A Taguchi-Based Approach to Tune Bio-Inspired Guidance Systems for Tactical UAVs

Shardul N. Amrite

Dissertation submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master's of Science

in

Mechanical Engineering

Dr. Andrea L'Afflitto, Chair

Dr. Kwok Tsui

Dr. Steve Southward

Dr. Kaveh Akhbari Hamed

December 9, 2021

Blacksburg, Virginia

Keywords: Taguchi Analysis, ANOVA, UAV, Design of experiments

A Taguchi-Based Approach to Tune Bio-Inspired Guidance Systems for Tactical UAVs

Shardul N. Amrite

(ABSTRACT)

This thesis aims to tune the control parameters of a bio-inspired guidance system designed to confer a tactical behavior to unmanned aerial vehicles (UAVs). This bio-inspired guidance system is capable of reducing exposure to threats, while traversing previously uncharted, and potentially hostile territories. UAVs employing this guidance system may exhibit a more or less tactical behavior by tuning 9 user-defined parameters within specified intervals. Although the UAV's behavior can be easily forecasted whenever all parameters are set to exhibit the most cautious behavior or the most reckless behavior, it is difficult to devise a taxonomy of flight behavior whenever these parameters are not set at the boundaries of their admissible intervals. The scope of this thesis is to analyze and forecast the UAV's behavior as a function of these user-defined parameters. To this goal, the Taguchi analysis method is employed to deduce those parameters that affect the UAV's behavior more than others. Successively, 81 software-in-the-loop simulations have been performed to analyze the UAV's behavior as a function of the most influential user-defined parameters. Finally, 10 flight tests were performed to validate the numerical results.

A Taguchi-Based Approach to Tune Bio-Inspired Guidance Systems for Tactical UAVs

Shardul N. Amrite

(GENERAL AUDIENCE ABSTRACT)

This thesis aims to tune the control parameters of a bio-inspired guidance system designed to confer a tactical behavior to unmanned aerial vehicles (UAVs). This bio-inspired guidance system is capable of reducing exposure to threats, while traversing previously uncharted, and potentially hostile territories. UAVs employing this guidance system may exhibit a more or less tactical behavior by tuning 9 user-defined parameters within specified intervals. Although the UAV's behavior can be easily forecasted whenever all parameters are set to exhibit the most cautious behavior or the most reckless behavior, it is difficult to devise a taxonomy of flight behavior whenever these parameters are not set at the boundaries of their admissible intervals. The scope of this thesis is to analyze and forecast the UAV's behavior as a function of these user-defined parameters. To this goal, the Taguchi analysis method is employed to deduce those parameters that affect the UAV's behavior more than others. Successively, 81 software-in-the-loop simulations have been performed to analyze the UAV's behavior as a function of the most influential user-defined parameters. Finally, 10 flight tests were performed to validate the numerical results.

Acknowledgments

I would like to sincerely thank Dr. L'Afflitto, my primary advisor and mentor for this research, for his continued insights and guidance throughout this research. Over the past year and a half, he has supported me from the very inception of this research, all the way to the end, through technical insights and guidance on writing this thesis. The projects I have worked on under him have provided me with invaluable experience, which is helping me in my professional career. I owe special gratitude to Dr. Tsui. Having a subject matter expert like him, providing me with his inputs at every stage of this research, gave me confidence in the work I did. I would also like to sincerely thank Dr. Southward and Dr. Hamed for serving on my committee and thoroughly assessing this research and providing their valuable inputs. I believe my peers at the Advanced Control Systems Lab had great contributions to this research as well. I would like to especially thank Julius Marshall for his sharing his ideas about this research with me. Along with his inputs, he has also shared a plethora of his experimental data and results with me, which has greatly helped this research.

Contents

List of Figures viii

List of Tables x

1 Introduction 1

1.1 Scope of study 1

1.2 Thesis Outline 2

2 Overview of a Tactical Guidance System 3

2.1 Overview of Guidance System 3

2.1.1 Introduction to the bio-inspired guidance system 3

2.1.2 Nomenclature for tactical guidance system 4

2.1.3 Overview of the path planning subsystem 6

2.1.4 Overview of the trajectory planning subsystem 7

2.1.5 Equations of motion and dynamic constraints 9

2.1.6 Collision avoidance, yaw angle, and saturation constraints 10

2.1.7 Soft Constraints 12

2.2 Conclusion 13

3 Overview of the Taguchi Methodology **14**

3.1 Literature Review of the Taguchi Method 14

3.2 Overview of the Taguchi Methodology 19

3.2.1 Selection of factors and interactions 19

3.2.2 Selection of the number of levels of factors. 22

3.2.3 Selection of orthogonal array 23

3.2.4 Assignment of factors and interactions to columns of an OA 28

3.2.5 Conducting tests 33

3.3 Analysis of Results 33

3.3.1 Response parameter 34

3.3.2 Analysis of variance 40

3.4 Conclusion 46

4 Experimentation and Results of the Taguchi Analysis **47**

4.1 Overview of the UAV 48

4.2 Flight Maps and Simulations 51

4.2.1 Simulations 51

4.2.2 Maps Employed in Simulations 51

4.3 Flight Tests 54

4.4 Simulation Results 55

4.4.1 Results of Taguchi analysis 55

4.4.2 Results of plotting analysis and column effect method 66

4.5 ANOVA Results 67

4.6 Flight Tests Results 72

4.7 Conclusion 77

5 Conclusion **80**

Bibliography **82**

List of Figures

3.1 Example of a linear graph for L8 OA. 29

3.2 Linear Graph for L27 OA. 31

3.3 Modified linear graph for L27 OA employed in this research. 31

3.4 An example of plotting analysis 46

4.1 UAV used in this research. 48

4.2 Software architecture of the tactical GNC system as shown in [1]. 50

4.3 Obstacle configuration used in Map 1 52

4.4 Obstacle configuration used in Map 2 53

4.5 Obstacle configuration used in Map 3 54

4.6 Top view of the simulations on Map 1 58

4.7 Side view of the simulations on Map 1 58

4.8 Top view of the simulations on Map 2 59

4.9 Side view of the simulations on Map 2 59

4.10 Top view of the simulations on Map 3 60

4.11 Side view of the simulations on Map 3 60

4.12 Averages of R per level 61

4.13 Averages of SNR per level 62

4.14 Interaction plots for average R 63

4.15 Interaction plots for average SNR 64

4.16 Comparison of percentage contributions towards SNR and R 71

4.17 Trends observed in flight tests and simulations over Map 1 73

4.18 Trends observed in flight tests and simulations over Map 2 74

4.19 Trends observed in flight tests and simulations over Map 3 75

4.20 Paths observed for Experiments 5, 15, 21, and 26 over Map 1 76

4.21 Paths observed for Experiments 5, 15, and 26 over Map 2 77

4.22 Paths observed for Experiments 15, 21, and 26 over Map 3 78

List of Tables

3.1 Factors For Taguchi Analysis 21

3.2 Level settings for factors 23

3.3 L8 Inner Array with L4 Outer Array 25

3.4 L 27 OA (3^{13}) 27

3.5 Triangular table for an L8 OA 29

3.6 Column assignment of Factors in L27 OA 32

4.1 Moment of inertia of maps 54

4.2 Results of 27 numerical simulations. The columns denoted by d_{avg} show the average distance to the closest obstacle, and the columns denoted by R show the response over the three maps, respectively. The experiment with the largest SNR (most tactical), Experiment 19, has been highlighted in green. Experiment 16, which has the smallest SNR (most reckless), has been highlighted in red. It can be seen that, from Experiment 19 onward, larger SNR values are observed. This can be chalked down to the fact that all these experiments have a high value of μ_2. We can also see a generally decreasing trend in d_{avg} values from Map 1 to Map 3, since Map 1 has the smallest obstacle density. 56

4.3 Average R for each level 57

4.4 Average SNR for each level 57

4.5 Comparison of results of Experiment 19 and confirmation experiment 63

4.6 Comparison of parameter settings of Experiment 19 and optimal level-settings 65

4.7 ANOVA table based on SNR 68

4.8 F-ratios of contributing factors based on SNR 69

4.9 ANOVA table based on R 70

4.10 F-ratios of contributing factors based on R 70

4.11 Comparison of d_{avg} between flight test and simulation over Map 1 73

4.12 Comparison of d_{avg} between flight test and simulation over Map 2 74

4.13 Comparison of d_{avg} between flight test and simulation over Map 3 75

List of Abbreviations

ANOVA	Analysis of Variance
DOE	Design of experiments
DoF	Degrees of Freedom
FFE	Full Factorial Experiments
HB	Higher the Better condition
LB	Lower the Better condition
MSD	Mean Square Deviation
NB	Nominal is Best condition
OA	Orthogonal Array
PFE	Partial Factorial Experiments
SNR	Signal to Noise Ratio
UAV	Unmanned Aerial Vehicle

Chapter 1

Introduction

1.1 Scope of study

Unmanned aerial vehicles (UAVs) are becoming increasingly popular for the versatility of their application. Surveillance and mapping of hostile territories is one such application, where employing a UAV mitigates the risk to human life. At the Advanced Control Systems Lab at Virginia Tech, a guidance system for UAVs to perform such tasks has been created. This novel guidance is inspired by the cautious behavior of prey animals, and tactics used by ground troops while traversing hostile territories, and it is capable of performing this task without prior knowledge of the surrounding environment. This guidance system instills a tactical behavior by instructing the UAV to seek cover around obstacles and coast walls while moving toward their goal. The degree of tactical behavior conferred to the UAV depends on nine user-defined parameters characterizing the two subsystems of this guidance system, namely, the path planning subsystem and the trajectory planning subsystem.

The goal of this research is to analyze and forecast the UAV's behavior as a function of the tunable parameters underlying this guidance system for tactical operations. It is analytically impossible to compute the effect of a parameter on the behavior of the UAV, and thus an experimental approach is needed. For this purpose, we implement the Taguchi methodology, which is a variation of DOE introduced by Genichi Taguchi [2]. This method is particularly beneficial over the full factorial design of experiments as it reduces the number of experiments required to analyze the effect of the main control parameters. Subsequently, this thesis proposes the use of a specific mathematical response parameter to analyze the degree of tactical or cautious behavior shown by the UAV, irrespective of the environment traversed. Furthermore, we attempt to analyze the influence of different control parameters on the behavior of this system by applying tools proposed by Taguchi, such as the column effect method, plotting analysis, and ANOVA. The numerical results obtained by applying the Taguchi analysis are then validated experimentally.

1.2 Thesis Outline

This thesis is outlined as follows. Chapter 2 provides an overview of this novel guidance system for tactical UAVs. Concurrently, the role of user-defined parameters in setting the UAV's level of cautiousness or recklessness is explained. Chapter 3 provides an extensive literature review on the Taguchi methodology and its applications in different domains. This chapter also explains the procedures undertaken to apply the Taguchi method, while concurrently applying them to this particular research. Finally, Chapter 3 explains tools used to analyze the data obtained through experimentation, in particular, ANOVA (analysis of variance). Next, Chapter 4 begins by explaining the methodology undertaken while performing experimentation for this particular research. This includes an exposition on the UAV, the architecture of software implementation of this GNC system, the setups for simulations and flight tests, and the maps used in this research. Subsequently, this chapter displays the results of this research, and the results obtained via numerical analysis. Lastly, the results presented in this thesis and future work directions are outlined in Chapter 5.

Chapter 2

Overview of a Tactical Guidance System

The primary goal of this research is to study the effect of tuning the design parameters of a novel guidance system for a multirotor UAV developed at the Advanced Control Systems Lab at Virginia Tech under the guidance of Dr. Andrea L'Afflitto. This chapter describes the details of this novel guidance system as described in [1], and begins with an overview of the necessity of such a guidance system, followed by the nomenclature used to describe the different subsystems of this guidance system. Subsequently, the path planning, trajectory planning, and collision avoidance subsystems are explained in detail. Concurrently, this chapter also explains the expected effect of altering different control parameters.

2.1 Overview of Guidance System

2.1.1 Introduction to the bio-inspired guidance system

This research is based on an original guidance system that can confer a 'tactical' behavior to multi-rotor UAVs, such as quad-copters that operate in potentially hostile, unknown, cluttered environments [1]. This guidance system allows an autonomous vehicle to reach a goal set, whose position relative to the vehicle's initial position is given, without any prior knowledge of the environment [1]. It is inspired by the cautious behavior of animals, especially house mice [3], while reaching their goal set by minimizing the exposure to threats and by the tactics of ground troops while stealthily traversing hostile territories [4]. It does not depend on external sources for information about the UAV's position, velocity, or attitude.

This guidance system comprises a path planner, which implements an optimization-based algorithm, an algorithm to reconstruct collision avoidance constraints from voxel maps, which implements a discrimination algorithm based on quadratic programming, and a trajectory planner, which implements a fast model predictive control algorithm. Both the path planner and trajectory planner underlie a cost function, which influences the decision to move toward the goal or to seek shelter by coasting obstacles.

2.1.2 Nomenclature for tactical guidance system

Path planning subsystem's notation

Let the orthonormal inertial reference frame centered at O and with axes $X, Y, Z \in \mathbb{R}^3$ be denoted by $\mathbb{I} \triangleq \{O; X, Y, Z\}$. The path planning algorithm in this guidance system generates the UAV's reference path, which is a sequence of waypoints $\{\hat{r}_k\}_{k=0}^{n_\mathrm{p}} \subset \mathbb{R}^3 \setminus \mathcal{O}$ for the UAV expressed in the inertial reference frame $\mathbb{I}$. The UAV's initial position as defined by the user and is denoted by $\hat{r}_0$. The goal position in the inertial reference frame $\mathbb{I}$ is denoted by $\hat{r}_{n_\mathrm{p}}$ and $\hat{r}_{n_\mathrm{p}} \in \mathcal{G}$, where $\mathcal{G} \subset \mathbb{R}^3$ denotes the goal set. The obstacles' set is denoted by $\mathcal{O}$ and $\mathcal{O} \subset \mathbb{R}^3$. The obstacles' set $\mathcal{O}$ is the union of those voxels produced by the navigation system, which have an occupancy probability higher than a user-defined threshold. The integer $k \in \{0, \ldots, n_\mathrm{p}\}$ is employed as an index to denote a generic waypoint and to express functional dependencies on waypoints.

2.1. Overview of Guidance System

Trajectory planning subsystem's notation

Time is denoted by $t \geq 0$, and we assume that the UAV is able to fly from $\hat{r}_k$, $k \in \{0, \ldots, n_\mathrm{p}-1\}$, to $\hat{r}_{k+1}$ in $n_\mathrm{t}\Delta T$ time units, where both $n_\mathrm{t} \in \mathbb{N}$ and $\Delta T > 0$ are user-defined. In general, both n_t and ΔT are different for each pair of consecutive waypoints. The *UAV's position* is captured by $r_k : [0, n_\mathrm{t}\Delta T] \to \mathbb{R}^3 \setminus \mathcal{O}$ in the inertial reference frame $\mathbb{I}$, where $r_k(0) = \hat{r}_k$ and $r_k(n_\mathrm{t}\Delta T) = \hat{r}_{k+1}$. The *UAV's roll angle* is denoted by $\phi_k : [0, n_\mathrm{t}\Delta T] \to [0, 2\pi)$, $k \in \{0, \ldots, n_\mathrm{p}-1\}$, the *UAV's pitch angle* is denoted by $\theta_k : [0, n_\mathrm{t}\Delta T] \to \left(-\frac{\pi}{2}, \frac{\pi}{2}\right)$, the *UAV's yaw angle* is denoted by $\psi_k : [0, n_\mathrm{t}\Delta T] \to [0, 2\pi)$, the *UAV's velocity* with respect to $\mathbb{I}$ is denoted by $v_k : [0, n_\mathrm{t}\Delta T] \to \mathbb{R}^3$, and the UAV's *angular velocity* with respect to $\mathbb{I}$ is denoted by $\omega_k : [0, n_\mathrm{t}\Delta T] \to \mathbb{R}^3$. The *total thrust force* produced by the UAV's propellers is denoted by $u_{1,k}(\cdot)$, $k \in \{0, \ldots, n_\mathrm{p}-1\}$, the *roll moment* produced by the UAV's propellers is denoted by $u_{2,k}(\cdot)$, the *pitch moment* produced by the UAV's propellers is denoted by $u_{3,k}(\cdot)$, and the *yaw moment* produced by the UAV's propellers is denoted by $u_{4,k}(\cdot)$.

The UAV's state vector is given by

$$x_k(j\Delta T) \triangleq \left[r_k^\mathrm{T}(j\Delta T), \phi(j\Delta T), \theta(j\Delta T), \psi(j\Delta T), v_k^\mathrm{T}(j\Delta T), \omega_k^\mathrm{T}(j\Delta T)\right]^\mathrm{T}, \ j \in \{i, \ldots, n_\mathrm{t}\},$$

$i \in \{0, \ldots, n_\mathrm{t}\}$, $k \in \{0, \ldots, n_\mathrm{p}-1\}$. The trajectory planner generates reference states for the state vector $x_k(j\Delta T)$ and corresponding *control input* $u_k(j\Delta T) \triangleq [u_{1,k}(j\Delta T), u_{2,k}(j\Delta T), u_{3,k}(j\Delta T), u_{4,k}(j\Delta T)] \in \mathbb{R}^4$.

The reference state and control inputs $(x_k(\cdot), u_k(\cdot))$, are calculated by applying the model predictive control algorithm and are recomputed at each time step $j\Delta T$, starting from current time step $i\Delta T$. The integer $i \in \{0, \ldots, n_\mathrm{t}\}$ is used to count iterations of the model predictive control algorithm for a given pair of waypoints, and $j \in \{i, \ldots, n_\mathrm{t}\}$ is used to indicate the time step $j\Delta T$ within the interval $[i\Delta T, n_\mathrm{t}\Delta T]$.

The proposed guidance algorithm allows generating reference trajectories for a UAV, whose degree of cautiousness can be imposed by tuning the user-defined parameters $\mu_q \in \mathbb{R}$, $q \in \{1, \dots, 9\}$. Additional user-defined parameters, which do not affect directly the UAV's degree of cautiousness, are denoted by $\nu_q \in \mathbb{R}$, $q \in \{1, \dots, 4\}$.

2.1.3 Overview of the path planning subsystem

The path planning subsystem generates a solution to an optimization problem by applying the A^* search algorithm in a map. This map is assumed to be divided into cubical sections of equal size (voxels). The obstacles' set is represented by occupied voxels and the UAV is free to move to any unoccupied voxel cluster adjacent to the voxels currently occupied by the UAV. The reference path is a sequence of unoccupied voxels adjacent to one another starting from the UAV's initial position to the goal set and minimizing a user-defined cost function.

The cost function underlying the path planning algorithm is given by

$$f_k \triangleq g_k + h_k, \qquad k \in \overline{\mathbb{N}}, \tag{2.1}$$

where

$$g_k \triangleq \sum_{q=1}^{k} \left[\kappa(\mathrm{d}_2(\hat{r}_q, \mathcal{O}))\mathrm{d}_2(\hat{r}_q, \hat{r}_{q-1})\right] \tag{2.2}$$

denotes the *cost-to-come function*,

$$h_k \triangleq (1 - \mu_2)\mathrm{d}_2(\hat{r}_k, \mathcal{G}) \tag{2.3}$$

denotes the *heuristic function*,

$$\kappa(\alpha) \triangleq 1 - \mu_2 e^{4\mu_1\mu_3 - \left[\mu_3\alpha + \mu_1\alpha^{-1}\right]^2}, \qquad \alpha > 0, \tag{2.4}$$

denotes the *weighing function*, and $\mu_1, \mu_3 > 0$ and $\mu_2 \in [0, 1)$ are user-defined parameters.

Practically, (2.1) is the weighted sum of the distance traveled by the UAV, and an under-estimate of the Euclidean distance between the voxel occupied by the UAV and the goal set. The weighting function $\kappa(\cdot)$ in (2.4) instills a tactical behavior by rewarding paths closer to obstacles. From (2.2) and (2.4), it can be concluded that smaller values of μ_2 reduce the attractive effect of the obstacles' set and the UAV displays a more reckless behavior. For a smaller μ_2 value, the algorithm investigates a smaller number of voxels and is thus faster. Also, (2.2) and (2.4) show that for smaller values of $\mu_1\mu_3^{-1}$ and consequently larger values of μ_3, the algorithm produces paths that are closer to the obstacles, thus providing a more tactical path.

2.1.4 Overview of the trajectory planning subsystem

The path planner provides waypoints but does not take into account the dynamics of the motion between two waypoints provided by the path planner. It also can not control the velocity of the UAV in between obstacles and while coasting obstacles. For this reason, the trajectory planner is employed to compute reference trajectories as a solution to an optimal control problem solved by a fast model predictive control algorithm, which uses the waypoints from the path planner as points to interpolate between.

Cost Function

The cost function used to obtain tactical reference trajectories is given by

$$\tilde{J}[\hat{r}_k, u_k(\cdot)] \triangleq \ell_{\mathrm{f}}(r_k(n_{\mathrm{t}}\Delta T)) + \sum_{i=0}^{n_{\mathrm{t}}-1} \tilde{\ell}(r_k(i\Delta T), u_k(i\Delta T)), \qquad k \in \{0, \ldots, n_{\mathrm{p}} - 1\}, \tag{2.5}$$

where

$$\tilde{\ell}(r_k, u_k) \triangleq \begin{bmatrix} \tilde{r}_k \\ u_k \end{bmatrix}^{\mathrm{T}} \tilde{R} \begin{bmatrix} \tilde{r}_k \\ u_k \end{bmatrix} + \tilde{q}_r^{\mathrm{T}} \tilde{r}_k + \tilde{q}_u^{\mathrm{T}} u_k, \qquad (r_k, u_k) \in \mathbb{R}^3 \times \mathbb{R}^4, \tag{2.6}$$

$$\ell_{\mathrm{f}}(r_k) \triangleq (r_k - \hat{r}_{k+1})^{\mathrm{T}} R_{r,\mathrm{f}} (r_k - \hat{r}_{k+1}) + q_{r,\mathrm{f}}^{\mathrm{T}} (r_k - \hat{r}_{k+1}), \tag{2.7}$$

$\tilde{R} \triangleq \begin{bmatrix} \tilde{R}_r & \tilde{R}_{r,u} \\ \tilde{R}_{r,u}^{\mathrm{T}} & R_u \end{bmatrix}$, $\tilde{R}_r \in \mathbb{R}^{3\times 3}$ is symmetric, $\tilde{R}_{r,u} \in \mathbb{R}^{3\times 4}$, and $R_u \in \mathbb{R}^{4\times 4}$ are user-defined and such that R_u is positive-definite and

$$\tilde{R}_r - 2\tilde{R}_{r,u}^{\mathrm{T}} R_u^{-1} \tilde{R}_{r,u} > 0, \tag{2.8}$$

$R_{r,\mathrm{f}} \in \mathbb{R}^{3\times 3}$ is symmetric and nonnegative-definite, $\tilde{q}_r \in \mathbb{R}^3$, $q_{r,\mathrm{f}} \in \mathbb{R}^3$, and $\tilde{q}_u \in \mathbb{R}^4$ are user-defined,

$$\tilde{r}_k(i\Delta T) \triangleq \mu_4 [r_k(i\Delta T) - \hat{r}_{k+1}] + (1 - \mu_4) f_{\mathrm{sat}} (\mu_5(\hat{r}_k - r_{\mathcal{O}})) [r_k(i\Delta T) - r_{\mathcal{O}}], \qquad i \in \{0, \ldots, n_{\mathrm{t}} - 1\}, \tag{2.9}$$

$\mu_4 \in (0, 1]$ and $\mu_5 > 0$ are user-defined,

Mayer's term in equation (2.7) captures the UAV's need to reach the next waypoint.

The first term on the right-hand side of (2.9) captures the UAV's distance from the next waypoint and the second term captures the UAV's distance from the obstacles' set. Thus the *Lagrangian function* (2.6) portrays the UAV's competing needs to get to the next waypoint and to get closer to the obstacles' set.

Setting $\mu_4 = 1$ minimizes (2.5) and induces a reckless behavior to the UAV maximizing the priority to reach the goal. Minimizing μ_4 causes the UAV to give higher priority to coasting obstacles. The user-defined parameter μ_5 affects the distance from the UAV to which an obstacle's attractive effect is experienced. The attractive effect of obstacles beyond μ_5^{-1} is diminished.

2.1.5 Equations of motion and dynamic constraints

To capture the UAV's linearized equations of motion, let $m > 0$ denote the UAV's mass, $g > 0$ the *gravitational acceleration*, $I_x, I_y, I_z \in \mathbb{R}$ the *UAV's principal moments of inertia* about the roll, pitch and yaw axes respectively. The discrete-time, linearized equations of motion of the quadcopter are given by

$$x_k((j+1)\Delta T) = Ax_k(j\Delta T) + Bu_k(j\Delta T) \tag{2.10}$$

$$\begin{bmatrix} r_k(i\Delta T) \\ v_k(i\Delta T) \end{bmatrix} = \begin{bmatrix} r_{\text{init}} - r_{\text{e}} \\ v_{\text{init}} \end{bmatrix}, \qquad \begin{bmatrix} r_k(n_{\text{t}}\Delta T) \\ v_k(n_{\text{t}}\Delta T) \end{bmatrix} = \begin{bmatrix} \hat{r}_{k+1} - r_{\text{e}} \\ v_{\text{end}} \end{bmatrix},$$
$$j \in \{i, \ldots, n_{\text{t}} - 1\}, \qquad i \in \{0, \ldots, n_{\text{t}} - 1\}, \qquad k \in \{0, \ldots, n_{\text{p}} - 1\}, \tag{2.11}$$

where $A = e^{\tilde{A}\Delta T}$, and $B = \int_0^{\Delta T} e^{\tilde{A}\sigma} \mathrm{d}\sigma \tilde{B}$, $\tilde{A} \in \mathbb{R}^{12\times 12}$ is such that $\tilde{A}_{1,7} = \tilde{A}_{2,8} = \tilde{A}_{3,9} = \tilde{A}_{4,10} = \tilde{A}_{5,11} = \tilde{A}_{6,12} = 1$, $\tilde{A}_{7,5} = g$, $\tilde{A}_{8,4} = -g$, $\tilde{A}_{i,j}$ denotes the element of the ith row and jth column of $\tilde{A}$, every other element of $\tilde{A}$ is equal to zero, $\tilde{B} \in \mathbb{R}^{12\times 4}$ is such that $\tilde{B}_{9,1} = m^{-1}$, $\tilde{B}_{10,2} = I_x^{-1}$, $\tilde{B}_{11,3} = I_y^{-1}$, $\tilde{B}_{12,4} = I_z^{-1}$, every other element of $\tilde{B}$ is equal to zero, $r_{\mathrm{e}} = [0, 0, h_{\mathrm{e}}]^{\mathrm{T}} \in \mathbb{R}^3$, $h_{\mathrm{e}} \geq 0$ denotes the hover altitude for the UAV.

2.1.6 Collision avoidance, yaw angle, and saturation constraints

The guidance system presented in [1] guarantees collision avoidance, constraints on the maximum yaw angles, and saturation constraints on the control input. To find solutions to the trajectory planning problem, these constraints are captured by

$$F_k(i\Delta T) \begin{bmatrix} x_k(j\Delta T) \\ u(j\Delta T) \end{bmatrix} \leq\leq f_k(i\Delta T),$$

$$j \in \{i, \ldots, n_{\mathrm{t}}\}, \qquad i \in \{0, \ldots, n_{\mathrm{t}}\}, \qquad k \in \{0, \ldots, n_{\mathrm{p}} - 1\}, \quad (2.12)$$

where $F_k(i\Delta T) \triangleq \begin{bmatrix} F_{r,k}(i\Delta T) & 0_{l\times 2} & 0_{l\times 1} & 0_{l\times 6} & 0_{l\times 4} \\ 0_{2\times 3} & 0_{2\times 2} & F_\psi(i\Delta T) & 0_{2\times 6} & 0_{2\times 4} \\ 0_{8\times 3} & 0_{8\times 2} & 0_{8\times 1} & 0_{8\times 6} & F_u(i\Delta T) \end{bmatrix} \in \mathbb{R}^{(l+10)\times 16}$,
l denotes the number of collision avoidance constraints,
$f_k(i\Delta T) \triangleq \begin{bmatrix} f_{r,k}(i\Delta T) \\ f_{\psi,k}(i\Delta T) \\ f_u(i\Delta T) \end{bmatrix}$, $F_{r,k}(i\Delta T) \in \mathbb{R}^{l\times 3}$, $F_{\psi,k}(i\Delta T) \in \mathbb{R}^2$, $F_u(i\Delta T) \in \mathbb{R}^{8\times 4}$, $f_{r,k}(i\Delta T) \in \mathbb{R}^l$, $f_{\psi,k}(i\Delta T) \in \mathbb{R}^2$, and $f_u(i\Delta T) \in \mathbb{R}^8$.

To enforce collision avoidance constraints on the UAV's reference trajectory, the UAV's trajectory is outlined within closed ellipsoids defined by

$$\overline{\mathcal{E}}_k(i\Delta T) \triangleq \left\{ w \in \mathbb{R}^3 : (w - r_k(i\Delta T))^{\mathrm{T}} P_k(i\Delta T)(w - r_k(i\Delta T)) + c_k(i\Delta T) \leq 0 \right\},$$
$$i \in \{0, \ldots, n_{\mathrm{t}}\}, \qquad k \in \{0, \ldots, n_p - 1\}, \tag{2.13}$$

where $P_k(i\Delta T) \in \mathbb{R}^{3\times 3}$ and $c_k(i\Delta T) \in \mathbb{R}$ are solutions of the quadratic discrimination problem, whose cost function is given by

$$\min \mathbf{e}_{8,8}^{\mathrm{T}} b_k(i\Delta T). \tag{2.14}$$

By computing the hyperplanes tangent to $\partial\mathcal{E}_k(\cdot)$, $k \in \{0, \ldots, n_{\mathrm{p}} - 1\}$, at user-defined sampling points, collision avoidance constraints are captured by

$$F_{r,k}(i\Delta T) \triangleq \left[\sum_{q=1}^{l} \mathbf{e}_{q,l} \otimes (s_q(i\Delta T) - r_k(i\Delta T))^{\mathrm{T}} \right] P_k(i\Delta T),$$
$$i \in \{0, \ldots, n_{\mathrm{t}}\}, \qquad k \in \{0, \ldots, n_{\mathrm{p}} - 1\}, \tag{2.15}$$

$$f_{r,k}(i\Delta T) \triangleq \sum_{q=1}^{l} \left[\mathbf{e}_{q,l} \otimes (s_q(i\Delta T) - r_k(i\Delta T))^{\mathrm{T}} P_k(i\Delta T) s_q(i\Delta T) \right]. \tag{2.16}$$

The optical axes of the UAV's cameras are aligned with the UAV's roll axis. The reference yaw angle $\psi_k(\cdot)$, $k \in \{0, \ldots, n_{\mathrm{p}} - 1\}$, is constrained in a way that the end point $\hat{r}_{k+1}$ is always in the cameras' field of view. This requirement is captured by

$$-\psi_k(j\Delta T) \leq -\hat{\psi}_k(i\Delta T) + \psi_{\max}, \quad j \in \{i, \ldots, n_{\mathrm{t}}\}, \quad i \in \{0, \ldots, n_{\mathrm{t}}\}, \quad k \in \{0, \ldots, n_{\mathrm{p}} - 1\}, \tag{2.17}$$

$$\psi_k(j\Delta T) \leq \hat{\psi}_k(i\Delta T) + \psi_{\max}, \tag{2.18}$$

where $\hat{\psi}_k(i\Delta T) \triangleq \tan^{-1}\left(\frac{\mathbf{e}_{2,3}^{\mathrm{T}}\left(\hat{r}_{k+1} - r_k(i\Delta T)\right)}{\mathbf{e}_{1,3}^{\mathrm{T}}\left(\hat{r}_{k+1} - r_k(i\Delta T)\right)}\right)$, $\psi_{\max} > 0$ denotes the *cameras' half field of view*, and $\tan^{-1}(\cdot)$ denotes the signed inverse tangent function. Therefore,

$$F_\psi(i\Delta T) \triangleq [-1, 1]^{\mathrm{T}}, \qquad i \in \{0, \ldots, n_{\mathrm{t}}\}, \tag{2.19}$$

and

$$f_{\psi,k}(i\Delta T) \triangleq \left[\psi_{\max} - \hat{\psi}_k(i\Delta T), \psi_{\max} + \hat{\psi}_k(i\Delta T)\right]^{\mathrm{T}}, \qquad k \in \{0, \ldots, n_{\mathrm{p}} - 1\}. \tag{2.20}$$

The saturation constraints on the control input $u_k(\cdot)$ are captured by

$$-u_k(j\Delta T) \leq\leq u_{\max}, \quad j \in \{i, \ldots, n_{\mathrm{t}}\}, \quad i \in \{0, \ldots, n_{\mathrm{t}}\}, \quad k \in \{0, \ldots, n_{\mathrm{p}} - 1\}, \tag{2.21}$$

$$u_k(j\Delta T) \leq\leq u_{\max}, \tag{2.22}$$

where $u_{\max} \in \mathbb{R}^4$ is user-defined and such that $u_{\max} \geq\geq 0$. Therefore,

$$F_u(i\Delta T) \triangleq [-\mathbf{1}_4, \mathbf{1}_4]^{\mathrm{T}}, \qquad i \in \{0, \ldots, n_{\mathrm{t}}\}, \qquad k \in \{0, \ldots, n_{\mathrm{p}} - 1\}, \tag{2.23}$$

$$f_u(i\Delta T) \triangleq [u_{\max}^{\mathrm{T}}, u_{\max}^{\mathrm{T}}]^{\mathrm{T}}. \tag{2.24}$$

2.1.7 Soft Constraints

Constraints that can not be violated are known as *hard constraints*. To ensure that the trajectories of the UAV verify user-defined safety margins and to avoid a sudden increase in control inputs due to the activation of hard constraints, soft constraints are introduced in the cost function of the hard constraints. In particular, the reference state vector $x_k(j\Delta T)$ and control input $u_k(j\Delta T)$ are computed as solutions of the optimization problem given by the cost function

$$\hat{I}_{i,k,\mathrm{lb}}(z_{i,k}) \triangleq I_{i,\mathrm{lb}}(z_{i,k}) + \sum_{q=1}^{(l+10)(n_\mathrm{t}-i)} \frac{1}{\nu_{4,i,k,q}} \log\left(1 + e^{\nu_{4,i,k,q}\left[p_{i,k,q} z_i - \hat{h}_{i,k,q}\right]}\right),$$

$$z_{i,k} \in \mathbb{R}^{16(n_\mathrm{t}-i)}, \qquad i \in \{0,\dots,n_\mathrm{t}-1\}, \qquad k \in \{0,\dots,n_\mathrm{p}-1\}. \quad (2.25)$$

The user-defined parameter $\nu_{4,i,k,q}$, $i \in \{0,\dots,n_\mathrm{t}-1\}$, $k \in \{0,\dots,n_\mathrm{p}-1\}$, $q \in \{1,\dots,(l+10)(n_\mathrm{t}-i)\}$, is defined as follows. If there exists $d_{i,k} \in \mathcal{N}(P_{i,k})$, $i \in \{0,\dots,n_\mathrm{t}-1\}$, $k \in \{0,\dots,n_\mathrm{p}-1\}$, $q \in \{1,\dots,(l+10)(n_\mathrm{t}-i)\}$, such that $\|d_{i,k}\|_\infty < 1$ and $d_{i,k} >> 0$, then

$$\nu_{4,i,k,q} \triangleq \frac{1}{\hat{h}_{i,k,q}} \log\left[\frac{1}{d_{i,k,q}} - 1\right], \quad (2.26)$$

otherwise we set $\nu_{4,i,k,q} = 200$. Setting a high value of $\nu_{4,i,k,q}$ guarantees that there is no control offset induced.

2.2 Conclusion

This chapter has briefly recalled the guidance system presented in [1]. In this thesis, those multiple user-defined parameters that characterize this guidance system will be tuned and their effects on the behavior of the UAV will be analyzed. The next chapter presents an overview of the Taguchi method employed to tune and analyze the effect of these parameters.

Chapter 3

Overview of the Taguchi Methodology

In this chapter, we briefly outline the Taguchi method, which is the principal tool used in this research to study the effect of various user-defined parameters on a UAV's behavior. This chapter presents a brief overview of the Taguchi method, followed by a literature review of the application of this methodology in various industries. Successively, a detailed explanation of the procedure followed in the Taguchi analysis is provided. Concurrently, we present how this procedure is applied to analyze the performance of the guidance system outlined in Chapter 2. Additionally, multiple interpretations of the Taguchi analysis and ANOVA are presented. In the next chapter, experimental results are analyzed by applying the theoretical results outlined in this chapter.

3.1 Literature Review of the Taguchi Method

DOE is a powerful experimental tool used by researchers to plan, conduct, analyze, and interpret the effect of different control factors on the outcome of a process. The DOE method was first conferred by Sir R. A. Fisher in the 1920s in the UK [5]. The primary method in the DOE is the full factorial design. The full factorial design gives a complete understanding of the effects of main factors and the interaction effects of these factors on the outcome of the process. For a full factorial design with n control factors and k levels of each factor, k^n experiments must be performed [6]. This method requires a large number of experiments, which in many cases is particularly time-consuming and expensive. For example, an experimental setup with 5 control factors and 3 levels per control factor requires $3^5 = 243$ experiments. Statisticians have thus developed fractional factorial (FFEs) or partial factorial experiments (PFEs). These techniques use only a fraction of all possible experiments and estimate the main factor effect and some interaction effects. FFEs are often represented by their respective orthogonal arrays (OA). Orthogonal arrays have been used as early as 1897 by a French Mathematician called Jaques Hadamard [2].

The 'Taguchi Method' was developed in 1954 at the Electrical Communication Laboratory in Japan by Dr. Genichi Taguchi [6]. Several Japanese companies have since explored the Taguchi method to achieve higher productivity and better resulting products. Since the mid-1960s, this method is widely taught to engineers in Japan. In 1980, Dr. Taguchi received a grant from Aoyama-Gakuin University to lecture on his methods in the United States, before which there was little knowledge of his ideas in the Western world. He visited several institutions and companies like Xerox and Bell Labs, AT&T. These companies, along with Ford and organizations like the American Supplier Institute were instrumental in promoting this philosophy in the United States. The two Monhonk Conferences in 1984 and 1985, which the Quality Assurance Center of AT&T Bell Labs organized, exposed these ideas to a larger statistics and engineering community [2]. The Taguchi philosophy has two cornerstones, which are

- the reduction in variation of a product or process represents a lower loss to society;
- proper strategies can be developed to reduce variation in process or products to reduce variation [6].

The main objective of this method is to design systems that are robust under external, uncontrolled disturbances. Taguchi proposed an 8 step process to optimize a process or a product to get better results [6, 7]:

1. Identify the main function, side effects, and failure mode;
2. Identify the noise factors, testing conditions, and quality characteristics;
3. Identify the objective function to be optimized;
4. Identify the control factors and their levels;
5. Select the orthogonal array matrix experiment;
6. Conduct the matrix experiment;
7. Analyze the data and predict the optimum levels and performance;
8. Perform the verification experiment and plan the future action.

In the Taguchi approach, the design of a product is viewed as a three-phase process, namely system design, parameter design, and tolerance design. System design is the initial phase, where new processes, plans, ideas, and designs are developed to obtain a product with superior quality. Parameter design, which is the main focus of Taguchi's study is an experimental method to determine what level of control factors deliver the best result with minimum variation for a process or a product [7, 8]. By exploring the interactions between control parameters and noise parameters, parameter design aims at finding optimal settings of the control parameters so that the system's performance is robust [2]. Tolerance design is the last part of the process, where the product or process quality is improved by tightening the tolerances on a control parameter. Appropriate parameter designing often makes tolerance designing easier or more cost-efficient [6].

The Taguchi method relies on partial factorial experiments, which reduce the number of experiments required to investigate the effect of main control factors as compared to a full factorial design. Instead of using the full factorial designs, Dr. Taguchi developed his partial factorial designs and provided orthogonal arrays for them. These orthogonal array designs allow the selection of a subset of combinations of multiple factors at multiple levels to produce a design with equally balanced levels for all of the factors [9, 10]. Orthogonality allows separating the effects of individual control factors at each level [11]. For example, applying the Taguchi method to a problem with 5 control factors and 3 levels per control factor requires only 18 experiments (by using the L18 OA) instead of 243. Although the Taguchi method reduces the number of required experiments, it only gives a partial understanding of the effects of factor interactions.

Since the 1980s, the Taguchi method has largely been used to optimize manufacturing processes [12, 13]. For instance, *Venu Gopal et al.* have used the Taguchi method to investigate the maximum permissible material removal rate while maintaining the surface finish and material damage as constraints for a brittle SiC work-piece. They observed that with an increase in grit size and a decrease in depth of cut, feed rate, and grit density, the tangential grinding force, and surface roughness decreased, which was in accordance with their expectations from their literature review. They concluded that the feed rate, depth of cut, and grit size were the primary contributing factors to surface finish while grinding silicon carbide [14].

A product manufactured by fused deposition modeling (FDM) with higher dimensional accuracy and mechanical strength was obtained by *Alafaghani et al.* by setting ideal levels of building direction, infill ratio, layer height, extrusion temperature, printing speed, and infill pattern by using the Taguchi method [15]. *Anitha et al.* conducted their independent analysis on the factors influencing FDM using the Taguchi techniques and found optimal settings for speed of deposition, road width, and layer thickness [16]. In [17], the effects of hybridization of carbon fiber reinforced polymer composite laminates were investigated on high-velocity responses of laminates using a validated FEM model, the Taguchi analysis, and artificial neural networks. *Mahapatra et al.* optimized the wire electrical discharge machining process by applying this method to the process parameters [18].

The Taguchi method has also been applied to processes in the biotechnology domain. For instance, *Shahavi et al.* have applied this method to study the antibacterial effects of clove oil nano-emulsions. Xylitol production using anaerobic bacteria *Candida sp.* was optimized by *Rao et al* [19]. Concurrently, *Hou et al.* have applied the Taguchi method to investigate the factors affecting the droplet density of insecticides sprayed from UAVs on citrus trees [10]. This methodology has also been used in economics. *Wang et al.* have applied the Taguchi method to find ideal configurations for the controllable factors of a forecasting model [20].

The Taguchi analysis also finds applications in the aerospace industry. For instance, *Kapsalis et al.* have used the Taguchi analysis with computational fluid dynamics to optimize design parameters of a blended wing body UAV, like quarter-chord sweep, aspect ratio, and taper ratio to achieve desired performance characteristics on maximum velocity, required runway length, and gross take-off weight [21]. *Soylak et al.* used the Taguchi method to analyze the effects of aerofoil shape, wing angle of attack, and Reynolds's number on the wing performance at low speeds [22]. In [23], *Abhiram et al.* improved design parameters like rotor radius, tip chord, blade linear twist, and rotor speeds to obtain better hover performance for small unmanned helicopters.

For its ability to efficiently extrapolate relevant information from systems characterized by numerous parameters, the Taguchi analysis is a potentially promising candidate to study the effects of altering different control parameters of the guidance system on the behavior of the UAV. It can also be a powerful tool to determine levels of different user-defined parameters in the guidance system, presented in Chapter 2, to obtain desired (tactical or reckless) behaviors.

3.2 Overview of the Taguchi Methodology

This section contains a detailed exposition of the methodology associated with the Taguchi philosophy. This section also serves as the basis for the setup of the design of experiments used in this research.

3.2.1 Selection of factors and interactions

In the Taguchi method, the primary classification of the factors is done between *noise factors* or *control factors*. A factor, which can be altered at will is referred to as a control factor. A noise factor, on the other hand, is an uncontrolled factor that has an effect on the process or the product's performance. An 'interaction' is a condition wherein the effect of one factor on the result is dependant on the level of another factor [5]. This subsection provides details on how to select control factors.

Control factors can be selected in two ways, namely, knowledge of the system or cause-effect diagrams. Generally, factors to be investigated are decided by brainstorming and finding the factors that affect the system response. If the significant control factors are unknown, then it is recommended that preliminary experiments be done with a large number of factors, and with 2 levels each [6]. From these results, it is possible to narrow down the factors which are not in contention. Once the significant factors are confirmed, then the Taguchi analysis can be performed to investigate the effects and interactions in depth.

In this research, factors are selected based on the knowledge of the system, namely the factors that directly affect the path planning subsystem described in Section 2.1.3, the trajectory planning subsystem described in Section 2.1.4, and the soft constraints described in Section 2.1.7 are selected. The selected factors include three factors from the path planner subsystem, which are μ_1, μ_2, and μ_3, five factors from the trajectory planner subsystem, which are μ_4, μ_5, $\tilde{R}_r$, $\tilde{R}_u$, and T, and one factor from the soft constraints, which is ν_4. The rationale behind the selection of these parameters is given in this subsection.

One of the most important user-defined parameters in the path planning subsystem is μ_2. This user-defined parameter influences the $\kappa(\cdot)$ function in (2.4), which ultimately affects the cost-to-come function in (2.2). Altering the value of μ_2 directly affects the attractiveness of obstacles and the number of voxels investigated for potential paths. This is expected to have a large effect on the paths generated by the path planner and is thus a significant factor for this study.

Apart from μ_2, both μ_1 and μ_3 also have an effect on the $\kappa(\cdot)$ function in (2.4). As described in the Section 2.1.3, the ratio of μ_1/μ_3 has an effect on the paths generated. For a small value of this ratio, paths closer to obstacles are generated. To investigate this effect, μ_1 and μ_3 are included in this study. The user-defined factors μ_4 and μ_5 are employed in (2.9) of the trajectory planner subsystem. The UAV's need to reach the next waypoint is influenced by μ_4. The other user-defined factor, μ_5, captures the need of the UAV to coast the obstacles' set more closely. These factors are included in the study for the following reasons. Firstly, both these factors directly affect the trajectories planned for the UAV. Secondly, these parameters allow us to study the behavior of the UAV when the potentially competing needs of getting to the next waypoint and getting closer to the obstacles, are both set high or low.

The positive-definite matrices characterizing the optimal control problem solved by the fast model predictive control algorithm are given by $\tilde{R}_r$ and $\tilde{R}_u$. Specifically, the penalty imposed on the trajectory tracking error is altered by changing the values of the coefficients of

$\tilde{R}_r$, a diagonal matrix in $\mathbb{R}^{12\times12}$. Of these 12 coefficients, in this study, we modify only those coefficients that correspond to the lateral position (x), longitudinal position (y), vertical position (z), and the yaw angle (ψ). The user-defined coefficient $\tilde{R}_u$ is a diagonal matrix in $\mathbb{R}^{4\times4}$, which induces a penalty on the control input of the UAV. In this thesis, we alter the coefficients of $\tilde{R}_u$ corresponding to the control inputs on the roll, pitch, and yaw moments. Altering $\tilde{R}_r$ and $\tilde{R}_u$ will influence the trajectory and the behavior of the UAV between two waypoints. Another factor from the trajectory planner which is included in this study is the time horizon T.

As mentioned in Section 2.1.7, soft constraints are included to avoid the UAV from reaching hard constraints, which would result in a sudden change in the control effort of the UAV. The user-defined parameter ν_4 is used to include these constraints into the cost function of the optimization problem (2.25), whose solutions provide the reference state $x_k(\cdot)$ and the control inputs $u_k(\cdot)$.

Parameter	**Brief Description**	**Range**
μ_1	Affects path's distance from the obstacles. Used to calculate $\kappa(\cdot)$ in (2.4)	$\mu_1 > 0$
μ_2	Alters the attractive effect of obstacles. Used in equations (2.4),(2.3)	$\mu_2 \in (0, 1]$
μ_3	Affects path's distance from the obstacles. Used to calculate $\kappa(\cdot)$ in (2.4)	$\mu_3 > 0$
μ_4	Influences the UAV's need to get to the next waypoint. Used in equation (2.8)	$\mu_4 \in (0, 1]$
μ_5	Affects the distance beyond which the attractive effect of obstacles is diminished	$\mu_5 > 0$
$\tilde{R}_r$	Induces penalty on trajectory tracking error	$\tilde{R}_r > 0$
$\tilde{R}_u$	Induces penalty on control input in optimal control	$\tilde{R}_u > 0$
T	Time horizon of model predictive control	$T > 0$
ν_4	Soft constraint coefficient introduced while calculating state and control input	$\nu_4 > 0$

Table 3.1: Factors For Taguchi Analysis

The Taguchi method provides an opportunity to investigate the interactions among selected parameters. In this research, we investigate the presence of interactions between factors from the trajectory planning subsystem with a factor from the path planning subsystem. Since the user-defined path planner factor μ_2 directly affects the attractive effect of the obstacles' set, it has been selected as a factor from the path planning subsystem. On the other hand, from the trajectory planning subsystem, the factors μ_4 and μ_5 have been selected to study their respective interactions with μ_2. The interaction $\mu_2 \times \mu_4$ is an interesting one to study as μ_2 has the capacity to define how 'tactical' or 'reckless' the paths generated by the path planner are, and altering μ_4 affects how 'tactically' or 'recklessly' the UAV will behave in between those waypoints. An interesting condition would be where, due to μ_2, 'tactical' paths are generated but μ_4 induces a reckless behavior in-between waypoints. Similarly, $\mu_2 \times \mu_5$ is an interesting interaction because there could arise scenarios, where the paths may be reckless and the trajectory tactical.

To conclude this subsection, nine factors have been selected for their various traits, out of which, three are from the path planner subsystem, five from the trajectory planning subsystem, and a factor from the soft constraints. Furthermore, the study of two interactions are proposed, which are, $\mu_2 \times \mu_4$ and $\mu_2 \times \mu_5$. The factors selected are presented in Table 3.1 with brief descriptions and their ranges. These, along with the two interactions ($\mu_2 \times \mu_4$ and $\mu_2 \times \mu_5$) will be studied using the Taguchi method.

3.2.2 Selection of the number of levels of factors.

Once the factors and interactions for the study have been picked, it is necessary to determine the number of level settings for each factor necessary to analyze the effect on the response. There are two kinds of factors involved in the testing, namely continuous and discrete. Continuous factors are the ones that can vary to their extremes and assume any value in between. Discrete factors are the ones that can only assume particular values.

All the factors selected for this research are continuous. The selected factors have been studied at three levels to analyze the possibility of quadratic or higher-order effects of the factors. Table 3.2 provides the level settings of all the selected factors. For factors that are defined over a compact interval, the levels have been selected in a manner that Level 1 is just above the minimum value, Level 2 is near the middle of the interval and Level 3 is just below the maximum permissible value.

Parameter	**Level 1**	**Level 2**	**Level 3**	**Range**
μ_1	0.01	1.00	5.00	$\mu_1 > 0$
μ_2	0.10	0.50	0.95	$\mu_2 \in (0, 1]$
μ_3	0.01	1.00	5.00	$\mu_3 > 0$
μ_4	0.20	0.05	0.95	$\mu_4 \in (0, 1]$
μ_5	0.10	0.30	0.60	$\mu_5 > 0$
$\tilde{R}_r$	350.00	700.00	1000.00	$\tilde{R}_r$>0
$\tilde{R}_u$	100.00	300.00	500.00	$\tilde{R}_u$>0
T	10.00	50.00	100.00	T>0
ν_4	10.00	100.00	1000.00	ν_4>0

Table 3.2: Level settings for factors

3.2.3 Selection of orthogonal array

Selecting the appropriate OA is an extremely important part of the Taguchi method. Orthogonal arrays are named with ‘L’ followed by the number of trials in the respective OA. Thus, for example, an OA with 8 experiments is tagged as L8. The arrays provided by Taguchi qualify as ‘orthogonal’ as they satisfy the following conditions: firstly, they are internally balanced, which means that every column (representing a parameter) has an equal number of experiments at each level. For example, each column of the L8 array depicted in Table 3.3 has 4 experiments at Level 1 and Level 2. Secondly, any two columns, together, are balanced, which means any two columns will have the same level-combinations as any other two columns. For instance, in an L8 array, if we analyze the levels prescribed for different experiments for any two columns, then we will obtain similar pairs, which are Level 1 - Level

1, Level 1 - Level 2, Level 2 - Level 1, Level 2 - Level 2. By satisfying these two conditions, an array is said to be orthogonal [5].

The above-mentioned OAs are also referred to as *'inner arrays'*, as it only accommodates control parameters. It is possible to repeat each experiment several times, assuming the error variation is an aggregate of all noise factors equally distributed in all conditions [6]. To incorporate the noise factors into the design strategy, the noise factors are included in an *'outer array'*. This design approach separates the control factors from the noise factors. By using an outer array, each experiment is performed at various levels of the external noise. It is then possible to find control factors and their specific levels, which are not highly sensitive to the noise factors.

Table 3.3 provides an example of the application of an outer array to accommodate 3 noise factors (X, Y, Z) to an L8 experiment with 7 control factors. This would provide a total of 28 separate conditions. Including an outer array often greatly increases the number of experiments to be conducted, and thus usually only one, or two noise factors are included at their extreme (minimum and maximum) conditions.

The selection of an appropriate OA depends on the number of factors and interactions, and the number of levels associated with each factor and interaction. Given the Factors A and B, the following formulas provide details on how the degrees of freedom (DoF) of individual factors and interactions are calculated

$$dof_A = K_A - 1, \tag{3.1}$$

$$dof_B = K_B - 1, \tag{3.2}$$

$$dof_{A\times B} = dof_A \times dof_B, \tag{3.3}$$

								L4 outer array			
							X	1	2	2	1
							Y	1	2	1	2
							Z	1	1	2	2
	L8 inner array										
	A	B	C	D	E	F	G	Result			
	Column Number										
Trial	1	2	3	4	5	6	7	y_1	y_2	y_3	y_4
1	1	1	1	1	1	1	1	*	*	*	*
2	1	1	1	2	2	2	2	*	*	*	*
3	1	2	2	1	1	2	2	*	*	*	*
4	1	2	2	2	2	1	1	*	*	*	*
5	2	1	2	1	2	1	2	*	*	*	*
6	2	1	2	2	1	2	1	*	*	*	*
7	2	2	1	1	2	2	1	*	*	*	*
8	2	2	1	2	1	1	2	*	*	*	*

Table 3.3: L8 Inner Array with L4 Outer Array

where K_A and K_B denote the levels of A and B respectively. To select an OA, it is necessary to know the total DoF required for the proposed experiment, which is denoted by dof_{OA}. The total DoF required for the experiments, dof_{OA}, is defined as the sum of the individual DoF of each factor and interaction. Once the required DoF is known, an appropriate OA can be selected from the OAs provided by Taguchi's method, which can accommodate for required number of levels and satisfies the following condition

$$dof_{OA} \geq dof_{experiment}, \tag{3.4}$$

where

$$dof_{OA} = N - 1 \tag{3.5}$$

and N denotes the number of experiments in the orthogonal array.

It is possible to modify OAs to accommodate factors that have a different number of levels than the ones prescribed in the OA. This can be done by *merging columns*, that is, accommodating a four-level factor in a two-level OA, by *dummy treatment*, that is, accommodating a three-level factor in a two-level array, by *idle column method*, that is, accommodating multiple three-level factors in a two-level OA, or by *combination method*, that is, accommodating two-level factors in a three-level OA [6].

Following the guidelines stated above, the total DoF for the experiment is calculated for this research. So, from (3.1), each control factor in this research will have 2 DoF, since each control factor has 3 levels. As there are nine factors involved in this study, the total DoF of factors is equal to 18 (two DoF per factor). From (3.3), the DoF of interaction for an interaction of two factors will be 4, since each interacting factor has a DoF of 2, that is,

$$dof_{\mu_2 \times \mu_4} = dof_{\mu_2} \times dof_{\mu_4} = 2 \times 2 = 4. \tag{3.6}$$

Next, the total DoF required for this study is the sum of DoF of all factors and interactions, that is,

$$dof = dof_{factors} + dof_{interactions} = 18 + 8 = 26. \tag{3.7}$$

This L27 OA, which has 27 experiments and 26 DoF, satisfies the condition stated in (3.4). This OA also has a provision to accommodate 3 levels for each factor. Since the available DoF matches the required DoF and has the required number of levels, the L27 OA has been selected as the basis for this research. The L27 OA has 13 columns, which means it can support up to 13 individual 3-level factors and prescribes a total of 27 experiments. Table 3.4 provides a structure of this OA.

Column	1	2	3	4	5	6	7	8	9	10	11	12	13
Factors	A	B	C	D	E	F	G	H	I	J	K	L	M
Trial													
1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	2	2	2	2	2	2	2	2	2
3	1	1	1	1	3	3	3	3	3	3	3	3	3
4	1	2	2	2	1	1	1	2	2	2	3	3	3
5	1	2	2	2	2	2	2	3	3	3	1	1	1
6	1	2	2	2	3	3	3	1	1	1	2	2	2
7	1	3	3	3	1	1	1	3	3	3	2	2	2
8	1	3	3	3	2	2	2	1	1	1	3	3	3
9	1	3	3	3	3	3	3	2	2	2	1	1	1
10	2	1	2	3	1	2	3	1	2	3	1	2	3
11	2	1	2	3	2	3	1	2	3	1	2	3	1
12	2	1	2	3	3	1	2	3	1	2	3	1	2
13	2	2	3	1	1	2	3	2	3	1	3	1	2
14	2	2	3	1	2	3	1	3	1	2	1	2	3
15	2	2	3	1	3	1	2	1	2	3	2	3	1
16	2	3	1	2	1	2	3	3	1	2	2	3	1
17	2	3	1	2	2	3	1	1	2	3	3	1	2
18	2	3	1	2	3	1	2	2	3	1	1	2	3
19	3	1	3	2	1	3	2	1	3	2	1	3	2
20	3	1	3	2	2	1	3	2	1	3	2	1	3
21	3	1	3	2	3	2	1	3	2	1	3	2	1
22	3	2	1	3	1	3	2	2	1	3	3	2	1
23	3	2	1	3	2	1	3	3	2	1	1	3	2
24	3	2	1	3	3	2	1	1	3	2	2	1	3
25	3	3	2	1	1	3	2	3	2	1	2	1	3
26	3	3	2	1	2	1	3	1	3	2	3	2	1
27	3	3	2	1	3	2	1	2	1	3	1	3	2

Table 3.4: L 27 OA (3^{13})

3.2.4 Assignment of factors and interactions to columns of an OA

Once an OA is selected, the factors need to be assigned to the OA in a particular order. The assignment of factors to columns is specific to the experiment. To aid this process, Taguchi provided two tools to assign columns and interactions in an OA. These tools are *linear graphs* and *triangular tables*. This subsection briefly presents these tools.

Linear graphs

A linear graph provides a graphical method to represent the interactions available in an OA; for an example of a linear graph, see Figure 3.1. In linear graphs, the dots at the end of a line represent the interacting columns. The numbers specified over the line segment represent the columns in which the interaction effect is seen. For 2 level factors, there is a single column that represents the interaction, since the DoF of the interaction is one (each column has 1 DoF in a two-level OA). Similarly, for 3 level factors, 2 columns represent the interaction since it has 4 DoF (each column has 2 DoF in a three-level OA). Some methods allow to alter these linear graphs and either reduce or add an interaction to the OA. However, the sum of DoF of all interactions and factors can not exceed the DoF of the OA. According to the linear graphs, the factors, whose interactions are to be studied, are assigned.

Figure 3.1 provides an example of a linear graph for an L8 OA; the L8 OA has 2 levels per factor and 7 columns. According to this figure, the interaction between Columns 1 and 2 can be seen in Column 3. The interaction between Columns 1 and 4 can be seen in Column 5. The interaction between Columns 2 and 4 can be seen in Column 6. Lastly, Column 7 does not support interactions.

Consider an experimental setup that has 4 control factors A, B, C, and D, and the interactions $A \times B$ and $A \times C$ are to be investigated. One could place Factor A in Column

1, Factor B in Column 2, and Factor C in Column 4. Factor D will be assigned to Column 7 since it does not interact with any other factors. The interaction $A \times B$ can be observed in Column 3, and the interaction $A \times C$ can be observed in Column 5.

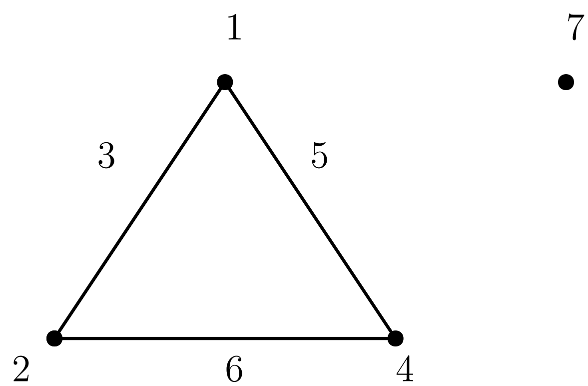

Figure 3.1: Example of a linear graph for L8 OA.

	Col No.					
Col No.	2	3	4	5	6	7
1	3	2	5	4	7	6
2	-	1	6	7	4	5
3	-	-	7	6	5	4
4	-	-	-	1	2	3
5	-	-	-	-	3	2
6	-	-	-	-	-	1

Table 3.5: Triangular table for an L8 OA

Triangular tables

Triangular tables provide an alternative method to assign factors to the columns of an OA and provide a list of all the possible interacting columns in the OA. To check for columns representing the interaction between a selected pair of factors, the row in the triangular table representing the first interacting factor is selected. The entry in this row, which lies in the column represented by the second interacting factor, will provide the columns where interactions are observed.

Referring to the L8 OA, the associated triangular table is portrayed in Table 3.5. To check for interaction between Column 1 and Column 4, we have to access the third column of the first row, which indicates Column 5 contains the interaction information. Since these interactions have only one DoF, there is a single column represented there. For OAs with 3 level-factors, there are two columns representing an interaction, since it has 4 DoF (each column has 2 DoF in a three-level-factor OA).

In this research, the L27 OA and its associated linear graphs and triangular tables are used. From the linear graph in Figure 3.2, it can be seen that the interaction of Columns 1 and 2 is captured by Columns 3 and 4, the interaction of Columns 1 and 5 are captured by Columns 6 and 7, and the interaction of Columns 2 and 3 are captured by Columns 8 and 11. Columns 9, 10, 12, and 13 are reserved for individual factors and do not represent any interaction. As we need to study two interactions, and not three, the linear graph needs to be modified to accommodate only two interactions. This can be done by 'borrowing' four DoF (2 columns) from one of the interactions. The modified linear graph for this research is shown in Figure 3.3. Columns 8 and 11, which represent the interactions of Columns 2 and 5 in Figure 3.2, are now represented as individual factors, which do not represent any interactions. So, in the modified linear graph, Columns 8, 9, 10, 11, 12, and 13 are used for individual factors, while Columns 1, 2, and 5 house factors for which interaction effects are going to be studied. The interaction effect of Columns 1 and 2 is seen in Columns 3 and 4. The interaction effect of Columns 1 and 5 is seen in Columns 6 and 7.

Using this information from the linear graph in Figure 3.3, we start assigning factors to respective columns. Since the interaction of factor μ_2, with μ_4 and μ_5, is of interest, μ_2 is assigned to Column 1 (as it interacts with Columns 2 and Column 5). The factor μ_4 is then assigned to Column 2 and the effects of its interaction with μ_2, are seen in Columns 3 and 4. Similarly, the user-defined factor μ_5 is placed in Column 5. By doing so, its interaction

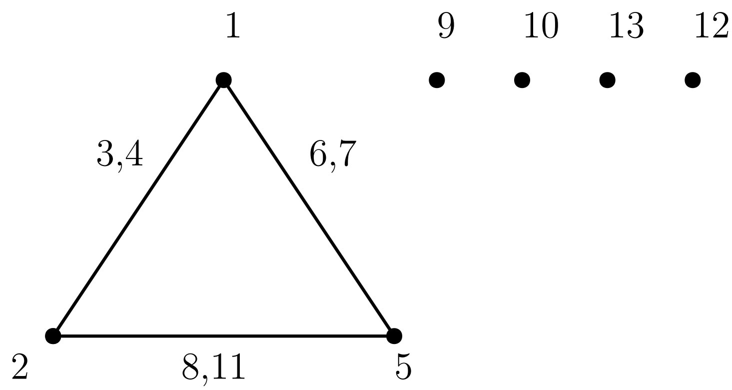

Figure 3.2: Linear Graph for L27 OA.

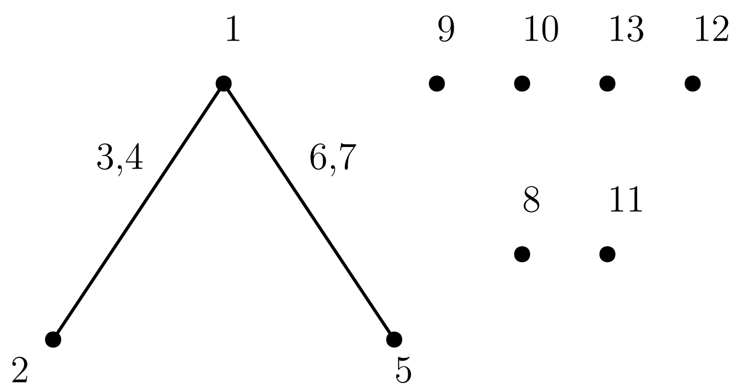

Figure 3.3: Modified linear graph for L27 OA employed in this research.

effect with μ_2 can be observed in Columns 6 and 7. Since we are only studying the main factor effects of the rest of the factors, they are placed in Columns 8, 9, 10, 11, 12, and 13. The factor μ_1 is placed in Column 8, μ_3 in Column 9, $\tilde{R}_r$ in 10, $\tilde{R}_u$ in 11, T in 12, and finally ν_4 in 13. In this manner, the total table is occupied and the DoF adds up to 26, which is equal to the DoF of the OA. The final column assignment, with level settings, is represented in Table 3.6.

Factors	μ_2	μ_4	$(\mu_2 \times \mu_4)_1$	$(\mu_2 \times \mu_4)_2$	μ_5	$(\mu_2 \times \mu_5)_1$	$(\mu_2 \times \mu_4)_2$	μ_1	μ_3	$\tilde{R}_r$	$\tilde{R}_u$	T	ν_4
1	0.1	0.2	1	1	0.1	1	1	0.01	0.01	350	100	10	10
2	0.1	0.2	1	1	0.3	2	2	1	1	700	300	50	100
3	0.1	0.2	1	1	0.6	3	3	5	5	1000	500	100	1000
4	0.1	0.5	2	2	0.1	1	1	1	1	700	500	100	1000
5	0.1	0.5	2	2	0.3	2	2	5	5	1000	100	10	10
6	0.1	0.5	2	2	0.6	3	3	0.01	0.01	350	300	50	100
7	0.1	0.99	3	3	0.1	1	1	5	5	1000	300	50	100
8	0.1	0.99	3	3	0.3	2	2	0.01	0.01	350	500	100	1000
9	0.1	0.99	3	3	0.6	3	3	1	1	700	100	10	10
10	0.5	0.2	2	3	0.1	2	3	0.01	1	1000	100	50	1000
11	0.5	0.2	2	3	0.3	3	1	1	5	350	300	100	10
12	0.5	0.2	2	3	0.6	1	2	5	0.01	700	500	10	100
13	0.5	0.5	3	1	0.1	2	3	1	5	350	500	10	100
14	0.5	0.5	3	1	0.3	3	1	5	0.01	700	100	50	1000
15	0.5	0.5	3	1	0.6	1	2	0.01	1	1000	300	100	10
16	0.5	0.99	1	2	0.1	2	3	5	0.01	700	300	100	10
17	0.5	0.99	1	2	0.3	3	1	0.01	1	1000	500	10	100
18	0.5	0.99	1	2	0.6	1	3	1	5	350	100	50	1000
19	0.95	0.2	3	2	0.1	3	2	0.01	5	700	100	100	100
20	0.95	0.2	3	2	0.3	1	3	1	0.01	1000	300	10	1000
21	0.95	0.2	3	2	0.6	2	1	5	1	350	500	50	10
22	0.95	0.5	1	3	0.1	3	2	1	0.01	1000	500	50	10
23	0.95	0.5	1	3	0.3	1	3	5	1	350	100	100	100
24	0.95	0.5	1	3	0.6	2	1	0.01	5	700	300	10	1000
25	0.95	0.99	2	1	0.1	3	2	5	1	350	300	10	1000
26	0.95	0.99	2	1	0.3	1	3	0.01	5	700	500	50	10
27	0.95	0.99	2	1	0.6	2	1	1	0.01	1000	100	100	100

Table 3.6: Column assignment of Factors in L27 OA

3.2.5 Conducting tests

While conducting tests, various trials should be randomized to protect the experiment from any uncontrolled factors. In this research, a bulk of the tests are performed through simulations, and thus randomization is not necessary. The tools for the analysis of results are provided in the subsequent section. Following the analysis, the parameter combination which results in the optimum result is deduced. This parameter combination is used to perform experiments or 'confirmation tests'; this is the final step of the Taguchi methodology. Usually, it is recommended to perform several runs of the confirmation experiments. The results of confirmation tests are then compared to the anticipated averages.

3.3 Analysis of Results

The methods of analyzing the data gathered from the experiments are explained in this section. These methods allow us to deduce the optimal level settings of the factors, the influence of each factor and interaction toward the response, and estimates of the response at the optimal level setting. Since the Taguchi philosophy revolves around the reduction of variation in a product or a process, it is important to analyze the sources of variance. Furthermore, since the OAs provided by Taguchi's method are partial factorial experiments, it is necessary to analyze the confidence associated with the results [5]. These analyses can be achieved through a statistical tool called the analysis of variance (ANOVA). ANOVA can be used to analyze any differences in the average performance of groups of items to be tested [6]. This section provides details on the response parameter used in this thesis, the methodology followed in ANOVA, and we conclude this section by describing two methods prescribed by Taguchi to interpret the results.

3.3.1 Response parameter

The response parameter is a measure of the performance of the process. In the Taguchi analysis, only one response parameter is used. Other methods, such as the grey-Taguchi method, consider multiple response variables and condense them to a single response parameter through a weighting function [24]. The response parameters are usually of three types, which are *lower the better, higher the better*, and *target values.*

Taguchi prescribes the use of signal-to-noise ratio (SNR) to analyze the sources of variance in the process or product. The change in response due to the alteration of a control factor is referred to as a 'signal'. The SNR measures the sensitivity of the quality characteristic being investigated in a controlled manner to those influencing factors, such as noise, that are not under control [5]. A high SNR indicates that a control parameter is more robust against noise [10]. The SNR of different experiments can be used to determine optimum settings for the desired result since the experiment with the highest SNR gives the optimum level setting for the control factors among all the combinations prescribed in the selected OA. A higher SNR also implies that the experiment has the minimum variance, and the higher the SNR, the lower the deviation of the respective experiment [5].

There are three primary approaches to calculate the mean square deviation (MSD), which is required to calculate the SNR, according to the desired nature of the result. These approaches are

- Lower the better (LB) response: This condition is used when the response with low values provides an ideal scenario for the functioning of the system. In this case, the MSD is computed as

$$MSD = \frac{(y_1^2 + y_2^2 + y_3^2 + ... + y_n^2)}{n}, \tag{3.8}$$

where y_i denotes the result of the i^{th} repetition of the k^{th} experiment and n denotes the total number of repetitions.

- Higher the better (HB) response: This condition is used when the response with high values provides an ideal scenario for the functioning of the system. In this case, the MSD is computed as

$$MSD = \frac{(1/y_1^2 + 1/y_2^2 + 1/y_3^2 + ... + 1/y_n^2)}{n}. \tag{3.9}$$

- Target value or nominal is best condition (NB): This condition is used when the response is required to tend to a predetermined value. In this case, the MSD is computed as

$$MSD = \frac{(y_1 - y_0)^2 + (y_2 - y_0)^2 + (y_3 - y_0)^2 + .. + (y_n - y_0)^2}{n}. \tag{3.10}$$

Using the MSD calculated according to the above conditions, the SNR is then calculated as

$$SNR = -10 \times log_{10}(MSD). \tag{3.11}$$

As this research aims at examining how 'tactical' the behavior of the UAV is, which is an intangible quality, we have defined a dedicated response parameter, R, for this research. The main aim of the guidance system is to protect the UAV in hostile territories. One way in which this is achieved is by flying close to obstacles, thus minimizing the exposure to threats from *any* direction. For this reason, the average distance from the UAV to the closest obstacle, at every time step, over the entire flight is considered as a part of the response parameter, R. The average distance from the UAV to the closest obstacle is henceforth denoted by d_{avg}. Since it is desirable to have lower values of the average distance to the wall, the 'lower the better' condition is used. To calculate the average distance to an obstacle, it is necessary to have the topology of the environment in which the UAV flying. Since the UAVs employed in this research are equipped with a simultaneous navigation and mapping system, which creates in real-time a map of the environment, it is possible to obtain this information. From the diagnostic logs recorded for each flight or simulation by the UAV, the coordinates of the UAV at every time step are obtained. Then, at every time step through the duration of the flight, the distance to the closest occupied voxel is found by comparing the Euclidean distances to the occupied voxels from the UAV's position. The average of all the minimum obstacle distances at each time step is then used as a part of the response parameter.

In this research, we are interested in characterizing the ability of the given guidance system to produce tactical behaviors, irrespective of the map employed to perform numerical simulations and flight tests. For this reason, in this research, we account for the density of obstacles and the obstacle distribution in the given simulation or flight test scenarios as noise factors. The obstacle density of a map is the ratio of the total volume occupied by obstacles in a map and the total volume of the map.

To capture the obstacle distribution and the obstacle density of the map, we calculate the 'moment of inertia' of the map. This notion of moment of inertia, which is borrowed from computer graphics [25], is analogous to calculating the moment of inertia of an object. It would convey both how spread are the obstacles and the obstacle density of the map. The average obstacle distance is then 'normalized' for the map by dividing by the moment of inertia of the map. By doing this, we are essentially making the response parameter more immune to the obstacle in the map. This can be looked at as a scaling factor for the response parameter, which depends on the nature of the map being used.

To calculate the moment of inertia of a map, the following methodology is applied. Firstly, the 'mass' of the map is calculated, which is the zero-order moment; the order of the moment is the sum of powers of x, y, and z distances. The powers are represented by the superscripts i, j and k for x, y, and z distances, respectively. The zero-order moment is the sum of the 'mass' of all the occupied voxels over the entire map. An occupied voxel is assumed to have a mass of 1, and an unoccupied voxel is assumed to have a mass of 0. The zero-order moment is defined as

$$m_{000} \triangleq \sum_x \sum_y \sum_z x^0 y^0 z^0 \mathcal{I}, \tag{3.12}$$

where $\mathcal{I}$ denotes the indicator function, that is,

$$\mathcal{I} = 1, \tag{3.13}$$

if the voxel is occupied, and

$$\mathcal{I} = 0, \tag{3.14}$$

if the voxel is unoccupied. In a similar manner, the first order moments are calculated for each obstacle individually, along the three axes by the using following expressions

$$m_{100} \triangleq \sum_{x,y,z} x y^0 z^0 \mathcal{I} = \sum_{x,y,z} x \mathcal{I}, \tag{3.15}$$

$$m_{010} \triangleq \sum_{x,y,z} x^0 y z^0 \mathcal{I} = \sum_{x,y,z} y \mathcal{I}, \tag{3.16}$$

$$m_{001} \triangleq \sum_{x,y,z} x^0 y^0 z \mathcal{I} = \sum_{x,y,z} z \mathcal{I}, \tag{3.17}$$

where m_{100}, m_{010}, and m_{001} denote the first order moments along the three axes, and x, y, and z denote the distances to the voxel from the origin in an orthonormal reference frame. Using the zero order moment and the first order moments, the centroid of a map is calculated for individual obstacles as follows

$$\bar{x} \triangleq m_{100}/m_{000}, \tag{3.18}$$

$$\bar{y} \triangleq m_{010}/m_{000}, \tag{3.19}$$

$$\bar{z} \triangleq m_{001}/m_{000}. \tag{3.20}$$

Using the centroid, the central moments (moments of inertia) of each individual obstacle is calculated about their centroid as

$$c^p_{i,j,k} \triangleq \sum_x \sum_y \sum_z (x-\bar{x})^i (y-\bar{y})^j (z-\bar{z})^k \mathcal{I}, \tag{3.21}$$

where i, j, and k can take the values of 0, 1, and 2, and the sum of i, j, and k is always equal to 2, and the index p denotes the p^{th} obstacle. The distances of the obstacle from the origin are represented by x, y, and z in the respective directions. The second order moments constitute the inertia matrix, which is given by

$$C^p \triangleq \begin{bmatrix} c^p_{200} & c^p_{110} & c^p_{101} \\ c^p_{110} & c^p_{020} & c^p_{011} \\ c^p_{101} & c^p_{011} & c^p_{002} \end{bmatrix}. \tag{3.22}$$

This inertia matrix is calculated for all obstacles individually. The Frobenius norm of the inertia matrices is used to calculate the 'magnitude' of the inertia of each obstacle as

$$M^p \triangleq \|C^p\|_F, \tag{3.23}$$

where C^p denotes the inertia matrix of the p^{th} obstacle and M^p denotes its Frobenius norm.

Similarly, the Frobenius norm of the inertia matrix is calculated for the entire map as a single entity. For this calculation, the indicator function $\mathcal{I}$ allows to define the occupancy of the map by eliminating the effect of the voxels that are not occupied. This is analogous to calculating the moment of inertia of an object with internal voids. The ratio of the sum of the Frobenius norms of inertia matrices of all obstacles and the Frobenius norm of the inertia matrix of the entire map is given by

$$M \triangleq \frac{\sum_{p=1}^{n} M^p}{M^t}, \tag{3.24}$$

where n denotes the total number of obstacles, and M^t denotes the Frobenius norm of the inertia matrix of the entire map as a single object. This quantity provides a measure of how scattered the map is and how densely populated it is with obstacles.

This map dependant quantity, M, is used to normalize the average distance from the UAV to the nearest obstacle (d_{avg}). Thus, the response parameter for this research, R, which represents the measure of how 'tactical' the behavior of the UAV is defined as

$$R \triangleq \frac{d_{avg}}{M}, \tag{3.25}$$

where d_{avg} denotes the average distance from UAV to closest obstacle over the entire flight and M is the moment of inertia of the map, obtained from (3.24). Following the experiments and simulations, the value of R will be computed for every experiment in Section 4.4.1. This will then be used to calculate the SNR and perform ANOVA to investigate the effects of varying different control parameters.

3.3.2 Analysis of variance

Several types of ANOVA can be used to investigate the effect of factors and interactions depending on the number of factors and the number of levels of the factors. ANOVA performed on the raw data identifies control factors that affect the means, whereas ANOVA performed on the SNR identifies control factors that affect the variance [6]. Based on these two ANOVAs, the control factors can be classified into 4 groups, which are as follows

- **Class 1**: Factors affecting both average and variance;
- **Class 2**: Factors affecting only variation;
- **Class 3**: Factors affecting only average;
- **Class 4**: Factors with negligible effect.

For experiments with two or more factors with two or more levels, the 'two-way ANOVA' is used. This subsection provides details on the steps used in this research to perform the two-way-ANOVA.

Sum of squares

First, we introduce the *'sum of squares'*, which provides a measure of the deviation of the responses from the average of the response. This quantity is defined as

$$S_T \triangleq \sum_{i=1}^{n}(y_i - \bar{y})^2, \tag{3.26}$$

where y_i denotes the response of the i^{th} experiment and $\bar{y}$ denotes the mean of all responses. There are alternative ways to calculate the sum of squares. For instance, the correction factor (CF) can be used to calculate the sum of squares, and it is defined as

$$CF \triangleq T^2/n, \tag{3.27}$$

where T denotes the sum of all results and n denotes the number of experiments. Using the correction factor, the sum of squares is calculated as

$$S_T \triangleq \sum_{1}^{n} y_i^2 - CF. \tag{3.28}$$

It is also necessary to calculate the sum of squares of individual factors and interactions. The sum of squares of an individual factor, in this example, Factor A, is defined as

$$S_A \triangleq \sum_{i=1}^{k} A_i^2/N_{A_i} - CF, \tag{3.29}$$

where k denotes the number of levels of Factor A, N_{A_i} denotes the number of experiments where Factor A has Level i, and A_i denotes the sum of all responses where the Factor A is at Level i. In the case where Factors A and B interact, their sum of squares is given by

$$S_{A\times B} \triangleq \sum_{i=1}^{c} [(A\times B)_i^2/N_{A\times B}] - CF - S_A - S_B, \tag{3.30}$$

where $(A \times B)_i$ represents the sum of responses of the i^{th} combination of the interactions in the OA, $N_{(A\times B)_i}$ denotes number of experiments where i^{th} combination exists, and c denotes the total number of combinations of the interacting factors at different levels. For a two-factor interaction with two levels each, the combinations would be $A_1 \times B_1$, $A_1 \times B_2$, $A_2 \times B_1$, and $A_2 \times B_2$, and thus $c = 4$. Similarly, for a two-factor interaction with 3 levels per factor, $c = 9$. In this manner, the sum of squares of all factors and interactions are calculated. For an intuitive example, consider Table 3.4, which is a three-level per factor OA. Assuming the first column is Factor A and second column is Factor B, the response $(A \times B)_1$, which is basically $A_1 \times B_1$, is calculated as the sum of the first three experiments as the entries in the first and second column denote the Level 1 for both the factors. Similarly, if one were to calculate the sum of $A_3 \times B_1$, they would add the responses of the experiments 19, 20, and 21 since for these experiments, Factor A has Level 3 and Factor B has Level 1.

Degrees of freedom

For further calculations, it is necessary to calculate the DoF of the total experiment, individual factor, and each interaction. The expression used to calculate the total DoF of the experiment is portrayed in (3.5). The expressions used to calculate the DoF of a factor and an interaction are represented in (3.1) and (3.3), respectively.

Mean squares

Mean square is a measure of the distribution of responses about the mean of the response. Once the sum of squares and DoF of each factor and interaction are calculated, the mean squares are calculated for each factor and interaction. The mean square of a factor or an interaction is defined as

$$V_A \triangleq S_A / f_A, \tag{3.31}$$

where S_A denotes the sum of squares of Factor A and f_A denotes the DoF of Factor A.

Percentage contribution

Percentage contribution represents the ability of the factor or interaction to reduce the total variation. The portion of the total variation observed in the experiment from each factor or interaction is reflected in the percentage contribution. In general, if the total of the percentage contributions of the significant factors is greater than 75%, then this is an indicator of the fact that appropriate factors have been selected for experimentation [6]. The percentage contribution for a Factor A is defined as

$$P_A \triangleq \frac{S_A}{S_T} \times 100, \tag{3.32}$$

where S_A denotes the sum of squares of Factor A and S_T denotes the total sum of squares.

F-ratio

As the number of repetitions of the experiment is increased, it is easier to predict accurate responses, but for a small number of repetitions, and considering the fact that Taguchi's method uses partial factorial experiments, it is important to verify the results. The F-ratios or variance ratios can be used for this purpose, and are defined as

$$F_A \triangleq V_A/V_e, \tag{3.33}$$

where V_e denotes the mean square of the error term, V_A denotes the mean square of Factor A. The values of the F-ratios are then compared to the standard values provided in the F-tables at various significance levels. If the F-ratio is lower than the value at the selected level of confidence in the F-table, then it does not contribute to the sum of squares at that confidence level.

Generally, if the DoF of the error term is zero, then Taguchi's method prescribes 'pooling' terms together which have low contributions. It is recommended to pool factors to account for nearly half the total DoF of the experiment. So, for an experiment with 10 DoF, factors with the lowest percent contributions will be pooled together to form an error term with approximately 5 DoF.

Estimation of results

With the help of the above analyses, the parameter combination at which the process or product has the most optimum response is identified. If this combination of level settings is one prescribed by an experiment in the OA, then the expected mean of the optimum conditions is directly calculated as a mean of repetitions of that experiment. If the required condition does not match any experiment prescribed in the OA, then the mean of that can be estimated. To calculate the estimated mean, the averages of the responses of the contributing factors and interactions, at the selected levels, are summed, and a term $k\overline{T}$ is subtracted from this, where k denotes a term which is equal to the number of factors or interactions involved in the estimation minus 1, and $\overline{T}$ denotes the average response of all experiments. For example, consider a condition, where factor levels are A_1, B_2, C_1, D_1, and Factors A and B are interacting, the estimated mean is calculated as

$$E(m)_{A_1 \times B_2 C_1 D_1} = \overline{A_1 \times B_2} + \overline{C_1} + \overline{D_1} - k\overline{T}, \tag{3.34}$$

where $\overline{A_1 \times B_2}$ represents the average of responses in case A_1 and B_2 exist together, $\overline{C_1}$ and $\overline{D_1}$ denote the average of responses where Factors C and D are at Level 1, respectively. In this case, $k = 2$, since we are considering 3 individual factors or interactions $(A \times B, C, D)$.

Column effect method

There are several other methods, which are complementary to ANOVA, to interpret the data of the experiments due to the structure of the orthogonal array. One of such methods is the *'column effects method'*. For each column (factor), the difference between the averages of the level with the highest average response and the lowest average response is calculated. This difference is directly proportional to the effect that factor has on the response. The column with the largest difference in absolute value has a stronger effect on the response parameter. A positive or negative sign conveys a positive or negative correlation of the factor with the response of the process. This method is used in conjunction with ANOVA to develop a deeper understanding of the effects of factors and interactions. In this research, the column effect is calculated using both the average response and the SNR.

Plotting analysis

Plotting analysis is a graphical method to investigate interactions among factors. The means of the response are plotted at each level setting present in the analysis. The *x-axis* contains the levels of the first interacting factor and the *y-axis* contains the average of the response or the SNR. The individual levels of the second interacting factor are represented by individual line segments. If these lines intersect, then the factors interact with each other. The magnitude of interaction is proportional to the angle between these two lines. If lines are almost parallel, then it can be assumed that there exists no interaction at any level between the two factors. Figure 3.4 has been provided as an example to aid this exposition. The average of responses for the four different conditions ($A_1B_1, A_2B_1, A_1B_2, A_2B_2$) are plotted, and the responses where Factor B is at the same level are connected by the line segment. It can be noted that in the first case, the lines representing the response do not intersect. In this condition, the factors are assumed to show no interaction. In the second case, represented on the right-hand side, the lines representing the interacting factors intersect. Thus it can be concluded that the Factors A and B are interacting in case two and not in case one.

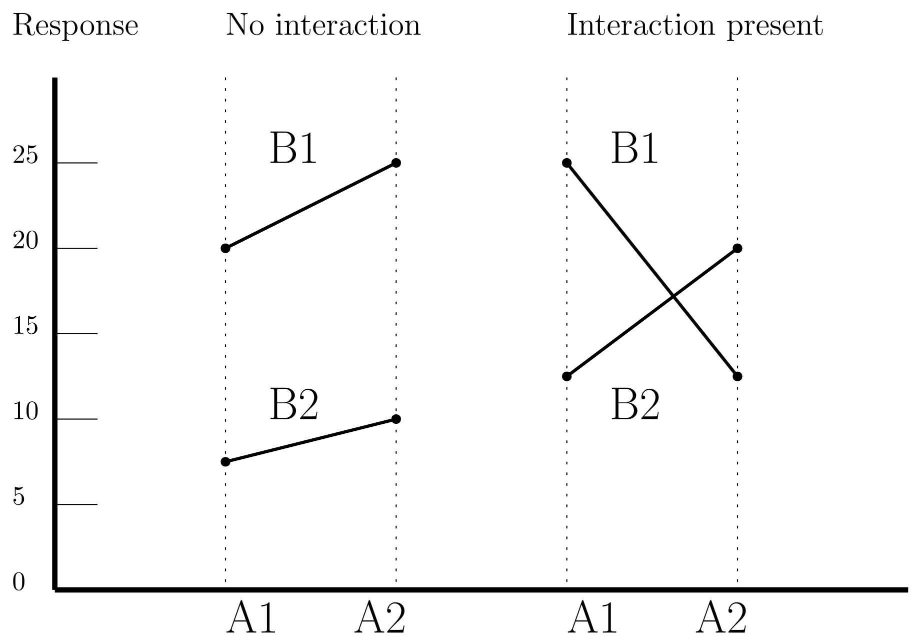

Figure 3.4: An example of plotting analysis

3.4 Conclusion

In this chapter, a detailed explanation of the Taguchi method was provided. Specifically, Section 3.1 provided various applications of Taguchi's method in the industry and presented the work undertaken by other researchers related to this domain. Subsequently, a detailed log of the process undertaken in Taguchi's method was provided. Following the guidelines stated in the methodology, the L27 OA was selected for this research and in the subsequent chapters, serves as the basis for experimentation. Following the experimentation, the methodology provided in the Section 3.3.2 will be used to analyze and interpret the results in the following chapters. As the foundation for the experimentation has been laid, the next chapter describes the process of experimentation, the different maps used for experimentation, and the results associated with these tests.

Chapter 4

Experimentation and Results of the Taguchi Analysis

In the previous chapters, the experiments for this research were designed. Additionally, the L27 OA, which has 13 columns and 3 levels for each column, has been employed. As per Taguchi's methodology, the next step in this process is conducting experiments and gathering data. This chapter provides the details of the procedures undertaken during the experimentation process. Furthermore, this chapter presents the maps used for this research, the architecture of the navigation system as portrayed in [1], and the UAV used for this research, along with an explanation of the setup used for flight tests. Subsequently, we will perform the Taguchi analysis and interpret the data gathered via simulations. Using the results of this analysis, we will deduce the parameter level settings for which the UAV exhibits a highly tactical behavior. Additionally, the results of ANOVA, performed as presented in Section 3.3.2, will be employed to obtain a quantitative estimate of the effect of each control factor on the behavior of the UAV. Lastly, to verify the behavior exhibited by the UAV in the simulations, the results of flight tests will be investigated for the presence of similar trends encountered in the simulations.

Figure 4.1: UAV used in this research.

4.1 Overview of the UAV

A UAV similar to the one used in [1] has been employed in this research, and is shown in Figure 4.1 This multi-rotor UAV has been designed in the Advanced Control Systems Lab at Virginia Tech. The length, width, and height of the UAV are $0.4m, 0.4m$, and $0.3m$ respectively. The frame used in this UAV is an *iFlight XL7 V4 True X*, which is a full 3k carbon fiber frame. The supporting components of the UAV are 3D printed in *polyethylene terephthalate glycol*. Using a high-fidelity CAD model, the mass of the UAV is found to be 2.0 kg and the principal moments of inertia are $I_x = 0.0205kg \cdot m^2$, $I_y = 0.0143kg \cdot m^2$, and $I_z = 0.0281kg \cdot m^2$. The *Intel NUC 7i7DNBE* single-board computer is used by the UAV to implement the GNC system. This single-board computer has a 4.20GHz *Intel i7-8650* processor and 4 GB RAM. This tactical GNC system is coded in C++ on the Ubuntu 18.04 operating system. The *Intel RealSense D435i* camera serves as the depth camera and has an 86.00° horizontal field of view and a 57.00° vertical field of view. For tracking, an *Intel*

RealSense T265 camera is used, which has a 69.40° horizontal field of view and a 42.50° vertical field of view. Both these cameras are equipped with *BMI055* inertia measurement units. The *Pixhawk PX4* autopilot serves as an inertia measurement unit, controls the propellers, and has an ST Micro L3GD20H gyroscope on board. The autopilot communicates with the single-board computer using a USB FTDI serial line. The motors used are *AirA1*, 200kV motors, on which 7 × 4.5 dual-blade propellers are mounted.

As inferred from [1], the tactical guidance system described in Chapter 2 is supported by a vision-based navigation system. This navigation system is capable of detecting obstacles, creating a binary occupancy map, and localizing the UAV relative to the environment. A set of stereo depth and tracking cameras detect obstacles and track the position of the UAV with respect to an initial position provided by the user. The tracking camera can also estimate the UAV's attitude, angular displacement, and transnational velocity with respect to the inertial reference frame $\mathcal{I}$. Finally, the environment is captured by a voxel map. If the probability of a voxel to be occupied is higher than some user-defined threshold, then it is marked as occupied. Hence, a binary occupancy map is employed by the guidance system to outline reference trajectories.

The architecture of the software underlying this tactical GNC system, as described in [1], is shown in Figure 4.2. The 'flight stack' is central to the implementation of this GNC system. The navigation system provides estimates of the UAV's position and yaw angle to the flight stack at approximately 20 to 60 Hz. Concurrently, the Pixhawk autopilot also provides the estimates of the UAV's pitch angle, roll angle, translational velocity, and angular velocity to the flight stack. The estimated accuracy of the attitude, angular velocity, and angular acceleration are $\pm 0.05°$, $\pm 0.004°/s$, and $\pm 7°/s^2$ respectively. Using estimates of the obstacles' positions and the position of the UAV provided by the navigation system, the path planning subsystem generates a reference path from the UAV's position to the goal point

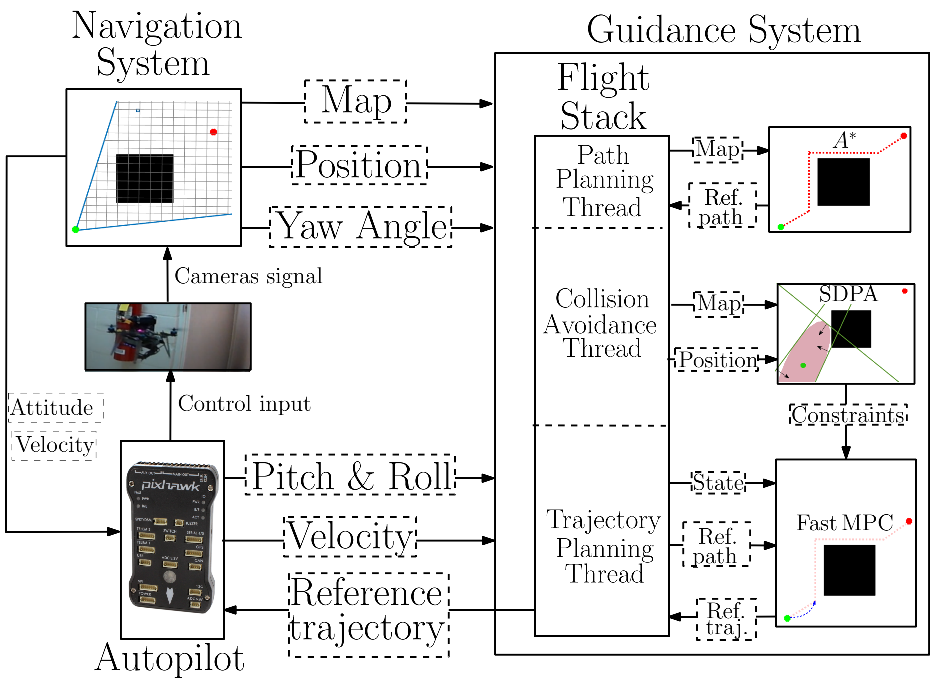

Figure 4.2: Software architecture of the tactical GNC system as shown in [1].

of the mission. As stated in Section 2.1.3, the number of voxels scanned while generating paths depend on the user-defined parameters μ_1, μ_2, and μ_3, thus altering the rate at which reference paths are generated (between 0.5 and 20 Hz). The semi-definite programming algorithm (SDPA) used by the collision avoidance subsystem, with the knowledge of the obstacles' positions, generates a constraint set for collision avoidance, that is subsequently used while generating reference trajectories. Reference trajectories are generated by the trajectory planning subsystem at around 50 Hz and are sent to the Pixhawk at a rate of $(n_t \Delta T)^{-1}$. With the knowledge of the state of the UAV and the reference trajectory provided by the trajectory planning subsystem, the Pixhawk autopilot uses a PID controller to actuate the motors.

4.2 Flight Maps and Simulations

4.2.1 Simulations

In this research, we investigate the effect of the obstacle distribution, which is considered as the noise factor, by simulating the behavior of the UAV over three different maps with varying obstacle densities and configurations. This approach is analogous to using an outer array, as explained in Section 3.2.3. Therefore, a total of 81 experiments are required to complete this study (as the L27 OA prescribes 27 experiments per map). In this section, we explain how the Taguchi method is applied in a software-in-the-loop simulation environment.

To gather data for this research, the experiments were performed through software-in-the-loop simulations using an *Intel NUC 7i7DNBE*. The parameter levels to simulate each experiment are read by the flight stack through two different text files. It is cumbersome to manually change these values in two different text files for the 81 simulations. To overcome this, the L27 OA was generated in MATLAB, with the values of the parameter level settings as per Table 3.6. This OA was then used to generate the two-parameter text files required according to each experiment. The C++ codes were modified to receive an input, that specified which experiment was being simulated. According to the input, the specific parameter files were read, thus expediting the simulation process and avoiding manual error.

4.2.2 Maps Employed in Simulations

To perform the simulations mentioned in the previous subsection, voxel maps of the different environments were generated in MATLAB. The dimensions and layout of the testing facility at the Advanced Control Systems Lab at Virginia Tech served as a basis for these maps. By adding a unique obstacle set, each with significantly different obstacle density, positioning,

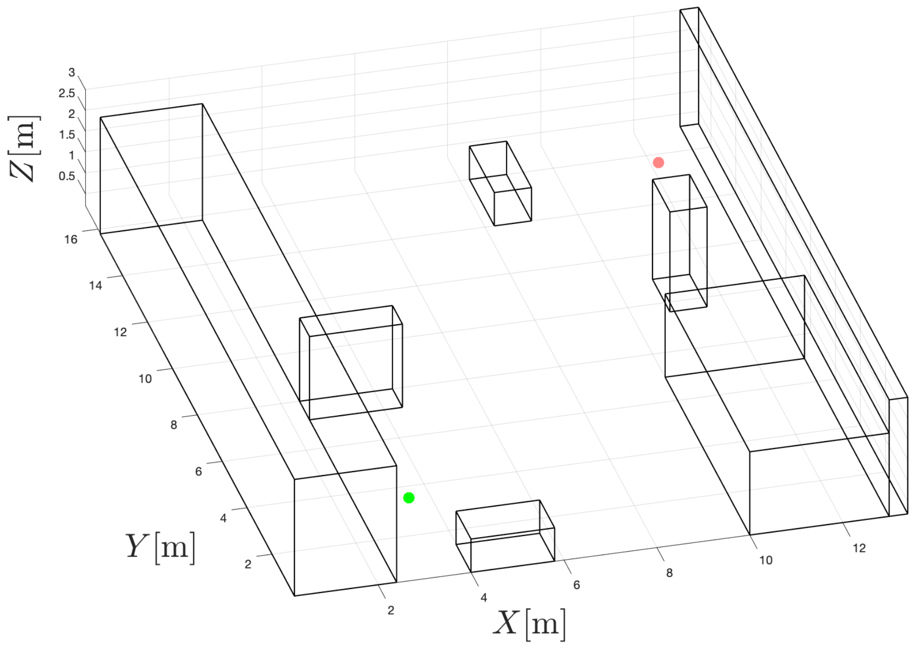

Figure 4.3: Obstacle configuration used in Map 1

and scattering, three maps with a varying moment of inertia were generated. The obstacle configurations used in this research can be seen in Figures 4.3, 4.4, and 4.5. By visually inspecting the different obstacle configurations, it can be observed that Map 1, shown in Figure 4.3, has the lowest number of obstacles and the lowest obstacle density among the three maps. Map 2, shown in Figure 4.4, represents a configuration that has a higher obstacle density and the obstacles are relatively more scattered than Map 1. It can also be observed that Map 3, shown in Figure 4.5, has the highest obstacle density, and has obstacles scattered all over the footprint of the map.

Following the generation of these maps, their moment of inertia was calculated according to the procedure presented in Section 3.3.1. The inertia matrices of each obstacle were calculated and the sum of the Frobenius norms of these inertia matrices was obtained to calculate the moment of inertia of the map, as per (3.24). Table 4.1 provides the resulting

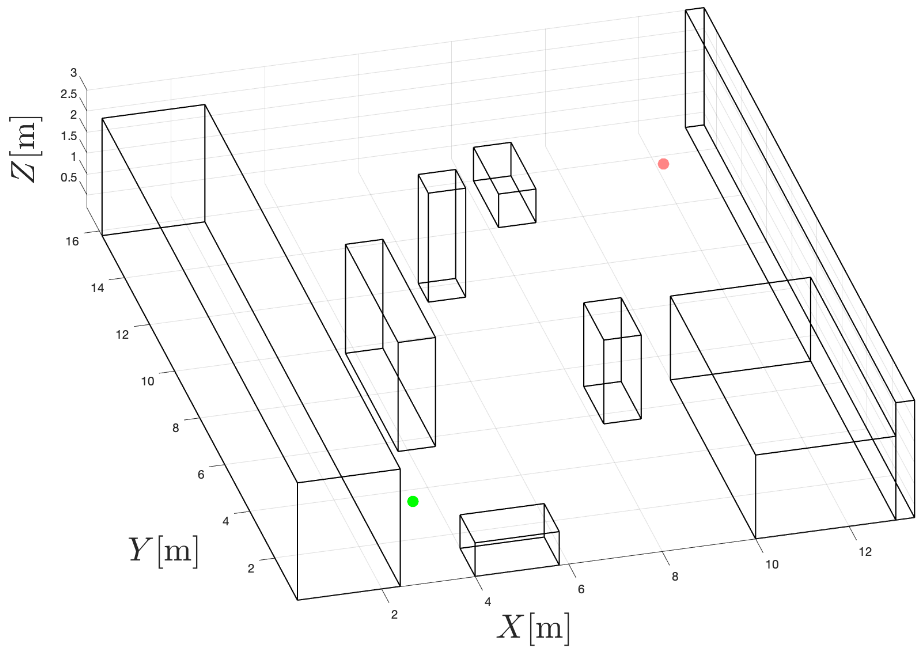

Figure 4.4: Obstacle configuration used in Map 2

moment of inertia for these maps, and shows that Map 1 has the lowest moment of inertia. Map 2 and Map 3 have a significantly larger moment of inertia, with Map 3 having the largest value.

Subsequently, for the simulations performed on these maps, the initial position of the UAV was set at $(1, 2.2, 1)m$ in (x, y, z) directions, and it is represented by the green marker in Figures 4.3, 4.4, and 4.5. The goal for these missions is set at $(12, 15, 0.6)m$, and it is represented by the orange marker in Figures 4.3, 4.4, and 4.5. The start and goal points are kept same for all the maps to observe the difference in behavior for similar missions as environment varies.

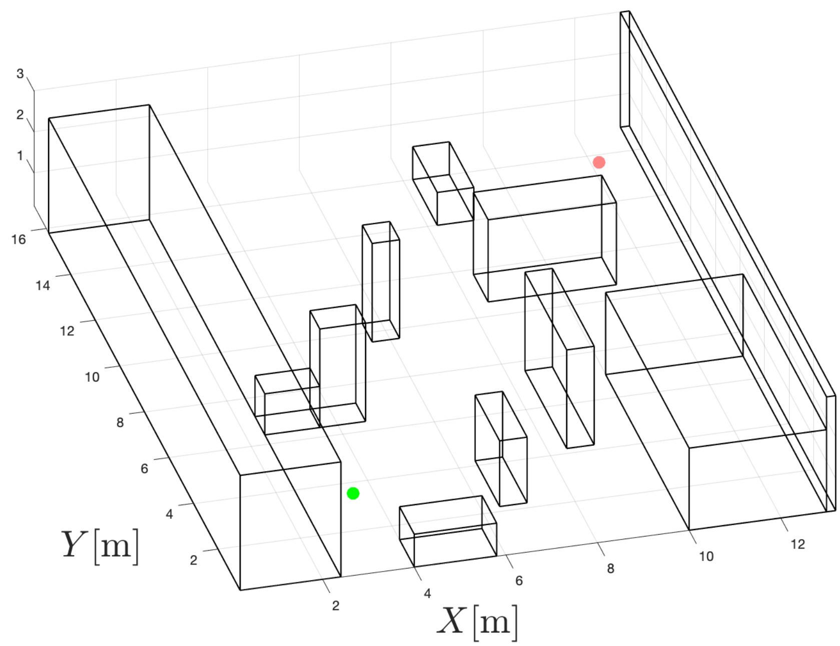

Figure 4.5: Obstacle configuration used in Map 3

	Map 1	Map 2	Map 3
Sum of Frobenius norms of individual obstacles	21613.043	35558.939	66178.060
Frobenius norm of the entire map as one entity	170395.8	174995.9	204234.5
Moment of inertia of map	0.1268	0.2031	0.3240

Table 4.1: Moment of inertia of maps

4.3 Flight Tests

Following the simulations, we perform flight tests for a few selected experiments to confirm if similar trends are observed in the actual flights. The maps employed for the Taguchi analysis through software-in-the-loop simulations are reproduced at the Advanced Control Systems Lab at Virginia Tech, and flight tests are performed. In this thesis, we present the results of 4 flight tests, whose goal is to validate the realism of the numerical simulations employed for the Taguchi analysis. Flights for Experiments 5, 15, 21, and 26, from Table 3.6, were performed over the three maps shown in Figures 4.3, 4.4, 4.5, using the UAV described in

Section 4.1. To calculate the average distance of the UAV to the closest obstacles for these flight tests, maps were recorded with the help of the navigation system, as the UAV uses this same navigation system to localize and detect obstacles. The start and goal positions were kept the same as the simulations to maintain uniformity in the mission and to analyze differences, if any, between the simulations and flight tests. Whenever possible, multiple repetitions of each flight test were done to get a larger data set to analyze the behavior. The flight tests were completely autonomous from take-off to landing, and the coordinates of the position of the UAV at each time step were recorded.

4.4 Simulation Results

4.4.1 Results of Taguchi analysis

This subsection presents the results of the data acquired through the simulations performed in Section 4.2. Figures 4.6 and 4.7 show the different paths taken by the UAV over Map 1. Similarly, Figures 4.8 and 4.9 represent the paths taken by the UAV over Map 2, and Figures 4.10 and 4.11 represent the paths taken by the UAV over Map 3. Table 4.2 presents the results of these simulations. The average distance from UAV to the closest obstacle for Maps 1, 2, and 3 are represented by d_{avg_1}, d_{avg_2}, and d_{avg_3}, respectively, expressed in meters. This table also shows the resulting response parameter R, calculated by applying (3.25), and SNR, calculated using (3.11), for each experiment. Subsequently, the averages of R and SNR for each factor, at each level are calculated, as explained in Section 3.3.2, and presented in Tables 4.3 and 4.4.

To begin the analysis of these results, we select the experiment with the highest SNR from Table 4.2, as it provides the optimal parameter combination from the given set of exper-

Exp no.	d_{avg_1}	d_{avg_2}	d_{avg_3}	R_1	R_2	R_3	SNR
1	1.7738	1.6579	0.9801	13.9845	8.1596	3.0251	-19.563
2	2.0022	1.7101	0.9993	15.7853	8.4165	3.0841	-20.408
3	2.0303	1.7240	1.0145	16.0071	8.4847	3.131	-20.518
4	1.8938	1.6915	1.0205	14.9309	8.325	3.1497	-20.031
5	1.9140	1.7992	1.0460	15.0898	8.8548	3.2285	-20.233
6	1.9623	1.6686	0.9785	15.4708	8.212	3.0201	-20.224
7	2.0662	1.7177	0.9754	16.2896	8.4535	3.0106	-20.618
8	1.9885	1.6602	0.9164	15.6771	8.1708	2.8283	-20.288
9	1.9156	1.5895	1.0579	15.1026	7.8225	3.2652	-19.999
10	1.9376	1.6569	0.9025	15.2759	8.1547	2.7856	-20.109
11	1.8822	1.7202	1.0254	14.8395	8.4658	3.1649	-20.027
12	2.1563	1.5061	1.0010	17.0000	7.4123	3.0864	-20.712
13	1.9202	1.5746	1.1109	15.1389	7.7493	3.4288	-20.014
14	2.1294	1.5465	0.9736	16.7882	7.6111	3.0049	-20.655
15	1.8872	1.6570	0.8544	14.8786	8.1549	2.637	-19.925
16	2.1851	1.5522	1.0299	17.2274	7.6391	3.1787	-20.855
17	2.0023	1.6712	0.9166	15.7861	8.2247	2.8290	-20.346
18	1.8087	1.5049	1.0757	14.2600	7.4064	3.3201	-19.530
19	1.0001	1.3740	1.4639	7.8840	6.7623	4.5182	-16.311
20	1.192	1.1259	1.0691	9.3975	5.5410	3.2997	-16.365
21	1.8511	1.4539	1.3251	14.5939	7.1553	4.0897	-19.714
22	1.5005	1.3506	1.2125	11.8300	6.6468	3.7422	-18.198
23	1.7759	1.7311	1.6474	14.0012	8.5197	5.0845	-19.919
24	1.1899	1.1230	1.1210	9.3809	5.5271	3.4598	-16.386
25	1.2979	1.1358	1.1038	10.2326	5.5897	3.4069	-16.918
26	1.4476	1.2683	2.6202	11.4128	6.2419	8.0869	-18.932
27	1.6191	1.6254	1.4578	12.7652	7.9994	4.4994	-19.159

Table 4.2: Results of 27 numerical simulations. The columns denoted by d_{avg} show the average distance to the closest obstacle, and the columns denoted by R show the response over the three maps, respectively. The experiment with the largest SNR (most tactical), Experiment 19, has been highlighted in green. Experiment 16, which has the smallest SNR (most reckless), has been highlighted in red. It can be seen that, from Experiment 19 onward, larger SNR values are observed. This can be chalked down to the fact that all these experiments have a high value of μ_2. We can also see a generally decreasing trend in d_{avg} values from Map 1 to Map 3, since Map 1 has the smallest obstacle density.

iments. For this design of experiments, Experiment 19 provides the highest SNR (-16.311 dB), and thus displays the most tactical behavior from the set of experiments prescribed by the L27 OA. As explained in Chapter 3, OAs provided by Taguchi are partial factorial OAs, and there is a possibility that the optimal combination of level settings might not be one of the experiments provided in the L27 OA. Therefore, we deduce a parameter combination, that is expected to provide the most tactical behavior. For individual factors ($\mu_1, \mu_3, \tilde{R}_r, \tilde{R}_u, T$, and ν_4), the level that shows the highest SNR in Figure 4.13 is selected. For the interactions $\mu_2 \times \mu_4$, the largest SNR can be observed when μ_2 is at Level 3 and μ_4 is at Level 1, and for $\mu_2 \times \mu_5$, the largest SNR is observed when μ_2 is at Level 3, and μ_5 is at Level 1, as observed in Figure 4.15. The optimal parameter combination thus deduced, can be seen in Table 4.6, along with a comparison of the parameter settings used in Experiment 19.

Level	μ_2	μ_4	μ_5	μ_1	μ_3	$\tilde{R}_r$	$\tilde{R}$	T	ν_4
1	4.950	4.658	4.602	4.650	4.711	4.776	4.808	4.477	4.885
2	4.903	4.747	4.907	4.667	4.804	4.766	4.517	4.906	4.899
3	4.399	4.847	4.743	4.936	4.738	4.711	4.927	4.870	4.469
Delta	0.551	0.189	0.305	0.286	0.093	0.065	0.410	0.429	0.431
Rank	1	7	5	6	8	9	4	3	2

Table 4.3: Average R for each level

Level	μ_2	μ_4	μ_5	μ_1	μ_3	$\tilde{R}_r$	$\tilde{R}_u$	T	ν_4
1	-14.45	-13.75	-13.64	-13.73	-13.91	-13.97	-14.01	-13.36	-14.23
2	-14.44	-13.91	-14.23	-13.75	-14.07	-13.97	-13.47	-14.31	-14.22
3	-12.93	-14.15	-13.95	-14.34	-13.84	-13.88	-14.35	-14.16	-13.37
Delta	1.53	0.40	0.59	0.61	0.22	0.10	0.88	0.95	0.86
Rank	1	7	6	5	8	9	3	2	4

Table 4.4: Average SNR for each level

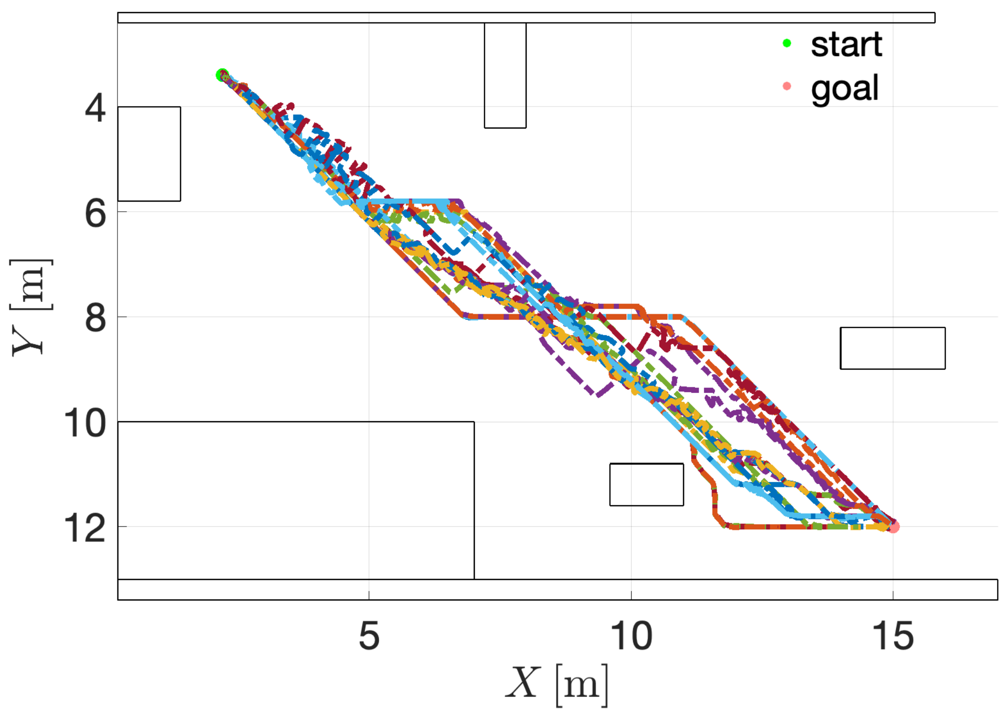

Figure 4.6: Top view of the simulations on Map 1

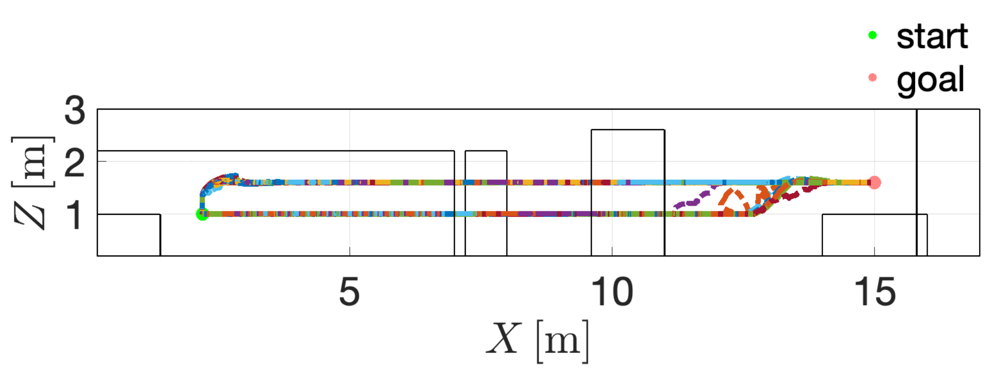

Figure 4.7: Side view of the simulations on Map 1

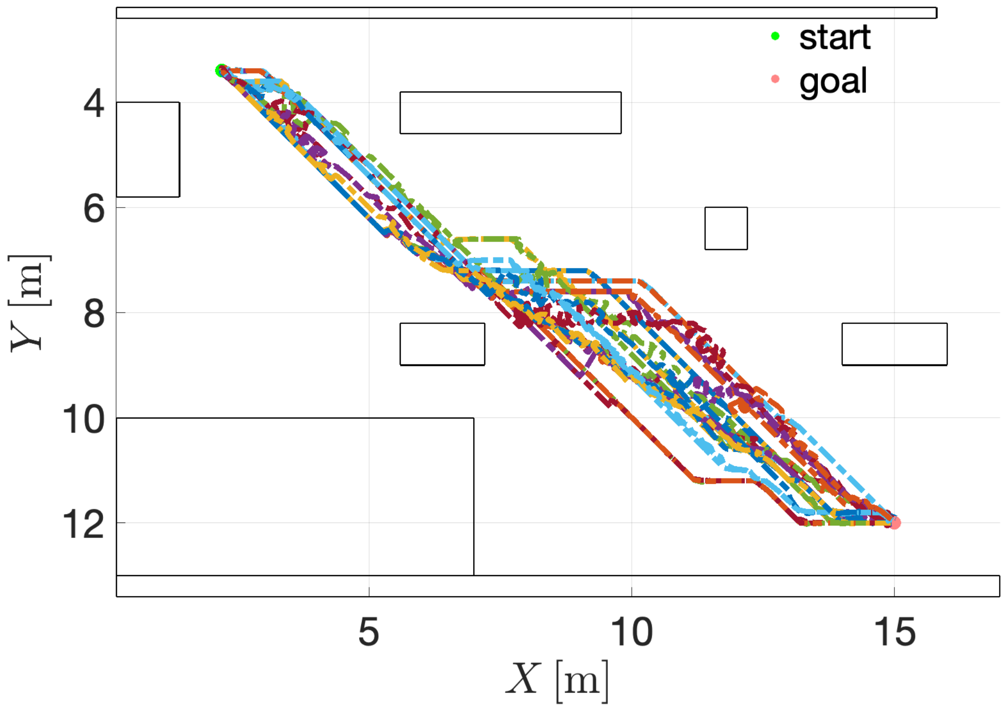

Figure 4.8: Top view of the simulations on Map 2

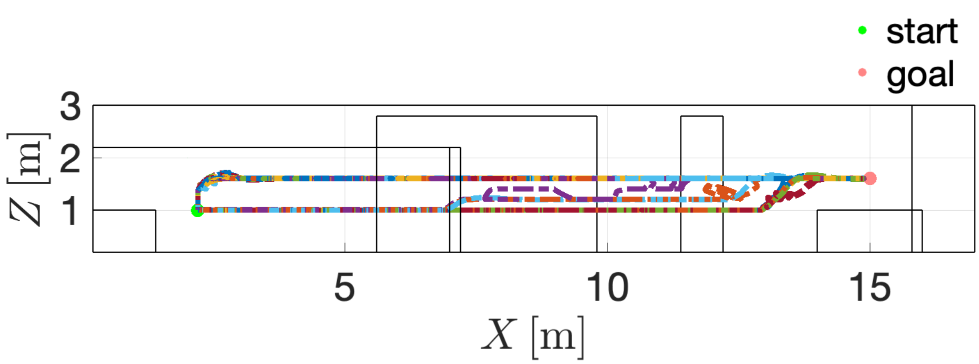

Figure 4.9: Side view of the simulations on Map 2

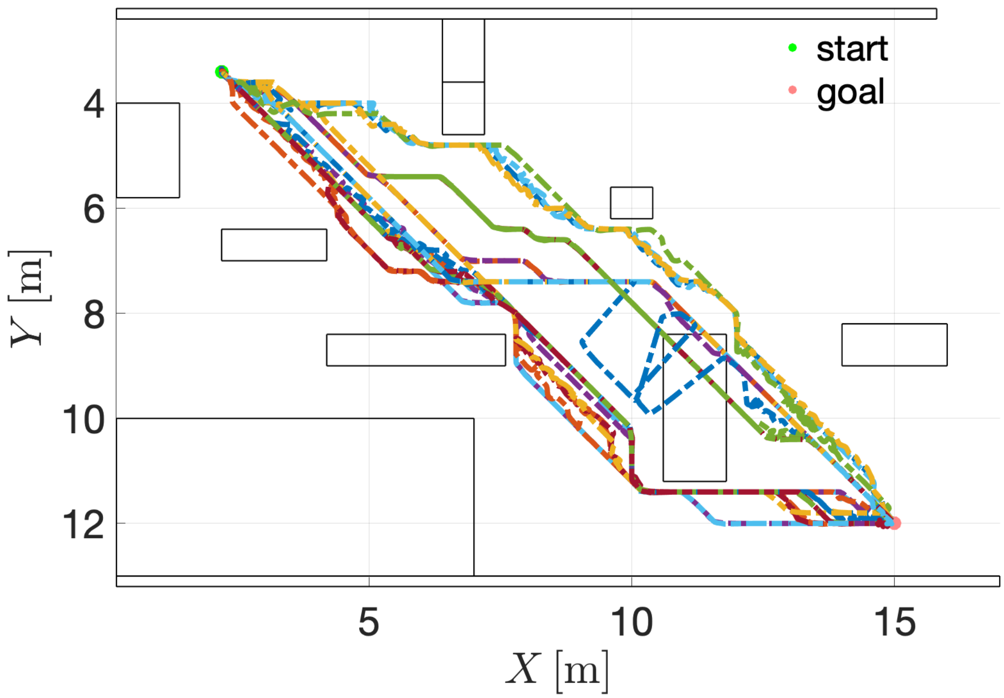

Figure 4.10: Top view of the simulations on Map 3

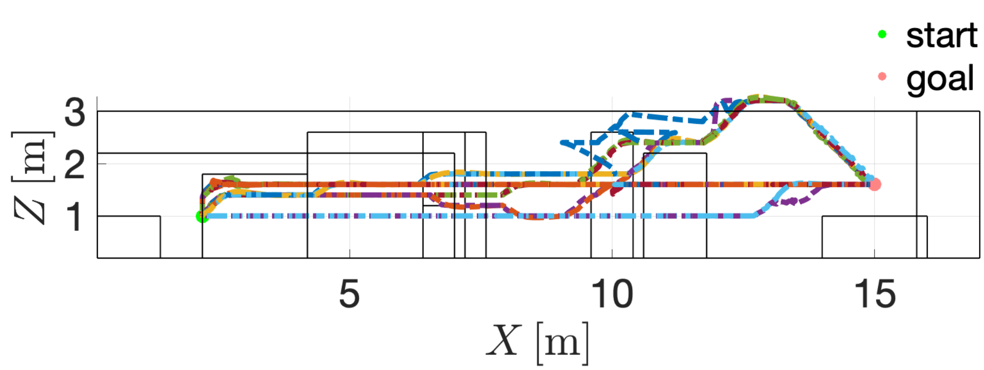

Figure 4.11: Side view of the simulations on Map 3

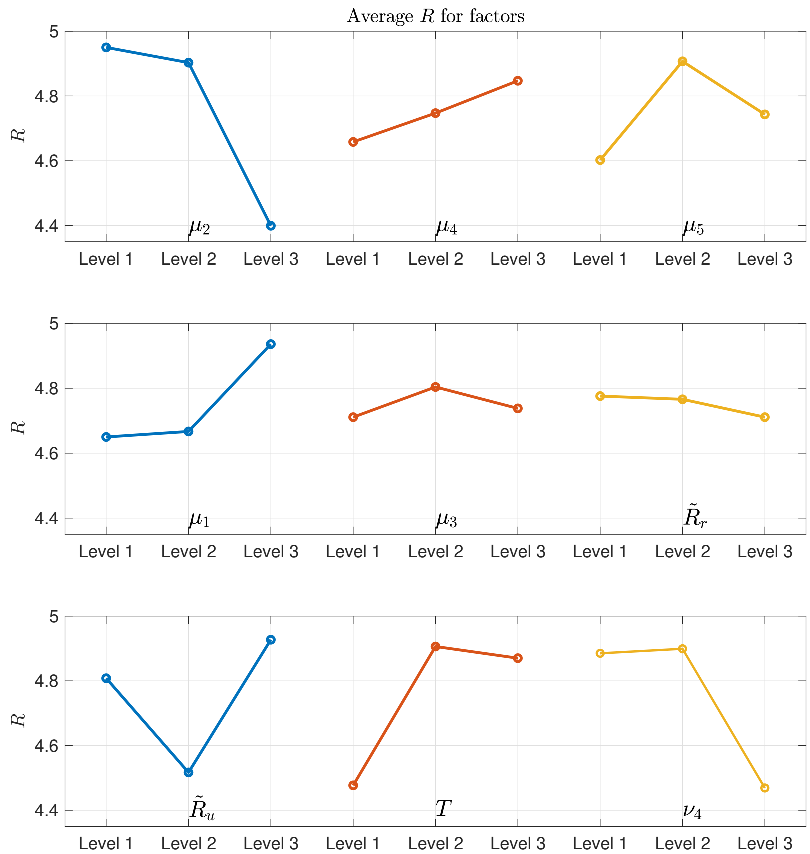

Figure 4.12: Averages of R per level

Subsequently, using this optimal level-set, the confirmation experiments are performed to verify if the deduced level-set, is in fact, the most tactical. A comparison of the simulations performed using this combination, deduced from the Taguchi analysis, with Experiment 19, the most tactical of the 27 experiments performed, is shown in Table 4.5. It can be seen that the optimal level-set deduced from the Taguchi analysis provides a marginally larger SNR (-16.163 dB), than that of Experiment 19 (-16.311 dB), and thus a more tactical behavior. As explained in Section 3.3, the parameter combination that results in the highest SNR is also the most resistant to external noise, which in this research is the obstacles' configuration. We

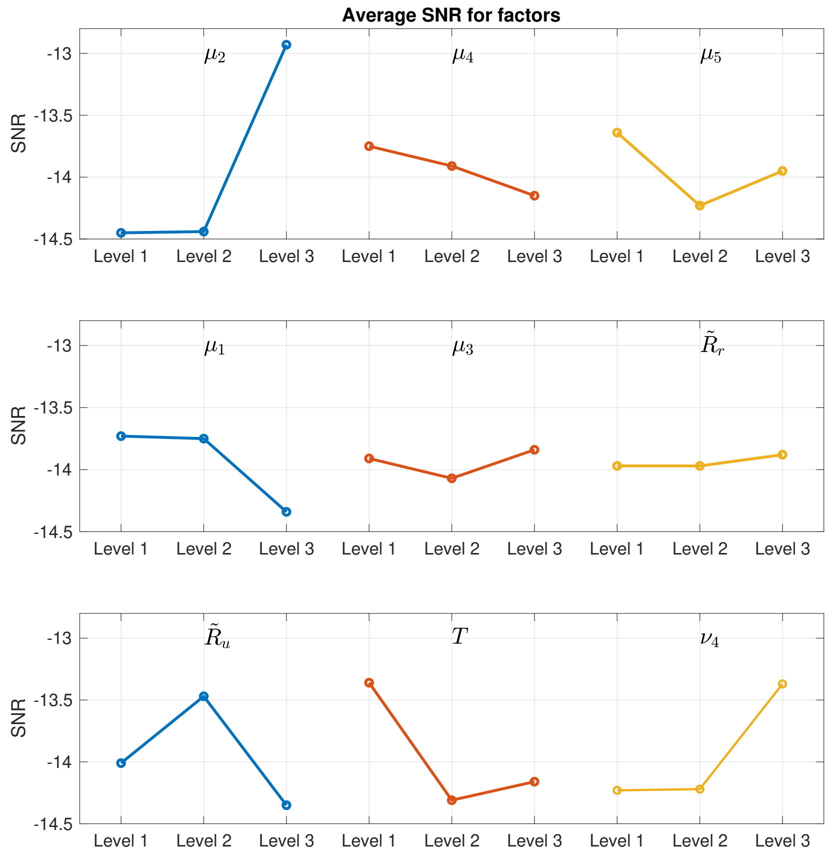

Figure 4.13: Averages of SNR per level

can thus conclude, that with the help of Taguchi analysis that the level-settings portrayed in Table 4.6 provide the most tactical behavior, and are the least influenced by the obstacles' configuration.

Further investigating the optimal level-settings seen in Table 4.6, the path planner parameters, μ_1, μ_2, and μ_3, follow the trends forecasted in Section 2.1.3. The user-defined parameters, μ_2 and μ_3 are at Level 3, and μ_1 is at Level 1. For a higher value of μ_2, the attractive effect of the obstacles is increased and the path planning algorithm scans a higher number of voxels, thus achieving a more tactical behavior. Since the lowest level-setting of

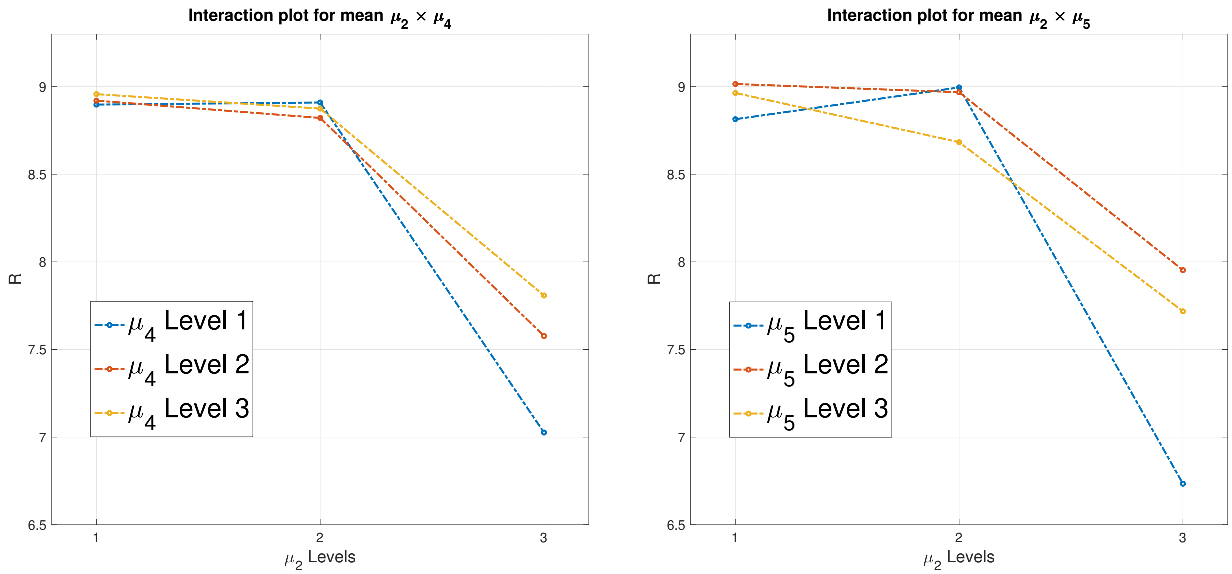

Figure 4.14: Interaction plots for average R

Level-set	d_{avg_1}	d_{avg_2}	d_{avg_3}	R_1	R_2	R_3	SNR
Experiment 19	1.0000	1.3740	1.4639	7.8840	6.7623	4.5182	-16.311
Optimal settings	1.1654	1.1056	0.9833	9.1879	5.4412	3.03487	-16.136

Table 4.5: Comparison of results of Experiment 19 and confirmation experiment

μ_1 and the highest level-setting of μ_3 are used, the lowest possible value for μ_1/μ_3, for this particular study, is obtained. Consequently, paths that are closer to obstacles are generated, which results in a more tactical behavior.

While investigating the results of the trajectory planner parameters, we find that the user-defined parameter μ_4 is at Level 1, the lowest level setting. This result follows the expectations stated in Section 2.1.4, since minimizing μ_4 gives a higher priority to coasting obstacles. Level 1, the lowest level setting, is also observed to be optimal for μ_5, which confirms that the attractive effect of obstacles that are at a greater distance from the UAV will also be considered while generating trajectories between two waypoints, thus providing a more tactical behavior.

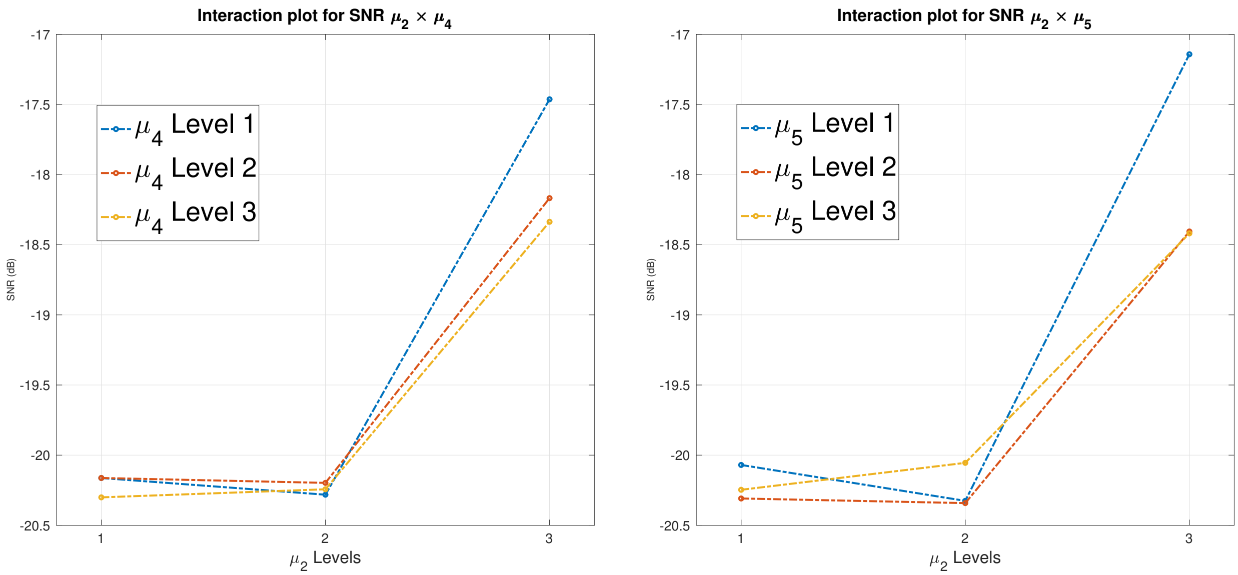

Figure 4.15: Interaction plots for average SNR

We can also confirm the most suitable coefficients for the optimal control problem by observing at the levels of $\tilde{R}_r$ and $\tilde{R}_u$. It can be observed that a higher penalty is associated with the trajectory tracking error, than the penalty on the control input. This implies that the trajectories generated by the UAV's trajectory planner provide a highly tactical behavior, and as the deviation increases from these trajectories, the UAV tends to be more reckless. Furthermore, the UAV displays a tactical behavior when T is at Level 1, that is, when the MPC algorithm generates references for only 10 time steps. Lastly, the optimal level for ν_4 is found to be Level 3. A high value of ν_4 helps in a better approximation of the obstacles' set in the trajectory planner, and thus trajectories closer to the obstacles' set can be achieved.

Lastly, we compare the level settings used in Experiment 16, the trial with the smallest SNR (-20.855 dB), with the optimal level settings. For the path planner subsystem, the parameter μ_1 is at Level 3, and μ_3 is at Level 1. This results in the largest possible value of μ_1/μ_3, for this study, thus inducing reckless behavior. It is interesting to note that for the reckless behavior displayed by Experiment 16, μ_2 is at Level 2, and not at Level 1, which is

	Exp. 19		Optimal set	
Factor	Level	Value	Level	Value
μ_2	3	0.95	3	0.95
μ_4	1	0.2	1	0.2
μ_5	1	0.1	1	0.1
μ_1	1	0.01	1	0.01
μ_3	3	5	3	5
$\tilde{R}_r$	2	700	3	1000
$\tilde{R}_u$	1	100	2	300
T	3	100	1	10
ν_4	1	100	3	1000

Table 4.6: Comparison of parameter settings of Experiment 19 and optimal level-settings

supposed to generate highly reckless paths. The trajectory planner parameter μ_4 uses Level 3, which confers a highly reckless behavior. A peculiar observation is that μ_5 uses Level 1, which is supposed to confer a tactical behavior, and thus, an anomaly to our expectations. This reckless parameter set also uses Level 1 for ν_4, thus providing an inferior estimate of the obstacles' set. The MPC algorithm generates trajectories for 100-time steps, as opposed to 10 for the optimal parameter settings. Using this information, we can conclude that a more tactical behavior is observed when the MPC algorithm generates trajectories for a smaller number of time steps. This conclusion can be further strengthened by inspecting the SNR plot of T, which loosely shows an inverse proportionality between T and SNR. Lastly, Experiment 16 uses the same level setting for $\tilde{R}_r$ as Experiment 19 (Level 3). Considering the fact that the most tactical, and the most reckless experiments, both have the same value of $\tilde{R}_r$, suggests that it has a small degree of influence on the response of this system. This can be further confirmed by observing the SNR plot in Figure 4.13, which shows no significant differences in SNR over three levels.

4.4.2 Results of plotting analysis and column effect method

To study the *column effect method*, as described in Section 3.3.2, the difference between the response of the level with the highest and lowest average R and SNR is portrayed in Tables 4.3 and 4.4, respectively. The last row of these tables assigns a rank to the factors to ascertain a comparative degree of influence on R and SNR, based on the *column effect method*.

By observing the ranks assigned in these tables, it can be concluded that the path planner parameter μ_2, is the highest contributing factor towards both, R and SNR. This can be further observed by analyzing Figure 4.13, where μ_2 encounters a large spike in average SNR at Level 3, as compared to the first two levels. Furthermore, it can be seen that factors $\nu_4, T, \tilde{R}_u, \mu_5$, and μ_1, also have a significantly large influence on R and SNR. Analyzing their respective SNR plots shows considerable differences in response at different levels. The user-defined factors μ_3, μ_4, and $\tilde{R}_r$ show a relatively small influence on the responses of the system. This result can again be observed in Figure 4.13, where the SNR of these factors does not vary significantly over the three levels. The column effect method further confirms the hypothesis stated in the previous subsection, that $\tilde{R}_r$ has a low contribution toward the response R and SNR (ranked last for both). To obtain a quantitative estimation of the factor's influence on the response of the system, their percentage contributions are calculated using ANOVA in the subsequent section.

Lastly, to perform a *plotting analysis*, as described in Section 3.3.2, a graphical representation of the data provided in Tables 4.3 and 4.4 is portrayed in Figures 4.12 and 4.13, respectively. Similarly, Figures 4.14 and 4.15 are used to support the analysis of the interactions $\mu_2 \times \mu_4$ and $\mu_2 \times \mu_5$. The primary intention of using plotting analysis, in this research, is to investigate the presence of interactions, and the relative degree of interaction for $\mu_2 \times \mu_4$ and $\mu_2 \times \mu_5$.

By analyzing the interaction plot representing $\mu_2 \times \mu_4$, in Figure 4.15, it can be observed that the line segments representing Level 1 and Level 3 of μ_4, intersect between Levels 1 and 2, and again between Levels 2 and 3 of the user-defined parameter μ_2. Thus, there exists an interaction between μ_2 and μ_4. Similarly, inspecting the interaction plot representing $\mu_2 \times \mu_5$ in Figure 4.15, the intersection of the line segments representing Level 1 and Level 3 of μ_5, can be observed between Levels 1 and 2, and between Levels 2 and 3 of the user-defined parameter μ_2, thus confirming the presence of an interaction between μ_2 and μ_5. Furthermore, by inspecting the angle formed by these intersecting segments, we can confirm that there exists a significantly larger interaction between μ_2 and μ_5, as compared to μ_2 and μ_4.

Thus, from the plotting analysis, we can conclude, that the effect of μ_4 and μ_5 on the response R and SNR, depends on the level-setting of the user-defined parameter μ_2, and vice versa. Finally, the effect of μ_5 has a higher degree of co-relation with μ_2, than the co-relation observed between μ_2 and μ_4.

4.5 ANOVA Results

In this section, we apply the methodology explained in Section 3.3.2, to perform ANOVA and investigate the effects of different control parameters on the response of the system. To begin, the sum of squares, percentage contributions, and mean squares are calculated for all control parameters and the interactions. Subsequently, to validate these results, we calculate the F-ratios and compare them to the 95% confidence threshold. As stated in Section 3.3.2, ANOVA performed on the SNR provides information on factors affecting the variance, whereas ANOVA performed on the response R, provides details on factors affecting

Factors	Sum of Squares	Mean Squares	% contributions
μ_2	30.002	15.001	59.433
μ_1	4.029	2.014	7.981
T	3.914	1.957	7.754
ν_4	3.409	1.705	6.753
$\tilde{R}_u$	2.749	1.374	5.445
$\mu_2 \times \mu_5$	2.201	0.681	4.361
μ_3	1.361	0.636	2.696
μ_5	1.272	0.55	2.52
$\mu_2 \times \mu_4$	0.852	0.242	1.687
μ_4	0.484	0.213	0.959
$\tilde{R}_r$	0.207	0.103	0.409

Table 4.7: ANOVA table based on SNR

the mean of the response. In this research, ANOVA has been performed on both, R and SNR, to study the effects of control factors on the behavior of the UAV.

Inferring from the results of ANOVA performed on SNR, as displayed in Table 4.7, factors $\mu_2, \mu_1, T, \nu_4, \tilde{R}_u$, and the interaction $\mu_2 \times \mu_5$ have a comparatively large effect on reducing variance in the SNR, as conveyed by their percentage contributions. The path planner factor μ_2, has an overwhelmingly large effect on SNR (59%), as compared to the other factors. This result validates the hypothesis drawn from the *column effect method* in Table 4.4, which stated μ_2 was the highest contributing factor to SNR. Furthermore, the control parameters $\mu_5, \mu_3, \mu_4, \tilde{R}_r$, and the interaction $\mu_2 \times \mu_4$ are found to have a relatively small effect on SNR. These results are again, in accordance with the results of the *column effect method*. Subsequently, these inferences drawn from the percentage contributions are further verified by analyzing the corresponding F-ratios.

In this research, since the DoF of the error term is zero, we pool factors with the lowest percentage contributions to form the error term. As per Section 3.3.2, Taguchi prescribes to form an error term, that has approximately half the DoF of the selected OA. Thus, in this

Factor / interaction	DoF	F-ratio	95% confidence threshold	Contributing factor
μ_2	2	43.107	3.89	Yes
$\mu_2 \times \mu_5$	4	1.581	3.26	No
μ_1	2	5.788	3.89	Yes
$\tilde{R}_u$	2	3.949	3.89	Yes
T	2	5.624	3.89	Yes
ν_4	2	4.898	3.89	Yes

Table 4.8: F-ratios of contributing factors based on SNR

research, we pool the least contributing factors till we obtain an error term, that has 12 DoF (the selected OA has 26 DoF). According to the percentage contributions shown in Table 4.7, $\tilde{R}_r, \mu_4, \mu_2 \times \mu_4, \mu_5$, and μ_3 are pooled to form an error term with 12 DoF. The sum of squares of error is equal to the total sum of squares of these pooled terms. Subsequently, the mean square of the error term is calculated as

$$V_e \triangleq S_e/f_e, \tag{4.1}$$

where S_e represents the sum of squares of the error term and f_e represents the DoF of the error term.

Finally, the F-ratios of the un-pooled terms, also referred to as the contributing terms, are calculated using (3.33), and presented in Table 4.8. The factors or interactions with an F-ratio higher than the threshold value at 95% confidence level, are considered as significantly contributing to the SNR. It is observed that all the un-pooled factors, apart from $\mu_2 \times \mu_4$, have an F-ratio larger than the threshold. Thus, μ_2, μ_1, ν_4, and T are considered significantly contributing to the SNR of the system, in turn verifying the results of the column effect method and the percentage contributions.

Finally, ANOVA is performed on the response parameter R to investigate the relative

Factors	Sum of squares	Mean squares	% contribution
μ_2	12.243	6.121	46.868
T	2.830	1.415	10.835
ν_4	2.787	1.394	10.671
μ_1	2.212	1.106	8.467
$\tilde{R}_u$	1.929	0.964	7.383
$\mu_2 \times \mu_5$	1.685	0.520	6.452
μ_5	1.041	0.421	3.985
$\mu_2 \times \mu_4$	0.649	0.182	2.483
μ_3	0.365	0.163	1.397
μ_4	0.327	0.162	1.25
$\tilde{R}_r$	0.054	0.027	0.207

Table 4.9: ANOVA table based on R

Factor / interaction	DoF	F-ratio	95 % confidence threshold	Contributing factor
μ_2	2	30.164	3.89	Yes
μ_1	2	5.449	3.89	Yes
T	2	6.973	3.89	Yes
$\tilde{R}_u$	2	4.751	3.89	Yes
ν_4	2	6.868	3.89	Yes
$\mu_2 \times \mu_5$	4	2.076	3.26	No

Table 4.10: F-ratios of contributing factors based on R

influence of factors on R and the mean of the response, and the results are presented in Table 4.9. Similar to SNR, μ_2 remains the highest contributing factor toward R. It can be observed that μ_1, ν_4, and T have significantly large contributions toward the response. The user-defined parameters $\mu_3, \mu_4, \mu_5, \tilde{R}_r$, and the interaction $\mu_2 \times \mu_4$ have the lowest contributions toward the response and are pooled to form an error term of 12 DoF, similar to the previous ANOVA. Lastly, the F-ratios are calculated, and the results are shown in Table 4.10. Similar to SNR, the factors $\mu_2, \mu_1, \nu_4, \tilde{R}_u$, and T are found to be significantly contributing to the response.

By observing the percentage contributions obtained from the two ANOVAs, we can see that, in general, a similar trend is observed for a factor's or interaction's contribution.

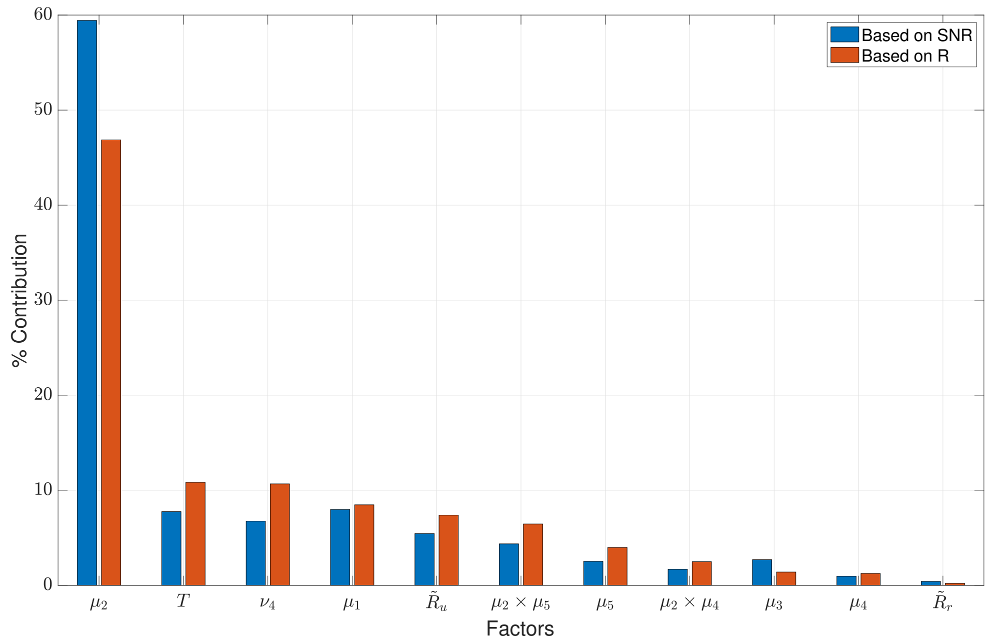

Figure 4.16: Comparison of percentage contributions towards SNR and R

A comparison of the percent contributions of factors toward R and SNR can be seen in Figure 4.16. It can be observed that μ_2 has a larger effect on SNR (59.43%), as compared to R (46.86 %). Whereas, the other significant contributing factors, μ_1, ν_4, T, and $\tilde{R}_u$, all have a comparatively higher effect on the response R. The factors μ_4 and $\tilde{R}_u$ are the two least contributing factors in both cases. It can also be confirmed from this research, that the penalty associated with the trajectory tracking error, $\tilde{R}_r$, does not largely affect how tactical the behavior of the UAV is. While investigating the implications of these results on the trajectory planner, it can be observed that the behavior of the UAV is significantly dependant on the number of time steps considered while generating the reference trajectory (T), and the soft constraints used to approximate the obstacles' set (ν_4).

4.6 Flight Tests Results

As explained in Section 4.3, flight tests have been used to validate the realism of simulations and verify the accuracy of the results obtained by analyzing the outcomes of numerical simulations. This section analyzes the results of flight tests performed for Experiments 5, 15, 21, and 26, over the three obstacle configurations. To limit the effect of non-deterministic factors each experiment was performed multiple times, and the average results have been analyzed. The results of these flight tests are shown in Tables 4.11, 4.12, and 4.13, for the three obstacle configurations, respectively. In these tables, each entry in the column labeled 'Flight', denotes the average of d_{avg}, expressed in meters, over multiple flights. Similarly, the columns labeled 'Sim' denote d_{avg} obtained for the respective experiment through simulations. Figures 4.17, 4.18, and 4.19 provide a graphical representation of the trends observed in Tables 4.11, 4.12, and 4.13, over the respective obstacle configurations. These figures also show the individual d_{avg} observed for individual flight tests. The paths taken by the UAV for different flight tests, along with the corresponding simulations, can be seen in Figures 4.20, 4.21, and 4.22, respectively, for the three obstacle configurations.

It must be noted that several factors, that are not considered in the simulations, have an influence on the behavior of the UAV while conducting these flight tests. The turbulence generated when the UAV is hovering close to an obstacle or wall is an example of such an external influencing factor. This turbulence has the capacity to destabilize the UAV. Altering the probability threshold set for a voxel to be occupied also changes the way the UAV behaves. Finally, there are instances, where due to the parameter settings of the guidance system, the UAV tends to go extremely close to the obstacles, often causing it to crash into them. Due to the interference of such external factors, gathering data for Experiment 5 for Map 3, and Experiment 26 for Map 2 was met with persistent difficulties.

	Experiment 5		Experiment 15		Experiment 21		Experiment 26	
	Sim	Flight	Sim	Flight	Sim	Flight	Sim	Flight
Map 1	1.9140	1.8846	1.8872	1.8376	1.8511	1.3187	1.4476	1.167

Table 4.11: Comparison of d_{avg} between flight test and simulation over Map 1

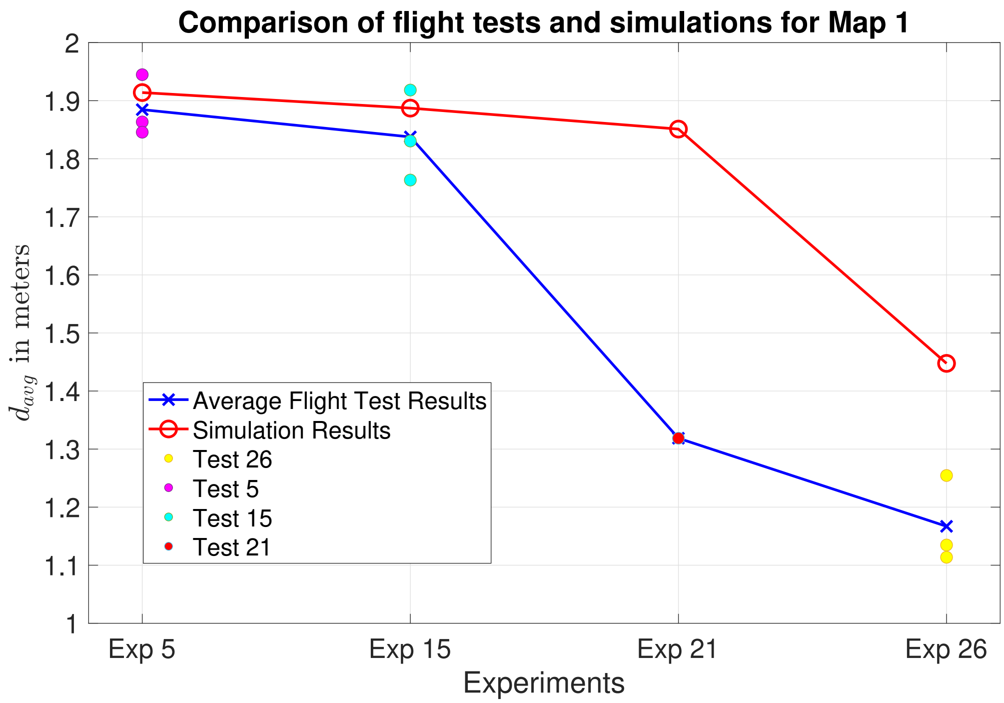

Figure 4.17: Trends observed in flight tests and simulations over Map 1

We begin by investigating the trends observed over the flight tests on Map 1. As observed in Table 4.11, for simulations performed over Map 1, Experiment 5 has the highest d_{avg}, followed by Experiment 15, Experiment 21, and finally Experiment 26, thus showing a gradually decreasing trend. This trend is represented by the red line in Figure 4.17. While comparing the trends displayed by the flight tests, represented by the blue line in the same figure, we can observe, a decreasing trend, similar to that shown by the simulations is observed. Subsequently, we can note that the average results of the flight tests for Experiments 5 and 15 are very close to those of the simulations. For Experiments 21 and 26, the difference between the results of the flight tests and simulations marginally increases, with Experiment

26 showing a difference of 0.28 m between the flight tests and simulations, as compared to less than 0.1 m for Experiment 5. Even though a higher difference is witnessed, the general trends are followed throughout these tests. Thus a similar trend can be confirmed for simulations over Map 1.

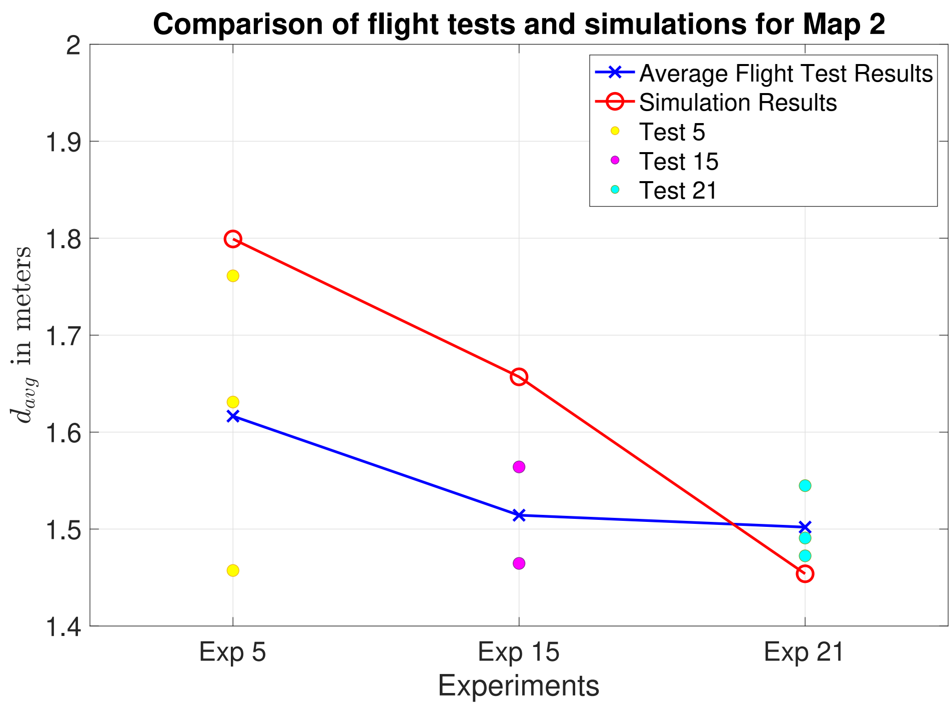

Figure 4.18: Trends observed in flight tests and simulations over Map 2

	Experiment 5		Experiment 15		Experiment 21	
	Sim	Flight	Sim	Flight	Sim	Flight
Map 2	1.7992	1.6165	1.6570	1.5143	1.4539	1.5020

Table 4.12: Comparison of d_{avg} between flight test and simulation over Map 2

Similarly, we analyze the behavior displayed by the UAV over Map 2 by analyzing the results portrayed in Table 4.12 and Figure 4.18. For Map 2, a prominently decreasing trend, for d_{avg}, is seen through Experiments 5, 15, and 21, as portrayed by the red line in Figure 4.18. Compared with the results of the flight tests, a similar decreasing trend is witnessed.

Experiment 5 has the highest d_{avg}, followed by Experiment 15, and lastly Experiment 21. It must be noted though, that for this obstacle configuration, the flight tests for Experiment 21 result in marginally smaller values (difference of 0.12m) with that of Experiment 15. In contrast, the simulations show a larger difference (0.20m) between Experiments 15 and 21. Thus, although a high degree of correlation can be seen in the behavior over Experiments 5 and 15, due to external influences, we observe a higher-than-expected d_{avg} for flight tests of Experiment 21, even though a decreasing trend is observed.

	Experiment 15		Experiment 21		Experiment 26	
	Sim	Flight	Sim	Flight	Sim	Flight
Map 3	0.8544	1.3762	1.3251	1.5746	2.6202	1.5846

Table 4.13: Comparison of d_{avg} between flight test and simulation over Map 3

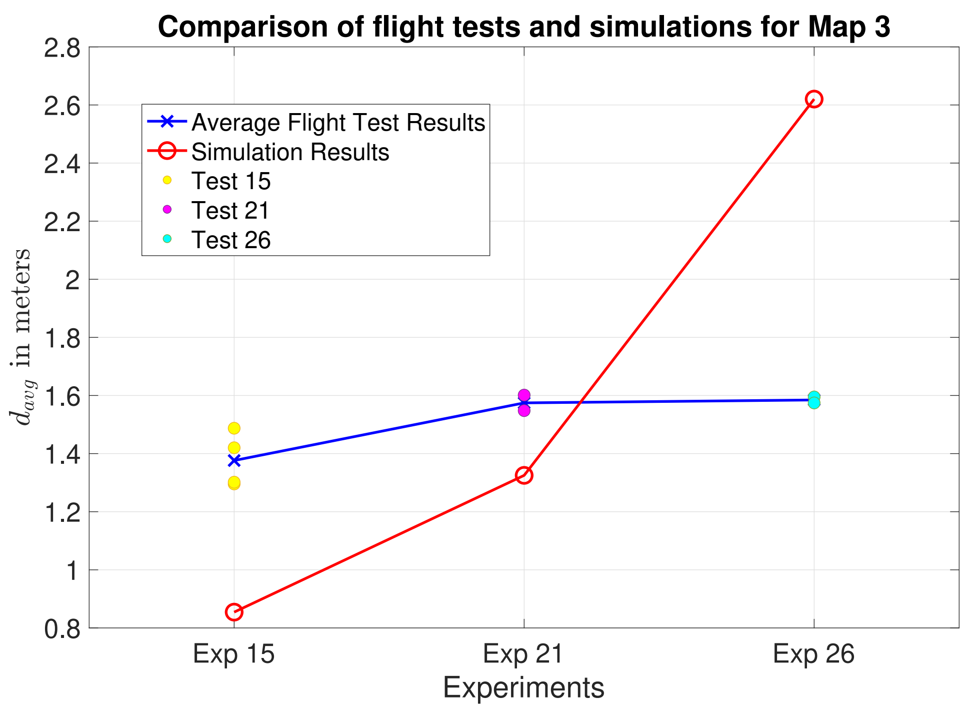

Figure 4.19: Trends observed in flight tests and simulations over Map 3

Lastly, through Table 4.13 and Figure 4.19, we compare the results of the flight tests and simulations over Map 3. Inspecting the results of the simulations, a progressively increasing

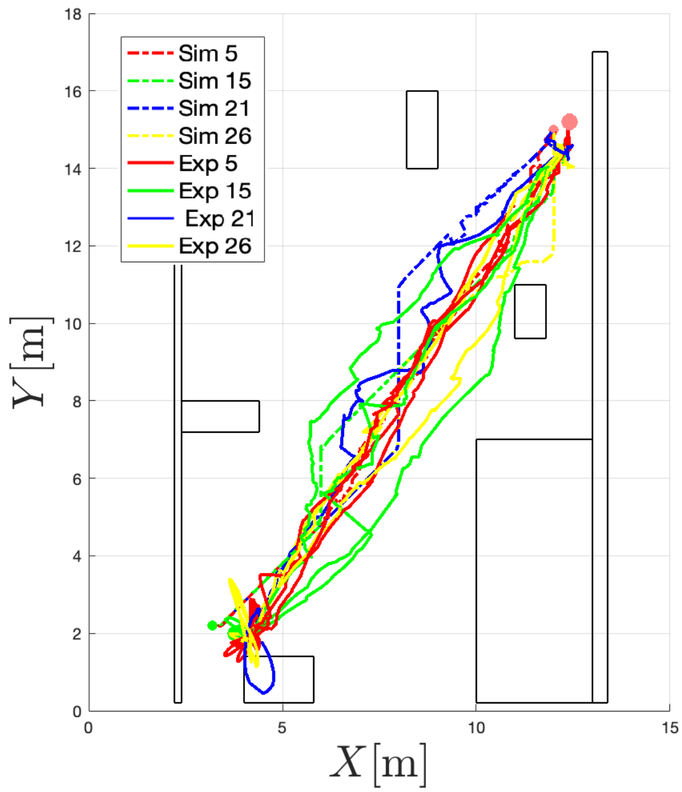

Figure 4.20: Paths observed for Experiments 5, 15, 21, and 26 over Map 1

trend is seen in d_{avg}, as represented by the red line in Figure 4.19. While comparing the corresponding results of the flight tests, we can see a similar behavior over Experiments 15 and 21. Flight tests performed over Experiment 26 show a d_{avg} result that is almost similar to that of Experiment 21. This is an anomaly to our expectations.

Hence, out of the 10 resulting comparison points (four over Map 1, three over Map 2, and three over Map 3), we can see clear co-relation for 8 of these points. Furthermore, the two points which do not closely follow the expectations, still display the expected (increasing or decreasing) trends. Thus, with the help of these flight tests, we can confirm the realism of the simulations.

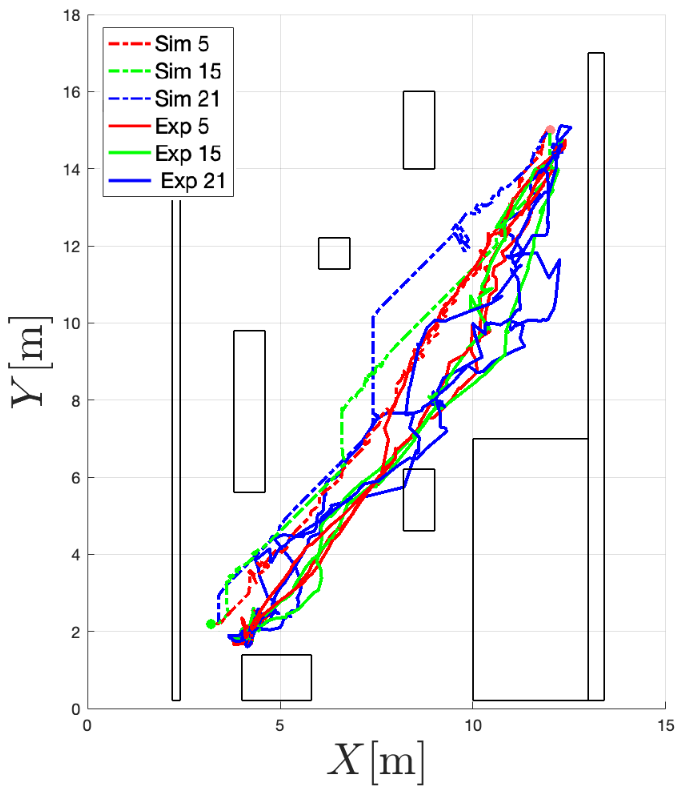

Figure 4.21: Paths observed for Experiments 5, 15, and 26 over Map 2

4.7 Conclusion

In this chapter, the architecture of the guidance system was presented. The implementation of this tactical guidance system was explained with the specific details of the UAV's components in Section 4.1. Subsequently, the process of generating maps for simulations, along with the nature of these maps was presented in Section 4.2. This chapter also recalls the procedure of simulating the behavior of the UAV. Furthermore, the rationale behind conducting the flight tests was explained followed by its procedure.

Subsequently, the results of the simulations performed in Section 4.2 were presented in Table 4.2, and Experiment 19 was observed to be the most tactical out of the 27 experiments performed in this research. The behavior of the UAV observed through these simulations,

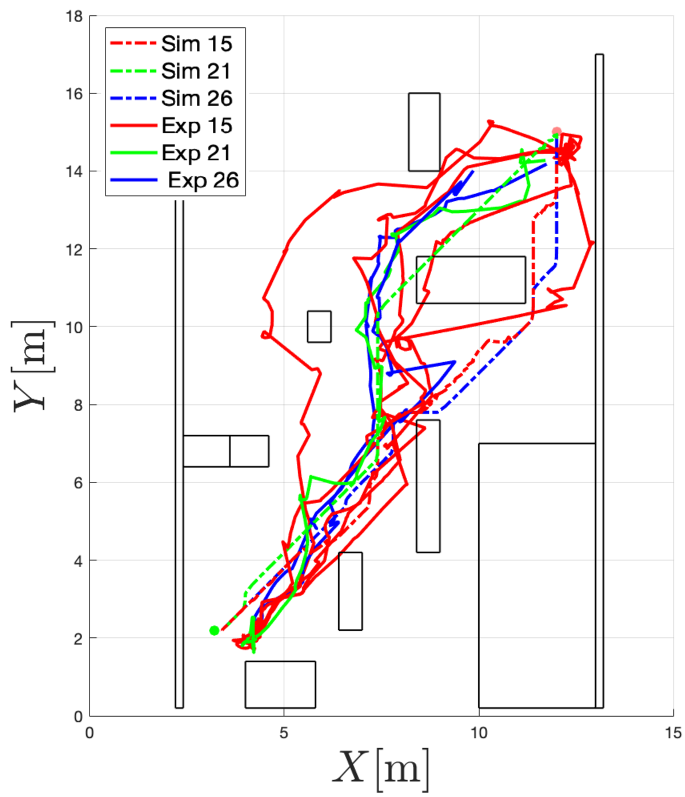

Figure 4.22: Paths observed for Experiments 15, 21, and 26 over Map 3

over the three maps, have been portrayed in Figures 4.5, 4.4, and 4.3, respectively. The averages of SNR and R as shown in Tables 4.4 and 4.3, and their graphical representation in Figures 4.13 and 4.12, were used to deduce an optimal parameter combination. This optimal level set, shown in Table 4.6, was used to perform confirmation experiments. Based on the results of these confirmation tests, represented in Table 4.5, the level settings exhibited in Table 4.6 were concluded to provide the most tactical behavior to the UAV.

Furthermore, the column effect method was used to get a primary estimation of the degree of influence of the factors on the response of the system. Concurrently, plotting analysis was used to confirm the presence of interaction between μ_2 and μ_4, and between μ_2 and μ_5. Interactions were observed in both cases, with $\mu_2 \times \mu_5$ showing a higher degree of interaction. Subsequently, ANOVA was performed on SNR and R, to quantitatively estimate the contributions of the factors and interaction, and the results were shown in Tables 4.7 and 4.9. Several hypotheses developed from the inferences of plotting analysis were confirmed by referencing the percentage contributions and the F-ratios, generated via ANOVA.

It was found that $\mu_2, \mu_1, T, \tilde{R}_u, \nu_4$, and $\mu_2 \times \mu_5$ were contributing toward both the mean and the variance of the results. Concluding from the F-ratios deduced by performing ANOVA on the SNR, $\mu_2, \mu_1, T, \tilde{R}_u$, and ν_4 were the significant contributing factors. Similarly, based on the F-ratios deduced from the ANOVA performed on the response R, factors $\tilde{R}_u$, μ_2, μ_1, and T qualify as significantly contributing at 95% significance threshold. Lastly, the results of the flight tests performed were analyzed to examine if similar trends were displayed by flight tests and simulations. It was found that the flight tests follow the results expected from the simulations, even though the d_{avg} for the flight tests is generally higher than that expected in the simulations.

Finally, this chapter presented the trends observed in flight tests through Tables 4.11, 4.12, and 4.13. Figures 4.17, 4.18, and 4.19 showed a graphical representation of this data, where similar trends were observed between flight tests and simulations.

Chapter 5

Conclusion

The goal of this thesis was to analyze and forecast the behavior of autonomous UAVs implementing a novel, bio-inspired guidance system designed to confer a tactical behavior to the vehicle, while exploring unknown and potentially hostile areas. To undertake this task, we outlined the provided an overview of this guidance system's architecture and described the role of 9 user-defined parameters, whose role is to confer a tactical or reckless behavior to the UAV. As it is analytically impossible to predict the effect of each parameter on the behavior of the system, the use of Taguchi design of experiments has been implemented.

As a first step in the proposed Taguchi analysis, 9 parameters were selected for their more significant effect on the vehicle's behavior. Based on the guidelines set by Taguchi, the L27 OA was selected for this research, which can accommodate these nine factors and study the interactions among two of them. Next, to quantify the intangible quality of being tactical, we developed a specialized response parameter R. This response parameter depends on the average distance of the UAV to the closest obstacle throughout the flight d_{avg}, and it also considers the effect of the obstacle configuration, which is a noise factor in this research. The noise factor was subsequently quantified by calculating the moment of inertia of these obstacle configuration.

To analyze the effect of the obstacle configuration, we simulated the behavior of the UAV over three obstacle configurations with a significantly varying moment of inertia, which was analogous to using an outer array. By analyzing the results of these simulations, we deduced a parameter combination that was expected to confer a highly tactical behavior to the UAV. We performed confirmation tests using this deduced parameter set, over the three obstacle configurations and compared it to the results of the most tactical experiment, out of the 27 experiments performed, which showed that using the optimal parameter set resulted in a more tactical behavior. This optimal parameter set was further used to deduce the ideal values for all the control parameters. The path planner parameters and the trajectory planner parameters followed the trends forecasted during our literature review. Lastly, we observed that a more tactical behavior was obtained when the MPC generated trajectories for a smaller number of time steps. Thus, using the Taguchi method, we deduced a parameter combination that confers a highly tactical behavior to the UAV irrespective of the topology of the map it is flying in.

Subsequently, using the *column effect method*, we deduced that path planning parameters have a relatively high influence on the behavior of this system. Successively, using *plotting analysis*, we concluded that there exists an interaction effect between the parameters belonging to the path planning subsystem and the trajectory planning subsystem. By observing the percentage contributions and F-ratios deduced via ANOVA, the hypotheses deduced from the Taguchi analysis were verified. Lastly, flight experiments validated the results achieved by means of software-in-the-loop simulations

Bibliography

[1] J. A. Marshall, R. B. Anderson, W.-Y. Chien, E. N. Johnson, and A. L'Afflitto, "A guidance system for tactical autonomous unmanned aerial vehicles," *Journal of Intelligent & Robotic Systems*, vol. 103, no. 4, p. 71, 2021.

[2] V. N. Nair, B. Abraham, J. MacKay, G. Box, R. N. Kacker, T. J. Lorenzen, J. M. Lucas, R. H. Myers, G. G. Vining, J. A. Nelder, M. S. Phadke, J. Sacks, W. J. Welch, A. C. Shoemaker, K. L. Tsui, S. Taguchi, and C. F. J. Wu, "Taguchi's parameter design: A panel discussion," *Technometrics*, vol. 34, no. 2, pp. 127–161, 1992. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/00401706.1992.10484904

[3] S. P. Jensen, S. J. Gray, and J. L. Hurst, "How does habitat structure affect activity and use of space among house mice," *Animal Behaviour*, vol. 66, no. 2, pp. 239–250, 2003.

[4] U.S. Army, "An Infantyman's guide to combat in built-up areas," U.S. Department of the Army, 2013, Technical report.

[5] R. K. Roy, *A primer on the Taguchi method.* Society of Manufacturing Engineers, 2010.

[6] P. Ross, *Taguchi Techniques for Quality Engineering: Loss Function, Orthogonal Experiments, Parameter and Tolerance Design.* McGraw-Hill, 1996. [Online]. Available: https://books.google.com/books?id=CiunygZ90TsC

[7] S. K. Karna, R. Sahai *et al.*, "An overview on taguchi method," *International Journal of Engineering and Mathematical Sciences*, vol. 1, no. 1, pp. 1–7, 2012.

[8] L. A. Ealey, *Quality by design : Taguchi methods and US industry / Lance A. Ealey.*, 2nd ed. Burr Ridge, IL: Irwin, 1994.

[9] M. S. Phadke, *Quality Engineering Using Robust Design*, 1st ed. USA: Prentice Hall PTR, 1995.

[10] C. Hou, Y. Tang, S. Luo, J. Lin, Y. He, J. Zhuang, and W. Huang, "Optimization of control parameters of droplet density in citrus trees using uavs and the taguchi method," *International Journal of Agricultural and Biological Engineering*, vol. 12, no. 4, pp. 1–9, 2019.

[11] M. N. Islam and A. Pramanik, "Comparison of design of experiments via traditional and taguchi method," *Journal of Advanced Manufacturing Systems*, vol. 15, no. 03, pp. 151–160, 2016.

[12] B. Gopalsamy, B. Mondal, and S. Ghosh, "Taguchi method and anova: An approach for process parameters optimisation of hard machining while machining hardened steel," *Journal of scientific and industrial research*, vol. 68, pp. 686–695, 08 2009.

[13] D. Deepak and B. Rajendra, "Optimization of machining parameters for turning of al6061 using robust design principle to minimize the surface roughness," *Procedia Technology*, vol. 24, pp. 372–378, 2016.

[14] A. Venu Gopal and P. Venkateswara Rao, "Selection of optimum conditions for maximum material removal rate with surface finish and damage as constraints in sic grinding," *International Journal of Machine Tools and Manufacture*, vol. 43, no. 13, pp. 1327–1336, 2003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0890695503001652

[15] A. Qattawi, B. Alrawi, A. Guzman *et al.*, "Experimental optimization of fused deposition modelling processing parameters: a design-for-manufacturing approach," *Procedia Manufacturing*, vol. 10, pp. 791–803, 2017.

[16] R. Anitha, S. Arunachalam, and P. Radhakrishnan, "Critical parameters influencing the quality of prototypes in fused deposition modelling," *Journal of Materials Processing Technology*, vol. 118, no. 1, pp. 385–388, 2001, pART 1: CONTAINING PAPERS PRESENTED AT INTERNATIONAL CONFERENCE ON ADVANCES IN MATERIALS PROCESSING TECHNOLOGY. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0924013601009803

[17] M. V. Mousavi and H. Khoramishad, "The effect of hybridization on high-velocity impact response of carbon fiber-reinforced polymer composites using finite element modeling, taguchi method and artificial neural network," *Aerospace Science and Technology*, vol. 94, p. 105393, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1270963819309290

[18] S. Mahapatra and A. Patnaik, "Optimization of wire electrical discharge machining (wedm) process parameters using taguchi method," *International Journal of Advanced Manufacturing Technology*, vol. 34, pp. 911–925, 01 2007.

[19] R. S. Rao, R. Prakasham, K. K. Prasad, S. Rajesham, P. Sarma, and L. V. Rao, "Xylitol production by candida sp.: parameter optimization using taguchi approach," *Process Biochemistry*, vol. 39, no. 8, pp. 951–956, 2004.

[20] T.-Y. Wang and C.-Y. Huang, "Improving forecasting performance by employing the taguchi method," *European Journal of Operational Research*, vol. 176, no. 2, pp. 1052–1065, 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0377221705007150

[21] S. Kapsalis, P. Panagiotou, and K. Yakinthos, "Cfd-aided optimization of a tactical blended-wing-body uav platform using the taguchi method," *Aerospace Science and Technology*, vol. 108, p. 106395, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1270963820310774

[22] M. Soylak, N. K. Gökçe, and E. S. Topal, "Aircraft wing design at low speeds using taguchi method," *Aircraft Engineering and Aerospace Technology*, 2018.

[23] A. R., R. Ganguli, D. Harursampath, and P. Friedmann, "Robust design of small-scale unmanned helicopter for hover performance using taguchi method," 01 2017.

[24] A. K. Sood, R. Ohdar, and S. S. Mahapatra, "Improving dimensional accuracy of fused deposition modelling processed part using grey taguchi method," *Materials & design*, vol. 30, no. 10, pp. 4243–4252, 2009.

[25] M. W. Spong, S. Hutchinson, M. Vidyasagar *et al.*, *Robot modeling and control.* wiley New York, 2006, vol. 3.