

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E
INFORMÁTICA INDUSTRIAL**

RUBISSON DUARTE LAMPERTI

**DOUBLE WAVE SWARM: UMA ESTRATÉGIA DE COMUNICAÇÃO
EM SISTEMAS MULTI-ROBÔS PARA TAREFAS DE NAVEGAÇÃO EM
FORMAÇÃO**

TESE

CURITIBA

2023

RUBISSON DUARTE LAMPERTI

**DOUBLE WAVE SWARM: UMA ESTRATÉGIA DE
COMUNICAÇÃO EM SISTEMAS MULTI-ROBÔS PARA
TAREFAS DE NAVEGAÇÃO EM FORMAÇÃO**

**DOUBLE WAVE SWARM: A COMMUNICATION STRATEGY IN
MULTI-ROBOT SYSTEMS FOR NAVIGATION TASKS IN
FORMATION**

Tese apresentada ao Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial(CPGEI), da Universidade Tecnológica Federal do Paraná (UTFPR) como requisito parcial para obtenção do grau de "Doutor em Ciências" – Área de Concentração: Engenharia De Automação E Sistemas.

Orientador(a): Prof^ª. Dr^ª. Lúcia Valeria Ramos de Arruda

CURITIBA

2023



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es).

Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Curitiba



RUBISSON DUARTE LAMPERTI

**DOUBLE WAVE SWARM: UMA ESTRATÉGIA DE COMUNICAÇÃO EM SISTEMAS MULTI-ROBÔS PARA
TAREFAS DE NAVEGAÇÃO EM FORMAÇÃO**

Trabalho de pesquisa de doutorado apresentado como requisito para obtenção do título de Doutor Em Ciências da Universidade Tecnológica Federal do Paraná (UTFPR). Área de concentração: Engenharia De Automação E Sistemas .

Data de aprovação: 11 de Dezembro de 2023

Dra. Lucia Valeria Ramos De Arruda, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Andre Schneider De Oliveira, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Erivelton Geraldo Nepomuceno, Doutorado - Maynooth University - Mu

Dr. Flavio Neves Junior, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Gustavo Medeiros Freitas, Doutorado - Universidade Federal de Minas Gerais (Ufmg)

Documento gerado pelo Sistema Acadêmico da UTFPR a partir dos dados da Ata de Defesa em 11/12/2023.

I dedicate this work to my family, my friends
and to all those who fight for a future better
through science and education.

ACKNOWLEDGEMENTS

This work would not have been possible without the invaluable support of several people and institutions, to whom I express my sincere tribute. I recognize that it is impossible to thank everyone who played a significant role in this important phase of my academic life. Therefore, in advance, I apologize to those who are not mentioned in this brief text, but who are equally worthy of my gratitude.

To my beloved family, I express deep gratitude for the constant affection, encouragement and support throughout the development of this thesis and at all crucial moments of my career.

To my dedicated advisor, Prof. Dr. Lúcia Valéria Ramos de Arruda, I express my gratitude for guiding me along the necessary paths and placing confidence in my potential.

I sincerely thank you for the collaboration, insights, and support that made this academic journey possible.

“Amidst adversities, we find true opportunities.
In challenges, we forge our resilience. Through
mutual trust, we weave the foundations of
tomorrow. May this academic journey inspire
overcoming, strengthen faith in progress, and
illuminate the path toward a promising future.”

RESUMO

LAMPERTI, Rubisson Duarte. **DOUBLE WAVE SWARM: UMA ESTRATÉGIA DE COMUNICAÇÃO EM SISTEMAS MULTI-ROBÔS PARA TAREFAS DE NAVEGAÇÃO EM FORMAÇÃO**. 2023. 130 f. Tese (Doutorado em Engenharia Elétrica e Informática Industrial) – Universidade Tecnológica Federal do Paraná. Curitiba, 2023.

Multi-robôs podem realizar tarefas complexas, como exploração, forrageamento, formação e navegação em grupo. A comunicação eficiente entre robôs pode contribuir para a realização de tarefas coletivas por meio da troca eficiente de mensagens. Este trabalho propõe um método baseado no Wave Swarm para a tarefa de formação inspirada no Problema do Caixeiro Viajante. Dada uma distribuição aleatória de robôs em um enxame, o Problema do Caixeiro Viajante é usado para estabelecer a relação parental entre os robôs do enxame. Além disso, uma estratégia de comunicação para robôs de enxame denominada Double-Wave Swarm é proposta neste trabalho. O Double-Wave Swarm utiliza a troca de mensagens entre robôs vizinhos para a sincronização de subtarefas e a troca de informações para a execução de tarefas. O Double-Wave Swarm é uma melhoria do Wave Swarm. A abordagem líder-seguidor é usada em ambiente desconhecido e comunicação local para validar ambas as propostas. Experimentos com o robô simulador CoppeliaSim (V-REP) validam a proposta inspirada no Problema do Caixeiro Viajante e o Double-Wave Swarm.

Palavras-chave: robótica de enxame. comunicação. controle de formação. localização.

ABSTRACT

LAMPERTI, Rubisson Duarte. **DOUBLE WAVE SWARM: A COMMUNICATION STRATEGY IN MULTI-ROBOT SYSTEMS FOR NAVIGATION TASKS IN FORMATION**. 2023. 130 p. Thesis (PhD in Electrical and Computer Engineering) – Universidade Tecnológica Federal do Paraná. Curitiba, 2023.

Multi-robots can perform complex tasks such as exploration, foraging, formation, and flocking. Efficient communication between robots can contribute to bringing about collective tasks through efficient message exchange. This work proposes a method based on the Wave Swarm for the formation task inspired by the Traveling Salesman Problem. Given a random distribution of robots in a swarm, the Traveling Salesman Problem is used to establish the parental relationship between the robots in the swarm. Also, we propose a communication strategy for swarm robots, the Double-Wave Swarm. The Double-Wave Swarm uses the exchange of messages between neighbors robots for the synchronization of subtasks and the exchange of information for the execution of tasks. The Double-Wave Swarm is an improvement of a prior Wave Swarm communication approach. We adopted the leader-follower approach in an unknown environment and local communication to validate both proposals. Experiments with the robot simulator CoppeliaSim (V-REP) validate the proposed inspired by the Traveling Salesman Problem and the Double-Wave Swarm.

Keywords: swarm robotic system. communication. formation control. localization.

LIST OF ALGORITHMS

Algoritmo 1 – Wave Swarm Algorithm	47
Algoritmo 2 – Control Algorithm	57
Algoritmo 3 – Navigation Task	57
Algoritmo 4 – Wave Recruitment Algorithm	60
Algoritmo 5 – <i>Q</i> -Learning Algorithm	64
Algoritmo 6 – Double-Wave Swarm Algorithm	78
Algoritmo 7 – Son robot rotation: <i>rotationSon()</i>	83
Algoritmo 8 – Alignment: <i>alignment()</i>	83
Algoritmo 9 – Navigation Task: <i>navigation()</i>	85
Algoritmo 10 – Control Algorithm: <i>controlPose()</i>	89

LISTA DE ILUSTRAÇÕES

Figura 1 – Simulation of robots in the shape of the letter C	23
Figura 2 – Experiment with 10 robots	24
Figura 3 – Scenario of the experiment proposed in the V-REP simulator	25
Figura 4 – Aggregation task steps: a) robots in a random position; b) robots in an intermediate position; c) robots in a final position	25
Figura 5 – Performing the formation task - images the transition (a - b- c) between the different formations using the ceiling-mounted camera system	26
Figura 6 – Self-assembly task: nine robots generating a structure with 4 legs in locomotion	26
Figura 7 – The trajectory of four robots with obstacle avoidance	27
Figura 8 – Exploration task: trajectory and maps that each robot built during online exploration. The black spots correspond to observable obstacles, and the gray spots correspond to unobservable obstacles	28
Figura 9 – Foraging task: (a) Simulation environment without obstacles; (b) Simulation environment with obstacle - the arrows are the agents, the white cell in the center is the nest, the white circles are the food and the gray rectangular shapes are the obstacles	28
Figura 10 – Flocking task: movement of three robots in a circle - the red and blue lines represent the trajectory of the virtual leader and the centroid	29
Figura 11 – Transport task: illustration of the initial condition of the simulation of an object moved by a group of four robots	29
Figura 12 – Distribution of the pheromone in a trajectory with an obstacle: a) one food source, and b) with two food sources	30
Figura 13 – Results of experiments for a variable triangular formation with obstacle avoidance, using the proposed method	30
Figura 14 – Experiment results for a triangular formation with obstacle avoidance	31
Figura 15 – Multi-robots formation control. robot 1: (green); robot 2: (black); robot 3: (blue); robot 4: (magenta)	32
Figura 16 – Different instants for the collective movement of the swarm: the red circle and yellow circles are the leader and followers, and the black rectangles are the obstacles. The leader moves along the red arrow while following the followers	34
Figura 17 – Navigation in formation with obstacle avoidance	36
Figura 18 – Precedence diagram for two subtasks	37
Figura 19 – The communication steps used in the cluster topology	38
Figura 20 – Result of the numerical simulation: a) movement of the robots, and b) communication graphs between the robots	38
Figura 21 – The trajectory followed by the robots when changing the leader, following a V formation	39
Figura 22 – Location estimation algorithm using a beacon - illustration of visible and non-visible robots	40
Figura 23 – DPE and EKF present estimates individually and relative detection and communication graphs	41
Figura 24 – e-puck2 robot	42
Figura 25 – Kilobot robot	43

Figura 26 – Swarmanoid: Foot-Bot robot	43
Figura 27 – Wave Swarm - parental relationship modeled as a graph	46
Figura 28 – subtasks set	47
Figura 29 – Diagram of sending and receiving messages for two subtasks.	48
Figura 30 – Exchange messages by Wave Swarm: parental relationship initial	53
Figura 31 – Exchange messages by Wave Swarm: a new parental relationship achieved through the TSP solution	53
Figura 32 – Areas surrounding the robots called Green Area, Gray Area, and Red Area.	55
Figura 33 – State Machine - navigation	56
Figura 34 – Robots in formation: the distance and angle to the reference point	58
Figura 35 – Robots in random position and unknown neighborhood - illustration of obstacle detection area	59
Figura 36 – Recruitment phase completed	60
Figura 37 – Solution of the Traveling Salesman Problem: the red line shows the shortest possible path that connects all robots	61
Figura 38 – Experiment with twenty robots: initial position	66
Figura 39 – Simulator CoppeliaSim (VRep): e-Puck	66
Figura 40 – Experiment berlin52-RL, variable $t_b(s)$: mean equal to 0.256, median equal to 0.245, skewness equal 1.01 and kurtosis equal to 1.07 - a) Histogram; and, b) Box plot	68
Figura 41 – Experiment berlin-GA, variable $t_b(s)$: mean equal to 8.05, median equal to 7.79, skewness equal 1.81 and kurtosis equal to 2.11 - a) Histogram; and, b) Box plot	69
Figura 42 – Experiment berlin52-RL, variable $S(\%)$: mean equal to 13.9, median equal to 14.3, skewness equal -0.694 and kurtosis equal to 0.811 - a) Histogram; and, b) Box plot	69
Figura 43 – Experiment berlin52-GA, variable $S(\%)$: mean equal to 45.8, median equal to 46.2, skewness equal 0.0686 and kurtosis equal to -0.909 - a) Histogram; and, b) Box plot	70
Figura 44 – Experiments berlin52-RL and berlin52-GA, variable $t_b(s)$: mean equal to 0.255, and 8.045 - Bootstrap mean with 95% confidence interval	70
Figura 45 – Experiments berlin52-RL and berlin52-GA, variable $S(\%)$: mean equal to 13.879, and 45.774 - Bootstrap mean with 95% confidence interval	71
Figura 46 – Experiments A (inline), B (square), C (circle), and N (letter N): bar graph of the number of successfully completed simulations, N_s	72
Figura 47 – Experiment C20, variable $t_s(s)$: mean equal to 412.281, median equal to 423.377, skewness equal -0.245 and kurtosis equal to 2.485 - a) Histogram; and, b) Box plot	72
Figura 48 – Experiment C20, variable $D_{dist}(m)$: mean equal to 8.509, median equal to 8.346, skewness equal to +0.039 and kurtosis equal to 2.362 - a) Histogram; and, b) Box plot	73
Figura 49 – Experiment C20, variable N_{msg} : mean equal to 2155.820, median equal to 2184.458, skewness equal to -0.216 and kurtosis equal to 2.443 - a) Histogram; and, b) Box plot	73
Figura 50 – Experiments C6, C10, C15, and C20: mean equal to 203.789, 249.372, 328.732 and 410.153 - Bootstrap mean with 95% confidence interval of the variable t_s	74

Figura 51 – Experiments C6, C10, C15, and C20: mean equal to 4.328, 5.185, 6.775 and 8.509 - Bootstrap mean with 95% confidence interval of the variable D_{dist}	74
Figura 52 – Experiments C6, C10, C15, and C20: mean equal to 1072.38, 1300.361, 1708.613 and 2155.878 - Bootstrap mean with 95% confidence interval of the variable N_{msg}	75
Figura 53 – Experiment C20: Spearman’s Correlation - $\rho = 0.812$ and P -value < 0.001 ; Linear regression (predictor variable D_q and dependent variable t_s) - model fit measures: $R = 0.809$, $R^2 = 0.654$, $F = 79.5$, $df^1 = 1$, $df^2 = 42$ and P - value < 0.001	75
Figura 54 – Experiment C20: Spearman’s Correlation - $\rho = 0.805$ and P -value < 0.001 ; Linear regression (predictor variable D_q and dependent variable D_{dist}) - model fit measures: $R = 0.801$, $R^2 = 0.642$, $F = 70.6$, $df^1 = 1$, $df^2 = 42$ and P - value < 0.001	76
Figura 55 – Experiment C20: Spearman’s Correlation - $\rho = 0.803$ and P -value < 0.001 ; Linear regression (predictor variable D_q and dependent variable D_{Nmsg}) - model fit measures: $R = 0.809$, $R^2 = 0.659$, $F = 79.6$, $df^1 = 1$, $df^2 = 42$ and P - value < 0.001	76
Figura 56 – Double-Wave Swarm - parental relationship modeled by a cyclic graph: the black line shows the relationship between the Father robots and the Son robots; and the red line points out the relationship between the Friends robots	78
Figura 57 – Robots unaligned with their Father: R_A (Father), R_B (Son 01), e, R_C (Son 02)	82
Figura 58 – Robots aligned with their Father: R_A (father), R_B (Son 01), e, R_C (Son 02) .	82
Figura 59 – Alignment task - the lines connecting the robots show the parental relationship between a Father robot and a Son robot	84
Figura 60 – Navigation with obstacle avoidance and total communication failure between Father and Son robots signaled with a red circle	86
Figura 61 – Navigation with obstacle avoidance and indirect communication between Father and Son robots through a Friend robot, signaled with a blue circle . .	86
Figura 62 – Sensors coverage area, forming a local communication among the R1, R2 and R3 robots	87
Figura 63 – Illustration of a complete loss of communication between robots R10, R6 and R9	88
Figura 64 – Triangular formation composed of ten e-pucks robots	88
Figura 65 – State Machine - navigation task	89
Figura 66 – Robot swarm performing navigation task	90
Figura 67 – Detection areas surrounding the robots	91
Figura 68 – Simulator CoppeliaSim (V-REP): e-puck	92
Figura 69 – Simulator CoppeliaSim (V-REP): e-puck with eight IR sensors	93
Figura 70 – Illustration of the alignment task	94
Figura 71 – Scenes from the proposed experiments	95
Figura 72 – Histogram of the number of message exchanges for the alignment task. . . .	97
Figura 73 – Boxplot of the number of message exchanges - Experiments A1 and A2: average equal to 1272.2 and 2099.2, median equal to 1265.2 and 2105.4, and IQR equal to 196 and 261	98
Figura 74 – Estimate of the average of the message number exchanged, N_{msg} , with a 95% bootstrap confidence interval: experiments A1 and A2	98
Figura 75 – Estimate of the median of the message number exchanged, N_{msg} , with a 95% bootstrap confidence interval: experiments A1 and A2	98

Figura 76 – Simulation time histogram for the alignment task	99
Figura 77 – Simulation time boxplot - Experiments A1 and A2: median equal to 4.494 and 5.017 and IQR equal to 0.562 and 0.689	100
Figura 78 – Estimate of the average of the simulation time, t_s , with a 95% bootstrap confidence interval: experiments A1 and A2	100
Figura 79 – Estimate of the median of the simulation time, t_s , with a 95% bootstrap confidence interval: experiments A1 and A2	101
Figura 80 – Histogram of the number of robot that completed the navigation, β	102
Figura 81 – Boxplot of the number of robot that completed the navigation, β - Experiments B2 and B3: median equal to 3 and 4, and IQR equal to 2.75 and 3.50	103
Figura 82 – Estimate of the average number of robots completing the navigation, β , with a 95% bootstrap confidence interval: experiment B2 and B3	103
Figura 83 – Estimate of the median number of robots completing the navigation, β , with a 95% bootstrap confidence interval: experiment B2 and B3	103
Figura 84 – Histogram of the number of robot that completed the navigation, β	104
Figura 85 – Boxplot of the number of robot that completed the navigation, β - Experiments C2 and C3: median equal to 7 and 8, and IQR equal to 2 and 2.75	105
Figura 86 – Estimate of the average of the number of robot that completed the navigation, β , with a 95% bootstrap confidence interval: experiments C2 and C3	105
Figura 87 – Estimate of the median of the number of robot that completed the navigation, β , with a 95% bootstrap confidence interval: experiments C2 and C3	105
Figura 88 – Histogram of the number of robot that completed the navigation, β	107
Figura 89 – Boxplot of the number of robot that completed the navigation, β - Experiments B and C: median equal to 4 and 7, and IQR equal to 3.5 and 3	108
Figura 90 – Estimate of the average of the number of robot that completed the navigation, β , with a 95% bootstrap confidence interval: Experiment B and C	108
Figura 91 – Estimate of the median of the number of robots that completed the navigation, β , with a 95% bootstrap confidence interval: Experiment B and C	108
Figura 92 – Percentage of number of simulations versus β - experiments B (Wave Swarm) and C (Double-Wave Swarm)	109

LISTA DE TABELAS

Tabela 1 – Time complexity of exact methods for TSP.	54
Tabela 2 – Time complexity of heuristic algorithms for TSP.	54
Tabela 3 – Time complexity of meta-heuristic algorithms for TSP.	55

LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

SIGLAS

ACO	Ant Colony Optimization
CDTA	Clustered Dynamic Task Allocation
DPE	Decentralized Pose Estimation
DWS	Double-Wave Swarm
GA	Genetic Algorithm
GPS	Global Positioning System
ID	Identification
IQR	Interquartile Range
IR	Infrared Signal
PSO	Particle Swarm Optimization
RL	Reinforcement Learning
ROS	Robot Operation System
SA	Simulated Annealing
TSP	Traveling Salesman Problem
V-REP	Virtual Robot Experimentation Platform
WS	Wave Swarm

LISTA DE SÍMBOLOS

LETRAS LATINAS

l	Total number of level of the tree-like graph
$M _W$	Number of messages, using Wave Swarm
$M _{DWS}$	Number of messages, using Double-Wave Swarm
N	Total number of robots
$O(N)$	Message and time complexity
$T(N) _W$	Total transmission time, using Wave Swarm
$T(N) _{DWS}$	Total transmission time, using Double-Wave Swarm

SUMÁRIO

1	INTRODUCTION	18
1.1	OBJECTIVE	19
1.2	THESIS CONTRIBUTIONS	20
1.3	RELATED PAPERS	21
1.4	THESIS STRUCTURE	21
2	SWARM ROBOTICS	22
2.1	CENTRALIZED AND DISTRIBUTED SYSTEM	23
2.2	COLLECTIVE BEHAVIOR	24
2.2.1	Space Organization	25
2.2.2	Navigation	27
2.3	FORMATION TASK	29
2.4	FLOCKING TASK	32
2.5	COMMUNICATION	35
2.5.1	Global communication	35
2.5.2	Local Communication	36
2.6	LOCALIZATION	39
2.7	ROBOTIC PLATFORMS	41
2.7.1	Physical Robots	42
2.7.2	Robot Simulators	44
3	WAVE SWARM	46
3.1	MESSAGES EXCHANGED BETWEEN ROBOTS	48
3.2	CONTRIBUTIONS BASED ON WAVE SWARM	49
4	A STRATEGY BASED ON WAVE SWARM FOR THE FORMATION TASK INSPIRED BY THE TRAVELING SALESMAN PROBLEM	51
4.1	TRAVELING SALESMAN PROBLEM	51
4.1.1	TSP solution	53
4.2	DEVELOPED STRATEGY FOR SWARM FORMATION TASK	55
4.2.1	Navigation	55
4.2.2	Desired robot group shapes	58
4.2.3	Leader robot's role	58
4.2.4	Strategy's phases	59
4.2.5	Communication failures	62
4.3	EXPERIMENTS AND METRICS FOR THE TSP	63
4.3.1	Metrics	65
4.4	EXPERIMENTS SET UP AND METRICS FOR PROPOSED STRATEGY	65
4.4.1	Metrics	67
4.5	SIMULATED RESULTS	67
4.5.1	Experiments for the TSP	68
4.5.2	Experiments for the proposed strategy with RL	71

5	DISTRIBUTED STRATEGY FOR COMMUNICATION BETWEEN MULTIPLE ROBOTS DURING FORMATION NAVIGATION TASK. .	77
5.1	DOUBLE-WAVE SWARM	77
5.1.1	Messages exchanged between robots	78
5.2	CONNECTIVITY	80
5.3	SWARM FORMATION NAVIGATION WITH DOUBLE-WAVE COMMUNICATION	81
5.3.1	Alignment task	81
5.3.2	Navigation task	83
5.3.3	Local communication	86
5.3.4	Rigid and semi-rigid formation	87
5.3.5	Formation control	88
5.3.6	Obstacles detection	89
5.4	SIMULATED EXPERIMENTS	92
5.4.1	Experiments A: alignment task	93
5.4.2	Experiments B: navigation task - Wave Swarm	93
5.4.3	Experiments C: navigation task - Double-Wave Swarm	94
5.4.4	Metrics	95
5.5	RESULTS AND DISCUSSION	96
5.5.1	Experiments A1 and A2 - alignment task	96
5.5.2	Experiment B2 and B3 - navigation task	101
5.5.3	Experiment C2 and C3 - navigation task	101
5.5.4	A comparison between experiments B and C.	104
6	CONCLUSION	110
6.1	FUTURE WORK	111
	REFERÊNCIAS	112

1 INTRODUCTION

In the multi-robot environment, there are several challenges, such as collision-free movement of robots (LAFMEJANI; BERMAN, 2021), maintenance of communication between robots (GUO *et al.*, 2019), sensing (MATSUKA *et al.*, 2021) and the lack of complete knowledge about the environment (YOUSSEFI; ROUHANI, 2021).

The formation task refers to the behavior of a group of mobile autonomous robots that aim to achieve a specific formation while moving along a trajectory. Among the existing approaches to carry formation tasks are: behavior-based (LEE; CHWA, 2017), virtual leader approach (DIN *et al.*, 2018), and leader-follower (URREA; MATTEODA, 2020).

The formation navigation of multiple robots is a cooperative task in which robots move towards a point while maintaining a pre-established shape of the team (OLCAY *et al.*, 2020; ELKILANY *et al.*, 2020). However, performing this task in an autonomous and distributed way can become complex, considering the total lack of knowledge of the environment, accurate information about the collective and individual location, constrained sensing, and local communication. In particular, local communication takes place between robots that are within an area covered by their transmitters. Another essential factor is the presence of unexpected obstacles that can hamper the navigation task.

In the presence of obstacles, robots must, in addition to avoiding collisions between them, also deviate from obstacles (SU *et al.*, 2021; ALMEIDA *et al.*, 2020). For this, during formation navigation, the robots must implement maneuvers that prevent crashes while maintaining the collective and coordinated navigation of the group.

These maneuvers often prioritize anti-collision actions over group formation. The change in shape formation leads to the modification of the topological structure involved in local communication (KHATERI *et al.*, 2020; ALMEIDA *et al.*, 2019; NEDJAH *et al.*, 2021). This modification can be minimal or severe, and it can compromise the maintenance of the formation due to a restructuring in the communication between the robots.

We can compare the local communication during the formation navigation of multiple robots with the communication between networked computers if we consider the robot swarm a distributed and independent processing system.

The elementary distributed algorithms are part of the theory of computer networks and deal with the propagation of messages in distributed systems. The wave propagation elementary

algorithms achieve global synchronization between processes by triggering the execution of some event for each process and transmitting data through messages exchange, ensuring the participation of all processes (TEL, 2000).

The wave propagation strategy is adopted to develop a new approach for the navigation of multiple robots in formation, originating the Wave Swarm algorithm (WS) (SILVA-JR; NEDJAH, 2017; SILVA-JR.; NEDJAH, 2016; SILVA-JR; NEDJAH, 2015). This approach uses a tree-like graph to model the swarm configuration in which each robot corresponds to a graph node, and the graph edges constitute the local communication links. However, the formation navigation of multiple robots may present group shape variations, mainly in the presence of obstacles. During obstacle avoidance maneuvers, some local communication links are frequently broken, compromising the fixed topological communication structure and preventing the correct navigation task performance. Thus, in this work, we propose an improvement in the existing solution (WS) to increase the connectivity of the communication network between the robots to fulfill navigation with obstacles, called Double-Wave Swarm (DWS).

We also introduce a solution for a multi-robot system in an unknown environment for the formation task. We use the WS (SILVA-JR; NEDJAH, 2017) to establish communication between robots inserted into an environment without landmarks, without information about the robots' position, and with limited sensing. WS establishes a Father-Son relationship between robots for message propagation. During the formation task, this relationship produces a reference for the robots, helping overcome the lack of environmental knowledge. However, some combinations of the Father-Son relationship may produce different results during the formation task, impacting the simulation time, the average distance traveled by each robot, and the number of messages exchanged among robots. We adopt a model inspired by the Traveling Salesman Problem (TSP) to define a Father-Son relationship aiming to maximize the performance of the robot group during the formation task.

1.1 OBJECTIVE

General Objective

The general objective is to develop communication and control strategies for the formation navigation of multiple robots during the execution of collective tasks, both in a completely unknown environment, with obstacles and local communication.

Specific objectives

Based on the general objective, we designed the following specific objectives to perform this work:

- investigate the main and current communication and formation control strategies, involving multiple robots during the execution of tasks, such as formation and flocking;
- develop a local communication strategy to reduce possible communication failures, which occur more frequently during the execution of maneuvers to avoid obstacles;
- simulate, analyze and validate a local communication strategy, using the CoppeliaSim robot simulator (formerly V-Rep);
- develop a strategy for formation control applied in an unknown environment, limited sensing and local communication;
- simulate, analyze and validate the strategy for formation control, using the CoppeliaSim robot simulator (formerly V-Rep);

1.2 THESIS CONTRIBUTIONS

The main contributions of this thesis are summarized in the following topics:

- we use wave propagation for the navigation of multiple robots in formation as the Wave Swarm algorithm. Nevertheless, we developed a new model based on a cyclic graph to establish the communication links between all robots in the swarm, called as Double-Wave Swarm. This new model enhances the local communication between robots during navigation while influencing the robustness of the network, making it less susceptible to failures during obstacle deviation maneuvers as its connectivity increases.
- given a random distribution of robots into a swarm, we model the problem of assigning one position and route for each robot to achieve its place in the formation as a Traveling Salesman Problem. We solve the TSP by reinforcement learning. The routes resulting from the TSP solution establish the parental relationship between the robots for Wave Swarm communication, assuring an efficient execution of the formation task.

1.3 RELATED PAPERS

Based on the results achieved during the development of this thesis, the following articles were published (in chronological order):

- LAMPERTI, RUBISSON DUARTE; DE ARRUDA, LUCIA VALÉRIA RAMOS. Distributed strategy for communication between multiple robots during formation navigation task. **ROBOTICS AND AUTONOMOUS SYSTEMS**, v.169, p.104509, 2023.
- LAMPERTI, RUBISSON DUARTE; DE ARRUDA, LUCIA VALÉRIA RAMOS. A strategy based on Wave Swarm for the formation task inspired by the Traveling Salesman Problem. **ENGINEERING APPLICATIONS OF ARTIFICIAL INTELLIGENCE**, v.126, p.106884, 2023.

1.4 THESIS STRUCTURE

In Chapter 2, we explored the area of Swarm Robotics. We begin by discussing centralized and distributed systems, followed by the analysis of collective behaviors, such as spatial organization, navigation and decision making. We also address formation and flocking tasks, as well as aspects of communication, localization and the robotic platforms used.

Chapter 3 focuses on the concept of the Wave Swarm. We detail the messages exchanged between robots and explore the contributions based on this model.

In Chapter 4, we cover the practical application of the Wave Swarm to solve the Traveling Salesperson Problem (TSP). We present a solution for the TSP, as well as a developed strategy for the formation task, including navigation, desired shapes of robot groups, the role of the lead robot, and the phases of the strategy. We also discuss communication failures and detail the experiments and analyzes associated with TSP.

In Chapter 5, we present a distributed communication strategy between multiple robots during formation navigation using the concept of Double Wave Swarm. We detail the messages exchanged between robots, connectivity, alignment task, navigation task, local communication, rigid and semi-rigid formation, formation control and obstacle detection. We detail the simulated experiments and their analyses.

In Chapter 6, we present the conclusion of this thesis, highlighting the results of the experiments and future work in the area.

2 SWARM ROBOTICS

Swarm Robotics is a research area that studies systems composed of large autonomous robots with relatively simple hardware architecture.

These robotic swarms replace a single robot or when performing collective tasks, which require a group of robots, for example, a physical restriction (RUTISHAUSER *et al.*, 2009) or to guarantee effective execution (BAYINDIR, 2016; ŞAHIN, 2004).

Robot swarms are designed using a centralized (SHURIJI *et al.*, 2019; QIAN *et al.*, 2016; GHOMMAM *et al.*, 2010) or distributed (ISSA; RASHID, 2020; YAMAGISHI; SUZUKI, 2017; ALOUACHE; WU, 2017) system. In a centralized system, there is a central point that processes some type of information and makes other information used in executing the task available to the swarm robots. In a distributed system, there is no central point. The processing of any information is performed individually by the swarm of robots.

Various works in the literature delve into the main collective behaviors, including spatial organization, navigation, and collective decision-making, among others (WANG *et al.*, 2017; El Ferik *et al.*, 2016; XIDIAS; AZARIADIS, 2016; CHEN *et al.*, 2013). However, there is also great interest in communication (KHATERI *et al.*, 2020) and estimating the location of robots (MATSUKA *et al.*, 2021), since these characteristics can support in the execution of complex tasks (DRAGANJAC *et al.*, 2020; JOSHI *et al.*, 2019; SZWAYKOWSKA *et al.*, 2015).

In robotics, prior knowledge about the environment can facilitate the execution of tasks. Therefore, executing tasks in completely unknown environments (LIN *et al.*, 2019) increases the complexity of the algorithms/strategies used by the swarm. In completely unknown environments, robots can not receive information, such as the precision location of robots in the swarm, while executing the task.

Robotic swarm researchers use robotic platforms to analyze and validate their ideas. These robotic platforms can be composed of real or simulated robots.

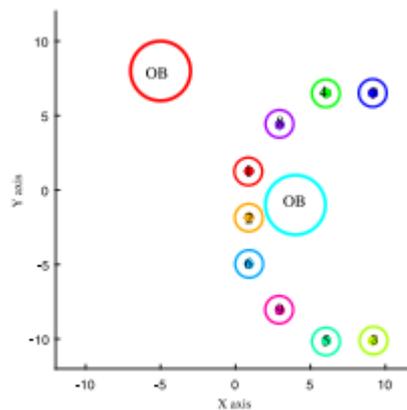
In this section, we will again address some, such as centralized and distributed systems, collective behaviors, communication, localization and the main robotic platforms. Specifically, we will detail the formation and flocking tasks.

2.1 CENTRALIZED AND DISTRIBUTED SYSTEM

A centralized system requires a central agent responsible for gathering information and guiding robots to perform the task. This type of system presents less complexity compared to the distributed system. However, the centralized system requires regular and uninterrupted communication between the central agent and the robots, making the system vulnerable to communication failures and malfunctions of the central agent. Furthermore, the system's response during task execution may be slow due to the large flow of information exchange. Obviously, swarms with a high number of robots require a greater number of message exchanges when compared to a smaller swarm.

Zhang *et al.* (2020) propose a new method for forming a group of robots in an optimally collision-free manner in the presence of static obstacles (Figure 1). From a centralized system, collision-free trajectories are calculated using the convex quadratic programming to minimize the distance and to get the optimal forming parameters under certain constraints.

Figura 1 – Simulation of robots in the shape of the letter C



Fonte: Zhang *et al.* (2020)

The distributed system uses collective processing in which each robot in the swarm reads its sensors, exchanges information with other robots, and then executes individual actions. The collective behavior of the swarm emerges from the execution of all individual actions of robots. However, the distributed robotic systems design is complex when compared to centralized ones. One advantage of the distributed system is providing complete autonomy in the decision-making process to the swarm robots. The system's response to environmental changes is faster and less vulnerable to failure since processing is individual and parallel, making the system flexible and

scalable due to the distributed characteristic.

The research by Vásárhelyi *et al.* (2014) presents a group of multi-copters using a decentralized system. The flight in an external environment is stable with up to 10 individuals (Figure 2), who rely on local communication to exchange information. Multi-copters have GPS receivers for location. Collective behavior is inspired by the physical statistical modeling of swarms of animals, with the aim of achieving stable flight in formation.

Figura 2 – Experiment with 10 robots



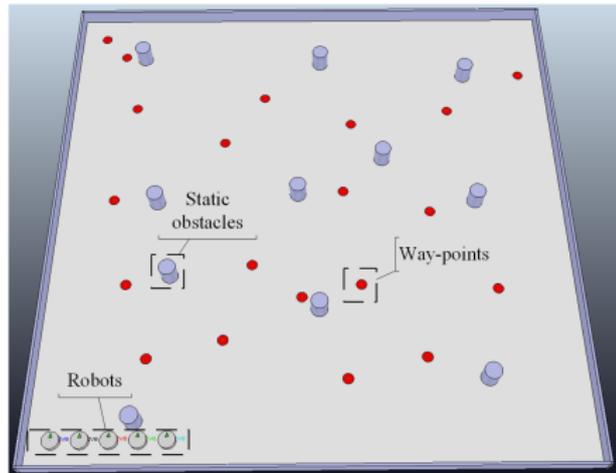
Fonte: Vásárhelyi *et al.* (2014)

In the work of Almeida *et al.* (2019), the authors present a cooperative and distributed navigation strategy. The team's robots are homogeneous, independent, and have limited communication skills. Local communication is used to share information between neighboring robots. An artificial pheromone is used to mark the explored regions to avoid redundancy. The main idea is to use an online trajectory planner for a multi-robot system designed to navigate and explore an unknown environment with the objective of finding and reaching way-points to the known final position (Figure 3). The cooperative and distributed navigation strategy through a proposed online trajectory planner is evaluated in different simulated environment scenarios.

2.2 COLLECTIVE BEHAVIOR

The collective behaviors works include the most frequent tasks and problems involving swarms of robots. Brambilla *et al.* (2013) proposes two taxonomies: methods and collective behaviors. According to Brambilla *et al.* (2013), collective behaviors classify into four main categories: spatially organizing behaviors, navigation behaviors, collective decision-making, and

Figura 3 – Scenario of the experiment proposed in the V-REP simulator



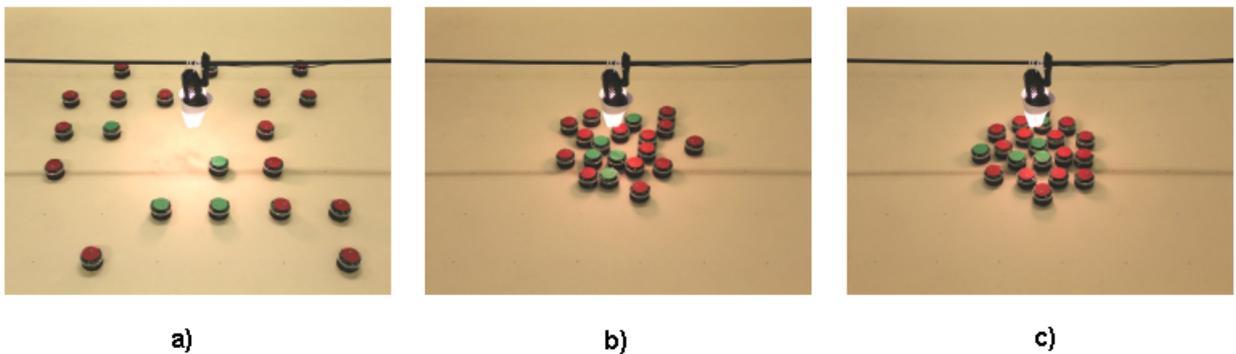
Fonte: Almeida *et al.* (2019)

other collective behaviors.

2.2.1 Space Organization

The spatial organization of a robot swarm is identified by the movement of robots for a particular purpose. Among the purposes or tasks for the swarm spatial organization found in the literature are the aggregation (or dispersion) (CHEN *et al.*, 2012) (Figure 4), the formation (FREUDENTHALER; MEURER, 2020) (Figure 5) and self-assembly (LI *et al.*, 2015) (Figure 6).

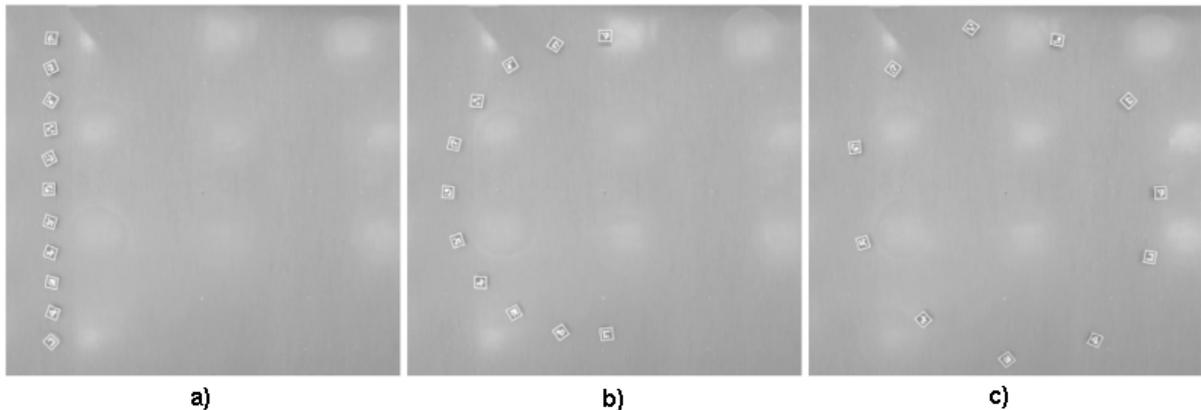
Figura 4 – Aggregation task steps: a) robots in a random position; b) robots in an intermediate position; c) robots in a final position



Fonte: Chen *et al.* (2012)

Aggregation is researched by authors Yang *et al.* (2019). They propose a collaborative

Figura 5 – Performing the formation task - images the transition (a - b- c) between the different formations using the ceiling-mounted camera system



Fonte: Freudenthaler e Meurer (2020)

Figura 6 – Self-assembly task: nine robots generating a structure with 4 legs in locomotion



Fonte: Li *et al.* (2015)

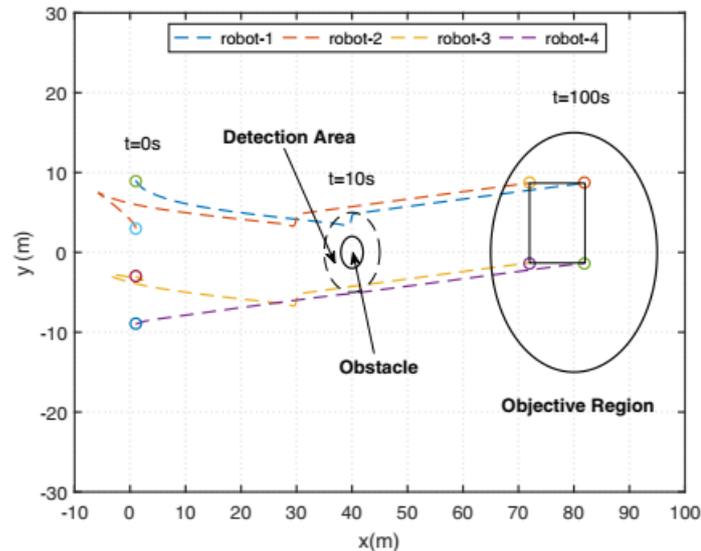
method based on by the Particle Swarm technique (also known as PSO) for robotic swarms subject to the practical constraints of relative positioning, local sensing, and communication with the limitations of the field of view. They associate the number of neighboring robots with the aggregation level. The proposed method updates the forward speed and angular velocity of the robot using the nonholonomic¹ model to realize the motion control of each robot.

Yu *et al.* (2019) use a model of a multi-robot system in the presence of uncertainties and external disturbances to explore a new form of robust control during the formation task. A novel neural network-based robust control scheme combined with an adaptive compensator and

¹ In mobile robotics, the term refers specifically to the kinematic constraints of the robot chassis. A holonomic robot is a robot that has zero nonholonomic kinematic constraints. Conversely, a nonholonomic robot is a robot with one or more nonholonomic kinematic constraints. Siegwart *et al.* (2011)

adaptive gain control is proposed to achieve formation control within the region with possible collision and obstacles. They show through numerical simulations that the proposed control method maintained the group formation as desired (Figure 7).

Figure 7 – The trajectory of four robots with obstacle avoidance



Fonte: Yu *et al.* (2019)

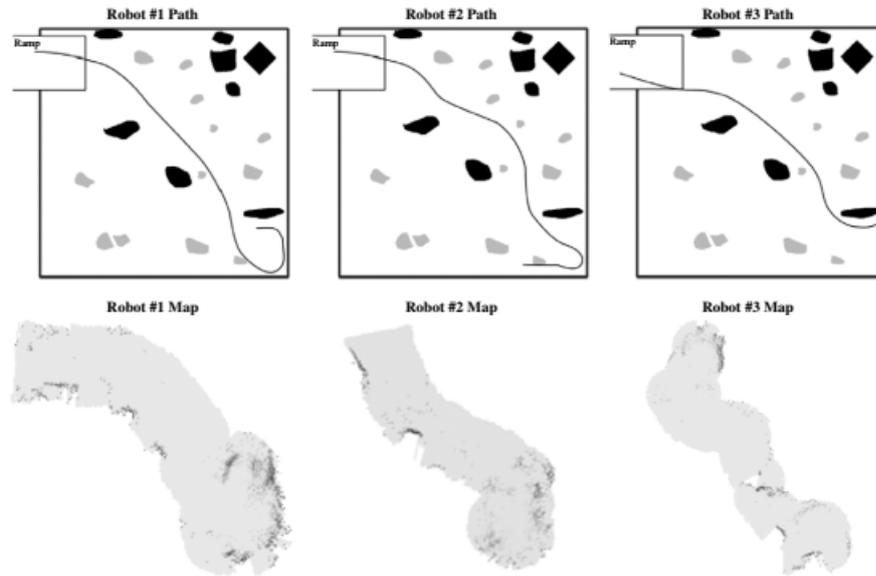
2.2.2 Navigation

There are different ways and objectives to navigate a swarm of robots, for example, exploration/mapping or the collective transport of an object. The robots swarm navigation covers topics such as exploration (FAIGL; KULICH, 2015) (Figure 8), foraging (CAMPO *et al.*, 2010) (Figure 9), flocking (OLCAY *et al.*, 2020) (Figure 10), and transport task (FENG *et al.*, 2020) (Figure 11).

The foraging problem is addressed by Song *et al.* (2020) in two scenarios: with one food source and two sources, Figure 12. In this work, the authors present a new pheromone model of swarm foraging behavior based on a neural network. The neural network is updated through a new evaporation model. Furthermore, the researchers present an optimization method to determine key parameters of cooperative foraging based on mathematical modeling. The experimental results present the efficiency of the proposed pheromone model.

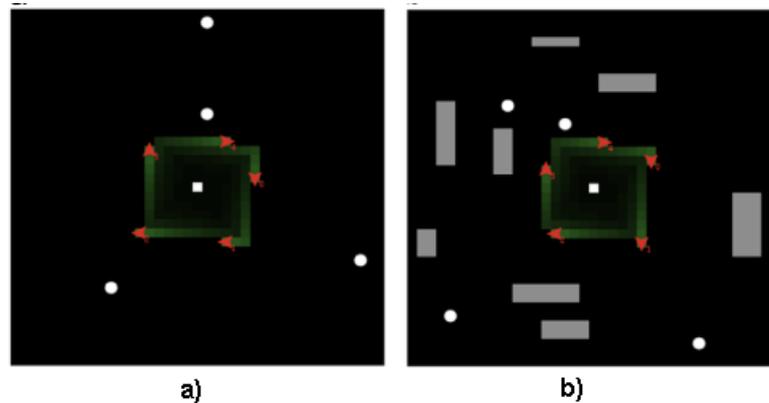
In the work of Tran *et al.* (2020), the authors investigate a new method for shape-shifting a group of heterogeneous autonomous vehicles, which include unmanned ground and aerial

Figura 8 – Exploration task: trajectory and maps that each robot built during online exploration. The black spots correspond to observable obstacles, and the gray spots correspond to unobservable obstacles



Fonte: Gifford *et al.* (2010)

Figura 9 – Foraging task: (a) Simulation environment without obstacles; (b) Simulation environment with obstacle - the arrows are the agents, the white cell in the center is the nest, the white circles are the food and the gray rectangular shapes are the obstacles



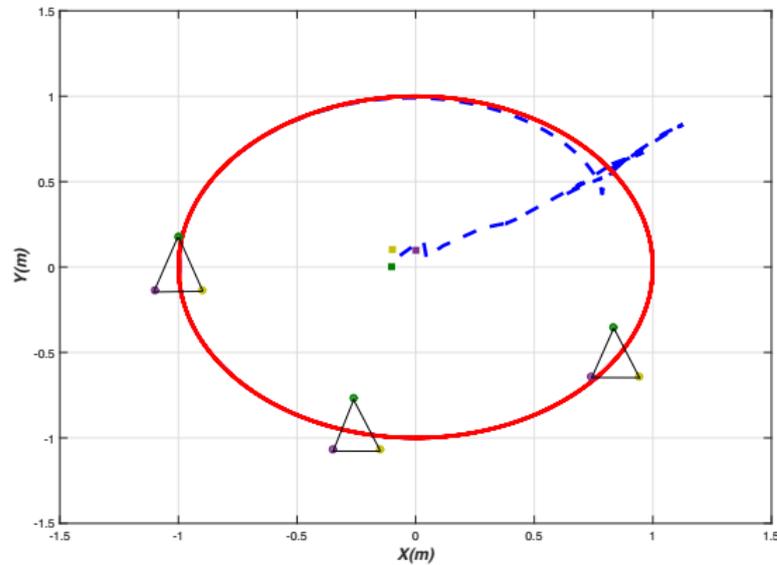
a)

b)

Fonte: Zedadra *et al.* (2016)

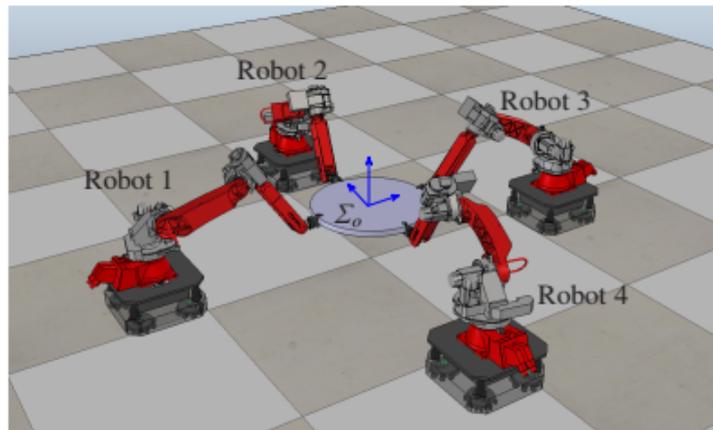
vehicles, to perform the Flocking task (Figure 13) . They extend the Negative Imaginary (NI) consensus control approach to change the formation of robots using only the relative distances between agents and combining the distances between agents and obstacles. All agents are free to seek a new safe formation to avoid obstacles. However, they must restore the initial formation as soon as the obstacles are overcome. The authors validate the proposed method by comparing it with existing methods in the literature, such as the conventional Artificial Potential Field method, using experiments with real robots.

Figura 10 – Flocking task: movement of three robots in a circle - the red and blue lines represent the trajectory of the virtual leader and the centroid



Fonte: Chu *et al.* (2018b)

Figura 11 – Transport task: illustration of the initial condition of the simulation of an object moved by a group of four robots

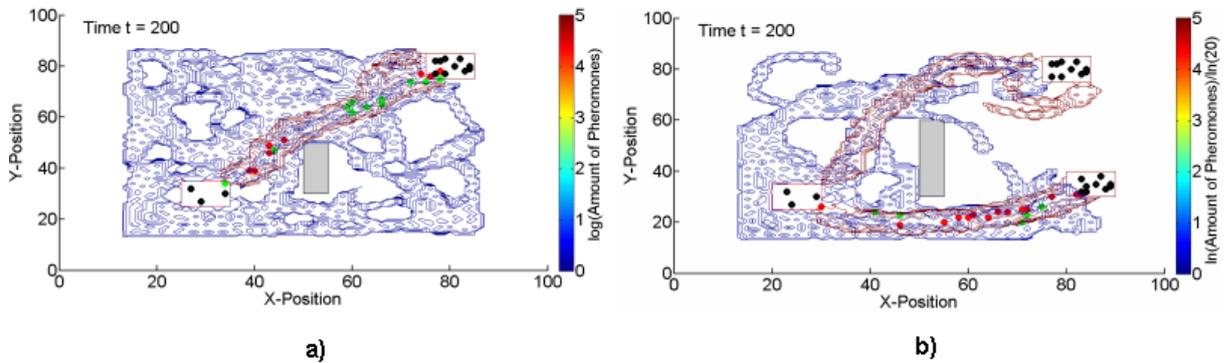


Fonte: Marino e Pierri (2018)

2.3 FORMATION TASK

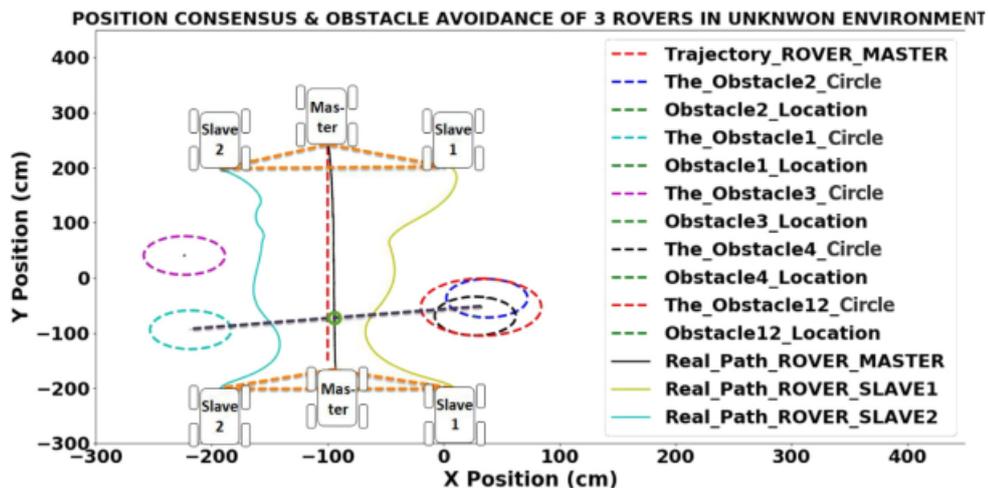
The formation task belongs to the subclass of spatial organization behaviors proposed by Brambilla *et al.* (2013). Formation is used when a swarm of robots needs to assume a specific shape from random positions of the robots and/or to maintain a formation (YAMAGISHI; SUZUKI, 2017; XU *et al.*, 2014). The swarms in some works take on basic geometric shapes (SILVA-JR; NEDJAH, 2017; SZWAYKOWSKA *et al.*, 2016; GARCIA-B *et al.*, 2013), but in others they take on more complex shapes (ZHANG *et al.*, 2018; RUBENSTEIN *et al.*, 2014;

Figure 12 – Distribution of the pheromone in a trajectory with an obstacle: a) one food source, and b) with two food sources



Fonte: Song *et al.* (2020)

Figure 13 – Results of experiments for a variable triangular formation with obstacle avoidance, using the proposed method



Fonte: Tran *et al.* (2020)

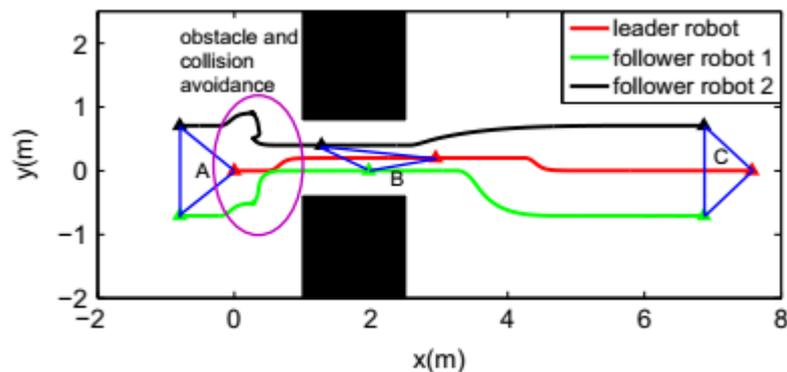
JEON; LEE, 2014; MENG *et al.*, 2013; HOU; CHEAH, 2012; CHEAH *et al.*, 2009). Several types of robot swarms have been used in works: aerial (SHURIJI *et al.*, 2019; CHUNG *et al.*, 2018; VÁSÁRHELYI *et al.*, 2014; LEE *et al.*, 2013; TAHK *et al.*, 2005), terrestrial (ZHANG; PHAM, 2019; SIEMONSMA *et al.*, 2018; DIMAROGONAS *et al.*, 2006; FREDSLUND; MATARIC, 2002) and aquatic (EDWARDS *et al.*, 2004).

The formation of a swarm, in most works in the literature, is achieved through the follower-leader robot (MISWANTO *et al.*, 2012; CONSOLINI *et al.*, 2008), in which the leader can be a robot from the swarm and/or a virtual leader robot (ASKARI *et al.*, 2013; DAI; LEE, 2012; LEWIS; TAN, 1997).

During the formation task, communication failures may arise between the robots (LIU; HU, 2014) and obstacles may appear (TEIXEIRA *et al.*, 2020; YANG *et al.*, 2010; BELTA; KUMAR, 2004). Despite the problems that may arise, the swarm must be able to execute the formation in the best possible way.

The authors Dai *et al.* (2015) use a strategy that adopts a leader robot to plan a safe trajectory using the geometric obstacle avoidance method. This strategy switches training to avoid and overcome obstacles safely (Figure 14). Upon detecting obstacles, the leader calculates a new distance and angle value between the leader and follower robots.

Figura 14 – Experiment results for a triangular formation with obstacle avoidance



Fonte: Dai *et al.* (2015)

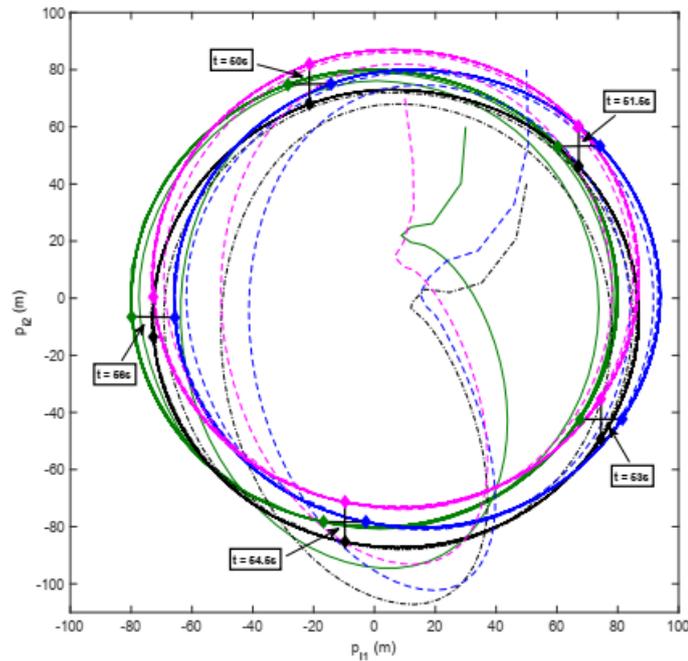
In the work of Yuan *et al.* (2019), the virtual leader is used for the formation problem for a group of mechanical systems with nonlinear dynamic uncertainty, Figure 15. The virtual leader is a linear system subject to unknown limited inputs, able to compute reference signals for the formation task.

Formation algorithms are used for swarm inserted in both previously known or completely unknown environment. The previously known environment, whether through offline training (LIN *et al.*, 2013) or the existence of reference points (HAN *et al.*, 2017), offers a means for calculating the estimated location of the robots. In completely unknown environments (LIN *et al.*, 2019), the robots only have information get during the task execution, making hard task accomplishing.

Regardless of the environment in which the swarm is inserted, the formation techniques frequently apply the module and angle measurements of the distance vector, and the orientation between neighboring robots (MARTINEZ-CLARK *et al.*, 2018; SOLEA *et al.*, 2010).

Research work on the formation task often focuses on the development of control

Figura 15 – Multi-robots formation control. robot 1: (green); robot 2: (black); robot 3: (blue); robot 4: (magenta)



Fonte: Yuan *et al.* (2019)

laws (CHEN *et al.*, 2019; CHU *et al.*, 2018a; QIAN *et al.*, 2017; QIAN *et al.*, 2016; HOU; CHEAH, 2012), the application of techniques - Potential Fields (ELKILANY *et al.*, 2020), Fuzzy (TEIXEIRA *et al.*, 2020; CHEN; CHANG, 2018; TUTUKO *et al.*, 2018), Neural Network, Genetic Algorithm (LÓPEZ-GONZÁLEZ *et al.*, 2020; LIN *et al.*, 2013; JIN *et al.*, 2009), PSO (Particle Swarm Optimization) (NURMAINI *et al.*, 2014), Reinforcement Learning (PRASAD *et al.*, 2017), GRN (Gene Regulatory Network) (JIN *et al.*, 2009) and H-GRN (Hierarchical Gene Regulatory Network) (JIN *et al.*, 2012), in the design of innovative algorithms (ISSA; RASHID, 2020).

2.4 FLOCKING TASK

Flocking is a behavior observed in nature in many bird species, which form large groups of individuals moving together toward a common target location (BAYINDIR, 2016).

Flocking task corresponds to multi-robots that move around an environment with the aiming of reaching a specific point. The navigation environment can be partially known (LEMAIGNAN *et al.*, 2017), or unknown (WANG *et al.*, 2016).

Navigation of multiple robots can be accomplished in several ways. Among these,

navigation in groups or flocks in formation has aroused interest among researchers in recent years (OLCAY *et al.*, 2020; DAI *et al.*, 2015; NASCIMENTO *et al.*, 2013). Specifically, in swarms of robots, flock navigation is known as Flocking (OLFATI-SABER, 2006; TANNER *et al.*, 2003b; TANNER *et al.*, 2003a). In collective terms, robots can navigate using different strategies, such as in the presence of a leader (LEE; CHWA, 2017; SILVA-JR; NEDJAH, 2017) as part of the group of robots, with only one virtual leader (YUAN *et al.*, 2019), and, in the absence of a leader (MENDONCA *et al.*, 2012).

Known environments have been frequently previously already visited by the robots or have mapping for navigation. However, even in a known environment, some changes may occur, such as the unexpected appearance of obstacles, providing the group of robots with an unknown situation - making the environment partially known.

The unknown environment requires complex strategies to deal with different unexpected situations, making navigation difficult for the group of robots. In most applications, robots must navigate with limited communication and location estimates that are sometimes uncertain and imprecise because of incomplete information.

There is the possibility of navigate with the help of sensors embedded into robots. Among the commonly used sensors are inertial sensors (ZHAO *et al.*, 2021; CHEN *et al.*, 2012). However, inertial sensors can be noise sources, making it difficult to estimate the values of the variables involved (CARVALHO, 2011).

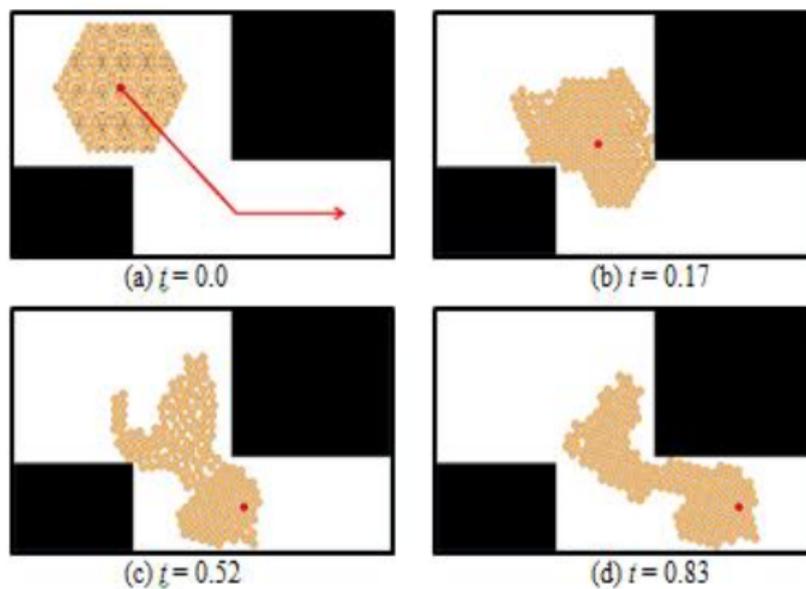
Robots can estimate their positions using radio navigation (SERGIYENKO *et al.*, 2019). The radio signal can be used in communication between robots, in identifying objects and robots and in localization techniques, such as Triangulation (SERGIYENKO *et al.*, 2016).

The maneuvers are performed autonomously and individually by each group member, but always with the same goal. The absence of obstacles (EDWARDS *et al.*, 2004) makes navigation easier. However, the appearance of obstacles (KOBAYASHI *et al.*, 2018; NASCIMENTO *et al.*, 2014) may require an abrupt change in the maneuvers performed, compromising the collective. Therefore, these obstacles must be avoided, maintaining the swarm formation with as little change as possible. Some techniques, such as Potential Fields (ELKILANY *et al.*, 2020), Fuzzy (TUTUKO *et al.*, 2018), Bresenham Algorithm (ALI *et al.*, 2016), Pattern Generation Strategy (ZHANG *et al.*, 2018), are developed and have been tested by several researchers to obstacle avoidance.

In the work of Yamagishi e Suzuki (2017), a distributed collective motion control

method is proposed for a swarm of robots based on a thermodynamic internal energy model. The robots move through a changeable aggregation formation, helping to avoid obstacles in the simulated environment (Figure 16). Navigation in formation is maintained throughout the trajectory because of the virtual thermodynamic model used. The swarm of robots has a leader who directs the group during navigation. Simulations are carried out to confirm the ability of the proposed method to provide robustness and flexibility to a swarm of robots during the navigation task.

Figura 16 – Different instants for the collective movement of the swarm: the red circle and yellow circles are the leader and followers, and the black rectangles are the obstacles. The leader moves along the red arrow while following the followers



Fonte: Yamagishi e Suzuki (2017)

Mendonça *et al.* (2017) propose a cooperative architecture for navigating a swarm of robots based on Dynamic Fuzzy Cognitive Maps (DFCM). The architecture is formed by a distributed system, in which the swarm robots are homogeneous autonomous and have abilities for learning, self-adaptation, behavior management and cooperative data sharing. The proposed Fuzzy model has parameters that need to be adjusted. These parameters are self-adjusting through computational intelligence, using the Reinforcement Learning technique. In this work, two strategies are analyzed: navigational memory sharing, and a strategy inspired by the behavior of ants.

Teixeira *et al.* (2020) present a Fuzzy approach for forming a group of four drones called Quadral-Fuzzy. The approach is composed of four central systems: leader drone control,

work drone control, formation maintenance, and self-preservation. The proposed approach uses a leader robot to map the environment using a 3D sensor. The leader has position and obstacle avoidance controls, making it possible to change direction in the event of an obstacle. In the presence of a detected obstacle, the formation is rotated, causing all drones to move away from it. The leader communicates with the working drones and passes information about the target point, and then the drones move to this position - formation maintenance system. The self-preservation system aims to avoid a collision between drones. Diversion maneuvers are based on the inaccuracy level of a drone moving in a particular direction. In the event of a probable collision, a repulsive force proportional to its inaccuracy level is applied, moving the two drones apart. Simulations show that the proposed approach provides safe navigation for multiple drones, avoiding collisions between drones and obstacles.

2.5 COMMUNICATION

2.5.1 Global communication

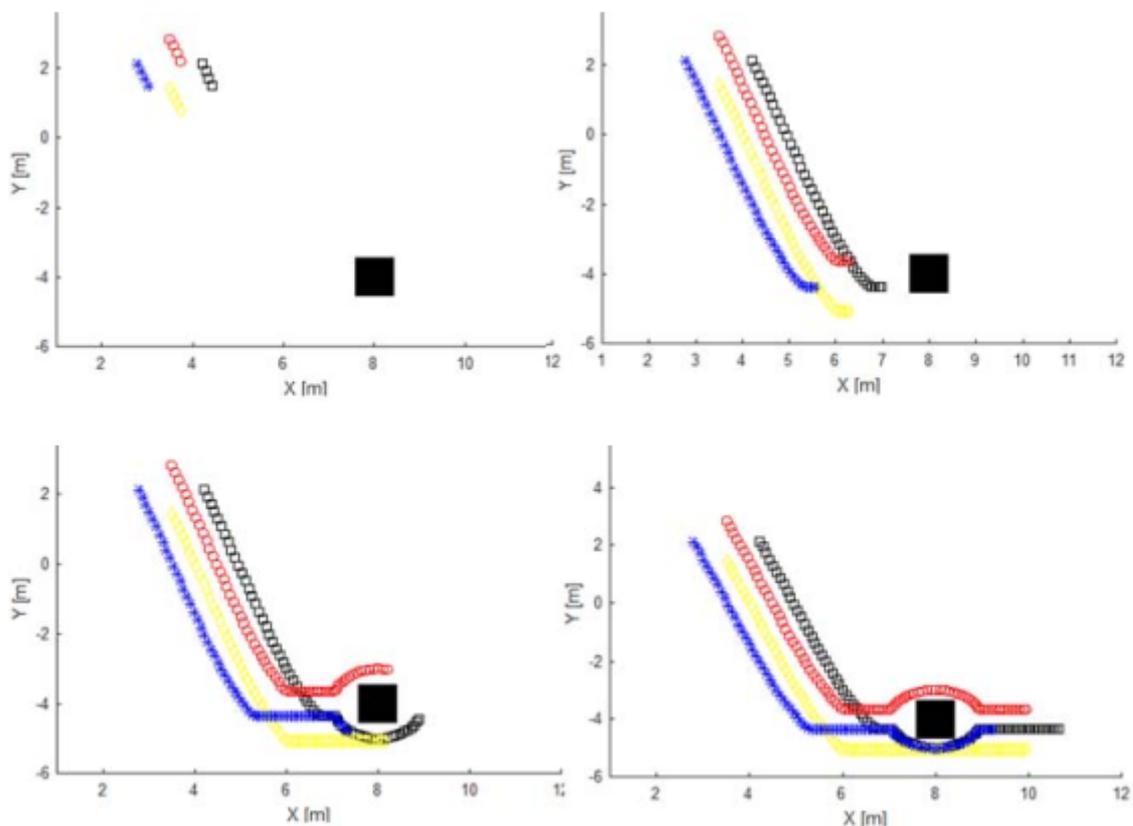
Global communication is an unrestricted way in which all robots in a swarm communicate with all other robots. In this type of communication, robots use wireless communication, such as Wi-Fi (BRAEM *et al.*, 2006) and Bluetooth. Global communication is often used for identification, position, communication with a central station (ALMEIDA *et al.*, 2020) and location.

An advantage of global communication is the range of the signal used for communication and not the distance between the robots. However, a disadvantage of this type of communication is the high number of data collisions. The collision occurs because of various messages exchanged by the same robot or central station, causing a delay in communication (BAI *et al.*, 2016; MIJALKOV *et al.*, 2016; LINDLEY *et al.*, 2012).

In the work of Szwaykowska *et al.* (2016), the authors use a swarm of robots through mixed reality. They investigate swarm behavior based on network connectivity, heterogeneity in agent dynamics, and acceleration capabilities. Authors show the existence of stable coherent patterns that can only be achieved with delay coupling and that are robust to decreases in network connectivity and the heterogeneity of agent dynamics. The results are validated through simulations and experimental results of delay-induced pattern formation in a mixed reality swarm (experiments with 46 virtual robots and 4 physical robots).

Yamchi e Esfanjani (2017) use a delayed data communication network between robots in the group to evaluate a new distributed predictive controller. The proposed controller guarantees stability by maintaining formation between mobile robots during their movement along a trajectory with obstacles, Figure 17. The system dynamics are modeled with adjustable parameters and delay. The adjustable parameters are determined synchronously in each agent by applying the predictive strategy. The efficiency of the proposed approach is proven through numerical simulations.

Figura 17 – Navigation in formation with obstacle avoidance



Fonte: Yamchi e Esfanjani (2017)

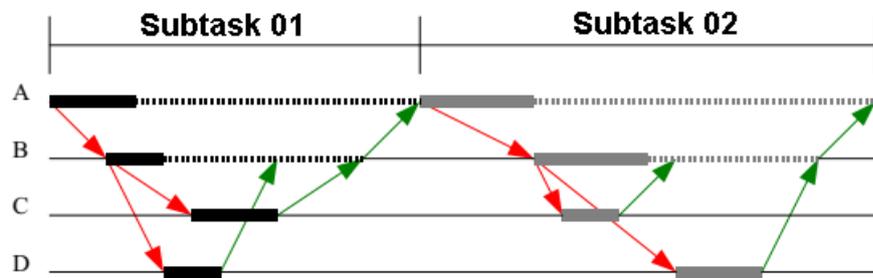
2.5.2 Local Communication

Local communication limits the exchange of information between robots close to each other. This is called neighborhood (HASAN *et al.*, 2016b). The local communication uses the neighborhood to exchange information, limiting communication between neighboring robots.

In local communication, robots use wireless communication, such as Wi-Fi (LATRÉ *et al.*, 2011), Bluetooth and Infrared Signal (IR) (SILVA-JR; NEDJAH, 2017; MENDONCA *et al.*, 2012).

Silva-Jr e Nedjah (2017) propose a new wave algorithm applied to collective navigation. This algorithm, called Wave Swarm, is used in local communication through the message propagation mechanism. Wave Swarm is presented as a general strategy for managing the sequence of subtasks (Figure 18) that make up collective navigation. Message propagation is used to delimit the beginning and end of each subtask among the robots in the swarm.

Figura 18 – Precedence diagram for two subtasks



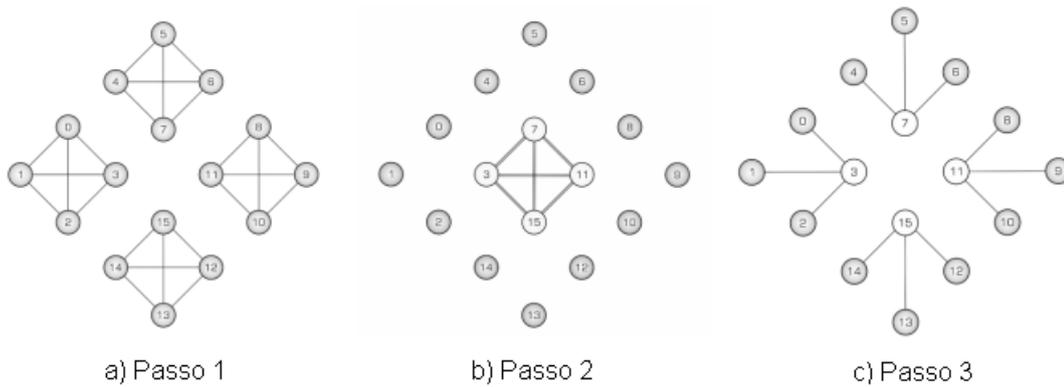
Fonte: Silva-Jr e Nedjah (2017)

In the work of Nedjah *et al.* (2020), the authors propose an algorithm for allocating tasks to a swarm of robots in a distributed system. The algorithm, called Clustered Dynamic Task Allocation (CDTA), is inspired by the particle swarm optimization strategy. Therefore, executing the algorithm requires an intense exchange of information between the robots (Figure 19), which can make it difficult to allocate tasks to large swarms. The authors use of a cluster topology between robots to the communication process. The communication is optimized, facilitating the efficient allocation of tasks for large swarms of robots.

Lack or failure of communication between robots can cause the swarm to collapse during collective navigation. Efficient formation control can ensure the maintenance of connectivity in the swarm (SZWAYKOWSKA *et al.*, 2015). High connectivity between robots is extremely important for local communication, therefore, maintaining the connectivity of a group or swarm of robots has aroused great interest.

An approach is proposed to preserve the connectivity during the encounter task by Luo *et al.* (2019). The main idea behind this approach is adaptation to dynamic changes in the network topology: problems with the communication link and situations of failures caused by

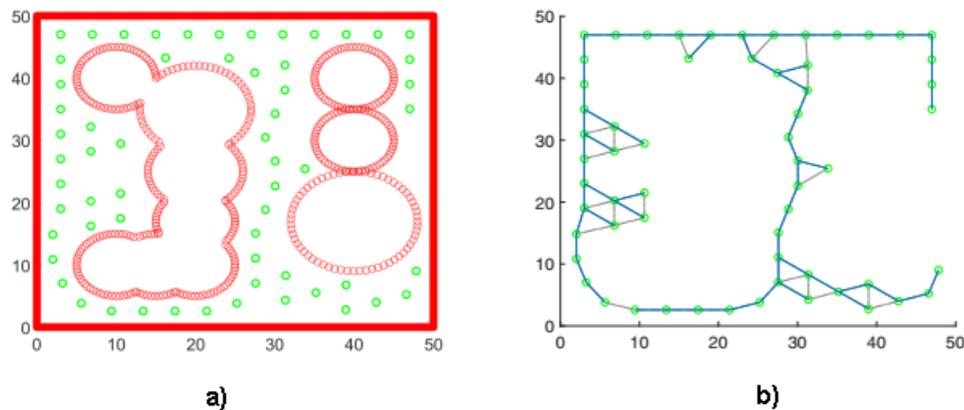
Figura 19 – The communication steps used in the cluster topology



Fonte: Nedjah *et al.* (2020)

robot mobility (Figure 20). Numerical simulations and real experiments are used to show the efficiency of the proposed approach.

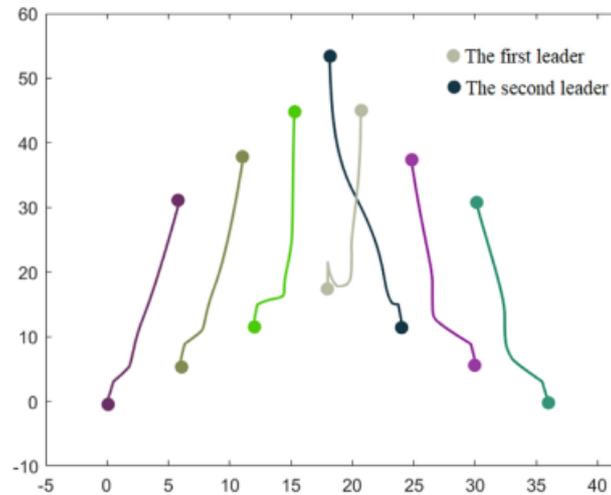
Figura 20 – Result of the numerical simulation: a) movement of the robots, and b) communication graphs between the robots



Fonte: Luo *et al.* (2019)

Khateri *et al.* (2020) suggest a new local connectivity maintenance method to gain more movement flexibility while preserving the properties and simplicity of a traditional local connectivity maintenance method. The proposed method is based on maintaining traditional local connectivity with a basic operation to exchange neighbors between two adjacent robots. This method makes it possible to exchange robots from the same group (Figure 21). Considering that the exchange of functions within a group is necessary, permutation without loss of connectivity is the advantage presented by the proposed method.

Figura 21 – The trajectory followed by the robots when changing the leader, following a V formation



Fonte: Khateri *et al.* (2020)

2.6 LOCALIZATION

Localization in robot swarms plays a key role in the ability of robots to coordinate and perform complex tasks, such as flocking and formation. In most swarms, the Global Positioning System (GPS) is not used, especially in indoor environments or in robots that can not integrate GPS receivers into their structure. The complexity of the swarm localization task is not directly related to the absence of GPS, but to the dynamics of the swarm itself and the coordination of the robots involved. Ensuring that each robot knows its relative location is crucial to successful task execution.

A localization system can rely on a specific, previously established coordinate system to reference all robots in the swarm. This coordinate system called the global coordinate system, allows you to reference robots on a map and create landmarks, making it easier to locate robots in a swarm (SÁ *et al.*, 2017; TRON *et al.*, 2016). Another way of localization is through the relative coordinate system, in which robots have their own coordinates within a swarm topology (RASHID *et al.*, 2015). In this way, measurements such as distance are related to the coordinate of the robot that measured it, making it impossible to build a map of the environment. Despite this, a relative coordinate system makes it possible to locate and apply it to a swarm of robots (SASAOKA *et al.*, 2016).

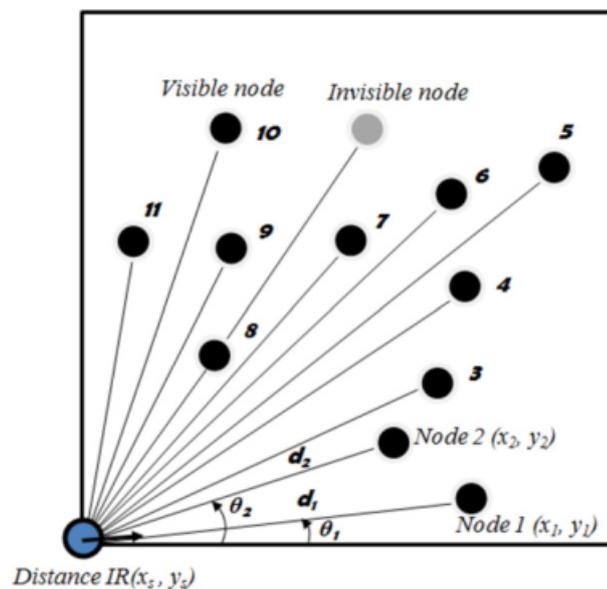
Localization problems go beyond the choice of a coordinate system. A strategy for processing data obtained by robots, whether by a global or relative system, must be designed

to transform the data into estimates of the positions of the robots in a swarm. The processing of this data is often carried out following a centralized (ISSA; RASHID, 2019; HASAN *et al.*, 2016a; ZHANG *et al.*, 2008; SHANG *et al.*, 2003) or distributed (SÁ *et al.*, 2017; HASAN *et al.*, 2016b; RUBENSTEIN, 2009) processing framework.

In the centralized processing, all swarm-relevant data is brought together at a single central station for processing. This central station has the function of calculating the position and returning the results to all robots in the swarm. However, in the event of a central station failure, all robots in the swarm cannot know their position.

In the work of Rashid *et al.* (2015), a new algorithm with centralized processing is presented for multi-robot localization and orientation. This algorithm considers which each robot can estimate its orientation in relation to neighboring robots that are within its vision (transmission) range (Figure 22). The environment has an infrared distance sensor that scans the robots and estimates the absolute value of their positions and orientations. The identification of each robot is computed by combining the orientation achieved by the IR distance sensor with the estimated orientation using the sensors embedded in each robot. The location and orientation of robots not visible to the IR distance sensor are achieved through the exchange of messages between the base (IR distance sensor) and the robots, reconstructing a complete map of the team distribution. The accuracy in estimating the location of these robots is improved by introducing a new algorithm that is based on the location of neighboring robots.

Figura 22 – Location estimation algorithm using a beacon - illustration of visible and non-visible robots

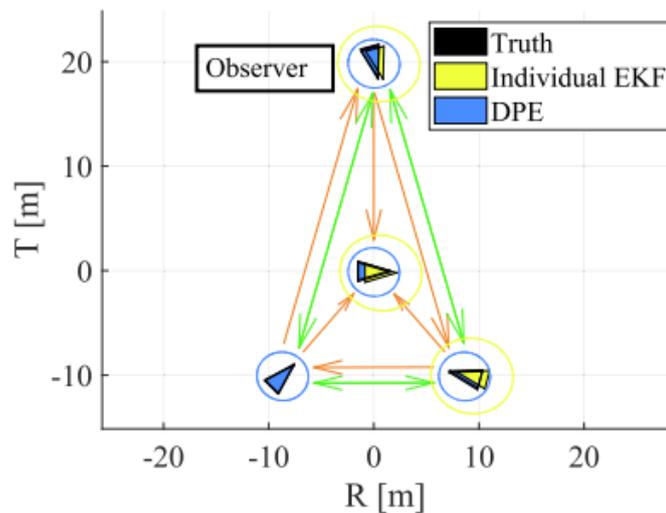


Fonte: Rashid *et al.* (2015)

In the distributed processing strategy, data processing is performed in a distributed manner by all robots in the swarm. Each robot must be able to estimate its position, using its data and that of its neighboring robots (KRISHNAN *et al.*, 2020). Swarms of robots with distributed processing are more robust when compared to centralized processing, and in the event of a failure (in one robot), in most applications, the other robots can estimate their positions. However, distributed processing increases the processing overhead on each robot. The efficiency of position estimation in a swarm is directly proportional to the connectivity between robots (SÁ *et al.*, 2016).

Matsuka *et al.* (2021) propose a new decentralized and scalable algorithm for localizing swarms of robots, called Decentralized Pose Estimation (DPE). DPE considers communication and relative detection to define an observable local formation (Figure 23). Each robot establishes a subset of neighboring robots through direct measurements and communication (ad hoc network). DPE provides a scalable, fully decentralized navigation solution that can be used for swarm control and movement planning. Numerical simulations and experiments using robotic spacecraft simulators are used to show the scalability and efficiency of the proposed algorithm.

Figura 23 – DPE and EKF present estimates individually and relative detection and communication graphs



Fonte: Matsuka *et al.* (2021)

2.7 ROBOTIC PLATFORMS

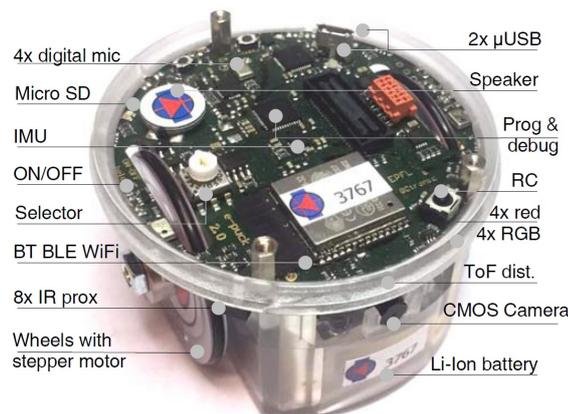
Software and hardware platforms for robotics enable the testing and validation of architectures and approaches using real robots or robot simulators. The cost of purchasing or

building real robots is high. Therefore, the use of robot simulators can partially make up for the lack of real robots. Below are some real robots and robot simulators frequently used in robotics research involving multiple robots.

2.7.1 Physical Robots

The e-puck2, Figure 24, is the latest mini mobile robot developed by GCtronic and EPFL (MONDADA, 2022). Launched in 2018, this is an evolution of the e-puck robot used in many research and education institutes. e-puck2 is backward compatible with its predecessor but is powered by a STM32F4 microcontroller and has various sensors such as proximity infrared, sound I/O, 9xIMU, ToF distance sensor, camera, uSD storage. The robot is a complete system with USB hub, debugger/programmer, Wi-Fi module.

Figura 24 – e-puck2 robot



Fonte: Mondada (2022)

The Kilobot, Figure 25, is a low-cost robotic system. It is designed to provide a basis test to advance understanding of collective behavior and realize its potential to provide solutions to a range of challenges (KILOBOT, 2010). According to the developer, the Kilobot has a diameter of 33 mm, fine motor control (255 different power levels), RGB LED, rechargeable battery and hundreds of robots can be programmed at the same time. Communication between neighboring robots can be performed up to 7 cm.

The Swarmanoid system is formed of three types of robots: foot-bots (13 cm in diameter and 28 cm in height) (Figure 26) specialized in moving on flat and irregular terrain, capable of self-assembly and transport objects or other robots; hand-bots (29 cm high and 41 cm wide) can climb some vertical surfaces and manipulating small objects; and eye-bots are autonomous flying robots that can attach to an internal ceiling, capable of analyzing the environment from a

Figura 25 – Kilobot robot



Fonte: Kilobot (2010)

privileged position to collect information inaccessible to foot-bots and hand-bots.

All Swarmanoids have a multiprocessor architecture, consisting of a main processor that handles CPU-intensive tasks such as vision and higher-level control, and multiple microcontrollers that handle real-time sensor reading and actuator control. The ARM 11 *i.MX31* main processor clocks in at 533 MHz and has 128 MB of RAM, 64 MB of Flash, a USB 2.0 host controller, and an add-on power and I/O chip. The microcontrollers are based on the DsPIC 33 as it offers good computing power, includes fixed-point and DSP instructions, and has low power consumption. Communication between robots allows a relative location from 10 cm to 5 m for foot-bots and handbots and up to 12 m for eye-bots. Signals from infrared sensors and radio signals are combined to minimize interference in communication (DORIGO *et al.*, 2013).

Figura 26 – Swarmanoid: Foot-Bot robot



Fonte: Dorigo *et al.* (2013)

2.7.2 Robot Simulators

CoopelliaSim or V-REP (Virtual Robot Experimentation Platform) (ROHMER *et al.*, 2013) is a robust, versatile, and scalable real robot simulator. A general-purpose robot simulator that provides various tools and functionality simultaneously while abstracting robotic systems and their complexities. It has distributed and script-driven control: each scene object can have an embedded script, all operating at the same time, in a chained or non-chained manner. Another important feature is the ability to self-duplicate or self-destruct the model, along with associated calculation objects and embedded scripts. Experiments performed in the simulator can be recorded by the V-REP's integrated video recorder. Communication is easily extensible within V-REP or with the outside world. V-REP has exhaustive documentation, facilitating understanding and increasing efficiency in simulating real robots.

The Player Project provides freely distributed software that enables the development of robot and sensor systems. The Player robot server is probably one of the most used robot control interfaces in the world. The Player provides a network interface for a variety of robots and hardware sensors. The Player's client/server model allows robot control programs to be written in any programming language and run on any computer with a network connection to the robot. The Player supports multiple simultaneous client connections to devices, creating new possibilities for distributed, collaborative discovery and control. Distributed under the General Public License (GNU), the project has a completely free interface for use, distribution and modification, called Player/Stage (STAGE, 2003).

Stage simulates a population of mobile robots moving and sensing in a two-dimensional environment. Various sensor models are provided, including sonar, laser sensors, camera, and odometry. Stage devices feature a standard Player interface, so little to no changes are required to move between simulation and hardware. Various controllers designed in the Stage can be used together with physics robots.

Gazebo (2003) is a multi-robot outdoor simulator. Like Stage, it can simulate a population of robots, sensors and objects, but it does so in a three-dimensional world. It generates realistic sensor feedback and physically plausible interactions between objects. Controllers written for the Stage can be used with the Gazebo without modification and vice versa.

Webots (2004) is an open-source, cross-platform application used for robot simulation. It provides a complete development environment for modeling, programming and simulating

robots. It is designed for professional use and is widely used in industry, education and research. Cyberbotics Ltd. has maintained Webots as its core product continuously since 1998.

ARGoS (2012) is a simulator that uses the plug-in concept for various situations, such as physics engines (TODOROV *et al.*, 2012), physical space, and media. Thus, the user can choose which physics engine to use for a simulation. Physical space can be divided into multiple regions, each controlled by a specific physics engine. As robots navigate the environment, they are transferred from one engine to another automatically. In ARGoS, there is the concept of medium. One medium is a plug-in that implements algorithms to simulate a robot's means of communication, such as Wi-Fi and Bluetooth.

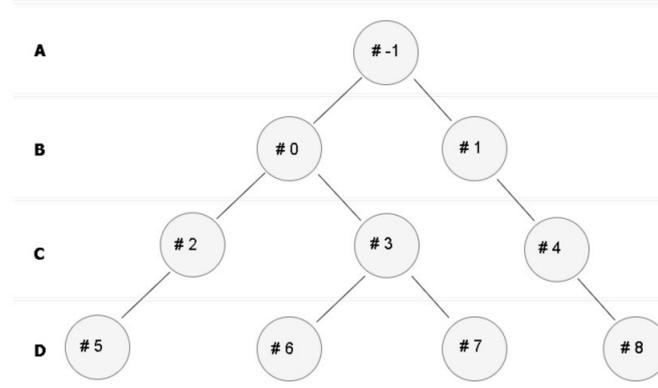
Simulation on multi-core processors is an option in ARGos. The main simulation loop is distributed across multiple threads. There are two types of threads: master and slave. The master thread assigns tasks to the slaves. A task consists of updating a single plug-in (sensor, actuator, entity component, motor, etc.). The user can define the number of threads as part of the experiment setup.

3 WAVE SWARM

Wave Swarm is a strategy for message propagation developed for robot swarm applications by Silva-Jr e Nedjah (2015). This strategy uses message propagation through a wave to manage tasks and share information between robots during swarm navigation (SILVA-JR; NEDJAH, 2017).

The Wave Swarm strategy adopts a parental relationship between Father robot and Son robots as displayed in Figure 27). This figure shows a tree-like graph of a robot swarm in a triangular formation with 10 robots. The #0 robot has two Son robots, #2 and #3, and a Father robot, # - 1. In this type of graph, there are three vertices groups (robots): the root (A level), the intermediate vertices (B and C level) and the leaves (D level).

Figura 27 – Wave Swarm - parental relationship modeled as a graph



Fonte: Own authorship

Robot swarm leader, # - 1, is the tree root and the origin of the wave propagation process. The Wave Swarm algorithm (Algorithm 1) describes the procedure for messages exchange between robots. This algorithm starts the wave propagation through the Origin robot (or Leader robot), which sends a message to its Son robots. The Son robots get the message and send it to their Son robots. The leaves (Son robots) get the message and perform an event, and after, send a feedback message to their Father robot. The Origin robot finishes the propagation process of sending and getting messages after it performs an event and receives all feedback messages from its Son robots.

Collective tasks are complex, and they are often divided into subtasks such as recruitment, alignment, and navigation to facilitate their implementation by a swarm. However, these

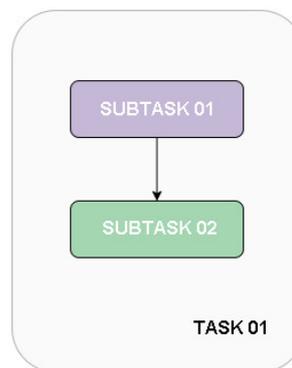
Algorithm 1 Wave Swarm Algorithm

```

requer Identification of the Father robot and the Son robots;
0: se ORIGIN-FATHER então
0:   Send the information to Son robots
0:   Perform the event
0:   Get the feedback message
0: senão,
0:   Get the message
0:   se The message contains an information então
0:     Send the information to Son robots
0:   senão, se Feedback message então
0:     Perform the event
0:     Send the feedback message
0:   finaliza se
0: finaliza se=0
  
```

subtasks need to run sequentially to guarantee the fulfillment of the main task. For example, subtask 1 and 2 compose the main task 1 (see Figure 28). By using Algorithm 1, subtask 1 must be carried out after subtask 2 (see Figure 29).

Figura 28 – subtasks set



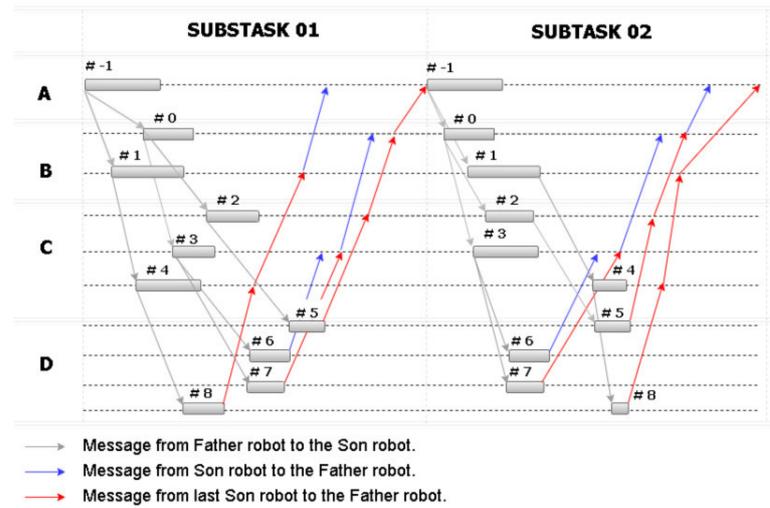
Fonte: Own authorship

The Origin robot, # – 1, sends a message to its Son robots, #0 and #1. These replicate the message to their Son robots. The feedback message starts with robot #8. However, the Origin robot only will start a new message sending after receiving all feedback messages from all of its Son robots.

In Figure 29, the gray arrows illustrate the sending and the red and blue arrows sign the receiving of messages by the Father robot. If a Father robot has only one robot Son, it will receive a single message (red arrow). In case of the Father robot has two or more Son robots, it will receive messages with blue arrows, except for the last message with a red arrow.

The run time of each subtask is different for each robot due to the distributed, asynchronous, and autonomous nature of the swarm. In the diagram, rectangles of different lengths

Figura 29 – Diagram of sending and receiving messages for two subtasks.



Fonte: Own authorship

illustrate the execution time of each subtask (see Figure 29).

3.1 MESSAGES EXCHANGED BETWEEN ROBOTS

The Wave Swarm algorithms use the exchange of messages between robots to synchronize subtasks and exchange information relevant to the execution of the formation task. Message and time complexity point out the resource consumption of these distributed algorithms. The message complexity is the total number of messages exchanged by the algorithm.

To measure the message complexity of the algorithms, we represent the parental relationship between the Father and Son robots through a graph $G=(V,E)$, where V and E are sets of N nodes and $(N - 1)$ edges. In this way, we can express the total number of messages exchanged between the robots using Wave Swarm is equal to:

$$M|_W = 2 * (N - 1) \quad (1)$$

where N is the total number of robots.

In distributed algorithms, the notion of time is not obvious, so we use some assumptions to characterize the time complexity (TEL, 2000).

- The time to process an event is zero time units.
- The transmission time (the time between sending and receiving a message) is one time unit.

In this way, the total transmission time for the Wave Swarm (Equation 16) algorithm is equal to:

$$T(N)|_W = 2N - 2 \quad (2)$$

where N is the total number of robots.

In the worst case, Wave Swarm has a message complexity and a time complexity equal to $O(N)$.

3.2 CONTRIBUTIONS BASED ON WAVE SWARM

The Wave Swarm (WS) uses message propagation through a wave to manage tasks and share information between robots during swarm navigation.

The WS has been used for alignment, recruitment, and formation tasks (SILVA-JR; NEDJAH, 2017; SILVA-JR.; NEDJAH, 2016; SILVA-JR; NEDJAH, 2015).

Silva-Jr. and Nedjah (2017) used a swarm (group) of robots with a known initial formation to navigate an environment with obstacles.

This thesis proposes the test of the WS in an environment with obstacles using a swarm (group) robots with an unknown initial formation (random initial position) for the formation task.

As shown in Section 4, establishing a relationship between Father and Son robot, also randomly, can increase the number of deviation maneuvers at the beginning of the task, impacting the time running, the average distance traveled by robots, and the number of message exchanges.

The Wave Swarm performance depends on an appropriate choice of Father-Son pairs when the robots' initial positions are unknown.

Such a choice establishes a combinatorial problem that can be modeled and solved as the Traveling Salesman Problem (TSP).

In Section 4, we show a distributed strategy for the communication between multiple robots, in which, first, we solve the TSP and then use the WS during the formation task in an environment with obstacles (random initial robots' position).

Silva-Jr.'s approach (SILVA-JR; NEDJAH, 2017) uses a tree-like graph to model the swarm configuration in which each robot corresponds to a graph node, and the graph edges establish the local communication links.

However, the formation navigation of multiple robots may present group shape variations, mainly in the presence of obstacles.

During obstacle avoidance maneuvers, some local communication links are frequently

broken, compromising the fixed topological communication structure and preventing the correct navigation task performance.

In this thesis, we also use wave propagation for the navigation of multiple robots in formation as the Wave Swarm algorithm (Section 5).

We developed a new model based on a cyclic graph to establish the communication links between all robots in the swarm.

This new model enhances the local communication between robots during navigation while influencing the robustness of the network, making it less susceptible to failures during obstacle deviation maneuvers as its connectivity increases. The Wave Swarm (WS) uses message propagation through a wave to manage tasks and share information between robots during swarm navigation.

4 A STRATEGY BASED ON WAVE SWARM FOR THE FORMATION TASK INSPIRED BY THE TRAVELING SALESMAN PROBLEM

Multi-robots can perform complex tasks such as exploration, foraging, and formation. Efficient communication between robots can contribute to the accomplishment of collective tasks through efficient message exchange.

This thesis proposes a strategy based on the message's propagation technique, Wave Swarm (SILVA-JR; NEDJAH, 2015; SILVA-JR; NEDJAH, 2017), for the formation task inspired by the Traveling Salesman Problem (TSP).

The Wave Swarm communication approach uses the concept of wave propagation for message exchange between neighbors, establishing a Father and Son relationship between robots. However, different pairs of Father and Son robots can impact the simulation time, the average distance traveled by each robot, and the number of messages exchanged during the formation task. Thus, given a random distribution of robots into a swarm, we model the problem of assigning one position and route for each robot to achieve its place in the formation as a Traveling Salesman Problem.

The routes resulting from the TSP solution establish a new parental relationship between the robots in the swarm. We performed preliminary experiments to define the technique used in the TSP resolution. We tested reinforcement learning and the genetic algorithm using the TSPLIB95 library. Thus, we develop a strategy for formation tasks based on Wave Swarm and TSP solved with reinforcement learning. We adopted the leader-follower approach in an unknown environment to validate the proposal. The results show the behavior of different sizes of robot groups for various desired shapes. Experiments with the robot simulator CoppeliaSim (V-REP) validate the proposed strategy and highlight its efficiency and robustness while running the formation task.

4.1 TRAVELING SALESMAN PROBLEM

The Traveling Salesman Problem (TSP) is a discrete combinatorial optimization problem studied since 1930. It is one of the most notorious problems in computer science and operational research.

The TSP is described as a completely undirected graph $G = (N, E)$ with a set of nodes $N = \{1, 2, \dots, n\}$ representing cities. Each edge $\{i, j\} \in E$ has a non-negative cost d_{ij} .

The Euclidean distance for calculating the distance between any two cities C_1 and C_2 is:

$$d = \sqrt{(c_{1x} - c_{2x})^2 + (c_{1y} - c_{2y})^2} \quad (3)$$

where (c_{1x}, c_{1y}) and (c_{2x}, c_{2y}) are the coordinates of the cities c_1 and c_2 .

The TSP is called symmetric when the cost of traveling between any two cities is the same regardless of the direction adopted, $d_{ij} = d_{ji}$.

The mathematical model of the symmetric TSP is (ZHANG; HAN, 2022):

Minimize:

$$Z = \sum_{i=1}^n \sum_{j=1}^n d_{ij} x_{ij} \quad (4)$$

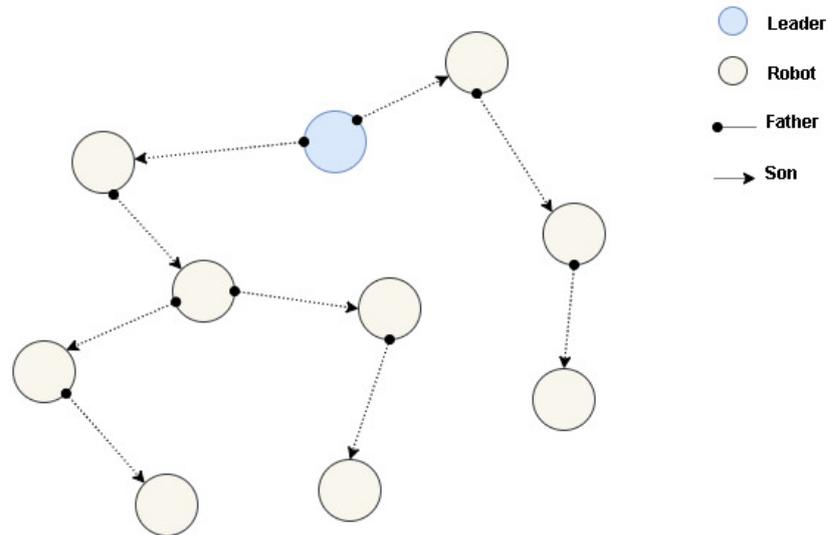
Subject to

$$\begin{aligned} \sum_{j=1}^n x_{ij} &= 1, i \in \{1, 2, 3, \dots, n\} \\ \sum_{i=1}^n x_{ij} &= 1, j \in \{1, 2, 3, \dots, n\} \\ \sum_{i, j \in S} x_{ij} &\leq |S| - 1, 2 \leq |S| \leq n - 2, S \subset \{1, 2, 3, \dots, n\} \\ x_{ij} &\in \{0, 1\}, \forall i, j \in \{1, 2, 3, \dots, n\} \& i \neq j \end{aligned}$$

The equation 4 is the objective function of the optimization problem, which minimizes the total distance traveled by the salesman. The value of $x_{ij} = 1$ represents a path taken by the salesman between cities i to j , otherwise $x_{ij} = 0$, i.e., the path has not occurred. The salesman can visit each city only once.

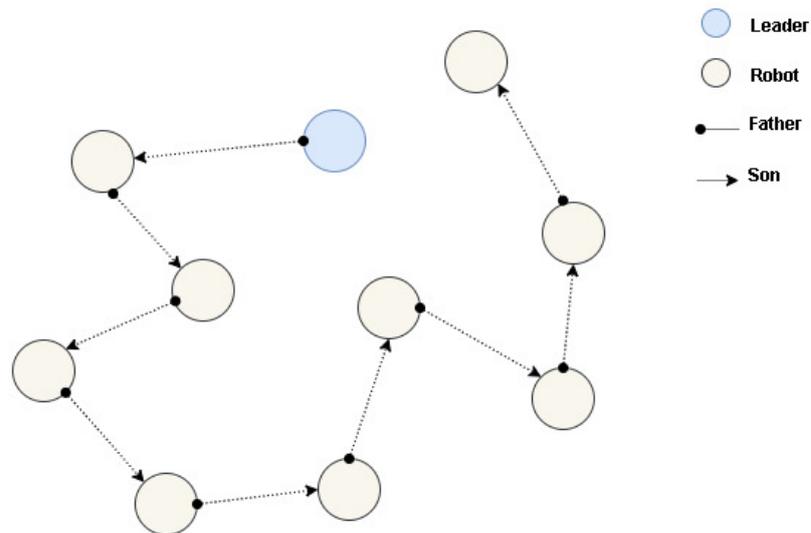
The proposed strategy uses the Wave Swarm algorithm to recruit neighborhood robots, generating an initial set of Father-Son relationships as illustrated in Figure 30. The TSP solution establishes a new parental relationship in a second moment, as illustrated in Figure 31. This new parental relationship restricts to one robot the maximum number of Son robots for each Father robot. It is worth noting that this choice is independent of the desired shapes for the execution formation task.

Figura 30 – Exchange messages by Wave Swarm: parental relationship initial



Fonte: Own authorship

Figura 31 – Exchange messages by Wave Swarm: a new parental relationship achieved through the TSP solution



Fonte: Own authorship

4.1.1 TSP solution

We can search for the optimal TSP solution by testing all solutions through an exhaustive search. However, the exhaustive search algorithm has a high computational cost with time complexity equal to $O(n!)$. The literature reports many other more efficient methods to solve the TSP. These methods are classified into exact, heuristic, and meta-heuristic methods.

Exact methods supply the optimal solution for the TSP. However, these methods can offer a high computational cost, as seen in Table 1.

Tabela 1 – Time complexity of exact methods for TSP.

Algorithm	Time complexity
Exhaustive search	$O(n!)$
Held-Karp algorithm (HELD; KARP, 1962)	$O(n^2 2^n)$
Ambainis et. al. (AMBAINIS <i>et al.</i> , 2019)	$O(1.728^n)$

Heuristic methods find a feasible solution for the TSP fast. However, these methods approach the optimal solution but do not ensure optimality. The nearest neighbor algorithms (HEINS *et al.*, 2023; AGRAWAL *et al.*, 2021; MARTÍNEZ; GARCÍA, 2021), 2-opt (ZHANG; HAN, 2022; GUNDUZ; ASLAN, 2021; PANWAR; DEEP, 2021), 3-opt (ZHONG, 2021; TUANI *et al.*, 2020; KHAN; MAITI, 2019), and Lin-Kernighan algorithms (LIN; KERNIGHAN, 1973; ZHENG *et al.*, 2023; TAILLARD, 2022; ZHOU *et al.*, 2022) are some of the well known heuristic methods. Table 2 shows the time complexity of some heuristic methods.

Tabela 2 – Time complexity of heuristic algorithms for TSP.

Algorithm	Time Complexity
Nearest neighbor	$O(n^2)$
Christofide heuristic	$O(n^3)$
Christofides and Serdyukov (van Bevern; SLUGINA, 2020)	$O(n^3)$
2-opt and 3-opt	$O(n^2)$ and $O(n^3)$
Lin-Kernighan (LK)	$O(n^{2.2})$

A meta-heuristic is a higher-level technique for solving optimization problems (BARA'A *et al.*, 2021). Some usual meta-heuristic methods used for the TSP are simulated annealing (SA) (EZUGWU *et al.*, 2017; LIN *et al.*, 2016), genetic algorithm (GA) (ZHOU *et al.*, 2022; LU *et al.*, 2020; XING *et al.*, 2008), ant colony optimization (ACO) (GONZÁLEZ *et al.*, 2022; ZHOU *et al.*, 2022; EBADINEZHAD, 2020), particle swarm optimization (PSO) (ZHOU *et al.*, 2018; ZHONG *et al.*, 2018; PHUNG *et al.*, 2017) and reinforcement learning (RL) (LIU; ZENG, 2009; OTTONI *et al.*, 2021; ZHOU *et al.*, 2022; LUO *et al.*, 2022; HU *et al.*, 2020). Table 3 shows the time complexity of some meta-heuristic methods.

The choice of method depends on the instance size, the desired accuracy, and the available computational resources. Some metrics used to evaluate the search methods for the TSP are the time complexity and the solution quality (closeness to the global optimum).

Tabela 3 – Time complexity of meta-heuristic algorithms for TSP.

Algorithm	Time Complexity
Tabu search (TS)	$O(n^3)$
Simulated annealing (SA)	$O(n^2)$
Genetic algorithm (GA)	$O(n^2)$
Ant colony optimization (ACO)	$O(n^2)$
Reinforcement learning (RL)	$O(n^2)$

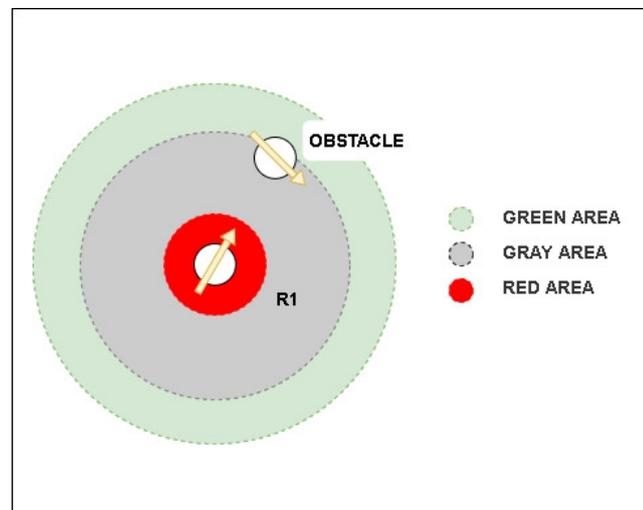
4.2 DEVELOPED STRATEGY FOR SWARM FORMATION TASK

In this section, we detail the group of robots (swarm) used in the formation task, the control algorithm used for navigation, the desired robot group shapes, highlighting the leader robot's role, and the several steps of the proposed strategy.

4.2.1 Navigation

The robots navigate to reach a specific formation during the formation run phase. For this, each Son robot must have the magnitude and angle of the distance vector regarding the Father robot and the Father's orientation. The robots get the distance vector data through IR sensors and the Father's orientation by exchanging messages (Algorithm 1). The robots exchange messages through a wireless network, each with a media access control address (MAC address).

Robots have eight infrared sensors for the detection of objects. Therefore, we limit three areas surrounding the robots: the Green Area, the Gray Area, and the Red Area (see Figure 67). These areas have a circular shape centered on the robot's center of mass.

Figura 32 – Areas surrounding the robots called Green Area, Gray Area, and Red Area.

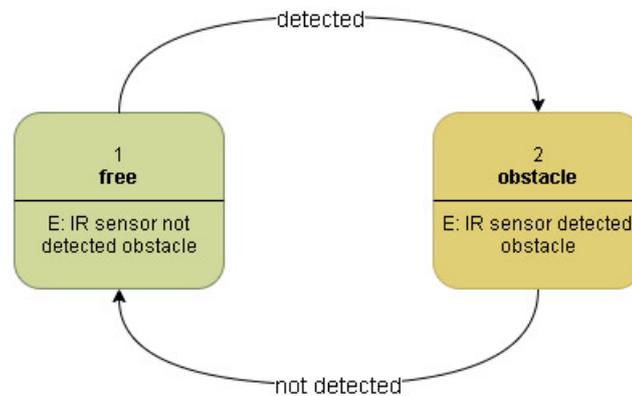
Fonte: Own authorship

Robots detect obstacles placed within the green and Gray Area, with a diameter equal to Φ_{green} and Φ_{gray} . The Red Area with a diameter equal to Φ_{red} is a safety area looking to guarantee the physical integrity of the robot. This resource is only used if there is a failure in controlling obstacles diversion. The area's diameters obey the following relation:

$$\Phi_{red} < \Phi_{gray} < \Phi_{green} \quad (5)$$

In this work, robots fall into two possible states: 1- Free and 2 - Obstacles (see Figure 33). The Free state implies the lack of obstacles within the detection area. Thus, the robot, in a Free state, can execute its control actions without route deviations.

Figura 33 – State Machine - navigation



Fonte: Own authorship

The Obstacle state suggests obstacles within the detection area. In an Obstacle state, the robot changes its wheels' speed, implementing a maneuver to divert the obstacle.

The Son robot achieves the distance, angle, and orientation set to its position within the robot group through proportional control actions. The wheels' speed control is carried out according to the Algorithm 10. The wheels can act one by one, providing a linear and/or angular displacement of the robot. The set point variables are *setDist* and *setAng*. The *setDist* variable is the distance, and the *setAng* variable is the angle, using the Father robot as a reference. The output variables are the angular speed of the left and right wheels.

The switching from the Free to the Obstacle state occurs when the robot detects any obstacle. Otherwise, it remains in the Free state. When a robot, through onboard sensors, detects an obstacle (another robot), it seeks to communicate with this robot.

Algorithm 2 Control Algorithm

```

requer setDist, setAng, currentDist, currentAng;
inserir sLeft, sRight;
0: errorAng = normalize(setAng - currentAng)
0: errorDist = setDist - currentDist;
0: enquanto (errorAng>tolerAng) or (errorDist>tolerDist) faça
0:    $sLeft = sLeft + K_D * errorDist + K_A * errorAng$ ;
0:    $sRight = sRight + K_D * errorDist - K_A * errorAng$ ;
0: finaliza enquanto=0
  
```

A robot communicates by sending a message with its identification, ID . Then, the robot that receives the message responds through another message with its identification.

All robots have an ID for several functions within the robot group. For instance, the ID defines which robot has the movement priority when a robot rendezvous with other robots during a formation task. We ruled the robot with movement priority to have the lowest ID among the robots with the chance of collision.

The navigation starts by running Algorithm 3 through local communication among robots. The Father robot provides the Son robot with the distance vector angle between Father and Son, θ_{PF} (line 2).

Algorithm 3 Navigation Task

```

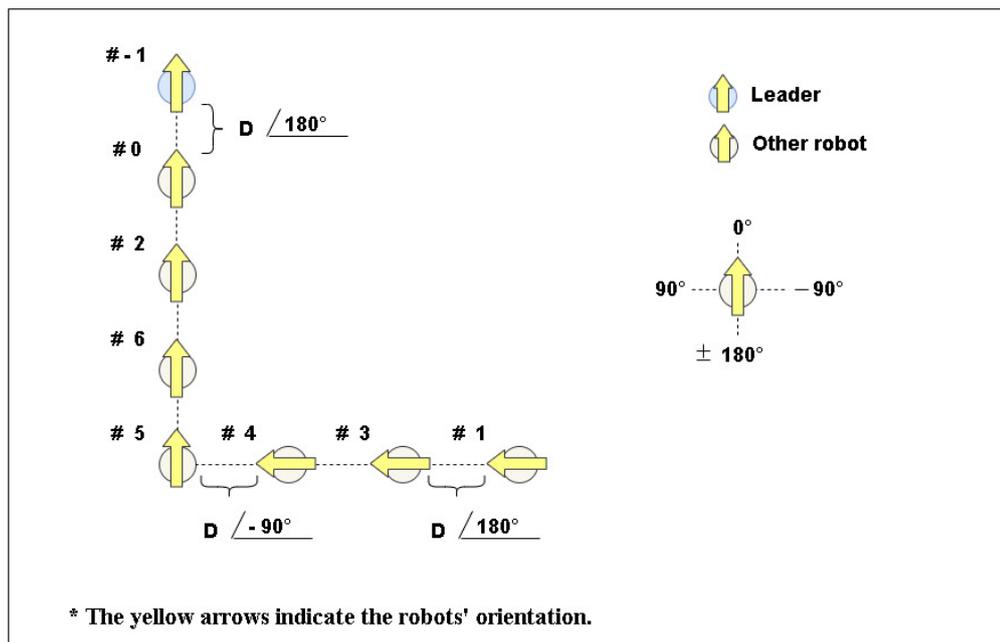
requer robotType, robotNavigating;
0: enquanto robotNavigating==true faça
0:    $get(\theta_{PF}, \theta_{FP})$ ; (Algorithm 1)
0:   calculate(currentDist, currentAng);
0:   control(setDist, setAng, currentDist, currentAng); (Algorithm 10)
0:   control(currentDist,  $\theta_{PF}$ , currentDist,  $\theta_{FP}$ );
   (Algorithm 10)
0:   send("Navigation Task Running.");
0:   se Failure==true então
0:     robotNavigating=false
0:     send("Robot Stopped.");
0:   finaliza se
0: finaliza enquanto
0: send("Navigation Task Completed.")=0
  
```

The sensors embedded in robots provide the data used to estimate the distance and angle of the distance vector, used by the control variables $currentDist$ and $currentAng$ (line 3). The controller uses the information about the priority variables to act in the movement of robots (lines 4 and 5). While the navigation task does not end or there is a complete communication failure between robots, the control repeats several times. In case of a total loss of communication, the Son robot cannot carry out the control actions due to the lack of information about its father, remaining stopped when the failure occurs.

4.2.2 Desired robot group shapes

The navigation of a robot swarm while maintaining a geometric shape formation requires controlling the robot's distance, angle, and orientation regarding a reference point (see Figure 34). Thus, the solution found through the Algorithm 5 for the shortest path problem provides the robots with the optimal path. This path establishes the positions of each robot within the group, providing a reference point for the formation task.

Figura 34 – Robots in formation: the distance and angle to the reference point



Fonte: Own authorship

The reference for a robot is always its father, except the leader. For example, for the L formation shown in Figure 34, each robot must be at a distance D and angle of 180° regarding its reference robot (father), except the robot #4, which must maintain a distance D and angle -90° .

4.2.3 Leader robot's role

The leader initiates the formation task by recruiting unknown neighbors. Besides recruitment, the leader performs calculations providing the robots with the optimal path, i.e., the relationship between Father and Son. The leader is also responsible for starting and finishing the communication between robots during the execution of the formation task (Algorithm 1).

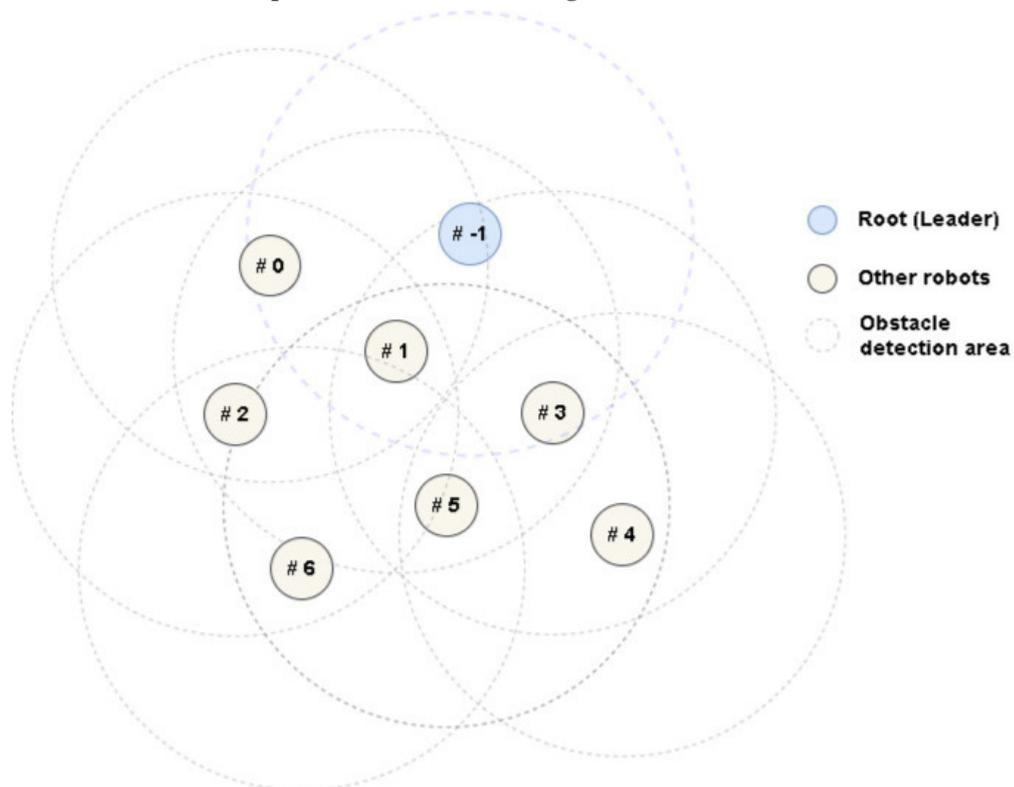
4.2.4 Strategy's phases

The proposed strategy for the formation task comprises three sequential steps or sub-tasks: 1 - recruitment, 2 - planning, and 3 - execution.

Recruitment

Robots are unaware of their environment, they must use their sensors to detect and communicate with neighboring robots. The range of the sensors allows direct communication only between robots in the same neighborhood, as shown in Figure 35. This way, the communication between two robots from different neighborhoods happens only through other robots, using wave propagation. As shown in Figure 36, the leader can send a message to robot #6 through robots #0 and #2.

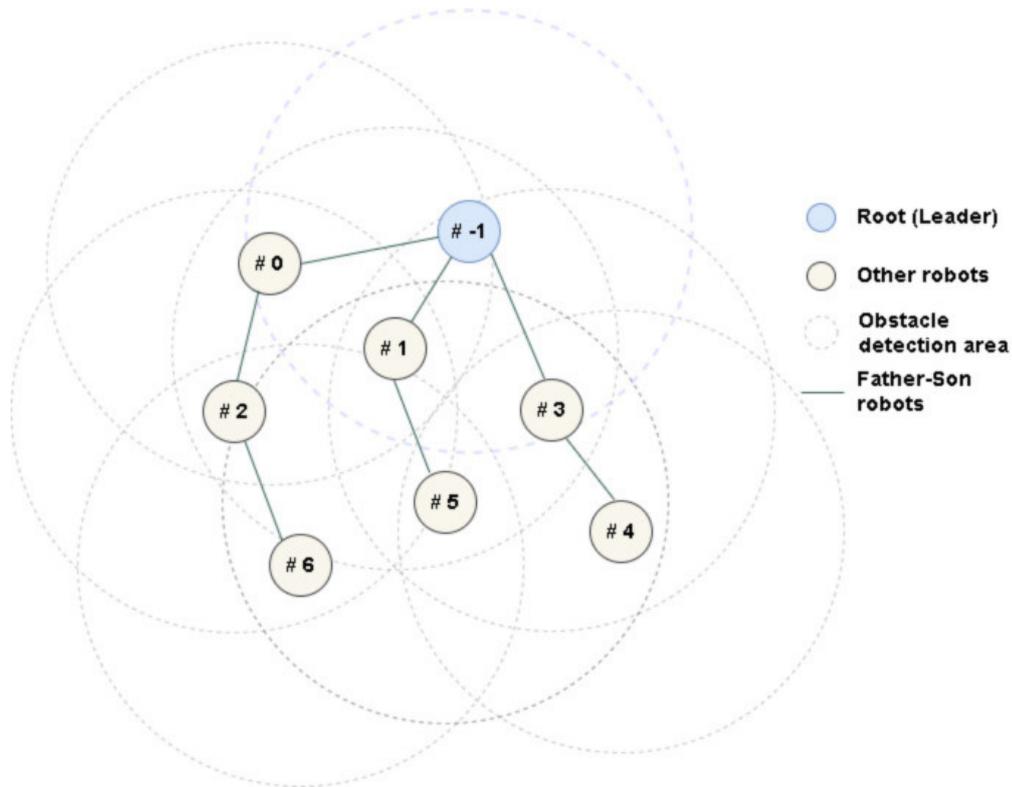
Figura 35 – Robots in random position and unknown neighborhood - illustration of obstacle detection area



Fonte: Own authorship

The leader initiates communication with its neighbors by sending them a message, as described in the procedural algorithm 4. Robots that receive a message are called Son robots. Robots that send a message to other robots are called Father robots. All other robots except the leader can act as Father and Son robots. However, a robot can only send a message to another

Figura 36 – Recruitment phase completed



Fonte: Own authorship

robot after receiving a message from a Father robot. At this stage, a robot can have many Son robots but only one Father robot. In this stage, the Father-Son relationship is randomly set.

Algorithm 4 Wave Recruitment Algorithm

```

requer  $ID$ ;
inserir  $feedback\_message$ ;
0: se leader então
0:   Send a  $message\_ID$  to your neighbors;
0:   Get a  $feedback\_message$ ;
0: senão, se fatherless então
0:   Get a  $message\_ID$ ;
0:   se  $message\_ID$  então
0:     Send a  $message\_ID$  to your neighbors;
0:     Perform the event;
0:   finaliza se
0: senão,
0:   Get a  $feedback\_message$ ;
0:   Send a  $feedback\_message$ ;
0: finaliza se=0

```

Feedback starts with a message back to the Father robot. This message contains information, such as its identification (ID), the module and the angle of the distance vector between the Father and Son robot.

The last feedback message received by the leader points out the end of the recruitment.

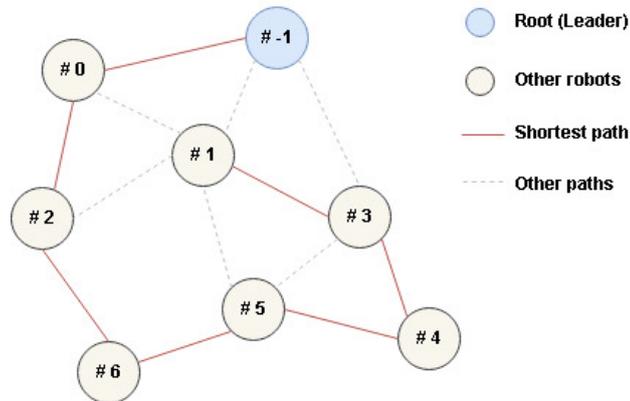
Planning

In the planning phase, the leader defines a new Father-Son relationship, where each Father robot can have only one child. The leader maps the robot group according to the information obtained during the recruitment phase, building the TSP model. After that, the leader seeks a new set of Father-Son relationships. These relationships must minimize the sum of the distance between a Father robot and a Son robot by solving the TSP, as discussed in Section 4.1.

The newly established Father-Son relationship will remain the same throughout the Execution phase (until the end of the formation task). In this way, the network topology for communication and sensing between swarm robots will remain fixed.

Inspired by the Traveling Salesman Problem, robots (vertices) represent cities, and distances between robots (edges) represent a path between cities (Figure 37). The leader seeks to find a path that minimizes the sum of the distance between robots, according to Equation 4.

Figura 37 – Solution of the Traveling Salesman Problem: the red line shows the shortest possible path that connects all robots



Fonte: Own authorship

The found path establishes a new relationship between the Father and Son robots used for wave propagation during the robots' communication. In addition, this new establishment determines the robots' positions when reaching the desired formation and the reference needed for executing the formation task, as seen in Section 4.2.2. After that, the leader sends the robots the new Father and Son relationship and the desired shape for the formation task. Robots receive this information through exchanging messages (Algorithm 1) before starting the execution phase.

Execution

The robots must navigate to reach the magnitude and angle of the distance vector concerning its reference (Father robot) during the execution phase. For this, the Father and Son robots must exchange messages between them using Wave Swarm (Algorithm 1) and the sensing of the environment. A necessary premise for the implementation of Wave Swarm for the formation task is:

- A Father robot has only one Son robot.

The existence of a single Son robot reduces the probability of sensing and communication failures occurring during navigation and reduces the number of messages exchanged.

The execution phase ends with a permanent failure in communication between the robots or after the group has reached the desired formation.

4.2.5 Communication failures

Maintaining communication in a swarm of robots is a huge challenge. Communication can fail at any time. However, the probability of failures can increase or decrease according to the robot's mobility.

In the Recruitment phase, the robots use the infrared signal to communicate. Since they remain stopped, the number of communication failures is reduced.

In the Planning phase, the robots stand stopped but they use the radio frequency signal to communicate.

In the Execution phase, the robots navigate the environment, seeking the desired shape and using the radio frequency signal to communicate. The probability of failures in this phase increases due to a possible sensing failure between the Father and Son robots. The sensing failure occurs when one of the robots has impaired vision due to the appearance of an obstacle (another robot). This failure can be temporary or permanent. Temporary failures happen more frequently, but the robots reestablish sensing after the deviation maneuver end. Permanent failures occur less frequently; the robots remain stopped and cannot perform tasks since the communication link is permanently broken. The experiment is over when a permanent failure is detected, and we must start the next one.

4.3 EXPERIMENTS AND METRICS FOR THE TSP

In this work, we conduct experiments to compare the performance for TSP of two meta-heuristics, RL and AG, using the library TSPLIB95 (REINELT, 1995) before implementing the planning algorithm.

The data files chosen for the simulation are burma14, ulysses22, berlin52, pr76 and kroD100. The number in the filenames represents the number of cities. The name of the experiments is formed by the filename-method, for example, berlin52-RL and berlin52-GA. The total number of experiments performed is equal to 10. We performed 100 simulations for each experiment using Matlab software (version R2022b).

All experiments ran in a notebook with an Intel Core Processor i5-2430M 2.4 GHz, 8GB of RAM memory, and a Microsoft Windows 10 Pro operating system.

Reinforcement learning - Q -learning

The Q -Learning algorithm enables learning through the interaction of an optimal π^* policy in the absence of a system model. The algorithm idea is to learn through interaction with the environment, using the best actions to find the optimal value function $Q^*(s,a)$.

The acquired knowledge is stored in an array $Q(s,a)$, which has the estimated values for each state-action pair. The Q function is estimated through equation 6 (SUTTON; BARTO, 2005):

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \max_a Q(s',a) - Q(s,a)] \quad (6)$$

where, α is the learning rate, r is the immediate reward, γ is the discount factor, and, $\max_a Q(s',a)$ is the utility of state s .

We must ensure that each pair of states is visited many times so that convergence from Q to Q^* (optimal value function) is guaranteed gradually and slowly (SUTTON; BARTO, 2005).

Algorithm 5 implemented in this work has all parameters set to: α is equal to $\frac{1}{1+S_v}$, γ to 0.01 and r to $-D_r$; where S_v is the number of visited states; D_r is the euclidean distance between the robots (OTTONI *et al.*, 2021).

Algorithm 5 *Q*-Learning Algorithm

```

0: Initialize  $Q(s,a) = 0$  for each pair (s,a)
0: Observe the state s
0: enquanto lastEpisode faça
0:   Initialize s (initial)
0:   enquanto finalState faça
0:      $a \leftarrow$  action for s using policy derived from Q;
0:     Take action  $a$ , observe the reward  $r$  and the next state  $s'$  ;
0:     Update  $Q(s,a)$  (equation 6)
0:      $s \leftarrow s'$ 
0:   finaliza enquanto
0: finaliza enquanto=0
  
```

Genetic Algorithm

Genetic algorithm use the evolution of a population of individuals to find the best TSP solution. Individuals are possible routes, and the population is a set of N individuals. Each individual has a fitness value, calculated by the fitness function, f_i , Equation 7.

$$f_i = 1/Z \quad (7)$$

where Z is given by Equation 4.

The main objective is to find the best route for the TSP. For this, we seek to maximize the fitness function, following the steps below:

- **Selection:** it is used to select a number of individuals from the population, N_S through the tournament selection. The selected individuals compete against each other. The individual with the highest fitness wins and will be included as one of the next-generation population. The number of individuals competing in each tournament corresponds to the tournament size, T_s . In this way, diversity is guaranteed, and the most and least fit individuals are selected according to the tournament. There are other forms of selection, such as roulette wheel and rank-based roulette wheel selection (RAZALI *et al.*, 2011).
- **Crossover:** This operation combines two individuals from the population. Once the parent individuals are selected (selection phase), the crossover operators are applied with a p_C probability, resulting in N_C individuals.
- **Mutation:** is used to modify an individual with probability p_m , increasing the population's genetic variability. A population with more significant genetic variability guarantees increased search space.

The population of children created by selection, crossover, and mutation replaces the original population of parents in the next generation, but the worst children $W_C(\%)$ are replaced by the best parents in the current population.

The parameters of the GA are the same for all experiments: N is 10 times larger than the number of instances, N_S to $0.4N$, T_S to 2, p_C to 0.85, p_M to 0.01 and $W_C(\%)$ to 10%. The total number of generations equals 1000.

4.3.1 Metrics

We evaluate the performance of RL and AG through the average simulation time, $t_b(s)$, and the quality of the solution through a percentage given by the equation 8.

$$S(\%) = \frac{S - S^*}{S^*} \cdot 100\% \quad (8)$$

where S is the average best solution found by RL or AG, and S^* is the best solution provided by the TSPLIB95 library.

Descriptive statistics such as mean, median, skewness, and kurtosis allow us to analyze experiments' results.

We conducted data normality tests using the Shapiro-Wilk test (SHAPIRO; WILK, 1965) ($p \leq 0.05$). Also, we performed the Bootstrap test (EFRON; TIBSHIRANI, 1994) (non-parametric) to assess the difference between the averages with a confidence interval of 95% Bootstrap with 10^6 Bootstrap samples.

The algorithm (RL or AG) with the best performance in these experiments will be used in the planning phase of the proposed strategy.

4.4 EXPERIMENTS SET UP AND METRICS FOR PROPOSED STRATEGY

In this work, we carried out some experiments to validate and evaluate the efficiency of the proposed strategy for the formation task.

The considered environment is unknown, and the leader-follower approach is adopted in such experiments. The results show the behavior of four groups of robots of different sizes: 6, 10, 15, and 20, as illustrated in Figure 38. These robots must take four shapes during the formation task: inline, square, circle, and letter N.

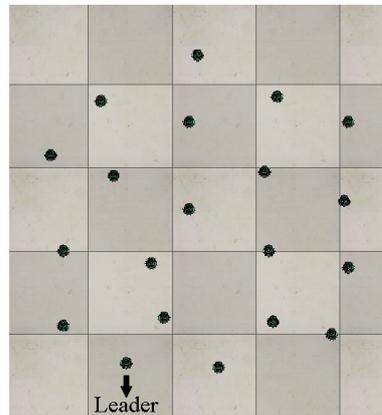
The experiments' stopping criterion is the end of the formation task or the simulation

time, t_s , is greater than or equal to the maximum time allowed for the simulation, t_{max} , which ranges with the robot group and with the desired robot group shaped.

We chose the CoppeliaSim PRO EDU (version 3.5.0 - rev.4) simulator to execute the experiments. The implementation is coded through scripts in the LUA language, following the standard defined by the simulated e-puck developers.

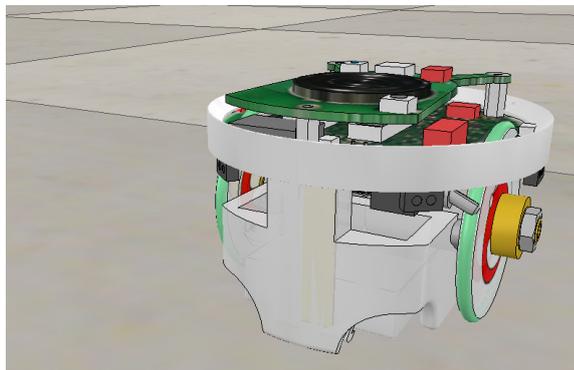
The e-puck2 is a differential robot with eight infrared (IR) sensors around it. These IRs are used to measure the proximity of objects and for communication, as illustrated in Figure 39. The IR sensor can measure a range of 3 meters. Visual interaction with the user is possible through eight light-emitting diodes (LED) surrounding the e-puck2. Also, the e-puck2 has a network of wireless technology, Wi-Fi.

Figura 38 – Experiment with twenty robots: initial position



Fonte: Own authorship

Figura 39 – Simulator CoppeliaSim (VRep): e-Puck



Fonte: Own authorship

All experiments ran in a notebook with an Intel Core Processor i5-2430M 2.4 GHz, 8GB of RAM memory, and a Microsoft Windows 10 Pro operating system.

We performed 50 simulations for the same group, adding up to 200 simulations for each experiment (A, B, C, and N). The robots' positions are randomly initialized, and their distance ranges from 1 to 2 meters in each experiment. Each simulation represents a solution found for the TSP.

4.4.1 Metrics

The approach proposed in this work uses the concept of message propagation to exchange information to assist in the sequential execution of tasks, forming complex tasks as the formation task. However, information propagation can demand many exchanged messages proportionally to the number of robots, increasing tasks' processing and execution time.

Therefore, the number of successfully completed simulations, N_s , and the simulation time, t_s , are some of the metrics used to evaluate the processing effort spent on the execution of experiments.

We also measure the average distance traveled by the robots, D_{dist} , the number of messages exchanged by the leader, N_{msg} , as well as the sum of the distance between a Father robot and a Son robot through solving the TSP, D_q .

Descriptive statistics such as mean, median, skewness, and kurtosis allow us to analyze experiments A, B, C, and N data.

We conducted data normality tests using the Shapiro-Wilk test (SHAPIRO; WILK, 1965) ($p \leq 0.05$). Also, we performed the Bootstrap test (EFRON; TIBSHIRANI, 1994) (non-parametric) to assess the difference between the averages with a confidence interval of 95% Bootstrap with 10^6 Bootstrap samples.

The Spearman's correlation analysis and the linear regression models are also used, in which the predictor variable D_q and the dependent variables t_s (model 1), D_{dist} (model 2) and N_{msg} (model 3).

4.5 SIMULATED RESULTS

This section shows the simulation results and discusses the obtained data with all experiments described in sections 4.3 and 4.4. The performance of RL and AG for the TSP is evaluated using the t_b and $S(\%)$ metrics. In addition, the metrics N_s , D_q , t_s , D_{dist} , and N_{msg} are used to validate the proposed strategy.

The results of experiments for the TSP are labeled with the name of the method file. For example, berlin52 data results using RL, we have berlin52-RL.

The results are labeled with a letter identifying the shape to form and a number setting the group's robot size. For example, for the formation task in line with 20 robots, we have experiment A20.

In this work, all experiments' results will be addressed. However, only the data in tabular and graphical form from the berlin52-RL, berlin52-AG, and C20 experiments will be available in this section.

The descriptive statistic, bootstrap mean, and Shapiro-Wilk test are shown in Appendix.

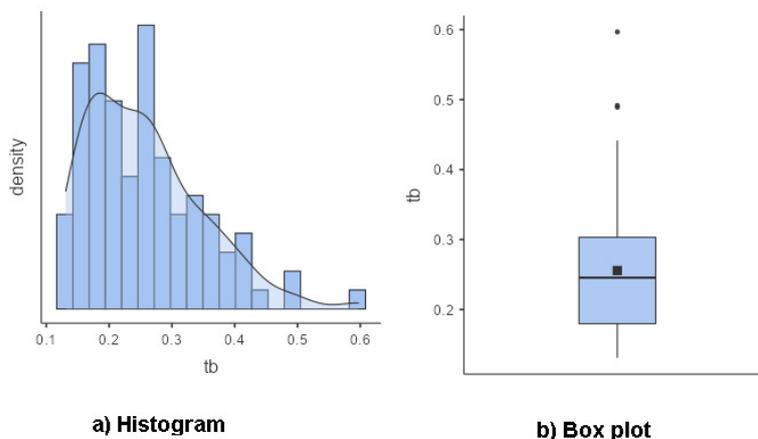
4.5.1 Experiments for the TSP

The data distribution from performed experiments shows different behaviors.

The data from all experiments, t_b , are extremely skewed (positive skewed), as illustrated in Figures 40 and 41.

In addition, the data from the burma14-RL, burma14-GA, ulysse22-RL, ulysse22-GA, and pr76-GA experiments showed extreme values of kurtosis, leptokurtic type. These high kurtosis represent the existence of heavy tails or outliers. The data from the berlin52-RL, berlin52-GA, pr76-RL, kroD100-RL, and kroD100-GA experiments present a distribution function of the platykurtic type.

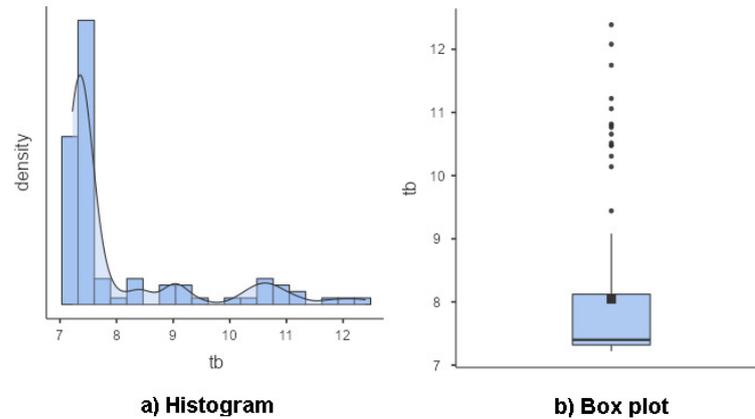
Figura 40 – Experiment berlin52-RL, variable t_b (s): mean equal to 0.256, median equal to 0.245, skewness equal 1.01 and kurtosis equal to 1.07 - a) Histogram; and, b) Box plot



Fonte: Own authorship

The data from the burma14-RL, berlin52-GA, pr76-GA, kroD100-RL and kroD100-GA

Figura 41 – Experiment berlin-GA, variable $t_b(s)$: mean equal to 8.05, median equal to 7.79, skewness equal 1.81 and kurtosis equal to 2.11 - a) Histogram; and, b) Box plot

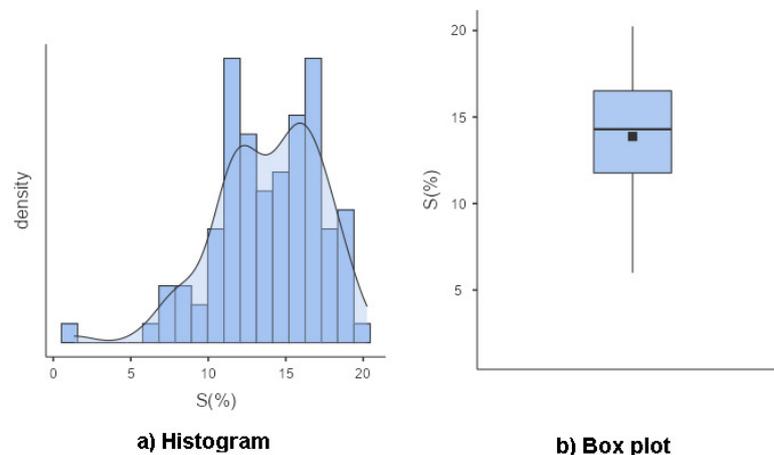


Fonte: Own authorship

experiments for variable $S(\%)$ present a nearly symmetrical (positive and negative skewed), as illustrated in Figure 43. The data from the ulysses22-RL, ulysses22-GA, berlin52-RL, and pr76-RL experiments introduce a slightly skewed (positive and negative skewed), as illustrated in Figure 42. The data from the burma14-GA experiment shows an extremely skewed (positive skewed).

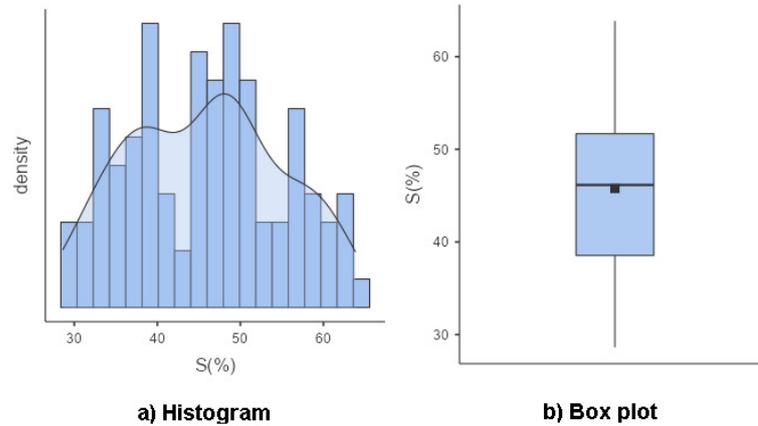
The kurtosis measures of the variable $S(\%)$ for burma14-RL, ulysses22-RL, ulysses22-GA, berlin52-RL, berlin52-GA, pr76-RL, pr76-GA, kroD100-RL and kroD100-GA present a distribution function of the platykurtic type, i.e., a flatness if compared to the Gaussian function. The data from the burma14-GA experiment shows a distribution function of the leptokurtic type, i.e., the existence of heavy tails.

Figura 42 – Experiment berlin52-RL, variable $S(\%)$: mean equal to 13.9, median equal to 14.3, skewness equal -0.694 and kurtosis equal to 0.811 - a) Histogram; and, b) Box plot



Fonte: Own authorship

Figura 43 – Experiment berlin52-GA, variable $S(\%)$: mean equal to 45.8, median equal to 46.2, skewness equal 0.0686 and kurtosis equal to -0.909 - a) Histogram; and, b) Box plot

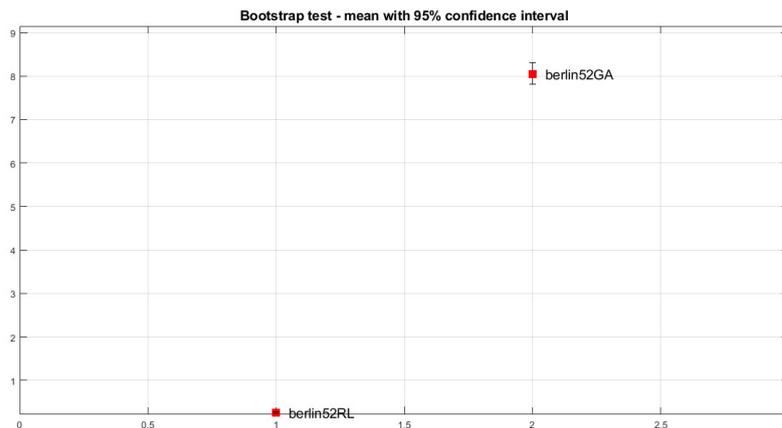


Fonte: Own authorship

The Shapiro–Wilk test point out the non-normality of the sampled data to t_b for all experiments and, $S(\%)$ for burma14-RL, burma14-GA and ulysses22-GA experiments. The Shapiro–Wilk test presents the data normality to $S(\%)$, for the ulysses22-RL, berlin52-RL, berlin52-GA, pr76-RL, pr76-GA, kroD100-RL and kroD100-GA experiments.

The data obtained from the experiment for t_b and $S(\%)$ presented different mean values. The Bootstrap test point out differences between the means of t_b and $S(\%)$, as illustrated in Figures 44 and 45.

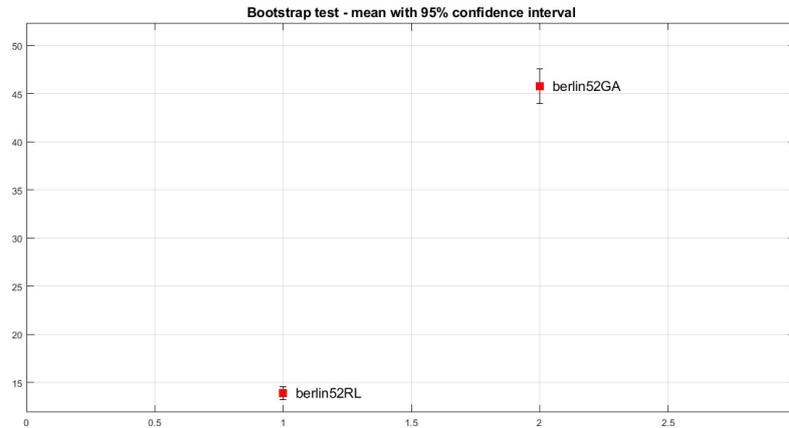
Figura 44 – Experiments berlin52-RL and berlin52-GA, variable $t_b(s)$: mean equal to 0.255, and 8.045 - Bootstrap mean with 95% confidence interval



Fonte: Own authorship

By analyzing these results, RL presented a better performance when compared to GA. Using RL, the results point to a lower value of t_b and $S(\%)$. Thus, in this work, we chose the RL to implement the Planning phase of the proposed strategy. Right away, we perform the experiments for the formation task with e-Puck as described in section 4.4.

Figura 45 – Experiments berlin52-RL and berlin52-GA, variable $S(\%)$: mean equal to 13.879, and 45.774 - Bootstrap mean with 95% confidence interval



Fonte: Own authorship

4.5.2 Experiments for the proposed strategy with RL

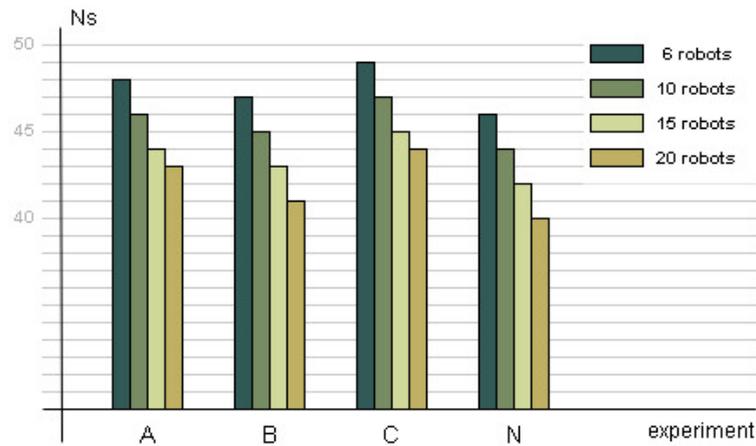
The successful experiments pointed out different values for the N_s variable, as shown in Figure 46. The proposed strategy resulted in an efficiency greater than 80% for the formation task. When comparing the results of experiments with the same number of robots, we observed a higher value of N_s for experiments C, followed by experiments A, B, and N. The results show that the higher the number of robots in the group, the lower the number of N_s . The value of N_s for experiments A, B, C, and N varies between 43 (86%) and 48 (96%), 41 (82%) and 47 (94%), 44 (88%) and 49 (98%), and, 40 (80%) and 46 (92%). The value of N_s for experiments with 6, 10, 15, and 20 robots varies between 46 (92%) and 49 (98%), 44 (88%) and 47 (94%), 42 (84%) and 45 (90%) and 40 (80%) and 44 (88%). The total value of N_s equals 714, representing a percentage of 89.25% of the number of simulations successfully performed.

The skewness measures of the variable t_s , D_{dist} and N_{msg} represent a weak asymmetry for all results, i.e., average value close to the median and the mode, as illustrated in Figures 47 to 49. The kurtosis measures of the variable t_s , D_{dist} and N_{msg} correspond to a distribution function of the platykurtic type for all, i.e., a flatness if compared to the Gaussian function.

The Shapiro–Wilk test point out the non-normality of the sampled data to D_q , and data normality to t_s , D_{dist} and N_{msg} , for the experiment A, B, C, and N, as seen in Table 11.

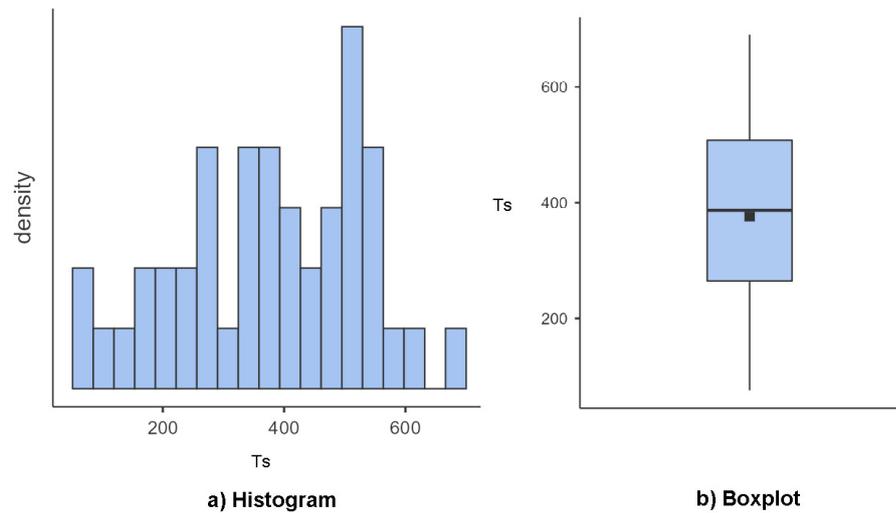
The data got from experiments A, B, C, and N for t_s , D_{dist} , and N_{msg} presented different mean values. The Bootstrap test indicates a difference between the means of t_s , D_{dist} and N_{msg} when comparing the data of each desired shape, as illustrated in Figures 50 to 52. Furthermore, it is possible to state that an increase in the number of robots implies an increase in the average

Figura 46 – Experiments A (inline), B (square), C (circle), and N (letter N): bar graph of the number of successfully completed simulations, N_s



Fonte: Own authorship

Figura 47 – Experiment C20, variable t_s (s): mean equal to 412.281, median equal to 423.377, skewness equal -0.245 and kurtosis equal to 2.485 - a) Histogram; and, b) Box plot

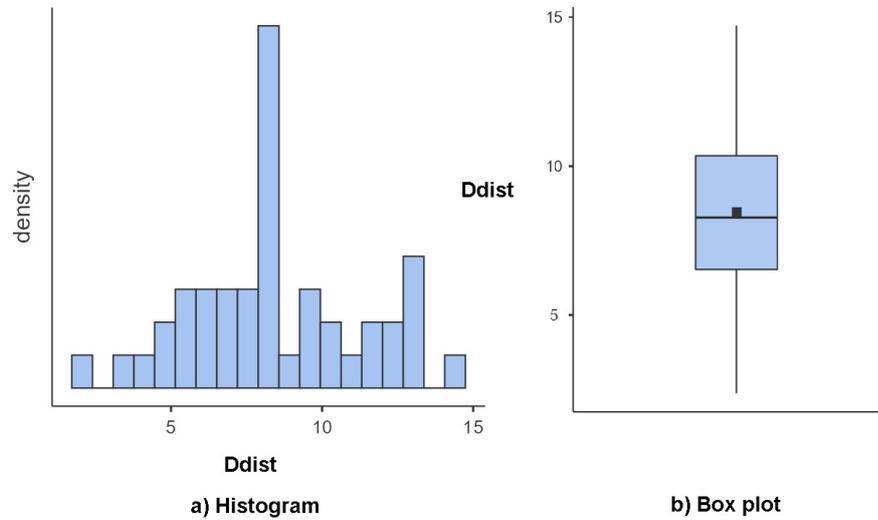


Fonte: Own authorship

values of t_s , D_{dist} , and N_{msg} .

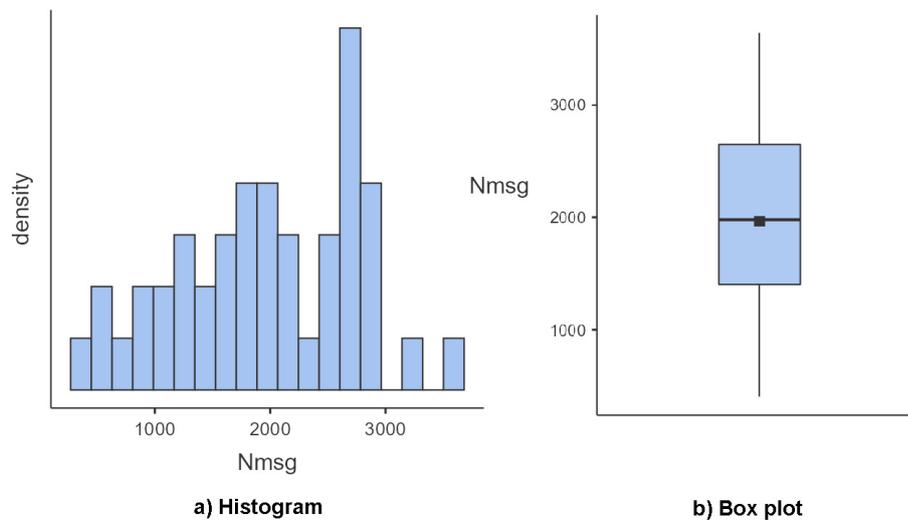
We carried out a Spearman's correlation analysis between the variables D_q and t_s , D_q and D_{dist} , and D_q and N_{msg} for experiment A, B, C, and N. The Spearman's correlation presents values between 0.801 to 0.925, D_q and t_s , 0.801 to 0.918, D_q and D_{dist} , 0.803 to 0.922, D_q and N_{msg} , as shown in Tables 12 to 14. The calculated P-value is less than 0.001 between the variables D_q and t_s , D_q and D_{dist} , and D_q and N_{msg} , in both showing a statistically significant correlation. In addition, Spearman's ρ presents different values between experiments, in which it demonstrates either a strong ($0.6 \leq |\rho| \leq 0.79$) or very strong ($0.80 \leq |\rho| \leq 1.0$) association, as shown in Table 12 to 14.

Figura 48 – Experiment C20, variable $D_{dist}(m)$: mean equal to 8.509, median equal to 8.346, skewness equal to +0.039 and kurtosis equal to 2.362 - a) Histogram; and, b) Box plot



Fonte: Own authorship

Figura 49 – Experiment C20, variable N_{msg} : mean equal to 2155.820, median equal to 2184.458, skewness equal to -0.216 and kurtosis equal to 2.443 - a) Histogram; and, b) Box plot

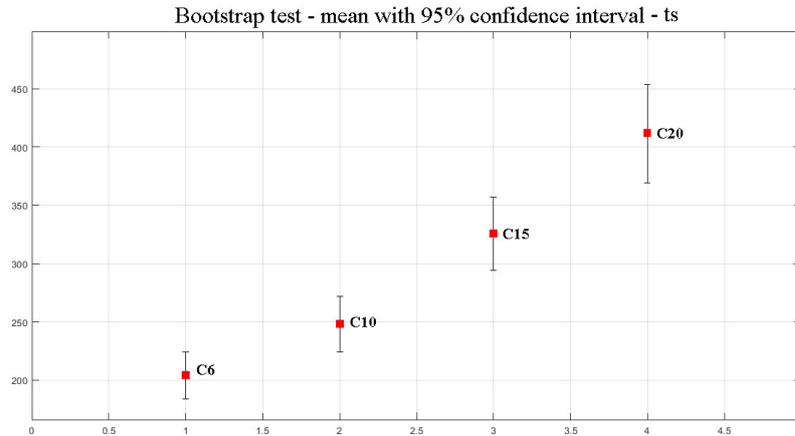


Fonte: Own authorship

We used a linear regression model to quantify the relationship between the predictor variable D_q and the dependent variables t_s (model 1), D_{dist} (model 2), and N_{msg} (model 3) for the experiments, as illustrated in Figures 53 to 55. The coefficient of determination R^2 present values between 0.650 to 0.854, model 1, 0.661 to 0.823, model 2, 0.646 to 0.865, and model 3, as shown in Tables 12 to 14.

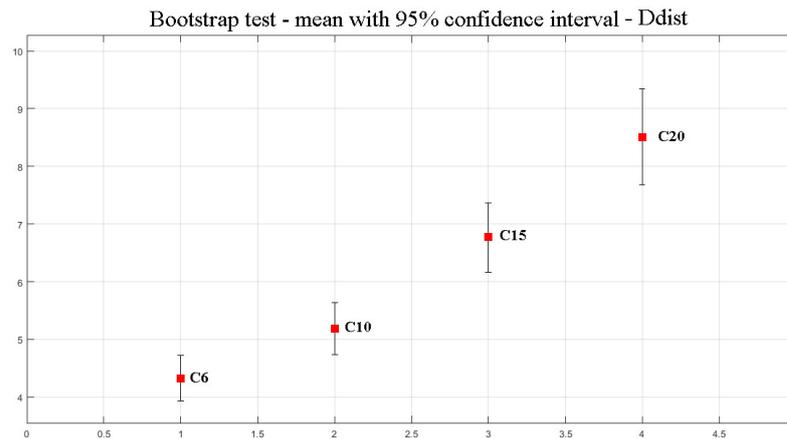
In addition, the P-value (F test) is less than 0.001, demonstrating that the predictor variable significantly influences each dependent variable in the three models. Thus, it is possible to observe that the lower the D_q value, the lower the t_s , D_{dist} , and N_{msg} values.

Figura 50 – Experiments C6, C10, C15, and C20: mean equal to 203.789, 249.372, 328.732 and 410.153 - Bootstrap mean with 95% confidence interval of the variable t_s



Fonte: Own authorship

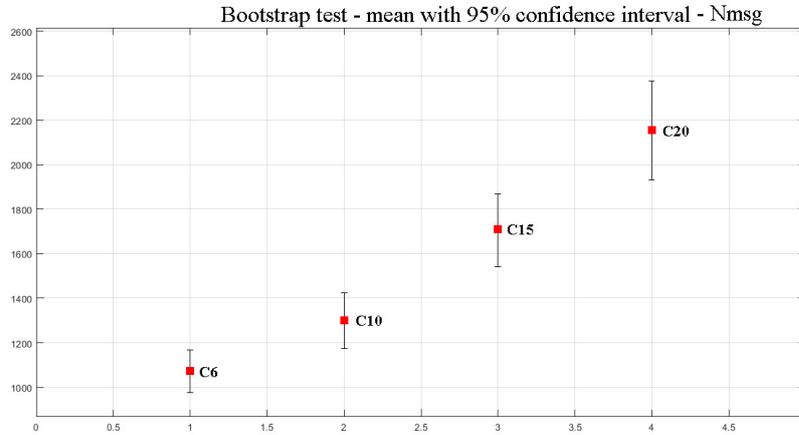
Figura 51 – Experiments C6, C10, C15, and C20: mean equal to 4.328, 5.185, 6.775 and 8.509 - Bootstrap mean with 95% confidence interval of the variable D_{dist}



Fonte: Own authorship

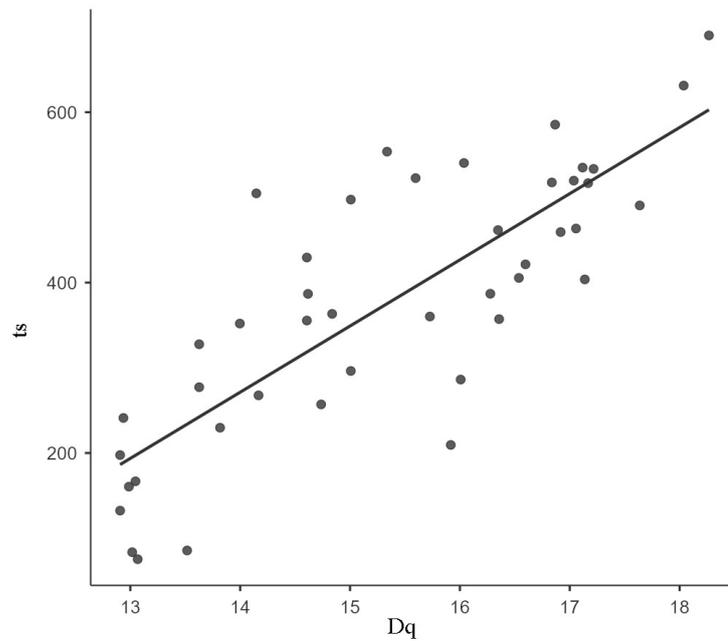
Multi-robot systems present great complexity by themselves. Therefore, when performing the formation task with different sizes of robot groups, it is possible to observe a decrease in the number of simulations successfully performed for experiments A, B, C, and N. In addition, the average results for simulation time, t_s , distance traveled by each robot, D_{dist} , and the number of messages exchanged by the leader, N_{msg} showed to be directly proportional to the number of robots regardless of the desired shape. However, when analyzing the data results from statistical tests, we observe that the strategic choice of the Father and Son robots (Wave Swarm) with TSP reduces all metrics: simulation time, the average distance traveled by each robot, and the number of messages exchanged by the leader, considering the same group of robots.

Figura 52 – Experiments C6, C10, C15, and C20: mean equal to 1072.38, 1300.361, 1708.613 and 2155.878 - Bootstrap mean with 95% confidence interval of the variable N_{msg}



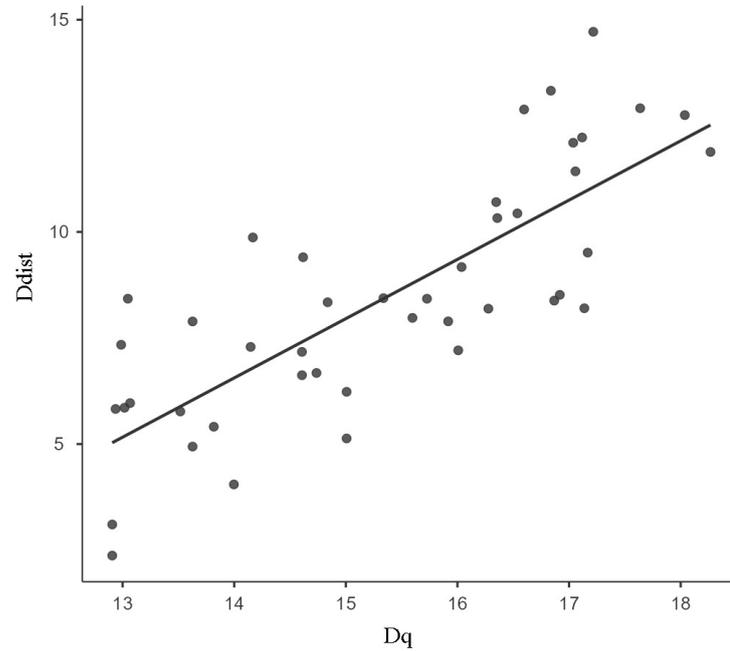
Fonte: Own authorship

Figura 53 – Experiment C20: Spearman’s Correlation - $\rho = 0.812$ and $P - value < 0.001$; Linear regression (predictor variable D_q and dependent variable t_s) - model fit measures: $R = 0.809$, $R^2 = 0.654$, $F = 79.5$, $df^1 = 1$, $df^2 = 42$ and $P - value < 0.001$



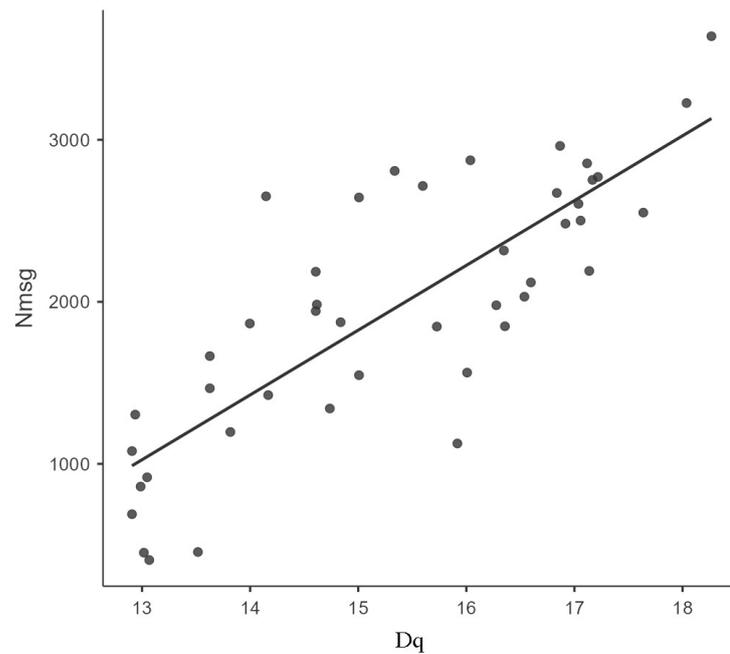
Fonte: Own authorship

Figura 54 – Experiment C20: Spearman's Correlation - $\rho = 0.805$ and $P - value < 0.001$; Linear regression (predictor variable D_q and dependent variable D_{dist}) - model fit measures: $R = 0.801$, $R^2 = 0.642$, $F = 70.6$, $df^1 = 1$, $df^2 = 42$ and $P - value < 0.001$



Fonte: Own authorship

Figura 55 – Experiment C20: Spearman's Correlation - $\rho = 0.803$ and $P - value < 0.001$; Linear regression (predictor variable D_q and dependent variable D_{Nmsg}) - model fit measures: $R = 0.809$, $R^2 = 0.659$, $F = 79.6$, $df^1 = 1$, $df^2 = 42$ and $P - value < 0.001$



Fonte: Own authorship

5 DISTRIBUTED STRATEGY FOR COMMUNICATION BETWEEN MULTIPLE ROBOTS DURING FORMATION NAVIGATION TASK.

Swarm robotics involves the study of the behavior of a set of robots in carrying out collective tasks, such as alignment, navigation and formation. During cooperative tasks execution, the communication among robots can contribute to successfully performing tasks through an efficient messages exchange.

This thesis proposes a communication strategy for swarm robots, the so-called Double-Wave Swarm, aiming at alignment and navigation tasks. The Double-Wave Swarm is an improvement of a prior Wave Swarm communication approach (SILVA-JR; NEDJAH, 2015; SILVA-JR; NEDJAH, 2017) that uses the concept of wave propagation for message exchange between neighbors.

5.1 DOUBLE-WAVE SWARM

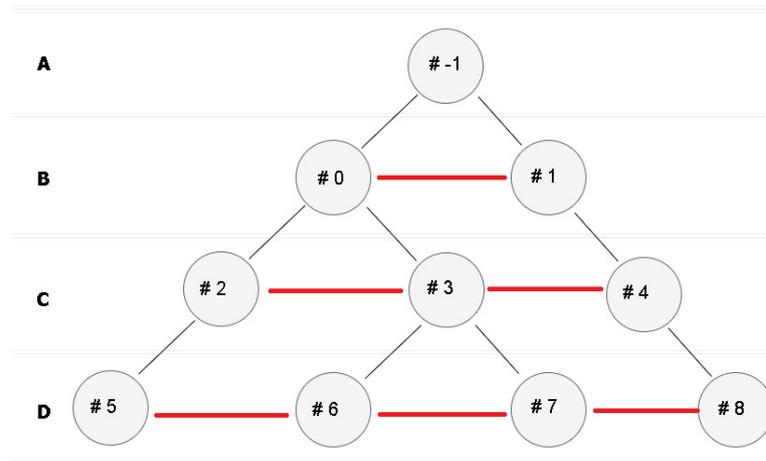
The Double-Wave Swarm strategy proposed in this thesis enhances the Wave Swarm Algorithm discussed above. This new strategy uses message propagation to information exchange between robots and also to manage tasks from the swarm.

Double-Wave Swarm establishes a relationship between robots at the same level (A, B, C or D), labeled as Friends, besides the relationship between Father and Son observed in the Wave Swarm (see Figure 56). Thus communication between individuals with the same degree of kinship beyond Father-Son's hierarchy is enabled. For example, robot #0 must communicate with its two Son robots, #2 and #3, besides its Father robot, # - 1 (see Figure 56). The management of swarm tasks occurs through communication between Father robot and Son robot.

The message propagation through Double-Wave Swarm starts with the Origin robot (Algorithm 6) and spreads through Son robots. The communication between Friends robots begins after the communication of Father robots with their Son robots (lines 7 to 15). In Figure 56, the robot #2 sends a message to the robot #3, and then sends a message to the robot #4, following the concept of wave propagation. As it is a propagation with feedback, the robot #4 sends an acknowledge message in the opposite direction, reaching the robot #2, confirming the receipt of the message by all Friends. After receiving the feedback message from a robot Friend and at the end of the event, the robot #2 sends the feedback message to its robot Father, #0.

Thus DWS propagation occurs in the graph depth as in the WS but also establishes

Figura 56 – Double-Wave Swarm - parental relationship modeled by a cyclic graph: the black line shows the relationship between the Father robots and the Son robots; and the red line points out the relationship between the Friends robots



Fonte: Own authorship

wave propagation through graph breadth.

Algorithm 6 Double-Wave Swarm Algorithm

requer Identification of the Father robot, the Son robots, the Friend robots and the Initiator robot;

```

0: se ORIGIN-FATHER então
0:   Send the information "1" to Son robots
0:   Perform the event
0:   Get the feedback message
0: senão,
0:   Get the information "1" from the Father robot
0:   Send the information "1" to Son robots
0:   se ORIGIN-FRIEND então
0:     Send the information "2" to the Friend robot
0:     Get the feedback message "2" from the Friend robot
0:   senão,
0:     Get the information "2" from the A1 Friend robot
0:     Send the information "2" to the A2 Friend robot
0:     Get the feedback message "2" from the A2 Friend robot
0:     Send the information "2" to the A1 Friend robot
0:   finaliza se
0:   Get the feedback message "1" from Son robots
0:   Perform the event
0:   Send the feedback message "1" to the Father robot
0: finaliza se=0

```

5.1.1 Messages exchanged between robots

The Double-Wave Swarm and Wave Swarm algorithms use the exchange of messages between robots for the synchronization of subtasks and the exchange of information relevant

to the execution of the formation navigation task. Message complexity and time complexity measures the resource consumption of these distributed algorithms. The message complexity is the total number of messages exchanged by the algorithm.

To measure the message complexity of the algorithms, we represent the parental relationship between the Father and Son robots (see Figure 27) through of a tree graph $G=(V,E)$ with $l > 1$ level, where V and E are sets of N nodes and $(N - 1)$ edges. Moreover, for each i level of the tree has i nodes. In this way, we can express the total number of nodes, N , with the number of levels, l , as follows:

$$N = \sum_{i=1}^l i = (1 + l) * \frac{l}{2}, \text{ or,} \quad (9)$$

$$l = \frac{1}{2}[(8N + 1)^{\frac{1}{2}} - 1] \quad (10)$$

For each message transmitted by the Origin robot (root), the number of messages exchanged between the robots using Wave Swarm is equal to:

$$M|_W = 2 * (N - 1), \text{ or,} \quad (11)$$

$$M|_W = l^2 + l - 2 \quad (12)$$

where N is the total number of robots, and l is the total number of levels of the tree-like graph.

The number of messages exchanged between the robots using Double-Wave Swarm is equal to:

$$M|_{DW} = M|_W + l * (l - 1), \text{ or,} \quad (13)$$

$$M|_{DW} = 2l^2 - 2, \text{ or,} \quad (14)$$

$$M|_{DW} = 4N - (8N + 1)^{\frac{1}{2}} - 1 \quad (15)$$

where N is the total number of robots, and l is the total number of levels of the tree-like graph.

We can see that for a large l , this implies a large N and $M|_{DW}$ twice as large as $M|_W$.

In distributed algorithms, the notion of time is not obvious, so we use some assumptions to characterize the time complexity (TEL, 2000).

- The time to process an event is zero time units.

- The transmission time (that is, the time between sending and receiving a message) is one unit of time.

In this way, the total transmission time for the Wave Swarm (Equation 16) and Double-Wave Swarm (Equation 17) algorithms is equal to:

$$T(N)|_W = 2N - 2, \quad (16)$$

$$T(N)|_{DW} = 2N + (8N + 1)^{\frac{1}{2}} - 5 \quad (17)$$

where N is the total number of robots.

In summary, Wave Swarm and Double-Wave Swarm have a message complexity and a time complexity equal to $O(N)$ in the worst case.

5.2 CONNECTIVITY

Connectivity is a fundamental concept from graph theory. Connectivity uses vertices and edges to depict the vertices' minimum number, k , and edges' minimum number, k' , to be removed to disconnect the remaining vertices from each other.

Furthermore, Whitney's Theorem (TEL, 2000) relates the connectivity of vertices and edges and the minimum degree of a graph: If G is a connected graph, then $k(G) \leq k'(G) \leq \delta(G)$, where, $k(G)$ is the connectivity of vertices of G or the connectivity of G ; $k'(G)$ is the edge connectivity of G ; and $\delta(G)$ is the minimum degree of G .

The connectivity of a graph is a well-known measure of a network's robustness. The larger the number $k(G)$ or $k'(G)$, the more robust the network represented by the graph will be.

The graph in Figure 27 from Wave Swarm is by definition, a tree, i.e., a connected and acyclic graph (SILVA-JR; NEDJAH, 2017). Thus, there is exactly one path between any pair of vertices, then $k = k' = 1$. The graph in Figure 56, derived from Double-Wave Swarm, is cyclic, so it does not represent a tree. Therefore, after calculating the values of $k(G)$ and $k'(G)$ we have both equal to 2. Thus, it is possible to conclude that the graph in Figure 56 has greater connectivity when compared to the graph in Figure 27.

The previous conclusion reinforces the hypothesis that the Double-Wave Swarm collaborates with robust communication between robots with a lower probability of failure when

compared to the Wave Swarm (SILVA-JR; NEDJAH, 2017) to navigation of multi-robots in formation.

Next, we intend to quantify, through the swarm and proposed experiments, the efficiency of the Double-Wave Swarm and its superiority when compared to the Wave Swarm, using the robot simulator CoppeliaSim (V-REP) and e-puck robots.

5.3 SWARM FORMATION NAVIGATION WITH DOUBLE-WAVE COMMUNICATION

In this work, we design a robot swarm with rigid formation to validate the Double-Wave Swarm and compare it with the former approach, Wave Swarm. The number of robots can change during navigation due to communication failures between robots caused by the diversion maneuvers of obstacles.

The scenarios and algorithms described below implement alignment and navigation tasks, formation control, and communication between robots and obstacles.

5.3.1 Alignment task

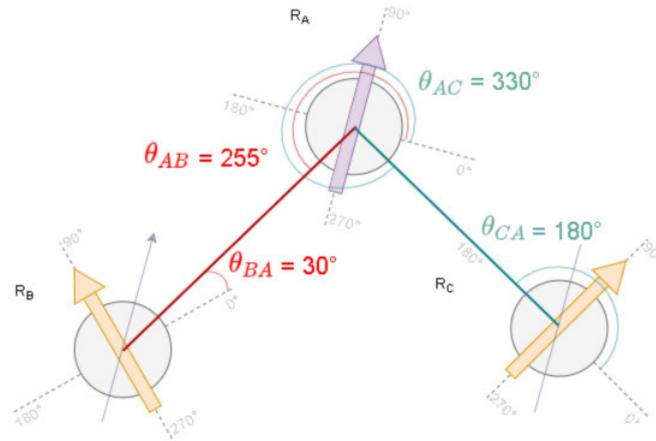
The alignment task begins with the Leader robot. During the execution of the task, each Father robot gets the distance vector angle concerning its Son robot through its embedded sensors. And then, the Father robot sends the angle value to its Son robot.

In turn, the Son robot acquires the distance vector angle concerning its Father robot through its embedded sensors. It sends this information to its Father robot during the message propagation process.

In Figure 57, there is a parental relationship between the Father robot, R_A , and its Son robots, R_B e R_C . In this example (Figure 57), the distance vector angle between R_A and R_B is equal to θ_{AB} ($\theta_{AB} = 255^\circ$), and between R_A and R_C is equal to θ_{AC} ($\theta_{AC} = 330^\circ$), considering the R_A coordinates. Following the same nomenclature, we have θ_{BA} ($\theta_{BA} = 30^\circ$) and θ_{CA} ($\theta_{CA} = 180^\circ$) the distance vector angle between the Son robots R_B and R_C in relation to R_A robot in the coordinates of R_B and R_C .

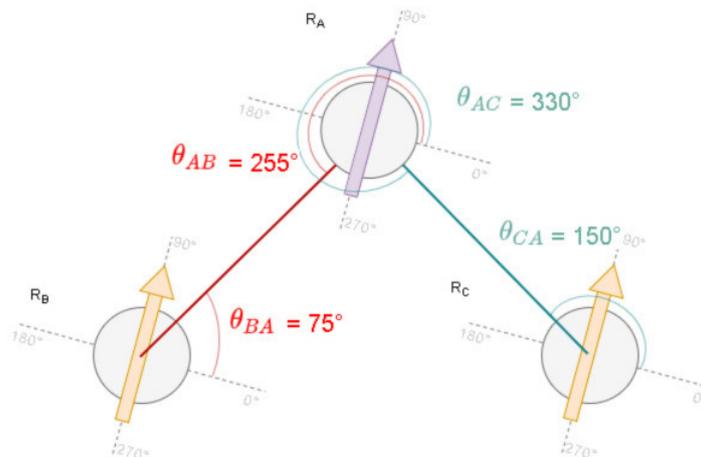
The Son robot aligns with the Father robot, if $|\Delta\theta|$ defined in Equation 18 is equal to π , where θ_{PF} is the angle between Father robot and Son robot, considering the coordinates of the Father robot, and θ_{FP} is the angle between Son robot and Father robot, considering Son robot's coordinates.

Figura 57 – Robots unaligned with their Father: R_A (Father), R_B (Son 01), e, R_C (Son 02)



Fonte: Own authorship

Figura 58 – Robots aligned with their Father: R_A (father), R_B (Son 01), e, R_C (Son 02)



Fonte: Own authorship

$$\theta_{PF} - \theta_{FP} = \Delta\theta \quad (18)$$

Thus figure 57 illustrates the unaligned Son robots, and Figure 58 shows the aligned Son robots.

The alignment task requires a rotation control looking for Son robots meet equation 18. Therefore, Algorithm 7 provides the information about the alignment of the Son robot and as

well acts on its orientation.

Algorithm 7 Son robot rotation: *rotationSon()*

```

requer  $\theta_{PF}, \theta_{FP}$ ;
inserir robotUnaligned;
0:  $\Delta\theta = \theta_{PF} - \theta_{FP}$ 
0: se  $\Delta\theta == 180$  então
0:   robotUnaligned = false;
0: senão,
0:   controlPose(currentDist,  $\theta_{PF}$ , currentDist,  $\theta_{FP}$ ); (Algorithm 10)
0:   robotUnaligned = true;
0: finaliza se=0

```

Algorithm 8 implements the alignment task. On line 1 of this algorithm, the Son robot calculates the value of θ_{FP} through its embedded sensors. The Double-Wave Swarm (Algorithm 6) or Wave Swarm (Algorithm 1) supplies the value of θ_{PF} (line 4), and then, the Son robot moves to align with its Father robot (line 6).

Algorithm 8 Alignment: *alignment()*

```

requer  $\theta_{PF}$ ; (Algorithm 1 or 6)
inserir robotUnaligned;
0: calculate( $\theta_{FP}$ );
0: robotUnaligned = rotationSon( $\theta_{PF}, \theta_{FP}$ ); (Algorithm 7)
0: enquanto robotUnaligned==true faça
0:   get( $\theta_{PF}$ ); (Algorithm 1 or 6)
0:   calculate( $\theta_{FP}$ );
0:   robotUnaligned = rotationSon( $\theta_{PF}, \theta_{FP}$ );
0: finaliza enquanto=0

```

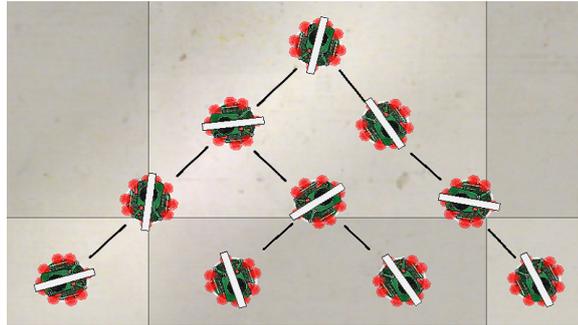
The Double-Wave Swarm or Wave Swarm ensures all robots receive the same message before the next one. We use e-puck robots to show this property. The red LEDs around the e-pucks light up to point out the start of sending and not receiving the feedback message. The blue LEDs signal the sending of the Son robot feedback message to the Father robot.

Figure 59(a) illustrates the start of the i th message transmission, where all robots are in red color, meaning only the message sent by Father robots. Figure 59(b) shows the j th message transmission, where some robots send the feedback message (blue) and other does not (red). Finally, Figure 59(c) shows the end of the transmission of the k th message with the receipt of the feedback message by all Father robots. And thus, all robots have completed the alignment task.

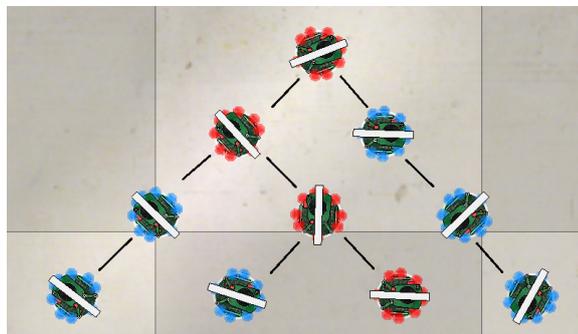
5.3.2 Navigation task

The swarm formation adopted in this thesis is a triangular formation with the Leader robot at the top of the triangle. The navigation task starts with Leader robot $\# - 1$. The Leader

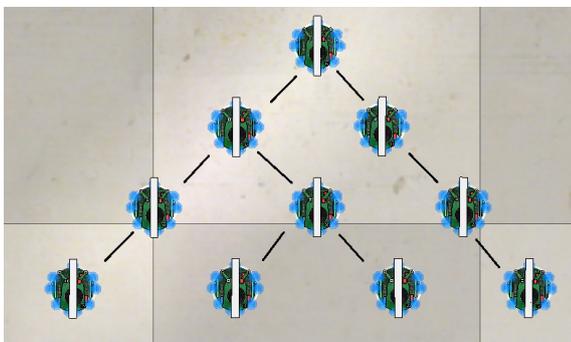
Figura 59 – Alignment task - the lines connecting the robots show the parental relationship between a Father robot and a Son robot



(a) Unaligned robots with red LEDs, showing the start of i message transmission



(b) Unaligned robots with red and blue LEDs, showing the j message transmission



(c) Aligned robots with blue LEDs, showing the end of k message transmission

Fonte: Own authorship

robot must follow a predetermined route: a straight line with 5 meters, obstacles free. The leader's goal is to drive on the straight line while managing the communication among robots. The task finishes when the leader reaches the end of the route.

During formation navigation, a swarm should preserve its original swarm shape. The Son robots must maintain priority values of distance, angle, and orientation in respect of their Father. However, the obstacles can appear during the navigation, implying in contrary actions to

the control actions for the formation navigation, prioritizing the diversion obstacles. The execution of successive obstacle avoidance maneuvers can lead to communication failures between the robots, resulting in the loss of the swarm shape.

Algorithm 9 assesses the robots' swarm behavior during the free and obstacles navigation. This algorithm waits for the robots' alignment to begin the navigation task. After the alignment, the local communication among robots starts with Algorithm 1 or 6 (line 7). Through the local communication, the Father robot provides to the Son robot the distance vector angle between Father and Son, θ_{PF} .

Algorithm 9 Navigation Task: *navigation()*

```

requer robotAlignment, robotType, robotNavigating;
0: se robotType!=Leader então
0:   enquanto robotAlignment==true faça
0:     print("Alignment Task Running.");
0:   finaliza enquanto
0: finaliza se
0: enquanto robotNavigating==true faça
0:   get( $\theta_{PF}$ ,  $\theta_{FP}$ ); (Algorithm 1 or 6)
0:   calculate(currentDist, currentAng);
0:   controlPose(setDist,setAng,currentDist,currentAng); (Algorithm 10)
0:   controlPose(currentDist,  $\theta_{PF}$ , currentDist,  $\theta_{FP}$ ); (Algorithm 10)
0:   print("Navigation Task Running.");
0:   se Failure==true então
0:     robotNavigating=false
0:     print("Robot Stopped.")
0:   finaliza se
0: finaliza enquanto
0: print("Navigation Task Completed.")=0

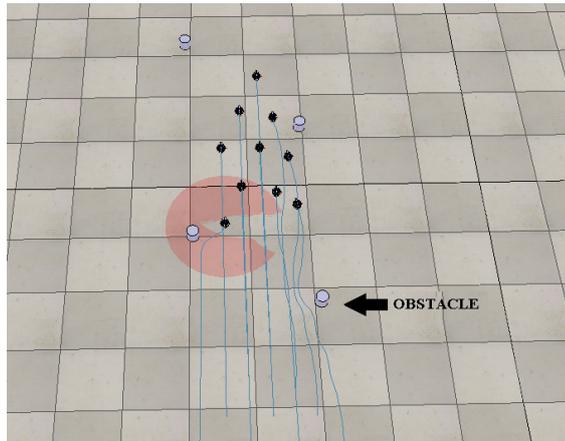
```

The sensors embedded in robots provide the data used to estimate the distance and angle of the distance vector, used by the control variables *currentDist* and *currentAng* (line 8). The controller uses the information about the priority variables to act in the movement of robots (lines 9 and 10). While the navigation task does not end or there is a complete communication failure between robots, the control repeats several times. In case of a total loss of communication, the Son robot cannot carry out the control actions due to the lack of information about its father, remaining stopped in its position when the failure has occurred. The red circle appears around the robot to illustrate a complete loss of communication (see Figure 60).

Double-Wave Swarm

In addition to Father and Son relationship, the Double-Wave Swarm also provides a connection between Friends. In this way, the communication among Friends can compensate for

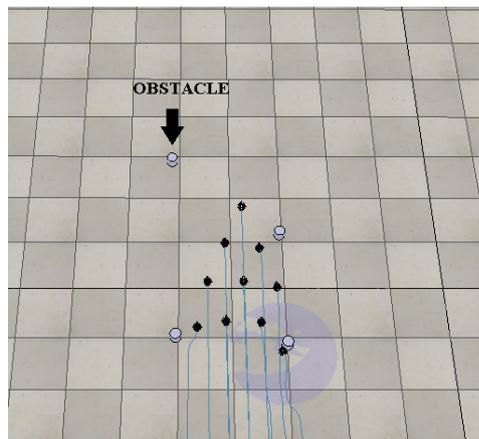
Figura 60 – Navigation with obstacle avoidance and total communication failure between Father and Son robots signaled with a red circle



Fonte: Own authorship

the communication failures of a Father and Son robot. When the communication link among father and son is lost but the communication may occur through a friend, a blue circle appears around the robot to show the fault (see Figure 61). In case of complete communication loss, a red circle appears around the robot under fault, highlighting there is no communication link between Father, Son and Friends robots (see Figure 60).

Figura 61 – Navigation with obstacle avoidance and indirect communication between Father and Son robots through a Friend robot, signaled with a blue circle



Fonte: Own authorship

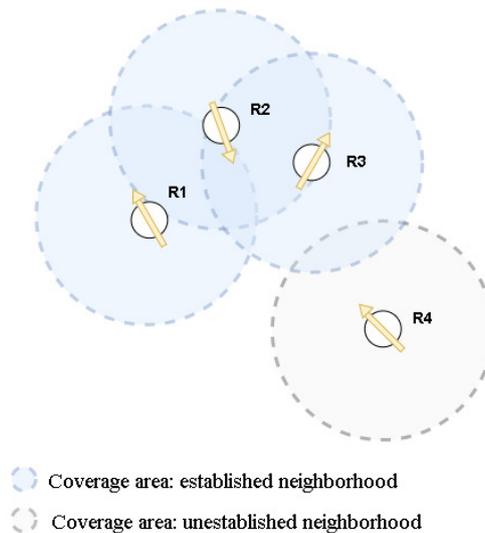
5.3.3 Local communication

In this thesis, the local communication establishes the message exchange only among the neighbors, i.e., robots out of the sensors coverage area can't communicate straight (see Figure

62). The sensor coverage area incorporates a circle with a radius equal to or smaller than the maximum range of the sensors embedded into the robot. These sensors provide distance, angle and neighbor robot identification (ID) information.

Figure 62 shows the communication between four robots, in which three share the same neighborhood (shaded area in blue). The $R1$ robot communicates with $R2$, which communicates with $R3$. Thus, $R1$, $R2$ and $R3$ robots form a unique neighborhood where the $R4$ robot does not belong.

Figura 62 – Sensors coverage area, forming a local communication among the R1, R2 and R3 robots



Fonte: Own authorship

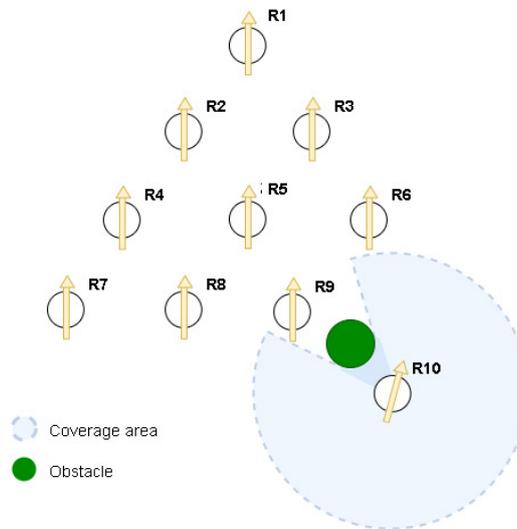
Navigation in formation with obstacles increases the chance of communication loss and affects the sensing between robots. Obstacle navigation may require diversion maneuvers that affect swarm performance during task execution. The obstacle avoidance maneuvers performed by the robot Son, $R10$, cause it to leave the coverage area of its robot Father, $R6$, besides losing the sensing ability of its robot Friend $R9$, as seen in Figure 63.

5.3.4 Rigid and semi-rigid formation

In this work, we adopted a triangular formation composed of ten e-pucks robots, as can see in Figure 64. This shape is the same designed by Souza-Jr (SILVA-JR; NEDJAH, 2017) since we intend to compare the Double-Wave Swarm and Wave Swarm.

During an avoiding obstacles maneuver, it can occur a change in the swarm of robots shape implying in new form so-called semi-rigid formation. The semi-rigid shape is a condition

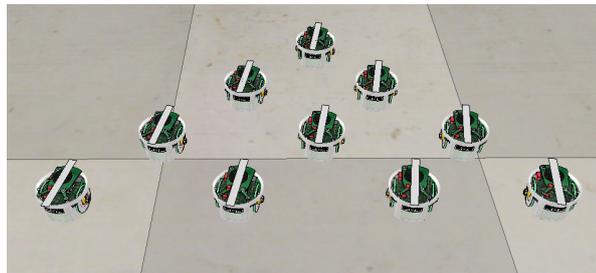
Figura 63 – Illustration of a complete loss of communication between robots R10, R6 and R9



Fonte: Own authorship

arising from losing one or more robots from the swarm due to communication failures.

Figura 64 – Triangular formation composed of ten e-pucks robots



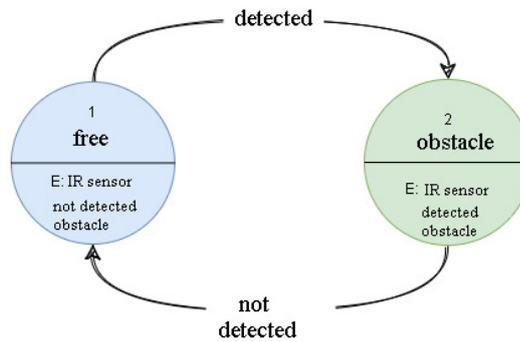
Fonte: Own authorship

5.3.5 Formation control

In this work, robots' routes fall two states possible into: 1- Free and 2 - Obstacles (see Figure 65). The Free state implies in the lack of obstacles within of robot detection area - dashed circle in Figure 66(a) and 66(b). Thus, a free state robot can run its formation control actions without diverting the route.

The Obstacle state suggests the presence of obstacles within the robot detection area. Thus, an Obstacles state robot changes the wheels' speed carrying an obstacle diversion. During such maneuver, the robot can lose communication with its Father robot (Figure 66(c) and 66(d)). As a result, the swarm formation switches from rigid to semi-rigid, i.e., the group loses some robots during the navigation task.

Figura 65 – State Machine - navigation task



Fonte: Own authorship

The Son robot achieves the distance, angle and orientation given by its position within the swarm through a proportional controller. The controller acts at the wheels' speed, as can observe in the Algorithm 10. The wheels can act one by one, providing a linear and / or angular displacement of the robot. The set-point variables for distance and angle are respectively *setDist* and *setAng*, both referenced to the Father robot. The output variables are the angular speed of the left and right wheels.

Algorithm 10 Control Algorithm: *controlPose()*

```

requer setDist, setAng, currentDist, currentAng;
inserir sLeft, sRight;
0: errorAng = normalize(setAng - currentAng)
0: errorDist = setDist - currentDist;
0: enquanto (errorAng>tolerAng) or (errorDist>tolerDist) faça
0:   sLeft = sLeft +  $K_D * errorDist + K_A * errorAng$ ;
0:   sRight = sRight +  $K_D * errorDist - K_A * errorAng$ ;
0: finaliza enquanto=0
  
```

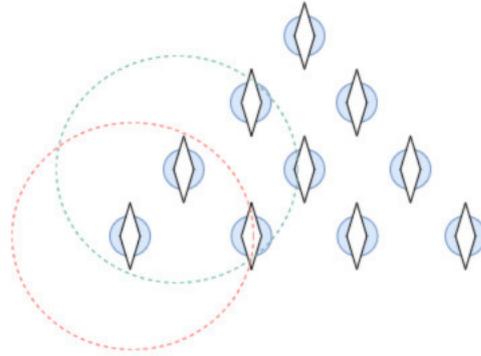
5.3.6 Obstacles detection

A robot classifies an obstacle as static or dynamic through communication between robots. A robot establishes communication from a message that has its identification, ID. Then, the robot that receives the message responds through another message with its identification.

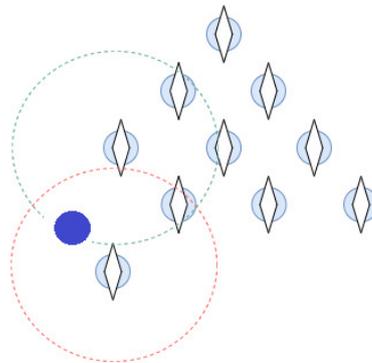
All robots have an ID for various purposes within the robot swarm. For instance, the ID defines which robot has the movement priority when a robot rendezvous with other robots during navigation. We ruled the robot with movement priority to have the lowest ID among the robots with the chance of collision.

Robots detect dynamic and static obstacles in their Green and Gray Area, with a diameter

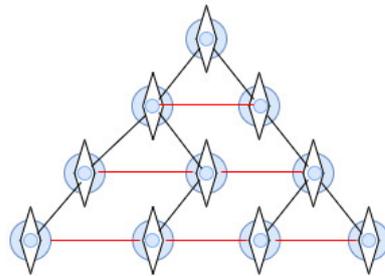
Figura 66 – Robot swarm performing navigation task



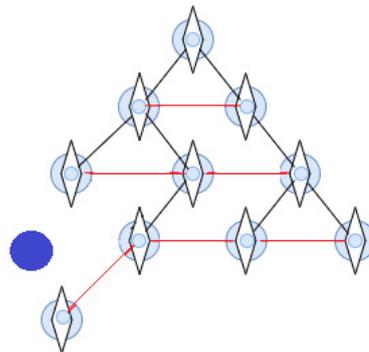
(a) Free obstacles.



(b) Obstacle located in the detection area of one of the swarm robots.



(c) Graph representation of the figure (a).



(d) Graph representation of the figure (b).

Fonte: Own authorship

equal to Φ_{green} and Φ_{gray} .

When an obstacle is in the Green Area, the robot starts an attempt at communication through the message sending containing your ID. In case of a dynamic obstacle, the robot receives a message containing another ID. At this moment, the robot having a higher priority ID starts the obstacle avoidance maneuver, and the other one stays still until the end of the maneuver. If the obstacle is a static one, the robot does not receive any message and keeps on navigating.

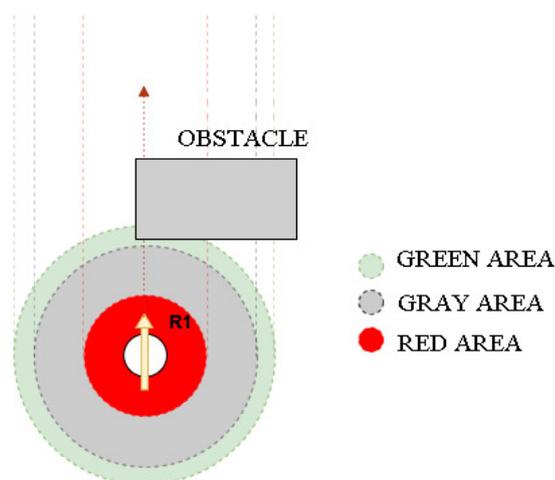
When an obstacle is detected in the Gray Area, the robot starts the obstacle avoidance maneuver, considering the obstacle as static.

The Red Area with a diameter equal to Φ_{red} is an emergency area. When an obstacle is in Red Area, the robot immediately stops and starts a communication attempt. For dynamic obstacles, the robot with priority ID begins the obstacle avoidance maneuver, and the other stays still until the end of the maneuver. In case of a static obstacle, the robot starts the obstacle avoidance maneuver. The Red Area is a safe area that assures the robot's physical integrity when a failure occurs in the obstacles deviation control algorithm and communication, as seen in Figure 67.

The relation between the diameters area is:

$$\Phi_{red} < \Phi_{gray} < \Phi_{green} \quad (19)$$

Figura 67 – Detection areas surrounding the robots



Fonte: Own authorship

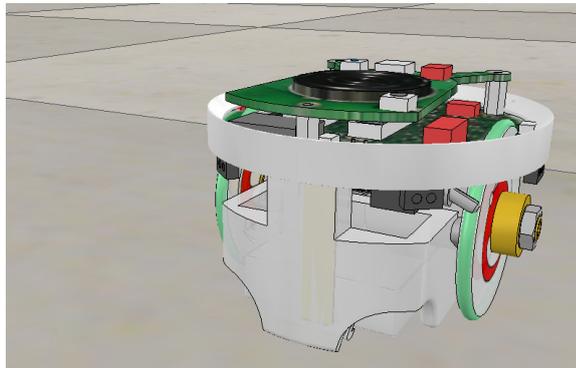
5.4 SIMULATED EXPERIMENTS

In this work, we carry out some experiments to prove the efficiency of the Double-Wave Swarm for alignment and formation navigation tasks. The Double-Wave Swarm and the Wave Swarm (SILVA-JR; NEDJAH, 2017) are compared during the alignment and navigation task. We essayed different scenarios running the alignment task (experiments A) and navigation task (experiments B and C).

Experiments A, B and C had different stopping criteria. The stopping criterion for experiments A is when all robots had the same orientation, meaning that all robots are aligned. In experiments B and C, the stopping criterion is when the Leader robot arrives at the end point of the priority established route.

The experiments are carried out into CoppeliaSim PRO EDU (version 3.5.0 - rev.4) simulator. The developed code is implemented through scripts in LUA language, following the pattern defined by the simulated e-puck developers (see Figure 68).

Figura 68 – Simulator CoppeliaSim (V-REP): e-puck

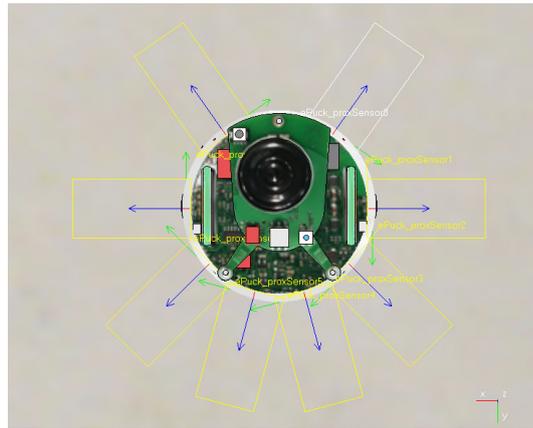


Fonte: Own authorship

The e-puck robots store the information received through the propagation message (Double-Wave Swarm and Wave Swarm) and use it in formation control during navigation. They replace the information after executing the control action and receiving a new message. Thus, the e-puck memory (RAM: 8 KB and Flash: 144 KB) is enough for this operation (MONDADA *et al.*, 2009).

The e-puck is a differential robot with eight infrared (IR) sensors placed around it. These IRs are used to measure the proximity of obstacles, as illustrated in Figure 69. The visual interaction with the user is possible through eight light-emitting diodes (LED) that surround the e-puck.

Figura 69 – Simulator CoppeliaSim (V-REP): e-puck with eight IR sensors



Fonte: Own authorship

We used a notebook with Intel Core Processor i5-2430M 2.4 GHz, 8GB of RAM memory and Microsoft Windows 10 Pro operation system.

5.4.1 Experiments A: alignment task

Experiments A goal is to assess the efficiency of the Double-Wave Swarm and to compare its performance with Wave Swarm in the execution of alignment task. We carried out two experiments: A1 (Wave Swarm) and A2 (Double-Wave Swarm).

In experiments A, the initial orientation of the robots followed a distribution function of uniform probability, ranging from 0° to 359° , as shown the Figure 70(a).

The lead robot starts and ends the alignment task after the alignment of all robots of the type of Son with relation to their Father (Figure 70(b)).

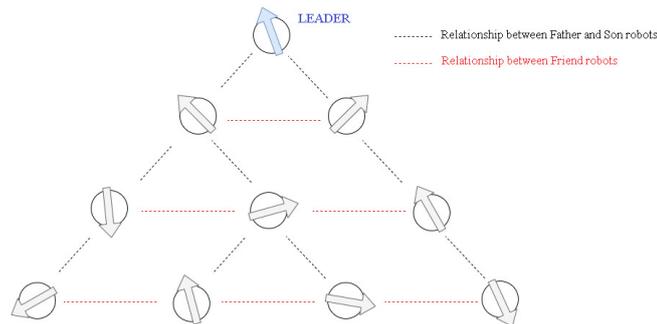
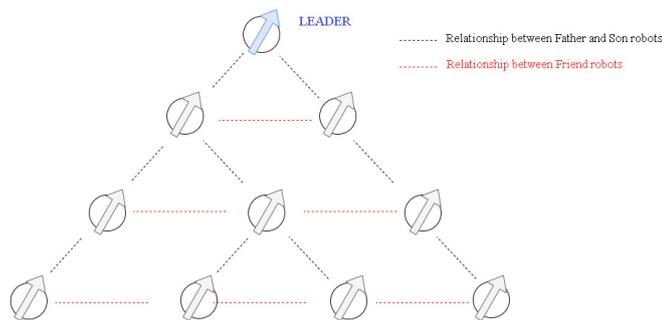
We carried out 30 simulations for each experiment. Moreover, we established the relation between Father robot and Son robots, and Friend robots to the experiments A1 and A2.

It is important to emphasize that the robots only change their orientation during the alignment task, while the X, Y positions remain unchanged.

5.4.2 Experiments B: navigation task - Wave Swarm

Experiments B apply the Algorithm 9 for navigation and the Wave Swarm algorithm for messages exchange.

The scenarios 1, 2 and 3 (Figure 71(a), 71(b) and 71(c)) simulate the free navigation and with obstacles. These experiments are labeled B1, B2 and B3.

Figura 70 – Illustration of the alignment task**(a) Initial orientation (unaligned robots)****(b) End of the simulation (aligned robots)****Fonte: Own authorship**

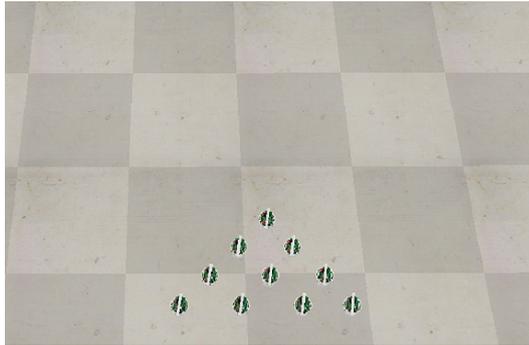
The obstacles are placed within four priority established areas. We carried out 30 simulations for each experiment, matching the four obstacles in their respective area. In each simulation, the position of the obstacles assumes a minimum distance between them equal to 5 times the diameter of the robots.

5.4.3 Experiments C: navigation task - Double-Wave Swarm

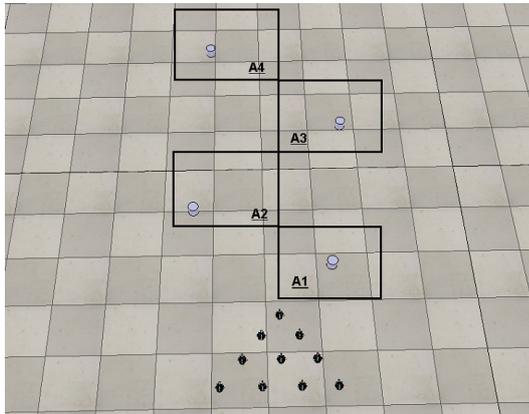
In this experiment, the swarm leader has the same goal and route from experiment B. We used the scenarios 1, 2 and 3 (Figure 71(a), 71(b) and 71(c)) to simulate the free navigation and with obstacles. These experiments are labeled C1, C2, and C3. We carried out 30 simulations for each experiment using the same obstacle positions from experiment B.

Experiments C apply the Algorithm 9 for navigation and the Double-Wave Swarm algorithm for communication between the robots (line 7).

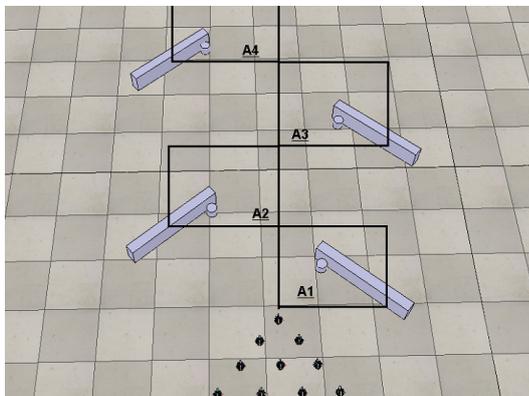
Figura 71 – Scenes from the proposed experiments



(a) Scene 1



(b) Scene 2



(c) Scene 3

Fonte: Own authorship

5.4.4 Metrics

The proposed approach uses the concept of message propagation to information exchange and to support the execution of tasks. Tasks such as alignment and formation navigation are complex and may run sequentially. However, the message propagation can demand a more significant message number exchanged proportionally to the number of robots in the swarm,

increasing the task processing and execution time.

Therefore, the message number exchanged, N_{msg} , and the simulation time, t_s , are metrics used to evaluate the processing spent on the execution of alignment task - experiments A1 and A2.

The number of robots can change during navigation with obstacles due to communication failures between robots during the diversion obstacles maneuvers. Hence, the number of robots (β) that completed the navigation, following the leader path, is computed - experiments B1, B2, B3, C1, C2 and C3.

Descriptive statistics such as mean, median, standard deviation, interquartile range (IQR), skewness, and kurtosis allow us to analyze experiments' results.

We conducted data normality tests using the Shapiro-Wilk test (SHAPIRO; WILK, 1965) ($p \leq 0.05$). Also, we performed the Bootstrap test (EFRON; TIBSHIRANI, 1994) (non-parametric) to assess the difference between the averages with a confidence interval of 95% Bootstrap with 10^6 Bootstrap samples.

5.5 RESULTS AND DISCUSSION

In this section, we show the simulation results and discuss the data gotten. The previously established metrics are used to validate Double-Wave Swarm: N_{msg} (experiments A1 and A2), t_s (experiments A1 and A2), e , β (experiments B1, B2, B3, C1, C2 and C3).

We propose to show Double-Wave Swarm is a tool to message exchange, and, also, for the sequential execution of tasks, forming complex tasks, as the alignment and the free navigation and with obstacles. We intend to highlight the robustness and efficiency of the Double-Wave Swarm by comparing it to the Wave Swarm.

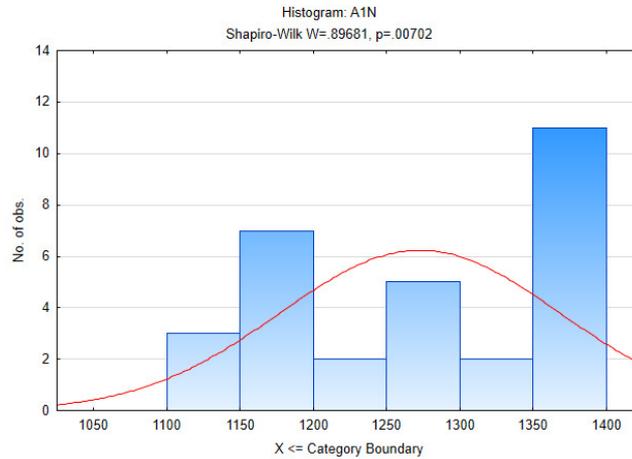
5.5.1 Experiments A1 and A2 - alignment task

The Double-Wave Swarm algorithm efficiently accomplished the alignment task, completing this task in all simulations carried out.

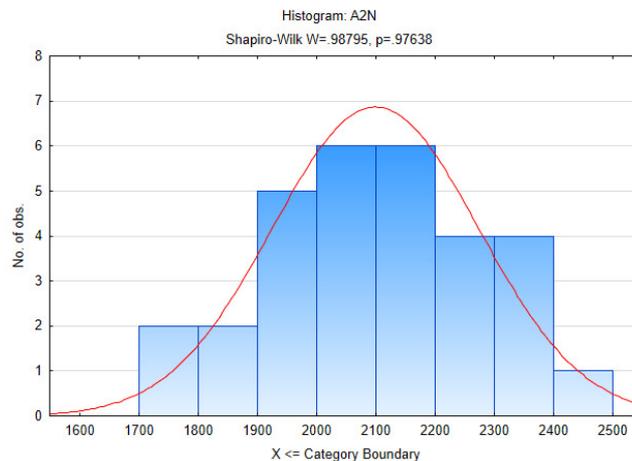
The histograms, Figure 72(a) and 72(b), display the number of the exchanged message, N_{msg} , during experiments A1 and A2. These histograms correspond to a distribution function of the platykurtic type, implying a flatness if compared to the Gaussian function, i.e., a data dispersion around the average. Moreover, the histograms have a positive asymmetry, in both

cases, considered a weak asymmetry, showing an average value close to the median and the mode.

Figura 72 – Histogram of the number of message exchanges for the alignment task.



(a) Experiment A1: average equal to 1272.2, standard deviation equal to 94.331, medians to 1265.2, kurtosis equal to 1.55, asymmetry equal to -0.080, and Shapiro-Wilk test with p-value equal to 0.007



(b) Experiment A2: average equal to 2099.2, standard deviation equal to 171.340, medians to 2105.4, kurtosis equal to -0.46, asymmetry equal to -0.023, and Shapiro-Wilk test with p-value equal to 0.976

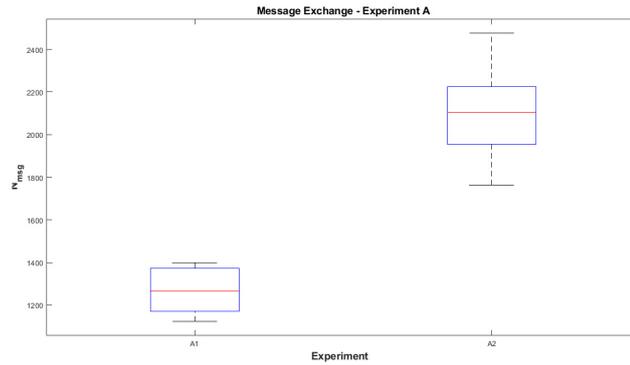
Fonte: Own authorship

The results obtained from experiments A1 and A2 for the value N_{msg} show greater variability for A2, as seen in Figure 73.

The Shapiro–Wilk test presented a p value of 0.007 and 0.976 to the experiments A1 and A2, Figure 72. These p values point out non-normality of the sampled data from experiment A1 and data normality for experiment A2.

The average and median values of N_{msg} , experiments A1 and A2, presented difference, as illustrated in Figures 74 and 75. Moreover, the average and median values of N_{msg} for

Figura 73 – Boxplot of the number of message exchanges - Experiments A1 and A2: average equal to 1272.2 and 2099.2, median equal to 1265.2 and 2105.4, and IQR equal to 196 and 261

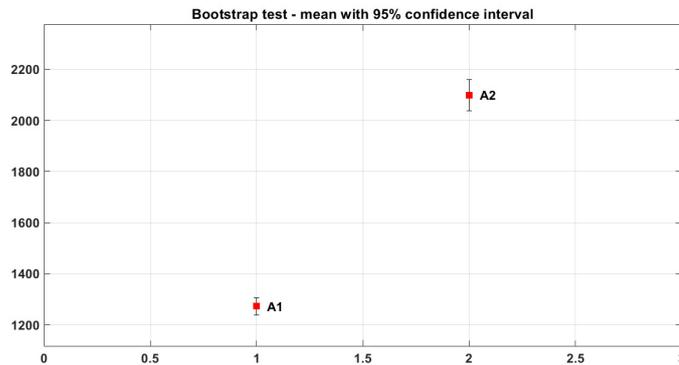


Fonte: Own authorship

experiment A2 have a superior percentage if compared to experiment A1 (around 65% and 66%).

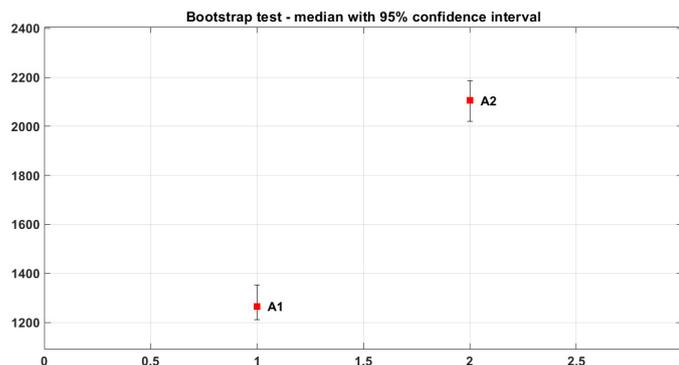
This increase of N_{msg} is expected due to the increase in the number of communication links between robots using the DWS algorithm. As mentioned in Section 5.2, DWS and WS have connectivity equal to 2 and 1.

Figura 74 – Estimate of the average of the message number exchanged, N_{msg} , with a 95% bootstrap confidence interval: experiments A1 and A2



Fonte: Own authorship

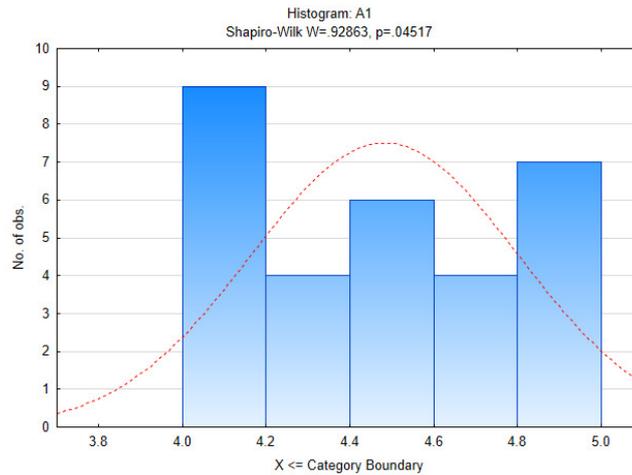
Figura 75 – Estimate of the median of the message number exchanged, N_{msg} , with a 95% bootstrap confidence interval: experiments A1 and A2



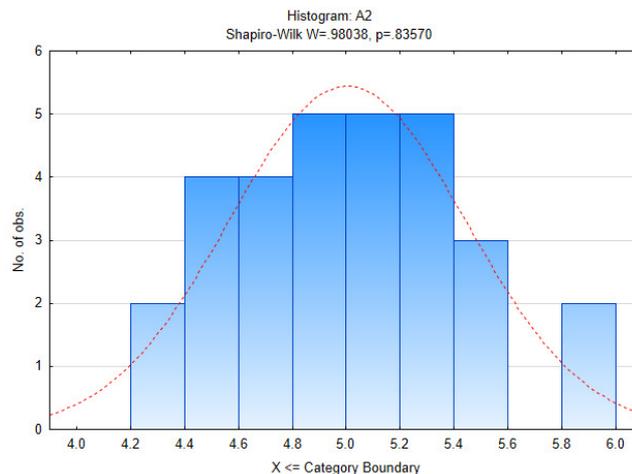
Fonte: Own authorship

The histograms, Figure 76(a) and 76(b), display the simulation time, t_s , during experiments A1 and A2. The shown distribution function is a platykurtic type, implying a flatness if compared to the Gaussian function, i.e., a dispersion of the data around the average. In addition, the histograms have a positive asymmetry, in both cases, considered a weak asymmetry, showing an average value close to the median and the mode.

Figure 76 – Simulation time histogram for the alignment task



- (a) Experiment A1: average equal to 4.483, standard deviation equal to 0.313, and medians to 4.494, kurtosis equal to -1.33, asymmetry equal to 0.0132, and Shapiro-Wilk test with p-value equal to 0.0452**



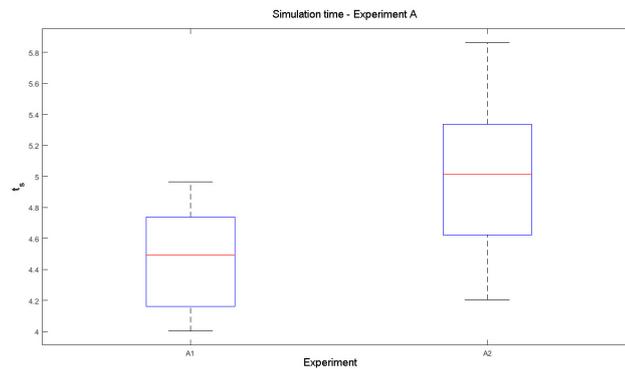
- (b) Experiment A2: average equal to 5.004, standard deviation equal to 0.433, and medians to 5.017, kurtosis equal to -0.635, asymmetry equal to 0.080, and Shapiro-Wilk test with p-value equal to 0.836**

Fonte: Own authorship

The results obtained from experiments A1 and A2 for the value t_s show greater variability for A2, as seen in Figure 77.

The Shapiro–Wilk test presented a p value of 0.0452 and 0.836 to the experiments A1 and A2, Figure 76. These p values point out non-normality of the sampled data from experiment

Figura 77 – Simulation time boxplot - Experiments A1 and A2: median equal to 4.494 and 5.017 and IQR equal to 0.562 and 0.689

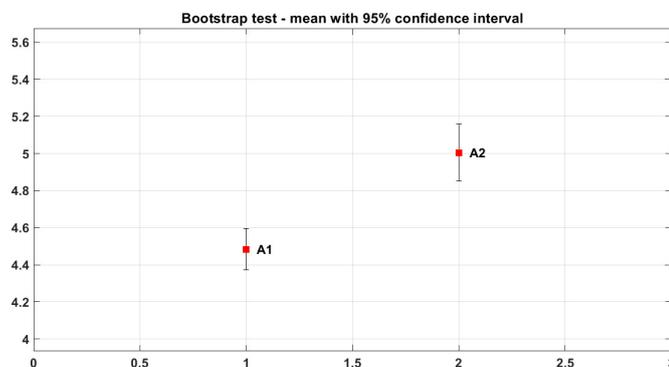


Fonte: Own authorship

A1 and data normality for experiment A2.

The average and median values of t_s , experiments A1 and A2, presented difference, as illustrated in Figures 78 and 79. Moreover, the average and median values of t_s , experiment A2, have a superior percentage if compared to experiment A1 (around 11.607% and 11.638%). This increase is due to the increment in the number of messages needed to exchange information between Friends robots, as previously discussed.

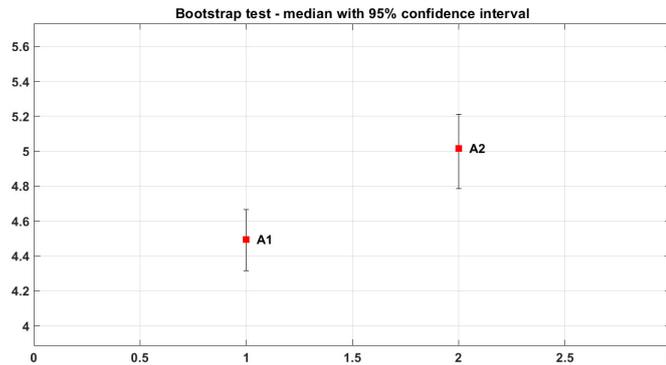
Figura 78 – Estimate of the average of the simulation time, t_s , with a 95% bootstrap confidence interval: experiments A1 and A2



Fonte: Own authorship

In summary, experiments A1 and A2 results show a more significant demand for message exchange and simulation time when using the approach proposed in this work, increasing the computational cost compared to the former approach Wave Swarm (SILVA-JR; NEDJAH, 2017). The results establish an increase in the number of messages exchanged leads to an increase in the simulation time. The way Double-Wave Swarm propagates the messages optimizes the time for task execution. Double-Wave Swarm presented an increase in computational cost to ensure increased connectivity between swarm robots.

Figura 79 – Estimate of the median of the simulation time, t_s , with a 95% bootstrap confidence interval: experiments A1 and A2



Fonte: Own authorship

5.5.2 Experiment B2 and B3 - navigation task

The number of robots that completed the navigation, β , can range between 0 and 10 robots in experiments B2 and B3. The histograms of figure 80(a) and 80(b) allude to the datasets provisions by experiments B2 and B3.

The histogram of the experiment B2 and B3, Figure 80, shows a distribution function of the platykurtic type, implying a flatness function if compared to the Gaussian function, i.e., a dispersion of the data around the average. The histograms B2 and B3 have a moderate positive asymmetry, meaning an average greater than the median and mode.

The results obtained from experiments B2 and B3 for the value β show greater variability for B3, as seen in Figure 81.

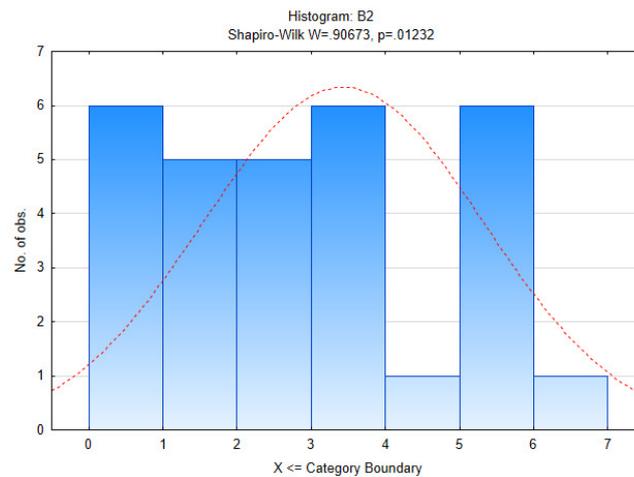
The Shapiro–Wilk test presented a p value of 0.012 and 0.106 to the experiments B2 and B3, respectively. These p values point out non-normality of the sampled data from experiment B2 and data normality for experiment B3.

The average and median values of beta in the experiments B2 and B3 not presented difference, as illustrated in Figures 82 and 83. Thus, the approach by Silva-Jr. et al. (SILVA-JR; NEDJAH, 2017) behaved similarly in the experiments B2 and B3, considering the average values of β .

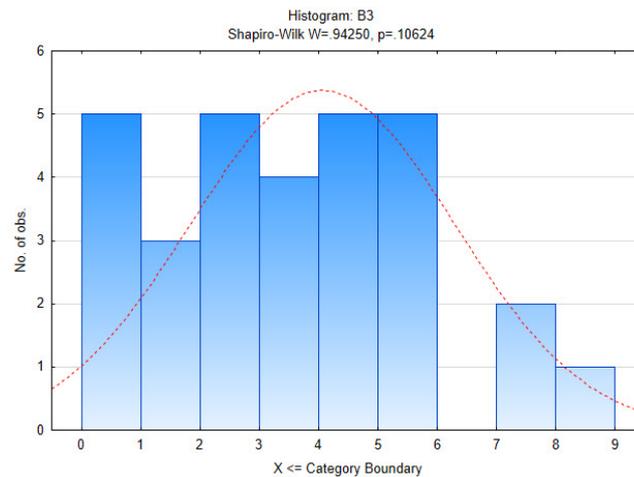
5.5.3 Experiment C2 and C3 - navigation task

The number of robots that completed the navigation, β , can range between 0 and 10 robots in experiments C2 and C3. The histograms of figure 84(a) and 84(b) concern to the

Figura 80 – Histogram of the number of robot that completed the navigation, β



(a) Experiment B2: average equal to 3.433, standard deviation equal to 1.856, medians to 3, kurtosis equal to -1.13, asymmetry equal to 0.276, and Shapiro-Wilk test with p-value equal to 0.012.



(b) Experiment B3: average equal to 4.067, standard deviation equal to 2.190, medians to 4, kurtosis equal to -0.497, asymmetry equal to 0.352, and Shapiro-Wilk test with p-value equal to 0.106.

Fonte: Own authorship

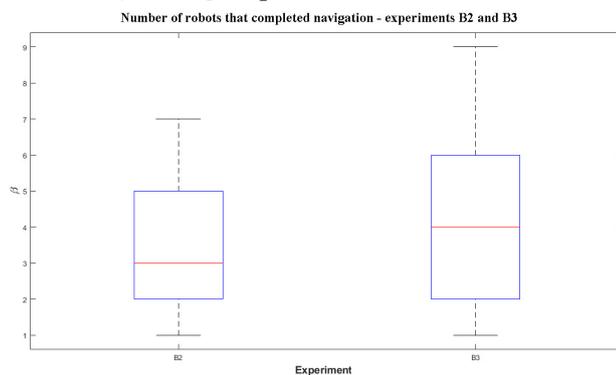
datasets provisions by experiments C2 and C3.

The experiment C2 and C3 histogram, Figure 84(a) and 84(b), shows a platykurtic type distribution function, implying a flatness if compared to the Gaussian function, i.e., a data dispersion around the average.

The histograms C2 and C3 have a moderate negative asymmetry, meaning an average lower than the median and mode. The experiments C2 and C3 results got show greater variability for C3, as seen in Figure 85.

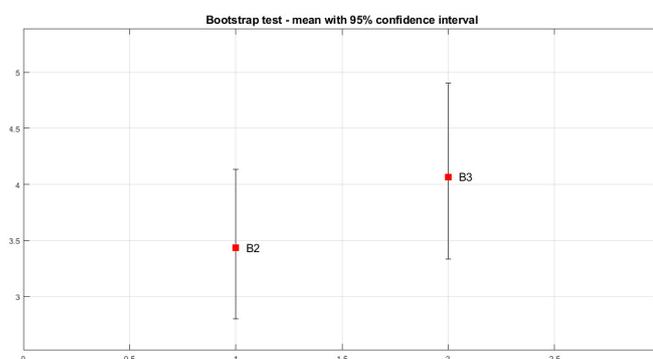
The Shapiro–Wilk test has presented a p value of 0.031 and 0.027 to experiments C2 and C3, respectively. These p values point out a non-normality of the data sampled.

Figura 81 – Boxplot of the number of robot that completed the navigation, β - Experiments B2 and B3: median equal to 3 and 4, and IQR equal to 2.75 and 3.50



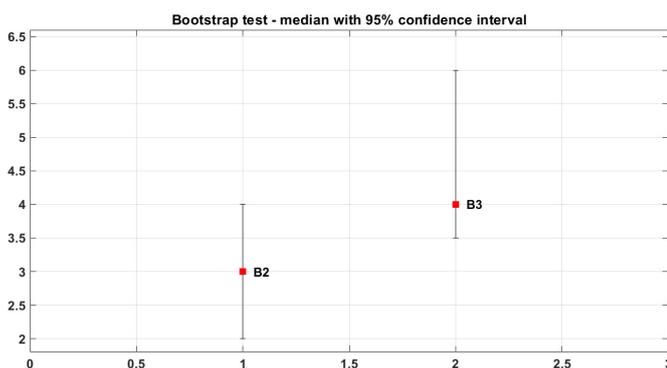
Fonte: Own authorship

Figura 82 – Estimate of the average number of robots completing the navigation, β , with a 95% bootstrap confidence interval: experiment B2 and B3



Fonte: Own authorship

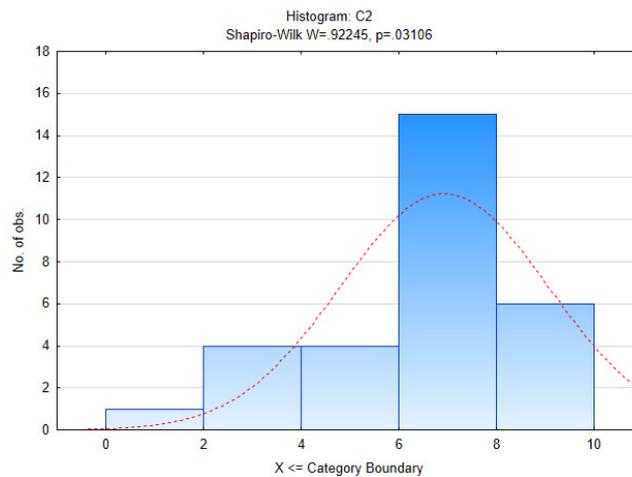
Figura 83 – Estimate of the median number of robots completing the navigation, β , with a 95% bootstrap confidence interval: experiment B2 and B3



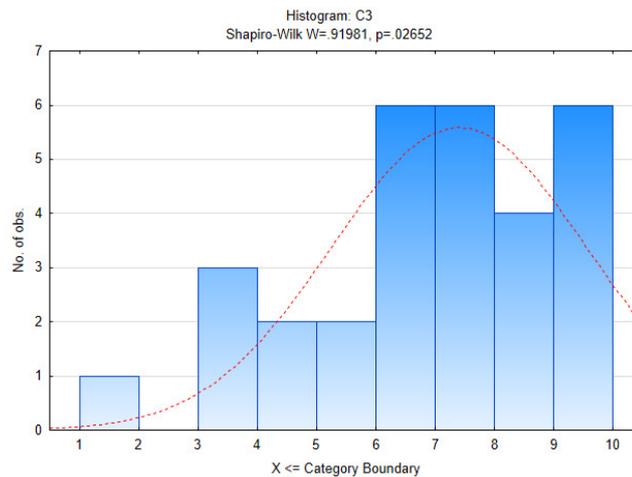
Fonte: Own authorship

The average and median β values from experiments C2 and C3 did not present a difference, as illustrated the Figure 87. Thus, the proposed approach behaved similarly in experiments C2 and C3, regarding the average β values.

Figura 84 – Histogram of the number of robot that completed the navigation, β



(a) Experiment C2: averages equal to 6.933, standard deviations equal to 2.097, medians to 7, kurtosis equal to 0.861, asymmetry equal to -0.890, and Shapiro-Wilk test with p-value equal to 0.031.



(b) Experiment C3: averages equal to 7.4, standard deviations equal to 2.107, medians to 8, kurtosis equal to -0.067, asymmetry equal to -0.697, and Shapiro-Wilk test with p-value equal to 0.027.

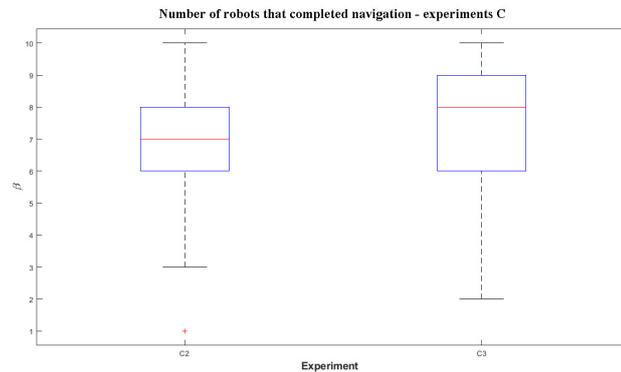
Fonte: Own authorship

5.5.4 A comparison between experiments B and C.

Obstacle-free navigation experiments, B1 and C1, did present an β average equal to 10 robots, showing the efficiency of both approaches in accomplishing navigation tasks. However, a decrease of β had occurred in the presence of some obstacles randomly posed, running Double-Wave Swarm and Wave Swarm, Figure 88. These data show that diversion maneuvers can decrease communication capability between neighboring robots and imply the non-conclusion of formation navigation tasks.

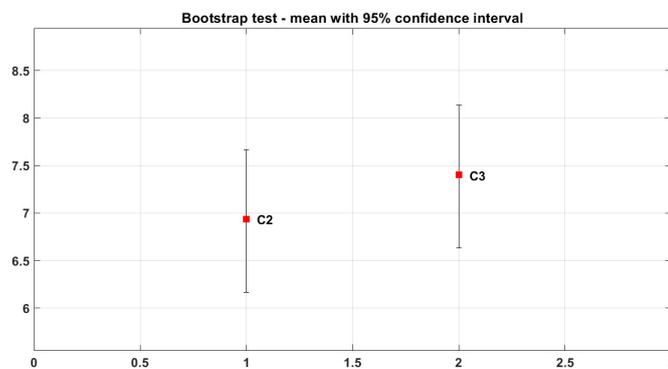
We merged the data from experiments B2 and B3, building a new and single data-set,

Figura 85 – Boxplot of the number of robot that completed the navigation, β - Experiments C2 and C3: median equal to 7 and 8, and IQR equal to 2 and 2.75



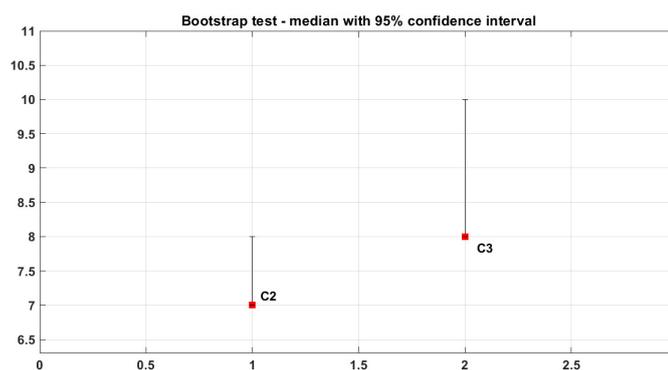
Fonte: Own authorship

Figura 86 – Estimate of the average of the number of robot that completed the navigation, β , with a 95% bootstrap confidence interval: experiments C2 and C3



Fonte: Own authorship

Figura 87 – Estimate of the median of the number of robot that completed the navigation, β , with a 95% bootstrap confidence interval: experiments C2 and C3



Fonte: Own authorship

so-called experiment B. Similarly, we merged the data from experiments C2 and C3, building a data-set labeled experiment C. Figures 88(a) and 88(b) put forward the values of β in histogram plots for 60 carried out simulations.

The number of robots completing navigation, β , can range between 0 and 10 robots in

experiments B and C. Figure 88(a) and 88(b) show histograms concerning the data-sets from experiments B and C.

The experiment B and C histogram, Figure 88(a) and 88(b), introduce a platykurtic type distribution function, implying a flatness if compared to the Gaussian function, i.e., a dispersion of the data around the average. The B histogram has a moderate positive asymmetry, meaning an average greater than the median and mode.

The C histogram has a moderate negative asymmetry, meaning an average lower than the median and mode.

The experiments B and C results got show greater variability for B, as seen in Figure 89.

The Shapiro–Wilk test presented a p value of 0.003 and 0.001 to the experiments B and C, respectively. These p values point out a non-normality of the data sampled.

The average and median values of β from experiments B and C are different, as illustrated the Figure 91. Moreover, the experiment C average and median values of β have a superior percentage than experiment B (around 91% and 75%).

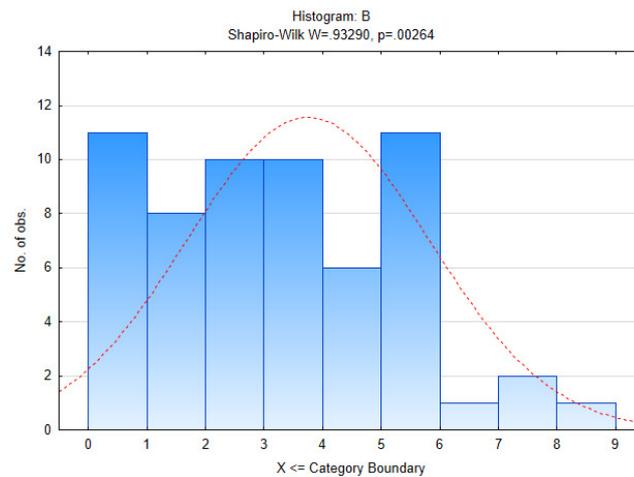
The simulations of experiments C introduced results with β equal to 10 robots in 15% of the simulations. The results of the simulations of B experiments showed the highest beta value equal to 9 robots in 1.67% of the total simulations.

In addition, the simulation data displayed that 70% or more of the robots, i.e., $\beta \geq 7$, completed the task in 71.67% and 6.67% of the simulations for experiments C and B. We can see these data and others in Figure 92.

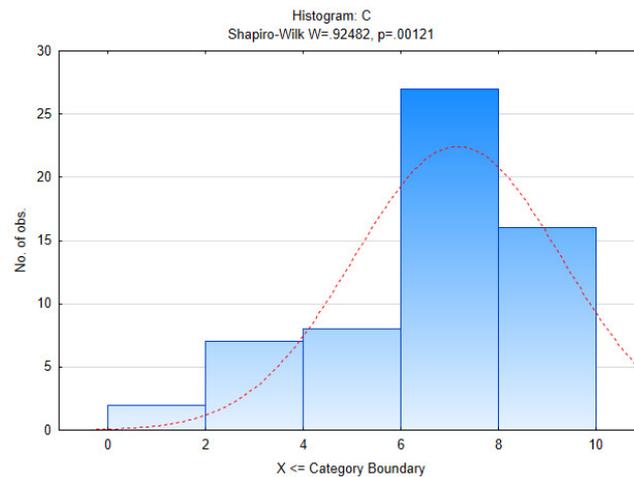
In summary, the experiments for the alignment task resulted in an average and median increase in the number of messages exchanged (65% and 66%) and simulation time (11.607% and 11.638%) using the proposed approach. The experiments for the formation navigation task without obstacles resulted in navigation without communication failures. Therefore, all robots completed the navigation. The experiments for the formation navigation task with obstacles resulted in an average and median increase in the number of robots that completed the navigation (91% and 75%) when comparing DWS to WS.

Thereby, we can conclude that the Double-Wave Swarm and Wave Swarm are valid approaches in the sequential execution of subtasks, aiding in complex tasks, such as obstacle-free and navigation. Indeed, the navigation with obstacles presented communication failures between robots in the task's execution, using both approaches. However, the results display the robot

Figura 88 – Histogram of the number of robot that completed the navigation, β



(a) Experiment B: averages equal to 3.75, standard deviations equal to 2.072, medians to 4, kurtosis equal to -0.585, asymmetry equal to 0.382, and Shapiro-Wilk test with p-value equal to 0.003

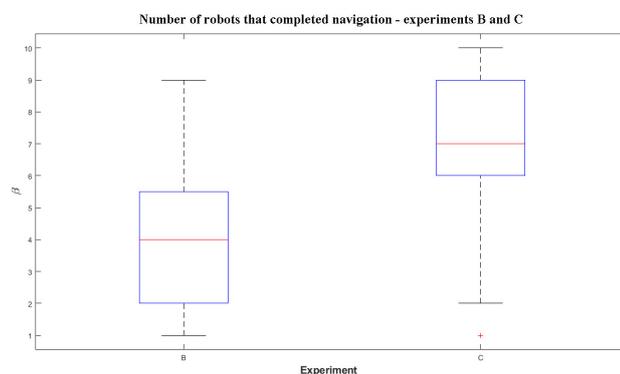


(b) Experiment C: averages equal to 7.167, standard deviations equal to 2.133, medians to 7, kurtosis equal to 0.292, asymmetry equal to -0.756, and Shapiro-Wilk test with p-value equal to 0.001

Fonte: Own authorship

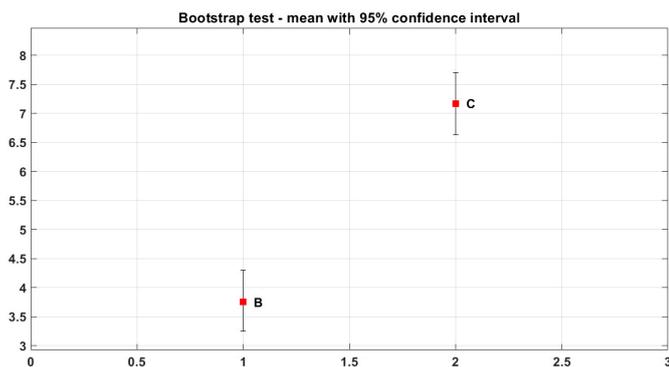
swarm fulfilled the navigation task despite the failures, ensuring robustness and an efficient performance using Double-Wave Swarm. Robustness and superior performance of DWS is achieved due to increased network connectivity when compared to Wave Swarm.

Figura 89 – Boxplot of the number of robot that completed the navigation, β - Experiments B and C: median equal to 4 and 7, and IQR equal to 3.5 and 3



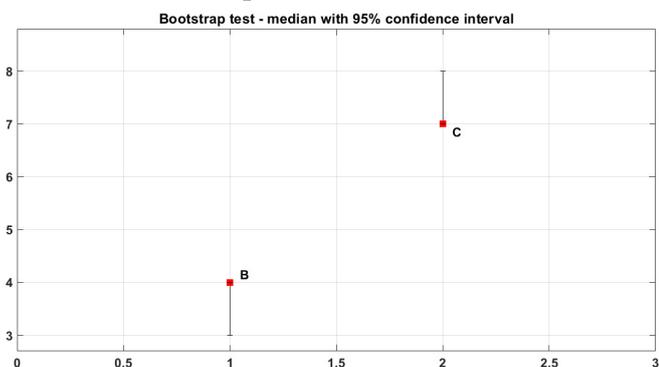
Fonte: Own authorship

Figura 90 – Estimate of the average of the number of robot that completed the navigation, β , with a 95% bootstrap confidence interval: Experiment B and C



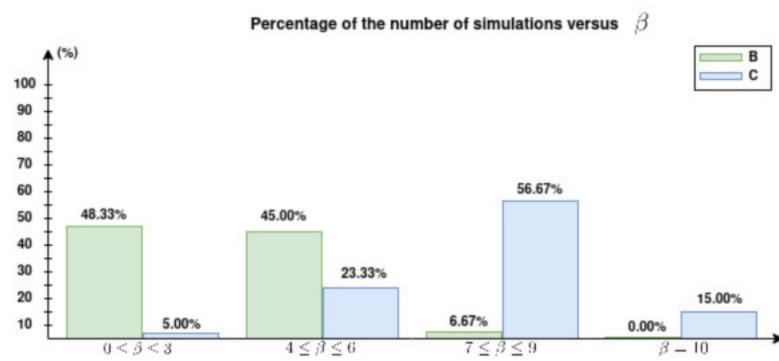
Fonte: Own authorship

Figura 91 – Estimate of the median of the number of robots that completed the navigation, β , with a 95% bootstrap confidence interval: Experiment B and C



Fonte: Own authorship

Figura 92 – Percentage of number of simulations versus β - experiments B (Wave Swarm) and C (Double-Wave Swarm)



Fonte: Own authorship

6 CONCLUSION

This work presents a strategy based on Wave Swarm for the formation task inspired by the Traveling Salesman Problem solved with reinforcement learning. This strategy uses communication through a wave propagation algorithm to exchange messages between robots and thus manage the formation task. The message propagation occurs between pairs of robots, linked as father and son. Information sharing supports robot localization and orientation within the group of robots, helping reach the goal. We performed experiments with different sizes of robot groups to execute the formation task with different designs to validate the proposed strategy. Such experiments used e-puck2 robots in an unknown environment, i.e., an unknown neighborhood without landmarks and/or mapping, built with the robot simulator CoppeliaSim (V-REP). The strategy proposed in this work provides a relationship between the Father and Son robots that minimize the metrics mentioned above, solving the TSP via the reinforcement learning.

Overall, the proposed strategy showed superior efficiency over WS for the formation task and proved to be an alternative approach for the formation task of multi-robots.

This work also presents the Double-Wave Swarm approach for communication in a swarm robotics application. Double-Wave Swarm uses a wave propagation algorithm to allow message exchange and subtask coordination. The sharing of information through the exchange of messages helps in the location and orientation of the robots within the swarm. DWS faces the three challenges for communication in swarm robotics: high number of messages exchanged, connectivity maintenance, and dynamic changes in the network topology.

Despite using wave propagation for local communication, the WS algorithm frequently fails to maintain connectivity while performing complex cooperative tasks, such as formation navigation. Such fails are less frequent for DWS. DWS improves WS by increasing swarm connectivity while maintaining similar complexity. Indeed, the DWS increased connectivity results in a reduction in miscommunications during formation navigation with obstacles when compared to WS. Moreover, the number of messages exchanged for each message transmitted by the lead robot is approximately twice using DWS instead of WS as the number of robots in the swarm grows. However, the total transmission time for this to happen is approximately the same for DWS and WS. Thus, message complexity and time complexity are equal to $O(N)$ for DWS and WS. The Double-Wave Swarm was superior during the navigation task, with obstacles

in different scenarios compared to the Wave Swarm.

The advantage of DWS is increased connectivity. However, an inherited disadvantage of Wave Swarm is the dependency on an Origin robot for successful execution.

We performed experiments with to execute the group navigation task with obstacles to validate the proposed strategy. Such experiments used e-puck robots in an unknown environment, i.e., an unknown neighborhood without landmarks and/or mapping, built with the robot simulator CoppeliaSim (V-REP).

In summary, the Double-Wave Swarm proved robust and efficient in performing alignment tasks, navigation in obstacle-free scenarios, and navigation in environments with obstacles of different shapes.

It is important to recognize the contributions of the proposed improvement to Wave Swarm and the contributions of Double-Wave Swarm. However, there are challenges and limitations that require further investigation and development.

This doctoral work allowed us to identify critical areas that require attention. Therefore, in the next section, Future Work, we will discuss the strategies we plan to adopt to address these limitations.

Understanding and proposing solutions to limitations will contribute to advancement in the field of swarm robots and their applications.

6.1 FUTURE WORK

We observed the vulnerability of DWS and WS to the failure of the Origin robot in a robotic swarm. However, we intend to implement a backup mechanism; the Backup Designated Router (BDR) (LIGANG *et al.*, 2023), as a solution to reduce this dependency.

Furthermore, we will address crucial issues, such as the influence of the Origin robot's position on wave propagation and task execution. We will also explore Origin choice strategies that optimize these processes.

The presence of obstacles during navigation can increase the chances of communication failures, making swarm mapping essential. Our next steps will involve exploring efficient communication protocols to facilitate this mapping.

Besides that, we will investigate the impact of noisy messages on DWS and WS performance. Finally, we plan to apply our proposed strategy to a larger-scale swarm and explore parallel processing techniques to solve the Traveling Salesman Problem (TSP).

REFERÊNCIAS

- AGRAWAL, Anubha; GHUNE, Nitish; PRAKASH, Shiv; RAMTEKE, Manojkumar. Evolutionary algorithm hybridized with local search and intelligent seeding for solving multi-objective Euclidian TSP. **Expert Systems with Applications**, v. 181, p. 115192, 2021. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417421006278>.
- ALI, Abduladhem A.; RASHID, Abdulmuttalib T.; FRASCA, Mattia; FORTUNA, Luigi. An algorithm for multi-robot collision-free navigation based on shortest distance. **Robotics and Autonomous Systems**, Elsevier BV, v. 75, p. 119–128, 2016. Disponível em: <https://doi.org/10.1016/j.robot.2015.10.010>.
- ALMEIDA, João Paulo Lima Silva De; NAKASHIMA, Renan Taizo; NEVES-JR, Flávio; ARRUDA, Lúcia Valéria Ramos De. Bio-inspired on-line path planner for cooperative exploration of unknown environment by a multi-robot system. **Robotics and Autonomous Systems**, v. 112, p. 32–48, 2019. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889018303567>.
- ALMEIDA, João Paulo Lima Silva De; NAKASHIMA, Renan Taizo; NEVES-JR., Flávio; ARRUDA, Lúcia Valéria Ramos De. A global/local path planner for multi-robot systems with uncertain robot localization. **Journal of Intelligent & Robotic Systems**, Springer Science and Business Media LLC, v. 100, p. 311–333, 2020. Disponível em: <https://doi.org/10.1007/s10846-020-01196-y>.
- ALOUACHE, Ali; WU, Qinghe. Performance comparison of consensus protocol and l-&phi approach for formation control of multiple nonholonomic wheeled mobile robots. **Journal of Mechatronics, Electrical Power, and Vehicular Technology**, Indonesian Institute of Sciences, v. 8, n. 1, p. 22–32, 2017.
- AMBAINIS, Andris; BALODIS, Kaspars; IRAIDS, Jānis; KOKAINIS, Martins; PRŪSIS, Krišjānis; VIHROVS, Jevgēnijs. Quantum Speedups for Exponential-Time Dynamic Programming Algorithms. In: _____. **Proceedings of the 2019 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)**. [s.n.], 2019. p. 1783–1793. Disponível em: <https://epubs.siam.org/doi/abs/10.1137/1.9781611975482.107>.
- ARGoS. 2012. Disponível em: <https://www.argos-sim.info/>.
- ASKARI, A.; MORTAZAVI, M.; TALEBI, H. A. UAV formation control via the virtual structure approach. **Journal of Aerospace Engineering**, American Society of Civil Engineers, v. 28, n. 1, p. 04014047, 2013.

BAI, Jing; WEN, Guoguang; SONG, Yu; RAHMANI, Ahmed; YU, Yongguang. Distributed formation control of fractional-order multi-agent systems with relative damping and communication delay. **International Journal of Control, Automation and Systems**, Springer Science and Business Media LLC, v. 15, n. 1, p. 85–94, 2016. Disponível em: <https://doi.org/10.1007/s12555-015-0132-x>.

BARA'A, A. Attea; ABBOOD, Amenah D.; HASAN, Ammar A.; PIZZUTI, Clara; AL-ANI, Mayyadah; ÖZDEMİR, Suat; AL-DABBAGH, Rawaa Dawoud. A review of heuristics and metaheuristics for community detection in complex networks: current usage, emerging development and future directions. **Swarm and Evolutionary Computation**, Elsevier, v. 63, p. 100885, 2021.

BAYINDIR, Levent. A review of swarm robotics tasks. **Neurocomputing**, Elsevier, v. 172, p. 292–321, 2016.

BAYINDIR, Levent. A review of swarm robotics tasks. **Neurocomputing**, v. 172, p. 292–321, 2016. ISSN 0925-2312. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231215010486>.

BELTA, Calin; KUMAR, Vijay. Abstraction and control for groups of robots. **IEEE Transactions on robotics**, IEEE, v. 20, n. 5, p. 865–875, 2004.

BRAEM, Bart; LATRE, Benoit; MOERMAN, Ingrid; BLONDIA, Chris; DEMEESTER, Piet. The wireless autonomous spanning tree protocol for multihop wireless body area networks. *In: IEEE. 2006 Third Annual International Conference on Mobile and Ubiquitous Systems: Networking & Services. [S.l.]*, 2006. p. 1–8.

BRAMBILLA, Manuele; FERRANTE, Eliseo; BIRATTARI, Mauro; DORIGO, Marco. Swarm robotics: a review from the swarm engineering perspective. **Swarm Intelligence**, Springer, v. 7, n. 1, p. 1–41, 2013.

CAMPO, Alexandre; GUTIÉRREZ, Álvaro; NOUYAN, Shervin; PINCIROLI, Carlo; LONGCHAMP, Valentin; GARNIER, Simon; DORIGO, Marco. Artificial pheromone for path selection by a foraging swarm of robots. **Biological cybernetics**, Springer, v. 103, n. 5, p. 339–352, 2010.

CARVALHO, Adriano G. **Influência da modelagem dos componentes de bias instabilidade dos sensores inerciais no desempenho do navegador integrado sni/gps**. 2011. Tese (Doutorado) — Instituto Militar de Engenharia, Rio de Janeiro, 2011.

CHEAH, Chien Chern; HOU, Saing Paul; SLOTINE, Jean Jacques E. Region-based shape control for a swarm of robots. **Automatica**, Elsevier, v. 45, n. 10, p. 2406–2411, 2009.

CHEN, Cheng; CHAI, Wennan; ROTH, H.; NASIR, A. Low cost imu based indoor mobile robot navigation with the assist of odometry and wi-fi using dynamic constraints. *In: Proceedings of IEEE/ION PLANS 2012*. [S.l.: s.n.], 2012. p. 1274–1279.

CHEN, Chi-Chung; CHANG, Han-Wen. Design of robots formation controllers using Evolutionary Fuzzy systems. **International Journal of Electrical Engineering**, v. 11, n. 1, p. 31–38, 2018.

CHEN, Jianing; GAUCI, Melvin; GROSS, Roderich. A strategy for transporting tall objects with a swarm of miniature mobile robots. *In: IEEE. 2013 IEEE International Conference on Robotics and Automation*. [S.l.], 2013. p. 863–869.

CHEN, Jianing; GAUCI, Melvin; PRICE, Michael J.; GROSS, Roderich. Segregation in swarms of e-puck robots based on the brazil nut effect. *In: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*. [S.l.: s.n.], 2012. p. 163–170.

CHEN, Jintao; SHI, Zongying; ZHONG, Yisheng. Robust formation control for uncertain multi-agent systems. **Journal of the Franklin Institute**, Elsevier BV, v. 356, n. 15, p. 8237–8254, 2019. Disponível em: <https://doi.org/10.1016/j.jfranklin.2019.07.014>.

CHU, Xing; PENG, Zhaoxia; WEN, Guoguang; RAHMANI, Ahmed. Distributed fixed-time formation tracking of multi-robot systems with nonholonomic constraints. **Neurocomputing**, Elsevier BV, v. 313, p. 167–174, 2018. Disponível em: <https://doi.org/10.1016/j.neucom.2018.06.044>.

CHU, Xing; PENG, Zhaoxia; WEN, Guoguang; RAHMANI, Ahmed. Distributed formation tracking of multi-robot systems with nonholonomic constraint via event-triggered approach. **Neurocomputing**, Elsevier BV, v. 275, p. 121–131, 2018. Disponível em: <https://doi.org/10.1016/j.neucom.2017.05.007>.

CHUNG, Soon-Jo; PARANJAPE, Aditya Avinash; DAMES, Philip; SHEN, Shaojie; KUMAR, Vijay. A survey on aerial swarm robotics. **IEEE Transactions on Robotics**, Institute of Electrical and Electronics Engineers (IEEE), v. 34, n. 4, p. 837–855, 2018. Disponível em: <https://doi.org/10.1109/tro.2018.2857475>.

CONSOLINI, Luca; MORBIDI, Fabio; PRATTICCHIZZO, Domenico; TOSQUES, Mario. Leader–follower formation control of nonholonomic mobile robots with input constraints. **Automatica**, Elsevier, v. 44, n. 5, p. 1343–1349, 2008.

DAI, Yanyan; KIM, YoonGu; WEE, SungGil; LEE, DongHa; LEE, SukGyu. A switching formation strategy for obstacle avoidance of a multi-robot system based on robot priority model. **ISA Transactions**, v. 56, p. 123–134, 2015. ISSN 0019-0578. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0019057814002572>.

DAI, Yanyan; LEE, Suk-Gyu. The leader-follower formation control of nonholonomic mobile robots. **International Journal of Control, Automation and Systems**, Springer, v. 10, n. 2, p. 350–361, 2012.

DIMAROGONAS, Dimos V.; EGERSTEDT, Magnus; KYRIAKOPOULOS, Kostas J. A leader-based containment control strategy for multiple unicycles. *In: IEEE. Proceedings of the 45th IEEE Conference on Decision and Control. [S.l.]*, 2006. p. 5968–5973.

DIN, Ahmad; JABEEN, Meh; ZIA, Kashif; KHALID, Abbas; SAINI, Dinesh Kumar. Behavior-based swarm robotic search and rescue using fuzzy controller. **Computers Electrical Engineering**, v. 70, p. 53–65, 2018. ISSN 0045-7906. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0045790618314216>.

DORIGO, Marco; FLOREANO, Dario; GAMBARDELLA, Luca Maria; MONDADA, Francesco; NOLFI, Stefano; BAABOURA, Tarek; BIRATTARI, Mauro; BONANI, Michael; BRAMBILLA, Manuele; BRUTSCHY, Arne; BURNIER, Daniel; CAMPO, Alexandre; CHRISTENSEN, Anders Lyhne; DECUGNIERE, Antal; CARO, Gianni Di; DUCATELLE, Frederick; FERRANTE, Eliseo; FORSTER, Alexander; GONZALES, Javier Martinez; GUZZI, Jerome; LONGCHAMP, Valentin; MAGNENAT, Stephane; MATHEWS, Nithin; OCA, Marco Montes de; O'GRADY, Rehan; PINCIROLI, Carlo; PINI, Giovanni; RETORNAZ, Philippe; ROBERTS, James; SPERATI, Valerio; STIRLING, Timothy; STRANIERI, Alessandro; STUTZLE, Thomas; TRIANNI, Vito; TUCI, Elio; TURGUT, Ali Emre; VAUSSARD, Florian. Swarmanoid: a novel concept for the study of heterogeneous robotic swarms. **IEEE Robotics & Automation Magazine**, v. 20, n. 4, p. 60–71, 2013. Disponível em: <https://doi.org/10.1109/mra.2013.2252996>.

DRAGANJAC, Ivica; PETROVIĆ, Tamara; MIKLIĆ, Damjan; KOVAČIĆ, Zdenko; ORŠULIĆ, Juraj. Highly-scalable traffic management of autonomous industrial transportation systems. **Robotics and Computer-Integrated Manufacturing**, v. 63, p. 101915, 2020. ISSN 0736-5845. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0736584518303843>.

EBADINEZHAD, Sahar. DEACO: Adopting dynamic evaporation strategy to enhance ACO algorithm for the traveling salesman problem. **Engineering Applications of Artificial Intelligence**, v. 92, p. 103649, 2020. ISSN 0952-1976. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0952197620300993>.

EDWARDS, D. B.; BEAN, T. A.; ODELL, D. L.; ANDERSON, M. J. A leader-follower algorithm for multiple AUV formations. *In: IEEE. 2004 IEEE/OES Autonomous Underwater Vehicles (IEEE Cat. No. 04CH37578). [S.l.]*, 2004. p. 40–46.

EFRON, Bradley; TIBSHIRANI, Robert J. **An introduction to the bootstrap. [S.l.]**: CRC press, 1994.

El Ferik, Sami; Tariq Nasir, Mohammad; BAROUDI, Uthman. A behavioral adaptive Fuzzy controller of multi robots in a cluster space. **Applied Soft Computing**, v. 44, p. 117–127, 2016. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494616301272>.

ELKILANY, Basma Gh.; ABOUELSOUD, A. A.; FATHELBAB, Ahmed M. R.; ISHII, Hiroyuki. A proposed decentralized formation control algorithm for robot swarm based on an optimized potential field method. **Neural Computing and Applications**, Springer Science and Business Media LLC, v. 33, n. 1, p. 487–499, 2020. Disponível em: <https://doi.org/10.1007/s00521-020-05032-0>.

EZUGWU, Absalom El-Shamir; ADEWUMI, Aderemi Oluyinka; FRÎNCU, Marc Eduard. Simulated annealing based symbiotic organisms search optimization algorithm for traveling salesman problem. **Expert Systems with Applications**, v. 77, p. 189–210, 2017. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417417300702>.

FAIGL, Jan; KULICH, Miroslav. On benchmarking of frontier-based multi-robot exploration strategies. *In: IEEE. 2015 european conference on mobile robots (ECMR). [S.l.]*, 2015. p. 1–8.

FENG, Zhi; HU, Guoqiang; SUN, Yajuan; SOON, Jeffrey. An overview of collaborative robotic manipulation in multi-robot systems. **Annual Reviews in Control**, v. 49, p. 113–127, 2020. ISSN 1367-5788. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1367578820300043>.

FREDSLUND, Jakob; MATARIC, Maja J. A general algorithm for robot formations using local sensing and minimal communication. **IEEE transactions on robotics and automation**, IEEE, v. 18, n. 5, p. 837–846, 2002.

FREUDENTHALER, Gerhard; MEURER, Thomas. PDE-based multi-agent formation control using flatness and backstepping: analysis, design and robot experiments. **Automatica**, v. 115, p. 108897, 2020. ISSN 0005-1098. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0005109820300959>.

GARCIA-B, Abel; PÉREZ-T, Cristian G.; ESPINOZA-Q, Eduardo S.; TREJO-M, Francisco R.; RAMOS-V, Luis E.; ROMERO-T, Hugo; M., Muñoz P. J. Comparative study of three different path tracking controls in mobile robots. **International Journal of Computer Applications**, v. 975, p. 8887, 2013.

GAZEBO. 2003. Disponível em: <http://gazebosim.org/>.

GHOMMAM, Jawhar; MEHRJERDI, Hasan; SAAD, Maarouf; MNIF, Faiçal. Formation path following control of unicycle-type mobile robots. **Robotics and Autonomous**

Systems, v. 58, n. 5, p. 727–736, 2010. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889009001894>.

GIFFORD, Christopher M.; WEBB, Russell; BLEY, James; LEUNG, Daniel; CALNON, Mark; MAKAREWICZ, Joseph; BANZ, Bryan; AGAH, Arvin. A novel low-cost, limited-resource approach to autonomous multi-robot exploration and mapping. **Robotics and Autonomous Systems**, v. 58, n. 2, p. 186–202, 2010. ISSN 0921-8890. Selected papers from the 2007 European Conference on Mobile Robots (ECMR '07). Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889009001535>.

GONZÁLEZ, Patricia; OSORIO, Roberto R.; PARDO, Xoan C.; BANGA, Julio R.; DOALLO, Ramón. An efficient ant colony optimization framework for HPC environments. **Applied Soft Computing**, v. 114, p. 108058, 2022. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494621009704>.

GUNDUZ, Mesut; ASLAN, Murat. DJAYA: a discrete Jaya algorithm for solving traveling salesman problem. **Applied Soft Computing**, v. 105, p. 107275, 2021. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494621001988>.

GUO, Yu; ZHANG, Yun; MI, Zhenqiang; YANG, Yang; OBAIDAT, Mohammad S. Distributed task allocation algorithm based on connected dominating set for WSANs. **Ad Hoc Networks**, v. 89, p. 107–118, 2019. ISSN 1570-8705. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1570870519302562>.

HAN, Qing; REN, Shan; LANG, Hao; ZHANG, Changliang. **Bearing-based localization for leader-follower formation control**. Public Library of Science (PLoS), 2017. Disponível em: <https://doi.org/10.1371/journal.pone.0175378.s001>.

HASAN, Ola A.; ALI, Ramzy S.; RASHID, Abdulmuttalib T. Centralized approach for multi-node localization and identification. **architecture**, v. 11, p. 12, 2016.

HASAN, Ola A.; RASHID, Abdulmuttalib T.; ALI, Ramzy S. Hybrid approach for multi-node localization and identification. **Basrah Journal for Engineering Sciences**, v. 16, n. 2, p. 11–20, 2016.

HEINS, Jonathan; BOSSEK, Jakob; POHL, Janina; SEILER, Moritz; TRAUTMANN, Heike; KERSCHKE, Pascal. A study on the effects of normalized TSP features for automated algorithm selection. **Theoretical Computer Science**, v. 940, p. 123–145, 2023. ISSN 0304-3975. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0304397522006089>.

HELD, Michael; KARP, Richard M. A Dynamic Programming Approach to Sequencing Problems. **Journal of the Society for Industrial and Applied Mathematics**, v. 10, n. 1, p. 196–210, 1962. Disponível em: <https://doi.org/10.1137/0110015>.

HOU, S. P.; CHEAH, C. C. Dynamic compound shape control of robot swarm. **IET control theory & applications**, IET, v. 6, n. 3, p. 454–460, 2012.

HU, Yujiao; YAO, Yuan; LEE, Wee Sun. A reinforcement learning approach for optimizing multiple traveling salesman problems over graphs. **Knowledge-Based Systems**, v. 204, p. 106244, 2020. ISSN 0950-7051. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950705120304445>.

ISSA, B.; RASHID, A. RP LiDAR sensor for multi-robot localization using leader follower algorithm. **Iraqi Journal of Electrical and Electronic Engineering**, v. 15, n. 2, p. 2019, 2019.

ISSA, Bayadir A.; RASHID, Abdulmuttalib T. Two algorithms for static polygon shape formation control. **Iraqi Journal for Electrical And Electronic Engineering**, Basrah University, n. 3RD, 2020.

JEON, Jae D.; LEE, Beom H. Multi-robot formation shape control using convex optimization and bottleneck assignment. **Journal of Industrial and Intelligent Information Vol**, v. 2, n. 1, 2014.

JIN, Yaochu; GUO, Hongliang; MENG, Yan. Robustness analysis and failure recovery of a bio-inspired self-organizing multi-robot system. *In*: IEEE. **2009 Third IEEE International Conference on Self-Adaptive and Self-Organizing Systems**. [S.l.], 2009. p. 154–164.

JIN, Yaochu; GUO, Hongliang; MENG, Yan. A hierarchical gene regulatory network for adaptive multirobot pattern formation. **IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)**, IEEE, v. 42, n. 3, p. 805–816, 2012.

JOSHI, Apurva; WALA, Ankit; LUDHIYANI, Mohit; CHAKRABORTY, Debraj; CHUNG, Hoam; MANJUNATH, D. Outdoor cooperative flight using decentralized consensus algorithm and a guaranteed real-time communication protocol. **Control Engineering Practice**, v. 88, p. 128–140, 2019. ISSN 0967-0661. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0967066118303757>.

KHAN, Indadul; MAITI, Manas Kumar. A swap sequence based Artificial Bee Colony algorithm for Traveling Salesman Problem. **Swarm and Evolutionary Computation**, v. 44, p. 428–438, 2019. ISSN 2210-6502. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2210650216304588>.

KHATERI, Koresh; POURGHOLI, Mahdi; MONTAZERI, Mohsen; SABATTINI, Lorenzo. A connectivity preserving node permutation local method in limited range robotic networks. **Robotics and Autonomous Systems**, v. 129, p. 103540, 2020. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889019309959>.

KILOBOT. 2010. Disponível em: <https://www.k-team.com/mobile-robotics-products/kilobot>.

KOBAYASHI, Yuusuke; ENDO, Takahiro; MATSUNO, Fumitoshi. Distributed formation for robotic swarms considering their crossing motion. **Journal of the Franklin Institute**, Elsevier BV, v. 355, n. 17, p. 8698–8722, 2018. Disponível em: <https://doi.org/10.1016/j.jfranklin.2018.09.012>.

KRISHNAN, Shravan; RAJAGOPALAN, Govind Aadithya; KANDHASAMY, Sivanathan; SHANMUGAVEL, Madhavan. Continuous-time trajectory optimization for decentralized multi-robot navigation. **IFAC-PapersOnLine**, v. 53, n. 1, p. 494–499, 2020. ISSN 2405-8963. 6th Conference on Advances in Control and Optimization of Dynamical Systems ACODS 2020. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2405896320301026>.

LAFMEJANI, Amir Salimi; BERMAN, Spring. Nonlinear MPC for collision-free and deadlock-free navigation of multiple nonholonomic mobile robots. **Robotics and Autonomous Systems**, v. 141, p. 103774, 2021. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889021000592>.

LATRÉ, Benoît; BRAEM, Bart; MOERMAN, Ingrid; BLONDIA, Chris; DEMEESTER, Piet. A survey on wireless body area networks. **Wireless networks**, Springer, v. 17, n. 1, p. 1–18, 2011.

LEE, Daero; BUTCHER, Eric A.; SANYAL, Amit K. Sliding mode control for decentralized spacecraft formation flying using geometric mechanics. 2013.

LEE, Giroung; CHWA, Dongkyoung. Decentralized behavior-based formation control of multiple robots considering obstacle avoidance. **Intelligent Service Robotics**, Springer Science and Business Media LLC, v. 11, n. 1, p. 127–138, 2017. Disponível em: <https://doi.org/10.1007/s11370-017-0240-y>.

LEMAIGNAN, Séverin; WARNIER, Mathieu; SISBOT, E. Akin; CLODIC, Aurélie; ALAMI, Rachid. Artificial cognition for social human–robot interaction: an implementation. **Artificial Intelligence**, v. 247, p. 45–69, 2017. ISSN 0004-3702. Special Issue on AI and Robotics. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0004370216300790>.

LEWIS, M. Anthony; TAN, Kar-Han. High precision formation control of mobile robots using virtual structures. **Autonomous robots**, Springer, v. 4, n. 4, p. 387–403, 1997.

LI, Haiyuan; WEI, Hongxing; XIAO, Jiangyang; WANG, Tianmiao. Co-evolution framework of swarm self-assembly robots. **Neurocomputing**, v. 148, p. 112–121, 2015. ISSN 0925-2312. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231214009394>.

LIGANG, Han; RUI, Sun; JINGLI, Mao; XIANZHI, Li; LIHUI, Han; JING, Wen; SHUO, Ren; YANHUA, Wang; QING, Ma; LEILEI, Ding; SHENGCHUN, Li; XIAOPENG, Li;

YUANCHEN, Wang; XUEGUANG, Wang; XIAODONG, Wang; ZHIPENG, Guo; KAN, Xu. Data Communications and Network Technologies. *In: _____*. Singapore: Springer Nature Singapore, 2023. cap. Dynamic Routing, p. 193–220. ISBN 978-981-19-3029-4.

LIN, Chien-Chou; CHEN, Kun-Cheng; HSIAO, Po-Yuan; CHUANG, Wei-Ju. Motion planning of swarm robots using potential-based genetic algorithm. **International Journal of Innovative Computing, Information and Control ICIC**, v. 9, p. 305–318, 2013.

LIN, Honggui; CHEN, Kang; LIN, Ruiquan. Finite-time formation control of unmanned vehicles using nonlinear sliding mode control with disturbances. **Int. J. Innovat. Comput. Inf. Control**, v. 15, n. 6, p. 2341–2353, 2019.

LIN, Shen; KERNIGHAN, Brian W. An Effective Heuristic Algorithm for the Traveling-Salesman Problem. **Operational Research**, v. 21, p. 498–516, 1973.

LIN, Yu; BIAN, Zheyong; LIU, Xiang. Developing a dynamic neighborhood structure for an adaptive hybrid simulated annealing – tabu search algorithm to solve the symmetrical traveling salesman problem. **Applied Soft Computing**, v. 49, p. 937–952, 2016. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494616304306>.

LINDLEY, Brandon; TERAN-ROMERO, Luis Mier-y; SCHWARTZ, Ira B. Randomly distributed delayed communication and coherent swarm patterns. *In: IEEE. 2012 IEEE International Conference on Robotics and Automation. [S.l.]*, 2012. p. 4260–4265.

LIU, Fei; ZENG, Guangzhou. Study of genetic algorithm with reinforcement learning to solve the TSP. **Expert Systems with Applications**, v. 36, n. 3, Part 2, p. 6995–7001, 2009. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417408006064>.

LIU, Yutian; HU, Junjie. The analysis and avoidance of fault agent in flocking of multi-agent. **SBC**, v. 2, p. A4, 2014.

LU, Hao-Chun; HWANG, F. J.; HUANG, Yao-Huei. Parallel and distributed architecture of genetic algorithm on Apache Hadoop and Spark. **Applied Soft Computing**, v. 95, p. 106497, 2020. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494620304361>.

LUO, Jia; LI, Chaofeng; FAN, Qinqin; LIU, Yuxin. A graph convolutional encoder and multi-head attention decoder network for TSP via reinforcement learning. **Engineering Applications of Artificial Intelligence**, v. 112, p. 104848, 2022. ISSN 0952-1976. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0952197622001038>.

LUO, Shaocheng; KIM, Jonghoek; PARASURAMAN, Ramvijas; BAE, Jun Han; MATSON, Eric T.; MIN, Byung-Cheol. Multi-robot rendezvous based on bearing-aided hierarchical

tracking of network topology. **Ad Hoc Networks**, v. 86, p. 131–143, 2019. ISSN 1570-8705. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1570870518301100>.

LÓPEZ-GONZÁLEZ, A.; CAMPAÑA, J. A. Meda; MARTÍNEZ, E. G. Hernández; CONTRO, P. Paniagua. Multi robot distance based formation using parallel Genetic Algorithm. **Applied Soft Computing**, v. 86, p. 105929, 2020. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494619307100>.

MARINO, Alessandro; PIERRI, Francesco. A two stage approach for distributed cooperative manipulation of an unknown object without explicit communication and unknown number of robots. **Robotics and Autonomous Systems**, v. 103, p. 122–133, 2018. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889017307807>.

MARTINEZ-CLARK, Rigoberto; PLIEGO-JIMENEZ, Javier; CRUZ-HERNANDEZ, Cesar. Leader-follower formation control for mobile robots based on master-slave approach. 2018.

MARTÍNEZ, Pablo A.; GARCÍA, José M. ACOTSP-MF: A memory-friendly and highly scalable ACOTSP approach. **Engineering Applications of Artificial Intelligence**, v. 99, p. 104131, 2021. ISSN 0952-1976. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0952197620303687>.

MATSUKA, Kai; FELDMAN, Aaron O.; LUPU, Elena S.; CHUNG, Soon-Jo; HADAEGH, Fred Y. Decentralized formation pose estimation for spacecraft swarms. **Advances in Space Research**, v. 67, n. 11, p. 3527–3545, 2021. ISSN 0273-1177. Satellite Constellations and Formation Flying. Disponível em: <https://www.sciencedirect.com/science/article/pii/S027311772030418X>.

MENDONCA, Márcio; CHRUN, Ivan R.; NEVES-JR., Flávio; ARRUDA, Lúcia Valéria Ramos De. A cooperative architecture for swarm robotic based on dynamic fuzzy cognitive maps. **Engineering Applications of Artificial Intelligence**, Elsevier BV, v. 59, p. 122–132, 2017. Disponível em: <https://doi.org/10.1016/j.engappai.2016.12.017>.

MENDONCA, Rafael Mathias De; NEDJAH, Nadia; MACEDO, Luiza Mourelle De. Swarm robots with queue organization using infrared communication. *In: SPRINGER. International Conference on Computational Science and Its Applications. [S.l.]*, 2012. p. 136–147.

MENG, Yan; GUO, Hongliang; JIN, Yaochu. A morphogenetic approach to flexible and robust shape formation for swarm robotic systems. **Robotics and Autonomous Systems**, Elsevier BV, v. 61, n. 1, p. 25–38, 2013. Disponível em: <https://doi.org/10.1016/j.robot.2012.09.009>.

MIJALKOV, Mite; MCDANIEL, Austin; WEHR, Jan; VOLPE, Giovanni. Engineering sensorial delay to control phototaxis and emergent collective behaviors. **Physical Review X**, APS, v. 6, n. 1, p. 011008, 2016.

MISWANTO, I. Pranoto; NAIBORHU, J.; ACHMADI, S. Formation control of multiple Dubin's car system with geometric approach. **IOSR Journal of Mathematics (IOSRJM)**, v. 1, n. 6, p. 16–20, 2012.

MONDADA, Francesco. **Epuck**. 2022. Disponível em: <https://e-puck.gctronic.com/>.

MONDADA, Francesco; BONANI, Michael; RAEMY, Xavier; PUGH, James; CIANCI, Christopher; KLAPTOCZ, Adam; MAGNENAT, Stephane; ZUFFEREY, Jean-Christophe; FLOREANO, Dario; MARTINOLI, Alcherio. The e-puck, a robot designed for education in engineering. *In: IPCB: INSTITUTO POLITÉCNICO DE CASTELO BRANCO. Proceedings of the 9th conference on autonomous robot systems and competitions. [S.l.]*, 2009. v. 1, n. CONF, p. 59–65.

NASCIMENTO, Tiago P.; CONCEIÇÃO, André G. S.; MOREIRA, Antônio Paulo. Multi-robot nonlinear model predictive formation control: the obstacle avoidance problem. **Robotica**, Cambridge University Press (CUP), v. 34, n. 3, p. 549–567, 2014. Disponível em: <https://doi.org/10.1017/s0263574714001696>.

NASCIMENTO, Tiago P.; MOREIRA, Antônio Paulo; CONCEIÇÃO, André G. Scolari. Multi-robot nonlinear model predictive formation control: moving target and target absence. **Robotics and Autonomous Systems**, v. 61, n. 12, p. 1502–1515, 2013. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889013001310>.

NEDJAH, Nadia; RIBEIRO, Luigi Maciel; MACEDO, Luiza Mourelle De. Communication optimization for efficient dynamic task allocation in swarm robotics. *In: SPRINGER. International Conference on Bioinspired Methods and Their Applications. [S.l.]*, 2020. p. 110–124.

NEDJAH, Nadia; RIBEIRO, Luigi Maciel; MACEDO, Luiza Mourelle De. Communication optimization for efficient dynamic task allocation in swarm robotics. **Applied Soft Computing**, v. 105, p. 107297, 2021. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494621002209>.

NURMAINI, Siti; HASHIM, M.; ZAITON, Siti; TRIADI, Agus. Swarm robots control system based Fuzzy-PSO. *In: INSTITUTE OF ADVANCED ENGINEERING AND SCIENCE. 1st International Conference on Electrical Engineering, Computer Science and Informatics 2014. [S.l.]*, 2014.

OLCAY, Ertug; SCHUHMANN, Fabian; LOHMANN, Boris. Collective navigation of a multi-robot system in an unknown environment. **Robotics and Autonomous Systems**, v. 132, p. 103604, 2020. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889020304449>.

OLFATI-SABER, Reza. Flocking for multi-agent dynamic systems: algorithms and theory. **IEEE Transactions on automatic control**, IEEE, v. 51, n. 3, p. 401–420, 2006.

OTTONI, André L. C.; NEPOMUCENO, Erivelton G.; OLIVEIRA, Marcos S. de; OLIVEIRA, Daniela C. R. de. Reinforcement learning for the traveling salesman problem with refueling. **Complex Intelligent Systems**, 2021.

PANWAR, Karuna; DEEP, Kusum. Discrete Grey Wolf Optimizer for symmetric travelling salesman problem. **Applied Soft Computing**, v. 105, p. 107298, 2021. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494621002210>.

PHUNG, Manh Duong; QUACH, Cong Hoang; DINH, Tran Hiep; HA, Quang. Enhanced discrete particle swarm optimization path planning for uav vision-based surface inspection. **Automation in Construction**, v. 81, p. 25–33, 2017. ISSN 0926-5805. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0926580517303825>.

PRASAD, B. K. Swathi; MANJUNATH, Aditya G.; RAMASANGU, Hariharan. Multi-agent polygon formation using Reinforcement Learning. *In: ICAART (1)*. [S.l.: s.n.], 2017. p. 159–165.

QIAN, Dianwei; TONG, Shiwen; LI, Chengdong. Leader-following formation control of multiple robots with uncertainties through sliding mode and nonlinear disturbance observer. **ETRI Journal**, Wiley, v. 38, n. 5, p. 1008–1018, 2016. Disponível em: <https://doi.org/10.4218/etrij.16.0116.0048>.

QIAN, D. W.; TONG, S. W.; LI, C. D. Observer-based leader-following formation control of uncertain multiple agents by integral sliding mode. **Bulletin of the Polish Academy of Sciences Technical Sciences**, Walter de Gruyter GmbH, v. 65, n. 1, p. 35–44, 2017. Disponível em: <https://doi.org/10.1515/bpasts-2017-0005>.

RASHID, Abdulmuttalib T.; FRASCA, Mattia; ALI, Abduladhem A.; RIZZO, Alessandro; FORTUNA, Luigi. Multi-robot localization and orientation estimation using robotic cluster matching algorithm. **Robotics and Autonomous Systems**, Elsevier BV, v. 63, p. 108–121, 2015. Disponível em: <https://doi.org/10.1016/j.robot.2014.09.002>.

RAZALI, Noraini Mohd; GERAGHTY, John *et al.* Genetic algorithm performance with different selection strategies in solving TSP. *In: INTERNATIONAL ASSOCIATION OF ENGINEERS HONG KONG, CHINA. Proceedings of the world congress on engineering*. [S.l.], 2011. v. 2, n. 1, p. 1–6.

REINELT, Gerhard. Tsplib95. **Interdisziplinäres Zentrum für Wissenschaftliches Rechnen (IWR), Heidelberg**, v. 338, p. 1–16, 1995.

ROHMER, Eric; SINGH, Surya P. N.; FREESE, Marc. V-REP: a versatile and scalable robot simulation framework. *In: IEEE. 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. [S.l.]*, 2013. p. 1321–1326.

RUBENSTEIN, Michael. **Self-assembly and self-healing for robotic collectives. [S.l.]**: University of Southern California, 2009.

RUBENSTEIN, Michael; CORNEJO, Alejandro; NAGPAL, Radhika. Programmable self-assembly in a thousand-robot swarm. **Science**, American Association for the Advancement of Science, v. 345, n. 6198, p. 795–799, 2014.

RUTISHAUSER, Samuel; CORRELL, Nikolaus; MARTINOLI, Alcherio. Collaborative coverage using a swarm of networked miniature robots. **Robotics and Autonomous Systems**, v. 57, n. 5, p. 517–525, 2009. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889008001590>.

ŞAHIN, Erol. Swarm robotics: from sources of inspiration to domains of application. *In: SPRINGER. International workshop on swarm robotics. [S.l.]*, 2004. p. 10–20.

SASAOKA, Toshiki; KIMOTO, Isao; KISHIMOTO, Yosuke; TAKABA, Kiyotsugu; NAKASHIMA, Haya. Multi-robot SLAM via information Fusion Extended Kalman Filters. **IFAC-PapersOnLine**, Elsevier BV, v. 49, n. 22, p. 303–308, 2016. Disponível em: <https://doi.org/10.1016/j.ifacol.2016.10.414>.

SERGIYENKO, Oleg; FLORES-FUENTES, Wendy; MERCORELLI, Paolo. **Machine vision and navigation. [S.l.]**: Springer Nature, 2019.

SERGIYENKO, O. Yu.; IVANOV, M. V.; TYRSA, V. V.; KARTASHOV, V. M.; RIVAS-LÓPEZ, M.; HERNÁNDEZ-BALBUENA, D.; FLORES-FUENTES, W.; RODRÍGUEZ-QUIÑONEZ, J. C.; NIETO-HIPÓLITO, J. I.; HERNANDEZ, W.; TCHERNYKH, A. Data transferring model determination in robotic group. **Robotics and Autonomous Systems**, v. 83, p. 251–260, 2016. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889015303146>.

SHANG, Y.; RUML, W.; ZHANG, Y.; FROMHERZ, M. P. J. Localization from mere connectivity. *In: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing. [S.l.: s.n.]*, 2003. p. 201–212.

SHAPIRO, Samuel Sanford; WILK, Martin B. An analysis of variance test for normality (complete samples). **Biometrika**, JSTOR, v. 52, n. 3/4, p. 591–611, 1965.

SHURIJI, Mushreq Abdulhussain; SALMAN, Traiq Mohammed; ABDULNABI, Hussein A. Robots swarm communication control based on biological behavior inspiration. **Indonesian Journal of Electrical Engineering and Computer Science**, v. 16, n. 3, p. 1379–1391, 2019.

SIEGWART, Roland; NOURBAKHSH, Illah Reza; SCARAMUZZA, Davide. **Introduction to autonomous mobile robots**. [S.l.]: MIT press, 2011.

SIEMONSMA, Hector Garcia de Marina1 Johan; JAYAWARDHANA, Bayu; CAO, Ming. Multi-robot motion-formation distributed control with sensor self-calibration: experimental validation. 2018.

SILVA-JR, Luneque; NEDJAH, Nadia. Wave algorithm for recruitment in swarm robotics. *In*: SPRINGER. **International Conference on Computational Science and Its Applications**. [S.l.], 2015. p. 3–13.

SILVA-JR., Luneque; NEDJAH, Nadia. Efficient strategy for collective navigation control in swarm robotics. **Procedia Computer Science**, Elsevier, v. 80, p. 814–823, 2016.

SILVA-JR, Luneque; NEDJAH, Nadia. Wave algorithm applied to collective navigation of robotic swarms. **Applied Soft Computing**, Elsevier, v. 57, p. 698–707, 2017.

SOLEA, Razvan; CERNEGA, Daniela; FILIPESCU, Adrian; SERBENCU, Adriana. Formation control of multi-robots via sliding-mode technique. *In*: **ICINCO (2)**. [S.l.: s.n.], 2010. p. 161–166.

SONG, Yong; FANG, Xing; LIU, Bing; LI, Caihong; LI, Yibin; YANG, Simon X. A novel foraging algorithm for swarm robotics based on virtual pheromones and neural network. **Applied Soft Computing**, v. 90, p. 106156, 2020. ISSN 1568-4946.

STAGE. 2003. Disponível em: <http://playerstage.sourceforge.net/index.php?src=stage>.

SU, Yun; WANG, Ting; SHAO, Shiliang; YAO, Chen; WANG, Zhidong. GR-LOAM: LiDAR-based sensor fusion SLAM for ground robots on complex terrain. **Robotics and Autonomous Systems**, v. 140, p. 103759, 2021. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889021000440>.

SUTTON, Richard S.; BARTO, Andrew G. **Reinforcement learning: an introduction**. [S.l.]: MIT press, 2005.

SZWAYKOWSKA, Klementyna; ROMERO, Luis Mier-y-Teran; SCHWARTZ, Ira B. Collective motions of heterogeneous swarms. **IEEE Transactions on Automation Science and Engineering**, IEEE, v. 12, n. 3, p. 810–818, 2015.

SZWAYKOWSKA, Klementyna; SCHWARTZ, Ira B.; ROMERO, Luis Mier-y-Teran; HECKMAN, Christoffer R.; MOX, Dan; HSIEH, M. Ani. Collective motion patterns of swarms with delay coupling: theory and experiment. **Physical Review E**, APS, v. 93, n. 3, p. 032307, 2016.

SZWAYKOWSKA, Klementyna; TERAN-ROMERO, Luis Mier-y; SCHWARTZ, Ira B. Patterned dynamics of delay-coupled swarms with random communication graphs. *In: IEEE. 2015 54th IEEE Conference on Decision and Control (CDC)*. [S.l.], 2015. p. 6496–6501.

SÁ, Alan Oliveira De; NEDJAH, Nadia; MACEDO, Luiza Mourelle De. Distributed efficient localization in swarm robotics using min–max and particle swarm optimization. **Expert Systems with Applications**, Elsevier, v. 50, p. 55–65, 2016.

SÁ, Alan Oliveira De; NEDJAH, Nadia; MACEDO, Luiza Mourelle De. Distributed and resilient localization algorithm for swarm robotic systems. **Applied Soft Computing**, Elsevier, v. 57, p. 738–750, 2017.

TAHK, Min-Jea; PARK, Chang-Su; RYOO, Chang-Kyung. Line-of-sight guidance laws for formation flight. **Journal of Guidance, Control, and Dynamics**, v. 28, n. 4, p. 708–716, 2005.

TAILLARD éric D. A linearithmic heuristic for the travelling salesman problem. **European Journal of Operational Research**, v. 297, n. 2, p. 442–450, 2022. ISSN 0377-2217. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0377221721004628>.

TANNER, Herbert G.; JADBABAIE, Ali; PAPPAS, George J. Stable flocking of mobile agents part i: dynamic topology. *In: IEEE. 42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*. [S.l.], 2003. v. 2, p. 2016–2021.

TANNER, Herbert G.; JADBABAIE, Ali; PAPPAS, George J. Stable flocking of mobile agents, part i: fixed topology. *In: IEEE. 42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*. [S.l.], 2003. v. 2, p. 2010–2015.

TEIXEIRA, Marco Antonio Simoes; NEVES-JR, Flavio; KOUBAA, Anis; ARRUDA, Lucia Valeria Ramos De; OLIVEIRA, Andre Schneider De. A Quadral-Fuzzy control approach to flight formation by a fleet of unmanned aerial vehicles. **IEEE Access**, Institute of Electrical and Electronics Engineers (IEEE), v. 8, p. 64366–64381, 2020. Disponível em: <https://doi.org/10.1109/access.2020.2985032>.

TEL, Gerard. **Introduction to distributed algorithms**. [S.l.]: Cambridge university press, 2000.

TODOROV, Emanuel; EREZ, Tom; TASSA, Yuval. MuJoCo: A physics engine for model-based control. *In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012. Disponível em: <https://doi.org/10.1109/iros.2012.6386109>.

TRAN, Vu Phi; GARRATT, Matthew; PETERSEN, Ian R. Switching time-invariant formation control of a collaborative multi-agent system using negative imaginary systems theory. **Control Engineering Practice**, v. 95, p. 104245, 2020. ISSN 0967-0661. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0967066119302035>.

TRON, R.; THOMAS, J.; LOIANNO, G.; DANILIDIS, K.; KUMAR, V. A distributed optimization framework for localization and formation control: applications to vision-based measurements. **IEEE Control Systems**, Institute of Electrical and Electronics Engineers (IEEE), v. 36, n. 4, p. 22–44, 2016. Disponível em: <https://doi.org/10.1109/mcs.2016.2558401>.

TUANI, Ahamed Fayeez; KEEDWELL, Edward; COLLETT, Matthew. Heterogenous Adaptive Ant Colony Optimization with 3-opt local search for the Travelling Salesman Problem. **Applied Soft Computing**, v. 97, p. 106720, 2020. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S156849462030658X>.

TUTUKO, Bambang; NURMAINI, Siti; SAPARUDIN, Saparudin; FITRIANA, Gita Fadila. Enhancement of non-holonomic leader-follower formation using interval type-2 Fuzzy logic controller. **International Journal of Online Engineering (iJOE)**, International Association of Online Engineering (IAOE), v. 14, n. 09, p. 124, 2018. Disponível em: <https://doi.org/10.3991/ijoe.v14i09.8568>.

URREA, Claudio; MATTEODA, Rodrigo. Development of a virtual reality simulator for a strategy for coordinating cooperative manipulator robots using cloud computing. **Robotics and Autonomous Systems**, v. 126, p. 103447, 2020. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889019307651>.

van Bevern, René; SLUGINA, Viktoriia A. A historical note on the $3/2$ -approximation algorithm for the metric traveling salesman problem. **Historia Mathematica**, v. 53, p. 118–127, 2020. ISSN 0315-0860. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0315086020300240>.

VÁSÁRHELYI, Gábor; VIRÁGH, Cs; SOMORJAI, Gergo; TARCAI, Norbert; SZÖRÉNYI, Tamás; NEPUSZ, Tamás; VICSEK, Tamás. Outdoor flocking and formation flight with autonomous aerial robots. *In: IEEE. 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. [S.l.]*, 2014. p. 3866–3873.

WANG, Dongshu; WANG, Haitao; LIU, Lei. Unknown environment exploration of multi-robot system with the FORDPSO. **Swarm and Evolutionary Computation**, v. 26, p. 157–174, 2016. ISSN 2210-6502. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2210650215000711>.

WANG, Qin; CHEN, Zuwen; LIU, Peng; HUA, Qingguang. Distributed multi-robot formation control in switching networks. **Neurocomputing**, v. 270, p. 4–10, 2017. ISSN 0925-2312.

WEBOTS. 2004. Disponível em: <https://cyberbotics.com/>.

XIDIAS, Elias K.; AZARIADIS, Philip N. Computing collision-free motions for a team of robots using formation and non-holonomic constraints. **Robotics and Autonomous Systems**, v. 82, p. 15–23, 2016. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889016302160>.

XING, Ling-Ning; CHEN, Ying-Wu; YANG, Ke-Wei; HOU, Feng; SHEN, Xue-Shi; CAI, Huai-Ping. A hybrid approach combining an improved genetic algorithm and optimization strategies for the asymmetric traveling salesman problem. **Engineering Applications of Artificial Intelligence**, v. 21, n. 8, p. 1370–1380, 2008. ISSN 0952-1976. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0952197608000110>.

XU, Dongdong; ZHANG, Xingnan; ZHU, Zhangqing; CHEN, Chunlin; YANG, Pei. Behavior-based formation control of swarm robots. **Mathematical Problems in Engineering**, Hindawi, v. 2014, 2014.

YAMAGISHI, Kouhei; SUZUKI, Tsuyoshi. Collective movement method for swarm robot based on a thermodynamic model. **International Journal of Advanced Computer Science and Applications**, The Science and Information Organization, v. 8, n. 11, 2017.

YAMCHI, Mohammad Hosseinzadeh; ESFANJANI, Reza Mahboobi. Distributed predictive formation control of networked mobile robots subject to communication delay. **Robotics and Autonomous Systems**, v. 91, p. 194–207, 2017. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889016301014>.

YANG, Jian; WANG, Xin; BAUER, Peter. Extended PSO based collaborative searching for robotic swarms with practical constraints. **IEEE Access**, Institute of Electrical and Electronics Engineers (IEEE), v. 7, p. 76328–76341, 2019. Disponível em: <https://doi.org/10.1109/access.2019.2921621>.

YANG, L.; CAO, Z.; ZHOU, C.; CHENG, L.; TAN, M. Formation control and switching for multiple robots in certain environments. **International Journal of Robotics and Automation**, ACTA Press, v. 25, n. 3, 2010. Disponível em: <https://doi.org/10.2316/journal.206.2010.3.206-3345>.

YOUSSEFI, Khalil Al-Rahman; ROUHANI, Modjtaba. Swarm intelligence based robotic search in unknown maze-like environments. **Expert Systems with Applications**, v. 178, p. 114907, 2021. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417421003481>.

YU, Jinwei; JI, Jinchun; MIAO, Zhonghua; ZHOU, Jin. Neural network-based region reaching formation control for multi-robot systems in obstacle environment.

Neurocomputing, v. 333, p. 11–21, 2019. ISSN 0925-2312. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231218315091>.

YUAN, Chengzhi; HE, Haibo; WANG, Cong. Cooperative deterministic learning-based formation control for a group of nonlinear uncertain mechanical systems. **IEEE Transactions on Industrial Informatics**, Institute of Electrical and Electronics Engineers (IEEE), v. 15, n. 1, p. 319–333, 2019. Disponível em: <https://doi.org/10.1109/tii.2018.2792455>.

ZEDADRA, Ouarda; SERIDI, Hamid; JOUANDEAU, Nicolas; FORTINO, Giancarlo. A cooperative switching algorithm for multi-agent foraging. **Engineering Applications of Artificial Intelligence**, v. 50, p. 302–319, 2016. ISSN 0952-1976. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0952197616000294>.

ZHANG, Fangfang; WANG, Tingting; LI, Qiyan; XIN, Jianbin. An iterative optimization approach for multi-robot pattern formation in obstacle environment. **Robotics and Autonomous Systems**, v. 133, p. 103645, 2020. ISSN 0921-8890. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921889020304851>.

ZHANG, Qingguo; WANG, Jinghua; JIN, Cong; YE, Junmin; MA, Changlin; ZHANG, Wei. Genetic algorithm based wireless sensor network localization. *In: IEEE. 2008 Fourth International Conference on Natural Computation. [S.l.]*, 2008. v. 1, p. 608–613.

ZHANG, Shuai; LIU, Mingyong; LEI, Xiaokang; HUANG, Yunke; ZHANG, Feihu. Multi-target trapping with swarm robots based on pattern formation. **Robotics and Autonomous Systems**, Elsevier BV, v. 106, p. 1–13, 2018. Disponível em: <https://doi.org/10.1016/j.robot.2018.04.008>.

ZHANG, Xu; PHAM, Quang-Cuong. Planning coordinated motions for tethered planar mobile robots. **Robotics and Autonomous Systems**, Elsevier BV, v. 118, p. 189–203, 2019. Disponível em: <https://doi.org/10.1016/j.robot.2019.05.008>.

ZHANG, Zhen; HAN, Yang. Discrete sparrow search algorithm for symmetric traveling salesman problem. **Applied Soft Computing**, v. 118, p. 108469, 2022. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494622000321>.

ZHAO, Shibo; ZHANG, Hengrui; WANG, Peng; NOGUEIRA, Lucas; SCHERER, Sebastian. Super odometry: IMU-centric LiDAR-visual-inertial estimator for challenging environments. **arXiv preprint arXiv:2104.14938**, 2021.

ZHENG, Jiongzhi; HE, Kun; ZHOU, Jianrong; JIN, Yan; LI, Chu-Min. Reinforced Lin–Kernighan–Helsgaun algorithms for the traveling salesman problems. **Knowledge-Based Systems**, v. 260, p. 110144, 2023. ISSN 0950-7051. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950705122012400>.

ZHONG, Xianghui. On the approximation ratio of the 3-Opt algorithm for the (1,2)-TSP. **Operations Research Letters**, v. 49, n. 4, p. 515–521, 2021. ISSN 0167-6377. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0167637721000808>.

ZHONG, Yiwen; LIN, Juan; WANG, Lijin; ZHANG, Hui. Discrete comprehensive learning particle swarm optimization algorithm with Metropolis acceptance criterion for traveling salesman problem. **Swarm and Evolutionary Computation**, v. 42, p. 77–88, 2018. ISSN 2210-6502. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2210650216304680>.

ZHOU, Bo; ZHOU, Rui; GAN, Yahui; FANG, Fang; MAO, Yujie. Multi-Robot multi-station cooperative spot welding task allocation based on stepwise optimization: an industrial case study. **Robotics and Computer-Integrated Manufacturing**, v. 73, p. 102197, 2022. ISSN 0736-5845. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0736584521000806>.

ZHOU, Honglu; SONG, Mingli; PEDRYCZ, Witold. A comparative study of improved GA and PSO in solving multiple traveling salesmen problem. **Applied Soft Computing**, v. 64, p. 564–580, 2018. ISSN 1568-4946. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494617307561>.

ZHOU, Yi; LI, Weidong; WANG, Xiaomao; QIU, Yimin; SHEN, Weiming. Adaptive gradient descent enabled ant colony optimization for routing problems. **Swarm and Evolutionary Computation**, v. 70, p. 101046, 2022. ISSN 2210-6502. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2210650222000189>.