Prototypical digital twin of multi-rotor UAV control and trajectory following

L.V. Nguyen[1], T.H. Le[1], and Q.P. Ha[1]

[1]School of Electrical and Data Engineering, University of Technology Sydney (UTS), Australia

{vanlanh.nguyen; hoang.t.le; quang.ha}@uts.edu.au

Abstract -

This paper presents the development of a prototypical digital twin package for multi-rotor aerial vehicle (MAV) systems, with potential application for monitoring a construction site. The focus is how to create a virtual representation of an actual surveillance system that can be updated from real-time data to effectively help with its control, conditional monitoring, and decision-making via a versatile platform. Here, a simulation framework is developed to include the system dynamics, operation environment, and other control and tracking requirements for a wide range of MAV mission circumstances. Real-world examples from a construction site are used to illustrate the concept. A prominent advantage is that we can thoroughly test, validate, verify and evaluate the MAV control and monitoring in abnormal conditions without the need for physical implementation and field experiments for the whole system in reality. This will substantially reduce extensive testing efforts throughout the development cycle to achieve optimal performance in terms of cooperation, safety, smoothness, fault tolerance, and energy efficiency.

Keywords -

Digital twin, multi-rotor aerial vehicle, 3D modelling, control and tracking.

1 Introduction

With the technological advancements of the Industrial Revolution 4.0, multi-rotor aerial vehicle (MAV) systems are gradually becoming popular in civilian life and industrial activities. In the construction sector, they are well-established in multiple project stages such as site monitoring and 3D mapping [1], building and damage assessment [2], as well as package delivery logistics [3], indicating the potential for widespread use of MAVs.

The MAV technology recognizes simulation as an effective approach for testing and validating drone functionality. The flying environment includes a wide range of factors such as cooperative missions, traveling distances, vehicle dynamics, and obstacles. The number of possibilities increases exponentially with the number of factors and has the potential to explode. To obtain dependability certification via real drone testing, the vehicle often has to fly hundreds of miles, even in complicated circumstances, which is highly time-consuming and risky. Furthermore, these circumstances cannot be simply manufactured and replicated in real life. It is worth noting that real flight testing is only valid for a certain mechanical, electrical, and software setup. If a new configuration or software update is required, the test must be repeated. As a result, simulations will be used to test the majority of MAV features, and a framework that can mix multiple technological components and validate the design development in the early stages of functional development is critical.

Digital twin, an emerging technology, could be used for that purpose. The digital twin concept was first introduced in [4] by creating virtual digital representations of physical systems. To imitate dynamic behaviours of actual objects, their virtual versions are developed in digital form. The computerized model can describe the object's state using perceptual data, anticipating, evaluating, and interpreting dynamic changes. So far, the use of digital twins has been proven successful in industrial robots and self-driving cars [5, 6]. This technology enables an autonomous system to comprehend complex commands, determine the best action, and carry out tasks independently.

In the field of MAVs, a digital twin framework has been recently introduced in [7], made up of three parts: the virtual simulation environment, the actual MAV, and the service centre. Physical and virtual MAVs can communicate bidirectionally, allowing for the continuous assembly and updating of physical models. For multiple MAVs, a digital twin-based intelligent cooperative architecture for MAV swarms was presented in [8]. In this framework, a digital twin model is built to accurately replicate the physical entity, i.e., the MAV swarm, with high fidelity and monitor its entire process. Integrated with this model, a machine learning algorithm is built to explore the global optimal solution and control the behaviors of the swarm.

In civil engineering areas, the idea of a digital twin can find applications in architecture, infrastructure, machinery, and construction processes [9]. From the virtual viewpoint, it makes extensive use of modelling, simulation, calculation, analysis tools, and cutting-edge algorithms. Accordingly, the development of smart construction is promoted. For instance, a digital twin-enabled intelligent modular integrated construction system was introduced with a testbed robotic demonstration for cooperative decision-making and daily operation during on-site assembly [10]. The focus of existing works, however, is mostly on replicating the digital structures and tracking the

building's state. Including MAV systems, a digital twin of a building inspection scenario for reanimation and data validation has been conducted in [11]. In this study, realistic system workflow creation, comprehensive research for regulating the digital twin in the settings, and the cross-application of geographic information systems (GIS) are used to manage the geographic information production process. It includes a variety of virtual environment creation procedures, such as analyzing available GIS data, installation in the simulator, applying GIS-based real-world environment transformation tools, and gathering data with MAVs.

Although certain digital twin platforms for MAV are available in the literature, the above-mentioned works are mainly developed for system verification or communication. To the best of our knowledge, no digital twin of MAV control and trajectory following has been designed. Therefore, this paper proposes a prototype digital twin for incorporating the dynamic behaviors of typical MAVs, as well as the implementation of a MAV swarm and a cooperative path-planning algorithm. The co-simulation framework will be illustrated through construction applications in normal and faulty operations. The contribution of this paper is twofold: (i) a 3D model of the MAV and its environment for better visualization; (ii) a testing framework based on the co-simulation to demonstrate the response of MAV control and trajectory following algorithms when performing an on-site surveillance task.

2 Preliminaries

This section presents an overview of MAV operation, modelling, cooperative path planning, and control algorithms. These concepts are crucial to the design of a prototypical digital twin of MAV.

2.1 Description of MAV

Consider MAVs powered by $2N_p$ motors, where N_p denotes the number of rotor pairs, as shown in Figure 1. For symmetry, each pair of rotors are arranged on two opposite sides of the airframe. For $i = 1, 2, ..., 2N_p$, the i-th rotor rotates clockwise if i is odd or anticlockwise if i is even. Each rotor is supported by its arm of length l_i. The angles formed by two successive arms are equal. By varying the speed of the rotors, the MAV can generate roll, pitch, yaw, and altitude movements with six degrees of freedom. The pitch angle, varied in accordance with the longitudinal motion of the MAV, is produced by unbalancing the sum of the front and rear propellers' velocities, which generate the total front force, $(F_1 + F_2 + F_8)$, and the total rear force, $(F_4 + F_5 + F_6)$. Meanwhile, its lateral displacement is governed by the roll angle, which is regulated by unbalancing the sum of the speeds of the right and left rotors, resulting in total right forces, $(F_2 + F_3 + F_4)$, and total left forces, $(F_6 + F_7 + F_8)$. Yaw torque is obtained by adjusting the average speed of the clockwise and anticlockwise rotating rotors. Finally, all of the rotors contribute to the overall thrust input. In general, the MAVs operate on a similar operating principle. Differentiating features include rotor layout, size, and quantity. In practice, numerous alternative variations of MAVs are conceivable by changing the main configuration.

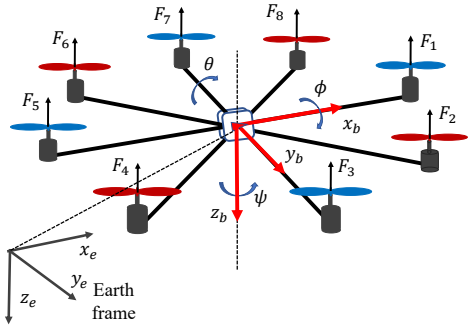


Figure 1. Configuration of the MAV system.

2.2 MAV Modelling

The goal of modelling is to express in mathematical terms the physical principles that govern the behaviours of a system in an environment or the motion of a control system. From this interpretation, calculations and simulations can be performed to analyze the system performance numerically and graphically. For MAVs, a parametric model of the vehicle is also required in order to execute the proper control. In this study, a multi-rotor system with $N_p \geq 2$ pairs of rotors is taken into consideration, assuming that the framework is rigid and symmetrical with regard to the roll and pitch axes. Besides, it is considered that the MAV's centre of gravity (CoG) is located at the origin of the body-fixed frame, and each propeller spins in the opposite direction of its two neighbors.

As illustrated in Figure 1, an earth frame, $\{x_e, y_e, z_e\}$, is attached to the ground, and a body frame, $\{x_b, y_b, z_b\}$, is mounted to the MAV's CoG, both with the z axis pointing downward. A vector $P = (x, y, z)^T$ defines the location of the CoG in the earth frame. Angles $(\phi, \theta, \psi)^T$, which correspond to roll, pitch, and yaw motion, respectively, are used to indicate the MAV orientation. These angles are limited for attitude control as $\phi \in [-\pi/2, \pi/2], \theta \in [-\pi/2, \pi/2]$ and $\psi \in [-\pi, \pi]$. According to the MAV description, the total thrust force F, represented in the body coordinate, is computed as

$$F = \sum_{i=1}^{2N_p} f_i = b \sum_{i=1}^{2N_p} \omega_i^2, \tag{1}$$

where $f_i = b\omega_i^2$ is the thrust force exerted by ith rotor, ω_i is its corresponding rotational speed, and $b > 0$ is

the thrust coefficient. The components of torque vector $\tau = [\tau_\phi\ \tau_\theta\ \tau_\psi]^T$, which represent rotation in the roll, pitch, and yaw directions, can be obtained as:

$$\tau_\phi = bl \sum_{i=1}^{2N_p} -\sin\left((i-\epsilon)\frac{\pi}{N_p}\right)\omega_i^2, \tag{2a}$$

$$\tau_\theta = bl \sum_{i=1}^{2N_p} \cos\left((i-\epsilon)\frac{\pi}{N_p}\right)\omega_i^2, \tag{2b}$$

$$\tau_\psi = \beta\left(\sum_{i=1}^{N_p}\omega_{2i}^2 - \sum_{i=1}^{N_p}\omega_{2i-1}^2\right) = \beta\left(\sum_{i=1}^{2N_p}(-1)^i\omega_i^2\right), \tag{2c}$$

where l is the length of rotor's arm, β is the apparent radius for converting the force into the yaw torque, and $\epsilon = 1$ if the MAV has a "+" configuration, or $\epsilon = 1/2$ if the MAV has an "X" configuration. In the MAV dynamic, we consider the propeller gyroscopic, τ_p, and aerodynamic torques, τ_a, as external disturbances, i.e., $d = \begin{bmatrix} d_\phi & d_\theta & d_\psi \end{bmatrix}^T = \tau_p - \tau_a$, where d_ϕ, d_θ, and d_ψ are disturbance components.

It is important to note that the thrust force F is the only control input of the 3D translational dynamics, whereas the active torques govern completely the MAV orientation. Hereafter, the vector of virtual control inputs will be denoted as $u = (u_1,\ u_2,\ u_3,\ u_4)^T$, with $u_1 = F, u_2 = \tau_\phi, u_3 = \tau_\theta$, and $u_4 = \tau_\psi$. A unified dynamical model for MAV is then given as below [12]:

$$\ddot{x} = \frac{1}{m}(c_\phi s_\theta c_\psi + s_\phi s_\psi)u_1, \tag{3a}$$

$$\ddot{y} = \frac{1}{m}(c_\phi s_\theta s_\psi + s_\phi c_\psi)u_1, \tag{3b}$$

$$\ddot{z} = \frac{c_\phi c_\theta}{m}u_1 - g, \tag{3c}$$

$$\ddot{\phi} = I_{xx}^{-1}\left[(I_{yy} - I_{zz})\dot{\theta}\dot{\psi} + u_2 + d_\phi\right], \tag{3d}$$

$$\ddot{\theta} = I_{yy}^{-1}\left[(I_{zz} - I_{xx})\dot{\phi}\dot{\psi} + u_3 + d_\theta\right], \tag{3e}$$

$$\ddot{\psi} = I_{zz}^{-1}\left[(I_{xx} - I_{yy})\dot{\phi}\dot{\theta} + u_4 + d_\psi\right], \tag{3f}$$

where s_x and c_x respectively represent $\sin x$ and $\cos x$, m is the total mass of the MAV, and $I = \mathrm{diag}(I_{xx}, I_{yy}, I_{zz})$ is the inertia matrix of the MAV.

2.3 MAV cooperative path planning

Consider a group of M vehicles that operate in a particular flight region with multiple obstacles. The MAV group's location in an earth frame is defined as $\mathcal{P} = [P_1, P_2, ..., P_M]$, where $P_m = [x_m, y_m, z_m]^T$ is the position of the m-th vehicle. The goal of cooperative path planning is to find the best routes for all MAVs to take from their starting point to target positions while meeting all formation shape, path length, threat avoidance, and turning angle limit requirements. The problem can be expressed as [13]

$$P_m(0) \xrightarrow[s.t.\ J(\mathcal{P}_m(k),\mathcal{P}_m^-(k))]{\mathcal{P}_m(k)} P_m(end),\ m = 1, 2, ..., M, \tag{4}$$

where $P_m(0)$ and $P_m(end)$ represent the initial and final postures of m-th MAV, respectively, k signifies the waypoint instant, $\mathcal{P}_m(k)$ and $\mathcal{P}_m^-(k)$ represent MAV$_m$'s path and a set of its neighbors' paths, which includes a total of K nodes subject to the multi-vehicle cost $J(\mathcal{P}_m(k), \mathcal{P}_m^-(k))$. We formulate it into an optimization problem, and we solve this problem with a cost function for each MAV in the team. The cost function for one MAV is defined as [14]:

$$J(\mathcal{P}(k)) = \omega_1 \sum_{k=1}^{K} E(k) + \omega_2 \sum_{k=1}^{K-1} L(k) + \omega_3 \sum_{k=1}^{K-1}\sum_{\tau=1}^{\mathcal{T}} D_\tau(k) + \omega_4 \sum_{k=1}^{K} H(k) + \omega_5 \sum_{k=1}^{K-2} \theta(k) + \omega_6 \sum_{k=1}^{K-1} |\varphi(k) - \varphi(k+1)|, \tag{5}$$

where formation error $E(k)$, path length $L(k)$, and safety cost $D_\tau(k)$ associated with threats $\mathcal{T}$ are concerned. The altitude payoff is denoted by $H(k)$, the turning angle and climbing angle are denoted by $\theta(k)$ and $\varphi(k)$, respectively, and the weight coefficients are denoted by ω_i for $i = 1, 2, ..., 6$.

Given the cost function $J(\mathcal{P}_m(k))$ defined for each MAV, the cooperative path planning problem becomes finding paths $\mathcal{P}_m, m = 1, 2, ..., M$ to simultaneously minimize $J(\mathcal{P}_m, \mathcal{P}_m^-)$. Since this cost depends on not only path $\mathcal{P}_m$ generated for MAV$_m$ but also its rivals' paths $\mathcal{P}_m^-$, finding optimal solutions is a challenging problem. Meanwhile, game theory has been well-recognized for resolving conflicts and handling interactions. The players of a game can decide to pursue their individual objectives by simultaneously considering the possible goals, behaviors, and countermeasures of other decision-makers to achieve a win-win situation. Therefore, this paper employs a recently developed game theory-based particle swarm optimization (GPSO) [14] to effectively generate MAV cooperative paths. The GPSO solves the collaborative path planning problem by finding the equilibrium of a Stackelberg-Nash game using a PSO hierarchical optimization framework.

2.4 MAV trajectory tracking control

The aim of trajectory tracking control is to derive the control laws that apply to each MAV's actuator to reach the desired position and attitude. Figure 2 depicts a general cascaded control structure for MAVs. The controller is designed in a hierarchical scheme, in which the inner loop is the control of the angles, and the outer loop is the position control. There are four control signals. For the fully-actuated subsystem, the control action u_1 is to control the altitude and u_4 is to control yaw motion. For the under-actuated subsystem, u_2 and u_3 are control roll angle and the pitch angle, respectively, and hence control x, y positions. The reference or desired position is typically

given by the ground operation remotely, or by a trajectory planner. To fine-tune the output of the MAV, i.e., the position and orientation, the classical PID (proportional-integrator-derivative) controller is widely employed due to its simple structure and implementation, overall good control performance, and robust design.

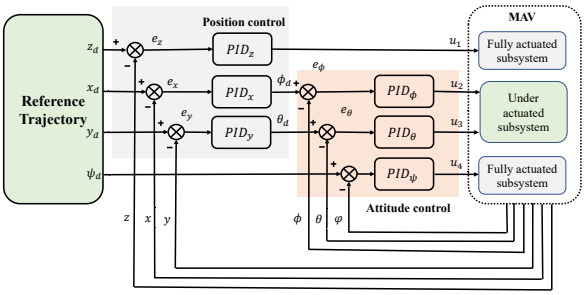


Figure 2. Control architecture of MAVs.

3 Tool development and Simulation

Given flying environments, vehicle dynamics, cooperative path planning, and control algorithms, this section provides a co-simulation framework for testing, verifying, and validating MAV technologies. Firstly, the 3D CAD model of a MAV is assembled in Solidworks to visually depict its physical twin along with its dynamic parameters obtained from the geometry and material inputs to the software. The virtual aerial vehicle is then exported to Matlab/Simulink through the supported plug-in between the two environments. Similarly, the virtual world of a case study, reflecting a construction site, is realized in the Simulink Simscape Multibody environment. The tower cranes, earthmoving vehicles, and incomplete buildings, which pose obstacles to drone navigation, are imported from their respective CAD files.

3.1 Computer-aided design MAV model

The use of computer-aided design (CAD) has proved useful and profitable for a wide range of professions. This practice assures effortless perception for users and promises growing productivity and straightforward modification in engineering endeavours. Specifically, a CAD model possesses the capability to accurately represent its physical counterpart. Hence, using this strategy encourages the visualization of the multi-rotor system, which is an essential part of the proposed digital twin platform.

In the first step, the CAD model of an actual MAV, such as the 3DR Solo drone in this work, is reconstructed and assembled using component-based and parametric modeling for visualization of the model. Therefore, modular design is possible with consideration for compatibility with the digital twin platform. The quadcopter's structural frame consists of an attached battery, replicated legs forming the landing gear, and a propulsion system comprising two pairs of contra-rotating brushless DC motors and corresponding propellers. Lastly, the materials and surface textures of each component are specified. This is a crucial step to ensure the modelling software numerically produces acceptable mass, the moment of inertia, and other intrinsic values and visually depicts its physical twin, as illustrated in Figure 3.

Figure 3. Rendered model of the 3DR Solo Drone.

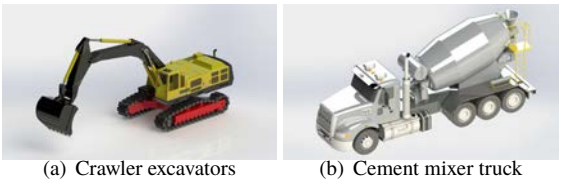
The software allows for obtaining all dynamic parameters of the drone from the manufacturer's specifications and measurements. The numerical values can be stored in a compatible file format for later use. Securing inertial figures, whether by rigorously performing measurements on specialized instruments or analytically conducting calculations, is an arduous task. Hence, by embracing CAD modeling, the developed digital twin can quickly accommodate a variety of drones on the platform.

3.2 Virtual flight environment

In general, a digital twin for MAV systems requires a virtual flight environment that accurately simulates the motion of aerial vehicles. While existing simulators such as AirSim, Gazebo, jMAVSim, and Flight-gear are useful, these simulators do not reflect in full dynamics of the UAVs in a visualized environment without a significant computational cost. To address this limitation, Solidworks is used to create 3D models of complex objects while being compatible with numerical simulation tools, such as Matlab and Simulink, reducing additional software dependencies.

To demonstrate the digital twin platform, simplified versions of construction objects were created to simulate a hypothetical construction site. These objects include crawler excavators, cement mixer trucks, luffing jib tower cranes, and unfinished buildings, all of which are utilized to simulate realistic construction activities in the virtual scenario. For example, earthwork operations and concrete-pouring processes in real construction require the use of excavators and mixer trucks, which are selected from open-source designs [15], as shown in Figures 4(a) and 4(b). Similarly, the tower crane for heavy lifting and moving materials, such as steel beams or other construction materials, is represented using component-based and parametric design

processes, as illustrated in Figure 4(c). The construction site is also simulated with unfinished buildings featuring exposed concrete floors and columns, and a general floor plan is sketched and duplicated for the upper levels, with supportive pillars patterned accordingly, as depicted in Figure 4(d). In addition, the tower blocks of the building complex appear here as an obstacle for aerial missions. By incorporating these elements into the simulated environment, the digital twin platform provides a more comprehensive and realistic representation of a construction site, enabling users to test and optimize different construction scenarios before real-world execution. Finally, the desired waypoints resulting from the cooperative path planning algorithm are represented as spherical markers, and the interpolated polynomial trajectory is featured to visually examine the MAVs' responses. A virtual environment is then created, as shown in Figure 5.



(a) Crawler excavators (b) Cement mixer truck

(c) Tower crane (d) Uncompleted building

Figure 4. CAD models of equipment and structure

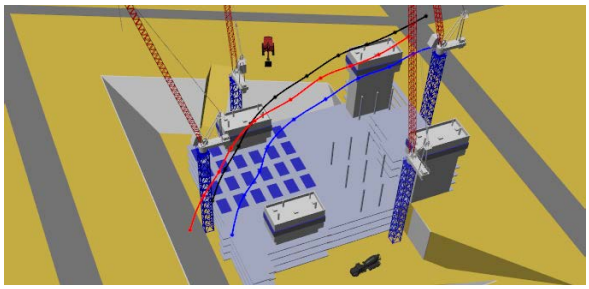


Figure 5. Virtual flight environment.

3.3 Numerical simulation

Given the MAV dynamic model, cooperative path planning and trajectory control algorithms, presented in Section 2, Matlab/Simulink are used to encode the algorithms. The digital twin of the 3DR Solo drone is also rigorously tested through tailored scenarios to verify the validity of the hierarchical control scheme. Figure 6 depicts the simulation model, including three main subsystems: trajectory generation and control, the MAV swarm, and the environment. Details of these subsystems are presented as follows.

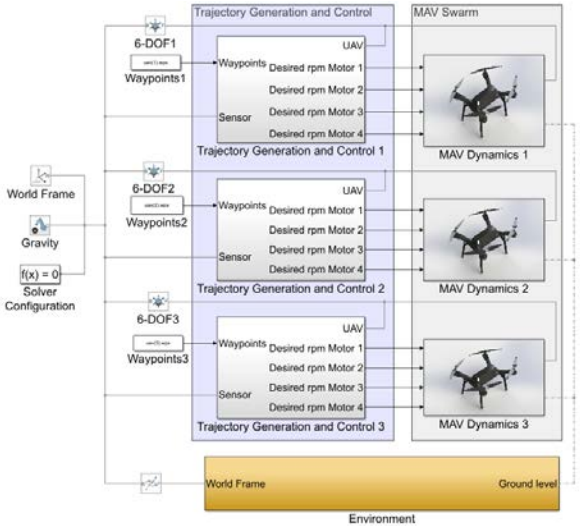


Figure 6. Simulation model

Figure 7 shows a subsystem of a MAV quadcopter. The highest level of this subsystem comprises its structural frame and the propulsion system as designed in Section 3.1. The virtual aerial vehicle is exported from Solidworks through the supported Simscape Multibody Link plug-in between the two environments. As a result, an extensible markup language (.xml) file is generated that describes all of the intrinsic, geometric relationships between each part in a CAD assembly model. After loading this file, a Simscape Multibody model of the multi-rotor is automatically constructed. Minor modifications are conducted to align the CAD model built with common practices in 3D modeling software with the simulation purposes of the digital twin platform. A graphically intuitive way of arranging the propulsion system is intentional for user-friendly purposes. Immobile parts of the drone, such as the four legs and battery, are grouped with the "Quadcopter Chasssis" block. This step should vary with different multi-rotor vehicles and depend on how intricate the models need to be in the simulations.

Figure 7. MAV subsystem

Figure 8 displays the environment subsystem, consist-

ing of the blocks for the visualization of the terrain and surroundings of the construction site along with the trajectories of the MAV swarm. For the Simscape Multibody simulation, the world origin connects to every block that is visually represented in the animation. Thus, the "World Frame" is the input to all internal blocks. Exploring the options from the TouchTerrain website, the topography of the construction site is secured in the standard triangulation language (.stl) file format. The terrain is included in the digital twin platform via a script that reads and converts a .stl file into a compatible data type for the "Rigid Terrain Grid Surface" block. Besides, the "Construction Site" block allows the direct integration of stationary CAD models of background objects into the interested test cases. Meanwhile, the "Trajectory" block enables the incorporation of markers for waypoints and the visualization of planned paths. Utilizing these blocks, the surroundings for different MAV missions can be realized within the Mechanics Explorer.

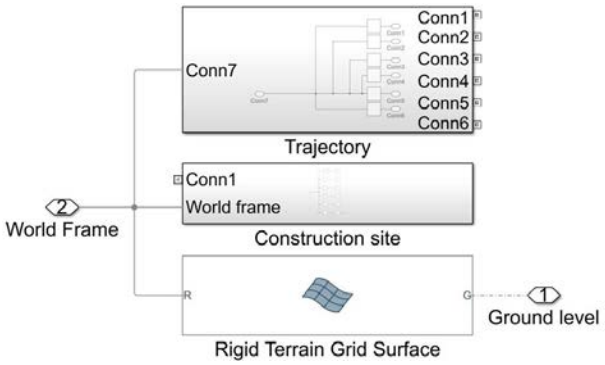
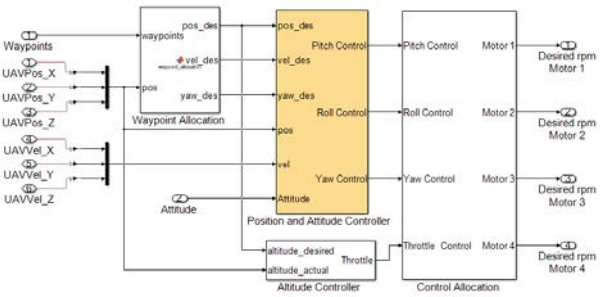
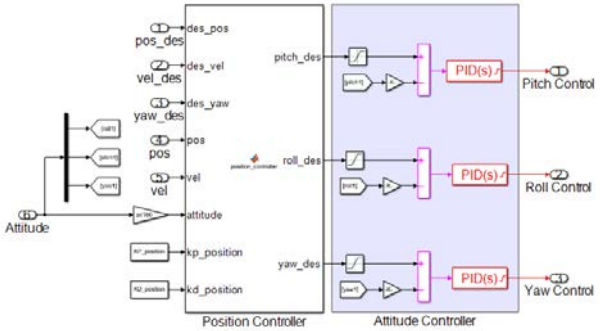


Figure 8. Environment subsystem

Figure 9(a) depicts the outer layer of the "Trajectory Generation and Control" block for one MAV. In the figure, the "Waypoint Allocation" block receives the waypoints produced by the cooperative path planning algorithm. The logic of the block sequentially assigns a new destination to the vehicle when an acceptable range to a waypoint is reached. The block will attempt to steer the vehicle to the final waypoint. The positional references are then sent to the controllers. In the "Position and Attitude Controller" block, as expanded in Figure 9(b), the position control initially receives the desired longitudinal and latitudinal values. Three major steps are executed in this block. Firstly, corrections are applied to the position errors in the form of PD gains. Then, an intermediate conversion from x and y positions to the required orientation, in this case, roll-pitch-yaw angles, is completed. Finally, the attitude PID controller attempts to improve the current rotational values accordingly. Simultaneously, the desired altitude is fed to the "Altitude Controller", at the outer layer, for appropriate adjustments to the resultant thrust. The outputs



(a) Outer layer



(b) Position and attitude controller block

Figure 9. Trajectory generation and control block

of the attitude and altitude controllers are converted into more intuitive motor revolutions for the following operations, performed in the "Control Allocation" block. The required RPM for each specific motor is then available for employment from the "MAV Dynamics" subsystem.

4 Results

4.1 Case study

To demonstrate the capabilities of our digital twin model, we conducted a virtual surveillance scenario at the Western Sydney International Nancy-Bird Walton Airport, located at coordinates $33°53'17''S$ $150°42'53''E$, which is currently under construction in Badgerys Creek, New South Wales, Australia [16]. The digitalized airport comprises integrated 3D models organized into a singular scenario. The terrain within the terminal building has been tailored to mimic specific steep and smooth areas, while the surrounding landscape remains flat due to the initial earthwork phase of construction. To supplement a realistic layer, asphalt roads for material transportation and future runways were introduced. The number of earthmoving equipment in the simulation was carefully selected to balance computational demands and visualization characteristics, resulting in a necessary number of construction vehicles being imported into the simulation. The rising tower blocks of the main building and the height of the luffing jib cranes pose the primary challenges for the aerial inspection task, with all objects being positioned on the

map to reflect the actual construction site.

The aim is to deploy a team of MAVs, configured with desired geometrical shapes from the initial to final locations, to carry out an on-site surveillance task. Our model utilized the GPSO algorithm to generate optimized, collision-free, and formation-preserving routes for the team, taking into account the environment's obstacles, as depicted in Figure 10. To ensure safe navigation of MAVs around obstacles while maintaining the desired formation, yellow cylinders were created around the tower mast to represent threat zones. The MAVs' altitudes were also constrained by the tops of the unfinished buildings and the minimum height of the crane's boom. Additionally, a cascade trajectory tracking controller was employed to ensure that the MAVs followed their planned routes accurately. This controller continuously monitored each MAV's position and velocity, comparing them to the planned trajectories, and made necessary adjustments to keep the MAVs on track.

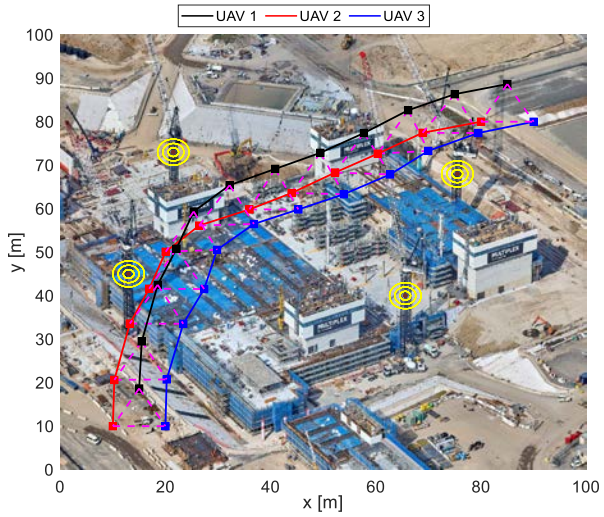


Figure 10. MAV paths on a construction site.

4.2 Normal operation

In the first test, a team of three MAVs using cascade controllers performed their mission, i.e., tracked their desired trajectory as in a normal operation. To evaluate the controller's robustness under any external excitation sources, e.g. winds, a random disturbance was introduced into the simulation model. Using the proposed digital twin platform, one can evaluate the response of MAVs as well as the performance of control and trajectory tracking algorithms, and hence improve them. A simulation scenario is depicted in Figure 11. As can be seen in the figure, the team of MAV can follow their reference without colliding with obstacles. However, the tracking error employing the current cascade controller is apparent. Accordingly, other advanced control techniques are suggested, such as iterative learning sliding mode control [17]. This is a huge benefit of the proposed digital twin.

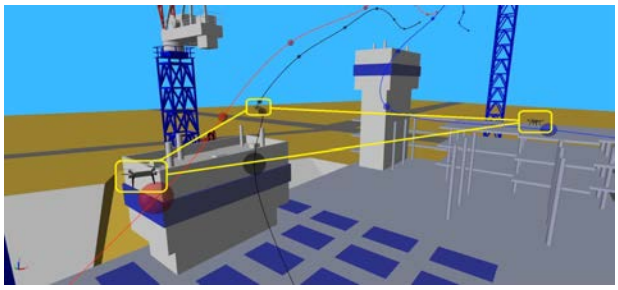


Figure 11. A visualization scenario.

4.3 Faulty operation

To further demonstrate the merit of the digital twin, another simulation is conducted with a faulty motor in one MAV. In this test, the electrical signal sent to one motor of the leader drone is disconnected during the mission. The result is shown in Figure 12. As can be expected, the 3DR Solo drone using cascade PID control is unable to continue flying and drops to the ground. For tolerance control, it is suggested to develop other types of MAV with more rotors, e.g., hexacopters or octocopters, and design a fault controller such as the one presented in [18].

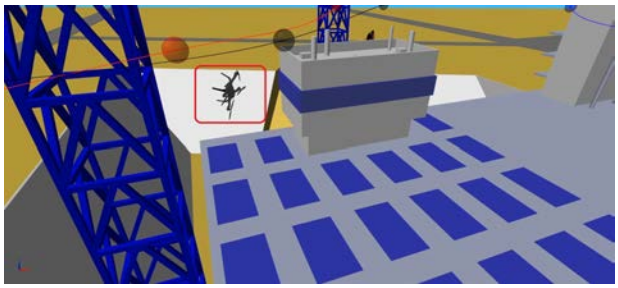


Figure 12. MAV in a faulty operation.

4.4 Discussion

Simulation results demonstrate that the digital twin model was able to generate optimal routes and trajectories for the MAV team, resulting in a safe and efficient completion of the inspection mission. The real-world scenario highlights the potential of the digital twin model to visually test the operation of a MAV team in complex and dynamic environments, such as construction sites, for safety purposes. By providing a virtual replica of the environment, the digital twin model can help optimize the flight paths of MAV teams before they are deployed to the actual site, reducing the risk of accidents and improving the quality and efficiency of the operation.

5 Conclusion

This paper has presented a digital twin platform for MAV path planning and trajectory tracking control. The physical quadcopter, one of the most typical MAVs, was digitalized into a 3D CAD model in the Solidworks software. The virtual construction site was formed by integrating CAD models of structures, vehicles, and MAVs in the common Simscape Multibody environment. Matlab/Simulink was employed to encode the working principles, path planning, and tracking algorithms of the MAV swarm. The simulation platform was validated with a team of MAVs employed in an inspection task on a construction site, demonstrating the efficacy of the proposed digital twin approach. While the system is computationally efficient and accurate in simulating the motion of aerial vehicles, validation in complex real-world scenarios will be necessary to fully assess its effectiveness. Our future work will also focus on extending the framework to dynamic environments.

References

[1] A. Bulgakov, D. Sayfeddine, T. Bock, and A. Fares. Generation of orthomosaic model for construction site using unmanned aerial vehicle. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, volume 37, pages 900–904. IAARC Publications, 2020.

[2] N. Bolourian, M. M. Soltani, A. H. Albahria, and A. Hammad. High level framework for bridge inspection using lidar-equipped uav. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, volume 34. IAARC Publications, 2017.

[3] J. Grzybowski, K. Latos, and R. Czyba. Low-cost autonomous uav-based solutions to package delivery logistics. In *Advanced, Contemporary Control*, pages 500–507. Springer, 2020.

[4] Michael W Grieves. Product lifecycle management: the new paradigm for enterprises. *International Journal of Product Development*, 2(1-2):71–84, 2005.

[5] E. G. Kaigom and J. Roßmann. Value-driven robotic digital twins in cyber–physical applications. *IEEE Trans. Industr. Inform.*, 17(5):3609–3619, 2020.

[6] Z. Hu, S. Lou, Y. Xing, X. Wang, D. Cao, and C. Lv. Review and perspectives on driver digital twin and its enabling technologies for intelligent vehicles. *IEEE trans. intell. veh.*, 2022.

[7] Y. Yang, W. Meng, H. Li, R. Lu, and M. Fu. A digital twin platform for multi-rotor uav. In *2021 40th Chinese Control Conference (CCC)*, pages 7909–7913. IEEE, 2021.

[8] L. Lei, G. Shen, L. Zhang, and Z. Li. Toward intelligent cooperation of uav swarms: When machine learning meets digital twin. *IEEE Network*, 35(1): 386–392, 2020.

[9] F. Jiang, L. Ma, T. Broyd, and K. Chen. Digital twin and its implementations in the civil engineering sector. *Autom. Constr.*, 130:103838, 2021.

[10] Y. Jiang, M. Li, D. Guo, W. Wu, R. Y. Zhong, and G. Q. Huang. Digital twin-enabled smart modular integrated construction system for on-site assembly. *Computers in Industry*, 136:103594, 2022.

[11] J. Zhang, R. Wang, G. Yang, K. Liu, C. Gao, Y. Zhai, X. Chen, and B. M. Chen. Sim-in-real: Digital twin based uav inspection process. In *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 784–801. IEEE, 2022.

[12] K. Benzaid, N. Mansouri, and O. Labbani-Igbida. A generalized dynamical model and control approach applied to multirotor aerial systems. In *2016 8th International Conference on Modelling, Identification and Control (ICMIC)*, pages 225–230. IEEE, 2016.

[13] L. V. Nguyen, I. T. Herrera, T. H. Le, M.D. Phung, R. P. Aguilera, and Q. P. Ha. Stag hunt game-based approach for cooperative uavs. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, volume 39, pages 367–374. IAARC Publications, 2022.

[14] L. V. Nguyen, M. D. Phung, and Q. P. Ha. Game theory-based optimal cooperative path planning for multiple uavs. *IEEE Access*, 10:108034–108045, 2022.

[15] URL `https://grabcad.com/library`.

[16] URL `https://en.wikipedia.org/wiki/Western_Sydney_Airport`.

[17] L. V. Nguyen, M. D. Phung, and Q. P. Ha. Iterative learning sliding mode control for uav trajectory tracking. *Electronics*, 10(20):2474, 2021.

[18] G. Michieletto, M. Ryll, and A. Franchi. Control of statically hoverable multi-rotor aerial vehicles and application to rotor-failure robustness for hexarotors. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2747–2752. IEEE, 2017.