HAL open science

Bio-inspired collective exploration and cultural organisation

Nicolas Cambier

▶ To cite this version:

Nicolas Cambier. Bio-inspired collective exploration and cultural organisation. Robotics [cs.RO]. Université de Technologie de Compiègne, 2019. English. NNT : 2019COMP2511 . tel-02501072

HAL Id: tel-02501072

https://theses.hal.science/tel-02501072v1

Submitted on 6 Mar 2020

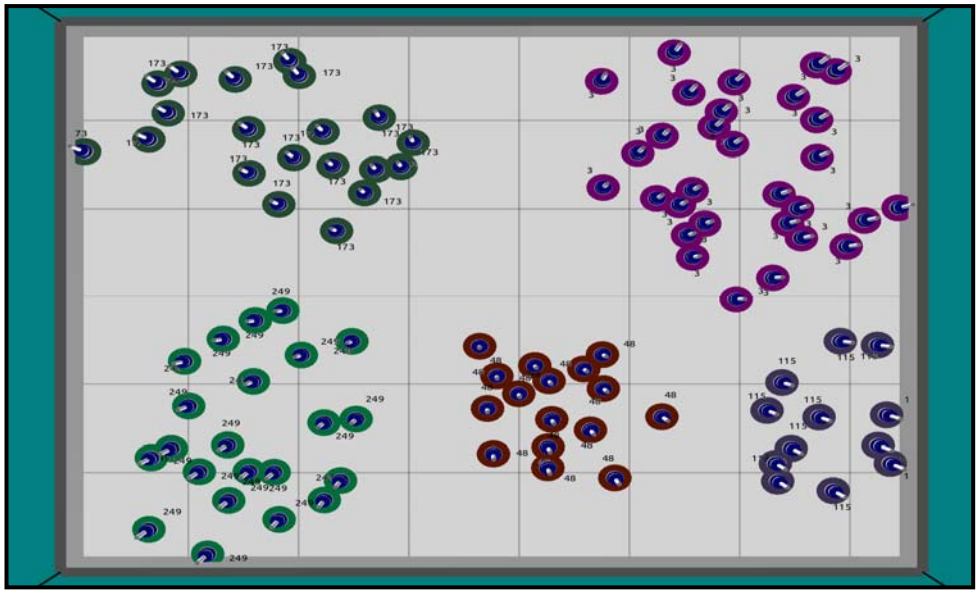
HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Par **Nicolas CAMBIER**

Bio-inspired collective exploration and cultural organisation

Thèse présentée
pour l'obtention du grade
de Docteur de l'UTC



Soutenue le 23 octobre 2019
Spécialité : Informatique : Unité de recherche Heudyasic (UMR-7253)
D2511

UNIVERSITÉ DE TECHNOLOGIE DE COMPIÈGNE

THÈSE

Présentée pour l'obtention du grade de
Docteur de l'Université de Technologie de Compiègne

Spécialité: Informatique
Unité de Recherche: Heudiasyc – UMR CNRS 7253

Par

Nicolas Cambier

Bio-inspired Collective Exploration and Cultural Organisation

23 octobre 2019

Soutenu devant le jury composé de:

M. **EIBEN Gusz**, Full Professor, Vrije Universiteit Amsterdam
M. **SIMONIN Olivier**, Professeur des Universités, CITI-Inria, INSA de Lyon
Mme. **PROROK Amanda**, Assistant Professor, University of Cambridge, Cambridge
M. **BREDECHE Nicolas**, Professeur des Universités, ISIR, Sorbonne Université, Paris
M. **DESTERCKE Sébastien**, chercheur CNRS, université de technologie de Compiègne, Heudiasyc, UMR CNRS 7253
M. **FERRANTE Eliseo**, Lecturer, PhD, University of Birmingham, Birmingham
M. **FRÉMONT Vincent**, Professeur des Universités, S2N UMR CNRS 6004, Ecole Centrale de Nantes, Nantes

(Invité) M. **TRIANNI Vito**, CNR Tenured Researcher, ISTC-CNR, Roma

Acknowledgements

First, I would like to thank Eliseo who closely supervised this thesis despite the distance (which I sincerely appreciate). Working with him has been absolutely great. He has trusted me from the moment we met and knowing that has really allowed me to take this work in the direction I wanted. He has been a terrific support during the course of this thesis and I learned a lot by working with him. Many thanks also to Vincent, my other supervisor who, again, fully trusted me and allowed me to make this work what it is. He also gave me plenty of insights into the world of academia, which are aiding me on a daily basis.

I also have to thank Vito, who I worked with quite a lot in the course of this thesis. Working with him has been extremely demanding but also extremely rewarding as I improved tremendously, in my writing and in my research. Thank you also for the two weeks in Rome, which were a great (although exhausting) end to my thesis. I am also thanking Roman, which is slightly odd, as we only started working together in the week before I handed out my manuscript. Nevertheless, I think that the conversations we had in the following couple of months helped me to clarify many of my ideas which allowed my defence to be complimentary to my thesis instead of a mere summary. Working with him has also just been really fun!

I would also like to thank the reviewers of my manuscript: Prof. Eiben and Prof. Simonin. Their questions and remarks also contributed to a very satisfying (at least to me) presentation. These thanks also go the other members of my jury: Dr. Prorok, Prof. Bredeche and Dr. Sébastien Destercke. The latter also read a couple of my reports during the course of my thesis (as mandated by the lab) and I must also thank him for that as some of the questions he raised following these readings actually kick-started the thought-process that lead me to a few important ideas in this thesis.

Thanks also to the people at Heudiasyc for the discussions we had. I will not cite everybody to save space but I want to mention Ariane and Manel especially. Many thanks also to the administrative staff of the lab who are all incredibly good at their jobs. I should especially mention Bérengère, Nathalie and Laurie as I actually gave them quite a lot of work. But it was just really nice to come to their office!

I am very thankful to my friends who maintained a communication channel with me despite the distance and the occasional radio silences. Maxime, Léni, Thérèse, thank you for that. Hadrien; we're both trying. Many thanks to my parents for all they gave to me and for their support.

Finally, a huge thank you to Gwendoline, for her support, her insights and for everything we shared in those three years (and more).

Résumé

Les systèmes autonomes sont récemment devenus une solution efficace pour des applications telles que l'exploration et la surveillance d'environnements. Dans ces situations, l'utilisation de plusieurs robots pourrait améliorer l'efficacité des solutions proposées, bien que cela nécessite des stratégies d'organisation qui soient à la fois robuste, flexible et adaptables à la taille de la flotte de robots. En robotique en essaim, ces qualités sont assurées par la décentralisation, la redondance (plusieurs/tous les robots effectuent la même tâche), des interactions locales et des règles simples. Les interactions et communications locales sont une composante clef de la robotique en essaim. Jusqu'ici, la communication n'a été utilisée que pour des tâches relativement simple, tels que signaler les préférences ou l'état d'un robot. Cependant, la communication peut être bien plus riche et similaire aux langages humains. Dans ces conditions, elle permettrait aux essaims de robots de gérer de nouvelles situations qui ne seraient pas prévues par leurs concepteurs. De riches communications sont donc nécessaires pour obtenir des essaims entièrement autonomes, en particulier dans des environnements inconnus.

Dans cette thèse, nous proposons une approche pour faire émerger des communications riches dans des essaims de robots en utilisant les jeux de langages comme protocole de communication et l'agrégation probabiliste comme cas d'étude. L'agrégation probabiliste est un prérequis pour de nombreuses tâches en robotique en essaim mais elle est aussi extrêmement sensible aux conditions expérimentales. Elle requiert donc un réglage spécifique de ses paramètres pour chaque nouvelle condition, y compris les changements d'échelle ou de densité.

Avec notre approche, nous avons observé que l'exécution simultanée du jeu de nommage et de l'agrégation mène, dans certaines conditions, à un nouveau comportement d'agglomération en plusieurs groupes, chacun avec son propre nom, qui est contrôlable via les paramètres de l'agrégation. En poussant ces interactions plus loin, nous démontrons que les dynamiques sociales du jeu de nommage peuvent sélectionner des paramètres d'agrégation efficaces. Cette sélection culturelle crée donc des contrôleurs résilients, qui évoluent en-ligne en fonction du contexte courant.

Abstract

Automatically-controlled artificial systems have recently been used in numerous settings including environmental monitoring and explorations, with great success. In such cases, the use of multiple robots could increase efficiency, although we should ensure that their communication and organisation strategies are robust, flexible, and scalable. These qualities can be ensured through decentralisation, redundancy (many/all robots perform the same task), local interaction, and simplistic rules, as is the case in swarm robotics. One of the key components of swarm robotics is local interaction or communication. The later has, so far, only been used for relatively simple tasks such as signalling a robot's preference or state. However, communication has more potential because the emergence of meaning, as it exists in human language, could allow robots swarms to tackle novel situations in ways that may not be a priori obvious to the experimenter. This is a necessary feature for having swarms that are fully autonomous, especially in unknown environments.

In this thesis, we propose a framework for the emergence of meaningful communications in swarm robotics using language games as a communication protocol and probabilistic aggregation as a case study. Probabilistic aggregation can be a prerequisite to many other swarm behaviours but, unfortunately, it is extremely sensitive to experimental conditions, and thus requires specific parameter tuning for any setting such as population size or density.

With our framework, we show that the concurrent execution of the naming game and of probabilistic aggregation leads, in certain conditions, to a new clustering and labelling behaviour that is controllable via the parameters of the aggregation controller. Pushing this interplay forward, we demonstrate that the social dynamics of the naming game can select efficient aggregation parameters through environmental pressure. This creates resilient controllers as the aggregation behaviour is dynamically evolved online according to the current environmental setting.

Contents

1 Introduction **1**

2 Swarm Robotics: State of the Art **5**
2.1 Design Methodologies 5
2.1.1 Behaviour-based Design 6
2.1.2 Automatic Design 7
2.2 Signalling 10
2.2.1 Stigmergy 11
2.2.2 Direct Interaction 12
2.2.3 Direct Communication 13
2.3 Self-Organised Aggregation 14
2.3.1 Aggregation with Shelters 14
2.3.2 Aggregation without Shelters 15
2.4 Applications to Collective Exploration 17
2.4.1 Area Coverage 18
2.4.2 Simultaneous Localisation and Mapping 19

3 Evolution of Language: State of the Art **25**
3.1 Biological Evolution 26
3.1.1 The Biolinguistic Paradigm 26
3.1.2 Synthetic Models 28
3.1.3 Applications to Artificial Systems 30
3.2 Cultural Evolution 33
3.2.1 Cultural Evolution of Language Universals 34
3.2.2 Origins of Language Games 36
3.2.3 The Naming Game 38

4 Methodology **43**
4.1 Parametric Aggregation 43
4.1.1 Linear Transition Function 45
4.1.2 Exponential Transition Functions 46
4.2 Integration of the Hearer-Only Minimal Naming Game 47
4.2.1 Bidirectional Interaction 48
4.2.2 Circular Interaction 49

5 Group-size Regulation in Self-Organised Aggregation through the Naming Game **55**
5.1 Linear Transitions 56
5.1.1 Experimental Setup 56
5.1.2 Results 58
5.2 Exponential Transitions 63
5.2.1 Experimental Setup 63
5.2.2 Results 64
5.3 Discussion 65

6 Dynamics of Culturally Evolving Probabilistic Aggregation **68**
6.1 Preliminary experiments 69
6.1.1 Experimental Setup 69
6.1.2 Results 70
6.2 In-depth study 74
6.2.1 Experimental Setup 74
6.2.2 Results 75
6.3 Discussion 79

7 Conclusion **82**

List of Figures

2.1 Example of collective chain formation. *From [Nouyan et al., 2009], reproduced with permission ©2009 IEEE.* 12
2.2 Example of a Voronoi diagram. Each region V_i has a centre point p_i. Any point in V_i is closer to p_i than to any other centre point. 19
2.3 Paths of agents following a noisy random walk (a) and passed through a Kalman filter (b) (colour schemes are random). 22

3.1 Crothers [1978]'s hierarchy of vowel systems. 35
3.2 Example of the AIBO game dialog between robot and a human. The robot performs image segmentation and tries to associate words spoken to the human (i.e. "ball") to the actual ball. For the interaction to be successful, the robot first has to be put in the right conditions by specific keywords. *Reproduced from [Steels, 2001] ©2001 IEEE.* . 39

4.1 (a) Base PFSM of our aggregation controller. (b) PFSM of our aggregation controller with linear transition functions. 44
4.2 Plotting of P_{join} and p_{Leave} according to (resp.) Eq. 4.2 and 4.3 and parametrised with the values for ϵ, ρ, a and b that best fit the values from Table 4.1 [Correll and Martinoli, 2007] (circles). 47
4.3 Positive feedback-loop 51

5.1 A picture of the ARGoS simulator in the course of an experiment with realistic footbots. The simulator can display extraneous information such as ongoing communications. 57

5.2 Mean (dot) and standard deviation (bars) of the quantity of aggregates in vanilla/NG aggregation (resp. left/middle) and of words (right) in stabilised swarms of 20/100 (resp. top/bottom) robots or after 100 000 seconds. These charts show that the Vanilla algorithm displays a normal aggregation behaviour as, with appropriate p_{Base} [Soysal and Sahin, 2005], the robots consistently gather themselves in a single aggregate. Moreover, we observe a visual correlation between the final quantity of aggregates and the final quantity of words in the populations playing a MNG. Finally, in cases where the swarm does not stabilises (0 aggregates), the whole population still converges on one word. 60
5.3 Examples of stabilised swarms with a $N = 100$ and p_{Base} configured as (resp.) 0.14 (*top*), 0.17 (*middle*), and 0.20 (*bottom*). For visibility, each word is associated with a different colour in this display. We see that the quantity of aggregates increases with p_{Base} and that each aggregate converged on a different word. 61
5.4 A picture of the grid simulator in the course of an experiment. Black squares are walking agents and other colours are uniquely associated to each possible word and are displayed only when agents having selected the corresponding word are staying. 63
5.5 Mean (dot) and standard deviation (bars) of the quantity of aggregates in vanilla/NG aggregation (resp. left/middle) and of words (right) in swarms of 20/100 (resp. top/bottom) agents after 30 000 time-steps. 65
5.6 Examples of stabilised swarms with a $N = 100$ and parameter a configured as (resp.) 0 ($left$), 0.50 (*middle*), and 2.50 ($right$). For visibility, each word is associated with a different colour in this display. We see that the quantity of aggregates increases with a and that each aggregate converged on a different word. 66

6.1 Average size of the largest cluster for each value of the parameters a and b, evaluated with the cluster metric and averaged over 10 independent runs. 71
6.2 Averaged time evolution over i runs of the three variations: CE ($i = 20$), *baseline* controller with parameters fixed to match the probability table from [Correll and Martinoli, 2007], $i = 20$) and *optimal* ($i = 10$). 72
6.3 Total occurrences of each possible parameter settings (with CE) at the end of 20 runs in setting with $N = \{25, 50, 100\}$ and a circular arena of radius $r = 10$. 73

6.4 Averaged time evolution over 20 runs of the three variations with constant density: CE, baseline behaviour and optimal controller for $N = 25$ and $r = 5m$ (i.e. $(a, b) = (1.5, 2.75)$). 73
6.5 Box-plot of the average performances obtained by 10 optimisation runs in irace for each configurations. A downward trend is clearly visible as the quantity of agents (and/or the density) increases (resp. decreases). 76
6.6 Comparison between CE aggregation and optimal fixed aggregation with a constant density of 1/12. 77
6.7 Comparison between CE aggregation and optimal fixed aggregation with a constant density of 1/25. 79
6.8 Performances of CE aggregation implemented on kilobots. The diameter of the area $d = \{1, 1.4\}$ metres for, resp., $N = \{25, 50\}$ in order to keep the density constant. 80

List of Tables

4.1 Averaged probability to join and leave a cluster as computed by Correll and Martinoli [2007] from observation of gregarious arthropods [Jeanson et al., 2005]. 46
4.2 Summary of the features of the bidirectional and circular interactions. Circular is presented in two versions as v1 was used for preliminary experiments (in Section 6.1), which enabled us to develop v2: the definitive version (Section 6.2). 52

5.1 Table of Experiments 57
5.2 Average 2nd moment of aggregates 62
5.3 Average 2nd moment of aggregates for two population sizes and six parameter settings 66

6.1 Grid sizes 75
6.2 Optimal fixed parameters 75
6.3 Wilcoxon rank-sum with the null hypothesis that the final cluster metrics of CE aggregation and fixed aggregation (in high-density arenas) are drawn from the same distribution. 78
6.4 Wilcoxon rank-sum with the null hypothesis that the final cluster metrics of CE aggregation and fixed aggregation (in low-density arenas) are drown from the same distribution. 80

Chapter 1

Introduction

Cooperation is ubiquitous in nature, from social insects (ants, bees, etc) to humans [Kropotkin, 1902]. Humans are indeed social animals, which is especially evident with labour [Marx, 1875, p.142]. For example, building work requires many part of the product to be completed at the same time and is often made more efficient through bucket brigading. Cooperation also allows time-critical tasks, such as sheep shearing or crop harvesting, to be carried out in large areas. A major factor in these successful examples of cooperation is communication. Although various communication modalities can be found in, e.g. bees, birds or chimps, the forms used by humans are the most complex [Hauser et al., 2002].

Our main form of communication is obviously natural language, which acquired its complexity by evolving during cooperative task execution [Wittgenstein, 1953]. For a long time, philosophers have tried to understand how language connected words and meaning. In this endeavour, they were partially motivated by a desire to reach less ambiguous communication [Stokhof, 2018], which would have facilitated scientific cooperation and, thus, progress. These efforts led to new logical formalisms, such as boolean and propositional logic. These latter formalisms, combined with mechanisation, which was devised to better exploit cooperative human labour, laid the foundations to computer sciences [Church, 1962] and, then, robotics.

Nowadays, automatically-controlled artefacts are posed to replace humans in perilous, difficult or time-critical tasks such as search-and-rescue missions [Kumar et al., 2004], crop monitoring [Albani et al., 2017] or infrastructure repairing [Dams et al., 2017]. Such missions have similar requirements to the human tasks we highlighted above, and could thus benefit from the deployment of multiple and cooperative robots. Ideally, the algorithm executed during these missions would work efficiently with varying number of robots as, e.g., larger areas may require more robots. Fur-

thermore, the environments of deployment are not always controllable. Robots thus need to be able to achieve their task in varying conditions. Finally, robots are prone to system faults and have limited embedded resources: sensors, actuators, processing power and energy. They can therefore stop working during the course of a mission, which should not alter the functioning of the others.

Swarm robotics takes inspiration from social animals in order to solve such challenges [Brambilla et al., 2013]. Indeed, using only simplistic rules and local interactions, swarm of insects have shown to be able to collectively build elaborate structures [Grassé, 1959] or, even, make qualitative choices in a way that resembles a single complex brain [Seeley et al., 2012]. Such behaviours can even create state-of-the-art computational optimisers [Dorigo et al., 2006]. This instance of intelligence emerging from local interactions between comparatively simple individuals highlights features that are typical of System-of-Systems (SoS). Indeed, although a robotic swarm can be perceived as a single entity executing a task, just as a humanoid robot could [Mondada et al., 2004], the major difference between the two is that, in a swarm bot, the parts (i.e. the robots as opposed to the various circuitries in a typical robot) are autonomous, following their own individual purpose. Consequently, they dynamically connect, rather than statically feeding into each other, to fulfil a design that they are not explicitly working towards. These are features that separate SoS from mere systems [Boardman and Sauser, 2006]. Besides, the robustness thus obtained improves deployment time and maintenance as failure or unavailability of any part of the SoS does not have to stop the deployment. Conversely, the simplicity of the different parts and the absence of hardware connection between them reduces operational and maintenance costs as well as energy consumption, which is an increasingly significant concern. These qualities ensure that robot swarms are scalable, flexible and robust.

Communication, the main focus of this thesis, also has a central role in swarm robotics, although it has, so far, been mostly limited to signalling. However, robots are embodied entities and thus exist in the real world, which is open-ended. For robots to be able to carry their task in such an environment, they need to be able to bring forth new meanings according to the situation [Steels, 2003], just as humans do. Previous work in that direction have highlighted a co-dependency between task-solving and language development [Tuci et al., 2011]. Furthermore, linguistic and philosophical research have shown that language and action can co-evolve in a complex social system (where a language is embedded in a cooperative action) through self-organisation only. This phenomenon, which is highly compatible with the *ethos* of swarm robotics, is often called cultural evolution.

This thesis aims to introduce the framework of cultural evolution into swarm

robotics. This endeavour showed that cultural evolution impacts the dynamics of known swarm behaviours in a way that make new features emerge. Moreover, coupled with stochastic communication noise, cultural evolution can tune a controller's parameters according to the local and current context. In this regard, we used self-organised aggregation as a case study. We highlighted the similarities and common interests between swarm robotics and cultural evolution as both depend on collective self-organisation to enable the emergence of complex structures. This thesis presents a number of original contributions, which were published in international conferences and journals.

We studied the effect self-organised aggregation and cultural evolution have on each other, leading to new behaviours. In most settings, cultural evolution does not affect the aggregation but provides an additional collective decision-making mechanism. We also showed that, in specific conditions, agents form a controllable quantity of coalitions, each with its own identifier. These findings were published in an international conference:

- **"Group-size regulation in self-organised aggregation through the naming game,"** Nicolas Cambier, Vincent Frémont, and Eliseo Ferrante. In *International Symposium on Swarm Behavior and Bio-Inspired Robotics (SWARM 2017)*, Kyoto, Japan, Oct 2017.

In an effort to better understand the dynamics of self-organised aggregation, we studied the effect of informed robots in probabilistic aggregation with shelters. We discovered that low proportion of informed robots were able to make an entire swarm converge to the best choice. This study was presented in an international conference.

- **"Self-organised Aggregation in Swarms of Robots with Informed Robots,"** Ziya Firat, Eliseo Ferrante, Nicolas Cambier, and Elio Tucci. In *International Conference on Theory and Practice of Natural Computing (TPNC 2018)*, Dublin, Ireland, Dec 2018.

We also proposed to use the dynamics of cultural evolution in order to create a new, dynamic, approach to self-organised aggregation. This proposition hinges on using culturally evolving words as encodings of aggregation parameters (comparable to a genetic code). In this approach, social dynamics maintain good aggregation parameters that appear through stochastic communication noise. This proposition was the topic of the following publication:

- **"Embodied evolution of self-organised aggregation by cultural propagation,"** Nicolas Cambier, Vincent Frémont, Vito Trianni, and Eliseo Ferrante. In *Eleventh International Conference on Swarm Intelligence (ANTS 2018)*, Rome, Italy, Oct 2018.

This approach was subsequently considerably improved on, to the point that our new model was able to largely outcompete an optimised controller performing regular aggregation. We established several experiments in various environments in order to propose a thorough analysis of the mechanisms enabling this achievement. A journal paper covering this topic is currently in preparation:

- **"Cultural Evolution of Probabilistic Aggregation,"** Nicolas Cambier, Vincent Frémont, Vito Trianni, and Eliseo Ferrante. *Swarm Intelligence*. (in preparation)

This research has application for the exploration of unknown environments with fleets of Unmanned Aerial Vehicles (UAVs) as it can be extended to area coverage and active SLAM.

Lastly, we produced a review on the emergence of communication in robotic swarms, which supports the approach presented in this thesis, i.e., self-organising communications to produce adaptable and autonomous swarms:

- **"Language Evolution in Swarm Robotics: a Perspective,"** Nicolas Cambier, Roman Miletitch, Vincent Frémont, Marco Dorigo, Eliseo Ferrante, and Vito Trianni. *Frontiers in Robotics and AI*, 2020. (accepted)

As well as an extensive state of the art, our contributions and their possible applications will be presented in this thesis. In Chapter 2, we will survey the state of the art in swarm robotics, focusing especially on communication and self-organised aggregation. Then, in Chapter 3, we will examine existing work regarding the evolution of language in artificial systems under a linguistic and philosophical lens. In Chapter 4, we will define a methodological framework to merge swarm robotic and the evolution of language, using insightful examples identified in the previous chapters. This framework will be exploited in Chapter 5, with promising results towards heightened swarm control and group/area labelling. It will be used again in Chapter 6 to show that cultural evolution can enable online parameter optimisation. Finally, in Chapter 7, we will present our conclusions and the direction of our future works.

Chapter 2

Swarm Robotics: State of the Art

Swarm robotics is the scientific field that takes inspiration from social animals in order to solve challenging tasks with robots. Swarm robotic systems are entities built of several individual robots and hold three properties (Robustness, Flexibility, and Scalability), which are ensured through decentralisation, redundancy (many–often all–robots perform the same task), local interaction, and simplistic rules [Brambilla et al., 2013]. As they are composed of many interacting individuals whose collaboration provokes the emergence of complex and interesting behaviours, swarms are a prime example of System-of-Systems (SoS) [Boardman and Sauser, 2006].

In order to design adaptable swarm robotic behaviour, Section 2.1, will first present a taxonomy of the different design methodologies that exist in swarm robotics. Then, Section 2.2, will address signalling, which is the main modality of communication in swarm robotic systems, and a necessity to produce adaptable behaviours. In Section 2.3, we will review the state of the art regarding self-organised aggregation, which will be our case study for collective behaviour in this thesis. Finally, in Section 2.4, we will consider some examples of applications that could benefit from swarm robotics, namely area coverage and simultaneous localisation and mapping (SLAM).

2.1 Design Methodologies

In this section we will use a taxonomy similar to a review of the swarm robotics literature [Brambilla et al., 2013]. The authors proposed to classify the design methods used in the literature in two categories: Behaviour-based and Automatic design methods. In this section, we will explore the design tools available in swarm robotics for these two categories in, respectively, Section 2.1.1 and Section 2.1.2.

2.1.1 Behaviour-based Design

Behaviour-based design methods consist in combining different behaviours to define the way a robot interacts with its environment and its neighbours, in a way that these interactions create an emerging behaviour that solves the task at hand. One major model of behaviour-based controller is Probabilistic Finite State Machine (PFSM). As emerging behaviours are difficult to predict, these approaches are often bio-inspired.

A survey of the best-of-n problem [Valentini et al., 2017] further divides behaviour-based design methods (or "bottom-up" approaches, as they call them) in the "opinion-based" and "*ad-hoc*" subcategories. Although this survey was concerned only with the best-of-n problem ("an abstraction capturing the structure and logic of discrete consensus achievement problems" and encompassing the majority of the tasks studied in swarm robotics), we believe that the categorisation it presents is applicable to most collective behaviours.

In **"opinion-based"** approaches, the robots have an internal representation of their individual choices and the crux of the design consists in enabling them to communicate their decisions to each other and to change opinion in order to reach a consensus. This results in generic controllers that are suited to different tasks such as monitoring [Albani et al., 2018] or foraging [Miletitch et al., 2018] and transition from one target to the other according to some utility function.

In contrast, "ad-hoc" approaches are designed to solve a specific task. In best-of-n problems, they can be divided in two further subcategories: aggregation strategies and navigation strategies [Valentini et al., 2017].

On one hand, **aggregation strategies** enable a swarm of robots to gather in a single area. In behaviour-based aggregation, individuals usually explore the entire environment randomly until they find a shelter or an aggregate and stop. A second mechanism is required to reach a collective decision on which area to actually converge to (otherwise, the robots would just form several small clusters). This mechanism is based on a probabilistic stopping and leaving criteria. Aggregation strategies are thus modelled as two-states PFSM with probabilities modelled after real cockroaches [Garnier et al., 2005, Correll and Martinoli, 2007, Bayindir and Sahin, 2009] or honeybees [Bodi et al., 2012]. Less direct emulations do exist and can use a third state that ensures a robot has left the vicinity of the aggregate before having the possibility to join again. Other, non-behaviour-based, approaches to self-organised aggregation exist, which will be reviewed in Section 2.3, along with a more extensive discussion on behaviour-based aggregation.

On the other hand, **navigation strategies** are mostly concerned with exploring the environment efficiently, sometimes to find some target object. They can take

inspiration from birds to produce a flocking behaviour [Ferrante et al., 2014]. They can also use some kind of stigmergic communication (see in Section 2.2), which require them to alternate between exploring and acting as a trail for other robots by joining the tail of a chain according to some probability [Nouyan et al., 2009, Campo et al., 2010]. The switch between the two roles can become unnecessary when different kinds of robots are used for each role [Ducatelle et al., 2011a].

Behaviour-based design methods have limits as, as explained above, the properties of such systems-of-systems are difficult to predict. Therefore, formal design methodologies have yet to emerge and designers are often left to tune their controllers by trial-and-error [Francesca et al., 2015]. Moreover, bio-inspiration does not ensure well-functioning algorithms, especially when such algorithms have been tuned so much that their original inspiration is more akin to a remote metaphor than to a real template [Camacho-Villalón et al., 2018].

2.1.2 Automatic Design

In opposition to behaviour-based design methods, automatic design methods propose to automatically optimise controllers according to a high-level description of the problem to solve. However, our survey of the literature regarding automatic design methods suggests that the schism between automatic and behaviour-based methods is not as obvious as it appears at first sight as they all have to be *a priori* designed to a certain degree. In this section, we will review four approaches to automatic design: Evolutionary swarm robotics [Nolfi and Floreano, 2000], Automatic Modular design [Francesca et al., 2014], Turing Learning [Li et al., 2016] and Embodied Evolution [Bredeche and Montanier, 2010].

Evolutionary swarm robotics proposes to use the principles of Darwinian evolution in order to efficiently optimise a robot's controller. The narrative behind this use of genetic algorithms, is that this process is responsible for incredibly efficient and robust structures in nature.

Evolutionary swarm robotics features the traditional methods of genetic algorithms; a pool of random genetic sequences (each an encoding of a possible solution to the problem at hand) are replaced in subsequent generations by the produce of reproduction (by recombination and mutation) between the best individuals of the previous run. In evolutionary swarm robotics, the robots' controller is modelled as neural networks that maps sensory inputs to actuators [Nolfi and Floreano, 2000, Trianni et al., 2008]. Thus, the genetic sequence is an encoding of the parameters of this neural network. The initial population's individual parameters/gene sequences are randomly set and then selected for reproduction according to a fitness function;

the high-level description of the problem mentioned earlier.

Despite their efficiency, evolutionary approaches such as evolutionary swarm robotics or automatic modular design have many problems [Valentini et al., 2017]. Even though evolutionary swarm robotics is the most widely utilised approach to develop efficient collective behaviours, it has a specific problem as, from the usage of neural networks, it follows that the evolved controller are black-boxes. They are thus difficult to model and analyse mathematically and to maintain accordingly.

Automatic Modular Design (AutoMoDe), in contrast, uses optimisation methods to build a Finite State Machine (FSM) instead of a neural network [Francesca et al., 2014]. In this approach, a wide variety of base behavioural modules and condition-defined events are available. Examples of behavioural modules include random walk, phototaxis and obstacle avoidance whilst events include the detection of white or black floor, neighbours count, fixed probability, *etc.* AutoMoDe then encodes a FSM wherein states are modules and transitions are events into a mathematical problem that it tries to optimise. As a matter of fact, AutoMoDe has been shown to outperform both evolutionary robotics and behaviour-based approaches [Francesca et al., 2015][1], even more so when given the ability to evolve communication [Hasselmann et al., 2018].

Despite their remarkable efficiency, both performance- and design-wise, offline approaches (evolutionary or otherwise) suffer from a series of problem when applied to swarm robotics [Francesca and Birattari, 2016]. Indeed, the many runs necessary for the optimisation mean that it has to be executed in simulation. In addition to being a centralised approach, this also risks overfitting the controller to the simulated setting, which means that the performances suffer during the deployment in real world [Jakobi et al., 1995]. Moreover, this "reality gap" might not be induced by reality itself but simply by the slight differences between settings [Ligot and Birattari, 2018], which highlights the sensitivity of evolved controller to experimental conditions [Trianni et al., 2003]. Therefore, on one hand, off-line approaches are computationally intensive but, on the other hand, they are often efficient for extremely specific scenarios and must thus be re-executed for any new setting.

Turing Learning is a coevolutionary approach to designing bio-inspired controllers [Li et al., 2016]. The aim of this approach is to reproduce existing natural behaviours through observation of their motion only and by evolving two models: the *replica* and the *classifier*. The *replica* is the controller whose goal is to trick the classifier into categorising the replica's motion as genuine, i.e. as data observed

[1]Let us note that manually-built controllers were also more efficient if they were designed only with the modules and event available to AutoMoDe. Beating them even required an update to AutoMoDe's optimisation algorithm. An interesting example of creativity benefiting from constraints.

from the natural behaviour. Conversely, the goal of the classifier is to be able to discriminate between genuine data and data produced by the *replica*. This approach has been used to reproduce two canonical problems in swarm robotics: self-organised aggregation (*cf.* Section 2.3) and object clustering.

Turing Learning has a couple of advantages compared to other evolutionary methods. Firstly, it can be performed on-line and in the real-world (although the optimisation is still performed on a centralised server), thus greatly reducing the problems of the reality gap and of adaptation to new environments. Secondly, it does not require any kind of metric and/or fitness function and is thus able to reproduce the original behaviour more accurately as swarm behaviours are too unpredictable to be inferred from metric-based method.

The obvious weakness of Turing Learning is that it is only able to *reproduce* existing behaviours. Although many natural swarm behaviours are extremely efficient, this fundamentally limits the possibility of innovation. Moreover, as swarm behaviours are indeed complex and can vary depending on the environment, experiments with the original natural system might need to be performed for new challenging settings before being able to teach them to the robots.

Embodied Evolution is the only automatic design method that enables robots to learn continuously in their lifetime. Thus, it bypasses the problem of adapting from one environment to another, including *in silica* to real life. It it a distributed adaption of evolutionary robotics in the sense that is uses the same tools but that the selection and replacement are performed locally by each robot. This local evolution can follow three models: distributed (each robot has a unique genome which changes when mating with other robots), encapsulated (each robot performs the evolutionary process internally without exchanging genetic material) or a hybrid between the two, also called the physically distributed island model [Silva et al., 2015]. Thus, embodied evolution adds the mating operator[Bredeche et al., 2018].

Mating is the local exchange of genetic code between two robots. It can be symmetrical [Bredeche and Montanier, 2010] or, oppositely, robots can stop at a given point in their lifetime to "receive" genetic material [Noskov et al., 2013]. Moreover, mating can be more or less distinct from the replacement operator as new genetic material can be integrated to current genome on the fly [Watson et al., 2002] or added to a genetic pool specific to each robot [Noskov et al., 2013]. The local and distributed nature of the mating operator also create a greater genetic diversity that eventually increases performances for a given task compared to centralised evolution [Pérez et al., 2018].

Further, the ability to mate can also serve as an implicit selection operator thus rendering a fitness function unnecessary [Bianco and Nolfi, 2004]. The idea under-

lying this implicit fitness is that robots that are more capable to feed at an energy source (foraging) or to catch other robots (self-assembly) will also communicate with more robots (resp. because it continues to move and because it seeks them) and thus spread its genetic code accordingly. This approach with implicit fitness characterises algorithms like minimal Environment-driven Distributed Evolutionary Adaptation (mEDEA) [Bredeche and Montanier, 2010]. A subsequent variation of this concept, MONEE [Noskov et al., 2013] ported this implicit fitness to other, less fitting tasks, by rewarding robots that successfully accomplish their task with a reward. These rewards are then transmitted, along with their genetic code, to "eggs", which are inactive robots (because they have reached the end of their life-cycle) gathering genetic material. Before starting a new life-cycle, the eggs thus select their new genome according to the rewards associated. This is called "parental investment". Embodied evolution has recently been adapted to low-cost robots by replacing the sexual reproduction metaphor (i.e. mating-selection-replacement) by the horizontal transfer of genetic code found in bacteria [Bredeche, 2019]. In this algorithm, only small bits of the genome are transmitted to conspecifics and these bits override the current ones in the receiver only if the reward associated to the donor is higher than the receiver's current reward. This transfer takes advantage of the constraints of low-cost robots as the size of the bits of code transmitted depend on the bandwidth of the robots. Smaller bandwidth therefore provide higher performances (as various genomes are eliminated less quickly and more variations appear) at the cost of a slower convergence time.

As we saw in this section, although all of these approaches have an automatic optimisation dimension to them, latter ones also display aspects of behaviour-based design as their controller learn to mimicry specific biological behaviours (Turing Learning) or are *a priori* designed to some extent (Embodied Evolution). This is not, however, necessarily a problem as these approaches can conversely avoid the requirement to formulate a fitness function for the task, which can fail to encompass every aspect of the problem or lead to large reality gaps [Nelson et al., 2009].

2.2 Signalling

Local communication is a central aspect of swarm robotics and often takes inspiration from the communication behaviour of social insects that use these kind of interactions to organise themselves. Animal communication, or signalling, has a long history in ethology. Indeed, signalling was initially defined as rigid signals trig-

gering a pre-determined and hard-wired reaction in conspecifics[2]. As information theory grew in popularity, signals started to be viewed as encodings, first, of the signaller's internal state and, later, of environmental information. In any of these cases, signalling is always conceived as having biologically evolved by bringing an evolutionary advantage, either because cooperation is essential to survival or egoistically, as helping conspecifics with similar genetic make-up (i.e. children) propagates one's own genome. Eventually, empirical studies demonstrated that signalling can even be symbolic as, e.g. vervet monkeys have specific alarm calls for each of their predator species. As a result, when threatened, signallers can give their listeners the opportunity to use the strategy most effective to escape this specific predator.

These information-based definitions were later replaced by an influence based one: Animal signalling is the use of *"specialized, species-typical morphology or behavior to influence the current or future behavior of another individual"* [Owren et al., 2010]. This definition has the advantage of encompassing previous definitions while being much more rigorous. Indeed, this new definitions more clearly rejects behaviours such as, e.g. incidental cueing/implicit information transfer (i.e. passive perception of conspecifics such as in self-organised aggregation).

A previous research on communication in swarm robotics [Trianni and Dorigo, 2006] isolated three communication behaviours in insect societies: stigmergy (i.e. indirect communication), direct interaction and direct communication.

The definition of signalling emphasised above includes all three of these communication (whereas earlier definitions would have excluded "direct interactions" as there is no encoding of information involved in these behaviours). The rest of this section will therefore review the state of the art for signalling in swarm robotics by focusing on stigmergy (Section 2.2.1), direct interaction (Section 2.2.2) and direct communication (Section 2.2.3).

2.2.1 Stigmergy

Stigmergy is a behaviour wherein communication is effectuated through modifications of the environment. It was first introduced in the midst of the XXth century to describe the behaviour of termites [Grassé, 1959], which are capable of building complex nest structures only by depositing their small balls of mud next to other balls with higher probability.

This behaviour can easily be reproduced with robots [Beckers et al., 2000] for object clustering or object aggregation (the latter involves objects that have to be

[2]The claims in the following paragraph are all largely based on Owren *et al.*'s historical notes on animal signalling [Owren et al., 2010]

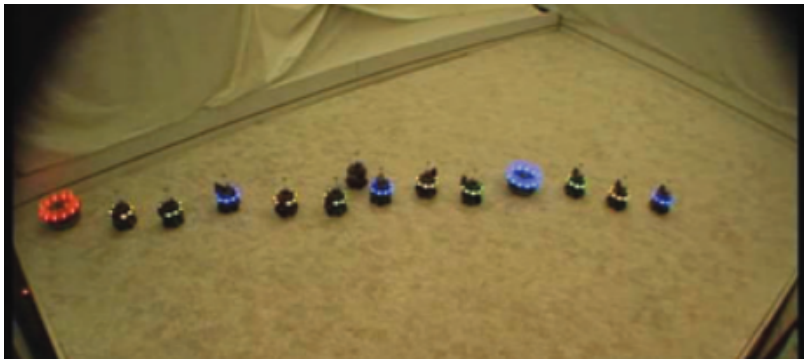


Figure 2.1: Example of collective chain formation. *From [Nouyan et al., 2009], reproduced with permission ©2009 IEEE.*

connected to each other) [Brambilla et al., 2013].

Another classical example of stigmergy can be found with the exploration of ant colonies who use pheromone trails to find the shortest path between their nest and food sources [Deneubourg et al., 1990].

However, pheromones are difficult to reproduce in robotics swarms. A few work on this topic propose particular actuator/sensors for this purpose, such as UV-light emitters that leave a mark on a specifically-designed floor [Alers et al., 2014]. Because these approaches are not exactly versatile, many research on the topic of stigmergy in swarm robotics prefer, as illustrated in Figure 2.1, to use a part of the swarm as pheromone trails that guide the way for other robots [Nouyan et al., 2009, Campo et al., 2010]. The remaining robots (those that do not constitute de chain) can then follow the chain to the target. The difference between "pheromone" and regular robots can also be formalised by using a different kind of robot for each role [Ducatelle et al., 2011a].

2.2.2 Direct Interaction

In direct interaction, an individual directly influences another by physical contact which provokes a response. This is a somewhat unintuitive example of communication if one focuses their understanding of communication on information theory. However, under an influence-based definition, direct physical contacts can be understood as signalling as long as they trigger a behaviour on the other side of the interaction (i.e. merely pushing another robot is not signalling, unless the pushed robot eventually starts to walk in the same direction by itself).

Although examples of direct interaction are rare in swarm robotics (because contact between robots is generally avoided or discarded [Trianni and Dorigo, 2006]), some examples can be found in coordinated motion, which can apply to self-assembly and collective transport [O'Grady et al., 2010].

Self-assembled swarms can cross obstacles that individual robots cannot, such as hills (by multiplying motor power) or holes (by forming structures that are larger than the hole). In both cases, the individual parts of the structure have to move in a common direction, even if some of them are not able to see the target and, thus, to decide said direction. In such cases, there are two challenges. The first is to self-assemble, which can be complicated as choosing who is the gripper and who is the grippee is not straightforward. This can however be negotiated via direct interactions [Ampatzis et al., 2009]. The second challenge is then to move in a common direction, which can be done by aligning oneself with the direction of traction sensed by the robot's torque [Groß et al., 2006].

This is quite similar to the collective transport behaviour of ants [Kube and Bonabeau, 2000], which can effectively be reproduced in robots in a similar fashion [Baldassarre et al., 2006, Groß et al., 2006].

2.2.3 Direct Communication

Direct communication is non-mediated transmission of information that requires no physical interaction, such as the waggle dance that bees use to inform their colleagues of the position of pollination sites [Riley et al., 2005].

This is the most common form of communication in swarm robotics. It can be used for, e.g., foraging [Ducatelle et al., 2011b], self-organised aggregation [Soysal and Sahin, 2005] or morphogenesis [O'Grady et al., 2009], and its various possible modalities have been thoroughly studied [Mathews et al., 2010, Trianni et al., 2016]. In fact, direct communication supersedes all other forms of communication in swarm robotics. Indeed, stigmergic implementations are often closer to a metaphor that actually uses directly communicating robots as pheromone trails [Ducatelle et al., 2011a]. Furthermore, direct interaction requires specific sensors (torque feedback) and physical contact between robots, which make real-world implementations difficult. Therefore, it is often replaced—or complemented [Trianni and Dorigo, 2006]—in collective transport and coordinated motion by direct communication of the robots' direction to similarly effective outcome [Campo et al., 2006, Ferrante et al., 2013], using strategies that are more similar to flocking [Ferrante et al., 2014].

Numerous works in evolutionary swarm robotics have also grappled with direct communication. They showed that the latter can emerge spontaneously [Floreano

et al., 2007], i.e. without any explicit requirement for communication in the fitness function or in the task at hand. Through these works, direct communication was shown to provide a significant advantage to foraging [Floreano et al., 2007] and categorisation [Ampatzis et al., 2008, Tuci, 2009] as well as creating more resilient controllers for such tasks as categorisation, synchronization and aggregation [Hasselmann et al., 2018].

2.3 Self-Organised Aggregation

Self-organised aggregation is a decision-making process ubiquitous in nature [Camazine, 2003] whereby agents gather all around the same area, without relying on global information, global communication, or any kind of centralised information or decision. In addition to being a widely studied behaviour of swarm robotics, it is also one of its most fundamental building blocks. Indeed, many problems in swarm robotics require the swarm to gather in order to be solved. These include self-assembly and obstacle avoidance [O'Grady et al., 2010], coordinated motion [Trianni et al., 2006, Ferrante et al., 2014] and cooperative object transport [Tuci et al., 2018].

Depending on the objective, the site of aggregation can be either an area in the environment that can be clearly perceived by all agents (and that has to be selected amongst several similar shelters) or non-specific and non-differentiable from any other part of the arena. Approaches to aggregation with shelters will be reviewed in Section 2.3.1 whilst Section 2.3.2 will review approaches that do not require shelter to form aggregates.

2.3.1 Aggregation with Shelters

Early studies on aggregation were inspired from *Blattella germanica* cockroach larvae's behaviour. In this model, agents collectively choose a shelter among several possibilities [Deneubourg et al., 2002]. In order to choose a shelter, each agent explores the entire environment randomly and stops according to the quantity X_i of conspecifics in shelter i, following a probability P_i given by Equation 2.1:

$$P_i = \frac{a}{1 + bX_i^2} \tag{2.1}$$

with $a = 0.01$ and $b = 0.16$ [Deneubourg et al., 2002].

By only using this mechanism, agents would aggregate randomly in all of the shelters. However, agents can also leave the cluster with probabilities that, according to later models [Amé et al., 2006], increase with n in order to avoid overcrowding as

shelters have a limited carrying capacity (see Equation 2.2). The models suggest that the probability to join can also be affected by the carrying capacity of the shelter.

$$R_i = \mu(1 - \frac{X_i}{S}) \tag{2.2}$$

where S is the quantity of agents that can fit in a shelter.

The decision making dynamics of this behaviour was also studied using differential equations [Amé et al., 2006]. This work analysed the decision making dynamics of a swarm of size N when it had to choose between two or more identical shelters of total carrying capacity S. An important find in this study is that of the mechanism of the decision making of the agents according to the ratio $\sigma = S/N$. Indeed, agents only make a collective choice if and only if there is a shelter for which $\sigma \geq 1$. In any other case, they distribute themselves equally in as few shelters as possible (even if the shelters have room to fill). Later studies continues this analysis with experiments on robots and shelters of carrying capacity [Garnier et al., 2009, Campo et al., 2011]. These works showed that agents following this kind of probabilistic aggregation are able to collectively select shelters that are neither too small nor to big.

Because, as expressed above, self-organised aggregation has many applications for collective behaviours, this model has inspired many algorithms in swarm robotics. The closest studies [Garnier et al., 2005, 2009], performed a very similar aggregation task, and even used the probability tables reported from observations of actual cockroaches [Jeanson et al., 2005].

Other insect species have also been taken as inspiration to perform self-organised aggregation. Following a principle very similar to cockroaches, honeybees can aggregate on hotspots [Bodi et al., 2012]. The only significant difference between the two behaviours is that, here, the probability distribution to stay do not increase with quantity of neighbours but with the local temperature, though the decision to stay or not is still triggered only when meeting other agents or aggregates.

Finally, automatic designs of aggregation with shelters are possible [Hasselmann et al., 2018], especially modular designs which perform better than any other evolved behaviour and resist better to the reality gap.

2.3.2 Aggregation without Shelters

The behaviour of cockroaches has also been studied in continuous environments. To model the relevant distribution of probabilities, Jeanson et al. [Jeanson et al., 2005] designed separate experiments with groups of two to four larvae in a featureless arena. By tracking the paths of the larvae, they were able to compute the resting times of

agents depending on the quantity of conspecifics around them. Their observations show that the walk of the larvae depend on their position in the arena (random walk in the center against wall-following on the periphery) and that the distribution of the resting times is bimodal; pauses can belong to either of two distributions: short or long pauses. In addition to both distributions increasing with n, the probability to choose a long pause also increases with n. From this, they were able to recreate their model artificially for validation by inverting the resting times into leaving probabilities by time-steps of one second. This model was also ported to robots for further study, which demonstrated that successful robotic aggregation highly depends on the probability of encounter between robots and, thus, on external parameters such as the density of robots and their communication range [Garnier et al., 2008].

Later works [Correll and Martinoli, 2007, 2011] proposed a simplified implementation by averaging the original bimodal distribution into a unimodal one and ignoring the variation of the walk mentioned above. As a result, this model of aggregation can be summarised as a probabilistic finite state machine (PFSM) with only two states, one to walk and one to stay, and simple transition probabilities that vary with n.

A modification of these distribution similar to [Correll and Martinoli, 2011] also allowed [Bayindir and Sahin, 2009] an update of the formula proposed in [Amé et al., 2004] (observations with shelters) to perform aggregation without shelters. Shelterless aggregation is also achievable using a PFSM with three states as, in addition to walking, they can approach or be repelled by aggregates [Soysal and Sahin, 2005], which is not bio-inspired.

One can also use the framework of evolutionary swarm robotics (see Section 2.1.2) to create optimal aggregation behaviours [Trianni et al., 2003, Dorigo et al., 2004, Şahin, 2004], which can cause the formation of either static clusters (as with probabilistic aggregation) or denser dynamic flocks. Embodied evolution can also ensure self-organised aggregation. OdNeat, is a multi-purpose embodied evolution algorithm, implemented as a distributed island model [Silva et al., 2015]. In this model, the neural network controller is evolved locally in each agent but some horizontal genetic exchanges are possible for robots in close proximity. The evolution algorithm optimises the topology as well as the weights of the neural network and also maintains a tabu list of the worst controllers. When the rewards used to calculate the fitness of the controller are the quantity of horizontal genetic exchanges, this algorithm successfully evolves aggregation controllers that perform similarly well to offline evolutionary methods.

Abandoning the biologically-inspired approaches of evolutionary robotics or ethology, a later model [Gauci et al., 2014] of aggregation reached scalable results with deterministically controlled robots equipped with a single one-bit sensor,

without using any PFSM but only a simple sensors-actuators mapping. However, they also had to perform experiments with every possible parameter tunings before selecting the most efficient. This controller also yielded dynamic flocks moving in circular motions and merging with each other. More recently, [Katada, 2018] created a probabilistic aggregation controller by formalising it as an optimisation problem that they solved with *Particle Swarm Optimisation* [Eberhart and Kennedy, 1995]. In addition to being scalable and interpretable, the controller they ended up with was very similar to [Soysal and Sahin, 2005]. Finally, likening robots to *Brownian Particles* [Einstein, 1905], shows that sensorless agents that can only sense when they run into obstacles can distribute themselves in distinct areas of high and low densities following a random walk under some speed conditions [Mayya et al., 2019]. This simplistic behaviour is sufficient to aggregate around a third of robots and has applications in contexts where aggregating a significant part of the swarm is necessary for the exchange of information whilst the spread of the whole is still important (e.g. distributed sensing or area surveillance).

To conclude, self-organised aggregation is amongst the simplest collective behaviours. Its various models are also very similar to those of other behaviours such as chain formation [Deneubourg et al., 2002, Nouyan et al., 2009], PFSM for foraging [Albani et al., 2018] or, more generally, evolutionary robotics solutions. It is also an instance of collective decision-making, which can itself take the form of many swarm robotics behaviours [Valentini et al., 2017]. Therefore, in addition to being a prerequisite for other, more complex behaviours, self-organised aggregation can also be exploited as an initial case study for swarm robotics in general.

2.4 Applications to Collective Exploration

Exploration of unknown environments is a challenging task, that is often accomplished with unmanned vehicles in order to limit human hazard. Performing exploration simultaneously with several vehicles can be a necessity in cases where the area of exploration is especially large.

Robust, flexible and scalable solutions are of paramount importance for this task as real robots are prone to communication and detection errors and other failures. This concern is even more prevalent for UAVs as the weight of aerial vehicles is critical to their battery usage [Kumar and Michael, 2017]. As a consequence, UAVs are bound by a trade-off between payload and battery, leading them to be severely limited with regard to the reliability of their equipment (even more so considering the challenges induced by three-dimensional motions). Moreover, the consequently short-

lived batteries induce the risk of several agents deactivating during the execution of a task. Therefore, collective exploration should be ensured by fully distributed bio-inspired strategies [Albani et al., 2017].

This section presents an overview of the challenges of collective exploration of unknown environments in robotics along with some examples of solutions swarm robotics has or could bring to them. This section is divided into two parts, which deal, respectively with area coverage (Section 2.4.1) and active SLAM[3] (Section 2.4.2).

2.4.1 Area Coverage

Multi-robot area coverage is a task wherein robots have to organise in order to maximise their exploration of an environment as quickly as possible and avoiding coverage redundancy (i.e. several robots exploring the same area) as much as possible. The two main approaches to this problem are blanket and sweeping coverages [Papatheodorou and Tzes, 2018].

On one hand, Blanket coverage consists in distributing the robots in a static network that covers as much as the environment as possible. Therefore, this approach presents a trade-off between remaining within communication range and maximising the coverage of the environment. In this regard, most effort exploit the works of the wireless sensor networks (WSN) community on the same topic [Mulligan and Ammari, 2010]. A good solution for homogeneous environments is to divide them in Voronoi diagram (as illustrated in Figure 2.2) and to assign each robot to a region [Vieira et al., 2003]. These non-overlapping regions are ideal for multi-robot area coverage because they reduce redundant data whilst optimising the constraints related to radio range, energy waste and communication interference. For an unknown environment, however, a blanket strategy would need to possess a dynamic variation. Existing examples include area negotiation [Ahmadi and Stone, 2006] and dynamic segmentation in Voronoi regions [Wurm et al., 2008].

Swarm robotics can tackle this challenge in a distributed and flexible fashion by, as usual, getting inspiration from natural sciences. Indeed, equations of electrostatic potential fields can maximize the area covered and form a well-connected communication network [Howard et al., 2002], without requiring any knowledge about the area.

On the other hand, sweeping strategies consist on following an *a priori* set path optimised to the constraints of the exploration task and can be exploited with one

[3]Simultaneous Localisation and Mapping

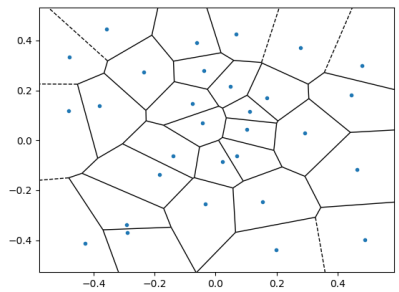


Figure 2.2: Example of a Voronoi diagram. Each region V_i has a centre point p_i. Any point in V_i is closer to p_i than to any other centre point.

[Ahmadi and Stone, 2005] or multiple robots [Kurabayashi et al., 1996]. However, these sweeping strategies heavily rely on planning and, thus, known environment, which is a strong assumption.

Swarm robotic approach can improve the flexibility of sweeping coverage. If recognisable landmarks are available, swarm robots can discretise their environment by building a dynamic graph wherein vertices are the landmarks [Rutishauser et al., 2009]. Individual robots can locally exchange the vertices they have mapped as well as the ones they have already explored and then move to the closest unexplored vertex. If no landmark is available, sweeping can still be performed using basic behaviours such as random walk, wall-following, obstacle avoidance, *etc*, or combination thereof [Doty et al., 1993]. These behaviours can be tuned for maximum efficiency in robot swarms with in-depth mathematical [Correll, 2008] or experimental [Dimidov et al., 2016] study of their dynamics.

2.4.2 Simultaneous Localisation and Mapping

Simultaneous localisation and mapping (SLAM) is a twofold challenge between estimating the state of one or more robots and building a model of the information acquired by their sensors, i.e. a map. SLAM has applications in all manner of situations as soon as robot's displacements in an uncharted area are required. It is also an alternative to user-built maps and is especially useful indoor where Global Navigation Satellite System (GNSS) localisation is often unavailable. In addition to serving as a support for other operations such as path planing, place recognition or loop closure, the maps also limits the error in the estimation of the pose. SLAM is

thus a task of paramount importance in robotic exploration.

Applications where robots do not passively perceive and process data but also explore the environment to maximise the exhaustivity and accuracy of the map and of the localisation are called *active* SLAM. Although area coverage (see Section 2.4.1) can obviously be used for such purpose, the modalities of SLAM algorithms create some specific requirements. Therefore, before we can look into applications of swarm robotics for active SLAM, we have to describe the workings of SLAM and its specific objective when it comes to navigation.

On the purely computational side of SLAM, modern systems can be divided into two modules: the *front-end* and the *back-end* [Cadena et al., 2016], with the front-end extracting relevant features and abstracting the sensor output into general models, whilst the *back-end* processes data into the map. On a more engineering-oriented perspective, SLAM includes three features: the formal encoding of the map, the sensors used and data processing [Saeedi et al., 2016]. Whilst the *back-end* is equivalent to data processing in this perspective, the *front-end* should be associated mainly with the sensors as it is an abstraction of them.

Map Representation. As mapping is half of SLAM, the representation of the map we eventually want to obtain is of paramount importance. As most usages of SLAM are in 2D space, the state of a robot is most often described as its pose, i.e. position and orientation [Cadena et al., 2016]. The literature exposes six types of such representations, each adapted to different uses [Saeedi et al., 2016]:

- *Grid maps* build a discrete representation of the space by dividing it in adjacent cells. The most common maps are occupancy grid maps [Mahdoui et al., 2017] where each cell represents a rectangular area and is associated with a probability that there is an object in the cell.

- *Feature maps*, or *landmark maps* [Atanasov et al., 2015] only register landmarks and their global positions.

- *Topological maps* are abstract representation of space in a network-like configuration wherein vertices are places and edges show accessibility from one place to another [Beeson et al., 2010]. This representation has been likened to the way animals and even humans build a cognitive representation of large-scale spaces [Kuipers, 2000].

- *Appearance maps* connect images of an environment in a graph wherein edges link views that are sufficiently similar to picture the same place [Saeedi et al., 2016].

- *Hybrid maps* combine different mapping methods to fill the deficits of each other.

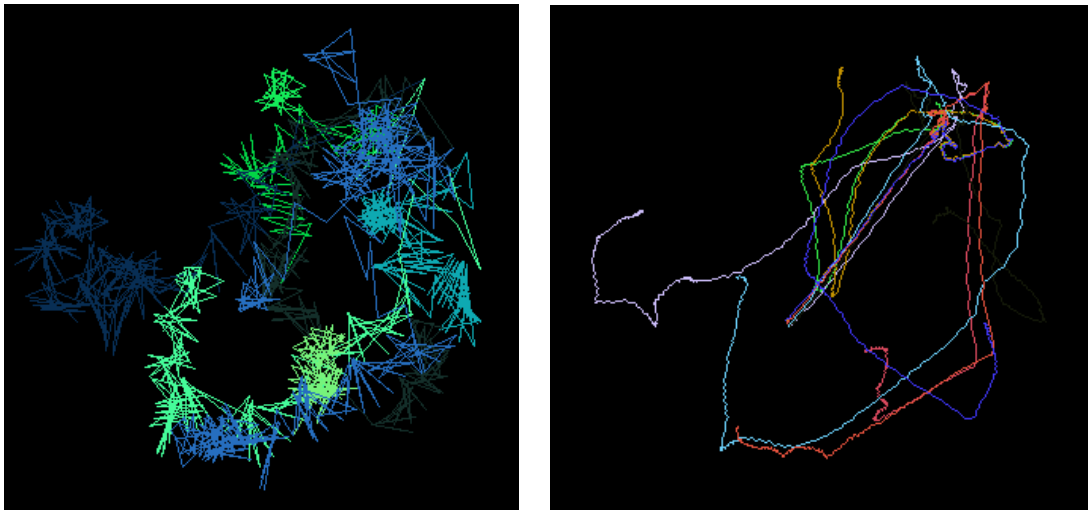
The choice of representation for the map is thus highly dependant on the goals the SLAM algorithm is deployed for. Nevertheless, some of these formalism are only applicable with certain sensors. Thus, in any SLAM implementation, the map representation will also be dependant on the available sensors.

Sensors and *front-end*. Robots all suffer from limitations with regards to the amount of equipment they can carry. These limitations mainly impact the quality and diversity of exteroceptive sensors (i.e. sensors that perceive and obtain measurements from the environment) which are most often heavy and energy-intensive. These sensors include sonars, range lasers, GNSS and cameras [Fuentes-Pacheco et al., 2015]. Among these, cameras are the most advantageous as they are lightweight and have low power consumptions. Nevertheless, a single source of measurements is insufficient to ensure a robust and accurate estimation of a robot's position [Castellanos et al., 2001]. This problem can be mitigated using proprioceptive sensors (e.g. accelerometers or gyroscopes) to estimate the robot's current pose as its previous pose translated according to the sensors' measurement.

Likewise, one can also directly use the robot's odometry [Thrun et al., 2005] which is not the measurement of displacement but the displacement that should correspond to the activation of the actuators. Odometry is a good example to explain the role of the *front-end*. Indeed, robots can have different actuators (wheels, propellers, etc.) with different sizes of effectiveness. As a SLAM implementation aims to be portable, it can not directly feed the actuation data (i.e. actuators have been activated for x seconds with a power of y.) into the *back-end*. Instead the *front-end* should be specific to the robot and translate the effect of the actuation, according to its own specifications, into a common measurement, e.g. a distance.

Data processing or *back-end*. The predictive transform from the previous position to the current position, however, are not deterministic [Thrun et al., 2005]. Indeed, both proprioceptive sensors and odometry suffer from noisy measurements. Consequently, left as is, this approach accumulates error and leads to entirely false localisations. The role of the *back-end* is to correct these estimations in order to produce accurate localisation and mapping.

The most straightforward solution is using probabilistic (Bayes) filters to correct the noise according to previous displacements. In this regard, the Kalman filter [Kalman, 1960] is the most widely used approach (e.g. EKF-SLAM Bailey et al.



(a) (b)

Figure 2.3: Paths of agents following a noisy random walk (a) and passed through a Kalman filter (b) (colour schemes are random).

[2006]). From previous displacements, this filter builds a correlation matrix between the x,y,z coordinates and crosses its prediction with the current motion data to predict the most probable position. Its effect is illustrated in Figure 2.3. Though extremely useful, the Kalman filter is somewhat limited as it would not correct the consistent skewing that could arise, e.g. from asymmetrically calibrated wheels in odometry.

Another way to correct a localisation is to average it with other estimations that were produced at the same location; a robot can recognise a previously visited location by comparing its exteroceptive sensors measurements and then correct its current estimation of its pose using its estimation at the time of the previous visit. This is called "loop closure" and is one of the cornerstone of SLAM as the accuracy of the process depends on this [Mur-Artal and Tardós, 2017]. Loop closure can also be performed using other robots data. This possibility provides a higher quantity of corrections but also introduces the challenges of transmitting the relevant data from robot to robot whilst accounting for the robot's capabilities and ensuring the continued exploration of the environment.

Active SLAM. SLAM presents a trade-off between exploration and communication. On one hand, exploration is obviously the goal and requires the robots to spread

for maximum efficiency. On the other hand, robots need to group occasionally as the "mapping" part of the process includes data fusion and loop-closure which are enabled by direct communication (that is, unless the data processing is performed on a central server [Karrer et al., 2018]). Moreover, certain areas might need to be updated regularly (e.g. in surveillance tasks) which means that robot need to either explore these areas multiple times or remain in their vicinity. This balance between exploration and exploitation is most often tackled by computing the utility of each possible localisation. This utility depends on the action of other robots, which therefore have to communicate their current maps. As the data to transmit and to compute in this case can be of a huge size, the robots are often limited to transmitting frontier points [Mahdoui et al., 2017], i.e. the limits of the explored area. The challenges of active SLAM are thus threefold as robots need to choose between exploration, exploitation and communication [Cadena et al., 2016]

Applications of Swarm Robotics. In addition to some proposition of Section 2.4.1 (e.g. [Rutishauser et al., 2009] is perfectly adapted to active SLAM with landmark maps), other swarm robotic approaches to these challenges are also possible. Firstly, localisation errors can be greatly reduced by performing social odometry, i.e. correcting each other's measurements as robots encounter. Social odometry has been shown to improve the performances of robots that had to forage "food" from different sources by strengthening their confidence in the localisation of their target [Miletitch et al., 2013].

More generally, behaviour-based approaches to foraging could easily be adapted to active SLAM as these approaches already implement some kind of utility functions and, e.g., frontier points can be considered as "food sources".

There are typically two approaches to foraging: threshold-based approaches or stochastic task-switching. The formers [Momen and Sharkey, 2009] use the division of labour model observed in social insects [Bonabeau et al., 1996]. Here, the probability to switch to a task decreases with the number of robots that are already executing it. This decreasing is not linear but rather follows a threshold function. In other words, the probability to join a task remains extremely low until the threshold value is passed, in which case the probability converges relatively abruptly (depending on n) to 1. Stochastic task-switching consists in a PFSM whereby robots move towards a food source with a fixed probability or remain idle with another fixed probability [Correll, 2008]. In this approach, all target-commitment are independent states that all transition by an idle state. Robots can also recruit or inhibit their conspecifics according to the size of the task [Albani et al., 2018] (e.g. if they discover a new room).

Despite its remoteness to swarm robotics, even passive SLAM (i.e. the perception and processing of data), can benefit from swarm robotic approaches, especially from its communication modalities. Indeed, an essential feature of swarm communication is its situatedness [Campo, 2011]. Contrary to abstract communication, whose content only is relevant (e.g. transmission of absolute position), situated communication uses its content as well as its context [Stoy, 2001]. This property is the reason approaches such as social odometry [Miletitch et al., 2013] or embodied evolution with implicit fitness [Bredeche and Montanier, 2010] work so well in swarm robotics. The simple fact that the communication is possible is already an information that can be exploited (resp. "we are in the same area" and "this robot survived with the attached genomes"). In Chapter 3, we will argue that this situated communication can be pushed even further in order to make a cultural system emerge, which, as has already been suggested [Albani et al., 2017], could help to build and label exploration maps.

Chapter 3

Evolution of Language: State of the Art

As we have seen in Chapter 2, one of the key components of swarm robotics is local interaction or communication. The later has, so far, only been used for relatively simple tasks such as signalling a robot's preference or state. However, new environments can greatly affect the efficiency of pre-designed interaction strategies. Indeed, pre-programmed communication is, by definition, insufficient to tackle an open-ended world [Steels, 2003]. For swarms to be fully autonomous, the emergence of meaning, as it exists in human language, is therefore necessary.

This chapter will thus explore the possibilities offered to us with regards to evolving useful communication by leaning on theories from fields related to natural languages such as linguistic and the philosophy of language. As we said in Section 2.1, swarm robotic controllers can either be automatically or manually designed. This mirrors the main debate regarding the question of the evolution of language, which is about whether language is innate or learned [Bates, 2003]. In the first view, our ability to master the complexities of language is biologically imprinted and results from millennia of evolution. In the second view, our abilities are not especially optimised for language and the latter would have gained its complexity through self-organisation in order to be transmittable.

Providing evidences for either position is not actually straightforward as language is a complex system at the intersection between three processes: socio-ecological evolution[1], biological evolution, and cultural evolution [Steels, 2011]. Although every-

[1]Socio-ecological evolution postulates that the pressure to use symbolic communication due to the humanity's initial environmental niche would be conserved and heightened through the integration of language into social structures, even though it stopped being an environmental necessity.

body agrees that socio-ecological evolution is the background against which the two other processes act [Steels, 2011], the relative importance of biological and cultural evolution remain unknown.

Section 3.1 will be an overview of works focusing on biological evolution of language complexity which, as we will see, has been a dominant paradigm that influenced applications in artificial systems. Section 3.2 will propose an alternative approach to evolving complex communication systems based on cultural evolution.

3.1 Biological Evolution

Biolinguistics is a field of linguistics that studies the evolution of natural languages. Its main assumption is that the capacity to use, understand and produce natural languages is mostly genetic and thus evolves biologically. In section 3.1.1, we will detail this assumption and the evidences that support it.

In order to better understand the evolution of such a complex system as language, artificial models have been proposed. As we will see in Section 3.1.2, by simulating the process of language inheritance across multiple generations, multi-agent systems can clarify the role of biological evolution in language.

Finally, the biolinguistic view is fully compatible with modern optimisation tools that use genetic evolution. Section 3.1.3 will show how engineering has therefore exploited this paradigm to create multi-agent systems that would solve practical tasks.

3.1.1 The Biolinguistic Paradigm

The biolinguistic paradigm[2] emphasises the importance of biological evolution in forming a faculty of language. The Faculty of Language in a Broad sense (FLB) is indeed biological as language is enabled by anatomical (vocal tract, auditory system, etc) and "computational" (brain power) features. These features are not unique to humans as they are found in many animals [Hauser et al., 2002]. Moreover, robots are already equipped with actuator-sensor systems and computational power that enable communication. The Faculty of Language in a Narrow sense (FLN), however, is arguably unique to humanity [Hauser et al., 2002]. The FLN is mostly an ability to understand and build recursive structures. Recursion, in turn, enables the

[2]This is sometimes called the nativist, innate or Chomskyan paradigm, the latter from Noam Chomsky who as been its major proponent. However, in this work, we will restrain ourselves to the denominations laid out in [Steels, 2011] as our work is in its continuation.

compositionality—i.e. creating new sentences by recombining learned examples—and the double articulation of language [3] [Smith, 2014].

Recursivity ensures complex languages which, nonetheless, children master relatively quickly and, more importantly, with a comparatively tiny subset of examples. To explain this success in face of the "poverty of linguistic stimulus", Noam Chomsky proposed that human brains are innately equipped with a Language-Acquisition Device (LAD) [Chomsky, 1964]. The LAD would be a cognitive unite in the brain that would have evolved biologically and that would be hard-wired for the understanding and production of language.

According to Chomsky, the LAD would provide humans with an endowed universal grammar that underlies all existing grammars. As this universal grammar would be part of the FLN, it would be recursive and so would any of its derivative. This enables to formalise any grammar as a set of recursive rules as in the minimal example below:
S → NP VP
NP → D NP
NP → A N
VP → V
N → box
A → blue
V → flies
D → the
Consequently, the sentence "The blue box flies" can be grammatically verified by backtracking the rules as follows:
The blue box flies → D A N V
D A N V → D NP V
D NP V → NP V
NP V → NP VP
NP VP → S
The fact that we can return to the root of this grammar (i.e. S) demonstrates that "The blue box flies" is grammatically correct. Generative grammars have had a substantial impact, including in computer science as they are widely used in the implementation of parsing algorithms to compile programming languages and they can also model finite state automata [Reghizzi et al., 2013].

A major argument supporting universal grammar and, therefore, the LAD and biolinguistics, is the existence of language universals (LU) [Chomsky, 1980]. LUs are

[3] meaning depends on two levels at the same time as (1) words are specific patterns of sounds and (2) sentences are specific patterns of words.

features or constraints that are found in the majority of languages across the globe. For example, no language expresses tense using noun affixes [Pinker and Bloom, 1990]. As we will see in Section 3.2.1, some LUs also express an implication between several features. LUs seem difficult to take into account without supposing that all humans are biologically (and identically) wired for speech following a universal grammar that dictates the presence of these LUs.

Despite this agreement between biolinguists, the latter are opposed by two antagonist views. The *adaptationist* view of biolinguistic [Pinker and Bloom, 1990], supports that the LAD has *"evolved through selection for the function it now fulfils"* [Kirby, 1999]. Conversely, the *exaptationist* view supports that the LAD would have evolved for other purposes and the fact that it enables the FLN would only be a side-effect.

3.1.2 Synthetic Models

Many synthetic models have been developed in order to study the evolution of language in long time-scales. Indeed, studying multi-agent simulations that can reproduce and interact at a very fast pace, informs researcher in ways that human observation can not. Moreover, robots offer the opportunity of grounded simulations as the complexity of the actual world (which could not be reproduced in an abstract simulation) plays an important part in the development of a language. Indeed, language must, for example, account for the difference of perspective between agents [Steels and Loetzsch, 2008] or be sufficiently powerful to discriminate an object against a continuous spectrum of stimuli [Steels and Loetzsch, 2012].

Examples of the linguistic models implemented on robotic platforms [Tuci, 2009, Tuci et al., 2011] use the framework of Evolutionary Swarm Robotics (see Section 2.1.2) as well as Evolutionary Robotics to show how language can evolve in co-dependence with performing actions. The latter [Tuci, 2009] was a categorisation task wherein groups of robots had to decide the direction of rotation of a revolving door according to perceptual cues that, like the robots, were distributed on both sides of the door. This only demonstrates that robots can evolve a simple signalling system to categorise the environment. Nevertheless, the latter [Tuci et al., 2011] showed that a uniquely linguistic feature such as compositionality can evolve through generations of single robots obeying to varying orders.

Agent-based computational models have also been proposed to simulate the evolution of language across several generations. In this model, the capacities of an agent, such as a neural network [Niyogi and Berwick, 1997], are encoded as a DNA string. These agents then reproduce to form a new generation by transmitting their

genes (according to Darwinian theory) and/or by teaching their language to their offspring.

Such models allows researchers to study the interplay between biological and cultural evolution such as in the Baldwin effect. The Baldwin effect is a well-known biological phenomenon that could support the *adaptionist* view of language. It states that skills acquired within a lifespan can be easier to learn for each new generation as a stronger predisposition to this skill becomes a selective advantage, which leads to new generations with a genome more adapted to this task [Baldwin, 1896]. In this explanation, babies born with a higher predisposition to a language could participate earlier to social activities and, thus have a higher change to survive which, generation after generation, would encode these predispositions into their genetic code, which would eventually have become the LAD.

Nonetheless, an agent-based model such as described above enabled researchers to test this hypothesis. In this model, genes are associated with an advantages for some features of language. Comparing different possible kind of influences (genes influence language, language influences genes–i.e. genes compatible with language are a selective advantage–, and combination thereof) over several generations showed that languages changes too rapidly for specifically adapted organs/instinct to appear because of the selective pressure. The LAD could not, therefore, have evolved because of a societal pressure to speak. Nonetheless, the Baldwin Effect could refine these biological modules once a complex language is used in a population.

This study does not entirely refutes an important role of biology in the formation of language. However, it clearly contradicts the *adaptationist* view, leaving only the *exaptationist* to fully support the biolinguistic paradigm. The shape of the LAD would thus have no particular evolutionary advantage. However, some mathematical modelling [Nowak et al., 2001] indicate that coherence might be a strong factor in the natural evolution of UG under selection pressure.

Similar computational models show that the transmission of a language from generation to generation can suffice to transform a weak learning bias into a given linguistic feature (i.e. a LU) with almost certainty [Kirby et al., 2007]. This learning bias could be biological but could also be induced by the constraints of the learning to the point, claim the authors, that culture may completely neuter innate bias in the resulting language or even weaken the genes that encode them (by reducing selective pressure). Finally, compositionality can also emerge without any reproduction but, rather, by repetitions of a teacher-learner cycle. Using the iterated model, it has been shown that the "poverty of linguistic stimulus", rather than being an argument for UG, can actually be responsible for the cultural emergence of compositionality. In fact, compositionality allows

learners to slip past this bottleneck (i.e. the subset of data they are taught) in order to learn the full language [Smith et al., 2003], without requiring biological adaptation.

The findings presented above strongly undermine the fundamental hypothesis of biolinguistics, as does the fact that even the existence of language universals is increasingly called into question as it may have resulted from ethnocentric observations [Evans and Levinson, 2009]. It is worth reiterating, however, that the impact of evolution on natural languages should not be entirely dismissed as other biological experiments have shown that even apes are not capable to manage recursive phrase structures contrary to infants [Fitch and Hauser, 2004]. In this section, we only demonstrated that cultural evolution is also an important part of the evolution of language and can explain many of its more complex aspects which tended to be obstructed by fully adaptationist approaches until recently. As we have just seen, simulated studies strongly imply that self-organisation can indeed induce many LU without requiring any biological explanation. This hypothesis will be developed in Section 3.2.1.

Despite the theoretical debate regarding biology as an explanation of linguistic complexities, its assumptions still can be (and have been) useful for artificial systems as they are highly compatible with one of the most common modern optimisation tools.

3.1.3 Applications to Artificial Systems

Due to its strong propensity towards an evolutionary robotics framework, it is not surprising that all previous work related to evolving communication in swarm of robots share the assumptions of biolinguistics. This is; however, also a weakness, as this strong focus on offline optimisation does not allow open-ended communication and, as we have seen in Section 2.2 constrains these works to developing mere signalling.

In the first work in this direction, small colonies of robots were evolved within a particular scenario that did not especially encourage communication [Floreano et al., 2007]. In this experiment, the robots were assigned a foraging task (i.e. find a food source in order to feed). However, the environment also hosted poison sources undistinguishable from food sources as both emitted red light. At the end of a classical evolutionary process, both kind of colonies (light equipped or not) had significantly improved their performance, especially in those equipped with lights wherein two types of signalling emerged in different populations. Indeed, agents either activated their blue lights near the food sources or they activated them near the

poison sources. This study also showed how, as in nature, suboptimal communication systems can be adopted and maintained as the individual can not afford to pass through a chaotic period in order to find a new optimum. Moreover, these results also demonstrate that communication with emerged meaning can provide an advantage to robots swarms in a typical scenario of swarm robotics.

Later works [Ampatzis et al., 2008] (see also Section 3.1.2 for [Tuci, 2009]) henceforth used ER to evolve signalling for other behaviours, expecting these signalling systems to produce more adaptable behaviours, especially when porting controllers evolved *in silico* to the real world. This outcome was successfully produced in a categorisation task wherein two, then four, robots had to collectively discriminate between two different environments and perform phototaxis or antiphototaxis accordingly [Ampatzis et al., 2008]. In the successfully evolved controllers, signalling had emerged (again, without any incentive in the fitness function) as a social and personal cue to switch from phototaxis to antiphototaxis (i.e. that the robots where in the second environment). Although the controller thoroughly resisted the reality gap, the authors emphasise that the uniformity of the controllers was most certainly a reason of its success as the robots did not have to deal with dishonest communication. Moreover, useful signalling systematically emerged in generations succeeding the first successes in the main task.

More recently, it has been shown that automatic modular design evolves more resilient communication than ER for such classical examples of swarm behaviours as aggregation (see Section 2.3), coordination[4] and categorisation [Hasselmann et al., 2018].

Evolutionary approaches to language have also been used for task-solving in abstract multi-agent settings with The Prey-Predator pursuit problem [Jim and Giles, 2000]. This problem involves five agents (one prey and four predators) moving on a two dimensional grid-world. The goal of the predators is to cooperate in order to surround the prey by all four sides at the same time, thus capturing it. Jim and Giles [Jim and Giles, 2000] used this pursuit problem in order to study the effectiveness of language as a cooperation scheme. We describe this more abstract implementation below because it allowed more comprehensive efforts and could well find applications with newer swarm robotics platforms such as ARK [Reina et al., 2017] or the kilogrid [Valentini et al., 2018].

In this experiment, besides choosing one of the four directions to move towards, a predator executes another action: sharing a message of l bits on a message board, accessible to all predators. Of course, this message is not always the same and

[4]In this setting, robots had to explore an arena wherein a dark spot was painted. The task was for all robots to stop as soon as possible after one of the robots had found the spot.

depends on the state of the agents, which is defined by the data it perceives: the range and bearing of the prey and the messages currently on the message board. So, in effect, for a predator, one turn consists of three actions:

1. Read all characters on the message board.
2. Taking these characters into account, choose and move towards a direction.
3. Write a new message on the message board.

with all predators executing their turns simultaneously (thus acting from the same messages).

The strategy of the predators (i.e. the decision they take depending on the message board and on their own sensory information) is homogeneous and remains constant during each scenario. This strategy is coded as a binary chromosome string. This chromosome string associates each combination of sensory information (range, bearing, and message board) to one move and one message.

The researchers also implemented a growing process which consists of taking a population of predators that have already reached an efficient language, increasing the length l of the messages, and then evolving them further. This process makes the language developed adaptive, which means that the language is capable of adapting itself to integrate new meanings or accommodate new individuals.

Eventually, this work shows that this communication scheme does help (as they produced the best results for this problem at the time) and that the time necessary to capture the prey decreases as the length of the messages increases. Moreover, the results showed that, for the same length, grown languages are as efficient as non-grown ones but that they are faster to evolve.

In parallel, the same problem was addressed using the Enforced Population Method (ESP), which consists of evolving the neurons of a neural network individually [Yong and Miikkulainen, 2001]. In this experiment, the prey actively tries to escape as it always chooses the direction opposite the nearest predator. Using ESP to evolve both communicating and non-communicating predators, they found that both performed as well although the communicating agents where slower to evolve and adjusted more poorly against preys exhibiting behaviours that were not considered during the training phase. However, they note that the communicating agents are more flexible (compared to non-communicating agents that each evolve a specific role) and that their unsatisfactory effectiveness is probably due to them switching role too many times as the scenario develops.

Other works recently studied the advantages procured by communication scheme for the pursuit problems [Goings et al., 2014], but none highlighted a definitive ad-

vantage of communication, even when the setting was especially designed to require communication [Rahman and Jain]. Overall, in the case of a pursuit problem, the evidences towards communication being an advantage seem contradictory, although, we can note that it helps for very challenging settings as the prey's behaviour as programmed in [Jim and Giles, 2000] make it more difficult to catch. Moreover, this survey showed that adaptive languages are faster to evolve and as efficient as non-adaptive ones.

The works cited above, simulated as well as embodied, all share the assumption that language or signalling evolve biologically, which is unsurprising as they are using the framework of automatic design available in swarm robotics and, more generally, optimisation. These implementations all share the same weaknesses as any work under the framework of evolutionary designs: they are computationally expensive, centralised and do not adapt well to new experimental conditions.

3.2 Cultural Evolution

Cultural evolution explains the complexity of language by self-organisation [Steels, 2011]. It supposes that no particular area of the brain is dedicated to language. Instead, specific forms of language would emerge through usage under constraints such as unreliable channels or limitations on memory and perception [Bates, 2003]. These forms are thus common, not because they are innate, but because they are among the only possible solutions to conveying a rich set of meanings onto these constraints.

To study this emerging phenomenon, evolutionary linguists use computational models with large quantities of interacting agents. These models do not require replacement nor reproduction but play out in a single open-ended experiment. This field has largely been influenced by Language Games (LG): a category of games aimed to study the way a population can self-organise in order to enable the emergence of languages that map words to meanings [Steels, 2011]. Moreover, the languages thus created are adaptive as they are capable to take new meanings into account (and actually emerged that way in the first place) and to include new agents (even if these agents already have a language).

LG are interesting from an epistemological perspective: They draw their origin from the philosophy of language, the study of how words and grammar acquire meaning, but they are used as a linguistic tool to explain the organisation of language. Linguistic and philosophy of language are very different fields of inquiry with few bridges between them and LGs are one of these bridges. Section 3.2.1 will revisit

LU and show that LG can be a linguistic explanation for them whilst Section 3.2.2 will delve into the origins of LG. Eventually, Section 3.2.3 will focus on the simplest instance of LG, which is the Naming Game, and highlight some common points of interests between the NG and swarm robotics.

3.2.1 Cultural Evolution of Language Universals

LUs are not just linguistic features that are found in every language. As a matter of fact, several LUs express a dependency between some of these features [Greenberg, 1963]. These dependency can be hierarchical [Kirby, 1999] and, thus, formalised as follows:

$$\forall L[(P_1(L) \rightarrow P_2(L))\&(P_2(L) \rightarrow P_3(L))\&...\&(P_{n-1}(L) \rightarrow P_n(L))]$$

These LUs are most often written by linguists as $P_n > P_{n-1} > ... > P_3 > P_2 > P1$ to express their relative height in a hierarchy of type, in the sense that P_n is true of many languages but P_{n-1} is found only in a subset of them and so on and so forth.

Vowel systems are one of these LUs [Crothers, 1978]. Indeed, the observation of more than 400 languages showed languages having the same number of vowels turned out to use resembling sets (among 37 existing vowels) and that, like BCT, larger vowel systems tended to contain all the vowels found in smaller systems.

These vowel systems actually follow a complex taxonomy and various implications are possible with different statistical probabilites (it is therefore a statistical rather than an absolute universal [Kirby, 1999]). However, we can illustrate this universal with languages using only three vowels, such as Classical Arabic, Inuktitut (one of the main Inuit languages), or Quechua. All three of these language only use /i, a, u/. These vowels are also used in the majority of languages with more vowels, such as Spanish, which has /a, e, i, o, u/. A tentative hierarchy of vowel systems [Crothers, 1978] is presented in Figure 3.1.

A language game that can explain these similarities is the imitation game [De Boer, 2000]. In this game, agents are equipped with an articulatory synthesizer, a module for calculating the distances between different vowels (according to human perception) and a repertoire for storing vowel prototypes. Then, two agents (among many) are selected randomly and start the game proper. The first agent (*the initiator*) selects a random vowel from its repertoire and utters it. The second agent (*the imitator*) then tries to imitate this vowel by uttering the closest likeness in its own repertoire. The initiator subsequently has to find the closest vowel to the one uttered by the imitator in its own repertoire, the goal being to thus find the initial vowel. Depending on the issue of previous games and on the success of the current

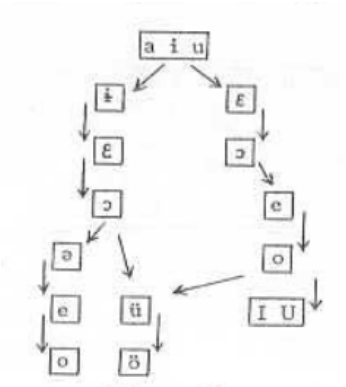


Figure 3.1: Crothers [1978]'s hierarchy of vowel systems.

one, both agents then either "merge" their vowels (they shift their vowel in the articulatory space towards the one they perceived) or add a new one. This protocol, coupled with some communication noise, causes the emergence of vowel systems that are strikingly similar to those found in actual human languages because the agents self-organise in order to produce vowels that are as distinguishable from each other as possible.

Another example of LU is the hierarchy of Basic Colour Terms (BCTs). According to surveys with 20 then 110 languages [Berlin and Kay, 1969, Kay and Regier, 2003]. BCTs are categories supposedly delimiting the colour spectrum consistently across every language, the categorisation being established by a word. A BCT also has a "focus": the colour value that corresponds the most to the BCT. In [Berlin and Kay, 1969], the authors argue that, even though languages do not all contain the same set of BCTs in their lexicon, these sets obey a hierarchy according to which any lexicon containing a lower-ranked colour term will also have every higher-ranked colour terms (e.g. the presence of red necessarily implies black and white). The hierarchy of BCTs is $black, white > red > green, yellow > blue > brown > purple, pink, orange, grey$ (note that green and yellow do not necessarily appear simultaneously but there is no universal order of precedence between them) [Kay and Regier, 2003].

For example, Darkinyung, an Australian language, only has two colour terms: *mining* (black) and *barag* (white), which they use, for any (resp.) dark/light colour. Additionally, Ibiobio (from Nigeria) uses four colours focusing on white, black, red and green but no example exist that uses, e.g. white, red, green and blue.

BCTs can also appear by self-organisation, as shown by the category game

[Baronchelli et al., 2010]. The category game is a variation of the guessing game in which only the speaker knows the topic and the hearer actually has to guess the topic amongst various objects [Steels, 2001]. The guessing game is relatively similar to the imitation game except, in this instance, the topic is exterior to the agents and the second agent, rather than imitating the first, has to point the topic it believes to be correct. In the category game, the topics that have to be guessed are values on a continuous scale (such as colours in the colour spectrum). Thus the creation of words for random topics drawn from this scale results in discrete categories that can be named. When agents perceive colours the way humans do (i.e. according to the Just Noticeable Difference scale), the category game indeed self-organises to create the BCTs found in human languages.

As explained in Section 3.1.1, the existence of LUs is one of the main evidence for the existence of the LAD, and thus, for the importance of biological evolution in the formation of language. However, as we have just seen, LGs show that LUs could have been single-handedly shaped by self-organisation around biological limitations, but without resorting to the LAD explanation. Therefore, LGs offer a different account of the role of both culture and biology in the development of language. In this account, biological evolution only provides us articulatory, perceptional and neuronal capacities and limitations to perform language. It is cultural evolution which enables the self-organisation of complex subsystems in language itself. Nevertheless, these studies are not the reason LGs were created as a philosophical tool.

3.2.2 Origins of Language Games

LGs were initially introduced by Ludwig Wittgenstein in his *Philosophical Investigations* [Wittgenstein, 1953]. In this work, he opposes what he calls the "Augustinian Conception of Language". Although this concept is a very wide umbrella, that imperfectly encompasses many intellectual and opposing currents, the Augustinians' core idea is that words are mere denominations of actual objects [Younes, 2016]. This view can be attributed to virtually all Western philosophers until then, from Plato to Gottlob Frege and even Wittgenstein in his *Tractatus* (an earlier work) [Rousseau, 1990].

The most straighforward example of an "Augustinian Conception of Language" is what Gottlob Frege called *psychologism*. *Psychologism* is similar to information-based definitions of communication as seen in Section 2.2. It imagines that each word has a straightforward link to an actual object, as if it were just a code.

Frege, who was one of the most immediate predecessor to Wittgenstein, tried

to break away from *psychologism*, which he viewed as the governing paradigm until then. Frege wanted to establish purely logical formalisms and remove any ambiguity in scientific ventures. This was the foundation of *analytical philosophy*. He tried to do so for mathematics (which eventually failed due to the so-called "Russel Paradox‘" [Irvine and Deutsch, 2016]) and then for language[5].

Frege realised that *psychologism* lacked rigour by remarking that, $A = B$ sentences referring to the same object B can have very different meaning (which *psychologism* did not permit). For example, the sentence "Harold Saxon is Harold Saxon" ($A = A$) is simply a tautology, it provides no information. Conversely, the sentence "Harold Saxon is The Master" ($A = B$) provides information as to the true (and malevolent) identity of Harold Saxon, even though "Harold Saxon" and "The Master" are just different encoding of the same person/object.

Consequently, Frege introduces an intermediate between the Sign (i.e. the word) and the Reference (i.e. the actual object referred to): the Sense [Frege, 1892]. To continue with our earlier example, the Sign "Harold Saxon", has the Sense of "The Prime Minister of Britain who replaced Harriet Jones"[6]. The Sign "The Master", however, has the Sense of "The renegade Time-Lord who'd get dizzy if he tried to walk in a straight line"[7]. Both, however, refer to the same person. Frege's theory can then be expanded to sentences, even subordinate ones. The goal, with this clarification, is to be able to ascertain whether a sentence is true or not by ensuring that every reference in it exists and that the stated relation between them is true. To make these verifications, Frege uses mathematical functions that denote propositions. For example, the proposition "Nyssa is asleep"[8] can be denoted $f(Nyssa)$, where $f(x)$: x is asleep [Younes, 2016]. This laid the foundations of propositional logic.

This system becomes exponentially complex with subordinates as implicit senses appear between each part of the sentence. Moreover, some references simply do not exist because they are fictional (as is actually the case of our first example). To answer this second problem, Frege draws a strict delimitation between a scientific use of language (to acquire or share knowledge) and an aesthetic use (to entertain). He calls this the presupposition of reference. Wittgenstein, however, sees this model as overtly idealistic. Indeed, Frege might have, with the presupposition of reference, removed the necessity for a Platonic World of Ideal (*cf.* Allegory of the Cave), but

[5]These efforts can be connected to generative grammars as described in Section 3.1.1. However, generative grammars formalise natural language by focusing on its structure, whereas analytical philosophy focuses on meaning. The former is the most effective [Stokhof, 2018].

[6] *"The Sound of Drums,"* Doctor Who, BBC, 5 Feb. 2008.

[7] *"The Mark of the Rani,"* Doctor Who, BBC, 2 Feb. 1985.

[8] *"Kinda,"* Doctor Who, BBC, 1 Feb. 1982.

he is still trying to fit language into the "perfect" box of logic, despite evidences that this is just not how language works [Younes, 2016]. As a result, Frege is still an "Augustinian" according to Wittgenstein.

It is worth noting that the "Augustinian Conception of Language" can be considered to epistemologically underlie computer science as it sets the basis of both Shannon's communication theory [Shannon, 1948] and propositional logic, which is at the centre of the theory of computability and complexity [Church, 1962].

Wittgenstein deliberately seceded with these kind of majestic edifices of logic (which, earlier in his life, he himself pushed even further). For him, the meaning of a word is not the actual object it refers to but simply its usage in a context.

He calls this context of usage a "language game" and defines it as *"consisting of language and the actions into which it is woven"* [Wittgenstein, 1953]. This is a break from his earlier, propositional view that envisioned language as a calculus rather than as a game. Wittgenstein explains that, though they encompass all human activity, these game can be extremely simplistic or very complex. They are all games only by virtue of a *family resemblance*, and not by a specific set of shared features. Therefore, imagining a language independently from a form of life is futile. As forms of life are collective, this introduces, for the first time a social and cultural aspect to the philosophy of language.

In Wittgenstein's view, explaining a name by pointing at the bearer of the name (cf. Figure 3.2), as earlier philosopher thought words got their meaning from, is simply one possible usage of the name in a specific context; a "naming game" of sorts.

3.2.3 The Naming Game

As we have seen in this section, evolutionary linguistics postulates that language acquires meaning through social activities. This opposes both the linguistic idea than the complexity of language is biologically imprinted and the philosophical idea that meaning is referring to actual objects or objective truth.

Evolutionary linguists propose to prove this by implementing specific game protocols into computer simulations or robotic supports. Many LGs have been developed so far (e.g. imitation, guessing and category games in Section 3.2.1) but the first and simplest LG to be introduced was the naming game (NG) [Steels, 1995], which will be presented in this section. Though it was originally developed to study how words were linked to meaning, it more generally illustrates how a norm can become shared among many agents and can thus inform our understanding of other social, economic and ecological phenomena [Steels and Loetzsch, 2012, Baronchelli, 2018].

Figure 3.2: Example of the AIBO game dialog between robot and a human. The robot performs image segmentation and tries to associate words spoken to the human (i.e. "ball") to the actual ball. For the interaction to be successful, the robot first has to be put in the right conditions by specific keywords. *Reproduced from [Steels, 2001] ©2001 IEEE.*

The NG is played by two agents, chosen randomly in the population, who must agree on a word to name a predetermined topic. Each agent has an individual lexicon, containing a list of possible words for each topic and a score of association between each word-topic couple. In order to reach their goal, the agents will take different roles that can change between each game; one will be the *speaker* and the other one will be the *hearer*. There are many variations of the NG, but it generally proceeds similarly to what follows:

1. *Initiation:* The speaker chooses a topic in the environment and introduces it in an extralinguistic way (e.g. pointing).

2. *Communication:* The speaker looks in its lexicon for words associated with the topic.

   - If it has at least one word, it chooses the word the most associated with the topic and tells this word to the hearer.
   - If it has no word, it adds a new random word to its lexicon and tells this word to the hearer.

3. *Reply:* The hearer hears the word.

   - If the word is already associated with the topic (through previous naming games with other agents of the population), its association score is increased and the hearer transmits the success to the speaker.
   - If the word was unknown, it is now associated with the topic with a base score and the hearer transmits the failure to the speaker.

4. *Confirmation:* The speaker receives the status of the game.

   - If it is a success, the association score between the word chosen and the topic is increased.
   - If it is a failure, the association score between the word chosen and the topic is decreased.

Variations of the NG are possible. For example, the word spoken by the speaker can be chosen stochastically (with probability proportional to the association scores) or, in case of success, the speaker and hearer can both drop all words associated to the topic except the one that just provoked the success (minimal naming game) [Loreto et al., 2010]. Furthermore, the set of possible words is theoretically infinite so we can assume that the probability that two agents invent the same word is

almost null. Therefore, we can reduce the environment to a single topic (and delete the *initiation* step) without loss of generality as the word-topic associations are completely independent [Baronchelli et al., 2006].

Regardless of these variations, the NG always results in the whole population agreeing on one word for the topic and suffices to show how self-organisation can result in a shared and efficient communication system without any generational transmission [Baronchelli et al., 2006].

Mathematical analyses of the NG, showed that the latter is characterised by a sudden pass from a phase of disorder (large quantity of synonymous) to a phase of order wherein one word breaks the symmetry and imposes itself steadily across the whole population [Baronchelli et al., 2008]. This dynamics follows the S-shape, or *snowball effect*, which fits actual linguistic data as well as logistic spreading models which can explain this pattern [Solé et al., 2010].

In a general model, if x_i is the number of agents knowing the word W_i, and R_i is the rate of learning W_i, the population dynamics of word spreading can be expressed as:

$$\frac{dx_i}{dt} = R_i x_i (1 - x_i) - x_i \tag{3.1}$$

which has two possible equilibrium points: $x_i^\star = 0$ and

$$x_i^\star = 1 - \frac{1}{R_i} \tag{3.2}$$

This means that $R_i > 1$ is necessary for x_i to continue growing, i.e. that under this threshold of learning rate, the population x_i will die [Solé et al., 2010]. In the NG this happens when a specific word W_j starts to win more games, which means that more lexicons contain only W_j and that the other words will fail more often and stop to propagate.

Furthermore, different update schemes can be envisioned for the lexicon. Indeed, both agents can update their lexicon after the game but, alternatively, only one of the hearer/speaker could make the update. Mathematical analysis of the spread of the words with these different schemes actually demonstrated that the speaker do not actually need to update its lexicon after the game to conserve a similar S-shaped dynamic [Baronchelli, 2011]. As a consequence, it appears that the hearer does not need to communicate the success or failure of the game which, in turns, renders the *confirmation* step of the NG useless. Eventually, it means that the naming game can be played with several hearers at once as the speaker can just broadcast its chosen word without needing to wait for any answer, as per Algo. 1. Moreover, in this variation of the naming game, agents with more neighbours have a higher

chance to be "elected", which means that the word they initially promote has a higher chance to be the chosen name for the predetermined topic, at least in static networks [Baronchelli, 2011].

LGs have been directly compared to ants foraging [Nolfi and Mirolli, 2009, p. 225] and bees' building of honeycombs [De Boer, 2000] as all three cases form macroscopic structures through the self-organisation of locally interacting agents. It is indeed striking how similar the epistemology of swarm intelligence (and, by extension swarm robotics) and of evolutionary linguistics are. It thus seems that both field could inform each other. Seminal work [Trianni et al., 2016] in this direction implemented the NG in a swarm of kilobots ("low-cost robot designed to make testing collective algorithms on hundreds or thousands of robots accessible to robotics researchers" [Rubenstein et al., 2012]). In this study, whose main aim was to investigate whether the dynamics of the NG change when implemented on agents that are mobile and embodied, the swarm was not engaged in any collective behaviour. Rather, the robots were all executing an individual random walk. This implementation benefited from the hearer-only update scheme (as discussed above) as kilobots, and swarm bots in general, are not build for directed communication; their only possible transmission scheme is to broadcast. The major conclusion from this work is that the embodiment of agents playing a NG reduces the strain on their memory as the collision between transmissions results in a loss of data (and thus the abortion of a part of the games). Conversely these collisions lead to the formation of aggregates of robots that do not interact much, leading to slower convergence than with simulated agents. Despite this, the algorithm still makes the swarm converge to a single word.

Still related to swarm robotics, LGs were mentioned as an interesting lead for foraging [Miletitch et al., 2018] to, e.g. assign labels to profitable resources, and develop a language grounded in their activity. To the best of our knowledge, no result as yet come from this suggestion. Nevertheless, this is a commendable commitment as it returns to the core of Wittgenstein's argument that LG are language interwoven with an activity, whereas existing implementation of LG still focus on very "linguistic" tasks such as naming or categorising.

Chapter 4

Methodology

Communication is of paramount importance to swarm robotics and, as we have seen in Chapter 3, it can emerge culturally from local interaction in order to be adaptive, just like any other swarm behaviour. This adaptiveness is necessary to efficiently tackle new situations in an open-ended world [Steels, 2003].

In Chapters 2 and 3, we highlighted self-organised aggregation and the naming game (more specifically, the hearer-only minimal naming game, henceforth MNG) as fruitful use cases for, resp., swarm behaviours and cultural evolution. Consequently, the remaining of this thesis aims to propose a framework to merge swarm behaviours and cultural communication, taking self-organised aggregation and the MNG as examples.

Both behaviour-based (Section 2.1.1) and automatic (Section 2.1.2) design methodologies have been tackled on self-organised aggregation (see Section 2.3). However, as the main argument of this thesis is to propose a methodology for designing communication that relies on self-organisation rather than generational transmission, it is only natural that our methodology will lean towards behaviour-based design.

In Section 4.1, we will introduce two models of aggregation without shelter, which were especially designed to better study the influence the MNG could have on them. Then, in Section 4.2, we will propose two ways to link the aggregation with the MNG.

4.1 Parametric Aggregation

The most used behaviour-based approach for self-organised aggregation is the one taking inspiration from cockroaches [Deneubourg et al., 2002, Jeanson et al., 2005,

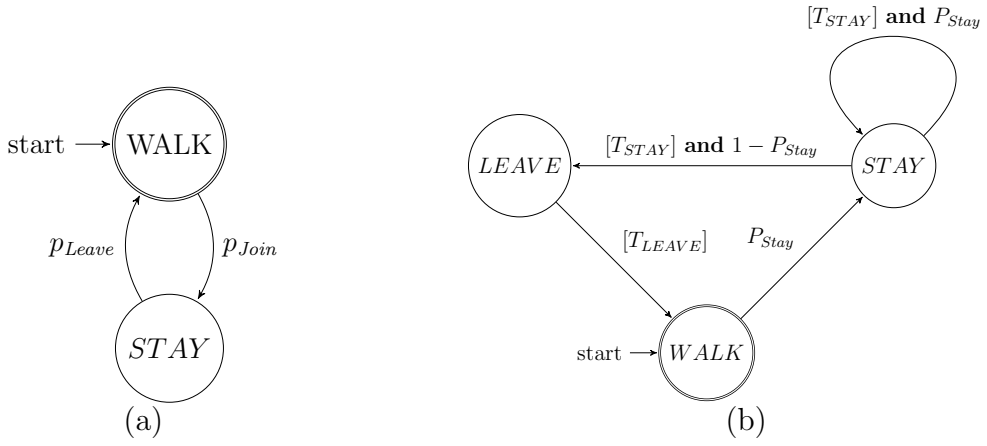


Figure 4.1: (a) Base PFSM of our aggregation controller. (b) PFSM of our aggregation controller with linear transition functions.

Garnier et al., 2005], which is also called *probabilistic aggregation* and can be used both for aggregation with and without shelters (see Section 2.3 for more details). However, in aggregation without shelter, the success and quality of aggregation is highly dependant on the encounter probability of agents, which is itself dependant on a high quantity of parameters such as the communication range, the robots' speed, the density of the population, etc. [Correll and Martinoli, 2007].

We designed two parametric aggregation controllers whose internal parameters can counteract external conditions in order to be able to reach effective aggregation whatever the setting. Both controllers have the same base. In order to design our PFSM, we started with an extremely simplified modelling of the cockroaches' behaviour, similar to [Correll and Martinoli, 2007, Bayindir and Sahin, 2009]. In this model, the agent's behaviour is independent from one's location in the environment (as opposed to the cockroaches' differing behaviour depending on whether they roam in an open area or follow a wall) and unimodal, which halves both the quantity of states and transitions. Consequently, as shown in Figure 4.1a, our automaton had only two states at that point:

1. $WALK$: the robot explores the area following a random walk.

2. $STAY$: the robot stops.

The difference between the two controllers lies mainly in their transition functions.

The linear transition function and its implications will be introduced in Section 4.1.1. The exponential transitions functions will be presented in Section 4.1.2.

4.1.1 Linear Transition Function

In order to make our initial controller parametric, we devised a simple linear probability function as expressed in Equation 4.1.

$$p_{transition} = min(1, p_{base}.n) \tag{4.1}$$

wherein n is the quantity of neighbours in the $STAY$ state and p_{base} is a parameter to set according to the aggregation strength desired and to the density of the population.

This equation determines the probability to join/stay in an aggregate and, thus, its complementary $(1 - p_{transition})$ gives the probability to leave the aggregate. The main strength of this function is that its single parameter (p_{base}) is instinctively understandable: $g = 1/p_{base}$ is the minimal quantity of neighbours necessary to make sure a robot stays in its place[1]. Any quantity lesser than that leads to a proportionately less solid position.

However, Jeanson et al. [Jeanson et al., 2005] observed that the probability to leave an aggregate is an order of magnitude weaker than the probability to join it (this is most obvious in Correll and Martinoli's averaged model [Correll and Martinoli, 2011]). In order to respect this necessary difference whilst keeping a single probability equation, we introduced a period of time T_{STAY} that must elapse before each trial to leave the $STAY$ state by a robot in that state.

Finally, the robots need to be able to leave an aggregate without risking to join it back immediately in the next time-step, or they would then never really leave as, again, they can join at any time-step but only try to leave once every T_{STAY} time-steps. Consequently, we introduced a third state in our PFSM, $LEAVE$, which is entered by the robots leaving the $STAY$ state (according to the transition rule exposed earlier) and which is left deterministically after a time period of T_{LEAVE} time-steps.

The resulting three-state PFSM of this controller, which is somewhat similar to [Soysal and Sahin, 2005], is illustrated in Figure 4.1b. Due to the presence of waiting periods (i.e. T_{STAY} and T_{LEAVE}), this approach can be slow. For this reason, we designed a second set of exponential transition functions that would not require these periods by being closer to the original cockroach model.

[1]Obviously, this means that, for the aggregation to perform optimally, g should be greater or equal to the maximum quantity of neighbours perceivable by a robot at any point. Otherwise, some aggregates could potentially become indestructible.

n	p_{Join}	p_{Leave}
0	0.03	n.a.
1	0.42	1/49
2	0.50	1/424
3	0.51	1/700
4+	0.51	1/1306

Table 4.1: Averaged probability to join and leave a cluster as computed by Correll and Martinoli [2007] from observation of gregarious arthropods [Jeanson et al., 2005].

4.1.2 Exponential Transition Functions

In the present case, we used two transition functions. The first, p_{Join} expresses the probability to transition from $WALK$ to $STAY$ while the second, p_{Leave} handles the transition from $STAY$ to $WALK$. This is similar to a previous model [Correll and Martinoli, 2007] that averaged the original bimodal distributions [Jeanson et al., 2005] into a single table reproduced in Figure 4.1. However, here, we propose a parametric variation of this model by fitting the probability table from Figure 4.1 through probability functions with parametrisable steepness, allowing to strengthen or weaken the agents' alignment and dispersion at will.

For both p_{Join} and p_{Leave}, we used exponential decay functions of the form e^{-qn} where q is a parameter. p_{Join} was then modified to fit some requirements of this controller. In particular, p_{Join} need not be null when $n = 0$. Indeed, as n only counts staying robots, if $p_{Join}(0) = 0$, no robot would ever stop. As a result, p_{Join} can be expressed as in Equation 4.2.

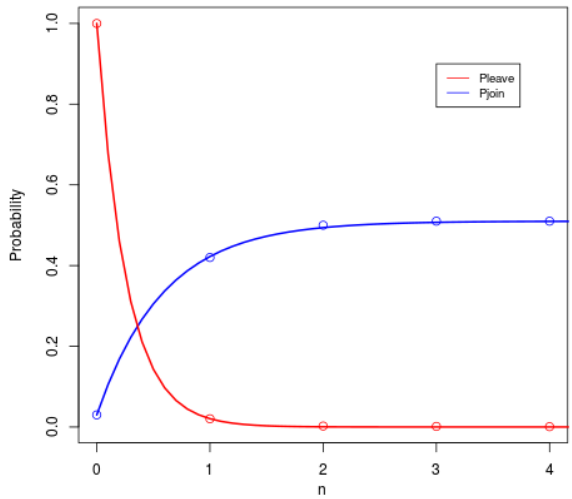
$$p_{Join}(n) = \epsilon + \rho(1 - e^{-an}) \tag{4.2}$$

In this equation, ϵ is the base join probability and ρ "squeezes" the function so that $p_{Join} \leq 1$. a is a parameter that handles the strength of the alignment. Indeed p_{Join} becomes steeper as a increases.

p_{Leave}, on the other hand, is a straightforward exponential decay function:

$$p_{Leave}(n) = e^{-bn} \tag{4.3}$$

Here, b handles the strength of the dispersion as p_{Leave} becomes steeper when b increases, and thus dispersion weakens. These functions can easily be fitted to the baseline values from Figure 4.1, with ϵ and ρ manually set to respect the limits of Table 4.1 (i.e. $p_{Join}(0) = 0.03$ and $\lim_{n\to\inf} p_{Join}(n) = 0.51$) and a and b fitted



$\epsilon = 0.03$
$\rho = 0.49$
$a = 1{,}70188$ (RMSE $= 0.003549$)
$b = 3.88785$ (RMSE $= 0.001262$)

Figure 4.2: Plotting of P_{join} and p_{Leave} according to (resp.) Eq. 4.2 and 4.3 and parametrised with the values for ϵ, ρ, a and b that best fit the values from Table 4.1 [Correll and Martinoli, 2007] (circles).

mathematically. The fit between these functions and the baseline are shown in Figure 4.2 and feature a remarkable matching.

This controller was first introduced in [Cambier et al., 2018]. Furthermore, we also developed a three-state automaton using Equations 4.2 and 4.3 in order to achieve aggregation with shelters [Firat et al., 2018].

We can broadly understand the effect of parameters a and b as strengthening (increase a or b) or weakening (decrease a or b) the cohesion of a cluster. The difficulty of self-organised aggregation is, of course, to find a trade-off between these two forces.

4.2 Integration of the Hearer-Only Minimal Naming Game

Self-organised aggregation and the MNGs are both collective decision-making processes [Valentini et al., 2017, Trianni et al., 2016]. These processes do not happen on the same time-scale nor progress according to identical conditions as the MNG depends mostly on the quantity of interactions whilst aggregation depends on the quality of the clusters encountered (i.e. size, concentration, etc.).

Consequently, integrating them within each other can provoke the emergence

of new behaviours as both processes will affect each other. This will be shown in Section 4.2.1, which proposes an approach to making these processes interact bidirectionally. Then, Section 4.2.2 shows that interactions between these dynamics can reinforce them as the MNG closes a positive-feedback loop that automatically selects aggregation parameters.

4.2.1 Bidirectional Interaction

Making the MNG interact with an aggregation controller is not straightforward. Indeed, we can not just run the MNG independently from the aggregation as, then, if the aggregation would impact the MNG (because the latter's dynamic is dependant upon the shape of the communication network), the MNG would change nothing to the aggregation.

Thus, we have to find a way for the MNG to affect the aggregation. To this end, we decided to articulate the MNG around the PFSM's $STAY$ state, as the agents should eventually converge to this state for the aggregation to succeed. Consequently, our modifications to the model were twofold:

- Agents in the $STAY$ state are **speakers**.
- The transition functions now depend on the quantity s of MNG won as an **hearer** in this time-step instead of the quantity n of agents.

The speaking and hearing processes are described in pseudocode in Algo. 1.

For example, $p_{transition}$ from Section 4.1.1 is now expressed as per Equation 4.4.

$$p_{transition} = min(1, p_{base}.s) \tag{4.4}$$

with s as the quantity of successful MNG in a row during the last time-step. The MNG is played only with the neighbours in the $STAY$ state.

Although the exact consequences of these interactions are studied in Chapter 5, we can already predict a few features of the new behaviours they will create.

Within a time-step, an agent plays the MNG in a random order with every single word received. If all the words received are identical (as it will often happen within an aggregate), two cases can happen. In the first case the *hearer* already knows that word, all the MNG will therefore be successful and $s = n$. In the second case, wherein *hearer* does not know the word, the first MNG will be a failure but the word will be added to the lexicon and all subsequent MNGs will be successes so $s = n - 1$.

If even one word received is different from the others, the game will be lost at least once. In that case, s will be reset to 0 (as s is the quantity of subsequent

wins). This means that a single mismatched word can, depending on its (random) place in the list of words, dramatically change the final value of s. For example, if the mismatch word is at the start of the list, the first game will be a failure but all the games following will be successes so $s = n - 1$. This is the best case scenario. However, if the mismatch word is at the end of the list, then s is reset at the very end too and all the wins are discarded: $s = 0$. Generally speaking, any failure will discard every prior victory in the time-step. Consequently, a single agent with a differing word can scatter a strong aggregate with a non-zero probability.

Moreover, as the MNG requires a minimum number of interactions before a success, initial clusters will not be able to form immediately, which will thus delay the start of the aggregation process. However, if the MNG has enough time for the whole swarm to converge to a single word, then this model would behave exactly as an MNG-less aggregation as the MNG thus becomes a straightforward signal of agents in the $STAY$ state.

4.2.2 Circular Interaction

With bidirectional interactions, we made the MNG and the aggregation affect each other. This interaction interfered with the normal process of both dynamics to create new behaviours. In the present section, however, we propose circular interactions, i.e. the dynamics reinforce each other. The key of this positive-feedback loop is to use the words of the MNG as the parameters (e.g. a, b and ρ) of the aggregation. In this scenario, the agents are heterogeneous as their controllers' parameters are free and evolve culturally according to local interactions.

The positive feedback-loop is articulated around three points:

- Aggregation parameters are set according to the values encoded in a word selected from the lexicon.
- Communication is noisy.
- Only robots in $STAY$ are speaking (all robots are hearing).

Designing an effective positive-feedback loop required an in-depth understanding of the behaviour aimed at. We started with the effect of cohesion in self-organised aggregation which, as we have already said in Section 4.1.2, needs to be regulated very accurately. This is the reason why, in both our controllers, we have made this feature parametric. Indeed, if cohesion is too weak, no durable cluster will form. Conversely, if cohesion is too strong, the agents will aggregate in several static clusters that will never break and, thus, never join in a single aggregate. Moreover,

in that case, agents will form sparser clusters because, as they need few neighbours to stay, they will stop at the first occasion rather than exploring further to find more densely populated areas. Following the above observations, we formulated the following premise, wherein, by analogy to graph theory, the degree of an agent is its number of neighbours:

Premise 1 *Agents with (near) optimal aggregation parameters have a higher degree, on average, than agents with suboptimal parameters.*

With the role of cohesion clearer, we continued to design our positive-feedback loop by studying the characteristics of the MNG. Indeed, as we have seen in Section 3.2.3, in addition to allowing broadcasting, the major difference between the MNG and other variations of the naming game is that agents with more neighbours (i.e., higher degree) have a higher chance to be "elected". Therefore, we formulated a second premise:

Premise 2 *Words promoted by agents with higher degree propagate more on average.*

From Premises 1 and 2, we concluded that words promoted by robots with (near) optimal aggregation parameters should propagate more on average. Therefore, by using words from the lexicon as values of the parameters (encoded in a bit string), we closed a positive feedback-loop (illustrated in Figure 4.3) whereby better parameter settings propagate more and, as they are shared by new robots, propagate even more. The parameters can be set according to the latest winning word (as we will see in Section 6.1) or to a random word from the lexicon (see Section 6.2).

Thus the encoded values of the parameters represent the "meme" in a cultural evolution process. An essential component of any evolutionary processes are mutations, which introduce the needed novelty required to search for optimal parameters. Mutations also exist in language due to the noise inherent in sound-based messaging [Steels and Kaplan, 1998] and can also contribute to its self-organisation [De Boer, 2000].

To implement mutations, we added noise to the messages broadcasted by the speakers as described by Shannon in his Communication Theory [Shannon, 1948]. The use of this model of noise is rather fitting as Shannon's model of communication (messages passing through a noisy channel) can represent biological mutation [Nolfi and Floreano, 2000] as well as a constraint for language [Nowak and Komarova, 2001, Steels, 2005, Lyon et al., 2007]. In our model, each bit of any message received has a probability m to flip.

Besides being able to generate new parameters in order to explore the solution landscape, mutations have another impact on our model. As mentioned above, the

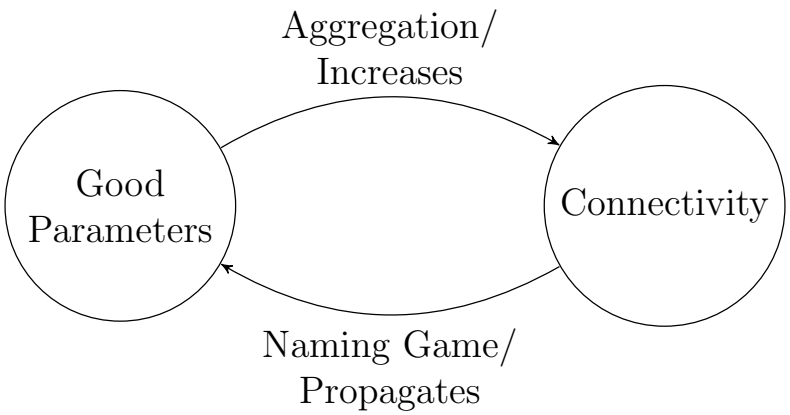


Figure 4.3: Positive feedback-loop

MNG always converges to a single solution, although in some cases it can take a very long time. Therefore, variability within the population disappears as different words are replaced by the most common one, meaning that sparse mutations are not sufficient to perturb this equilibrium. Furthermore, convergence in the MNG is a rather fast process [Baronchelli et al., 2006] in well-mixed populations, meaning that the evolutionary process can stagnate after a very short exploration of the solution landscape, granting too short time for aggregation to take place. To solve this problem, we could allow the robots to play the MNG at a lower pace. However, mutations are a much better solution because, by creating new words during the game, they increase failed games by an order of $l.m$ per time-step (a message is l bits, each with a probability m to flip) and therefore greatly slow the MNG down. In other words, mutations allow the MNG convergence to happen in a time-scale more compatible with the self-organised aggregation process.

Finally, in line with Section 4.1.1, and because Premise 2 comes from observations on static networks, we made sure that only robots in the $STAY$ state can become speakers. In preliminary experiments, we used s as the variable in the transition functions (as in Section 4.1.1). However, as we'll see in Section 6.1.2, these preliminary experiments revealed that s was incompatible with communication noise and we thus reverted to using n. As counting the number of successful MNG is not a necessity in that case, the agents only play a single, random, MNG per time-step, in order to reduce the pace of the MNG. In this latter version, we also initialise all lexicons at the start of the experiments (rather than waiting for robots to speak) in order to sample a larger quantity of initial parameters. The differences between

	Bidirectional	Circular v1	Circular v2
Speakers	Agents in $STAY$	Agents in $STAY$	Agents in $STAY$
Hearers	Every agent	Every agent	Every agent
Word creation	Speak & lexicon=$\emptyset$	Speak & lexicon=$\emptyset$	At initialisation
#MNGs per step	All	All	1 (random)
Variable	s	s	n
Parameters	Fixed	Winning word	Random word
Noise	No	Yes	Yes

Table 4.2: Summary of the features of the bidirectional and circular interactions. Circular is presented in two versions as v1 was used for preliminary experiments (in Section 6.1), which enabled us to develop v2: the definitive version (Section 6.2).

these two version and also the bidirectional link are summarised in Table 4.2.

In the end, circular interaction creates an on-line and embodied cultural evolution of aggregation controllers. Although, as we will see in Chapter 6, this process does not actually *optimises* the parameters (but still produces extremely resilient controllers), it introduces several components that are comparable to genetic optimisation (as presented in Section 2.1.2.

Lexicon The lexicon is a list of possible words. Each word encodes a specific parameter setting. This is comparable to a genetic pool in biological evolution.

Speaking In traditional self-organised aggregation, staying robots send a signal to their neighbours when they are stopped so that they can be counted. In our algorithm, the robots use this signal to transmit a word selected randomly from their lexicon (mating).

Hearing When a robot receives a word, either of two things can happen. If the hearer does not already know the word, it adds it to its lexicon (i.e. collection of cultural material). If the hearer already knows the word, it reduces the lexicon to this single word. As the words received are from robots that successfully aggregated (i.e. stopped), the lexicon should converge towards words that enable aggregation (selection). The aggregation parameters are set according to a word from this lexicon (replacement).

Communication noise Stochastic noise forces the exploration of new parameters (mutation).

This model was originally proposed in [Cambier et al., 2018], although it has since received several improvements. The exact effect of this model of interaction, as well as the dynamics of this culturally evolving aggregations will be studied in Chapter 6, but we can already see that the self-organisation in this approach emerges from existing constraints as is the case in natural languages (see Section 3.2.1).

Algorithm 1 Minimal Naming Game

```
procedure NAMING GAME
    function HEAR(word)
        if word inside Lexicon then
            Lexicon.clear()
            Lexicon.add(word)
            return true                                  ▷ Successful Game
        else
            Lexicon.add(word)
            return false                                 ▷ Unsuccessful Game
    function SPEAK
        if Lexicon == ∅ then
            Lexicon.add(generateRandomWord())
        word ← Lexicon[rand()]
        Broadcast(word)
```

Chapter 5

Group-size Regulation in Self-Organised Aggregation through the Naming Game

Self-organised aggregation and the MNG are both self-organised behaviour that can be played simultaneously and affect each other. Our objectives in this chapter are twofold. Firstly, we will investigate the influence of the differing time-scales between self-organised aggregation and the MNG on the dynamics of self-organised aggregation. Secondly, we will aim to understand the effect of a dynamic swarm network on the spread of words in a MNG. Differently from [Trianni et al., 2016], the aim here is to study what happens when the swarm is undergoing the dynamics dictated by a collective, coordinated behaviour, rather than engaged in an individual random walk.

To meet these objectives, we will study the dynamics of both processes when their interactions are bidirectional, as presented in Section 4.2.1. In order to generalise our observations, we will use this scheme on controllers using linear (see Section 4.1.1) and exponential (Section 4.1.2) transition functions. These observations will also be performed with different experimental setup. Section 5.1 will present experiments with a linear transition function on a physics-based simulator and section 5.1 will focus on experiments with exponential transition functions on a conceptual grid-world simulator. Our conclusions for this study will be exposed in Section 5.3.

5.1 Linear Transitions

In this section, we present published results [Cambier et al., 2017] on the study of bidirectional interactions between the MNG and a probabilistic aggregation controller using linear transition functions. Section 5.1.1 will detail our experimental setup and section 5.2.2 will discuss our results.

5.1.1 Experimental Setup

The algorithm described in Section 4.2.1 can help us answer two questions: "How does the MNG influence aggregation?" and "How does aggregation influence the MNG".

To this end, we implemented our algorithm on a simulated version of the foot-bot, a version of the MarXbot [Bonani et al., 2010] which is a differential wheeled robot. As MarXbot are equipped with proximity sensors, we implemented the random walk as walking in a straight line until encountering an obstacle and then turning according to a uniform random angle [Nouyan and Dorigo, 2006, Roduit, 2009]. This implementation thus performs obstacle avoidance as well as a brownian motion. Besides the wheels, the foot-bots involved in the experiments were equipped with IR sensors in order to detect obstacles and with the range and bearing module developed for the swarmanoid project [Dorigo et al., 2013]: a combination of four IR sensors permanently rotating to make 360° scans and allowing to exchange data via IR light. This module is very suitable for the setting of this work as infra-red communication suffices to emulate the interactions between cockroaches [Garnier et al., 2005, Correll and Martinoli, 2007] and the possibility to exchange (a limited amount of) data is useful to play language games. We studied the behaviour of these robots under our model by running several experiments in a **6x7.5m arena** on the ARGoS simulator [Pinciroli et al., 2012], which, as shown in Figure 5.1, offers the possibility to visualise a range of information as well as the simulation itself.

The experimental design that we used is described in Table 5.1. We varied only two parameters: the population size N and the p_{Base} probability. Moreover, the experiments were run with two versions of our algorithm: Vanilla (game-less, as presented in Section 4.1.1) and NG (bidirectional interactions with the MNG, as presented in Section 4.2.1). As T_{STAY} and T_{LEAVE} obviously influence alignment/dispersion, their values were set, after some parameter tuning, to (resp.) 200 and 50 in all experiments. We performed 20 independent runs for each parameter configuration in order to have statistically meaningful results.

The experiments were stopped when the swarm stabilised. The condition we used

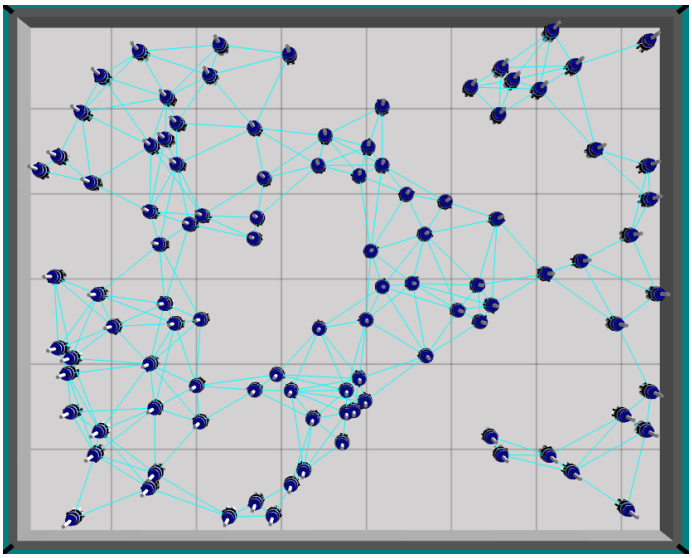


Figure 5.1: A picture of the ARGoS simulator in the course of an experiment with realistic footbots. The simulator can display extraneous information such as ongoing communications.

Table 5.1: Table of Experiments

Version	Label	NG?	N	p_{Base}	Runs
Vanilla	s20p14	No	20	0.14	20
	s20p17	No	20	0.17	20
	s20p20	No	20	0.20	20
	s100p14	No	100	0.14	20
	s100p17	No	100	0.17	20
	s100p20	No	100	0.20	20
NG	s20p14	Yes	20	0.14	20
	s20p17	Yes	20	0.17	20
	s20p20	Yes	20	0.20	20
	s100p14	Yes	100	0.14	20
	s100p17	Yes	100	0.17	20
	s100p20	Yes	100	0.20	20

to determine whether stabilisation had happened or not was that every individual remain still for more than 600 time-steps. With T_{STAY} set at 200, it means that **every** robot has decided to stay three times in a row. However, it is possible that stabilisation never happens, in which case the experiments were stopped after 1 000 000 time-steps (100 000 seconds).

To study the impact the MNG and an aggregation behaviour have on each other, we manually analysed the final state of each run and we computed two quantities: the amount of aggregates and the amount of words in the game. Obviously, to evaluate the former quantity, one has to define an aggregate. For the purpose of this analysis, we viewed the final state as a graph where each node is a robot and distances lesser than the range and bearing's reach are edges. With these notions, we defined an aggregate as a set of nodes where each node is linked to any other node by at least two paths. With this definition, elongated aggregates are possible but chains of robots cannot bridge aggregates together nor can be considered aggregates by themselves. Moreover, if an experiment stopped because of the time constraint and if less than 90% of the robots were part of aggregates, the aggregates were not counted as they were deemed unstable. The quantity of words in the game is much simpler to count as it is straightforwardly the size of the set of all the single-word lexicons.

Finally, following [Gauci et al., 2014], we computed the dispersion of the aggregates using the second moment, or variation, of the robots' positions. Using p_i as the position of robot i (among n in the aggregate), and $\bar{p}$ the centroid of these positions,

$$\bar{p} = \frac{1}{n}\sum_{i=1}^{n} p_i \tag{5.1}$$

the second moment of the robots is given by:

$$v = \frac{1}{4r^2 n}\sum_{i=1}^{n} ||p_i - \bar{p}||^2 \tag{5.2}$$

where r is the radius of a robot. As the robots are not mere points and occupy space (wherein other robots can not fit), $4r^2$ normalises v to render it independent of r. As aggregates may have different sizes, we also added n in the denominator, which normalises v with regard to the size of the aggregate.

5.1.2 Results

The data retrieved from our experiments are presented in Fig. 5.2. These plots were computed from our counts of the quantities of aggregates and words for each run

of each experiment. The dots represent the mean number of aggregates/words by experiment and the error bars represent the standard deviation. Consequently, a shorter bar indicates that the final outcome of an experiment is consistent. Longer bars indicate that the outcome vary more widely.

The first observation to make is that, in small populations ($N = 20$), low values of p_{Base} are often insufficient to secure the emergence of aggregates. Nevertheless, with a sufficiently high value of p_{Base} (higher than 0.17), the NG version forms a single aggregate with a quorum of 90%.
We should note that, as the mean quantity of aggregates remains near one, the Vanilla version has a better success rate than the NG version when p_{Base} is lower or equal to 0.17. However, as the count of the words shows, the NG-swarms consistently converge on a single word. Therefore, as explained in Section 4.2.1, $s = n$ and they will eventually behave identically to the Vanilla-swarms. This means that, given more time, they would reach the same success rate. In any case, with the NG algorithm, a swarm of 20 robots *displays a normal aggregation behaviour* as the impact of the stay probability on the efficiency of aggregation is well-known [Soysal and Sahin, 2005].

However, with a larger population ($N = 100$), the results with NG are more surprising. Interestingly, the mean quantity of aggregates neatly increases with p_{Base}. As the error bars show low deviations, this quantity is also relatively consistent from run to run. Thus, with the right value, traditional aggregation in a single aggregate is still possible but the algorithm also allows the experimenter to *divide the swarm in a configurable quantity of aggregates*.

Furthermore, we observe that, apart from the cases where no aggregation happened, the quantity of words in the game remains extremely close to the quantity of aggregates. This correlation is made even clearer with the visual examples that can be seen in Fig. 5.3. We can see that each aggregate has its own word–which can act as a label–as if different MNGs were played in each aggregate. This is incidentally also the case in small swarms but less ostensibly as they end up with one or zero aggregates and a single word. Consequently, the NG algorithm also *labels each robot of an aggregate with an identical word*.

Finally, looking at the average dispersion of the aggregate (or of the whole population in cases no aggregate formed) in Table 5.2, we can see that, in small populations ($N = 20$) with Vanilla, the variation of the positions of the robots (as computed from Eqs. (5.1) and (5.2)) is lower than the equivalent experiments with NG, which means that the aggregates are less dispersed with Vanilla than with NG. However, this difference is only significant with the *s20p17* experiment, which can easily be explained by having a look back at Fig. 5.2. Indeed, as the proximity of the mean quantity of aggregates to zero shows, the NG version of *s20p17* often failed to aggregate in

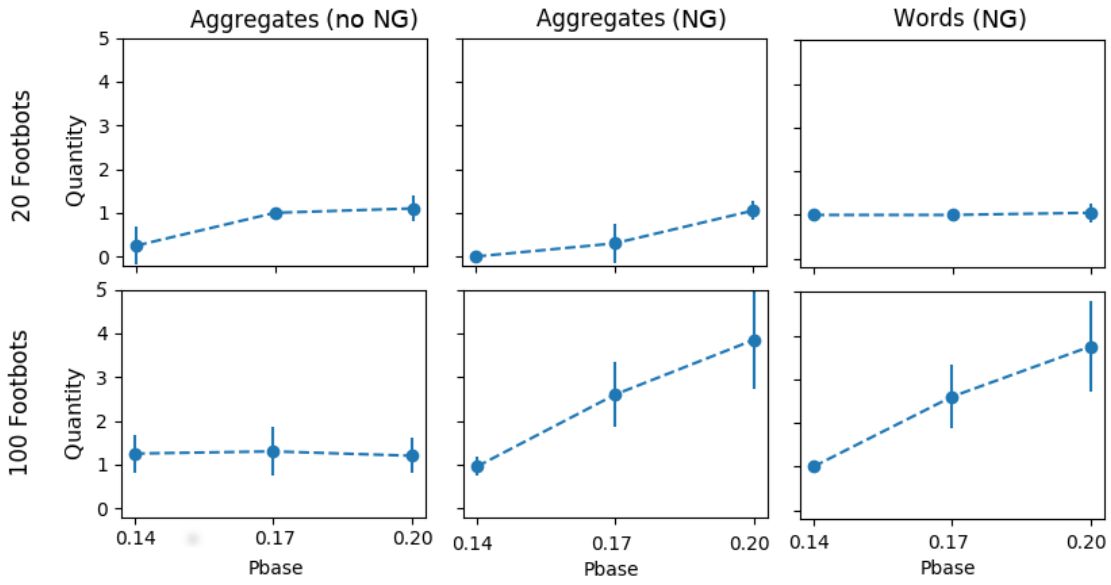


Figure 5.2: Mean (dot) and standard deviation (bars) of the quantity of aggregates in vanilla/NG aggregation (resp. left/middle) and of words (right) in stabilised swarms of 20/100 (resp. top/bottom) robots or after 100 000 seconds. These charts show that the Vanilla algorithm displays a normal aggregation behaviour as, with appropriate p_{Base} [Soysal and Sahin, 2005], the robots consistently gather themselves in a single aggregate. Moreover, we observe a visual correlation between the final quantity of aggregates and the final quantity of words in the populations playing a MNG. Finally, in cases where the swarm does not stabilises (0 aggregates), the whole population still converges on one word.

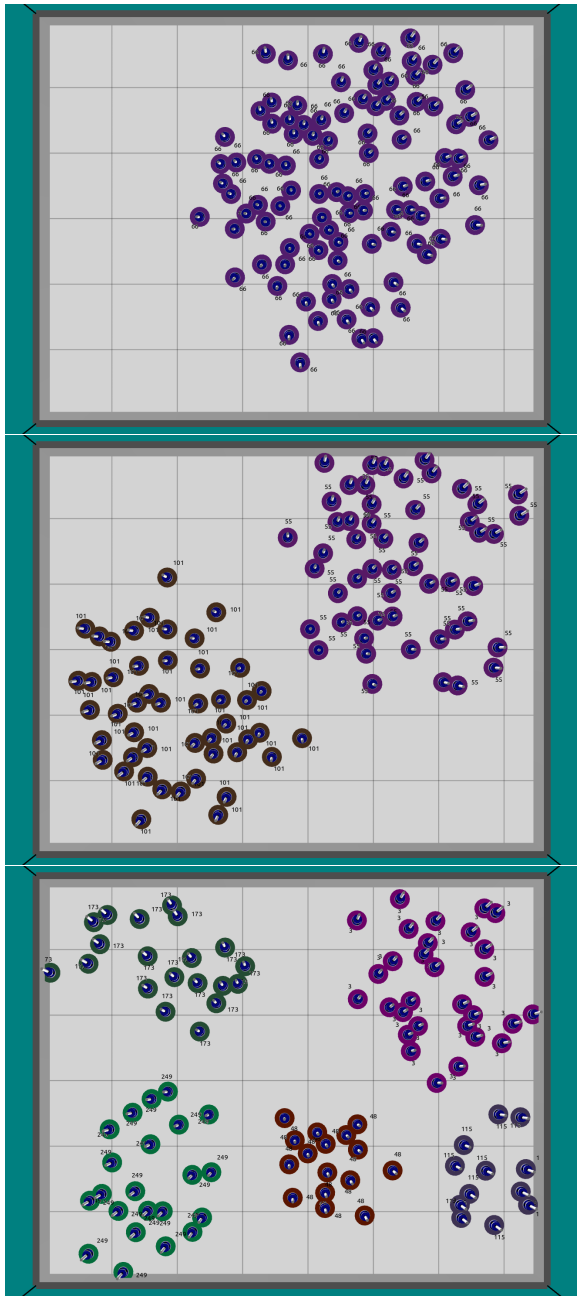


Figure 5.3: Examples of stabilised swarms with a $N = 100$ and p_{Base} configured as (resp.) 0.14 (*top*), 0.17 (*middle*), and 0.20 (*bottom*). For visibility, each word is associated with a different colour in this display. We see that the quantity of aggregates increases with p_{Base} and that each aggregate converged on a different word.

Table 5.2: Average 2nd moment of aggregates

	Vanilla	NG
s20p14	70.3017	85.6672
s20p17	12.5599	56.6773
s20p20	16.4397	17.8883
s100p14	65.0273	37.1738
s100p17	80.9566	19.9819
s100p20	85.5426	18.1084

this setting. Consequently, the average variation of the positions of the robots in the *s20p17* experiment with NG presented in Table 5.2 is increased by the dispersion of non-aggregated swarms. As the equivalent for Vanilla always aggregated, the variation is obviously much lesser.

Yet, in large populations ($N = 100$) the roles are reversed and NG is now much less dispersed. The explanation here is more subtle: As the density of robots is quite high (N increased but the size of the arena remained the same), many are within communication range from the initialisation of the runs. Consequently, in the Vanilla version, many robots stop almost immediately and, as they are already well surrounded, remain in this state indefinitely. The scarce robot that did not stop immediately thus quickly meets the aggregate (which spans the whole arena) and enters the $STAY$ state too. Thus the aggregation process is fast but non-qualitative. In the NG version, however, the robots cannot agree on a word at the outset. They are therefore **forced** too explore the arena and aggregate much later, but in tighter aggregates. Thus, *the aggregation is slower but qualitative.*

To summarise, the influence of the MNG on aggregation is minor with small populations ($N = 20$) but provides remarkable benefits with larger population sizes ($N = 100$) as the swarm *can* then *divide itself in several, tighter, aggregates.* However, more experiments with more population sizes are needed to fully understand the connection between the MNG and the multiplication of aggregates. For instance, it is not clear here whether the density or the population size is the main cause of this quantitative change in behaviour. Conversely, the impact of aggregation on the MNG is that, by reducing robot mobility through aggregation, different MNGs are played in parallel, thus preventing the population from converging to a single word.

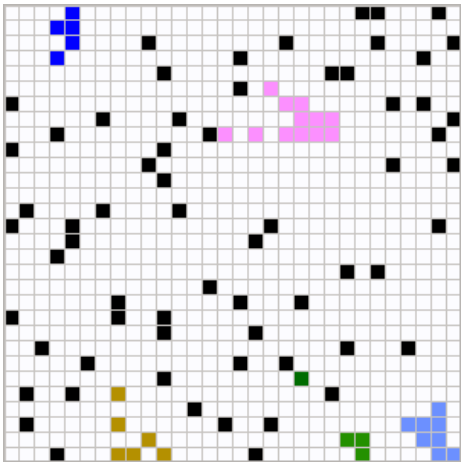


Figure 5.4: A picture of the grid simulator in the course of an experiment. Black squares are walking agents and other colours are uniquely associated to each possible word and are displayed only when agents having selected the corresponding word are staying.

5.2 Exponential Transitions

In order to generalise our results and for coherency with Chapter 6, this section will present experiments using an aggregation controller with exponential transition functions simulated on a grid world. Our experimental setup, which is in line with the methodology that will be used in Chapter 6, will be presented in Section 5.2.1. The results of these experiments will be shown in Section 5.2.2.

5.2.1 Experimental Setup

To run our experiments, we implemented a grid world model illustrated in Figure 5.4. In this model, the world is a bounded grid and the agents can move in four directions: up, down, right, left (which they select randomly). We chose to implement this model as a grid (rather than a continuous space) because, as collisions are an essential feature of embodied experiments and as this work emphasises the importance of on-the-field deployment, collisions are of paramount importance for our purpose. A grid model makes them much easier to implement.

In the following experiments, agents can communicate with agents at a Manhattan distance of two or less i.e. a diagonal adjacent cell counts as two because it needs

two movements to be reached. This distance allows agents to move while aggregated within aggregates, which would not be possible if the range was set to one, as all agents would be densely packed close to each other. Conversely, we used the Manhattan distance rather than the Chebyshev distance (i.e. diagonals counting as one) in order to limit the size of the neighbourhood. The experiments were performed in a $15X15$ grid on this abstract simulator[1].

This controller has four parameters, ϵ, ρ, a and b and, due to the fact that decision-making happens on every time-step (i.e. no waiting period between transition in the PFSM), is much less stable than the linear controller on which these experiments were made in the first place [Cambier et al., 2017]. In order to obtain results that would be comparable to the original experiments, we fixed ϵ and ρ as, resp., 0.03 and 0.49, following our baseline [Correll and Martinoli, 2011]. Moreover, b (that handles the $STAY \rightarrow LEAVE$ transitions) was set to an arbitrary high value of 150 which provides stability similar to the linear controller.

We are consequently left with a single parameter a, as was the case in Section 5.1. We ran our experiments with six values of a, ranging regularly from 0 to 2.5.

Moreover, we implemented a new definition of aggregates in order to detect them automatically. This definition states that aggregates are independent graphs wherein the vertices are agents connected by edges if and only if they are within communication range. Apart from that, our metrics remained identical to those presented in Section 5.1.1.

5.2.2 Results

Figure 5.5 shows that, with $a \in [0, 2.5]$ this controller still aggregates normally without the MNG and/or with $N = 20$. Moreover, with $N = 100$ and symmetric interaction, the quantity of aggregates formed follows an upward tendency as a increases.

Conversely, the MNG continues to converge on a single word (even when no aggregation happens), except when several aggregates are formed, in which case each aggregate gets a different word (this is illustrated in Figure 5.6). These are all similar behaviours to what we observed in Section 5.1.2, although the increase is not as linear, which can be explained by the fact that the transition function is now exponential. Indeed, the quantity of aggregates doubles from $a = 0$ (1 aggregate) to $a = 0.5$ (2 aggregates) but a has to go up to 2.5 for a similar increase to happen

[1] the difference in the proportion of area occupied by the agents compared to experiments in Section 5.1 is irrelevant. Indeed, the probability of interaction are different anyway since the communication range and agents' size and speed are also different.

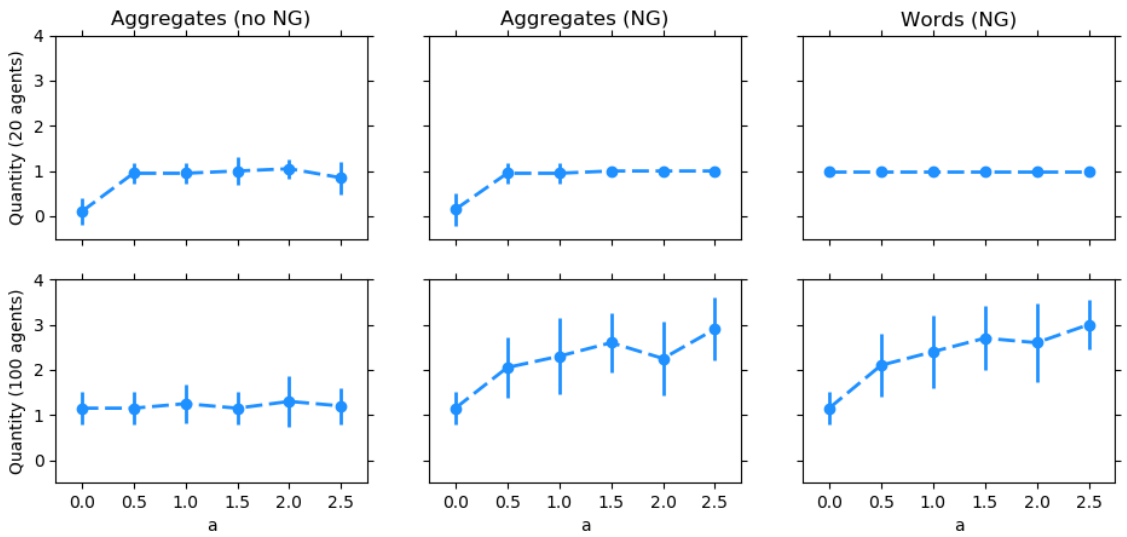


Figure 5.5: Mean (dot) and standard deviation (bars) of the quantity of aggregates in vanilla/NG aggregation (resp. left/middle) and of words (right) in swarms of 20/100 (resp. top/bottom) agents after 30 000 time-steps.

again (3 aggregates).

Finally, the spread of the aggregates follow the same pattern as before with Vanilla and NG-aggregation both producing similarly sparse aggregates, except in dense scenarios, wherein NG-aggregation yields more, tighter, aggregates (see Table 5.3).

Overall, these results are extremely similar to those presented in Section 5.1.2.

5.3 Discussion

We believe that evolutionary linguistic models such as the MNG can offer robotic swarms a way to make collective decisions when several unforeseen alternatives are available. This chapter focused on implementing such a model in an aggregation behaviour, which can be seen as a prerequisite for other types of collective behaviours and tasks [Brambilla et al., 2013].

To study the interplay between self-organised aggregation and the MNG, we strongly linked both dynamics by (1) allowing robots to speak only when staying in

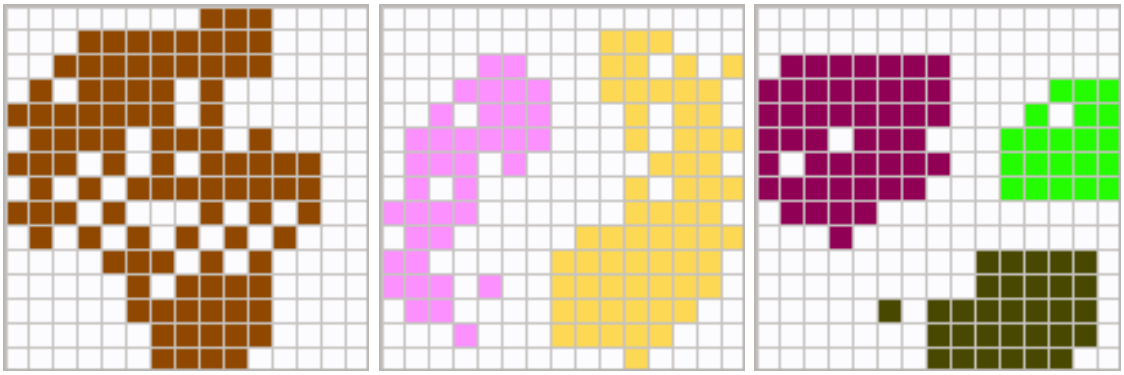


Figure 5.6: Examples of stabilised swarms with a $N = 100$ and parameter a configured as (resp.) 0 ($left$), 0.50 ($middle$), and 2.50 ($right$). For visibility, each word is associated with a different colour in this display. We see that the quantity of aggregates increases with a and that each aggregate converged on a different word.

Table 5.3: Average 2nd moment of aggregates for two population sizes and six parameter settings

	Vanilla	NG
s20a0	0.7847	1.1667
s20a50	3.25	3.25
s20p100	3.235	3.65
s20p150	4.63	3.7375
s20p200	3.935	4.95
s20p250	3.2375	3.235
s100a0	13.2693	25.1182
s100a50	13.2698	13.2947
s100p100	32.3291	4.8461
s100p150	26.2767	14.0400
s100p200	27.6019	15.0285
s100p250	29.3686	7.8441

an aggregate and (2) making the strength of the aggregates directly proportional to the rate of successes in the MNG.

The robustness of our observations was demonstrated by implementing our approach on two different aggregation controllers: the first, using linear transition functions was assessed in a physics-based simulator (Section 5.1) and the second, with exponential functions, demonstrated similar dynamics in an abstract simulator (Section 5.2).

The results of our simulations with the ARGoS simulator show that, in small population sizes ($N = 20$), the aggregation dynamics with the MNG remains unchanged with respect to a vanilla aggregation algorithm without the MNG, but with the added feature of the convergence on a single word, which could be used later in other collective behaviours. However, with larger population sizes ($N = 100$), the swarm displays three interesting emerging features: (a) The possibility to select the quantity of final aggregates by increasing the value of the aggregation parameter; (b) more compact aggregates; (c) aggregates associated each to a different word (labelling).

Chapter 6

Dynamics of Culturally Evolving Probabilistic Aggregation

A major setback of probabilistic aggregation is that its efficiency is extremely sensitive to experimental conditions such as population size [Bayindir and Sahin, 2009] or agent's capabilities (e.g., speed, communication range) [Correll and Martinoli, 2011]. As the target behaviour is an "emergent property of the interaction between the robot and the environment" [Nolfi and Floreano, 2000], the correct settings are almost impossible to predict. Probabilistic aggregation therefore requires supervised tuning of internal model parameters in order to be effective in a specific settings [Soysal and Sahin, 2005, Bayindir and Sahin, 2009]. As seen in Section 2.3 other approaches to self-organised aggregation exist but they usually require a fair amount of offline tuning, whether by automatic optimisation [Trianni et al., 2003, Dorigo et al., 2004, Şahin, 2004] or brute-force search [Gauci et al., 2014], with only few effort on online optimisation [Silva et al., 2015].

In this chapter, we show that the circular interactions suggested in Section 4.2.2 results in an alternative approach to implementing an embodied evolutionary process, using a cultural rather than a biological evolution metaphor [Steels, 2011]. Cultural Evolution (CE) postulates that good ideas spread widely in a population as a result of social dynamics [Castellano et al., 2009]. Here, we propose that these social dynamics can be coupled with the self-organised aggregation dynamics in a way that makes parameters setting that are good for aggregation spread widely in the robot swarm, hence promoting a suitable tuning of the parameters to the specific conditions encountered by the robots.

To demonstrate this concept, we ran preliminary experiments on a physics-based simulator (Section 6.1). Then, by implementing our system on a conceptual simula-

tor, we gained a better understanding of our system (Section 6.2), which enabled us to improve it. These improvements then enabled us to successfully implement this algorithm on kilobots [Rubenstein et al., 2012]. In both cases, our controller uses exponential transition functions (see Section 4.1.2), although it is set differently to highlight different dynamics. We will discuss our conclusions in Section 6.3.

6.1 Preliminary experiments

This section presents the experimental setup (Section 6.1.1) and results (Section 6.1.2) of preliminary studies we performed for the cultural evolution of aggregation parameters using circular interactions. This section extends on a research that was originally published in [Cambier et al., 2018].

6.1.1 Experimental Setup

The proposed embodied evolution was implemented on simulated MarXbots [Bonani et al., 2010] within the ARGoS simulator [Pinciroli et al., 2012]. The MarXbots were moving at a speed of 10cm/s and communicating with robots within a range of 70cm. As in Section 5.2.1, we set ϵ and ρ (from Equation 4.2) as, resp., 0.03 and 0.49 in order to optimise only two parameters of our controller: a and b (the latter from Equation 4.3). Furthermore, we decided to discretise a and b in the range (1,5] with steps of 0.25 as this was the most dynamic range we could find. This allowed us to encode these parameters on 1 byte (four bits for each parameter) and to analyse the performances of each parameter setting by displaying them in two dimensions, as in Figure 6.1.

Also as in Section 5.2.1, in these preliminary experiments, the transition probabilities were computed using s (the quantity of MNG successes) rather than n (the quantity of neighbours).

We used a circular arena of constant radius $r = 10m$ and with three different population sizes $N = \{25, 50, 100\}$. We evaluated the aggregation behaviour using the cluster metric [Gauci et al., 2014], which is the ratio between the size of the biggest cluster and the swarm size N. We first performed a brute-force analysis of the parameter space of our aggregation behaviour without CE (following only Section 4.1.2).

To highlight the dynamics produced by embodied evolution, we contrasted it with selected non-evolving instantiations, namely the *baseline* controller obtained by fitting the parameters a and b to the probability table from [Correll and Martinoli, 2007], and the *optimal* controller obtained with the parameter settings that

maximised the cluster metric, as identified through brute-force search. We ran the following experiments:

- 20 runs of embodied evolution with mutation rate $m = 0.001$
- 20 runs of the baseline controller featuring fixed parameters $(a, b) = (1.70188, 3.88785)$, set to fit [Correll and Martinoli, 2007]
- 10 runs of the optimal controller with fixed parameters:
  - $N = 25 : (a, b) = (2.25, 3.5)$
  - $N = 50 : (a, b) = (1.25, 2.0)$
  - $N = 100 : (a, b) = (1.25, 1.25)$

In addition to the cluster metric, we recorded the variation over time of the number of clusters formed and the number of free agents.

In addition, we confronted three experimental conditions in which density of robots was maintained constant: $N = 25$ with $r = 5m$, $N = 50$ with $r = 7m$, and $N = 100$ with $r = 10m$. We contrasted CE with the baseline controller and with the *optimal* controller obtained with brute force search for $N = 25$ and $r = 5m$. The latter is tested also in the other conditions, to verify whether fixed parametrisations that prove optimal on a given scale also perform well with larger scales.

6.1.2 Results

The results are displayed in Figure 6.1. A close look reveals that most differences in the quality of the aggregation are attributable to parameter b, i.e., dispersion (at least within the range we tested). Moreover, the size of the largest aggregate increases as b decreases until a threshold value (specific to N), after which aggregation suddenly becomes impossible due to too strong dispersion. Finally, we can observe that, the larger N, the stronger the dispersion needs to be to obtain large aggregates, which reduces the amount of good parameter settings as they are lower-bounded by the threshold. These observations are compatible with Premise 1 (from Section 4.2.2) and show the cohesion trade-off evoked to build this premise.

Note also that these results are obtained by fixed parametrisations, and clearly reveal the sensitivity of the probabilistic aggregation to the working conditions, as the parametrisations that maximise the cluster metric strongly vary with the swarm size N. In contrast, the proposed CE model should be able to tune the parameters of each robot online, and, therefore, to exploit both adaptation of the parameters and some heterogeneity within the swarm.

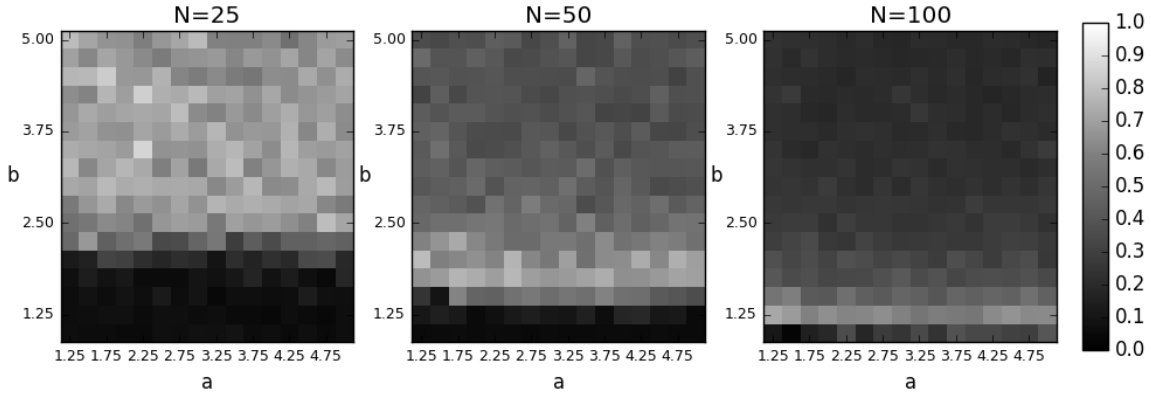


Figure 6.1: Average size of the largest cluster for each value of the parameters a and b, evaluated with the cluster metric and averaged over 10 independent runs.

A comparison between fixed and evolving aggregation in an arena of constant radius $r = 10$ (i.e. variable density) is shown in Figure 6.2. It is possible to notice that the CE model fails to produce stable aggregates when $N = 25$. This is because the MNG is particularly slow at low densities, because interactions among agents happen with very low probability [Trianni et al., 2016, Cambier et al., 2017]. As a consequence, the number of successful games is small—also due to mutations disturbing the language dynamics—and clusters quickly disband. However, as N increases, we can see that CE presents dynamics that are very close to the baseline aggregation behaviour [Correll and Martinoli, 2007], i.e. a short phase of building aggregates followed by stagnation. For $N = 100$, CE attains values for the cluster metric that are higher than the baseline controller. Additionally, evolution is different from the baseline in respect of the number of free agents. Indeed, almost all the agents of the baseline behaviour stay in clusters after the build-up phase, whilst the evolutionary model continues to explore for a longer time and never entirely stops. This demonstrates a better handling of the dispersion trade-off, which explains our model's higher scalability especially for large N. The optimal controllers, instead, proceed by slowly and constantly building up a large aggregate, maintaining at the same time a large fraction of exploring robots. This slow process represents the only means to increase the size of the largest cluster at the expenses of small clusters, when parameters are fixed and the system is homogeneous. However, a very specific parameterisation is necessary to observe this behaviour, especially for large N, as shown in Figure 6.1, making these controllers very sensitive to variation in the experimental conditions.

To understand the heterogeneity of parameters obtained through CE, Figure 6.3

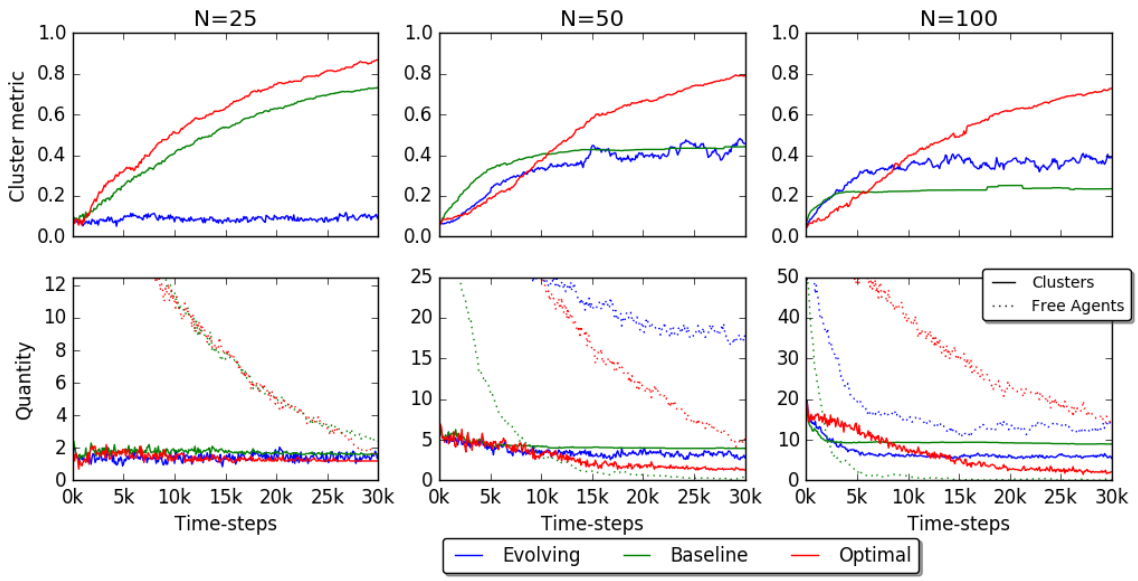


Figure 6.2: Averaged time evolution over i runs of the three variations: CE ($i = 20$), *baseline* controller with parameters fixed to match the probability table from [Correll and Martinoli, 2007], $i = 20$) and *optimal* ($i = 10$).

shows the frequencies of the different parametrisations found within the swarm at the end of the 20 performed runs, for $N = \{25, 50, 100\}$. It is possible to notice that some parameter settings have been selected multiple times across different scales and evolutionary runs. Furthermore, judging by the values distribution, the model converges towards the higher region of the maps, i.e. low dispersion parameter settings. As the optimal parameter setting for the non-evolving model (Fig. 6.1) lies on the other end of the dispersion spectrum, we can understand that the CE builds up aggregates in a different way, by varying the parameters during the run and by exploiting heterogeneity among the robots. Indeed, robots can, for example, develop a parametrisation leading to high-cohesion that barely allows leaving, and just wait for other robots with low-cohesion parameters to disperse and wander into their trap-like cluster which would, as a consequence, grow and become more effective to capture other robots.

We have observed that the CE results in larger clusters when the number of robots is high. As exhibited in Figure 6.4, with constant density, the CE is sufficiently scalable. Indeed, we can see that, with sufficient robot density, the evolutionary model initially performs as well as the baseline behaviour [Correll and Martinoli, 2007] and

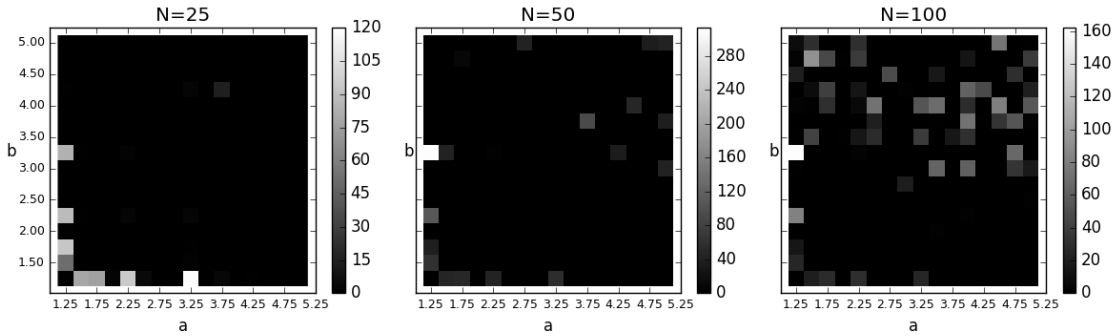


Figure 6.3: Total occurrences of each possible parameter settings (with CE) at the end of 20 runs in setting with $N = \{25, 50, 100\}$ and a circular arena of radius $r = 10$.

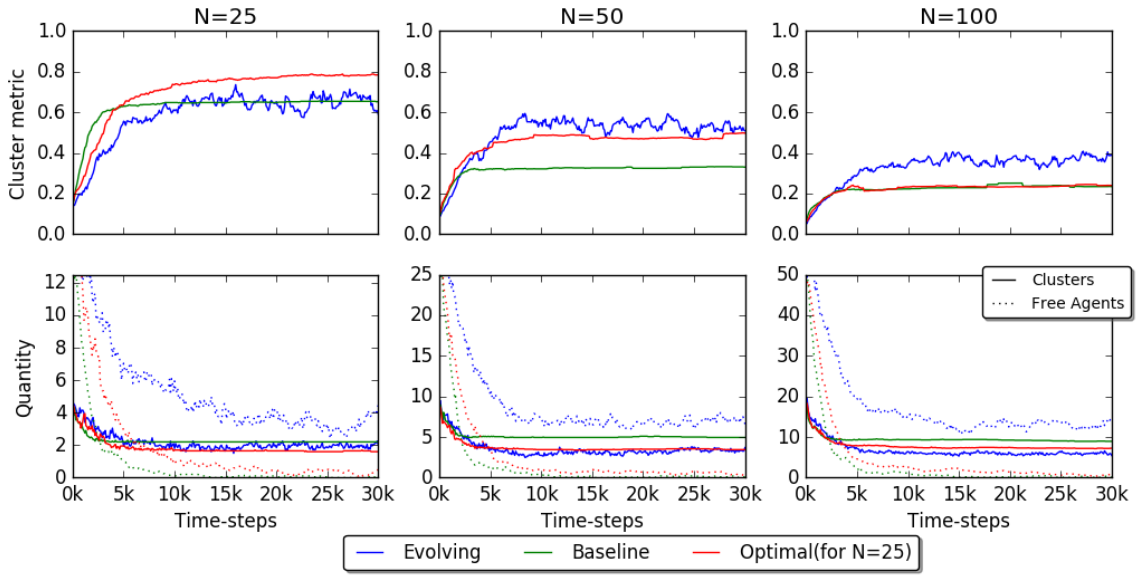


Figure 6.4: Averaged time evolution over 20 runs of the three variations with constant density: CE, baseline behaviour and optimal controller for $N = 25$ and $r = 5m$ (i.e. $(a, b) = (1.5, 2.75)$).

not too distant from the optimal behaviour. However, CE scales up better than either of the fixed-parameters alternatives. We conclude that CE represents a promising solution for scalable behaviour rather than optimal probabilistic aggregation, provided that the density of robots remains sufficiently high.

6.2 In-depth study

The results presented in Section 6.1 only consider a small parameter space. In order to be able to study a much larger parameter space, we thus moved to an abstract simulator that could run faster experiments. The simulator, as well as our parameter space and the details of our experimentations will be presented in Section 6.2.1. Then, our experimental results will be examined in Section 6.2.2.

6.2.1 Experimental Setup

The simulator we used for the following experiments is a grid world similar to the one we presented in Section 5.2.1. The only difference is that, in the present case, we set the communication range to 1 in order to favour packed aggregates and to make the aggregation more difficult as agent encounters become rarer. Conversely, here, we used the Chebyshev distance (i.e. diagonals also count as a distance of 1) rather than the Manhattan in order to avoid excessively restricting the neighbourhood. Indeed, this way, the maximum neighbour size is 8 instead of 4 with the Manhattan distance.

Contrary to Section 6.1, in the following experiments, n remains the number of neighbours and keeps being used as is in the transitions functions (instead of s). This is because, as we have observed in Section 6.1.2, the noise increases the rate of loss in the MNG, which means that increasing the noise would stop the aggregation as much as it would allow to find new parameters. Furthermore, agents play a single game per time-step with a single word selected randomly from all the words received in this time-step. This is to slow the dynamics of the MNG down.

For the transition functions of the controller, we maintained the exponential equations (cf. Section 4.1.2) and set $\epsilon = 1 - \rho$ so that p_{Join} would always converge to 1. The remaining three parameters remained free in the range $[0, 1]$ (for ρ) and $[0, 5]$ (for a and b). We did not explore the range beyond 5 for a and b as, at this point, p_{Join} and p_{Leave} already converge to, resp., 1 and 0 for $n = 1$. All the parameters were encoded on 24 bits (1 byte each), which ensures really small steps of an order of 10^{-2}.

The experiments were performed with $N = \{25, 50, 100, 200\}$ agents in grids of constant density. Two densities were considered: high (1 agent every 12 cells) and low (1 agent every 25 cells). The corresponding grid sizes are detailed in Table 6.1. All experiments were run for 300 000 time-steps and over 100 independent runs.

We ran our CE algorithm with two mutation rates $m = \{0.01, 0.001\}$. Each setting was run 100 times.

In order to compare the CE aggregation with its fixed variation, we optimised

Table 6.1: Grid sizes

	High Density (1/12)	Low Density (1/25)
N=25	17x17	25x25
N=50	24x24	35x35
N=100	35x35	50x50
N=200	49x49	71x71

Table 6.2: Optimal fixed parameters

Density	Param.	N=25	N=50	N=100	N=200
High (1/12)	a	2.86	4.27	0.61	0.18
	b	2.57	2.08	1.8	1.42
	ρ	0.35	0.95	0.22	0.11
Low (1/25)	a	3.55	2.36	3.4	2.38
	b	2.67	2.1	1.89	1.42
	ρ	0.89	0.7	0.6	0.62

the latter using irace [López-Ibáñez et al., 2016]. The controller was optimised independently for each setting with a budget of 20000 runs each time. The mean cluster metrics of all parameter settings thus obtained are presented in Figure 6.5. We selected the best parameter setting of each configurations to run them against our cultural evolution algorithm. The optimal parameter thus found are available in Table 6.2.

As a proof of concept, we also implemented this algorithm on kilobots [Rubenstein et al., 2012] simulated in ARGoS. In order to make sure the kilobots could get out of their aggregate's communication range when they decide to leave, we set a decision time of 60 time-steps. The PFSM thus ticks every 6 seconds. These experiments were performed with $N = \{25, 50\}$, over 20 runs each.

6.2.2 Results

We expect CE-aggregation to select efficient parameters online. To verify this hypothesis, we compared the CE-aggregation with a regular aggregation algorithm set with optimal (fixed) parameters. This comparison was performed in eight different

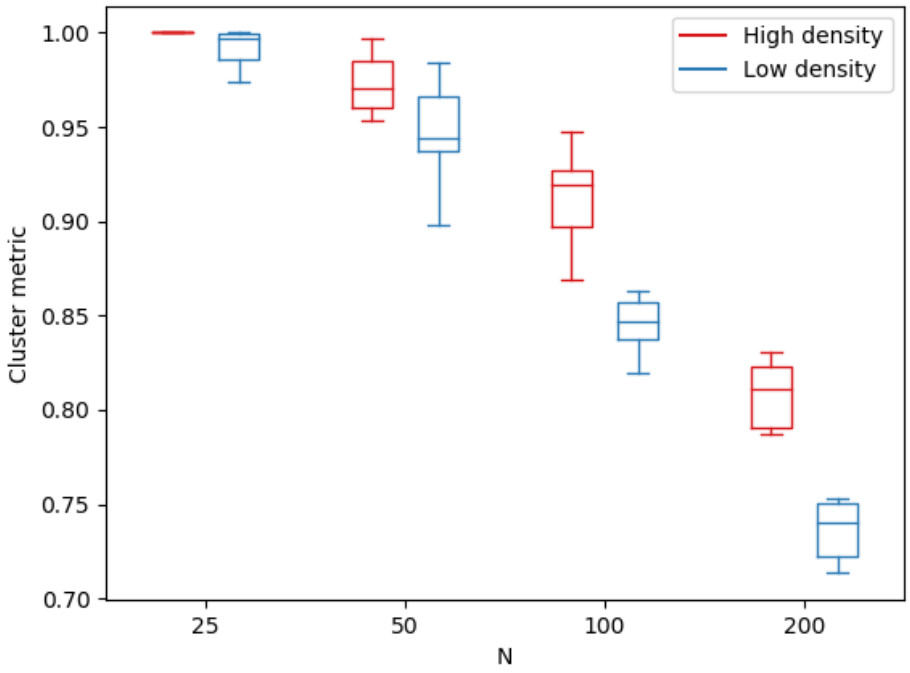


Figure 6.5: Box-plot of the average performances obtained by 10 optimisation runs in irace for each configurations. A downward trend is clearly visible as the quantity of agents (and/or the density) increases (resp. decreases).

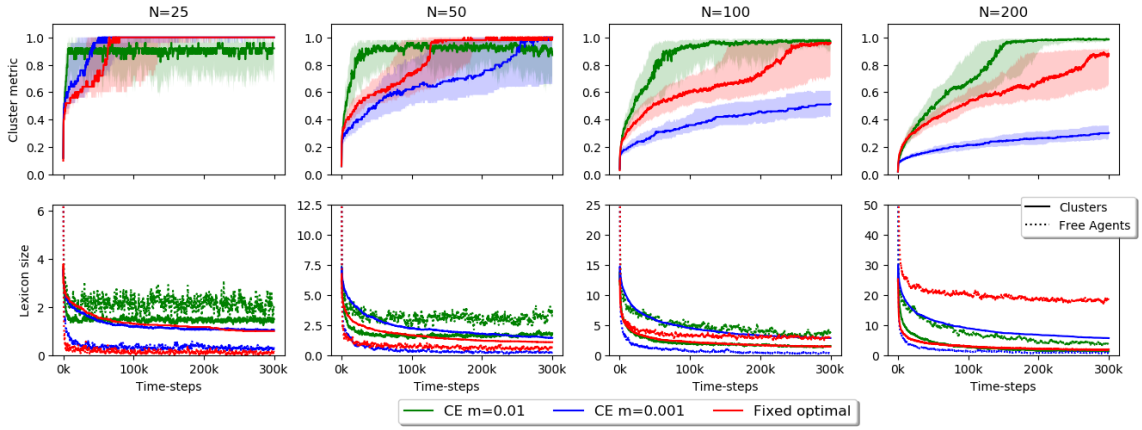


Figure 6.6: Comparison between CE aggregation and optimal fixed aggregation with a constant density of 1/12.

experimental conditions and each optimal setting is specific to the given conditions. We refer the reader to Table 6.2 for these settings.

Figure 6.6 presents this comparison with the first four conditions, i.e. with four different populations sizes but maintaining a high population density. Solid lines represent the median cluster metric and shaded regions indicate the 1st and 3rd quartiles. From the outset, we see that CE-aggregation can perform as well as—or better—than optimal parameters. Indeed, a low mutation rate ($m = 0.001$) enables the whole swarm to aggregate in a single cluster for low population sizes, i.e. $N = 25$ and $N = 50$, although the process is longer in the latter case. Nevertheless, with this mutation rate, CE-aggregation does not scale up well and quickly yields outright bad performances. This is not comparable to the optimal, which not scale up perfectly either (although it does better), but requires a complete change of parameter for each experimental conditions. Even though a higher mutation rate ($m = 0.01$) gives slightly lesser performances at low scales, it also outperforms the optimal fixed aggregation for $N = 200$.

These impressions can be confirmed using the Wilcoxon rank-sum test, as in Table 6.3. In this table, positive (resp. negative) "statistic" values indicate that CE-aggregation yields higher (lower) performances than the optimal. The p-value gives a confidence interval as values lower than 0.05 reliably indicate that the values yielded by both algorithm are drawn from two different distributions. In addition to confirming the observations we already made, this test shows that the high-mutation CE-aggregation also outperforms the optimal for $N = 100$.

These trends are even more visible in low densities, as presented in Table 6.4.

m	N	statistic	p-value
0.01	25	-8.46145571612	2.64060339719e-17
	50	-7.33138835582	2.27780537046e-13
	100	2.97971274785	0.00288518801021
	200	10.5517749091	4.98464345723e-26
0.001	25	-0.564422832926	0.572466403302
	50	-2.32854960943	0.0198829382166
	100	-9.40460382655	5.22265814727e-21
	200	-12.0397987413	2.19490470991e-33

Table 6.3: Wilcoxon rank-sum with the null hypothesis that the final cluster metrics of CE aggregation and fixed aggregation (in high-density arenas) are drawn from the same distribution.

The lack of scalability for low mutation rates is explained by the fact that, in large aggregates, the few mutations happening are "drowned out", in the MNG, by the numerous successful words. As a result, the agents completely stop exploring new parameter settings and, as shown by the lack of free agents in Figure 6.7, they settle in many small clusters; up to ten in large populations ($N = 200$). Conversely, high mutations rates allow new mutations to take hold of large aggregates and to completely change their dynamics, resulting in a larger proportion of free agents. This is obviously limiting in small populations as the agents never aggregate entirely, but this is also the reason CE-aggregation can outperform a fixed, optimal, aggregation controller.

Indeed, the combination of the MNG and of mutations enable two new dynamics. Firstly, as said in Section 6.1.2, the agents controllers become heterogeneous. Therefore, agents that aggregate can tune their parameters to form extremely strong aggregates that just can not disperse. These aggregates form "traps" which wait for agents with weaker parameters to come to them and be captured. This reinforcement is possible because, if a weaker parameter setting appears in the population, the agent leaves with a high probability and immediately stops spreading these weak parameters.

Secondly, the MNG has a stochastic component and therefore, weak mutations can survive and infect a major part of an aggregate, leading to an haemorrhage of agents. This event appears with a probability that decreases with the size of the aggregate (as, again, mutations are "drowned out" more easily). Therefore, this dynamic reinforces the main mechanism of self-organised whereby small aggregates

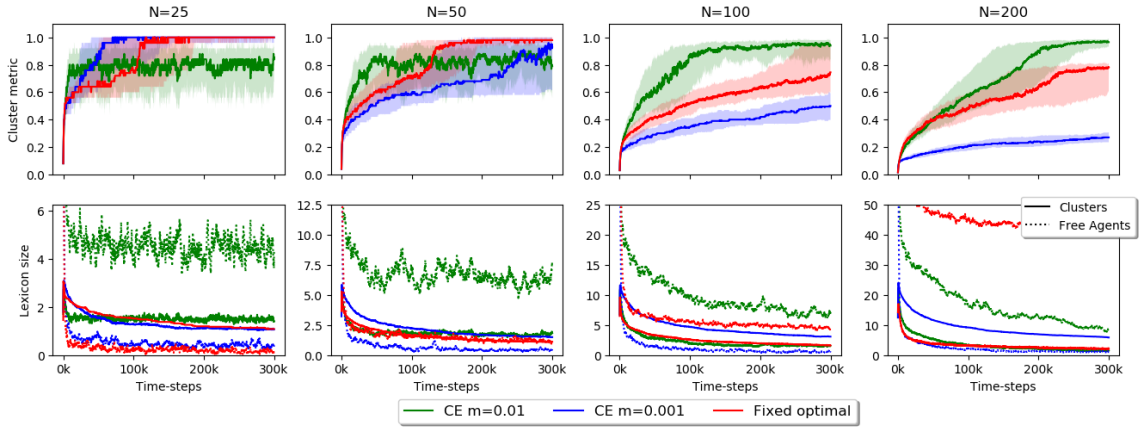


Figure 6.7: Comparison between CE aggregation and optimal fixed aggregation with a constant density of 1/25.

disband more easily.

As Tables 6.3 and 6.4 show, CE-aggregation with $m = 0.01$ is the most resilient setting. We thus used this parameter to assess our implementation on kilobots. The ensuing performances, plotted in Figure 6.8, are not as impressive as in our abstract model. Nevertheless, they compare favourably to the results presented in Section 6.1, despite being obtained with kilobots, which are much more challenging due to asynchronous messaging and a complete absence of proximity sensors [Rubenstein et al., 2012]. The latter provokes many collisions that force aggregates to disband. Moreover, the large decision-time, which is necessary for kilobots to leave their aggregates, considerably slows the dynamics down. This means that aggregation with even 50 robots can take a while to stabilise, as indicated by the large dispersion of the cluster metric towards the end of the experiments. To the best of our knowledge, this is the only successful attempt at implementing self-organised aggregation on kilobots.

6.3 Discussion

In this chapter, we presented a novel Cultural Evolution (CE) approach for swarm robotics based on social dynamics. The main idea underlying this proposal is linking the spreading of opinions in the population to the self-organising processes displayed by the robot swarm. Starting from the CE metaphor, which postulates that good ideas spread widely in a population, we proposed that good parameters of a self-organising behaviour can spread following a social dynamics process, leading to a

m	N	statistic	p-value
0.01	25	-8.77420949367	1.72108567037e-18
	50	-8.29652696624	1.07199196017e-16
	100	4.90388149646	9.39611881503e-07
	200	9.50844785425	1.93528452088e-21
0.001	25	-1.22902461022	0.21906257901
	50	-2.2686865817	0.0232873921706
	100	-8.9770107713	2.78222662183e-19
	200	-12.1302041301	7.30670741954e-34

Table 6.4: Wilcoxon rank-sum with the null hypothesis that the final cluster metrics of CE aggregation and fixed aggregation (in low-density arenas) are drown from the same distribution.

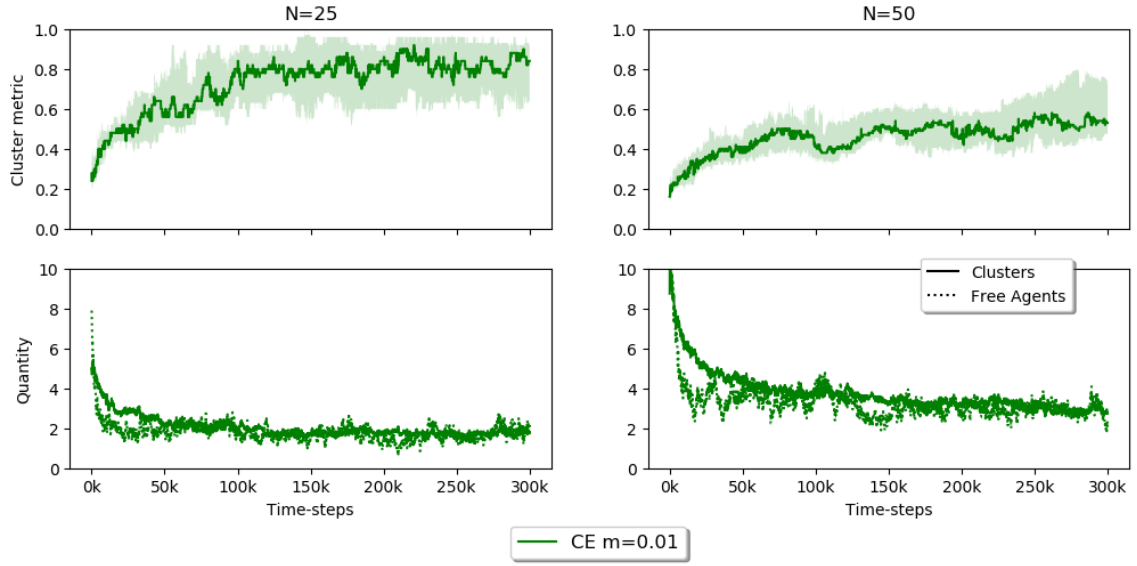


Figure 6.8: Performances of CE aggregation implemented on kilobots. The diameter of the area $d = \{1, 1.4\}$ metres for, resp., $N = \{25, 50\}$ in order to keep the density constant.

swarm capable of adapting its behaviour to the current environmental conditions it is experiencing. This was possible to the extent that communication noise created new parameters settings.

We assessed this idea on probabilistic aggregation, using the MNG as the cultural transmission operator. After a phase of preliminary experiments on a physics-based simulator (Section 6.1), we were able to improve this idea by weakening the coupling between aggregation and MNG (Section 6.2) as the only necessary link between the two is to use the MNG's words as the aggregation parameters and to only let the staying robots speak.

Experimental results on an abstract grid world simulator showed that the proposed CE process (a) alters the dynamics of aggregation by introducing varying and heterogeneous alignment and dispersion across the swarm, and (b) it autonomously selects parameter settings that outperform regular aggregation with optimal parameters. The robustness of this approach was assessed with simulated kilobots.

Chapter 7

Conclusion

With this thesis, we aimed to devise a framework for the emergence of meaningful communication in robotics in order to enable the exploration of unknown and open-ended environments. For this purpose, we presented the various design methods of swarm robotics and possible applications as well as detailing the importance of communication and the usages of self-organised aggregation (Chapter 2). Then, we emphasised the importance of communication in emerged behaviours and proposed to focus on self-organising language rather than evolving it in a biological fashion as this course of action is at the same time linguistically realistic and coherent with the principles of swarm robotics (Chapter 3). Afterwards, we used self-organised aggregation and the minimal naming game (MNG) as examples to build a methodological framework to integrate swarm robotics and evolutionary linguistics (Chapter 4).

We demonstrated that bidirectional interactions between the dynamics of self-organised aggregation and the MNG provoked the emergence of a new clustering behaviour, with each aggregate having its own associated word (Chapter 5). Subsequently, we showed that both dynamics could also reinforce each other as they led to a form of embodied evolution (Chapter 6). This find paves the way towards swarms of robots that can adapt to any situation regardless of the unpredictability of the environment. This is obviously of paramount importance for the exploration of unknown environments.

Several directions are enabled by the present contribution. As a matter of fact, the dynamics we have observed in our contributions can be relevant to other task than aggregation as several swarm behaviours use similar stochastic models. A natural extension would be to (1) integrate the naming game into foraging and to compare the resulting cultural evolution to adaptive strategies. Self-organised aggregation is also very similar to area coverage as both put emphasis on maintaining a consistent

topology. Hence, (2) cultural evolution could naturally be applied to area coverage.

Moreover, the present contribution is only a first stone towards a new framework, we should thus also consider more ambitious models of languages than mere one-word lexicon. As previous works have shown that the observations obtained from language games for lexicon formation can be applied to grammar [Nolfi and Mirolli, 2009, p. 285], future work should aim to (3) exploit the characteristics that make natural languages so powerful.

Furthermore, the work presented in this thesis very much assumes a conversational (language game protocols) and sociological frame [Goffman, 1974] (the base aggregation controller). Future work should try and (4) make these frames emerge as well, as is usually the case in evolutionary robotics.

An immediate application of the dynamics offered by symmetric interactions can be envisioned for 3D mapping and exploration with UAVs. Indeed, the differing word for each aggregate can be used as a label for each area of interest. This means that, even without accurate positioning, data with the same label can be interpreted as coming from the same area. This can later help the fusion of the data into topological/semantic or even appearance maps. This opens a way for self-organising language to enable a topological map representation, which is similar to the way humans conceptualise large spaces. Swarm robotic models could therefore also serve to (5) investigate the relationship between space conceptualisation and language formation in human populations. This line of enquiry could have far-reaching consequences in cognitive and linguistic theory as well as improving the quality of simultaneous localisation and mapping.

Bibliography

Mazda Ahmadi and Peter Stone. Continuous area sweeping: A task definition and initial approach. In *ICAR'05. Proceedings., 12th International Conference on Advanced Robotics, 2005.*, pages 316–323. IEEE, 2005.

Mazda Ahmadi and Peter Stone. A multi-robot system for continuous area sweeping tasks. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 1724–1729. IEEE, 2006.

Dario Albani, Joris IJsselmuiden, Ramon Haken, and Vito Trianni. Monitoring and mapping with robot swarms for agricultural applications. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2017.

Dario Albani, Tiziano Manoni, Daniele Nardi, and Vito Trianni. Dynamic uav swarm deployment for non-uniform coverage. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 523–531. International Foundation for Autonomous Agents and Multiagent Systems, 2018.

Sjriek Alers, Karl Tuyls, Bijan Ranjbar-Sahraei, Daniel Claes, and Gerhard Weiss. Insect-inspired robot coordination: foraging and coverage. In *Artificial Life Conference Proceedings 14*, pages 761–768. MIT Press, 2014.

Jean-Marc Amé, Colette Rivault, and Jean-Louis Deneubourg. Cockroach aggregation based on strain odour recognition. *Animal behaviour*, 68(4):793–801, 2004.

Jean-Marc Amé, José Halloy, Colette Rivault, Claire Detrain, and Jean Louis Deneubourg. Collegial decision making based on social amplification leads to optimal group formation. *Proceedings of the National Academy of Sciences*, 103(15): 5835–5840, 2006.

Christos Ampatzis, Elio Tuci, Vito Trianni, and Marco Dorigo. Evolution of signaling in a multi-robot system: Categorization and communication. *Adaptive Behavior*, 16(1):5–26, 2008.

Christos Ampatzis, Elio Tuci, Vito Trianni, Anders Lyhne Christensen, and Marco Dorigo. Evolving self-assembly in autonomous homogeneous robots: Experiments with two physical robots. *Artificial Life*, 15(4):465–484, 2009.

Nikolay Atanasov, Jerome Le Ny, Kostas Daniilidis, and George J Pappas. Decentralized active information acquisition: Theory and application to multi-robot slam. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4775–4782. IEEE, 2015.

Tim Bailey, Juan Nieto, Jose Guivant, Michael Stevens, and Eduardo Nebot. Consistency of the ekf-slam algorithm. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3562–3568. IEEE, 2006.

Gianluca Baldassarre, Domenico Parisi, and Stefano Nolfi. Distributed coordination of simulated robots based on self-organization. *Artificial life*, 12(3):289–311, 2006.

J Mark Baldwin. A new factor in evolution. *The american naturalist*, 30(354): 441–451, 1896.

Andrea Baronchelli. Role of feedback and broadcasting in the naming game. *Physical Review E*, 83(4):046103, 2011.

Andrea Baronchelli. The emergence of consensus: a primer. *Royal Society open science*, 5(2):172189, 2018.

Andrea Baronchelli, Maddalena Felici, Vittorio Loreto, Emanuele Caglioti, and Luc Steels. Sharp transition towards shared vocabularies in multi-agent systems. *Journal of Statistical Mechanics: Theory and Experiment*, 2006(06):P06014, 2006.

Andrea Baronchelli, Vittorio Loreto, and Luc Steels. In-depth analysis of the naming game dynamics: the homogeneous mixing case. *International Journal of Modern Physics C*, 19(05):785–812, 2008.

Andrea Baronchelli, Tao Gong, Andrea Puglisi, and Vittorio Loreto. Modeling the emergence of universality in color naming patterns. *Proceedings of the National Academy of Sciences*, 107(6):2403–2407, 2010.

Elizabeth Bates. On the nature and nurture of language. *Frontiers of biology: The brain of Homo sapiens*, pages 241–265, 2003.

Levent Bayindir and Erol Sahin. Modeling self-organized aggregation in swarm robotic systems. In *Swarm Intelligence Symposium, 2009. SIS'09. IEEE*, pages 88–95. IEEE, 2009.

Ralph Beckers, Owen E Holland, and Jean-Louis Deneubourg. From local actions to global tasks: Stigmergy and collective robotics. In *Prerational Intelligence: Adaptive Behavior and Intelligent Systems Without Symbols and Logic, Volume 1, Volume 2 Prerational Intelligence: Interdisciplinary Perspectives on the Behavior of Natural and Artificial Systems, Volume 3*, pages 1008–1022. Springer, 2000.

Patrick Beeson, Joseph Modayil, and Benjamin Kuipers. Factoring the mapping problem: Mobile robot map-building in the hybrid spatial semantic hierarchy. *The International Journal of Robotics Research*, 29(4):428–459, 2010.

Brent Berlin and Paul Kay. *Basic color terms: Their universality and evolution*. Univ of California Press, 1969.

R. Bianco and S Nolfi. Toward open-ended evolutionary robotics: evolving elementary robotic units able to self-assemble and self-reproduce. *Connection Science*, 16(4):227–248, 2004.

John Boardman and Brian Sauser. System of systems-the meaning of of. In *2006 IEEE/SMC International Conference on System of Systems Engineering*, pages 6–pp. IEEE, 2006.

Michael Bodi, Ronald Thenius, Martina Szopek, Thomas Schmickl, and Karl Crailsheim. Interaction of robot swarms using the honeybee-inspired control algorithm beeclust. *Mathematical and Computer Modelling of Dynamical Systems*, 18(1): 87–100, 2012.

Eric Bonabeau, Guy Theraulaz, and Jean-Louis Deneubourg. Quantitative study of the fixed threshold model for the regulation of division of labour in insect societies. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 263 (1376):1565–1569, 1996.

Michael Bonani, Valentin Longchamp, Stéphane Magnenat, Philippe Rétornaz, Daniel Burnier, Gilles Roulet, Florian Vaussard, Hannes Bleuler, and Francesco Mondada. The marxbot, a miniature mobile robot opening new perspectives for

the collective-robotic research. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 4187–4193. IEEE, 2010.

Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, and Marco Dorigo. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7 (1):1–41, 2013.

Nicolas Bredeche. Hit-ee: a novel embodied evolutionary algorithm for low cost swarm robotics. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 109–110. ACM, 2019.

Nicolas Bredeche and Jean-Marc Montanier. Environment-driven embodied evolution in a population of autonomous agents. In *International Conference on Parallel Problem Solving from Nature*, pages 290–299. Springer, 2010.

Nicolas Bredeche, Evert Haasdijk, and Abraham Prieto. Embodied evolution in collective robotics: A review. *Frontiers in Robotics and AI*, 5:12, 2018.

Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6):1309–1332, 2016.

Christian Leonardo Camacho-Villalón, Marco Dorigo, and Thomas Stützle. Why the intelligent water drops cannot be considered as a novel algorithm. In *International Conference on Swarm Intelligence*, pages 302–314. Springer, 2018.

Scott Camazine. *Self-organization in biological systems*. Princeton University Press, 2003.

Nicolas Cambier, Vincent Frémont, and Eliseo Ferrante. Group-size regulation in self-organised aggregation through the naming game. In *International Symposium on Swarm Behavior and Bio-Inspired Robotics (SWARM 2017)*, Kyoto, Japan, Oct 2017. URL `https://hal.archives-ouvertes.fr/hal-01679600`.

Nicolas Cambier, Vincent Frémont, Vito Trianni, and Eliseo Ferrante. Embodied evolution of self-organised aggregation by cultural propagation. In *ANTS 2018*, Rome, Italy, Oct 2018.

Alexandre Campo. *On the design of self-organized decision making in robot swarms*. PhD thesis, Université libre de Bruxelles, 2011.

Alexandre Campo, Shervin Nouyan, Mauro Birattari, Roderich Groß, and Marco Dorigo. Enhancing cooperative transport using negotiation of goal direction. Technical report, University of Namur, 2006.

Alexandre Campo, Álvaro Gutiérrez, Shervin Nouyan, Carlo Pinciroli, Valentin Longchamp, Simon Garnier, and Marco Dorigo. Artificial pheromone for path selection by a foraging swarm of robots. *Biological cybernetics*, 103(5):339–352, 2010.

Alexandre Campo, Simon Garnier, Olivier Dédriche, Mouhcine Zekkri, and Marco Dorigo. Self-organized discrimination of resources. *PLoS One*, 6(5):e19888, 2011.

Claudio Castellano, Santo Fortunato, and Vittorio Loreto. Statistical physics of social dynamics. *Reviews of Modern Physics*, 81(2):591–646, 2009.

José A Castellanos, José Neira, and Juan D Tardós. Multisensor fusion for simultaneous localization and map building. *IEEE Transactions on Robotics and Automation*, 17(6):908–914, 2001.

Noam Chomsky. Aspects of the theory of syntax. Technical report, MASSACHUSETTS INST OF TECH CAMBRIDGE RESEARCH LAB OF ELECTRONICS, 1964.

Noam Chomsky. Rules and representations. *Behavioral and brain sciences*, 3(1): 1–15, 1980.

Alonzo Church. Logic, arithmetic and automata. In *Proceedings of the international congress of mathematicians*, volume 1962, pages 23–35, 1962.

Nikolaus Correll. Parameter estimation and optimal control of swarm-robotic systems: A case study in distributed task allocation. In *2008 IEEE International Conference on Robotics and Automation*, pages 3302–3307. IEEE, 2008.

Nikolaus Correll and Alcherio Martinoli. Modeling self-organized aggregation in a swarm of miniature robots. In *IEEE 2007 International Conference on Robotics and Automation Workshop on Collective Behaviors inspired by Biological and Biochemical Systems*, number SWIS-CONF-2007-002, 2007.

Nikolaus Correll and Alcherio Martinoli. Modeling and designing self-organized aggregation in a swarm of miniature robots. *The International Journal of Robotics Research*, 30(5):615–626, 2011.

John Crothers. Typology and universals of vowel systems in phonology. *Universals of human language*, 2:95–152, 1978.

Barrie Dams, Sina Sareh, Ketao Zhang, Paul Shepherd, Mirko Kovac, and Richard Ball. Aerial additive building manufacturing: three-dimensional printing of polymer structures using drones. *Proceeding of the Institution of Civil Engineers: Construction Materials.*, pages 1–12, 2017.

Bart De Boer. Self-organization in vowel systems. *Journal of phonetics*, 28(4):441–465, 2000.

J-L Deneubourg, Serge Aron, Simon Goss, and Jacques M Pasteels. The self-organizing exploratory pattern of the argentine ant. *Journal of insect behavior*, 3 (2):159–168, 1990.

Jean-Louis Deneubourg, Arnaud Lioni, and Claire Detrain. Dynamics of aggregation and emergence of cooperation. *The Biological Bulletin*, 202(3):262–267, 2002.

Cristina Dimidov, Giuseppe Oriolo, and Vito Trianni. Random Walks in Swarm Robotics: An Experiment with Kilobots. In Marco Dorigo, Mauro Birattari, Xiaodong Li, Manuel López-Ibáñez, Kazuhiro Ohkura, Carlo Pinciroli, and Thomas Stützle, editors, *Swarm Intelligence: 10th International Conference, ANTS 2016, Brussels, Belgium, September 7-9, 2016, Proceedings*, pages 185–196. Springer International Publishing, 2016.

Marco Dorigo, Vito Trianni, Erol Şahin, Roderich Groß, Thomas H Labella, Gianluca Baldassarre, Stefano Nolfi, Jean-Louis Deneubourg, Francesco Mondada, Dario Floreano, et al. Evolving self-organizing behaviors for a swarm-bot. *Autonomous Robots*, 17(2):223–245, 2004.

Marco Dorigo, Mauro Birattari, and Thomas Stutzle. Ant colony optimization. *IEEE computational intelligence magazine*, 1(4):28–39, 2006.

Marco Dorigo, Dario Floreano, Luca Maria Gambardella, Francesco Mondada, Stefano Nolfi, Tarek Baaboura, Mauro Birattari, Michael Bonani, Manuele Brambilla, Arne Brutschy, et al. Swarmanoid: a novel concept for the study of heterogeneous robotic swarms. *IEEE Robotics & Automation Magazine*, 20(4):60–71, 2013.

Keith L Doty, Reid R Harrison, and Instantiating Real. Sweep strategies for a sensory-driven. In *Behavior-Based Vacuum Cleaning Agent, AAAI 1993 Fall Symposium Series, Instantiating Real-World Agents, Research Triangle Park, Raleigh, NC*, pages 1–6. Citeseer, 1993.

Frederick Ducatelle, Gianni A Di Caro, Carlo Pinciroli, and Luca M Gambardella. Self-organized cooperation between robotic swarms. *Swarm Intelligence*, 5(2):73, 2011a.

Frederick Ducatelle, Gianni A Di Caro, Carlo Pinciroli, Francesco Mondada, and Luca Gambardella. Communication assisted navigation in robotic swarms: self-organization and cooperation. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4981–4988. IEEE, 2011b.

Russell Eberhart and James Kennedy. A new optimizer using particle swarm theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43. Ieee, 1995.

Albert Einstein. Investigations on the theory of the brownian movement. *Ann. der Physik*, 1905.

Nicholas Evans and Stephen C Levinson. The myth of language universals: Language diversity and its importance for cognitive science. *Behavioral and brain sciences*, 32(5):429–448, 2009.

Eliseo Ferrante, Manuele Brambilla, Mauro Birattari, and Marco Dorigo. Socially-mediated negotiation for obstacle avoidance in collective transport. In *Distributed autonomous robotic systems*, pages 571–583. Springer, 2013.

Eliseo Ferrante, Ali Emre Turgut, Alessandro Stranieri, Carlo Pinciroli, Mauro Birattari, and Marco Dorigo. A self-adaptive communication strategy for flocking in stationary and non-stationary environments. *Natural Computing*, 13(2):225–245, 2014.

Ziya Firat, Eliseo Ferrante, Nicolas Cambier, and Elio Tuci. Self-organised aggregation in swarms of robots with informed robots. In *International Conference on Theory and Practice of Natural Computing*, pages 49–60. Springer, 2018.

W Tecumseh Fitch and Marc D Hauser. Computational constraints on syntactic processing in a nonhuman primate. *Science*, 303(5656):377–380, 2004.

Dario Floreano, Sara Mitri, Stéphane Magnenat, and Laurent Keller. Evolutionary conditions for the emergence of communication in robots. *Current biology*, 17(6): 514–519, 2007.

Gianpiero Francesca and Mauro Birattari. Automatic design of robot swarms: achievements and challenges. *Frontiers in Robotics and AI*, 3:29, 2016.

Gianpiero Francesca, Manuele Brambilla, Arne Brutschy, Vito Trianni, and Mauro Birattari. Automode: A novel approach to the automatic design of control software for robot swarms. *Swarm Intelligence*, 8(2):89–112, 2014.

Gianpiero Francesca, Manuele Brambilla, Arne Brutschy, Lorenzo Garattoni, Roman Miletitch, Gaëtan Podevijn, Andreagiovanni Reina, Touraj Soleymani, Mattia Salvaro, Carlo Pinciroli, et al. Automode-chocolate: automatic design of control software for robot swarms. *Swarm Intelligence*, 9(2-3):125–152, 2015.

Gottlob Frege. 'on sense and reference'. *Basic Topics in the Philosophy of Language, Prentice-Hall, Englewood Cliffs, NJ*, pages 142–160, 1892.

Jorge Fuentes-Pacheco, José Ruiz-Ascencio, and Juan Manuel Rendón-Mancha. Visual simultaneous localization and mapping: a survey. *Artificial Intelligence Review*, 43(1):55–81, 2015.

Simon Garnier, Christian Jost, Raphaël Jeanson, Jacques Gautrais, Masoud Asadpour, Gilles Caprari, and Guy Theraulaz. Aggregation behaviour as a source of collective decision in a group of cockroach-like-robots. In *European Conference on Artificial Life*, pages 169–178. Springer, 2005.

Simon Garnier, Christian Jost, Jacques Gautrais, Masoud Asadpour, Gilles Caprari, Raphaël Jeanson, Anne Grimal, and Guy Theraulaz. The embodiment of cockroach aggregation behavior in a group of micro-robots. *Artificial life*, 14(4):387–408, 2008.

Simon Garnier, Jacques Gautrais, Masoud Asadpour, Christian Jost, and Guy Theraulaz. Self-organized aggregation triggers collective decision making in a group of cockroach-like robots. *Adaptive Behavior*, 17(2):109–133, 2009.

Melvin Gauci, Jianing Chen, Wei Li, Tony J Dodd, and Roderich Groß. Self-organized aggregation without computation. *The International Journal of Robotics Research*, 33(8):1145–1161, 2014.

Erving Goffman. *Frame analysis: An essay on the organization of experience.* Harvard University Press, 1974.

Sherri Goings, Emily PM Johnston, and Naozumi Hiranuma. The effect of communication on the evolution of cooperative behavior in a multi-agent system. In *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pages 999–1006. ACM, 2014.

Plerre-P Grassé. La reconstruction du nid et les coordinations interindividuelles chezbellicositermes natalensis etcubitermes sp. la théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs. *Insectes sociaux*, 6 (1):41–80, 1959.

Joseph H Greenberg. Some universals of grammar with particular reference to the order of meaningful elements. *Universals of language*, 2:73–113, 1963.

Roderich Groß, Elio Tuci, Marco Dorigo, Michael Bonani, and Francesco Mondada. Object transport by modular robots that self-assemble. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 2558–2564. IEEE, 2006.

Ken Hasselmann, Frédéric Robert, and Mauro Birattari. Automatic design of communication-based behaviors for robot swarms. In *International Conference on Swarm Intelligence*, pages 16–29. Springer, 2018.

Marc D Hauser, Noam Chomsky, and W Tecumseh Fitch. The faculty of language: what is it, who has it, and how did it evolve? *science*, 298(5598):1569–1579, 2002.

Andrew Howard, Maja J Matarić, and Gaurav S Sukhatme. Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem. In *Distributed Autonomous Robotic Systems 5*, pages 299–308. Springer, 2002.

Andrew David Irvine and Harry Deutsch. Russell's paradox. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, winter 2016 edition, 2016.

Nick Jakobi, Phil Husbands, and Inman Harvey. Noise and the reality gap: The use of simulation in evolutionary robotics. In *European Conference on Artificial Life*, pages 704–720. Springer, 1995.

Raphael Jeanson, Colette Rivault, Jean-Louis Deneubourg, Stephane Blanco, Richard Fournier, Christian Jost, and Guy Theraulaz. Self-organized aggregation in cockroaches. *Animal behaviour*, 69(1):169–180, 2005.

Kam-Chuen Jim and C Lee Giles. Talking helps: Evolving communicating agents for the predator-prey pursuit problem. *artificial life*, 6(3):237–254, 2000.

Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1), 1960.

Marco Karrer, Patrik Schmuck, and Margarita Chli. Cvi-slam—collaborative visual-inertial slam. *IEEE Robotics and Automation Letters*, 3(4):2762–2769, 2018.

Yoshiaki Katada. Evolutionary design method of probabilistic finite state machine for swarm robots aggregation. *Artificial Life and Robotics*, 23(4):600–608, 2018.

Paul Kay and Terry Regier. Resolving the question of color naming universals. *Proceedings of the National Academy of Sciences*, 100(15):9085–9089, 2003.

Simon Kirby. *Function, selection, and innateness: The emergence of language universals*. OUP Oxford, 1999.

Simon Kirby, Mike Dowman, and Thomas L Griffiths. Innateness and culture in the evolution of language. *Proceedings of the National Academy of Sciences*, 104(12): 5241–5245, 2007.

Piotr Alexeievich Kropotkin. *Mutual Aid: A Factor Or Evolution*. McClure Phillips and Company, 1902.

C Ronald Kube and Eric Bonabeau. Cooperative transport by ants and robots. *Robotics and autonomous systems*, 30(1-2):85–101, 2000.

Benjamin Kuipers. The spatial semantic hierarchy. *Artificial intelligence*, 119(1-2): 191–233, 2000.

Vijay Kumar and Nathan Michael. Opportunities and challenges with autonomous micro aerial vehicles. In *Robotics Research*, pages 41–58. Springer, 2017.

Vijay Kumar, Daniela Rus, and Sanjiv Singh. Robot and sensor networks for first responders. *IEEE Pervasive computing*, 3(4):24–33, 2004.

Daisuke Kurabayashi, Jun Ota, Tamio Arai, and Eiichi Yoshida. Cooperative sweeping by multiple mobile robots. In *Proceedings of IEEE international conference on robotics and automation*, volume 2, pages 1744–1749. IEEE, 1996.

Wei Li, Melvin Gauci, and Roderich Groß. Turing learning: a metric-free approach to inferring behavior and its application to swarms. *Swarm Intelligence*, 10(3): 211–243, 2016.

Antoine Ligot and Mauro Birattari. On mimicking the effects of the reality gap with simulation-only experiments. In *International Conference on Swarm Intelligence*, pages 109–122. Springer, 2018.

Manuel López-Ibáñez, Jérémie Dubois-Lacoste, Leslie Pérez Cáceres, Thomas Stützle, and Mauro Birattari. The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43–58, 2016. doi: 10.1016/j.orp.2016.09.002.

Vittorio Loreto, Andrea Baronchelli, and Andrea Puglisi. Mathematical modeling of language games. In *Evolution of communication and language in embodied agents*, pages 263–281. Springer, 2010.

Caroline Lyon, Chrystopher L Nehaniv, and Bob Dickerson. Clues from information theory indicating a phased emergence of grammar. In *Emergence of communication and language*, pages 71–85. Springer, 2007.

Nesrine Mahdoui, Vincent Frémont, and Enrico Natalizio. Cooperative exploration strategy for micro-aerial vehicles fleet. In *2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 180–185. IEEE, 2017.

Karl Marx. *Le capital*, volume 1. Lachâtre, 1875.

Nithin Mathews, Anders Lyhne Christensen, Eliseo Ferrante, Rehan O'Grady, and Marco Dorigo. Establishing spatially targeted communication in a heterogeneous robot swarm. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*, pages 939–946. International Foundation for Autonomous Agents and Multiagent Systems, 2010.

Siddharth Mayya, Gennaro Notomista, Dylan Shell, Seth Hutchinson, and Magnus Egerstedt. Achieving non-uniform densities in vibration driven robot swarms using phase separation theory. *CoRR*, abs/1902.10662, 2019. URL `http://arxiv.org/abs/1902.10662`.

Roman Miletitch, Vito Trianni, Alexandre Campo, and Marco Dorigo. Information aggregation mechanisms in social odometry. In *Artificial Life Conference Proceedings 13*, pages 102–109. MIT Press, 2013.

Roman Miletitch, Marco Dorigo, and Vito Trianni. Balancing exploitation of renewable resources by a robot swarm. *Swarm Intelligence*, 12(4):307–326, 2018.

Sifat Momen and A Sharkey. An ant-like task allocation model for a swarm of heterogeneous robots. In *Proceedings of the 2nd swarm intelligence algorithms and applications symposium (SIAAS)*, pages 31–38, 2009.

Francesco Mondada, Giovanni C Pettinaro, Andre Guignard, Ivo W Kwee, Dario Floreano, Jean-Louis Deneubourg, Stefano Nolfi, Luca Maria Gambardella, and Marco Dorigo. Swarm-bot: A new distributed robotic concept. *Autonomous robots*, 17(2-3):193–221, 2004.

Raymond Mulligan and Habib M Ammari. Coverage in wireless sensor networks: a survey. *Network Protocols & Algorithms*, 2(2):27–53, 2010.

Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33(5): 1255–1262, 2017.

Andrew L Nelson, Gregory J Barlow, and Lefteris Doitsidis. Fitness functions in evolutionary robotics: A survey and analysis. *Robotics and Autonomous Systems*, 57(4):345–370, 2009.

Partha Niyogi and Robert C Berwick. Evolutionary consequences of language learning. *Linguistics and Philosophy*, 20(6):697–719, 1997.

Stefano Nolfi and Dario Floreano. *Evolutionary robotics: The biology, intelligence, and technology of self-organizing machines*. MIT press, 2000.

Stefano Nolfi and Marco Mirolli. *Evolution of communication and language in embodied agents*. Springer Science & Business Media, 2009.

Nikita Noskov, Evert Haasdijk, Berend Weel, and Agoston Endre Eiben. Monee: Using parental investment to combine open-ended and task-driven evolution. In *European Conference on the Applications of Evolutionary Computation*, pages 569–578. Springer, 2013.

Shervin Nouyan and Marco Dorigo. Chain based path formation in swarms of robots. In *International Workshop on Ant Colony Optimization and Swarm Intelligence*, pages 120–131. Springer, 2006.

Shervin Nouyan, Roderich Groß, Michael Bonani, Francesco Mondada, and Marco Dorigo. Teamwork in self-organized robot colonies. *IEEE Transactions on Evolutionary Computation*, 13(4):695–711, 2009.

Martin A Nowak and Natalia L Komarova. Towards an evolutionary theory of language. *Trends in cognitive sciences*, 5(7):288–295, 2001.

Martin A Nowak, Natalia L Komarova, and Partha Niyogi. Evolution of universal grammar. *Science*, 291(5501):114–118, 2001.

Rehan O'Grady, Anders Lyhne Christensen, and Marco Dorigo. Swarmorph: multirobot morphogenesis using directional self-assembly. *IEEE Transactions on Robotics*, 25(3):738–743, 2009.

Michael J Owren, Drew Rendall, and Michael J Ryan. Redefining animal signaling: influence versus information in communication. *Biology & Philosophy*, 25(5):755–780, 2010.

Rehan O'Grady, Roderich Groß, Anders Lyhne Christensen, and Marco Dorigo. Self-assembly strategies in a group of autonomous mobile robots. *Autonomous Robots*, 28(4):439–455, 2010.

Sotiris Papatheodorou and Anthony Tzes. Theoretical and experimental collaborative area coverage schemes using mobile agents. In *Applications of Mobile Robots*. IntechOpen, 2018.

Iñaki Fernández Pérez, Amine Boumaza, and François Charpillet. Maintaining diversity in robot swarms with distributed embodied evolution. In *International Conference on Swarm Intelligence*, pages 395–402. Springer, 2018.

Carlo Pinciroli, Vito Trianni, Rehan O´Grady, Giovanni Pini, Arne Brutschy, Manuele Brambilla, Nithin Mathews, Eliseo Ferrante, Gianni Di Caro, Frederick Ducatelle, et al. Argos: a modular, parallel, multi-engine simulator for multi-robot systems. *Swarm intelligence*, 6(4):271–295, 2012.

Steven Pinker and Paul Bloom. Natural language and natural selection. *Behavioral and brain sciences*, 13(4):707–727, 1990.

Tahmid Rahman and Shantanu Jain. Evolving cooperative communicative behaviour.

Stefano Crespi Reghizzi, Luca Breveglieri, and Angelo Morzenti. *Formal languages and compilation*. Springer, 2013.

Andreagiovanni Reina, Alex J Cope, Eleftherios Nikolaidis, James AR Marshall, and Chelsea Sabo. Ark: Augmented reality for kilobots. *IEEE Robotics and Automation letters*, 2(3):1755–1761, 2017.

Joe R Riley, Uwe Greggers, Alan D Smith, Don R Reynolds, and Randolf Menzel. The flight paths of honeybees recruited by the waggle dance. *Nature*, 435(7039): 205, 2005.

Pierre Roduit. Trajectory analysis using point distribution models. 2009.

André Rousseau. Quelques aspects de la philosophie du langage (frege, husserl, wittgenstein) et leur incidence en linguistique. *Germanica*, (8):203–241, 1990.

Michael Rubenstein, Christian Ahler, and Radhika Nagpal. Kilobot: A low cost scalable robot system for collective behaviors. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3293–3298. IEEE, 2012.

Samuel Rutishauser, Nikolaus Correll, and Alcherio Martinoli. Collaborative coverage using a swarm of networked miniature robots. *Robotics and Autonomous Systems*, 57(5):517–525, 2009.

Sajad Saeedi, Michael Trentini, Mae Seto, and Howard Li. Multiple-robot simultaneous localization and mapping: A review. *Journal of Field Robotics*, 33(1):3–46, 2016.

Erol Şahin. Swarm robotics: From sources of inspiration to domains of application. In *International workshop on swarm robotics*, pages 10–20. Springer, 2004.

Thomas D. Seeley, P. Kirk Visscher, Thomas Schlegel, Patrick M. Hogan, Nigel R. Franks, and James A. R. Marshall. Stop signals provide cross inhibition in collective decision-making by honeybee swarms. *Science*, 335(6064):108–111, 2012. ISSN 0036-8075. doi: 10.1126/science.1210361. URL `https://science.sciencemag.org/content/335/6064/108`.

CE Shannon. (1948),” a mathematical theory of communication”, bell system technical journal, vol. 27, pp. 379-423 & 623-656, july & october. 1948.

Fernando Silva, Paulo Urbano, Luís Correia, and Anders Lyhne Christensen. odneat: An algorithm for decentralised online evolution of robotic controllers. *Evolutionary Computation*, 23(3):421–449, 2015.

Andrew DM Smith. Models of language evolution and change. *Wiley Interdisciplinary Reviews: Cognitive Science*, 5(3):281–293, 2014.

Kenny Smith, Henry Brighton, and Simon Kirby. Complex systems in language evolution: the cultural emergence of compositional structure. *Advances in Complex Systems*, 6(04):537–558, 2003.

Ricard V Solé, Bernat Corominas-Murtra, and Jordi Fortuny. Diversity, competition, extinction: the ecophysics of language change. *Journal of The Royal Society Interface*, 7(53):1647–1664, 2010.

Onur Soysal and Erol Sahin. Probabilistic aggregation strategies in swarm robotic systems. In *Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE*, pages 325–332. IEEE, 2005.

Luc Steels. A self-organizing spatial vocabulary. *Artificial life*, 2(3):319–332, 1995.

Luc Steels. Language games for autonomous robots. *IEEE Intelligent systems*, 16 (5):16–22, 2001.

Luc Steels. Evolving grounded communication for robots. *Trends in cognitive sciences*, 7(7):308–312, 2003.

Luc Steels. The emergence and evolution of linguistic structure: from lexical to grammatical communication systems. *Connection science*, 17(3-4):213–230, 2005.

Luc Steels. Modeling the cultural evolution of language. *Physics of Life Reviews*, 8 (4):339–356, 2011.

Luc Steels and Frédéric Kaplan. Stochasticity as a source of innovation in language games. In *Proceedings of artificial life vi*, pages 368–376, 1998.

Luc Steels and Martin Loetzsch. Perspective alignment in spatial language. *Spatial language and dialogue*, pages 70–89, 2008.

Luc Steels and Martin Loetzsch. The grounded naming game. *Experiments in cultural language evolution*, 3:41–59, 2012.

Martin Stokhof. Can natural language be captured in a formal system? In *Introduction to Formal Philosophy*, pages 273–288. Springer, 2018.

K Stoy. Using situated communication in distributed autonomous mobile robotics. In *SCAI'01: Proceedings of the Seventh Scandinavian Conference on Artificial Intelligence*, pages 44–52, 2001.

Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT press, 2005.

Vito Trianni and Marco Dorigo. Self-organisation and communication in groups of simulated and physical robots. *Biological cybernetics*, 95(3):213–231, 2006.

Vito Trianni, Roderich Groß, Thomas H Labella, Erol Şahin, and Marco Dorigo. Evolving aggregation behaviors in a swarm of robots. In *European Conference on Artificial Life*, pages 865–874. Springer, 2003.

Vito Trianni, Stefano Nolfi, and Marco Dorigo. Cooperative hole avoidance in a swarm-bot. *Robotics and Autonomous Systems*, 54(2):97–103, 2006.

Vito Trianni, Stefano Nolfi, and Marco Dorigo. Evolution, Self-organization and Swarm Robotics. In *Swarm Intelligence*, pages 163–191. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

Vito Trianni, Daniele De Simone, Andreagiovanni Reina, and Andrea Baronchelli. Emergence of consensus in a multi-robot network: from abstract models to empirical validation. *IEEE Robotics and Automation Letters*, 1(1):348–353, 2016.

Elio Tuci. An investigation of the evolutionary origin of reciprocal communication using simulated autonomous agents. *Biological Cybernetics*, 101(3):183–199, 2009.

Elio Tuci, Tomassino Ferrauto, Arne Zeschel, Gianluca Massera, and Stefano Nolfi. An experiment on behavior generalization and the emergence of linguistic compositionality in evolving robots. *IEEE Transactions on Autonomous Mental Development*, 3(2):176–189, 2011.

Elio Tuci, Muhanad Hayder Mohammed Alkilabi, and Otar Akanyeti. Cooperative object transport in multi-robot systems: A review of the state-of-the-art. *Frontiers in Robotics and AI*, 5:59, 2018.

Gabriele Valentini, Eliseo Ferrante, and Marco Dorigo. The best-of-n problem in robot swarms: Formalization, state of the art, and novel perspectives. *Frontiers in Robotics and AI*, 4:9, 2017. ISSN 2296-9144.

Gabriele Valentini, Anthony Antoun, Marco Trabattoni, Bernát Wiandt, Yasumasa Tamura, Etienne Hocquard, Vito Trianni, and Marco Dorigo. Kilogrid: a novel experimental environment for the kilobot robot. *Swarm Intelligence*, 12(3):245–266, 2018.

Marcos AM Vieira, Luiz FM Vieira, Linnyer Beatrys Ruiz, Antonio Alfredo Ferreira Loureiro, Antônio Otávio Fernandes, and José Marcos S Nogueira. Scheduling nodes in wireless sensor networks: A voronoi approach. In *28th Annual IEEE International Conference on Local Computer Networks, 2003. LCN'03. Proceedings.*, pages 423–429. IEEE, 2003.

Richard A Watson, Sevan G Ficici, and Jordan B Pollack. Embodied evolution: Distributing an evolutionary algorithm in a population of robots. *Robotics and Autonomous Systems*, 39(1):1–18, 2002.

Ludwig Wittgenstein. Philosophical investigations, trans. *GEM Anscombe*, 261:49, 1953.

Kai M Wurm, Cyrill Stachniss, and Wolfram Burgard. Coordinated multi-robot exploration using a segmentation of the environment. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1160–1165. IEEE, 2008.

Chern Han Yong and Risto Miikkulainen. Cooperative coevolution of multi-agent systems. *University of Texas at Austin, Austin, TX*, 2001.

Rola Younes. *Introduction à Wittgenstein*. La Découverte, 2016.