

UNIVERSIDADE DO ESTADO DE SANTA CATARINA – UDESC
CENTRO DE CIÊNCIAS TECNOLÓGICAS – CCT
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA – PPGCAP

MARIA EDUARDA SILVA DE MACEDO

**ALGORITMOS PARA AGREGAÇÃO EM ENXAMES DE ROBÔS: MODELAGEM E
IMPLEMENTAÇÃO DE COMPORTAMENTOS COLETIVOS EMERGENTES**

JOINVILLE

2026

MARIA EDUARDA SILVA DE MACEDO

**ALGORITMOS PARA AGREGAÇÃO EM ENXAMES DE ROBÔS: MODELAGEM E
IMPLEMENTAÇÃO DE COMPORTAMENTOS COLETIVOS EMERGENTES**

Dissertação apresentada ao Programa de Pós-Graduação em Computação Aplicada do Centro de Ciências Tecnológicas da Universidade do Estado de Santa Catarina, como requisito parcial para a obtenção do grau de Mestre em Computação Aplicada.

Orientador: Prof. Dr. Roberto Silvio Ubertino Rosso Junior

Coorientador: Prof. Dr. Yuri Kaszubowski Lopes

JOINVILLE

2026

**Ficha catalográfica elaborada pelo programa de geração automática da
Biblioteca Universitária Udesc,
com os dados fornecidos pelo(a) autor(a)**

Silva de Macedo, Maria Eduarda
ALGORITMOS PARA AGREGAÇÃO EM ENXAMES DE
ROBÔS: MODELAGEM E IMPLEMENTAÇÃO DE
COMPORTAMENTOS COLETIVOS EMERGENTES / Maria
Eduarda Silva de Macedo. -- 2026.
140 p.

Orientador: Roberto Silvio Ubertino Rosso Junior
Coorientador: Yuri Kaszubowski Lopes
Dissertação (mestrado) -- Universidade do Estado de Santa
Catarina, Centro de Ciências Tecnológicas, Programa de
Pós-Graduação em Computação Aplicada, Joinville, 2026.

1. Robótica de Enxame. 2. Comportamento de Enxame. 3.
Controle Minimalista. 4. Agregação Multitarefa. 5. Robôs Sem
Memória. I. Ubertino Rosso Junior, Roberto Silvio. II. Kaszubowski
Lopes, Yuri. III. Universidade do Estado de Santa Catarina, Centro
de Ciências Tecnológicas, Programa de Pós-Graduação em
Computação Aplicada. IV. Título.

MARIA EDUARDA SILVA DE MACEDO

**ALGORITMOS PARA AGREGAÇÃO EM ENXAMES DE ROBÔS: MODELAGEM E
IMPLEMENTAÇÃO DE COMPORTAMENTOS COLETIVOS EMERGENTES**

Dissertação apresentada ao Programa de Pós-Graduação em Computação Aplicada do Centro de Ciências Tecnológicas da Universidade do Estado de Santa Catarina, como requisito parcial para a obtenção do grau de Mestre em Computação Aplicada.

Orientador: Prof. Dr. Roberto Silvio Ubertino Rosso Junior

Coorientador: Prof. Dr. Yuri Kaszubowski Lopes

BANCA EXAMINADORA:

Dr. Roberto Silvio Ubertino Rosso Junior
Universidade do Estado de Santa Catarina (UDESC)
Presidente/Orientador

Membros:

Dr. Marco Antonio Simões Teixeira
Pontifícia Universidade Católica do Paraná (PUCPR)

Dra. Vivian Cremer Kalempa
Universidade do Estado de Santa Catarina (UDESC)

Dr. Marcelo Eduardo Pellenz - Suplente
Pontifícia Universidade Católica do Paraná (PUCPR)

Dr. Douglas Wildgrube Bertol - Suplente
Universidade do Estado de Santa Catarina (UDESC)

Joinville, 20 de fevereiro de 2026

Dedico este trabalho ao Senhor Jesus Cristo:
Porque Dele e por Ele e para Ele, são todas as
coisas; Glória, pois, a Ele eternamente. Amém.
Romanos 11:36

AGRADECIMENTOS

Agradeço a Deus por sempre estar comigo nesta jornada, por seu amor, fidelidade e infinita bondade.

Agradeço a meus pais, por sempre me apoiarem e me proporcionarem condições de me dedicar aos estudos, por várias vezes integralmente e por vibrarem comigo esta conquista.

Agradeço a meus avós, que não puderam acompanhar esta jornada, mas me ensinaram grandes lições que contribuíram ricamente para este resultado.

Agradeço ao meu melhor amigo que dividiu comigo a caminhada ao longo deste desafio.

Agradeço aos meus amigos da Igreja Presbiteriana do Brasil e da UDESC pelo companheirismo e amizade ao longo deste tempo.

Agradeço aos meus orientadores Roberto Silvio Ubertino Rosso Junior e Yuri Kaszubowski Lopes pelos aprendizados, conselhos e apoio desde o começo da jornada.

Agradeço a minha colega Ana Paula Chiarelli de Souza pela colaboração neste trabalho, sendo sempre exemplar e dedicada em tudo o que faz.

Agradeço a todos os meus professores por contribuíram com esta conquista diretamente.

Agradeço também aos demais servidores da UDESC que contribuem para o excelente funcionamento da instituição.

Agradeço a FAPESC (bolsa e TO 2024TR2555), a CAPES e ao PROMOP/UDESC por terem tornado esta pesquisa viável.

Agradeço a todos que de alguma forma contribuíram com este resultado.

“Não fui eu que ordenei a você? Seja forte e corajoso! Não se apavore nem desanime, pois o Senhor, o seu Deus, estará com você por onde você andar.” Josué 1:9

RESUMO

A robótica de enxame explora como comportamentos coletivos emergem de interações locais entre múltiplos robôs e é inspirada nos comportamentos de animais sociais. Por sua flexibilidade e adaptação, um enxame adota um controle minimalista, autônomo e descentralizado capaz de realizar tarefas mesmo com recursos limitados em ambientes desconhecidos, sem comunicação (ou com limitada comunicação local) ou memória. Este desafio fica ainda mais interessante quando, além do controle minimalista, o problema é generalizado para mais de um enxame simultâneo. Exemplo disso é a agregação multitarefa, na qual diferentes grupos de robôs, distintos por cores, simultaneamente devem se separar dos demais e agregar todos os seus integrantes em um agregado compacto. A partir de um controlador reativo, é possível mapear leituras de um sensor de visão e assim determinar as velocidades de cada roda dos robôs, o que permite que o comportamento coletivo emerja. Para encontrar os parâmetros ideais deste controlador, duas estratégias de otimização foram comparadas: (1) uma busca exaustiva (*grid search*) em um subespaço de parâmetros, aproveitando configurações já validadas na literatura; e (2) uma otimização automática no espaço de busca completo, utilizando algoritmos como DIRECT, MLSL, SBPLX e COBYLA. Após essa investigação o resultado é combinado com a literatura em uma nova proposta: alternância. Esta é uma arquitetura de controle híbrida que integra navegação visual reativa com um sistema de memória de curto prazo (*buffer*) alimentado por comunicação local. Este controlador opera através de uma alternância autônoma entre estados de Exploração e Agregação, dirigida pela densidade local de vizinhos percebida na memória do robô. Os resultados demonstram que a introdução do *buffer* permite otimização de desempenho, gerando os múltiplos agregados compactos e estáveis em diferentes cenários, mantendo o princípio da descentralização. Assim, identifica-se uma solução simples e promissora que pode ser explorada em trabalhos futuros para novos comportamentos ou, até mesmo, a experimentação com robôs físicos no mundo real.

Palavras-chave: Robótica de Enxame. Comportamento de Enxame. Controle Minimalista. Agregação Multitarefa. Robôs Sem Memória.

ABSTRACT

Swarm robotics explores how collective behaviors emerge from local interactions among multiple robots and is inspired by the behaviors of social animals. Due to its flexibility and adaptability, a swarm adopts a minimalist, autonomous, and decentralized control approach capable of performing tasks even with limited resources in unknown environments, without global communication (or with limited local communication) or memory. This challenge becomes even more interesting when, beyond minimalist control, the problem is generalized to more than one simultaneous swarm. An example of this is multi-task aggregation, in which different groups of robots, distinguished by colors, must simultaneously separate from the others and aggregate all their members into a compact cluster. Using a reactive controller, it is possible to map readings from a vision sensor to determine the velocities of each of the robots' wheels, allowing the collective behavior to emerge. To find the ideal parameters for this controller, two optimization strategies were compared: (1) an exhaustive search (grid search) in a parameter subspace, leveraging configurations already validated in the literature; and (2) an automatic design optimization in the full search space, using algorithms such as DIRECT, MLSL, SBPLX, and COBYLA. Based on this investigation, the result is combined with the literature into a new proposal: alternation. This is a hybrid control architecture that integrates reactive visual navigation with a short-term memory system (buffer) powered by local communication. This controller operates through autonomous alternation between Exploration and Aggregation states, driven by the local density of neighbors perceived in the robot's memory. The results demonstrate that the introduction of the buffer enables performance optimization, generating multiple compact and stable aggregates across different scenarios while maintaining the principle of decentralization. Thus, a simple and promising solution is identified that can be explored in future work for new behaviors or even experimentation with physical robots in the real world.

Keywords: Swarm Robotics. Swarm Behavior. Minimalist Control. Multi-task Aggregation. Memoryless Robots.

LISTA DE ILUSTRAÇÕES

Figura 1 – Principais robôs em SR	23
Figura 2 – Robô <i>e-puck</i>	24
Figura 3 – Formigas agrupando-se em busca de alimento.	27
Figura 4 – Etapas da Pesquisa	52
Figura 5 – Mapa de calor mostrando os resultados da <i>grid search</i> para $\bar{v}_{\ell,2}$ e $\bar{v}_{r,2}$	59
Figura 6 – Desempenho do controlador $\bar{v}_0$ durante o treinamento utilizando <i>grid search</i>	61
Figura 7 – Comparativos com $\bar{v}_0$	63
Figura 8 – Dispersão $\mathbf{v}_0$ Sem Cilindro de Referência	65
Figura 9 – Dispersão $\bar{\mathbf{v}}_0$ Com Cilindro de Referência	66
Figura 10 – Proporção $\bar{\mathbf{v}}_0$ Sem Cilindro de Referência	67
Figura 11 – Proporção $\bar{\mathbf{v}}_0$ Com Cilindro de Referência	68
Figura 12 – Comportamento $\bar{\mathbf{v}}_{\alpha_2}$ para $r = 90$ robôs com a presença de cilindros de referência.	70
Figura 13 – Comportamento $\bar{\mathbf{v}}_0$ para $r = 75$ robôs com a presença de cilindros de referência.	71
Figura 14 – Comportamento $\bar{\mathbf{v}}_0$ para $r = 90$ robôs sem a presença de cilindros de referência.	71
Figura 15 – Comportamento $\bar{\mathbf{v}}_0$ para $r = 90$ robôs com a presença de cilindros de referência.	72
Figura 16 – Comportamento $\bar{\mathbf{v}}_0$ para $r = 150$ robôs sem a presença de cilindros de referência.	72
Figura 17 – Comportamento $\bar{\mathbf{v}}_0$ para $r = 150$ robôs com a presença de cilindros de referência.	73
Figura 18 – Mapa de calor da <i>grid search</i> para $\bar{\mathbf{v}}_\beta = (\bar{v}_{\ell,0}, \bar{v}_{r,0}, \bar{v}_{\ell,1}, \bar{v}_{r,1})$	74
Figura 19 – Comparação Mapas de Calor	76
Figura 20 – Comportamento $\bar{\mathbf{v}}_1$ para $r = 75$ robôs com a presença de cilindros de referência.	77
Figura 21 – Comportamento $\bar{\mathbf{v}}_2$ para $r = 75$ robôs com a presença de cilindros de referência.	77
Figura 22 – Comportamento $\bar{\mathbf{v}}_3$ para $r = 75$ robôs com a presença de cilindros de referência.	78
Figura 23 – Comportamento $\bar{\mathbf{v}}_4$ para $r = 75$ robôs com a presença de cilindros de referência.	78
Figura 24 – Resultados $\bar{\mathbf{v}}_1$	80
Figura 25 – Resultados $\bar{\mathbf{v}}_2$	81
Figura 26 – Resultados $\bar{\mathbf{v}}_3$	82
Figura 27 – Resultados $\bar{\mathbf{v}}_4$	83
Figura 28 – Dispersão $\bar{\mathbf{v}}_4$ Sem Cilindro de Referência	84
Figura 29 – Dispersão $\bar{\mathbf{v}}_4$ Com Cilindro de Referência	85
Figura 30 – Proporção $\bar{\mathbf{v}}_4$ Sem Cilindro de Referência	86
Figura 31 – Proporção $\bar{\mathbf{v}}_4$ Com Cilindro de Referência	87
Figura 32 – Intersecção Mapas de Calor $\bar{\mathbf{v}}_0$ e $\bar{\mathbf{v}}_4$	88
Figura 33 – Resultados $\bar{\mathbf{v}}_{4_a}$	89
Figura 34 – Comparação comportamento de $\bar{\mathbf{v}}_{4_a}$	90
Figura 35 – Dispersão $\bar{\mathbf{v}}_{4_a}$ Sem Cilindro de Referência	91
Figura 36 – Dispersão $\bar{\mathbf{v}}_{4_a}$ Com Cilindro de Referência	92
Figura 37 – Proporção $\bar{\mathbf{v}}_{4_a}$ Sem Cilindro de Referência	93

Figura 38 – Proporção $\bar{v}_{4a}$ Com Cilindro de Referência	94
Figura 39 – Etapas da abordagem de otimização por algoritmos de busca global.	95
Figura 40 – Resultados $\bar{v}_5$	99
Figura 41 – Resultados $\bar{v}_6$	100
Figura 42 – Comparação comportamento de $\bar{v}_5$	101
Figura 43 – Comparação comportamento de $\bar{v}_6$	101
Figura 44 – Dispersão $\bar{v}_5$ Sem Cilindro de Referência	103
Figura 45 – Dispersão $\bar{v}_5$ Com Cilindro de Referência	104
Figura 46 – Proporção $\bar{v}_5$ Sem Cilindro de Referência	105
Figura 47 – Proporção $\bar{v}_5$ Com Cilindro de Referência	106
Figura 48 – Dispersão $\bar{v}_6$ Sem Cilindro de Referência	107
Figura 49 – Dispersão $\bar{v}_6$ Com Cilindro de Referência	108
Figura 50 – Proporção $\bar{v}_6$. Sem Cilindro de Referência	109
Figura 51 – Proporção $\bar{v}_6$. Com Cilindro de Referência	110
Figura 52 – Resultados $\bar{v}_7$ sem cilindro de referência	115
Figura 53 – Resultados $\bar{v}_7$ com cilindro de referência	116
Figura 54 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 75 robôs. . .	117
Figura 55 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 75 robôs. .	118
Figura 56 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 125 robôs. .	118
Figura 57 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 125 robôs. .	119
Figura 58 – Dispersão $\bar{v}_7$ Sem Cilindro de Referência	120
Figura 59 – Dispersão $\bar{v}_7$ Com Cilindro de Referência	121
Figura 60 – Proporção $\bar{v}_7$ Sem Cilindro de Referência	122
Figura 61 – Proporção $\bar{v}_7$ Com Cilindro de Referência	123
Figura 62 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 30 robôs. . .	131
Figura 63 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 30 robôs. .	131
Figura 64 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 50 robôs. . .	132
Figura 65 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 50 robôs. .	132
Figura 66 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 45 robôs. . .	133
Figura 67 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 45 robôs. .	133
Figura 68 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 75 robôs. . .	134
Figura 69 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 75 robôs. .	134
Figura 70 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 60 robôs. . .	135
Figura 71 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 60 robôs. .	135
Figura 72 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 100 robôs. .	136
Figura 73 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 100 robôs. .	136
Figura 74 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 75 robôs. . .	137
Figura 75 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 75 robôs. .	137
Figura 76 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 125 robôs. .	138

Figura 77 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 125 robôs. .	138
Figura 78 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 90 robôs. . .	139
Figura 79 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 90 robôs. .	139
Figura 80 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 150 robôs. .	140
Figura 81 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 150 robôs. .	140

LISTA DE TABELAS

Tabela 1	– Resultados dos experimentos de otimização com Algoritmo Global MLSL .	96
Tabela 2	– Primeira Comparação das Configurações Analisadas.	111
Tabela 3	– Simulações para validação da configuração $\bar{\mathbf{v}}_7$	114
Tabela 4	– Segunda Comparação das Configurações Analisadas.	124

LISTA DE ABREVIATURAS E SIGLAS

CMA-ES	Covariance Matrix Adaptation Evolution Strategy
COBYLA	Constrained Optimization By Linear Approximations
DIRECT	DIviding RECTangles
ID	Identificador
IR	<i>Infrared</i> (Infravermelho)
μ AV	<i>Micro Aerial Vehicles</i>
MLSL	Multi-Level Single-Linkage
MRS	<i>Multi-Robot Systems</i>
UAV	<i>Unmanned Aerial Vehicles</i>
Udesc	Universidade do Estado de Santa Catarina
USV	<i>Unmanned Surface Vehicles</i>
UUV	<i>Unmanned Submarine Vehicles</i>
SBPLX	Subplex
SI	<i>Swarm Intelligence</i>
SR	<i>Swarm Robotics</i>

SUMÁRIO

1	INTRODUÇÃO	16
1.1	JUSTIFICATIVA	17
1.2	DEFINIÇÃO DO PROBLEMA	18
1.3	OBJETIVOS	18
1.4	ESCOPO	19
2	FUNDAMENTAÇÃO TEÓRICA	21
2.1	ROBÓTICA DE ENXAME	21
2.2	PLATAFORMAS ROBÓTICAS	22
2.3	PLATAFORMA DE SIMULAÇÃO	24
2.4	COMPORTAMENTO DE AGREGAÇÃO EM ENXAME	25
2.5	CONTROLE DE ENXAMES ROBÓTICOS	27
2.5.1	Arquiteturas de Controle: Centralizada <i>versus</i> Descentralizada	28
2.5.2	O Desafio do <i>Design</i>: Da Abordagem Manual à Automática	29
2.5.3	Aspectos da Coordenação e Composição do Enxame	30
2.5.4	Controle Minimalista	31
2.6	MÉTODOS DE OTIMIZAÇÃO	32
2.6.1	<i>GRID SEARCH</i>	32
2.6.2	Otimização de Parâmetros para Controle de Enxames	32
2.6.3	Algoritmos de Otimização para Problemas de Caixa-Preta	33
2.7	CONSIDERAÇÕES DO CAPÍTULO	34
3	REVISÃO BIBLIOGRÁFICA E TRABALHOS RELACIONADOS . . .	35
3.1	PESQUISA BIBLIOGRÁFICA SISTEMÁTICA	35
3.1.1	Revisão Bibliográfica	37
3.2	TRABALHOS RELACIONADOS	39
3.2.1	Agregação Sem Computação	40
3.2.2	Limitações Teóricas da Agregação Bimodal	43
3.2.3	Generalização para Agregação Multitarefa	44
3.2.3.1	<i>Dispersão</i>	46
3.2.3.2	<i>Proporção de Agrupamento</i>	46
3.2.4	Outras Investigações em Controle Minimalista	47
3.3	CONSIDERAÇÕES DO CAPÍTULO	50
4	MÉTODO DE SOLUÇÃO PROPOSTO	51
4.1	ESTRUTURA DE CONTROLE E AMBIENTE DE SIMULAÇÃO	52
4.1.1	Estrutura do Controlador	53
4.1.2	Métricas de Desempenho	53

4.1.3	Configuração do Ambiente de Simulação	54
4.2	ABORDAGEM 1: OTIMIZAÇÃO POR <i>GRID SEARCH</i>	55
4.3	ABORDAGEM 2: OTIMIZAÇÃO AUTOMÁTICA COM ALGORITMOS DE BUSCA GLOBAL	56
5	RESULTADOS	58
5.1	ABORDAGEM DE OTIMIZAÇÃO POR <i>GRID SEARCH</i>	58
5.1.1	<i>Grid Search</i> : Estatísticas e Resultados	59
5.1.2	Validação e Escalabilidade Experimental	64
5.1.3	Exploração de Novos Parâmetros	73
5.1.4	Comparação dos Resultados	88
5.2	ALGORITMOS DE OTIMIZAÇÃO	95
5.2.1	Algoritmos de Otimização: Estatísticas e Resultados	98
6	ALGORITMO DE ALTERNÂNCIA	111
6.1	MÉTODO	111
6.2	RESULTADOS	114
7	CONCLUSÃO E TRABALHOS FUTUROS	125
	REFERÊNCIAS	127
	APÊNDICE A – SNAPSHOTS $\bar{v}_7$	131

1 INTRODUÇÃO

A robótica de enxame é uma área inspirada nos comportamentos coletivos observados em animais sociais, como por exemplo, formigas e abelhas. Esses sistemas são tolerantes a falhas, adaptativos, lidam bem com o aumento do número de robôs e ainda são capazes de realizar tarefas como agregação, alocação de tarefas e formação de padrões (Dorigo; Theraulaz; Trianni, 2021).

Os comportamentos coletivos são provenientes da interação local entre agentes simples, autônomos e distribuídos, que no caso do enxame são os próprios robôs (Dorigo; Theraulaz; Trianni, 2021). Em relação ao controle dos robôs, ele é descentralizado, não havendo uma unidade central de controle, o que impõe desafios significativos, como a coordenação de decisões coletivas, a navegação em ambientes dinâmicos e a comunicação limitada entre agentes (Devi et al., 2024).

Brambilla et al. (2013) propõem uma abordagem sistemática denominada "engenharia de enxames", composta por quatro etapas fundamentais: modelagem e especificação de requisitos, projeto e implementação, verificação e validação, e operação e manutenção. Os autores também introduzem uma taxonomia para os métodos utilizados em robótica de enxame, classificando-os em métodos de projeto e métodos de análise.

Os métodos de projeto são divididos em três categorias: métodos manuais, nos quais o projetista define regras de comportamento individuais com base em sua intuição ou experiência; métodos automáticos, que utilizam técnicas como aprendizado por reforço e robótica evolutiva para gerar comportamentos com base em um objetivo global; e métodos híbridos, que combinam os dois anteriores (Brambilla et al., 2013). Quanto aos métodos de análise, são frequentemente utilizadas equações diferenciais com atraso, também chamado retardo, que permitem descrever e prever a alocação de robôs em tarefas ou posições, além de avaliar a estabilidade das configurações atingidas.

Ademais, a definição e a categorização dos comportamentos coletivos são importantes na robótica de enxame. Esses comportamentos podem ser agrupados em três categorias principais: comportamentos espaciais (e.g., agregação, dispersão, cobertura, movimentação em formação e construção), comportamentos orientados a tarefas (e.g., forrageamento, transporte coletivo e execução de tarefas sequenciais), e tomada de decisão coletiva (e.g., consenso, seleção de locais e alocação de tarefas) (Brambilla et al., 2013).

Entre eles, destaca-se a autoagregação, considerada fundamental por servir de base para outras formas de cooperação (Camazine et al., 2001), (Dorigo et al., 2004). A autoagregação consiste no agrupamento espontâneo e compacto de robôs em um ambiente, sem controle centralizado, geralmente por meio de controladores reativos ou redes neurais recorrentes.

Gauci et al. (2014b) demonstraram a eficácia de abordagens evolutivas para calibrar controles reativos e controladores recorrentes para o comportamento de autoagregação. No mesmo trabalho, controladores recorrentes produziram formações compactas em 99 de 100

execuções, enquanto os reativos obtiveram sucesso em apenas 19.

Em outro trabalho, Gauci et al. (2014a) apresentam uma solução para a agregação auto-organizada de robôs que não requer computação aritmética nem memória. Para isso, foi utilizada uma abordagem de otimização exaustiva, busca em grade ou, do inglês, *grid search*, para encontrar o controlador ótimo, testando sistematicamente o espaço de controladores possíveis. A busca revelou que um controlador simples, no qual os robôs giram no próprio eixo ao detectar outro robô e, caso contrário, deslocam-se para trás em curva. A eficácia dessa abordagem foi validada pelos autores com simulações com até 1000 robôs e em experimentos físicos com um enxame de 40 robôs *e-puck*.

No trabalho de Krischanski et al. (2023), os autores generalizaram este comportamento de agregação encontrado por Gauci et al. (2014a) para um ambiente multitarefa, permitindo múltiplas instâncias da mesma tarefa em execução simultânea. Para tornar a busca menos dependente do projetista, os autores utilizaram o *design* automático, ou seja, treinaram um dado algoritmo de busca global a fim de encontrar o melhor controlador e, em seguida, avaliaram seu desempenho. No entanto, os comportamentos resultantes, embora funcionais, apresentaram dificuldade em formar corretamente agregados de robôs, especialmente quando o número deles aumentava. Em particular, observa-se a emergência de um padrão circular não compacto, caracterizado por robôs que seguem uns aos outros.

Assim, a literatura mostra diferentes estratégias já testadas para controle minimalista. Entretanto, ainda há lacunas a serem investigadas. Dessa forma, este trabalho busca refinar e combinar soluções estudadas por autores como Gauci et al. (2014a) e Krischanski et al. (2023). As abordagens de otimização *grid search* e *design* automático serão comparadas, buscando encontrar a melhor configuração para agregação de enxame de robôs com múltiplos grupos, sem nenhum tipo de comunicação ou informação do ambiente.

1.1 JUSTIFICATIVA

A implementação da agregação de maneira eficaz permite que o enxame aumente sua robustez a falhas e opere de maneira coordenada. Paralelamente, o controle minimalista, emprega robôs com capacidades sensoriais e computacionais restritas, surge como um paradigma de grande interesse, devido à sua capacidade de reduzir a complexidade e o custo dos robôs, promovendo escalabilidade e facilitando a generalização de comportamentos em diferentes cenários.

O estudo da agregação com controle minimalista aplicado a cenários com divisão do enxame em múltiplos grupos, pode ser muito útil a aplicações realistas com multitarefa, mas ainda é um desafio em aberto. Trabalhos anteriores, como o de Krischanski et al. (2023), obtiveram sucesso funcional na divisão de grupos, mas com limitações em obter agregados de robôs compactos, ou seja, com a mínima distância entre os robôs, e dificuldade em se manter estável conforme o número de robôs aumentava no ambiente (escalabilidade) indicando uma lacuna na

literatura.

Portanto, neste trabalho busca-se desenvolver e avaliar estratégias de controle minimalista que permitam solucionar essas limitações, obtendo múltiplos grupos de robôs em agregados compactos e estáveis. Isso contribuirá para a criação de enxames robóticos mais flexíveis, adaptáveis e aptos a operar em ambientes dinâmicos.

Desta forma, a necessidade de superar as limitações de trabalhos anteriores justifica esta pesquisa. Os resultados iniciais da investigação, que constituem a principal contribuição deste trabalho, foram consolidados em um artigo científico curto publicado na conferência internacional TAROS 2025 (*Towards Autonomous Robotic Systems*) (Macedo et al., 2025).

1.2 DEFINIÇÃO DO PROBLEMA

Em um ambiente virtual de duas dimensões de formato quadrado, considera-se um enxame robótico homogêneo composto por $r \geq 2$ robôs móveis, organizados em $g \geq 2$ grupos distintos, definidos pela cor atribuída a cada robô. Os robôs não possuem memória, nem quaisquer informações a respeito do ambiente ou comunicação entre si. Eles são equipados apenas com: sensor de visão, capaz de interpretar a cor de um único elemento localizado à sua frente (no centro da imagem), e um par de rodas diferenciais, no qual cada roda pode assumir velocidades distintas no mesmo instante de tempo, sendo importante ao percorrer trajetórias curvilíneas.

O enxame adota controle descentralizado e uma estrutura de controle responsável por indicar as velocidades do par de rodas diferenciais a cada leitura do sensor de visão. Além disso, a simulação no ambiente virtual inicia com cada robô espalhado aleatoriamente pelo ambiente e o objetivo é que, ao final, cada grupo tenha reunido todos os seus robôs formando um agregado compacto.

O sucesso da solução é medido após a simulação por meio de métricas como dispersão e proporção de agrupamento, as quais indicam, respectivamente, o quão espalhados os robôs de um mesmo grupo estão e quantos robôs já foram agregados corretamente ao longo do tempo. Essas definições foram estabelecidas a partir da análise de trabalhos relacionados, sobretudo destacam-se: (Gauci et al., 2014a), (Gauci et al., 2014b) e (Krischanski et al., 2023), discutidos na Seção 3.2.

1.3 OBJETIVOS

O objetivo geral deste trabalho é investigar e encontrar um controlador capaz de proporcionar a formação de agregados compactos para múltiplos grupos. Para isso, são estabelecidos os seguintes objetivos específicos:

- Utilização de controle minimalista reativo para robótica de enxame;
- Comparar as soluções encontradas por diferentes abordagens: *grid search* e *design* automático, que combina técnicas de busca global e local;

- Avaliar diferentes configurações de parâmetros para o controlador por meio das métricas de dispersão e proporção de agrupamentos;
- Identificar o desempenho das configurações quanto à capacidade de gerar agregados compactos ao invés de orbitas em formato de anéis;
- Otimizar a segregação dos grupos antes da formação de agrupamentos; e
- Investigar a escalabilidade dessas configurações ao variar o número de robôs (r) e de grupos (g).

1.4 ESCOPO

Dadas as características do problema, uso de controle minimalista reativo para *Swarm Robotics* (SR), este trabalho é limitado pela lacuna teórica que aponta a impossibilidade de garantir a agregação para um número arbitrário de robôs ($r > 2$) com controladores bimodais sem memória (Steinberg; Solovey, 2024). Apesar destas restrições, busca-se encontrar soluções que sejam funcionais no mundo real. Dessa forma, pode-se definir essa abordagem como empírico-prática.

Embora já haja soluções funcionais para a agregação multi-grupo, elas produzem agregados pouco compactos, como por exemplo formações circulares, onde a distância entre cada robô não é a mínima possível, Krischanski et al. (2023). Em termos de escalabilidade, elas apresentam dificuldade em separar os grupos quando o número de robôs totais aumenta. Assim, a contribuição central desta pesquisa é encontrar e validar configurações de controle minimalistas que superem esses desafios, gerando múltiplos agregados que sejam simultaneamente estáveis e compactos.

O controlador é projetado para robôs com um par rodas diferenciais (cada roda pode assumir valores distintos de velocidade) e capacidades sensoriais mínimas, com um único sensor de visão, operando de forma totalmente descentralizada, sem qualquer tipo de comunicação explícita ou memória de estados passados. Ainda para delimitar a pesquisa e utilizar apenas recursos disponíveis, a implementação em *hardware* físico está de fora dos limites deste trabalho e a investigação é realizada em um ambiente virtual de simulação de duas dimensões, sem a presença de obstáculos, a fim de explorar o comportamento de agregação no enxame.

Também foi definido percurso metodológico. Ele inicia por uma revisão bibliográfica para selecionar a literatura científica, seguida pela formulação do problema e da proposta de solução e, então, da experimentação por meio de simulações para testar hipóteses, com a validação dos resultados feita por meio de métricas adequadas e comparação com estudos anteriores.

O trabalho está organizado de modo progressivo. Inicialmente, a Introdução contextualiza o tema (Capítulo 1), apresentando os objetivos e a motivação. Em seguida, a Fundamentação Teórica revisa os conceitos essenciais que sustentam a investigação (Capítulo 2), enquanto a Revisão Bibliográfica e Trabalhos Relacionados analisa estudos relevantes da literatura, situando

a contribuição desta pesquisa (Capítulo 3). O Capítulo 4, denominado Método de Solução Proposto, descreve os procedimentos adotados nos experimentos, o Capítulo 5, denominado Resultados, apresenta os dados obtidos, discutindo e interpretando os resultados e a proposta final com o Algoritmo de Alternância (Capítulo 6). Por fim, Conclusão e Trabalhos Futuros sintetizam as principais contribuições do estudo e indicam direções para etapas futuras da pesquisa (Capítulo 7).

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os fundamentos teóricos que sustentam o trabalho e conceitos relacionados a robótica de enxame. São abordados o comportamento de agregação, técnicas de *design* de controle e otimização, e métricas de avaliação que são utilizadas ao longo do trabalho, de modo a contextualizar o problema de pesquisa e a solução proposta.

2.1 ROBÓTICA DE ENXAME

A Robótica de Enxame, do inglês *Swarm Robotics* (SR), é inspirada pelo comportamento de sistemas biológicos, como colônias de insetos e cardumes de peixes (Shahzad et al., 2023). Essa abordagem pode ser definida como a coordenação de múltiplos agentes inteligentes, robôs autônomos, os quais configuram o enxame. Segundo Brambilla et al. (2013), cada enxame deve conter pelo menos três agentes, os quais seguem um conjunto homogêneo de regras de comportamento locais que visam objetivos coletivos, operando com mínima intervenção humana e sem controle centralizado.

Um princípio fundamental da SR é a simplicidade relativa de cada agente individual, tanto em termos de *hardware* quanto de *software*, quando comparados à complexidade da tarefa coletiva que realizam (Bayındır, 2016). Essa característica é superada pelo poder de interação do enxame, uma vez que as capacidades coletivas se sobressaem as habilidades individuais. Essa arquitetura emergente confere à SR vantagens sobre sistemas de robô único (Kuckling, 2023).

Entre essas vantagens está a robustez, que é a redundância estrutural do enxame, a qual garante que falhas individuais como desconexões, perdas, remoção de um robô, entre outros, tenham impacto mínimo sobre o desempenho geral (Shahzad et al., 2023), permitindo a continuidade da tarefa e estabilidade do sistema (Gautam; Mohan, 2012; Brambilla et al., 2013). Pode-se citar também a flexibilidade, que refere-se à adaptação do SR a ambientes e tarefas dinâmicas sem necessidade de reprogramação dos agentes, respondendo de forma eficaz a mudanças contextuais (Bayındır, 2016; Devi et al., 2024). Outra vantagem é a escalabilidade, a qual corresponde à eficiência do enxame em permanecer estável mesmo com variações no número de robôs (Kuckling, 2023).

Essas características permitem diferentes aplicações para a SR. A capacidade do enxame de cobrir grandes áreas de forma eficiente e se adaptar a cenários imprevisíveis o torna particularmente promissor para tarefas complexas no mundo real, das quais se destacam:

- Exploração Cooperativa e Busca e Resgate: Em ambientes desconhecidos ou atingidos por desastres (Mattson et al., 2025), um SR pode explorar rapidamente grandes áreas (Devi et al., 2024);
- Monitoramento Ambiental: Um enxame pode ser utilizado na detecção de resíduos, prevenção de incêndios (Albani et al., 2017), em monitoramento aquático (Mattson et al., 2025) e aéreo (Shahzad et al., 2023);

- Agricultura de Precisão: Para tarefas como monitoramento e mapeamento de ervas daninhas em campos, avaliação da saúde de plantações, gerenciamento de tarefas agrícolas (Albani et al., 2017) e tarefas de semeadura (Shahzad et al., 2023);
- Logística e Automação de Armazéns: Utilizado para coleta de pedidos (Schranz et al., 2020);
- Transporte Cooperativo: Movimentação de objetos grandes ou pesados demais para um único robô (Chen et al., 2015);
- Exploração Espacial: Explorar áreas fora do planeta Terra com o uso de enxames de satélites de pequena escala (Chung et al., 2018) ou sistemas robóticos reconfiguráveis para exploração planetária (Shahzad et al., 2023); e
- Educação: Plataformas de robóticas de baixo custo utilizadas para pesquisa e fins educacionais (Schranz et al., 2020), como os robôs *e-puck* (Lopes et al., 2016).

2.2 PLATAFORMAS ROBÓTICAS

A aplicação das abordagens SR depende diretamente das plataformas robóticas utilizadas, pois é o *hardware* que determina as capacidades práticas e as limitações de cada robô no sistema. Essas plataformas podem ser classificadas em: voltadas à pesquisa e voltadas à aplicação industrial. O primeiro grupo trata-se de ferramentas laboratoriais utilizadas para demonstrar, verificar e experimentar algoritmos de enxame, enquanto o segundo representa as implantações desses sistemas em cenários reais (Schranz et al., 2020).

Quanto ao meio de operação, as plataformas podem ser classificadas em espaciais, aéreas, terrestres e aquáticas. Dentre elas, os robôs terrestres têm sido os mais estudados e empregados na literatura, sendo também os mais relevantes para as investigações deste trabalho (Shahzad et al., 2023).

Neste contexto, pode-se afirmar: plataformas aéreas, como *Unmanned Aerial Vehicles* (UAV) e *Micro Aerial Vehicles* (μ AV), são comumente empregados em atividades de monitoramento, vigilância e resgate (Dorigo; Theraulaz; Trianni, 2021); aquáticas, como *Unmanned Surface Vehicles* e *Unmanned Submarine Vehicles* (USV e UUV), são aplicados a tarefas como exploração subaquática, monitoramento ambiental e avaliação da qualidade da água; sistemas espaciais geralmente envolvem satélites em formação de enxame, com foco em missões de exploração planetária e pesquisa orbital. Já em relação as plataformas terrestres voltadas à pesquisa, alguns exemplos são elencados na Figura 1:

Figura 1 – Principais robôs utilizados em SR. (a) Jasmine; (b) Alice; (c) Kilobots; (d) e-pucks; (e) Swarm-bots; e (f) Swarmanoid.



Fonte: Adaptado de Dorigo, Theraulaz e Trianni (2021).

- Jasmine e Kilobots: pequenos e de baixo custo, projetados para operações escaláveis com centenas ou milhares de unidades, com capacidades de processamento bem limitadas (Kuckling, 2023);
- Alice: plataforma de microrrobô autônomo, útil para enxames de larga escala (Schranz et

al., 2020);

- *Swarm-bots*: robôs móveis autônomos com capacidade de auto-montagem (Schranz et al., 2020);
- *Swarmanoid*: integrado ao *design*, desenvolvimento e controle de sistemas robóticos de enxames heterogêneos (Schranz et al., 2020);
- *e-puck*: este robô, mostrado com mais detalhes na Figura 2, é uma das plataformas mais utilizadas em SR (Dorigo; Theraulaz; Trianni, 2021) e, por isso, foi escolhido para este trabalho. O *e-puck* possui uma estrutura com diâmetro, altura e peso iguais a 7,4 cm, 5,5 cm e 150 g, respectivamente. Além disso, é equipado com diversos sensores e atuadores, incluindo uma câmera localizada em sua parte frontal. Também possui um par de rodas diferenciais acionadas por motores de passo que podem ser controladas de modo independente para mover-se para frente e para trás, com velocidade mínima e máxima de $-12,8$ cm/s e $12,8$ cm/s, respectivamente.

Figura 2 – Robô *e-puck*.



Fonte: Adaptado de Mondada et al. (2009).

Apesar dos avanços, existe uma lacuna entre a pesquisa teórica, focada em simulação, e a industrial, focada no desenvolvimento de *hardware* (Schranz et al., 2020) com aplicações industriais ainda limitadas (Dorigo; Theraulaz; Trianni, 2021). Isso reforça a necessidade de reduzir a distância entre os dois campos. Neste trabalho, este robô será utilizado em simulações.

2.3 PLATAFORMA DE SIMULAÇÃO

As plataformas de simulação reproduzem virtualmente cenários dinâmicos e as capacidades e interações dos robôs (Bayındır, 2016). Elas são muito úteis para avaliar o desempenho do

enxame e otimizar algoritmos quando comparados a experimentos com robôs reais, que podem ser demorados e caros (Brambilla et al., 2013).

Alguns exemplos de plataformas de simulação são mostradas a seguir:

- ARGoS: destaca-se pelo suporte nativo à pesquisa em SR, modularidade, paralelismo e capacidade de simular até 10.000 robôs ao mesmo tempo e suporta modelos como *e-pucks* e *kilobots* (Brambilla et al., 2013);
- *Flightmar*: simulador para robôs aéreos, capaz de renderizar o ambiente e útil para estudos de navegação visual e sensoriamento remoto (Dorigo; Theraulaz; Trianni, 2021);
- *Player/Stage*: muito utilizado para programar aplicações multirrobô (Gautam; Mohan, 2012); e
- Enki: biblioteca de simulação para robótica com suporte para *e-puck* (Magenat; Waibel; Beyeler, 2005). O *Enki* é uma plataforma de simulação de código aberto, desenvolvida em C++ e especializada na modelagem da cinemática e dinâmica de corpos rígidos em duas dimensões. Ele possui suporte nativo para a plataforma robótica *e-puck*, fornecendo um modelo virtual validado para a experimentação (Magenat; Waibel; Beyeler, 2005). Como ferramenta, o *Enki* é muito útil para o desenvolvimento e a avaliação de algoritmos de robótica de enxame, permitindo a execução de experimentos sistemáticos em um ambiente controlado (Community, 2025). Por essas características, e por ter sido utilizado em trabalhos relacionados (Krischanski et al., 2023), ele foi escolhido para realização desta pesquisa.

Vale destacar o desafio da “lacuna da realidade” que simulações em ambientes virtuais enfrentam, o qual consiste na diferença entre o desempenho da simulação e os resultados alcançados em experimentos com robôs reais (Pagliuca; Vitanza, 2025; Kuckling, 2023). Isso se deve ao fato de ser inviável simular todos os aspectos da realidade de forma perfeita, como ruído e interações físicas complexas (Brambilla et al., 2013).

2.4 COMPORTAMENTO DE AGREGAÇÃO EM ENXAME

Os princípios de SR são considerados bioinspirados, pois baseiam-se na inteligência de enxame, do inglês *Swarm Intelligence* (SI) (Shahzad et al., 2023). A SI, refere-se aos fenômenos naturais nos quais comportamentos complexos e de larga escala, surgem das interações locais entre muitos agentes individuais simples, tal como observado em animais sociais (Pagliuca; Vitanza, 2025; Fengying et al., 2024).

O conceito central de SI é a auto-organização, responsável por descrever a capacidade do sistema de gerar padrões e formações complexas sem um controle central ou um plano global (Kuckling, 2023). O resultado deste processo é chamado de comportamento emergente, ou seja, um padrão global que “emerge” como consequência das interações locais entre os robôs, as quais

seguem um conjunto de regras de controle simples (Gauci et al., 2014a; Firat et al., 2020). Assim, o principal desafio da área consiste em projetar essas regras locais para que o comportamento desejado emergja (Devi et al., 2024).

O padrão global, resultante do comportamento emergente, é chamado de comportamento coletivo. Ele descreve as ações coordenadas de um grupo de robôs que interagem para alcançar um objetivo comum, focado na cooperação e a coordenação entre os agentes (Bayındır, 2016). Os comportamentos coletivos podem ser classificados de acordo com sua função principal, destacando-se (Shahzad et al., 2023):

- Organização Espacial: Ações em que os robôs se organizam no ambiente, como na formação de padrões, montagem de estruturas e agregação;
- Navegação Coordenada: Tarefas que envolvem o movimento conjunto do enxame, como exploração de áreas, localização e transporte coletivo; e
- Tomada de Decisão Coletiva: Processos nos quais o enxame chega a uma escolha unificada, como a alocação de tarefas, a busca por consenso e a percepção coletiva.

Pode-se dizer que os comportamentos coletivos são o “o quê” o enxame faz, enquanto comportamentos emergentes explicam o “como” esse comportamento complexo surge das interações simples. E essa capacidade intrínseca do sistema de gerar ordem e padrões complexos sem um controle externo é definida como auto-organização (Kuckling, 2023). Apesar da diversidade de comportamentos que podem ser gerados, sua implantação em aplicações no mundo real ainda enfrenta desafios significativos de *hardware* e a dificuldade de lidar com a imprevisibilidade de ambientes dinâmicos (Pagliuca; Vitanza, 2025).

Além disso, a emergência dos comportamentos coletivos em sistemas robóticos não ocorre de forma espontânea como nos sistemas biológicos, o que implica na necessidade de ser projetada. O principal desafio da engenharia de enxames está em alcançar o objetivo coletivo desejado, macro-comportamento, a partir de um conjunto de regras individuais simples que cada robô deve seguir, micro-comportamento (Brambilla et al., 2013). A implementação dessas regras é a função do controlador dos robôs, fundamental em projeto de SR, e discutido na Seção 2.5.

Dentre os diversos comportamentos coletivos, a agregação é um dos mais estudados e aplicados em SR. Seu objetivo é reunir os agentes de forma autônoma em uma ou mais regiões do ambiente, aumentando a densidade local para aumentar as interações entre eles (Shahzad et al., 2023).

Na biologia, ela é muito aplicada em fenômenos biológicos, como na formação de colônias de insetos para proteção contra predadores ou para otimização da busca por alimento, sendo caracterizada pela ausência de controle centralizado e pela auto-organização, vide Figura 3. Frequentemente a agregação é utilizada em tarefas como seleção de área, nas quais o enxame deve se reunir de forma autônoma em um ou mais locais específicos do ambiente, como "ninhos" ou "abrigos" (Pagliuca; Vitanza, 2025).

Figura 3 – Formigas agrupando-se em busca de alimento.



Fonte: A autora (2025), com base em código Python adaptado de inteligência artificial (Gemini).

Geralmente, a coordenação de agregação ocorre por meio da estigmergia, tipo de comunicação implícita, na qual os robôs modificam o ambiente ou utilizam características ambientais existentes para transmitir informações inspirando-se, por exemplo, em trilhas de feromônios deixadas por formigas (Devi et al., 2024). Experimentos mais complexos, usam esse conceito para testar a robustez do enxame em tarefas como, a formação de agregados em um abrigo com acesso restrito ou a união dos robôs em um local específico, usando apenas as pistas ambientais para orientação (Francesca et al., 2015). Ademais, a importância da agregação em SR está em sua característica como “bloco construtivo” para tarefas mais complexas. Isso porque, a partir de um enxame agregado, pode-se iniciar comportamentos como forrageamento, vigilância distribuída ou transporte cooperativo. Assim, a agregação demonstra ser uma estratégia eficiente e escalável para o desenvolvimento de sistemas de controle distribuído robustos (Brambilla et al., 2013).

Em aplicações reais, a agregação já é usada para: para coordenar a exploração e o monitoramento de longo prazo em ecossistemas aquáticos complexos, como a Lagoa de Veneza, enxame de aéreos que se agrupam antes de executar formações para missões de vigilância e na inspeção e limpeza de cascos de grandes navios (Schranz et al., 2020).

2.5 CONTROLE DE ENXAMES ROBÓTICOS

O controlador é o componente de *software* e *hardware* responsável por implementar as regras individuais dos robôs, conforme mencionado na Seção 2.4. A partir do mapeamento das percepções sensoriais, o controlador dita as decisões tomadas pelos robôs, o que é expresso por uma estrutura de controle com um conjunto de parâmetros definidos.

A escolha da abordagem de controle, desde a sua arquitetura até o método de *design* desses parâmetros, é uma decisão crucial no projeto de qualquer Sistema de Múltiplos Robôs (MRS - *Multi-robot Systems*), incluindo SR. Isso porque, definirá como o desafio micro-macro será resolvido para obter o comportamento coletivo desejado. Entre as arquiteturas mais comuns estão: centralizada e descentralizada (Schranz et al., 2020).

2.5.1 Arquiteturas de Controle: Centralizada *versus* Descentralizada

As duas abordagens predominantes, centralizada e descentralizada, afetam diretamente características como robustez, escalabilidade e tipo de tarefa que o sistema pode realizar. Ao compará-las, nota-se a necessidade de diminuir uma qualidade ou vantagem para ganhar ou aumentar outra, ou seja, um *trade-off* (em português, troca compensatória). Este *trade-off* ocorre entre a previsibilidade da engenharia tradicional e a adaptabilidade bioinspirada da robótica de enxame.

No controle centralizado, comum em MRS, uma única unidade de controle (CCU - *Central Control Unit*) é responsável por gerenciar e tomar as decisões para todo o grupo, a partir da comunicação constante e direta entre a CCU e cada robô (Schranz et al., 2020). Essa característica torna a arquitetura vantajosa para pequenos grupos de robôs, e possibilita a execução de cálculos complexos em tempo real, permitindo realizar tarefas que exigem alto poder computacional e conhecimento global do ambiente, como operações de alta precisão, sendo as restrições de volume de dados geralmente mais flexíveis (Brambilla et al., 2013).

A preferência pelo controle centralizado em aplicações industriais, geralmente é fundamentada pela dificuldade de garantir a segurança e disponibilidade com sistemas emergentes, tornando-se preferível escolher a previsibilidade no lugar dos benefícios da auto-organização (Schranz et al., 2020). Outra vantagem, reside no fato destes sistemas serem mais eficazes que sistemas descentralizados em determinadas situações, conferindo estabilidade, melhor custo-benefício e respostas mais diretas em condições complexas. Também são adaptáveis para tarefas específicas, principalmente quando há planejamento de rotas ou rastreamento dos ambientes.

Apesar disso, sua principal desvantagem é a fragilidade, pois a CCU representa um ponto único de falha, com escalabilidade do sistema limitada, já que o controle se torna mais complexo com o aumento do número de robôs (Schranz et al., 2020). A adaptação a ambientes dinâmicos também é comprometida, já que os agentes têm pouca ou nenhuma capacidade de decisão independente, o que reduz a capacidade exploratória e a resposta a emergências de curto prazo (Duan et al., 2022). Assim, sistemas centralizados podem apresentar maiores taxas de falha, maior consumo de energia e tempo para conclusão de tarefas mais longo em comparação com sistemas descentralizados, além da dificuldade de criar modelos simplificados do problema e do ambiente em que os robôs operarão.

Já o controle descentralizado, comumente utilizado em SR, não possui uma entidade que lidera os robôs, o comportamento coletivo é auto-organizado e a tomada de decisão é distribuída entre todos os membros do enxame (Brambilla et al., 2013). Isso confere robustez e tolerância

a falhas, permitindo que o enxame continue operando mesmo se um ou vários robôs falharem, afinal a redundância inerente ao enxame, em que nenhuma função depende de um único robô, garante que a falha de um indivíduo não comprometa a realização da tarefa coletiva.

Além disso, há flexibilidade e adaptabilidade, uma vez que os robôs podem se adaptar a ambientes diferentes e dinâmicos (Schranz et al., 2020). Outro ponto relevante é a escalabilidade, pois sistemas descentralizados lidam bem com aumento ou remoção de robôs sem que o desempenho seja significativamente afetado, isso se deve à comunicação e às interações locais. O controle descentralizado favorece a auto-organização, possibilitando que comportamentos coletivos emergjam das interações locais entre os robôs, reduzindo a dependência de um projetista humano para implementar o *software* de controle (Kuckling, 2023).

Ainda há o barateamento da implementação, pois a construção de múltiplos robôs simples é geralmente mais econômica do que a de um único robô complexo, conferindo melhor custo-benefício (Gautam; Mohan, 2012). Também pode-se otimizar a comunicação, minimizando-a com protocolos específicos, para que o desempenho do enxame não seja comprometido com o aumento do número de agentes.

Entre suas desvantagens, estão a dificuldade de concepção e de previsibilidade. Isso porque, projetar o comportamento individual de cada robô de modo que o desempenho global seja máximo requer que o projetista antecipe como as interações locais contribuirão para o comportamento coletivo emergente (Brambilla et al., 2013). Prever com precisão se o comportamento do enxame será adequado em aplicações, industriais por exemplo, é um desafio.

Outro aspecto refere-se à complexidade em desenvolver modelos tanto no nível microscópico (individual) quanto no nível macroscópico (coletivo), devido a auto-organização (Brambilla et al., 2013). Muitos métodos matemáticos formais ainda são baseados em suposições simplificadas que não se aplicam à robótica de enxame devido a fatores como ruído, assincronia, estocasticidade e ausência de informação global.

Devido às capacidades restritas de sensoramento e comunicação, os robôs em sistemas descentralizados apresentam limitações que dificultam a modelagem das regras de controle de maneira precisa (Bayındır, 2016). Além disso, as redes de comunicação em enxame tendem a ser menos estáveis e mais vulneráveis, podendo falhar quando alguns robôs estão fora alcance ou apresentam defeitos.

Já, para obter as vantagens de um enxame, como robustez e escalabilidade é necessário um projeto cauteloso. Afinal, exercer controle sobre o enxame em operação quando ocorrem comportamentos indesejados é muito difícil, já que não há um ponto central de comando (Dorigo; Theraulaz; Trianni, 2021).

2.5.2 O Desafio do *Design*: Da Abordagem Manual à Automática

A dificuldade em resolver o problema micro-macro no controle descentralizado motivou a evolução das metodologias de *design*. Inicialmente, o projeto de controladores era um processo manual, baseado em tentativa e erro e nas habilidades do pesquisador (Francesca et al., 2015).

Para superar essa limitação, foram desenvolvidas as metodologias automáticas, nas quais o problema de *design* é formulado como um problema de otimização (Francesca; Birattari, 2016).

Dentre as abordagens de *design* automático, a robótica evolutiva consolidou-se como uma ferramenta importante em SR. Inspirada na seleção natural, ela utiliza algoritmos para gerar e selecionar iterativamente os comportamentos individuais mais aptos a alcançar o objetivo coletivo, tratando o *design* do controlador como uma busca por uma solução ótima em espaço de possibilidades (Kuckling, 2023).

Apesar dos avanços, a área ainda apresenta escassez de práticas empíricas sistematizadas para o *design* de enxames (Francesca; Birattari, 2016). Destacam-se seis necessidades principais para o amadurecimento da área:

1. Definição de modelos de referência formais para garantir comparações justas (Francesca et al., 2015);
2. Descrição detalhada e reproduzível dos métodos, com implementações públicas (Francesca; Birattari, 2016);
3. Estabelecimento de *benchmarks* públicos e padronizados (Chakraborty et al., 2023);
4. Realização de experimentos com robôs reais para superar a lacuna da realidade (Birattari et al., 2019);
5. Como o enxame é auto-organizado e descentralizado, há dificuldade para o humano estabelecer um canal de comunicação claro com o grupo (Dorigo; Theraulaz; Trianni, 2020); e
6. Ainda há fragmentação pelo desenvolvimento de *hardware ad hoc*, desenvolvido para um problema específico, e não otimizado para enxames, tornando a parte experimental complexa e sem a opção de reutilização de componentes em diferentes contextos (Dorigo; Theraulaz; Trianni, 2020).

2.5.3 Aspectos da Coordenação e Composição do Enxame

Além da arquitetura de controle e do método de *design*, outros dois aspectos são importantes para a coordenação do enxame: a sua composição e a forma como os agentes se comunicam. Os enxames podem ser homogêneos, quando todos os robôs possuem o mesmo *hardware* e *software*, o que favorece a redundância e a tolerância a falhas (Dorigo; Theraulaz; Trianni, 2021; Duan et al., 2022). Já enxames heterogêneos, são constituídos por robôs distintos ou com diferentes especializações, oferecendo maior flexibilidade comportamental, mas com maior complexidade de projetar seu comportamento auto-organizado (Dorigo; Theraulaz; Trianni, 2021).

A comunicação é outro aspecto relevante da coordenação em SR. A comunicação explícita refere-se à troca direta de mensagens entre os agentes, sendo vantajosa para reações rápidas, mas

podendo sobrecarregar a rede em enxames grandes e prejudicando a escalabilidade (Gautam; Mohan, 2012; Fengying et al., 2024). Já a comunicação implícita, ou stigmergia, ocorre quando os agentes interagem indiretamente, modificando o ambiente, consistindo em uma forma de coordenação mais robusta e escalável, pois não depende de canais de comunicação diretos (Devi et al., 2024; Gautam; Mohan, 2012).

Assim, nota-se que os desafios em SR se intensificam quando a comunicação é limitada ou totalmente ausente (Krischanski et al., 2023), quando comparadas com abordagens que assumem alguma forma de troca de informações. Todavia, propostas recentes demonstram a viabilidade de alcançar comportamentos coordenados complexos sem qualquer tipo de comunicação no enxame, chamado de controle minimalista (Gauci et al., 2014b).

2.5.4 Controle Minimalista

O controle minimalista propõe a redução da complexidade dos robôs individuais ao máximo, restringindo suas capacidades de armazenar informações e realizar cálculos (Gauci et al., 2014a). O objetivo é investigar o potencial de um enxame executar determinadas tarefas mesmo com essas limitações (Gauci et al., 2014b). As motivações para esta linha de pesquisa incluem: o barateamento do sistema; a viabilidade de aplicações em micro e nanoescala, nas quais os recursos são intrinsecamente escassos; e o aumento da escalabilidade, uma vez que os requisitos de processamento sofrem pouca influência do tamanho do enxame (Chen et al., 2015).

O controle minimalista é comumente reativo, isto é, a ação de um robô corresponde a uma resposta imediata à sua percepção sensorial, sem utilização de memória de eventos passados ou comunicação explícita com seus vizinhos (Gauci et al., 2014a). Geralmente, os robôs são equipados com sensores simples, como por exemplo um único sensor binário de linha de visão que apenas detecta a presença de um objeto à sua frente (Krischanski et al., 2023).

Ademais, o projeto de controladores para estes robôs se torna um desafio, podendo recorrer tanto a métodos manuais quanto de *design* automático para otimizar os parâmetros que promovem o comportamento coletivo desejado (Francesca; Birattari, 2016; Gauci et al., 2014a).

O comportamento de agregação é um caso de estudo central para o controle minimalista. Em seus trabalhos seminais, Gauci et al. (2014a) e Gauci et al. (2014), os autores demonstraram por meio de simulações e experimentos físicos, que um enxame de robôs sem memória e com um simples controlador bimodal poderia se auto-organizar e agregar corretamente para o caso de dois robôs, mostrando que um comportamento complexo poderia emergir de regras extremamente simples.

Entretanto, a convergência para um número arbitrário de robôs ($r \geq 2$) é um desafio. Isso porque controladores bimodais simples podem falhar em cenários com mais de dois robôs devido a configurações de impasse (*deadlock*) e oclusão, sendo impossível garantir matematicamente a agregação para todos os casos (Steinberg; Solovey, 2024). Apesar disso, em cenários realistas, o ruído inerente ao sistema, como pequenas imprecisões de movimento e sensoriamento, frequentemente colaboram para que o enxame escape desses impasses e atinja seu objetivo, reforçando a

validade da busca por soluções empiricamente robustas (Steinberg; Solovey, 2024).

Mesmo com estas limitações teóricas na agregação, o paradigma minimalista foi aplicado com sucesso a uma variedade de outras tarefas, como o agrupamento de objetos (Gauci et al., 2014b), o transporte cooperativo (Chen et al., 2015), o forrageamento (Talamali et al., 2020) e a captura de alvos (Lei et al., 2023). Estes estudos reforçam que comportamentos coletivos sofisticados podem emergir de regras individuais simples, mesmo que a garantia de sucesso universal não seja possível de provar matematicamente (Talamali et al., 2020; Steinberg; Solovey, 2024).

2.6 MÉTODOS DE OTIMIZAÇÃO

Nesse contexto, os métodos de otimização tornam-se ferramentas fundamentais, pois permitem a busca sistemática por parâmetros de controle que gerem o comportamento desejado. Esta Seção detalha duas estratégias de otimização utilizadas para automatizar o *design* de comportamentos em enxames de robôs: busca exaustiva e algoritmos de otimização sem derivadas.

2.6.1 GRID SEARCH

A técnica de *grid search* é particularmente atraente nesse contexto porque, apesar de sua aparente simplicidade, ela oferece uma abordagem sistemática e exaustiva de exploração do espaço de busca para os parâmetros do controlador (Mitrano et al., 2019). Consiste, essencialmente, na definição de um conjunto discreto de valores para cada parâmetro de interesse e na avaliação de todas as combinações possíveis entre esses valores.

Essa característica a torna particularmente adequada em estágios iniciais de exploração de um problema, quando se busca compreender a sensibilidade do sistema aos parâmetros e identificar regiões promissoras do espaço de busca. Essa técnica já foi implementada por autores como Gauci et al. (2014a) e Mitrano et al. (2019), os quais exploram o controle minimalista em SR, vide Seção 3.2.

A aplicação da *grid search* em um enxame robótico requer algumas decisões que impactam diretamente a qualidade dos resultados obtidos. Em primeiro lugar, é necessário determinar quais parâmetros serão incluídos na busca. Outra consideração fundamental é o desenho dos cenários de teste.

2.6.2 Otimização de Parâmetros para Controle de Enxames

Com o intuito de resolver o problema micro-macro, uma abordagem consolidada na literatura é o *design* automático. Ele remove a necessidade de um projetista humano especificar manualmente as regras individuais, tratando o projeto do controlador como um problema de otimização (Francesca; Birattari, 2016).

O objetivo dessa otimização é encontrar um conjunto de parâmetros para o controlador que maximize uma métrica de desempenho (ou minimize uma função de custo), a qual quantifica o sucesso do comportamento coletivo. Como a função que mapeia os parâmetros do desempenho do enxame é geralmente caixa-preta, ou seja, sem expressão analítica, ruidosa e com alto custo computacional, pois depende de simulações, algoritmos que não exigem o cálculo de derivadas são os mais indicados neste caso (Birattari et al., 2019).

Esses algoritmos podem ser divididos em locais e globais.

- **Algoritmos de Otimização Local:** Partem de uma solução candidata e buscam refiná-la explorando sua vizinhança no espaço de busca. São eficientes para convergir rapidamente para uma solução ótima, mas possuem maior probabilidade de ficarem presos em ótimos locais, que não representam a melhor solução possível em todo o espaço de busca (Johnson, 2022). Métodos como o de *Nelder-Mead* e suas variações são exemplos clássicos desta abordagem.
- **Algoritmos de Otimização Global:** São projetados para explorar todo o espaço de busca, com o objetivo de encontrar o ótimo global. Para isso, utilizam estratégias como a amostragem de múltiplos pontos de partida (*multistart*) ou a divisão sistemática do espaço de busca em sub-regiões (Johnson, 2022). Embora mais robustos contra ótimos locais, geralmente exigem um número maior de avaliações da função objetivo.

A combinação de estratégias de otimização local e global é uma abordagem híbrida poderosa e eficiente para a *design* de controladores. Enquanto os algoritmos globais são robustos para mapear amplamente o espaço de parâmetros e identificar regiões promissoras, eles podem ser computacionalmente caros para convergir com alta precisão. Por outro lado, algoritmos locais, embora rápidos no refinamento de uma solução, falham em garantir que essa solução seja o ótimo global (Brambilla et al., 2013).

Portanto, a associação entre os dois métodos permite superar as limitações individuais, pois há uma fase de exploração global que identifica pontos de partida promissores, posteriormente entregues a um otimizador local para uma busca de exploração rápida e precisa.

2.6.3 Algoritmos de Otimização para Problemas de Caixa-Preta

Para problemas de caixa-preta, algoritmos que não exigem o cálculo de derivadas são essenciais. A biblioteca NLOpt (*Non-Linear Optimization*), por exemplo, é uma coleção de *software* que fornece implementações de código aberto para diversas dessas estratégias de otimização não linear, tanto locais quanto globais (Kumar; Soman; Jayadeva, 2016). Dentre os algoritmos relevantes para este trabalho, destacam-se:

SBPLX é um algoritmo de otimização local e uma variação do método de *Nelder-Mead*. Sua principal característica é a abordagem *Subplex*, que decompõe o espaço de busca em subespaços de menor dimensão para aplicar o método Simplex. Essa estratégia aumenta a

eficiência e a estabilidade da busca, especialmente em problemas com muitas variáveis (Johnson, 2022).

O *Constrained Optimization By Linear Approximations* (COBYLA) também é um método de otimização local. Ele destaca-se por sua capacidade de lidar com restrições não lineares complexas, tanto de igualdade quanto de desigualdade, por meio de aproximações lineares sucessivas da função objetivo e das restrições (Kumar; Soman; Jayadeva, 2016).

O *Multi-Level Single-Linkage* (MLSL) é um algoritmo de otimização global que exemplifica a abordagem híbrida *multistart*. Ele combina uma fase de exploração global, por meio da amostragem aleatória de pontos de partida, com a eficiência de otimizadores locais para refinar as soluções mais promissoras encontradas. Uma heurística de agrupamento é utilizada para evitar buscas redundantes (Johnson, 2022).

O algoritmo *Dliding RECTangles* (DIRECT) é uma técnica de otimização global determinística. Ele opera dividindo iterativamente o espaço de busca em uma coleção de hiper-retângulos de diferentes tamanhos. A estratégia de seleção de quais retângulos dividir equilibra a exploração de novas áreas (regiões maiores) com a exploração de regiões já conhecidas como promissoras (regiões menores), sendo eficaz mesmo em funções com múltiplos mínimos locais, apesar de não utilizar otimizadores locais (Kumar; Soman; Jayadeva, 2016).

2.7 CONSIDERAÇÕES DO CAPÍTULO

A análise dos conceitos apresentada neste capítulo demonstra a progressiva maturação de SR como uma disciplina de engenharia. Foi evidenciada a transição de uma abordagem puramente bioinspirada para um campo que emprega metodologias rigorosas, como o *design* automático, para solucionar o desafio de projetar regras locais que resultem em um comportamento global coeso, problema micro-macro.

Esta discussão aprofundou-se no paradigma do controle minimalista, que se mostra um campo de pesquisa promissor e de grande relevância. Além disso, foi observado que se por um lado, há o sucesso empírico em gerar comportamentos complexos, como a agregação, com agentes de capacidade limitada, por outro, há a existência de limitações teóricas que impedem uma garantia formal de sucesso em todos os cenários. É nesta lacuna entre a viabilidade prática e a impossibilidade teórica que este trabalho se insere.

Portanto, este capítulo cumpre seu objetivo ao estabelecer o panorama conceitual, as ferramentas de análise e delimitar a lacuna de pesquisa a ser investigada. A partir disso, segue-se para o Capítulo 3, no qual são discutidos os trabalhos relacionados.

3 REVISÃO BIBLIOGRÁFICA E TRABALHOS RELACIONADOS

Neste capítulo, apresenta-se a metodologia de busca por trabalhos relevantes, seguida da revisão do material selecionado e da discussão dos estudos diretamente relacionados ao tema. Para isso, adotou-se uma abordagem sistemática, com critérios e métodos previamente definidos para a condução da revisão de literatura.

3.1 PESQUISA BIBLIOGRÁFICA SISTEMÁTICA

A partir dos princípios metodológicos recomendados por Marconi e Lakatos (2003), a presente pesquisa bibliográfica sistemática (PBS) foi elaborada com o objetivo de garantir uma análise ampla, rigorosa e replicável sobre o estado da arte em robótica de enxame, pois uma pesquisa bem estruturada permite levantar oportunidades de pesquisa, evitar redundâncias e identificar lacunas relevantes.

Quanto à classificação da pesquisa, esta PBS enquadra-se na grande área de Ciências Exatas e da Terra, abrangendo especificamente os campos da Ciência da Computação e da Engenharia Elétrica (Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), 2024). É uma revisão sistemática, com critérios de busca claros, etapas documentadas e reprodutibilidade. Quanto à classificação, é inexata, factual, dura e nomotética, lidando com fenômenos complexos, usando modelagem matemática e algoritmos, aplicando métodos rigorosos e buscando princípios replicáveis em enxames robóticos (Marconi; Lakatos, 2003). Em relação à natureza, é exploratória e tecnocrática, focando no impacto de variáveis de controle no comportamento coletivo e propondo soluções práticas. O raciocínio é hipotético-dedutivo, com dados quantitativos, oriundos de simulações e revisão bibliográfica, e variáveis contínuas, discretas, independentes e dependentes. A experimentação é feita por simulações computacionais, visando testar e validar modelos de forma controlada e reproduzível.

Com o intuito de orientar a revisão bibliográfica, foram formuladas as perguntas primárias da pesquisa com base na estratégia PICO (*Population/Problem, Intervention, Comparison, Outcome*), conforme descrito a seguir: grupos de robôs homogêneos (P), controle reativo (I), comparação entre controle reativo e outros modelos de controle (C) e impacto no comportamento coletivo (O).

A partir dessa estrutura, as seguintes perguntas foram definidas:

- Quais são os parâmetros que influenciam comportamentos multitarefa utilizando controle descentralizado?
- Quais são as principais abordagens e algoritmos utilizados para a modelagem e implementação de comportamentos coletivos em robótica de enxame?
- Quais são os principais desafios e as soluções propostas na literatura para garantir escalabilidade e robustez em sistemas de controle minimalistas?

E deste modo, essa PBS tem como objetivo principal identificar trabalhos de relevância científica sobre: Comportamentos emergentes e coletivos; Controle descentralizado e minimalista; Execução de múltiplas tarefas por grupos robóticos; Uso de sensores e capacidades computacionais limitadas.

Ademais, foi considerado o intervalo de publicação como 2010 a 2025, adotado com base no crescimento significativo da área de robótica de enxame a partir da última década, com o avanço de plataformas como *e-puck* (Mondada et al., 2009) e surgimento de novas abordagens bioinspiradas. Esse critério permite filtrar publicações maduras, recentes e relevantes.

Posteriormente, foi empregado o modelo 3W2H, também citado por (Marconi; Lakatos, 2003), para definir a *string* de busca conforme a seguir.

- *What?* Comportamentos coletivos: *clustering, task, behavior*;
- *Who?* Robôs: *swarm robotics, multi-robot systems, robot*;
- *How?* Controle minimalista auto-organizado: *minimalist controllers, binary sensors, without computation*;
- *When?* A partir de 2010.

Assim, para encontrar os materiais da literatura base desta pesquisa, foi selecionado como mecanismo de busca acadêmica (MBA), o *Google Scholar*. Ele foi empregado devido à sua capacidade de indexar diferentes bases de dados, garantindo uma vasta disponibilidade de materiais acadêmicos (Martín et al., 2018).

Com isso, sabendo que o *Google Scholar* necessita de operadores *booleanos* explícitos como *OR*, *AND* e aspas duplas, mas considera variações como singular e plural, verbos conjugados, sinônimos, desconsidera acentos e letras maiúsculas, faz casamento de termos compostos e considera ordens diferentes de palavras, ainda que com menor prioridade nesses dois últimos, foi definida a seguinte *string*:

("swarm robotics"OR "multi-robot"OR "multi-robot systems"OR "robot") AND ("clustering"OR "clusters"OR "self-organized") AND ("not compute"OR "without computation"OR "binary sensors"OR "minimalist controllers") AND ("task"OR "instance"OR "multi-task"OR "multi-instance"OR "behavior"OR "behaviour")

E sua variação, a fim de incluir artigos de revisão:

("swarm robotics"OR "multi-robot"OR "multi-robot systems"OR "robot") AND ("behavior"OR "behaviour") AND ("clustering"OR "clusters"OR "self-organized"OR "not compute"OR "without computation"OR "binary sensors"OR "minimalist controllers"OR "task"OR "instance"OR "multi-task"OR "multi-instance"OR) AND ("review"OR "survey")

Em seguida, foram realizadas três buscas, a primeira com a *string* original, a segunda também com ela, mas especificando explicitamente o intervalo desejado, 2010 a 2025, e a última com a variação para artigos de revisão. Isso foi feito para verificar se considerar o intervalo de

publicação no início ou ao final, como critério de exclusão, interferiria na busca, o que permitiu encontrar mais cinco materiais relevantes para a pesquisa.

Dessa forma, cada busca foi tratada individualmente, primeiro utilizando o *script* público (Wittmann, 2024), capaz de selecionar previamente os 100 materiais mais citados, e depois seguindo os critérios de inclusão:

- 10 mais citados (impacto consolidado);
- 10 mais citados por ano (novidade e relevância atual);
- 10 mais recentes (tendências emergentes);
- 10 mais relacionados à *string* (relevância semântica direta).

Então, foram aplicados os critérios de exclusão como:

- Materiais pagos, com opção de *download* ausente ou sem acesso completo (quando não disponíveis pelo portal de periódicos da CAPES);
- Teses, dissertações ou capítulos de livros;
- Publicações anteriores a 2010;
- Materiais com menos de 2 páginas;
- Material duplicado;
- Baixa correlação com a pesquisa a partir da leitura do título, resumo e conclusões.

Após a aplicação dos critérios de exclusão, restaram 28 artigos utilizados nesta revisão. Esses foram fichados com o auxílio de planilhas eletrônicas e *software* de gerenciamento de referências bibliográficas.

3.1.1 Revisão Bibliográfica

O aumento de especialização das tarefas desempenhadas por sistemas robóticos em ambientes reais tem impulsionado o desenvolvimento de abordagens distribuídas baseadas em múltiplos agentes autônomos. Nesse contexto, os sistemas de múltiplos robôs (*Multi-Robot Systems* – MRS) destacam-se como soluções tolerantes a falhas e escaláveis.

Essas características se devem ao controle descentralizado típicos em MRS, as quais diferem das abordagens centralizadas que, embora mais precisas, não favorecem a autonomia distribuída e a interação local. Entretanto, tais sistemas enfrentam desafios particulares, como a fragmentação da formação e dificuldades de adaptação a ambientes dinâmicos, principalmente em cenários com limitações de comunicação (Gautam; Mohan, 2012).

O crescimento desta área nas últimas décadas provem do potencial de aplicações em contextos civis e militares (Shahzad et al., 2023). Nesse cenário, SR é um subcampo dos MRS,

focado na coordenação de um grande número de robôs relativamente simples para atingir comportamentos coletivos complexos, inspirado em sistemas auto-organizados observados na natureza, como colônias de abelhas, cardumes de peixes e bandos de pássaros (Shahzad et al., 2023; Brambilla et al., 2013).

A partir da década de 1990, SR começou a ser pesquisado com publicações de trabalhos seminais e surgimento do termo *swarm robotics* em 1991, pouco utilizado até 2003 (Dorigo; Theraulaz; Trianni, 2021). Neste período, os primeiros testes robóticos já demonstravam auto-organização através de interações indiretas e locais, influenciadas pela inteligência de enxames, seu conceito de estigmergia (Brambilla et al., 2013).

Nos anos 2000, especialmente até 2005, a capacidade de criar enxames de robôs cooperativos expandiu-se para uma variedade de tarefas (Dorigo; Theraulaz; Trianni, 2021). O crescimento de publicação em SR nesse período, foi influenciado pelos avanços tecnológicos que permitiram a construção de robôs pequenos e relativamente baratos, como a plataforma robótica *e-puck* criada em 2009 (Mondada et al., 2009).

Os interesses mais comuns eram o desenvolvimento de modelos analíticos para direcionar a implementação robótica, a adoção de abordagens de otimização e o desenvolvimento de metodologias de *design* e verificação. Projetos como o *Swarm-bots*, desenvolvidos entre 2002 e 2006, demonstraram a capacidade de auto-montagem dos enxames, nos quais os robôs podiam formar cadeias para movimentar cargas ou estruturas em terrenos difíceis (Shahzad et al., 2023).

Em 2010, houve um crescente número de estudos descrevendo implementações reais de SR com novas plataformas como *Kilobots*, que permitiram experimentos em larga escala com até milhares de robôs. Abordagens de *design* automático, como a robótica de enxame evolucionária, ganharam destaque, porém ainda carecendo de uma prática consistente para comparar métodos e avaliar sua generalidade em diferentes tarefas (Francesca; Birattari, 2016). Ademais, novas aplicações foram apresentadas como em construção, monitoramento aquático e transporte cooperativo.

Recentemente, de 2020 a 2025, a pesquisa continua fortemente dedicada a controle de SR (Dorigo; Theraulaz; Trianni, 2021). Nesse aspecto, SR favorece abordagens distribuídas, nas quais os robôs operam com autonomia e interação localmente para coordenar o comportamento coletivo (Duan et al., 2022). Isso proporciona alguns temas diversificados que vêm sendo estudados.

Entre eles estão a alocação de tarefas (Chakraa et al., 2023), aprendizado de máquina, evolução para robótica de enxame (Kuckling, 2023), o consenso descentralizado (Fengying et al., 2024), os controles minimalistas para multi-tarefas (Krischanski et al., 2023), entre outros. Apesar de grandes avanços nas últimas décadas, a área ainda está limitada a laboratórios e simulações, sem aplicação prática em larga escala no mundo real (Dorigo; Theraulaz; Trianni, 2021).

Adicionalmente, embora o conhecimento do operador humano ainda desempenhe um papel relevante no processo de desenvolvimento de SR, observa-se uma tendência na adoção de

métodos semi-automáticos para o *design* do comportamento individual dos robôs, fundamentais para alcançar objetivos coletivos (Dorigo; Theraulaz; Trianni, 2021). Dentre esses métodos, destaca-se a robótica evolucionária, que utiliza estratégias evolutivas para desenvolver comportamentos coletivos e a neuroevolução, que evolui redes neurais para o controle de robôs (Kuckling, 2023).

Também há técnicas de *design* modular automático, como o *AutoMoDe*. (Francesca et al., 2015; Francesca; Birattari, 2016). Elas permitem a construção de *software* de controle a partir da composição de módulos comportamentais básicos e aprendizagem robótica, incluindo aprendizagem por reforço (Orr; Dutta, 2023) e aprendizagem por imitação (Kuckling, 2023).

Adicionalmente, acredita-se que as contribuições de SR influenciarão significativamente a trajetória do desenvolvimento tecnológico futuro (Devi et al., 2024) com a integração da robótica à vida cotidiana. Todavia, para que isso ocorra, é fundamental melhorar a engenharia do sistema em vários âmbitos, como modelagem de requisitos, *design*, realização, verificação, validação, operação, manutenção (Brambilla et al., 2013) e limitações de *hardware* e redes de comunicação (Dorigo; Theraulaz; Trianni, 2021).

Demais aplicações futuras devem exigir o uso mais intensivo de dados de sensores e partilha de informação, como por exemplo em sistemas industriais produtivos (Dorigo; Theraulaz; Trianni, 2021). Também há grandes potenciais para aplicações na agricultura (Albani et al., 2017), logística e armazéns, monitorização ambiental, busca e salvamento, infraestrutura, e na medicina com uso de enxames com nanorobôs para o tratamento de doenças (Shahzad et al., 2023).

Para o futuro próximo, a principal expectativa é que as publicações avancem para além das atuais provas de conceito, apresentando experimentações físicas que abordem problemas mais complexos, de larga escala e em cenários realistas. Esse avanço é muito importante para a definição de uma metodologia de engenharia confiável e eficiente.

Todavia, isso exigirá maior maturidade do campo, pois os desafios intrínsecos a SR, como o controle descentralizado, a auto-organização e o problema micro-macro continuarão sendo desafios a serem considerados (Dorigo; Theraulaz; Trianni, 2021). A partir deste panorama apresentado, o capítulo avança para a discussão de trabalhos relacionados.

3.2 TRABALHOS RELACIONADOS

Na presente Seção é desenvolvida uma discussão sobre trabalhos que formam a base desta pesquisa. As contribuições e limitações dos estudos são analisadas para contextualizar o problema da agregação minimalista e fundamentar a necessidade da solução proposta. A análise se concentra na linha de pesquisa investigada por Gauci et al. (2014a), suas limitações teóricas e sua generalização para cenários multitarefa.

3.2.1 Agregação Sem Computação

Gauci et al. (2014) investigam a agregação auto-organizada de um enxame com robôs sem nenhuma informação do ambiente e que utilizam apenas sensores binários de linha de visão, comparando longo alcance sensorial com o processamento mínimo de informação. A abordagem utiliza robótica evolucionária para otimizar dois tipos de controladores, um reativo, que sem uso de memória responde a leitura do sensor, e um controlador recorrente, com uso de memória usando rede neural recorrente de 2 neurônios.

Em suas simulações, o autor mostra que ambos os controladores podem alcançar agregação completa se o alcance do sensor for suficiente, com o controlador recorrente apresentando melhor desempenho e escalabilidade para até 1000 robôs. Também foi observado que o ruído falso negativo no sensor pode acelerar a agregação com o controlador recorrente.

Ademais, o trabalho foi validado por experimentos com 20 robôs físicos *e-puck*, alcançando uma taxa de agregação média de 86-89% em 10 minutos para ambos os controladores. Assim, este trabalho demonstra a viabilidade da agregação com sensores extremamente simples, destacando o potencial para sistemas robóticos de menor escala.

Já em Gauci et al. (2014b), a mesma abordagem de controle reativo do trabalho anterior é utilizada, também sendo sintetizado usando robótica evolucionária, a partir da Estratégia de Evolução de Adaptação de Matriz de Covariância (CMA-ES). Entretanto, agora o objetivo é que robôs agrupem objetos inicialmente espalhados pelo ambiente.

A pesquisa compara o desempenho quando os robôs podem distinguir entre objetos e outros robôs e quando não podem, ou seja, interpretam outros robôs como nada. O autor conclui que o agrupamento é possível mesmo sem distinguir robôs de objetos, mas observa que o desempenho melhora significativamente quando essa distinção é feita. Além disso, os controladores evoluídos demonstraram boa robustez a ruído sensorial e mostraram-se escaláveis com o aumento do número de robôs e objetos.

Para realizar as simulações foi utilizado o simulador Enki, enquanto para validação foram realizados 10 experimentos físicos com 20 objetos cilíndricos e 5 robôs *e-puck*, equipados apenas com o sensor de linha de visão. Obteve-se uma média de 86,5% dos objetos em um único agrupamento após 10 minutos. Desta forma, o trabalho demonstrou mais uma vez que tarefas coletivas complexas podem ser realizadas com processamento de informação mínimo.

Em Gauci et al. (2014a), também há implementação de uma solução minimalista para a agregação auto-organizada para SR, eliminando a necessidade de computação aritmética e uso de memória. Novamente os robôs utilizam apenas um sensor binário de linha de visão e um controlador reativo simples.

Neste trabalho, é apresentada a prova teórica de que, para garantir a agregação, o sensor precisa ter um alcance suficientemente longo, pois, caso contrário, a agregação não pode ser garantida, independentemente do controlador utilizado. Embora que, teoricamente, o alcance finito possa falhar, os resultados experimentais sugerem que a solução não é comprometida por

essa limitação no mundo real.

Para mostrar isso, os autores utilizam um argumento lógico: dividem todos os comportamentos possíveis do robô em quatro categorias (parado, girando no lugar, movendo-se para frente ou para trás) e analisam todas as 16 combinações de controle possíveis. Para cada combinação eles demonstram a existência de configurações iniciais patológicas nas quais os robôs, inevitavelmente, se separam, mesmo que comecem conectados. Provando assim que a agregação não pode ser garantida com um sensor de alcance finito.

Dessa forma, os autores utilizaram uma estrutura de controle que realiza o mapeamento da leitura do sensor I para um par $(\bar{v}_\ell, \bar{v}_r) \in [-1, 1]^2$ de velocidades angulares normalizadas das rodas diferenciais esquerda e direita do robô, respectivamente quando $I = i$. Assim, as leituras do sensor são $I = 0$ quando o robô identifica parede (ou nada) e $I = 1$ quando encontra outro robô. Ademais, valores negativos indicam movimento para trás, enquanto valores positivos indicam movimento para frente.

Para encontrar o controlador ótimo, ou seja, aquele que permite a agregação do enxame, foi utilizada a estratégia *grid search*. Com base na faixa de velocidades que o robô *e-puck* assume, -12,8 a 12,8 cm/s, os autores normalizaram estes valores e definiram 21 valores possíveis para cada um dos quatro parâmetros de controle, variando de -1 a $+1$ com incrementos de 0,1. A combinação de todas as possibilidades resultou em um espaço de busca contendo 194.481 controladores distintos, ou seja, 21^4 .

Para avaliar cada controlador, foram realizadas 100 simulações, cada uma com 10 robôs posicionados aleatoriamente no ambiente e duração de 180 s. O desempenho em cada simulação foi medido por uma métrica de dispersão, função de custo, projetada para recompensar tanto a velocidade de agregação dos robôs quanto a compactação do agrupamento final. A pontuação final de cada controlador correspondia à média dos resultados obtidos nas 100 simulações.

A função de custo, $d(t)$, utilizada é apresentada a seguir. Considerando r robôs, $p_i(t)$ é a posição do robô i no tempo t e $\bar{p}(t)$ é o centroide da posição dos robôs no tempo t .

$$d(t) = \frac{1}{r} \sum_{i=1}^r \|p_i(t) - \bar{p}(t)\| \quad (1)$$

O desempenho do controlador em uma simulação, $U(\bar{v})$, é obtido pela soma ponderada de $d(t)$ ao longo de T intervalos de tempo:

$$U(\bar{v}) = \sum_{t=1}^{T-1} t \cdot d(t) \quad (2)$$

Conforme Gauci et al. (2014), otimizar a Equação 2, buscando seu menor valor possível, visa reduzir a dispersão entre os robôs no tempo final, bem como promover uma rápida convergência. Isso é alcançado por meio da atribuição de pesos crescentes a $d(t)$ ao longo do tempo t .

Essa ponderação confere duas propriedades desejáveis à métrica: penaliza formações que não convergem ou não se mantêm estáveis, e favorece aquelas que convergem rapidamente.

Assim, após todos os controladores serem testados e avaliados, os autores selecionaram aquele que obteve a melhor pontuação média. A partir disso, os autores encontraram o controlador ótimo:

$$\bar{\mathbf{v}}_{\beta} = (\bar{v}_{\ell,0}, \bar{v}_{r,0}, \bar{v}_{\ell,1}, \bar{v}_{r,1}) = (-0,70; -1,00; 1,00; -1,00), \quad \bar{\mathbf{v}} \in [-1, 1]^4 \quad (3)$$

Onde:

- $v_{\ell,i}$ é a velocidade normalizada da roda diferencial esquerda;
- $v_{r,i}$ é a velocidade normalizada da roda diferencial direita;
- i é a leitura do sensor, onde $i \in \{0, 1\}$.

Com o controlador definido por Gauci et al. (2014a), é fornecida uma prova matemática formal de que ele sempre agrega um robô a um agrupamento circular estático ou dois robôs em movimento em tempo finito, com um limite superior de tempo fornecido. A prova formal para a agregação de um robô móvel com um alvo estático, se baseia em análise geométrica e relações de recorrência.

Para isso, os autores modelam o sistema em um plano 2D, utilizando um sistema de coordenadas para definir as posições do robô e de seu centro de curvatura e, através de manipulação algébrica e trigonométrica, eles derivam uma equação que demonstra como a distância entre o robô e o alvo diminui de forma previsível e garantida a cada ciclo de interação. Essa relação é então expressa como uma recorrência matemática, mostrando que o robô se aproximará do alvo até a agregação ser inevitável.

Posteriormente, os autores realizaram uma nova fase de simulações para validar o desempenho e a robustez da solução encontrada. Nestes experimentos, era utilizado um grupo de 100 robôs, os quais eram inicializados em posições aleatórias dentro de um ambiente virtual quadrado, mantendo-se uma densidade constante. Diferente da fase de otimização, estas simulações de validação foram executadas por 1800 s, para permitir a observação completa do comportamento do enxame. Foi observado que o robô gira sobre si mesmo ao detectar outro robô e, caso contrário, move-se para trás em trajetória circular.

Por fim, as simulações demonstraram que o método é robusto a ruídos no sensor e a uma redução de até 70% no alcance do sensor, além de ser escalável, agregando corretamente até 1000 robôs. A validação em experimentos físicos com 40 robôs *e-puck* confirmou a viabilidade da solução, alcançando uma taxa de agregação de 98,6%, e revelou que o processo depende de um movimento coletivo complexo do agrupamento, semelhante à convecção.

3.2.2 Limitações Teóricas da Agregação Bimodal

Apesar dos bons resultados dos trabalhos discutidos na Subseção 3.2.1, com controlador reativo comprovadamente funcional para $r = 2$ robôs, Daymude et al. (2021) questionam a afirmação de que o enxame sempre agrega para sistemas com $r > 2$. Os autores alegam que existem configurações de *deadlock*, impasse em tradução direta, a partir das quais o algoritmo não consegue atingir a agregação para $r > 3$ robôs, quando o movimento dos robôs é uniforme e determinístico.

Entretanto, no mundo real, esses *deadlocks* não são observados na prática devido ao ruído inerente nos robôs físicos, como colisões e escorregamentos, que perturbam o sistema e permitem que os robôs se desprendam, mas quantidades excessivas de ruído também podem interferir significativamente na agregação. Ademais, Daymude et al. (2021) provam que o uso de um sensor em formato de cone de visão, em vez de linha de visão, acelera linearmente a agregação para $n = 2$, embora essa vantagem diminua para enxames maiores. Aqui vale observar que alguns autores divergem quanto ao número mínimo de robôs para considerar um enxame robótico, vide Seção 2.1.

Houve também adaptação discreta e ruidosa do algoritmo introduzida para análise rigorosa, mas sua complexidade teórica permanece um desafio. Assim, o trabalho identifica limitações teóricas do algoritmo original, mas reforça sua robustez prática devido ao ruído inerente e explora alternativas para melhoria.

Já Steinberg e Solovey (2024) exploram ainda mais esta limitação, provando que nenhum controlador bimodal pode alcançar a agregação para todos os $n > 2$, mostrando que o desafio não se limita apenas ao controlador proposto por Gauci et al. (2014a). Também identificam uma suposição implícita e irrazoável na prova de agregação para dois robôs de Gauci et al. (2014a), tornando-a incompleta, pois robôs podem estar em uma configuração onde a distância para agregação é atendida, mas eles nunca se veem e por isso não se agregam.

Para solucionar isso, Steinberg e Solovey (2024) apresentam um controlador alternativo, $\bar{\mathbf{u}} = (-a, a, b, b)$, no qual o robô gira no local quando não vê outro e se move para frente quando vê, além de fornecerem uma prova simples e rigorosa de agregação para $r = 2$ robôs. Em testes empíricos, este controlador $\bar{\mathbf{u}}$ superou o de Gauci et al. (2014a), que falhou em 4,24% dos testes para $r = 2$, além disso, uma versão revisada de Gauci et al. (2014a) alcançou 100% de sucesso para $r = 2$.

Ademais, provou-se matematicamente que as configurações iniciais que levam à não agregação para $r > 2$ robôs não são singulares, ou seja, a probabilidade de não agregação é maior que zero mesmo quando as posições iniciais são escolhidas aleatoriamente com pequenas perturbações, isso reforça que os *deadlocks* não são eventos raros que exigem configurações precisas. Assim, os resultados empíricos de Steinberg e Solovey (2024) também sugerem que a introdução de dinâmicas mais precisas como escorregamento e ruído não melhoram a situação para sistemas com $r > 2$, e que a porcentagem de agregação geralmente diminui com o aumento

do número de robôs.

Em resumo, enquanto Gauci et al. (2014a) introduziram um conceito inovador de agregação sem computação complexa, que se mostrou eficaz para $r = 2$ robôs e em muitos casos para enxames maiores em condições ideais, sendo beneficiados do ruído do sistema, os trabalhos de Daymude et al. (2021) e Steinberg e Solovey (2024) forneceram provas matemáticas sobre as limitações intrínsecas desses controladores minimalistas para enxames $r > 2$, identificando *deadlocks* e mostrando que a agregação universal é impossível sem capacidades adicionais dos agentes.

3.2.3 Generalização para Agregação Multitarefa

Em seu trabalho Krischanski et al. (2023), generaliza tarefas de agregação de robôs e agrupamento de objetos em SR. Isso permite que múltiplas instâncias da mesma tarefa, diferenciadas por cor, sejam executadas simultaneamente no mesmo ambiente virtual. O objetivo dos autores é fazer com que os robôs de diferentes grupos se organizem em agrupamentos distintos ou agrupem objetos de sua cor em locais separados da arena, a qual corresponde ao ambiente da simulação com formato quadrado.

Assim como no trabalho de Gauci et al. (2014a), é implementado um controle reativo, no qual os robôs operam sem memória, ou seja, não armazenam estados nem leituras passadas. Suas ações são determinadas exclusivamente pela leitura atual do sensor de visão I . A diferença entre os dois trabalhos consiste no fato de que Krischanski et al. (2023), ao generalizar as tarefas, adiciona novos pares de velocidades e propõe duas estruturas de controle para agregação e cinco para o agrupamento de objetos. Ainda, a Equação 1 é adaptada para considerar g grupos, conforme segue:

$$d(t) = \sum_{k=1}^g \max(\|p_i(t) - p_j(t)\| : i, j \in G_k) \quad (4)$$

Na qual os r robôs são divididos em g grupos, seja $G_k : k \in \{1, \dots, g\}$ o conjunto de robôs pertencentes ao grupo k . As posições dos robôs i e j no tempo t são representadas por $p_i(t)$ e $p_j(t)$, respectivamente.

A partir de experimentação, os autores validam as melhores estruturas de controle para cada caso. Para otimização dos parâmetros dos controladores é utilizado o algoritmo de otimização global DIRECT e uma função de custo, Equações 2 e 4, adaptadas de Gauci et al. (2014a). Após otimizados, os controladores são validados em simulações baseadas em física utilizando o *enki* com modelo de robô *e-puck*.

Neste trabalho, o controlador $\bar{v}_\beta$ foi generalizado para cenários de múltiplos grupos. Em configurações específicas, a estrutura de controle foi expandida com a adição de pares de velocidades correspondentes a novas leituras do sensor I . Além das leituras anteriores, $I = 0$ para ausência de obstáculos e $I = 1$ para robôs ou objetos do mesmo grupo, o sistema foi ampliado

para distinguir robôs de outros grupos ($I = 2$) e/ou objetos de outros grupos ($I = 3$). Isso pode ser observada nas Equações 5 e 6, que são equivalentes aos controladores $\bar{v}_4$ e $\bar{v}_2$ propostos por Krischanski et al. (2023).

Nos experimentos de agregação de robôs, realizados com 30 agentes divididos em 3 grupos, o controlador $\bar{v}_{\alpha_2}$ (equivalente ao $\bar{v}_2$ de Krischanski et al. (2023)) apresentou desempenho superior ao controlador equivalente a $\bar{v}_\beta$. Sendo capaz de distinguir robôs de grupos distintos, esta abordagem alcançou, em testes preliminares, uma média de 99,83% de robôs corretamente agrupados após 300 s.

Quanto ao agrupamento de objetos, as simulações com 15 robôs e 30 objetos (em 3 grupos) indicaram que os controladores $\bar{v}_4$ e $\bar{v}_6$ (Krishanski et al., 2023) possuem os melhores desempenhos globais, equilibrando baixa dispersão e alta proporção de agrupamento. Entretanto, o controlador $\bar{v}_{\alpha_1}$ ($\bar{v}_4$) mostrou-se preferível por exigir menor complexidade sensorial, atingindo uma taxa de 97,73% de objetos agrupados ao final de 1800 s.

Assim, os parâmetros do controlador para agrupamento de objetos de Krischanski et al. (2023) são dados pela Equação 5 e para agregação de robôs dados pela Equação 6, conforme:

$$\bar{v}_{\alpha_1} = (\bar{v}_{\ell,0}, \bar{v}_{r,0}, \bar{v}_{\ell,1}, \bar{v}_{r,1}, \bar{v}_{\ell,2}, \bar{v}_{r,2}), \quad \bar{v}_{\alpha_1} \in [-1, 1]^6 \quad (5)$$

$$\bar{v}_{\alpha_2} = (\bar{v}_{\ell,0}, \bar{v}_{r,0}, \bar{v}_{\ell,1}, \bar{v}_{r,1}, \bar{v}_{\ell,3}, \bar{v}_{r,3}), \quad \bar{v}_{\alpha_2} \in [-1, 1]^6 \quad (6)$$

Onde:

- $v_{\ell,i}$ é a velocidade normalizada da roda diferencial esquerda;
- $v_{r,i}$ é a velocidade normalizada da roda diferencial direita;
- i é a leitura do sensor, onde $i \in \{0, 1, 2, 3\}$.

Ademais, para orientar a formação dos aglomerados em locais específicos da arena, foram utilizados cilindros fixos de referência. Em relação à escalabilidade, a tarefa de agregação de robôs não apresentou bom desempenho com o aumento do número de robôs por grupo, uma vez que o comportamento de movimento circular otimizado se mostrou adequado apenas para uma quantidade limitada de robôs, aproximadamente 10, tendo seu desempenho comprometido em cenários com grupos mais numerosos.

Além disso, a agregação não foi capaz de gerar aglomerados compactos, mas sim um formato circular, em que os robôs formam um anel. Por outro lado, a tarefa de agrupamento de objetos demonstrou boa escalabilidade, com bom desempenho à medida que o número de robôs aumentava para manipular uma quantidade fixa de objetos.

Vale destacar as métricas de desempenho utilizadas pelos autores, já que elas são essenciais para quantificar e avaliar objetivamente a eficácia do comportamento coletivo. Com isso, as

métricas de dispersão e proporção de agrupamentos de Krischanski et al. (2023) são apresentadas na Seção 3.2.3.1.

3.2.3.1 Dispersão

Uma baixa dispersão em cada grupo indica que o objetivo de formar grupos compactos de robôs foi alcançado e o segundo momento é uma medida da dispersão dos elementos.

Os autores utilizaram o segundo momento baseado em grupo dos robôs em relação ao centróide de seu grupo, como medida de sua dispersão em cada instante discreto de tempo t . Dessa forma, seja o centróide $\bar{\mathbf{p}}_k^{(t)}$ dos n_k robôs do grupo k no instante de tempo discreto t definido por:

$$\bar{\mathbf{p}}_k^{(t)} = \frac{1}{n_k} \sum_{i=1}^r \mathbf{p}_i^{(t)} : i \in G_k. \quad (7)$$

Onde:

- $\mathbf{p}_i^{(t)}$ é a posição do elemento i no tempo t ;
- G_k é o conjunto de robôs do grupo k .

Note-se que todos os robôs do sistema estão distribuídos entre os g grupos, e a soma das quantidades em cada grupo dá o total geral de robôs:

$$r = \sum_{k=1}^g n_k. \quad (8)$$

Onde:

- r é o número total de robôs no sistema;
- g é o número total de grupos.

Seja r_o o raio dos robôs, o segundo momento baseado em grupo, $u^{(t)}$, no instante de tempo t é dado por:

$$u^{(t)} = \frac{1}{4r_o^2} \sum_{i=1}^r \|\mathbf{p}_i^{(t)} - \bar{\mathbf{p}}_k^{(t)}\|^2 : i \in G_k. \quad (9)$$

3.2.3.2 Proporção de Agrupamento

Já a proporção de robôs agrupados, $pc_k^{(t)}$, do grupo k indica quantos robôs já estão agrupados entre si no tempo t . Esta é definida pela razão entre o maior grafo de robôs conectados no grupo, $lc_k^{(t)}$ e o número total de robôs no grupo, n_k , dada por:

$$pc_k^{(t)} = \frac{lc_k^{(t)}}{n_k} \quad (10)$$

Dois robôs, i e j , do mesmo grupo k (ou seja, $i, j \in G_k$), são considerados conectados se não for possível que outro robô se encaixe entre eles. Isso significa que a distância entre eles é menor que quatro vezes o raio dos robôs (isto é, $|\mathbf{p}_i^{(t)} - \mathbf{p}_j^{(t)}| < 4ro$).

Sendo r o número total de robôs (de todos os grupos), a proporção de robôs agrupados de todos os grupos, $pc^{(t)}$, no tempo t , é dada por:

$$pc^{(t)} = \frac{1}{r} \sum_{k=1}^g lc_k^{(t)} \quad (11)$$

3.2.4 Outras Investigações em Controle Minimalista

Outros autores também investigaram controle reativo minimalista em SR. Mitrano et al. (2019) aborda o problema da segregação N-classes, separação espacial de robôs em N grupos distintos baseados em uma característica comum em SR focando em alcançar esse comportamento com princípios minimalistas.

Os autores propõem um algoritmo descentralizado em que os robôs utilizam apenas um sensor ternário de linha de visão. Ele é capaz de detectar a presença do robô mais próximo e indicar se ele é da mesma classe, de uma classe diferente ou se não há robôs na linha de visão, mas sem fornecer a distância e sem distinguir as classes.

Assim, seu controlador é reativo, mapeando diretamente as três leituras do sensor ($I = 0, 1, 2$) para as velocidades das rodas diferenciais esquerda e direita do robô por meio de uma estrutura simples, sem memória ou computação complexa. Também é realizada uma busca exaustiva (*grid search*) no espaço de parâmetros de 6 dimensões, ou seja, três pares de velocidades, usando simulações pelo simulador ARGoS e robôs *foot-bot*.

Neste trabalho, o comportamento investigado foi a segregação, e os resultados mostraram que a dinâmica que causa a convergência para o melhor conjunto de parâmetros encontrado, movimentos circulares específicos que promovem a adesão entre mesma classe e repulsão de classes distintas. Além de avaliarem a robustez do comportamento em cenários não ideais como ângulos de abertura finitos do sensor e diferentes configurações dos robôs.

Já Özdemir et al. (2018), apresenta uma estratégia de controle para o problema de escolha coletiva em SR. Os robôs *e-puck* físicos ou simulados virtualmente, não possuem nenhum tipo de comunicação, não armazenam informações em tempo de execução e seu controlador também é dado pela leituras do sensor de linha de visão ternário, podendo mapear opção, outro agente ou nada.

A metodologia utilizou a otimização *off-line* usando CMA-ES para determinar os seis parâmetros do controlador. Este processo, tem alto custo computacional, mas é realizado apenas

uma vez, não impactando a simplicidade do controlador em tempo de execução. A eficácia da estratégia foi avaliada por meio de 1.000 simulações com 20 robôs, e validada com 50 experimentos físicos usando 20 robôs *e-puck*.

Na simulação, o consenso foi atingido em 97,3% dos testes e nos experimentos físicos, em 96%, mesmo com falhas mecânicas que fizeram alguns robôs pararem de se mover. O controlador mostrou-se robusto contra ruído sensorial, mantendo bom desempenho quando o ruído afetava apenas a percepção das opções ou apenas a percepção de outros robôs.

Entretanto, o desempenho caiu rapidamente quando ambos os tipos de percepção foram afetados simultaneamente. A estratégia também se mostrou escalável, funcionando com até 100 robôs e atingindo consenso médio de 88,5%. Embora projetado para duas opções idênticas, o controlador conseguiu generalizar para cenários com mais de duas opções e para opções de tamanhos desiguais, preferindo repetidamente a maior.

Além disso, um comportamento emergente observado foi o movimento em órbita dos robôs ao redor da opção escolhida, o que favoreceu o consenso, os robôs giravam à esquerda ao detectar uma opção, à direita quando não detectavam nada, e seguiam em frente ao detectar outro robô, e esse padrão os levava a circular a opção. Também foi observado que a estratégia depende que o alcance do sensor seja suficiente, pois não funciona bem com sensoriamento estritamente local.

No trabalho Chen et al. (2012) é explorada a segregação espacial auto-organizada em enxames de robôs físicos. Nela os indivíduos separam-se em grupos distintos para formar padrões anelares ou de centro-periferia, sem que haja comunicação entre eles. A abordagem é diretamente inspirada no efeito *Brazil Nut*, um fenômeno natural em que partículas de diferentes tamanhos se separam num meio vibrante, com as maiores a tenderem a subir.

Em SR, mesmo sendo os agentes fisicamente idênticos, eles simulam discos de tamanhos virtuais diferentes e o movimento de cada robô é comandado por uma combinação de três componentes: atração em direção a um ponto central, movimento aleatório e repulsão de robôs próximos. O algoritmo implementado não exige que os robôs distingam entre robôs de grupos diferentes para alcançar a segregação, isso emerge das interações dessas propriedades elencadas, mas realiza cálculos aritméticos e processamento de informações no tempo de execução para determinar as forças e movimentos, diferenciando-o das abordagens sem computação.

A implementação utiliza robôs *e-puck* físicos, equipados com câmeras direcionais e sensores infravermelhos. Para obter uma visão omnidirecional, os robôs precisam girar e capturar várias imagens, o que aumenta o ciclo de controle em comparação com simulações anteriores. A visão baseada na câmera gera desafios como percepções erradas e para corrigir, o mecanismo de repulsão foi modificado em relação a simulação para priorizar os agentes mais próximos.

Nos experimentos realizados com um grupo de 20 robôs *e-puck* físicos em 100 testes de 20 minutos cada, verificou-se a qualidade da segregação. Isso foi feito medindo erro de segregação que conta o número de pares de discos incorretamente ordenados em relação à sua distância do ponto central.

Os resultados mostraram um desempenho muito preciso. Para os testes em que um grupo de robôs foi programado para simular um raio virtual 3, 4 ou 5 vezes maior que o outro (referido como fator de raio b), o erro médio de segregação foi inferior a 0,5%. A maioria dos erros observados em testes físicos foi atribuída a falhas mecânicas dos robôs, como parar de se mover, os robôs que representavam discos maiores se posicionaram espacialmente mais distantes do centro, confirmando o efeito esperado.

Por fim, Özdemir et al. (2019) abordam o problema de controlar um SR para cobrir cooperativamente um espaço bidimensional desconhecido, com cobertura contínua da área. Novamente os robôs não possuem comunicação entre si, computação ou armazenamento em tempo de execução, não conseguem localizar-se ou ter informações do ambiente ou do tamanho do enxame. O controlador é determinístico, mapeando diretamente a leitura binária do sensor para um par de velocidades das rodas diferenciais, suas leituras são se outro robô está presente ou não em sua linha de visão direta.

Este controlador foi obtido por meio de otimização *off-line*, também utilizando uma abordagem de robótica evolutiva (CMA-ES) e seu desempenho foi avaliado pelas métricas cobertura de células e cobertura de área. Para isso, foram realizadas simulações com 1000 testes com 25 robôs, comparando o controlador com outras estratégias e validando a solução com 20 experimentos físicos usando 25 robôs *e-puck* reais em um ambiente de 300 cm x 300 cm.

Assim os resultados de simulação mostraram que o controlador atingiu entre 71% e 76% de cobertura em um cenário sem redundância de robôs, sendo a cobertura média de célula de 76,0% e a cobertura de área de 71,2%. Isso representa uma redução de 26% a 39% na área não coberta quando comparado a uma caminhada aleatória otimizada. Ademais, o controlador exibiu robustez ao ruído sensorial, mantendo um desempenho superior a 60% de cobertura mesmo com um nível de 100% de ruído do tipo falso-negativo ou falso-positivo.

Todavia, seu desempenho degradou mais rapidamente na presença de ruído combinado, que afeta ambas as leituras do sensor, caindo para aproximadamente 20% com um nível de ruído de 100%. Para escalabilidade, o desempenho médio melhorou em enxames com até 40 robôs, depois estabilizou, apresentando uma pequena diminuição, e voltou a melhorar para 50 ou mais robôs.

Então, a generalidade da abordagem foi demonstrada pela eficácia do mesmo controlador em ambientes com diferentes formatos e dimensões e até mesmo na navegação de um labirinto. A análise da distribuição espacial dos robôs indicou que eles tendem a se concentrar perto dos limites do ambiente, principalmente nos cantos. Já para os experimentos com robôs físicos, foi observada uma queda moderada no desempenho em comparação as simulações, alcançando 65,2% de cobertura de célula e 64,9% de cobertura de área, apesar disso, os resultados ainda foram superiores aos *benchmarks* da caminhada aleatória.

3.3 CONSIDERAÇÕES DO CAPÍTULO

A revisão da literatura e a discussão de trabalhos relacionados demonstram SR como um campo de pesquisa amplo e promissor. Com destaque para a linha que investiga o potencial de controladores minimalistas para gerar comportamentos coletivos complexos, que se mostra ainda pouco explorada.

O trabalho seminal de Gauci et al. (2014a) estabeleceu a viabilidade da agregação e do agrupamento de objetos sem a necessidade de computação complexa ou memória, utilizando apenas interações locais e sensores simples. Essa abordagem minimalista foi subsequentemente validada e expandida por outros pesquisadores para tarefas como segregação N-classes e tomada de decisão coletiva.

No entanto, outras análises mostraram limitações teóricas importantes. Conforme apontado por Daymude et al. (2021) e Steinberg e Solovey (2024), os controladores bimodais propostos, embora eficazes na prática, não podem garantir matematicamente a agregação para enxames com mais de dois robôs ($r > 2$) devido à existência de configurações de impasse (*deadlock*). Isso evidencia uma lacuna entre a robustez observada experimentalmente, muitas vezes auxiliada pelo ruído inerente dos sistemas físicos, e a comprovação formal da convergência do sistema.

O trabalho de Krischanski et al. (2023) representa um avanço natural, generalizando a abordagem para cenários multitarefa, nos quais diferentes grupos de robôs ou objetos devem ser organizados simultaneamente. Contudo, essa generalização apresentou novas limitações, pois a agregação de robôs resultante não produziu agrupamentos compactos, mas sim formações circulares e foi pouco escalável.

Diante disso, neste trabalho se busca preencher as lacunas identificadas. Assim, busca-se superar as limitações de Krischanski et al. (2023), a partir da combinação das estratégias discutidas até aqui: *grid search* e algoritmos globais e locais associados (abordagem híbrida). Estas propostas são detalhadas a seguir, no Capítulo 4.

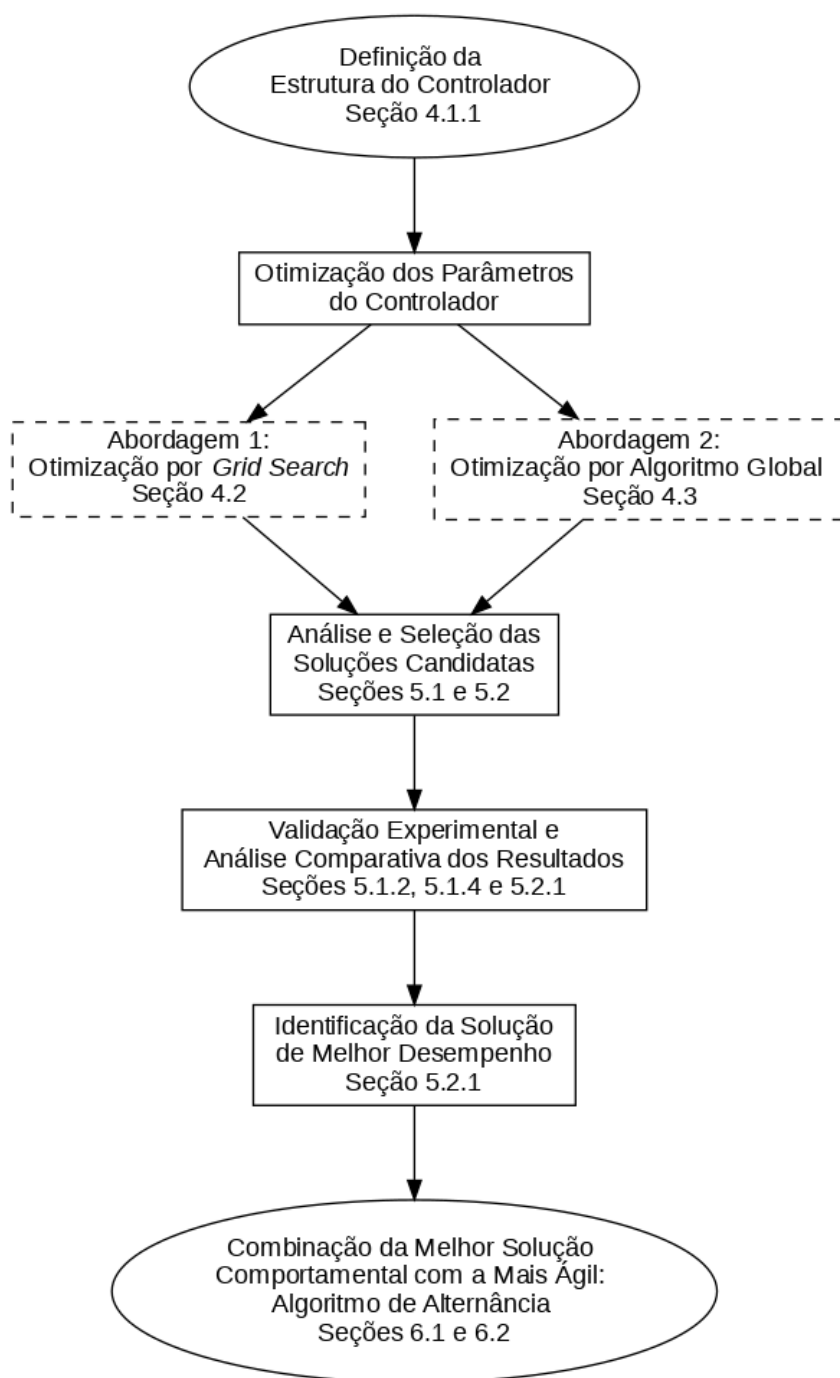
4 MÉTODO DE SOLUÇÃO PROPOSTO

Este capítulo detalha a metodologia empregada para investigar e aprimorar o comportamento de autoagregação multitarefa em enxames de robôs. O objetivo central foi desenvolver e validar configurações de controle minimalista capazes de produzir agregados compactos, superando as formações circulares.

Para isso, a pesquisa uniu as abordagens de Gauci et al. (2014a) e Krischanski et al. (2023), estruturando a investigação em duas fases experimentais principais, ambas executadas em ambiente de simulação. A primeira é baseada em busca exaustiva em grade (*grid search*) e a segunda em otimização automática com múltiplos algoritmos de busca global e local.

Estas duas fases serão comparadas, a fim de investigar qual delas promove a melhor solução. Todavia, deve-se considerar o fato de que a *grid search* explora um espaço de busca muito menor que os algoritmos de otimização automática, devido a sua natureza exaustiva e seu alto custo computacional. Na Figura 4, o trabalho é apresentado de maneira resumida. Nas próximas seções há o detalhamento de cada etapa.

Figura 4 – Etapas da Pesquisa



Fonte: A autora (2025).

4.1 ESTRUTURA DE CONTROLE E AMBIENTE DE SIMULAÇÃO

Ambas as fases experimentais deste trabalho compartilharam a mesma base de controle e o mesmo ambiente de simulação, garantindo que os resultados possam ser comparados quantitativamente. Nas próximas seções são apresentadas essas especificações.

4.1.1 Estrutura do Controlador

Neste trabalho foi utilizada a estrutura de controle apresentada na Equação 6. Nela há três pares de velocidades para as rodas diferenciais esquerda e direita, $(\bar{v}_{l,i}, \bar{v}_{r,i})$, um para cada estado i do sensor.

Os robôs são capazes de identificar: parede ou nada ($I = 0$), robô do mesmo grupo ($I = 1$) e robô de um grupo diferente ($I = 2$). Optou-se por chamar a leitura para robô de um grupo diferente como $I = 2$ e não como $I = 3$, como feito em Krischanski et al. (2023), para facilitar a compreensão do leitor, já que na prática são equivalentes, conforme discutido no capítulo anterior.

Quanto aos valores de velocidade possíveis assumidos por cada parâmetro, eles variam dentro do intervalo de -12,8 a 12,8 cm/s. Isso se justifica por considerar a faixa de operação da plataforma robótica escolhida, o *e-puck*.

O objetivo da otimização é encontrar a combinação dos parâmetros que resultem no comportamento de agregação para múltiplos grupos simultaneamente. Dessa forma, para avaliar as soluções, é empregada a mesma função de custo utilizada por Gauci et al. (2014a), Equações 1 e 2, a qual avalia como melhores desempenhos os valores mais baixos da função.

4.1.2 Métricas de Desempenho

Para quantificar e avaliar de modo objetivo a eficácia dos resultados encontrados nas simulações, foram aplicadas três métricas que permitem analisar a compactação dos grupos, a conectividade entre os robôs e a eficiência da convergência ao longo do tempo.

- **Função de Custo:** Utilizada para orientar os processos de otimização, esta função avalia o desempenho global ao longo de T intervalos de tempo. Ela utiliza uma soma ponderada da distância média dos robôs ao centroide para cada grupo, penalizando formações instáveis e favorecendo convergências rápidas por meio de pesos crescentes ao longo do tempo, conforme a Equação 2. Nesta aplicação multitarefa, $d(t)$ é adaptada para considerar g grupos, medindo a extensão máxima entre robôs de um mesmo conjunto, conforme a Equação 4;
- **Dispersão:** Esta métrica mensura o quão espalhados os robôs de um mesmo grupo estão em relação ao centroide de seu grupo em cada instante t . Uma baixa dispersão indica que o objetivo de gerar agregados compactos foi atingido, vide Equação 9;
- **Proporção de Agrupamento:** Indica a fração de robôs que já estão agregados entre si no tempo t . É calculada pela razão entre o maior grafo de robôs conectados no grupo ($lc_k^{(t)}$) e o número total de robôs no grupo (n_k). Para que dois robôs sejam considerados conectados, a distância entre eles deve ser menor que quatro vezes o raio do robô ($\|p_i^{(t)} - p_j^{(t)}\| < 4r_o$), impedindo que outro agente se encaixe entre eles. Para o enxame total de r robôs, a proporção global é dada pela Equação 11.

4.1.3 Configuração do Ambiente de Simulação

Em relação as simulações, foi empregada a plataforma *Enki*, a qual trata-se de uma ferramenta de código aberto desenvolvida em C++ que simula a cinemática e dinâmica de corpos rígidos em 2D e foi escolhido por possuir o modelo virtual do robô *e-puck* disponível em sua biblioteca (Magenat; Waibel; Beyeler, 2005), conforme descrito na Seção 2.3. Assim, implementados deste trabalho foram escritas em linguagem C++, mas para geração de imagens foi utilizada a linguagem Python.

A plataforma foi configurada com ciclos de controle de 0,1 s e uma taxa de atualização física de 100 Hz (10 atualizações por ciclo) para garantir a fidelidade das interações.

Os robôs são divididos em grupos distintos a partir de suas cores e espalhados em um ambiente virtual quadrado. No início das simulações padronizadas, segue-se a seguinte configuração: largura e altura da arena virtual = 450 cm, número de grupos = 3, número de robôs por grupos (r/g) = 25, tempo de simulação = 2400 s, valor mínimo de velocidade = $-12,8$ cm/s, valor máximo de velocidade = $12,8$ cm/s, semente aleatória variando de 1 a 30 e a presença de cilindros de referência pela arena. Estes valores foram escolhidos baseados no trabalho de Krischanski et al. (2023), para que os resultados encontrados possam ser comparados.

Cada semente aleatória indica uma simulação, totalizando 30 execuções para uma mesma configuração. Além disso, ela é responsável pela distribuição dos robôs pela arena, de modo uniforme e aleatório, com suas orientações iniciais sorteadas no intervalo de $[-180^\circ, 180^\circ]$.

Assim como em Krischanski et al. (2023), são utilizados cilindros de referência fixos no ambiente virtual. Eles são distribuídos uniformemente sobre um círculo concêntrico, com raio equivalente a 40% do comprimento do ambiente, ou seja, 180 cm ou ao longo de uma linha diagonal na arena. A avaliação dos resultados permitiu observar que, quando uma solução apresenta bom desempenho na distribuição circular, ela tende a ser igualmente eficaz na distribuição diagonal, entretanto, o inverso nem sempre se confirma. Por esse motivo, priorizou-se a exibição dos casos em que há essa discrepância de comportamento, já nas situações em que a solução é robusta para ambas as configurações, a representação diagonal é omitida para garantir a concisão do texto.

A localização final desejada dos grupos pode ser guiada pela presença desses cilindros, já que eles são estáticos e permanecem fixos durante toda a simulação. É atribuído um cilindro para cada grupo, os quais compartilham a mesma cor dos demais integrantes, sendo tratados pelos robôs como robôs do próprio grupo ($I = 1$).

A utilização de cilindros de referência foi investigada como estratégia para mitigar problemas de oclusão entre os grupos. Na ausência desses cilindros, observa-se a tendência de os grupos convergirem para a região central da arena; como consequência, agrupamentos de um grupo passam a obstruir a visão dos robôs pertencentes a outros grupos. Com a introdução de cilindros de referência distribuídos pela arena, esses elementos passam a atuar como pontos de atração, direcionando os diferentes grupos para regiões distintas. Isso resulta em uma melhora

na qualidade das soluções obtidas e na mitigação dos efeitos de oclusão.

Além disso, em enxames de robôs, pode ser desejável que um único elemento exerça influência sobre o comportamento coletivo do grupo. Nesse contexto, o cilindro de referência pode ser substituído por um robô líder, com comportamento diferenciado ou ainda controlado por um operador humano. Neste último caso, reduzindo a carga cognitiva do operador humano (Miyauchi; Lopes; Groß, 2023).

A partir disso, os resultados das simulações são avaliados por meio da função de custo e das métricas de desempenho, dispersão e proporção de agrupamento, vide Equações ??, 10 e 11. Então, a escalabilidade das melhores soluções é explorada com a finalidade de validar a solução em diferentes cenários.

Neste caso, foram realizadas mais 200 simulações. Esse total é definido por meio da combinação das configurações: com e sem cilindro de referência, para $g = \{3, 5\}$ com $r/g = \{10, 15, 20, 25, 30\}$ (até 150 robôs no total), e semente aleatória variando de 31 a 40 (10 simulações para uma mesma configuração). Este total de robôs foi definido para que os resultados encontrados fossem comparáveis com a literatura utilizada. As demais variáveis mantêm os mesmos valores das simulações padronizadas.

Por fim, a arena virtual utilizada neste trabalho possui sempre as mesmas medidas mesmo com o aumento do número total de robôs, diferentemente de Gauci et al. (2014a), que altera o tamanho da arena a fim de manter a densidade de robôs sempre a mesma. Essa escolha se justifica porque o objetivo é avaliar o impacto da oclusão na escalabilidade das soluções.

4.2 ABORDAGEM 1: OTIMIZAÇÃO POR *GRID SEARCH*

A primeira fase experimental utilizou a técnica de busca em grade (*grid search*) devido à necessidade de selecionar, de maneira sistemática e eficiente, os parâmetros de controle, semelhantemente ao que foi realizado por Gauci et al. (2014a). Do ponto de vista metodológico, a escolha pela *grid search* foi motivada tanto por sua transparência quanto por sua capacidade de fornecer uma base sólida para análises posteriores, conforme descrito na Seção 2.6.1.

A primeira decisão desta abordagem foi limitar o espaço de busca, adotando um incremento de 10% da velocidade máxima de 12,8 cm/s, dentro do intervalo de velocidades de $-12,8$ cm/s a $12,8$ cm/s, resultando em um total de 21 valores distintos. Entretanto, o espaço de busca ainda se manteve extenso para a abordagem escolhida, com 21^6 (ou 85.766.121) combinações possíveis.

Por isso, optou-se por fixar os dois primeiros pares de velocidade do controlador, responsáveis pela agregação de tarefa única ($I = 0$ e $I = 1$), utilizando o controlador ótimo encontrado por Gauci et al. (2014a), apresentado na Equação 3. Assim, o número de combinações possíveis foi reduzido para $21^2 = 441$ e a busca exaustiva foi realizada apenas para o último par de velocidades ($\bar{v}_{l,2}$, $\bar{v}_{r,2}$ corresponde a $I = 2$, mantendo a seguinte estrutura de controle:

$$\bar{\mathbf{v}}_{\beta_1} = (-0,70; -1,00; 1,00; -1,00; \bar{v}_{\ell,2}; \bar{v}_{r,2}), \quad \bar{\mathbf{v}}_{\beta_1} \in [-1, 1]^6 \quad (12)$$

Durante a execução da *grid search*, cada combinação de parâmetros é avaliada de maneira independente por 30 simulações com diferentes condições iniciais e, ao final das simulações por todo o espaço de busca, os dados obtidos são organizados em um mapa de calor e, então, segue-se a metodologia apresentada na Seção 4.1.3. Assim, há um total de 13.230 simulações (441 combinações com 30 rodadas, cada uma correspondendo a uma semente aleatória diferente).

As execuções foram paralelizadas para otimizar o tempo de processamento, distribuídas em 30 chamadas paralelas com cada uma correspondendo a um conjunto de simulações. Foram utilizados também 4 computadores, a fim de avaliarem diferentes regiões do espaço de busca de modo proporcional à capacidade de processamento. Os processadores utilizados foram Intel Core i5 de 4ª geração e Intel Core i7 de 4ª, 9ª e 12ª gerações.

O tempo de execução para 13.230 simulações foi de 34 h 40 min. Já para 200 experimentos de escalabilidade foi de 1 h 20 min, paralelizados em 10 chamadas com 20 simulações cada uma. Assim, o tempo total de todas as execuções foi de 36 h. Ademais, os resultados obtidos são apresentados no Capítulo 5.

4.3 ABORDAGEM 2: OTIMIZAÇÃO AUTOMÁTICA COM ALGORITMOS DE BUSCA GLOBAL

A segunda fase experimental consiste na comparação sistemática de diferentes algoritmos de otimização para explorar o espaço de busca completo do controlador. Diferentemente da abordagem anterior, não foram impostas restrições ao espaço de busca, permitindo uma avaliação abrangente dos algoritmos. Assim, são considerados os três pares de velocidades do controlador, assumindo o incremento como 0,1 cm/s para o intervalo de -12,8 a 12,8 cm/s, resultando em 257^6 (ou 208.422.380.089.201) combinações possíveis.

Inspirado no trabalho de Krischanski et al. (2023), que utilizou um algoritmo de busca local, o DIRECT, como estratégia de otimização, esta análise comparou o desempenho deste algoritmo com abordagens híbridas baseadas na combinação do algoritmo global MLSL com os otimizadores locais SBPLX e COBYLA. O objetivo foi avaliar qual dessas estratégias se mostra mais eficiente para obter formações compactas em cenários multitarefa. A relevância desta análise reside no fato de que não existe um algoritmo universalmente superior, sendo fundamental avaliar qual estratégia se adapta melhor às características específicas deste problema.

A otimização foi realizada utilizando a biblioteca NLOpt, selecionados para abranger tanto abordagens puramente globais quanto estratégias híbridas. Cada um dos algoritmos selecionados, DIRECT, MLSL, SBPLX e COBYLA, descritos na Seção 2.6.3, foram submetidos a uma etapa de treinamento, na qual o objetivo era encontrar a configuração de controle que resultasse no menor valor da função de custo, a qual inicia com valor igual a 10^9 . A solução resultante então é validada por meio das métricas de dispersão e proporção de agrupamentos já mencionadas.

A configuração do treino difere da simulação porque ao invés de fornecer como entrada de dados o vetor de velocidades, são utilizadas as outras variáveis que permitem a exploração

do espaço de busca: avaliação e teste. A avaliação representa o número de configurações distintas avaliadas, enquanto o teste é a quantidade de repetições independentes por configuração, variando a semente aleatória, sendo muito importante para mitigar os efeitos de *outliers* e ruído estocástico.

Para o treino, as configurações utilizadas são: ambiente virtual com 200 cm de lado, 10 robôs por grupo, 2 grupos, tempo máximo de otimização (180 e 300 s), ausência de cilindros de referência, número de testes por avaliação (10, 30 e 100) e o número total de avaliações (1000, 6000 e 10000). Demais variáveis, são mantidas com os mesmos valores da simulação, descritos na Seção 4.1.3.

Os valores de tamanho da arena, tempo de execução, número de robôs, entre outros, no treino são reduzidos em relação a simulação para acelerar o processo de convergência do algoritmo, reduzindo a demanda de processamento e o ruído de interações excessivas que dificultariam o aprendizado inicial. Deste modo, o algoritmo consegue realizar milhares de iterações em poucos minutos, permitindo que o aprendizado das regras de comportamento e coordenação básica sem a necessidade da busca exaustiva. O objetivo é que o algoritmo desenvolva uma solução robusta e escalável que possa ser generalizada para o ambiente completo, tratando o cenário maior apenas como uma extensão lógica do que foi aprendido no ambiente simplificado.

Ainda assim há um custo computacional associado, e por isso, foram testadas mais de 50 configurações possíveis na etapa de treino, mas apenas as 25 mais consistentes foram avaliadas. Dentre os computadores mencionados anteriormente foram utilizados os com melhores processadores (Core i7 de 9ª e 12ª gerações), simultaneamente para executar mais de uma configuração. Entretanto, não houve paralelização do treinamento, cada treino foi integralmente conduzido em uma única execução contínua.

O tempo para cada treino variou muito. Os menores tempos foram cerca de 2 h 20 min (com 10 testes, 1000 avaliações e tempo máximo de otimização de 180 s), os intermediários foram de até 15 h (com 100 testes, 1000 avaliações ou 10 testes, 6000 avaliações ambos com tempo máximo de avaliação de 180 s), e os maiores tempos levaram mais de dois dias (para 100 testes e 10000 avaliações).

As melhores soluções encontradas foram então validadas por meio das simulações padronizadas e dos experimentos de escalabilidade, conforme a metodologia apresentada na Seção 4.1.3. No Capítulo 5 são apresentados os experimentos realizados.

5 RESULTADOS

Com base na análise da literatura apresentadas nos capítulos anteriores, foi identificada uma lacuna na capacidade de controladores minimalistas para alcançar agregação compacta em cenários multitarefa. Neste capítulo são apresentados os resultados das propostas de controle apresentadas no Capítulo 4.

O objetivo principal é encontrar um conjunto de parâmetros para um controlador reativo que contorne as limitações observadas nos trabalhos relacionados, formação de anéis em órbita em vez de agrupamentos compactos e seus desafios de escalabilidade. Para atingir esse objetivo, são implementadas e comparadas duas estratégias concorrentes de síntese de controladores: a busca em grade (*grid search*) e a otimização automática híbrida, busca global e local associadas.

Inicialmente há a implementação de cada abordagem e, em seguida, são apresentados e analisados os resultados obtidos. Por fim, realiza-se uma análise comparativa do desempenho dos melhores controladores encontrados, discutindo suas vantagens, desvantagens e o potencial para alcançar o comportamento de agregação compacta em multitarefa desejado.

5.1 ABORDAGEM DE OTIMIZAÇÃO POR *GRID SEARCH*

A primeira abordagem experimental deste trabalho emprega a técnica de *grid search* para mapear sistematicamente o comportamento do enxame e identificar os parâmetros de controle que geram a agregação multitarefa compacta. Conforme a metodologia, e para tornar a busca computacionalmente viável, a estratégia foi fixar os parâmetros já validados por Gauci et al. (2014a) para a agregação de tarefa única.

Assim, a exploração foi focada nos parâmetros que correspondem a interação entre robôs de grupos distintos ($I = 2$), permitindo uma análise aprofundada da regra de controle mais crítica para a segregação de grupos. Esta Seção detalha a configuração específica deste procedimento e os resultados obtidos.

Ademais, vale destacar a possibilidade de se observar, dentro do espaço de busca, múltiplas configurações com desempenhos semelhantes, o que pode indicar redundância nos parâmetros ou a presença de compensações entre eles. Entretanto, também há casos em que pode-se verificar que o desempenho é altamente sensível a variações em determinado parâmetro. Além disso, os dados obtidos com a *grid search* podem ser utilizados como base para métodos mais sofisticados de ajuste, como interpolações, modelagem estatística ou aprendizado supervisionado.

Em resumo, ao empregar a abordagem *grid search* em um contexto de agregação multitarefa, o presente trabalho busca não apenas demonstrar sua eficácia, mas também contribuir para a construção de metodologias replicáveis no campo da engenharia de enxames. Alinhando-se aos princípios de rigor experimental que regem a pesquisa científica em sistemas multirrobôs distribuídos.

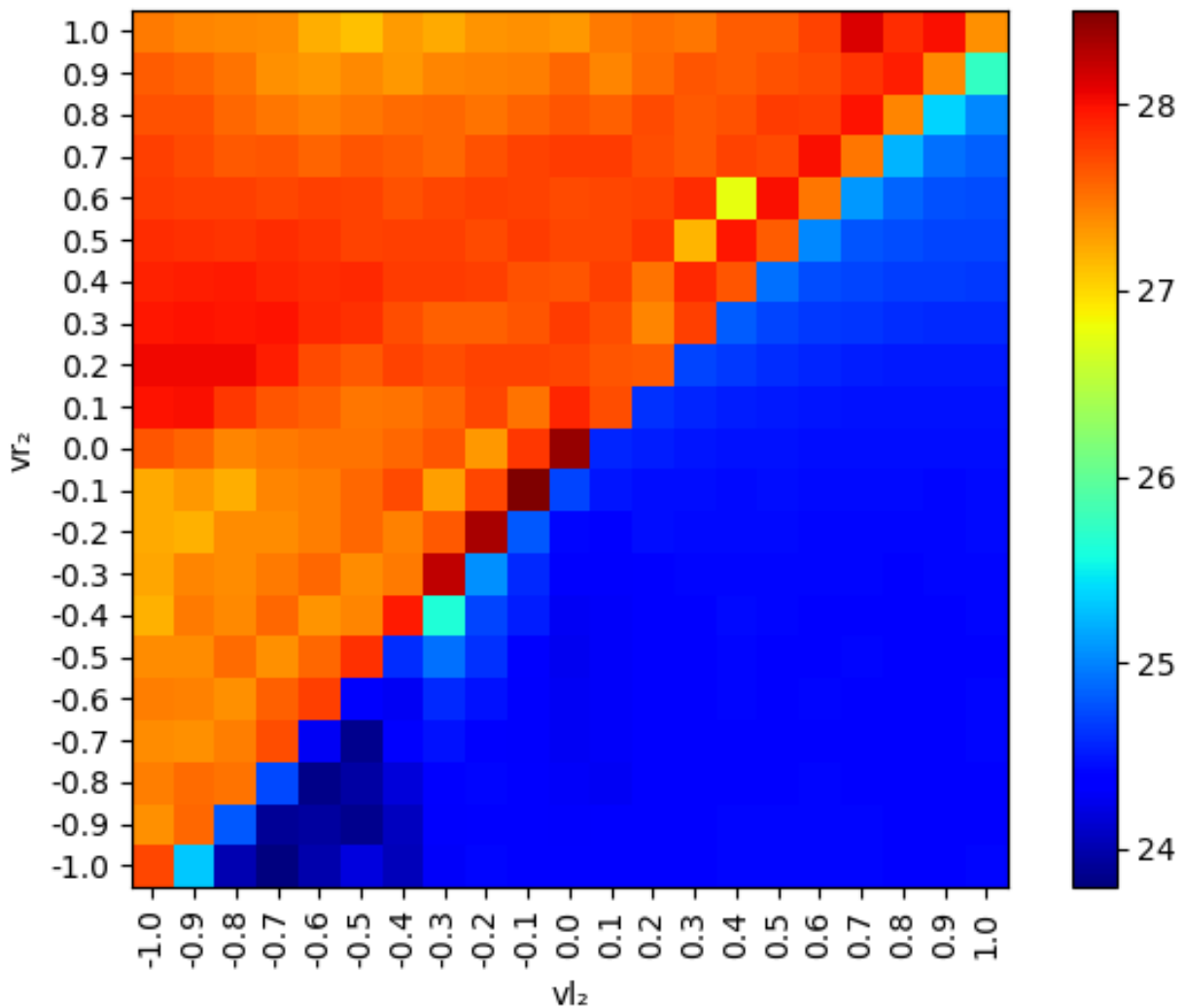
5.1.1 Grid Search: Estatísticas e Resultados

Primeiramente, foram executadas as simulações conforme descrito na Seção 4.2. Com os resultados obtidos pela *grid search*, os dados de desempenho de cada uma das 13.230 simulações foram coletados e organizados.

Na Figura 5, há um mapa de calor mostrando esses resultados para os dois parâmetros ($\bar{v}_{\ell,2}$ e $\bar{v}_{r,2}$), respectivamente, as velocidades esquerda (eixo x) e direita (eixo y) quando o sensor de linha de visão detecta um robô de outro grupo ($I = 2$). Cada quadrado do mapa de calor representa um incrementado de 0,1 das velocidades normalizadas, de $-1,0$ a $1,0$, para $\bar{v}_{\ell,2}$ e $\bar{v}_{r,2}$.

O gradiente de cores, varia de azul (baixo) a vermelho (alto), representando o logaritmo natural da função de custo média, vide Equação 2, em 30 simulações para cada combinação de parâmetros (uma para cada semente aleatória). Áreas azuis mais escuras indicam valores de custo mais baixos (melhores), enquanto áreas vermelhas representam custos mais altos. O mapa de calor auxilia na identificação de regiões ótimas de parâmetros, destacando combinações que minimizam a função de custo.

Figura 5 – Mapa de calor mostrando os resultados da *grid search* para $\bar{v}_{\ell,2}$ e $\bar{v}_{r,2}$.



Fonte: A autora (2025).

Assim, pode-se observar que os melhores parâmetros encontrados foram $(\bar{v}_{\ell,2}, \bar{v}_{r,2}) = (-0,70; -1,00)$, pois possuem a região com azul mais escuro. Resultando na configuração do controlador:

$$\bar{\mathbf{v}}_0 = (-0,70; -1,00; 1,00; -1,00; -0,70; -1,00) \quad (13)$$

Isso indica que o melhor controlador consiste em acionar as mesmas velocidades para as rodas do robô em ambos os casos: quando o sensor de linha de visão não detecta nada ($I = 0$) ou detecta um robô de outro grupo ($I = 2$).

Também observa-se a existência de um controlador simétrico, isto é, trocando as velocidades esquerda e direita para cada par de leituras do sensor tem-se que:

$$\bar{\mathbf{v}}_{0a} = (-1,00; -0,70; -1,00; 1,00; -1,00; -0,70) \quad (14)$$

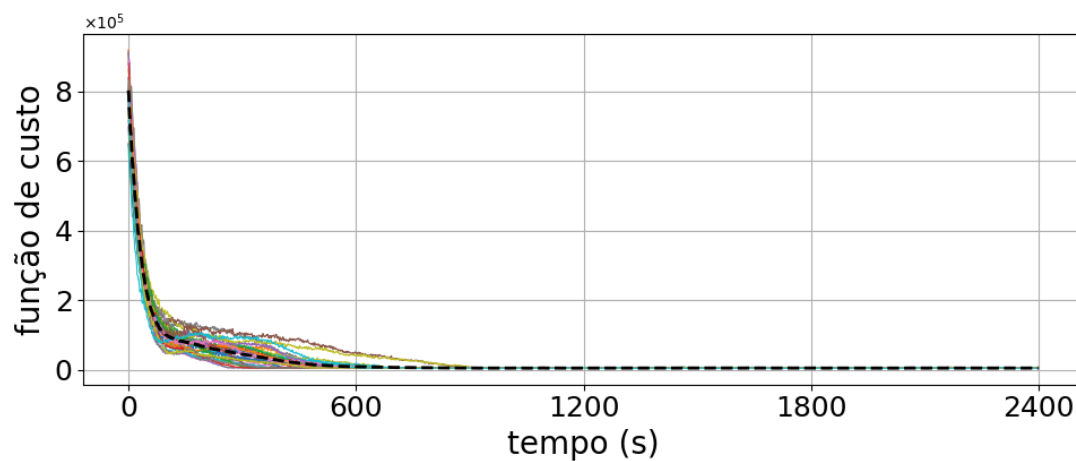
Este controlador simétrico é uma solução equivalente, pois gera o mesmo mapa de calor espelhado em torno do eixo horizontal, invertendo a posição dos vértices superior e inferior.

Em Krischanski et al. (2023), o comportamento principal envolve varrer o ambiente realizando curvas fechadas para frente até avistar um robô do mesmo grupo. Quando outro robô do mesmo grupo é detectado, o robô move-se em linha reta em sua direção. Se o robô deixa de detectar outro robô do mesmo grupo, retorna ao movimento de varredura.

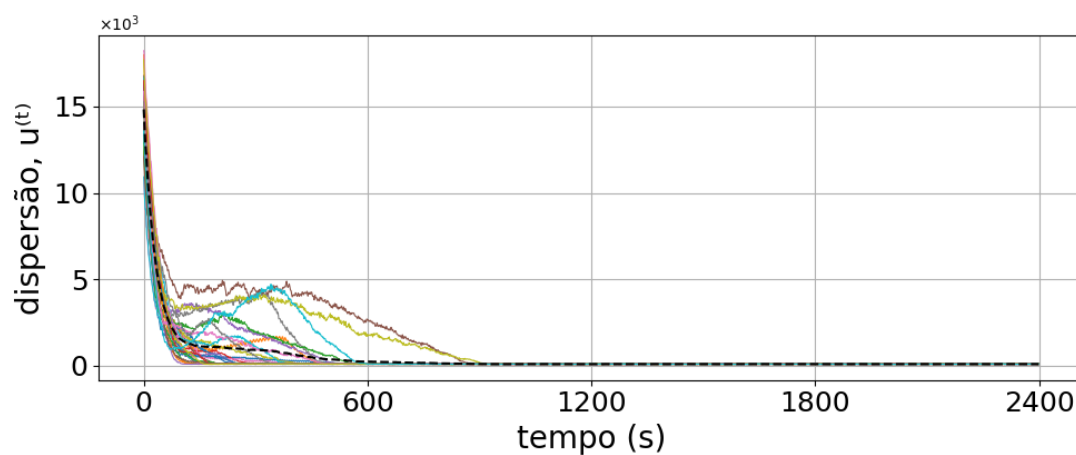
Já o comportamento principal observado aqui, consiste em girar no sentido horário em velocidade máxima até avistar um robô do mesmo grupo, uma estratégia de varredura similar, mas com rotação em vez de curva. Quando outro robô do mesmo grupo é detectado, o robô executa uma curva suave para trás no sentido horário. Se o robô deixa de detectar outro robô do mesmo grupo, retorna ao movimento de varredura.

A Figura 6 apresenta a evolução da função de custo (a), dispersão (b) e proporção (c) ao longo do tempo durante o treinamento para os melhores parâmetros do controlador. Linhas finas contínuas indicam a evolução de cada uma das 30 rodadas e a linha preta tracejada grossa indica o valor médio de todas as rodadas. Também é possível notar pela Figura 6, que a solução encontrada convergiu bem e rapidamente, antes de 1200 s, em todas as simulações. Assim, $\bar{\mathbf{v}}_0$ foi validado como a melhor solução para o controlador proposto e por isso foi explorado por meio dos experimentos de escalabilidade.

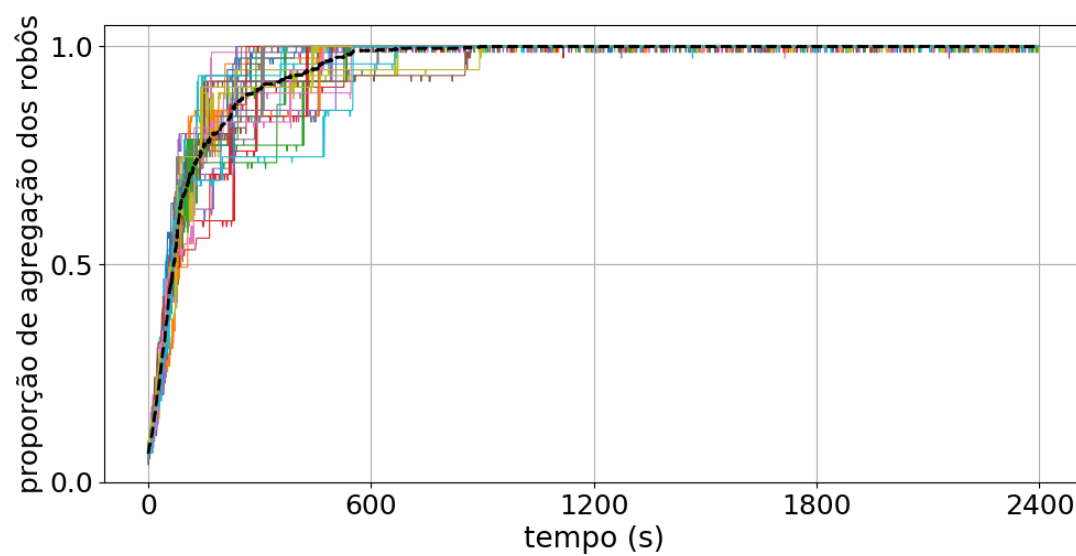
Figura 6 – Desempenho do melhor conjunto de parâmetros do controlador $\bar{v}_0$ obtidos por *grid search* durante o treinamento.



(a) Função de Custo



(b) Dispersão



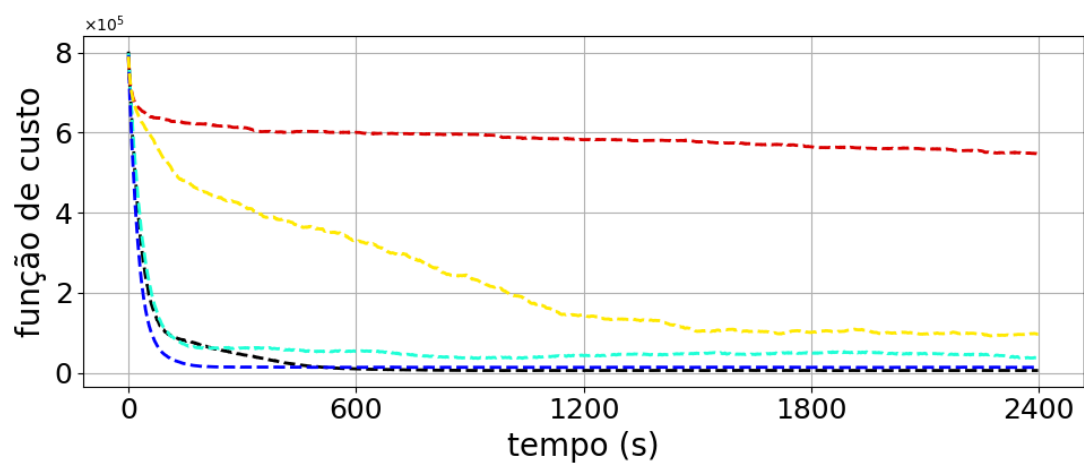
(c) Proporção

Fonte: A autora (2025).

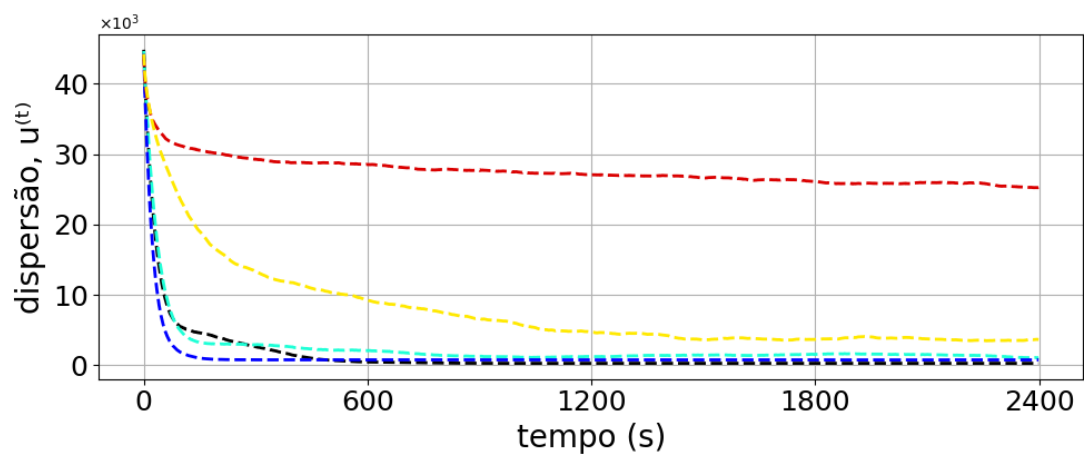
Para validar a hipótese de que $\bar{v}_0$ era de fato o valor ótimo, analisou-se a evolução da sua função de custo ao longo do tempo, e de outras quatro opções de vetores, os quais variam apenas o último par de velocidades. A escolha destes valores foi baseada nos resultados apresentados no mapa de calor, ou seja, valores entre 24 e 26 na escala de cores da Figura 5.

Assim, a seleção contemplou diferentes variações de comportamentos quando detectados robôs de outros grupos ($I = 2$), incluindo combinações de sinais opostos, puramente negativas e puramente positivas (distantes e adjacentes), permitindo deste modo avaliar e comparar o valor ótimo proposto com vetores de desempenho similar, porém com características distintas. Dessa forma, conforme a Figura 7, os valores são dados por: preto é $\bar{v}_0$ (mesmo da Figura 6), ciano é igual a $(-0,25; -0,40)$, azul representa $(0,60; -0,50)$, vermelho é $(0,70; 1,00)$ e amarelo corresponde a $(0,40; 0,60)$.

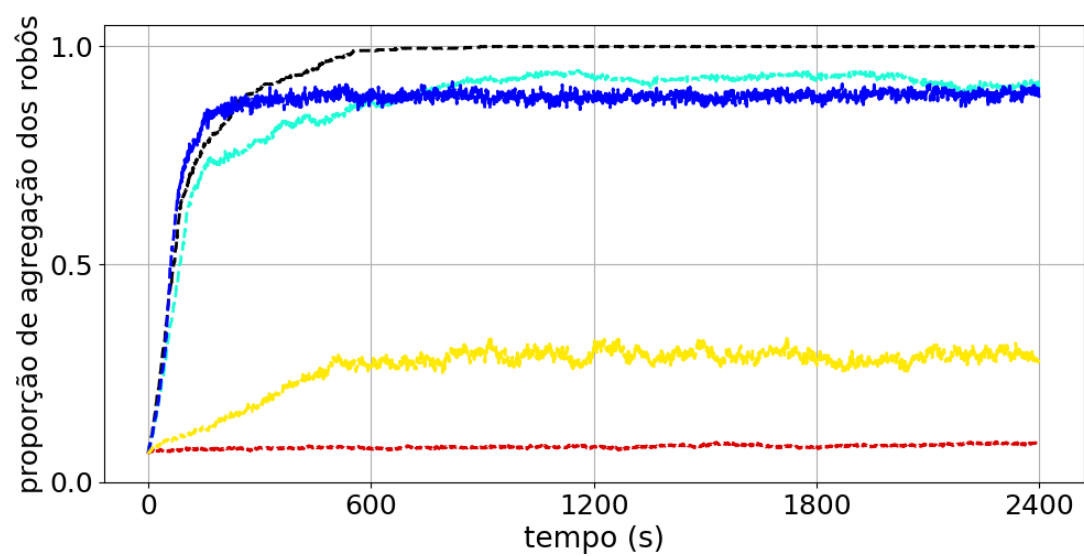
Na Figura 7 cada linha indica o valor médio de todas as rodadas para cada conjunto de parâmetros do controlador, sendo a evolução da função de custo (a), dispersão (b) e proporção (c) ao longo do tempo.

Figura 7 – Comparativos com $\bar{v}_0$ 

(a) Função de Custo



(b) Dispersão



(c) Proporção

Fonte: A autora (2025).

A configuração representada pela linha azul, apresentou uma boa convergência, o que era esperado, já que seus valores são muito próximos da configuração de $\bar{v}_0$. As demais configurações não apresentaram resultados satisfatórios, convergindo apenas parcialmente. Assim, $\bar{v}_0$ foi validado como a melhor solução para o controlador proposto e por isso foi explorado por meio dos experimentos de escalabilidade.

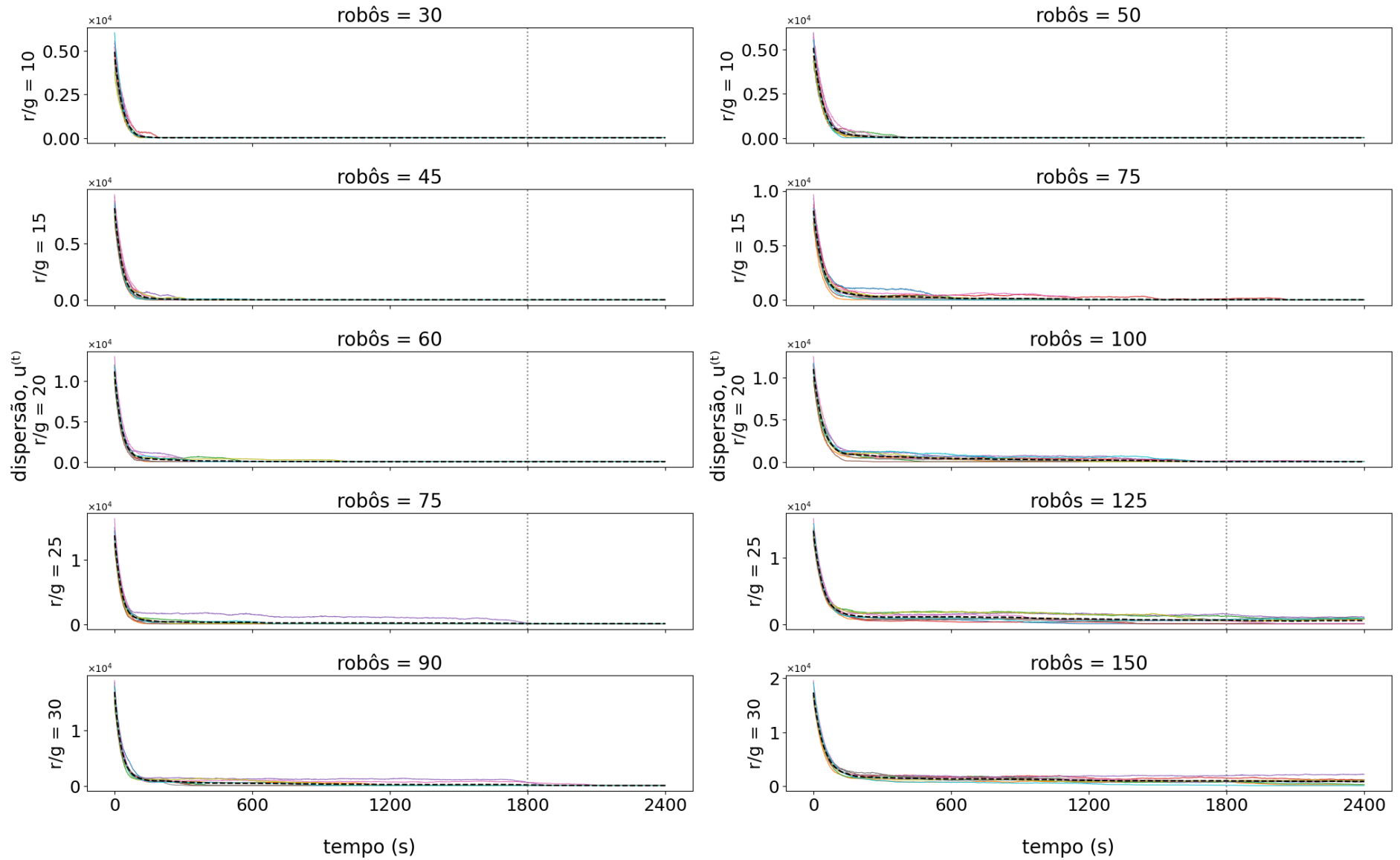
5.1.2 Validação e Escalabilidade Experimental

Nesta etapa, foram realizados testes sistemáticos para validação e investigação da escalabilidade em relação a variação do número de grupos e robôs por grupo. Conforme descrito na Seção 4.1.3, conduziram-se 10 testes com duração de 2400 s para cada configuração proposta, com cada teste sendo realizado com configurações iniciais (posição e ângulos) diferentes.

As Figuras 8a, 8b, 9a e 9b apresentam a dispersão ao longo do tempo. Nelas, é possível observar que apresenta uma evolução em praticamente todos os cenários, com um leve ruído quando o número total de robôs é de 150 e não há cilindro de referência. Já as Figuras 10a, 10b, 11a e 11b mostram a proporção de robôs agrupados ao longo do tempo. Neste caso, a partir de 125 robôs totais, o ruído aumenta e a solução não converge totalmente mas estabiliza. Entretanto, com a presença do cilindro de referência esse problema é solucionado.

Em ambas as métricas, dispersão e proporção de robôs agrupados, observa-se um período de convergência seguido por um estado estacionário. Isso indica que, após a formação dos agrupamentos, o comportamento emergente obtido é capaz de manter a formação compacta, já que o único critério de parada é atingir 2400 s do tempo de simulação. Em todas as figuras que apresentam os gráficos de evolução ao longo do tempo de função de custo, dispersão e proporção, cada linha representa uma simulação com semente aleatória diferente e a linha preta tracejada representa a média.

Figura 8 – Dispersão v_0 Sem Cilindro de Referência

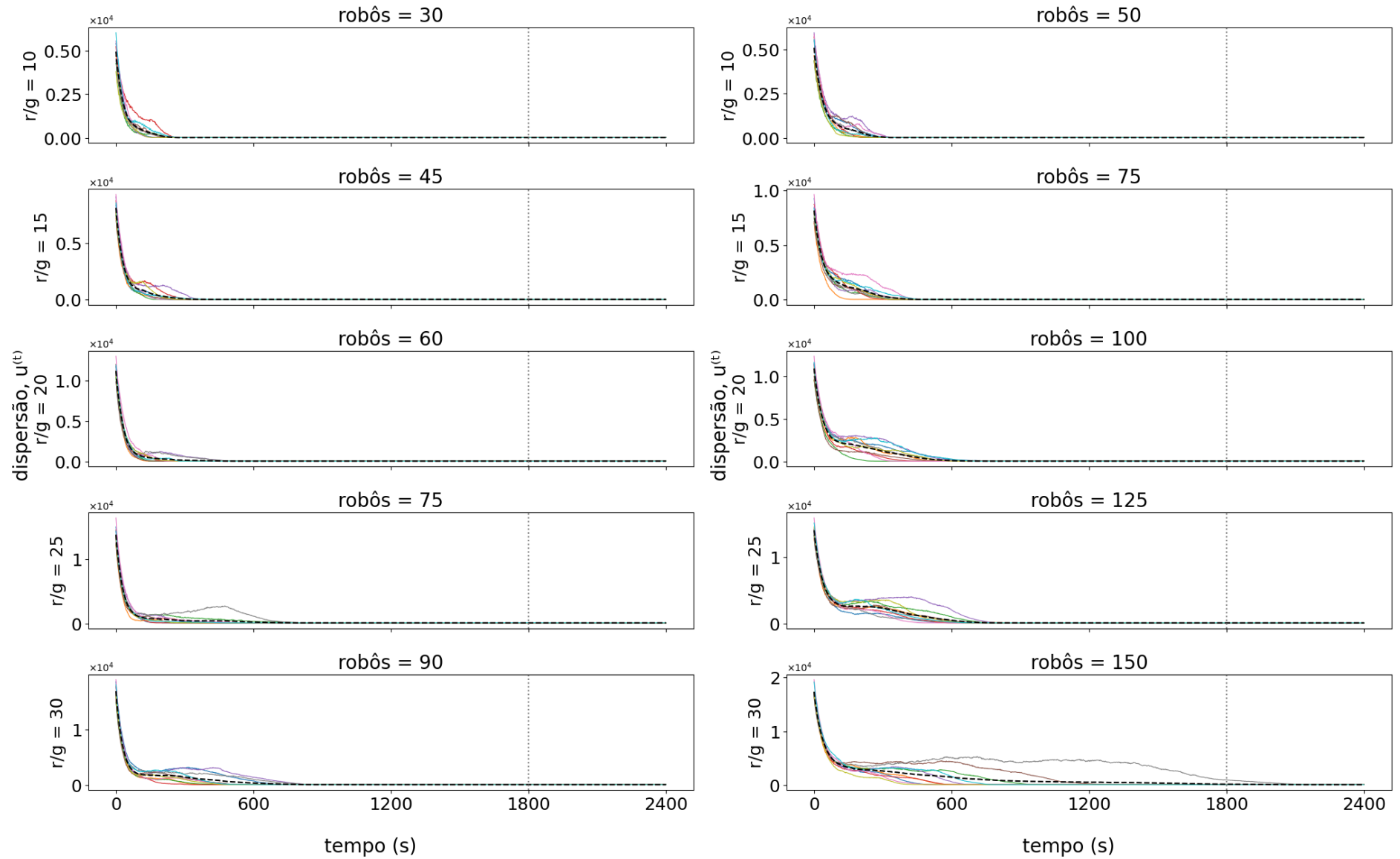


(a) Sem Cilindro de Referência - 3 grupos

(b) Sem Cilindro de Referência - 5 grupos

Fonte: A autora (2025).

Figura 9 – Dispersão $\bar{v}_0$ Com Cilindro de Referência

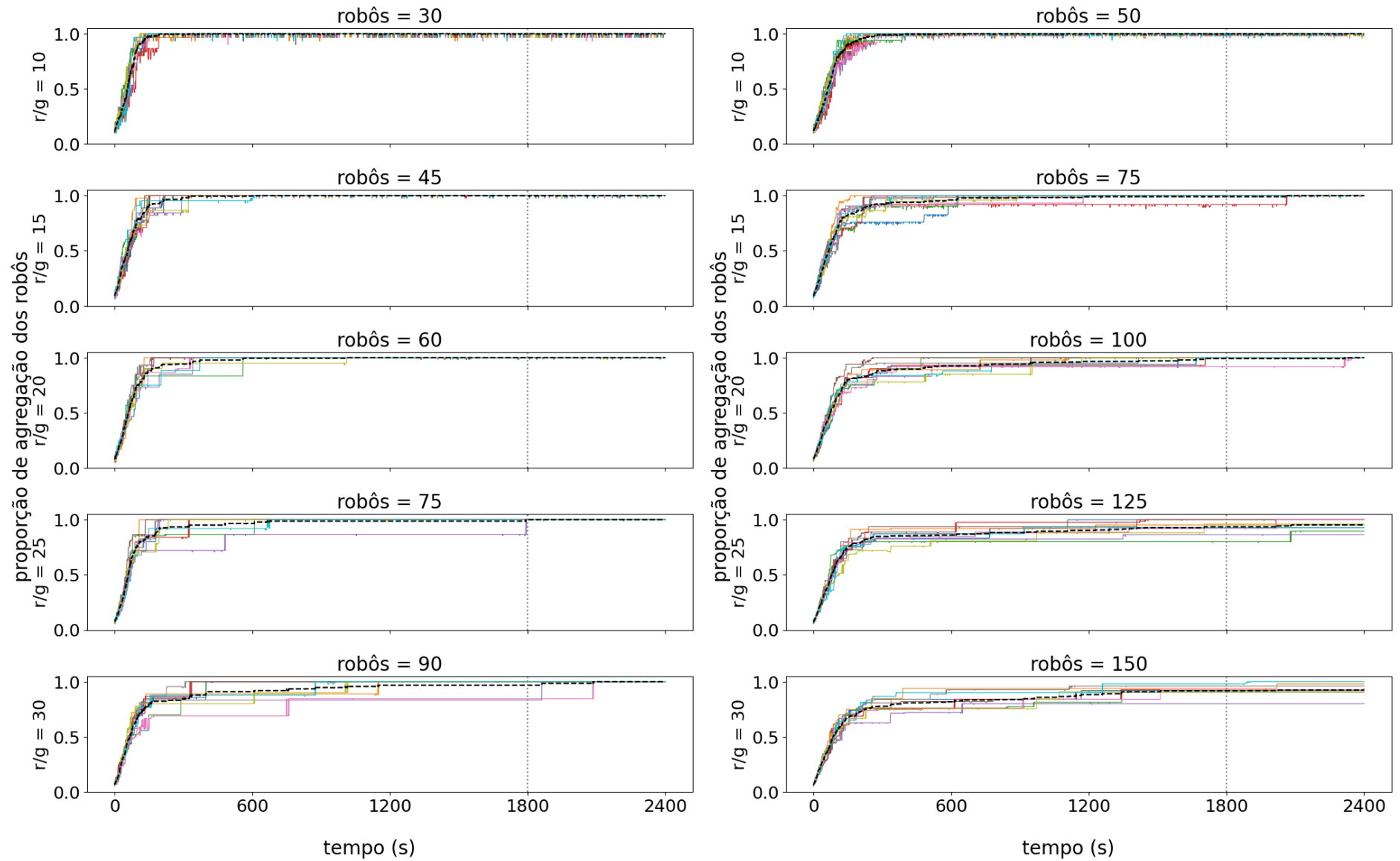


(a) Com Cilindro de Referência - 3 grupos

(b) Com Cilindro de Referência - 5 grupos

Fonte: A autora (2025).

Figura 10 – Proporção $\bar{v}_0$ Sem Cilindro de Referência

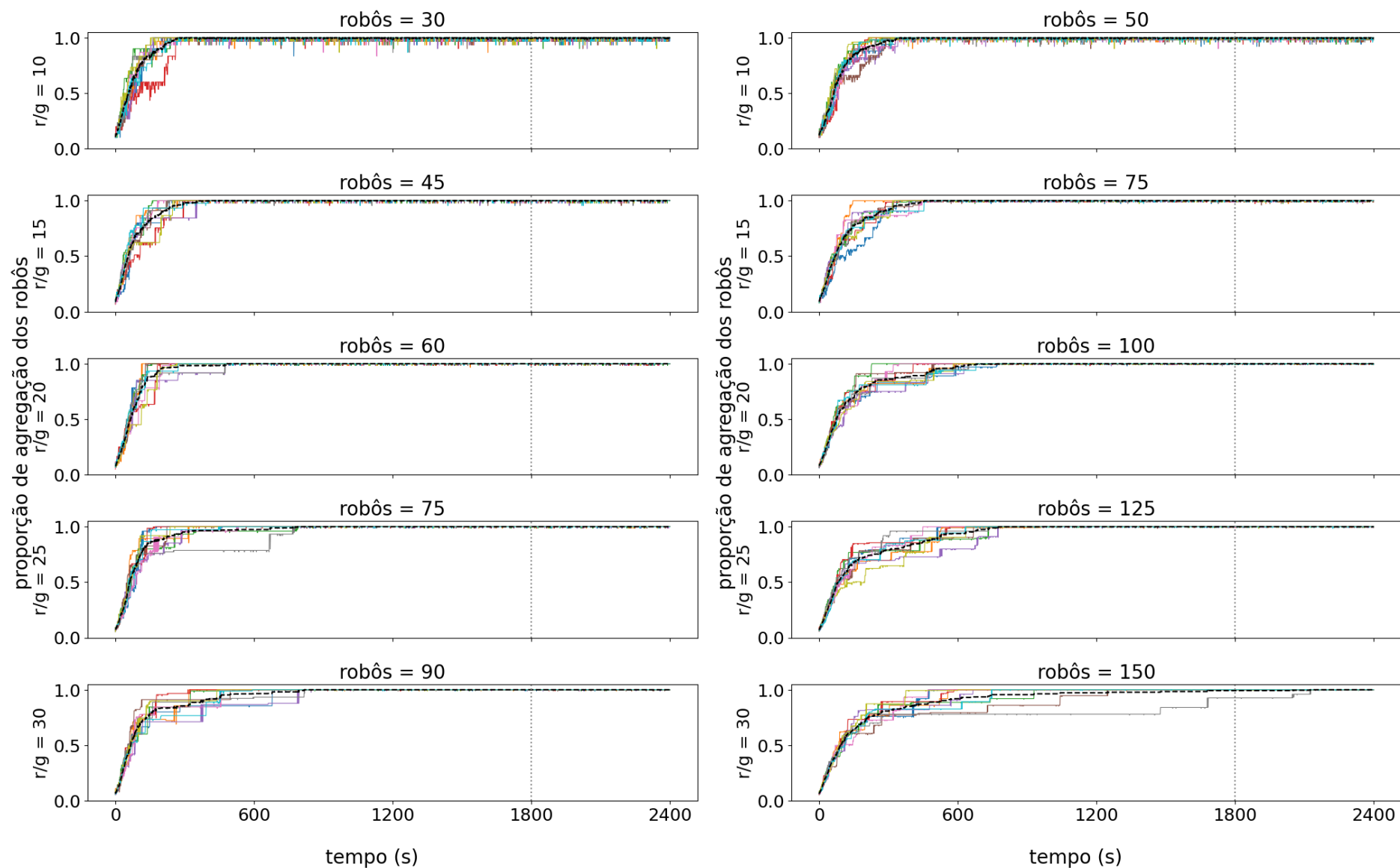


(a) Sem Cilindro de Referência - 3 grupos

(b) Sem Cilindro de Referência - 5 grupos

Fonte: A autora (2025).

Figura 11 – Proporção $\bar{v}_0$ Com Cilindro de Referência



Além disso, na dinâmica de dispersão ao longo do tempo, ao comparar experimentos com $g = 3$ e $g = 5$, tanto a dispersão entre os grupos quanto o tempo necessário para atingir o estado estacionário aumentam, ligeiramente, em todos os cenários (com ou sem cilindros de referência e diferentes valores de robôs por grupo). A dispersão acumulada e o tempo até a estabilização também aumentam, de forma leve, à medida que cresce o número de robôs por grupo.

Esse comportamento é esperado, dado que grupos maiores representam um desafio maior, por haver maior incidência de oclusão, já que o tamanho da arena é mantido. Ainda assim, o tempo adicional necessário para atingir o estado estacionário corresponde apenas a uma fração do aumento do problema, em termos do número de robôs.

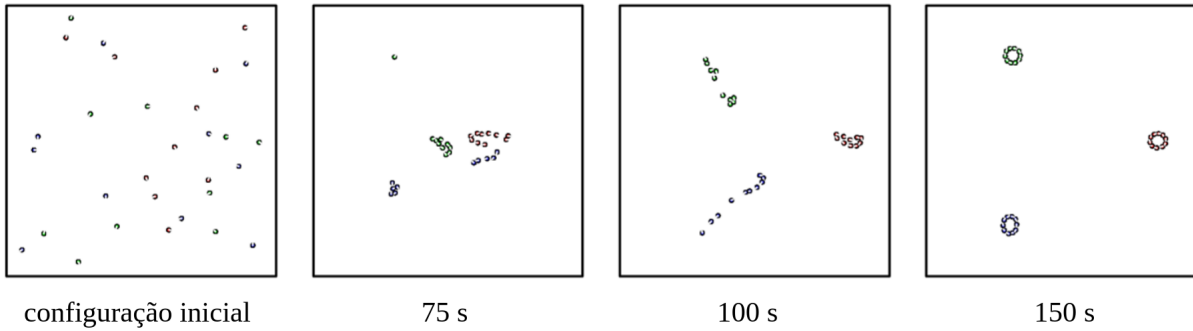
Ao comparar a dispersão em cenários com e sem o cilindro de referência, nota-se que a adição do cilindro aumenta tanto a dispersão acumulada quanto o tempo para estabilização. Esse aumento é quase imperceptível para $g = 3$, mas se torna mais evidente para $g = 5$. Isso se justifica porque deslocar-se até a posição do cilindro de referência é uma etapa adicional, que demanda tempo extra. Os cilindros de referência foram utilizados em Krischanski et al. (2023) para mitigar os efeitos de oclusão, mas o comportamento observado neste trabalho mostra-se suficientemente robusto, em termos de dispersão, para não depender do uso desses cilindros de referência.

Com relação à proporção de robôs agrupados, os experimentos com três grupos ($g = 3$) apresentaram um bom desempenho, especialmente com o uso do cilindro de referência. Ao considerar cinco grupos ($g = 5$), os cilindros de referência contribuíram significativamente para o aumento da proporção de robôs agrupados.

De fato, devido ao tamanho limitado da arena, a oclusão representa um fator de atraso na formação completa dos agrupamentos, e os cilindros de referência atenuam esse efeito negativo. Embora a oclusão ainda exerça influência sobre o comportamento obtido pelo método proposto neste trabalho, o resultado encontrado supera significativamente aquele relatado em Krischanski et al. (2023).

Considerando $r/g = 30$ e o uso dos cilindros de referência, Krischanski et al. (2023) relata que a proporção de robôs agrupados na marca de 1800 s foi de 40,22% para $g = 3$ e 38,53% para $g = 5$. No comportamento obtido neste trabalho, preservando as mesmas configurações de Krischanski et al. (2023), para a mesma marca temporal atingiu-se 100,00% e 99,33% para $g = 3$ e $g = 5$, respectivamente. Essa diferença é significativa devido ao fato de que em Krischanski et al. (2023), os robôs ficam em órbita circular e não em agrupamentos compactos, o que penaliza as métricas avaliadas, pois fica um espaço permanentemente vazio entre os robôs, porém seu controlador é mais rápido ao segregar os grupos conforme observado na Figura 12.

Figura 12 – Comportamento $\bar{v}_{\alpha_2}$ para $r = 90$ robôs com a presença de cilindros de referência.

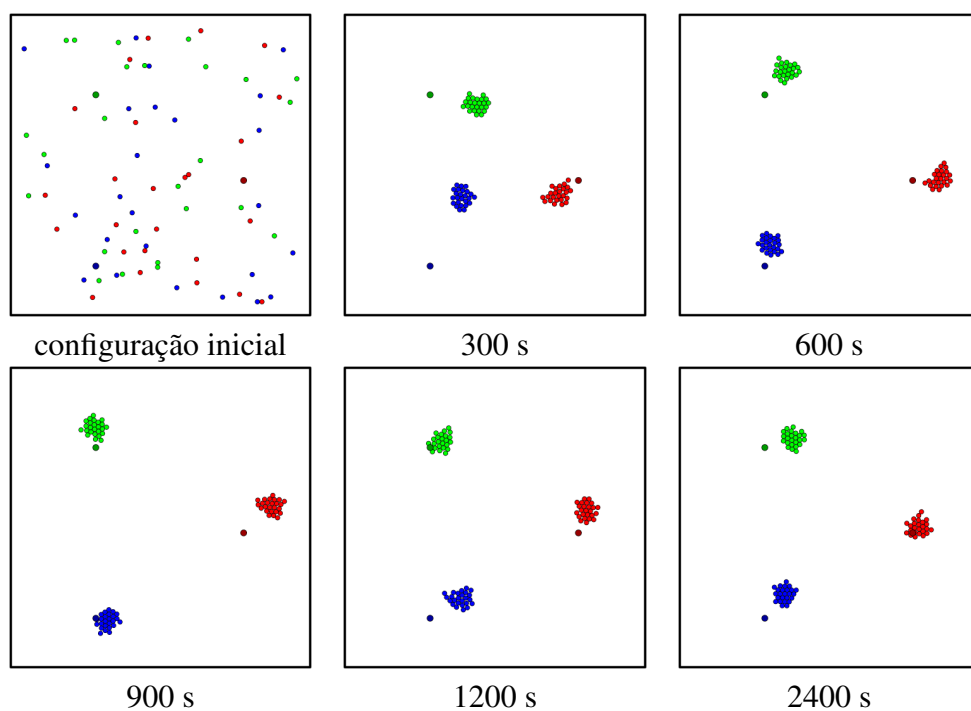


Fonte: Adaptado Krischanski et al. (2023).

Assim a Figura 13 representa o comportamento dos robôs com o controlador $\bar{v}_0$ com a presença de cilindros de referência com $r/g = 25$ em cada um dos três grupos, e as Figuras 14, 15, 16 e 17 apresentam uma sequência de imagens de experimentos com $r/g = 30$ robôs por grupo, ilustrando o comportamento de formação compacta obtido. As imagens variam conforme o número de grupos (g) e a presença ou não dos cilindros de referência. Os cilindros de referência podem ser distinguidos dos robôs na simulação, mas nos *snapshots* gerados eles são representados da mesma forma que os robôs.

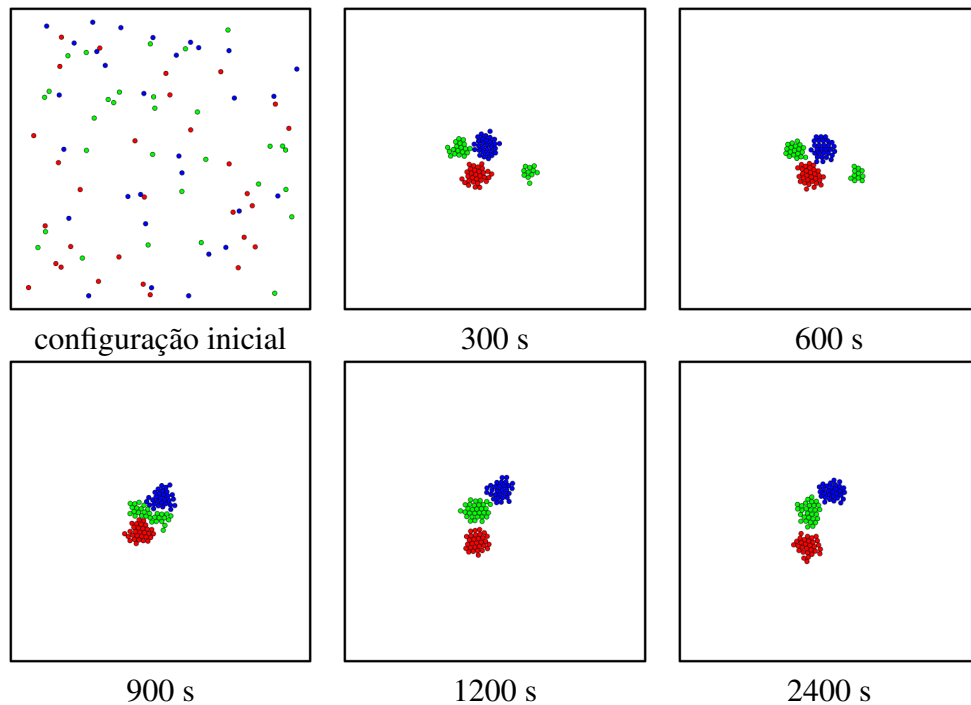
O uso dos cilindros de referência influenciou a distribuição dos agrupamentos ao redor do centro da arena, dentro do raio definido pela distribuição dos cilindros de referência. Na ausência deles, todos os agrupamentos tendem a se formar próximos ao centro da arena, além de observa-se que o comportamento é capaz de permitir que dois agrupamentos de robôs do mesmo tipo (cor) eventualmente se unam ao contornar outros agrupamentos.

Figura 13 – Comportamento $\bar{v}_0$ para $r = 75$ robôs com a presença de cilindros de referência.



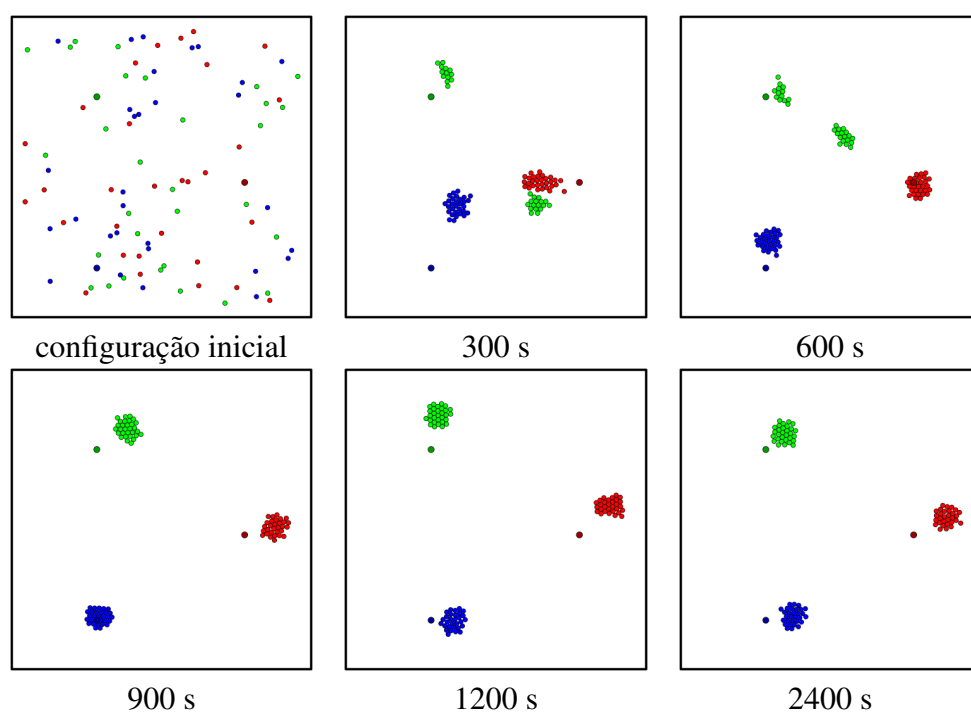
Fonte: A autora (2025).

Figura 14 – Comportamento $\bar{v}_0$ para $r = 90$ robôs sem a presença de cilindros de referência.



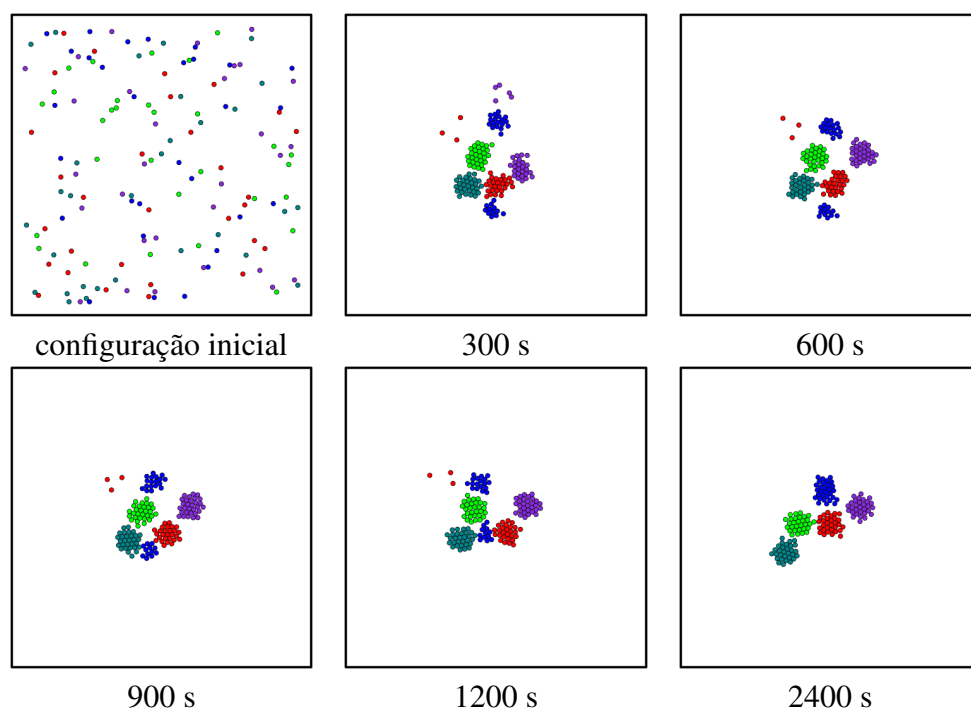
Fonte: A autora (2025).

Figura 15 – Comportamento $\bar{v}_0$ para $r = 90$ robôs com a presença de cilindros de referência.



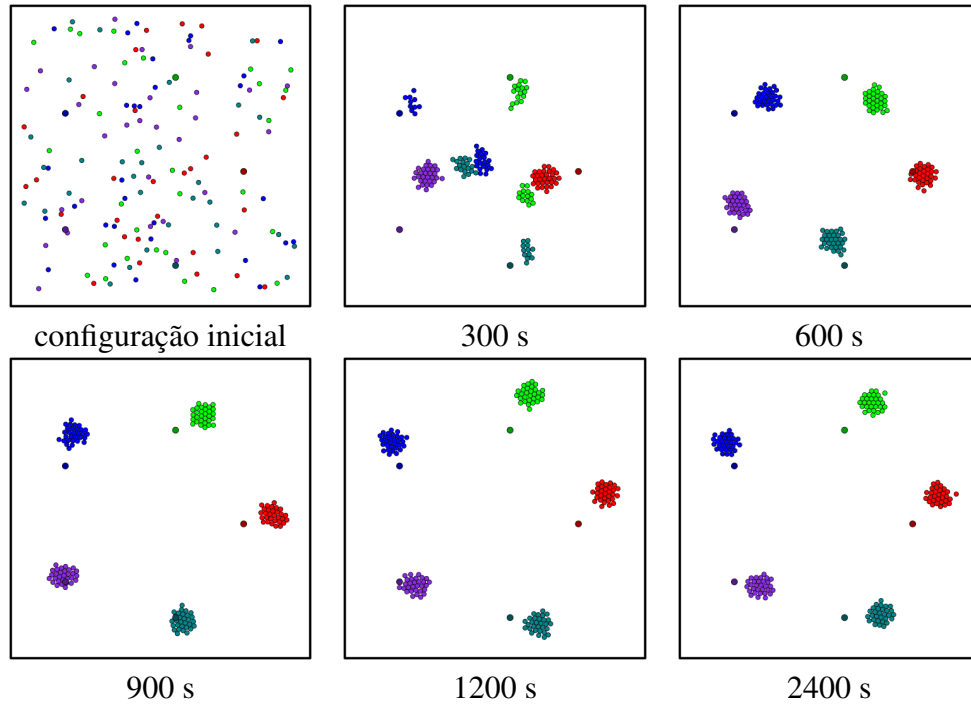
Fonte: A autora (2025).

Figura 16 – Comportamento $\bar{v}_0$ para $r = 150$ robôs sem a presença de cilindros de referência.



Fonte: A autora (2025).

Figura 17 – Comportamento $\bar{v}_0$ para $r = 150$ robôs com a presença de cilindros de referência.



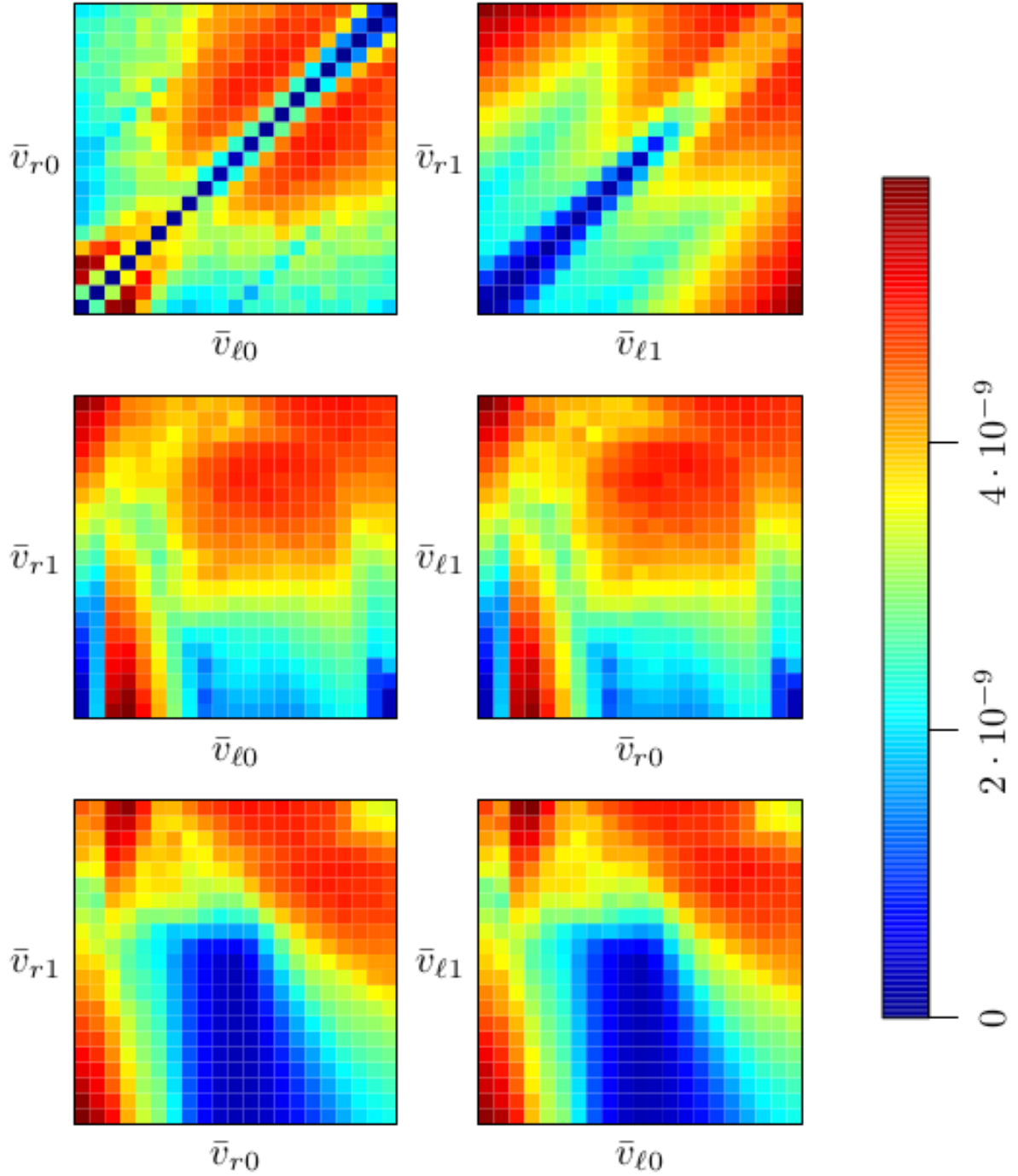
Fonte: A autora (2025).

5.1.3 Exploração de Novos Parâmetros

Posteriormente, uma nova análise exploratória foi feita para investigar outras configurações dos parâmetros, a fim de compará-los. Para validar a eficácia do controlador identificado como ótimo global, a saber, $\bar{v}_0 = (-0,70; -1,00; 1,00; -1,00)$, foram selecionadas quatro configurações adicionais ($\bar{v}_1, \bar{v}_2, \bar{v}_3, \bar{v}_4$) para uma análise comparativa. A seleção desses vetores baseou-se na interpretação do mapa de calor da Figura 18, bem como na metodologia do autor Gauci et al. (2014a), que utilizou este mesmo mapa de calor para encontrar os dois primeiros pares de velocidades fixados em $\bar{v}_0$.

Na Figura 18, é apresentado o panorama de desempenho em todo o espaço de busca dos controladores investigados por Gauci et al. (2014a), sendo cada eixo de todos os gráficos variando de $[-1,00; 1,00]$, que equivale as velocidades normalizadas máximas das rodas dos robôs, esquerda e direita. Os valores mais altos, ou mais avermelhados, significam um melhor desempenho de acordo com a métrica de dispersão.

Figura 18 – Mapa de calor da *grid search* para $\bar{\mathbf{v}}_\beta = (\bar{v}_{\ell,0}, \bar{v}_{r,0}, \bar{v}_{\ell,1}, \bar{v}_{r,1})$.



Fonte: Gauci et al. (2014a).

Como o mapa de calor apresenta vários pontos promissores em diferentes áreas do espaço de busca, a seleção de $\bar{\mathbf{v}}_1, \bar{\mathbf{v}}_2, \bar{\mathbf{v}}_3, \bar{\mathbf{v}}_4$ permite verificar se outros ótimos locais oferecem resultados comparáveis ao ótimo global, $\bar{\mathbf{v}}_0$. Deste modo, considerando a escala avermelhada (mediana superior), $\bar{\mathbf{v}}_1$ e $\bar{\mathbf{v}}_2$, possuem avaliações medianas, $\bar{\mathbf{v}}_3$, uma avaliação considerada ruim, e $\bar{\mathbf{v}}_4$ uma avaliação próxima a de $\bar{\mathbf{v}}_0$. Os controladores são:

1. $\bar{\mathbf{v}}_1 = (0, 50; -0, 40; 0, 40; 0, 50);$

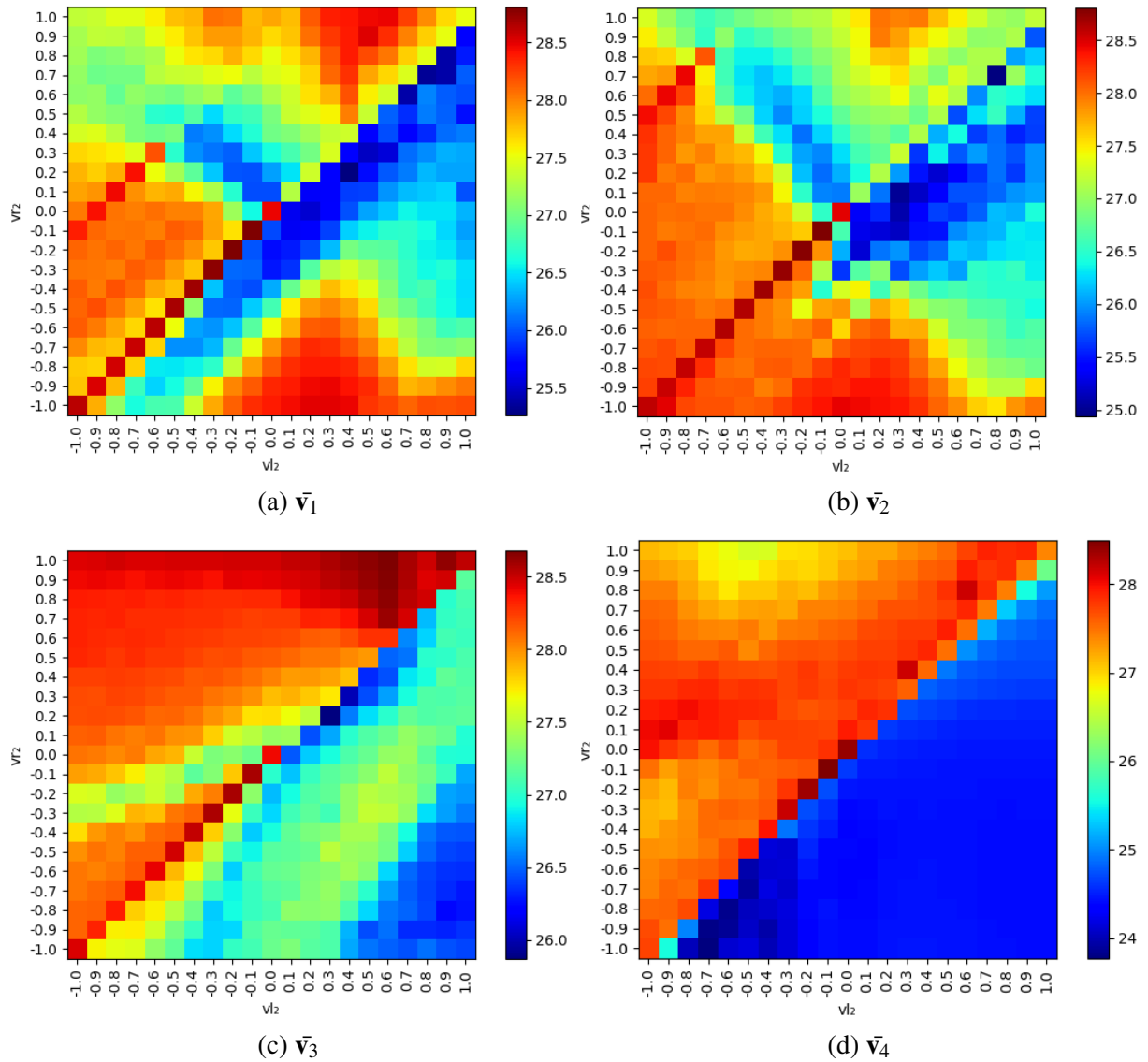
2. $\bar{v}_2 = (0, 60; -0, 90; 0, 30; 0, 60);$
3. $\bar{v}_3 = (0, 70; 0, 30; 0, 60; 1, 00);$
4. $\bar{v}_4 = (-0, 60; -0, 80; 1, 00; -1, 00).$

A diversidade de estratégias de movimento entre os controladores selecionados reside na forma como o espaço é explorado (estado $I = 0$) e como o robô reage ao detectar um vizinho (estado $I = 1$). Neste caso, os controladores seguem as seguintes lógicas:

- $\bar{v}_1$: Realiza uma varredura do ambiente através de uma rotação horária fechada com um leve deslocamento frontal quando não detecta ninguém. Ao detectar um robô do seu grupo, ele executa uma curva suave para a esquerda em avanço frontal, em uma estratégia de aproximação orbital;
- $\bar{v}_2$: Utiliza uma rotação horária rápida combinada a um recuo linear no estado de busca. Esta estratégia permite escanear o ambiente enquanto o robô aumenta sua área de cobertura ao se afastar da posição atual. Na aproximação, realiza uma curva frontal para a esquerda;
- $\bar{v}_3$: Executa uma busca puramente frontal através de uma trajetória circular para a direita. Ao detectar um vizinho, inverte a direção da curva para a esquerda, mantendo o avanço. Esta configuração testa a eficácia da agregação sem o uso de rotações sobre o próprio eixo ou movimentos de para trás; e
- $\bar{v}_4$: Assemelha-se à lógica de $\bar{v}_0$, utilizando um recuo circular horário no estado de exploração. Ao detectar um companheiro, realiza a rotação sobre o próprio eixo com velocidade máxima, o que maximiza a chance de manter o alvo dentro do campo de visão do sensor binário.

A partir disso, a metodologia utilizada foi a mesma aplicada a *grid search* na Seção 5.1.2. Após explorados o último par de velocidade para cada um destes vetores, foram gerados os mapas de calor apresentados na Figura 19.

Figura 19 – Comparação Mapas de Calor



Fonte: A autora (2025).

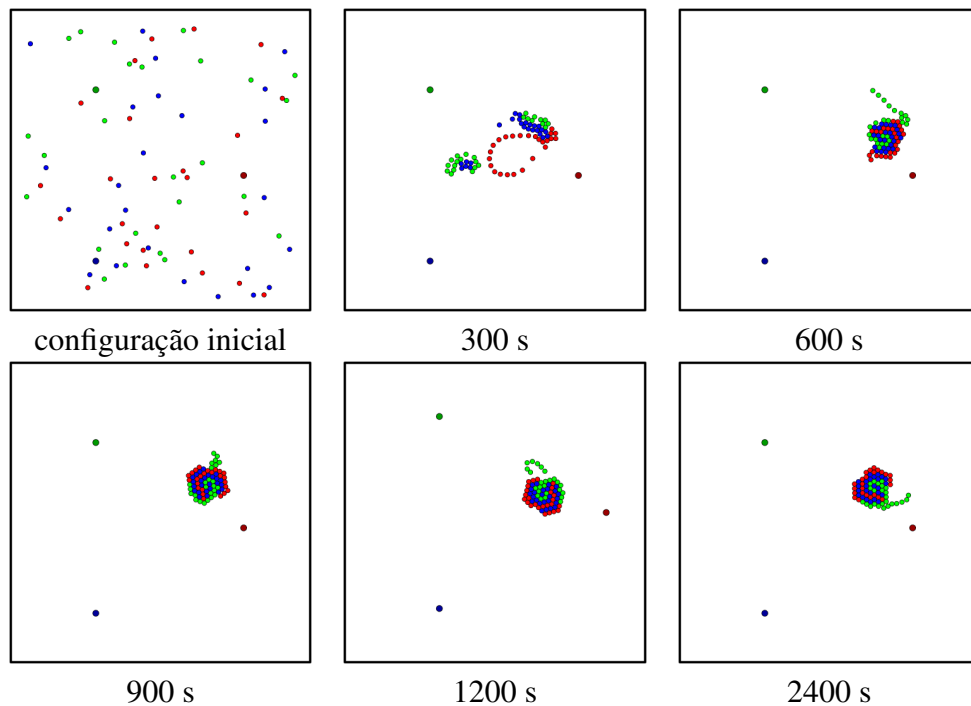
Com estes mapas de calor, foram encontradas as melhores combinações para cada configuração, sendo:

1. $\bar{v}_1 = (0, 50; -0, 40; 0, 40; 0, 50; 0, 40; 0, 20);$
2. $\bar{v}_2 = (0, 60; -0, 90; 0, 30; 0, 60; 0, 80; 0, 70);$
3. $\bar{v}_3 = (0, 70; 0, 30; 0, 60; 1, 00; 0, 30; 0, 20);$
4. $\bar{v}_4 = (-0, 60; -0, 80; 1, 00; -1, 00; -0, 70; -1, 00).$

Deste modo, pode-se notar que o vetor com melhores resultados foi $\bar{v}_4$, e assim como $\bar{v}_0$, encontrou como valores para o último par de velocidades $-0,70$ e $-1,00$. Os demais vetores, mesmo com sua melhor configuração, ainda apresentaram grande ruído em suas métricas de desempenho e uma formação final do enxame insatisfatória, apresentado a seguir. Isso fica

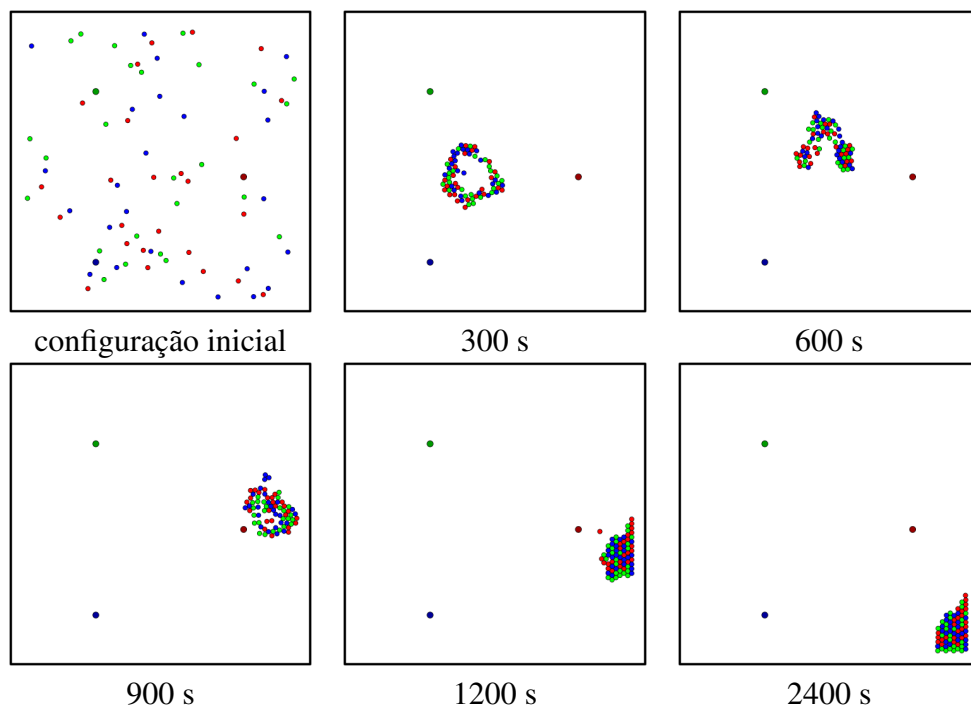
evidente no comportamento ao longo do tempo de cada controlador, mostrado nas Figuras 20, 21, 22 e 23. As simulações foram realizadas com a presença cilindros de referência em todas as configurações de vetores analisadas.

Figura 20 – Comportamento $\bar{v}_1$ para $r = 75$ robôs com a presença de cilindros de referência.



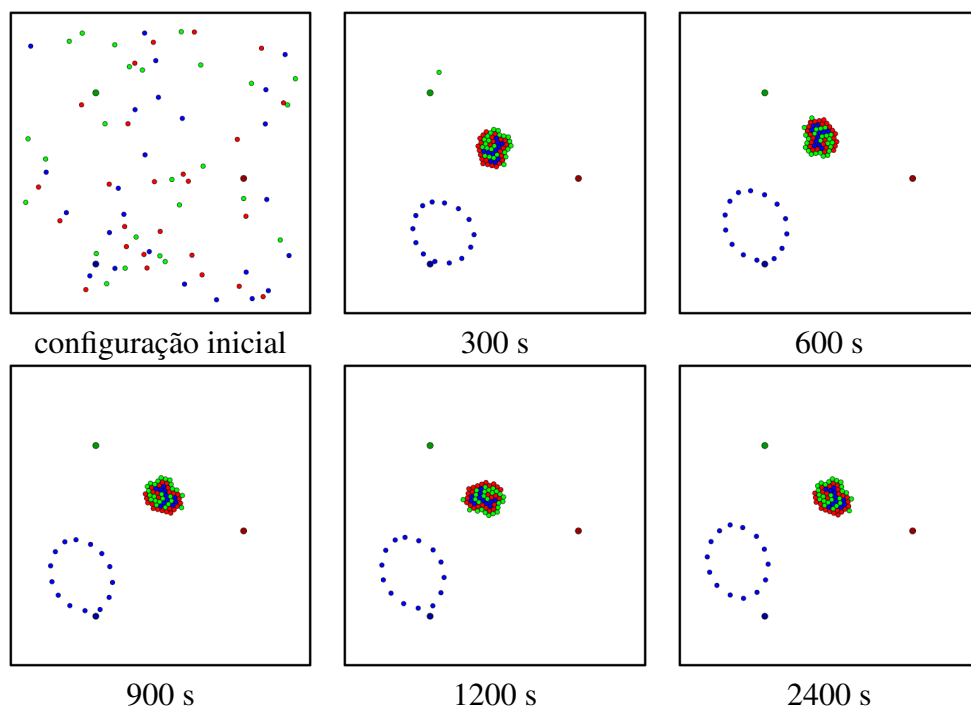
Fonte: A autora (2025).

Figura 21 – Comportamento $\bar{v}_2$ para $r = 75$ robôs com a presença de cilindros de referência.



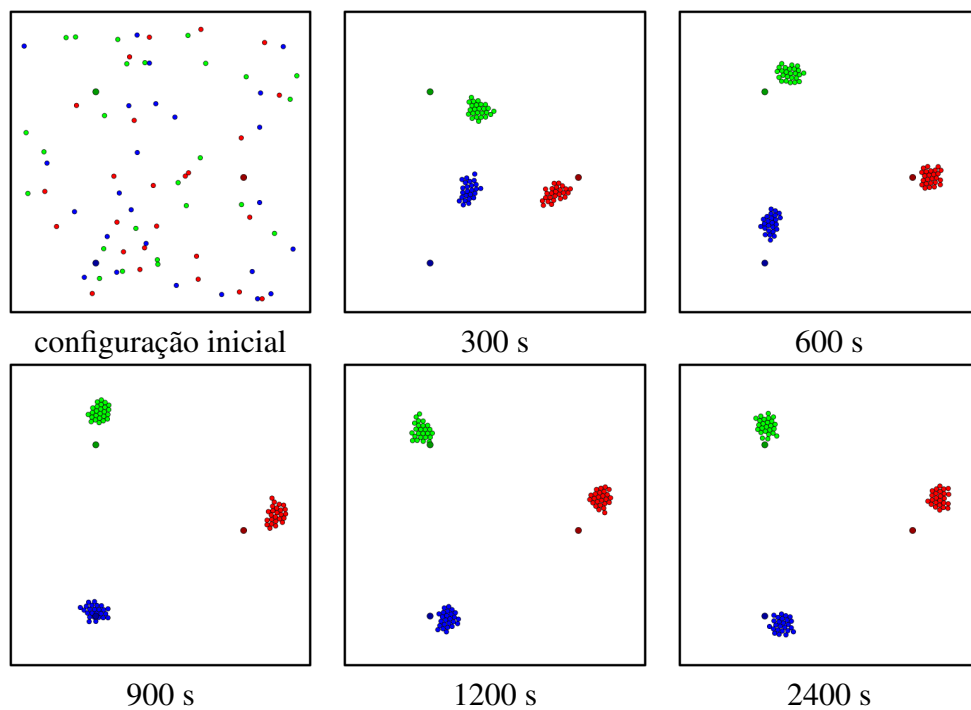
Fonte: A autora (2025).

Figura 22 – Comportamento $\bar{v}_3$ para $r = 75$ robôs com a presença de cilindros de referência.



Fonte: A autora (2025).

Figura 23 – Comportamento $\bar{v}_4$ para $r = 75$ robôs com a presença de cilindros de referência.



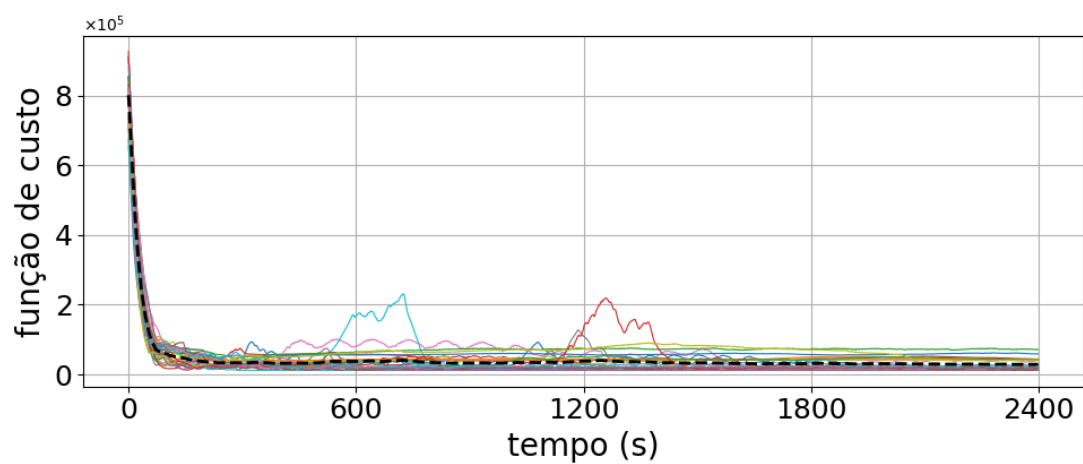
Fonte: A autora (2025).

As métricas de função de custo, dispersão e proporção dos quatro controladores são discutidas a seguir. Na Figura 24, apesar de $\bar{v}_1$ alcançar bons valores para a função de custo e para a dispersão, seus resultados não convergiram para a proporção de agrupamento de robôs.

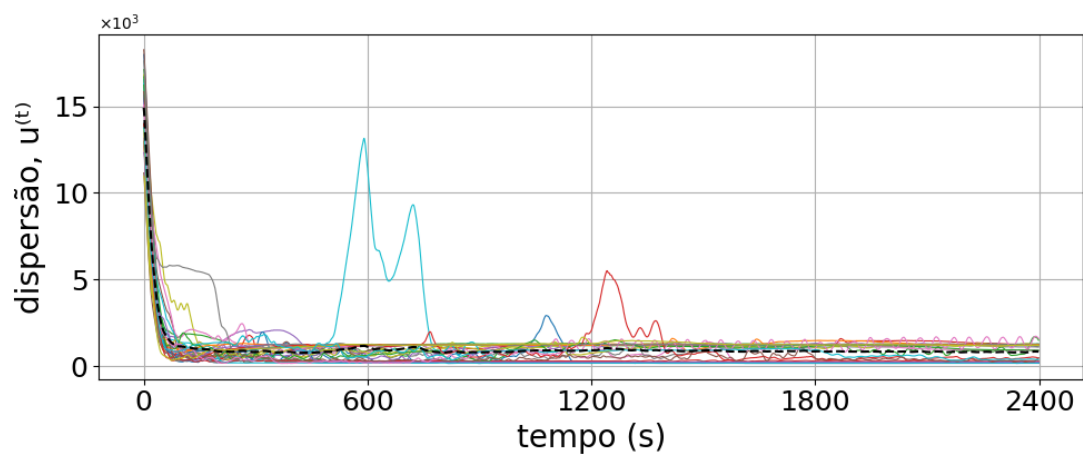
Isso ocorre porque sua formação final, apesar de separar os grupos, gera o formato circular, ou anel, assim como encontrado por Krischanski et al. (2023), vide Figura 20.

Já os vetores $\bar{v}_2$ e $\bar{v}_3$ tiveram desempenhos semelhantes. Conforme visto nas Figuras 21, 22, apesar desses vetores terem uma formação final compacta, eles não foram capazes de segregar os robôs em seus grupos. As Figuras 25 e 26 deixam isso mais evidente, as métricas avaliadas apresentaram muito ruído, não sendo solucionado até o final da simulação, especialmente em relação a proporção de agregação dos robôs. Entretanto, em situações com um único grupo estes vetores podem ser úteis, ainda que não sejam a configuração ótima (Gauci et al., 2014a).

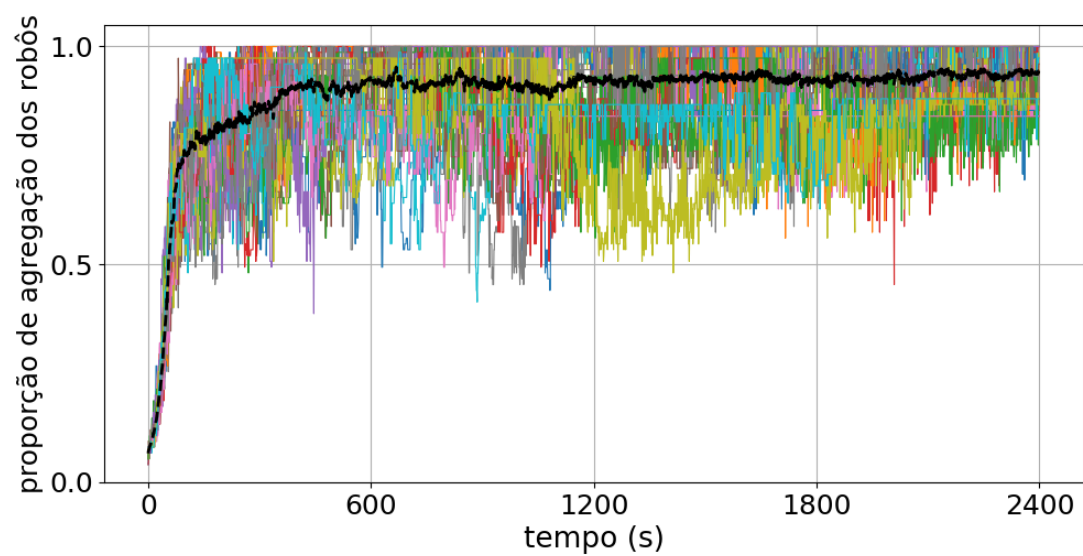
O vetor $\bar{v}_4$, por sua vez, obteve uma formação final satisfatória, com grupos compactados, segregados e espalhados pelo ambiente virtual, conforme mostrou a Figura 23. Também apresentou pouco ruído em suas métricas de desempenho, as quais convergiram bem e em um tempo inferior a 1200 s, conforme pode ser visto na Figura 27. Além disso, por ter sido a melhor configuração avaliada, $\bar{v}_4$ foi explorado a partir dos experimentos de escalabilidade. Ele demonstrou ser escalável e consistente, conforme mostram as Figuras 28, 29, 30 e 31.

Figura 24 – Resultados $\bar{v}_1$ 

(a) Função de Custo

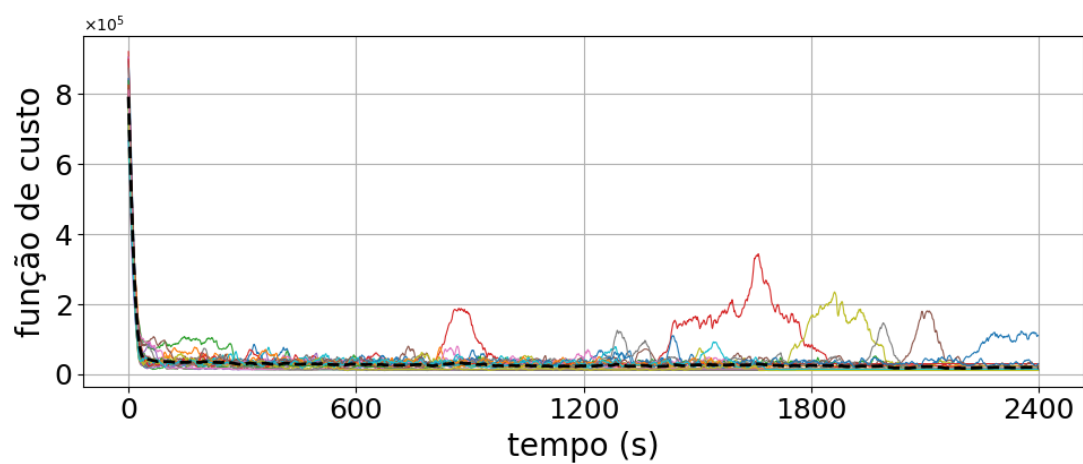


(b) Dispersão

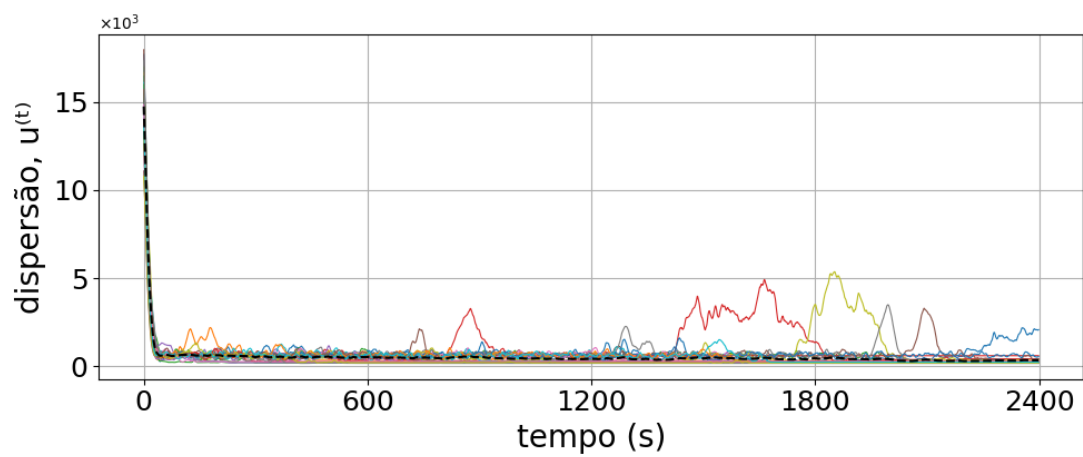


(c) Proporção

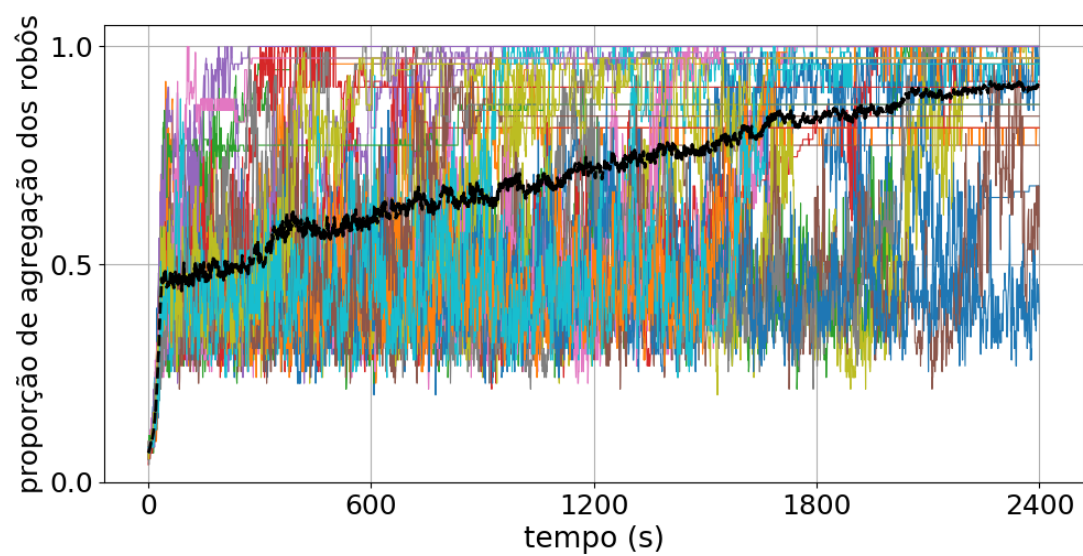
Fonte: A autora (2025).

Figura 25 – Resultados $\bar{v}_2$ 

(a) Função de Custo

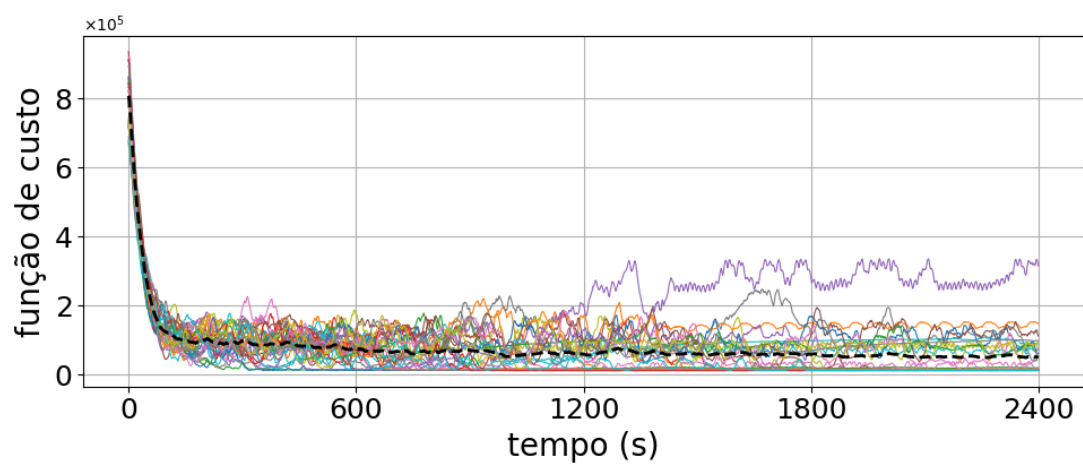


(b) Dispersão

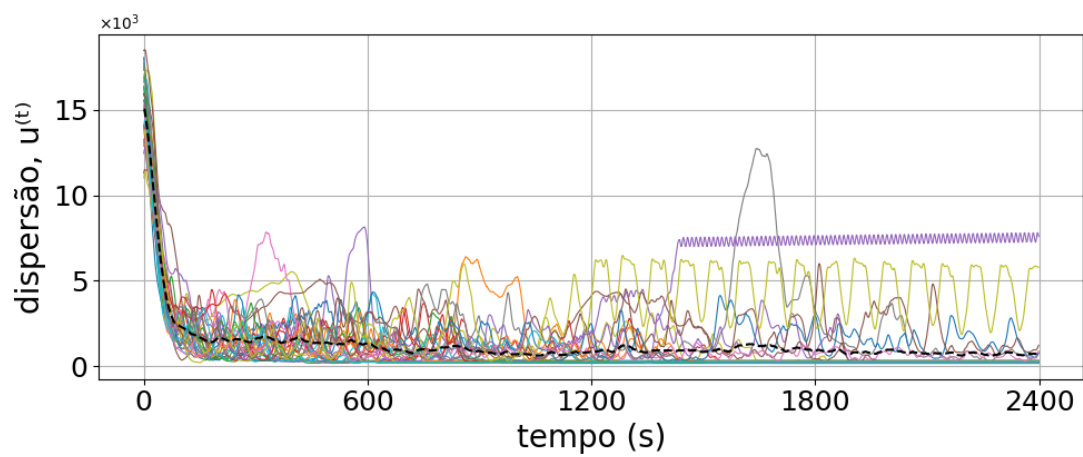


(c) Proporção

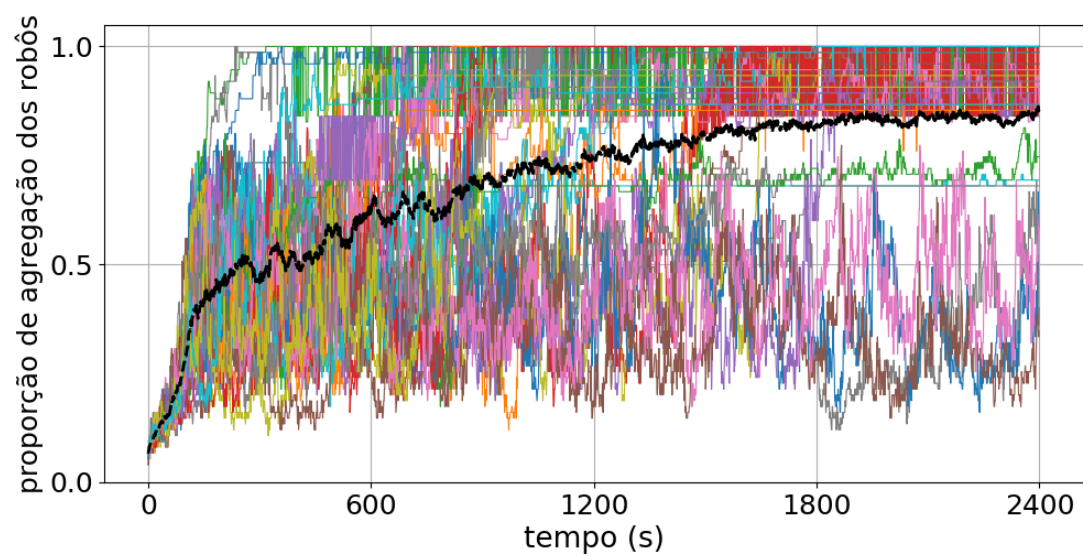
Fonte: A autora (2025).

Figura 26 – Resultados $\bar{v}_3$ 

(a) Função de Custo

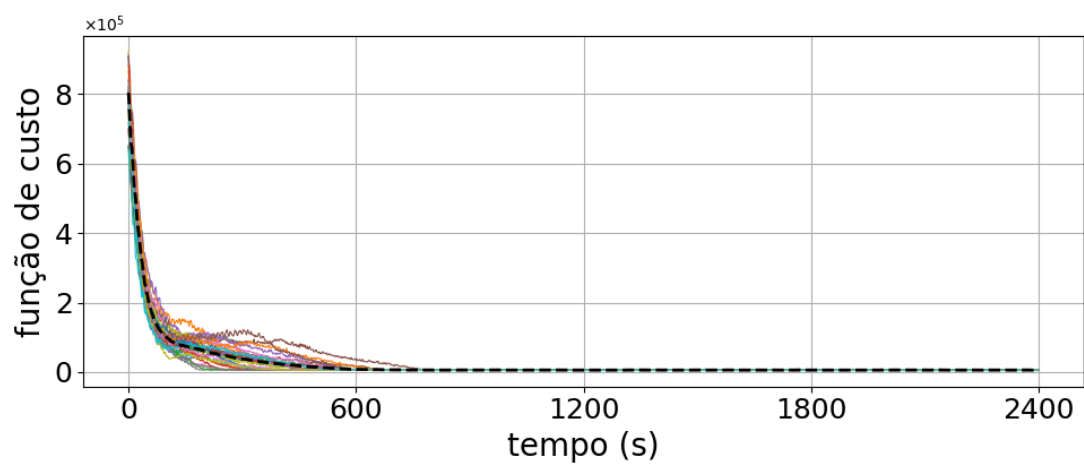


(b) Dispersão

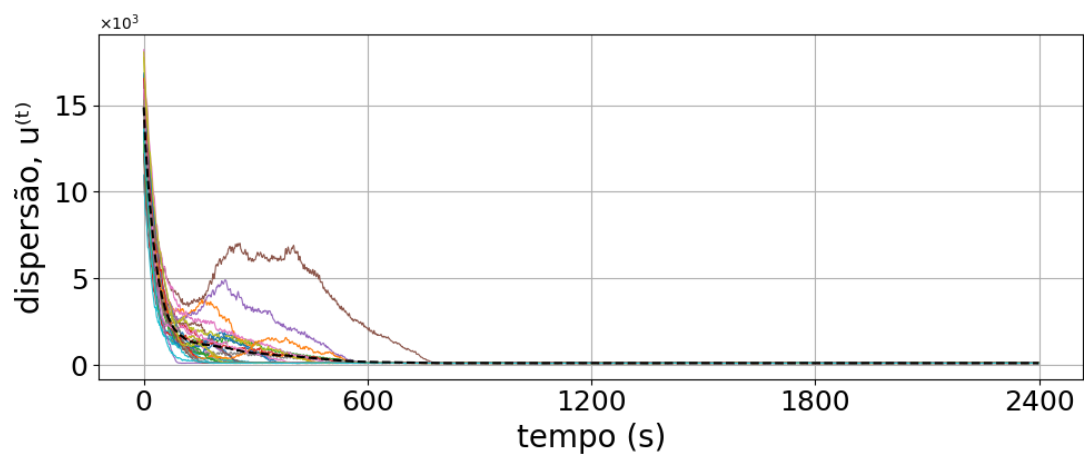


(c) Proporção

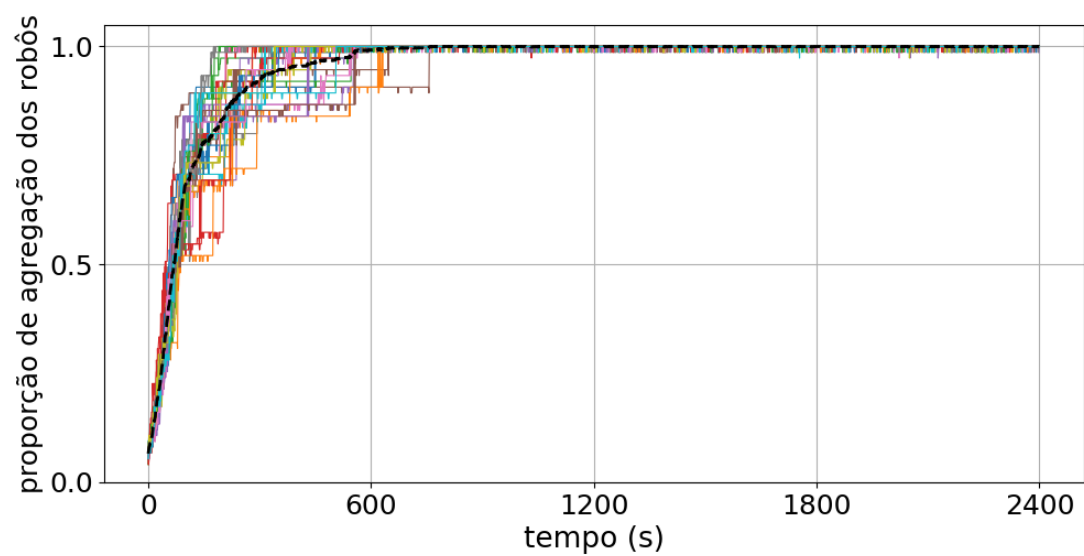
Fonte: A autora (2025).

Figura 27 – Resultados $\bar{v}_4$ 

(a) Função de Custo



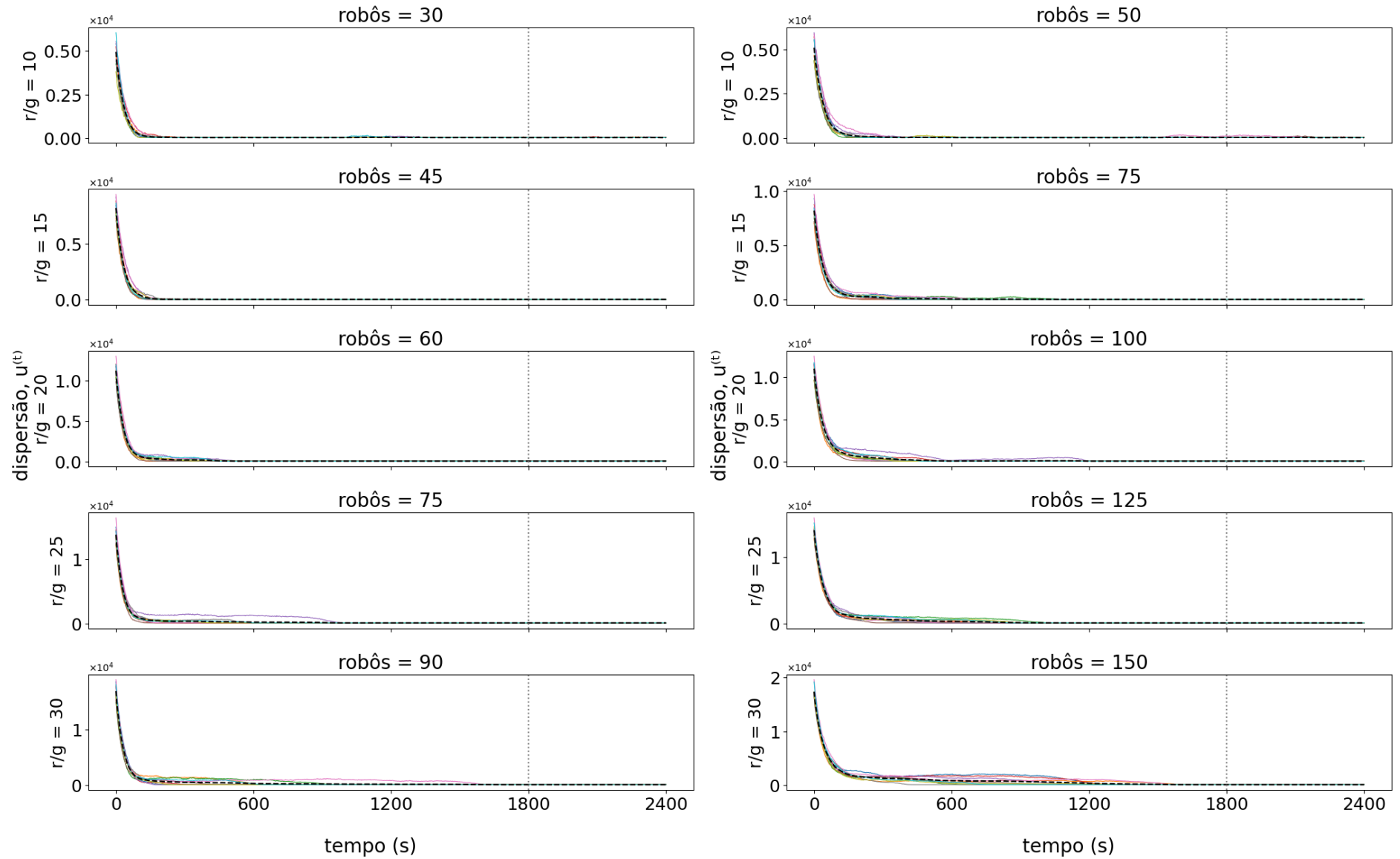
(b) Dispersão



(c) Proporção

Fonte: A autora (2025).

Figura 28 – Dispersão $\bar{v}_4$ Sem Cilindro de Referência



(a) Sem Cilindro de Referência - 3 grupos

(b) Sem Cilindro de Referência - 5 grupos

Fonte: A autora (2025).

Figura 29 – Dispersão $\bar{v}_4$ Com Cilindro de Referência

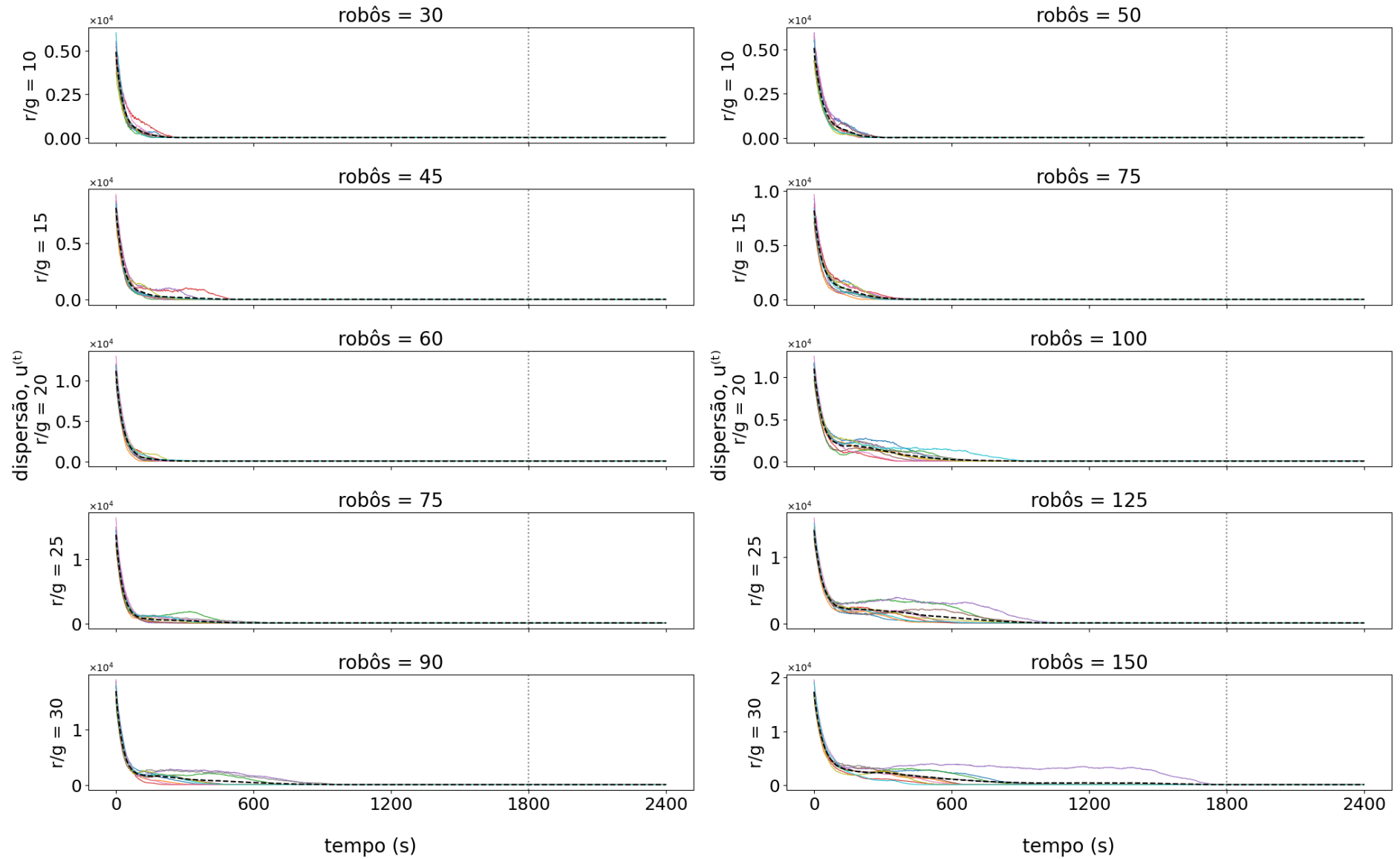
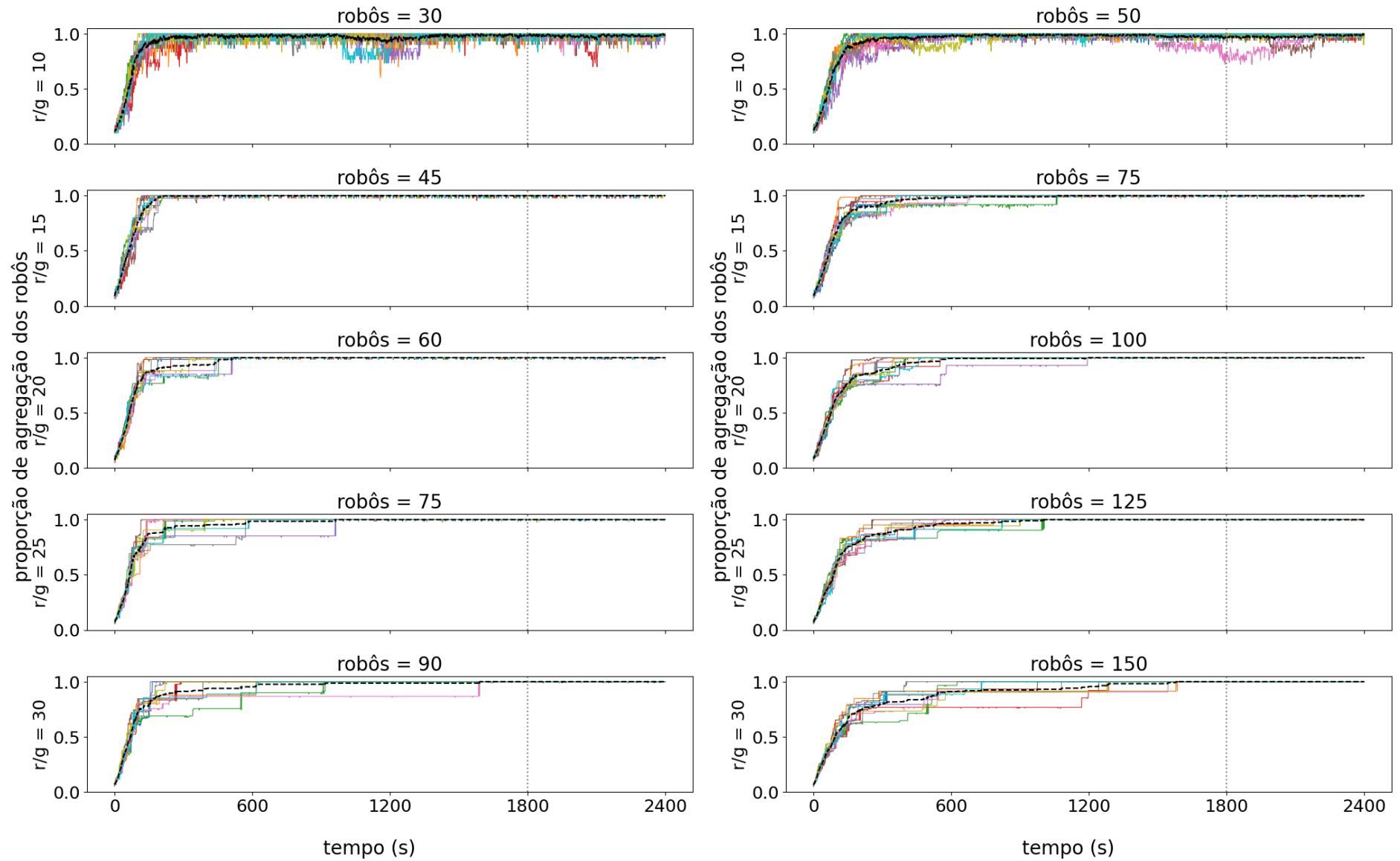


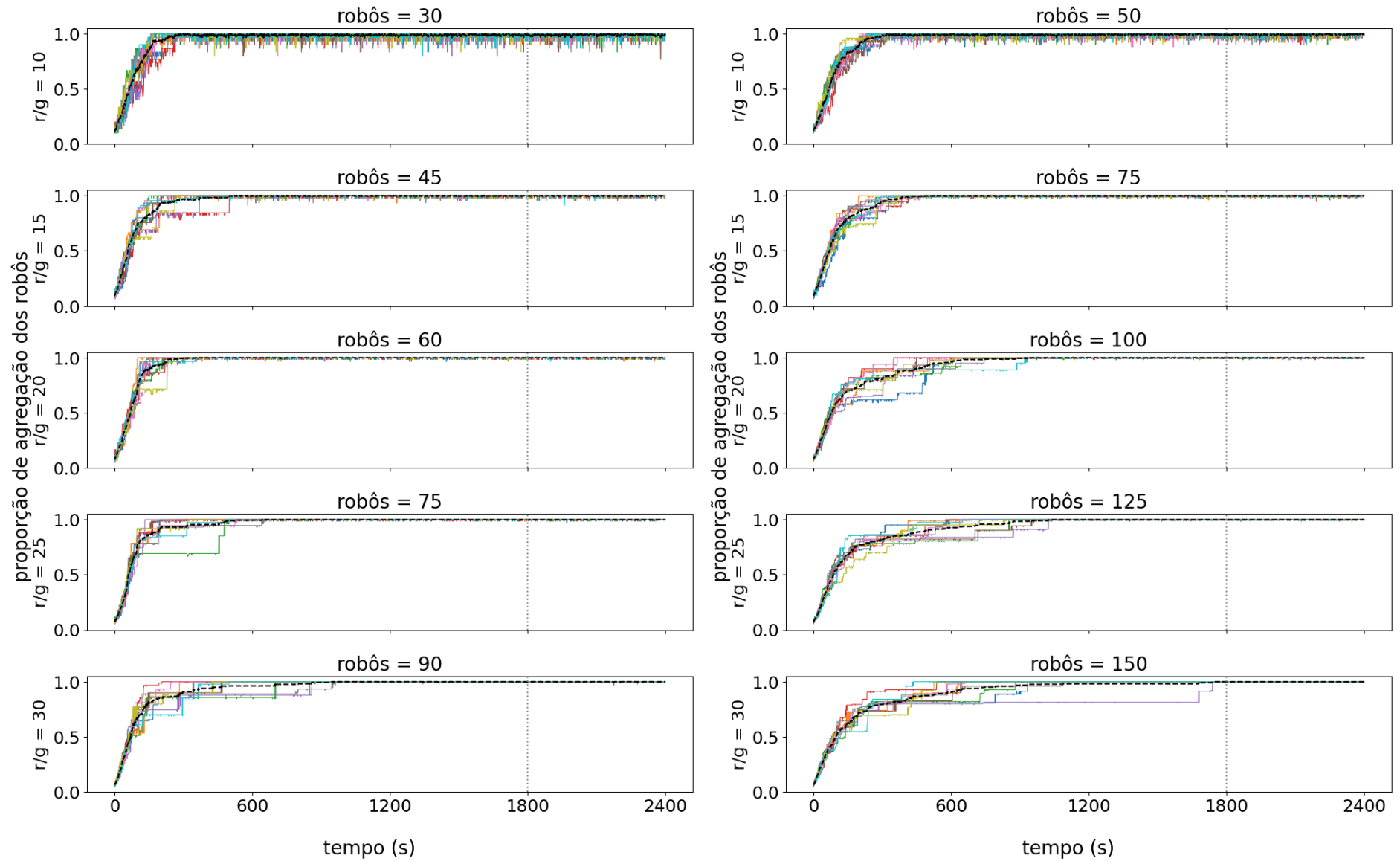
Figura 30 – Proporção $\bar{v}_4$ Sem Cilindro de Referência



(a) Sem Cilindro de Referência - 3 grupos

(b) Sem Cilindro de Referência - 5 grupos

Figura 31 – Proporção $\bar{v}_4$ Com Cilindro de Referência



(a) Com Cilindro de Referência - 3 grupos

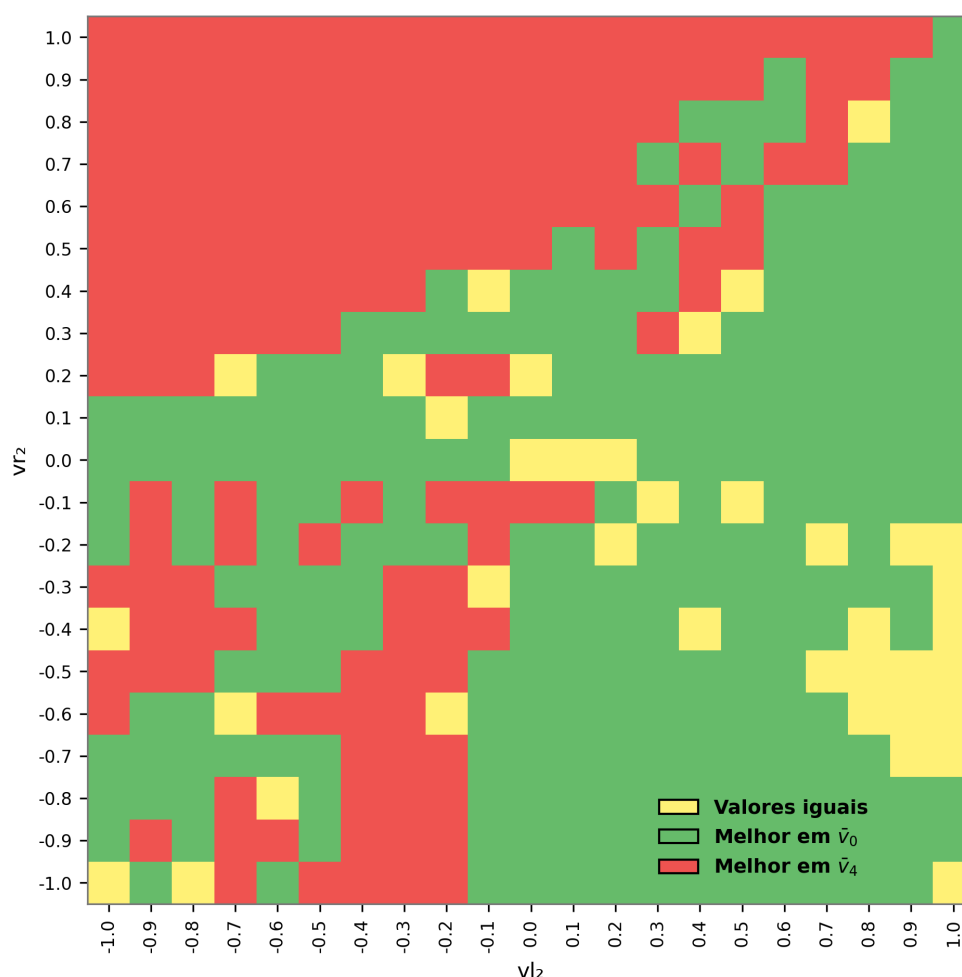
(b) Com Cilindro de Referência - 5 grupos

5.1.4 Comparação dos Resultados

A partir destes resultados, $\bar{v}_0$ e $\bar{v}_4$ foram comparados, a fim de identificar, por meio dos mapas de calor, configurações que tenham boa avaliação e possam ser exploradas. A partir da Figura 32 é possível fazer algumas observações.

Para melhor compreensão, pode-se traçar uma diagonal da origem até (1,00;1,00) e desconsiderar o triângulo superior esquerdo, uma vez que representa uma região com soluções ruins para os mapas de calor originais de cada vetor. Assim, pode-se dizer que $\bar{v}_0$ é superior na maior parte das combinações com bons resultados, região direita inferior da Figura 32, na qual verde indica soluções melhores para o vetor $\bar{v}_0$, vermelhas são melhores para $\bar{v}_4$ e amarelo são resultados iguais em ambos os vetores.

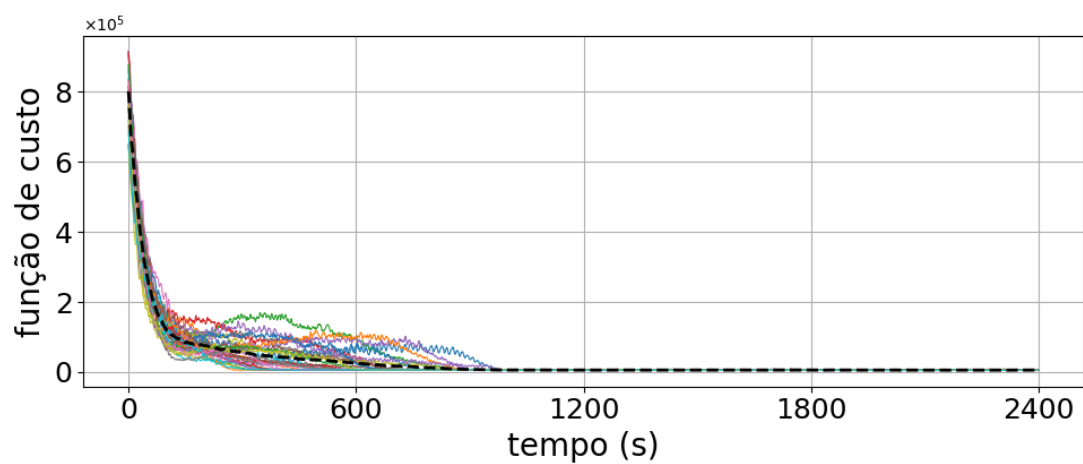
Figura 32 – Intersecção Mapas de Calor $\bar{v}_0$ e $\bar{v}_4$



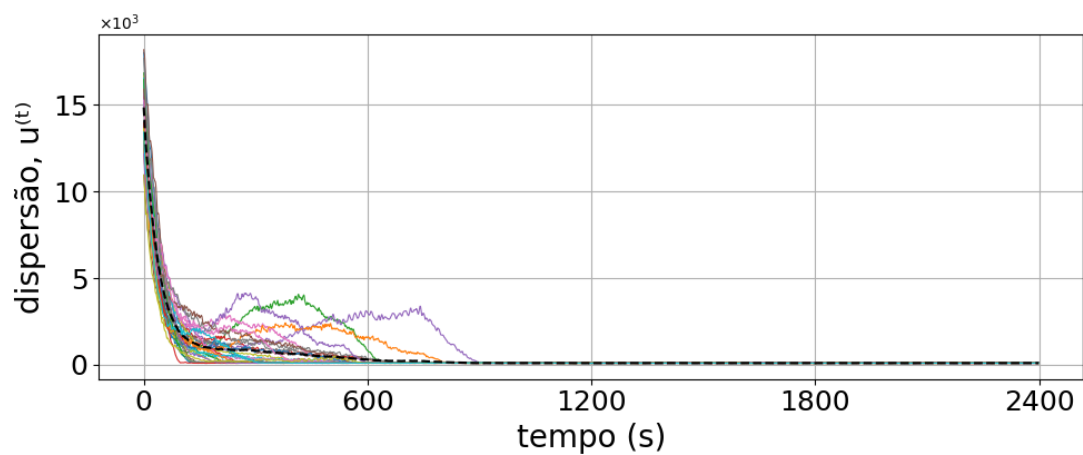
Fonte: A autora (2025).

Com isso, foi selecionado um novo par de velocidade em que $\bar{v}_4$ é superior a $\bar{v}_0$, a fim de confirmar seu desempenho mesmo com pequenas variações. Este vetor foi nomeado de $\bar{v}_{4a}$ e é dado por $\bar{v}_{4a} = (-0,60; -0,80; 1,00; -1,00; -0,40; -0,50)$.

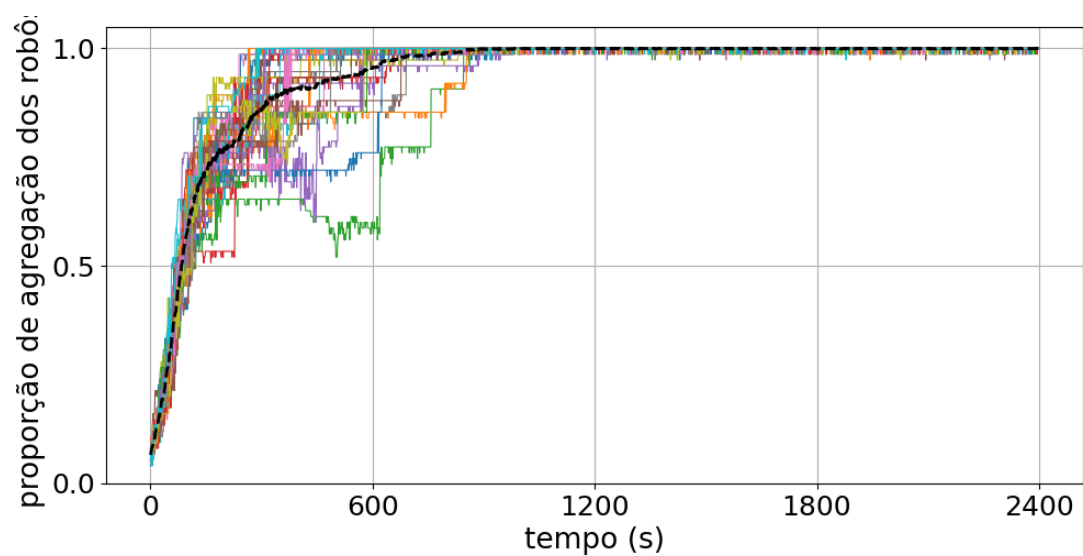
Seguindo a mesma metodologia utilizada, a avaliação pelas métricas de desempenho das simulações padronizadas mostraram o desempenho de $\bar{v}_{4a}$, apresentada na Figura 33.

Figura 33 – Resultados $\bar{v}_{4a}$ 

(a) Função de Custo



(b) Dispersão



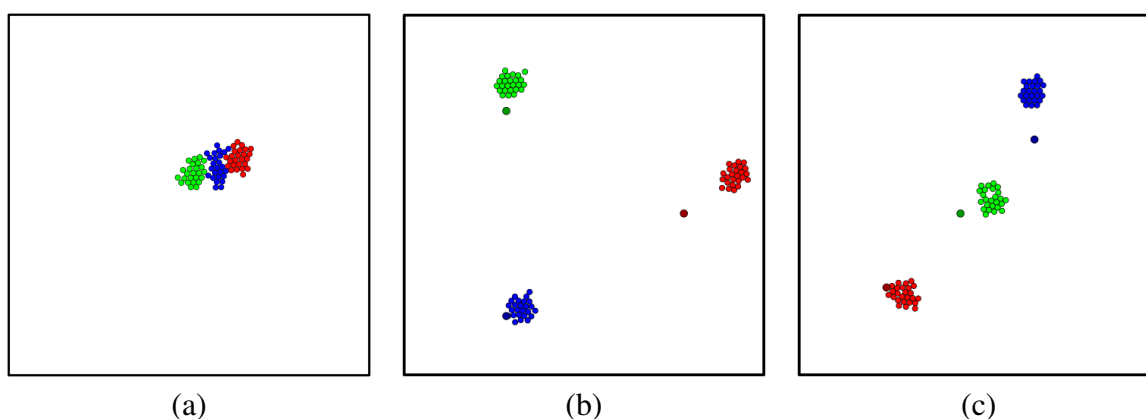
(c) Proporção

Fonte: A autora (2025).

Em relação a seu comportamento final nas simulações, ocorreu conforme esperado. Os grupos tenderam a ficar no meio do ambiente virtual, o que é facilmente resolvido com cilindro de referência no ambiente, conforme visto na Figura 34 (a).

Ainda na Figura 34, uma nova configuração de disposição dos cilindros de referência no ambiente foi avaliada, com eles distribuídos em diagonal pelo ambiente. Isso foi feito com o objetivo de avaliar o impacto da disposição dos cilindros de referência sobre o resultado final da simulação e observar como o vetor converge em diferentes cenários.

Figura 34 – (a) sem cilindros de referência (b) com cilindros de referência em modo circular (c) com cilindros de referência em modo diagonal.

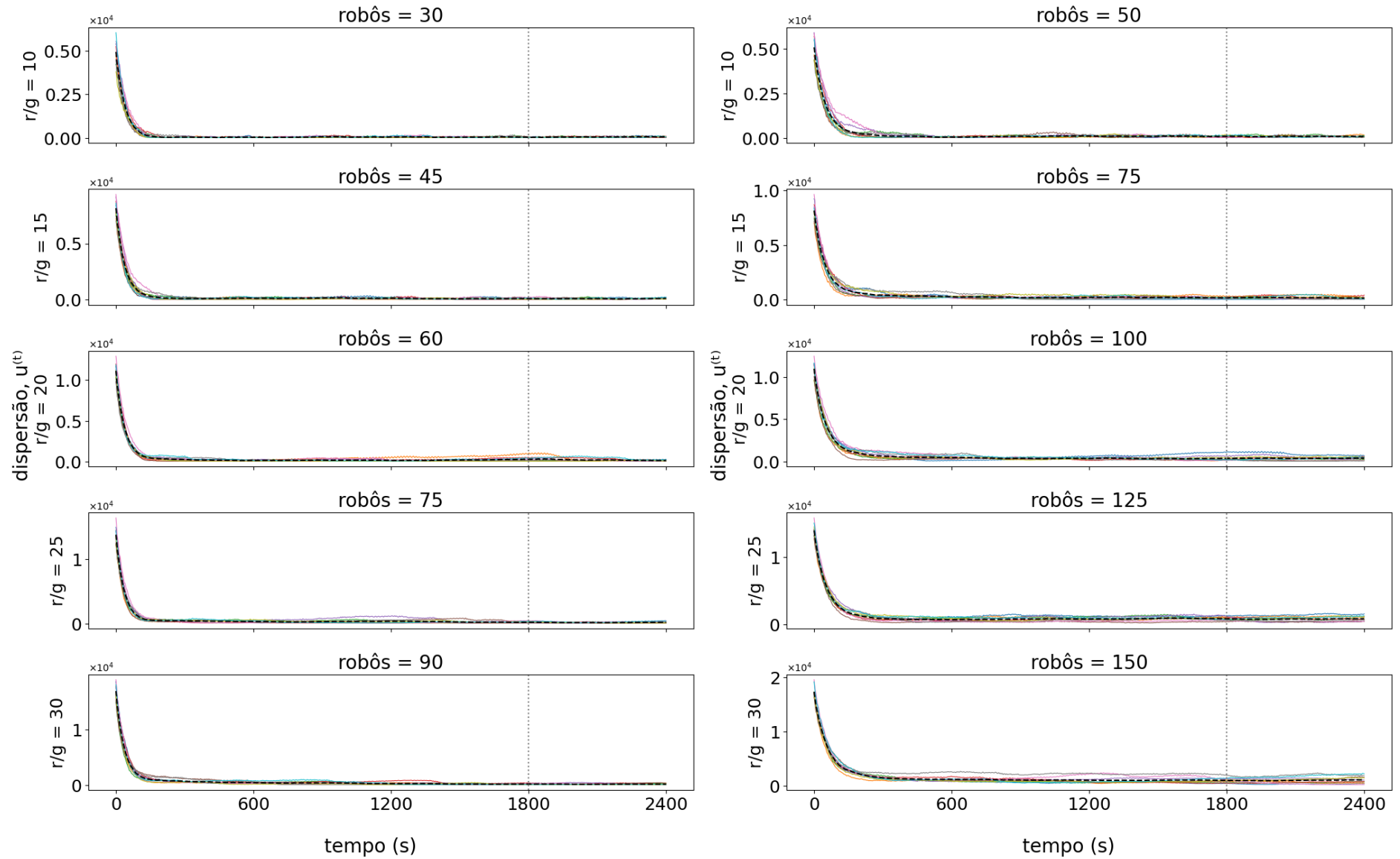


Fonte: A autora (2025).

Deste modo, $\bar{v}_{4_u}$ foi considerada uma solução válida para o problema de agregação de múltiplos robôs. Afinal, mesmo com uma nova disposição dos cilindros de referência, alcançou a correta agregação dos grupos.

Ademais, quanto à escalabilidade, o vetor manteve seu desempenho consistente mesmo com enxames maiores quando havia presença de cilindros de referência na arena. Sem eles, o ruído é alto para um número igual ou superior a 50 robôs totais, em termos de proporção de agrupamentos. Essas observações foram extraídas da interpretação das Figuras 35, 36, 37 e 38. Além disso, estes resultados também mostram-se superiores ao encontrado por Krischanski et al. (2023).

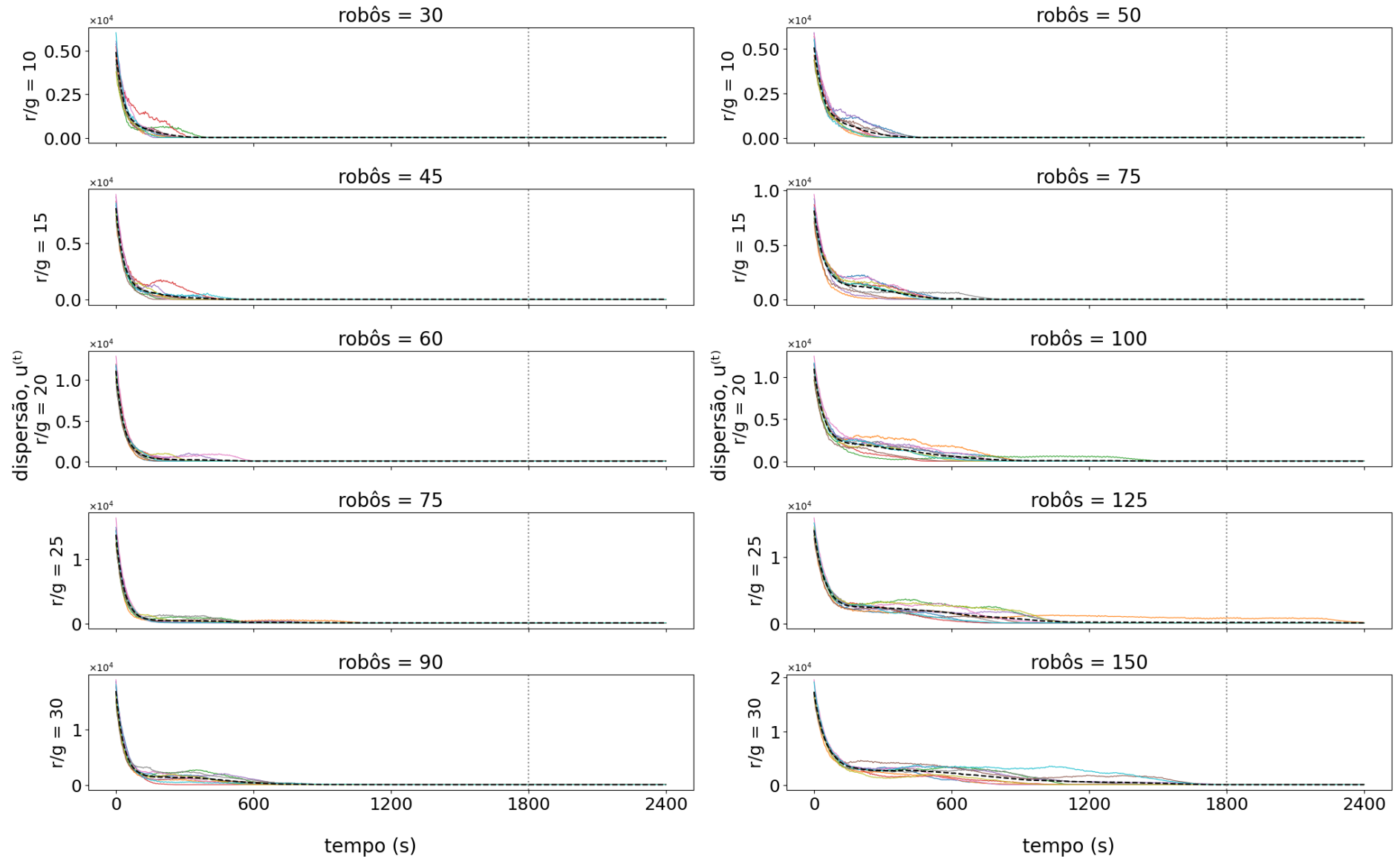
Figura 35 – Dispersão $\bar{v}_{4a}$ Sem Cilindro de Referência



(a) Sem Cilindro de Referência - 3 grupos

(b) Sem Cilindro de Referência - 5 grupos

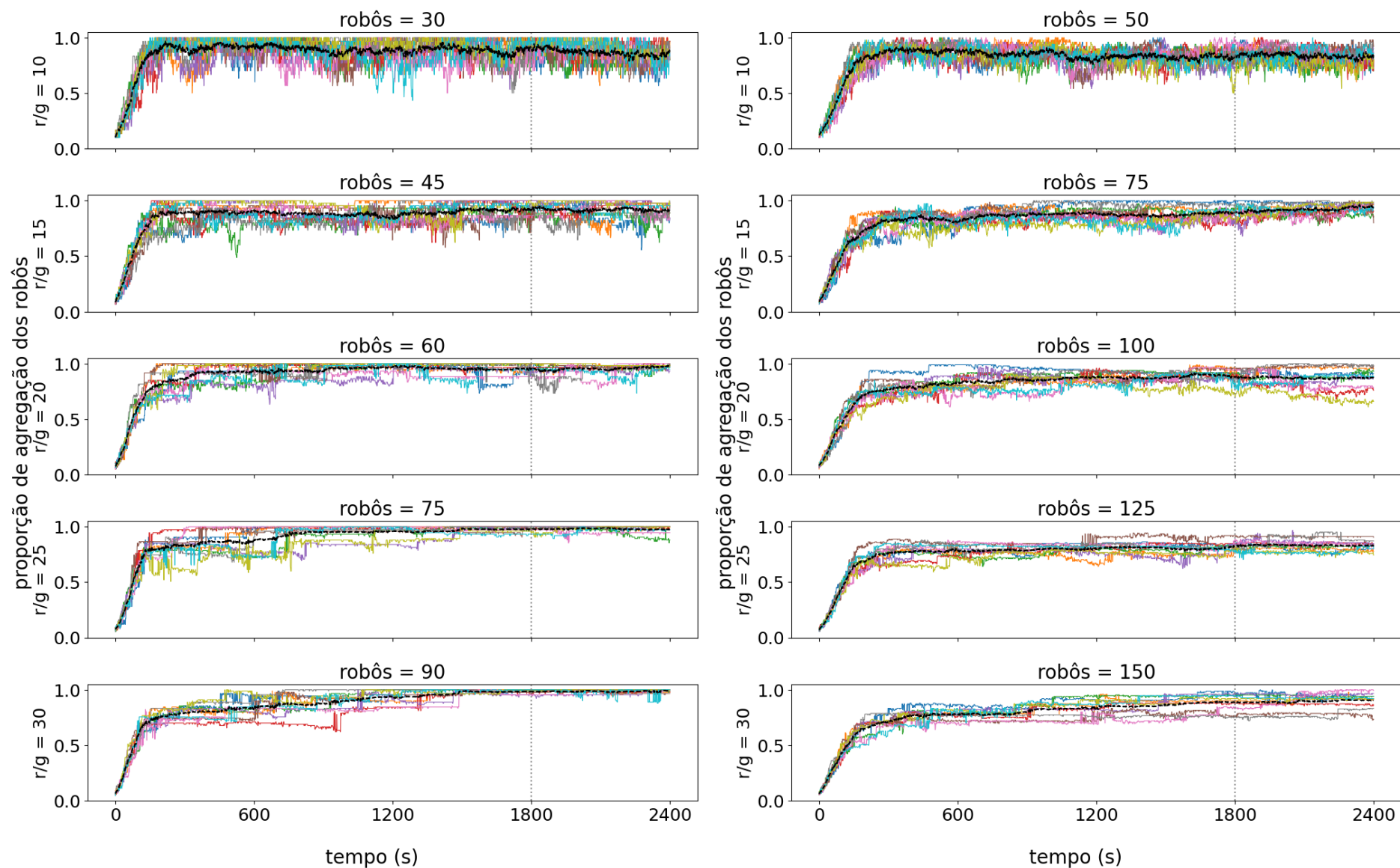
Figura 36 – Dispersão $\tilde{v}_{4a}$ Com Cilindro de Referência



(a) Com Cilindro de Referência - 3 grupos

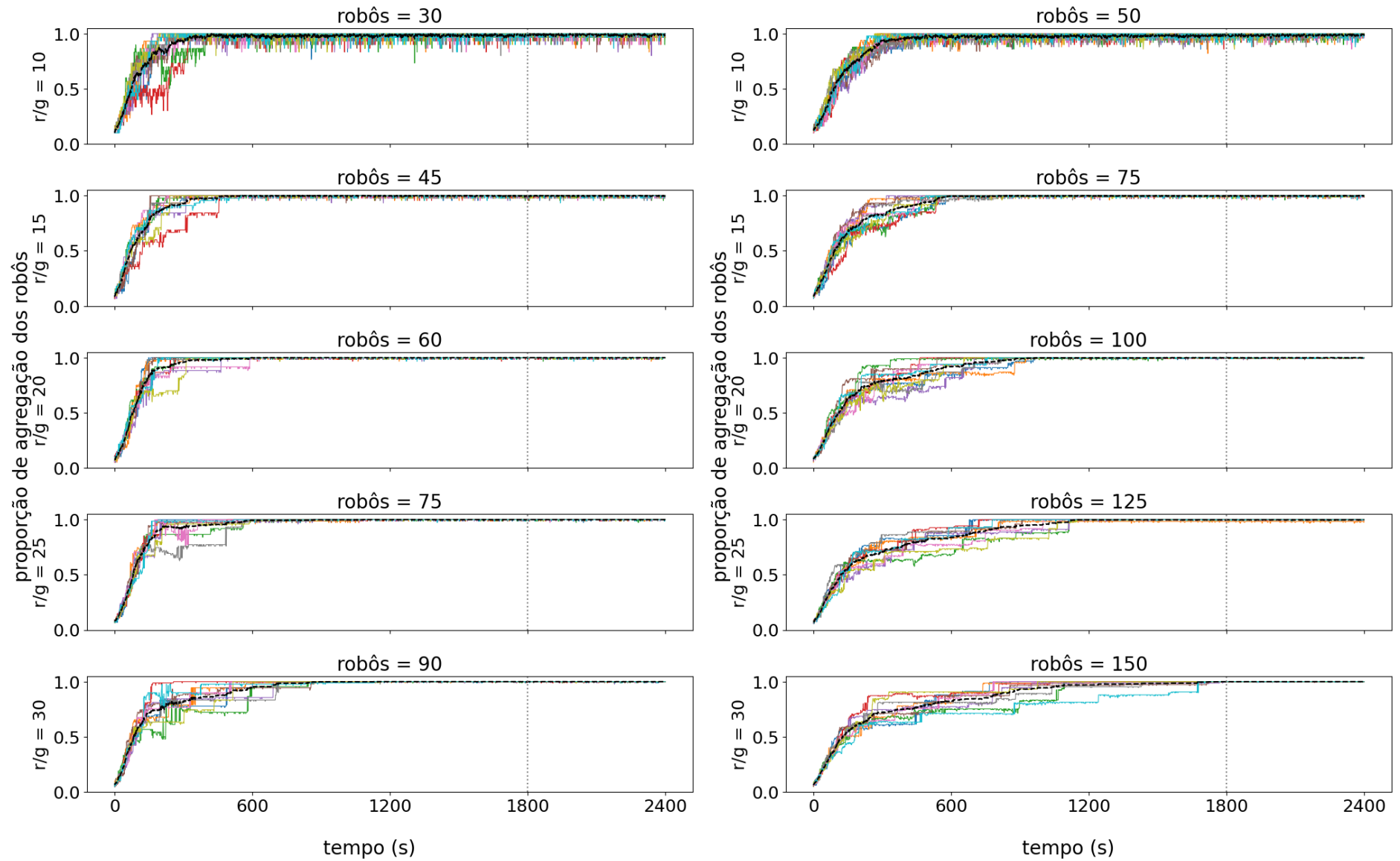
(b) Com Cilindro de Referência - 5 grupos

Figura 37 – Proporção $\bar{v}_{4_a}$ Sem Cilindro de Referência



Fonte: A autora (2025).

Figura 38 – Proporção $\bar{v}_{4_a}$ Com Cilindro de Referência



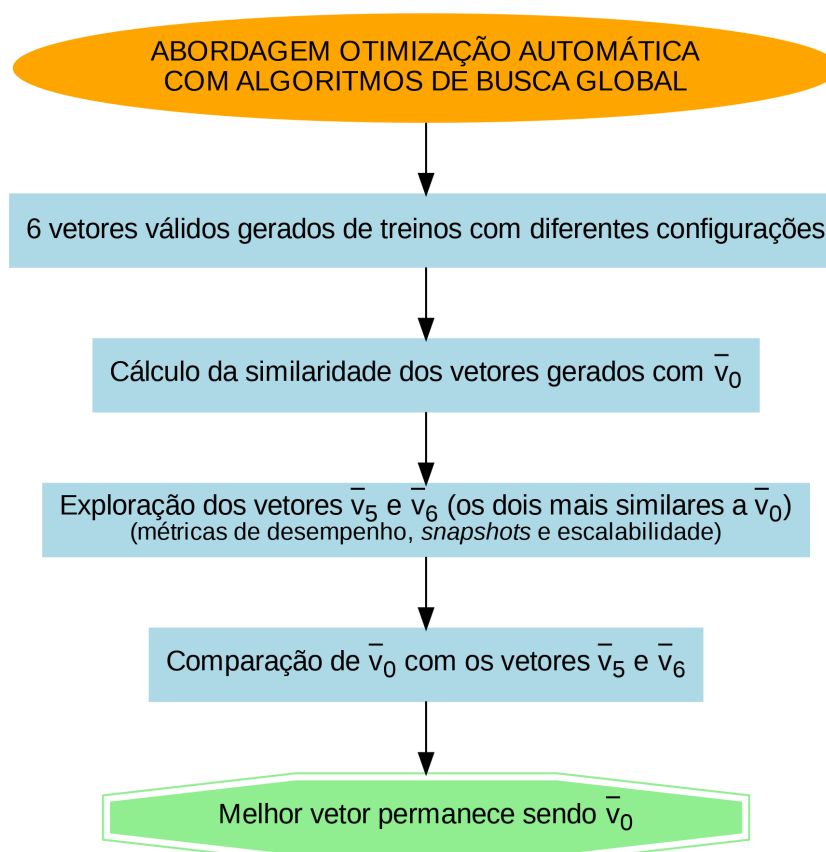
(a) Com Cilindro de Referência - 3 grupos

(b) Com Cilindro de Referência - 5 grupos

5.2 ALGORITMOS DE OTIMIZAÇÃO

Enquanto a *grid search* foi eficaz para explorar um subespaço de parâmetros, a segunda fase experimental investigou a capacidade de algoritmos de otimização automática para explorar o espaço de busca completo do controlador, com seis graus de liberdade. Nesta fase o objetivo foi avaliar se abordagens heurísticas poderiam, de forma mais eficiente, encontrar soluções próximas a encontrada pelo método exaustivo. Na Figura 39, são resumidas as etapas executadas nessa abordagem.

Figura 39 – Etapas da abordagem de otimização por algoritmos de busca global.



Fonte: A autora (2026).

Conforme a metodologia, foram avaliados diferentes algoritmos, incluindo o DIRECT, utilizado por Krischanski et al. (2023), e abordagens híbridas do algoritmo global MLSL combinado aos algoritmos locais SBPLX e COBYLA. Assim, foram avaliadas 25 configurações de treino, a fim de encontrar a abordagem que oferece a melhor solução para a agregação minimalista multitarefa.

Para uma configuração ser considerada bem-sucedida, o controlador resultante precisava demonstrar, ao final da simulação padronizada, a capacidade de formar agregados compactos e garantir a segregação entre os diferentes grupos. O algoritmo DIRECT não foi capaz de encontrar uma solução que atendesse a esses critérios em nenhum dos treinamentos realizados.

Em contrapartida, a abordagem híbrida MLSL e SBPLX mostrou-se a mais promissora, sendo responsável pela maioria dos bons controladores encontrados, em treinos de configurações distintas. Isso pode ser observado na Tabela 1, a qual apresenta um resumo dos melhores resultados obtidos por esta abordagem, com os vetores de velocidade que geraram o comportamento desejado.

As velocidades são apresentadas com seus valores normalizados e a similaridade de cada solução com o vetor $\bar{\mathbf{v}}_0$ (Simil. $\bar{\mathbf{v}}_0$), encontrado por *grid search*, também é calculada como métrica de comparação. Os vetores comparados são:

- $\bar{\mathbf{v}}_A = (-1,00; -0,62; -0,61; 1,00; -1,00; -0,37);$
- $\bar{\mathbf{v}}_B = (-1,00; -0,65; -0,99; 1,00; -1,00; -0,40);$
- $\bar{\mathbf{v}}_C = (-0,71; -1,00; 1,00; -1,00; -0,49; -0,80);$
- $\bar{\mathbf{v}}_D = (-0,59; -0,95; 1,00; -0,51; -0,30; -1,00);$
- $\bar{\mathbf{v}}_E = (-0,56; -1,00; 1,00; -0,96; -0,39; -1,00);$
- $\bar{\mathbf{v}}_F = (-0,94; -0,53; -0,26; 1,00; -0,81; -0,26).$

Tabela 1 – Resultados dos experimentos de otimização com Algoritmo Global MLSL

Simil. $\bar{\mathbf{v}}_0$	A. L.	T (s)	t	Aval.	V	M. A.	F. C.
86,16%	SBPLX	180	30	1000	$\bar{\mathbf{v}}_A$	334	$9,69 \times 10^{11}$
93,02%	SBPLX	180	100	6000	$\bar{\mathbf{v}}_B$	4950	$9,75 \times 10^{11}$
94,23%	SBPLX	300	10	1000	$\bar{\mathbf{v}}_C$	318	$2,62 \times 10^{12}$
78,89%	SBPLX	300	100	10000	$\bar{\mathbf{v}}_D$	1113	$9,97 \times 10^{11}$
90,95%	SBPLX	300	100	10000	$\bar{\mathbf{v}}_E$	707	$2,47 \times 10^{12}$
57,02%	COBYLA	300	100	10000	$\bar{\mathbf{v}}_F$	364	$2,51 \times 10^{12}$
–	–	–	–	–	$\bar{\mathbf{v}}_0$	–	$2,16 \times 10^{15}$

Fonte: A autora (2025).

Na Tabela 1, Simil. $\bar{\mathbf{v}}_0$ refere-se à similaridade com o vetor $\bar{\mathbf{v}}_0$ em %, seu cálculo é apresentado na Equação 17; A.L., algoritmo local utilizado; T, tempo de treinamento (em segundos); t, o número de testes por avaliação; Aval., total de avaliações realizadas; V, os vetores de velocidades (em *cm/s*); M.A., a melhor avaliação obtida; F.C., o valor da função de custo correspondente.

Foram analisadas 25 configurações, das quais foram selecionadas aquelas capazes de agregar os robôs e manter a separação entre os grupos para uma exploração mais aprofundada. Nessa etapa, observou-se que, embora algumas dessas configurações apresentassem bons resultados em termos de agregação e segregação dos grupos, elas enfrentavam certa dificuldade em manter os grupos afastados entre si, devido à tendência de convergência para a região central da arena virtual.

Adicionalmente, nota-se pela Tabela 1 que os vetores que promovem bons resultados seguem uma estrutura semelhante a $\bar{\mathbf{v}}_0$, conforme:

$$\bar{\mathbf{v}}_{0_b} = (-\bar{v}_{\ell,0}, -\bar{v}_{r,0}, \bar{v}_{\ell,1}, -\bar{v}_{r,1}, -\bar{v}_{\ell,2}, -\bar{v}_{r,2}) \quad (15)$$

Também observa-se um padrão entre os vetores, caracterizado pela organização dos pares laterais (1-2 e 5-6) com velocidades negativas de magnitude assimétrica, onde o segundo elemento do par geralmente apresenta maior intensidade. O par central mantém uma oposição de sinais, tipicamente contendo um valor positivo máximo e um valor negativo, representando movimentos diferenciais para rotação do robô. Os vetores analisados são simétricos, ou seja, o comportamento e o desempenho do sistema permanecerão válidos para:

$$\bar{\mathbf{v}}_{0_c} = (-\bar{v}_{r,0}, -\bar{v}_{\ell,0}, -\bar{v}_{r,1}, \bar{v}_{\ell,1}, -\bar{v}_{r,2}, -\bar{v}_{\ell,2}) \quad (16)$$

Ademais, como os valores de velocidade negativos indicam movimento para trás, isso permite que os robôs se afastem de elementos que não pertencem ao seu grupo e, ao mesmo tempo, se aproximem do agrupamento, mantendo uma percepção mais ampla do ambiente. Além disso, a comparação dos módulos entre os elementos de um par não mantém uma uniformidade de lado (esquerdo ou direito) dominante ao longo de todo o vetor, ou seja, se em um par o módulo do lado esquerdo é maior, isso não implica que será maior nos demais pares. Contudo, em $\bar{\mathbf{v}}_0$ isso é observado e parece ser um dos motivos pelos quais os vetores encontrados não apresentam um desempenho totalmente satisfatório quando comparados com ele.

A análise comparativa dos algoritmos mostrou que a combinação MLSL e SBPLX se destacou significativamente, demonstrando maior eficácia ao encontrar soluções satisfatórias quando comparada a MLSL e COBYLA, o qual em apenas um treino específico encontrou um bom vetor.

Posteriormente, tendo em vista que $\bar{\mathbf{v}}_0$ não foi superado, o erro médio relativo normalizado entre suas velocidades e dos demais vetores foi calculado, por meio das Equações 17 e 18. Isso foi feito a fim de permitir selecionar um dos vetores para exploração de suas métricas de desempenho.

$$E = \frac{1}{n} \sum_{i=1}^n \frac{|v_i - \bar{v}_i|}{2|\bar{v}_i|} \quad (17)$$

Onde:

- E representa o erro médio relativo normalizado;
- n é o número total de parâmetros;
- v_i é o valor do parâmetro i no vetor a ser avaliado;
- $2|\bar{v}_i|$ é o valor do parâmetro i no vetor de referência $\bar{\mathbf{v}}_0$, multiplicado pelo fator de normalização correspondente à amplitude total do intervalo operacional $[-1, 1]$.

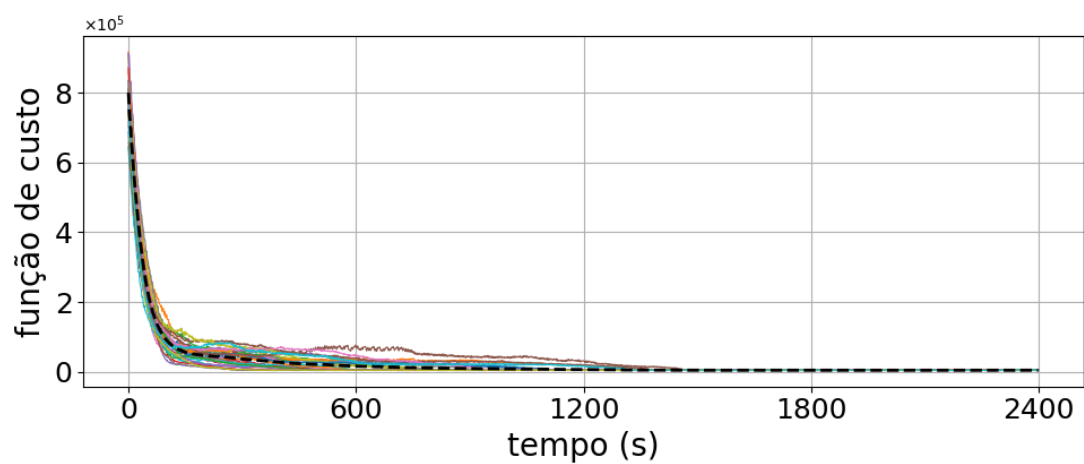
e subsequentemente convertida para similaridade percentual (S) através da relação:

$$S = (1 - E) \times 100 \quad (18)$$

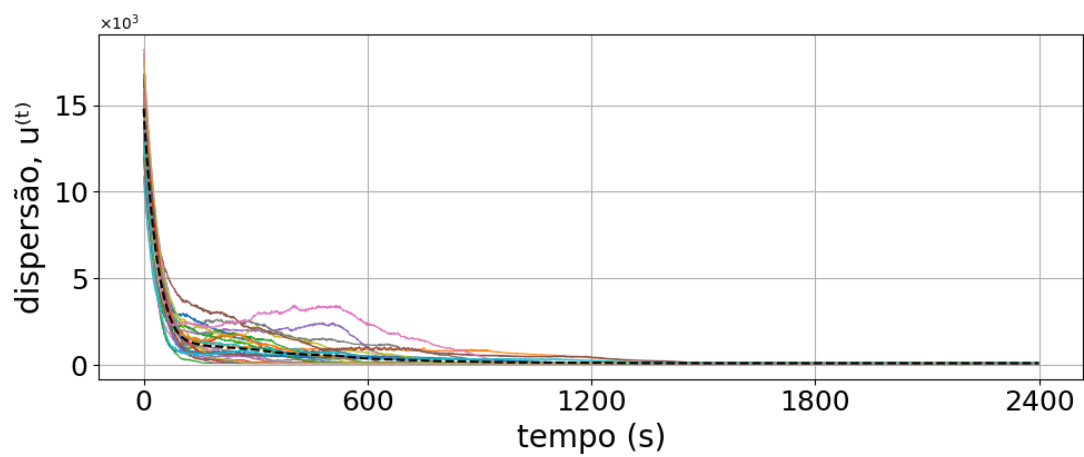
A partir disso, foram selecionados os três resultados mais semelhantes a $\bar{v}_0$ (com similaridades superiores a 90%), porém, para manter o texto conciso, apenas dois deles foram explorados. Como as similaridades de $\bar{v}_B = 93,02\%$ e $\bar{v}_C = 94,23\%$ eram muito próximas, $\bar{v}_C$ não foi explorado. Assim, os vetores $\bar{v}_5 = (-0,56; -1,00; 1,00; -0,96; -0,39; -1,00)$ e $\bar{v}_6 = (-1,00; -0,65; -0,99; 1,00; -1,00; -0,40)$, antigos $\bar{v}_E$ e $\bar{v}_B$ respectivamente, avançaram para as simulações.

5.2.1 Algoritmos de Otimização: Estatísticas e Resultados

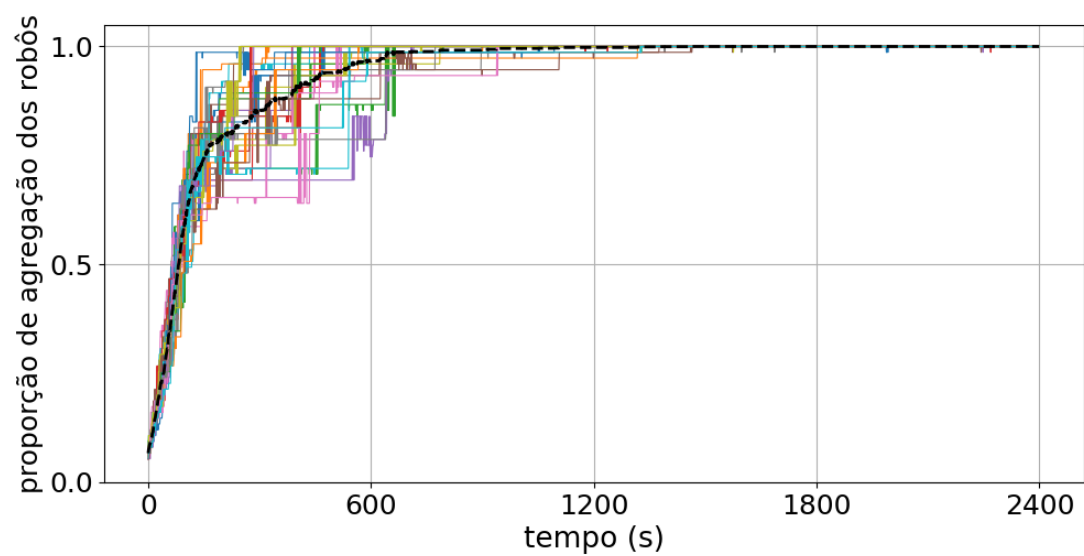
Por serem vetores muito similares, seus resultados apresentaram o mesmo comportamento, com pequenas variações. Assim, o desempenho das simulações padronizadas para $\bar{v}_5$ e $\bar{v}_6$ é visto a seguir. Ambos apresentaram bons valores para função custo, dispersão e proporção, conforme as Figuras 40 e 41.

Figura 40 – Resultados $\bar{v}_5$ 

(a) Função de Custo

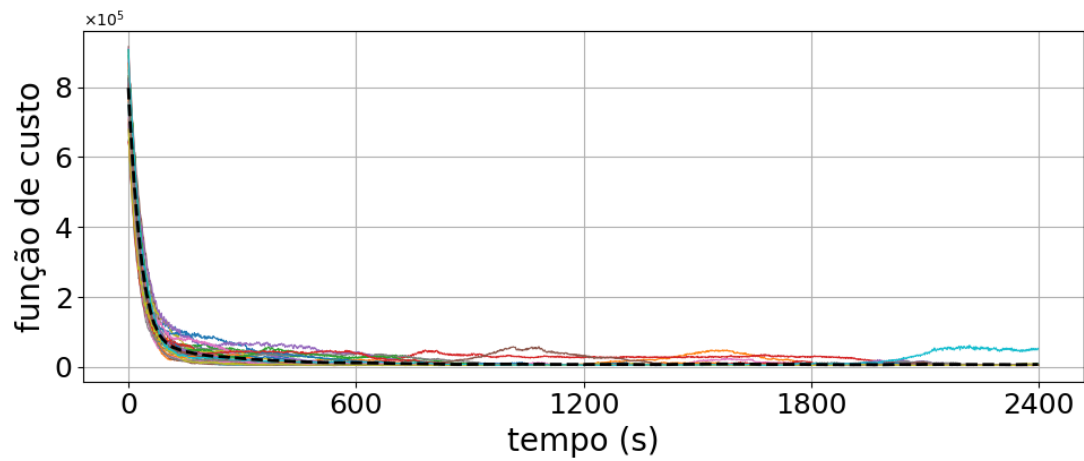


(b) Dispersão

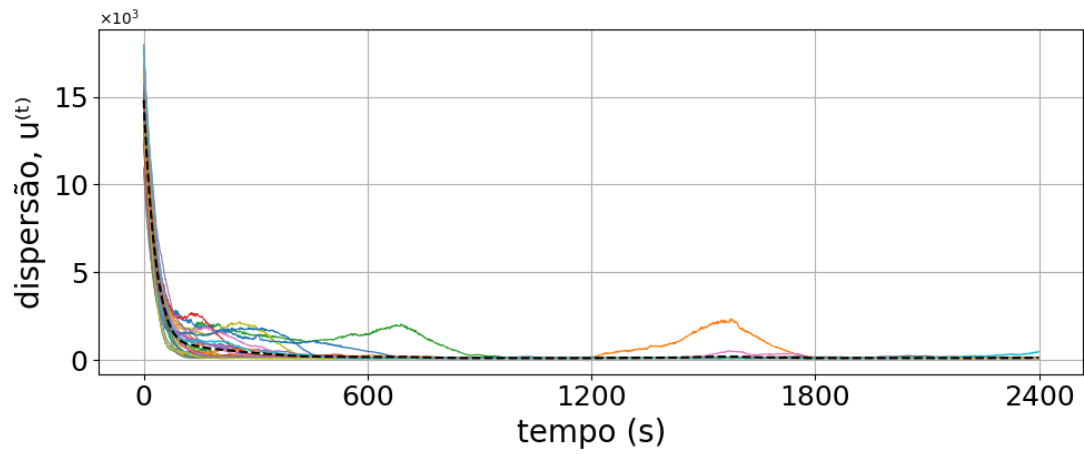


(c) Proporção

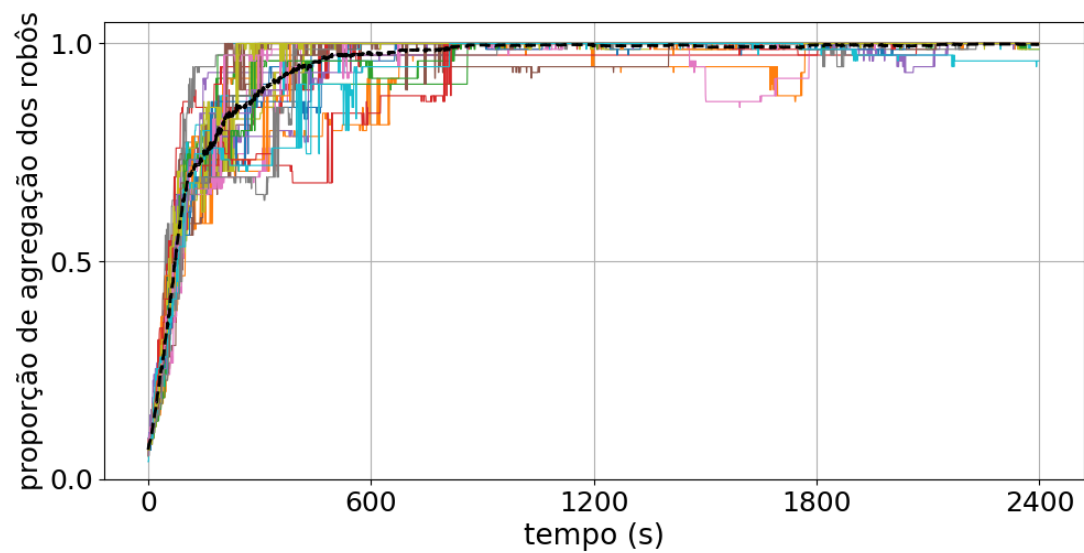
Fonte: A autora (2025).

Figura 41 – Resultados $\bar{v}_6$ 

(a) Função de Custo



(b) Dispersão



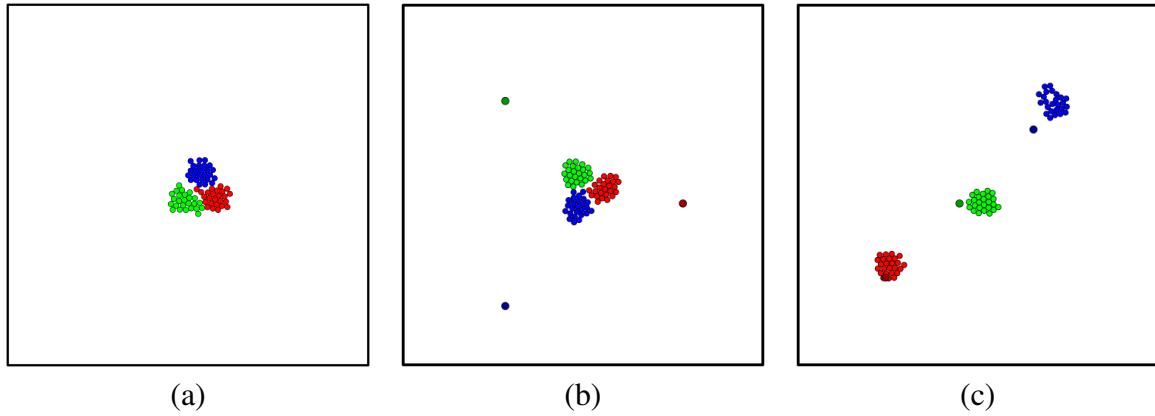
(c) Proporção

Fonte: A autora (2025).

Em relação a formação ao final das simulações, eles apresentam um comportamento curioso: os grupos tendem a ficar no meio do ambiente virtual, mesmo com cilindro de referência no ambiente, indicando uma relação com as velocidades do último par, respectivamente $\bar{v}_{r,2}$ para $\bar{v}_5$ e $\bar{v}_{l,2}$ para $\bar{v}_6$. Como estas velocidades possuem um valor menor que as demais, isso compromete o tempo de reação quando visualiza robôs de outro grupo.

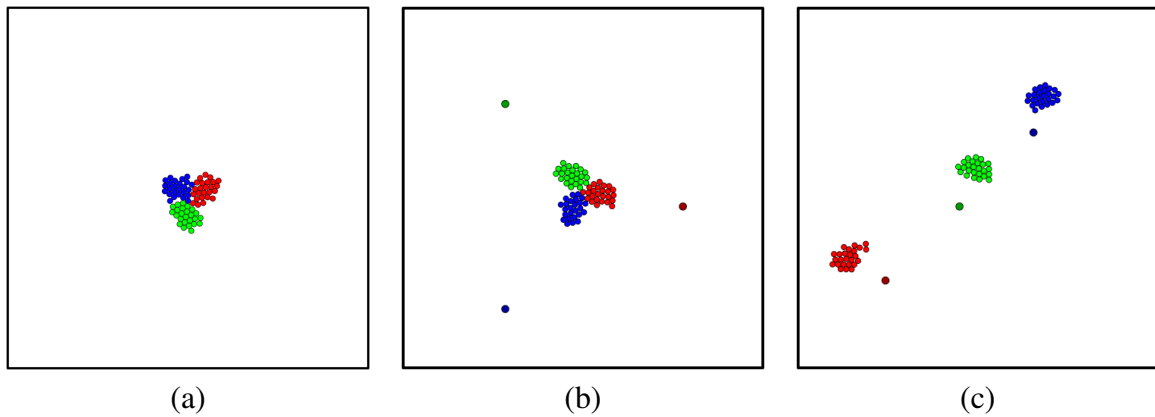
Entretanto, este problema pode ser resolvido mudando a disposição dos cilindros pelo ambiente, para o espalhamento diagonal (Figuras 42 e 43), porém isso gera dependência da solução em uma disposição específica e a algo externo ao enxame. Essas soluções ainda podem ser úteis em algumas aplicações. Assim, ambos os vetores demonstraram ser soluções válidas para o problema de agregação de múltiplos robôs.

Figura 42 – Comparação comportamento de $\bar{v}_5$. (a) sem cilindros de referência (b) com cilindros de referência em modo circular (c) com cilindros de referência em modo diagonal.



Fonte: A autora (2025).

Figura 43 – Comparação comportamento de $\bar{v}_6$. (a) sem cilindros de referência (b) com cilindros de referência em modo circular (c) com cilindros de referência em modo diagonal.



Fonte: A autora (2025).

Quanto à escalabilidade, os desempenhos dos vetores se mantiveram com bons resultados para dispersão, sem dificuldade em tratar enxames maiores. Contudo, para proporção de agrupa-

mentos, mesmo com a presença de cilindros de referência ainda houve ruído para as maiores quantidades de robôs totais no ambiente virtual.

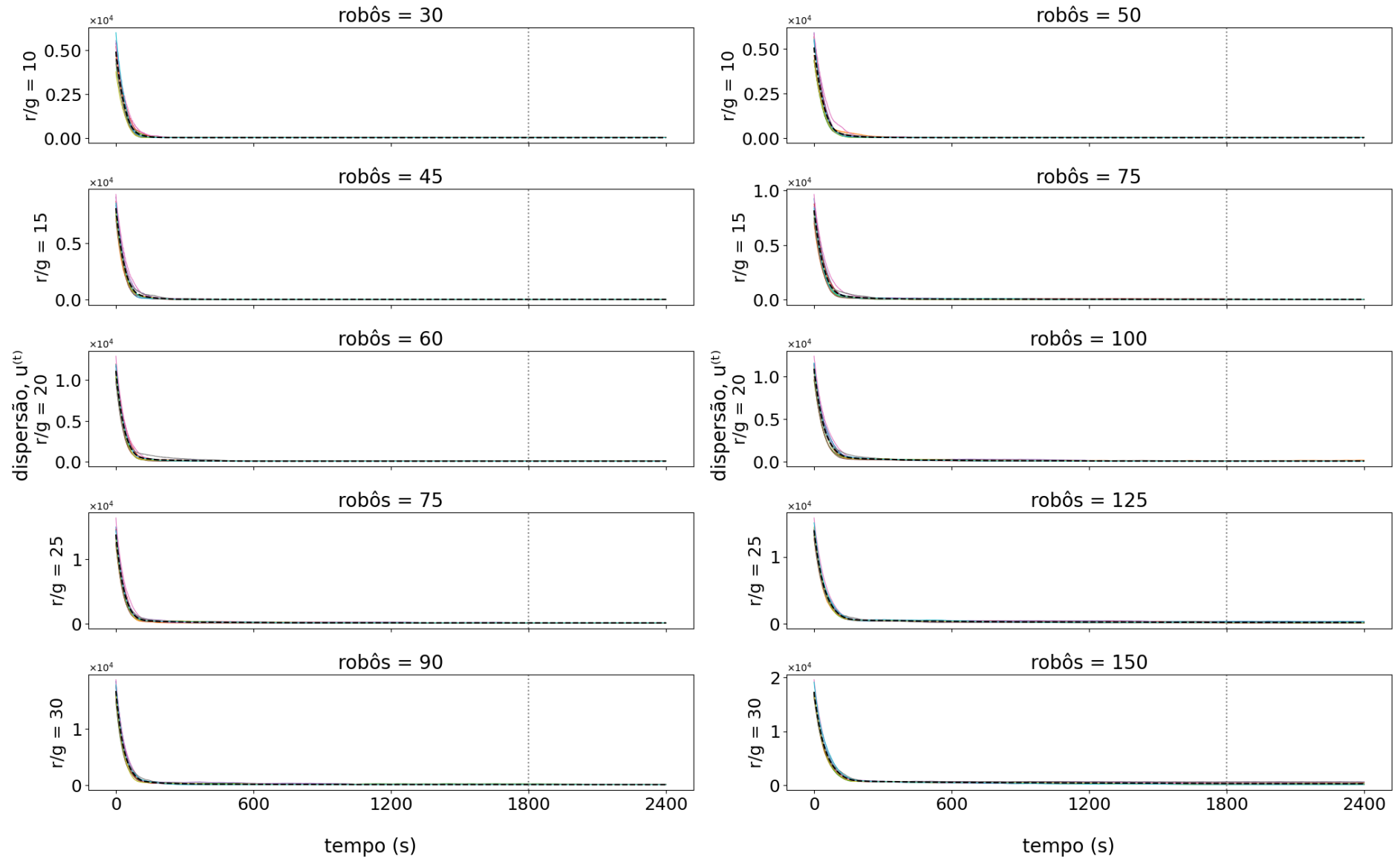
Vale destacar, que como em todos os outros experimentos de escalabilidade, a disposição dos cilindros é circular. Possivelmente, a escalabilidade para a disposição diagonal apresentaria menos ruído. Essas observações podem ser verificadas nas Figuras 44, 45, 46, 47, 48, 49, 50, e 51.

Esta abordagem também apresentou resultados superiores ao encontrado por Krischanski et al. (2023), mas inferiores a abordagem *grid search*. Porém, neste caso, o espaço de busca era muito mais extenso e não houve paralelização. Além disso, aqui a dependência de interferência humana é menor.

Assim, comparando todos os controladores explorados, a solução ótima encontrada foi $\bar{\mathbf{v}}_0$, selecionado por *grid search*. Em relação aos experimentos de otimização automática demonstrou-se que a abordagem híbrida entre os algoritmos MLSL e SBPLX é capaz de identificar controladores consistentes, que convergem para estruturas semelhantes a $\bar{\mathbf{v}}_0$. Porém há um custo computacional significativamente menor em comparação à busca exaustiva, mesmo explorando o espaço de busca completo, veja por exemplo o tempo de execução das duas abordagens, descritos nas Seções 4.2 e 4.3.

Embora as soluções encontradas, como $\bar{\mathbf{v}}_5$ e $\bar{\mathbf{v}}_6$, tenham apresentado uma dependência da disposição dos cilindros de referência para garantir a segregação e espalhamento dos grupos pela arena, seus resultados de escalabilidade validam a eficácia do *design* automatizado para encontrar boas soluções. Portanto, a otimização heurística demonstra ser uma ferramenta útil para identificar controladores minimalistas, oferecendo um equilíbrio viável entre autonomia no processo de *design* e desempenho coletivo em cenários de agregação multitarefa, com menor custo computacional.

Figura 44 – Dispersão $\bar{v}_5$ Sem Cilindro de Referência

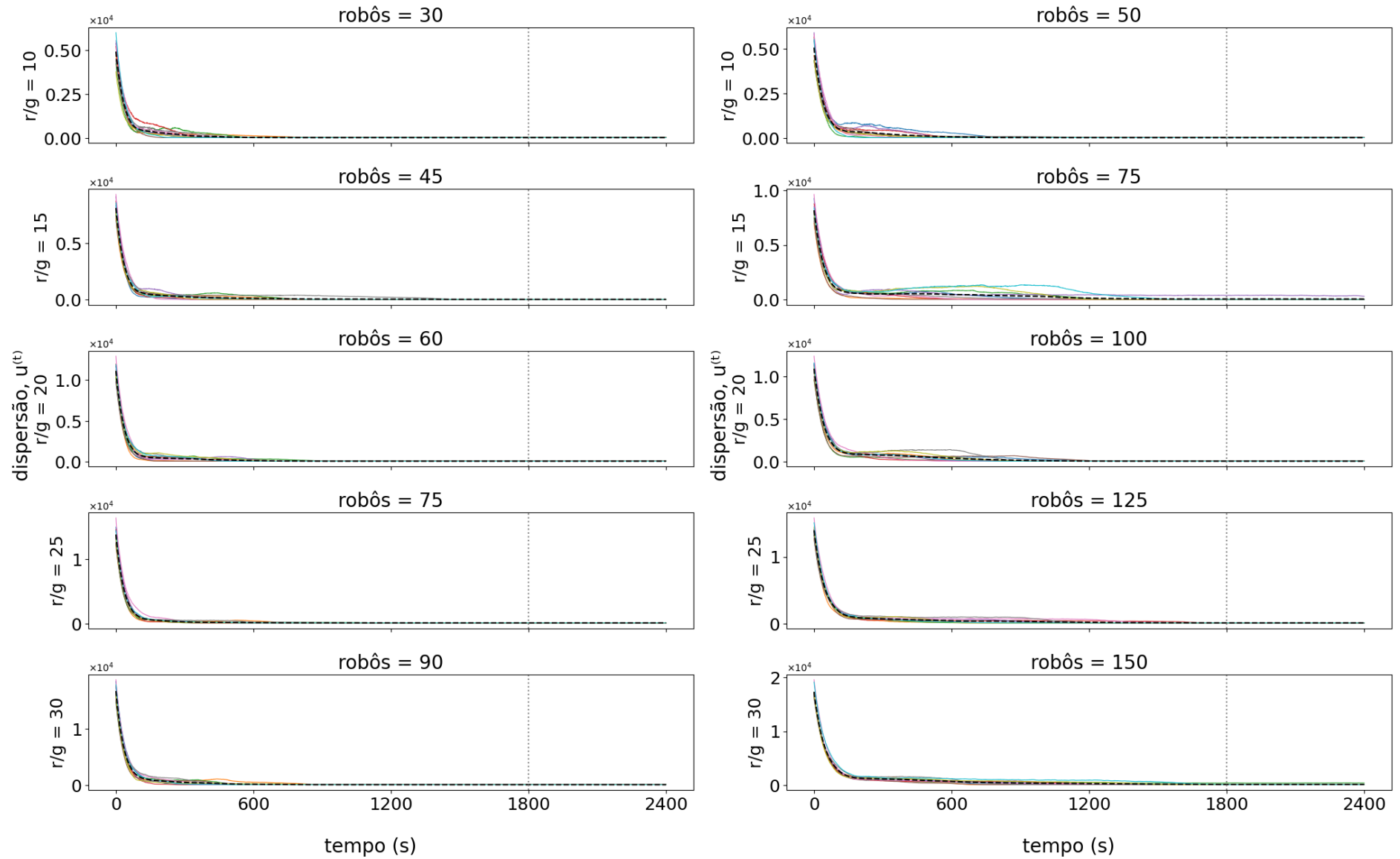


(a) Sem Cilindro de Referência - 3 grupos

(b) Sem Cilindro de Referência - 5 grupos

Fonte: A autora (2025).

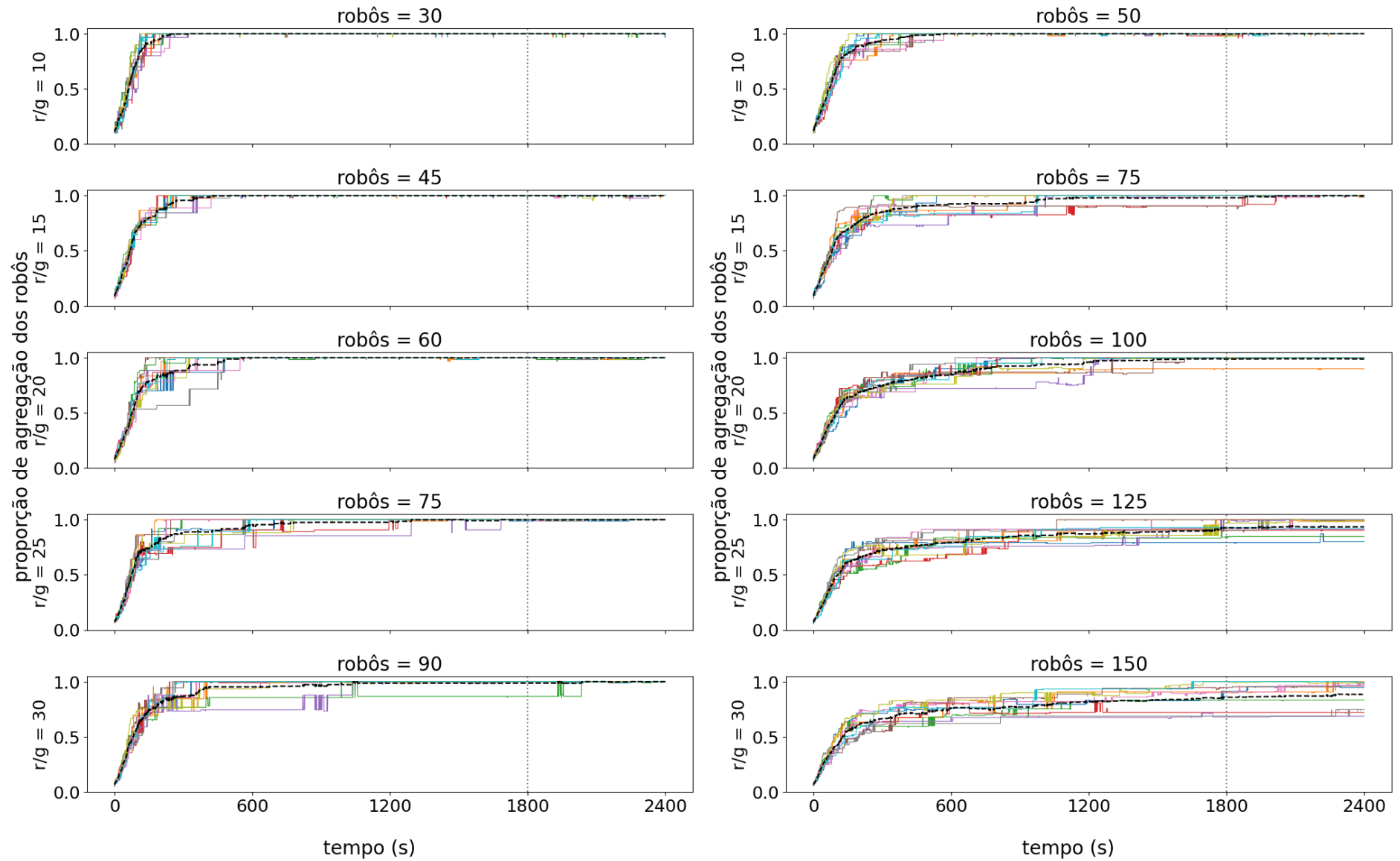
Figura 45 – Dispersão $\bar{v}_5$ Com Cilindro de Referência



(a) Com Cilindro de Referência - 3 grupos

(b) Com Cilindro de Referência - 5 grupos

Figura 46 – Proporção $\bar{v}_5$ Sem Cilindro de Referência

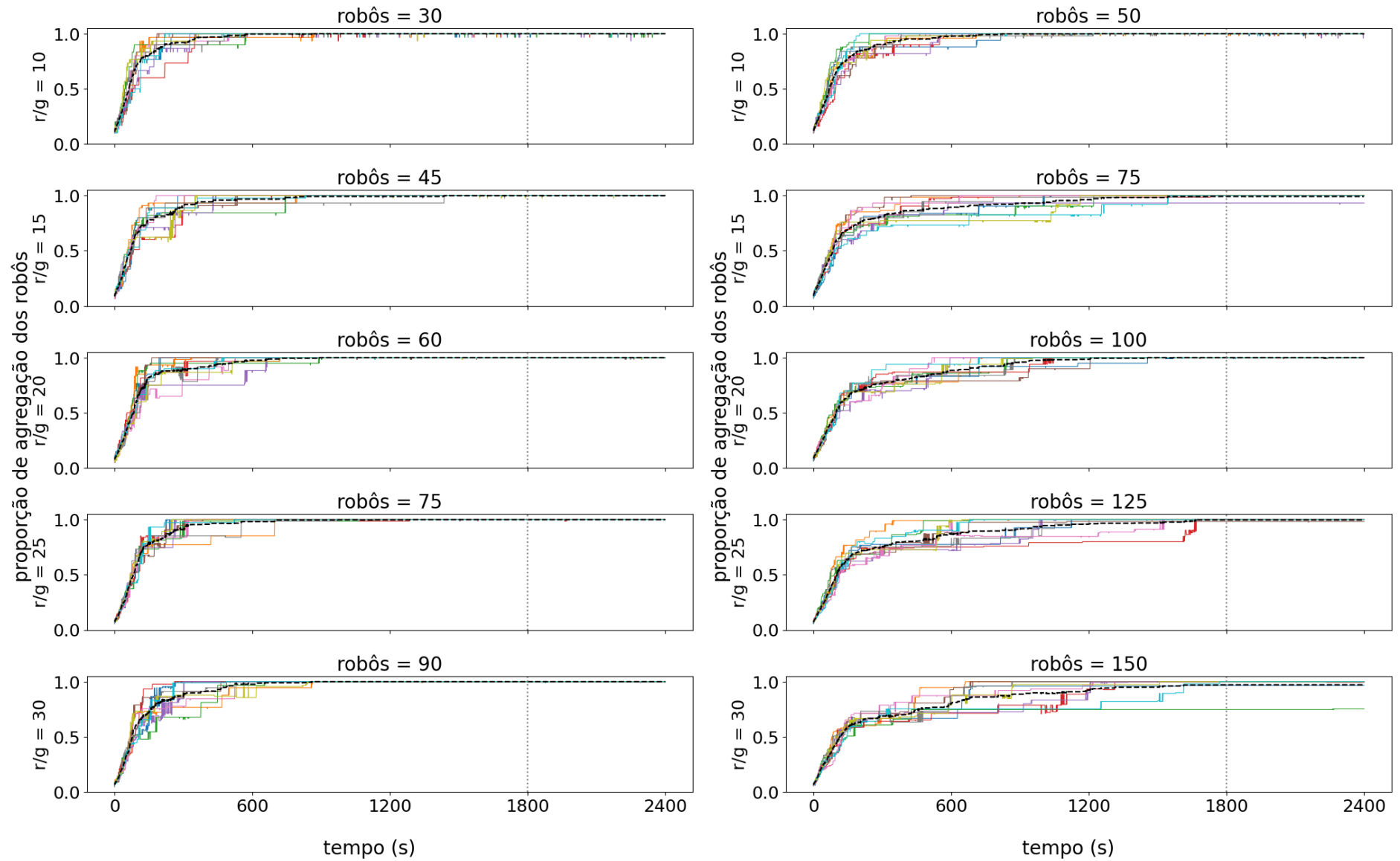


(a) Sem Cilindro de Referência - 3 grupos

(b) Sem Cilindro de Referência - 5 grupos

Fonte: A autora (2025).

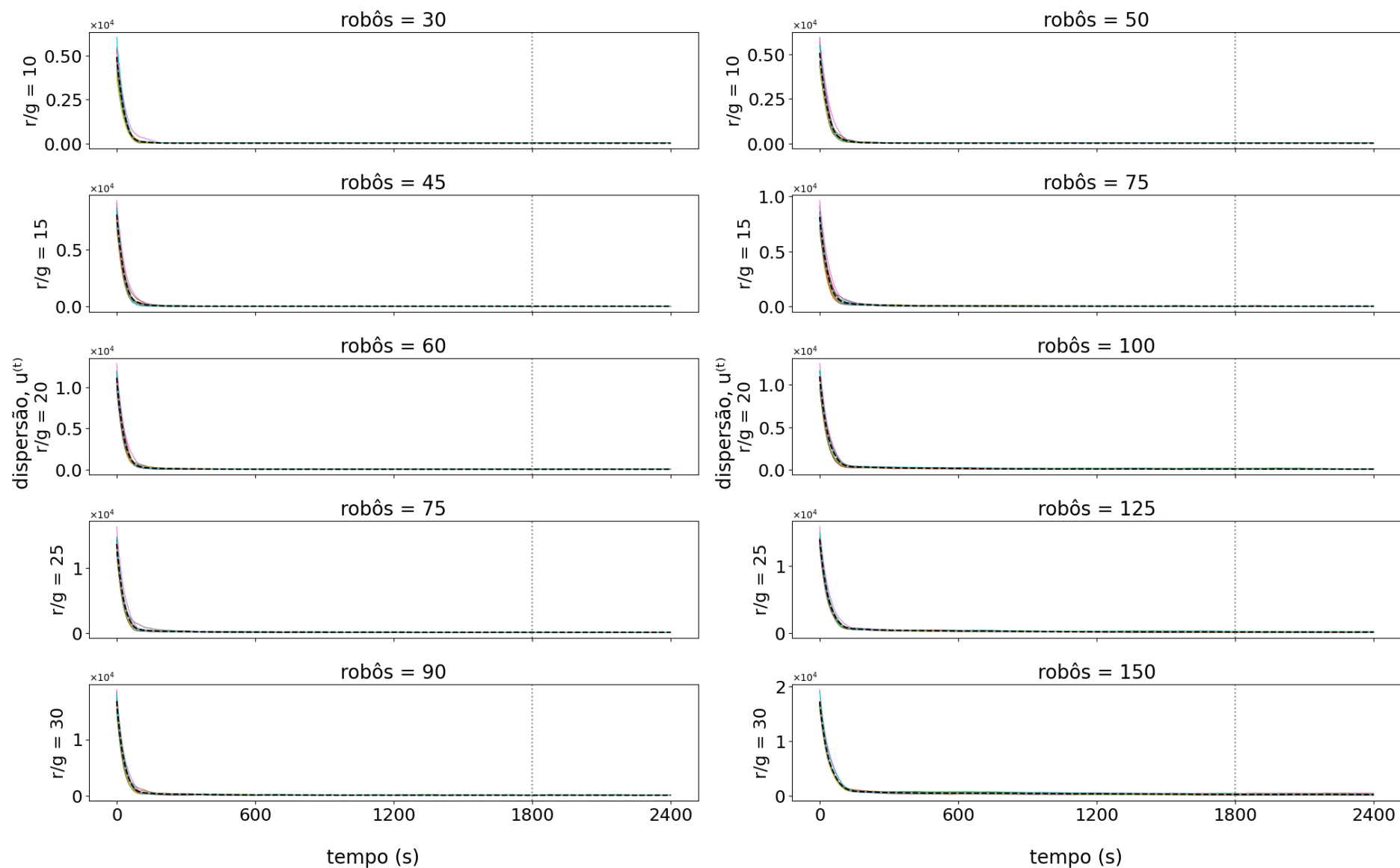
Figura 47 – Proporção $\bar{v}_5$ Com Cilindro de Referência



(a) Com Cilindro de Referência - 3 grupos

(b) Com Cilindro de Referência - 5 grupos

Figura 48 – Dispersão $\bar{v}_6$ Sem Cilindro de Referência

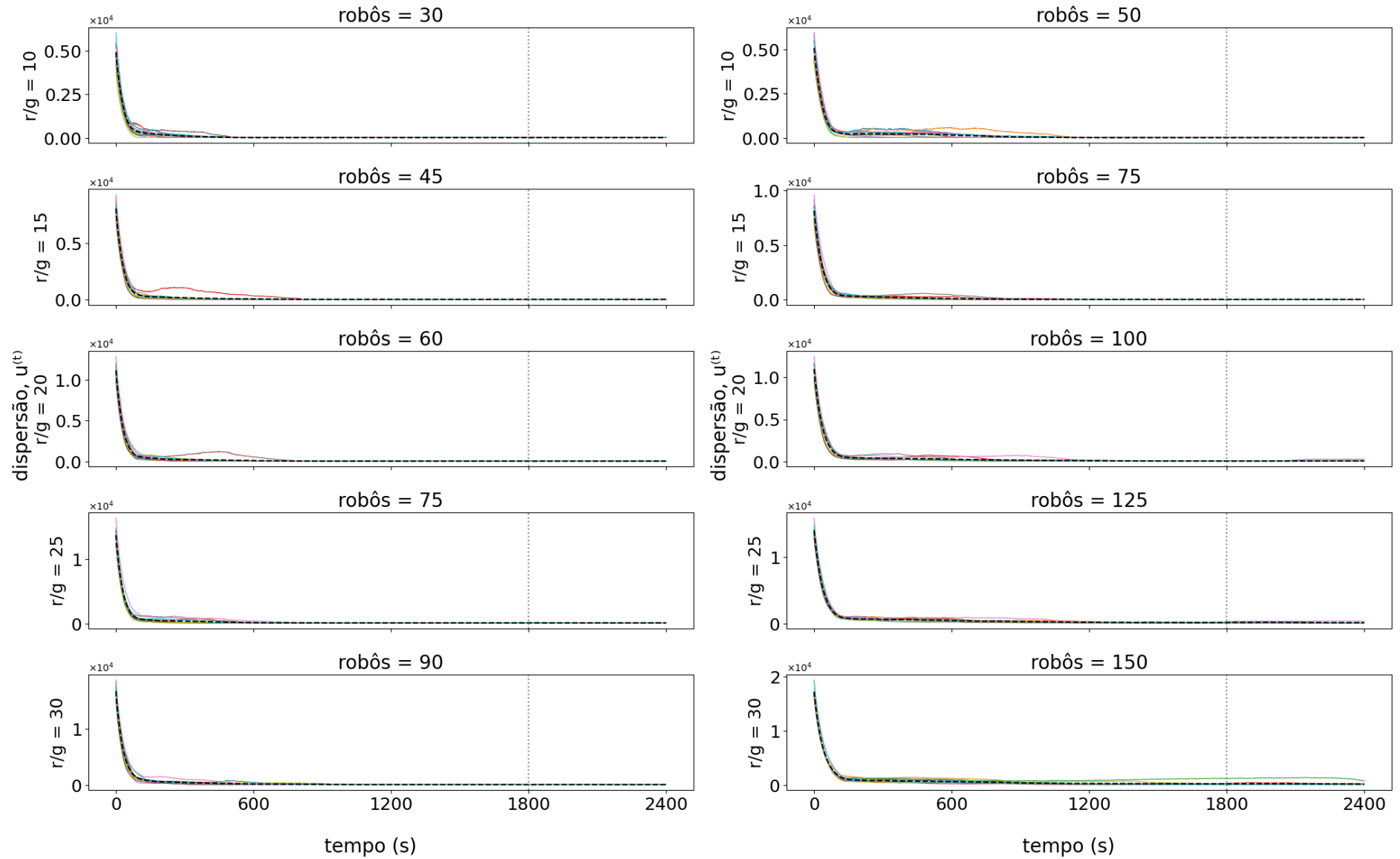


(a) Sem Cilindro de Referência - 3 grupos

(b) Sem Cilindro de Referência - 5 grupos

Fonte: A autora (2025).

Figura 49 – Dispersão $\bar{v}_6$ Com Cilindro de Referência



(a) Com Cilindro de Referência - 3 grupos

(b) Com Cilindro de Referência - 5 grupos

Figura 50 – Proporção $\bar{v}_6$. Sem Cilindro de Referência

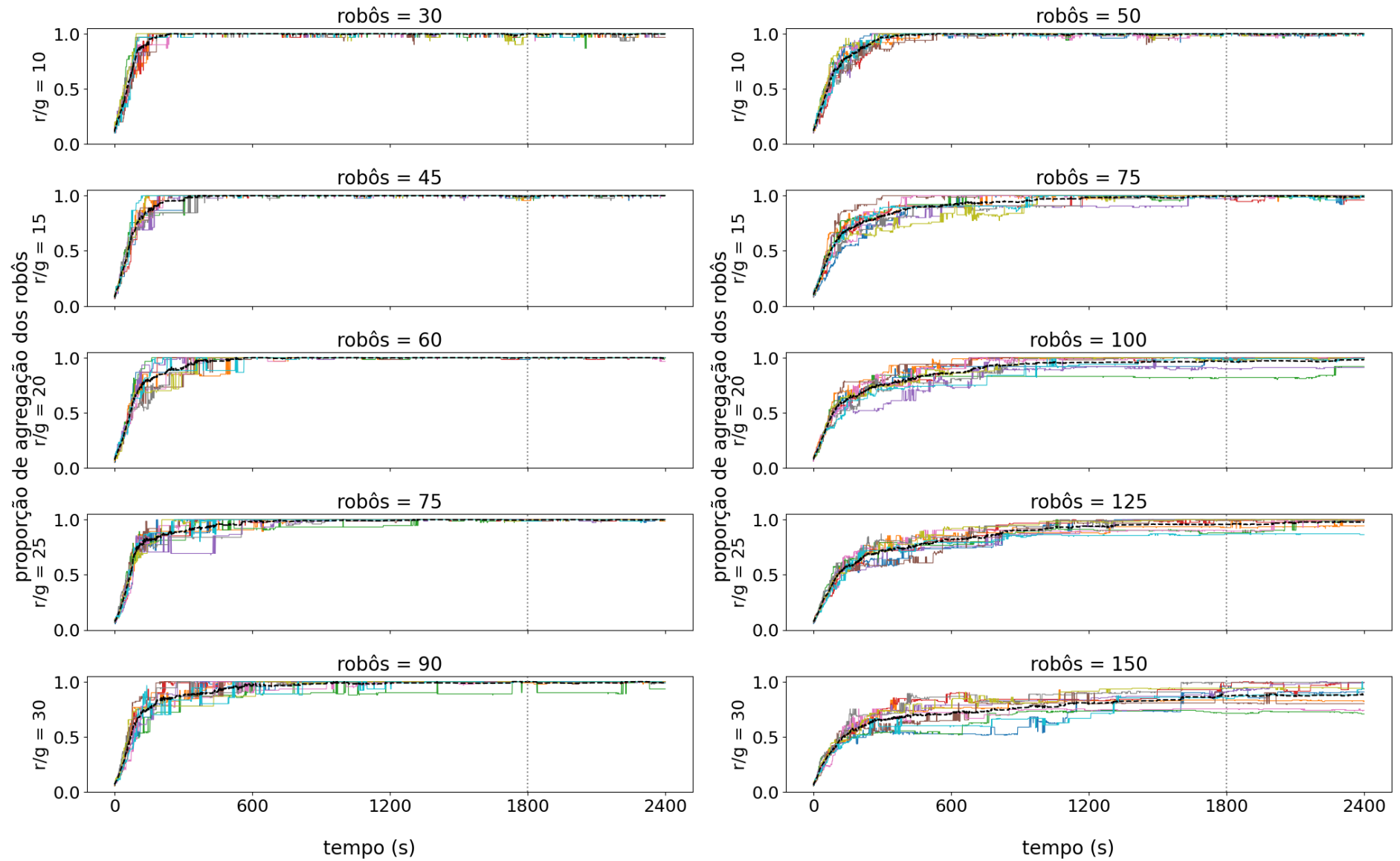
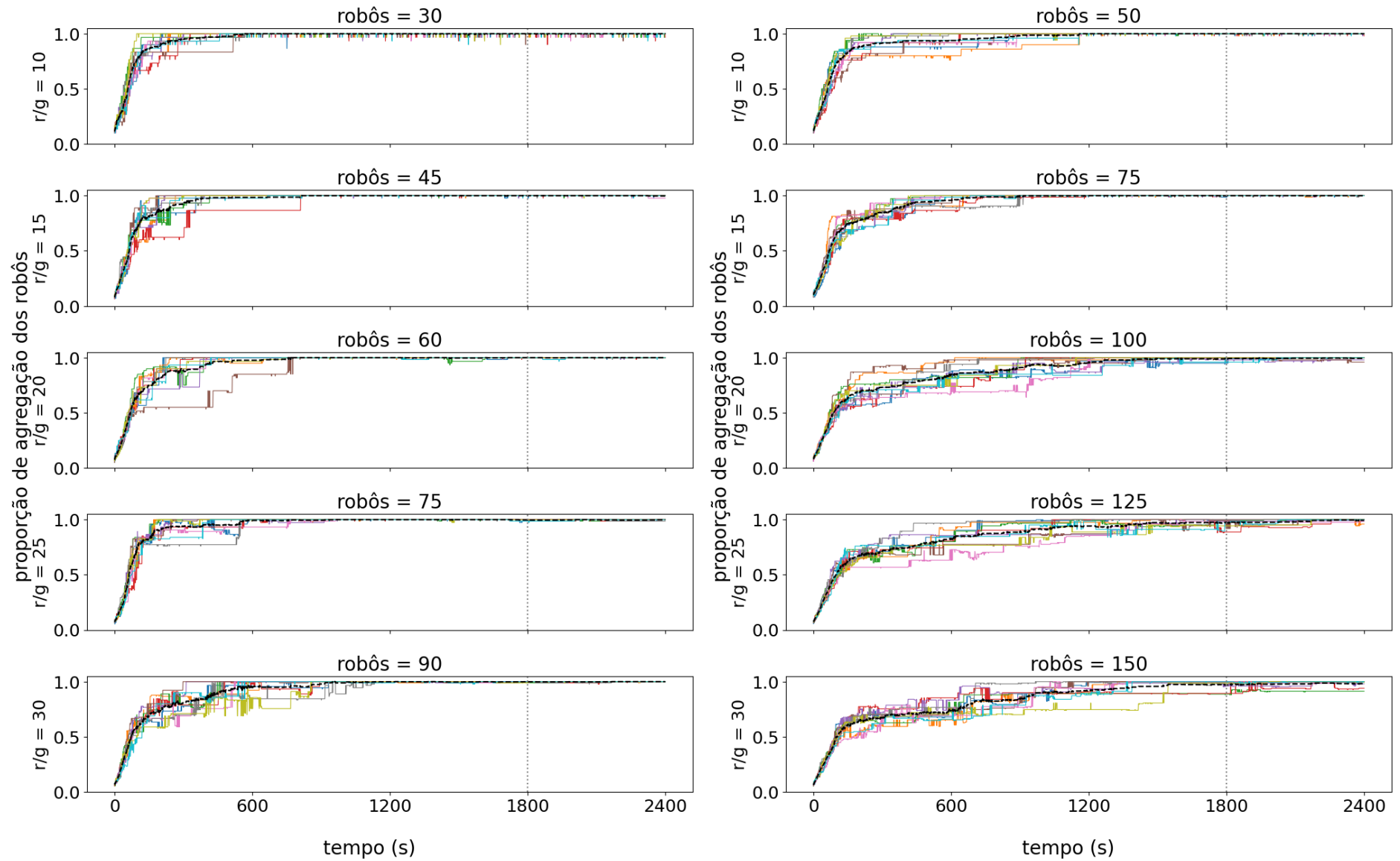


Figura 51 – Proporção $\bar{v}_6$. Com Cilindro de Referência



(a) Com Cilindro de Referência - 3 grupos

(b) Com Cilindro de Referência - 5 grupos

6 ALGORITMO DE ALTERNÂNCIA

A partir dos resultados encontrados e, comparando-os com a literatura, foi proposta uma nova abordagem, a fim de otimizar os resultados, encontrando um controlador que segrega os grupos corretamente, capaz de manter os grupos compactos e chega ao estágio estacionário rapidamente.

6.1 MÉTODO

Tendo $\bar{v}_0$ como a melhor solução encontrada neste trabalho, ele foi comparado com os trabalhos relacionados, mostrando-se superior para agregar múltiplos grupos de robôs simultaneamente em termos da qualidade da solução final, isto é, o quão compacto é cada grupo. No entanto, em relação ao tempo de convergência até atingir o estado estacionário, os resultados encontrado por Gauci et al. (2014a) e Krischanski et al. (2023) são melhores, conforme exposto na Tabela 2. Vale destacar que essa afirmação é feita com base na replicação das simulações dos autores, com as mesmas configurações utilizadas por cada um.

Tabela 2 – Primeira Comparação das Configurações Analisadas. (1) Gauci et al. (2014a), (2) Krischanski et al. (2023) e (3) Configuração $\bar{v}_0$.

Tb.	T (s)	Form.	Met.	Tarefa	D.L.A. (cm)	g	r/g	D (s)	N.P.	C.R.	E.F.
1	200	Compacta	G.S.	Única	316 a 3162	1	100 a 1000	1800	4	Não	Sim
2	145	Circular	DIRECT	Multi.	450	3/5	10 a 30	2400	6	Sim/Não	Não
3	540	Compacta	G.S.	Multi.	450	3/5	10 a 30	2400	6	Sim/Não	Não

Fonte: A autora (2025).

As abreviações utilizadas na Tabela 2 correspondem às seguintes definições: Tb. é trabalho; T representa o tempo até atingir o estado estacionário, em segundos, necessário para que cada proposta atinja o estado estacionário; Met. é o método; Form. indica a formação apresentada ao final das simulações; D.L.A. refere-se à dimensão do lado da arena virtual quadrada dada em centímetros e, para Gauci et al. (2014a), a variação deve-se ao fato dos autores utilizarem uma área sempre proporcional ao número de robôs; g é o número de grupos; r/g corresponde ao número de robôs por grupo; D indica o tempo total de duração das simulações, em segundos; N.P. é número de parâmetros; C.R. informa a presença ou ausência dos cilindros de referência; E.F. indica se houve experimentação física; G.S. é *grid search* e Multi. representa multitarefa.

A fim de superar essa limitação e conciliar as vantagens do controlador encontrado por (Krischanski et al., 2023), $\bar{v}_{\alpha_2}$, é proposta uma nova solução. Este novo conceito foi nomeado de Algoritmo de Alternância e refere-se à capacidade do robô de trocar autonomamente entre dois comportamentos distintos (Exploração ($\bar{v}_{\alpha_2}$) e Agregação Compacta ($\bar{v}_0$)), tornando o enxame capaz de segregar rapidamente os grupos e, após isso, iniciar a agregação.

Para o algoritmo de alternância, os robôs passam a utilizar comunicação mínima local (utilizando sensores de infravermelho(IR)), memória temporal (em um *buffer*) e uma única informação global (parâmetro conhecido no início da tarefa), o número total de robôs por grupo. Essas implementações foram necessárias para que os robôs:

- Soubessem quão próximo está outro robô de si para trocar informações de seu *buffer*;
- Pudessem otimizar suas buscas, considerando as informações vindas do *buffer* de seu companheiro;
- Utilizassem o *broadcast* para descartar informações obsoletas; e
- Comparassem o tamanho do seu *buffer* com o total de companheiros e identificassem quando alterar seu comportamento;

A partir disso, se dá a nova lógica de funcionamento do enxame. Primeiramente, cada robô é identificado com um código ID, de forma determinística durante a inicialização da simulação, formado por um endereçamento composto que codifica a estrutura social do enxame diretamente no identificador numérico de 32 bits, anteriormente cada robô era tratado de forma exclusivamente reativa a partir da leitura de sua cor pelo sensor de visão. Por meio de operações de manipulação binária, o identificador único de cada robô é sintetizado pela concatenação do índice do seu grupo (16 bits mais significativos) acompanhado por seu índice sequencial local (16 bits menos significativos), o que permite a identificação do grupo a qual a ID pertence por qualquer vizinho sem recorrer a tabelas centralizadas.

Este tipo de arquitetura de identificação é muito importante para o sistema distribuído, pois permite a filtrar pacotes de comunicação e o gerenciamento da memória (*buffer*). Isso interfere diretamente na tomada de decisão deste novo sistema, pois o robô só se comunica com seus companheiros do mesmo grupo.

Assim, quando a simulação inicia, cada robô mantém sua observação do ambiente baseada no sensor de visão, iniciando sempre com a configuração das velocidades sendo $\bar{v}_{\alpha_2}$. Então, conforme se aproxima dos robôs que lhe atraem, do mesmo grupo, o robô passa utilizar os IR para saber se o robô mais próximo a ele está perto o suficiente para ler seu ID e, possivelmente, trocar informações. Essa distância é dada pelo alcance do sensor somado a dois raios do robô, totalizando 17 cm.

Para garantir que o robô mantenha a coesão com o grupo mesmo quando perde o contato visual momentâneo, o sistema utiliza uma memória de curto prazo (*buffer*). Essa memória funciona como uma lista dinâmica: ela armazena os identificadores dos vizinhos, ID, detectados e associa a cada um deles um temporizador.

A atualização dessa lista de ID é feita através de um processo de comunicação cooperativa (*broadcast*). Quando dois robôs se aproximam, e são do mesmo grupo, eles trocam suas listas inteiras de vizinhos. Essa troca de informações permite que o robô tenha uma noção mais ampla e estável da densidade do grupo ao seu redor, superando as limitações da visão direta

ou a necessidade de mecanismos mais complexos para cálculos de densidade ou excesso de informações locais, o que exigiria uma infraestrutura de processamento global, comprometendo a natureza do controle descentralizado.

Além disso, se um ID é percebido novamente, seu temporizador é reiniciado para um valor máximo, porém, caso chegue a zero sem renovação, é removido do *buffer*. Este mecanismo assegura que o robô não abandone o grupo imediatamente após perder um vizinho de alcance, mas também não permaneça em estado de agregação perpetuamente se o grupo se dispersar. Devido a seu impacto na tomada de decisão, esse temporizador torna-se uma variável de interesse nesta etapa.

Já a alternância é dada por outra variável de interesse, o limiar de densidade, que é responsável pela decisão de troca de comportamento. A partir da divisão do tamanho do *buffer* pelo número de robôs por grupo há a proporção de vizinhos do mesmo grupo presentes na memória do robô, quando este valor é maior ou igual ao limiar de densidade ocorre a mudança de comportamento da exploração para a agregação. Assim pode-se definir:

$$\rho = \frac{N_{buffer}}{r/g} \quad (19)$$

Onde:

- ρ é o limiar de densidade adimensional;
- N_{buffer} representa a quantidade de IDs de robôs do mesmo grupo atualmente armazenadas;
- r/g é o total de robôs que cada grupo possui.

Assim, seguiu-se uma metodologia muito semelhante a do Capítulo 4, especialmente a Seção 4.3, pois o primeiro passo foi estabelecer as variáveis, limiar de densidade e temporizador do *buffer*, e otimizá-las a partir do treinamento do algoritmo, selecionando também os algoritmos global e local MLSL e SBPLX por seus bons resultados anteriores. Dessa forma foram obtidos os valores de: 61% para o limiar de densidade, ρ , e de 61 s para o temporizador, τ .

Com isso, a solução foi chamada de $\bar{\mathbf{v}}_7$ e corresponde a:

$$\bar{\mathbf{v}}_7 = (\bar{\mathbf{v}}_\alpha; \bar{\mathbf{v}}_0; \rho; \tau) = (\bar{\mathbf{v}}_\alpha; \bar{\mathbf{v}}_0; 0,61; 61,00), \quad \rho \in [0, 1], \tau \in \mathbb{R}^+ \quad (20)$$

Onde:

- τ representa o temporizador, expresso em segundos.

A fim de validar essa solução e seguir com os experimentos para compará-la posteriormente com a literatura, foram realizadas dez simulações com configurações diferentes, buscando identificar sua consistência em diferentes cenários. Cada simulação tem por objetivo identificar qual o tempo necessário para que os robôs segregassem e agregassem corretamente, até atingirem o estado estacionário, os resultados estão descritos na Tabela 3.

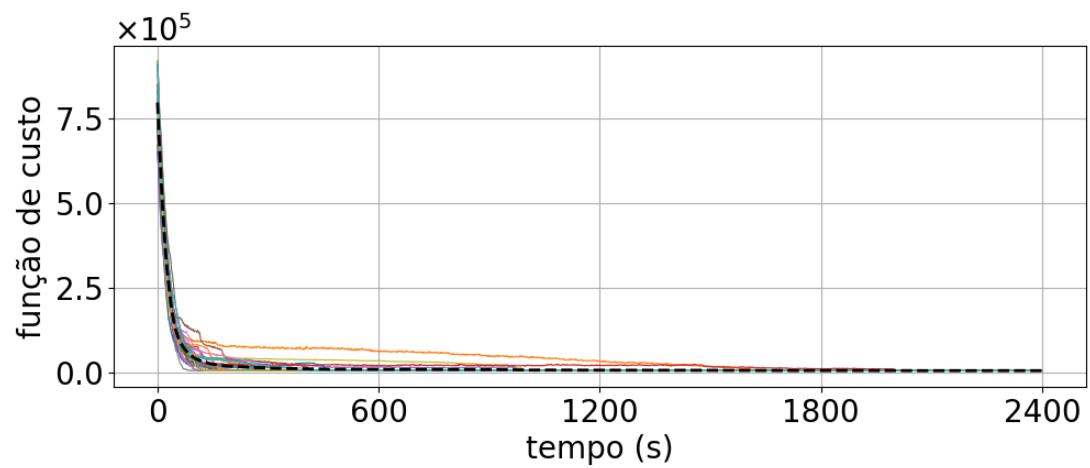
Tabela 3 – Simulações para validação da configuração $\bar{v}_7$.

Simulação	Nº de Grupos	Robôs por Grupo	Cilindro de Referência	Tempo (s)
1	3	10	Não	122
2	3	10	Sim	272
3	3	25	Não	184
4	3	25	Sim	515
5	5	15	Não	340
6	5	15	Sim	620
7	3	30	Não	137
8	3	30	Sim	709
9	5	30	Não	348
10	5	30	Sim	1370

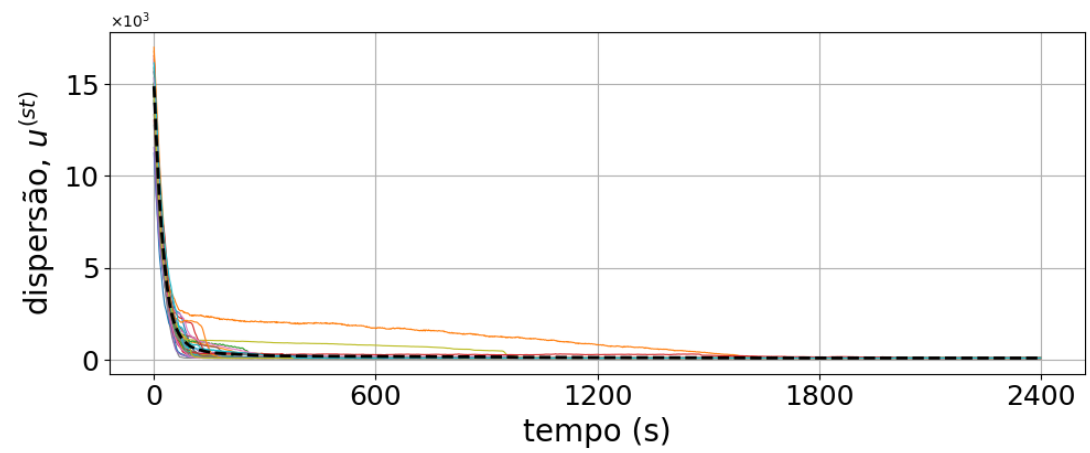
Pode-se notar que a solução encontrada, $\bar{v}_7$, é consistente e mostrou redução do tempo quando comparada a utilização apenas de $\bar{v}_0$, considerando-a válida para exploração das métricas de dispersão e proporção. Entretanto, nota-se um fato curioso, o uso de cilindros de referência aumentou o tempo em cerca de três vezes em média, quando comparado as simulações sem cilindros de referência.

6.2 RESULTADOS

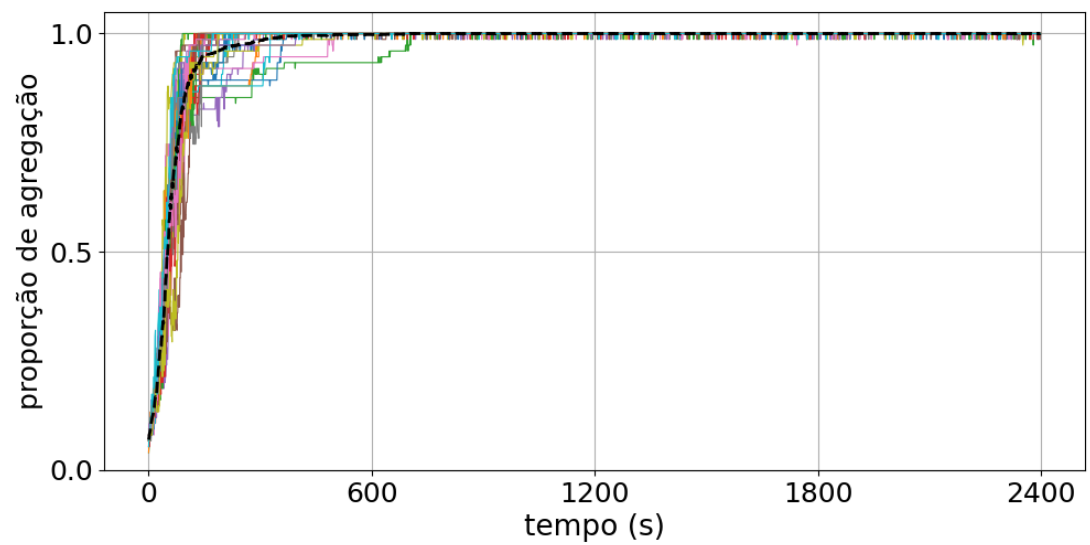
A partir disso, seguiu-se a mesma metodologia descrita no Capítulo 4. Inicialmente, obteve-se resultados para a função de custo, dispersão e proporção, com ou sem a presença dos cilindros. Conforme as Figuras 52 e 53, pode-se observar que essas métricas convergiram rapidamente, impactando em significativa diminuição do tempo para atingir o estado estacionário quando comparado com $\bar{v}_0$.

Figura 52 – Resultados $\bar{v}_7$ sem cilindro de referência

(a) Função de Custo

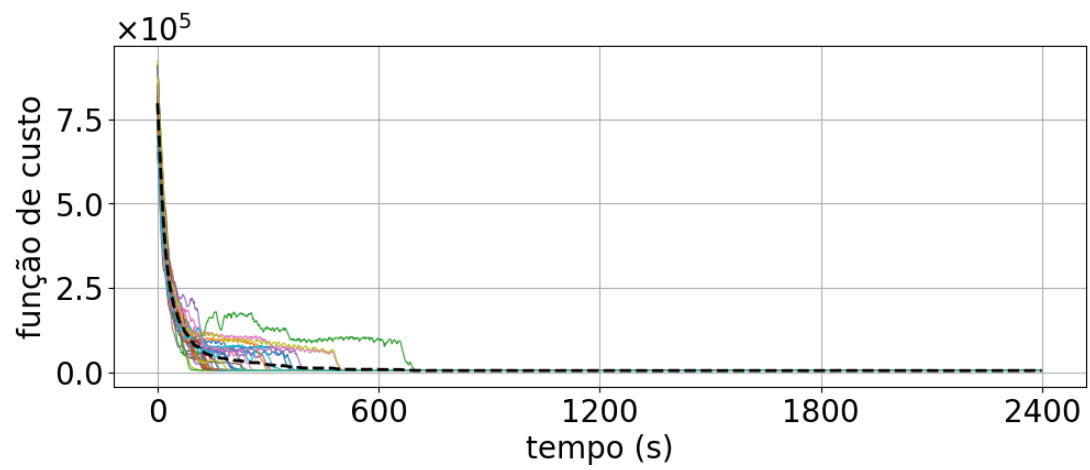


(b) Dispersão

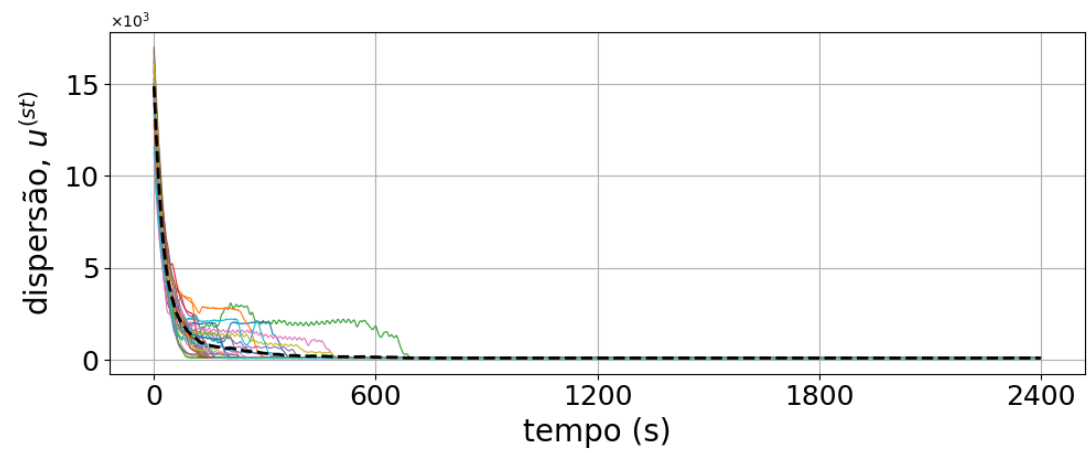


(c) Proporção

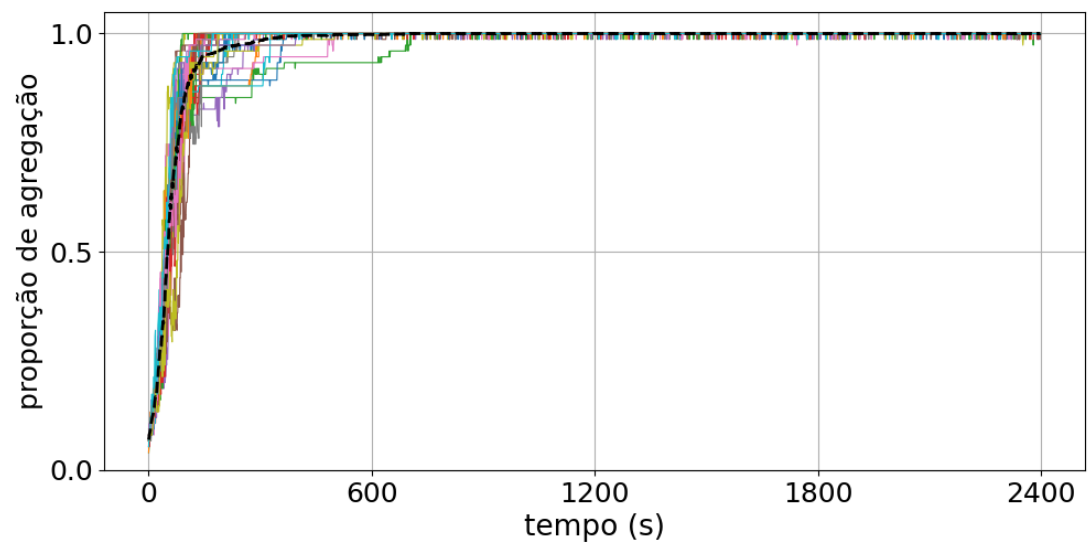
Fonte: A autora (2025).

Figura 53 – Resultados $\bar{v}_7$ com cilindro de referência

(a) Função de Custo



(b) Dispersão

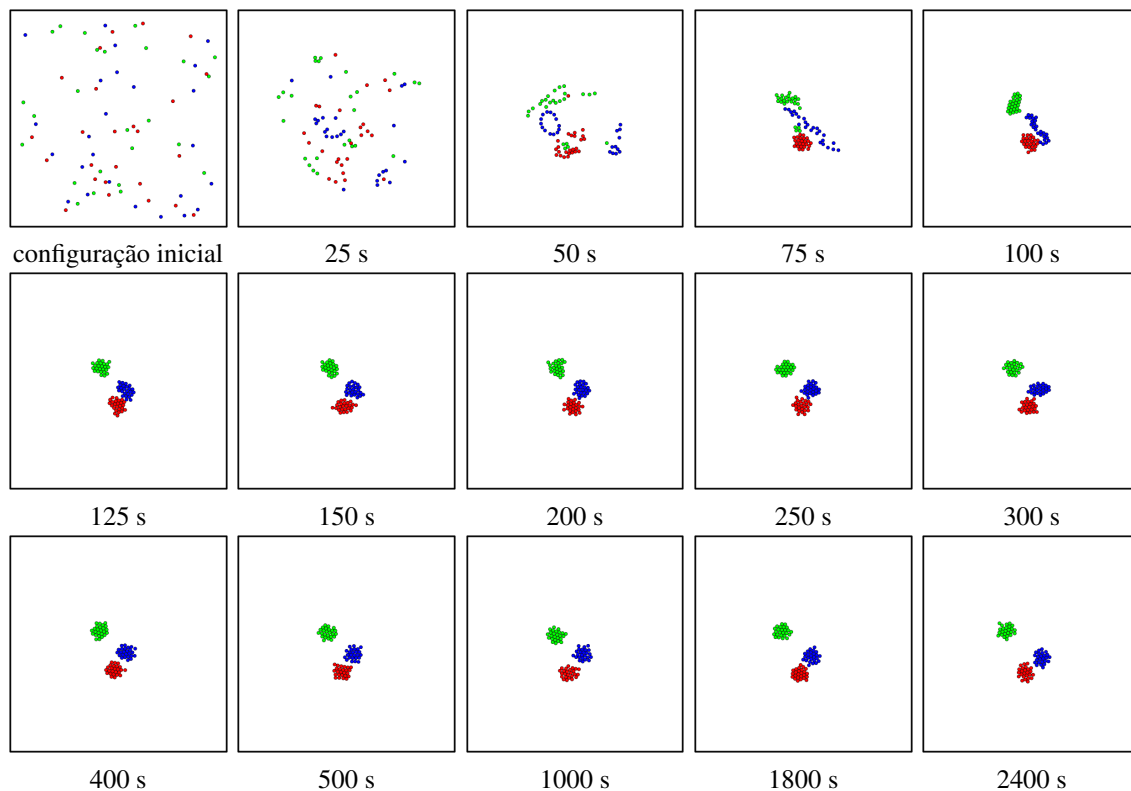


(c) Proporção

Fonte: A autora (2025).

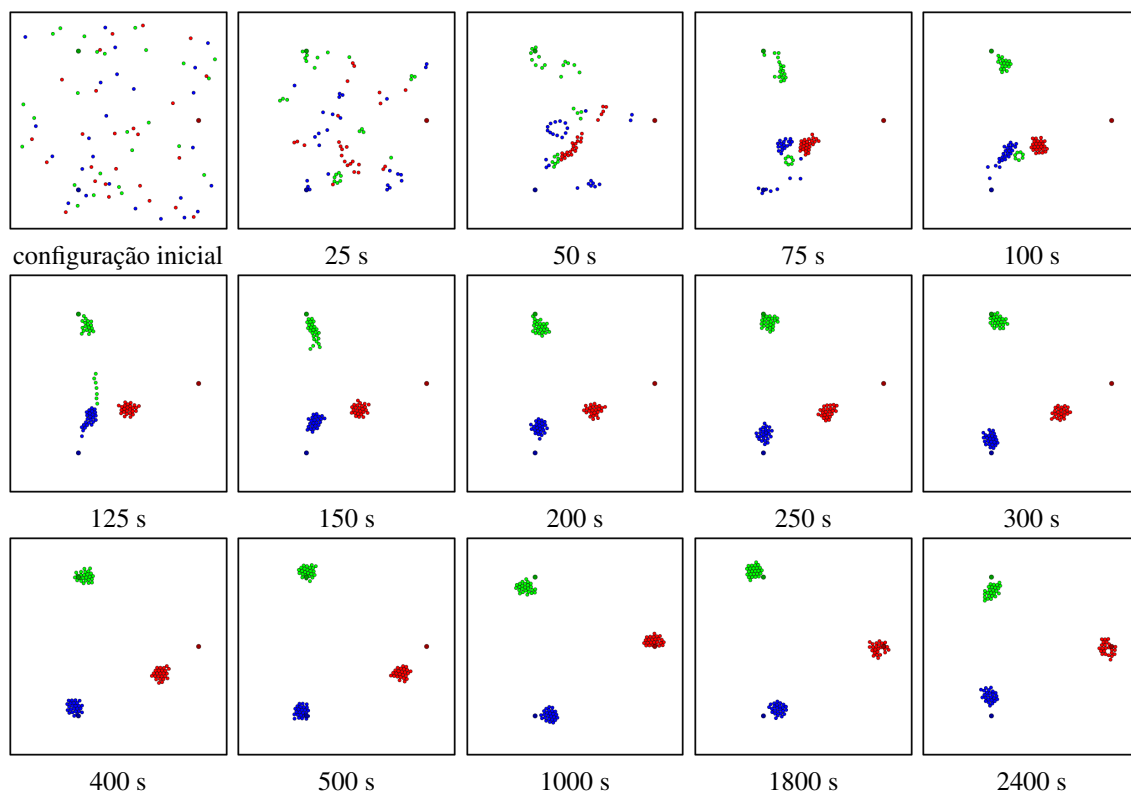
Isso fica mais evidente nas Figuras 54, 55, 56 e 57, as quais demonstram o comportamento do enxame ao longo da simulação. Nelas é possível observar que em todas as configurações o tempo para atingir o estado estacionário ficou abaixo dos 500 segundos, mesmo com aumento do número de robôs e grupos, havendo ou não cilindro de referência no ambiente. Isso demonstra que a solução foi capaz de superar o problema de tempo identificado em $\bar{v}_0$. Outras configurações foram exploradas e podem ser observadas no Apêndice A.

Figura 54 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 75 robôs.



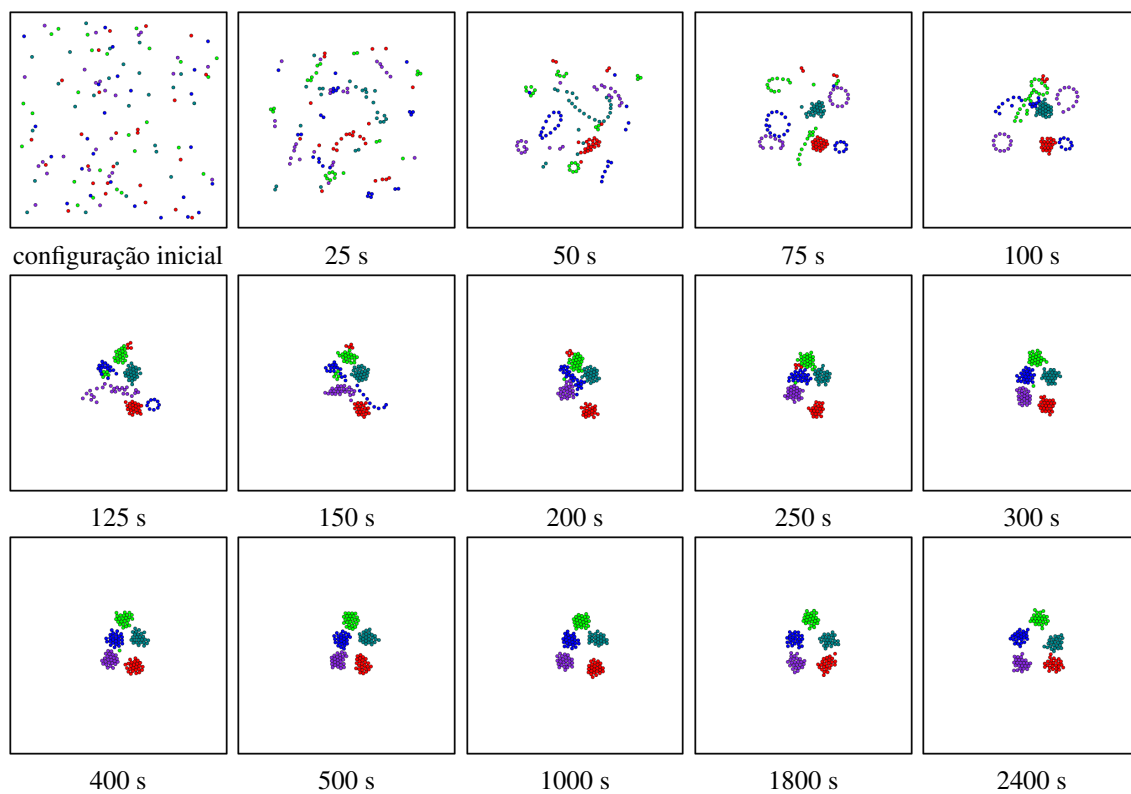
Fonte: A autora (2025).

Figura 55 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 75 robôs.



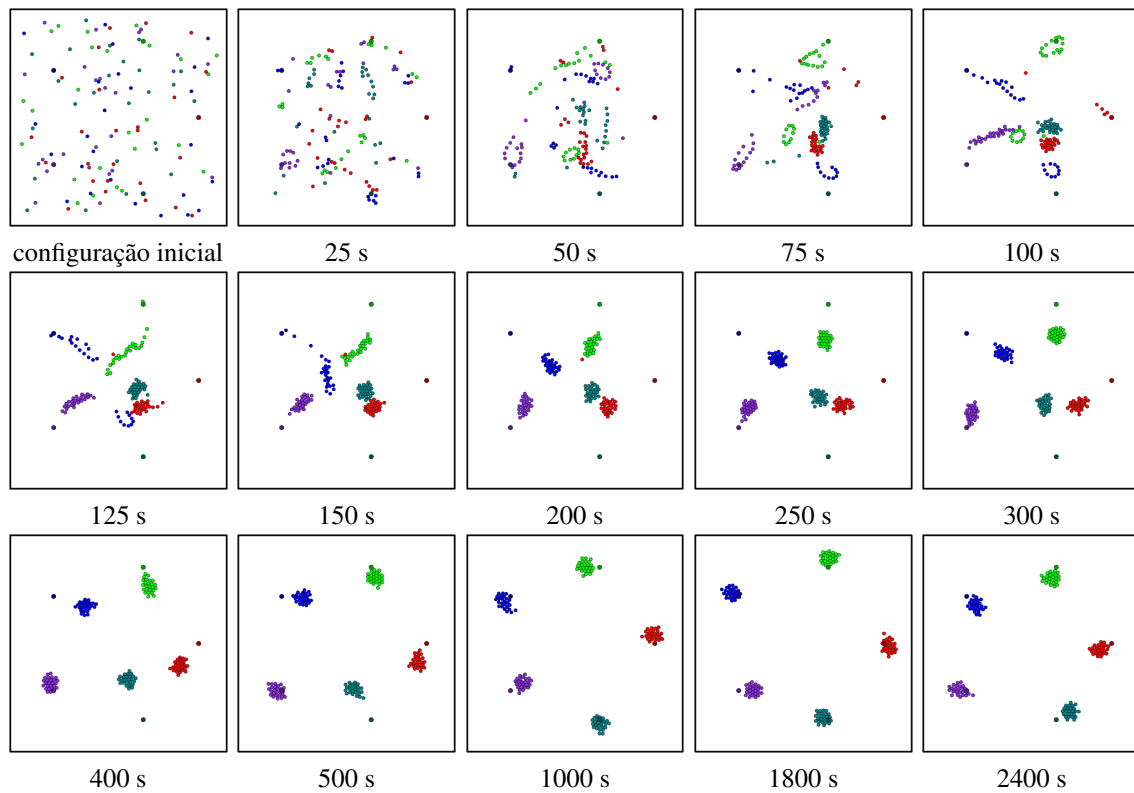
Fonte: A autora (2025).

Figura 56 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 125 robôs.



Fonte: A autora (2025).

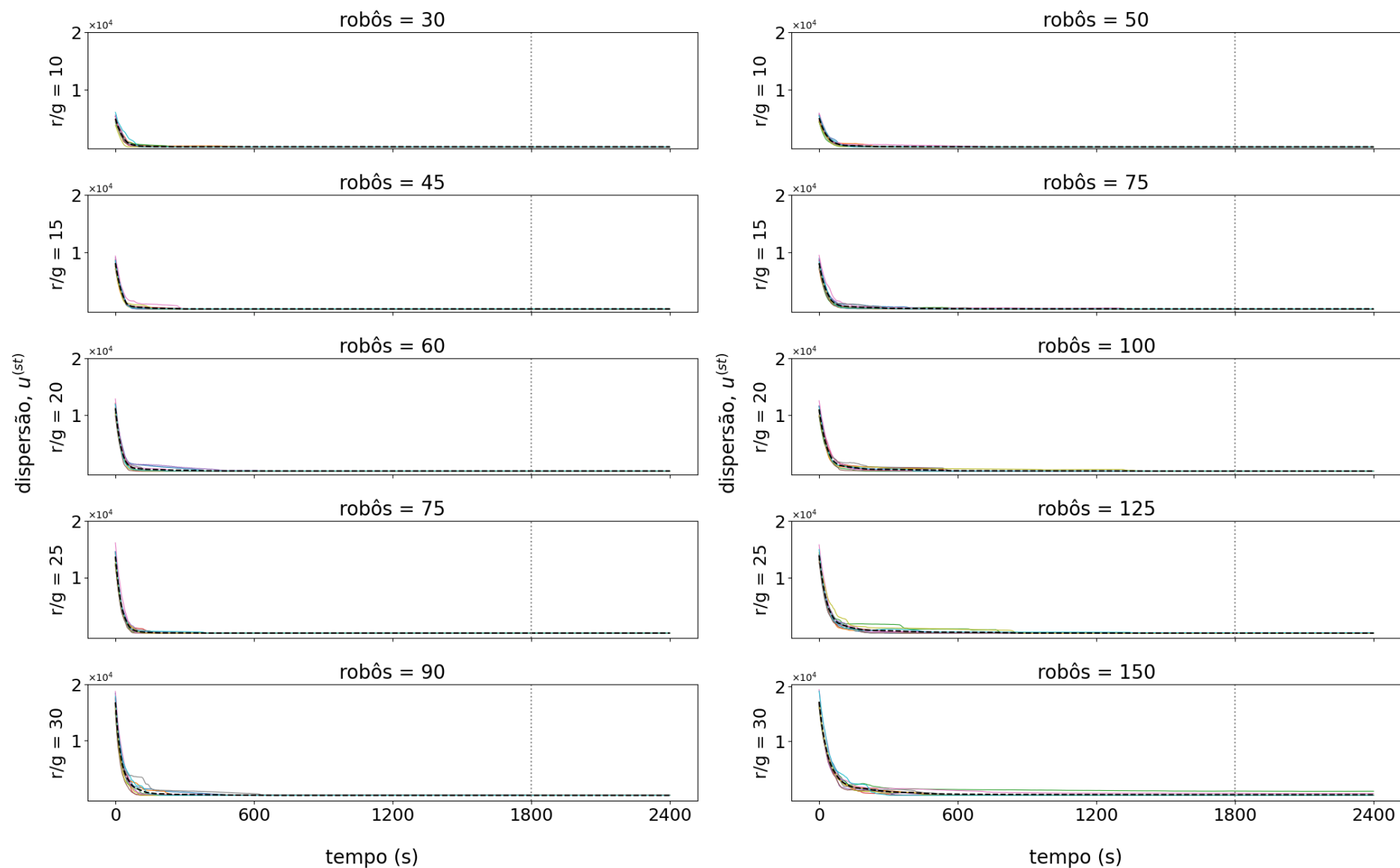
Figura 57 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 125 robôs.



Fonte: A autora (2025).

Em relação as simulações de escalabilidade, os resultados foram consistentes, mostrando que a proposta desenvolvida lida bem com diferentes cenários e possui uma menor dependência dos cilindros de referência para concluir a tarefa e chegar ao estado estacionário, ainda que estes sejam úteis para diminuição do tempo de conclusão da tarefa em cenários com muitos robôs. Além disso, nota-se que as Figuras 58, 59, 60 e 61 apresentaram pouco ruído e tempo de convergência baixo em todos os casos.

Figura 58 – Dispersão $\bar{v}_7$ Sem Cilindro de Referência

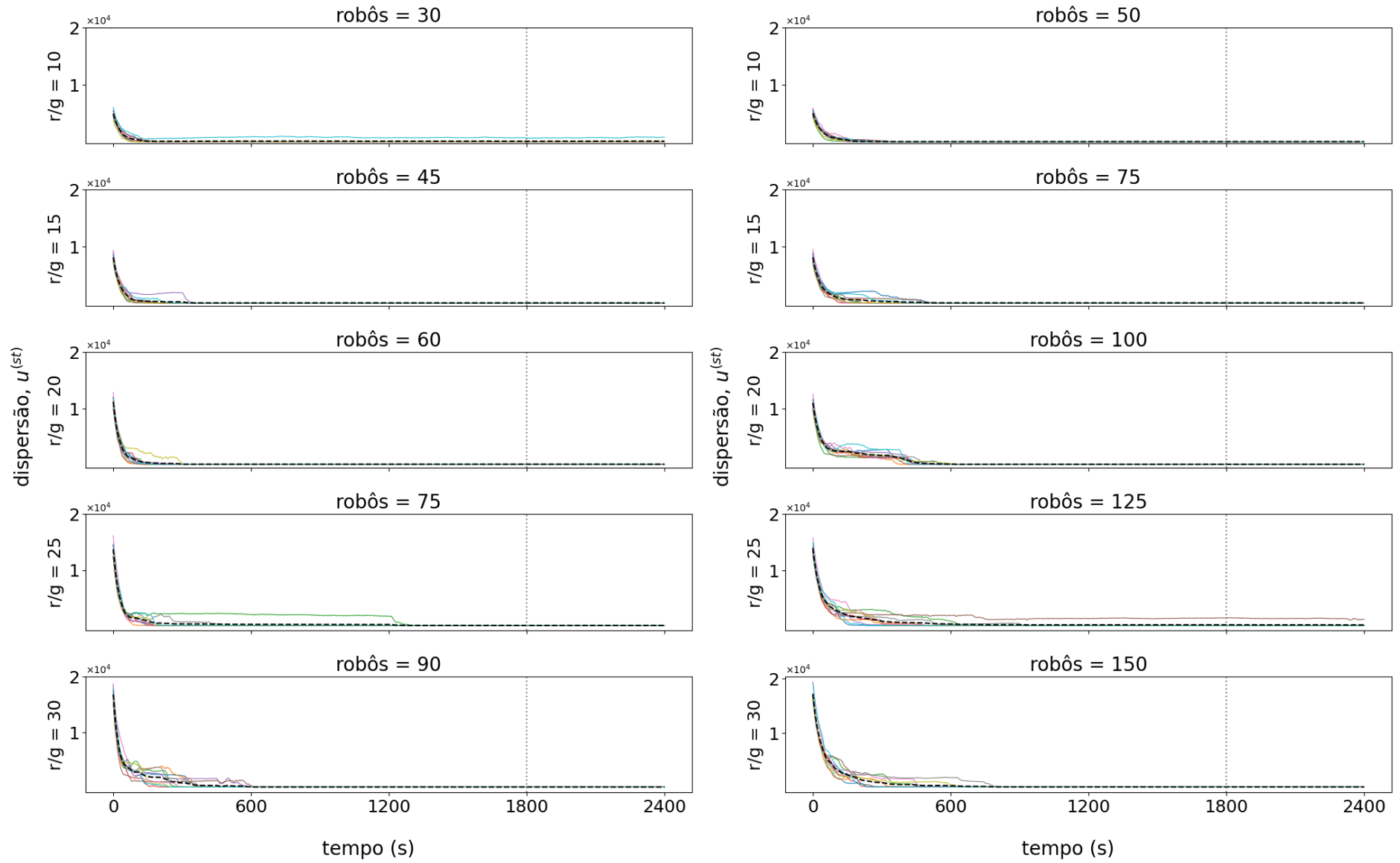


(a) Sem Cilindro de Referência - 3 grupos

(b) Sem Cilindro de Referência - 5 grupos

Fonte: A autora (2025).

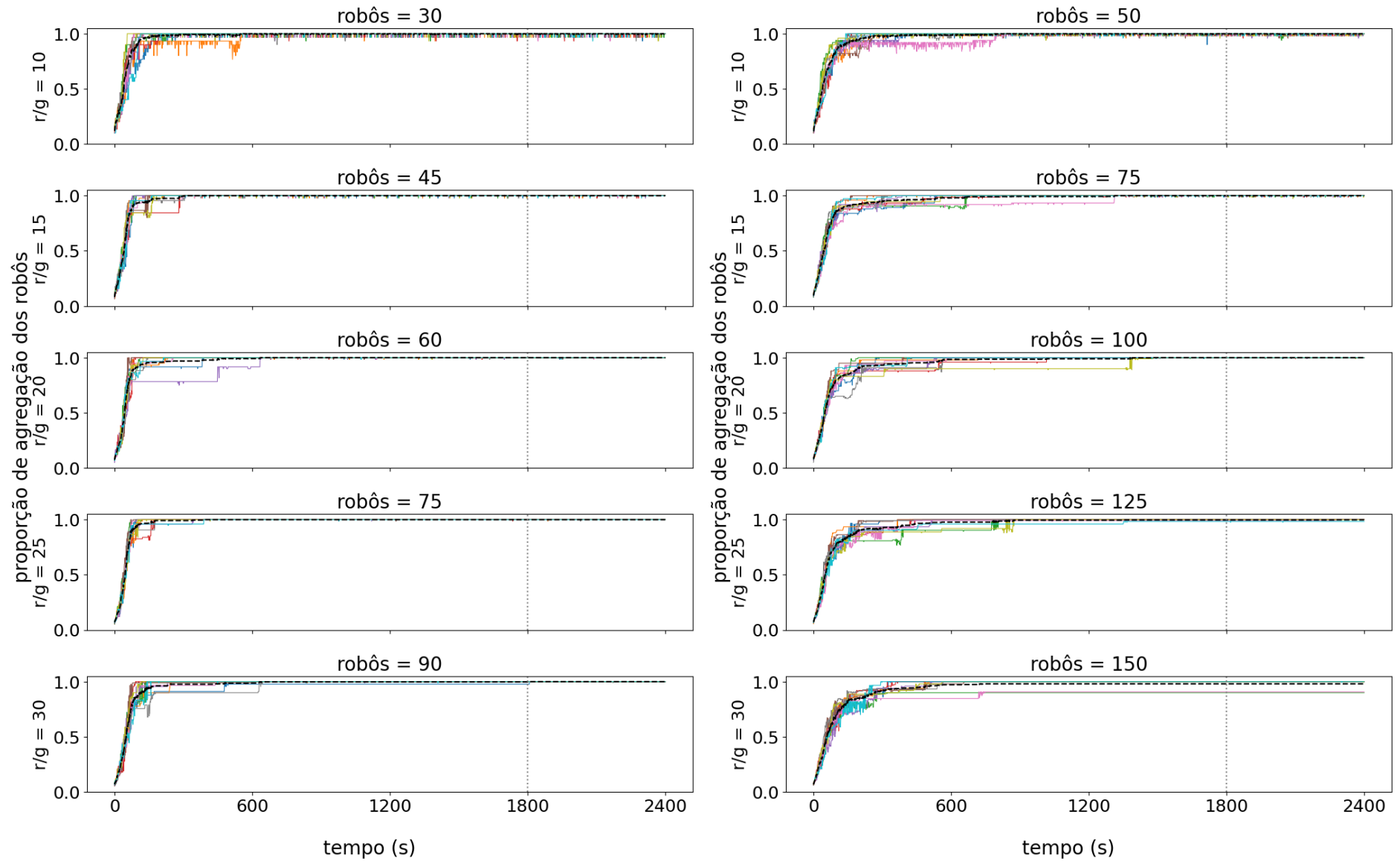
Figura 59 – Dispersão $\bar{v}_7$ Com Cilindro de Referência



(a) Com Cilindro de Referência - 3 grupos

(b) Com Cilindro de Referência - 5 grupos

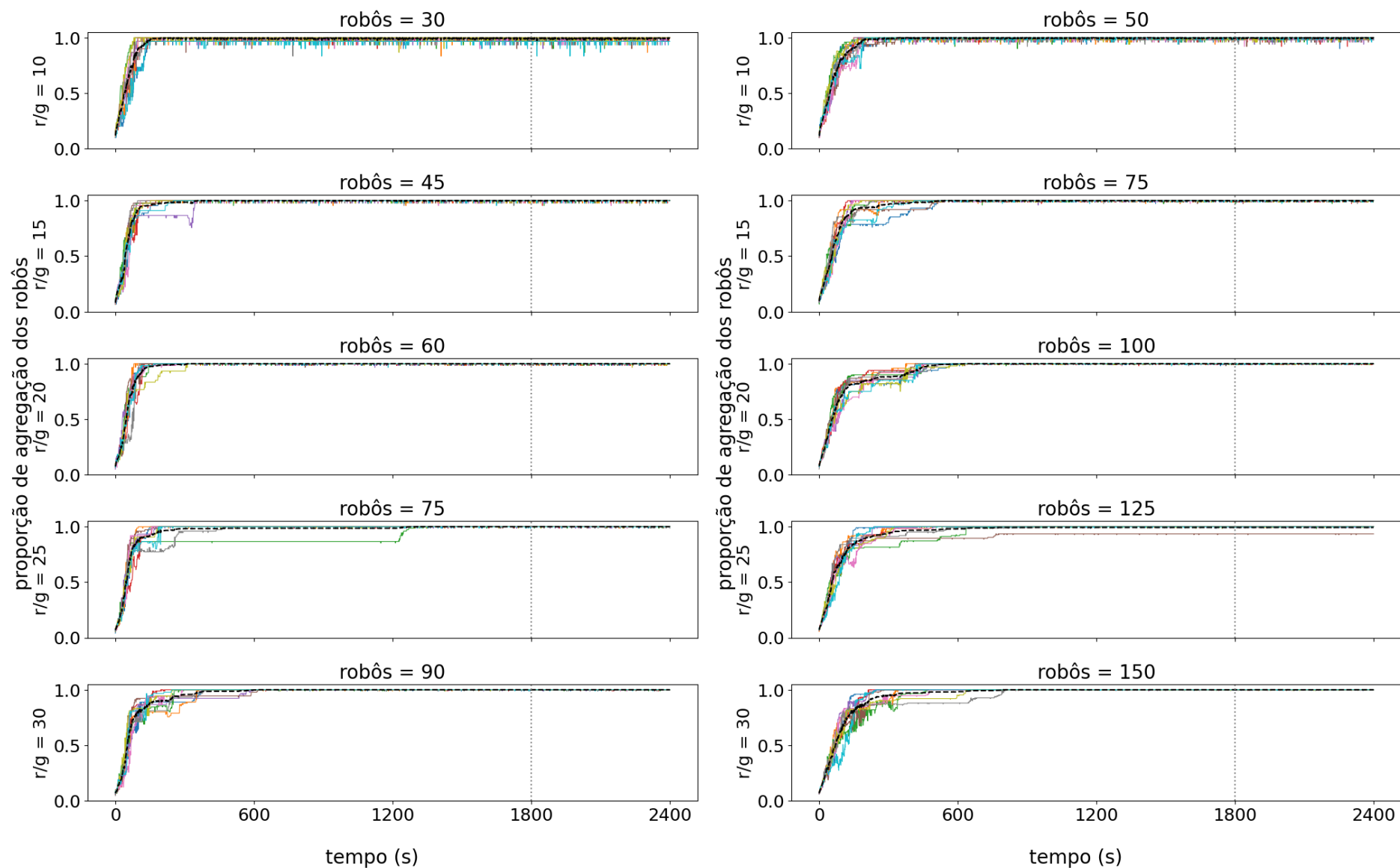
Figura 60 – Proporção $\bar{v}_7$ Sem Cilindro de Referência



(a) Sem Cilindro de Referência - 3 grupos

(b) Sem Cilindro de Referência - 5 grupos

Figura 61 – Proporção $\bar{v}_7$ Com Cilindro de Referência



(a) Com Cilindro de Referência - 3 grupos

(b) Com Cilindro de Referência - 5 grupos

Deste modo, foi possível comparar a abordagem Alternância com as anteriores. Para isso, na Tabela 4 considera-se para fins de comparação, os melhores resultados de cada proposta, após replicação dos experimentos, conforme descritos por cada autor.

Tabela 4 – Segunda Comparação das Configurações Analisadas. (1) Gauci et al. (2014a), (2) Krischanski et al. (2023) (3) Configuração $\bar{v}_0$ e (4) Configuração $\bar{v}_7$.

Tb.	T (s)	Form.	Met.	Tarefa	D.L.A. (cm)	g	r/g	D (s)	N.P.	C.R.	E.F.
1	200	Compacta	G.S.	Única	316 a 3162	1	100 a 1000	1800	4	Não	Sim
2	145	Circular	DIRECT	Multi.	450	3/5	10 a 30	2400	6	Sim/Não	Não
3	540	Compacta	G.S.	Multi.	450	3/5	10 a 30	2400	6	Sim/Não	Não
4	125	Compacta	Alternância	Multi.	450	3/5	10 a 30	2400	6	Sim/Não	Não

Fonte: A autora (2025).

Assim, $\bar{v}_7$, baseado na lógica de alternância, se mostrou uma solução eficiente e robusta, capaz de combinar as melhores características de seus antecessores adicionando estratégias simples. Essa solução destaca-se em todos os critérios analisados: segrega corretamente os grupos, os mantém compactados, atinge o estado estacionário no menor tempo e é menos dependente dos cilindros de referência que os demais.

7 CONCLUSÃO E TRABALHOS FUTUROS

A proposta exposta neste trabalho foi de investigar e solucionar uma lacuna na robótica de enxame: a formação de agrupamentos compactos em cenários de agregação multitarefa com controladores minimalistas. A partir de uma metodologia que combinou uma *grid search* e a otimização por algoritmos automáticos, globais e locais, foi possível desenvolver e validar uma solução que supera as formações circulares e os desafios de escalabilidade reportados por Krischanski et al. (2023). Isso resultou em uma nova solução que combina as vantagens principais dos resultados encontrados aqui e por autores anteriores, o que foi chamado de Algoritmo de Alternância.

As simulações realizadas em ambiente virtual demonstraram que a abordagem de *grid search* foi fundamental para mapear o espaço de parâmetros e identificar o vetor de controle ótimo, $\bar{v}_0$. Este controlador revelou que a estratégia mais eficaz para a segregação de grupos reside em tratar robôs de grupos distintos como obstáculos ou parede.

Como resultado, o $\bar{v}_0$ consolidou-se como uma solução capaz de gerar agregados compactos e estáveis, mantendo uma escalabilidade consistente para os valores testados de 150 robôs totais distribuídos em 3 ou 5 grupos. Paralelamente, o uso dos algoritmos MLSL e SBPLX mostrou um caminho promissor para a automação do *design*, apresentando um custo computacional significativamente inferior à busca exaustiva, embora com menor precisão em espaços de busca completos.

Já a principal contribuição técnica deste trabalho, foi o desenvolvimento do Algoritmo de Alternância. Esta solução híbrida solucionou de forma satisfatória o *trade-off* entre a velocidade de segregação inicial e a qualidade da compactação final. Ao alternar entre o comportamento exploratório acelerado de Krischanski et al. (2023), $\bar{v}_{\alpha_2}$, e a precisão de agrupamento do vetor $\bar{v}_0$. Ela não apenas alcançou taxas de agregação de 100% em cenários complexos, como também superou as limitações de formação de órbitas e anéis observadas em $\bar{v}_{\alpha_2}$.

Esse avanço técnico valida a hipótese de que a inclusão de mecanismos de memória temporal, como o uso de *buffers* e identificação de vizinhança, é importante para estabilizar comportamentos emergentes em enxames de robôs de larga escala. Ainda assim, preservando os princípios de controle descentralizado e auto-organização.

Os resultados obtidos com a solução otimizada $\bar{v}_{\alpha_2}$ demonstraram superioridade em relação as demais soluções exploradas neste trabalho, mantendo-se consistente mesmo em simulações de escalabilidade. Assim, foi possível promover a formação de agregados compactos e segregados, evitando formações circulares e ainda reduzindo o tempo de convergência para o estado estacionário quando comparado a $\bar{v}_0$. O Algoritmo de Alternância demonstra que é possível atingir comportamentos coletivos complexos e rápidos mesmo sob restrições de *hardware* típicas da robótica de enxame.

Apesar dos avanços, é necessário reconhecer as limitações deste trabalho. Primeiramente, toda a validação foi conduzida em um ambiente de simulação, o que implica a existência de

uma lacuna da realidade, a qual depende de experimentos físicos para ser corrigida. Em segundo lugar, o foco foi de natureza prática e empírica, não buscando abordar a impossibilidade teórica da garantia de agregação para controladores bimodais sem memória (Steinberg; Solovey, 2024).

As considerações e limitações deste trabalho abrem caminho para diversas perspectivas futuras. Sugere-se a realização de experimentos físicos para validar a robustez do Algoritmo de Alternância em cenários reais, bem como a avaliação de novos valores para o limiar de densidade. Também é proposta a investigação exaustiva sobre o impacto da lógica de *buffer* na estabilidade global do sistema, pois, diferentemente de abordagens puramente reativas, a introdução deste mecanismo de memória de curto prazo permite que a coesão do enxame seja mantida mesmo diante de oclusões visuais momentâneas. Além disso, a aplicação desta lógica de alternância em outros comportamentos coletivos, como o agrupamento de objetos e o transporte cooperativo, apresenta-se como uma alternativa para expandir as aplicações desta solução em cenários de robótica de enxame.

Por fim, a estrutura lógica estabelecida neste trabalho permite que trabalhos futuros implementem a modelagem formal por meio da Teoria de Controle Supervisório e Autômatos Híbridos. Enquanto até aqui identificou-se os vetores de controle ótimos para a agregação e segregação, a formalização do Algoritmo de Alternância como um autômato permitiria gerenciar as transições entre o comportamento exploratório e a compactação final de forma analítica. Essa abordagem possibilitaria tratar a solução proposta não apenas como uma heurística de sucesso para controladores minimalistas, mas como um sistema verificável, capaz de garantir a estabilidade e a segurança operacional em situações críticas de enxames robóticos em larga escala.

REFERÊNCIAS

- ALBANI, Dario et al. Monitoring and mapping with robot swarms for agricultural applications. In: **2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)**. Lecce, Italy: IEEE, 2017. p. 1–6. ISBN 978-1-5386-2939-0. Authorized licensed use limited to: UDESC - Universidade do Estado de Santa Catarina. Disponível em: <https://doi.org/10.1109/AVSS.2017.8078502>.
- BAYINDIR, Levent. A review of swarm robotics tasks. **Neurocomputing**, Elsevier, v. 172, p. 292–321, 2016.
- BIRATTARI, Mauro et al. Automatic off-line design of robot swarms: A manifesto. **Frontiers in Robotics and AI**, Frontiers Media S.A., v. 6, 7 2019. ISSN 22969144.
- BRAMBILLA, Manuele et al. Swarm robotics: A review from the swarm engineering perspective. **Swarm Intelligence**, Springer Science and Business Media, LLC, v. 7, p. 1–41, 3 2013. ISSN 19353820.
- CAMAZINE, S et al. **Self-Organization in Biological Systems**. Princeton, NJ, USA: Princeton University Press, 2001.
- CHAKRAA, Hamza et al. **Optimization techniques for Multi-Robot Task Allocation problems: Review on the state-of-the-art**. [S.l.]: Elsevier B.V., 2023.
- CHEN, Jianing et al. Occlusion-based cooperative transport with a swarm of miniature mobile robots. **IEEE Transactions on Robotics**, Institute of Electrical and Electronics Engineers Inc., v. 31, p. 307–321, 4 2015. ISSN 15523098.
- CHEN, Jianing et al. Segregation in swarms of e-puck robots based on the brazil nut effect. In: CONITZER et al. (Ed.). **Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)**. Valencia, Spain, 2012. Disponível em: https://www.ifaamas.org/Proceedings/aamas2012/papers/aamas12_chenj.pdf.
- CHUNG, Soon Jo et al. A survey on aerial swarm robotics. **IEEE Transactions on Robotics**, Institute of Electrical and Electronics Engineers Inc., v. 34, p. 837–855, 8 2018. ISSN 15523098.
- COMMUNITY, Enki. **Enki Simulator**. 2025. <https://github.com/enki-community/enki>. Accessed: 2025-04-22.
- Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq). **Árvore do Conhecimento – Diretório dos Grupos de Pesquisa no Brasil**. 2024. Acesso em: 7 maio 2025. Disponível em: <https://lattes.cnpq.br/web/dgp/arvore-do-conhecimento>.
- DAYMUDE, Joshua J. et al. Deadlock and noise in self-organized aggregation without computation. **arXiv preprint**, arXiv:2108.09403, 2021. ArXiv:2108.09403v1 [cs.RO]. Disponível em: <https://arxiv.org/abs/2108.09403>.
- DEVI, Kskn Venkata Ramana et al. A review: Swarm robotics: Cooperative control in multi-agent systems. **E3S Web of Conferences**, EDP Sciences, v. 505, p. 03013, 2024. ICARAE2023. Disponível em: <https://doi.org/10.1051/e3sconf/202450503013>.
- DORIGO, Marco; THERAULAZ, Guy; TRIANNI, Vito. Reflections on the future of swarm robotics. **Science Robotics**, v. 5, n. 49, p. eabe4385, 2020. HAL Id: hal-03362864. Disponível em: <https://hal.science/hal-03362864v1>.

DORIGO, Marco; THERAULAZ, Guy; TRIANNI, Vito. **Swarm robotics: Past, present, and future**. Institute of Electrical and Electronics Engineers Inc., 2021. 1152-1165 p. Licensed under CC BY-NC-ND 4.0. Disponível em: <https://doi.org/10.1109/JPROC.2021.3072740>.

DORIGO, M. et al. Evolving self-organizing behaviors for a swarm-bot. **Autonomous Robots**, v. 17, n. 23, p. 223–245, 2004.

DUAN, Shiqi et al. Coordination of multi-robot systems. In: **Proceedings - 2022 International Conference on Electronics and Devices, Computational Science, ICEDCS 2022**. [S.l.]: Institute of Electrical and Electronics Engineers Inc., 2022. p. 361–367. ISBN 9781665455411.

FENGYING, Yang et al. Decentralized consensus in robotic swarm for collective collision and avoidance. **IEEE Access**, Institute of Electrical and Electronics Engineers Inc., v. 12, p. 72143–72154, 2024. ISSN 21693536.

FIRAT, Ziya et al. On self-organised aggregation dynamics in swarms of robots with informed robots. **Neural Computing and Applications**, Springer, v. 32, p. 13825–13841, 9 2020. ISSN 14333058.

FRANCESCA, Gabriele; BIRATTARI, Mauro. **Automatic design of robot swarms: Achievements and challenges**. [S.l.]: Frontiers Media S.A., 2016.

FRANCESCA, Gianpiero et al. Automode-chocolate: automatic design of control software for robot swarms. **Swarm Intelligence**, Springer New York LLC, Berlin, Germany, v. 9, n. 2, p. 89–112, 9 2015. ISSN 19353820.

GAUCI, M. et al. Evolving Aggregation Behaviors in Multi-Robot Systems with Binary Sensors. In: **Int. Symposium on Distributed Autonomous Robotic Systems (DARS)**. [S.l.]: Springer Verlag, 2014. (Springer Tracts in Advanced Robotics, v. 104), p. 355–367.

GAUCI, Melvin et al. Self-organized aggregation without computation. **The International Journal of Robotics Research**, SAGE Publications, v. 33, n. 8, p. 1145–1161, 2014a.

GAUCI, Melvin et al. **Clustering objects with robots that do not compute**. Natural Robotics Lab, Sheffield Centre for Robotics & Department of Automatic Control and Systems Engineering, The University of Sheffield, 2014b. Unpublished. Disponível em: www.ifaamas.org.

GAUTAM, Avinash; MOHAN, Sudeept. A review of research in multi-robot systems. In: **Proceedings of the 2012 IEEE 7th International Conference on Industrial and Information Systems (ICIIS)**. Chennai, India: IEEE, 2012. ISBN 978-1-4673-1155-5. ISSN 2164-7011.

JOHNSON, Steven G. **NLopt: Nonlinear Optimization Library**. [S.l.], 2022. Accessed: 2025-05-15. Disponível em: https://nlopt.readthedocs.io/en/latest/NLopt_Algorithms/.

KRISCHANSKI, Adilson et al. Multi-instance task in swarm robotics: Sorting groups of robots or objects into clusters with minimalist controllers. In: **IEEE International Conference on Intelligent Robots and Systems**. [S.l.]: Institute of Electrical and Electronics Engineers Inc., 2023. p. 8827–8832. ISBN 9781665491907. ISSN 21530866.

KUCKLING, Jonas. **Recent trends in robot learning and evolution for swarm robotics**. [S.l.]: Frontiers Media S.A., 2023.

KUMAR, Udit; SOMAN, Sumit; JAYADEVA. Benchmarking nlopt and state-of-the-art algorithms for continuous global optimization via iacor. **Swarm and Evolutionary Computation**, Elsevier, v. 27, p. 116–131, 2016. ISSN 2210-6502. Available online 22 October 2015. Disponível em: <https://doi.org/10.1016/j.swevo.2015.10.005>.

LEI, Xiaokang et al. Self-organized multi-target trapping of swarm robots with density-based interaction. **Complex and Intelligent Systems**, Springer International Publishing, v. 9, p. 5135–5155, 10 2023. ISSN 21986053. © The Author(s) 2023. Disponível em: <https://doi.org/10.1007/s40747-023-01014-6>.

LOPES, Yuri K. et al. Supervisory control theory applied to swarm robotics. **Swarm Intelligence**, Springer New York LLC, v. 10, p. 65–97, 3 2016. ISSN 19353820. Received: 16 June 2015; Accepted: 27 January 2016; Published online: 25 February 2016. Disponível em: <https://doi.org/10.1007/s11721-016-0119-0>.

MACEDO, Maria Eduarda Silva de et al. A minimalist controller for autonomously self-aggregating robotic swarms: Enabling compact formations in multitasking scenarios. **arXiv preprint arXiv:2507.13969**, jul. 2025. Short paper manuscript submitted to TAROS 2025. Disponível em: <https://doi.org/10.48550/arXiv.2507.13969>.

MAGNENAT, S.; WAIBEL, M.; BEYELER, A. **Enki Reference documentation**. 2005. <https://lis2.epfl.ch/resources/download/doc1.0/libenki/>. Accessed 01 August 2022.

MARCONI, Marina de Andrade; LAKATOS, Eva Maria. **Fundamentos de metodologia científica**. 5. ed. São Paulo: Atlas, 2003.

MARTÍN, A. A. et al. Google scholar, web of science, and scopus: a systematic comparison of citations in 252 subject categories. **Journal of Informetrics**, v. 12, n. 4, p. 1160–1177, 2018.

MATTSON, Connor et al. Discovery and deployment of emergent robot swarm behaviors via representation learning and real2sim2real transfer. In: VOROBAYCHIK, Yevgeniy; SEN, Sandip; NOWÉ, Ann (Ed.). **Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025)**. Detroit, Michigan, USA: International Foundation for Autonomous Agents and Multiagent Systems, 2025. Licensed under Creative Commons Attribution International 4.0 License. Disponível em: <https://www.ifaamas.org/Proceedings/aamas2025/>.

MITRANO, Peter et al. A minimalistic approach to segregation in robot swarms. In: **2019 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)**. Rutgers University, New Brunswick, NJ, USA: IEEE, 2019. p. 97–103. ISBN 978-1-7281-2876-4.

MIYAUCHI, Genki; LOPES, Yuri K.; GROß, Roderich. Sharing the control of robot swarms among multiple human operators: A user study. In: **2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**. [S.l.: s.n.], 2023. p. 574–580. Issued within the frame of the OpenSwarm project.

MONDADA, F. et al. The e-puck, a robot designed for education in engineering. In: **Conference on Autonomous Robot Systems and Competitions**. [S.l.: s.n.], 2009. v. 1, n. 1, p. 59–65.

ORR, James; DUTTA, Ayan. **Multi-Agent Deep Reinforcement Learning for Multi-Robot Applications: A Survey**. [S.l.]: MDPI, 2023.

ÖZDEMİR, Anıl et al. Finding consensus without computation. **IEEE Robotics and Automation Letters**, IEEE, v. 3, n. 3, p. 1346–1353, 2018. Supplementary material available at <http://ieeexplore.ieee.org>.

ÖZDEMİR, Anıl et al. Spatial coverage without computation. In: **2019 IEEE International Conference on Robotics and Automation (ICRA)**. Montreal, Canada: IEEE, 2019. p. May 20–24.

PAGLIUCA, Paolo; VITANZA, Alessandra. A comparative study of evolutionary strategies for aggregation tasks in robot swarms: Macro- and micro-level behavioral analysis. **IEEE Access**, v. 13, p. 72721–72735, 2025. ISSN 2169-3536. Disponível em: <https://ieeexplore.ieee.org/document/10938154/>.

SCHRANZ, Melanie et al. **Swarm Robotic Behaviors and Current Applications**. [S.l.]: Frontiers Media S.A., 2020.

SHAHZAD, Muhammad Muzamal et al. A review of swarm robotics in a nutshell. **Drones**, MDPI, v. 7, n. 4, p. 269, 2023. ISSN 2504-446X. Open access under CC BY 4.0 license. Disponível em: <https://www.mdpi.com/2504-446X/7/4/269>.

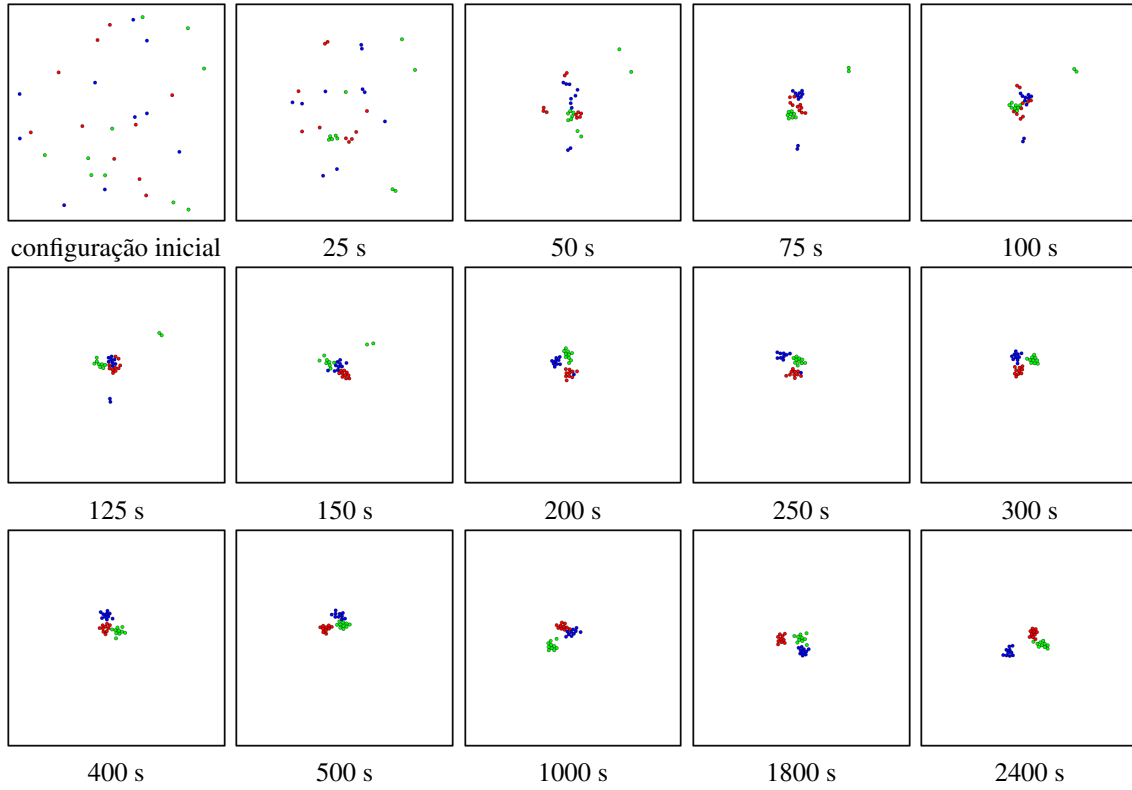
STEINBERG, Roy; SOLOVEY, Kiril. Impossibility of self-organized aggregation without computation. **arXiv preprint arXiv:2501.00390**, 2024. Submitted on 31 Dec 2024. Disponível em: <https://arxiv.org/abs/2501.00390>.

TALAMALI, Mohamed S. et al. Sophisticated collective foraging with minimalist agents: a swarm robotics test. **Swarm Intelligence**, Springer, v. 14, p. 25–56, 3 2020. ISSN 19353820.

WITTMANN, F. **sort-google-scholar**. 2024. Disponível em: <https://github.com/WittmannF/sort-google-scholar.git>. Acesso em: 25 nov. 2024.

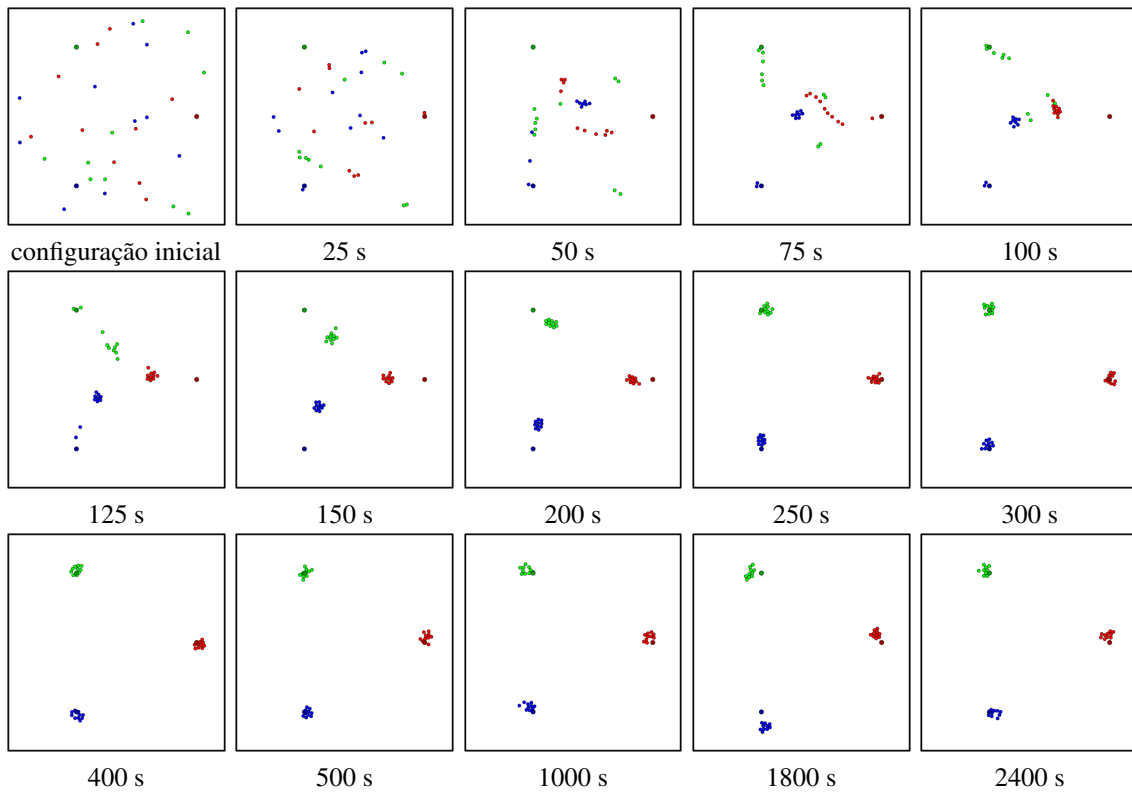
APÊNDICE A – SNAPSHOTS $\bar{v}_7$

Figura 62 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 30 robôs.



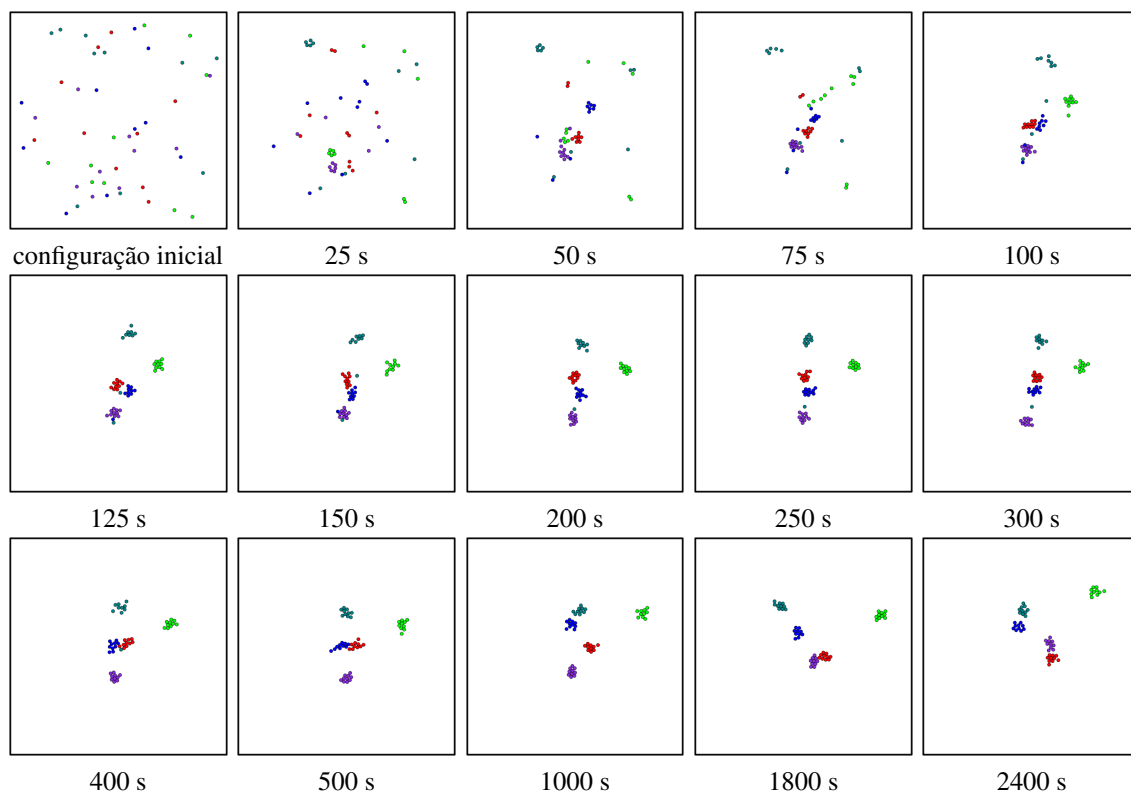
Fonte: A autora (2025).

Figura 63 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 30 robôs.



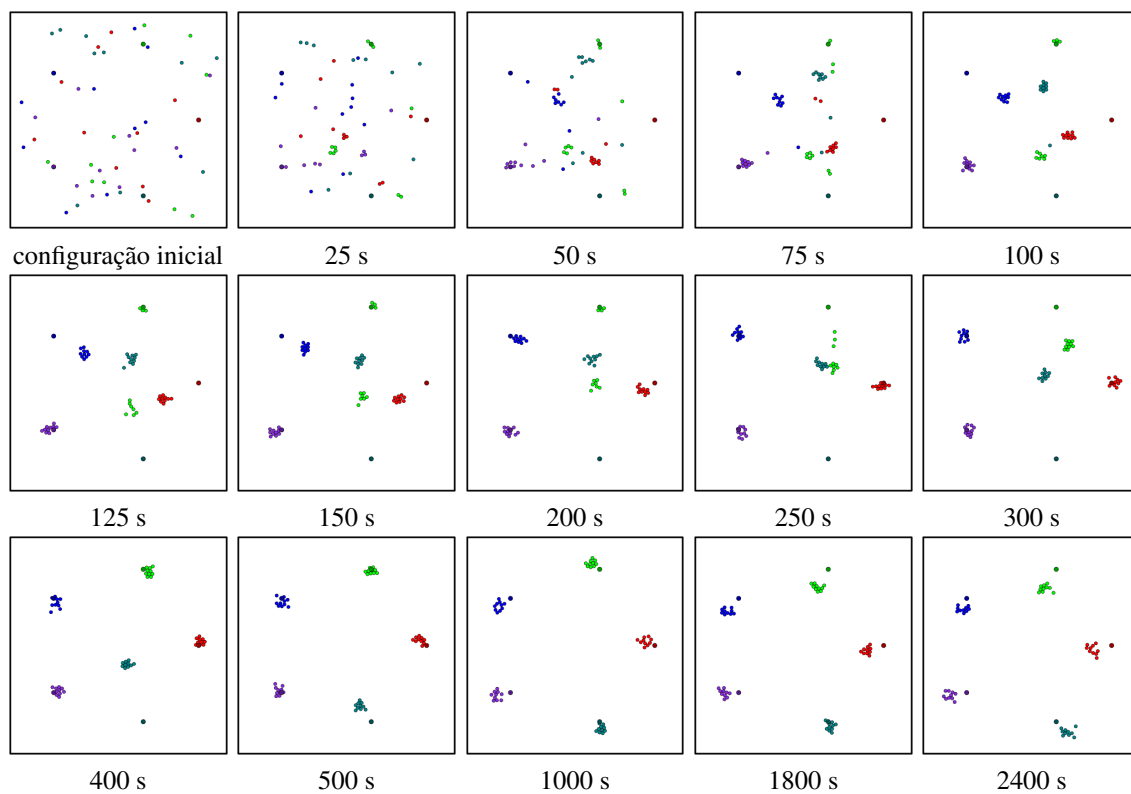
Fonte: A autora (2025).

Figura 64 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 50 robôs.



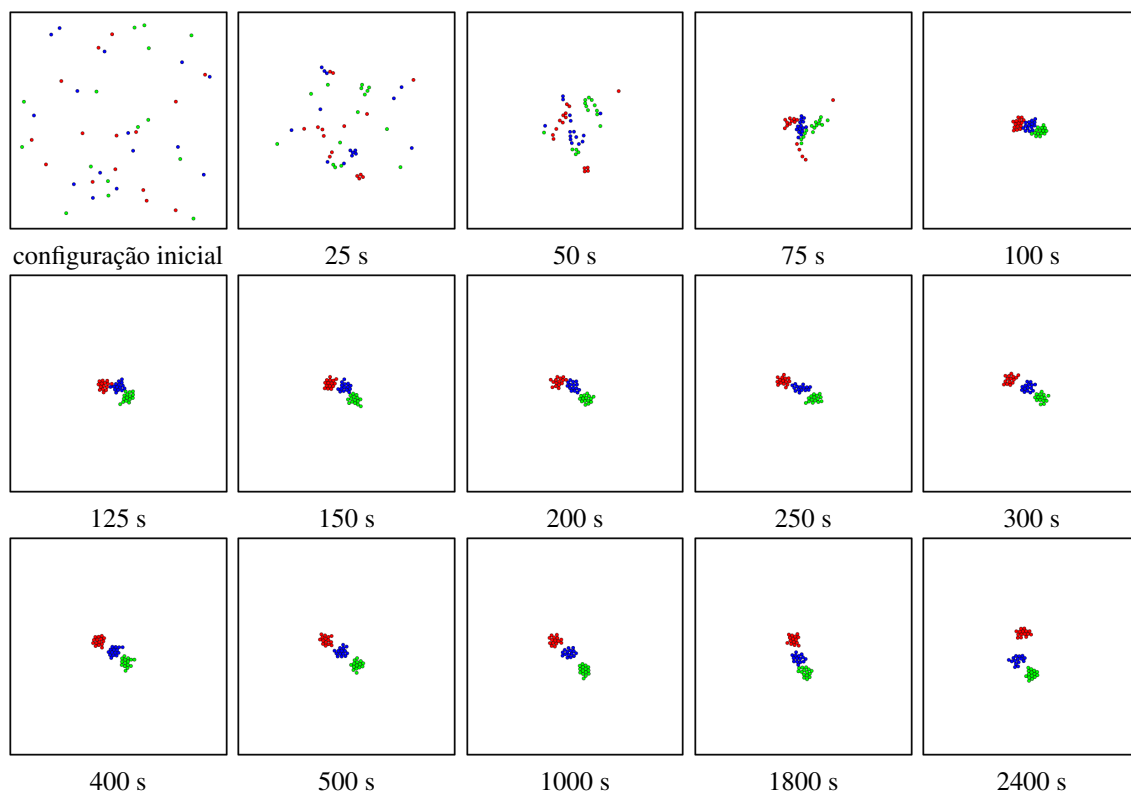
Fonte: A autora (2025).

Figura 65 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 50 robôs.



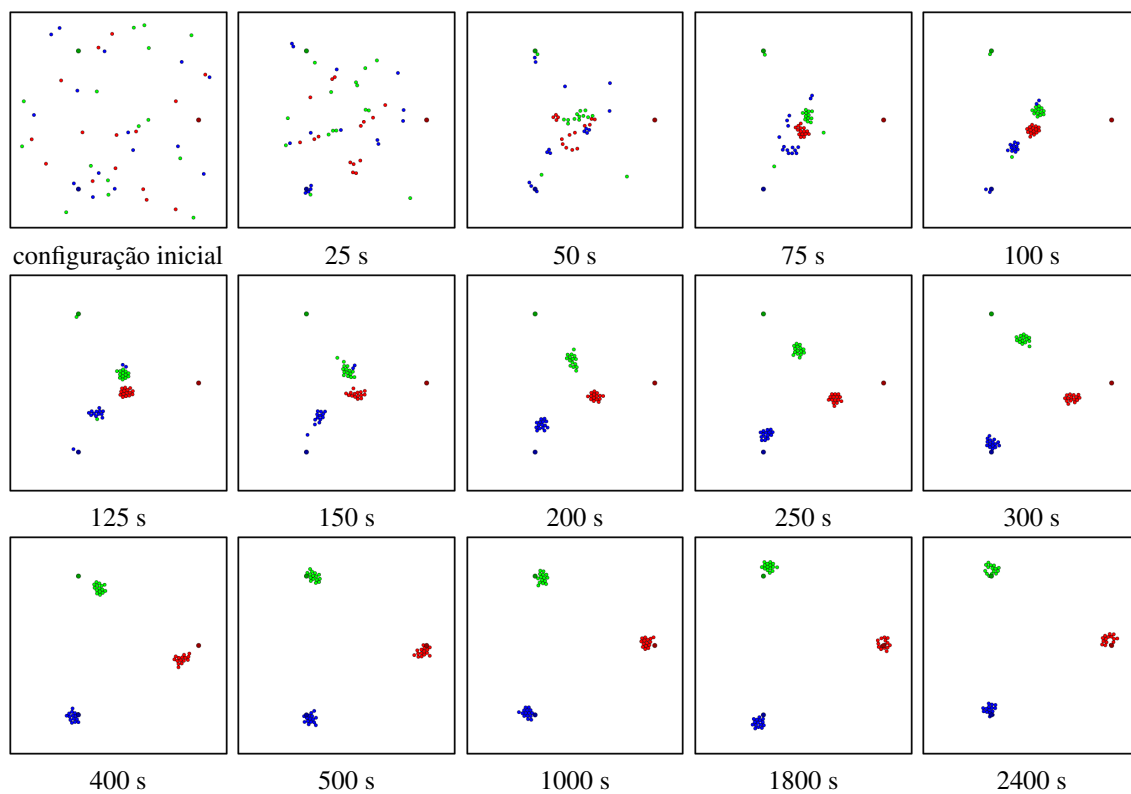
Fonte: A autora (2025).

Figura 66 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 45 robôs.



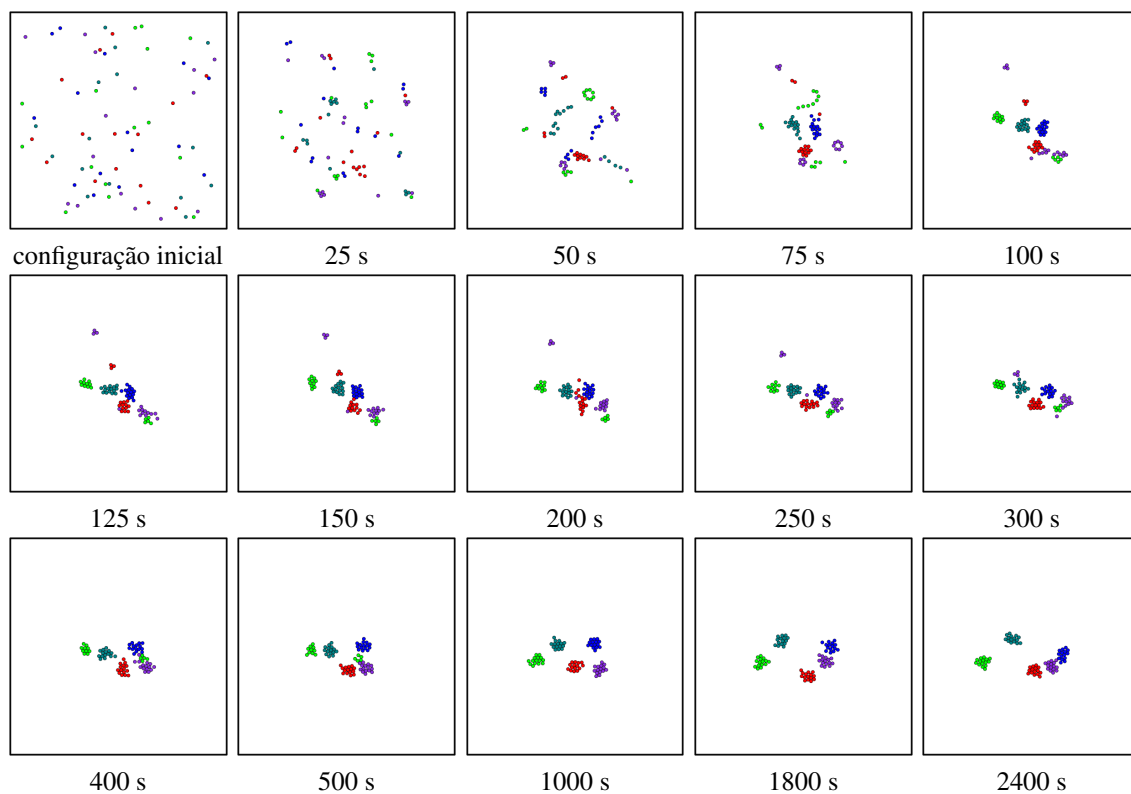
Fonte: A autora (2025).

Figura 67 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 45 robôs.



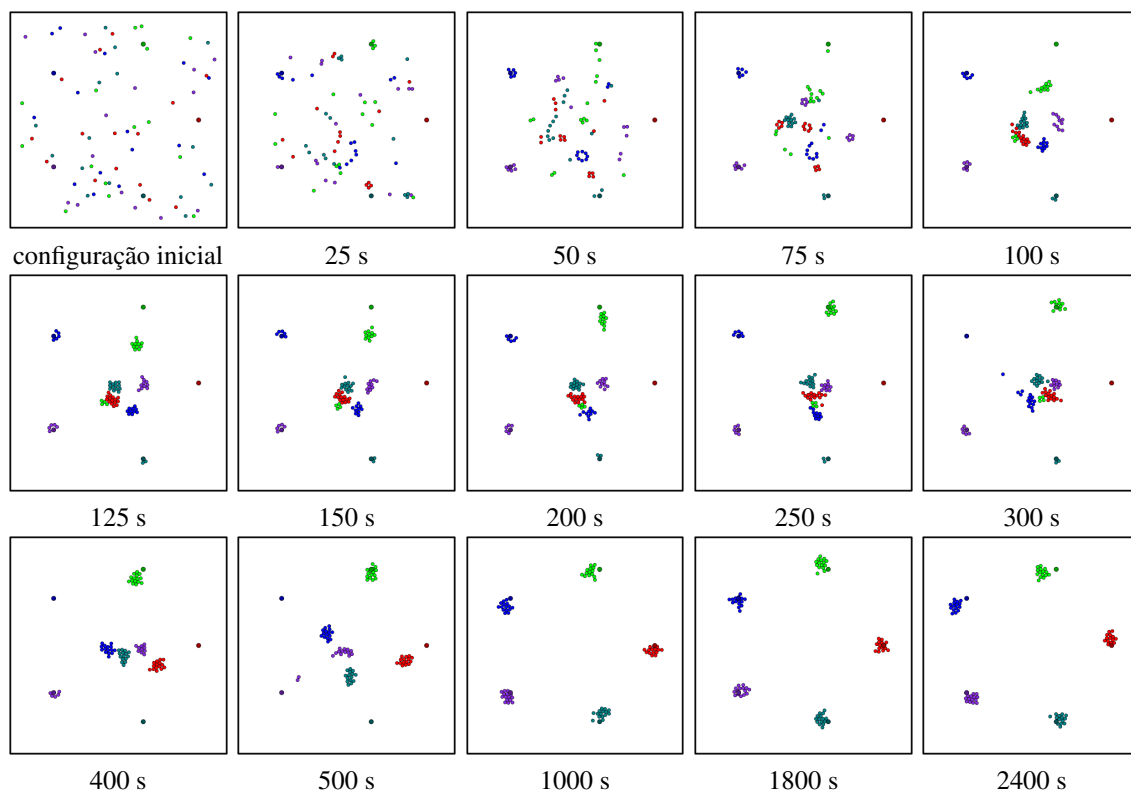
Fonte: A autora (2025).

Figura 68 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 75 robôs.



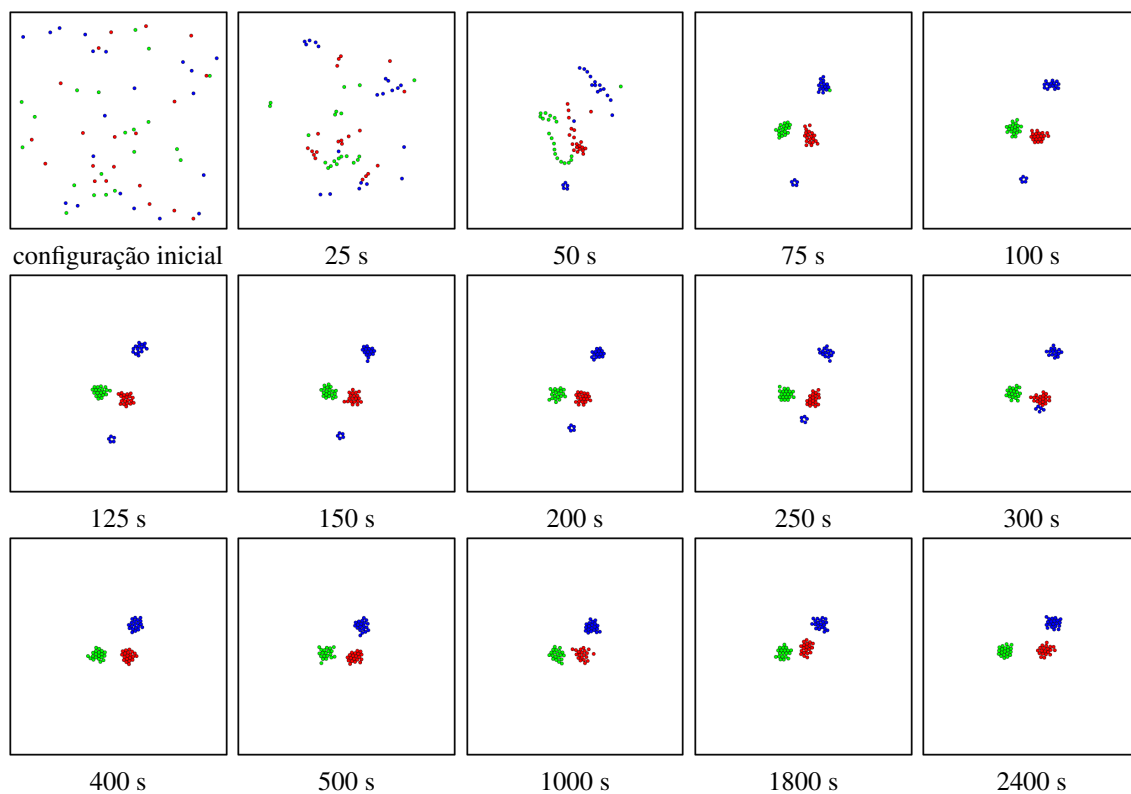
Fonte: A autora (2025).

Figura 69 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 75 robôs.



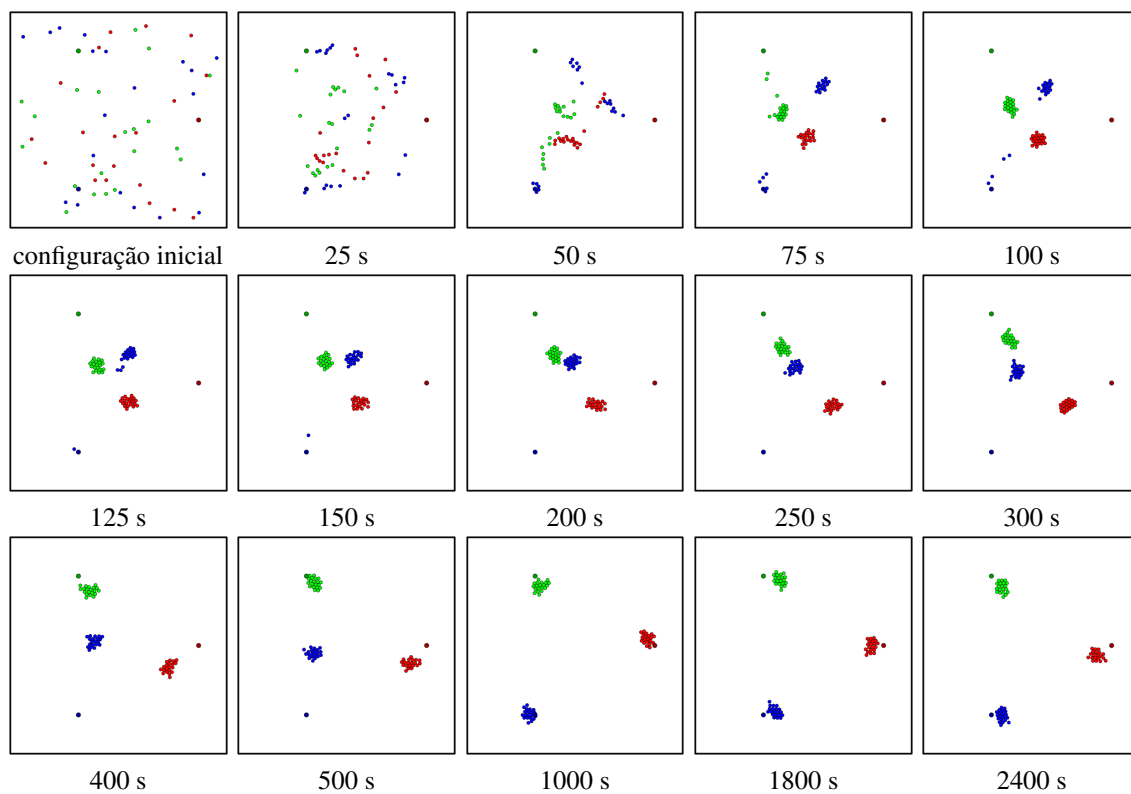
Fonte: A autora (2025).

Figura 70 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 60 robôs.



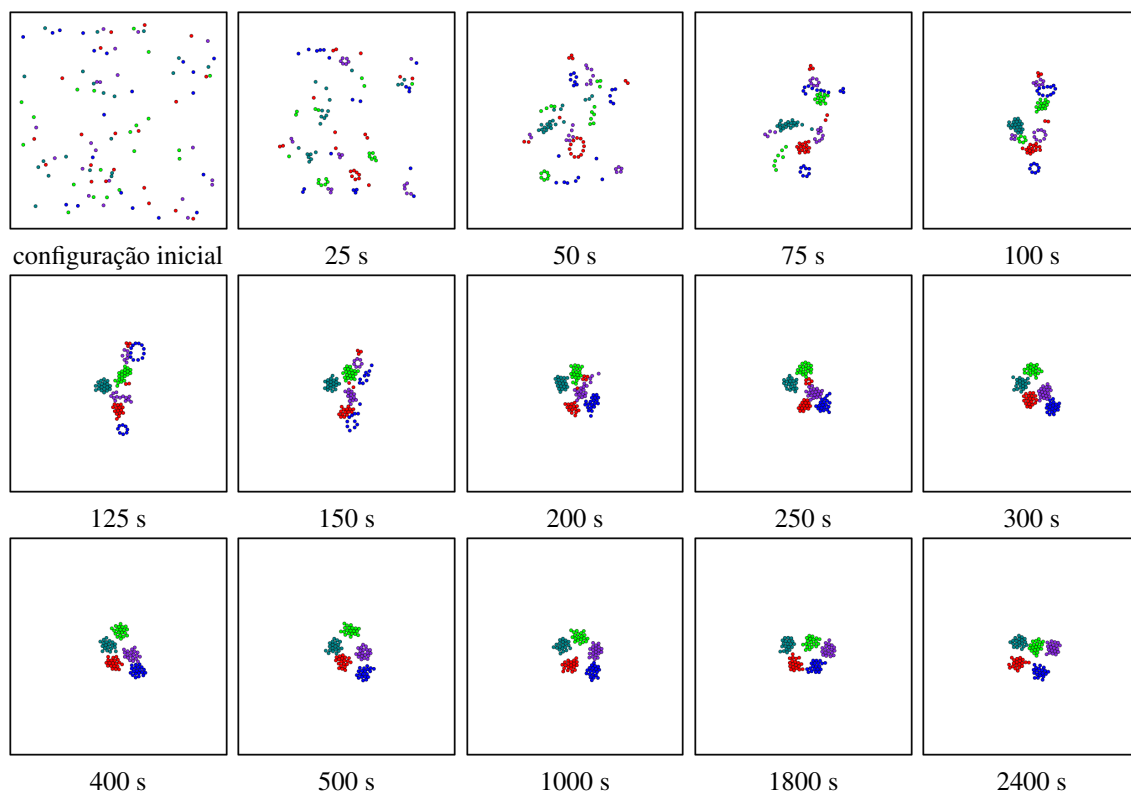
Fonte: A autora (2025).

Figura 71 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 60 robôs.



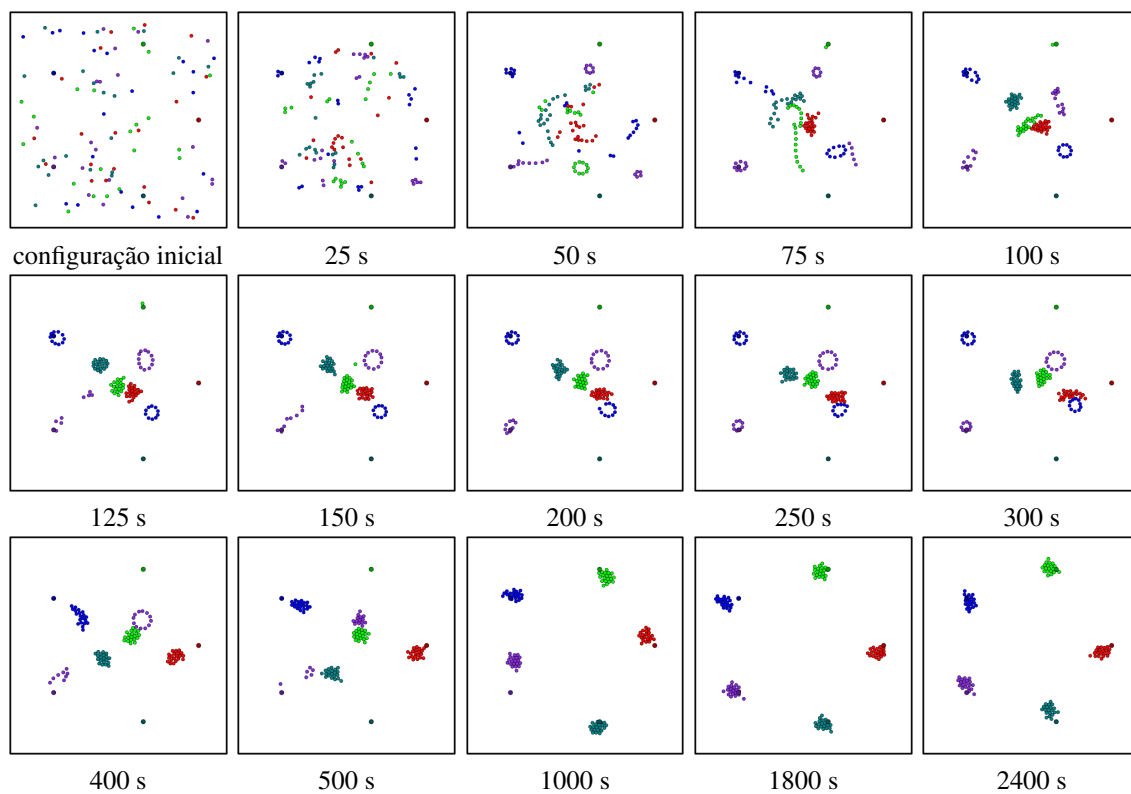
Fonte: A autora (2025).

Figura 72 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 100 robôs.



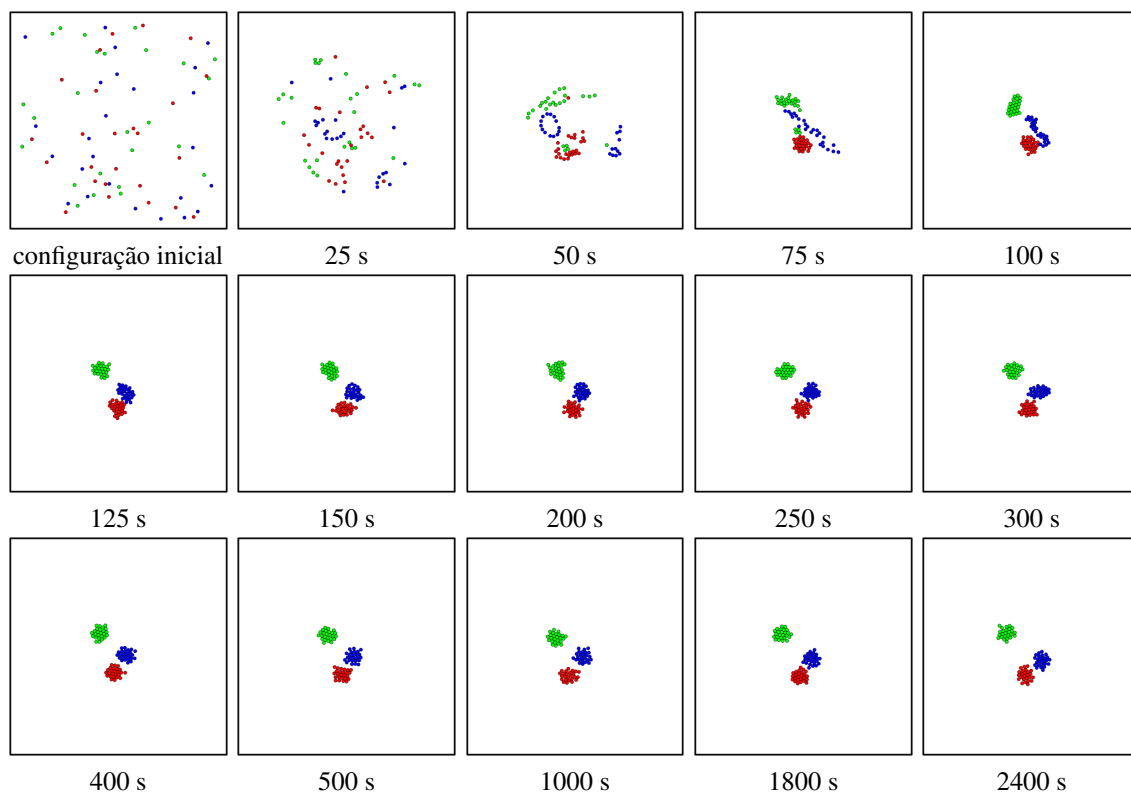
Fonte: A autora (2025).

Figura 73 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 100 robôs.



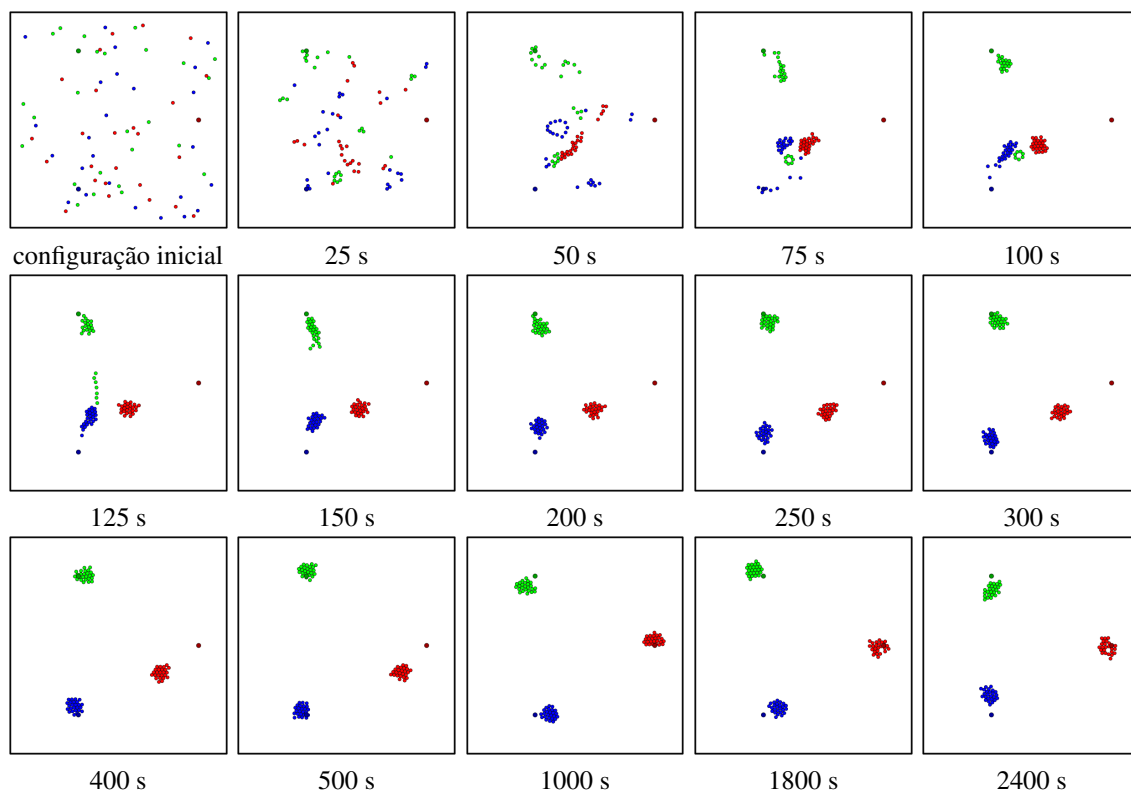
Fonte: A autora (2025).

Figura 74 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 75 robôs.



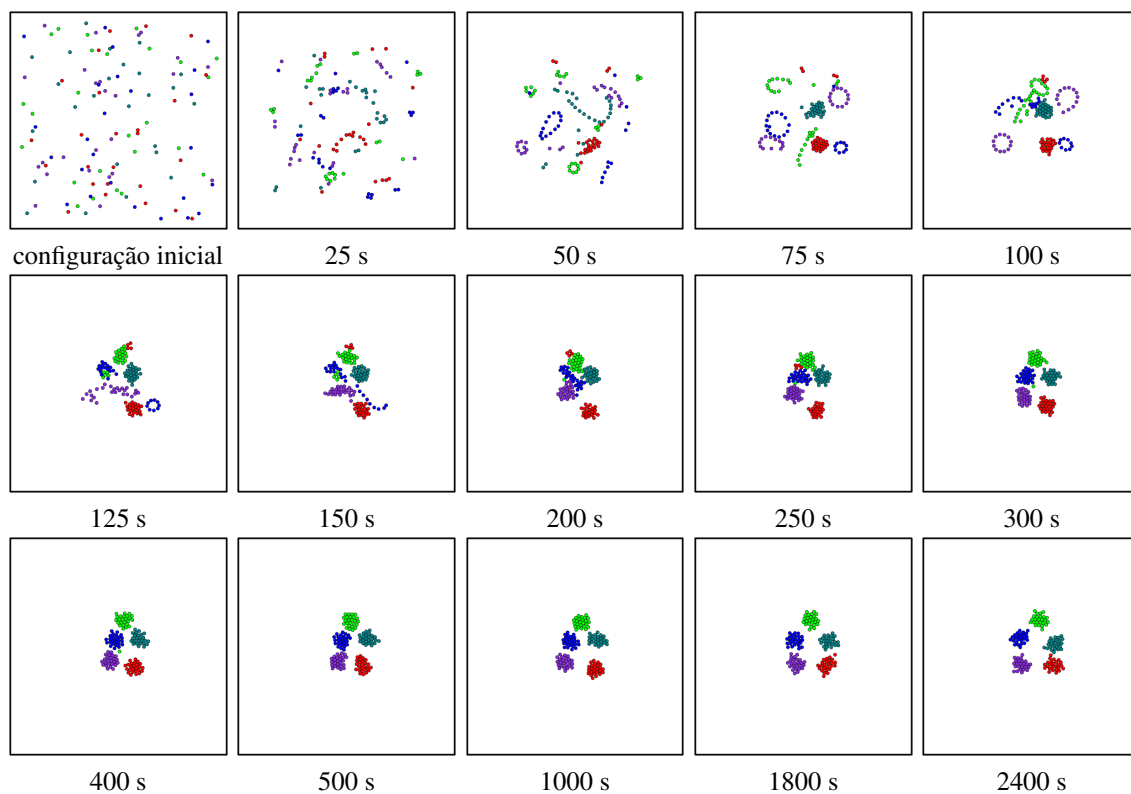
Fonte: A autora (2025).

Figura 75 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 75 robôs.



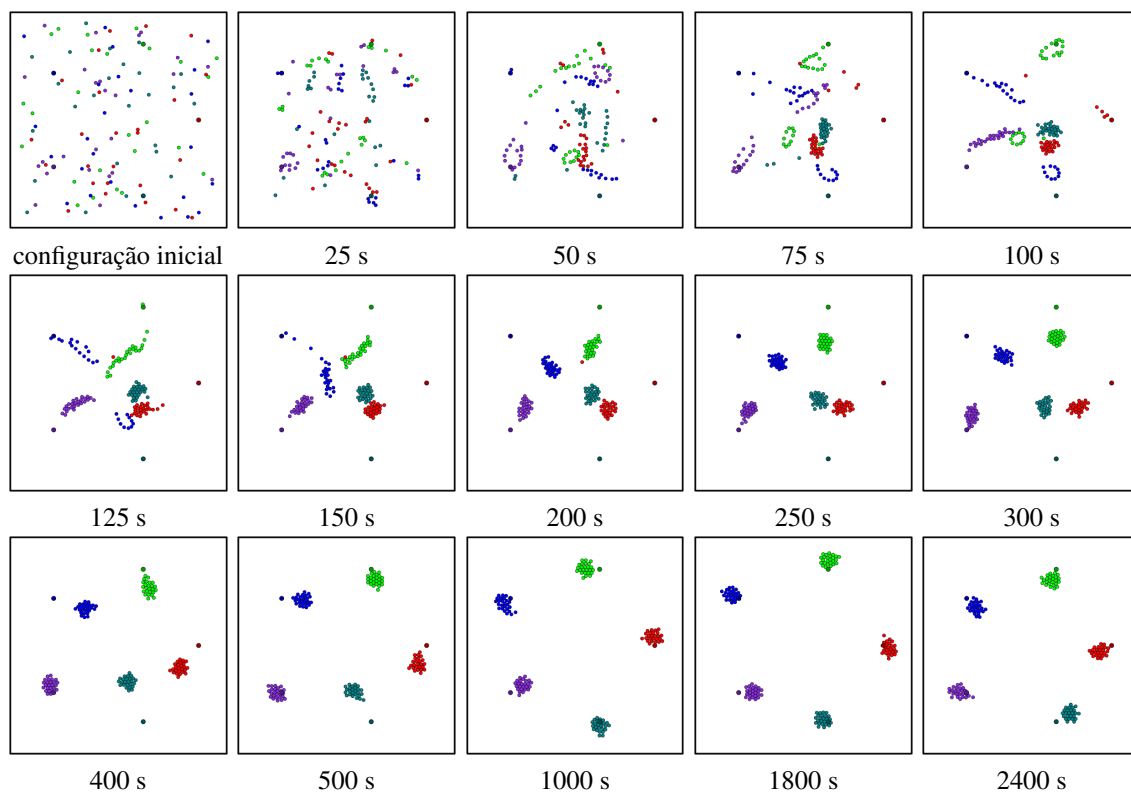
Fonte: A autora (2025).

Figura 76 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 125 robôs.



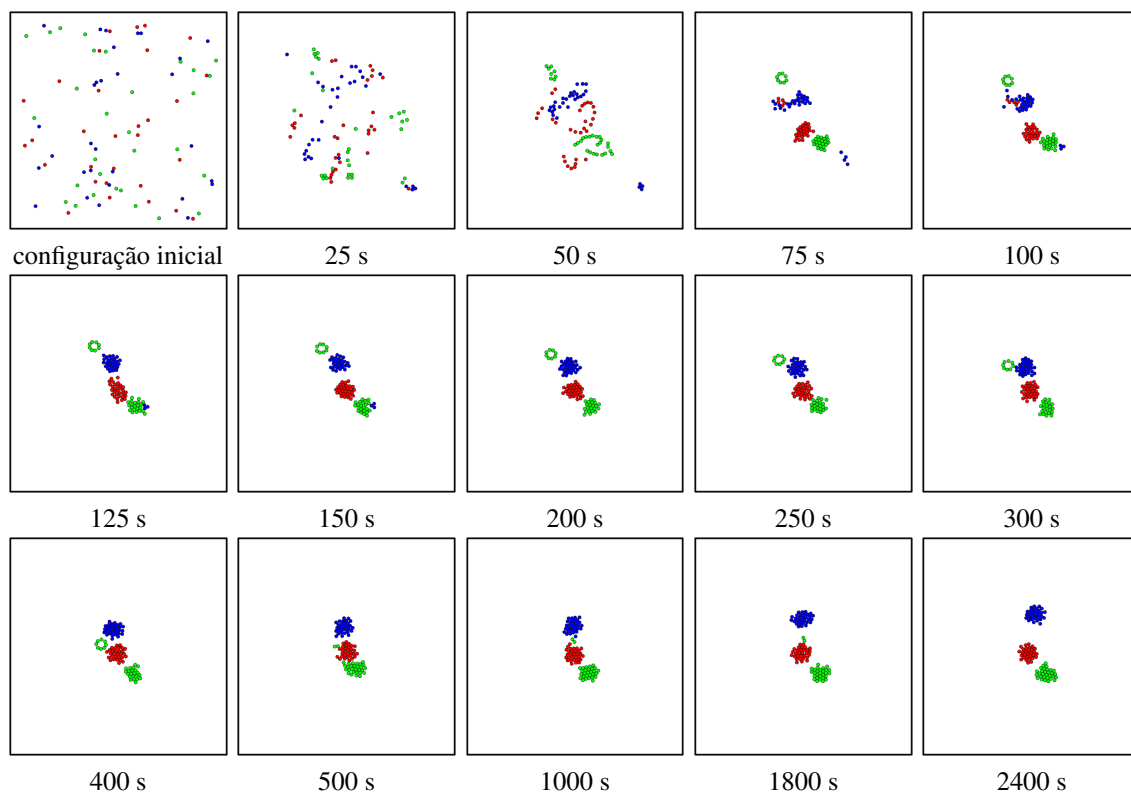
Fonte: A autora (2025).

Figura 77 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 125 robôs.



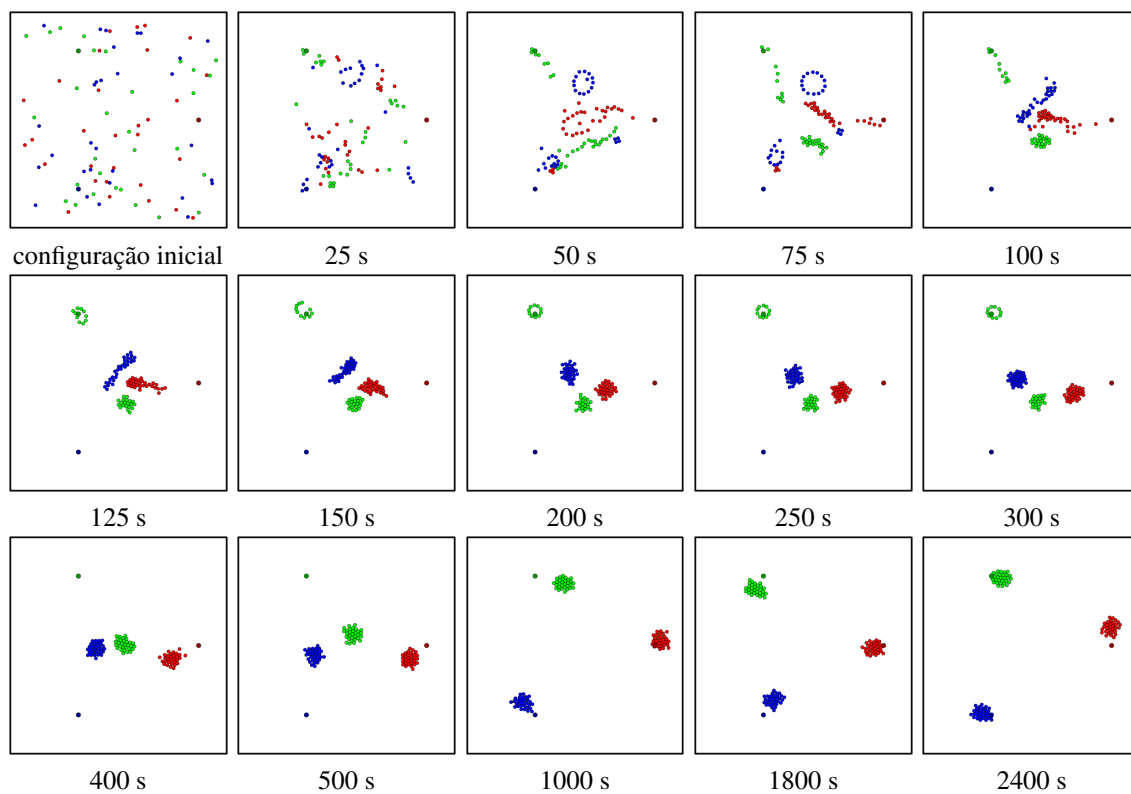
Fonte: A autora (2025).

Figura 78 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 3 grupos, 90 robôs.



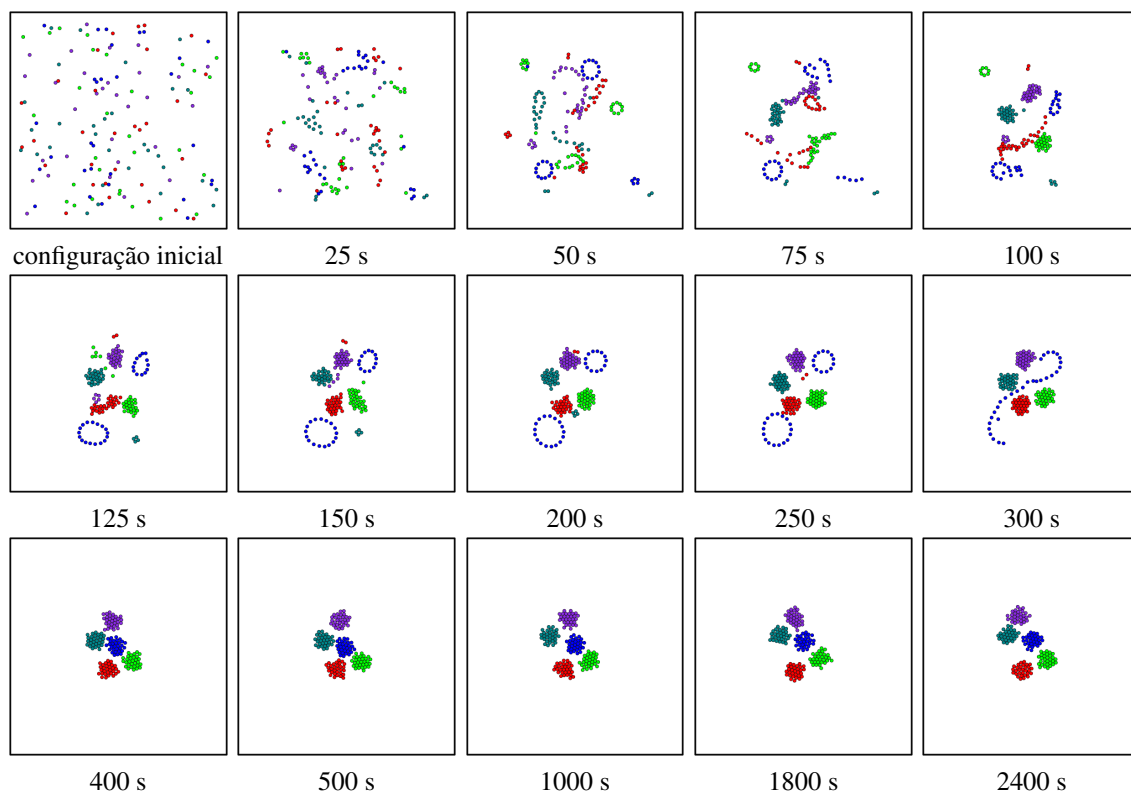
Fonte: A autora (2025).

Figura 79 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 3 grupos, 90 robôs.



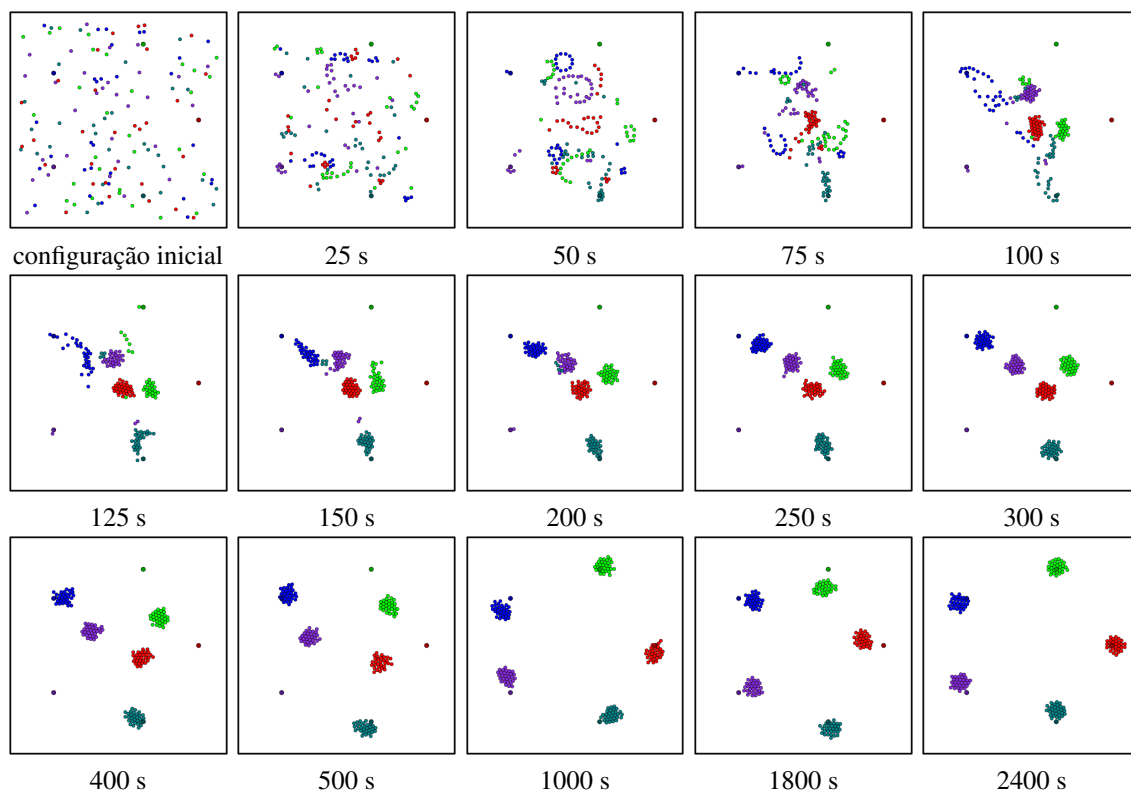
Fonte: A autora (2025).

Figura 80 – Comportamento $\bar{v}_7$ ao longo do tempo. Sem cilindro, 5 grupos, 150 robôs.



Fonte: A autora (2025).

Figura 81 – Comportamento $\bar{v}_7$ ao longo do tempo. Com cilindro, 5 grupos, 150 robôs.



Fonte: A autora (2025).

UNIVERSIDADE DO ESTADO DE SANTA CATARINA – UDESC
BIBLIOTECA UNIVERSITÁRIA
REPOSITÓRIO INSTITUCIONAL

CENTRO DE CIÊNCIAS TECNOLÓGICAS – CCT

ATESTADO DE VERSÃO FINAL

Eu, Roberto Silvio Ubertino Rosso Junior, professor(a) do curso de Mestrado em Computação Aplicada do Centro de Ciências Tecnológicas da Universidade do Estado de Santa Catarina, declaro que esta é a versão final aprovada pela comissão julgadora da dissertação intitulada: **“ALGORITMOS PARA AGREGAÇÃO EM ENXAMES DE ROBÔS: MODELAGEM E IMPLEMENTAÇÃO DE COMPORTAMENTOS COLETIVOS EMERGENTES”** de autoria do(a) acadêmico Maria Eduarda Silva de Macedo.

Joinville, 17 de abril de 2026.

Assinatura digital do(a) orientador(a):

Documento assinado digitalmente
 **ROBERTO SILVIO UBERTINO ROSSO JUNIOR**
Data: 17/04/2026 17:56:43-0300
Verifique em <https://validar.iti.gov.br>

Roberto Silvio Ubertino Rosso Junior