JJMIE

Volume 15, Number 4, October. 2021
ISSN 1995-6665
Pages 337 - 345

Jordan Journal of Mechanical and Industrial Engineering

Simulation of Trajectory Tracking and Motion Coordination for Heterogeneous Multi-Robots System

Samer Mutawe, Mohammad Hayajneh, Suleiman BaniHani, Mohammad Al Qaderi

Department of Mechatronics Engineering, Hashemite University, Zarqa 13115 Jordan

Received May 4 2021 *Accepted September 15 2021*

Abstract

The paper addresses developing a team of aerial and ground robots to accomplish multi-robot system navigation task in an accessible way. The motions of two different robotic structures, namely quadcopter, and differential drive mobile robots are simulated and controlled. Two-level controller has been adopted for the multi-robot system. A low-level controller is utilized for each robotic platform to insure its motion stability and robustness. Then, a formulation of high-level tracking controller is presented to allow each robot to avoid obstacles in a dynamic environment and to organize its motion with other flying/ground robots. The performance of the proposed system is demonstrated in a simulation environment. A modeling platform is adopted to construct the simulation environment, which allows the user to easily adjust the models and controller parameters as well as to implement different control algorithms. In addition, the simulation environment helps in analyzing the obtained results and performing several tasks in different conditions. The real-time motion of multi-robot system is monitored in the created environment that provides three-dimensional graphical displays of the robotic platforms. Simulation results show that the aerial and ground robots produced trajectories individually to reach different targets. Meanwhile, each robot in the system was able to navigate among obstacles without colliding with other agents in the network.

Keywords: Ground robot, Multi-robots, Path planning, Quadcopter, Trajectory tracking;

1. Introduction

Due to rapid revolution in micro-electronics and communication technologies, robots have been employed for autonomous tasks such as security [1], inspection [2], and monitoring [3, 4]. Therefore, authors in many interesting research have developed ground and aerial robots to work in uncertain environments [2], [5-9]. In such scenarios, the ability to reach the goal point without a collision with any obstacle is the task of the navigation system [6], [10-12]. However, it is difficult or impossible for an individual robot to fulfill its tasks in a mission. Thus, a team of small and inexpensive robots with various capabilities can cooperate with each other in one network to accomplish a common goal easier and faster.

In multi-agent or multi-robot systems, the cooperation takes place among the agents using the information available from the network. Therefore, accurately measuring the vehicle’s position with respect to other vehicles and the local environment helps in two ways; avoiding collisions between robots and obstacles and locating targets effectively [13]. Many researchers work on developing allocation and path planning strategies of multiple robots (i.e. flying and ground) system for obstacle avoidance [14-16]. More specifically, a group of people developed a framework of multiple robots cooperating with each other for building indoor geometric maps [17]. Other research studies have been conducted on locating accurately or tracking a ground-moving target by aerial robot [18-20].

Several authors have looked at differences of the multi-robot motion planning. Most of these algorithms are classified into coupled or decoupled approaches to find collision-free trajectories for many automatic ground vehicles (AGVs) moving toward separate targets within a common environment [21-23]. Other approaches of robotic manipulator optimal motion planning have been proposed in [24, 25]. In contrast with coupled approach, the decoupled methods cannot achieve completeness and optimality. However, by planning the motion of robots successively, the decoupled planner has lower complexity and greater scalability than a coupled planner.

In multi-robot systems, coordination is defined as the mechanism used for achieving cooperation. Coordination and cooperation can be viewed as " joint operation or action amongst a group of robots" [26]. Coordination approaches for multi-robot systems are classified based on various parameters, where each parameter is considered as one dimension for classification. One of the comprehensive classifications of coordination approaches for multi-robot system has been presented in [27]. The paper considered three key parameters, namely decision-making strategy, communication mode, and adaptivity as basis for the classification. Coordination can be static or dynamic

* Corresponding author e-mail: samerk@hu.edu.jo

* This work is supported by the Royal Academy of Engineering grant Transforming Systems through Partnership TSP1040

according to the environment and the associated task. In dynamic coordination approaches, implicit and explicit communication modes have been used to fulfill the task [28, 29]. The coordination approaches can be categorized into three groups, namely centralized, decentralized, and hybrid, based on decision-making strategy parameter. Task planning and motion planning are key problems that need to be addressed first for multi-robot system to accomplish any task efficiently [30]. Therefore, a mechanism is needed to coordinate robots in order to generate efficient path for each robot while taking into account static and dynamic obstacles in the environment as well as the movement of other robots [31]. Motion planning coordination approaches adopt mostly decoupling method which can be centralized or decentralized coordination [32, 33].

The complexity of mathematical description for most robotic systems such as in [34-37] needs more effort to analyze their dynamic character. Therefore, those people take the advantage of appropriate modeling and simulation methods to analyze diverse movements of the mechanical system, and to provide a convenient way for practical engineering application in order to reduce time and cost as well as to find error early [34, 35].

Visual based autonomy was mainly used for cooperative multi robot systems. A vision-guided quadrotor was presented in [38]. The quadrotor was equipped with a monocular camera. It was able to achieve smooth take-off, stable tracking and safe landing with respect to a moving ground robot where a marker-based and optical-flow-based pose/motion estimation methods were used. In [39] a vision based localization and target detection algorithm was proposed for a cooperative team of one UAV and a number of UGVs to be used in crowd detection and GIS localization. The UAVs localization algorithm converts the crowds' image locations into their real-world positions, using perspective transformation. The authors in [40] present an implementation of a hybrid system consisting of a low-cost quadrotor and a small pushcart, a RGB computer vision algorithm was used for tracking and landing a quadrotor on a moving carrier using a classical PID controller. In [41] a Wii remote infrared (IR) camera fixed under a miniature unmanned aerial vehicle (UAV) was used to allow for robust tracking and landing over a small carrier vehicle without communication between them and works with an onboard 8-bit microcontroller.

Another approach is to use intelligent control algorithms, in [46] a fuzzy logic controller and a neural network were used for positioning and maneuvering of aerial robots. A fuzzy logic controller with nine rules was implemented by [47] for autonomous tracking of a multi robot team. In [48] a robotic team is presented in hostile environment. Algorithms for human detection and deep learning architecture for terrorist is presented. In [49] a surveillance system for early detection of escapers from a restricted area is presented based on a new swarming mobility model called Chaotic Rössler Mobility Model for multi Swarms. A Genetic Algorithm was used to optimize the vehicles' parameters and escapers' evasion ability using a predator-prey approach. In [50] a path tracking control system was designed using an adaptive preview model. A camera is used to collect images that are fed to the monitoring module, then operation instructions are sent to the bottom control module to create a line to judge the UAV path, then the adaptive preview time model is constructed to complete the trajectory tracking of UAV.

The ambition of this work is not only to develop a high-level framework where aerial and ground robots interact with each other, but also to evaluate different control and navigation algorithms for building geometric paths cooperatively. The current work illustrates a high-level approach that utilized Simulink to perform modeling, simulation, and control for the robotic system. A quadcopter and a mobile robot were designed in ProEngineer CAD environment with all the mass and inertial properties defined there, and then these models were exported as XML files to be read by SimMechanics in MATLAB. At this stage, joints, actuators, sensors, and inertial bodies were defined properly. This work also introduces a design of a decoupled method for multi-robot motion planning by generating small variations of robot motions on a predefined trajectory and scheduled velocity. In other words, this method first finds obstacle-free paths, and then, regulates velocities of individual robots to avoid collisions.

This paper is organized as follows: Section 2 describes the overall architecture of the robotic platforms. Mathematical models of ground robot and quadcopter are provided in Section 3. Then control systems are designed for robots stability in Section 4. In addition, Section 5 describes the design of an algorithm for robot's path planning and motion coordination. The simulation tests used to demonstrate the robots performances exist in Section 6. Finally, some conclusive remarks on the work are given in Section 7.

2. Multi-robot System Structure

The purpose of the proposed dynamic simulator for the robotic system is to develop collaborative active simulation package that supports user's interactive operation in the loop. Furthermore, it supports the applied operations by the simulation environment.

This section describes the overall construction of a robotic system, which was created by means of Simulink modeling and SimMechanic tool in MATLAB. The system combines several aerial and ground robots as shown in Figure 1. In this system, the robotic platforms were created in CAD modeling environment (e.g. ProEngineer) and were interfaced with MATLAB tools for motion control developments. The proposed robotic system allows one to insert as many aerial and ground robots as needed to demonstrate a mission performed by a multi-agent system.

The geometric models of a quadcopter and ground robot were built for simulation. The parameter values for each robot adopted in the simulations are presented in Table 1.

Table 1. CAD-model robots

Robot platform	Dimensions (m)	Weight (Kg)	Inertia $(kg.m^2\text{x}10^{-3})$
Ground robot	0.45 x 0.65 x 0.9	8.4	$I_{robot} = 175$
Quadcopter	0.2 x 0.4 x 0.4	1.2	$I_{rotor} = 0.0105\ [4.836\ 0\ 0;\ 0\ 4.836\ 0\ ;\ 0\ 0\ 8.325]$

It is noteworthy here that during the simulation process, users can apply and change the parameters of forces and torques on the created components of each robot as well as change the constraint relationship between the robots in the system.

3. Robots

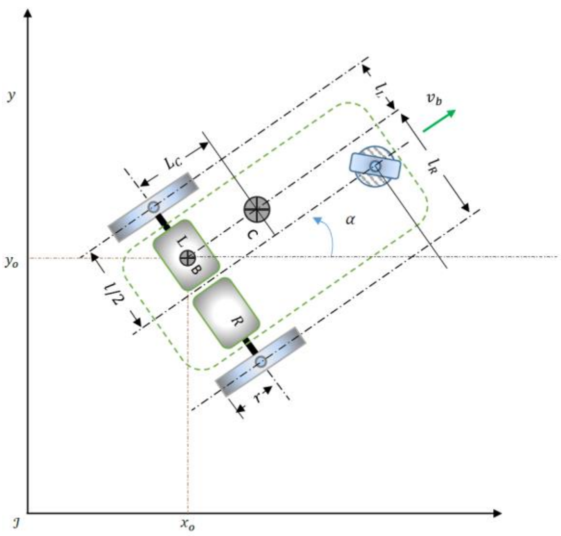


Figure 1. The schematic of differentially steered robot

Modeling

This section shows the dynamics and kinematics of a ground mobile robot and a quadcopter to derive their mathematical models. The objective of deriving mathematical models is to assist in developing controllers for physical robots.

3.1. Ground robot model

The dynamics of a differential steering robot can be defined with vector of translation speed v_b and its rotation with angular velocity ω_b which are produced by the changes of speeds of the right and the left wheels as shown in Figure 1. The robot is driven by two independent wheels while the third wheel is a castor wheel needed for static stability of mobile robot. Particularly when the two wheels rotate at the same speed, the robot will move forward. While turning left/right is achieved by driving the right/left wheel at a higher rate than the other wheel.

To develop the equation of motion of a differential steering system, an arbitrary point B is chosen and treated as a frame of reference as shown in Figure 1. By linear motion, the force vector causing the movement acts in point B and goes through center of gravity. Considering F_L and $F_R$as forces caused by the left and the right actuators in addition to $F_O$as a resistance force, the balance of forces influencing linear motion is given as follows:

$$F_L + F_R + F_O = m\dot{v}_b$$
$$\frac{M_L}{r} + \frac{M_R}{r} - k_v v_b = m\dot{v}_b \tag{1}$$

Where$m$ is robot mass. k_v is resistance coefficient against linear motion, M_L, M_R are drive torques. v_b is linear motion speed, and r is radius of the wheels.

By rotational motion, a moment M_E caused by Euler's force and a resistance moment M_O are considered. The total moments generated on the point B can be summed up in the following relation:

$$M_{BL} + M_{BR} + M_O + M_E = J_C\dot{\omega}_b$$
$$-\frac{M_L}{r}l_L + \frac{M_R}{r}l_R - k_\omega\omega_b - m\dot{\omega}_b {l_C}^2 = J_C\dot{\omega}_b \tag{2}$$

WhereM_{BL}, M_{BR} are moments by drive forces. k_ω is resistance coefficient and J_C is moment of inertia with respect to rotation axis in center of gravity. ω_b is angular speed in point B.

l_L, l_R, L_C are distances from point B to left drive, right drive, and center of gravity respectively.

Using the parallel axis theorem, the moment of inertia J_B with respect to rotation around point B is given by:

$$J_B = J_C + ml_C^2 \tag{3}$$

WhereJ_C is moment of inertia at point C.

It is important to study the mechanical performance of the mobile robot in order to build suitable control structure for a desired task. The coordinate system of the robot is governed by combined action of both the linear velocity v and the angular velocity ω. Therefore, the vehicle kinematic model is given by:

$$\begin{cases} \dot{x} = v_b\, cos\alpha \\ \dot{y} = v_b\, sin\alpha \\ \dot{\alpha} = \omega_b \end{cases} \tag{4}$$

Where(x, y) are the positions of point B in inertial frame coordinate. α is the heading angle of the robot with respect to x axis.

To minimize the error distance $\rho > 0$ between the desired location and the current one, the robot position can be represented in polar coordinates as follows:

$$\begin{cases} \dot{\rho} = -v_b\, cos\alpha \\ \dot{\theta} = -\omega_b + \frac{vsin\theta}{\rho} \\ \dot{\beta} = \frac{vsin\theta}{\rho} \end{cases} \tag{5}$$

Where$\rho = \sqrt{(x_d - x_c)^2 + (y_d - y_c)^2}$ is the measured distance between the current position and the desired position. $\theta = (\beta - \alpha)$ is the angle measured between the robot axes frame and the vector ρ. The system in (5) is valid when $\rho > 0$ and it will be engaged in creating feedback control law for robot maneuvering as discussed in section 4.

The linear speed $v_b$of point B and the angular velocity of rotation ω_b are directly produced by the change of the speeds of the right and left drives, v_R, and v_L respectively. Both drive wheels are linked to motors over ideal gearbox with gear ratio p_G that reduces the output velocities (ω_R, ω_L) and simultaneously increases the output torques.

$$\begin{cases} v_b = \frac{v_L l_R + v_R l_L}{l_L + l_R} = \frac{r}{p_G(l_L + l_R)}(l_R\omega_L + l_L\omega_R) \\ \omega_b = \frac{v_R - v_L}{l_L + l_R} = \frac{r}{p_G(l_L + l_R)}(\omega_R - \omega_L) \end{cases} \tag{6}$$

Where ω_L and ω_R are the angular speeds of the left and the right motors respectively.

3.2. Quadrotor model

The quadrotor is considered as a rigid body in the space SO(3) under external forces applied to the center of mass and expressed on earth fixed frame$J$.

$$\begin{cases} \sum_P F = M\dot{v}_p \\ -\Omega \times J_Q\Omega + \sum_P \tau = J_Q\dot{\Omega} \end{cases} \quad (7)$$

Where M is the rigid body mass, J_C is 3×3 inertia matrix around the center of mass expressed in the body-fixed frame, v is the linear velocity of the body expressed in the inertial frame J, Ω is the angular velocity of the body expressed in the body-fixed frame Q, f is the total forces on the body and, τ is the total torques on the body.

The kinematics of the quadrotor motion is given by the following relations

$$\begin{cases} \dot{\xi} = v_b \\ \dot{R} = R\hat{\Omega} \end{cases} \quad (8)$$

Whereξ is the body position with respect to inertial frame, $\hat{\Omega}$ is the skew-symmetric matrix of the body angular speed vector Ω, R is 3×3 rotation matrix that move the body vector into inertial frame.

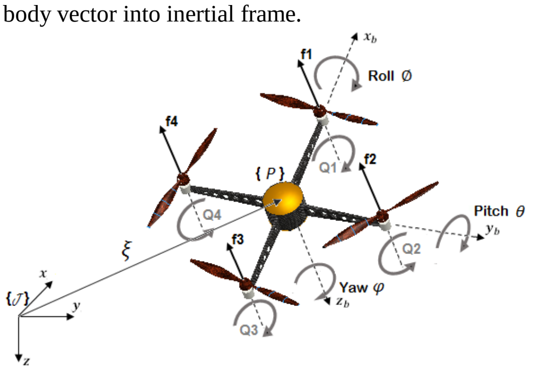


Figure 2. The quadrotor control inputs

Each rotor in the quadcopter generates a translational force f_i and torque Q_i as a function of the motor speed as shown in Figure 2. The gravitational force and the translational force (thrust) T_f are the only forces acting on the body and can be represented in inertial frame as follow:

$$\sum_P F = -R\begin{bmatrix} 0 \\ 0 \\ T_f \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ Mg \end{bmatrix} \quad (9)$$

Where g is the gravitational acceleration and the trust T_f is given as:

$$T_f = \sum_{i=1}^{4} f_i = k\sum_{i=1}^{4} {\omega_i}^2 \quad (10)$$

where ω_i is the angular speed of i^{th} motor. As shown in Figure 2, the torque produced by each i^{th} motor is denoted by Q_i. Considering motor's torque is related to an aerodynamic drag $\tau_d = k_Q\omega_i^2$, then the total torques generated at motor shaft are:

$$Q_i - \tau_d = J_M\dot{\omega}_i \quad (11)$$

Where J_M is the moment of inertia of the i^{th} motor, $k_Q > 0$ is a constant for quasi stationary maneuvers in free flight. The control torques applied at the quadrotor center of mass are thus

$$\tau_P = \begin{bmatrix} \tau_\theta \\ \tau_\phi \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} (f_1 - f_3)l \\ (f_4 - f_2)l \\ \sum_{i=1}^{4} Q_i \end{bmatrix} \quad (12)$$

Where l is the distance between the motor shaft and the center of gravity. τ_θ is the pitching moment around x_b. τ_ϕ is the rolling moment around y_b. τ_ψ is the yawing moment around z_b.

The rotation of each rotor with respect to a rotating body leads to a gyroscopic torque τ_G to the air frame and is given by the following relationship

$$\tau_G = \sum_{i=1}^{4} J_M\,(\Omega \times z_b)\omega_i = (\Omega \times z_b)\sum_{i=1}^{4} J_M\,\omega_i \quad (13)$$

Where $(\Omega \times z_b)$ denotes the cross product between the body angular speeds vector and the axis z_b. This yield to the final quadcopter model:

$$\begin{cases} \dot{\xi} = v_b \\ \dot{R} = R\hat{\Omega} \\ -R\begin{bmatrix} 0 \\ 0 \\ T_f \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ Mg \end{bmatrix} = M\dot{v}_b \\ -\Omega \times J_Q\Omega + \tau_P + \tau_G = J_Q\dot{\Omega} \end{cases} \quad (14)$$

4. Robots controllers

Several control approaches in the literature have been proposed for controlling the mobile robot and the quadcopter. Recently, many modern control methodologies such as nonlinear control [20, 51], optimal control [52], adaptive control [53], and fuzzy control [54, 55] have been widely intended for robots control. However, these approaches are difficult to implement in embedded systems. Therefore, the choice of a PID control approach in many world control problems refers to its simple structure and efficiency with its three-term functionality covering both transient and steady states. However, the difficulty in this type of controller is to obtain the optimal solution through tuning its gains.

Ground robot control

The control algorithm is designed to drive the robot from its current configuration (x_c, y_c, α_c) to the desired one (x_d, y_d, α_d) with respect to the inertial frame. Using kinematics motion models, the control inputs are the linear and angular velocities of the robot (i.e. $u = (v \;\; \omega)$) which are acting on its center of gravity. PI controllers are adopted here to control the robot as follows:

$$\begin{cases} v = \left(k_\rho\rho + k_i\int_0^{\delta T} \rho\, dt\right)cos(e_\alpha) \\ \omega = k_\alpha(e_\alpha) + k_{i\alpha}\int_0^{\delta T} e_\alpha\, dt \end{cases} \quad (15)$$

Where $\rho = \sqrt{(x_d - x)^2 + (y_d - y)^2}$ is the error distance, and $e_\alpha = \alpha_d - \alpha_c$ is the difference between the desired heading angle and the current angle from the x axis in the inertial frame. The control vectors stated in (15) makes the robot motion behavior to be smooth and stable.

The linear and angular velocities of the robot are transformed to right and left wheel speeds. Considering that the rotating point occurs at the middle between the right and left drive (i.e. $l_L = l_R = l$), then the motors speeds can be given as follows:

$$\begin{cases} \omega_R = \dfrac{v}{r} + \dfrac{l\omega}{2r} \\ \omega_L = \dfrac{v}{r} - \dfrac{l\omega}{2r} \end{cases} \quad (16)$$

Quadcopter control

The controlled inputs of the quadcopter are three torques and thrust. The attitude is controlled by the quadcopter torques which are related to angular velocities by $\tau = J_Q\dot{\Omega}$, when Coriolis and gyroscopic terms are neglected. The attitude stability depends mainly on the accuracy in altitude, which keeps the quadcopter aloft. Therefore, a PID controller is adopted for the attitude stability and another PID is used to stabilize the vehicle altitude. The controller outputs are given in the following relationships:

$$\begin{cases} \tau = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = J_Q\left(K_p e_\Omega(t) + K_i\int_0^{\delta T} e_\Omega(t)\,dt + K_d\dfrac{de_\Omega(t)}{dt}\right) \\ T = \dfrac{M}{cos\phi cos\theta}\left(g - \left(K_p e_z(t) + K_i\int_0^{\delta T} e_z(t)\,dt + K_d\dfrac{de_z(t)}{dt}\right)\right) \end{cases} \quad (17)$$

where K_p, K_i, and K_d are the proportional, integral, and derivative gains respectively. The difference between the desired state (i.e. ϕ_d, θ_d, ψ_d in attitude and z_d in altitude) and the current state is represented by the error signals e, which are computed in a sample period.

The quadcopter horizontal positions are coupled with thrust vector and are controlled by orienting the vector towards the desired direction. Referring to (14), the desired linear horizontal accelerations are calculated from PD feedback of the position and velocity errors by the following relationships:

$$\begin{cases} \ddot{x}_d = K_p e_x + K_d\dfrac{de_x(t)}{dt} + \ddot{x}_f \\ \ddot{y}_d = K_p e_y + K_d\dfrac{de_y(t)}{dt} + \ddot{y}_f \end{cases} \quad (18)$$

where K_p, K_i, and K_d are the proportional, integral, and derivative gains respectively. $e_x = x_d - x$ and $e_y = y_d - y$ are the position error vectors. The terms $\ddot{x}_f$ and $\ddot{y}_f$ represent feedforward accelerations. The horizontal positions are relative to the rolling and pitching angles ϕ and θ. By assuming small rolling and pitching angles and a thrust magnitude equal to Mg, the desired angles ϕ_d and θ_d can be given by the following relationships:

$$\begin{cases} \phi_d = \dfrac{1}{g}(\ddot{x}_d\ sin\psi_d - \ddot{y}_d\ cos\psi_d) \\ \theta_d = \dfrac{1}{g}(\ddot{x}_d\ cos\psi_d + \ddot{y}_d\ sin\psi_d) \end{cases} \quad (19)$$

Finally, the resultant thrust and torques that are generated by the controllers in (17) can be computed as four thrusts generated from quadrotor's propellers. For this task, proper allocation matrix is computed accordingly to mechanical parameters such as "plus" mounted quadrotor and is given as follows:

$$\begin{bmatrix} T \\ \tau \end{bmatrix} = \begin{bmatrix} -1 & -1 & -1 & -1 \\ 0 & -l & 0 & l \\ l & 0 & -l & 0 \\ -K_Q & K_Q & -K_Q & K_Q \end{bmatrix}\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \quad (20)$$

where l is the distance between the rotor and the center of gravity. For each $i = 1,\ldots,4$, f_i is the generated thrust from each rotor and it is a function of the angular speed of the motor (i.e. $f = k\omega^2$).

5. Path Planning and Motion Coordination

In this work a centralized decoupled planner is adopted for path planning and motion coordination of multi- robot system. In this approach an individual path is generated independently for each robot in order to reach its desired target where all of the state information is available to a single processor. Then another plan is involved to generate velocity profile for each robot to avoid collisions with other moving agent in the network.

Each robot in the proposed system is required to track a desired trajectory to reach a target. Specifically, the control structure proposed in section *4* for each robot is sufficient to control the tracking error to converge to zero, where the tracking error of the i_{th}-robot is defined as:

$$e_{ti} = q_{di} - q_i \quad (21)$$

Where$(i = 1,2,3,\ldots,n)$ denotes the i_{th} robot in the network and n is the total number of the individual robots. In addition, q_i is a vector of robot's positions in the inertial coordinate system and q_{di} is a vector of robot's desired trajectory.

Each robot in the proposed robotic system shares its position information with other robots. Moreover, the positions of unknown non-moving obstacles are assumed to be learned by robots using their own sensors. The robot is required to navigate to its target avoiding fixed and moving obstacles.

Based on the equation given in (21), the robot will move to the desired target by following the given trajectory, unless an obstacle is presented in the path of the robot or another robot is coming towards it. In this case, it is essential to look for alternative solution paths by feeding small variations of robot motions by affecting both its predefined path and its scheduled velocity. Hence, we define a new control law for the ground robot to provide feed forward signals on the robot motion as follows:

$$\begin{cases} \omega_{fi} = \displaystyle\sum_{j\neq i}^{n} k_f\dfrac{cos(\theta_{ij})}{e^{(|q_{ij}|-1)^2}} \\ v_{fi} = \dfrac{1}{k_v}\omega_{fi} \end{cases} \quad (22)$$

Where$i(1 \leq i \leq n)$ denotes the i_{th} robot in the network and n is the total number of the individual robots. In addition q_{ij} denotes the vector of displacements of the i_{th} robot coordinates from the j_{th} robot coordinates in inertial frame. θ is the angle between the q_{ij} vector and the vector of displacement of i_{th} robot from the target coordinates in inertial frame as shown in *3*. ω_{fi} and v_{fi} are the feed-forward angular and linear speed control inputs which are tuned by the gain constants k_f and k_v respectively. It is obvious from (22) that the value of ω_{fi} increases when θ_{ij} and q_{ij} decrease. For example, the effect on the value ω_f from robot 4 is greater than the effects coming from robot 2 and robot 3 because of the shortest distance between robot 1 and robot 4 as well as the smallest angle value (θ_{14}) as shown in Figure 3.

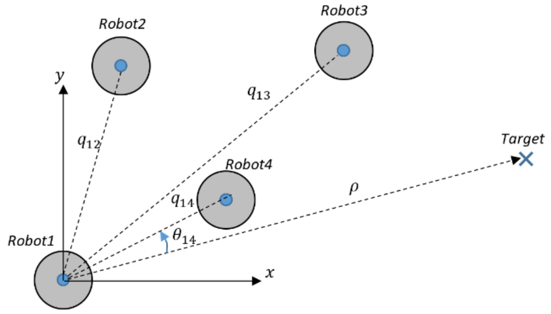


Figure 3. Robot coordinates with other robots in the network

In the case of the aerial robot, it is possible to avoid collision with other aerial vehicles by finding a path in three dimensions, but we assume that a number of quadcopters are required to stay at the same height. Therefore, the quadcopter should find a path in the horizontal plane to avoid collision with other quadcopters by the following control inputs:

$$\begin{cases} v_{fxi} = \sum_{j\neq i}^{n} k_r \frac{cos(\theta_{ij})}{e^{(|q_{ij}|-1)^2}} \\ v_{fyi} = \sum_{j\neq i}^{n} k_r \frac{sin(\theta_{ij})}{e^{(|q_{ij}|-1)^2}} \end{cases} \quad (23)$$

Wherev_{fxi} and v_{fyi} are the feed-forward linear velocities to the i_{th} quadcopter in x and $y$directions respectively. We assume that the feed-forward velocities are equal to zero when there is enough height distance between the i_{th} quadcopter and other quadcopters. As we discussed in the ground robot case, the velocity values in (23) are approaching to zero when the horizontal distances between i_{th} quadcopter and the others are very long. The velocities in (23) are represented in inertial frame. Therefore, the values required in body frame are calculated by considering the heading angle.

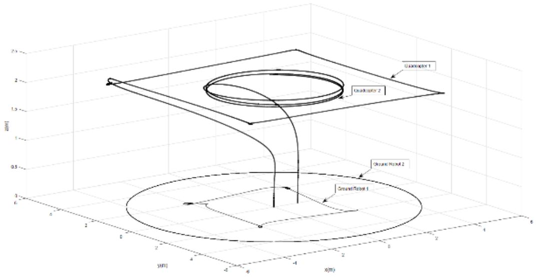


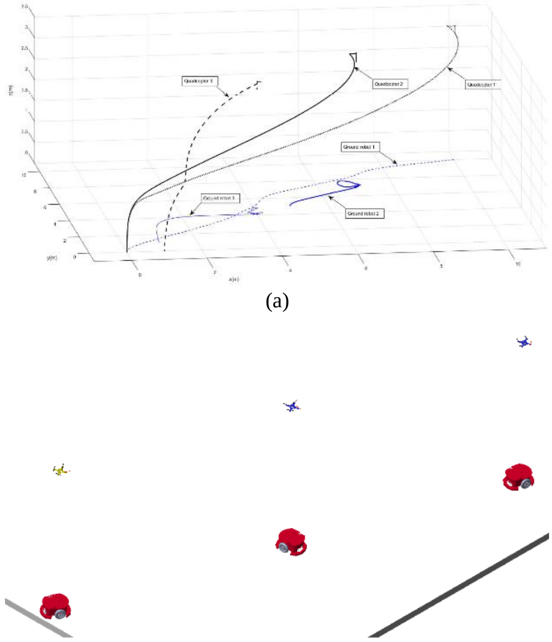
Figure 4. Motion of aerial and ground robots on assigned trajectories

6. Simulation Results

This section serves to illustrate the effectiveness of the proposed control algorithms for robot's trajectory tracking and path coordination. The simulation results are executed first to show the performance of each robot independently, and to study its stability and ability to track a pre-defined trajectory. Then the second part of the simulation shows the performance of each aerial and ground robot in generating new paths to avoid collision with other robots.

6.1. Trajectory tracking

Position and orientation stability of each robot were achieved by using PID controllers. The gains of the controllers were tuned properly to guarantee best performance. More precisely, attitude and position controller gains were chosen in order to minimize oscillations and settling time. The adjusted gains of the PIDs in the quadcopter and ground robot are given in Table. To ensure the validity of the used PID controllers in the proposed robots, external disturbances as random form signals were carried out on each robot. These signals are applied on all axes in body frame of each robot with a value range of $(-0.4\ to\ 0.4)N$.



(a)

(b)

Figure 5. Mission Scenario 2: (a) Robots trajectories, (b) The final positions of the system robots

Table 2:PIDs gains in the quadcopter and ground robot

Quadcopter PIDs				Ground robot PIDs			
	P	I	D		P	I	D
Alt	30	20	15	Position x	15	5	0
Positions x,y	5	0	3	Position y	15	5	0
Roll/Pitch	3	0	2	Rotation	5	3	0
Yaw	0.5	0.1	0.1				

Figure 4shows how the robots (i.e. two quadcopters and two ground robots) track the assigned paths correctly. In this scenario the robotic platforms were initialized on the ground from different positions. One of the ground robots and one of the quadcopters were demanded to follow rectangular-shape paths. Meanwhile, the other quadcopter and ground robot were needed to track a circular-shape trajectory. The vehicles maintain the performance of the motion in terms of stability and proves quite perfect characteristics in terms of ability to track the predefined trajectories.

6.2. Path planning for collision avoidance

The following two figures illustrate the performance of the robotic platforms which are demonstrated by two different scenarios in order to validate the ability of each robot for collision avoidance with other aerial or ground robots in a multi-robots network.

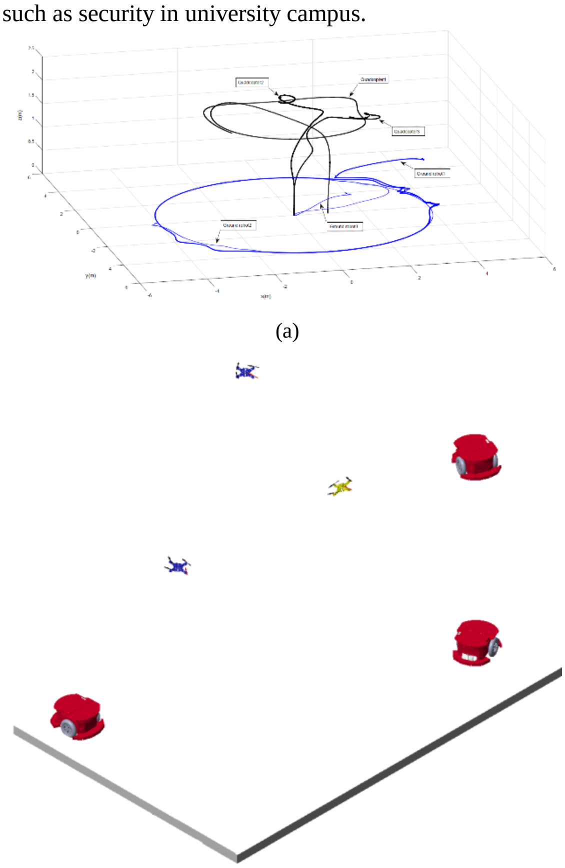
Figure 5 shows the performance of the proposed system in a mission scenario where each robot moves towards a specific target point and then it waits there for a while. More specifically, ground robot 1 , 2, and 3 are requested to move toward points (4,4), (7,7), and (10,10) respectively which are aligned on a straight line. Also, quadcopter 1, 2, and 3 are needed to move from their initial positions on the ground to the points (4,4,3), (7,7,3), and (10,10,3) respectively which are also falling on a straight line. It is clear from the figure that robot 3 (ground or aerial) is creating a curved-line instead of straight line during motion to its target, in order to avoid collision with robot 1, and 2. The same methodology is applied also on robot 2 to move away from robot 1.

When a robot reaches its desired fixed-point location, it stops over that point for a period of time. The stopped robot can react quickly to other robots approaching it, accordingly it moves enough distance from the coming robot to avoid an accident and at the same time to allow the robot to return to its desired position as soon as possible. This situation is characterized through the figure by the irregular path, which is formed at the end of the robot track around the fixed point.

Finally, Figure 6 shows the performance of the robotic platforms in a different scenario where two of the ground robots were required to follow a circular path. The ground robot 1 was moving in an opposite direction of robot 2. The third ground robot was required to travel to a home location at (0, 0). This scenario also included three quadcopters; one of the quadcopters (quadcopter 1) was trying to track a circular path, and two of them were targeting the points (2, 0, 2) and (0, 2, 2) which were located on the same path of quadcopter 1. The figure shows also how the ground robot and the quadcopter on the circular paths were reconstructing the desired trajectory when it was needed to avoid collisions between robots.

7. Conclusions

This work presented the modeling, simulation, and control of a multi-robot system, which contains models of aerial and mobile robots. Each of the robotic platforms was built by means of Simulink modeling and SimMechanics and was controlled by applicable control methods for motion stability and maneuverability. This work also proposed a new strategy for path planning of robot motion in a network of several aerial and ground robots. Providing this method, the robot can change its predefined path to avoid collisions with other robots on the path by manipulating feed-forward speeds via PID controller. The results demonstrated the validation of this strategy on a multi-robot system, and illustrated how the track path of a robot can be reshaped smoothly. The robotic system was built in a simulation environment that provides many advantages for multi-agent robotic systems; one can insert and remove several robots easily as well as modifying the parameters of the robot model in CAD platform, which can be reconsidered in the simulation, needs less time and effort. The performance of the robotic system was demonstrated and validated in different motion scenarios. The results showed high stability in robots' motions and good performance in trajectory tracking, even when disturbances exist. The performance of the proposed system when evaluated on different scenarios suggests that the robotics system has very good maneuverability for collision avoidance when multi-agent robots are engaged in common tasks. The present work will be extended to include real implementation of the proposed system in real application such as security in university campus.



(a)

(b)

Figure 6. Mission Scenario 2: (a) Robots trajectories, (b) The final positions of the system robots

References

[1] G. Marek, Š. Peter, "Design the robot as security system in the home". Procedia Engineering, vol. 96, pp. 126–130, 2014.

[2] P. Rea, E. Ottaviano, "Design and development of an inspection robotic system for indoor applications". Robotics and Computer-Integrated Manufacturing, vol. 49, pp. 143–151, 2018.

[3] A. Kurkin, D. Y. Tyugin, V. Kuzin, A. Chernov, V. Makarov, P. Beresnev, V. Filatov, D. Zeziulin, "Autonomous mobile robotic system for environment monitoring in a coastal zone". Procedia Computer Science, vol. 103, pp. 459–465, 2017.

[4] J. Peterson, W. Li, B. Cesar-Tondreau, J. Bird, K. Kochersberger, W. Czaja, M. McLean, "Experiments in unmanned aerial vehicle/unmanned ground vehicle radiation search," Journal of Field Robotics, vol. 36, no. 4, pp. 818–845, 2019.

[5] A. K. Nasir, A. Hsino, H. Roth, K. Hartmann, "Aerial robot localization using ground robot tracking–towards cooperative slam". IFAC Proceedings Volumes, vol. 46, no. 19, pp. 313–318, 2013.

[6] R. Abiyev, D. Ibrahim, B. Erin, "Navigation of mobile robots in the presence of obstacles". Advances in Engineering software, vol. 41, no. 10-11, pp. 1179–1186, 2010.

[7] M. Hayajneh, M. Melega, L. Marconi, "Design of autonomous smartphone based quadrotor and implementation of navigation and guidance systems". Mechatronics, vol. 49, pp. 119–133, 2018.

[8] M. Hayajneh, A. Badawi, "Automatic uav wireless charging over solar vehicle to enable frequent flight missions". 3rd international conference on automation, control and robots, 2019, pp. 44–49.

[9] M. Al-Fetyani, M. Hayajneh, A. Alsharkawi, "Design of an executable anfis-based control system to improve the attitude and altitude performances of a quadcopter drone". International Journal of Automation and Computing, vol. 18, no. 1, pp. 124–140, 2021.

[10] R. H. Abiyev, B. Erin, A. Denker, "Navigation of mobile robot using type-2 fuzzy system". in International Conference on Intelligent Computing. Springer, 2017, pp. 15–26.

[11] A. Meléndez, O. Castillo, F. Valdez, J. Soria, M. Garcia, "Optimal design of the fuzzy navigation system for a mobile robot using evolutionary algorithms". International Journal of Advanced Robotic Systems, vol. 10, no. 2, p. 139, 2013.

[12] S. BaniHani, M. Hayajneh, A. Al-Jarrah, S. Mutawe, "New control approaches for trajectory tracking and motion planning of unmanned tracked robot". Advances in Electrical and Electronic Engineering, vol. 19, no. 1, pp. 42– 56, 2021.

[13] J. Sebestyénová, P. Kurdel, "Simulation of self-organizing multi-robotic system used for area coverage and surround of found targets". WSEAS Transactions on Information Science and Applications, vol. 11, pp. 130–138, 2014.

[14] J. H. Liang, C. H. Lee, "Efficient collision-free pathplanning of multiple mobile robots system using efficient artificial bee colony algorithm". Advances in Engineering Software, vol. 79, pp. 47–56, 2015.

[15] D.H. Lee, "Resource-based task allocation for multi-robot systems". Robotics and Autonomous Systems, vol. 103, pp. 151–161, 2018.

[16] M. Saska, V. Vonásek , T. Krajník, L. Přeučil, "Coordination and navigation of heterogeneous MAV–UGV formations localized by a 'hawk-eye'-like approach under a model predictive control scheme". The International Journal of Robotics Research, vol. 33, no. 10, pp. 1393–1412, 2014.

[17] P. K. Penumarthi, A. Q. Li, J. Banfi, N. Basilico, F. Amigoni, J. O'Kane, I. Rekleitis, S. Nelakuditi, "Multirobot exploration for building communication maps with prior from communication models". in Multi-Robot and Multi-Agent Systems (MRS), 2017 International Symposium on. IEEE, 2017, pp. 90–96.

[18] G. Loianno, V. Spurny, J. Thomas, T. Baca, D. Thakur, D. Hert, R. Penicka, T. Krajnik, A. Zhou, A. Cho, M. Saska , "Localization, grasping, and transportation of magnetic objects by a team of mavs in challenging desert like environments". IEEE Robotics and Automation Letters, vol. 3, no. 3, pp. 1576– 1583, 2018.

[19] J.-E. Gomez-Balderas, G. Flores, L. G. Carrillo, R. Lozano, "Tracking a ground moving target with a quadrotor using switching control". Journal of Intelligent & Robotic Systems, vol. 70, no. 1-4, pp. 65–78, 2013.

[20] S. Mutawe, M. Hayajneh,S. BaniHani,"Robust Path Following Controllers for Quadrotor and Ground Robot". 2021 International Conference on Electrical, Communication, and Computer Engineering (ICECCE), vol. 12 pp. 1-6, 2021.

[21] M. Peasgood, C. M. Clark, J. McPhee, "A complete and scalable strategy for coordinating multiple robots within roadmaps". IEEE Transactions on Robotics, vol. 24, no. 2, pp. 283–292, 2008.

[22] L. E. Parker, "Path planning and motion coordination in multiple mobile robot teams". Encyclopedia of complexity and system science, pp. 5783–5800, 2009.

[23] B. Li, H. Liu, D. Xiao, G. Yu, Y. Zhang, "Centralized and optimal motion planning for large-scale agv systems: A generic approach". Advances in Engineering Software, vol. 106, pp. 33–46, 2017.

[24] Y. Chen, L. Li, "Collision-free trajectory planning for dual-robot systems using B-splines". International Journal of Advanced Robotic Systems, vol. 14, no. 4, 2017.

[25] G. Sanchez, J.-C. Latombe, "Using a PRM planner to compare centralized and decoupled planning for multi-robot systems". 2002 IEEE International Conference on Robotics and Automation,vol. 2, pp. 2112–2119, 2002.

[26] J. López, D. Pérez, E. Zalama,, "A framework for building mobile single and multi-robot applications". Robotics and Autonomous Systems, vol. 59, no.(3-4), pp.151-162, 2011.

[27] J. Verma, R. Virender, "Multi-Robot Coordination Analysis, Taxonomy, Challenges and Future Scope". Journal of Intelligent & Robotic Systems vol. 102, no.1, pp.1-36, 2021.

[28] Y. Hirata, K. Kosuge, H. Asama, H. Kaetsu, K. Kawabata,"Decentralized control of mobile robots in coordination". 1999 IEEE International Conference on Control Applications, Vol. 2,pp. 1129-1134, 1999.

[29] B. Gerkey,J. Mataric,"Are (explicit) multi-robot coordination and multi-agent coordination really so different".*spring symposium on bridging the multi-agent and multi-robotic research gap*, pp. 1-3. 2004.

[30] G. Korsah, A. Stentz, M. Dias, "A comprehensive taxonomy for multi-robot task allocation". The International Journal of Robotics Research, vol. 32, no.12, pp. 1495–1512, 2013.

[31] T. Standley "Finding optimal solutions to cooperative pathfinding problems". Twenty-Fourth AAAI Conference on Artificial Intelligence, vol. 24, no. 1, pp. 173–178, 2010.

[32] T. Standley, R. Korf, "Complete algorithms for cooperative pathfinding problems". IJCAI International Joint Conference on Artificial Intelligence, pp. 668–673, 2011.

[33] G. Wagner, H. Choset, "Subdimensional expansion for multirobot path planning". Artificial Intelligence,vol. 219, pp. 1–24 2015.

[34] N. Dung, A. Shimada, "Humanoid climbing robot modeling in matlab-simechanics". IECON 2013-39th Annual Conference of the IEEE, pp. 4162–4167, 2013.

[35] M. Silva, R. Barbosa, T. Castro, "Multi-legged walking robot modelling in MATLAB/Simmechanics TM and its simulation".Eighth EUROSIM Congress on Modelling and Simulation. pp. 226–231, 2013.

[36] R. Ismail, M. Ariyanto, W. Caesarendra, A. Nurmiranto, "Development of robotic hand integrated with simmechanics 3D animation," International Seminar on Intelligent Technology and Its Applications (ISITIA), pp. 629–634, 2016.

[37] M. D. Murphy, "A modular virtual laboratory for quadrotor control simulation". IFAC-PapersOnLine, vol. 49, no. 6, pp. 93–98, 2016.

[38] W. Li, T. Zhang, K. Kühnlenz, "A vision-guided autonomous quadrotor in an air-ground multi-robot system". 2011 IEEE International Conference on Robotics and Automation, pp. 2980–2985, 2011.

[39] S. Minaeian, J. Liu, Y.-J. Son, "Vision-based target detection and localization via a team of cooperative UAV and UGVs".

IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 46, no. 7, pp. 1005–1016, 2016.

[40] Y. Bi, H. Duan, "Implementation of autonomous visual tracking and landing for a low-cost quadrotor". Optik - International Journal for Light and Electron Optics, vol. 124, no. 18, pp. 3296–3300, 2013.

[41] K. Wenzel, A. Masselli, A. Zell, "Automatic take off, tracking and landing of a miniature UAV on a moving carrier vehicle". Journal of Intelligent Robotic Systems, vol. 61, no. 1, pp. 221–238, 2011.

[42] J. Higgins, N. Bezzo, "Negotiating visibility for safe autonomous navigation in occluding and uncertain environments". IEEE Robotics and Automation Letters, vol. 6, no. 3, pp. 4409–4416, 2021.

[43] E. Yel, N. Bezzo, "Gp-based runtime planning, learning, and recovery for safe UAV operations under unforeseen disturbances". 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020, pp. 2173–2180.

[44] C. Hui, C. Yousheng, L. Xiaokun, W. Shing, "Autonomous takeoff, tracking and landing of a UAV on a moving UGV using onboard monocular vision". 32nd Chinese Control Conference, pp. 5895–5901, 2013.

[45] J. Ni, M. Tang, Y. Chen, W. Cao, "An improved cooperative control method for hybrid unmanned aerial-ground system in multitasks". International Journal of Aerospace Engineering, vol. 2020, pp. 1-14, 2020.

[46] MF Lee, SF. Su, JW. Yeah, HM. Huang, J. Chen, "Autonomous landing system for aerial mobile robot cooperation". 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS), pp. 1306–1311, 2014.

[47] EH. Harik, F. Guerin , F. Guinand, JF. Brethé , H. Pelvillain, JY. Parédé , "Fuzzy logic controller for predictive vision-based target tracking with an unmanned aerial vehicle". Advanced Robotics, vol. 31, no. 7, pp. 368–381, 2017.

[48] A. Sawarkar, V. Chaudhari, R. Chavan, V. Zope, A. Budale, F. Kazi, "HMD vision-based teleoperating UGV and UAV for hostile environment using deep learning". CoRR, vol. abs/1609.04147, 2016.

[49] D. Stolfi, M. Brust, G. Danoy, P. Bouvry, "Uav-ugvumv multi-swarms for cooperative surveillance". Frontiers in Robotics and AI, vol. 8, p. 5, 2021.

[50] Yiping Li , "Design of Path Tracking Control System for UAV Based on Adaptive Preview Method". Jordan Journal of Mechanical and Industrial Engineering, vol.14, no.1, pp. 101 – 108, 2020.

[51] W. Dixon, D.Dawson, E. Zergeroglu, A. Behal. Nonlinear control of wheeled mobile robots, 2001.

[52] LC. Lai, CC. Yang, CJ. Wu, "Time-optimal control of a hovering quad-rotor helicopter". Journal of Intelligent and Robotic Systems, vol. 45, no. 2, pp. 115–135, 2006.

[53] P. Mukherjee and S. Waslander, "Direct adaptive feedback linearization for quadrotor control," in AIAA Guidance, Navigation, and Control Conference, 2012, p. 4917.

[54] M. Santos, V. Lopez, and F. Morata, "Intelligent fuzzy controller of a quadrotor".International Conference on Intelligent Systems and Knowledge Engineering (ISKE), pp. 141–146, 2010.

[55] T. Lee, F. Leung, P. Tam, "Position control for wheeled mobile robots using a fuzzy logic controller". Industrial Electronics Society, IECON'99 Proceedings. The 25th Annual Conference of the, vol. 2, pp. 525–528, 1999.