



Autopilot design of a class of miniature autonomous blimps enabled by switched controllers

Sungjin Cho¹ · Qiuyang Tao² · Paul Varnell² · Sean Maxon² · Fumin Zhang²

Received: 21 July 2020 / Accepted: 19 February 2022 / Published online: 4 April 2022
© The Author(s), under exclusive licence to Springer Nature Singapore Pte Ltd. 2022

Abstract

The Georgia Tech Miniature Autonomous Blimp (GT-MAB) is developed as an indoor flying robot for education and research. The GT-MAB features extended flight duration and safety for human–robot interaction. The influence of aerodynamics on the blimp is significant and must be considered to achieve autonomous flying. This paper presents the mathematical modeling, system identification, and a switched controller design approach to achieve waypoint and line following behaviors for the GT-MAB. The switched controllers incorporate a scheduling algorithm to handle uncertainties of propeller efficiency and asymmetric envelope. Experiments have demonstrated that the proposed controllers enable flights that collect light measurements in a lab environment. The collected data is used to establish a 3D map of light intensity in a closed space.

Keywords Autonomous blimp · Autopilot · Indoor light field mapping

1 Introduction

Over the past two decades, autonomous robots have assisted scientists to collect data from environmental processes (Dunbabin and Marques 2012; Zhang et al. 2015). Large-scale environmental mapping has been underscored, such as crude-oil concentration maps from oil spill survey (Mukhopadhyay et al. 2014) and temperature field maps for hurricane observation (Glenn et al. 2016). Mapping indoor environments has received less attention than mapping outdoor environments due to challenges associated with developing flying miniature autonomous robots. Small unmanned aerial vehicles such as quad-rotors and multi-copters have become popular for this purpose. However, the indoor applications of these unmanned aerial vehicles (UAVs) are constrained by a number of factors. UAVs usually have limited flight durations per battery charge, typically less than 20 min, which restricts the duration of missions. Some UAVs have large spinning propellers and some fly at relatively high speeds. This can cause safety concerns for humans sharing the same

space. Safety nets or cages are usually installed for human protection, which limits the potential for human–robot interaction experiments.

We develop the Georgia Tech Miniature Autonomous Blimp (GT-MAB) as small flying vehicles for indoor experiments that can support research on mobile sensor networks, human–robot interaction, 3D motion control, networked robotics, and other aspects of autonomy. The GT-MAB has relatively long flight durations of up to 2 h per battery charge. Furthermore, the blimps are naturally cushioned and do not cause any harm if they collide with a human. It offers a fun experience that encourages physical contact, as illustrated in Fig. 1. With its small size, low cost, and safe operation, the GT-MAB also serves to educate and excite children and students about robot design and control.

A significant body of literature exists for both outdoor and indoor and robotic blimps. Earlier developments are similar to airships (Elfes et al. 1998; Kantor et al. 2001). Indoor blimps have been previously developed for entertainment (Burri et al. 2013), artistic performance in museum (St-Onge et al. 2015), telepresence (Paulos and Canny 1998), and for the emulation of underwater vehicles (Zwann et al. 2000). A class of small blimps were previously designed for robotics research on indoor localization and mapping (Müller and Burgard 2013). The advantages of blimps over other aerial vehicles, including reduced energy consumption, lower cost, and better safety, have been noted in the literature reviewed.

✉ Fumin Zhang
fumin@gatech.edu

¹ Department of Guidance and Control, Agency for Defense Development, Daejeon 34186, South Korea

² School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA



Fig. 1 Safe interaction between children and the GT-MAB

The GT-MAB is a unique design in several perspectives. The blimp envelop has an ellipsoidal shape with the lengths of the semi-major and semi-minor axes as 0.36 and 0.23 m, respectively, which is smaller than the 0.5 m diameter design in Zufferey et al. (2006). Being small allows multiple blimps to be flown at the same time to support indoor experiments on swarming and sensor networks. But being small also causes limited load capacity, which has motivated us to design small and lightweight driving and sensing hardware systems. Recent advancements in electronics, computing, and MEMS allow us to achieve this goal. The GT-MAB is also highly maneuverable. Most previous designs (Zwann et al. 2000; Müller and Burgard 2013; Zhang and Ostrowski 1999; Gonzalez et al. 2009) inherited the envelop design of airships and employ longitudinal tail fins for maneuvering. This design favors stable forward cruise motion for long-range flights, but it is not the optimal choice for maneuverability in an indoor lab setting. The GT-MAB uses a “saucer-shaped” envelope that makes turning motions easier. Multiple propellers can provide vector thrusts that achieve better maneuverability in indoor environments, see Fig. 1.

The emphasis on maneuverability and low weight poses challenges for achieving autonomous flight control that overcomes the significant influence from aerodynamics in the face of unknown model parameters. We first model the aerodynamics associated with the envelop, and then identify the model parameters. The dynamic model of the GT-MAB differs from previous works on airship control (Gomes and Ramos 1998; Kim et al. 2003; do Valle et al. 2015). Our design of autopilot uses proportional-integral-derivative (PID) controllers that are tuned based on the identified models to achieve stable flights in three-dimensional space. In particular, altitude and forward-backward controllers incorporating a scheduling algorithm are designed to deal with uncertainties of propeller efficiency, aerodynamics, and

ambient noise. The autopilot allows the GT-MAB to follow a planned path in a 3D environmental sensing experiment. Since environmental sensing applications have been a major motivation for the development of outdoor robotic blimps, we demonstrate using the GT-MAB to sample a 3D light intensity field indoor. This scenario emulates outdoor surveying missions frequently encountered in practice (Mukhopadhyay et al. 2014; Elfes et al. 1998; Kantor et al. 2001), which require long flight duration and collection of a large number of measurements.

This paper extends our previous conference paper (Cho et al. 2017) where preliminary work on the GT-MAB modeling, identification and autopilot design was first introduced. We introduce a station keeping controller and provide more details on the switched controller designs for the forward/backward motion and the up/down motion. These controllers are different from the ones in Cho et al. (2017). We have also implemented a line following behavior for indoor light field mapping missions using the combination of these controllers, which was not achieved in Cho et al. (2017). Comparing to the waypoint based controller and the spiraling sampling path in Cho et al. (2017), the line following behavior is able to collect denser measurements of light intensity in the lab space, and provide more accurate reconstruction of the light intensity field.

Indoor light field mapping have been previously investigated for computer graphics (Levoy and Hanrahan 1996; Levoy 2006). Since 3D representation of objects from 2D images is altered according to the light fields (Debevec et al. 2000), multiple cameras are used to measure the 3D light field. Generally, it is difficult to estimate light fields in a closed indoor space in that light rays are reflected on the walls and their intensities are attenuated, which leads to complex spatial variations of light fields. The GT-MAB is able to navigate the indoor space so that light intensity can be accurately collected along its trajectory. This light field mapping method may have an application in optical wireless communication (Komine and Nakagawa 2004) where it is critical to measure light intensity in 3-D space to predict performance of optical communication.

The remainder of this paper has been organized into the following sections: in Sects. 1 and 2, we present design, modeling, and control of the GT-MAB. In Sect. 3, we demonstrate the application of the GT-MAB to 3D environmental sampling and mapping. In Sect. 4, we provide conclusions and future work.

2 Physical design

The physical design of the GT-MAB reflects a balance among design challenges such as stability of the structure, limited payload, and maneuverability. The GT-MAB has two

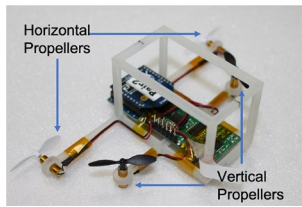


Fig. 2 The gondola of GT-MAB with two horizontal and two vertical propellers

major modules, a “saucer-shaped” envelope and a gondola (see Fig. 2). The center of gravity is below the center of buoyancy, which facilitates dampening of pitching and rolling motions and hence increases the stability of operation. The symmetric shape of the envelop allows easy spinning in place, which increases the maneuverability.

Figure 2 shows a gondola that is attached to the bottom of the envelope near the center. The gondola is a 3D printed housing for the control and sensing hardware including motors, propellers, a microprocessor, an inertial measurement unit (IMU), and other sensors. The limited lift provided by the envelope imposes a constraint on the total weight of the gondola. This constraint was satisfied by a careful selection of electronic components and a light-weight design of the mechanical structure. The total weight is 85.9 g, and offers 12.1 g for payload.

The motion of the GT-MAB is generated by four motors connected to small propellers that are mounted to the gondola. This particular configuration of the four motors gives the blimp high maneuverability. The two motors, mounted facing upwards near the center-line, provide counter torque to each other and stabilize the upward and downward motion of the blimp. The other two motors facing the head of the GT-MAB provide counter torque to stabilize the horizontal motion of the blimp and can also be used to provide differential thrusts for quick turning motion.

2.1 Blimp dynamics

We follow the established conventions in the literature (Zwann et al. 2000; Gomes and Ramos 1998) to derive the dynamic modeling equations of the GT-MAB. The blimp has six degree of freedom (6-DOF). The position of the blimp is determined in a world-fixed inertial coordinate frame. But it is more convenient to represent the linear and angular velocities in a body-fixed coordinate frame; see Fig. 3 for the frame convention. The control commands are supplied to an on-board thrusters which apply thrusts in the body-fixed frame, but the position and orientation of the blimp are described with respect to the inertial coordinate frame.

Let Euler angles ϕ, θ, ψ be the roll, pitch, and yaw angles respectively and x, y, z be the 3-D position with respect to

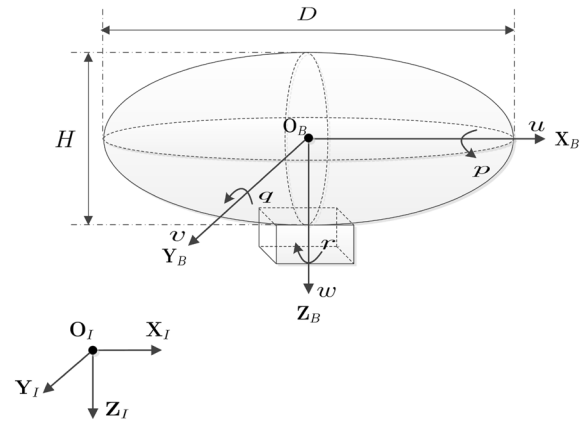


Fig. 3 Inertial and body coordinate frames. $O_B - X_B Y_B Z_B$ represents the body-fixed axis of the blimp. $O_I - X_I Y_I Z_I$ represents the inertial axis of the blimp. The origin O_B is the center of the buoyancy (CB) of the blimp. The center of buoyancy (CB) is collocated at the center of volume of the envelop

the inertial frame. Let $\eta^B = [u, v, w]^T$, $\eta^I = [\dot{x}, \dot{y}, \dot{z}]^T$ be the linear velocity vectors in the body-fixed frame and the inertial frame respectively. The linear velocity vector can be transformed to the body-fixed frame from the inertial frame using $\eta^B = C_B^I \eta^I$, where $C_B^I = C_\phi C_\theta C_\psi$, and C_ϕ, C_θ, C_ψ are the rotation matrices corresponding to roll, pitch and yaw respectively where all the rotation matrices are defined in Fossen (1993) (see equation (2.18)). Let $\omega = [p, q, r]^T$ be the angular velocity of the blimp in the body-fixed frame such that $\omega = J[\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$ where J is defined in Fossen (1993) (see equation (2.14)).

Let $\mathbf{f} = [f_x, f_y, f_z]^T$ represent the translational forces generated by the propellers of the blimp and $\boldsymbol{\tau} = [\tau_x, \tau_y, \tau_z]^T$ be the rotational torque generated by the propellers. Let $\mathbf{F} = [F_x, F_y, F_z]^T$ be a vector that represents the gravity forces, the buoyancy forces, and other aerodynamic forces acting on the blimp in the body-fixed frame. Let $\mathbf{M} = [M_x, M_y, M_z]^T$ be a vector that represents all the external moments exerted on the blimp except for those generated by the propellers. Let m be the mass of the blimp and

$$\mathbf{I} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & -I_{yz} & I_{zz} \end{bmatrix}$$

be the moment of inertia about the origin of the body-fixed frame. Then the dynamic equations of motion for the blimp can be derived as

$$m(\dot{\eta}^B + \omega \times \eta^B) = \mathbf{F} + \mathbf{f} \quad (1)$$

$$\mathbf{I}\dot{\omega} + \omega \times (\mathbf{I}\omega) = \mathbf{M} + \boldsymbol{\tau}. \quad (2)$$

For the GT-MAB design, the height H of the blimp is 0.46 m and the diameter D is 0.72 m. The translational forces generated by the propellers of the blimp is represented by $\mathbf{f} = [f_x, 0, f_z]^T$ because there is no sideways force in the body-fixed frame. The turning torque generated by the propellers is represented by $\boldsymbol{\tau} = [0, 0, \tau_z]$ which only controls the yaw moment.

2.2 Motion primitives

The 6-DOF model described by Eqs. (1) and (2) is nonlinear and coupled. A general controller design for such system is difficult, especially when some of the parameters of the model are unknown. Therefore, we need to find simplified models to achieve controllable flight. The GT-MAB is designed mainly to achieve four *motion primitives*:

1. Maintaining speed. The blimp should be able to maintain a desired constant speed along its X_B direction while having zero vertical speed, and zero yaw angular speed e.g., $u = u_0$, $w = 0$, and $r = 0$.
2. Station keeping. The blimp should be able to move forward or backward in order to move along its X_B direction to stay at a position while keeping the velocities close to zero e.g., $\eta^B \approx 0$.
3. Changing altitude. The blimp should be able to ascend or descend to a desired height while maintaining zero forward speed and zero yaw angular speed e.g., $z = z_0$, $u = 0$ and $r = 0$.
4. Changing orientation. The blimp should be able to spin in place so that its yaw angle can be stabilized at any desired value while maintaining zero forward speed, and zero vertical speed e.g., $\psi = \psi_0$, $u = 0$ and $w = 0$.

These motion primitives can be combined to achieve stable flights that allow the blimp to move in 3D space. The design goals for the autopilot are now reduced to stabilizing these four motion primitives.

In order to simplify the dynamics, we make the following practical assumption that holds for all four motion primitives.

Assumption 1 The roll angle and the roll angular velocity (ϕ, p), the pitch angle and the pitch angular velocity (θ, q), and the side-slipping velocity (v) are negligibly small during the transient phase of the flight and zero during the steady state flight.

Remark 1 The assumption is justified because the GT-MAB is under the influence from the restoring forces due to its bottom-heavy design. The restoring forces induced by gravity and buoyancy effectively damp out roll and pitch motion. Since the blimp is very light with a large envelop,

the side-slipping velocity of the blimp will vanish quickly due to air drag when the blimp flies forward. We understand that a sideways force will be generated by the term $\boldsymbol{\omega} \times \boldsymbol{\eta}^B$ while the blimp is spinning and flying forward at the same time. However, this term is viewed as a vanishing disturbance force that is damped out by air drag. The force can also be ignored under the assumption that the spinning speed is almost zero during forward flight, and the forward speed is almost zero during the spinning motion.

We are then able to separate the longitudinal and the spinning motion from the full nonlinear motion model and obtain three dynamic equations

$$m\dot{u} = F_x + f_x \quad (3)$$

$$m\dot{w} = F_z + f_z \quad (4)$$

$$I_{zz}\dot{r} = M_z + \tau_z. \quad (5)$$

Equations (3) and (4) describe the motion of the blimp in the $X_B - Z_B$ plane, assuming no sideways motion and ignoring the sideways forces. Equation (5) represents the spinning motion of the blimp around the Z_B axis.

Remark 2 Due to the symmetry of the blimp envelop, the inertia matrix \mathbf{I} becomes diagonal. One advantage of the “saucer-shaped” envelop is that $I_{xx} = I_{yy}$, which further simplifies the dynamics. The term $\boldsymbol{\omega} \times I\boldsymbol{\omega}$ does not generate rotation moments in Eq. (5) if the roll and pitch angular velocities are zero. In the case when roll and pitch moments are not zero, the contribution from the term $\boldsymbol{\omega} \times I\boldsymbol{\omega}$ to the roll and pitch moments are also small allowing them to be easily damped out by the restoring force from the gravity.

Remark 3 Note that the models (3–5) are still nonlinear because the external forces F_x, F_z and the external moment M_z include the forces and moments generated by the gravity and the ambient air. These forces are nonlinear functions of accelerations and velocities that are quite difficult to model. This is quite different from quad-rotors and multi-copters with powerful thrusters that generate forces f_x, f_z and torque τ_z at least a magnitude larger than the influences from ambient air, which allow F_x, F_z and M_z to be ignored. For the blimp, because of the relatively large envelop and the relatively weak thrusters, the aerodynamics need to be considered, which is very similar to the modeling of underwater vehicles.

2.3 System identification

The simplified models can be further linearized for each motion primitive.

- For flight at constant heading, we linearize the model around a desired forward speed, zero vertical speed, and zero yaw angular speed. The transfer function $P_1(s)$ for maintaining forward speed has f_x as its input and the forward speed u as its output. Forward motion is generated from positive f_x ; backward motion is generated from negative f_x .
- For a change of forward or backward position, we linearize the model around the desired position along a forward or backward direction, zero vertical speed and zero yaw angular speed. The transfer function $P_2(s)$ is for maintaining a position through low speed forward or backward motion. It has f_x as its input and the x as its output.
- For a change of altitude, we linearize the model around the desired height, zero forward speed and zero yaw angular speed. The transfer function $P_3(s)$ for changing altitude has f_z as its input and the z as its output. Since z is positive downward, the height of the blimp is negative.
- For a change of orientation, we linearize the model around a desired yaw angle, zero forward speed, and zero vertical speed. These models can be viewed as open-loop plants. The transfer function $P_4(s)$ for changing orientation has the torque τ_z as its input and the yaw angle ψ as its output.

A set of experiments have been performed where both input and output of the models are measured. Then the transfer functions $P_1(s) - P_4(s)$ are identified using the MATLAB system identification toolbox Ljung and Singh (2012) as follows:

$$P_1(s) = \frac{b_2 s^2 + b_1 s + b_0}{s^3 + a_2 s^2 + a_1 s + a_0} e^{-0.17s} \quad (6)$$

$$P_2(s) = \begin{cases} \frac{b_0}{a_2 s^2 + a_1 s} e^{-0.17s} & \text{if } f_x \leq 0 \\ \frac{b_0}{a_2 s^2 + a_1 s} e^{-0.17s} & \text{if } f_x > 0 \end{cases} \quad (7)$$

$$P_3(s) = \begin{cases} \frac{b_0}{a_2 s^2 + a_1 s} e^{-0.17s} & \text{if } f_z \leq 0 \\ \frac{b_0}{a_2 s^2 + a_1 s} e^{-0.17s} & \text{if } f_z > 0 \end{cases} \quad (8)$$

Table 1 Coefficients of identified transfer functions

Transfer functions	a_0	a_1	a_2	b_0	b_1	b_2
$P_1(s)$	4.69	14.62	0.76	12.26	0.58	0.96
$P_2(s), (f_x \leq 0)$	0	0.46	1	0.67	0	0
$P_2(s), (f_x > 0)$	0	0.30	1	0.82	0	0
$P_3(s), (f_z \leq 0)$	0	0.13	1	1.95	0	0
$P_3(s), (f_z > 0)$	0	0.02	1	0.93	0	0
$P_4(s)$	0	0.19	1	10.94	0	0

$$P_4(s) = \frac{b_0}{a_2 s^2 + a_1 s} e^{-0.17s}, \quad (9)$$

where all the coefficients of the identified transfer functions are listed in Table 1.

We notice that these models are greater than second order, most likely because they incorporate the parasitic effects caused by the non-ideal values of the state variables that are previously ignored. We then use the first order Pade approximation to approximate the time delays in the transfer functions. The Pade approximation $e^{-\tau s} = \frac{1}{\tau s + 1}$ (Nise 1995) leads to approximated open loop transfer functions as follows:

$$P_1(s) = \frac{b_2 s^2 + b_1 s + b_0}{s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}. \quad (10)$$

$$P_2(s) = \begin{cases} \frac{b_0}{a_3 s^3 + a_2 s^2 + a_1 s + a_0} & \text{if } f_x \leq 0 \\ \frac{b_0}{a_3 s^3 + a_2 s^2 + a_1 s + a_0} & \text{if } f_x > 0 \end{cases} \quad (11)$$

$$P_3(s) = \begin{cases} \frac{b_0}{a_3 s^3 + a_2 s^2 + a_1 s + a_0} & \text{if } f_z \leq 0 \\ \frac{b_0}{a_3 s^3 + a_2 s^2 + a_1 s + a_0} & \text{if } f_z > 0 \end{cases} \quad (12)$$

$$P_4(s) = \frac{b_0}{a_3 s^3 + a_2 s^2 + a_1 s + a_0}, \quad (13)$$

where all the coefficients of the identified transfer functions are listed in Table 2.

The locations of open loop poles and zeros of the transfer function $P_1(s)$ are plotted in blue in Fig. 4. For transfer functions $P_1(s)$ and $P_4(s)$, there are no poles on the right half of the complex plane, but there are poles on the imaginary axis or the origin. This implies that forward speed and orientation systems are marginally stable. The forward and backward position transfer functions $P_2(s)$ have similar pole characteristics; however, their numerators are different. The altitude transfer function $P_3(s)$ has different pole characteristics depending on input. When input f_z is non-positive, one pole

Table 2 Coefficients of identified transfer functions with approximation

Transfer functions	a_0	a_1	a_2	a_3	b_0	b_1	b_2
$P_1(s)$	27.59	90.71	19.11	6.65	72.12	3.40	5.66
$P_2(s), (f_x \leq 0)$	0	2.73	6.35	1	3.96	0	0
$P_2(s), (f_x > 0)$	0	1.77	6.18	1	4.84	0	0
$P_3(s), (f_z \leq 0)$	0	0.75	6.01	1	11.49	0	0
$P_3(s), (f_z > 0)$	0	0.10	5.9	1	5.50	0	0
$P_4(s)$	0	1.09	6.07	1	64.33	0	0

is located at the origin; however, double poles are located at the origin when input f_z is positive. It shows that the altitude system is marginally stable under non-positive input, and unstable under positive input. Therefore, the desired speed, forward or backward position, height, and yaw angle can not be achieved by open loop control.

In particular, the forward speed model $P_1(s)$ has one pair of complex conjugate poles very close to the imaginary axis, which are generated by the coupling between the forward motion and pitch oscillation that are ignored when deriving the theoretical model. These poles will lead to very slowly vanishing oscillatory modes in the pitch angle when the blimp flies forward.

The forward and backward position models $P_2(s)$ switches according to input sign. Although the symmetric envelope makes the forward and backward motion dynamics similar, there are differences between the numerators caused by the different slopes of thrust. The slope of backward thrust ($f_x \leq 0$) is smaller than that of the forward thrust ($f_x > 0$). Since propeller blades are mainly designed for generating forward thrust, reversal rotation of propellers produces lower backward thrust than forward thrust.

The altitude model $P_3(s)$ also switches according to the input sign. The altitude dynamics is different for ascending and descending motion due to propeller efficiency. When we compare numerators of transfer function $P_3(s)$, the slope of upward thrust ($f_z \leq 0$) is two times larger than that of downward thrust ($f_z > 0$). Comparing the pole positions of transfer function $P_3(s)$ also shows asymmetric motion vertically. When the GT-MAB goes up, the upward motion is marginally stable in that the gondola located at the bottom of the envelope plays a role as a stabilizer aerodynamically. However, when the GT-MAB goes down, the downward motion is unstable because the envelope blocks the airflow

Table 3 Speed and yaw PID controller gains

Controllers	P	I	D
Speed	1.095	1.095	0
Yaw	0.1955	0	0.192

Table 4 Transient performances of the speed controller and the yaw controller

Controllers	Rising time (s)	Settling time (s)	Overshoot (%)
Speed	1.3	8.0	14.8
Yaw	1.2	3.2	6.2

generated by the propellers. Thus, the upward and the downward motions are not symmetric.

3 Controller designs

3.1 Speed and heading controller designs

Based on the identified linear input-output speed and heading models represented by $P_1(s)$ in Eq. (10) and $P_4(s)$ in Eq. (13), we design PID controllers to achieve two motion primitives: maintaining speed and changing orientation. The goal is to make the two closed-loop systems asymptotically stable, and to compensate for the oscillations in the speed system. The PID gains for the two controllers are tuned in MATLAB based on the open-loop transfer functions $P_1(s)$ and $P_4(s)$ identified. Tables 3 and 4 show the gains and their transient performances, respectively.

The two closed-loop transfer functions under the PID controllers are:

$$G_1(s) = \frac{6.1988s^3 + 9.9265s^2 + 82.6865s + 78.9594}{s^5 + 6.6459s^4 + 25.3141s^3 + 100.6424s^2 + 110.2788s + 78.9594} \quad (14)$$

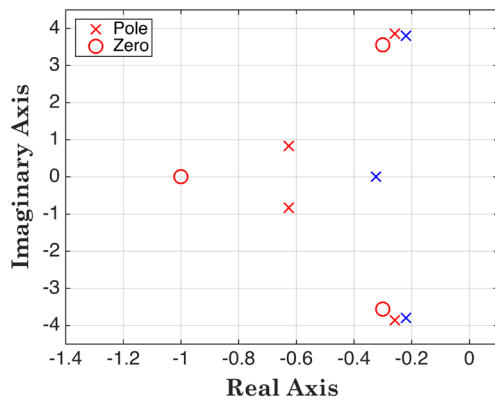


Fig. 4 Poles and zeros of the open loop transfer function $P_1(s)$ and the closed-loop transfer function $G_1(s)$ for forward speed control. Blue markers represent poles and zeros of the open-loop transfer function. Red markers represent poles and zeros of the closed-loop transfer function. One open-loop pole located at $(-5.88, 0)$ and one closed-loop pole located at $(-4.88, 0)$ do not appear in the figure (color figure online)

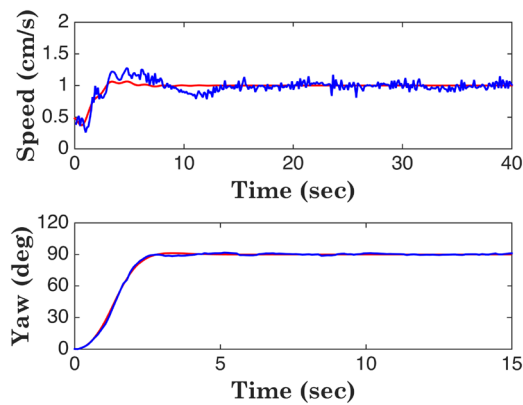


Fig. 5 Simulated step response (red) and measured step response (blue) for two motion primitives of the GT-MAB (color figure online)

$$G_4(s) = \frac{12.5771}{s^3 + 6.0676s^2 + 13.4429s + 12.5771}, \quad (15)$$

where $G_1(s)$ is closed-loop transfer function of speed, and $G_4(s)$ is closed-loop transfer function of yaw angle.

For the two closed-loop transfer functions, all poles are on the left half plane, hence all the two closed-loop transfer functions are asymptotically stable. Furthermore, there is no steady-state error under step function input. The poles and zeros of the closed-loop transfer function $G_1(s)$ are plotted

in red in Fig. 4. The oscillatory complex poles in $P_1(s)$ are compensated by a pair of complex conjugate zeros, which significantly reduce the oscillation. Note that we design the PID gain of the speed controller for the blimp with the desired speed 1 cm/s and apply the gains successfully for speed range from 1 cm/s to 3 cm/s .

Figure 5 shows the comparison between the simulated step response and the measured step response of the GT-MAB under each controller. It can be seen that the heading controller (bottom-graph) performs similarly to the simulated response and has a good rise and settling time. However, we still observe small oscillations in the speed controller (top graph). Due to lack of direct control input for the pitch motion, the linearized system is stabilizable but not controllable with respect to the pitch angle.

3.2 Scheduling algorithm for multiple PID controllers

We design a scheduling algorithm to accomplish motion primitives for changing altitude and horizontal position. Each open-loop plant, represented by Eq. (7) or (8), has two different dynamics according to the sign of control input. A single PID controller is not capable of controlling the two different dynamics. Hence we design two PID controllers to satisfy the stability and performance specified for the closed-loop systems. We design a scheduling algorithm that switches between the two controllers, which is presented as Algorithm 1.

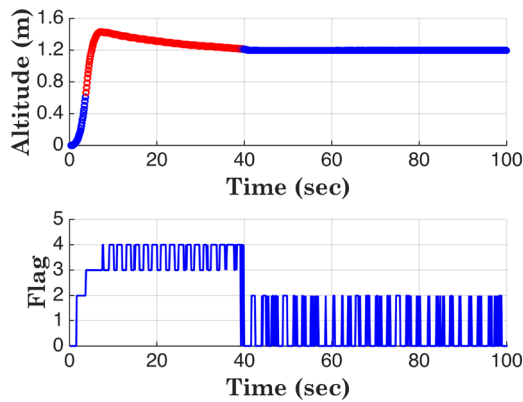
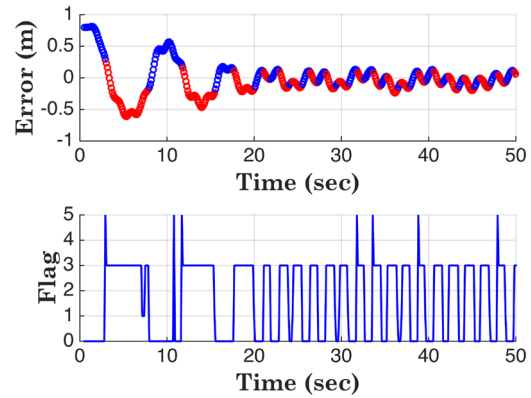
Consider the generic case where two PID controllers, #1 and #2 are to be scheduled. Let f_1 be the control force computed using #1 and f_2 be the control force computed using #2. Controller #1 is designed for the open-loop plant with non-positive f_1 while controller #2 is designed for open-loop plant with positive f_2 . Hence we can compute both controllers at the same time. If both f_1 and f_2 are non-positive, then controller #1 is chosen. Otherwise, if both f_1 and f_2 are positive, then controller #2 is chosen. There are two cases left when f_1 and f_2 have opposite signs. To make the correct choice, we will need to use the error signal e which is the difference between the desired output of the system and the current output value under control. A positive error indicates that the current output is less than desired value, hence controller #1 needs to be selected. A non-positive error indicates that the current value is greater than the desired value, hence controller #2 needs to be selected. A flag is introduced to mark all the cases in the Algorithm 1 according to which a controller is selected.

Table 5 Altitude PID controller gains

Controllers	P	I	D
# 1	0.6560	0.0087	0.7352
# 2	0.146	0.0027	1.955

Table 6 Position PID controller gains

Controllers	P	I	D
# 1	0.0749	0.0002	0.0615
# 2	0.0915	0.0004	0.0491

**Fig. 6** Measured step response for changing altitude (top) and switching modes (bottom): upward controller (blue) and downward controller (red) (color figure online)**Fig. 7** Measured step response for changing position (top) and switching modes (bottom): forward controller (blue) and backward controller (red) (color figure online)**Algorithm 1:** Scheduling Algorithm for two PID Controllers**Input:** #1 PID controller output f_1 , #2 PID controller output f_2 , error e **Output:** Control input f and switching mode flag

```

1 if  $f_1 \leq 0$  &  $f_2 \leq 0$  then
2    $f = f_1$ , flag = 0
3 else if  $f_1 \leq 0$  &  $f_2 > 0$  then
4   if  $e \leq 0$ 
5      $f = f_2$ , flag = 1
6   else
7      $f = f_1$ , flag = 2
8   end
9 else if  $f_1 > 0$  &  $f_2 > 0$  then
10   $f = f_2$ , flag = 3
11 else if  $f_1 > 0$  &  $f_2 \leq 0$  then
12  if  $e \leq 0$ 
13     $f = f_2$ , flag = 4
14  else
15     $f = f_1$ , flag = 5
16  end
17 else
18   break;
19 end

```

Note that the scheduling algorithm can be generalized to more than two PID controllers according to the levels of error and the desired control effort. For example, one can introduce another two controllers to handle the cases when the error is large. This will increase the complexity of the scheduling algorithm while improving performance.

3.3 Altitude controller

The scheduling algorithm is applied to the altitude controller to accomplish motion primitive for height control. Controller #1 is designed with the transfer function $P_3(s)$ associated with upward thrust ($f_z \leq 0$). Controller #2 is designed with the transfer function $P_3(s)$ associated with downward thrust ($f_z > 0$). The output of the two controllers are $f_z = f_1$ and

$f_z = f_2$ respectively so that Algorithm 1 applies. The PID gains are tuned in MATLAB and shown in Table 5. Figure 6 shows measured step response of altitude (top) and corresponding time-series switching modes (bottom) while Algorithm 1 is running. Transient altitude response settles down after 40 s by the altitude controller incorporating the scheduling algorithm. In the bottom panel of Fig. 6, flags 0 and 2 represent upward thrust that reduces positive altitude error. Flags 3 and 4 represent downward thrust that reduces overshoot and transient altitude error.

3.4 Station keeping controller

The station keeping controller aims to keep the blimp at a desired location with zero forward/backward speed. It is also designed using the scheduling algorithm. The controller #1 is designed with the transfer function $P_1(s)$ associated with forward thrust ($f_x > 0$). Controller #2 is designed with the transfer function $P_2(s)$ associated with backward thrust ($f_x \leq 0$). The PID gains are tuned in MATLAB and shown in Table 6. The output of the two controllers are $f_x = f_1$ and $f_x = f_2$ respectively so that Algorithm 1 applies. Figure 7 shows measured step response of position (top) and corresponding time-series switching modes (bottom) while Algorithm 1 is running. Transient position response settles down after 20 s by the position controller incorporating the scheduling algorithm. In the bottom panel of Fig. 7, flags 0 and 5 represent forward thrust that reduces positive position error. Flags 1 and 3 represent backward thrust that reduces overshoot and transient position error. Note that abrupt transition between controllers after 30 s is caused by position error oscillating with very small magnitude. It is generated by the oscillatory modes from the coupling of forward motion and pitch oscillation described in Sect. 2.3.

The GT-MAB design discussed in this paper does not offer direction control for sideways motions, hence there will be sideways drift while the blimp keeps its station along the forward/backward direction. This leads to a line following behavior where the blimp uses an initial sideways speed to move along a line. This behavior will be leveraged in measuring a light field.

4 Application demonstration: indoor light field mapping

We leverage the controller designs to generate robotic behaviors to measure an indoor light field. Two blue LED's were placed on the floor at coordinates $(-1.22 \text{ m}, 0, 0)$ and $(1.52 \text{ m}, 0, 0)$. The total sampling volume is approximately 4.0 m long, 1.2 m wide, and 1.0 m high in the lab. The blimp will move autonomously in this sampling volume to take light measurements. A 3D light field map is generated based

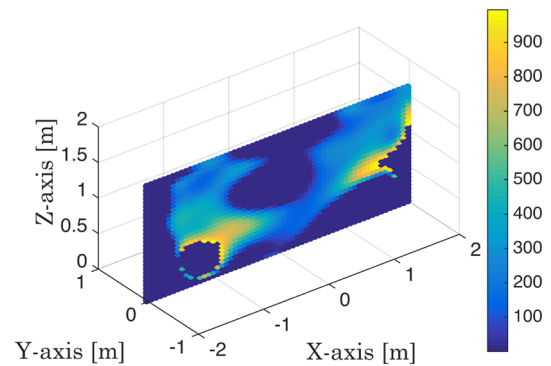


Fig. 8 One slice of the light field map along the X–Z plane when the Y coordinate is zero for field constructed in Cho et al. (2017). The estimated light intensity around the light source is not clearly identified because of low resolution of light measurements collected by waypoint behaviors

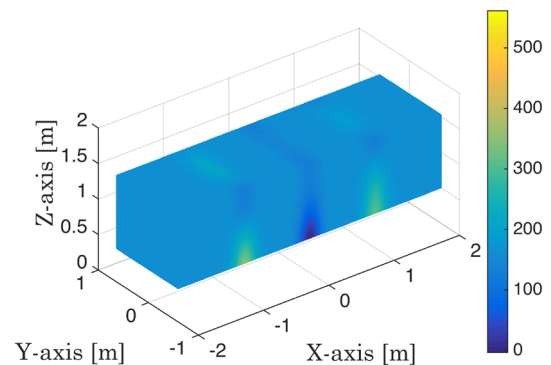


Fig. 9 The 3D light field map constructed from the light data collected by the GT-MAB with the line following behavior around the two light sources

on the collected measurements. Note that for ease of interpretation, the Z coordinates of all measurement locations are converted to positive values, corresponding to height above the ground.

In our previous work Cho et al. (2017), the motion primitives have been combined to implement a waypoint following behavior for the blimp that reaches a location specified as a waypoint in 3D space. The altitude controller controls the blimp to the desired altitude where the waypoint is located, and the heading controller controls the heading of the blimp to always point towards the waypoint. A distance controller will use the distance to the waypoint d as feedback and f_x as control input. The plant model is $\frac{P_1(s)}{s}$. A PID controller is designed so that the blimp will slow down when it gets close to the waypoint. The gains of the distance controller are selected as $P = 0.006271$, $I = 5.236 \times 10^{-5}$ and

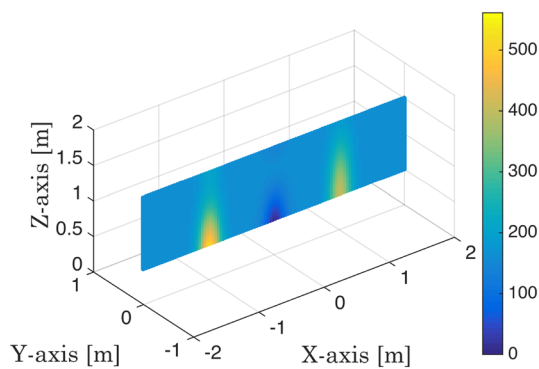


Fig. 10 One slice of the light field map along the X - Z plane when the Y coordinate is zero in Fig. 9: the estimated light intensity around the light source is much stronger than any other spot. There is perceivable improvements over Fig. 8

$D = 0.03288$. The distance, altitude, and heading controller work together to control the blimp that reaches the desired waypoints in space. This waypoint following controller was used in Cho et al. (2017) to follow a spiraling path to measure a 3D light field. The experimental results demonstrate that a 3D light field can be constructed from the data collected. However, one drawback of this strategy is the unevenly distributed sampling of the volume. Figure 8 provides a closer look at one slice of the reconstructed field based on data collected in Cho et al. (2017). We can observe the lack of data in certain locations.

To increase the sampling density, we leverage the station-keeping controller for the blimp developed in this paper. The blimp will follow a set of desired waypoints that are located at a given height along sampling lines that are parallel to the Y -axis. The altitude controller will maintain the desired height, and the heading controller keeps the orientation of the blimp to be aligned with the X -axis. A total of nine sampling paths are specified. Each of the nine lines is parallel to the Y axis, and each passes a unique point that is specified by choosing the X coordinate as one of the three values -1.5 , 0 , and 1.5 , and choosing the Z coordinate as one of the three values 0 , 0.5 , and 1 . For each line, the blimp is deployed with a small initial speed along the Y -axis so that it will drift sideways long the sampling line. Three measurements are taken along each line at the Y coordinates of -0.5 , 0 , and 0.5 , leading to a total of 27 direct light measurements in the volume. Then, a radial basis function-based smoothing scheme is used to produce a greater number of virtual light measurements in the volume. The interpolated 3D map is generated with the Kriging method Pilz and Spöck (2008) that uses those virtual light measurements. The line tracking behavior enables denser sampling in a 3-D volume of the indoor environment by collecting light data on the predetermined lines. Figures 9 and 10 show that the GT-MAB generates the

light field map in the indoor environment. We can observe improved performance of field construction.

5 Conclusions and future work

The Georgia Tech miniature autonomous blimps are designed as a versatile indoor flying platform to support a broad scope of research in robotics. This paper demonstrated that the GT-MAB can achieve autonomous motion primitives using an autopilot based on simple feedback control laws. System identification techniques allow the controllers to be tuned to account for the differences between simplified and actual flight dynamics. The autopilot incorporates a scheduling algorithm in order to handle uncertainties of aerodynamics and propulsion. The autopilot combines the motion primitives to achieve basic waypoint following behaviors and line following behaviors that are used for light field sensing experiments. The collection of motion primitives will be expanded in future work to support other research missions. For a more realistic light field map, we will study the map of the light field affected by various environmental factors such as human intervention and wind disturbance.

Funding Funding was provided by Office of Naval Research (N00014-19-1-2556, N00014-19-1-2266, N00014-16-1-2667), National Science Foundation (OCE-1559475, CNS-1828678, S&AS-1849228), U.S. Naval Research Laboratory (N00173-17-1-G001, N00173-19-P-1412), National Oceanic and Atmospheric Administration (NA16NOS0120028).

References

- Burri, M., Gasser, L., Kach, M., Laube, S., Ledergerber, A.: Design and control of a spherical omnidirectional blimp. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1873–1879 (2013)
- Cho, S., Mishra, V., Tao, Q., Vamell, P., King-Smith, M., Muni, A., Smallwood, W., Zhang, F.: Autopilot design for a class of miniature autonomous blimps. In: 2017 IEEE Conference on Control Technology and Applications (CCTA), pp. 27–30 (2017)
- Debevec, P., Hawkins, T., Tchou, C., Duiker, H.P., Sarokin, W., Sagar, M.: Acquiring the reflectance field of a human face. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Technique, pp. 145–156 (2000)
- do Valle, R.C., Menegaldo, L.L., Simões, A.M.: Smoothly gain-scheduled control of a tri-turbopan airship. *J. Guid. Control. Dyn.* **38**(1), 53–61 (2015)
- Dunbabin, M., Marques, L.: Robotics for environmental monitoring. *IEEE Robot. Autom. Mag.* **19**, 21–24 (2012)
- Elfes, A., Bueno, S.S., Bergerman, M., Ramos, J.J.G.: A semi-autonomous robotic airship for environmental monitoring missions. In: Proceedings of the 1998 IEEE International Conference on Robotics and Automations, pp. 614–619 (1998)

- Fossen, T.I.: Guidance and Control of Ocean Vehicles. Wiley, New York (1993)
- Glenn, S., Miles, T., Seroka, G., Xu, Y., Forney, R., Yu, F., Roarty, H., Schofield, O., Kohut, J.: Stratified coastal ocean interactions with tropical cyclones. *Nat. Commun.* **7**(10887), 1–10 (2016)
- Gomes, S.B.V., Ramos, J.J.G.: Airship dynamic modeling for autonomous operation. In: Proceedings of 1998 IEEE International Conference on Robotics and Automation (ICRA), pp. 3462–3467 (1998)
- Gonzalez, P., Burgard, W., Sanz, R., Fernandez, J.L.: Developing a low-cost autonomous indoor blimp. *J. Phys. Agents* **3**, 43–52 (2009)
- Kantor, G., Wettergreen, D., Ostrowski, J.P., Singh, S.: Collection of environmental data from an airship platform. *Proc. SPIE* **4571**, 76–83 (2001)
- Kim, J., Keller, J., Kumar, R.V.: Design and verification of controllers for airships. In: Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003), vol. 1, pp. 54–60 (2003)
- Komine, T., Nakagawa, M.: Fundamental analysis for visible-light communication system using led lights. *IEEE Trans. Consum. Electron.* **50**(1), 100–107 (2004)
- Levoy, M., Hanrahan, P.: Light field rendering. In: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, pp. 31–42 (1996)
- Levoy, M.: Light fields and computational imaging. *Computer* **39**(8), 46–55 (2006)
- Ljung, L., Singh, R.: Version 8 of the system identification toolbox. In: Proc. 6th IFAC Symposium on System Identification The International Federation of Automatic Control, pp. 1826–1831 (2012)
- Mukhopadhyay, S., Wang, C., Patterson, M., Malisoff, M., Zhang, F.: Collaborative autonomous surveys in marine environments affected by oil spills. In: Koubaa, A., Khelil, A. (eds.) *Cooperative Robots and Sensor Networks, Studies in Computational Intelligence*, vol. 554, pp. 87–113. Springer, Berlin (2014)
- Müller, J., Burgard, W.: Efficient probabilistic localization for autonomous indoor airships using sonar, air flow, and IMU sensors. *Adv. Robot.* **27**(January), 711–724 (2013)
- Nise, N.S.: *Control Systems Engineering*. Addison-Wesley, Boston (1995)
- Paulos, E., Canny, J.: PRoP: personal roving presence. In: CHI '98 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 296–303 (1998)
- Pilz, J., Spöck, G.: Why do we need and how should we implement bayesian kriging methods. *Stoch. Env. Res. Risk Assess.* **22**(5), 621–632 (2008)
- St-Onge, D., Imment Gosselin, C., Reeves, N.: Dynamic modelling and control of a cubic flying blimp using external motion capture. *J. Syst. Control Eng.* **229**(10), 970–982 (2015)
- Zhang, H., Ostrowski, J.P.: Visual servoing with dynamics: control of an unmanned blimp. In: Proceedings of the 1999 IEEE/RSJ International Conference on Robotics and Automation, pp. 618–623 (1999)
- Zhang, F., Marani, G., Smith, R., Choi, H.: Future trends in marine robotics. *IEEE Robot. Autom. Mag.* **22**, 14–21 (2015)
- Zufferey, J.-C., Guanella, A., Beyeler, A., Floreano, D.: Flying over the reality gap: from simulated to real indoor airships. *Auton. Robot* **21**, 243–254 (2006)

Zwann, S., Bernardino, A., Santos-Victor, J.: Vision based station keeping and docking for an aerial blimp. In: Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 614–619 (2000)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Sungjin Cho is a principal researcher in the department of Guidance and Control at the Agency for Defense Development, South Korea. He received a PhD degree in 2017 from the Georgia Institute of Technology (Atlanta) in Electrical and Computer Engineering. His research interests include marine and aerial robotics, mobile sensor network, and system and control theory.



Qiuyang Tao received the Ph.D. in Electrical Engineering from the Georgia Institute of Technology, Atlanta GA, USA, in 2020. His current research interests include the design, modeling and control of novel robotic and mechatronic systems.



Paul Varnell is a Ph.D. candidate in the School of Electrical and Computer Engineering with a focus in control theory and robotics. He received his B.S. degree in 2009 from Georgia Tech and then worked at Georgia Tech Research Institute (GTRI) on the Yellowfin underwater vehicle before enrolling in the Georgia Tech Ph.D. program. His research includes topics relating to computational tools for robust control, the influence

of communication on control, environmental sensing, and human-computer interaction.



Sean Maxon is a Ph.D. candidate in the school of Electrical and Computer Engineering at the Georgia Institute of Technology. In 2018 he joined the Center for High Assurance Computer Systems at the U.S. Naval Research Laboratory in Washington, DC. His current research interests include assured autonomy, multi-robot cooperative sensing, and wireless mesh networking for autonomous systems.



Fumin Zhang received the B.S. and M.S. degrees from Tsinghua University, Beijing, China, in 1995 and 1998, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Maryland, College Park, in 2004. He joined the School of ECE, Georgia Institute of Technology in 2007, where he is a Professor. He was a Lecturer and Postdoctoral Research Associate in the Mechanical and Aerospace Engineering Department, Princeton University from 2004

to 2007. His research interests include marine autonomy, mobile sensor networks, and theoretical foundations for battery supported cyber-physical systems. He received the NSF CAREER Award in 2009, and the ONR YIP Award in 2010.

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).

Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.

These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.

We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.

While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval, sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.

These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.

To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.

Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.

If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com