Formation-Based Odour Source Localisation Using Distributed Terrestrial and Marine Robotic Systems

THÈSE N^{O} 7080 (2016)

PRÉSENTÉE LE 15 AVRIL 2016

À L'ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
À LA FACULTÉ DE L'ENVIRONNEMENT NATUREL, ARCHITECTURAL ET CONSTRUIT
LABORATOIRE DE SYSTÈMES ET ALGORITHMES INTELLIGENTS DISTRIBUÉS

ET

À L'INSTITUTO SUPERIOR TÉCNICO (IST) DA UNIVERSIDADE DE LISBOA

PROGRAMME DOCTORAL EN ROBOTIQUE, CONTRÔLE ET SYSTÈMES INTELLIGENTS
ET
DOUTORAMENTO EM ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES (PhD)

PAR

Jorge Miguel DIAS ALMEIDA RODRIGUES SOARES

acceptée sur proposition du jury:

Prof. R. Valadas, président du jury
Prof. A. Martinoli, Prof. A. M. Santos Pascoal, directeurs de thèse
Prof. L. J. Fortes Marques, rapporteur
Prof. G. Antonelli, rapporteur
Prof. P. M. U. de Almeida Lima, rapporteur

TÉCNICO LISBOA

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Suisse
2016

To my grandparents…

Para os meus avós…

Abstract

This thesis tackles the problem of robotic odour source localisation, that is, the use of robots to find the source of a chemical release. As the odour travels away from the source, in the form of a plume carried by the wind or current, small scale turbulence causes it to separate into intermittent patches, suppressing any gradients and making this a particularly challenging search problem.

We focus on distributed strategies for odour plume tracing in the air and in the water and look primarily at 2D scenarios, although novel results are also presented for 3D tracing. The common thread to our work is the use of multiple robots in formation, each outfitted with odour and flow sensing devices.

By having more than one robot, we can gather observations at different locations, thus helping overcome the difficulties posed by the patchiness of the odour concentration. The flow (wind or current) direction is used to orient the formation and move the robots up-flow, while the measured concentrations are used to centre the robots in the plume and scale the formation to trace its limits.

We propose two formation keeping methods. For terrestrial and surface robots equipped with relative or absolute positioning capabilities, we employ a graph-based formation controller using the well-known principle of Laplacian feedback. For underwater vehicles lacking such capabilities, we introduce an original controller for a leader-follower triangular formation using acoustic modems with ranging capabilities.

The methods we propose underwent extensive experimental evaluation in high-fidelity simulations and real-world trials. The marine formation controller was implemented in MEDUSA autonomous vehicles and found to maintain a stable formation despite the multi-second ranging period. The airborne plume tracing algorithm was tested using compact Khepera robots in a wind tunnel, yielding low distance overheads and reduced tracing error. A combined approach for marine plume tracing was evaluated in simulation with promising results.

Keywords: odour source localisation, plume tracing, robotic olfaction, graph-based formations, range-based formations, formation control, distributed robotics, marine robots, multi-robot systems, environmental sensing

Resumo

A presente tese aborda o problema da localização de fontes de odor com recurso a robôs. Ao afastar-se da fonte, transportado pelo vento ou pela corrente, o odor forma uma pluma que, sob a acção de turbulência de pequena escala, se separa em pacotes discretos. A ausência de um gradiente torna a localização da fonte um problema particularmente desafiante.

Este trabalho visa o estudo de estratégias distribuídas para seguimento de plumas químicas no ar e na água. Lidamos maioritariamente com cenários 2D, embora apresentemos também resultados inovadores com plumas 3D. O fio condutor da tese é a utilização de formações de robôs equipados com sensores de odor e de fluxo (vento ou corrente). A utilização de mais de um robô permite-nos medir simultaneamente a concentração em diferentes locais, o que ajuda a superar a inconstância da pluma. O sentido do fluxo é usado para orientar a formação e deslocar os veículos na direcção de onde provém o odor, ao passo que as várias medidas de concentração servem para centrar os robôs na pluma e ajustar a formação para a delimitar.

Propomos dois métodos distintos de controlo de formação. No caso de robôs terrestres e de veículos aquáticos de superfície com acesso a sistemas de localização relativos ou absolutos, introduzimos um controlador baseado na teoria dos grafos, recorrendo nomeadamente à matriz de Laplace associada a um grafo. Para robôs subaquáticos sem tais capacidades de localização, oferecemos um método original para controlo de uma formação triangular guiada por dois líderes à superfície, utilizando sensores acústicos capazes de medir a distância entre veículos.

Os métodos desenvolvidos foram sujeitos a uma avaliação experimental detalhada, quer em simulação, quer em ensaios com robôs reais. O controlador de formação para robôs subaquáticos foi implementado nos veículos MEDUSA e demonstrou ser capaz de manter uma formação estável em testes de superfície, não obstante o longo intervalo entre medições de distância. O algoritmo para seguimento de plumas aéreas foi testado com robôs Khepera num túnel de vento, atingindo elevada eficiência e baixo erro na localização da fonte. Avaliámos ainda, em simulação, uma estratégia combinada para seguimento de plumas aquáticas, com resultados promissores.

Palavras-chave: localização de fontes de odor, seguimento de plumas químicas, olfacção robótica, formações baseadas em grafos, formações baseadas em distâncias, controlo de formação, robótica distribuída, robôs marinhos, sistemas multirrobô, monitorização ambiental

Résumé

Cette thèse aborde le problème de la localisation robotique d'une source odorante, c'est-à-dire, l'utilisation de robots pour trouver l'origine d'un dégagement chimique. Lors ce que l'odeur s'éloigne de le la source, sous la forme d'un panache transporté par un vent ou un courant, il est affecté par des turbulences à petite échelle, causant son délitement en de petits paquets, supprimant tout gradient et rendant le problème de recherche particulièrement difficile.

Nous nous concentrons sur des stratégies distribuées pour suivre un panache d'odeur dans l'air et dans l'eau, et nous intéressons essentiellement aux scénarios 2D, bien que nous présentions aussi des résultats novateurs pour la troisième dimension. Le fil conducteur de notre travail est l'utilisation de plusieurs robots en formation, chacun équipé de capteurs d'odeur et de flux. L'utilisation de plusieurs robots permet de mesurer la concentration d'odeur à différents endroits, aidant à surmonter les difficultés posées par une concentration irrégulière. La direction du flux est utilisée pour orienter la formation et faire remonter les robots à contre-courant, tandis que les mesures d'odeur sont utilisées pour centrer les robots sur le panache et pour adapter la taille de la formation en fonction de ses limites.

Nous proposons deux méthodes pour contrôler une formation. Pour des robots terrestres et des véhicules marins de surface capables de positionnement absolu ou relatif, nous utilisons un contrôleur basé sur la théorie des graphes en utilisant la matrice laplacienne. Pour des véhicules sous-marins sans cette capacité, nous introduisons un contrôleur original pour une formation triangulaire guidée par deux meneurs à la surface, utilisant seulement des modems acoustiques capables de mesures de distance.

Les méthodes que nous proposons ont été étudiées de façon extensive sous forme de simulations de haute fidélité et d'expériences matérielles. L'algorithme de contrôle de formations marines a été implémenté sur les véhicules MEDUSA et maintient une formation stable malgré la basse fréquence des mesures de distance. L'algorithme de suivi des panaches aériens a été testé en utilisant des robots Khepera dans une soufflerie, obtenant de faibles erreurs et de courtes distances. Une solution mixte pour suivre des plumes aquatiques a été évaluée en simulation, avec des résultats prometteurs.

Mots clefs : localisation de sources d'odeur, suivi de panaches, olfaction robotique, formations basées sur des graphes, formations basées sur des distances, contrôle de formations, robotique distribuée, robots marins, systèmes multi-robot, surveillance de l'environnement

Acknowledgements

Years ago, I decided to go for a PhD in a field I had little knowledge of, having never taken a course in robotics or control. I am not one to doubt, yet for the longest time I doubted this day would come. I am not one to quit, yet I more than once contemplated quitting.

The present document would never have come to fruition without the support and guidance of my supervisors, Prof. António Pedro Aguiar, Prof. Alcherio Martinoli, and Prof. António Pascoal. Thank you for keeping me on track and guiding me to completion. You have taught me a great deal.

I must thank Prof. Pedro Lima, who, besides educating me in robotics and helping with the first steps in the IST-EPFL program, was part of my CAT and jury, as well as the remaining members of the jury—Prof. Gianluca Antonelli, Prof. Lino Marques, Prof. Fernando Lobo Pereira, and Prof. Rui Valadas—for taking the time to read and assess my work.

Going back a few years, I owe a debt of gratitude to Prof. Rui Rocha for being a great mentor and for first introducing me to research—if only I had learnt the lesson. I owe Prof. Rui Cruz for the inspiration and for spawning the black hole that absorbed most of my free time during the last eight years. I am grateful to Prof. João Barros because sometimes a single Skype call can make all the difference in the course of one's life.

Many of my colleagues at DSOR and DISAL contributed to this effort in different capacities and deserve to be mentioned. João, Jorge, Luís, and Manuel, who provided invaluable technical and operational assistance for the MEDUSA experiments. Emmanuel, who helped with the Khepera hardware and the wind tunnel. Ali, Francisco, and Iñaki, with whom I had the pleasure of co-authoring publications. Anil and Jonathan, who joined us for semester projects and whose work made it into this thesis. Guillaume, because of whom I now have non-zero knowledge of orbital mechanics. Corinne, who made sure we had a roof over our heads during the first few months in Switzerland. Steven, who reviewed part of the thesis. This list would be incomplete without thanking Marco, Thomas, and Sven, whose previous developments laid the groundwork for my work.

But there is more to life than work, and friends are a big part of it. I am truly grateful to all of them, in Portugal, Switzerland and elsewhere around the world. Despite my constant neglect over the past few years, you make life worth living. I am happy to count everyone at DISAL within this group: by welcoming us with arms wide open and creating a genuinely friendly environment, you made our expat life in Switzerland infinitely better. The same applies to the good people at the EPFL International Spouses group. A special mention to Efthymia and Sara, who kept tabs on me during the final sprint; I may have avoided you at times, but I cannot thank you enough for being there. Also Sofia, who braved (skied, really) the *via dolorosa* ahead of me and switched on the light at the end of the tunnel. Maybe she will now forgive me for betraying her in my choice of college eleven years ago.

Next comes my family, who have incessantly supported and pushed me to do better. My parents, who made all of this possible. My grandparents, who made me who I am and did not all get to see the result. My jack-of-all-trades sister who is said to be outstanding and who proofread this thesis. In the interest of full disclosure, she was the only one to be compensated for her involvement; it is only fair that she takes the blame for any errors. My extended family—I could not have asked for a better one.

Finally, I will never be able to find the right words to thank Rute, who put her life on hold to follow me to Switzerland, and who was there through the good, the bad, and the really weird. You are the real MVP. I have no idea what comes next for us but I know it will be great.

Lisbon, 12 October 2015 J. M. S.

Funding institutions

- Fundação para a Ciência e a Tecnologia through doctoral grant [SFRH / BD / 51073 / 2010], project CONAV [PTDC / EEACRO / 113820 / 2009] and LARSyS-ISR strategic funding [PEst-OE / EEI / LA0009 / 2011 and UID / EEA / 50009 / 2013]
- European Union though project MORPH [EU FP7 ICT 288704]
- EPFL through discretionary funding for the DISAL salary envelope

Contents

Abstract **v**

Resumo **vii**

Résumé **ix**

Acknowledgements **xi**

List of figures **xvii**

List of tables **xix**

I Introduction **1**

1 Introduction **3**
1.1 Odour plume tracing 5
1.2 Formation control 6

2 Outline **9**

3 Contributions **11**

II Platforms and tools **13**

4 Introduction **15**

5 Robots **17**
5.1 Khepera III mobile robot 17
5.2 Khepera IV mobile robot 18
5.3 Khepera sensing modules 20
5.3.1 Wind sensing board 21
5.3.2 Odour sensing board 22
5.3.3 Relative positioning board 22
5.4 MEDUSA autonomous marine vehicle 23

6 Experimental sites **25**

6.1 Wind tunnel 25
6.1.1 Traversing system 26
6.1.2 Positioning system 27
6.1.3 Odour source 28
6.2 Lisbon Oceanarium lagoon 28

7 Simulators 31
7.1 MATLAB/Simulink 31
7.2 Webots 33
7.2.1 Wind simulation and sensing 33
7.2.2 Odour simulation and sensing 34
7.3 ANSYS Fluent 35

8 Summary 37

III Range-based formation control 39

9 Introduction 41

10 Related work 43

11 Technical approach 45
11.1 Range-only formation control 45
11.1.1 Error dynamics 47
11.1.2 Speed controller 47
11.1.3 Heading controller 48
11.1.4 Stability of the overall system 50
11.1.5 Formation heading estimation 52
11.2 Acoustics-based formation control for AUVs 52
11.2.1 Discrete noisy range handling 53
11.2.2 Outlier rejection 54
11.2.3 Heading estimation 54

12 Evaluation 57
12.1 Simplified simulation 57
12.2 Full simulation 58
12.3 MEDUSA real-world testing 60

13 Alternative formations 63

14 Summary 65

IV Formation-based airborne plume tracing 67

15 Introduction 69

16 Related work 71

17 Technical approach 73
17.1 Graph-based formation plume tracing 74
17.1.1 Sensing 75
17.1.2 Laplacian feedback 76
17.1.3 Upwind movement 77
17.1.4 Plume centring 77
17.1.5 Behaviour aggregation 78
17.2 Application to Khepera III robots 78
17.2.1 Relative position handling 79
17.2.2 Formation scaling 79
17.2.3 Collision avoidance 79

18 Evaluation 81
18.1 Webots simulation 81
18.1.1 Linear formation 82
18.1.2 Rectangular formation 84
18.2 Wind tunnel trials 85
18.2.1 Rectangular formation 86
18.2.2 Additional formations 88

19 Alternative behaviours 91
19.1 Plume finding 91
19.2 Source declaration 92
19.3 Windless plume tracing 92

20 Extension to 3D tracing 95
20.1 Algorithm 95
20.2 Evaluation 97

21 Summary 101

V Formation-based waterborne plume tracing 103

22 Introduction 105

23 Related work 107

24 Technical approach 109
24.1 Sensing and communication 110

24.2 Leader control . . . 111
24.2.1 Laplacian feedback . . . 111
24.2.2 Upflow movement . . . 112
24.2.3 Plume centring . . . 113
24.2.4 Behaviour aggregation . . . 113
24.3 Follower control . . . 113

25 Evaluation **115**
25.1 Set-up . . . 115
25.2 Results . . . 117

26 Alternative behaviours **119**
26.1 Scale adjustment . . . 119
26.2 Variable depth operation . . . 120

27 Summary **121**

VI Discussion **123**

28 Conclusions **125**

29 Outlook **127**

Glossary **129**

Bibliography **131**

Curriculum vitae **139**

List of Figures

Part I: Introduction

1.1 Examples of plumes in the air and water. 4
1.2 Possible trajectories of a single robot engaged in plume tracing. 5
1.3 Simulated trajectories of five robots running a graph-based formation controller. 6

Part II: Platforms and tools

5.1 Khepera III robots with sensor stack. 18
5.2 Khepera IV robot with sensor stack. 19
5.3 Individual Khepera sensing boards. 21
5.4 Wind sensor thermistor response. 21
5.5 Odour sensor response. 22
5.6 MEDUSA AMV fleet. 23

6.1 Wind tunnel test section. 26
6.2 Wind tunnel 3D traversing system. 26
6.3 SwisTrack visual tracking software. 27
6.4 Odour pump and ethanol reservoir. 28
6.5 Lisbon Oceanarium test site. 29

7.1 MEDUSA Simulink model. 32
7.2 Webots plume tracing simulation. 33
7.3 ANSYS Fluent unsteady plume simulation. 35

Part III: Range-based formation control

11.1 Marine formation diagram with parameters. 46
11.2 Structure and data flow in the MEDUSA implementation of the formation controller. 53

12.1 Simulation: results of a simple simulation run for a lawn mowing manoeuvre. 58
12.2 Simulation: results of a full simulation run for a lawn mowing manoeuvre. . . 59
12.3 MEDUSA trials: aerial view of the mission area and trajectories of the vehicles. 60
12.4 MEDUSA trials: range measurements, heading references, and control signals. 61

13.1 MEDUSA trials: vehicle trajectories using the alternative formation. 64

Part IV: Formation-based airborne plume tracing

17.1 Schematic view of the wind tunnel. 74

18.1 Webots simulation scenario. 82
18.2 Simulation: formation geometries tested. 82
18.3 Simulation: robot trajectories for the three-robot linear formation. 83
18.4 Simulation: control outputs for the three-robot linear formation. 83
18.5 Simulation: wind measurements and Kalman estimate for the three-robot linear formation. 84
18.6 Simulation: odour measurements and max-filtered estimate for the three-robot linear formation. 84
18.7 Simulation: max-filtered odour measurements for the left-side robots in the rectangular formation. 85
18.8 Simulation: robot trajectories for the five-robot rectangular formation. 85
18.9 Wind tunnel: geometry and graph for the five-robot rectangular formation. . 86
18.10 Wind tunnel: robot trajectories for the five-robot rectangular formation. . . . 86
18.11 Wind tunnel: odour measurements for the five-robot rectangular formation. . 87
18.12 Wind tunnel: wind direction estimates for the five-robot rectangular formation. 87
18.13 Wind tunnel: alternative formations tested. 88
18.14 Wind tunnel: trajectories obtained using the alternative formations. 88

19.1 Wind tunnel: robot trajectories obtained using the windless controller. 94

20.1 Simulation: trajectories obtained using the 3D controller. 97
20.2 Wind tunnel: trajectories obtained using the 3D controller. 98
20.3 Wind tunnel: odour readings obtained using the 3D controller. 99

Part V: Formation-based waterborne plume tracing

24.1 Triangular marine formation showing graph edges. 112

25.1 View of the freshwater plume generated in ANSYS Fluent. 116
25.2 Snapshot of the salinity levels with added noise. 116
25.3 Simulation: vehicle trajectories for different starting positions. 117
25.4 Simulation: max-filtered odour readings. 117
25.5 Simulation: speed and heading of the leaders. 118
25.6 Simulation: speed and heading of the follower. 118

List of Tables

Part IV: Formation-based airborne plume tracing

18.1 Wind tunnel: movement overhead and final error. 89

20.1 Webots simulation: mean final error for different source heights. 98

Introduction Part I

'Yeh'll get yer firs' sight o' Hogwarts in a sec,' Hagrid called over his shoulder, 'jus' round this bend here.' There was a loud 'Oooooh!'. The narrow path had opened suddenly onto the edge of a great black lake. Perched atop a high mountain on the other side, its windows sparkling in the starry sky, was a vast castle with many turrets and towers.

— J. K. Rowling, *Harry Potter and the Philosopher's Stone*

1 Introduction

Have you ever felt the urge to follow that unique scent of freshly baked bread to a nearby shop?

Olfaction, the sense of smell, has long been underrated and overlooked. Nevertheless, it is of the utmost importance to our lives. Recent results suggest that humans are capable of discriminating over 10^9 olfactory stimuli, a resolution several orders of magnitude higher than that of the other senses [1].

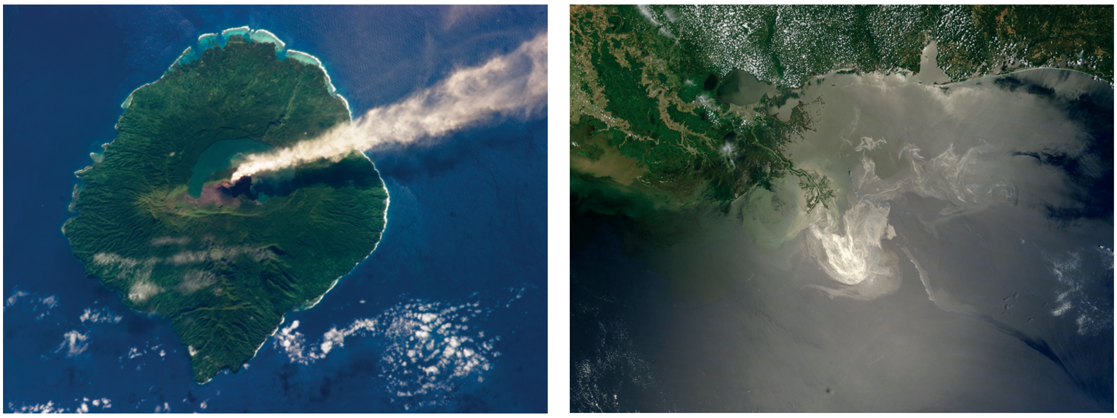
The characteristic scent of bread baking [2], our example, is one most people recognise and many describe as a favourite. Studies have identified some of its effects and potential uses, from improving quality of life in long-term care facilities [3] to modifying consumer behaviour in retail surfaces [4]. Bread and similarly pleasant scents may even influence people to behave more altruistically [5].

And yet, our olfaction is relatively poor when compared with that of many of the animals that surround us. Dogs have 220 million olfactory receptors, versus the 5 million in humans [6], and are 10^8 times more sensitive to certain compounds [7]. We perceive only a small fraction of the incredible volume of olfactory information we take in, and find it hard to imagine being able to pick up the odour of an animal carcass kilometres away [8].

This thesis is not, however, about bread, scents, or olfactory physiology. It is about finding the proverbial bakery, a process referred to as odour source localisation. It takes place all around us: bees use it to find nectar, sharks to locate prey, and dogs to find food, mates, and—with human encouragement—bombs, contraband, and cancer cells [9]. The goal of this work is to design strategies for robots to perform the same task.

As early as the 1950s, researchers were looking into the use of electronic devices for odour sensing, an application commonly referred to as machine olfaction [10], [11]. While it was not for decades that work in robotic olfaction began [12], it has steadily intensified ever since. Yet robotic odour source localisation continues to lag behind its biological equivalents.

What makes this such a challenging problem? Odour propagation is non-trivial, with concen-



(a) (b)

Figure 1.1 – Examples of plumes in the air and water. (a) Volcanic plume emanating from the Gaua Island eruption, Vanuatu, May 2013. (b) Oil plume resulting from the Deepwater Horizon spill, Gulf of Mexico, May 2010. (Image credit: NASA)

tration distribution far from a smooth gradient [13]. Instead, small scale turbulence causes odour to form packets, leading to an intermittent plume with wide concentration fluctuations. Figure 1.1 shows this effect in large-scale airborne and waterborne plumes.

Plumes are also dynamic and subject to meandering, their shape and position potentially varying. This limits the quantity and quality of information that can be collected by a single *in situ* odour sensor, a problem further exacerbated by relatively slow sensing technology with long response and recovery times.

Just as the challenges are great, the pay-off is formidable. Post-war de-mining is a prime example of a situation where odour source localisation comes into play. Current mainstream minefield clearing methods use dogs or rats that can sniff the chemical signature exuded by decaying explosives. This is a risky endeavour, both for the animals and for their human handlers. Urban gas leak detection, industrial plant safety [14], and post-disaster search and rescue [15] are but a few of the countless safety and security applications. The environment would also stand to benefit, with improved capabilities for pollution tracing and underwater pipeline leak localisation. Many of these and other scenarios are difficult or dangerous to operate in, and therefore leading candidates for the deployment of robotic solutions.

In this thesis, we address a particular aspect of odour source localisation: how to trace (or follow) a chemical plume. Of the several classes of solutions to this problem, we choose to focus on multi-robot approaches and, in particular, formation-based plume tracing. Consequently, our research work revolves around these core subjects: odour plume tracing and formation control. The following sections briefly introduce the two.

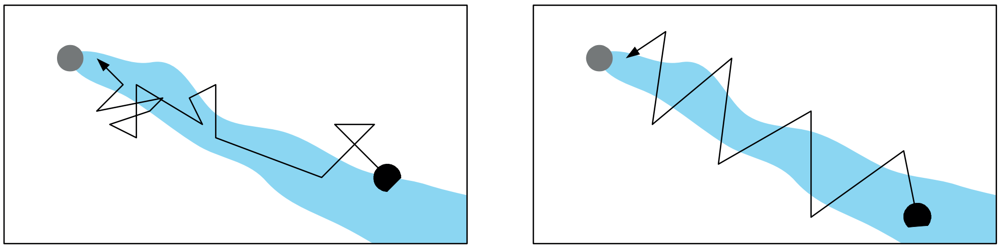


Figure 1.2 – Possible trajectories of a single robot engaged in plume tracing using different algorithms (biased random walk and casting). (Image credit: T. Lochmatter [16])

1.1 Odour plume tracing

Odour source localisation is, fundamentally, a search problem that consists of locating the source of some chemical substance present in the environment.

Over the past two decades, numerous studies have addressed this problem. Simple algorithms based on the behaviour of model insects gave way to a host of approaches of differing complexity and inspiration, including not only a wide variety of bio-inspired algorithms but also those based on probabilistic inference, optimisation meta-heuristics like particle swarm optimisation (PSO), or multi-robot swarms, to name but a few. The set of solutions also encompasses map-based approaches, as well as exclusion-based strategies such as plume or source avoidance. Figure 1.2 shows two possible strategies in operation.

Odour source localisation is usually divided into three processes or stages:

- plume finding, the process by which the robot acquires the plume
- plume tracing, the process by which the robot follows the plume
- source declaration, the process by which the robot determines that it reached the source

While the processes may appear sequential, the boundaries are less clear in practice: a robot may, for instance, lose the plume it is tracing and need to re-engage in plume finding. Source declaration, in particular, is often an ongoing process during plume tracing, and the two are deeply intertwined. Depending on the approach, the same algorithm may tackle one or more stages.

Instead of attacking the overall problem, we concentrate on the intermediate and most frequently studied problem, plume tracing (sometimes referred to as plume traversal or tracking). We have already seen that following a chemical plume presents particular challenges when compared with seemingly related problems: due to turbulent transport, chemical concentration in a plume tends to be very patchy, with packets of high concentration and periods of low or completely absent odour. Classical gradient ascent algorithms are unlikely to perform well.

We focus on multi-robot approaches, and specifically formation-based algorithms that take

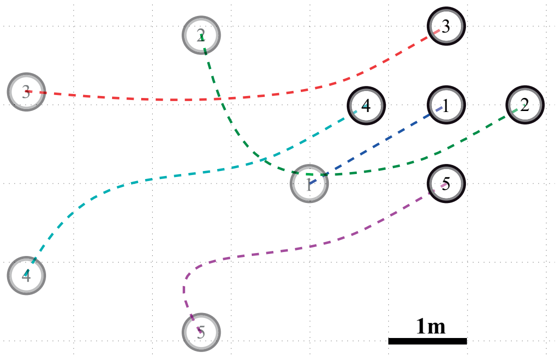


Figure 1.3 – Simulated trajectories of five robots running a graph-based formation controller. The leader R_1 moves independently while the other agents attempt to reach a diamond formation around it. (Image credit: S. Gowal [17])

advantage of cooperating robots moving as a group to locate the source with minimal wandering. Reducing wandering leads to improved time efficiency, an important consideration in emergency scenarios. We make a reasonable assumption that the odour is transported by the wind or current, and that the robots are able to measure both.

Under these constraints, our refined research goal becomes to drive a group of robots in a formation towards the source of a detected chemical plume using local odour and flow readings.

1.2 Formation control

We have thus far discussed formations as a tool in the scope of plume tracing algorithms. However, formation control is an active research field in its own right, both within robotics and control theory.

The base formation control problem can be described as driving a robot (or group of robots) to some desired position, generally specified as a set of relative coordinates, resulting in a particular spatial configuration. As the group moves, either under the action of a leader or of distributed control, the configuration should be maintained. Figure 1.3 shows an example of formation control in action, in which four robots start from arbitrary positions and converge to a diamond-shaped escort around a leader.

Classically, the problem is solved through feedback laws on some measure of the difference between the actual and desired positions. In ideal conditions and for a set of cooperating robots, this is a consensus problem with a simple solution in the form of Laplacian-based rendezvous augmented with bias vectors describing the final configuration. We base part of our work, using land robots, on a modified version of this approach.

The problem becomes more complicated as additional restrictions are put in place. Kinematic constraints on the movement of the robots, as well as their dynamics, limit the controllability of the system and have to be taken into account. Localisation errors and external perturbations can also adversely impact formation stability, as do fundamental limitations to sensing and, depending on the choice of controller, communication.

For all the reasons above, marine vehicles present an especially challenging case for formation control. Being immersed in a dynamic medium, autonomous marine vehicles (AMVs) may not retain full control over their own motion and have relatively complex dynamics; even staying in a fixed position may require active control.

Furthermore, in the case of autonomous underwater vehicles (AUVs), communication and positioning are normally effected through acoustic means that severely limit the quantity of information exchanged. In particular, many acoustic systems are only capable of providing ranges, measured using the time-of-flight of pings or messages, thereby only resolving the position of a neighbour to a circle around the vehicle.

While formations are used throughout our work, our research in the field is mostly focused on marine vehicles and the additional localisation and communication difficulties they entail.

2 Outline

The overarching subject of this thesis is formation-based odour plume tracing, but in diverse environments and using different robots. In order to facilitate comprehension of each of our lines of research, we split the write-up in six parts.

The core research occupies the three central parts, starting with range-based formation control for marine vehicles, continuing with plume tracing using ground robots, and finally joining the two in a combined approach to waterborne plume tracing. Each of these parts features its own introduction, related work, and summary.

The full thesis is organised as follows:

Part I: Introduction We briefly introduce the broader field of odour source localisation and the specific problems we address, odour plume tracing and formation control.

Part II: Platforms and tools We describe our hardware platforms and experimental set-up, including the Khepera and MEDUSA robots, the add-on sensing boards, the simulators, and the wind tunnel.

Part III: Range-based formation control We focus on formation control in the absence of fully resolved inter-vehicle positions and introduce a controller for a follower vehicle in a triangular formation using acoustic range measurements.

Part IV: Formation-based airborne plume tracing We address the problem of multi-robot odour plume tracing in the air and propose a graph-based distributed algorithm to drive a group of robots in an arbitrary formation to an odour source.

Part V: Formation-based waterborne plume tracing We combine the work of the previous two parts and introduce a solution for waterborne plume tracing using a triangular formation.

Part VI: Discussion We summarise the thesis, discuss our proposals and results, and provide final comments and avenues for future work.

3 Contributions

The work done in the scope of this thesis led to several contributions and related publications, listed below for each part.

Part II: Platforms and tools The first contribution is an analysis and experimental evaluation of the Khepera IV robot and its individual parts. As one of the first customers to receive the robot, we prepared an in-depth assessment as a service to the community. We also developed an open-source framework to help build applications for the robot, the Khepera IV Toolbox.

- J. M. Soares, I. Navarro and A. Martinoli, 'The Khepera IV mobile robot: performance evaluation, sensory data, and software toolbox', in *Iberian Robotics Conference*, ser. Advances in Intelligent Systems and Computing, vol. 417, Springer, 2015, pp. 767–781. DOI: 10.1007/978-3-319-27146-0_59.

Part III: Range-based formation control The second contribution is a controller to maintain a formation of marine robots that can only sense inter-robot distances using acoustic ranging devices. Our publications tackle different aspects of the problem, including the theoretical foundations, simulation results, experimental data, and alternative formulations.

- J. M. Soares, A. P. Aguiar and A. M. Pascoal, 'Triangular formation control using range measurements: an application to marine robotic vehicles', in *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles*, Porto, Portugal, 2012, pp. 112–117. DOI: 10.3182/20120410-3-PT-4028.00020.
- J. M. Soares, A. P. Aguiar, A. M. Pascoal and A. Martinoli, 'Joint ASV/AUV range-based formation control: theory and experimental results', in *IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, 2013, pp. 5579 –5585. DOI: 10.1109/ICRA.2013.6631378.
- J. M. Soares, A. P. Aguiar, A. M. Pascoal and A. Martinoli, 'Design and implementation of a range-based formation controller for marine robots', in *Iberian Robotics*

Conference, ser. Advances in Intelligent Systems and Computing, vol. 252, Springer, 2013, pp. 55–67. DOI: 10.1007/978-3-319-03413-3.

- F. Rego, J. M. Soares, A. M. Pascoal, A. P. Aguiar and C. Jones, 'Flexible triangular formation keeping of marine robotic vehicles using range measurements', in *IFAC World Congress*, Cape Town, South Africa, 2014, pp. 5145–5150. DOI: 10.3182/20140824-6-ZA-1003.02435.

Part IV: Formation-based airborne plume tracing The third contribution is a formation algorithm for odour plume tracing. Our solution uses a group of simple robots in a graph-based formation to follow a chemical plume to its source, with limited implementation or computational complexity. We present both simulation and real robot results, including pioneering 3D plume tracing results.

- J. M. Soares, A. P. Aguiar, A. M. Pascoal and A. Martinoli, 'A graph-based formation algorithm for odor plume tracing', in *International Symposium on Distributed Autonomous Robotic Systems*, ser. Springer Tracts in Advanced Robotics, vol. 112, Springer, 2014, pp. 255–269. DOI: 10.1007/978-4-431-55879-8_18.
- J. M. Soares, A. P. Aguiar, A. M. Pascoal and A. Martinoli, 'A distributed formation-based odor source localization algorithm: design, implementation, and wind tunnel evaluation', in *IEEE International Conference on Robotics and Automation*, Seattle, WA, USA, 2015, pp. 1830–1836. DOI: 10.1109/ICRA.2015.7139436.

Part V: Formation-based waterborne plume tracing The last contribution is an extension of our odour plume tracing algorithm to waterborne plumes and marine robots, combining range-based and graph-based formation control. The solution is used to trace a surface freshwater plume in simulation.

To the extent that this manuscript reuses material from our previous publications, referenced above, we recognise the copyrights transferred to their respective publishers.

Platforms and tools Part II

'I'm sorry to say I sold the wand that did it,' he said softly. 'Thirteen-and-a-half inches. Yew. Powerful wand, very powerful, and in the wrong hands . . . Well, if I'd known what that wand was going out into the world to do . . .'

— J. K. Rowling, *Harry Potter and the Philosopher's Stone*

4 Introduction

Our research unfolds along several lines, thereby requiring a diverse set of tools and platforms for development, simulation, and experimentation. There is, in particular, a major demarcation between the work using terrestrial robots and the work using marine robots, which come with distinct characteristics, traditions and established know-how.

In this part, we present the main tools we employ. The next chapter begins by introducing the robots (Khepera III, Khepera IV and MEDUSA) and the sensors with which they are equipped. We then describe our experimental settings, namely the lagoon for in-water testing and the wind tunnel where the plume tracing experiments took place. Finally, we go over the tools used to model the vehicles, simulate the transport phenomena, and test the algorithms.

5 Robots

Our land-based work makes use of the Khepera family of mobile robots. These are compact, differential-wheeled robots developed and built in Switzerland by K-Team. They are particularly well-suited for indoor robotic experiments, given their small size, reasonable capabilities, and easy expandability.

Our water-based work is developed around the MEDUSA-class AMVs. These robots have been designed and built at Instituto Superior Técnico (IST), before and during this project, and are small and lightweight marine vehicles especially fit for multi-robot experiments. Their feature set has been evolving, including a recent and significant shift from surface-bound operation to full AUV capabilities.

The robots are described in more detail in this chapter, as are the additional sensors installed and used specifically for this work.

5.1 Khepera III mobile robot

The Khepera III, shown in Figure 5.1, is a compact differential-wheeled mobile robot. It was first released in 2006, and its development saw contributions by the Distributed Intelligent Systems and Algorithms Laboratory (DISAL).

The robot is approximately cylindrical with a diameter of 13 cm and a height of 7 cm [25]. Two 41 mm wheels sit 8.41 cm apart and a plastic pin on the bottom of the battery package provides a third contact point. The complete robot with battery weighs 690 g.

Our robots carry the Korebot II board, which features an Intel XScale PXA-270, 32 MB of Flash memory and 128 MB of RAM, and runs a GNU/Linux OS. Communication is handled by a 802.11b CompactFlash card, and a dsPIC 30F5011 microcontroller manages the interface between the Korebot II and the built-in devices.

The Khepera III is powered by a 1350 mAh 7.4 V replaceable battery that can either be charged in-place or using an external charger. The documentation specifies a running time of up to

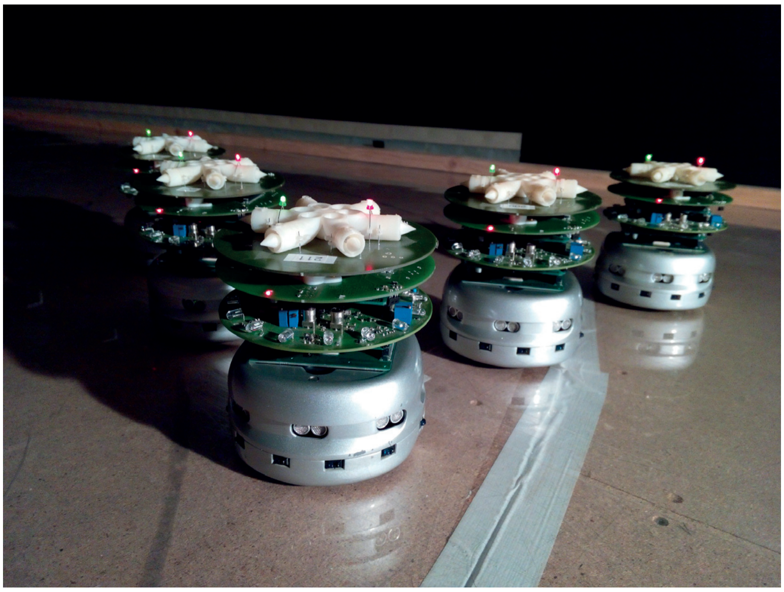


Figure 5.1 – Khepera III robot fleet. The robots are equipped with a wind sensing board (top), odour sensing board (middle) and relative localisation board (bottom). The wind sensing board features red and green LEDs for overhead tracking.

4 h, but our experiments with additional sensors and ageing batteries often brought this down to less than 10 min.

The base package includes the following sensors:

- eleven infrared reflective optical sensors, two of which facing downwards as ground detectors and the others mounted around the robot, functioning as ambient light sensors or proximity detectors with a published range of 20–250 mm
- five ultrasonic transceivers around the front of the robot, for obstacle detection at a range of 20–400 cm
- two wheels encoders, yielding approximately 22 pulses per millimetre of wheel movement

Although K-Team provides a reference application programming interface (API) with the robot, we instead use the Khepera III Toolbox, a simplified and enhanced interface previously developed at DISAL and maintained as an open-source project [26].

5.2 Khepera IV mobile robot

Building on the success of the Khepera III, K-Team released the Khepera IV in January 2015, making it the most recent iteration in the series. Being among the first users of the Khepera IV, we undertook a detailed analysis and experimental assessment of the robot [18], of which we

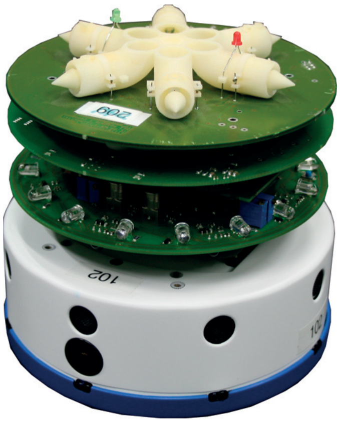


Figure 5.2 – Khepera IV robot equipped with the same sensor stack.

only summarise the major characteristics here.

Just like its predecessor, the Khepera IV (pictured in Figure 5.2 with additional sensor boards) is designed for indoor use. It is shaped as a cylinder, with a diameter of 14.08 cm and a height of 5.8 cm. Inside, it follows a similar stacked PCB design. The complete robot weighs 566 g.

The two actuated wheels are 42 mm in diameter (including the O-rings that act as tires) and are centred on the sides of the robot, spaced 10.54 cm apart. Two ball transfer units, at the front and at the back, provide the remaining contact points. This solution results in 0.5–1 mm of ground clearance, making the robot very stable but preventing its use on any surface that is not effectively flat and smooth.

The brain of the robot is a Gumstix Overo FireSTORM COM, an off-the-shelf embedded computer that carries a Texas Instruments DM3730 800MHz ARM Cortex-A8 Processor with a TMS320C64x Fixed Point DSP core, 512 MB of DDR LPDRAM, and 512 MB of NAND Flash memory. A combined transceiver provides both 802.11b/g (WiFi) and Bluetooth 2.0+EDR capabilities using internal antennae. Low-level aspects are managed by a dsPIC33FJ64 GS608 micro-controller that builds a bridge between the embedded computer and the built-in hardware.

Energy is supplied by a 3400 mA h 7.4 V lithium-ion polymer battery. The battery is not swappable and charges in approximately 5 h using the charging jack. Support is also provided for charging from the extension bus or from a set of contacts under the body of the robot.

Compared with the Khepera III, the Khepera IV is outfitted with a rich set of sensing devices:

- twelve reflective optical sensors, four of them downward-facing and eight equally spaced in a ring around the robot body, also capable of operating as ambient light sensors or proximity detectors with a published range of 2–250 mm
- five 40 kHz ultrasonic transceivers with a range of 25–200 cm

- two amplified omnidirectional MEMS microphones, mounted on each side of the robot, with gain of 20 dB and frequency range of 100–10 000 Hz.
- two high-resolution wheel encoders, yielding approximately 147.4 pulses per millimetre of wheel displacement
- a centre-mounted single-package inertial measurement unit (IMU) featuring a 3D accelerometer (±2 g range) and a 3D gyroscope (±2000 dps range)
- a front-mounted colour camera with a 1/3 " WVGA CMOS sensor, yielding a resolution of 752 px × 480 px

The robots are also equipped with three top-mounted RGB LEDs for tracking and identification and a loudspeaker for communication or user interaction.

K-Team released an improved and simplified *libkhepera* library for the new robot, but its out-of-the-box functionality is still limited and the very different API would make it time-consuming to port our existing Khepera III programs. For this reason, we have created a Khepera IV Toolbox, replicating the functionality of the Khepera III Toolbox and implementing support for the new hardware while maintaining, to the extent possible, a compatible API.

Updating most existing programs to work with the new robot should be a simple matter of rewriting some function calls and possibly adjusting for hardware differences (e.g. the position of the proximity sensors, in the case of an obstacle avoidance algorithm). All sixty example programs and additional routines bundled with the Khepera III Toolbox have been revised and are included with the new library.

Our evaluation of the Khepera IV and the source for the Khepera IV Toolbox are available in [18].

5.3 Khepera sensing modules

The native functionality of the robots can be extended through the use of generic USB, Wi-Fi, and, in the case of the Khepera IV, Bluetooth devices. In addition, the robots provide a specialised KB-250 bus allowing for the connection of stackable custom boards.

This 100-pin link, provided by both the Khepera IV and the Korebot II board, includes power, I^2C, SPI, RS-232 and USB buses, as well as specialised lines for, e.g. LCD or dsPIC interfacing. The interface on the two platforms is mostly compatible, with only seldom used pins re-purposed on the newer Khepera.

For the plume tracing experiments, we rigged the robots with a set of sensor boards allowing for an autonomous solution with no external dependencies. These sensors come in the form of three stackable add-on boards previously developed at DISAL, connected through the I^2C bus. Figure 5.3 shows all the boards in our custom stack.

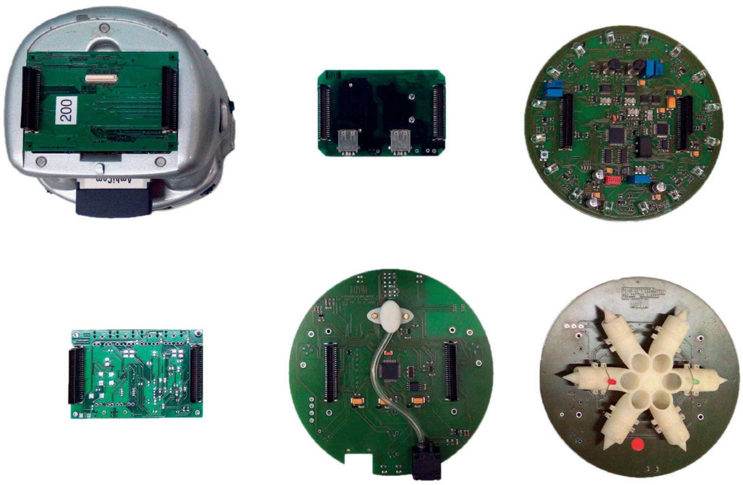


Figure 5.3 – Individual components of our Khepera III stack. Top row: Khepera III robot, power conversion board, relative positioning board. Bottom row: mechanical spacer, odour sensing board and wind sensing board.

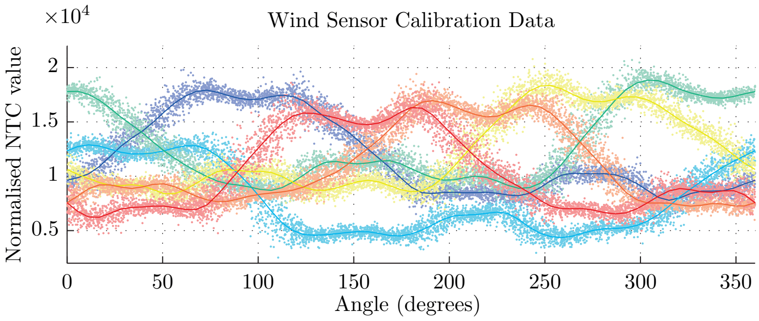


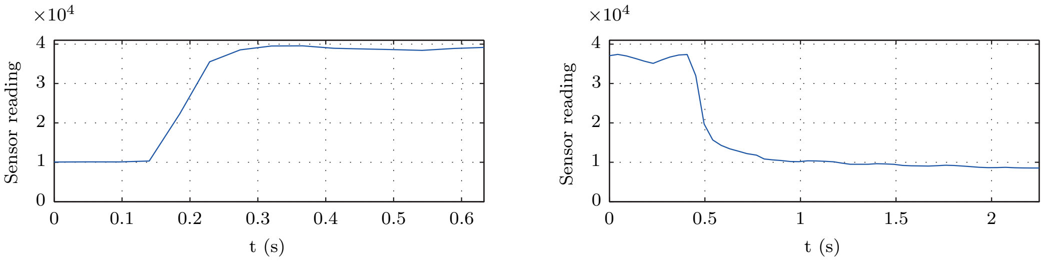
Figure 5.4 – Data obtained during the calibration of a wind sensing board. Each curve represents the response of one of the six thermistors over several complete rotations of the robot.

5.3.1 Wind sensing board

The wind direction sensing board [16] consists of a ring of six Honeywell 111-202CAK-H01 negative temperature coefficient (NTC) thermistors in a star-shaped 3D-printed enclosure. The board uses a log-likelihood model to estimate the wind direction from the raw response of the six sensors, an example of which is shown in Figure 5.4.

Accuracy depends on the characteristics of each specific board and varies for particular angles, but the standard error for a well-built and well-calibrated board is approximately 5–8°, normally distributed, measured over the whole range.

In reality, sensor error varies significantly across different boards and its distribution depends on the angle of incidence. Calibration is performed by rotating the robots in place in a uniform wind field and tuning the model using the captured data.



(a) **(b)**

Figure 5.5 – Odour sensor response (a) and recovery (b).

5.3.2 Odour sensing board

The odour sensing board [16] features a MiCS-5521 volatile organic compound (VOC) resistive sensor from SGX Sensortech. The sensor is responsive to a wide range of gases, including carbon monoxide, hydrocarbons, alcohols and glycols.

A small pump continuously drives air through the sensor package, resulting in comparatively short (sub-second) response times. Figure 5.5 presents the result of a rudimentary experiment to determine the sensor response, yielding a response time (to 90 %) of 0.13 s and a recovery time (to 10 %) of 0.41 s.

No response calibration is performed for the VOC sensors. Instead, before each trial, sensors are left to measure and average their readings over a three-minute period, leading to an approximate baseline that is then subtracted from the measurements taken during the experiments.

5.3.3 Relative positioning board

Inter-robot relative positions are provided by an infrared range and bearing board [27]. The board features a ring of sixteen infrared light-emitting diodes (LEDs) and eight receivers. The distance and bearing to each neighbour are estimated from the signal strength measured by the set of receivers.

The transmitted signal encodes the sender identifier so relative positions can be matched to a specific neighbour. The boards are configured to transmit at 10 Hz and calibrated by randomly moving a group of robots beneath an overhead tracking camera according to the procedure outlined in [28], with typical resulting standard errors of 10 % in range and 0.15 rad in bearing.

The calibration procedure works on the receiving side only and does not distinguish between different emitters, correcting for asymmetries in the receivers and for the average transmission power. Differences in transmission power among boards or emitters on the same board lead to less accurate distance measurements.

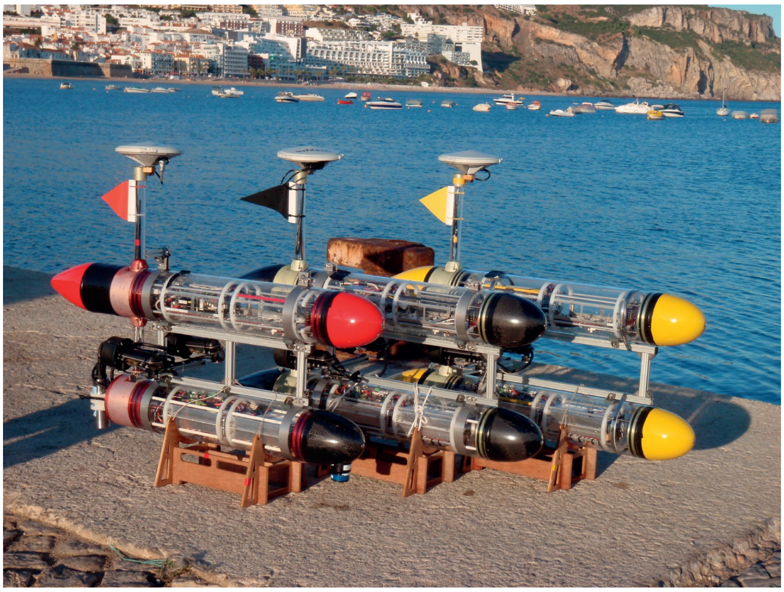


Figure 5.6 – The three MEDUSA AMVs at an experimental site.

5.4 MEDUSA autonomous marine vehicle

The MEDUSA, shown in Figure 5.6, is a small AMV developed by the Dynamical Systems and Ocean Robotics Laboratory (DSOR) at IST [29].

Each MEDUSA-class vehicle weighs approximately 30 kg and consists of two superimposed longitudinal acrylic housings of size 0.15 m × 1.035 m attached to an aluminium frame. Although the MEDUSA have since been extended with diving capabilities, at the time of use they were limited to surface operation.

The upper body is partially above the surface and carries an EPIC single-board computer with a 1.8 GHz Intel Atom D525 dual-core processor and 2 GB of RAM. Also on the upper body are an Ashtech Global Positioning System (GPS) receiver with real-time kinematic (RTK) capabilities and a VectorNav IMU and attitude and heading reference system (AHRS).

Most of the lower body is taken up by two seven-cell lithium polymer batteries, providing 32 A h of capacity and approximately 12 h of autonomy. Other sensors and devices, such as echosounders and Doppler velocity log (DVL) systems, can be added to the robots if required for an experiment.

The vehicle is propelled by two side-mounted, forward-facing SEABOTIX brushless stern thrusters that directly control surge and yaw motion, and can move at speeds up to $1.5\,\mathrm{m\,s^{-1}}$.

An 802.11 interface is used for surface communications, while a Tritech Micron acoustic modem enables underwater communication in the 20–28 kHz frequency band with maximum data rate of $40\,\mathrm{bit\,s^{-1}}$ and horizontal range of 500 m. The acoustic modem can be used for ranging, with a published accuracy of ±0.2 m when provided with the correct speed of sound.

The on-board computer runs a GNU/Linux distribution, and the software suite is a mix of C/C++ and MATLAB applications, running atop the MOOS middleware, a set of support and communication libraries and tools maintained by the Oxford Mobile Robotics Group. In practice, most low-level controllers are written in C, whereas high-level algorithms are implemented in MATLAB/Simulink.

6 Experimental sites

The bulk of the experimental work takes place in one of two settings: the DISAL wind tunnel at École Polytechnique Fédérale de Lausanne (EPFL), where we run experiments with the Khepera robots, and the lagoon surrounding the Lisbon Oceanarium, where we run experiments with the MEDUSA vehicles.

In this chapter, we describe these two experimental sites, albeit in different levels of detail: whereas in the case of the lagoon we are mere users, the wind tunnel required significant infrastructure work. Some preliminary tests and experiments took place in different settings that are not vital to the thesis.

6.1 Wind tunnel

The DISAL boundary layer wind tunnel occupies a specialised building at EPFL. The installation features two overlaid test channels, a lower channel for high wind speed trials, with a test section measuring 2 m × 1.5 m, and an upper channel for low wind speed trials, with a test section measuring 4 m × 2 m. Only the upper channel, shown in Figure 6.1, is used in our work. Owing to the construction of the wind tunnel and fixed gear installed inside, the usable volume is approximately 12 m × 3.5 m × 1.6 m.

The fan is actuated by a 90 kW motor driven by a variable speed controller. The tunnel is highly mechanised, with several parts being movable by hand or remotely, and can be operated in open or closed loop. Given the use of a chemical source inside the tunnel, our experiments always run in open loop, allowing for a maximum wind speed of $5\,\mathrm{m\,s^{-1}}$.

Most of the equipment installed in the wind tunnel is connected to an integrated control and automation system. A controller area network (CAN) field bus links the several components to a single Schneider Electric Lexium LMC20 programmable logic controller (PLC) that interacts with a supervisory control and data acquisition (SCADA) system over Open Platform Communications (OPC) on an Ethernet network. The same network also supports the NI-DAQ data acquisition system that is used to monitor sensors deployed inside the tunnel.

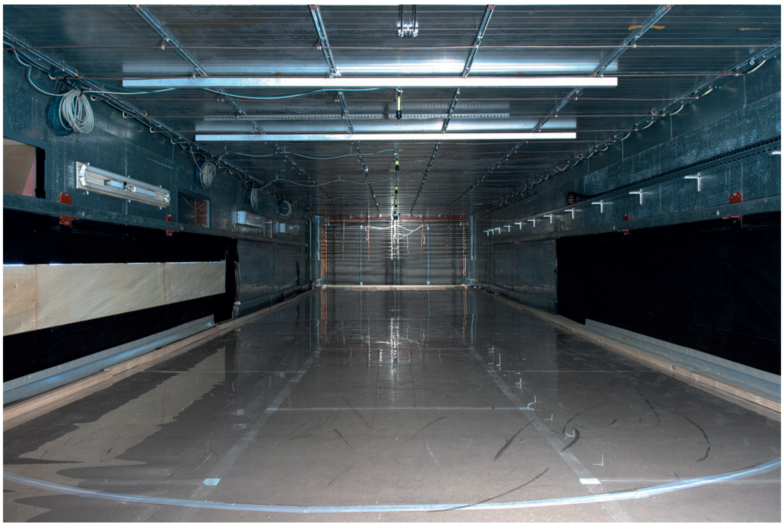


Figure 6.1 – Inside view of the wind tunnel test section.

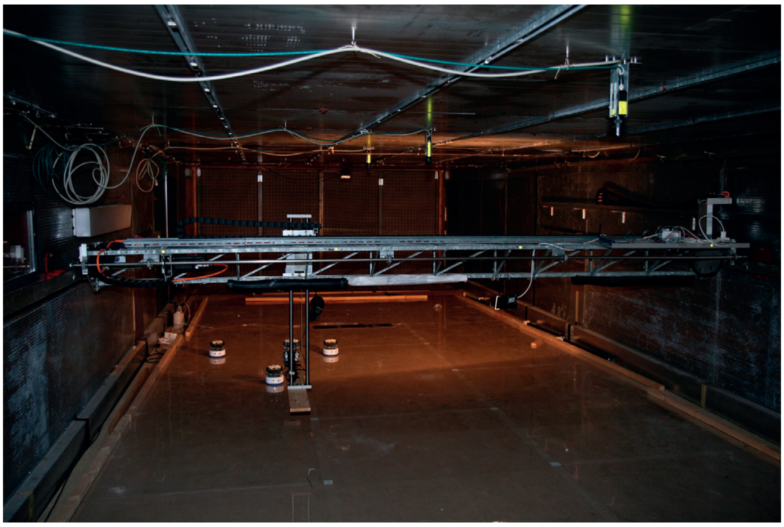


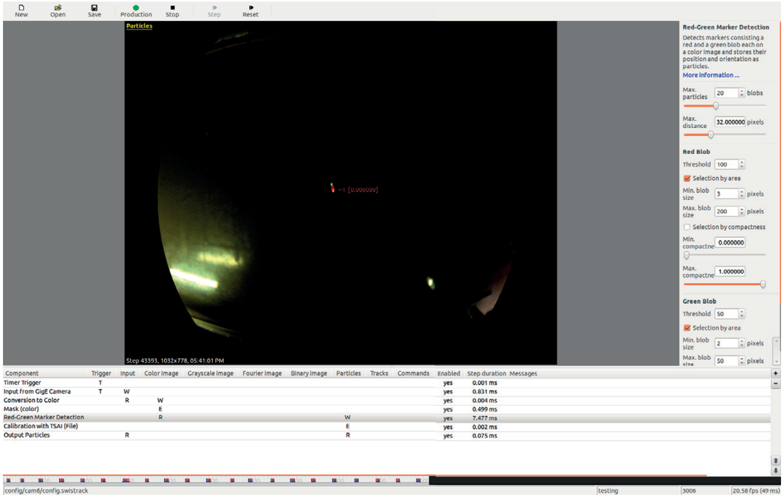
Figure 6.2 – Wind tunnel 3D traversing system, with Khepera IV robots in the background.

6.1.1 Traversing system

The wind tunnel is equipped with a three-axis traversing system, shown in Figure 6.2, which makes it possible to place a sensor or tool at arbitrary positions along the channel, remotely and with high accuracy.

The traversing system has previously been used to mount Pitot tubes, hot wire anemometers, and other analogue sensors, connected to the aforementioned NI-DAQ system. We use it to move a Khepera robot in 3D, emulating a flying robot.

We control the traversing system through a custom TCP-to-OPC proxy that allows any program to connect to a TCP port on the server and send simple text commands to obtain the system



Step 43393, 1032x778, 05:41:01 PM

Red-Green Marker Detection

Detects markers consisting a red and a green blob each on a color image and stores their position and orientation as particles.

Max. particles 20 blobs

Max. distance 32.000000 pixels

Red Blob

Threshold 100

☑ Selection by area

Min. blob size 3 pixels

Max. blob size 200 pixels

☐ Selection by compactness

Min. compactne 0.000000

Max. compactne 1.000000

Green Blob

Threshold 50

☑ Selection by area

Min. blob size 2 pixels

Max. blob size 50 pixels

Component	Trigger	Input	Color image	Grayscale image	Fourier image	Binary image	Particles	Tracks	Commands	Enabled	Step duration	Messages
Timer Trigger	T									yes	0.001 ms	
Input from GigE Camera	T	W								yes	0.831 ms	
Conversion to Color		R	W							yes	0.004 ms	
Mask (color)			E							yes	0.499 ms	
Red-Green Marker Detection			R				W			yes	7.477 ms	
Calibration with TSAI (File)							E			yes	0.002 ms	
Output Particles		R					R			yes	0.075 ms	

config/cam6/config.swistrack | testing | 3006 | 20.58 fps (49 ms)

Figure 6.3 – SwisTrack visual tracking software.

state or move to a new position.

6.1.2 Positioning system

In order to position the robots within the wind tunnel floor plane, we employ an overhead tracking system consisting of six Basler Scout Gigabit Ethernet cameras. The cameras are mounted on the ceiling along the centre of the tunnel and are connected to a single workstation that processes the incoming raw video to determine the absolute position of each LED robot marker.

Tracking is accomplished using the open-source SwisTrack software [30], developed at DISAL and shown in Figure 6.3. Six separate instances of the application translate the image coordinates from each camera into real-world coordinates, using a previously calibrated Tsai transform [31]. The data are then output to a TCP socket and received by an external application that merges the individual detections, ensuring that tracks are not disrupted when transitioning between cameras.

To cope with the volume of data generated by the cameras and the load of processing it, we set up a powerful Intel Xeon workstation featuring six 3.2 GHz processing cores. The cameras are connected to two Intel I350-T4 Gigabit Ethernet server adapters using the Basler performance driver, which provides direct memory access (DMA) capabilities for GigE Vision packets, significantly reducing CPU loads.

The low ceiling height mandates the use of wide angle objectives with significant optical distortion. Nevertheless, after manual simultaneous calibration using fifty points distributed along the tunnel, the system achieves a positioning accuracy under 8 cm, enough to prevent robot identifiers from being mixed up.

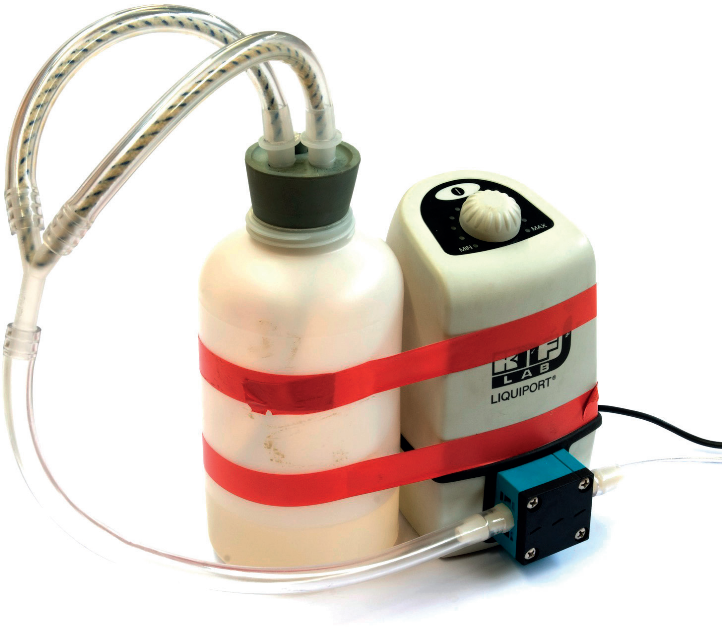


Figure 6.4 – Odour pump and ethanol reservoir.

6.1.3 Odour source

The odour tracing experiments use a plume of A15-A absolute alcohol (99.9 % purity ethanol). Although invisible, it is detectable by our VOC sensors, readily available, and harmless at low concentrations.

The plume is generated using a low speed pump (see Figure 6.4) to circulate $1.2\,\mathrm{l\,min^{-1}}$ of air through a bottle of ethanol, which evaporates at room temperature. The gas mixture is released through a rubber hose of diameter 1 cm, placed at different positions near the inlet end of the tunnel.

While we lack the calibrated equipment required to determine the absolute odour concentration at the outlet, it is primarily a function of the ambient air temperature, and therefore assumed to remain constant throughout an experiment. The absolute concentration is not, in any event, critical to our approach, which acts on concentration differences.

6.2 Lisbon Oceanarium lagoon

The Lisbon Oceanarium is located in Parque das Nações, a former industrial estate redeveloped for the 1998 Lisbon World Exposition. It sits atop a privately managed man-made lagoon used for small boat sailing training and other water sports. Owing to its convenient location and facilities, it serves as the primary testing ground for DSOR, where we run most MEDUSA

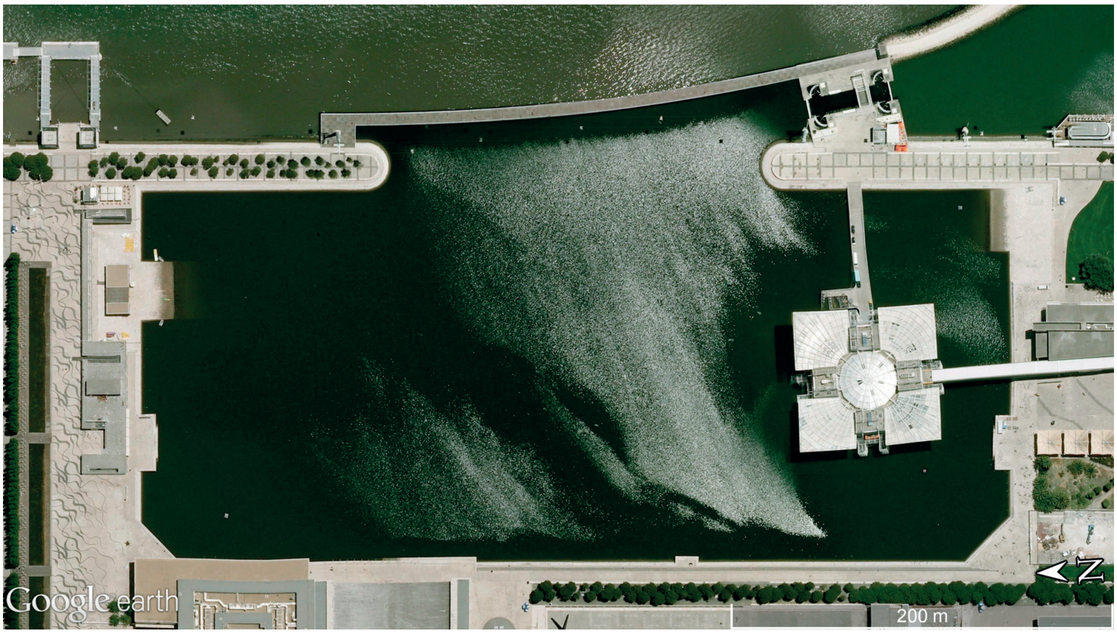


Figure 6.5 – Satellite view of the Lisbon Oceanarium test site. (Image credit: Google Inc.)

experiments.

Figure 6.5 shows a satellite view of the lagoon and surrounding areas. The oceanarium is visible on the right side of the image, and on the left is a concrete slipway for launching and hauling out vehicles, immediately in front of the building that houses our makeshift control centre.

A sliver of the Tagus river estuary can be seen along the top edge. The lagoon is separated from the estuary by a barrier underneath the walkway, and the two only communicate during high tide. The inflow is mostly composed of seawater and leads to high salinity in the lagoon, while the barrier guarantees minimal currents affecting the vehicles.

With a usable area of 330 m × 180 m, the lagoon provides extensive space for testing our algorithms, even when running multiple vehicles over long trajectories and in spaced-out formations. Depth, however, is generally under 5 m, which can impact the performance of the acoustic communication and ranging systems.

A wired Ethernet network connected to an 802.11 access point and a high-gain antenna supports communication between the vehicles, operators and other devices. A RTK GPS base station is deployed at roof level near the control centre and provides corrections to the robots over the network. While the bound on the absolute GPS error depends on the accuracy of the computed position of the base station, relative positions in the area of operations should be accurate to the sub-centimetre level.

7 Simulators

Different hardware platforms and scenarios require matching simulation environments. The three main tools used for this purpose are MATLAB/Simulink, Webots and ANSYS Fluent.

The use of MATLAB is transversal to the thesis but, as simulator, it is mostly used for experiments involving the MEDUSA, taking advantage of an existing detailed dynamical model of the vehicle. Webots is used as a simulator for the Khepera robots. Fluent is primarily used to study plume behaviour under different conditions and generate waterborne plumes.

The various simulation set-ups are described in the upcoming sections.

7.1 MATLAB/Simulink

MATLAB is a widely used numerical computing environment. Combined with Simulink, a graphical environment for modelling dynamic systems, it becomes a powerful tool for robot simulation.

We primarily use MATLAB and Simulink in simulations involving the MEDUSA vehicles. Not only do they provide a convenient toolchain for simulating complex systems, but the controllers on the real vehicles also use MATLAB. The same implementation of an algorithm is usable in simulation, in real-world experiments, and in hardware-in-the-loop (HIL) tests, an important step prior to deployment in an uncontrolled environment.

A detailed model of the MEDUSA was already available for our use [29], having been previously fitted with extensive experimental data and providing an accurate representation of the real dynamics of the vehicle. As we consider the MEDUSA to be bound to the surface plane, its kinematic equations take the form

$$\dot{x} = u\cos\psi - v\sin\psi \tag{7.1}$$

$$\dot{y} = u\sin\psi + v\cos\psi \tag{7.2}$$

$$\dot{\psi} = r \tag{7.3}$$

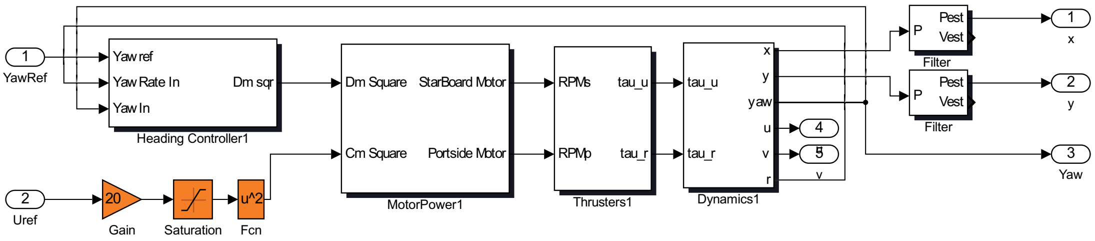

1
YawRef
Yaw ref
Yaw Rate In
Yaw In
Dm sqr
Heading Controller1
2
Uref
20
Gain
Saturation
u^2
Fcn
Dm Square
StarBoard Motor
Cm Square
Portside Motor
MotorPower1
RPMs
tau_u
RPMp
tau_r
Thrusters1
tau_u
tau_r
x
y
yaw
u
v
r
Dynamics1
4
5
v
P
Pest
Vest
Filter
P
Pest
Vest
Filter
1
x
2
y
3
Yaw

Figure 7.1 – Top-level view of the MEDUSA Simulink model.

where u (surge speed) and v (sway speed) are the body axis components of the velocity of the vehicle, x and y are the Cartesian coordinates of its centre of mass, ψ defines its orientation (heading angle), and r its angular velocity. The motions in heave, roll and pitch can be neglected as the vehicle has large enough metacentric height. The simplified dynamic equations of motion for surge, sway and yaw are

$$m_u \dot{u} - m_v v r + d_u u = \tau_u \tag{7.4}$$

$$m_v \dot{v} + m_u u r + d_v v = 0 \tag{7.5}$$

$$m_r \dot{r} - m_{uv} u v + d_r r = \tau_r \tag{7.6}$$

where τ_u stands for the external force in surge (thruster common mode), τ_r for the external torque (thruster differential mode), and the terms

$$m_u = m - X_{\dot{u}} \tag{7.7}$$

$$m_v = m - Y_{\dot{v}} \tag{7.8}$$

$$m_r = I_z - N_{\dot{r}} \tag{7.9}$$

$$m_{uv} = m_u - m_v \tag{7.10}$$

$$d_u = -X_u - X_{|u|u}|u| \tag{7.11}$$

$$d_v = -Y_v - Y_{|v|v}|v| \tag{7.12}$$

$$d_r = -N_r - N_{|r|r}|r| \tag{7.13}$$

represent vehicle masses, hydrodynamic added masses, and linear and quadratic hydrodynamic damping effects.

The simulation also incorporates a model of the thruster dynamics, as well as the MEDUSA low-level heading and speed controllers. The full details of the models and physical parameters can be found in [29], while a high-level view of their Simulink implementation is presented in Figure 7.1.

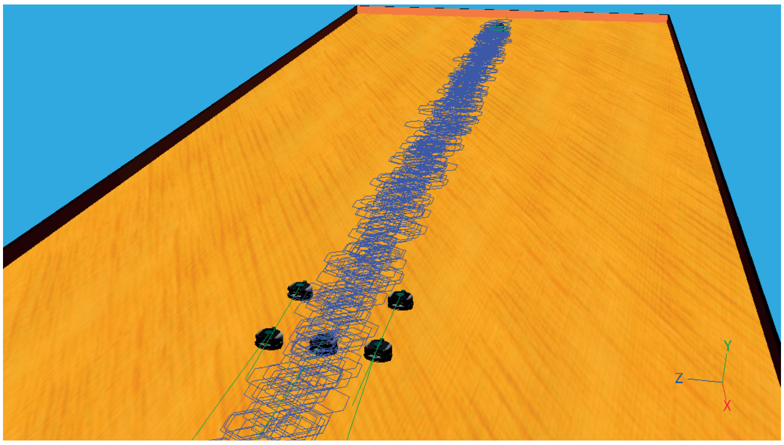


Figure 7.2 – Plume tracing simulation in Webots, using five Khepera III robots. The odour source is at the far end, and the filaments are represented in purple.

7.2 Webots

Webots is a 3D mobile robotics simulator that provides the means to model robots at the submicroscopic level, with simplified but realistic physics, resulting in a simulation behaviour that accurately mirrors that of real robots. Currently in its eighth major release, it has been developed since 1998 by Cyberbotics in close cooperation with EPFL laboratories and is used by universities and research centres worldwide.

Webots provides a flexible platform with built-in support for several programming languages (C, C++, Java, MATLAB, and Python), robot models (including the Aibo, Bioloid, e-puck, iRobot Create, Khepera III, Koala, Nao, Pioneer, and Surveyor SVR-1) and sensors (distance, light, cameras, LIDAR, GPS, IMU, compass, contact, among others). The Open Dynamics Engine (ODE) library is used for faithful simulation of physical phenomena, and a plug-in interface makes it possible to implement additional mechanisms, such as more complex physics or visual representations.

In spite of the extensive functionality provided by the software, it is not possible to simulate odour propagation out of the box. We take advantage of a plug-in previously developed at DISAL to model discrete particle propagation by action of the wind [16], which we have updated to work with modern versions of Webots. Figure 7.2 shows Webots running a plume tracing simulation using this plug-in.

7.2.1 Wind simulation and sensing

For most experiments, we use a constant wind vector of intensity $1\,\mathrm{m\,s^{-1}}$, simulating laminar flow. The wind sensor provides noisy wind velocity measurements, resulting of the sum of the wind velocity at the position of the robot, w, and a vector of three independent Gaussian random variables, $N(0,\sigma_a^2)$.

$$\mathbf{a_s}(p) = \begin{pmatrix} w_x \\ w_y \\ w_z \end{pmatrix} + \begin{pmatrix} N(0,\sigma_a^2) \\ N(0,\sigma_a^2) \\ N(0,\sigma_a^2) \end{pmatrix} \tag{7.14}$$

As the wind field is constant, the magnitude of w is also constant and only its orientation changes. The standard deviation of the Gaussian noise is set to $\sigma_a = 0.1\,\mathrm{m\,s^{-1}}$. This differs from the noise distribution of the real sensor, for which the experimentally determined distribution of the *directional* noise is $N(0,\sigma_d^2)$, $\sigma_d = 5$–8°.

7.2.2 Odour simulation and sensing

The odour propagation implementation is based on the filament model proposed in [32], and generates an intermittent plume similar to the one observed in the wind tunnel. Odour is simulated as a set of filaments ($i = 0,...,N$), each containing a constant amount $s = 8.3 \times 10^9$ of molecules. Each filament is defined by the (x, y, z) position of its geometric centre $\mathbf{p}_{i,t}$ and by its width $w_{i,t}$, both varying over time t.

At each simulation step, the position of a filament is updated according to the wind flow and a stochastic process $\mathbf{v}_p$, consisting of a vector of three independent Gaussian random variables, $N(0,\sigma_p^2)$, with $\sigma_p = 0.1\,\mathrm{m}$. Molecular dispersion is modelled by having the filament width increase with time while the peak concentration decreases. The resulting evolution of the filaments is described by

$$\mathbf{p}_{i,t+\Delta t} = \mathbf{p}_{i,t} + \mathbf{a}(\mathbf{p}_{i,t})\Delta t + \mathbf{v}_p \tag{7.15}$$

$$w_{i,t+\Delta t} = w_{i,t} + \frac{\gamma}{2w_{i,t}} \tag{7.16}$$

The filament dispersion rate approximating the wind tunnel conditions was previously determined to be $\gamma = 4 \times 10^{-7}\,\mathrm{m^2\,s^{-1}}$. The virtual odour source releases 100 filaments per second with initial width $w_{i,0} = 10\,\mathrm{cm}$ and initial position distributed over the circular area of the source.

The odour concentration at time t and position $\mathbf{p}$ is the sum of the concentration contribution of all filaments, which decays exponentially with the increasing distance to the centre of a filament, that is,

$$c_t(\mathbf{p}) = \sum_{i=0}^{N} \frac{s}{w_{i,t}^3} \exp\left(-\frac{|\mathbf{p}-\mathbf{p}_{i,t}|}{w_{i,t}^2}\right) \tag{7.17}$$

No noise is added to the measured concentration as the chemical-to-electrical transduction noise of the real sensor was observed to be negligible [16]. We also do not mimic the response and recovery times of the real sensor in the plug-in, but the use of a sliding window max filter in our algorithms overshadows the long recovery time.

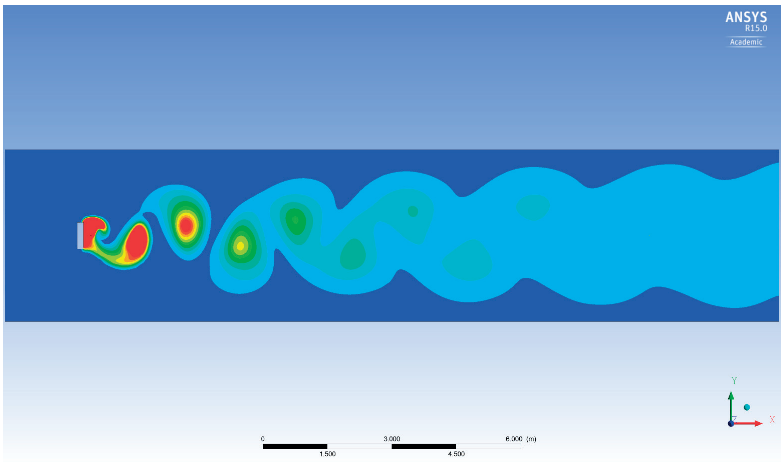


Figure 7.3 – Overhead view of an unsteady ethanol plume simulation using ANSYS Fluent.

7.3 ANSYS Fluent

Fluent is a powerful computational fluid dynamics (CFD) simulation package developed and commercialised by ANSYS. It is part of a comprehensive suite that includes tools for modelling, meshing, simulation, and post-processing, and that can interface with applications handling different physics, e.g. electromagnetism.

As a CFD tool, Fluent simulates the displacement of fluids by modelling the underlying physical phenomena. Broadly speaking, this translates to numerically solving the Navier-Stokes equations at each simulation step. Additional models can be integrated to handle, for instance, heat exchange or acoustics.

We use Fluent to ascertain the behaviour of the plume without the simplifications that underpin the Webots implementation or the overhead involved in wind tunnel experiments. The simulations we run generally use the pressure-based solver and the realisable k–ϵ turbulence model. Depending on the results of interest, the chemical plume is modelled using either discrete phase or species transport.

The full set of simulations carried out in Fluent include transient and steady state runs, in water and air, using different arenas and conditions. Figure 7.3 shows a snapshot of an unsteady ethanol plume in a virtual wind tunnel.

8 Summary

This part of the thesis introduced the major tools used in the course of our work, including the robots, infrastructure and simulators. No tool was used at every stage, each serving a particular purpose in one or more of our scenarios. Throughout this thesis, we list the specific set-up used for each activity and experiment.

The number and diversity of tools presented a significant challenge, with the resulting set described here being the result of extensive preparation, analysis, and development work. While no new hardware was developed, we integrated and assessed new solutions, including the newly released Khepera IV robot, and engaged in major and minor improvements and repairs, with considerable time and effort invested into the wind tunnel.

As part of our work, we also contributed numerous software fixes and improvements to SwisTrack, the Webots odour plug-in, the Khepera III Toolbox, and the new Khepera IV Toolbox.

Range-based formation control

Part III

‘Hawkshead Attacking Formation’ he read as he watched the three Irish Chasers zoom closely together, Troy in the centre, slightly ahead of Mullet and Moran, bearing down upon the Bulgarians.

— J. K. Rowling, *Harry Potter and the Goblet of Fire*

9 Introduction

Developments in the field of autonomous marine vehicles, with increasingly powerful and affordable vehicles coming to market, open the door for a multitude of novel applications in fields as diverse as biology, geophysics, and defence.

Many of the tasks envisioned to be within the reach of multi-AUV groups in the near future require the vehicles to actively cooperate; conventional examples include seabed mapping and ocean sampling. Some go further and call for the vehicles to move in formation, i.e. while maintaining relative positions to one another. We later introduce plume tracing strategies that share the latter requirement.

Underwater vehicles, however, face significant localisation and communication constraints when compared with their land or air equivalents. There is no single system providing global absolute localisation, and communication is severely restricted in range (for radio-frequency approaches) or in throughput and latency (for the more common acoustic approaches).

In this part of the thesis, we address the problem of triangular formation keeping under severe communication and localisation constraints. For a reference scenario consisting of two localised leader vehicles on the surface and an underwater follower vehicle, we use acoustic ranging and communications to establish and maintain a moving formation. Of the multiple real-world applications matching this scenario, a typical one is surface-guided underwater surveying. We make a realistic assumption that the AUV has independent depth control, and focus on formation control on a 2D plane only.

We start by introducing a range-only approach to formation control using a kinematic model for the follower vehicle written in terms of speed and heading [19]. We propose a control strategy that estimates the formation motion from the ranges measured to the two leading vehicles and uses simple feedback laws for speed and heading to drive suitably defined common- and differential-mode errors to zero. Along with the controllers, we include a local proof of convergence of the distance errors under straight-line motion.

We subsequently discuss adapting this solution to a MEDUSA AMV [20], describing the constraints imposed by the platform and the modifications required. These include, for instance, estimating the continuous distance signals from discrete samples and obtaining the formation heading through acoustic data exchange. Our range-only theoretical approach develops into a range-based practical implementation.

Results are presented from both MATLAB simulations and real-world trials with the MEDUSA vehicles. We finish by introducing an alternative formulation that generalises our control framework to support different formations.

10 Related work

Related work on formation control can be found in [33], [34], which describe a leader-follower control problem using a formation graph with an arbitrary number of robots. The authors discuss approaches for both range-bearing and range-range control, depending on the available sensors; in both cases, knowledge of the leader motion is assumed. A distinct graph-based leader-follower solution using range and bearing is proposed in [35] and supported by robot experiments.

A strategy for a four-robot station keeping problem is proposed in [36], using exclusively range measurements and assuming holonomic vehicles described by simple kinematic points. A similar scenario is considered in [37], although global convergence is only proved for a triangular formation. Bearing-only methods are also available for square [38] and triangular formations [39], [40].

In [41], the authors advance algorithms to coordinate a formation of mobile agents that are only able to measure ranges to their immediate neighbours. This solution requires that subsets of non-neighbouring agents determine the relative positions of their neighbours while these are stationary, and only then move to minimise a particular cost function.

There are numerous strategies designed with marine vehicles in mind. A solution that decouples the controllers for formation shape, formation motion and vehicle orientation yet requires position information is proposed in [42]. Coordinated path following approaches are presented in [43] and [44], the latter specifically dealing with underwater pipeline inspection. These assume that the path to be followed is known to all vehicles, and generally work by exchanging some along-path synchronisation measure.

An example real-world AUV operation is documented in [45], where a group of Slocum gliders adopt a virtual body and artificial potential (VBAP) approach [46] to maintain a triangular formation while performing various tasks. The vehicles independently navigate using dead reckoning while underwater and resurface periodically to get an updated mission plan from an onshore computer.

11 Technical approach

The control problem discussed in this part is illustrated in Figure 11.1, which shows two leading vehicles (vehicles 2 and 3, represented by their positions x_2 and x_3), moving along some path, and a follower (vehicle 1, represented by its position x) which we control. Through the remainder of this part, and unless otherwise noted, the absence of an index indicates a variable or parameter related to vehicle 1, the controlled, trailing, or following vehicle.

The objective is for the trailing vehicle to follow the leaders in a triangular formation, at the same desired distance d from each. In the figure, this is tantamount to x converging to the desired position x_d. There exists a symmetric solution to the problem, with the desired position x_d mirrored in relation to the segment defined by $\overline{x_2 x_3}$. We designate the solution shown in Figure 11.1 by *following motion*, and the mirrored solution by *leading motion*. We only deal with the case of following motion, to which the vehicle should converge provided it starts on that side of $\overline{x_2 x_3}$.

Let $z_i = \|x_i - x\|; i = 2,3$ denote the distances from the trailing vehicle to each of the leaders. From the range measurements, we define the common- and differential-mode errors

$$\epsilon = \frac{e_2 + e_3}{2} = \frac{z_2 + z_3}{2} - d \tag{11.1}$$

$$\delta = e_3 - e_2 = z_3 - z_2 \tag{11.2}$$

with $e_i = z_i - d; i = 2,3$. The errors loosely correspond to longitudinal and transversal errors relative to x_d and the $\overline{x_2 x_3}$ segment.

The control problem consist of deriving control laws to drive ϵ and δ to zero or, equivalently, to drive x to x_d.

11.1 Range-only formation control

In this section, we describe the theoretical derivation of the formation controller, respecting the constraints previously mentioned but ignoring practical considerations related to the

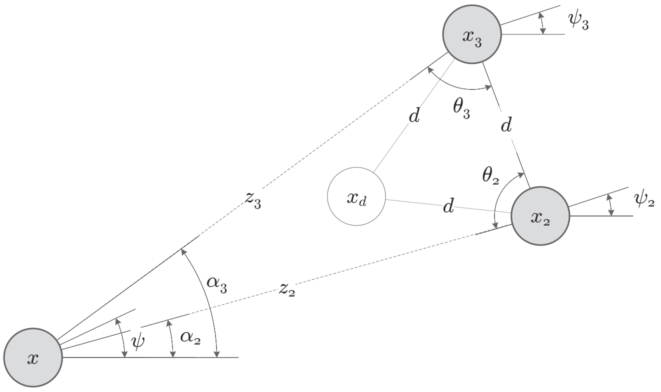


Figure 11.1 – System of three vehicles (1, 2, 3) and their intended triangular formation (x_d, x_2, x_3). Vehicle 1 is the follower, and vehicles 2 and 3 are the leaders. The image shows many of the relevant parameters, including the formation and independent vehicle headings, as well as the relationships (ranges and bearings) between them.

implementation using a real robot. The outcome is a strategy designed to regulate the motion of the controlled (trailing) vehicle using only range information.

We assume that the vehicle starts from a following position in order to converge to a following motion and that the leader vehicles (2 and 3) move at a distance d from each other, according to simple kinematics described by

$$\dot{x}_i = \begin{bmatrix} v_i \cos\psi_i \\ v_i \sin\psi_i \end{bmatrix}, \quad i = 2,3 \tag{11.3}$$

where v_i and ψ_i denote the speed and heading of vehicle i, and $x_i \in R^2$ denotes its position. The control signals are the speed v and heading ψ of the follower vehicle, whose kinematic model is given by

$$\dot{x} = \begin{bmatrix} v \cos\psi \\ v \sin\psi \end{bmatrix} \tag{11.4}$$

We further assume that both leaders move with a common heading $\psi_f = \psi_2 = \psi_3$, and that the total velocity vector of each leading vehicle is always perpendicular to the line segment that joins them. The heading ψ_f is unknown to vehicle 1, and the mean formation speed is given by $v_f = (v_2 + v_3)/2$.

Separate controllers are designed to stabilise each error measure, with the speed controller stabilising the common-mode error and the heading controller stabilising the differential-mode error. The following subsections describe the controllers and provide proofs of convergence under linear motion, first for each controller in isolation and then for the overall system.

11.1.1 Error dynamics

From the definition of the ranges z_i, it follows that

$$\dot{z}_i = v_i \cos(\alpha_i - \psi_f) - v \cos(\alpha_i - \psi) \tag{11.5}$$

Simple computations show that the relations between the angles α_i and the interior angles θ_i of the triangle in Figure 11.1 are given by

$$\alpha_2 = \theta_2 + \psi_f - \frac{\pi}{2} \tag{11.6}$$

$$\alpha_3 = -\theta_3 + \psi_f + \frac{\pi}{2} \tag{11.7}$$

The law of cosines yields the following expressions for θ_i:

$$\theta_2 = \arccos\left(\frac{z_2^2 + d^2 - z_3^2}{2\,d\,z_2}\right) \tag{11.8}$$

$$\theta_3 = \arccos\left(\frac{z_3^2 + d^2 - z_2^2}{2\,d\,z_3}\right) \tag{11.9}$$

Although the control strategy can be applied to other types of trajectories, the next sections assume the simpler case of constant-speed linear motion for the two leading vehicles. This means that $v_2 = v_3 = v_f$ and the simplified error dynamics for ϵ and δ become

$$\dot{\epsilon} = \cos\beta \big(v_f \cos\varphi - v \cos\big(\varphi + \tilde{\psi}\big)\big) \tag{11.10}$$

$$\dot{\delta} = 2 \sin\beta \big(v_f \sin\varphi - v \sin\big(\varphi + \tilde{\psi}\big)\big) \tag{11.11}$$

where

$$\beta = \frac{\theta_2 + \theta_3}{2} - \frac{\pi}{2} \tag{11.12}$$

$$\varphi = \frac{\theta_2 - \theta_3}{2} \tag{11.13}$$

and $\tilde{\psi} = \psi_f - \psi$ is the heading error.

11.1.2 Speed controller

The desired speed of the controlled vehicle depends on its position relative to its intended place in the formation. The vehicle should speed up if trailing behind and slow down if too close to the leaders. In steady state, the speed v should converge to the unknown mean leader speed v_f.

When the leader vehicles follow a linear trajectory with constant speed v_f, $\psi = \psi_f$ and $\delta = 0$ (i.e. x is on the perpendicular bisector of the $\overline{x_2 x_3}$ line segment), the dynamics of ϵ in (11.10) reduce to

$$\dot{\epsilon} = \cos\beta\left(v_f - v\right) \tag{11.14}$$

Since $\cos\beta \geq 0$, a control law $v = v_f + K_p^s \epsilon$, $K_p^s > 0$ exponentially stabilises the origin $\epsilon = 0$, provided $\beta \neq -\frac{\pi}{2}$, which would indicate a collapsed line formation.

As v_f is unknown, we must adapt to it. Therefore, we propose the following speed controller to regulate the common-mode error ϵ to zero:

$$v = K_p^s \epsilon + K_i^s \int_0^t \epsilon \, d\tau \tag{11.15}$$

where $K_p^s > 0$ and $K_i^s > 0$ are the proportional and integral gains, respectively. Theorem 1 outlines the conditions for guaranteed exponential stability of the system in closed loop with the proposed controller.

Theorem 1. Consider the overall system comprising two leader vehicles (2 and 3) describing a linear trajectory with constant but unknown speed $v_f > 0$. Suppose that $\psi = \psi_f$ and that the controlled vehicle lies in a neighbourhood of the perpendicular bisector of $\overline{x_2 x_3}$, in a following position. Then, for a sufficiently small integral gain K_i^s (with respect to K_p^s), control law (11.15) locally exponentially stabilises ϵ to the origin $\epsilon = 0$.

Proof. (Theorem 1) Defining $\dot{\xi} = K_i^s \epsilon$, we get $v = \xi + K_p^s \epsilon$. Introducing the error term $\tilde{v} = \xi - v_f$ yields $\dot{\tilde{v}} = \dot{\xi} - \dot{v}_f = K_i^s \epsilon$. From (11.14), we obtain

$$\begin{bmatrix} \dot{\epsilon} \\ \dot{\tilde{v}} \end{bmatrix} = \begin{bmatrix} -K_p^s \cos\beta & -\cos\beta \\ K_i^s & 0 \end{bmatrix} \begin{bmatrix} \epsilon \\ \tilde{v} \end{bmatrix} \tag{11.16}$$

Linearising system (11.16) at the equilibrium point corresponding to the equilateral triangular position ($\epsilon = 0$, $\delta = 0$, $\beta^* = -\frac{\pi}{6}$), it follows that the eigenvalues of the state matrix

$$\lambda_1 = \frac{1}{2}\left(-K_p^s \cos\beta^* - \sqrt{\cos\beta^* \left(-4K_i^s + K_p^{s\,2} \cos\beta^*\right)}\right) \tag{11.17}$$

$$\lambda_2 = \frac{1}{2}\left(-K_p^s \cos\beta^* + \sqrt{\cos\beta^* \left(-4K_i^s + K_p^{s\,2} \cos\beta^*\right)}\right) \tag{11.18}$$

are negative for $K_i^s < \frac{\sqrt{3}}{8} K_p^s$, thus proving the theorem. ■

11.1.3 Heading controller

In steady state (with $x = x_d$) and while moving forward, the heading of the follower vehicle should match the heading of the leaders. When the follower is not in its designated position in the formation, the heading must change to correct for any lateral deviations: if closer to x_2, it must steer towards x_3, and vice-versa.

For the same constant-speed linear motion but now with $\epsilon = 0$ and $v = v_2 = v_3$, (11.11) becomes

$$\dot{\delta} = 2 v_f \sin\beta \left(\sin\varphi - \sin\left(\varphi + \tilde{\psi}\right)\right) \tag{11.19}$$

This leads us to propose the following heading control law that uses the differential-mode error δ:

$$\psi = \hat{\psi}_f + \gamma\left(K_p^h \delta\right) \tag{11.20}$$

where $K_p^h > 0$, $\hat{\psi}_f$ denotes an estimate of the formation heading ψ_f, and γ is any function such that $\sin(\gamma(ay))y > 0, \forall a > 0$. An example is the saturation function $\gamma(y) = \frac{\pi}{2}\,\text{sat}(y)$. The use of γ is due to the control law appearing within a sine argument in (11.19).

We can now introduce Theorem 2, dealing with the stability of δ.

Theorem 2. Consider the overall system comprising two leader vehicles (2 and 3) describing a linear trajectory with constant speed $v_f > 0$. Suppose that ψ_f is known, $v = v_f$, the controlled vehicle lies in a trailing position, and there exist positive constants $D > \mu > 0$ such that $d + \mu < z_2 + z_3 < D$. Then, the control law (11.20) stabilises δ to the origin $\delta = 0$.

The following result is instrumental in proving Theorem 2:

Proposition 1. Suppose that $z_2 + z_3 > d$. Then, the sign of $\sin\varphi$ is the same as the sign of δ, i.e. $\sin(\varphi)\delta > 0$.

Proof. (Proposition 1) For $\delta = 0$, $z_2 = z_3$ and the terms in φ cancel out. For $\delta > 0$ and $z_2 + z_3 > d$, we have

$$\begin{aligned} \frac{z_3^2 + d^2 - z_2^2}{2\,d\,z_3} &> \frac{z_2^2 + d^2 - z_3^2}{2\,d\,z_2} \\ \implies \theta_2 &> \theta_3 \\ \implies \varphi &> 0 \end{aligned} \tag{11.21}$$

An analogous argument can be derived for $\delta < 0$. ■

Remark 1. The condition $z_2 + z_3 \geq d$ is a physical constraint, since vehicles 2 and 3 are separated by a distance d. When $z_2 + z_3 = d$, vehicle 1 is located along the line segment $\overline{x_2 x_3}$. This does not fit the definition of following motion.

Proof. (Theorem 2) Define the candidate Lyapunov function

$$V_h = \frac{1}{2}\delta^2 > 0 \tag{11.22}$$

Using (11.19) and the control law (11.20) we obtain

$$\begin{aligned}\dot{V}_h &= 2v_f \sin\beta\left(\sin\varphi - \sin\varphi\cos\tilde{\psi} - \cos\varphi\sin\tilde{\psi}\right)\delta \\ &= 2v_f \sin\beta\left(\left(1-\cos\left(\gamma\left(K_p^h\delta\right)\right)\right)\sin(\varphi)\delta + \cos\varphi\sin\left(\gamma\left(K_p^h\delta\right)\right)\delta\right) \\ &< 0\end{aligned} \tag{11.23}$$

The last condition stems from the fact that $\sin(\varphi)\delta > 0$, $\sin\left(\gamma\left(K_p^h\delta\right)\right)\delta > 0$ and that, because $d+\mu < z_2 + z_3 < D, D > \mu > 0$, there exist positive constants $c_i, i = 1,2,3$ such that $\sin\beta < -c_1$, $\cos\beta > c_2$ and $\cos\varphi > c_3$. ■

11.1.4 Stability of the overall system

We have shown that our two controllers, one for the speed v and one for the heading ψ, each converge in isolation. However, in practice, the two controllers need to operate in parallel. In Theorem 3, we outline conditions in which we are able to guarantee stability of the overall system with both controllers in closed loop.

While the theorem and proof only address local stability (i.e. for positions x within a small neighbourhood of x_d), experiments show that the controller stabilises ϵ and δ for an extended region of initial states.

Theorem 3. Consider the overall system comprising two leader vehicles (2 and 3) describing a linear trajectory with constant speed $v_f > 0$. Suppose that ψ_f is known and that the controlled vehicle lies in a small neighbourhood of the desired position x_d. Then, for appropriately chosen gains K_i^s and K_p^s, control laws (11.15) and (11.20) stabilise ϵ and δ to the origin $\epsilon = \delta = 0$.

Proof. (Theorem 3) Define $\mathbf{x} = \begin{bmatrix}\epsilon & \tilde{v} & \delta\end{bmatrix}^T$ and let

$$\dot{\mathbf{x}} = f(\mathbf{x}) \tag{11.24}$$

be the dynamics equations (11.10) and (11.11) in closed loop with (11.15) and (11.20). Also let

$$\dot{\mathbf{x}} = f^*(\mathbf{x}) \tag{11.25}$$

be the simplified dynamics (11.14) and (11.19), also in closed loop.

We first show that the equilibrium point $\mathbf{x} = 0$ is locally exponentially stable. To this effect, we consider the Lyapunov function

$$V = \mathbf{x}^T P\mathbf{x} \tag{11.26}$$

with

$$P = \begin{bmatrix} \frac{(K_i^s+\cos\beta^*)\sec^2\beta^*}{2K_p^s} & \frac{1}{2}\sec\beta^* & 0 \\ \frac{1}{2}\sec\beta^* & \frac{1+(K_p^s)^2+K_i^s\sec\beta^*}{2K_i^sK_p^s} & 0 \\ 0 & 0 & \frac{1}{2} \end{bmatrix} \tag{11.27}$$

Computing its time derivative along the trajectories (11.25) yields

$$\dot{V} \le a_\epsilon \epsilon^2 + a_{\tilde{v}}\tilde{v}^2 + a_\delta \delta \tag{11.28}$$

where

$$a_\epsilon = -\rho + \frac{1}{2}(1-\rho)\left(K_p^s + \frac{1}{K_p^s} + \frac{K_i^s}{K_p^s}\sec\beta^*\right) \tag{11.29}$$

$$a_{\tilde{v}} = -K_i^s \sec\beta^*(1-\rho) + a_\epsilon \tag{11.30}$$

$$a_\delta = 2v_f \sin\beta\left(\sin\varphi - \sin(\varphi - \gamma(K_p^h\delta))\right) \tag{11.31}$$

and $\rho = \cos\beta\sec\beta^*$ is viewed as an external signal.

It can be shown from the definition of ρ and β that, if ϵ is restricted to a bounded region $|\epsilon| \le \|\mathbf{x}\| < \frac{d}{2} - \mu, \mu > 0$ and appropriate values of K_p^s and K_i^s are chosen, there exist positive constants q_{11}, q_{22} and q_{33} such that $q_{11} \le -a_\epsilon$, $q_{22} \le -a_{\tilde{v}}$ and $q_{33}|\delta| \le -a_\delta\delta$. This, in turn, yields

$$\dot{V} \le -\mathbf{x}^T Q \mathbf{x} < 0 \tag{11.32}$$

with $Q = \mathrm{diag}(q_{11}, q_{22}, q_{33})$ positive definite, guaranteeing local exponential stability.

For the second step, we are now interested in the full dynamics and therefore introduce

$$\tilde{f}(\mathbf{x}) = f(\mathbf{x}) - f^*(\mathbf{x}) = \begin{bmatrix} \cos\beta(v_f(\cos\varphi - 1) - v(\cos(\varphi - \gamma(K_p^h\delta)) - 1)) \\ 0 \\ 2\sin\beta(v_f - v)\sin(\varphi - \gamma(K_p^h\delta)) \end{bmatrix} \tag{11.33}$$

In this case, $\dot{\mathbf{x}} = f^*(\mathbf{x}) + \tilde{f}(\mathbf{x})$. Using the same Lyapunov function V, we obtain

$$\dot{V} \le -\mathbf{x}^T Q \mathbf{x} + 2\mathbf{x}^T P \tilde{f}(\mathbf{x}) \tag{11.34}$$

From the fact that for every small $\eta > 0$ there exists $r > 0$ such that

$$||\tilde{f}(\mathbf{x})|| < \eta||\mathbf{x}||, \quad \forall_{||\mathbf{x}||<r} \tag{11.35}$$

it follows that

$$\dot{V} \le -\left(\lambda_{min}(Q) - 2\eta\lambda_{max}(P)\right)||\mathbf{x}||^2 \tag{11.36}$$

Choosing a small enough η guarantees that $\dot{V}$ is negative definite in a neighbourhood of the origin. ■

Remark 2. For the nominal system (11.25), we can obtain explicit convergence intervals for K_p^s and K_i^s by imposing appropriate restrictions on ϵ. As an example, $\epsilon < \frac{d}{3}$ yields $0.159 < K_p^s < 6.307$ and $0 < K_i^s < 3.016 - 0.433K_p^s - \frac{1.941}{0.5+K_p^s}$.

11.1.5 Formation heading estimation

We have so far assumed that the vehicle has access to an estimate of the formation heading. With no explicit communication, the exact heading is not immediately discoverable; we can, however, estimate it from the evolution of the observed distances. Either of the following estimates for ψ_f,

$$\hat{\psi}_f^2 = \arcsin\left(\frac{\xi}{v}\sin(\theta_2) - \frac{\hat{\dot{z}}_2}{v}\right) - \theta_2 + \psi \tag{11.37}$$

$$\hat{\psi}_f^3 = \arcsin\left(-\frac{\xi}{v}\sin(\theta_3) + \frac{\hat{\dot{z}}_3}{v}\right) + \theta_3 + \psi \tag{11.38}$$

or their circular mean can be used. The exact values of $\dot{z}_i$ are not known, but can be estimated from the distance observations z_i using a Kalman filter.

11.2 Acoustics-based formation control for AUVs

The theoretical derivation presented above is based on a set of idealised assumptions, of which the most important is access to noiseless and continuous range signals.

In reality, distances between AUVs are often determined using acoustic equipment that registers the time of flight of an echo request or synchronised ping. Owing to the throughput limitations of acoustic communication and potential collisions on the shared medium, ranges can only be obtained at a low rate and are prone to errors, creating both immediate and indirect complications.

As a result, the controllers developed under theoretical assumptions must be adapted to work in a real-world setting. Our customisation is centred on the MEDUSA AUV developed at DSOR and therefore takes the capabilities and constraints of this vehicle into account; nevertheless, the result should be applicable to comparable vehicles with only minor adjustments.

We operate the MEDUSAs as surface vehicles but with all the practical limitations of an AUV. The solution we propose is suitable for constant-depth underwater operations and, with minor changes, for variable-depth operations using vehicles equipped with a depth sensor and independent depth control.

An overview of the resulting implementation is presented in Figure 11.2. Note that the expres-

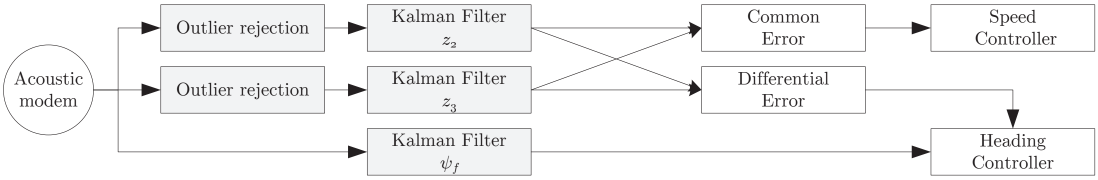


Figure 11.2 – Structure and data flow in the MEDUSA implementation of the formation controller. Modules in grey perform the conversion of available discrete data to the continuous signals required by the algorithm. The remaining modules are the same as in the preceding discussion.

sions for the speed and heading controllers in our implemented solution remain the same as previously presented: all changes take place in the earlier (leftmost) stages. We also assume the use of the standard low-level MEDUSA controllers to transform speed and heading requests into motor commands.

11.2.1 Discrete noisy range handling

Ranging in the MEDUSA vehicles is performed using a multifunction Tritech acoustic data modem. The low transmission speed of the modem makes it so that we can only issue one echo request every couple of seconds. Since transmissions cannot overlap on the common channel, time multiplexing must be used to obtain the ranges to each of the leader vehicles. We choose to explicitly query each one separately, although other solutions are possible, such as emitting a broadcast ping with vehicle-dependent delayed replies.

Each leader is queried every 2 s, so a complete range information update occurs every 4 s—longer in the event of packet loss. This is in stark contrast with our previous assumption of continuous measurement. To avoid fundamental changes to the controllers, we implement two hybrid Kalman filters that take the discrete samples and output a continuous estimate of the distances. The filters realise a simple model

$$\dot{\zeta}_1 = \zeta_2 \tag{11.39}$$

$$\dot{\zeta}_2 = w \tag{11.40}$$

$$z_k = \zeta_{1k} + n_k \tag{11.41}$$

where w and n_k respectively denote the process and measurement noise. While a more accurate internal model of the inter-vehicle range evolution could yield better results, it would require additional information not available to the vehicle.

The ranges received are never current, arriving with a latency of approximately 0.5 s, imposed by transmission times and I/O scheduling on both sender and receiver. We decided not to implement any mitigation techniques (e.g. back-dating the filter updates), instead prioritising simplicity.

In addition to providing updated predictions for the ranges, the Kalman filters help cope with the noise associated with ranging. Although the noise has not been fully characterised, data from non-systematic experiments show quasi-Gaussian noise with standard error of 0.22 m, in line with the specification.

11.2.2 Outlier rejection

Acoustic ranging is prone to outliers: floor geometry, clines, and non-uniform propagation in the water all may lead to an echo reply being received through a path other than the shortest, resulting in overestimation of the distance.

Seeing as an upper bound for the relative speed of the vehicles v_{sup} can be established, any range that differs from the previous by more than the maximum displacement is potentially an outlier. This presupposes, of course, that the ranges are properly initialised.

We implement a simple outlier filter for the ranges received from each leader, based on a sliding window. A measurement m is accepted if inside the interval

$$\left[\bar{z} - v_{sup}(t - t_l), \bar{z} + v_{sup}(t - t_l)\right] \tag{11.42}$$

where $\bar{z}$ is a moving average of previous headings and t_l is the time of the last accepted measurement. When a measurement is accepted, $\bar{z}$ and t_l are updated according to $t_l = t$ and $\bar{z}[k+1] = (1-k)\bar{z}[k] + k_a m$, where $k_a = 0.5$. When a measurement is not accepted but is inside the interval

$$\left[\bar{z} - 4v_{sup}(t - t_l), \bar{z} + 4v_{sup}(t - t_l)\right] \tag{11.43}$$

then $\bar{z}$ is updated as $\bar{z}[k+1] = (1 - k_r)\bar{z}[k] + k_r m$, where $k_r = 0.25$ but t_l remains unchanged, relaxing the limits for subsequent tests and stopping $\bar{z}$ from becoming locked to an old or invalid range. A measurement outside these ranges is discarded.

11.2.3 Heading estimation

The originally proposed approach was designed for continuous measurements, and performs aptly under that assumption. However, it depends on the first derivative of the measured ranges, and the low rate of the acoustic ranging makes an estimate of this derivative very unreliable. For this reason, the theoretical approach is not practical.

The need for formation heading estimation is, however, less than certain. Heading is hard to estimate from ranges alone and, unlike velocity, is widely available in surface and underwater vehicles through the use of a low-cost magnetic compass, a basic navigational aid. As the vehicles use full-featured acoustic modems to measure ranges, it is possible to piggyback a short amount of data on the ranging reply. Transmitting the current heading in the echo reply (compressed to 1 B) thus becomes an obvious solution.

In order to estimate the formation heading from the discrete observations, we feed another hybrid Kalman filter with the piggybacked value whenever a new range is received. The filter shares the same model as the range filters, and its implementation differs solely in that it properly handles circular quantities.

In our scenario, the heading of the two leaders should be close and converge to the formation heading, and so we treat all received headings as samples of the same variable. Thus, while each range estimator is, in the absence of losses, updated every 4 s, the common heading estimator is updated every 2 s, yielding a more responsive estimate. In cases where this assumption does not hold, it is preferable that the follower maintain separate filters, or that leaders send the reference formation heading instead.

12 Evaluation

We evaluate the controller using a multi-tiered approach that culminates in real-world tests with the MEDUSA vehicles. Not only does this allow us to validate the correctness and behaviour of the solution prior to deployment in the real vehicles, but it also makes it possible to discriminate the impact of the additional constraints and coping mechanisms on its performance.

In the following sections, we present the results obtained for each evaluation modality and discuss their significance. The first section deals with a simplified simulation with basic kinematics, the second with a full dynamics simulation, and the last with real-world trials.

12.1 Simplified simulation

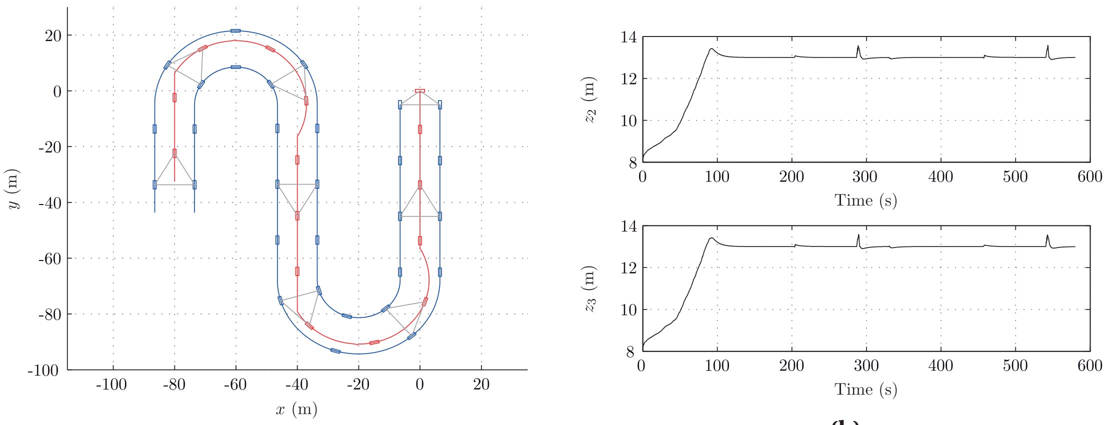
The theoretical derivation of the controller builds upon a unicycle kinematic model for the vehicle, easier to work with than the full dynamics of a marine vehicle. For this reason, and to establish a performance baseline, we begin the evaluation by using the same model.

The simulation work is performed using the MATLAB/Simulink suite we have previously described, and runs in continuous time. We implement the controller as described in Section 11.1, assuming continuous and instant access to range measurements and no explicit communication between the vehicles. The leader heading is not provided and is therefore estimated from range measurements according to (11.37) and (11.38).

We test the controller by having the leader vehicles describe a predefined lawn mowing (scanning) manoeuvre over a 600 s time span, at an average speed $v_f = 0.5\,\mathrm{m\,s^{-1}}$. As specified, the centre path extends for approximately 300 m, with turns of radius $r = 20\,\mathrm{m}$. The vehicles start in the upper right corner, and not in the final formation.

We define an equilateral triangular formation with $d = 13\,\mathrm{m}$. The controller is configured with gains $K_p^s = 1$, $K_i^s = 0.1$, $K_p^h = 1$.

Figure 12.1a shows the trajectories described by the simulated vehicles, with the leaders drawn



(a) **(b)**

Figure 12.1 – Results of a simple simulation run for a lawn mowing manoeuvre. (a) Trajectories described by the three vehicles. The starting position is in the upper right corner, and the follower is in the middle, in red. The position of the vehicles is plotted every 40 s. (b) Distances to leader vehicles.

in blue and the follower in red. Although the trajectory of the follower vehicle may appear unusual when in the turns, it is, in fact, the trajectory that maintains the desired equilateral triangle formation.

The distance plots in Figure 12.1b show this quite clearly. The follower starts out too close to the leaders and moves at a slower pace during part of the initial leg. From the moment it enters the intended formation with $z_2 = z_3 = 13\,\text{m}$, around $t = 100\,\text{s}$, it remains there through the rest of the experiment, with only minute deviations at the moment the vehicles enter or leave a turn, before it is able to adapt.

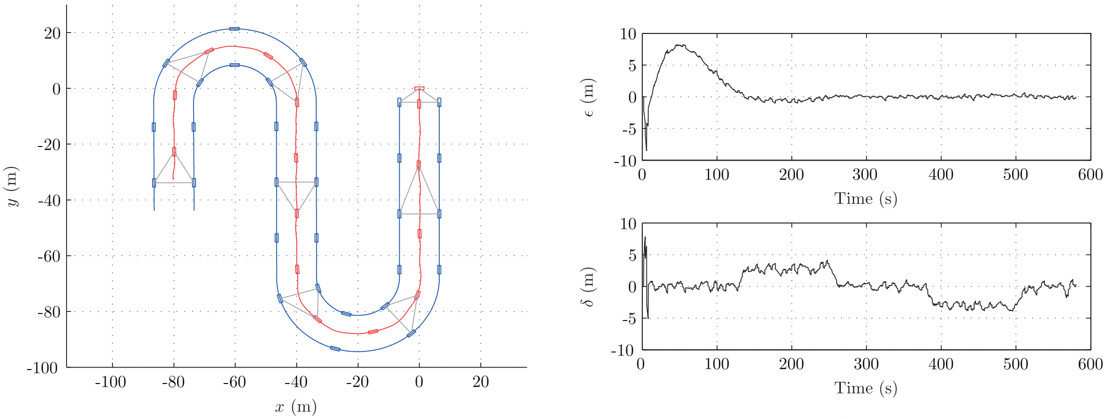
Given the theoretical work underlying our solution, this quasi-optimal behaviour is to be expected.

12.2 Full simulation

Our first simulation is based on a set of idealised assumptions that are not met in reality. In this section, we move towards real-world testing by introducing models and constraints matching those of the real MEDUSA vehicles.

Instead of the unicycle kinematics used in the previous section, the follower vehicle now obeys the full dynamical model of the MEDUSA, described in Section 7.1. The behaviour of the simulated agent should closely match that of the MEDUSA when given the same speed and heading commands.

The range sensing is also reworked. Instead of continuous range measurements, we implement discrete ranging mimicking the capabilities of the acoustic modem on the real vehicles. Range



(a)

(b)

Figure 12.2 – Results of a full simulation run for a lawn mowing manoeuvre. (a) Trajectories described by the three vehicles. The starting position is in the upper right corner, and the follower is in the middle, in red. The position of the vehicles is plotted every 40 s. (b) Common- and differential-mode errors.

measurements are extracted directly from the position difference, to which uncorrelated zero-mean Gaussian noise ($\sigma = 0.2\,\mathrm{m}$) is added. Ranges to each vehicle are obtained alternately with a 4 s period, meaning there is a new sample from one of them every 2 s.

The controller implemented here is as described in Section 11.2. Specifically, it adds hybrid Kalman filters to estimate the ranges and leader heading and implements outlier rejection. Leader headings are piggybacked on each range measurement and are therefore alternately available every 2 s. Our simulation does not consider packet loss: despite potentially serious effects, past experiments hinted that it was relatively infrequent.

For the sequence presented here, the leader vehicles are configured to navigate the same lawn mowing path at the same speed $v_f = 0.5\,\mathrm{m\,s^{-1}}$, well within the capabilities of the MEDUSA. The three vehicles are expected to form an equilateral triangle of side $d = 13\,\mathrm{m}$. The new set of gains, $K_p^s = 0.2$, $K_i^s = 0.005$, and $K_p^h = 0.1$, reflects the slower dynamics of the system.

The resulting trajectories are presented in Figure 12.2a. In contrast to the previous experiment, it is worth noting that the vehicle now remains centred throughout the run, even in the turns. This is a consequence of the tuned gain structure, which strikes a balance between maintaining the desired ranges and mimicking the received leader headings. We believe this approach is preferable as it results in a more natural trajectory and eliminates sharp course changes when entering or exiting a turn, which would, in any case, be cumbrous for the real vehicle to execute.

The impact of this change can be seen in Figure 12.2b. The differential-mode error δ is now significant when turning, as the ranges deviate from the 13 m goal. Still, the absolute error does not exceed 4 m during turns and is generally below 1 m during straight-line motion. Following the initial convergence period, the common-mode error ϵ remains stable around zero.

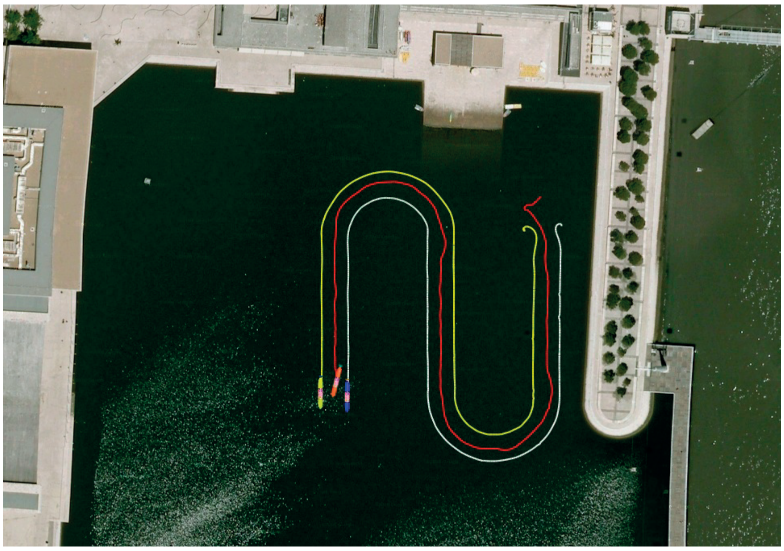


Figure 12.3 – Aerial view of the mission area and trajectories of the MEDUSA vehicles. The manoeuvre starts on the top right corner; the trajectory of the controlled vehicle is shown in red.

While not described here, the same implementation was used for the hardware-in-the-loop (HIL) simulation, to ensure that it would work correctly on the vehicle hardware.

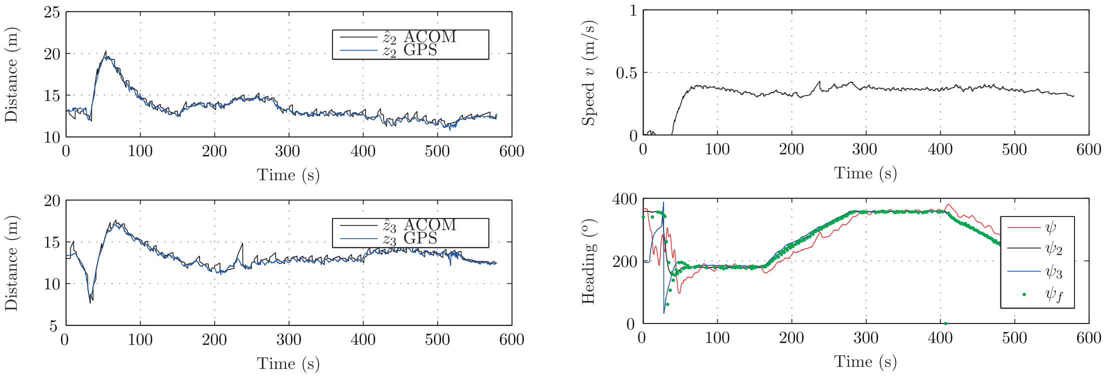
12.3 MEDUSA real-world testing

Real world trials took place in the saltwater lagoon described in Section 6.2. The vehicles were equipped with the full sensor suite, including RTK GPS, but the usual constraints of an AUV are observed. In particular, and while the follower remains at the surface, it only logs GPS information for ground truth and does not use it for navigation; distances to the leading vehicles are measured using acoustic signals. The leader vehicles are running the coordinated path following algorithm described in [29].

A series of increasingly complex validation and evaluation tests culminated in the three-legged lawn mowing manoeuvre previously simulated, navigating at an approximate speed $v_f = 0.4\,\mathrm{m\,s^{-1}}$. All controller settings and parameters are the same as in the full simulation.

Figure 12.3 shows a top view of the trajectories described by the three vehicles, starting in the upper right corner. A transient can be noticed at the beginning: none of the vehicles start in their designated position or heading and therefore need to adjust. The movement of the leaders and their rapidly varying reported headings impact the Kalman filter estimate, causing the follower to start by moving in a suboptimal direction.

The follower remains on track during the rest of the manoeuvre, even though its trajectory is not as smooth. A leading factor causing the vehicle to stray off-path is packet loss in the acoustic links; while not modelled in simulation, it turns out to be an important phenomenon



(a) **(b)**

Figure 12.4 – (a) Time evolution of the distances z_i measured using RTK GPS and estimates $\hat{z}_i$ obtained from the acoustic ranges using a Kalman filter. (b) Time evolution of the follower vehicle speed v and actual heading ψ, as well as leader headings ψ_2, ψ_3 and references received acoustically, ψ_f. The vehicle speed is estimated from GPS measurements with some associated noise.

in shallow water conditions such as those of the test site.

Figure 12.4a compares the real distances measured using GPS data with the estimates produced by the Kalman filters from the discrete acoustic range measurements. For the most part, the estimated ranges closely follow the real distances. However, there are times when ranges are missed and the controller makes decisions based on outdated information, as is the case for z_3 around $t = 230\,\mathrm{s}$.

The heading data in Figure 12.4b confirm these observations. While ψ_2, ψ_3, and, therefore, ψ_f samples overlap during the majority of the test, there are significant differences in the first forty seconds, before the leader vehicles converge to their predetermined path. Periods of missing samples are visible throughout the experiment.

The time delay between the leader and follower heading is not a shortcoming but intended behaviour. While there is a delay imposed by the Kalman filter and the overall response time of the system, the controlled vehicle is in fact trailing the leaders by several seconds; in order to mimic their trajectory, changes in heading must likewise be delayed.

13 Alternative formations

The work we presented in the previous chapters solves the particular problem of having a follower vehicle navigate in a triangular formation, behind the two leaders, using range information. The MORPH project, however, calls for a formation with two surface vehicles travelling in line and underwater vehicles following on the sides.

In order to adapt to the MORPH scenario, the problem is recast, abandoning the common- and differential-mode error formulation. Instead, an along track–cross track reference frame, ϵ–δ, is defined, with origin at p_d, the along-track axis ϵ pointing in a direction opposite the heading of the leaders (that is, $\psi_f + 180°$), and the δ axis pointing 90° anticlockwise.

Written in this frame, the kinematics of the vehicle become

$$\dot{p}_\epsilon = v_l - v\cos\left(\psi - \psi_f\right) \tag{13.1}$$

$$\dot{p}_\delta = -v\sin\left(\psi - \psi_f\right) \tag{13.2}$$

where p_ϵ and p_δ are the ϵ–δ coordinates of p. If the errors p_ϵ and p_δ go to zero, then p converges to the desired position p_d, subject to the same symmetry limitation discussed in Section 11.1.

A control law for the desired speed can be given by

$$v = \operatorname{sat}\left(K_p^s p_\epsilon + K_i^s \int_0^t p_\epsilon d\tau + v_{nom}\right) \tag{13.3}$$

where K_p^s is the proportional gain, K_i^s is the integral gain, and v_{nom} is an optional nominal velocity that, if close to the leader speed, helps the convergence. The result is saturated to a range $[v_{\min}, v_{\max}]$.

A control law for ψ is given by

$$\psi = \hat{\psi}_f + \operatorname{sat}\left(K_p^h p_\delta + K_i^h \int_0^t p_\delta d\tau\right) \tag{13.4}$$

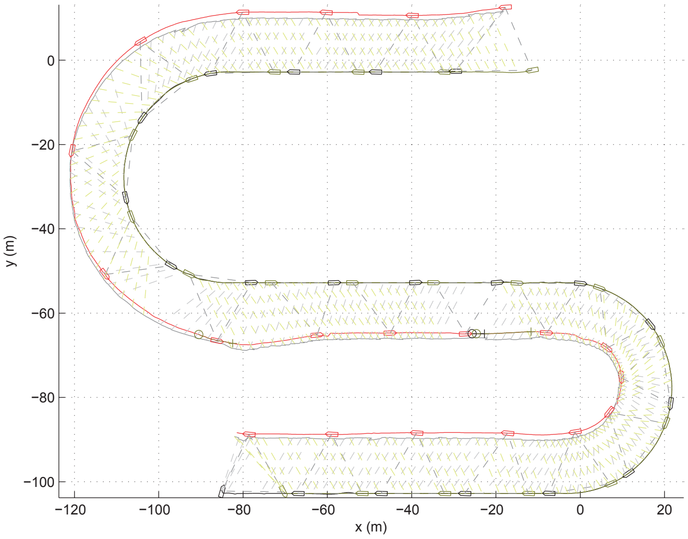


Figure 13.1 – Vehicle trajectories during trial, with the follower vehicle moving on the right side of the two in-line leaders. The follower is plotted in red, and the two leaders are plotted in black and yellow [22].

where K_p^h is the proportional gain and K_i^h is the integral gain. The heading controller tracks the reference heading estimated from the information sent by the leaders, $\hat{\psi}_f$, and adds a proportional-integral (PI) controller on the error p_δ. The output of the PI controller is saturated to a range such that, even for large errors, the vehicle does not move in a direction opposite that of the leaders.

This alternative problem formulation and controllers have been successfully tested using the MEDUSA vehicles. Figure 13.1 shows a trial run with two leader vehicles in line and a follower on the side. We refer the reader to [22] for a complete description and analysis.

14 Summary

In this part of the thesis, we studied the problem of how to maintain a follower vehicle in a triangular formation with two leaders using range measurements. This is a particularly pertinent problem in the marine domain, where underwater vehicles often lack other localisation methods and must navigate based on acoustic ranging.

We explained the theoretical derivation of separate speed and heading controllers, respectively using the common-mode and differential-mode range errors as input. Based on a simplified kinematic model and a set of assumptions on information availability, we outlined proofs of stability for these controllers.

Afterwards, we tackled the implementation of our solution in real MEDUSA vehicles. Constraints associated with the slow and unreliable acoustic ranging process forced us to introduce range estimation and outlier rejection mechanisms. As it is infeasible to reliably predict the formation heading from the infrequent range measurements, the leaders include their compass heading on the data field of the ranging replies.

The solution was first evaluated in a simplified simulation, using the same assumptions that underlie the theoretical derivation. We then tested it in a high-fidelity simulation using the MEDUSA dynamical model and completed the evaluation in real-world trials with three surface vehicles.

Finally, we explored how the same architecture can be used for a sideways triangular formation by adapting the controller expressions.

Part IV

Formation-based airborne plume tracing

A much smaller and warmer hand had enclosed his and was pulling him upward. He obeyed its pressure without really thinking about it. Only as he walked blindly back through the crowd did he realise, from a trace of flowery scent on the air, that it was Ginny who was leading him back into the castle.

— J. K. Rowling, *Harry Potter and the Half-Blood Prince*

15 Introduction

The difficulties surrounding robotic plume tracing have led to the development of a multitude of approaches, ranging from simple bio-inspired algorithms to varyingly complex probabilistic inference methods. Researchers have also experimented with different combinations of sensors, different numbers of robots, and different cooperation modalities.

Our work uses a group of robots in formation to trace a plume to its source in stable wind conditions, using odour and wind sensors. The use of multiple robots provides us with spatial diversity in odour sampling, yielding more information than a single robot could. Having robots distributed along the crosswind direction makes it possible, over time, to estimate the relative position of the formation in the plume, while upwind diversity provides us with additional data points that may be averaged, helping to overcome the patchiness of the odour distribution.

The robots exchange odour measurements that are used to drive the fleet to and along the centre of the plume, and sense the wind direction in order to move upwind in the plume. Relative position measurements enable a graph-based feedback controller to maintain the robots in an arbitrary leaderless formation. The controller is based on the principle of Laplacian feedback, which has been used extensively in other contexts [35], [47]. The formation can be dynamically scaled to better fit the plume or change more drastically in response to the situation.

Our solution is computationally light, easy to implement, and requires no sensor information other than wind, odour, and relative positions. It is validated and evaluated in a submicroscopic simulator that mimics our existing real-world infrastructure, including the Khepera III differential wheeled robots and add-on odour and wind sensing boards [23]. The simulation arena is a model of the DISAL wind tunnel with laminar wind flow and a filament-based odour propagation model. We also present real-world experiments in a wind tunnel, using Khepera III and Khepera IV robots and a fully distributed and self-contained implementation [24].

16 Related work

A survey of past research in odour source localisation and mapping can be found in [48], where a comprehensive taxonomy is defined and a multitude of solutions are discussed.

Some of the simplest algorithms are inspired by the strategies used by biological agents—such as bacteria [49] or silkworm moths [50]—and usually consist of a set of basic states with associated behaviours. These include, for instance, moving upwind when inside the plume and moving in an expanding spiral when the plume is lost. Braitenberg approaches have also been tested successfully [51]. Experiments with multiple robots [52], [53] have yielded mixed results, with authors reporting no clear improvement in the plume traversal phase, particularly in the absence of explicit collaboration [16].

Formation- and swarm-based algorithms [54], [55] are designed for cooperative multi-robot scenarios and work by coordinating the movement of several agents distributed over the area of interest. The information gained from simultaneous sampling at multiple locations and the close cooperation between the robots can greatly increase efficiency while retaining computational simplicity. Our work was originally inspired by the crosswind formation algorithm presented in [56].

Of the more complex approaches, probabilistic inference methods have been used extensively [16], [57]–[59]. Taking multiple odour samples obtained over time and space and a model of plume propagation, agents are able to generate a probability distribution for the source location. This estimate can then be used to optimise movement steps or trajectories with the goal of improving the belief.

Meta-heuristic optimisation methods applied to the problem include PSO [60]–[62], ant colony optimisation (ACO) [63], and others [64].

More recently, compact commercial tunable diode laser absorption spectroscopy (TDLAS) sensors eliminated the constraint of *in situ* odour measurements. Despite measuring the accumulated concentration over the beam path and being limited to line-of-sight readings, laser sensors open the door for novel remote sensing approaches [65], [66].

17 Technical approach

We start by establishing the three reference frames that are used throughout this part:

- the local frame, x–y, with origin at the centre of the robot and in which the x axis points towards the front of the robot and the y axis points 90° anticlockwise
- the wind frame, uw–cw, with origin at the centre of the specified formation and in which the uw axis points in the perceived upwind direction and the cw axis points 90° anticlockwise
- the global frame, x'–y', with origin at the upwind end of the arena and in which the x' axis points downwind and the y' axis points 90° anticlockwise

These frames are represented in Figure 17.1 in red, blue, and black, respectively. The global frame is not used in the algorithm, which relies entirely on local information, but solely for ground truth trajectory plotting.

We organise our algorithm in three parallel components: upwind movement, formation control and plume centring . Each fulfils a particular role, but all three are required for a functional solution. The components each yield a $(\dot{x}, \dot{y})$ desired velocity vector; the three vectors are combined and transformed into control signals.

Wind plays a central role in the problem and in our solution: as the odour tends to travel downwind, the direction of the wind provides a strong indication as to the relative position of the source. The upwind movement component therefore takes this indication and drives the robots upwind. Moreover, we do not assume access to a magnetic compass and so the perceived direction of the wind becomes the only common reference for alignment.

By itself, upwind movement will only lead a robot to the source if it starts centred in the plume and is able to obtain perfect error-free wind measurements, which are not realistic assumptions. The plume centring component takes the odour concentrations measured by the different robots in the formation and uses them to move crosswind towards the plume centre.

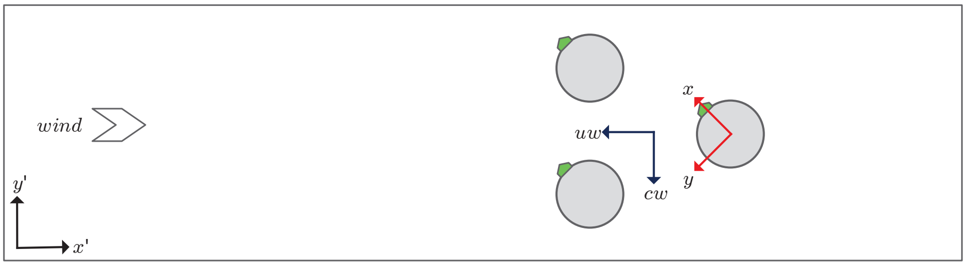


Figure 17.1 – Schematic view of three robots in the wind tunnel (not to scale), displaying the local, wind and global reference frames used in this work.

In order to obtain the odour measurements that allow us to track the plume, we must be able to position the robots appropriately. Graph-based formation control is the core component that makes the algorithm possible. The shape of the formation is specified by the desired position of each robot relative to the centre in a reference frame that rotates with the wind, so that the formation implicitly orients to the wind direction. The component takes the specified and actual positions of the neighbours and computes a direction of movement to drive the robot to its post. While the formation shape is specified *a priori*, it can adapt over time to better envelop the plume.

Control is distributed and asynchronous, with no external support or intervention. The robots do not have access to their absolute position and must therefore rely on relative positions. Because the distances are bound and relatively short, we assume that the robots are always in communication range. Nevertheless, the algorithm is designed to minimise information exchange and be robust to isolated instances of packet loss.

In the remainder of this part, the terms front and back are respectively used to refer to the robots upwind and downwind in the specified formation; the definitions of left-of-centre and right-of-centre follow naturally. We do not consider role assignment in this work, and so the position of each robot in the formation is fixed and defined by its identifier.

The next section describes the development of the algorithm, resorting to theory and simulation. We then address the challenges of implementing the solution in real robots.

17.1 Graph-based formation plume tracing

The design of the algorithm is driven by a quest for simplicity, taking into account the constraints of small and inexpensive robotic platforms. Accordingly, and while we strive to keep the algorithm general, some aspects are influenced by the limitations of the Khepera robots and of the wind and odour sensing hardware.

The three individual behaviours are described in the upcoming sections. As the control laws are locally formulated and homogeneous, the notation referencing the node itself is omitted except where required for comprehension.

17.1.1 Sensing

The algorithm uses three types of ephemeral information: relative positions to the neighbours, for formation control; wind direction, for orientation; and odour concentration, for centring and dynamical spacing. For the development and simulation work, we use the previously presented virtual sensors.

The range and bearing measurements received from the board are converted to the local Cartesian frame and stored along with a time stamp. Relative positioning using infrared sensors is fast when compared with the acoustic ranging in the previous part, and can generate multiple samples per second. For this reason, we opt to forgo the added complexity of target tracking and use the (unfiltered) most recent measurements.

Wind measurements are affected by significant process and measurement noise, and so we incorporate a discrete Kalman filter to estimate the real wind direction. The observations are complemented by odometry information to compensate for the robot rotation. The Kalman filter follows the model

$$\theta_k = \theta_{k-1} + \begin{bmatrix} \frac{1}{l} & \frac{-1}{l} \end{bmatrix} \begin{bmatrix} \delta^r_k \\ \delta^l_k \end{bmatrix} + w_k \tag{17.1}$$

$$z_k = \theta_k + n_k \tag{17.2}$$

where θ_k is the wind direction estimate at time k, z_k the observation, and w_k and n_k the process and measurement noise, respectively. The inputs δ^r_k and δ^l_k are the distances travelled by each wheel since the last prediction step and l is the robot axle length. The angles are normalised and discontinuities are dealt with.

Odour measurements obtained by the sensor are fed through a sliding window filter to compute the maximum of the N most recent observations. The estimate produced by the filter is broadcast to the other robots along with the source identifier. All robots store the last received concentrations c_j from all flock mates, in turn used to update three variables:

- c_c, the mean concentration measured by robots in the centre
- c_l, the mean concentration measured by robots to the left of the formation centre
- c_r, the mean concentration measured by robots to the right of the formation centre

Owing to the way information is exchanged and the assumption on communication ranges, all robots make decisions based on approximately the same data. While not true at all times (the controllers are not synchronised and packets may occasionally be lost),the update rate is such that temporal alignment discrepancies do not require specific handling.

17.1.2 Laplacian feedback

Our formation can be expressed as an undirected graph $G = (V, E)$, in which vertices $V = \{v_1, v_2, \dots, v_n\}$ correspond to robots and edges $E = \{e_1, e_2, \dots, e_n\}$ correspond to inter-robot relative positioning links, or a subset thereof. From here, we can use the work in [47] and standard results in graph theory to attain a provably stable solution to the formation control problem:

$$\dot{\mathbf{x}} = -\left(\mathscr{L} \otimes I_2\right)(\mathbf{x} - \mathbf{b}) \tag{17.3}$$

where $\mathbf{x}$ is the (x, y) absolute position vector for all robots, $\mathbf{b}$ is the bias vector containing the desired offsets to the formation centroid, and $\mathscr{L} = \mathscr{B}\mathscr{B}^T$ is the positive-semidefinite Laplacian matrix, obtained from the incidence matrix $\mathscr{B}$ that describes the edges of G. The Laplacian or admittance matrix is defined as the difference between the degree and adjacency matrices ($\mathscr{L} = \mathscr{D} - \mathscr{A}$), and its elements are given by

$$\mathscr{L}_{i,j} = \begin{cases} \deg(v_i) & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise} \end{cases} \tag{17.4}$$

The law (17.3) is only applicable under the assumptions of holonomic agents, access to absolute positions, and a connected graph (i.e. one in which there exists a path between every pair of vertices). The same approach can nonetheless be implemented in a more practical decentralised fashion, using only relative positions and accounting for nonholonomicity either natively or by offloading to a lower level controller [35]. The resulting velocity vector for formation control is

$$\mathbf{u_f} = -\begin{bmatrix} \sum_{j=0}^{N} \mathscr{L}_j \left(x_j - \beta_j^x\right) \\ \sum_{j=0}^{N} \mathscr{L}_j \left(y_j - \beta_j^y\right) \end{bmatrix} \tag{17.5}$$

where $\mathscr{L}_j = \mathscr{L}_{i,j}$ is the entry of the Laplacian matrix relating the controlled node i to neighbour j, and x_j and y_j are the relative positions to robot j in the body frame. As we want the formation to be oriented with respect to the wind, the bias vectors are specified in the uw–cw frame and must be rotated by the estimated wind angle θ; β_j is a local analogue to $\mathbf{b}$, describing the desired relative position between the two robots in the robot frame, i.e. $\beta_j = R(\theta_i)[\bar{\mathbf{p}}_\mathbf{j} - \bar{\mathbf{p}}_\mathbf{i}]$, $\bar{\mathbf{p}}_\mathbf{i}$ and $\bar{\mathbf{p}}_\mathbf{j}$ expressed in the wind frame.

Depending on the size and growth rate of the odour plume, a fixed formation might not be an optimal or even effective choice. Therefore, we implement a method to change the formation spacing by varying the bias vector as a function of the measurements. For simplicity, we define two scalar parameters, s_{uw} and s_{cw}, which respectively represent adaptive bias coefficients in the upwind and crosswind direction. Assuming that biases (and hence the formation) are symmetric, this results in a minor modification. Equation (17.5) remains valid, but β_j assumes a new formulation given by

$$\beta_j = R(\theta) \begin{bmatrix} s_{uw} & 0 \\ 0 & s_{cw} \end{bmatrix} [\bar{\mathbf{p}}_\mathbf{j} - \bar{\mathbf{p}}_\mathbf{i}] \tag{17.6}$$

We choose to use a constant upwind scaling factor and continuously vary the crosswind scaling according to

$$s_{cw} = k_{cw} \frac{c_l + c_r}{1 + c_c} \tag{17.7}$$

The underlying rationale is that, at our evaluation scale, the differences in plume structure are more pronounced in the crosswind direction. The scaling methods may be easily modified to adapt to different realities.

The exact distribution of robots in the plume depends on its aspect ratio, influenced by wind speed and other factors mediating odour dispersal; proper scaling limits need to be set to account for the minimum safe distance and the maximum movement range. We loosely tune k_{cw} to maintain the side robots on the detectable edges of the test plumes. For plumes with very different characteristics, the robots may not move along the edges but will still trace the plume shape and converge to the minimum safe distance as they approach a point source.

17.1.3 Upwind movement

Tracing an odour plume is hard due to the intermittency of the chemical cue. However, in stable enough wind conditions, the wind flow provides a continuous cue that is a strong indicator of the direction of the source. Therefore, when in the plume, moving upwind is a good strategy. This behaviour is known to biologists as odour-gated rheotaxis.

We implement rheotaxis by defining a movement urge in the direction of the apparent wind, given by

$$\mathbf{u_w} = R(\theta) \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{17.8}$$

The base behaviour is specified in the uw–cw frame, in which upwind movement corresponds to the vector $[1 \quad 0]^T$, and is transformed to the local frame using $R(\theta)$.

17.1.4 Plume centring

The formation is kept centred in the plume by a crosswind force depending on the difference between c_l and c_r, the aforementioned mean odour readings to the left and to the right of the formation centre. To prevent extreme changes in control outputs, reflecting the high-amplitude variations of the odour concentration, we implement a logistic response given by

$$\mathbf{u_c} = R(\theta)\begin{bmatrix} 0 \\ -u_c^{max} + \frac{2u_c^{max}}{1+e^{-(c_l-c_r)/k_l}} \end{bmatrix} \tag{17.9}$$

This asymptotically limits the maximum requested crosswind velocity to $\pm u_c^{max} = \pm 0.5$. Setting k_l to the same order of magnitude as the dynamic range of the sensor guarantees that these values are only approached for highly asymmetric odour readings.

17.1.5 Behaviour aggregation

We combine the requested velocity vectors from each behaviour by means of the weighted sum

$$\mathbf{u} = k_w\mathbf{u_w} + k_c\mathbf{u_c} + k_f\mathbf{u_f}. \tag{17.10}$$

Taking the particular functions described above and setting constant weights $k_w = k_c = k_f = 1$, the expression resolves to

$$\mathbf{u} = -\begin{bmatrix} \sum\limits_{j=0}^{N} \mathscr{L}_j\left(x_j - \beta_j^x\right) \\ \sum\limits_{j=0}^{N} \mathscr{L}_j\left(y_j - \beta_j^y\right) \end{bmatrix} + R(\theta)\begin{bmatrix} 1 \\ -u_c^{max} + \frac{2u_c^{max}}{1+e^{-(c_l-c_r)/k_l}} \end{bmatrix} \tag{17.11}$$

where β_j is given by (17.6). Proportional controllers take the resulting vector $\mathbf{u} = \begin{bmatrix} u_x & u_y \end{bmatrix}^T$ and compute the requested (dimensionless) linear and angular speeds, limited to forward movement and saturated to a reasonable maximum within the operating range:

$$v = k_v\mathbf{u}_x \qquad 0 \le v \le v_{max} \tag{17.12}$$

$$\omega = k_\omega\mathbf{u}_y \qquad -\omega_{max} \le \omega \le \omega_{max} \tag{17.13}$$

These simple controllers suffice because the velocity vector rotates with the perceived wind direction, in the local frame; as the robot turns based on $\mathbf{u}_y$, it eventually aligns with $\mathbf{u}$. The commands v and ω are transferred to a robot-specific motion controller that computes the actual control signals.

17.2 Application to Khepera III robots

Even when working with submicroscopic simulators and detailed models of the hardware, imperfectly modelled phenomena inevitably result in differences between the simulated and real performance of an algorithm. In the following sections, we address the changes enacted in the process of bringing our solution to the real Khepera III robots.

17.2.1 Relative position handling

Owing to component tolerances, no two relative position boards transmit the exact same power. Therefore, two emitters at the same physical distance are detected as if they were at different ranges. When two neighbouring robots in the formation measure disparate distances to each other, one will tend to chase in an attempt to reach the desired offset, while the other will move away with the same intent.

In order to achieve a stable formation, this behaviour must be precluded, which we accomplish by having robots broadcast their observed ranges. Received ranges are used to compute a pairwise mean range, guaranteeing each two robots see one another at the same distance. This suppresses the chasing behaviour and further increases the robustness of the formation without requiring changes to the controller.

17.2.2 Formation scaling

It became clear during preliminary experiments that calculating the formation scale factor from the latest measurements alone resulted in too quick and wide variations, causing the robots to spend time and energy adjusting their heading in spurious attempts to narrow or widen the formation. To improve efficiency, we replace the original expression for the scaling factor s_{cw} with an integral law

$$\dot{s}_{cw} = k_{cw}((c_l + c_r) - c_c) \tag{17.14}$$

The goal is to dimension the formation width in order for the side robots to detect approximately half the concentration of the centre robots, thereby tracing the plume shape. The bounds for s_{cw} are defined considering the minimum safety distance between two robots and the dimension of the tunnel. In contrast to the previous approach, it is not necessary to carefully optimise the gain k_{cw}, provided its value results in reasonably small adjustments.

17.2.3 Collision avoidance

When using Laplacian feedback for formation control, only robots to which an edge exists in the static graph bear direct influence on the movement; all other robots are, from the perspective of the algorithm, invisible. In reality, this might not be a desirable property: when two unconnected robots cross paths (for instance, when first moving into formation), it may give rise to a collision.

A trivial solution to the invisibility problem is to define a complete graph, one in which all pairs of vertices are connected. Making all robots visible should help prevent crashes but is not without its disadvantages. When the infrared beam between two robots crosses a third one, occlusion effects can cause ranges to be unavailable or overestimated (due to signal attenuation or reflection). If this is the case for the desired formation shape, consistently erroneous range measurements will prevent the robots from ever converging.

Our solution is to hybridise the two approaches. We define a static graph that includes all immediate neighbours and does not suffer from occlusion effects in steady state. To this graph, transient links are added for robots detected at very close range: these present a collision risk and are unlikely to be erroneous. Only range measurements obtained during the three most recent cycles are taken into account, and neighbours for which no recent range is available have their corresponding edges temporarily stricken from the graph.

This approach protects against spurious measurements that could destabilise the formation while simultaneously preventing collisions with robots that, although members of the formation, are not adjacent in the static graph.

We do not address collision avoidance with non-robot objects. A simple anticollision strategy could consist of an overriding Braitenberg controller using the proximity sensors [67], but the problem may be elegantly solved within the graph-based framework by dynamically creating virtual graph vertices with repulsive edges, corresponding to detected obstacles. A solution along these lines is explored in [17].

18 Evaluation

We take the algorithm through an in-depth testing and evaluation process, comprising both simulations and real-world experiments. In this chapter, we present and discuss the outcomes of this process.

The first section deals with simulations using Webots, which provide a baseline for the sensor performance and algorithm behaviour. In addition to testing two formations, we show how the formation adapts to the shape of the plume. The last section concerns robot trials in the wind tunnel, and presents qualitative and quantitative results for different formations and starting positions.

18.1 Webots simulation

The simulation arena, shown in Figure 18.1 and modelled on the wind tunnel, is approximately 20 m long and 4 m wide. We use the previously described odour and wind simulation plug-in and the Khepera III robot model provided with Webots.

The robots are placed approximately 14 m downwind from the source (the exact distance depends on the formation and randomised starting positions), and the source is placed 1 m from the virtual inlet, centred in the test section at floor level. The wind field is constant, with magnitude $1\,\mathrm{m\,s^{-1}}$.

Positions are measured using Webots emitters and receivers, resulting in range and bearing observations. For legacy reasons, we use radio emitters for which there is no line of sight requirement. This limitation notwithstanding, the formation graphs are designed to avoid links that would be subject to infrared occlusion once the robots are in formation.

The simulation step is 32 ms. The number of samples of the odour max filter is set to $N = 50$, corresponding to a sliding window of 1.6 s. In our experiments, $k_l = 200$ and $k_{cw} = 0.2$. No attempt is made to optimise the robot speed so as to minimise mission completion time. Instead, the cruising speed is defined by the low-level translation of a speed command $|u_w| = 1$,

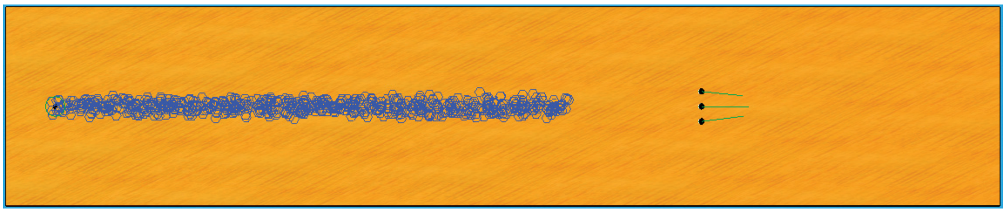


Figure 18.1 – Webots simulation scenario. The odour source is on the left end and the robots start on the right side. The wind blows from left to right. The plume is shown in blue and consists of filaments with growth rate $\gamma = 4 \times 10^{-7}\,\mathrm{m^2\,s^{-1}}$.

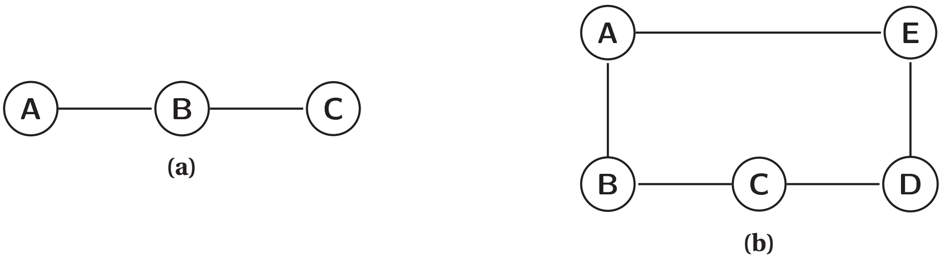


Figure 18.2 – Formations tested in simulation: (a) Three robots in a line; (b) Five robots in a rectangle.

which corresponds to approximately $0.07\,\mathrm{m\,s^{-1}}$.

As we are evaluating a plume tracing algorithm, we assume that at least one robot starts in the plume. Plume declaration also being outside the scope, experiments are interrupted externally when the robots reach the end of the simulation arena.

We perform several experiments, testing multiple formations and different plumes. We discuss two representative arrangements: a three-robot linear formation and a five-robot rectangular formation (see Figure 18.2). The plots shown are the result of individual runs but representative of the average performance.

18.1.1 Linear formation

The first simulation considers a three-robot linear formation, oriented along the crosswind axis, corresponding to predefined biases $\mathbf{b}_{uw} = [0 \quad 0 \quad 0]^T$ and $\mathbf{b}_{cw} = [1 \quad 0 \quad -1]^T$. To better highlight the adaptive formation spacing, this simulation uses a high filament growth rate $\gamma = 10^{-3}\,\mathrm{m^2\,s^{-1}}$. For the same reason, the robots start centred in the plume and in proximity—this is not a requirement, and the next section presents simulations with off-centre starting positions.

Figure 18.3 shows the trajectories described by the three robots over a total time of 180 s, overlapped with a snapshot of the simulation environment at an arbitrary time. Based on the odour measurements, robots begin by widening the formation and then successfully trace the limits of the plume to its source. At this scale, the trajectories appear smooth, with no major

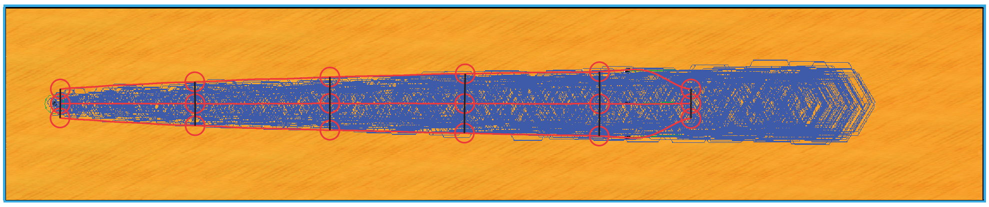


Figure 18.3 – Robot trajectories for a three-robot linear formation in a plume with filament growth rate $\gamma = 10^{-3}\,\mathrm{m^2\,s^{-1}}$. Black lines connect the robot positions at intervals of approximately 35 s.

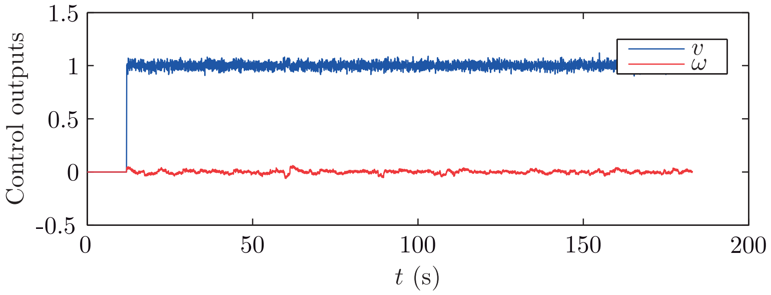


Figure 18.4 – Control outputs v and ω for the centre robot. Both are dimensionless quantities, converted by a low level controller into the actual linear and angular speeds. The robots are parked for the first 11 s, waiting for the first non-zero odour measurement.

disturbances.

In fact, looking at the control signals in Figure 18.4, we can see that the angular speed ω is almost constant and more stable than the linear speed v, which varies approximately 10 % around the base upwind speed $v = 1$.

Figure 18.5 shows the wind direction measurements along the complete trajectory, as well as the Kalman filter estimate of the true wind direction (in the local frame). Given the central role that the wind direction plays in our algorithm, high measurement noise would severely degrade its performance. The Kalman estimate is a considerable improvement over the raw data, and provides a usable reference.

In Figure 18.6, we illustrate our statements about the intermittency of the odour plume with the recorded odour measurements from the robots on the left (A) and in the centre of the plume (B). The centre robot reports higher odour readings, as expected, but both show high-frequency and high-amplitude variations. The fifty-slot sliding window maximum filter yields more relevant readings, but even its output includes substantial noise. The logistic response in centring is nevertheless able to minimise the impact of these variations, as evidenced by the stable ω in Figure 18.4.

Closer analysis shows that, in spite of the noisy wind and odour signals, the dominant source of noise present in the control outputs is formation control, a consequence of errors in the relative positions received from the range and bearing module. Improvements could be realised by

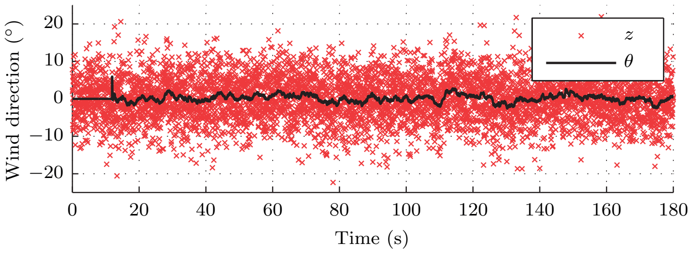


Figure 18.5 – Wind measurements z and Kalman estimate θ of the wind direction, relative to the front of the centre robot.

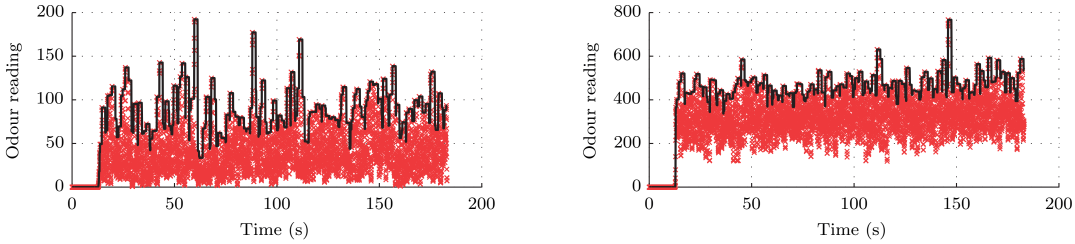


Figure 18.6 – Odour measurements for robots A (left) and B (centre). The red lines show the instant measurements and the black lines the result of sliding window maximum filtering. Note the different scales in each plot.

introducing a Bayesian filter for neighbour tracking, and further still by broadcasting the control outputs and wind direction estimate of each vehicle and using them to better predict future relative positions. However, the fast rate of the sensor allows us to obtain good overall results even without implementing more involved strategies.

18.1.2 Rectangular formation

The rectangular formation, portrayed in Figure 18.2b, is composed of five robots: four on the vertices and one at the centre of the rear edge. This corresponds to predefined biases $\mathbf{b}_{uw} = [0 \quad 1 \quad 0 \quad 1 \quad 0]^T$ and $\mathbf{b}_{cw} = [1 \quad 1 \quad 0 \quad -1 \quad -1]^T$, with each robot connected to the two closest neighbours along the perimeter of the formation. The robots start out in two clusters on the sides of the arena, to showcase the plume centring and formation control capabilities.

In contrast to the linear formation, the rectangular formation introduces spatial diversity in the upwind direction. Robots along the same upwind line are able to average their readings, in order to provide aggregate measurement c_l and c_r, less affected by individual odour packets. Figure 18.7 shows a sixty-second plot of the readings obtained by the two robots on the left edge of the rectangle and of the resulting mean value that is used as input to the controller.

The trajectories followed by the robots are presented in Figure 18.8. The robots converge to the desired formation and to the plume centre in the first 30 s, and continue upwind along a nearly

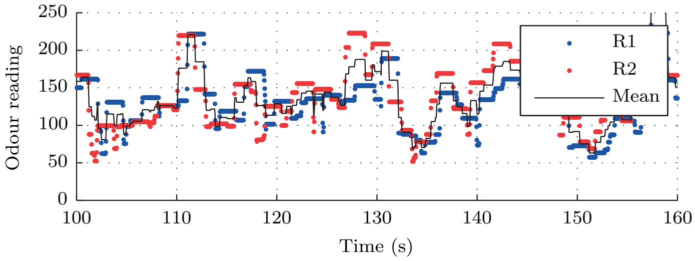


Figure 18.7 – Max-filtered odour measurements for robots A/R1 (left, front) and B/R2 (left, back), and resulting mean c_l.

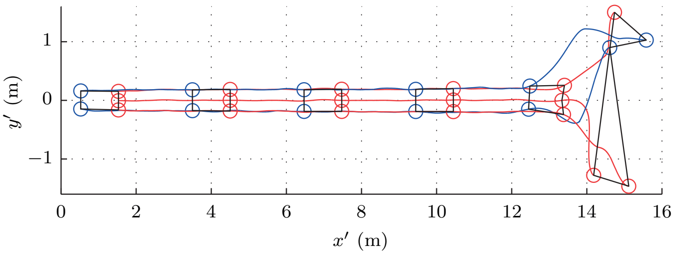


Figure 18.8 – Robot trajectories for a rectangular formation in a plume with growth rate $\gamma = 4 \times 10^{-7}\,\mathrm{m^2\,s^{-1}}$. The trajectories of the front robots are traced in blue and those of the back robots are traced in red. Black lines connect the robot positions at intervals of approximately 35 s.

straight path to the source. This simulation was run with the standard filament growth rate $\gamma = 4 \times 10^{-7}\,\mathrm{m^2\,s^{-1}}$, therefore no change in plume width is observable over this short distance.

18.2 Wind tunnel trials

Real world trials take place in the wind tunnel, using Khepera III robots equipped with the custom boards described in Chapter 5. The odour source is placed approximately 0.8 m downwind from the inlet and slightly off-centre at $x = 1.9$ m, and the wind speed is set to $1\,\mathrm{m\,s^{-1}}$.

Due to differences in sensor characteristics and the previously discussed changes to the algorithm, gains are adjusted to $k_{cw} = 1.5 \times 10^{-6}$ and $k_l = 4000$. Because of the slower response of the sensor, the odour max filter window is reduced to $N = 5$ samples. The relative positioning boards are set to transmit at 5 Hz.

The base experiments use a five-robot rectangular formation with one robot in the centre; subsequent experiments test other formations. The following sections describe the results obtained.

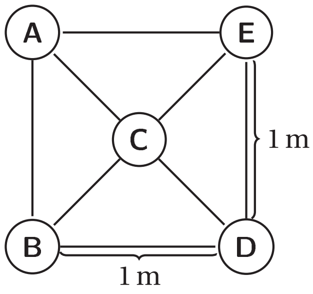


Figure 18.9 – Five-node rectangular formation showing graph edges and base distances. Actual desired distances are dynamically varied by the algorithm.

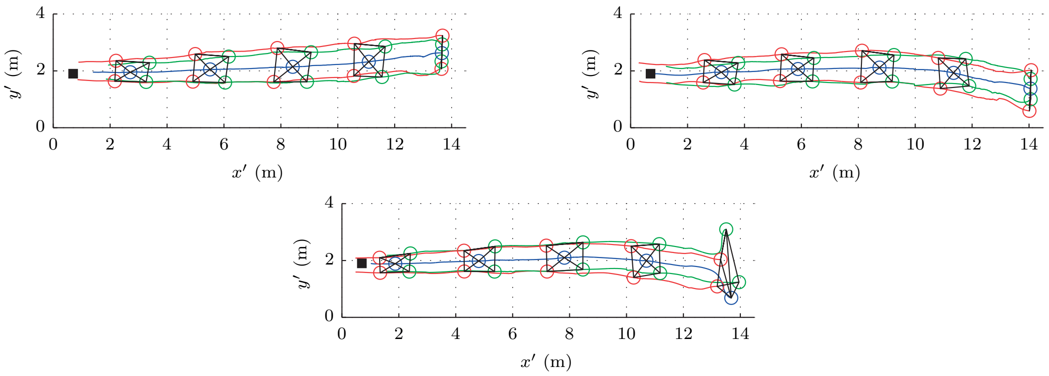


Figure 18.10 – Trajectories obtained using the five-robot rectangular formation, for three different sets of starting positions. Black lines are plotted connecting the robot positions every 40 s, and the odour source is represented by the black square (size not to scale) on the left side.

18.2.1 Rectangular formation

For the first series of experiments, we configure five nodes with the formation shown in Figure 18.9. Except for the centre node, each node is an endpoint to three edges in the graph, corresponding to the three relative positions that are generally considered for the purpose of formation control. The modified position of the centre robot harmonises the inter-robot distances and improves tracking reliability.

The robots start at the end of the tunnel opposite the source, approximately 13 m downwind. Figure 18.10 shows the results of separate runs with different starting positions:

- a crosswind line to the right of the plume centre (a)
- a crosswind line to the left of the plume centre (b)
- arbitrarily distributed in the environment (c)

The figure shows the robots quickly approaching the desired formation and reducing the crosswind spacing (i.e. narrowing the formation span) as they approach the source, marked by the black square on the left side of the plots. Some residual formation error is due to systematic

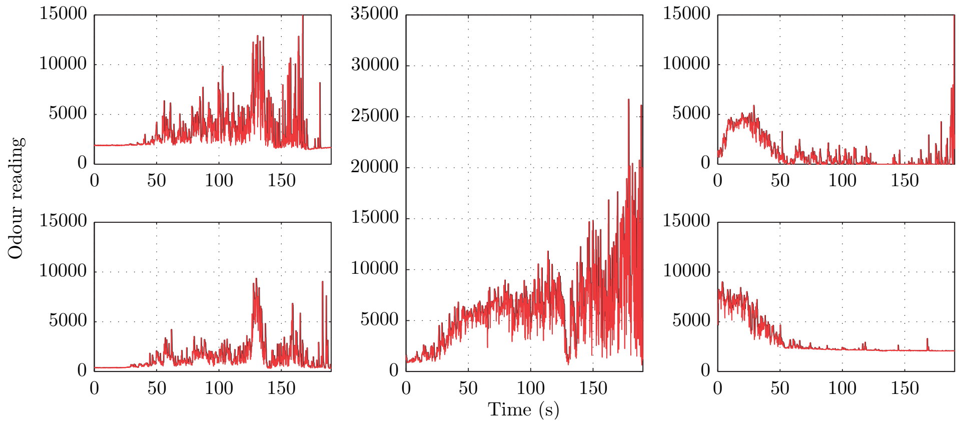


Figure 18.11 – Odour measurements (arbitrary units) obtained along the path during a complete experiment with five robots in a rectangular formation, starting on the left side of the plume. The plot position matches the robot positions, i.e. the top left plot corresponds to the front left robot.

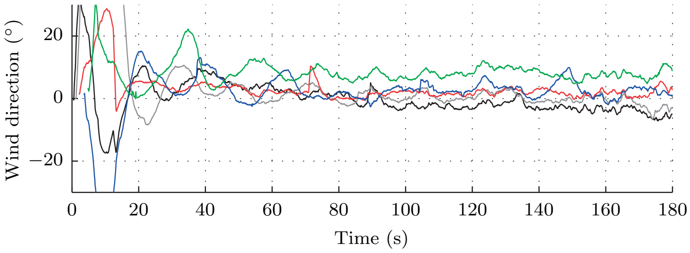
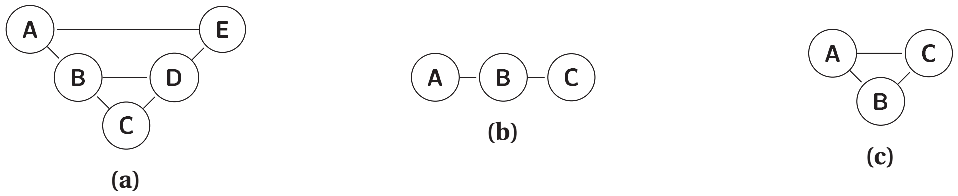


Figure 18.12 – Filtered wind direction estimates obtained for each robot during a complete experiment with five robots in a rectangular formation.

errors in relative positioning and, to a certain extent, inaccurate wind measurements. The tracking system introduces jitter to the ground truth, which is extraneous to the performance of the algorithm.

Figure 18.11 shows the odour measurements obtained by each robot during the second experiment with the five-robot rectangular formation. The plot positions correspond to the formation positions as defined in Figure 18.9. As the robots start on the left side of the plume, there is an initial period of higher readings for right-side robots—and, conversely, an initial period of low readings for left-side robots. Despite the increasing trend, particularly in the case of the centre robot, the measured concentrations continue to show extreme variations.

In line with simulation, real wind measurements are subject to substantial noise. Figure 18.12 shows the wind direction estimates for each robot, generated by the Kalman filter based on wind measurements and odometry information. Even post-filtering, both low- and high-frequency noise are non-negligible. Furthermore, one of the sensors is biased, returning consistently elevated measurements. Nevertheless, by explicitly and implicitly capitalising on information from all the robots, the algorithm is able to cope with imperfect sensing.



(a) (b) (c)

Figure 18.13 – Alternative formations: (a) five-robot inverted V formation; (b) three-robot linear formation; (c) three-robot inverted V formation.

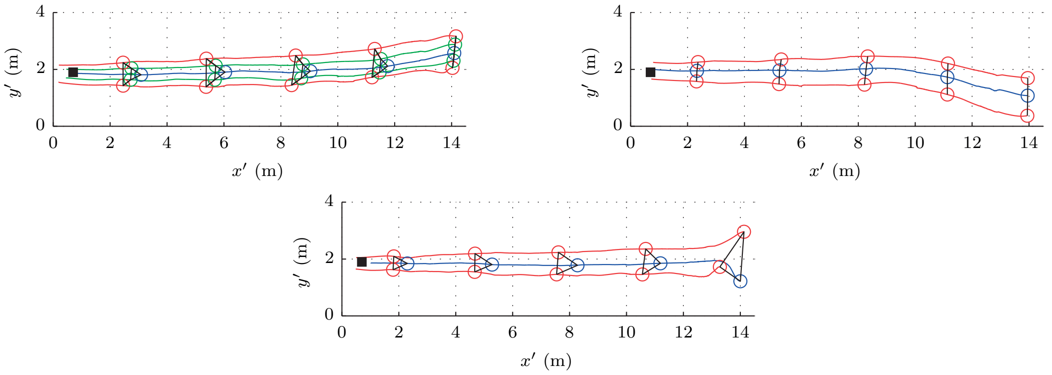


Figure 18.14 – Example trajectories obtained using the alternative formations, in the same order. Black lines are plotted connecting the robot positions every 40 s, and the odour source is represented by the black square (size not to scale) on the left side.

18.2.2 Additional formations

Additional experiments assessed three different formations: a five-robot inverted V formation (Figure 18.13a), a three-robot linear formation (Figure 18.13b), and a three-robot inverted V formation (Figure 18.13c). Three experiments were run for each case, with the starting positions of the robots following the same recipe as for the rectangular formation.

The outcomes do not differ substantially as a result of the starting conditions. Therefore, a single example run for each formation is shown in Figure 18.14, while Table 18.1 summarises the quantitative results. Prior to metric calculation, tracks are fed through a low-pass filter to abate SwisTrack noise; this operation does not substantially distort the actual trajectories, as system dynamics are of lower frequency.

The movement overhead is defined as $\alpha = d/\Delta - 1$, where d is the actual distance travelled by the formation centroid (i.e. the Riemann sum of its discrete step position variations) and Δ is the norm of its displacement (i.e. the straight-line distance between starting and final positions). The error is given by $\epsilon = |y_c - y_s|$, the distance between the crosswind position of the centre robot at the final coordinate (y_c) and that of the source (y_s).

While the number of runs does not allow us to draw statistically significant conclusions for each scenario, the average performance of the algorithm across the whole sample is $\bar{\alpha} = 0.54\,\%$, $\sigma_\alpha = 0.28\,\%$ and $\bar{\epsilon} = 0.04\,\text{m}$, $\sigma_\epsilon = 0.03\,\text{m}$. In the vast majority of cases, the robots

Table 18.1 – Movement overhead and final error obtained for each experiment.

		Five robots		Three robots	
		Rectangular	Inverted V	Linear	Inverted V
E1	α	0.30 %	0.32 %	0.42 %	0.89 %
	ϵ	0.05 m	0.04 m	0.04 m	0.01 m
E2	α	0.78 %	0.97 %	1.01 %	0.35 %
	ϵ	0.04 m	0.06 m	0.10 m	0.04 m
E3	α	0.38 %	0.34 %	0.51 %	0.24 %
	ϵ	0.01 m	0.04 m	0.01 m	0.04 m

trace the plume to a neighbourhood of the source smaller than one robot diameter, and do so while covering only a slightly longer distance than the optimal straight-line path.

There is not a clear winning formation. While the five-robot rectangular formation achieves the smallest mean overhead, the difference is not significant. From an efficiency perspective, it is preferable to attain the same performance with a smaller number of robots. Therefore, in this scenario, a three-robot formation could be regarded as a superior choice.

19 Alternative behaviours

In addition to a plume tracing algorithm, we have described a flexible framework for odour source localisation. Not only can the formations be dynamically adapted through the use of scaling factors, but the base formation biases may be switched on-line without jeopardising the stability of controller. The simple, isolated formulation of the components and their aggregation using a weighted sum make it easy to write new behaviours, enable and disable behaviours mid-run, and change their influence on the outcome.

This flexibility enables us to tackle different stages of the problem—or different problems—using the same core methods. In this chapter, we present possible additions to the framework that enable it to engage in plume acquisition at the beginning of the experiment and in source declaration at the end. We also introduce a modified algorithm for scenarios with no steady or measurable wind field.

19.1 Plume finding

The formation geometry used for tracing may not be well-suited for the initial plume acquisition phase. For instance, the authors of [55] investigate formation shapes for plume finding and discover that the best solution is a diagonal formation with equal spacing between the robots. The whole formation should move crosswind, with the optimal inter-robot distance depending on the wind speed.

This behaviour can be reproduced within our framework by setting bias vectors

$$\mathbf{b_{uw}} = \mathbf{b_{cw}} = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}^T \tag{19.1}$$

and scaling factors

$$s_{uw} = \begin{cases} -9.6e^{-8.8v_w} - 2.6v_w + 4.4 & \text{if } v_w \leq 0.3 \\ 2.8e^{-1.3v_w} - 0.07v_w + 1.1 & \text{if } v_w > 0.3 \end{cases} \tag{19.2}$$

$$s_{cw} = 8.3e^{-11.1v_w} - 0.007v_w + 0.15 \tag{19.3}$$

where v_w is the magnitude of the measured wind velocity vector and the optimised coefficients are copied from the reference. This assumes that the wind speed is roughly the same across all three nodes. Crosswind (as opposed to upwind) movement can be accomplished by changing the respective behaviour to $\mathbf{u_w} = R(\theta)\,[0 \quad 1]^T$ and setting $\mathbf{u_c} = 0$.

Finding the plume, as evidenced by measuring a concentration above some preset threshold, would trigger a switch to plume tracing mode, replacing the vectors, scaling factors, and behaviours with the ones defined in the previous chapters.

19.2 Source declaration

Odour source declaration can take advantage of select (e.g. rectangular) formations with differentiated positions along the upwind axis. While leading robots should, on average, get higher readings than trailing robots, this is not the case once they move past the odour source.

We can, therefore, outline a trivial source declaration method based on the continued difference between the readings of the leading and trailing robots. We define two new variables:

- c_f, the mean concentration measured by robots to the front of the formation centre

- c_b, the mean concentration measured by robots to the back of the formation centre

A source is declared when $c_f \ll c_b$, subject to proper thresholds; due to the intermittent aspect of the plume, a minimum time window must be ensured. Depending on the window and threshold, the method may be vulnerable to false positives, whereas false negatives should be a rarer occurrence in laminar flow conditions and for chemicals that are not otherwise present in the environment.

19.3 Windless plume tracing

Our work so far revolves around the tracing of odour plumes under the influence of measurable wind. There are other situations in which odour propagates through diffusion or slow convection. While our proposed algorithm cannot perform under these circumstances, the framework can be easily adapted to support adequate plume tracing strategies. Windless tracing strategies are also suitable when, despite the existence of a wind field, robots do not carry anemometers.

As a proof of concept, we propose a simple approach whose behaviour resembles an odour-seeking Braitenberg controller [68] but that uses several vehicles in formation instead of a single vehicle with multiple sensors. The core idea is to couple the rotation of this virtual vehicle to the concentrations detected on each side of the formation.

The readings influence the formation movement by either increasing or decreasing its reference heading. As we can no longer access the orientation reference provided by the wind, we assume the robots have access to a magnetic compass or equivalent global heading reference.

Heading control operates on the same principle as the plume centring behaviour in our standard algorithm. Robots calculate a formation centring error δ_c according to the expression

$$\delta_c = -u_c^{max} + \frac{2u_c^{max}}{1 + e^{-(c_l - c_r)/k_l}} \tag{19.4}$$

and a proportional-integral-derivative (PID) controller uses this error as input and outputs a normalised desired formation heading, ψ_f.

The plume tracing controller consists of two behaviours. The first is forward movement, replacing the previous upwind movement:

$$\mathbf{u_w} = R\left(\psi_f - \psi\right) \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{19.5}$$

Note the rotation by $\psi_f - \psi$, which translates to forward movement in the direction of the desired formation heading (and not the current robot heading). This is analogous to a forward bias but at the formation level.

The formation control behaviour $\mathbf{u_f}$ remains unchanged from (17.5). However, β_j takes a slightly different formulation, with the same ψ_f replacement and no scaling factors:

$$\beta_j = R\left(\psi_f - \psi\right) \left[\bar{\mathbf{p}}_\mathbf{j} - \bar{\mathbf{p}}_\mathbf{i}\right] \tag{19.6}$$

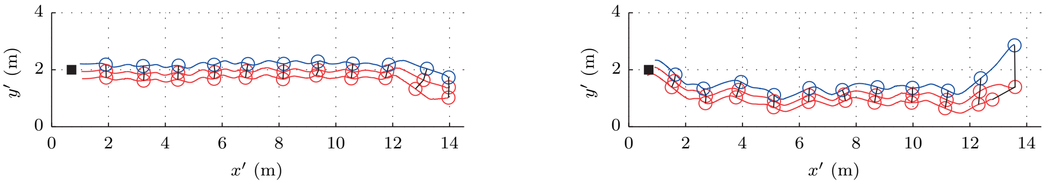
Instead of maintaining static alignment in the wind frame, the whole formation now rotates with the direction of movement.

The overall controller is again the simple weighted sum

$$\mathbf{u} = k_w \mathbf{u_w} + k_f \mathbf{u_f} \tag{19.7}$$

This strategy is tested in the wind tunnel using a linear formation with three Khepera IV robots. We run a first experiment with the same $1\,\mathrm{m\,s^{-1}}$ wind speed, followed by an experiment with wind speed under $0.03\,\mathrm{m\,s^{-1}}$ (the margin of error of our anemometer). In the latter case, the odour pump is kept running for ten minutes prior to the trial so as to flood the environment.

Figure 19.1a shows the trajectories followed by the robots using this controller in the $1\,\mathrm{m\,s^{-1}}$



(a)

(b)

Figure 19.1 – Example trajectories obtained using the windless controller: (a) 1 m s^{-1} wind and (b) <0.03 m s^{-1} wind. Black lines are plotted connecting the robot positions every 40 s, and the odour source is represented by the black square (size not to scale) on the left side.

wind field. Lacking the strong reference provided by the wind angle, the trajectories show significantly more oscillations. Nevertheless, when faced with the same airflow conditions as before, the robots trace the plume successfully.

In the absence of wind, the robots start by approaching the left wall of the tunnel and continue along that side before turning towards the source approximately 1 m out (Figure 19.1b). Whereas the final movement appears to validate our method, it is unclear why the robots start by moving left. This is, however, the observed behaviour in all experimental runs. We hypothesise that the low-speed airflow in the tunnel is biased and transports the wind closer to the wall—but, lacking an odour map for the tunnel, we cannot substantiate this theory.

20 Extension to 3D tracing

Our framework and algorithm are easily extensible to work in 3D, opening the way for implementation in AUVs or unmanned aerial vehicles (UAVs).

In this chapter, we present an extension to a heterogeneous team comprising multiple wheeled robots and a single flying robot, centred crosswind in the formation. This scenario was selected based on the capabilities of our experimental set-up.

The following sections describe the changes to the algorithm and present evaluation results gathered in Webots simulations and wind tunnel experiments.

20.1 Algorithm

The algorithm is essentially unchanged, with most modifications consisting of trivial extensions to the additional dimension. The same assumptions stand, and the only new piece of information available to the robots is the elevation to each team mate, a vertical counterpart to the bearing.

We assume that the wind is parallel to the floor. While accounting for non-parallel wind would not present major algorithmic complications, it would require the use of a 3D anemometer and would not result in a practical advantage, seeing as the wheeled robots cannot leave the ground plane. The expression for upwind movement is thus augmented with a null z component and becomes

$$\mathbf{u_w} = R_z(\theta)\begin{bmatrix}1\\0\\0\end{bmatrix} \tag{20.1}$$

where $R_z(\theta)$ denotes a rotation about the vertical axis by the horizontal wind direction θ.

The plume centring behaviour is likewise refashioned as

$$\mathbf{u_c} = R_z(\theta) \begin{bmatrix} 0 \\ -u_c^{max} + \frac{2u_c^{max}}{1+e^{-(c_l - c_r)/k_l}} \\ 0 \end{bmatrix} \tag{20.2}$$

It makes no sense to add a rule for vertical centring: centring, as defined, would require the entire formation to move up and down, which is not physically possible when using ground robots. Vertical adjustments are, for this reason, exclusively within the purview of formation control.

All substantive changes take place in the formation control component. First, the equation for $\mathbf{u_f}$ is extended with Laplacian feedback along the z axis:

$$\mathbf{u_f} = - \begin{bmatrix} \sum_{j=0}^{N} \mathscr{L}_j \left(x_j - \beta_j^x\right) \\ \sum_{j=0}^{N} \mathscr{L}_j \left(y_j - \beta_j^y\right) \\ \sum_{j=0}^{N} \mathscr{L}_j \left(z_j - \beta_j^z\right) \end{bmatrix} \tag{20.3}$$

Aligning the formation centroid with the source along the vertical axis requires manipulating the height of the flying robot, which is the same as scaling the formation in that direction. Therefore, the bias calculation β_j now includes a supplementary vertical scaling factor, s_v:

$$\beta_j = R_z(\theta) \begin{bmatrix} s_{uw} & 0 & 0 \\ 0 & s_{cw} & 0 \\ 0 & 0 & s_v \end{bmatrix} [\bar{\mathbf{p}}_\mathbf{j} - \bar{\mathbf{p}}_\mathbf{i}] \tag{20.4}$$

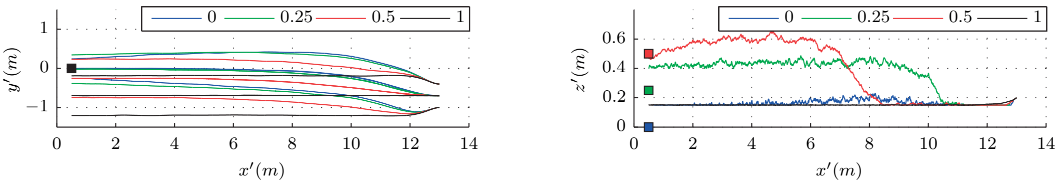
The expressions for the evolution of the scaling factors take the same form as in the 2D wind tunnel implementation:

$$\dot{s}_{cw} = k_{cw}((c_l + c_r) - c_c) \tag{20.5}$$

$$\dot{s}_v = k_v(0.66c_t - c_b) \tag{20.6}$$

The quantities c_t and c_b used for vertical scaling are the average concentrations measured by the top and bottom robots. The factors s_{cw} and s_v are bounded by appropriate minimum and maximum limits, taking into account operational constraints.

The goal of (20.6) is to match the odour concentration measured by robots on the ground and flying. Equation (20.5) results in a mean bottom concentration c_b that tends towards two-thirds of the bottom centre concentration; as we work with a single centred flying robot, we adjust (20.6) accordingly to maintain vertical symmetry.



(a) **(b)**

Figure 20.1 – Trajectories described using the 3D controller with the odour source placed at different heights. (a) Top view of the trajectories of all robots. (b) Side view of the trajectories of the flying robot. The odour sources are represented by the squares on the left side, except for the source at $h = 1\,\text{m}$, which is outside the operating range and not pictured.

20.2 Evaluation

The algorithm is first tested in Webots simulations, using a four-robot team consisting of three ground robots and one flying robot, capable of 3D movement. For simplicity, we adopt the built-in model of the Khepera III for all robots, including the flying one. As the Khepera is a wheeled robot and is not able to move vertically, we implement an external supervisor to displace the flying robot in 3D; this is a good approximation to the case in the wind tunnel. All other simulation parameters remain unchanged.

Figure 20.1 shows the trajectories followed by the robot in four discrete simulation runs with different source heights. In all runs, the robots start in the same positions, with the formation shifted 0.5 m off-centre.

The ground tracks are presented in the left plot. For low plumes (at $h = 0\,\text{m}$ and $h = 0.25\,\text{m}$), the robots quickly move to the plume centre and the formation reaches the end of the tunnel aligned with the plume source. In fact, the only clearly observable difference between the two trajectories is the scale of the formation, reflecting a change in the odour profile. As the source height increases, the tracing behaviour deteriorates. For $h = 0.5\,\text{m}$, the formation still moves towards the plume but reaches the end approximately 0.2 m off-centre. No centring effort is observed when the source is placed at a height of 1 m.

In the right plot, we can see the vertical trajectory of the flying robot. For a ground plume, the robot remains at its lowest defined z bias of 0.15 m. As the source height increases, so does the robot climb higher. However, when the source is at 1 m, the plume is no longer detected by any of the robots (readings under 0.1 in a 0–1000 range), the formation blindly moves upwind and the flying robot again stays at the minimum height.

It is clear from the results that this approach breaks down for plumes above a certain height. As we use a single flying robot, the centring behaviour is determined solely by the robots on the ground. Consequently, there can be no lateral adjustments when the plume does not reach ground level, and failure is inevitable.

In order to assess the performance impact of introducing 3D tracing for source heights at

Table 20.1 – Mean absolute final crosswind error (lateral distance from the formation centre to the source) for different source heights. Each entry (mean and standard deviation) is the result of 100 simulation runs.

	Source height (m)				
	0.10	0.15	0.20	0.25	0.30
2D	3.1 ± 2.3 mm	3.3 ± 2.4 mm	3.8 ± 3.4 mm	4.9 ± 3.5 mm	6.6 ± 4.6 mm
3D	2.5 ± 1.8 mm	2.5 ± 1.7 mm	2.4 ± 1.8 mm	3.3 ± 2.5 mm	5.2 ± 3.8 mm

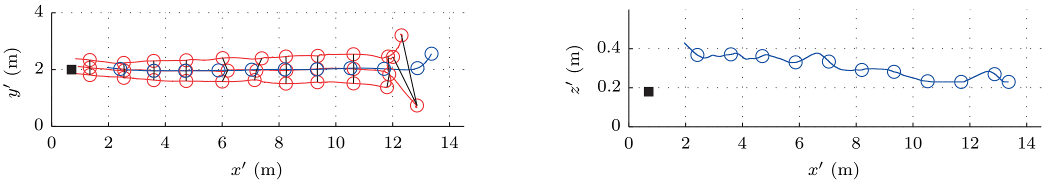


Figure 20.2 – Top and side view of the trajectories described using the 3D controller with the odour source placed at $h = 0.18$ m. The ground robots are drawn in red and the flying robot in blue. Black lines are plotted connecting the robot positions every 40 s, and the odour source is represented by the black square (size not to scale) on the left side.

which all robots can detect the plume, we perform a quantitative evaluation of our algorithm with and without the flying robot. For each configuration and source height (0.10 m, 0.15 m, 0.20 m, 0.25 m and 0.30 m), the simulation is repeated 100 times.

The final lateral errors, presented in Table 20.1, show that, in this range, the 3D formation consistently outperforms the solution featuring only ground robots, yielding lower mean error with lower standard deviation. The error is decreased by up to 36 %. Furthermore, the 3D algorithm is more resilient to increased plume heights, its performance only beginning to degrade when the source reaches 0.25 m.

We also test the algorithm in the wind tunnel using a team of Khepera IV robots, one of which is mounted on the traversing system. As our relative positioning boards only work in 2D, we obtain absolute positions from SwisTrack and the traversing system. These positions are merged in an external interface application, and the resulting range, bearing, and elevation data are sent to the robots. The formation is the same as in simulation, only with the flying robot now trailing 1 m behind so as not to obstruct the camera view.

The flying robot does not actuate its motors and instead sends commands to an auxiliary program that operates the traversing system. Due to safety and mechanical restrictions, the vertical range of the traversing system only extends from a bottom height of 0.23 m to a top height of 0.53 m. The limited range impacts our ability to test the algorithm: any source close to the floor or higher than 0.25 m results in saturation of the vertical control.

Figure 20.2 shows the trajectories followed by the robot when a source is placed laterally centred at a height of 0.18 m. The horizontal tracks are similar to those seen before, with the robots quickly aligning to the source. However, they are now seen straying from the path

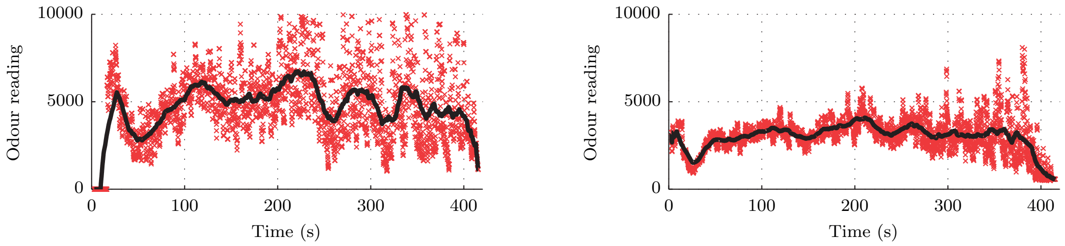


Figure 20.3 – Odour readings for top robot (left) and bottom robots (right) over time, corresponding to the experiment shown in Figure 20.2. The individual readings are plotted along with a 250-sample moving average for readability.

during the final 1 m stretch. Because the source is placed higher than the sensors, robots cease to detect the plume as they get too close, and start to turn based on spurious environmental readings or miscalibrated sensor offsets.

This final concentration drop can be seen in Figure 20.3, which plots the concentration readings over time (left-to-right evolution in the time plot corresponds to right-to-left evolution in the position plot), both in raw form and smoothed over a twenty-second sliding window for easier reading. The left plot shows the top robot readings and the right plot shows the averaged ground robot readings.

The concentration plots also help understand the vertical track, in which we see the flying robot start from the minimum height and stay low for the first 2 m of upwind movement. Afterwards, it starts detecting higher concentrations and climbs to trace the plume, reaching the end at approximately 0.42 m. The bottom concentration tracks two-thirds of the top concentration, as designed.

Our experimental set-up suffers from constraints that prevent testing a complete 3D scenario, the most relevant being the limitation to a single flying robot. Despite these constraints, we have shown that our augmented 3D algorithm can outperform the solution using only ground robots.

A formation-based algorithm is a natural candidate for implementation in teams in which all robots are capable of unrestricted movement. In such a scenario, adding a similar plume centring behaviour on the vertical axis would allow the formation to trace a plume in all three dimensions. It is, however, worth noting that a true flying robot would generate its own air flow, affecting the odour measurements and the plume itself in ways that are hard to predict.

21 Summary

This part of the thesis presented a decentralised framework for formation-based multi-robot odour plume tracing. Our algorithm consists of a set of simple behaviours, including graph-based formation control, upwind movement and plume centring. Starting anywhere in the plume, and using odour concentration and wind direction information, it enables a group of robots to trace the plume to its source.

We showed both Webots simulations and real-world experiments performed in our wind tunnel, using Khepera III robots equipped with custom sensors. The algorithm runs on board and is completely self-contained, with no dependence on external computing or data. It tracks the plume to the source with average distance overhead approaching 0.5 % in wind tunnel experiments.

In addition to our base algorithm, we discussed how the framework can be extended to tackle different problem phases, such as plume finding and source declaration, or different conditions, such as windless tracing. These require only minor modifications or additions, building upon the flexibility of our base formulation.

We finished by presenting an extension of the work to 3D plume tracing using a mixed team of ground and flying robots, this too only requiring minor additions to the algorithm. We tested this algorithm using Khepera IV robots in the wind tunnel, including one mounted on the traversing system, and show that it successfully traces the plume in three dimensions while reducing error by up to 36 %.

Part V

Formation-based waterborne plume tracing

The water was icy; Harry's waterlogged clothes billowed around him and weighed him down. Taking deep breaths that filled his nostrils with the tang of salt and seaweed, he struck out for the shimmering, shrinking light now moving deeper into the cliff.

— J. K. Rowling, *Harry Potter and the Half-Blood Prince*

22 Introduction

In the aftermath of the Deepwater Horizon rig explosion [69], the resulting oil spill jetting out of the Macondo wellhead at 62 000 barrel/day [70] extended across tens of kilometres in the Gulf of Mexico, yielding one of the most recognisable images of a chemical plume (pictured in Figure 1.1b). The Sentry AUV, equipped with a mass spectrometer, helped map the plume [71].

Tracing oil spills or pipeline leakage is but one potential application of marine plume tracing. There are other environmental applications, such as tracing the dispersion of wastewater outfall or unauthorised pollutant discharges, or searching for hydrothermal vents, benthic structures that support complex ecosystems and release large volumes of heated water with detectable chemical signatures. As with land mines, detection of underwater unexploded ordinance (UUXO) is a major concern and has traditionally been the subject of magnetic and sonar imaging methods [72], even though chemical tracing has the potential to reduce false positives.

In the previous part, we looked into odour plume tracing in the air. Now we shift our attention to waterborne plumes and the methods to track them, capitalising on our proposed formation control approaches.

We focus on constant-depth and near-surface operations. Although that may seem like a major limitation, water bodies are subject to stratification, with layers forming based on salinity, oxygenation, density, and temperature [73]. On short time scales, some plumes remain almost horizontal, as did the main Deepwater Horizon oil plume, at approximately 1120–1160 m depth [71]. In the case of freshwater plumes in the ocean, such as those produced by river discharges, buoyancy often leads to expansion primarily along the surface [74].

To trace a near-surface plume, we propose a triangular formation of two surface vehicles and an underwater vehicle. The two surface vehicles run an adapted version of our airborne plume tracing solution built around a graph-based formation, while the follower AUV runs the range-based formation controller and relays concentration data, but is otherwise unaware of the plume tracing task.

We evaluate our algorithm in simulation, using the MEDUSA vehicle models and a Fluent-generated freshwater plume, spanning a much larger area than the ethanol plumes in the wind tunnel. Using the combined approach, the vehicles are able to stay in a group, centre on the plume, and scale the formation while tracing the plume to its source.

23 Related work

Some early work in underwater odour source localisation tried to mimic the well-studied behaviour of bottom-dwelling lobsters [75], [76]. This led to the development of lobster-inspired robots [77] and Braitenberg-like controllers that link wheel speeds to the differential readings of the two antennae [78]. Other authors have looked at the active fanning behaviour of crayfish [79].

Related bio-inspired work includes [80], [81], where the authors propose a surge-cast algorithm in which robots switch between moving at a small offset angle to up-flow and a plume reacquisition manoeuvre. The approach was tested in experiments with the REMUS AUV [82]. Although tangential to our work, REMUS has also been used for odour mapping [83].

A significant body of work deals with hydrothermal vent localisation. The authors of [84], [85] propose an approach in which vehicles perform a preplanned sampling survey and switch to spiral movement when a chemical signature is detected. The same team also introduced an approach based on occupancy grid maps [86].

In [87], the authors advance a partially-observable Markov decision process (POMDP) formulation for the hydrothermal vent localisation problem and propose several planning methods building upon the aforementioned occupancy grid mapping algorithm. A Bayesian inference solution using a hydrothermal plume model and artificial potential fields is proposed in [88].

An approach for cooperative 3D plume tracing is described in [89] and based on previous 2D work with temperature plumes [90]. The authors derive a controller that allows a vehicle to independently decide its movement direction based on a quadratic plume model and the sharing of localised concentration measurements.

Recent recognition for the field of waterborne odour source localisation comes in the form of its inclusion in the euRathlon competition [91], in both the 2014 sea edition and the 2015 grand challenge that brought together land, water and flying robots. Modelled on the Fukushima Daiichi nuclear accident of 2011 [92], relevant challenges include finding workers missing at sea and locating underwater contaminant leaks.

24 Technical approach

The problem formulation for waterborne odour plume tracing does not depart significantly from its airborne equivalent. There are, nevertheless, practical considerations that require differentiated coping mechanisms.

Our target scenario features a neutrally buoyant surface plume, in particular a freshwater plume in a saltwater medium. The plume is detectable by sensors at the surface, as well as by sensors up to a limited depth, and is therefore traceable by a joint team of surface and underwater vehicles. Our goal, as in the previous part, is to follow an already detected odour plume to its source.

For a team entirely consisting of autonomous surface vehicles (ASVs), the approach outlined in Part IV is applicable with minor changes. However, the addition of an AUV lacking access to fully resolved relative positions precludes the exclusive use of our Laplacian feedback formation controller. We propose a set-up similar to that in Part III, with two leader surface vehicles and a trailing underwater vehicle, which enables the use of our acoustic ranging approach for controlling the underwater vehicle.

We assume that the thickness of the plume does not change substantially, making it possible for the AUV to maintain a constant depth, and that an independent depth controller provides that functionality. While transposing the vertical scaling mechanism from Chapter 20 is technically straightforward, it would require additional communication and further reduce the ranging frequency, thereby impacting performance.

Our work targets the capabilities and restrictions of the MEDUSA robots. We propose a controller for the surface vehicles that mirrors the one used in the Khepera robots, but taking into account the different sensing and actuation modalities. In particular, the algorithm works with absolute positions and generates a control output in heading instead of angular speed. The follower runs the unmodified range-based formation controller, and all vehicles transmit their concentration readings.

The next sections contain a description of our approach, divided into sensing, leader control, and follower control. The flexibility of our framework ensures that minimal changes are required. Accordingly, we restate the major points of the solution but refer the reader to the original description of the algorithms for full derivations and explanations.

24.1 Sensing and communication

All vehicles are outfitted with appropriate odour sensors. In our particular simulation scenario, we track a freshwater plume, discriminated by its salinity S. Salinity is normally measured using temperature-corrected conductivity probes, which are much faster than the VOC sensors we used in the air. We can therefore assume instant odour measurements.

Tracking a freshwater plume in saltwater requires searching for lower salinity readings, as opposed to our previous search for higher odour concentrations. To avoid changes to the algorithm, the measured salinity can be converted to a *freshness* value by subtracting it from an upper bound:

$$c_i = S_{max} - S_i \tag{24.1}$$

where c_i is the resulting concentration value for vehicle i, S_i is the salinity measured by its sensor, and S_{max} is a salinity upper bound appropriate to the environment, inferred from environmental measurements or known data. The mean sea water salinity is 34.7 ppt and, in the absence of local information, 50 ppt would be an adequate bound for the majority of surface waters. The concentrations readings are run through a 4 s sliding window max filter, with its output being used by the algorithm.

The underwater vehicle has no access to absolute positions, whereas the surface vehicles are assumed to be equipped with a compass and an accurate GPS receiver. We treat the GPS positions as error-free, which is essentially the case for short-range relative positions using RTK systems. Where RTK is not available, a sensor fusion approach including GPS, IMU, and AHRS data should yield accurate enough information.

We also assume at least one of the surface vehicles has some access to the direction of the current. This information may come from direct measurement, be estimated using either a DVL or GPS and a dynamical model of the vehicle, or be obtained from external sources. It may also consist solely of model data. While real-time point-accurate information may improve performance, it is not an absolute requirement. In strong currents, having access to the velocity of the current instead of just its direction also helps improve the behaviour of the vehicle.

Communication between surface vehicles is performed over short-range radios and is considered to be instant and unconstrained in data rate. This is a workable assumption given how little information is exchanged between the robots: only the most recent odour readings and positions. We do not implement any mechanisms to cope with packet loss, but the algorithm is resilient to isolated communication failures.

Communication with the underwater follower takes place over an acoustic link, subject to the previously discussed limitations. As was already the case, the leader vehicles send their current heading piggybacked on the ranging replies when alternately queried every 2 s. The solution departs from the pure formation-control approach in that the ranging requests sent by the follower now also include piggybacked data, namely its max-filtered odour concentration measurements. Both leaders listen to all requests.

24.2 Leader control

The two leaders implement the approach outlined in Part IV with minor adjustments. The follower is part of the formation graph, but only so we can properly classify odour readings; it is otherwise disconnected, and its formation control operates independently.

Relative positions, in the form of ranges and bearings to neighbours, are replaced with absolute positions transmitted over the radio. This is not because of a requirement of the algorithm (we already showed it to work with only noisy relative positions) but because GPS is more commonly available in ASVs than relative positioning devices. This change allows us to more accurately reflect vehicle capabilities and showcases the potential for alternative implementations.

To reflect the new circumstances, we use the terms up-flow and cross-flow where we previously referred to upwind and crosswind. The mathematical notation remains unchanged.

24.2.1 Laplacian feedback

We retain the approach from Part IV, only adapted to handle global positions. As we are now working with a specific triangular configuration, we propose the graph $G = (V, E)$ with

$$V = \{1, 2, 3\} \tag{24.2}$$

$$E = \{(2, 3)\} \tag{24.3}$$

and corresponding bias vectors

$$\mathbf{b_{uw}} = \begin{bmatrix} -1 & 0 & 0 \end{bmatrix}^T \tag{24.4}$$

$$\mathbf{b_{cw}} = \begin{bmatrix} 0 & -\frac{d}{2} & \frac{d}{2} \end{bmatrix}^T \tag{24.5}$$

This corresponds to the formation in Figure 24.1, with inter-vehicle base distance d. As no edge exists connecting the trailing vehicle, its bias is not considered for formation control; any negative constant may be used.

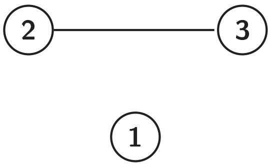


Figure 24.1 – Triangular marine formation showing graph edges.

The resulting velocity vector for formation control remains

$$\mathbf{u_f} = -\begin{bmatrix} \sum_{j=0}^{N} \mathscr{L}_j \left(x_j - \beta_j^x\right) \\ \sum_{j=0}^{N} \mathscr{L}_j \left(y_j - \beta_j^y\right) \end{bmatrix} \tag{24.6}$$

where $\mathscr{L}_j = \mathscr{L}_{i,j}$ is the entry of the Laplacian matrix relating the controlled node i to neighbour j. In contrast to the wind tunnel approach, x_j and y_j are now the absolute positions of vehicle j.

The dynamic bias β_j assumes a new formulation, replacing relative positions with the absolute positions and suppressing the up-flow scaling factor:

$$\beta_j = R(\theta) \begin{bmatrix} 1 & 0 \\ 0 & s_{cw} \end{bmatrix} \begin{bmatrix} \mathbf{b_j^{uw}} \\ \mathbf{b_j^{cw}} \end{bmatrix} \tag{24.7}$$

The matrix $R(\theta)$ acts as a coordinate transformation, with θ now the local estimate of the current direction in the global frame. The scale in the up-flow direction is controlled solely by the follower vehicle, therefore no scaling factor is warranted. The cross-flow scaling factor s_{cw} still allows the formation to be scaled to adapt to the plume width, and its evolution is governed by

$$\dot{s}_{cw} = k_{cw}((c_l + c_r) - c_c) \tag{24.8}$$

24.2.2 Upflow movement

We maintain an identical rheotaxis behaviour by defining a movement urge in the direction of the estimated current:

$$\mathbf{u_w} = R(\theta) \begin{bmatrix} 1 + \frac{|w|}{k_w} \\ 0 \end{bmatrix} \tag{24.9}$$

In scenarios with strong currents, the base up-flow urge may not suffice to overcome the force dragging the vehicle back. If an estimate of the current speed $|w|$ is available, it can be added to the base urge. If not, the vehicles are still able to trace the plume provided the current is weaker than the urge, although the up-flow movement speed decreases accordingly.

24.2.3 Plume centring

The centring behaviour assumes the same formulation of a generalised logistic response on the difference between side concentrations:

$$\mathbf{u_c} = R(\theta) \begin{bmatrix} 0 \\ -u_c^{max} + \frac{2u_c^{max}}{1+e^{-(c_l - c_r)/k_l}} \end{bmatrix} \tag{24.10}$$

24.2.4 Behaviour aggregation

The velocity vectors for each component are combined using the simple sum

$$\mathbf{u} = k_w \mathbf{u_w} + k_c \mathbf{u_c} + k_f \mathbf{u_f} \tag{24.11}$$

As the vehicle controller takes (v, ψ) inputs for navigation, the resulting desired movement vector $\mathbf{u} = \begin{bmatrix} u_x & u_y \end{bmatrix}^T$ is transformed into its magnitude and direction:

$$v = \|\mathbf{u}\| \qquad\qquad 0 \le v \le v_{max} \tag{24.12}$$

$$\psi = \text{atan2}(\mathbf{u}_y, \mathbf{u}_x) \tag{24.13}$$

24.3 Follower control

The follower controller was designed from the start to be independent of the navigation mechanism used by the leaders. This allows us to employ the controller introduced in Part III with no changes, keeping the follower blind to the plume tracing problem.

We restate the main speed and heading controller expressions

$$v = K_p^s \epsilon + K_i^s \int_0^t \epsilon \, d\tau \tag{24.14}$$

$$\psi = \hat{\psi}_f + \gamma(K_p^h \delta) \tag{24.15}$$

where ϵ and δ are the common- and differential-mode errors, derived from the ranges and distance set-point, and all other terms retain their meanings. We refer the reader to Chapter 11 for details.

The one difference in the follower implementation is that it now measures the odour concentration and relays it to the leaders as part of the acoustic ranging requests. As in the original work, we assume the follower has independent depth control and maintains a constant depth D.

For simpler configuration, we use the same desired inter-vehicle distance d in leader and follower controllers. As we do not correct for the cross-flow scaling of the leader formation, and in order to ensure the stability of the controller, the scale factor s_{cw} must fulfil

$$s_{cw} < 2\sqrt{1-\left(\frac{D}{d}\right)^2} \tag{24.16}$$

Larger values drive the leaders too far apart from one another and make it physically impossible for the follower to attain the desired ranges. This is not a significant limitation, as the base distance d can be jointly tuned with s_{cw} to achieve the desired minimum and maximum formation spans.

25 Evaluation

We evaluate the complete solution in a simulated environment approaching our target scenario for future real-world trials. We run the robot simulation in MATLAB/Simulink, using the MEDUSA vehicle model to trace a freshwater plume generated in ANSYS Fluent. The following sections detail the set-up and results obtained.

25.1 Set-up

Prior to the vehicle simulation, we generate our base plume in ANSYS Fluent. In a volume measuring 108 m × 40 m × 4 m, we place a 1 cm × 1 cm hose outlet at the surface, emitting freshwater in the direction of the current, at a mass flow rate of $1\,\mathrm{kg\,s^{-1}}$. The background current flows at $5\,\mathrm{cm\,s^{-1}}$ along the length of the volume.

The simulation models chemical species transport, with freshwater having salinity 0 ppt and density $998.2\,\mathrm{kg\,m^{-3}}$ and saltwater having salinity 35 ppt and density $1010\,\mathrm{kg\,m^{-3}}$. Both are at a default temperature of 24 °C.

We run a steady-state simulation, with standard gravity conditions and the k–ϵ viscosity model, for a total of 500 iterations, enough for the solution to converge. The plume obtained is pictured in Figure 25.1. Two horizontal profiles of the plume are obtained—at the surface for the ASVs, and at a depth of 0.5 m for the AUV—and are imported into MATLAB.

In the absence of real readings to calibrate the model, we choose to approximate the odour signatures seen in the wind tunnel and add error to the salinity readings drawn from a half-normal distribution with $\sigma = 0.1\,\mathrm{ppt}$, the same order of magnitude of the salinity differences between the plume and the medium. Because the freshwater plume can never cause a salinity increase, we saturate the resulting values to the medium salinity $S = 35\,\mathrm{ppt}$. An example snapshot is presented in Figure 25.2, showing the resulting patchy plume.

As we do not foresee having access to real-time current information when using the real MEDUSA vehicles, we choose to provide the robots only with the base current, $0.05\,\mathrm{m\,s^{-1}}$

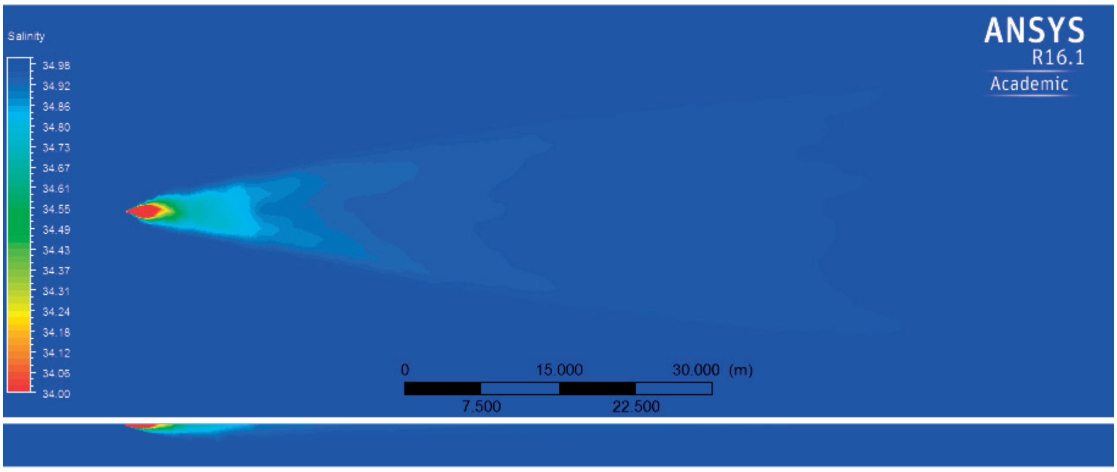


Figure 25.1 – Surface (top) and side (bottom) views of the freshwater plume in ANSYS Fluent. The plume measures approximately 75 m × 25 m × 2 m at $S \leq 34.98$ ppt.

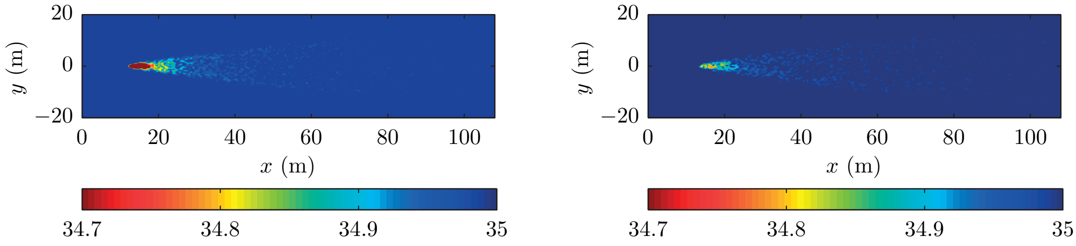


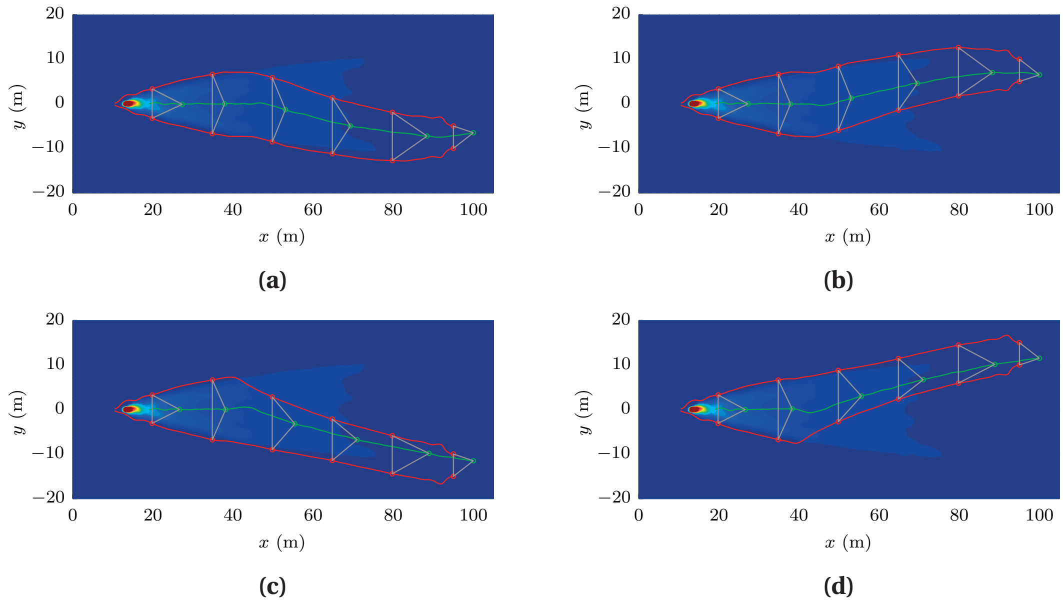
Figure 25.2 – Snapshot of the time-variant salinity levels with added half-normal noise, at the surface (left) and at a depth of 0.5 m (right).

along the length of the arena. The actual current field generating the plume is, however, non-uniform, due to the effects of the high-speed freshwater inlet and of turbulence.

The robot simulation uses the MEDUSA Simulink model for all three vehicles. The two surface vehicles run identical copies of the leader controller whereas the AUV runs the follower controller. We only consider movement on the 2D plane, and ignore the depth of the AUV for all purposes other than odour sampling.

The vehicles start on the right side, in the plume, and in their approximate target formation. The follower vehicle must start behind the leaders in order to converge to the desired following position.

All simulation parameters from Part III are maintained. Because of the different sensing range, the leader controller uses updated parameters $k_{cw} = 5 \times 10^{-2}$ and $k_l = 2 \times 10^{-2}$. Furthermore, as the response of the vehicles is slower, we decrease the formation control weight $k_f = 0.5$, while leaving $k_w = k_c = 1$.



(a)

(b)

(c)

(d)

Figure 25.3 – Vehicle trajectories for different starting positions, superimposed on the base plume. The leader vehicles are plotted in red and the follower vehicle in green.

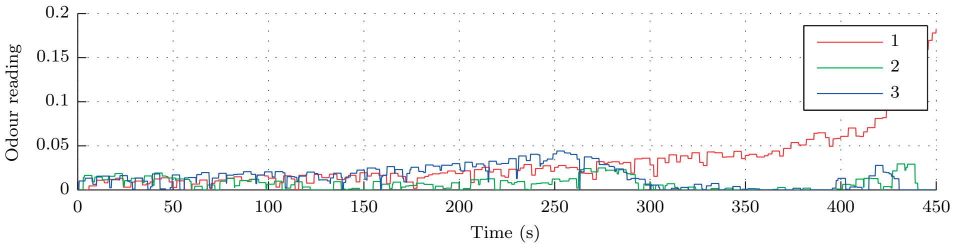


Figure 25.4 – Max-filtered odour readings of all the vehicles in the experiment in Figure 25.3a. Vehicle 1 is the follower, and vehicles 2 and 3 are the leaders.

25.2 Results

We run simulations for different starting positions on the sides of the plumes. Figure 25.3 shows the resulting trajectories followed by the vehicles. The robots converge to the plume in all cases, although the time taken to reach the centre line depends on the starting offset.

Increments to the cross-flow scaling factor are proportional to the difference in readings, which is more pronounced when exposed to the higher concentrations in the centre of the plume. Therefore, the formation tends to only widen to the plume span as the robots approach its centre, and formations starting closer to the plume begin to trace its limits earlier in the experiment.

Figure 25.4 shows the odour concentrations measured by each vehicle, after being put through the max filter. For the initial 250 s the readings increase as the vehicles make their way into and up the plume. The elevated concentrations on the right side (vehicle 3) cause them to

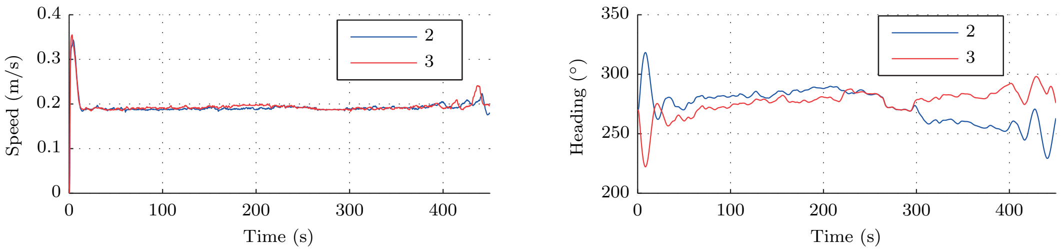


Figure 25.5 – Actual speed and heading of the leaders for the experiment in Figure 25.3a. The up-flow direction is 270°.

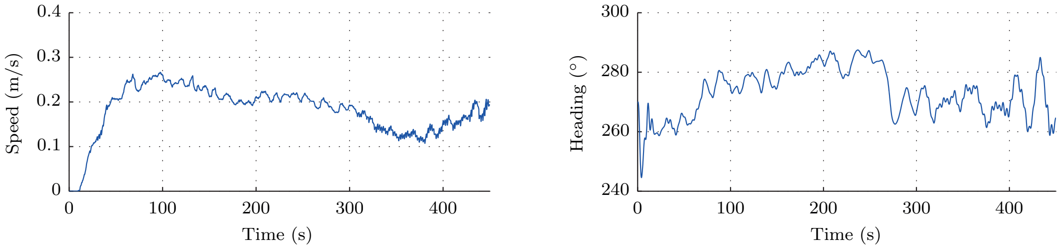


Figure 25.6 – Actual speed and heading of the follower for the experiment in Figure 25.3a. The up-flow direction is 270°.

move cross-flow. At $t = 270\,\mathrm{s}$ the formation is centred in the plume, and the concentrations measured by the follower quickly increase. The leaders are now on the edge of the plume and measure low concentrations, leading the formation to contract as it approaches the source.

The speed and heading of the leaders is plotted in Figure 25.5. Speed is mostly driven by up-flow movement and almost constant throughout the experiment, after the initial formation convergence. Whereas the up-flow heading is 270°, the actual heading reflects the behaviour of the vehicles: in a first phase, both move at elevated headings while centring on the plume; then, their headings progress in opposite directions as the formation scales.

The follower movement, in Figure 25.6, is less constant. The lack of accurate positioning and reliance on low-frequency range measurements introduces sawtooth disturbances in both signals. Speed is further affected by the change in inter-leader distance: while the leaders move at near-constant speed, formation scaling requires the follower to modulate its speed in order to move forward or back and keep its distance to the leaders. This phenomenon is most clear from $t = 300\,\mathrm{s}$, when the formation narrows for the final approach to the source. The heading of the follower tracks that of the leaders during the centring phase and afterwards remains close to the up-flow direction.

26 Alternative behaviours

We have so far introduced a balanced solution with various applications. Nevertheless, alternative mechanisms may be required to handle specific scenario requirements. In the following sections, we outline possible extensions to our methods.

26.1 Scale adjustment

In the interest of avoiding additional data transmission, we do not feed any scale information to the follower, which always tries to achieve a preset distance to the leaders. As the leaders increase or decrease the formation span, this results in changes to the up-flow distance to the follower, i.e. the closer the leaders get to one another, the farther behind the follower lags.

Although this effect has limited impact on the performance of the solution, there are situations in which it might be desirable to compensate for it. Doing so merely requires replacing d in (11.1) and (11.2) by

$$d_f = \sqrt{d_x^2 + \left(s_{cw}\frac{d}{2}\right)^2 + D^2} \tag{26.1}$$

where d_x is the desired up-flow follower offset, d is the base cross-flow bias between the leaders, and D is the depth of the follower. This also removes the restriction on s_{cw} imposed by (24.16).

Whereas all the other terms are constant and preconfigured, computing d_f requires that the cross-flow scale coefficient s_{cw} be sent to the follower over the acoustic link. Adding another numeric field to the ranging response packet, while trivial, comes at the cost of further decreased ranging frequency.

26.2 Variable depth operation

Assuming the follower vehicle is equipped with an appropriate depth controller, varying the depth to improve tracing performance or to gather a vertical profile of the plume can be accomplished by using the approach in Chapter 20.

The leaders, armed with knowledge of the odour concentrations at the surface and underwater, can compute a scaling factor s_v such that the follower traces the vertical shape of the plume. The follower, however, does not have access to the concentrations measured by the leaders and is unable to perform this computation.

In order to effect the depth changes, s_v must be transmitted to the follower. Because varying the depth affects the measured ranges and further constrains the cross-flow scaling factor s_{cw}, this approach should be combined with that outlined in the previous section.

To implement the integrated solution, the leader vehicles include (s_{cw}, s_v) in the acoustic replies sent to the follower. The follower transfers the depth reference $s_v D$ to the depth controller, and computes the target range d_f using

$$d_f = \sqrt{d_x^2 + \left(s_{cw}\frac{d}{2}\right)^2 + (s_v D)^2} \tag{26.2}$$

27 Summary

In this part of the thesis, we bridged the gap between our previously described marine formation control and airborne plume tracing approaches by introducing a solution for waterborne plume tracing that combines the two. Our method enables a triangular formation composed of two surface vehicles, acting as leaders, and an underwater vehicle, acting as follower, to trace a near-surface plume to its source.

The leaders in our combined approach implement the graph-based formation plume tracing algorithm introduced in Part IV. We make only minor changes, namely the replacement of relative positions with absolute GPS coordinates, which approximate the algorithm to the sensing capabilities and constraints of ASVs while preserving its core ideas.

The follower AUV uses the range-based formation controller from Part III to pursue the leaders, with no knowledge of the plume tracing task. In addition, it measures the odour concentration and relays the information as part of the ranging requests, with minimal overhead.

We demonstrated the approach working in simulation using the full dynamical model of the MEDUSA vehicles and a freshwater plume in a saltwater medium, generated using a high-fidelity CFD application. Despite the added complexity, the robots successfully trace the plume to its source.

A major take-away of this part is the little effort required to convert the land-based tracing algorithm to work with marine vehicles, and to merge graph-based and range-based formation controllers into a hybrid approach. This attests to the generality of our methods and the flexibility of the developed framework.

Part VI

Discussion

'That wand's more trouble than it's worth,' said Harry. 'And quite honestly,' he turned away from the painted portraits, thinking now only of the four-poster bed lying waiting for him in Gryffindor Tower, and wondering whether Kreacher might bring him a sandwich there, 'I've had enough trouble for a lifetime.'

— J. K. Rowling, *Harry Potter and the Deathly Hallows*

28 Conclusions

In the first lines of the document, we set forth an objective: *to follow that freshly baked bread smell to a nearby bakery*.

Over the course of this thesis, we examined the problem of odour source localisation, with a particular focus on odour plume tracing. The main outcome of our work is a flexible algorithm that allows a formation of robots to follow an odour plume to its source using the observations collected in spatially distributed points in the environment.

Formations are a central component of our work, allowing us to position robots in a manner that guarantees meaningful observations. We made use of Laplacian feedback to drive terrestrial robots equipped with relative localisation sensors to an intended formation, and employed a similar approach with absolute positions to control marine surface robots. For underwater robots lacking positioning capabilities, we introduced an original formation control method based on acoustic ranges.

Along the way, we explored additional mechanisms and applications for our framework, highlighting its flexibility. In particular, our proposed 3D extension yielded some of the first published experimental results using a heterogeneous team of ground and flying robots for 3D plume tracing.

Our work combines aspects of robotics and control theory while keeping simplicity as a core tenet. Our methods are easy to understand, easy to implement, and light on requirements, both computational and sensory. Although often overlooked, we believe simplicity to be of the utmost importance to drive adoption in early stage technology, and a clear advantage of our solution.

Development work was accompanied by extensive experimental evaluation, involving land and water robots, as well as a proxy for a flying vehicle. We worked with three simulation platforms (MATLAB/Simulink, Webots, and ANSYS Fluent), three different robots (Khepera III, Khepera IV, and MEDUSA), and two principal test beds, in two countries, targeting utterly different environments. This experimental effort, however challenging and often frustrating, provided us with results that are validated using real robots, sensors and environments.

We conclude by summarising the contributions and outcomes of this thesis:

- We performed a detailed analysis of the newly released Khepera IV mobile robot and made available an open-source programming interface for the robot, the Khepera IV Toolbox.
- We proposed a theoretical approach to range-only triangular formation control that allows an AUV to follow two leaders and navigate in formation using only range measurements to each leader.
- We implemented a practical controller that realises this formation control approach while accounting for the constraints of ranging with acoustic modems.
- We carried out simulations and real robot experiments with the range-based formation controller using the MEDUSA AMVs in an outdoor setting.
- We designed a graph-based formation algorithm that allows a set of simple robots and behaviours to trace an odour plume to its source based on odour concentration and wind direction measurements.
- We assessed the performance of the algorithm in simulation and performed real-world experiments using Khepera robots and an ethanol plume in a wind tunnel.
- We extended the plume tracing algorithm to a 3D formation comprising ground and flying robots, and presented pioneering real-robot experimental results using Khepera IV robots and a 3D traversing system.
- We merged graph-based formation control and range-based formation control into an integrated solution for waterborne plume tracing using surface and underwater vehicles.
- We demonstrated the combined solution in simulations with the MEDUSA vehicle model and a surface freshwater plume generated with the aid of CFD software.

29 Outlook

We have shown compelling evidence for the value of simpler formation-based plume tracing approaches. Despite requiring additional robots, the increased efficiency translates to less time wasted tracing the plume, which in critical scenarios can mean the difference between life and death.

We foresee any immediate follow-up to our work going in the direction of more complex scenarios, with non-laminar wind flow and odour distribution asymmetries caused by obstacles in the field. Although the algorithm will undoubtedly require adjustments,the general framework for distributed sensing should only be more advantageous in scenarios with higher information density.

Still in the context of our solution, real-world experiments with marine vehicles in close formation would be an exciting feat. Additional investment into theory and modelling would be worthwhile and could provide estimates and bounds for the performance in different circumstances, as well as yield new ideas for further development.

Three-dimensional plume tracing is likely to become a hot topic in the upcoming years. A great number of applications have clear 3D requirements and provide great opportunities for research. In early experiments, we have shown that departing from a plane formulation and adding vertically-mobile robots can improve tracing performance. However, we have done so in a constrained scenario with a traversing system substituting for a flying robot. A critical aspect will be the impact, both positive and negative, of the strong air flow generated by real flying robots.

Finally, we look forward to seeing how new capabilities provided by remote sensors like TDLAS will influence future trends in plume tracing and help the technology move from research labs to the real world.

Glossary

ACO ant colony optimisation. 71

AHRS attitude and heading reference system. 23, 110

AMV autonomous marine vehicle. 7, 17, 23, 41, 126

API application programming interface. 18, 20

ASV autonomous surface vehicle. 109, 111, 115, 121

AUV autonomous underwater vehicle. 7, 17, 41, 43, 52, 60, 95, 105, 107, 109, 115, 116, 121, 126

CAN controller area network. 25

CFD computational fluid dynamics. 35, 121, 126

DISAL Distributed Intelligent Systems and Algorithms Laboratory. 17, 18, 20, 25, 27, 33, 69

DMA direct memory access. 27

DSOR Dynamical Systems and Ocean Robotics Laboratory. 23, 28, 52

DVL Doppler velocity log. 23, 110

EPFL École Polytechnique Fédérale de Lausanne. 25

GPS Global Positioning System. 23, 29, 60, 61, 110, 111, 121

HIL hardware-in-the-loop. 31, 60

IMU inertial measurement unit. 20, 23, 110

IST Instituto Superior Técnico. 17, 23

LED light-emitting diode. 22, 27

NTC negative temperature coefficient. 21

ODE Open Dynamics Engine. 33

OPC Open Platform Communications. 25

PI proportional-integral. 64

PID proportional-integral-derivative. 93

PLC programmable logic controller. 25

POMDP partially-observable Markov decision process. 107

PSO particle swarm optimisation. 5, 71

RTK real-time kinematic. 23, 29, 60, 110

SCADA supervisory control and data acquisition. 25

TDLAS tunable diode laser absorption spectroscopy. 71, 127

UAV unmanned aerial vehicle. 95

UUXO underwater unexploded ordinance. 105

VBAP virtual body and artificial potential. 43

VOC volatile organic compound. 22, 28, 110

Bibliography

[1] C. Bushdid, M. O. Magnasco, L. B. Vosshall and A. Keller, 'Humans can discriminate more than 1 trillion olfactory stimuli', *Science*, vol. 343, no. 6177, pp. 1370–1372, 2014. DOI: 10.1126/science.1249168 (cit. on p. 3).

[2] I. H. Cho and D. G. Peterson, 'Chemistry of bread aroma: a review', *Food Science and Biotechnology*, vol. 19, no. 3, pp. 575–582, 2010. DOI: 10.1007/s10068-010-0081-3 (cit. on p. 3).

[3] J. Clay, 'The smell of baking bread : an inexpensive means of improving the quality of life for long-term care residents?', MSc Thesis, University of British Columbia, 2011 (cit. on p. 3).

[4] C. S. Gulas and P. H. Bloch, 'Right under our noses: ambient scent and consumer responses', *Journal of Business and Psychology*, vol. 10, no. 1, pp. 87–98, 1995. DOI: 10.2307/25092493 (cit. on p. 3).

[5] N. Guéguen, 'The sweet smell of ... implicit helping: effects of pleasant ambient fragrance on spontaneous help in shopping malls.', *The Journal of Social Psychology*, vol. 152, no. 4, pp. 397–400, 2012. DOI: 10.1080/00224545.2011.630434 (cit. on p. 3).

[6] W. O. Reece, H. Erickson, J. P. Goff and E. E. Uemura, *Dukes' physiology of domestic animals.* Wiley-Blackwell, 2015 (cit. on p. 3).

[7] A. Thesen, J. B. Steen and K. B. Doving, 'Behaviour of dogs during olfactory tracking', *Journal of Experimental Biology*, vol. 180, no. 1, pp. 247–251, 1993 (cit. on p. 3).

[8] F. C. Craighead, 'Grizzly bear ranges and movement as determined by radiotracking', *Bears: Their Biology and Management*, vol. 3, p. 97, 1976. DOI: 10.2307/3872759 (cit. on p. 3).

[9] D. Pickel, G. P. Manucy, D. B. Walker, S. B. Hall and J. C. Walker, 'Evidence for canine olfactory detection of melanoma', *Applied Animal Behaviour Science*, vol. 89, no. 1-2, pp. 107–116, 2004. DOI: 10.1016/j.applanim.2004.04.008 (cit. on p. 3).

[10] J. Hartman, 'A possible method for the rapid estimation of flavours in vegetables', *Proceedings of the American Society for Horticultural Science*, vol. 64, pp. 335–342, 1954 (cit. on p. 3).

[11] R. W. Moncrieff, 'An instrument for measuring and classifying odors', *Journal of Applied Physiology*, vol. 16, no. 4, pp. 742–749, 1961 (cit. on p. 3).

[12] R. Rozas, J. Morales and D. Vega, 'Artificial smell detection for robotic navigation', in *International Conference on Advanced Robotics*, Pisa, Italy, 1991, pp. 1730–1733. DOI: 10.1109/ICAR.1991.240354 (cit. on p. 3).

[13] P. J. W. Roberts and D. R. Webster, *Turbulent diffusion.* ASCE Press, 2002 (cit. on p. 4).

[14] C. Distante, G. Indiveri and G. Reina, ‘An application of mobile robotics for olfactory monitoring of hazardous industrial sites’, *Industrial Robot: An International Journal*, vol. 36, no. 1, pp. 51–59, 2009. DOI: 10.1108/01439910910924675 (cit. on p. 4).

[15] H. Ishida, T. Nakamoto, T. Moriizumi, T. Kikas and J. Janata, ‘Plume-tracking robots: a new application of chemical sensors’, *Biological Bulletin*, vol. 200, no. 2, pp. 222–226, 2001 (cit. on p. 4).

[16] T. Lochmatter, ‘Bio-inspired and probabilistic algorithms for distributed odor source localization using mobile robots’, PhD Thesis 4628, École Polytechnique Fédérale de Lausanne, 2010. DOI: 10.5075/epfl-thesis-4628 (cit. on pp. 5, 21, 22, 33, 34, 71).

[17] S. A. Gowal, ‘A framework for graph-based distributed rendezvous of nonholonomic multi-robot systems’, PhD Thesis 5845, École Polytechnique Fédérale de Lausanne, 2013. DOI: 10.5075/epfl-thesis-5845 (cit. on pp. 6, 80).

[18] J. M. Soares, I. Navarro and A. Martinoli, ‘The Khepera IV mobile robot: performance evaluation, sensory data, and software toolbox’, in *Iberian Robotics Conference*, ser. Advances in Intelligent Systems and Computing, vol. 417, Springer, 2015, pp. 767–781. DOI: 10.1007/978-3-319-27146-0_59 (cit. on pp. 11, 18, 20, 140).

[19] J. M. Soares, A. P. Aguiar and A. M. Pascoal, ‘Triangular formation control using range measurements: an application to marine robotic vehicles’, in *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles*, Porto, Portugal, 2012, pp. 112–117. DOI: 10.3182/20120410-3-PT-4028.00020 (cit. on pp. 11, 41, 141).

[20] J. M. Soares, A. P. Aguiar, A. M. Pascoal and A. Martinoli, ‘Joint ASV/AUV range-based formation control: theory and experimental results’, in *IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, 2013, pp. 5579 –5585. DOI: 10.1109/ICRA.2013.6631378 (cit. on pp. 11, 41, 141).

[21] ——, ‘Design and implementation of a range-based formation controller for marine robots’, in *Iberian Robotics Conference*, ser. Advances in Intelligent Systems and Computing, vol. 252, Springer, 2013, pp. 55–67. DOI: 10.1007/978-3-319-03413-3 (cit. on pp. 11, 141).

[22] F. Rego, J. M. Soares, A. M. Pascoal, A. P. Aguiar and C. Jones, ‘Flexible triangular formation keeping of marine robotic vehicles using range measurements’, in *IFAC World Congress*, Cape Town, South Africa, 2014, pp. 5145–5150. DOI: 10.3182/20140824-6-ZA-1003.02435 (cit. on pp. 12, 64, 141).

[23] J. M. Soares, A. P. Aguiar, A. M. Pascoal and A. Martinoli, ‘A graph-based formation algorithm for odor plume tracing’, in *International Symposium on Distributed Autonomous Robotic Systems*, ser. Springer Tracts in Advanced Robotics, vol. 112, Springer, 2014, pp. 255–269. DOI: 10.1007/978-4-431-55879-8_18 (cit. on pp. 12, 69, 140).

[24] ——, ‘A distributed formation-based odor source localization algorithm: design, implementation, and wind tunnel evaluation’, in *IEEE International Conference on Robotics and Automation*, Seattle, WA, USA, 2015, pp. 1830–1836. DOI: 10.1109/ICRA.2015.7139436 (cit. on pp. 12, 69, 140).

[25] A. Prorok, A. Arfire, A. Bahr, J. R. Farserotu and A. Martinoli, ‘Indoor navigation research with the Khepera III mobile robot: an experimental baseline with a case-study on ultra-wideband positioning’, in *International Conference on Indoor Positioning and Indoor Navigation*, Zurich, Switzerland, 2010. DOI: 10.1109/IPIN.2010.5647880 (cit. on p. 17).

[26] T. Lochmatter, *Khepera III Toolbox*. Wikibooks, The Free Textbook Project, 2013 (cit. on p. 18).

[27] J. Pugh, X. Raemy, C. Favre, R. Falconi and A. Martinoli, 'A fast onboard relative positioning module for multirobot systems', *IEEE/ASME Transactions on Mechatronics*, vol. 14, no. 2, pp. 151–162, 2009. DOI: 10.1109/TMECH.2008.2011810 (cit. on p. 22).

[28] S. Gowal, A. Prorok and A. Martinoli, 'Two-phase online calibration for infrared-based inter-robot positioning modules', in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, CA, USA, 2011, pp. 3313–3319. DOI: 10.1109/IROS.2011.6048655 (cit. on p. 22).

[29] J. Ribeiro, 'Motion Control of Single and Multiple Autonomous Marine Vehicles', MSc Thesis, Instituto Superior Técnico - Technical University of Lisbon, 2011, p. 76 (cit. on pp. 23, 31, 32, 60).

[30] T. Lochmatter, P. Roduit, C. Cianci, N. Correll, J. Jacot and A. Martinoli, 'SwisTrack - a flexible open source tracking software for multi-agent systems', in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 4004–4010. DOI: 10.1109/IROS.2008.4650937 (cit. on p. 27).

[31] R. Tsai, 'A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses', *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 323–344, 1987. DOI: 10.1109/JRA.1987.1087109 (cit. on p. 27).

[32] J. A. Farrell, J. Murlis, X. Long, W. Li and R. T. Cardé, 'Filament-based atmospheric dispersion model to achieve short time-scale structure of odor plumes', *Environmental Fluid Mechanics*, vol. 2, no. 1-2, pp. 143–169, 2002. DOI: 10.1023/A:1016283702837 (cit. on p. 34).

[33] J. Desai, J. Ostrowski and V. Kumar, 'Controlling formations of multiple mobile robots', in *IEEE International Conference on Robotics and Automation*, vol. 4, Leuven, Belgium: IEEE, 1998, pp. 2864–2869. DOI: 10.1109/ROBOT.1998.680621 (cit. on p. 43).

[34] J. Desai, J. Ostrowski and V. Kumar, 'Modeling and control of formations of nonholonomic mobile robots', *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 905–908, 2001. DOI: 10.1109/70.976023 (cit. on p. 43).

[35] R. Falconi, S. Gowal and A. Martinoli, 'Graph based distributed control of non-holonomic vehicles endowed with local positioning information engaged in escorting missions', in *IEEE International Conference on Robotics and Automation*, Anchorage, AK, USA, 2010, pp. 3207–3214. DOI: 10.1109/ROBOT.2010.5509139 (cit. on pp. 43, 69, 76).

[36] M. Cao and A. S. Morse, 'Station keeping in the plane with range-only measurements', in *American Control Conference*, New York, NY, USA, 2007, pp. 5419–5424. DOI: 10.1109/ACC.2007.4282748 (cit. on p. 43).

[37] K.-K. Oh and H.-S. Ahn, 'Formation control of mobile agents based on inter-agent distance dynamics', *Automatica*, vol. 47, no. 10, pp. 2306–2312, 2011. DOI: 10.1016/j.automatica.2011.08.019 (cit. on p. 43).

[38] A. N. Bishop, 'Distributed bearing-only formation control with four agents and a weak control law', in *IEEE International Conference on Control and Automation*, Santiago, Chile, 2011, pp. 30–35. DOI: 10.1109/ICCA.2011.6138062 (cit. on p. 43).

[39] ——, 'A very relaxed control law for bearing-only triangular formation control', in *IFAC World Congress*, Milano, Italy, 2011, pp. 5991–5998. DOI: 10.3182/20110828-6-IT-1002.00105 (cit. on p. 43).

[40] M. Basiri, A. N. Bishop and P. Jensfelt, 'Distributed control of triangular formations with angle-only constraints', *Systems & Control Letters*, vol. 59, no. 2, pp. 147–154, 2010. DOI: 10.1016/j.sysconle.2009.12.010 (cit. on p. 43).

[41] M. Cao, C. Yu and B. D. O. Anderson, 'Formation control using range-only measurements', *Automatica*, vol. 47, no. 4, pp. 776–781, 2011. DOI: 10.1016/j.automatica.2011.01.067 (cit. on p. 43).

[42] H. Yang and F. Zhang, 'Geometric formation control for autonomous underwater vehicles', in *IEEE International Conference on Robotics and Automation*, Anchorage, AK, USA, 2010, pp. 4288–4293. DOI: 10.1109/ROBOT.2010.5509748 (cit. on p. 43).

[43] R. Ghabcheloo, A. P. Aguiar, A. M. Pascoal, C. Silvestre, I. Kaminer and J. Hespanha, 'Coordinated path-following in the presence of communication losses and time delays', *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 234–265, 2009. DOI: 10.1137/060678993 (cit. on p. 43).

[44] X. Xiang, B. Jouvencel and O. Parodi, 'Coordinated formation control of multiple autonomous underwater vehicles for pipeline inspection', *International Journal of Advanced Robotic Systems*, vol. 7, no. 1, pp. 75–84, 2010. DOI: 10.5772/7242 (cit. on p. 43).

[45] E. Fiorelli, N. E. Leonard, P. Bhatta, D. A. Paley, R. Bachmayer and D. M. Fratantoni, 'Multi-AUV control and adaptive sampling in Monterey Bay', *IEEE Journal of Oceanic Engineering*, vol. 31, no. 4, pp. 935–948, 2006. DOI: 10.1109/JOE.2006.880429 (cit. on p. 43).

[46] P. Ögren, E. Fiorelli and N. E. Leonard, 'Cooperative control of mobile sensor networks: adaptive gradient climbing in a distributed environment', *IEEE Transactions on Automatic Control*, vol. 49, no. 8, pp. 1292–1302, 2004. DOI: 10.1109/TAC.2004.832203 (cit. on p. 43).

[47] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010 (cit. on pp. 69, 76).

[48] G. Kowadlo and R. Russell, 'Robot odor localization: a taxonomy and survey', *The International Journal of Robotics Research*, vol. 27, no. 8, pp. 869–894, 2008. DOI: 10.1177/0278364908095118 (cit. on p. 71).

[49] A. Dhariwal, G. Sukhatme and A. Requicha, 'Bacterium-inspired robots for environmental monitoring', in *IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2004, pp. 1436–1443. DOI: 10.1109/ROBOT.2004.1308026 (cit. on p. 71).

[50] L. Marques, U. Nunes and A. T. de Almeida, 'Olfaction-based mobile robot navigation', *Thin Solid Films*, vol. 418, no. 1, pp. 51–58, 2002. DOI: 10.1016/S0040-6090(02)00593-X (cit. on p. 71).

[51] A. Lilienthal and T. Duckett, 'Experimental analysis of gas-sensitive Braitenberg vehicles', *Advanced Robotics*, vol. 18, no. 8, pp. 817–834, 2004. DOI: 10.1163/1568553041738103 (cit. on p. 71).

[52] A. Hayes, A. Martinoli and R. Goodman, 'Distributed odor source localization', *IEEE Sensors Journal*, vol. 2, no. 3, pp. 260–271, 2002. DOI: 10.1109/JSEN.2002.800682 (cit. on p. 71).

[53] A. Khalili, A. Rastegarnia, M. K. Islam and Z. Yang, 'A bio-inspired cooperative algorithm for distributed source localization with mobile nodes.', in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Osaka, Japan, 2013, pp. 3515–3518. DOI: 10.1109/EMBC.2013.6610300 (cit. on p. 71).

[54] V. Genovese, P. Dario, R. Magni and L. Odetti, 'Self organizing behavior and swarm intelligence in a pack of mobile miniature robots in search of pollutants', in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, Raleigh, NC, USA, 1992, pp. 1575–1582. DOI: 10.1109/IROS.1992.594225 (cit. on p. 71).

[55] A. Marjovi and L. Marques, 'Optimal swarm formation for odor plume finding', *IEEE Transactions on Cybernetics*, vol. 99, no. 12, pp. 2302–2315, 2014. DOI: 10.1109/TCYB.2014.2306291 (cit. on pp. 71, 91).

[56] T. Lochmatter, E. Göl, I. Navarro and A. Martinoli, 'A plume tracking algorithm based on crosswind formations', in *International Symposium on Distributed Autonomous Robotic Systems*, ser. Springer Tracts in Advanced Robotics, vol. 83, Springer, 2013, pp. 91–102. DOI: 10.1007/978-3-642-32723-0_7 (cit. on p. 71).

[57] M. Vergassola, E. Villermaux and B. I. Shraiman, '"Infotaxis" as a strategy for searching without gradients', *Nature*, vol. 445, no. 7126, pp. 406–9, 2007. DOI: 10.1038/nature05464 (cit. on p. 71).

[58] J.-G. Li, Q.-H. Meng, Y. Wang and M. Zeng, 'Odor source localization using a mobile robot in outdoor airflow environments with a particle filter algorithm', *Autonomous Robots*, vol. 30, no. 3, pp. 281–292, 2011. DOI: 10.1007/s10514-011-9219-2 (cit. on p. 71).

[59] J.-G. Li, J. Yang, J. Liu, G.-D. Lu and L. Yang, 'Odor-source searching using a mobile robot in time-variant airflow environments with obstacles', in *Chinese Control Conference*, Nanjing, China, 2014, pp. 8559–8564. DOI: 10.1109/ChiCC.2014.6896437 (cit. on p. 71).

[60] G. Cabrita, L. Marques and V. Gazi, 'Virtual cancelation plume for multiple odor source localization', in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tokyo, Japan, 2013, pp. 5552–5558. DOI: 10.1109/IROS.2013.6697161 (cit. on p. 71).

[61] W. Jatmiko, K. Sekiyama and T. Fukuda, 'A PSO-based mobile robot for odor source localization in dynamic advection-diffusion with obstacles environment: theory, simulation and measurement', *IEEE Computational Intelligence Magazine*, vol. 2, no. 2, pp. 37–51, 2007. DOI: 10.1109/MCI.2007.353419 (cit. on p. 71).

[62] L. Marques, U. Nunes and A. T. Almeida, 'Particle swarm-based olfactory guided search', *Autonomous Robots*, vol. 20, no. 3, pp. 277–287, 2006. DOI: 10.1007/s10514-006-7567-0 (cit. on p. 71).

[63] M.-L. Cao, Q.-H. Meng, X.-W. Wang, B. Luo, M. Zeng and W. Li, 'Localization of multiple odor sources via selective olfaction and adapted ant colony optimization algorithm', in *IEEE International Conference on Robotics and Biomimetics*, Shenzhen, China, 2013, pp. 1222–1227. DOI: 10.1109/ROBIO.2013.6739631 (cit. on p. 71).

[64] G. de Croon, L. O'Connor, C. Nicol and D. Izzo, 'Evolutionary robotics approach to odor source localization', *Neurocomputing*, vol. 121, pp. 481–497, 2013. DOI: 10.1016/j.neucom.2013.05.028 (cit. on p. 71).

[65] V. M. H. Bennetts, A. J. Lilienthal, A. A. Khaliq, V. P. Sese and M. Trincavelli, 'Towards real-world gas distribution mapping and leak localization using a mobile robot with 3D and remote gas sensing capabilities', in *IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, 2013, pp. 2335–2340. DOI: 10.1109/ICRA.2013.6630893 (cit. on p. 71).

[66] V. H. Bennetts, E. Schaffernicht, T. Stoyanov, A. J. Lilienthal and M. Trincavelli, 'Robot assisted gas tomography: localizing methane leaks in outdoor environments', in *IEEE International Conference on Robotics and Automation*, Hong Kong, China, 2014, pp. 6362–6367. DOI: 10.1109/ICRA.2014.6907798 (cit. on p. 71).

[67] T. Lochmatter, N. Heiniger and A. Martinoli, 'Localizing an odor source and avoiding obstacles: experiments in a wind tunnel using real robots', in *International Symposium on Olfaction and Electronic Nose*, ser. AIP Conference Proceedings, vol. 1137, AIP, 2009, pp. 69–72. DOI: 10.1063/1.3156629 (cit. on p. 80).

[68] V. Braitenberg, *Vehicles: experiments in synthetic psychology*, ser. Bradford Books. MIT Press, 1986 (cit. on p. 93).

[69] T. Shroder and J. Konrad, *Fire on the horizon: the untold story of the gulf oil disaster*. HarperCollins, 2011 (cit. on p. 105).

[70] M. K. McNutt, R. Camilli, T. J. Crone, G. D. Guthrie, P. A. Hsieh, T. B. Ryerson, O. Savas and F. Shaffer, 'Review of flow rate estimates of the Deepwater Horizon oil spill', *Proceedings of the National Academy of Sciences of the United States of America*, vol. 109, no. 50, pp. 20 260–20 267, 2012. DOI: 10.1073/pnas.1112139108 (cit. on p. 105).

[71] R. Camilli, C. M. Reddy, D. R. Yoerger, B. A. S. Van Mooy, M. V. Jakuba, J. C. Kinsey, C. P. McIntyre, S. P. Sylva and J. V. Maloney, 'Tracking hydrocarbon plume transport and biodegradation at Deepwater Horizon.', *Science*, vol. 330, no. 6001, pp. 201–4, 2010. DOI: 10.1126/science.1195223 (cit. on p. 105).

[72] T. Clem, 'Sensor technologies for hunting buried sea mines', in *MTS/IEEE Oceans*, vol. 1, Biloxi, MS, USA, 2002, pp. 452–460. DOI: 10.1109/OCEANS.2002.1193312 (cit. on p. 105).

[73] S. A. Thorpe, *An introduction to ocean turbulence*. Cambridge University Press, 2007 (cit. on p. 105).

[74] D. A. Fong and W. R. Geyer, 'The alongshore transport of freshwater in a surface-trapped river plume', *Journal of Physical Oceanography*, vol. 32, no. 3, pp. 957–972, 2002. DOI: 10.1175/1520-0485(2002)032<0957:TATOFI>2.0.CO;2 (cit. on p. 105).

[75] P. A. Moore, N. Scholz and J. Atema, 'Chemical orientation of lobsters, homarus americanus, in turbulent odor plumes.', *Journal of Chemical Ecology*, vol. 17, no. 7, pp. 1293–307, 1991. DOI: 10.1007/BF00983763 (cit. on p. 107).

[76] J. Atema, 'Eddy chemotaxis and odor landscapes: exploration of nature with animal sensors', *Biological Bulletin*, vol. 191, no. 1, p. 129, 1996. DOI: 10.2307/1543074 (cit. on p. 107).

[77] T. Consi, J. Atema, C. Goudey, J. Cho and C. Chryssostomidis, 'AUV guidance with chemical signals', in *IEEE Symposium on Autonomous Underwater Vehicle Technology*, Cambridge, MA, USA, 1994, pp. 450–455. DOI: 10.1109/AUV.1994.518659 (cit. on p. 107).

[78] F. W. Grasso, T. R. Consi, D. C. Mountain and J. Atema, 'Biomimetic robot lobster performs chemo-orientation in turbulence using a pair of spatially separated sensors: Progress and challenges', *Robotics and Autonomous Systems*, vol. 30, no. 1-2, pp. 115–131, 2000. DOI: 10.1016/S0921-8890(99)00068-8 (cit. on p. 107).

[79] T. Moriizumi and H. Ishida, 'Robotic systems to track chemical plumes', in *Conference on Optoelectronic and Microelectronic Materials and Devices*, Sydney, Australia, 2002, pp. 537–540. DOI: 10.1109/COMMAD.2002.1237308 (cit. on p. 107).

[80] J. Farrell and R. Arrieta, 'Biologically inspired chemical plume tracing on an autonomous underwater vehicle', in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 7, The Hague, The Netherlands, 2004, pp. 5991–5996. DOI: 10.1109/ICSMC.2004.1401337 (cit. on p. 107).

[81] J. Farrell, S. Pang and W. Li, 'Chemical Plume Tracing via an Autonomous Underwater Vehicle', *IEEE Journal of Oceanic Engineering*, vol. 30, no. 2, pp. 428–442, 2005. DOI: 10.1109/JOE.2004.838066 (cit. on p. 107).

[82] J. Farrell, R. Arrieta, W. Li and S. Pang, 'Moth-inspired chemical plume tracing on an autonomous underwater vehicle', *IEEE Transactions on Robotics*, vol. 22, no. 2, pp. 292–307, 2006. DOI: 10.1109/TRO.2006.870627 (cit. on p. 107).

[83] B. Fletcher, ‘Chemical plume mapping with an autonomous underwater vehicle’, in *MTS/IEEE Oceans*, vol. 1, Honolulu, HI, USA, 2001, pp. 508–512. DOI: 10.1109/OCEANS.2001.968774 (cit. on p. 107).

[84] G. Ferri, M. V. Jakuba and D. R. Yoerger, ‘A novel method for hydrothermal vents prospecting using an autonomous underwater robot’, in *IEEE International Conference on Robotics and Automation*, Pasadena, CA, USA, 2008, pp. 1055–1060. DOI: 10.1109/ROBOT.2008.4543344 (cit. on p. 107).

[85] ——, ‘A novel trigger-based method for hydrothermal vents prospecting using an autonomous underwater robot’, *Autonomous Robots*, vol. 29, no. 1, pp. 67–83, 2010. DOI: 10.1007/s10514-010-9187-y (cit. on p. 107).

[86] M. V. Jakuba and D. R. Yoerger, ‘Autonomous search for hydrothermal vent fields with occupancy grid maps’, in *Australasian Conference on Robotics and Automation*, Canberra, Australia, 2008 (cit. on p. 107).

[87] Z. A. Saigol, ‘Automated planning for hydrothermal vent prospecting using AUVs’, PhD Thesis, University of Birmingham, 2011 (cit. on p. 107).

[88] S. Pang, ‘Plume source localization for AUV based autonomous hydrothermal vent discovery’, in *MTS/IEEE Oceans*, Seattle, WA, USA, 2010. DOI: 10.1109/OCEANS.2010.5664516 (cit. on p. 107).

[89] R. H. Byrne, S. E. Eskridge, J. E. Hurtado and E. L. Savage, ‘Algorithms and analysis for underwater vehicle plume tracing’, Sandia National Laboratories, SAND2003-2643, 2003 (cit. on p. 107).

[90] J. Hurtado, D. Adkins, S. Eskridge, J. Harrington, E. Heller and R. Byrne, ‘Miniature mobile robots for plume tracking and source localization research’, *Journal of Micromechatronics*, vol. 1, no. 3, pp. 253–261, 2001. DOI: 10.1163/156856301760132141 (cit. on p. 107).

[91] F. E. Schneider and D. Wildermuth, ‘From laboratory into real life: the EURATHLON and ELROB disaster response robotics competitions’, in *International Carpathian Control Conference*, Szilvasvarad, Hungary, 2015, pp. 452–457. DOI: 10.1109/CarpathianCC.2015.7145122 (cit. on p. 107).

[92] T. Ohnishi, ‘The disaster at Japan’s Fukushima-Daiichi nuclear power plant after the March 11 2011 earthquake and tsunami, and the resulting spread of radioisotope contamination’, *Radiation Research*, vol. 177, no. 1, 2012 (cit. on p. 107).

[93] J. M. Soares, M. Franceschinis, R. M. Rocha, W. Zhang and M. A. Spirito, ‘Opportunistic data collection in sparse wireless sensor networks’, *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, no. 401802, 2011. DOI: 10.1155/2011/401802 (cit. on p. 141).

[94] J. M. Soares, B. J. Goncalves and R. M. Rocha, ‘Practical issues in the development of a minimalistic power management solution for WSNs’, *International Journal of Sensor Networks*, vol. 8, no. 3/4, pp. 136–146, 2010. DOI: 10.1504/IJSNET.2010.036189 (cit. on p. 141).

[95] J. M. Soares, W. Zhang, M. Franceschinis, M. A. Spirito and R. M. Rocha, ‘Experimental and analytical performance evaluation of a real opportunistic wireless sensor network’, in *Future Network and Mobile Summit*, Florence, Italy, 2010 (cit. on p. 141).

[96] J. M. Soares, ‘CHARON: convergent hybrid-replication approach to routing in opportunistic networks – efficient collection routing for low-density mobile WSNs’, MSc Thesis, Instituto Superior Técnico - Technical University of Lisbon, 2009 (cit. on p. 141).

[97] J. M. Soares and R. M. Rocha, ‘CHARON: routing in low-density opportunistic wireless sensor networks’, in *IFIP Wireless Days*, Paris, France, 2009. DOI: 10.1109/WD.2009.5449666 (cit. on p. 141).

[98] J. M. Soares, B. Goncalves and R. M. Rocha, ‘Power management extensions for Tagus-SensorNet’, in *International Conference on Computer Communications and Networks*, San Francisco, CA, USA, 2009. DOI: 10.1109/ICCCN.2009.5235377 (cit. on p. 141).

Curriculum vitae

Jorge Miguel Soares

mail@jorgesoares.org

Education

2010–2016	PhD in Robotics, Control, and Intelligent Systems *École Polytechnique Fédérale de Lausanne (EPFL)* PhD in Electrical and Computer Engineering *Instituto Superior Técnico - University of Lisbon (IST-UL)*
2007–2009	MSc in Communication Networks Engineering *Instituto Superior Técnico - University of Lisbon (IST-UL)*
2004–2007	BSc in Communication Networks Engineering *Instituto Superior Técnico - University of Lisbon (IST-UL)*

Professional experience

10.2015–12.2015	Research Fellow *Faculty of Engineering, University of Porto*
06.2015–08.2015	Research Fellow *Dynamical Systems and Ocean Robotics Laboratory, IST-UL*
09.2012–05.2015	Doctoral Assistant *Distributed Intelligent Systems and Algorithms Laboratory, EPFL*
09.2010–08.2014	Doctoral Fellow *Fundação para a Ciência e Tecnologia*
09.2008-02.2009	Graduate Teaching Assistant *Department of Electrical and Computer Engineering, IST-UL*
07.2007-05.2008	Undergraduate Research Fellow *Data Management and Information Retrieval Group, INESC-ID*

Non-profit experience

01.2016–	Ad-Hoc Committee on Disruptive Innovation, Member
01.2015–	Nominations and Appointments Committee, Member
01.2015–	Region 8 Awards and Recognition Committee, Member
01.2015–12.2015	Ad-Hoc Committee on Digital Strategy, Member
01.2015–12.2015	History Committee, Member
01.2013–12.2014	MGA Nominations and Appointments Committee, Member
01.2013–12.2014	Region 8 Strategic Planning Committee, Member
04.2011–12.2013	Portugal Section, Vice-Chair
01.2012–12.2012	MGA SAC RSR Steering Committee, Chair
01.2011–12.2012	Region 8, Student Representative
11.2007–11.2009	IST Student Branch, Chair
	IEEE
09.2006–11.2009	Vice-President *NEERCI-IST*

Honours and awards

2015	Ordem dos Engenheiros, Best Electrical Engineering Internship
2011	IEEE Portugal Section, Outstanding Student Volunteer Award
2009	IST DEEC, Best Student Award
2008	IST, Merit Diploma
2006	IST, Merit Diploma
2005	IST, Merit Diploma

Publications

1. J. M. Soares, I. Navarro and A. Martinoli, ‘The Khepera IV mobile robot: performance evaluation, sensory data, and software toolbox’, in *Iberian Robotics Conference*, ser. Advances in Intelligent Systems and Computing, vol. 417, Springer, 2015, pp. 767–781. DOI: 10.1007/978-3-319-27146-0_59.
2. J. M. Soares, A. P. Aguiar, A. M. Pascoal and A. Martinoli, ‘A distributed formation-based odor source localization algorithm: design, implementation, and wind tunnel evaluation’, in *IEEE International Conference on Robotics and Automation*, Seattle, WA, USA, 2015, pp. 1830–1836. DOI: 10.1109/ICRA.2015.7139436.
3. J. M. Soares, A. P. Aguiar, A. M. Pascoal and A. Martinoli, ‘A graph-based formation algorithm for odor plume tracing’, in *International Symposium on Distributed Autonomous Robotic Systems*, ser. Springer Tracts in Advanced Robotics, vol. 112, Springer, 2014, pp. 255–269. DOI: 10.1007/978-4-431-55879-8_18.

4. F. Rego, J. M. Soares, A. M. Pascoal, A. P. Aguiar and C. Jones, ‘Flexible triangular formation keeping of marine robotic vehicles using range measurements’, in *IFAC World Congress*, Cape Town, South Africa, 2014, pp. 5145–5150. DOI: 10.3182/20140824-6-ZA-1003.02435.
5. J. M. Soares, A. P. Aguiar, A. M. Pascoal and A. Martinoli, ‘Design and implementation of a range-based formation controller for marine robots’, in *Iberian Robotics Conference*, ser. Advances in Intelligent Systems and Computing, vol. 252, Springer, 2013, pp. 55–67. DOI: 10.1007/978-3-319-03413-3.
6. J. M. Soares, A. P. Aguiar, A. M. Pascoal and A. Martinoli, ‘Joint ASV/AUV range-based formation control: theory and experimental results’, in *IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, 2013, pp. 5579 –5585. DOI: 10.1109/ICRA.2013.6631378.
7. J. M. Soares, A. P. Aguiar and A. M. Pascoal, ‘Triangular formation control using range measurements: an application to marine robotic vehicles’, in *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles*, Porto, Portugal, 2012, pp. 112–117. DOI: 10.3182/20120410-3-PT-4028.00020.
8. J. M. Soares, M. Franceschinis, R. M. Rocha, W. Zhang and M. A. Spirito, ‘Opportunistic data collection in sparse wireless sensor networks’, *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, no. 401802, 2011. DOI: 10.1155/2011/401802.
9. J. M. Soares, B. J. Goncalves and R. M. Rocha, ‘Practical issues in the development of a minimalistic power management solution for WSNs’, *International Journal of Sensor Networks*, vol. 8, no. 3/4, pp. 136–146, 2010. DOI: 10.1504/IJSNET.2010.036189.
10. J. M. Soares, W. Zhang, M. Franceschinis, M. A. Spirito and R. M. Rocha, ‘Experimental and analytical performance evaluation of a real opportunistic wireless sensor network’, in *Future Network and Mobile Summit*, Florence, Italy, 2010.
11. J. M. Soares, ‘CHARON: convergent hybrid-replication approach to routing in opportunistic networks – efficient collection routing for low-density mobile WSNs’, MSc Thesis, Instituto Superior Técnico - Technical University of Lisbon, 2009.
12. J. M. Soares and R. M. Rocha, ‘CHARON: routing in low-density opportunistic wireless sensor networks’, in *IFIP Wireless Days*, Paris, France, 2009. DOI: 10.1109/WD.2009.5449666.
13. J. M. Soares, B. Goncalves and R. M. Rocha, ‘Power management extensions for Tagus-SensorNet’, in *International Conference on Computer Communications and Networks*, San Francisco, CA, USA, 2009. DOI: 10.1109/ICCCN.2009.5235377.