Article

Proximal cooperative aerial manipulation with vertically stacked drones

https://doi.org/10.1038/s41586-025-09575-x

Received: 1 December 2024

Accepted: 27 August 2025

Published online: 24 September 2025

Check for updates

Huazi Cao[1,2], Jiahao Shen[1], Yin Zhang[1], Zheng Fu[1], Cunjia Liu[3], Sihao Sun[4] & Shiyu Zhao[1,5]✉

Enabling vertical-stack proximal cooperation between multirotor flying robots can facilitate the execution of complex aerial manipulation tasks. However, vertical-stack proximal flight is commonly regarded as a dangerous condition that should be avoided because of persistent and intense downwash interference generated between flying robots[1,2]. Here we propose a cooperative aerial manipulation system, called FlyingToolbox, that can work stably with sub-centimetre-level docking accuracy under vertical-stack flight conditions. The system consists of a toolbox micro-aerial vehicle (MAV) and a manipulator MAV. The robotic arm of the manipulator MAV can autonomously dock with a tool carried by the toolbox MAV, in which the docking accuracy reaches 0.80 ± 0.33 cm in the presence of downwash airflow of up to 13.18 m s^{-1}. By enabling midair tool exchange in proximity, FlyingToolbox resolves the paradox between flight proximity and manipulation accuracy, suggesting a new model for heterogeneous and interactive flying robot cooperation in diverse applications[3–5].

Aerial manipulators, which combine the mobility of multirotor micro-aerial vehicles (MAVs) with the dexterity of robotic arms, offer rapid and precise manipulation at hard-to-reach locations[3,6–8]. They have demonstrated promising potential in various tasks, including aerial pick-and-place[9–11], contact-based inspection and repair[12–15], aerial additive manufacturing[16,17], peg-in-hole assembly[18,19] and object manipulation[20–22]. However, complex aerial manipulation tasks often require versatile tools or materials, making single aerial robots incapable because of their restricted payload. A promising solution to overcome this limitation is through cooperation between multiple heterogeneous aerial robots, in which different robots carry payloads of versatile abilities. For instance, one aerial robot carries tools necessary for a task and hands them over to another aerial robot one at a time to perform a series of dedicated operations—much like a theatre nurse passing tools to the surgeon during a complex surgical procedure. This cooperation allows each aerial robot to specialize in its role, ensuring seamless task execution while reducing downtime and resource use. The cooperation can be extended to other applications, such as aerial battery replacement and material replenishment, providing solutions for complex aerial manipulation tasks.

Despite the appealing concept, these cooperative aerial systems face a paradox between flight proximity and manipulation accuracy. On the one hand, vertical-stack proximal flight is essential for two aerial robots exchanging payloads in the air. On the other hand, such a flight condition generates persistent and intense downwash interference that can greatly downgrade control accuracy, increasing the likelihood of task failure. Existing aerial swarming systems typically treat these flight conditions as dangerous and deliberately avoid them[4,23–26]. Although vertical-stack flight appears in aerial landing[27–30], aerial hanging[31] and swarming flight[1,32–34], these flight conditions are short-lived, and these tasks do not require precise aerial manipulation. Furthermore, tasks such as cooperative transportation[5,35–39], material delivery[40–43] and aerial assembly[44–46] involve physical interactions between MAVs, but they do not involve vertical-stack flight with aerial manipulation.

Here we propose a cooperative aerial manipulation system called FlyingToolbox that consists of a toolbox MAV and a manipulator MAV (Fig. 1a). The toolbox MAV carries a toolbox that contains a set of end-effector tools for dedicated tasks such as grasping, cutting and inspection. The manipulator MAV flies above the toolbox MAV and can autonomously dock with any tool using its robotic arm. After completing a manipulation task, the manipulator MAV can return the tool to the toolbox MAV or switch to another to perform a different task. The autonomy of the system is realized by having all necessary perception, estimation and control functions onboard. To ensure successful docking, the horizontal displacement between the bottom of the robotic arm and the top of a tool must be less than 1.5 cm. However, as the vertical distance between the two MAVs may reach as close as 0.6 m during aerial docking, intense downwash airflow up to 13.18 m s^{-1} can be generated and hence substantially downgrade the control performance.

To address these challenges, we designed three core modules: (1) compliant electromagnetic docking mechanisms; (2) airflow disturbance estimation and compensation methods; and (3) high-accuracy docking and manipulation control techniques. These modules enable high-accuracy aerial docking under conditions of persistent downwash interference. Specifically, 20 consecutive docking experiments demonstrate the high robustness of the system and quantify a docking accuracy of 0.80 ± 0.33 cm (mean ± s.d.). This sub-centimetre accuracy represents a substantial improvement over the accuracy (6–8 cm) of existing aerial docking systems without robotic arm compensation[2,31]. It is also comparable to the state-of-the-art accuracy (0.7–1.29 cm) of

[1]WINDY Lab, Department of Artificial Intelligence, Westlake University, Hangzhou, China. [2]Westlake Institute for Optoelectronics, Fuyang, Hangzhou, China. [3]Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough, UK. [4]Department of Cognitive Robotics, Delft University of Technology, Delft, The Netherlands. [5]Research Center for Industries of the Future, Westlake University, Hangzhou, China. ✉e-mail: zhaoshiyu@westlake.edu.cn

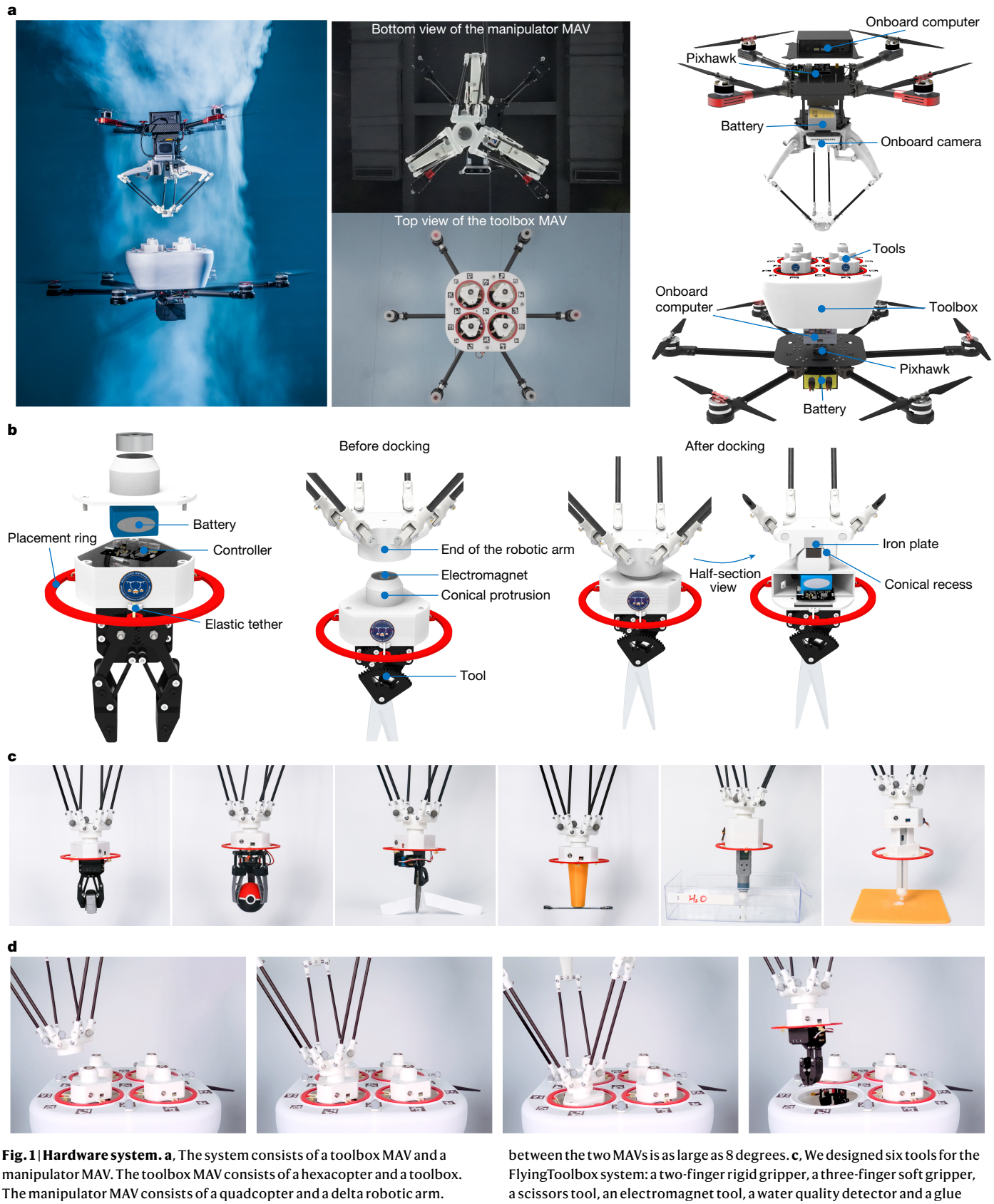


Fig. 1 | Hardware system. a, The system consists of a toolbox MAV and a manipulator MAV. The toolbox MAV consists of a hexacopter and a toolbox. The manipulator MAV consists of a quadcopter and a delta robotic arm. **b**, The docking mechanism. The top of a tool and the bottom of the robotic arm can be attracted by an iron plate and a wirelessly controlled electromagnet. A soft mechanism, in which elastic tethers connect the tool to its placement ring, ensures a high success rate of docking even when the relative pitch angle between the two MAVs is as large as 8 degrees. **c**, We designed six tools for the FlyingToolbox system: a two-finger rigid gripper, a three-finger soft gripper, a scissors tool, an electromagnet tool, a water quality detector and a glue dispenser. **d**, An illustration of the docking procedure, in which the soft docking mechanism ensures successful docking although the bottom of the robotic arm is non-parallel to the top surface of the tool.

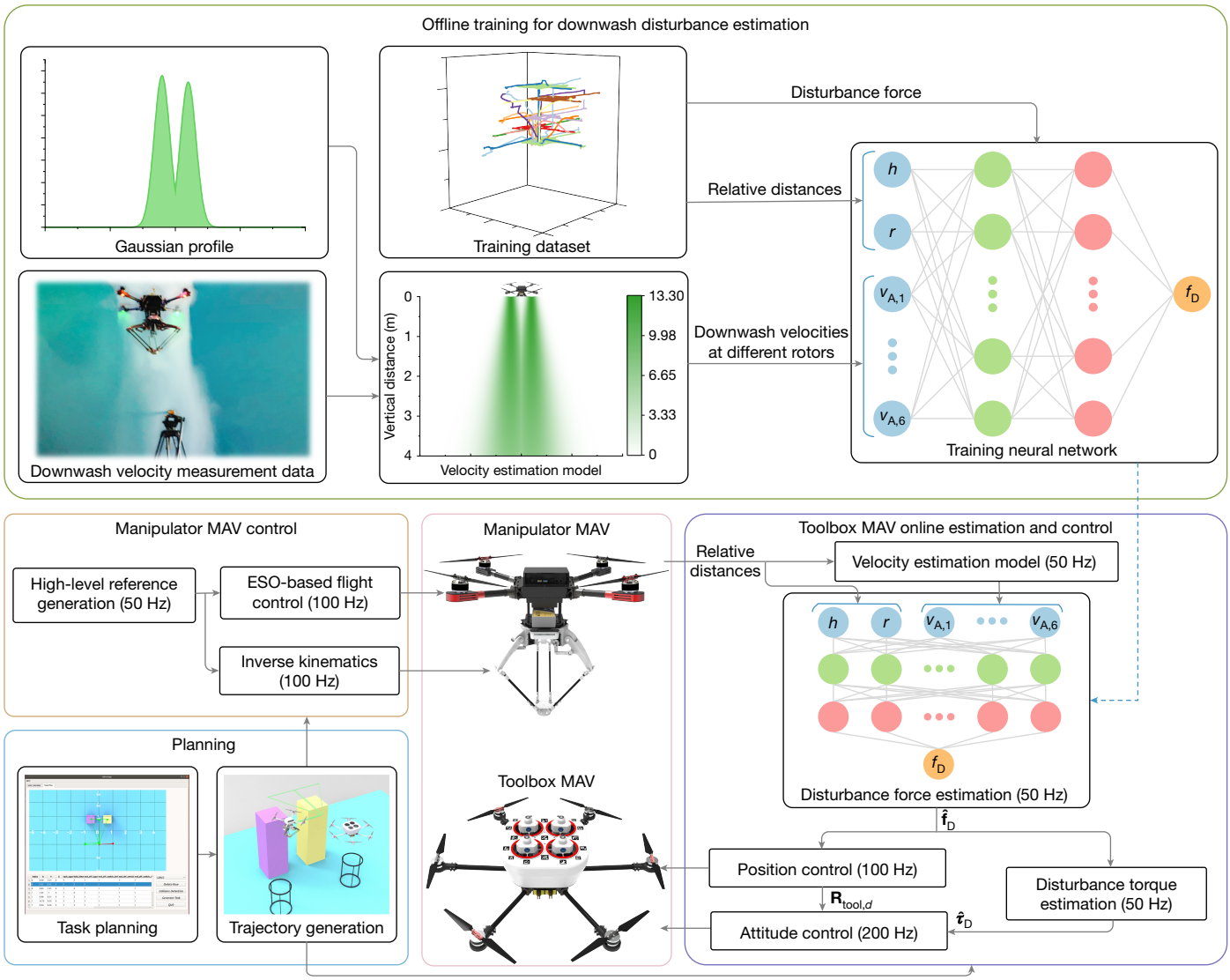


Fig. 2 | Estimation and control structure. The downwash velocity is estimated using a Gaussian profile. The downwash force is estimated by an MLP, which is trained offline and executed in real-time online. The design of the inputs of the neural network is inspired by a physics-based analytical model. The estimated downwash force and velocity are further used to estimate the disturbance torque. The control of the toolbox MAV incorporates the estimated force and torque to counter the downwash airflow. The control of the manipulator MAV consists of high-level reference generation and low-level flight control. Details are given in the Methods.

single-MAV manipulation with robotic arm compensation[18], despite our system addressing more challenging aerodynamic coupling effects arising from two MAVs flying in close proximity. Furthermore, it is demonstrated that multiple manipulator MAVs can reliably perform complex, multi-stage aerial manipulation through sequential tool switching using the toolbox MAV. By resolving the paradox between flight proximity and manipulation accuracy, FlyingToolbox suggests an unprecedented model that may open up new possibilities for complex aerial tasks in diverse applications.

Hardware system

The FlyingToolbox system consists of three hardware components: a toolbox MAV, a manipulator MAV and a docking mechanism (Fig. 1 and Extended Data Fig. 1).

The toolbox MAV consists of a hexacopter with a 1.02 m motor-to-motor distance and a specially designed toolbox (Fig. 1a and Extended Data Fig. 1a). The toolbox has four circular tool slots, each with a diameter of 0.12 m. We developed six end-effector tools with distinct functions, including rigid and soft grasping, cutting, magnetic attraction, liquid inspection and adhesive spraying (Fig. 1b,c). The manipulator MAV consists of a quadcopter with a 0.65 m motor-to-motor distance and a delta robotic arm (Fig. 1a and Extended Data Fig. 1a). The delta arm, controlled by three servos, enables three degrees of translational freedom at its bottom. The manipulator MAV is additionally equipped with a downward-looking vision system that can provide relative positioning during aerial docking (Extended Data Fig. 2).

The design of the docking mechanism directly affects the success rate of aerial docking. At the bottom of the robotic arm of the manipulator MAV is a conical recess containing an iron plate (Fig. 1b and Extended Data Fig. 1b). On top of each tool is a conical protrusion with an electromagnet (Extended Data Fig. 1c). When the electromagnet and iron plate attract each other, they generate an attraction force of 80 N, thereby linking the tool to the robotic arm. Each tool is additionally equipped with a wireless controller and a battery, allowing the electromagnet to be wirelessly activated or deactivated.

The relative attitude between the bottom of the robotic arm and the top of the tool also affects docking success. A large angle between the

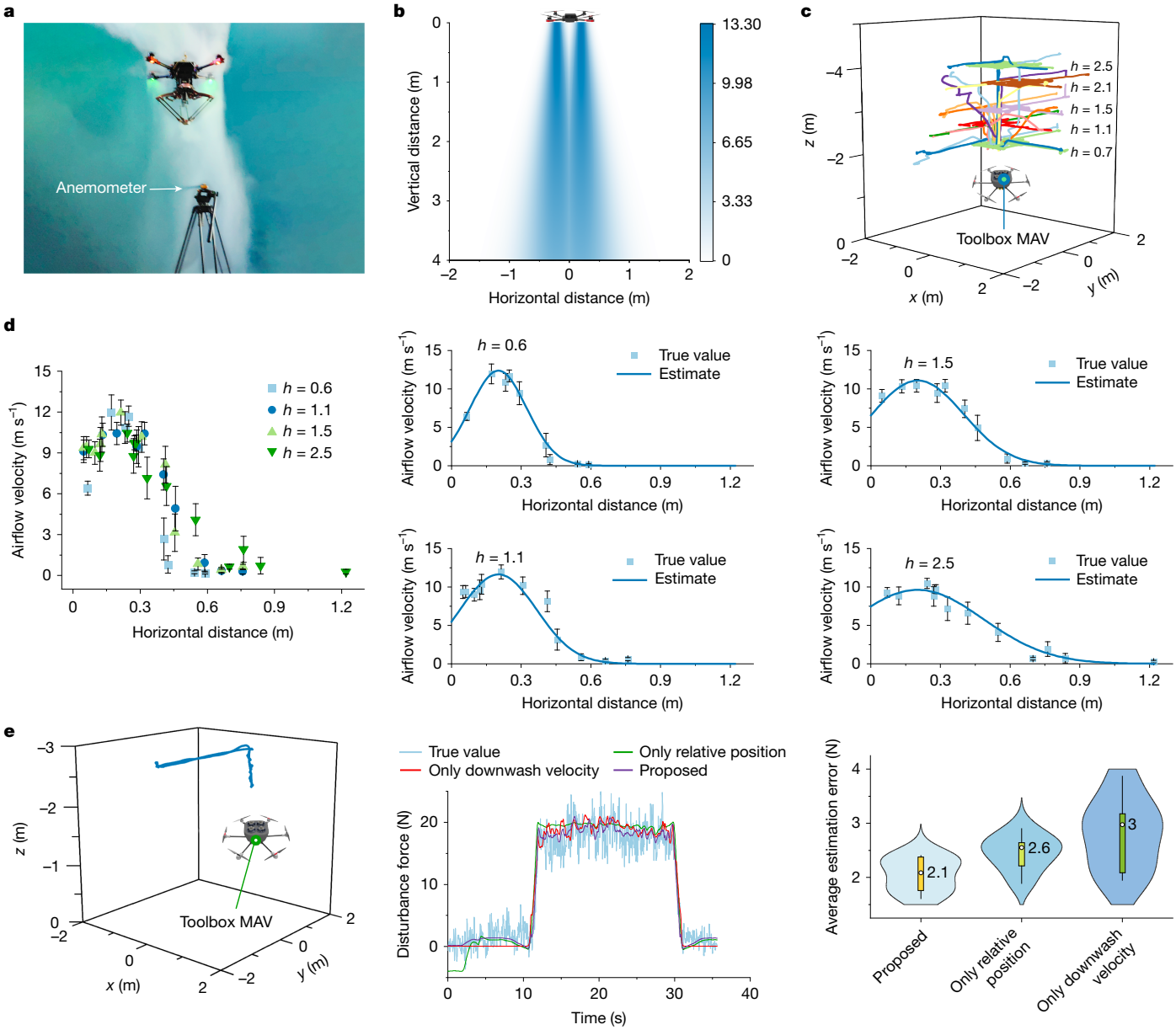


Fig. 3 | Estimation datasets and results. a, An anemometer was used to collect the airflow velocity measurement dataset at different positions beneath the manipulator MAV. **b**, A visualization of the airflow velocity field model that was fitted using the velocity measurement dataset shown in **d**. **c**, The measurement datasets were collected when the toolbox MAV hovers and the manipulator MAV moved along different trajectories within a cylindrical region of 1.2 m radius, at different heights of 0.7 m, 1.1 m, 1.5 m, 2.1 m and 2.5 m above the toolbox MAV. **d**, The collected velocity measurements at different heights and the fitted results. The error bars represent the standard deviation. **e**, Verification results of the trained force estimation model. The proposed model achieves an average estimation error of 2.1 N with a standard deviation of 0.31 N. The ablation study showed that inputting both relative position and downwash velocity to the neural network improves estimation accuracy.

two contact surfaces of the iron plate and the electromagnet prevents proper magnetic attraction between them. Experiments showed that docking can be stably achieved when the angle is less than 2°. However, the relative pitch angle between the two MAVs typically varies up to 5°. To address this challenge, we designed a soft mechanism in which elastic tethers are used to connect the tool to its placement ring (Fig. 1b). This mechanism allows the electromagnet to passively adjust its attitude when pressed by the bottom of the robotic arm, exhibiting compliance and ensuring proper magnetic attachment. It can also avoid rigid impacts between the bottom of the robotic arm and the top of the tool in the presence of control errors, thereby enhancing operational safety. Experimental results indicated that the soft mechanism ensures successful docking even when the relative angle is as large as 8°, effectively accommodating the attitude variations between the MAVs (Fig. 1d).

The relative position control between the bottom of the robotic arm and the top of the tool is essential for successful docking. The maximum allowable horizontal position error is 1.5 cm, determined by the radius of the conical recess of the robotic arm and the conical protrusion of the tool. Vertically, to ensure successful magnetic attraction, the robotic arm applies slight downward pressure on the tool when docking, but this downward distance should be precisely controlled within a margin of 1.5 cm to avoid a clash. To meet the high-accuracy requirement, we developed a series of estimation and control modules as detailed below.

Estimation and control

During aerial docking, the toolbox MAV hovers at a designated position. The manipulator MAV moves to a position about 1 m above the

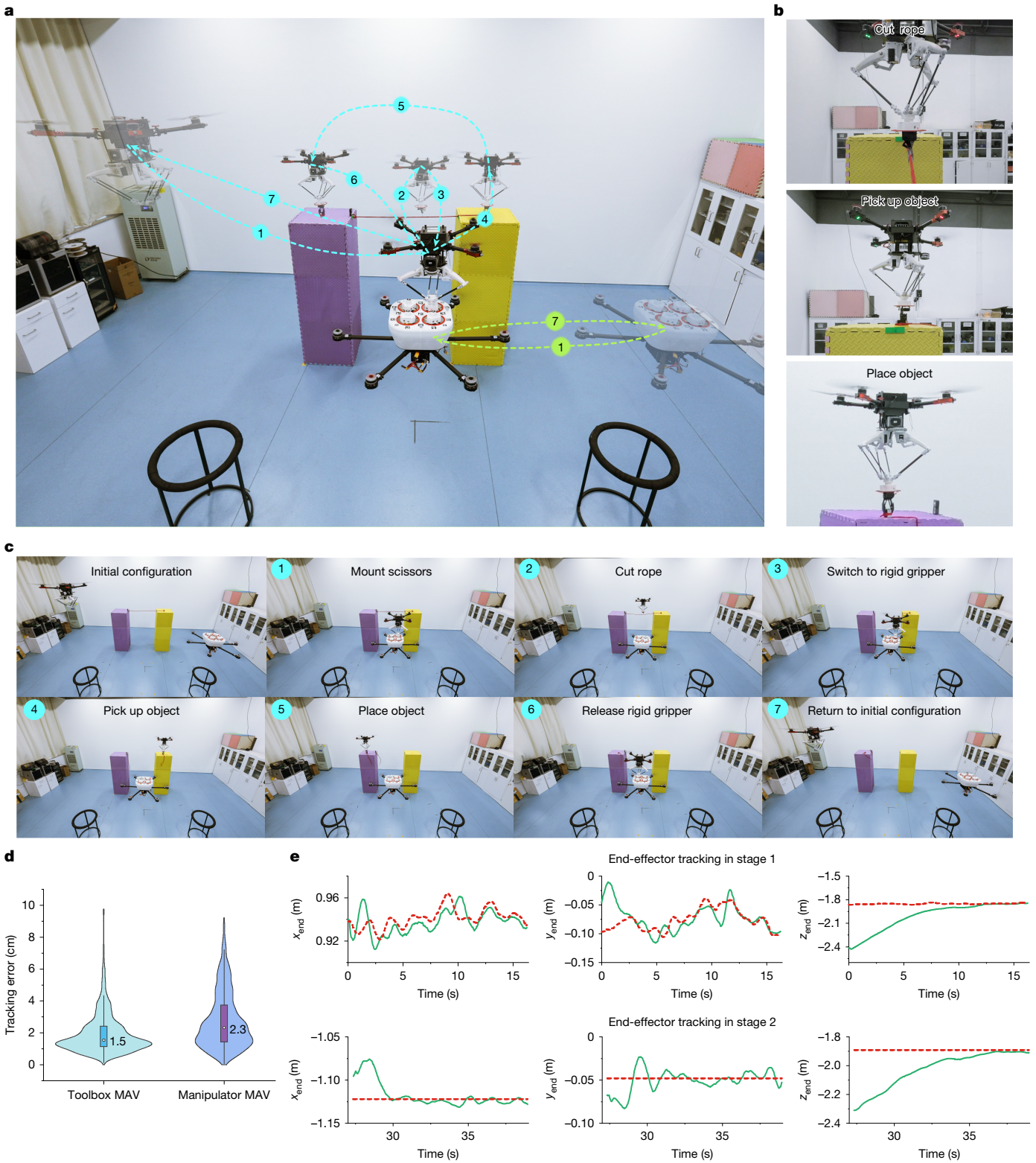


Fig. 4 | A multi-task demonstration. A manipulator MAV conducts multiple manipulation tasks with the support of a toolbox MAV fully autonomously. **a**, The complete procedure. Starting from an initial location, the manipulator MAV firstly mounted a scissors tool; secondly cut a rope; thirdly released the scissors tool and switched to a rigid gripper; fourthly picked up an object from the right pillar; fifthly placed the object on the left pillar; sixthly released the gripper tool; and seventhly flew back to a desired location. **b**, Snapshots of object manipulation during the task. **c**, Snapshots of the FlyingToolbox system during the task. **d**, The median values of the position tracking errors of the toolbox and manipulator MAV bases across the entire experiment are 1.5 cm and 2.3 cm, respectively. **e**, The curves of the end-effector position control in two representative stages. Dotted and solid curves represent reference and actual values, respectively.

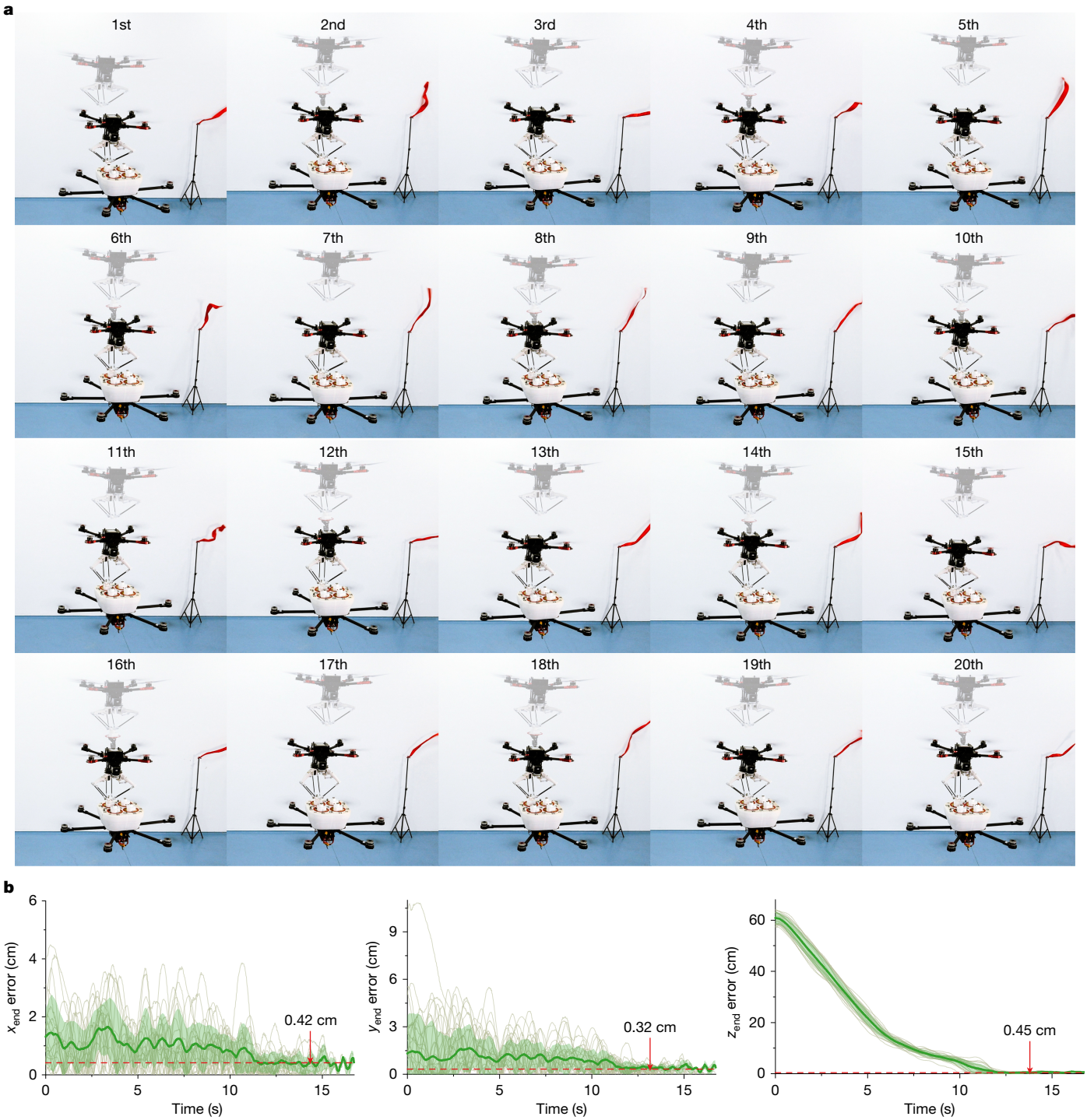


Fig. 5 | Repeatability and accuracy verification by an experiment with 20 consecutive docking processes. a, Snapshots of the 20 docking processes. The complete experimental procedure is shown in Supplementary Video 3. **b**, Tracking errors in three directions of the 20 docking processes (grey curves). The initiation time of each docking process is defined by the issuance of the docking command, whereas the completion time is marked by the fly-up command. The duration of the 20 docking processes ranges from 12.56 s to 16.72 s, with an average duration of 14.57 s. The plots align all error curves based on their initiation times. By considering the absolute errors during the final two seconds of each docking process, we obtained the average docking errors in the *x*-, *y*- and *z*-directions across the 20 trials as 0.42 cm, 0.32 cm and 0.45 cm, respectively. The overall average docking error in three directions was 0.80 cm and the standard deviation was 0.33 cm. The maximum instantaneous error caused by strong airflow disturbances during the final 2 s across all docking trials was 1.98 cm.

toolbox MAV and then descends at a speed of approximately 0.1 m s^{-1} until its robotic arm is able to reach a tool. Throughout this process, an onboard vision system on the manipulator MAV provides 50 Hz relative position measurement by recognizing the QR code array on the toolbox MAV (Extended Data Fig. 2). The average positioning error of the onboard vision system is 0.48 cm (Extended Data Fig. 2d), which is sufficient to support high-accuracy docking control.

During the docking process, the propellers of the manipulator MAV generate persistent and intense downwash disturbances on the toolbox MAV underneath. For instance, when the vertical distance between the

two MAVs is 0.6 m, the downwash speed can reach 13.18 m s^{-1}, exerting an unsettled disturbance up to 24.9 N, which is 40.2% of the weight of the toolbox MAV (6.32 kg). To counter the downwash airflow, it is important to estimate and then compensate for the induced disturbance force and torque. Here, we model the downwash velocity field as a Gaussian profile, which enables the prediction of airflow velocities at arbitrary positions below the manipulator MAV (Fig. 2 and Methods). The airflow velocity prediction is further used to estimate the disturbance force and torque. Specifically, the disturbance force is estimated through a multilayer perceptron (MLP), whose inputs include the relative horizontal and vertical distances between the manipulator and toolbox MAVs, along with the Gaussian-predicted airflow velocities at all six rotor locations of the toolbox MAV (Fig. 2). The design of the inputs of the neural network is inspired by a physics-based analytical model, as detailed in the Methods. The proposed method incorporates airflow velocities as additional inputs to the neural network estimator in contrast to existing methods[1,27]. As the airflow velocity model integrates aerodynamic principles with real-world velocity sensing data, it can provide rich information to enhance estimation accuracy. Furthermore, the estimated disturbance force and the downwash velocity are used to estimate the disturbance torque (Fig. 2 and Methods).

The effectiveness of the proposed estimation method was verified on seven docking trajectory datasets (Fig. 3e). The median of the average disturbance estimation errors is 2.1 N, corresponding to a 10.5% relative error against a typical 20 N disturbance force. An ablation study showed that incorporating both relative distance and downwash velocity information can improve the estimation accuracy by 19.2% compared with using only relative distance information, and by 30.0% compared with using only downwash velocity information (Fig. 3e).

The control of the manipulator MAV consists of two modules: high-level reference generation and low-level flight control. The goal of the reference generation is to guide the MAV base to a desired position above the toolbox MAV, and in the meantime, the end-effector reaches its desired position (Methods). The control of the toolbox MAV incorporates the estimated disturbance forces and torques (Methods) to counter downwash airflow so that the toolbox MAV can maintain a stationary position during docking. Experiments showed that the proposed controller works effectively (Extended Data Fig. 3a) and outperforms a PX4 open-source flight controller and the proposed controller without disturbance compensation, in terms of position error and settling time (Extended Data Fig. 3b). For instance, the median of the average height errors across trials is 0.7 cm for the proposed controller, compared with 12.3 cm and 3.2 cm for the other two controllers (Extended Data Fig. 3b). Ablation studies also showed that the compensation of disturbance torque markedly improves the horizontal position accuracy of the toolbox MAV by reducing the maximum error in the y-direction from 11.9 cm to 4.8 cm (Extended Data Fig. 3c).

Demonstrations

Figure 4 exhibits the complete procedure of a cooperative manipulation task where one manipulator MAV conducts multi-stage manipulation tasks with the support of a toolbox MAV fully autonomously. In this task, the manipulator MAV first mounted a scissors tool and cut a string using the scissors. Subsequently, it released the scissors tool and switched to a gripper tool, which was then used to pick up and place an object from one location to another. The complete procedure is shown in Supplementary Video 1. The aerial docking between the manipulator and toolbox MAVs was based on onboard vision sensing, whereas object manipulation was based on an indoor positioning system.

A more complex experiment in which two manipulator MAVs (denoted as M1 and M2) collaborate with one toolbox MAV to perform multi-stage manipulation tasks fully autonomously was conducted. As shown in Extended Data Fig. 4 and Supplementary Video 2, M1 first docked with an electromagnet tool, picked up the iron plate from the right pillar and placed it on the middle pillar. M2 first docked with a soft gripper tool, picked up the ball from the left pillar and eventually stacked the ball on the iron plate that had been placed on the middle pillar. As the experimental space was confined, the three flying MAVs generated highly dynamic airflow in the field, as evidenced by the movement of the window curtains and red ribbons in the videos. These experiments demonstrate the robustness and versatility of FlyingToolbox for complex multi-stage aerial tasks.

To further demonstrate the high accuracy and repeatability of FlyingToolbox, we conducted an experiment of 20 consecutive docking operations, which comprised 10 pick-ups and 10 returns (Fig. 5 and Supplementary Video 3). The primary challenge of this experiment lies in the repeated retrieval and return of the same tool, as any minor inaccuracy during the return phase in one docking trial could compromise the success of the subsequent pick-up. The average error of the 20 docking trials is 0.80 cm with the standard deviation as 0.33 cm. This sub-centimetre accuracy represents a substantial improvement over the 6–8 cm accuracy for docking systems without robotic arm compensation[2,31], whereas matching the state-of-the-art accuracy (0.7 ± 0.36 cm) for setpoint control tasks achieved with robotic arm compensation (table VI of ref. 18). While the toolbox MAV maintained a hovering state in the demonstrations presented earlier, we further challenged the system by considering a more complex aerial-docking-in-motion task, in which the toolbox MAV was in motion and the manipulator MAV successfully performed aerial docking to retrieve a tool, as shown in Extended Data Fig. 5 and Supplementary Video 4. This demonstration showcases the potential of the proposed system in more demanding scenarios, expanding the boundaries of its abilities.

Discussion

The FlyingToolbox system achieves physically interactive and high-accuracy aerial docking in the presence of persistent airflow interference. The accuracy, robustness and versatility of the system make it a potential solution for a wide range of real-world tasks. We believe that it may inspire the community to develop complex cooperative aerial manipulation systems, for example, for battery replacement and material replenishment, substantially expanding the abilities of aerial manipulators. FlyingToolbox has the following limitations that may be overcome in the future. First, although vision sensing was used for relative positioning during the docking process, an external positioning system was also used to navigate the MAVs for object manipulation tasks. Environmental mapping and object detection[47,48] may be integrated to achieve fully autonomous navigation for aerial manipulation in the future. Second, the robotic arm adopted by the manipulator MAV is a delta arm, which only has three degrees of translational freedom. It can be replaced by a multi-joint serial robotic arm[49] that has more degrees of freedom to adapt to more complex manipulation tasks.

Online content

Any methods, additional references, Nature Portfolio reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41586-025-09575-x.

1. Shi, G., Hönig, W., Shi, X., Yue, Y. & Chung, S. J. Neural-Swarm2: planning and control of heterogeneous multirotor swarms using learned interactions. *IEEE Trans. Robot.* **38**, 1063–1079 (2021).
2. Zhang, R., Zhang, D. & Mueller, M. W. Proxfly: robust control for close proximity quadcopter flight via residual reinforcement learning. *In Proc. 2025 IEEE International Conference on Robotics and Automation (ICRA)*, 13683–13689 (IEEE, 2025).
3. Ollero, A., Tognon, M., Suarez, A., Lee, D. & Franchi, A. Past, present, and future of aerial robotic manipulators. *IEEE Trans. Robot.* **38**, 626–645 (2021).
4. Kim, S., Seo, H., Shin, J. & Kim, H. J. Cooperative aerial manipulation using multirotors with multi-DOF robotic arms. *IEEE/ASME Trans. Mechatron.* **23**, 702–713 (2018).

5. Rossi, E. et al. Cooperative aerial load transportation via sampled communication. *IEEE Control Syst. Lett.* **4**, 277–282 (2019).
6. Spieler, P. et al. Parsec: an aerial platform for autonomous deployment of self-anchoring payloads on natural vertical surfaces. In *Proc. 2023 IEEE International Conference on Robotics and Automation (ICRA)*, 5331–5337 (IEEE, 2023).
7. Lee, J. et al. Introspective perception for long-term aerial telemanipulation with virtual reality. *IEEE Trans Field Robot.* **1**, 360–393 (2024).
8. Yiğit, A., Cuvillon, L., Perozo, M. A., Durand, S. & Gangloff, J. Dynamic control of a macro–mini aerial manipulator with elastic suspension. *IEEE Trans. Robot.* **39**, 4820–4836 (2023).
9. Guo, X. et al. Powerful UAV manipulation via bioinspired self-adaptive soft self-contained gripper. *Sci. Adv.* **10**, eadn6642 (2024).
10. Luo, W., Chen, J., Ebel, H. & Eberhard, P. Time-optimal handover trajectory planning for aerial manipulators based on discrete mechanics and complementarity constraints. *IEEE Trans. Robot.* **39**, 4332–4349 (2023).
11. Garimella, G. & Kobilarov, M. Towards model-predictive control for aerial pick-and-place. In *Proc. 2015 IEEE International Conference on Robotics and Automation (ICRA)*, 4692–4697 (2015).
12. Ryll, M. et al. 6D interaction control with aerial robots: the flying end-effector paradigm. *Int. J. Robot. Res.* **38**, 1045–1062 (2019).
13. Bodie, K. et al. Active interaction force control for contact-based inspection with a fully actuated aerial vehicle. *IEEE Trans. Robot.* **37**, 709–722 (2020).
14. Alexis, K., Darivianakis, G., Burri, M. & Siegwart, R. Aerial robotic contact-based inspection: planning and control. *Auton. Robot.* **40**, 631–655 (2016).
15. Chermprayong, P., Zhang, K., Xiao, F. & Kovac, M. An integrated delta manipulator for aerial repair: a new aerial robotic system. *IEEE Robot. Autom. Mag.* **26**, 54–66 (2019).
16. Zhang, K. et al. Aerial additive manufacturing with multiple autonomous robots. *Nature* **609**, 709–717 (2022).
17. Hunt, G., Mitzalis, F., Alhinai, T., Hooper, P. A. & Kovac, M. 3D printing with flying robots. In *Proc. 2014 IEEE International Conference on Robotics and Automation (ICRA)*, 4493–4499 (IEEE, 2014).
18. Wang, M. et al. Millimeter-level pick and peg-in-hole task achieved by aerial manipulator. *IEEE Trans. Robot.* **40**, 1242–1260 (2024).
19. Orsag, M., Korpela, C., Bogdan, S. & Oh, P. Dexterous aerial robots–mobile manipulation using unmanned aerial systems. *IEEE Trans. Robot.* **33**, 1453–1466 (2017).
20. Lee, D., Seo, H., Kim, D. & Kim, H. J. Aerial manipulation using model predictive control for opening a hinged door. In *Proc. 2020 IEEE International Conference on Robotics and Automation (ICRA)*, 1237–1242 (IEEE, 2020).
21. Estrada, M. A., Mintchev, S., Christensen, D. L., Cutkosky, M. R. & Floreano, D. Forceful manipulation with micro air vehicles. *Sci. Robot.* **3**, eaau6903 (2018).
22. Nguyen, H. & Alexis, K. Forceful aerial manipulation based on an aerial robotic chain: hybrid modeling and control. *IEEE Robot. Autom. Lett.* **6**, 3711–3719 (2021).
23. Vásárhelyi, G. et al. Optimized flocking of autonomous drones in confined environments. *Sci. Robot.* **3**, eaat3536 (2018).
24. Zhou, X. et al. Swarm of micro flying robots in the wild. *Sci. Robot.* **7**, eabm5954 (2022).
25. Uzun, S., Üre, N. K. & Açıkmeıe, B. Decentralized state-dependent Markov chain synthesis with an application to swarm guidance. *IEEE Trans. Autom. Control* **69**, 5759–5774 (2024).
26. Bandyopadhyay, S., Chung, S. J. & Hadaegh, F. Y. Probabilistic and distributed control of a large-scale swarm of autonomous agents. *IEEE Trans. Robot.* **33**, 1103–1123 (2017).
27. Shi, G., et al. Neural lander: stable drone landing control using learned dynamics. In *Proc. 2019 IEEE International Conference on Robotics and Automation (ICRA)*, 9784–9790 (IEEE, 2019).
28. Polvara, R. et al. Toward end-to-end control for UAV autonomous landing via deep reinforcement learning. In *Proc. 2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, 115–123 (IEEE, 2018).
29. Saripalli, S., Montgomery, J. F. & Sukhatme, G. S. Visually guided landing of an unmanned aerial vehicle. *IEEE Trans. Robot. Autom.* **19**, 371–380 (2003).
30. Gonçalves, V. M., McLaughlin, R. & Pereira, G. A. Precise landing of autonomous aerial vehicles using vector fields. *IEEE Robot. Autom. Lett.* **5**, 4337–4344 (2020).
31. Shankar, A., Woo, H. & Prorok, A. Docking multirotors in close proximity using learnt downwash models. In *Proc. International Symposium on Experimental Robotics*, 427–437 (Springer, 2023).
32. Gielis, J., Shankar, A., Kortvelesy, R. & Prorok, A. Modeling aggregate downwash forces for dense multirotor flight. In *Proc. International Symposium on Experimental Robotics*, 393–404 (Springer, 2023).
33. Smith, H., Shankar, A., Gielis, J., Blumenkamp, J. & Prorok, A. SO(2)-equivariant downwash models for close proximity flight. *IEEE Robot. Autom. Lett.* **9**, 1174–1181 (2023).
34. Wu, Z. et al. £1quad: £1adaptive augmentation of geometric control for agile quadrotors with performance guarantees. *IEEE Trans. Control Syst. Technol.* **33**, 597–612 (2025).
35. Ebel, H., Luo, W., Yu, F., Tang, Q. & Eberhard, P. Design and experimental validation of a distributed cooperative transportation scheme. *IEEE Trans. Autom. Sci. Eng.* **18**, 1157–1169 (2021).
36. Loianno, G. & Kumar, V. Cooperative transportation using small quadrotors using monocular vision and inertial sensing. *IEEE Robot. Autom. Lett.* **3**, 680–687 (2018).
37. Lee, H., Kim, H. & Kim, H. J. Planning and control for collision-free cooperative aerial transportation. *IEEE Trans. Autom. Sci. Eng.* **15**, 189–201 (2018).
38. Sanalitro, D., Savino, H. J., Tognon, M., Cortés, J. & Franchi, A. Full-pose manipulation control of a cable-suspended load with multiple UAVs under uncertainties. *IEEE Robot. Autom. Lett.* **5**, 2185–2191 (2020).
39. Li, G., Ge, R. & Loianno, G. Cooperative transportation of cable suspended payloads with MAVs using monocular vision and inertial sensing. *IEEE Robot. Autom. Lett.* **6**, 5316–5323 (2021).
40. Palunko, I., Cruz, P. & Fierro, R. Agile load transportation: safe and efficient load manipulation with aerial robots. *IEEE Robot. Autom. Mag.* **19**, 69–79 (2012).
41. Gawel, A. et al. Aerial picking and delivery of magnetic objects with MAVs. In *Proc. 2017 IEEE International Conference on Robotics and Automation (ICRA)*, 5746–5752 (IEEE, 2017).
42. Saunders, J., Saeedi, S. & Li, W. Autonomous aerial robotics for package delivery: a technical review. *J. Field Robot.* **41**, 3–49 (2024).
43. Kornatowski, P. M., Feroskhan, M., Stewart, W. J. & Floreano, D. Downside up: rethinking parcel position for aerial delivery. *IEEE Robot. Autom. Lett.* **5**, 4297–4304 (2020).
44. Lindsey, Q., Mellinger, D. & Kumar, V. Construction with quadrotor teams. *Auton. Robot.* **33**, 323–336 (2012).
45. Augugliaro, F. et al. The flight assembled architecture installation: cooperative construction with flying machines. *IEEE Control Syst. Mag.* **34**, 46–64 (2014).
46. Jimenez-Cano, A. E., Martin, J., Heredia, G., Ollero, A. & Cano, R. Control of an aerial robot with multi-link arm for assembly tasks. In *Proc. 2013 IEEE International Conference on Robotics and Automation (ICRA)*, 4916–4921 (IEEE, 2013).
47. Zhang, X., Zhang, Y., Liu, P. & Zhao, S. Robust localization of occluded targets in aerial manipulation via range-only mapping. *IEEE Robotics and Automation Letters* **7**, 2921–2928 (2022).
48. Scaramuzza, D. et al. Vision-controlled micro flying robots: from system design to autonomous navigation and mapping in GPS-denied environments. *IEEE Robot. Autom. Mag.* **21**, 26–40 (2014).
49. Lee, J. et al. Visual-inertial telepresence for aerial manipulation. In *Proc. 2020 IEEE International Conference on Robotics and Automation (ICRA)*, 1222–1229 (IEEE, 2020).

Methods

Downwash velocity estimation

The rotors of the manipulator MAV, due to their rapid spinning, create a downwash field beneath. The velocity field of the downwash airflow can be represented by a Gaussian profile[50,51]:

$$v_{\mathrm{A,max}}(h) = k_{\mathrm{a},1} - k_{\mathrm{a},2}h, \tag{1}$$

$$v_{\mathrm{A}}(r,h) = v_{\mathrm{A,max}}(h)\exp\left[-\frac{1}{2}\left(\frac{r-k_{\mathrm{a},3}}{k_{\mathrm{a},4}+k_{\mathrm{a},5}h}\right)^2\right], \tag{2}$$

where r, $h > 0$ are, respectively, the horizontal distance and vertical distance between the centre of the toolbox MAV and the centre of the manipulator MAV. During aerial docking, r and h are obtained through the onboard vision sensing system. Here, $v_{\mathrm{A,max}}(h) \in \mathbb{R}$ is the maximum airflow velocity at h, and $v_{\mathrm{A}}(r,h) \in \mathbb{R}$ is the airflow velocity at position (r,h). To determine the parameters of the Gaussian profile, $\{k_{\mathrm{a},i}\}_{i=1}^{5}$, we used an anemometer to measure the airflow velocity at 44 different positions underneath the manipulator MAV (Fig. 3a,d). The parameters were then determined by solving the following optimization problem:

$$\min_{\{k_{\mathrm{a},i}\}_{i=1}^{5}} \sum_{i=1}^{n_{\mathrm{A}}} (v_{\mathrm{A}}(r_i,h_i) - \tilde{v}_{\mathrm{A},i})^2, \tag{3}$$

where r_i, h_i, and $\tilde{v}_{\mathrm{A},i}$ represent the horizontal distance, vertical distance and measured airflow velocity in the ith observation, respectively, and n_{A} denotes the total number of measurements.

Downwash force and torque estimation

As the downwash flow field generated by a multirotor MAV adheres to the quasi-steady-state assumption during low-velocity operations[52,53], an MLP can effectively model the downwash-induced disturbance force as a nonlinear function. The neural network consists of two hidden layers, each containing 50 nodes with the activation function as ReLU. A mean squared error loss function and the Adam optimizer with a learning rate of 0.001 were used for training. The number of training epochs was 1,000. The output of the neural network is the one-dimensional estimated magnitude of the disturbance force. The design of the input of the neural network is inspired by the following analysis.

The thrust generated by the ith propeller of the toolbox MAV is $f_i = c_{\mathrm{p}}(v_{\mathrm{A},i})\omega_i^2 \in \mathbb{R}$ (ref. 54), where $v_{\mathrm{A},i}$ is the airflow velocity above the propeller, $c_{\mathrm{p}}(v_{\mathrm{A},i}) > 0$ is the thrust coefficient of the propeller, which is a function of $v_{\mathrm{A},i}$, and ω_i is the spinning speed of the propeller. In the absence of downwash disturbances, the thrust is $f_i = c_{\mathrm{p}}(0)\omega_i^2$ where $v_{\mathrm{A},i} = 0$. Hence, the disturbance force experienced by the rotors of the toolbox MAV can be expressed as $f_{\mathrm{D},r} = \sum_{i=1}^{n_r}[c_{\mathrm{p}}(v_{\mathrm{A},i}) - c_{\mathrm{p}}(0)]\omega_i^2 \in \mathbb{R}$, where n_{r} is the number of rotors on the toolbox MAV (here $n_{\mathrm{r}} = 6$). The first-order Taylor expansion of $c_{\mathrm{p}}(v_{\mathrm{A},i})$ yields $c_{\mathrm{p}}(v_{\mathrm{A},i}) \approx c_{\mathrm{p}}(0) + b_v v_{\mathrm{A},i}$, where $b_v = \partial c_{\mathrm{p}}/\partial v_{\mathrm{A},i} \in \mathbb{R}$. Then, $f_{\mathrm{D,r}}$ becomes

$$f_{\mathrm{D,r}} = b_v \sum_{i=1}^{n_{\mathrm{r}}} v_{\mathrm{A},i}\omega_i^2. \tag{4}$$

The expression of $f_{\mathrm{D,r}}$ in equation (4) requires the rotor speed ω_i of the toolbox MAV. However, MAVs usually cannot measure their rotor speeds with high frequency. Motivated by this, note that the toolbox MAV hovers nearly steadily during aerial docking, and hence the speeds of different rotors are approximately identical. Denote $\bar{\omega}$ as this identical value and $\bar{\omega} \approx \omega_i$ for any i. Then, equation (4) becomes

$$f_{\mathrm{D,r}} = b_v\bar{\omega}^2 \sum_{i=1}^{n_{\mathrm{r}}} v_{\mathrm{A},i}. \tag{5}$$

The expression of $f_{\mathrm{D,r}}$ in equation (5) requires b_v and $\bar{\omega}$, which are non-trivial to obtain because they both depend on the airflow. As a result, equation (5) cannot be directly used in practice, but it can guide us to design a neural network to estimate $f_{\mathrm{D,r}}$. First, equation (5) indicates that $f_{\mathrm{D,r}}$ is a function of the downwash velocities $v_{\mathrm{A},1}, \cdots, v_{\mathrm{A},n_{\mathrm{r}}}$. Second, the parameters b_v and $\bar{\omega}$ that depend on the airflow are also functions of h and r. Therefore, we identify the key factors determining the disturbance force at the rotors as $[v_{\mathrm{A},1}, \ldots, v_{\mathrm{A},n_{\mathrm{r}}}, h, r] \in \mathbb{R}^{n_{\mathrm{r}}+2}$ where $n_{\mathrm{r}} = 6$.

Notably, the total disturbance exerted by the manipulator MAV on the toolbox MAV includes not only disturbances at the rotors but also disturbances on the body. The irregular shape of the toolbox makes it challenging to mathematically describe the disturbance force on the body. Fortunately, the factors identified by the above theoretical analysis can not only describe the airflow velocity at the peripheral rotors but also effectively characterize, to a considerable extent, the airflow velocity field within the region enclosed by these rotors. Consequently, these factors can also effectively infer the disturbances acting on the body of the toolbox MAV. Therefore, we design the input of the neural network as $[v_{\mathrm{A},1}, \ldots, v_{\mathrm{A},n_{\mathrm{r}}}, h, r] \in \mathbb{R}^{n_{\mathrm{r}}+2}$ to estimate the total disturbance magnitude. The accurate estimation performance in Fig. 3e verifies the rationale of this design.

The neural network was trained offline on a dataset. To collect the dataset, the toolbox MAV was kept hovering while the manipulator MAV approached the toolbox MAV from different directions at different speeds. As the downwash of the manipulator MAV is primarily within a cylindrical region with a radius of 1.2 m (Fig. 3b), we collected trajectories within this region at different heights of 0.7 m, 1.1 m, 1.5 m, 2.1 m and 2.5 m above the toolbox MAV (Fig. 3c). The dataset consists of 15 trajectories, containing a total of 15,272 data points. The test set consists of seven actual aerial docking trajectories, comprising a total of 4,245 data points (Fig. 3e). Inspired by ref. 1, the ground truth of the disturbance force, denoted as $\mathbf{f}_{\mathrm{D}}^* \in \mathbb{R}^3$, can be obtained as $\mathbf{f}_{\mathrm{D}}^* = m_{\mathrm{tool}}\ddot{\mathbf{p}}_{\mathrm{tool}} - m_{\mathrm{tool}}\mathbf{g} + \mathbf{f}$, whose magnitude is $f_{\mathrm{D}}^* = \|m_{\mathrm{tool}}\ddot{\mathbf{p}}_{\mathrm{tool}} - m_{\mathrm{tool}}\mathbf{g} + \mathbf{f}\| \in \mathbb{R}$, where $m_{\mathrm{tool}} \in \mathbb{R}$ is the mass of the toolbox MAV, $\mathbf{g} = g[0,0,1]^{\mathrm{T}} \in \mathbb{R}^3$, $\ddot{\mathbf{p}} \in \mathbb{R}^3$ is the total acceleration of the toolbox MAV that can be measured by a HWT905 IMU, and the control force vector is given by $\mathbf{f} = \mathbf{R}_{\mathrm{tool}}[0,0,-f]^{\mathrm{T}} \in \mathbb{R}^3$, where $f = \sum_{i=1}^{n_{\mathrm{r}}} f_i$ is the total thrust from the rotors and $\mathbf{R}_{\mathrm{tool}} \in SO(3)$ is the rotation matrix of the toolbox MAV and can be obtained by the IMU. Here, f_i is calculated based on a speed-to-thrust model, calibrated by a Series 1585 thrust stand bundle, combined with the spinning speed of propeller i measured by an ALPHA DATA LINK v2 acquisition system. As the spinning speed can only be measured at 10 Hz, the speed measurements were merely used for offline neural network training. The deployed neural network can achieve 50 Hz estimation during flight.

The neural network outputs a scalar value $\hat{f}_{\mathrm{D}}$, which represents the estimated magnitude of the total disturbance force. The estimated disturbance force vector, expressed in the world frame with the z-axis pointing vertically downwards, is then obtained as $\hat{\mathbf{f}}_{\mathrm{D}} = [0,0,1]^{\mathrm{T}}\hat{f}_{\mathrm{D}} \in \mathbb{R}^3$. This formulation assumes that the disturbance force acts in the direction of the z-axis. This assumption is reasonable because the magnitude of the disturbance force along the z-axis is dominant[1]. By focusing on the dominant z-axis component, this simplification provides a tractable yet effective method for estimating the disturbance force.

The estimated disturbance force can be further applied to estimate the disturbance torque $\hat{\boldsymbol{\tau}}_{\mathrm{D}} \in \mathbb{R}^3$ caused by the downwash. As $\hat{f}_{\mathrm{D}} \approx \sum_{i=1}^{n_{\mathrm{r}}}\hat{f}_{\mathrm{p},i}$ and $\hat{f}_{\mathrm{p},i} = b_v\omega_i^2 v_{\mathrm{A}_i} \approx b_v\bar{\omega}^2 v_{\mathrm{A}_i}$, we have $\hat{f}_{\mathrm{D}} \approx b_v\bar{\omega}^2\sum_{i=1}^{n_{\mathrm{r}}} v_{\mathrm{A}_i}$ and hence $\hat{f}_{\mathrm{p},i}/\hat{f}_{\mathrm{D}} \approx v_{\mathrm{A},i}/\sum_{j=1}^{n_{\mathrm{r}}} v_{\mathrm{A},j}$. As a result, the magnitude of the disturbance force on the ith propeller is $\hat{f}_{\mathrm{p},i} = \hat{f}_{\mathrm{D}}\, v_{\mathrm{A},i}/\sum_{j=1}^{n_{\mathrm{r}}} v_{\mathrm{A},j}$. Then, the total disturbance torque in the body frame can be estimated as

$$\hat{\boldsymbol{\tau}}_{\mathrm{D}} = \sum_{i=1}^{n_{\mathrm{r}}} \mathbf{r}_{\mathrm{p},i}^{\mathrm{B}} \times \mathbf{n}^{\mathrm{B}}\hat{f}_{\mathrm{p},i}, \tag{6}$$

where $\mathbf{n}^{\mathrm{B}} = [0, 0, 1]^{\mathrm{T}} \in \mathbb{R}^3$ and $\mathbf{r}^{\mathrm{B}}_{\mathrm{p},i} \in \mathbb{R}^3$ is the centre position of the ith propeller in the body frame of the toolbox MAV.

The proposed learning-based method for estimating airflow disturbance offers several advantages over existing approaches. First, it fully leverages pretraining and real-time communication to exploit cooperative characteristics, achieving superior performance. Experiments comparing it with two representative reactive methods—the unscented Kalman filter (UKF)[55] and extended state observer (ESO)[56]—demonstrate a maximum error of 2.6 cm, representing a 63.3% reduction compared to the reactive methods (Extended Data Fig. 6). The reason that reactive methods exhibit higher latency and lower accuracy is that they can estimate disturbances only after state changes occur. Second, as a data-driven approach, the proposed method overcomes the limitations of fluid dynamics-based methods[57,58] that rely on simplified assumptions, and hence provides more accurate disturbance estimation for MAVs with complex body shapes, such as the toolbox MAV. Third, compared with the existing neural network methods[1], our method achieves higher accuracy by incorporating additional information on the airflow velocity field of the upper MAV, as demonstrated by the results in Fig. 3e. Furthermore, although the six velocity inputs constitute spatial information, given their sparsity, we selected an MLP rather than a convolutional neural network, which requires dense spatial inputs to function effectively, achieving high accuracy at modest training and computational costs.

Toolbox MAV control

The dynamical model of the toolbox MAV is

$$\dot{\mathbf{p}}_{\mathrm{tool}} = \mathbf{v}_{\mathrm{tool}}, \tag{7}$$

$$\dot{\mathbf{v}}_{\mathrm{tool}} = \mathbf{g} - \frac{\mathbf{f} - \mathbf{f}_{\mathrm{D}}}{m_{\mathrm{tool}}}, \tag{8}$$

$$\dot{\mathbf{R}}_{\mathrm{tool}} = \mathbf{R}_{\mathrm{tool}}[\boldsymbol{\omega}_{\mathrm{tool}}]_{\times}, \tag{9}$$

$$\dot{\boldsymbol{\omega}}_{\mathrm{tool}} = \mathbf{M}^{-1}_{\mathrm{tool}}[\boldsymbol{\tau} - \boldsymbol{\omega}_{\mathrm{tool}} \times (\mathbf{M}_{\mathrm{tool}}\boldsymbol{\omega}_{\mathrm{tool}}) + \boldsymbol{\tau}_{\mathrm{D}}], \tag{10}$$

where $\mathbf{p}_{\mathrm{tool}}, \mathbf{v}_{\mathrm{tool}} \in \mathbb{R}^3$ represent the position and velocity, respectively; $\mathbf{R}_{\mathrm{tool}} \in SO(3)$ and $\boldsymbol{\omega}_{\mathrm{tool}} \in \mathbb{R}^3$ denote the attitude and angular velocity, respectively; m_{tool} and $\mathbf{M}_{\mathrm{tool}} \in \mathbb{R}^{3\times3}$ denote the mass and the inertia matrix, respectively; $\mathbf{f}, \boldsymbol{\tau} \in \mathbb{R}^3$ represent the control force in the world frame and torque vectors in the body frame, respectively; $\mathbf{f}_{\mathrm{D}}, \boldsymbol{\tau}_{\mathrm{D}} \in \mathbb{R}^3$ represent the disturbance force and torque caused by the downwash, respectively; $[\cdot]_{\times}$ denotes the skew-symmetric operator.

The flight controller was designed based on our previous work[56]. In particular, it consists of a position controller and an attitude controller. The estimated disturbance force and torque are compensated in the controllers. The position control aims to generate the control force vector $\mathbf{f}$ for the toolbox MAV so that the desired position $\mathbf{p}_{\mathrm{tool,d}} \in \mathbb{R}^3$ can be tracked. It is designed as[56]

$$\mathbf{f} = m_{\mathrm{tool}}(\mathbf{g} + \mathbf{K}_v\mathbf{r}_v - \dot{\mathbf{v}}_{\mathrm{r}}) + \hat{\mathbf{f}}_{\mathrm{D}}, \tag{11}$$

where the reference velocity $\mathbf{v}_{\mathrm{r}} \in \mathbb{R}^3$ is expressed as $\mathbf{v}_{\mathrm{r}} = \dot{\mathbf{p}}_{\mathrm{tool,d}} - 2\boldsymbol{\Lambda}_{\mathrm{p}}\tilde{\mathbf{p}}_{\mathrm{tool}} - \boldsymbol{\Lambda}^2_{\mathrm{p}}\int_0^t \tilde{\mathbf{p}}_{\mathrm{tool}}(\tau)\mathrm{d}\tau$ and $\mathbf{r}_v \in \mathbb{R}^3$ is given as $\mathbf{r}_v = \dot{\tilde{\mathbf{p}}}_{\mathrm{tool}} + 2\boldsymbol{\Lambda}_{\mathrm{p}}\tilde{\mathbf{p}}_{\mathrm{tool}} + \boldsymbol{\Lambda}^2_{\mathrm{p}}\int_0^t \tilde{\mathbf{p}}_{\mathrm{tool}}(\tau)\mathrm{d}\tau$. Here, $\tilde{\mathbf{p}}_{\mathrm{tool}} = \mathbf{p}_{\mathrm{tool}} - \mathbf{p}_{\mathrm{tool,d}} \in \mathbb{R}^3$ represents the position error; $\mathbf{K}_v \in \mathbb{R}^{3\times3}$ and $\boldsymbol{\Lambda}_{\mathrm{p}} \in \mathbb{R}^{3\times3}$ are positive diagonal matrices.

The attitude control aims to generate the control torque vector $\boldsymbol{\tau}$ for the toolbox MAV to track the desired rotation matrix $\mathbf{R}_{\mathrm{tool,d}} = [\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3] \in SO(3)$. Let ψ_{d} denote the desired yaw angle of the MAV base. We define $\mathbf{a} = [\cos\psi_{\mathrm{d}}, \sin\psi_{\mathrm{d}}, 0]^{\mathrm{T}}$. Then, we have $\mathbf{b}_3 = \mathbf{f}/\|\mathbf{f}\|$, $\mathbf{b}_2 = \mathbf{b}_3 \times \mathbf{a}/\|\mathbf{b}_3 \times \mathbf{a}\|$, $\mathbf{b}_1 = \mathbf{b}_2 \times \mathbf{b}_3$. The attitude control is designed as[56]

$$\boldsymbol{\tau} = \mathbf{M}_{\mathrm{tool}}\dot{\boldsymbol{\omega}}_{\mathrm{tool},r} - \hat{\boldsymbol{\tau}}_{\mathrm{D}} + \boldsymbol{\omega}_{\mathrm{tool}} \times (\mathbf{M}_{\mathrm{tool}}\boldsymbol{\omega}_{\mathrm{tool}}) - \mathbf{K}_{\omega}\mathbf{r}_{\omega} - k_{\beta}\tilde{\boldsymbol{\beta}}_v, \tag{12}$$

where $\boldsymbol{\omega}_{\mathrm{tool},r} \in \mathbb{R}^3$ is designed as $\boldsymbol{\omega}_{\mathrm{tool},r} = \tilde{\mathbf{R}}^{\mathrm{T}}_{\mathrm{tool}}\boldsymbol{\omega}_{\mathrm{tool,d}} - 2\boldsymbol{\Lambda}_q\tilde{\boldsymbol{\beta}}_v$; $\mathbf{r}_{\omega} \in \mathbb{R}^3$ is designed as $\mathbf{r}_{\omega} = \boldsymbol{\omega}_{\mathrm{tool}} - \tilde{\mathbf{R}}^{\mathrm{T}}\boldsymbol{\omega}_{\mathrm{tool,d}} + 2\boldsymbol{\Lambda}_q\tilde{\boldsymbol{\beta}}_v$; $\tilde{\mathbf{R}}_{\mathrm{tool}} = \mathbf{R}^{\mathrm{T}}_{\mathrm{tool,d}}\mathbf{R}_{\mathrm{tool}}$ is the rotation error, $\boldsymbol{\omega}_{\mathrm{tool,d}} = [\mathbf{R}^{\mathrm{T}}_{\mathrm{tool}}\dot{\mathbf{R}}_{\mathrm{tool,d}}]^{\vee}$ is the desired angular velocity, and $\tilde{\boldsymbol{\beta}}_v = \frac{1}{4\tilde{\beta}_0}[\tilde{\mathbf{R}}_{\mathrm{tool}} - \tilde{\mathbf{R}}^{\mathrm{T}}_{\mathrm{tool}}]^{\vee}$ with $[\cdot]^{\vee}$ denoting the inverse operation of $[\cdot]_{\times}$; $\mathbf{K}_{\omega} \in \mathbb{R}^{3\times3}$ and $\boldsymbol{\Lambda}_{\mathrm{q}} \in \mathbb{R}^{3\times3}$ are positive diagonal matrices, and $k_{\beta} \in \mathbb{R}$ is a positive constant. This control approach simplifies the design process by enabling convenient control gain tuning and allowing easy parameter determination based on specific performance requirements. These features make the method efficient, adaptable and user-friendly, reducing complexity and saving time in controller design and optimization. More details of the MAV control can be found in ref. 56.

Manipulator MAV control

A unified controller was used for both aerial docking and manipulation tasks. Let $\mathbf{p}_{\mathrm{base}} \in \mathbb{R}^3$ and $\mathbf{p}_{\mathrm{end}} \in \mathbb{R}^3$ be the positions of the MAV base and the end-effector of the robotic arm, respectively. Denote $\mathbf{p}_{\mathrm{O}} \in \mathbb{R}^3$ as the position of the target. For aerial docking, $\mathbf{p}_{\mathrm{O}}$ refers to the position of the target tool, defined as 5.8 cm vertically above the geometric centre of the tool slot, in which the target tool is positioned. For aerial manipulation, $\mathbf{p}_{\mathrm{O}}$ refers to the position of the target object. Let $\mathbf{R}_{\mathrm{base}} \in SO(3)$ be the rotation matrix from the body frame of the manipulator MAV to the world frame. Then, we have

$$\mathbf{p}_{\mathrm{end}} = \mathbf{p}_{\mathrm{base}} + \mathbf{R}_{\mathrm{base}}\mathbf{p}^{\mathrm{base}}_{\mathrm{end}}, \tag{13}$$

$$\mathbf{p}_{\mathrm{O}} = \mathbf{p}_{\mathrm{base}} + \mathbf{R}_{\mathrm{base}}\mathbf{p}^{\mathrm{base}}_{\mathrm{O}}, \tag{14}$$

where $\mathbf{p}^{\mathrm{base}}_{\mathrm{end}} \in \mathbb{R}^3$ and $\mathbf{p}^{\mathrm{base}}_{\mathrm{O}} \in \mathbb{R}^3$ are the positions of the end effector and the target in the body frame of the manipulator MAV, respectively. The value of $\mathbf{p}^{\mathrm{base}}_{\mathrm{end}}$ can be obtained through the forward kinematics of the robotic arm, whereas $\mathbf{p}^{\mathrm{base}}_{\mathrm{O}}$ can be obtained from the onboard camera. Equations (13) and (14) imply $\mathbf{p}^{\mathrm{base}}_{\mathrm{end}} = \mathbf{R}^{\mathrm{T}}_{\mathrm{base}}(\mathbf{p}_{\mathrm{end}} - \mathbf{p}_{\mathrm{base}})$ and $\mathbf{p}^{\mathrm{base}}_{\mathrm{O}} = \mathbf{R}^{\mathrm{T}}_{\mathrm{base}}(\mathbf{p}_{\mathrm{O}} - \mathbf{p}_{\mathrm{base}})$.

Let $\mathbf{p}_{\mathrm{base,d}} \in \mathbb{R}^3$ be the designed position of the MAV base in the world frame. It is selected as a point above the target: $\mathbf{p}_{\mathrm{base,d}} = \mathbf{p}_{\mathrm{O}} + l_{\mathrm{s}}[0, 0, -1]^{\mathrm{T}}$, where $l_{\mathrm{s}} \in \mathbb{R}$ is the distance from the centre of the workspace of the robotic arm to the xy-plane of the body frame of the MAV base. The control objective is to make $\mathbf{p}_{\mathrm{base}}$, $\mathbf{p}_{\mathrm{end}}$ converge to $\mathbf{p}_{\mathrm{base,d}}$, $\mathbf{p}_{\mathrm{O}}$, respectively. The high-level velocity command is designed as

$$\dot{\mathbf{p}}_{\mathrm{base,c}} = \dot{\mathbf{p}}_{\mathrm{base,d}} - k_{\mathrm{B}}(\mathbf{p}_{\mathrm{base}} - \mathbf{p}_{\mathrm{base,d}}), \tag{15}$$

$$\dot{\mathbf{p}}^{\mathrm{base}}_{\mathrm{end,c}} = \dot{\mathbf{p}}^{\mathrm{base}}_{\mathrm{O}} - k_{\mathrm{E}}(\mathbf{p}^{\mathrm{base}}_{\mathrm{end}} - \mathbf{p}^{\mathrm{base}}_{\mathrm{O}}) - k_{\mathrm{V}}\mathbf{R}^{\mathrm{T}}_{\mathrm{base}}(\mathbf{p}_{\mathrm{base,d}} - \mathbf{p}_{\mathrm{base}}), \tag{16}$$

where $k_{\mathrm{B}} > 0$, $k_{\mathrm{E}} > 0$ and $k_{\mathrm{V}} > 0$ are control gains. The command in equation (16) drives $\mathbf{p}_{\mathrm{end}}$ to $\mathbf{p}_{\mathrm{O}}$ while helping the MAV base of the manipulator MAV maintain its position at $\mathbf{p}_{\mathrm{base,d}}$. When the robotic arm contacts the target, the resulting contact force pushes the MAV base. The calculation in equation (16) compensates for the movement of the MAV base, adjusting the motion of the arm to counteract the speed of the base. This reduces the force on the MAV base, helping it return to the desired position. The velocity command generated above may violate physical constraints, resulting in infeasible results for the manipulator MAV to track. To address this issue, we allocate velocity reference commands based on quadratic programming. The detailed method can be found in (section VI of ref. 56).

Task planning

The operation procedure of the manipulator MAV is represented as a simple state machine with five states: move, wait, operate, mount tool,

switch tool and release tool. The operation of the toolbox MAV corresponds to two states: move and wait. Based on the task requirements and the state machines, we developed a task planning software that can plan discrete waypoints while ensuring safe multi-MAV operation (Extended Data Fig. 7). After the waypoints have been planned, the motion trajectories for both the manipulator MAV and the toolbox MAV are generated using a Bezier curve-based method, which involves flight corridor generation and trajectory generation[59]. As waypoint planning can ensure that no MAVs are flying in the same area simultaneously, safe flight corridors can be generated by considering only static obstacles to avoid collisions. Task and trajectory planning for multiple mobile robots is a mature research field[60]. It is not the core contribution of this work. The above-proposed method is already sufficient to demonstrate the cooperative aerial manipulation tasks.

Data availability

The data supporting the findings of this study are presented in the paper or the extended data. Other source data related to the study are available from the corresponding author upon reasonable request. Source data are provided with this paper.

Code availability

The mechanical design and the code for disturbance estimation, visual sensing, trajectory generation and task planning are available at GitHub (https://github.com/WindyLab/FlyingToolbox).

50. Jain, K. P., Fortmuller, T., Byun, J., Mäkiharju, S. A. & Mueller, M. W. Modeling of aerodynamic disturbances for proximity flight of multirotors. In *Proc. 2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, 1261–1269 (IEEE, 2019).
51. Khan, W., Nahon, M. & Caverly, R. Propeller slipstream model for small unmanned aerial vehicles. In *Proc. AIAA Modeling and Simulation Technologies Conference (MST)*, 4907 (AIAA, 2013).
52. Davoudi, B., Taheri, E., Duraisamy, K., Jayaraman, B. & Kolmanovsky, I. Quad-rotor flight simulation in realistic atmospheric conditions. *AIAA J.* **58**, 1992–2004 (2020).
53. He, X. & Leang, K. K. Quasi-steady in-ground-effect model for single and multirotor aerial vehicles. *AIAA J.* **58**, 5318–5331 (2020).
54. Mahony, R., Kumar, V. & Corke, P. Multirotor aerial vehicles: modeling, estimation, and control of quadrotor. *IEEE Robot. Autom. Mag.* **19**, 20–32 (2012).
55. McKinnon, C. D. & Schoellig, A. P. Unscented external force and torque estimation for quadrotors. In *Proc. 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5651–5657 (IEEE, 2016).
56. Cao, H., Li, Y., Liu, C. & Zhao, S. ESO-based robust and high-precision tracking control for aerial manipulation. *IEEE Trans. Autom. Sci. Eng.* **21**, 2139–2155 (2023).
57. Bauersfeld, L., Muller, K., Ziegler, D., Coletti, F. & Scaramuzza, D. Robotics meets fluid dynamics: a characterization of the induced airflow below a quadrotor as a turbulent jet. *IEEE Robot. Autom. Lett.* **10**, 1241–1248 (2025).
58. Waslander, S. & Wang, C. Wind disturbance estimation and rejection for quadrotor position control. In *Proc. AIAA Infotech@Aerospace Conference*, 1983–1995 (AIAA, 2009).
59. Cao, H., Shen, J., Liu, C., Zhu, B. & Zhao, S. Motion planning for aerial pick-and-place with geometric feasibility constraints. *IEEE Trans. Autom. Sci. Eng.* **22**, 2577–2594 (2024).
60. Yu, J. & LaValle, S. M. Optimal multirobot path planning on graphs: complete algorithms and effective heuristics. *IEEE Trans. Robot.* **32**, 1163–1177 (2016).

Acknowledgements This work was supported by the STI 2030-Major Projects (grant no. 2022ZD0208804) and the Research Center for Industries of the Future at Westlake University (grant no. WU2022C027). We thank Y. Li, S. Guo, C. Zheng, Z. Ma and H. Chen for assistance with software and circuit development, and K. Li for assistance with the onboard vision system development. We also thank B. Zhu, R. Wang, H. Jiang, Y. Sun, and C. Ma for supporting this study.

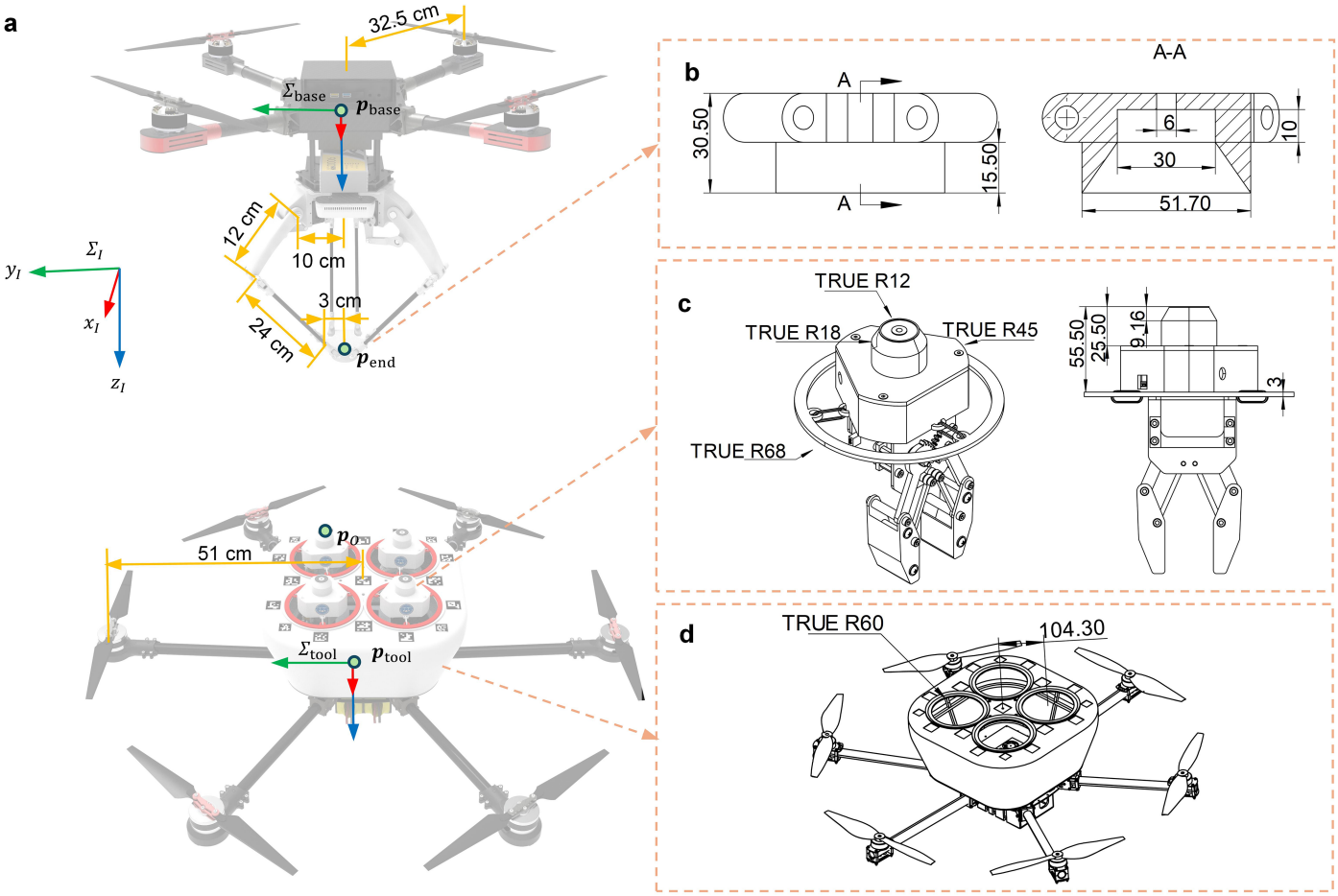
Author contributions S.Z., H.C., C.L. and S.S. designed the research; H.C., J.S., Y.Z. and Z.F. performed the research; H.C., S.Z., C.L. and S.S. analysed the data; and S.Z., H.C., C.L. and S.S. wrote the paper.

Competing interests The authors declare no competing interests.

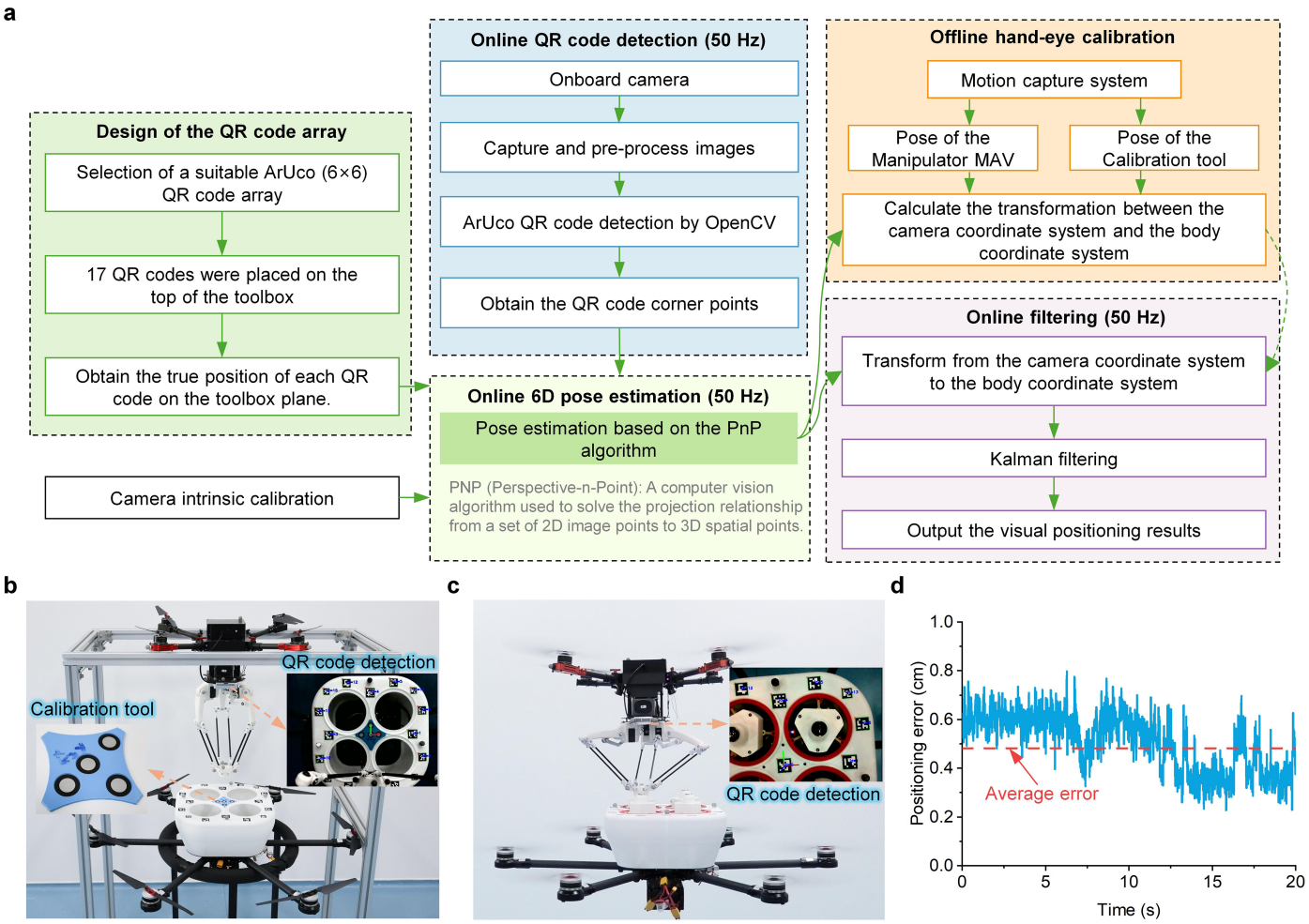
Additional information
Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41586-025-09575-x.

Correspondence and requests for materials should be addressed to Shiyu Zhao.

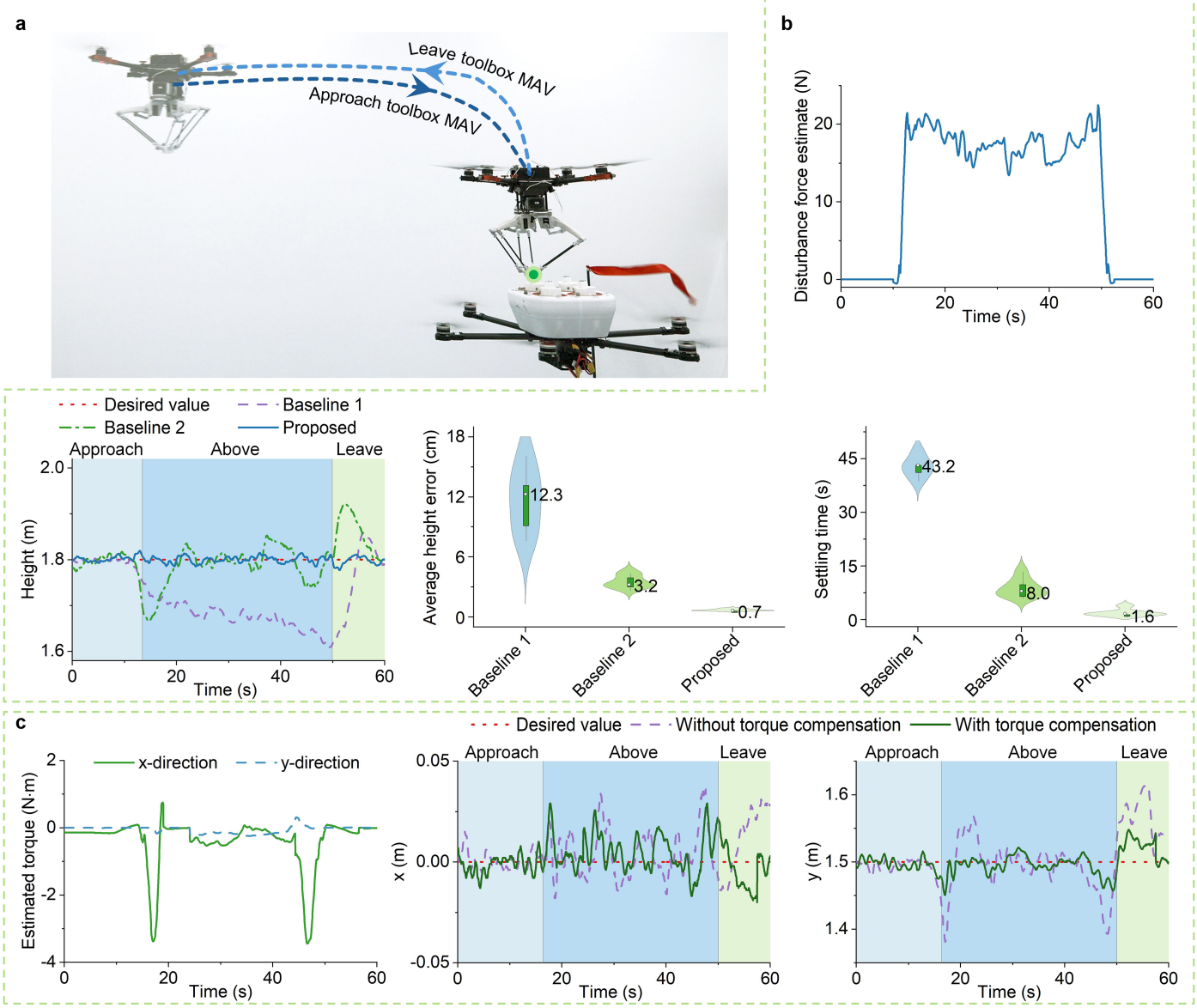
Peer review information *Nature* thanks Vladlen Koltun, Seungjae Lee and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.



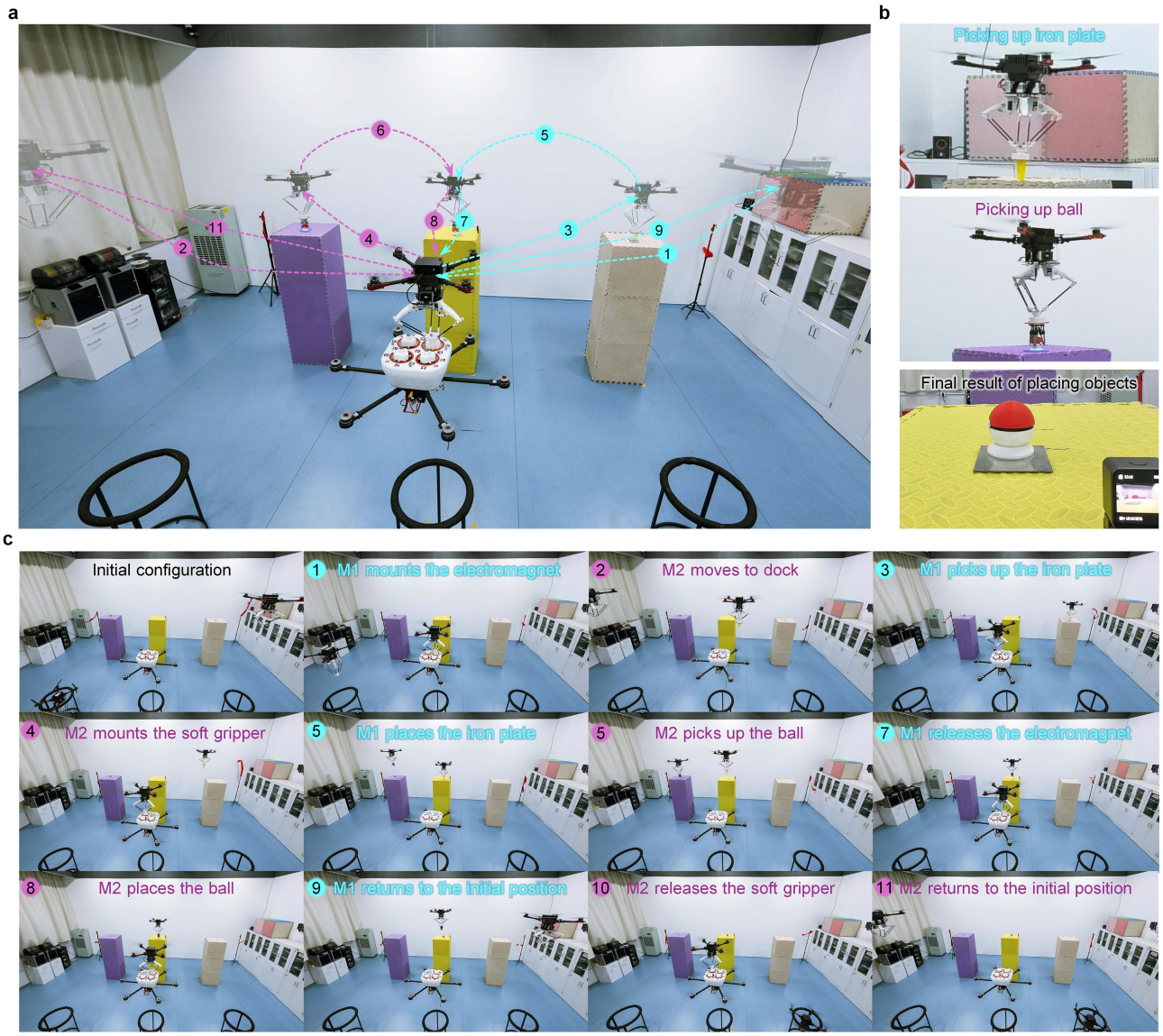
Extended Data Fig. 1 | Hardware components of FlyingToolbox. a. The motor-to-motor distances of the manipulator MAV and the toolbox MAV are 65 cm and 102 cm, respectively. The toolbox MAV is designed to be larger than the manipulator MAV to ensure sufficient payload capacity to support the toolbox and counteract disturbances caused by the manipulator MAV. The weight of the manipulator MAV including all payloads such as the robotic arm and batteries is 5.08 kg. The weight of the toolbox MAV including all payloads but tools is 6.32 kg. The six tools weigh between 0.20 kg and 0.34 kg. The dimensions of the Delta arm are shown in the figure. The reason to choose a Delta arm is that since the heavier components of the Delta arm, such as joint servos, are fixed to the MAV's body, the moving parts are lighter in mass, resulting in smaller disturbances and faster end-effector response speed. The Delta arm's end-effector can achieve a maximum linear velocity of 1.2 m/s, with a workspace of around 16 × 16 × 20 cm. When the base is fixed, its end-effector control accuracy reaches 0.2 cm. The manipulator MAV is equipped with a NUC12WSHi5 onboard computer with a hybrid Core i5 processor, while the toolbox MAV is equipped with an NVIDIA Jetson Xavier NX onboard computer incorporating a 6-core NVIDIA Carmel ARMv8.2 64-bit CPU cluster. **b.** At the bottom of the manipulator MAV's robotic arm is a conical recess containing an iron plate. The dimensions in the figure are in millimeters, and the same applies to the following two figures. **c.** Each tool features a conical protrusion with an electromagnet on top. The electromagnet can be activated or deactivated wirelessly. **d.** The top surface of the toolbox has four circular holes, each with a radius of 60 mm.



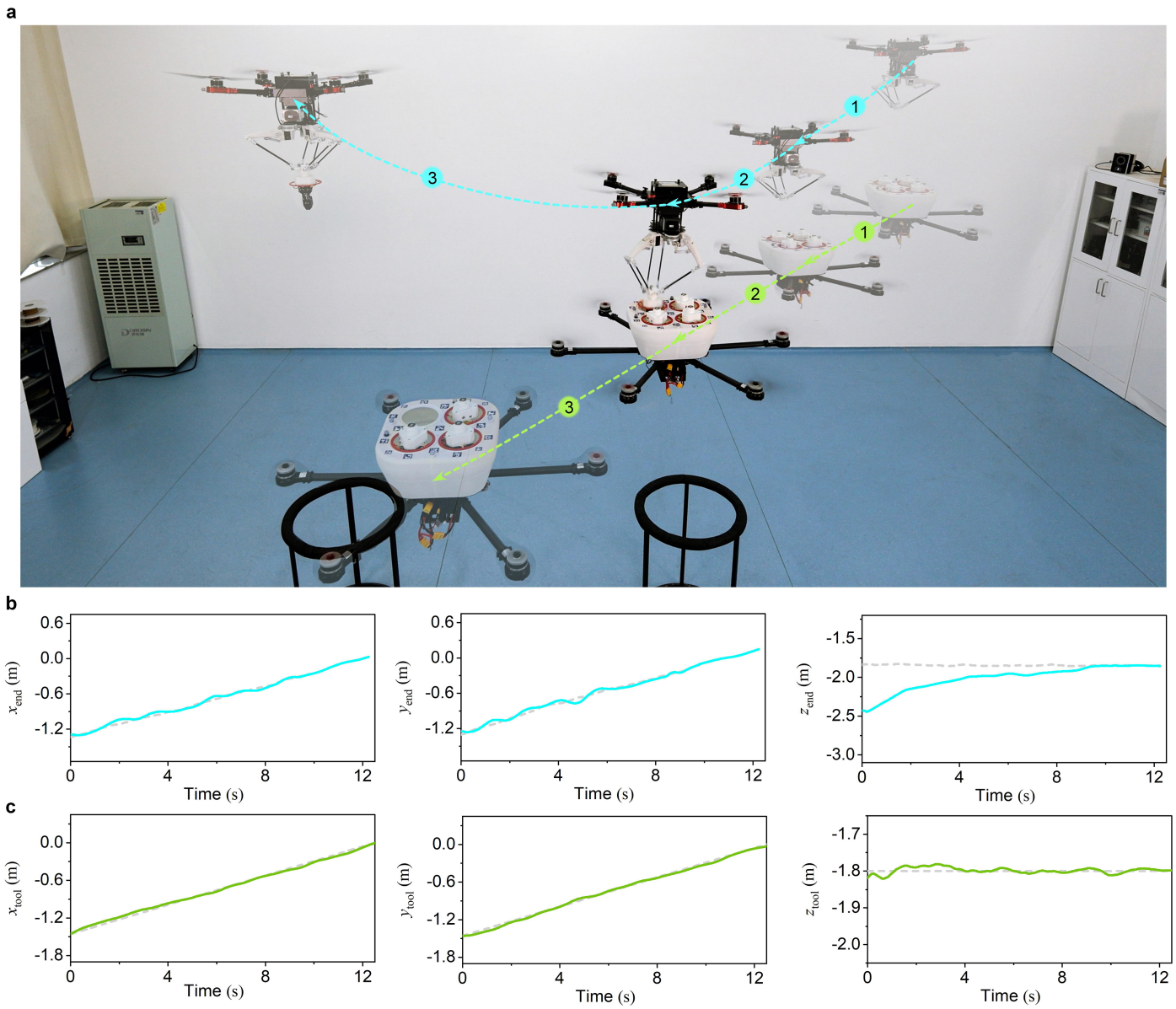
Extended Data Fig. 2 | Onboard vision system of FlyingToolbox. a. The design of the onboard vision system involves six steps: three offline steps (QR code array design, camera calibration, and hand-eye calibration) and three online steps (QR code detection, 6D pose estimation, and filtering at 50 Hz). **b.** Offline hand-eye calibration was performed using a motion capture system to determine the transformation between the camera frame and the manipulator MAV base frame. The two manipulator MAVs are equipped with an Intel RealSense L515 camera and a D435i camera, respectively, both operating at 640 × 480 pixel resolution. **c.** The onboard camera can detect a subset of the QR codes during aerial docking. **d.** The vision system achieves an average positioning error of 0.48 cm with a standard deviation of 0.12 cm when the vertical distance between the manipulator MAV and the toolbox MAV is 0.6 m.



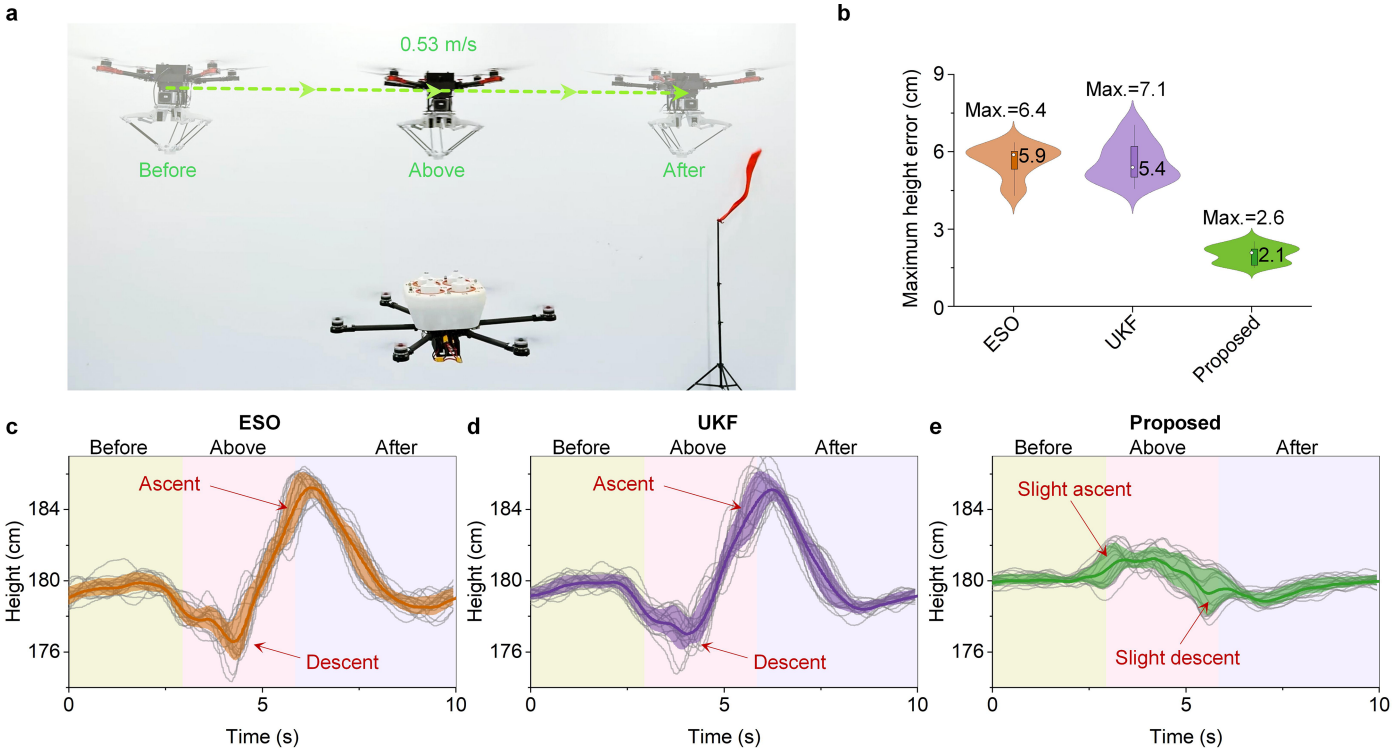
Extended Data Fig. 3 | Performance evaluation of the controller of the toolbox MAV. a. Experimental setup. The toolbox MAV remains stationary while the manipulator MAV gradually approaches from above until the bottom of the robotic arm is 6 cm above the toolbox. This creates three phases of influence on the toolbox MAV: approach, above, and leave. Here, "above" refers to the phase when at least one of the manipulator MAV's rotors enters the vertical projection area determined by the rotors of the toolbox MAV. **b**. Ablation study of force compensation. The proposed controller is compared to two baseline controllers, a PX4 open-source flight controller (baseline 1) and the proposed controller without force compensation (baseline 2). As can be seen, the proposed controller outperforms the baselines in terms of both control accuracy and settling time. The settling time is defined as the time at which the toolbox MAV's height first enters the ±1 cm error band. The statistical results were based on 5, 8, and 13 trials of experiments using baseline 1, baseline 2, and the proposed controller, respectively. **c**. Ablation study of torque compensation. As can be seen, torque compensation can improve control accuracy especially when the downwash of the manipulator MAV starts or ends impacting the toolbox MAV. For instance, the maximum error in the y-direction can be reduced from 11.9 cm to 4.8 cm with torque compensation.



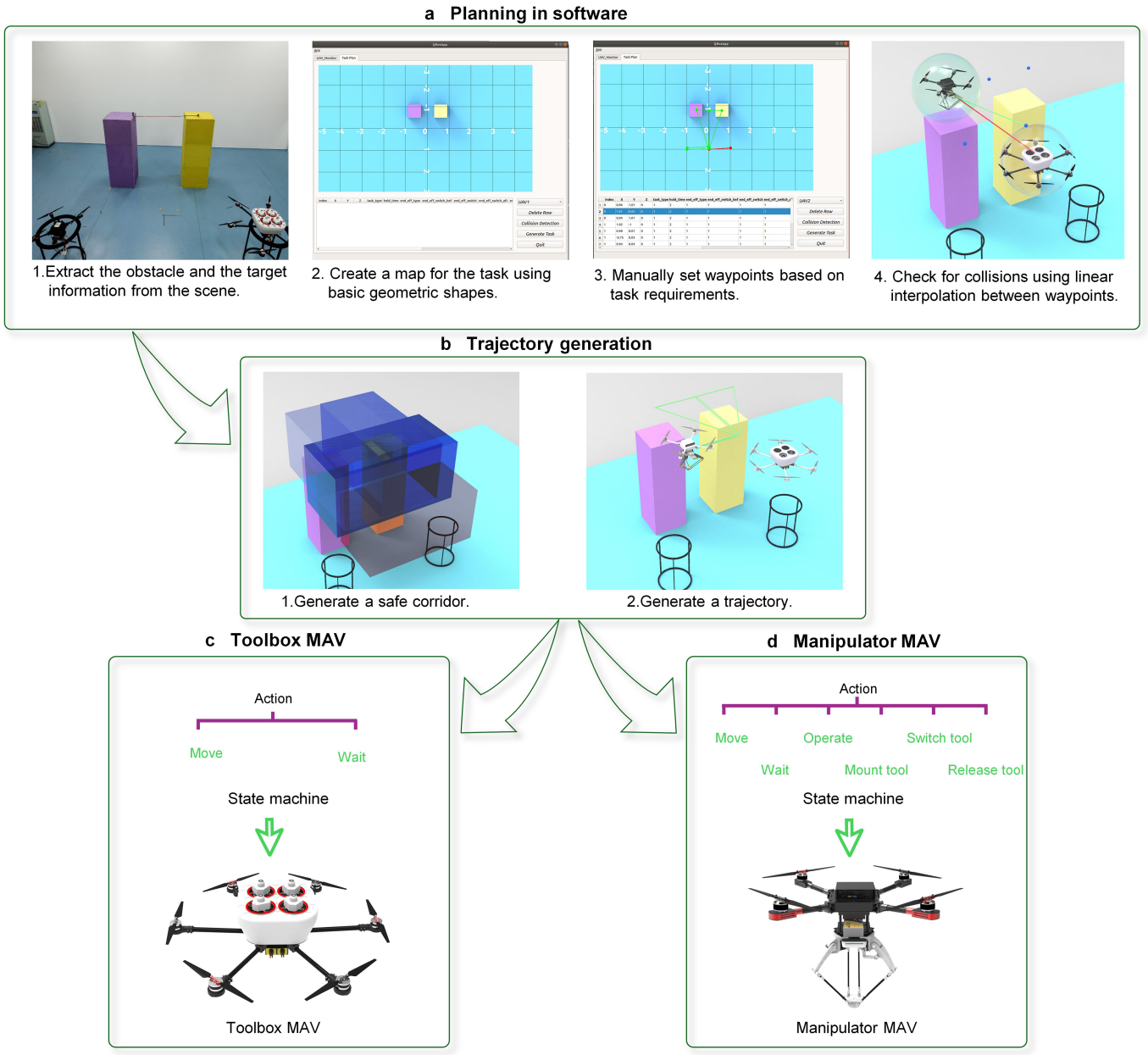
Extended Data Fig. 4 | A more complex multi-task demonstration that involves three MAVs. Two manipulator MAVs (denoted as M1 and M2) collaborate with a toolbox MAV to perform multiple manipulation tasks fully autonomously. **a**. The complete procedure. An iron plate and a ball were initially placed on top of the right and left pillars, respectively. The goal is to move the iron plate and the ball to the top of the middle pillar so that the ball is placed right on the iron plate. M1 first docked with an electromagnet tool, picked up the iron plate from the right pillar, and placed it on the middle pillar. M2 first docked with a soft gripper tool, picked up the ball from the left pillar, and eventually stacked the ball on the iron plate that had been placed on the middle pillar. **b**. Snapshots of object manipulation during the task. **c**. Snapshots of the FlyingToobox system during the task.



Extended Data Fig. 5 | Aerial docking in motion. a. The manipulator MAV docked with the toolbox MAV, which moves at a speed of 0.16 m/s. The manipulator MAV successfully retrieved a gripper tool from the toolbox. **b**. The reference (gray dotted curve) and the actual value (blue curve) of the position of the bottom of the robotic arm of the manipulator MAV before docking. **c**. The reference (gray dotted curve) and the actual value (green curve) of the position of the toolbox MAV before docking. The transition of the toolbox MAV from a hovering state to a moving state transforms the control problem from setpoint control to tracking control. The proposed control method demonstrates the capability to handle time-varying references. To obtain the time-varying position and velocity of the toolbox MAV, this experiment relies on a high-accuracy indoor positioning system rather than the onboard vision system. All other elements, such as disturbance estimation, remain consistent with the hovering case.



Extended Data Fig. 6 | Comparison of three disturbance estimation methods: unscented Kalman filter (UKF)[55], extended state observer (ESO)[56], and our proposed method. a. Experimental setup. The manipulator MAV moves along a straight line from one side of the toolbox MAV to the other side, creating three phases of influence on the toolbox MAV: before, above, and after. Here, "above" refers to the phase when at least one of the manipulator MAV's rotors enters the vertical projection area determined by the rotors of the toolbox MAV. Three methods were respectively used to estimate the vertical disturbance force, while the other system components remained the same. **b**. Maximum height error. For each estimation method, 18 experimental trials were conducted. Across the 18 trials, the proposed method's average maximum error was 2.1 cm, representing at least 61.1% reduction compared to 5.4 cm and 5.9 cm of the UKF and ESO methods. **c**,**d**, and **e** show the height variations of the toolbox MAV for the three estimation methods. Each gray curve represents a single trial. Each thick curve indicates the average over 18 trials, while the shaded area depicts the standard deviation. As can be seen, the height fluctuation by the proposed learning-based method was considerably smaller than that of the reactive methods. Notably, when the disturbance appeared, the learning-based method resulted in a slight height increase rather than a decrease in the reactive case, demonstrating its predictive characteristics.



Extended Data Fig. 7 | Motion planning of the FlyingToolbox system. **a**. Path planning for each MAV is performed by a specially designed user interface. **b**. Motion trajectories for both the manipulator MAV and toolbox MAV are generated using a Bezier curve-based method, which incorporates flight corridor and trajectory generation. Safe flight corridors are established by considering only static obstacles, ensuring that MAVs do not enter the same area simultaneously to avoid collisions. **c** & **d**. The toolbox MAV and manipulator MAV follow state machines that define the actions to be performed after completing their respective trajectories. These actions enable the MAVs to execute the corresponding manipulation tasks.