



**UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE
AUTOMAÇÃO E SISTEMAS**

**IRONS: FRAMEWORK PARA SIMULAÇÃO INTEGRADA
EM ROBÓTICA COLABORATIVA**

DANIEL COSTA RAMOS

**Florianópolis/SC
2018**

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Ramos, Daniel Costa
IRONS: FRAMEWORK PARA SIMULAÇÃO INTEGRADA EM
ROBÓTICA COLABORATIVA / Daniel Costa Ramos ;
orientador, Ubirajara Franco Moreno, 2018.
189 p.

Tese (doutorado) - Universidade Federal de Santa
Catarina, Centro Tecnológico, Programa de Pós
Graduação em Engenharia de Automação e Sistemas,
Florianópolis, 2018.

Inclui referências.

1. Engenharia de Automação e Sistemas. 2. Robôs
cooperativos. 3. Redes de comunicação. 4. Framework
de simulação. 5. Método para simulação. I. Moreno,
Ubirajara Franco. II. Universidade Federal de Santa
Catarina. Programa de Pós-Graduação em Engenharia de
Automação e Sistemas. III. Título.

Daniel Costa Ramos

**IRONS: FRAMEWORK PARA SIMULAÇÃO INTEGRADA
EM ROBÓTICA COLABORATIVA**

Esta Tese foi julgada adequada para obtenção do Título de Doutor em Engenharia de Automação e Sistemas e aprovada em sua forma final pelo Programa Pós-Graduação em Engenharia de Automação e Sistemas

Florianópolis, 23 de fevereiro de 2018.

Prof. Daniel Ferreira Coutinho, Dr.
Coordenador do Curso

Banca Examinadora:

Prof. Ubirajara Franco Moreno, Dr.
Orientador
DAS/Universidade Federal de Santa Catarina

Prof. Luiz Affonso Henderson Guedes de Oliveira, Dr.
Membro Externo (Videoconferência)
DCA/Universidade Federal do Rio Grande do Norte

Prof. Eugênio de Bona Castelan Neto, Dr.
DAS/Universidade Federal de Santa Catarina

Prof. Carlos Barros Montez, Dr.
DAS/Universidade Federal de Santa Catarina

AGRADECIMENTOS

Aos meus pais, irmão e namorada por todo apoio, paciência e carinho. Aos meus amigos e companheiros de trabalhos pelas alegrias, tristezas e dores compartilhas. Aos colegas de Portugal e ao professor Luis Almeida, por todo apoio e companheirismo durante e após o estágio sanduíche na Universidade do Porto. E ao professor Ubirajara, pela paciência na orientação e incentivo que tornaram possível a conclusão desta tese.

Daniel Costa Ramos

Resumo da defesa apresentada à UFSC como parte dos requisitos necessários para obtenção do grau de Doutor em Engenharia de Automação e Sistemas.

IRONS: FRAMEWORK PARA SIMULAÇÃO INTEGRADA EM ROBÓTICA COLABORATIVA

Daniel Costa Ramos

Fevereiro/2018

Orientador: Ubirajara Franco Moreno, Dr.

Área de Concentração: Controle, Automação e Sistemas.

Palavras-chave: Sistemas robóticos via rede; robôs cooperativos; redes de comunicação; *framework* de simulação; método para simulação.

Número de Páginas: 189

O aumento do uso de robôs cooperativos móveis em uma variedade de aplicações implica em um aumento significativo na pesquisa de soluções de estratégias de controle cooperativas. Para este propósito, a simulação é uma poderosa ferramenta para testar algoritmos rapidamente, possibilitando uma quantidade maior de testes exaustivos antes de uma implementação em robôs reais. Uma maneira de melhorar a precisão de uma simulação de robótica cooperativa é incorporar elementos pertinentes a sua comunicação e cooperação, ou seja, a rede de comunicação e o controle de topologia. No entanto, elaborar e analisar um estudo simulado envolvendo vários domínios simultaneamente é uma tarefa complexa, onde necessitam-se de procedimentos sistemáticos para a sua construção e de ferramentas adequadas para sua análise. A proposta desta tese é elaborar um *framework* de simulação aliado a um método, este contendo procedimentos e ferramentas, de forma simples e objetiva. A utilização e validação do método e do *framework* é realizado para um conjunto de técnicas, onde é exemplificado como obter respostas aos questionamentos relacionados à interação entre controle cooperativo e redes de comunicação. Após a exposição dos estudos de caso e de suas respectivas contribuições, conclui-se que a simulação integrada é capaz de fornecer informações importantes ao desenvolvedor ainda na fase de simulações e que o método se demonstra adequado para o desenvolvimento da mesma, proporcionando também uma padronização na apresentação e análise dos resultados.

Abstract of Thesis presented to UFSC as a partial fulfillment of the requirements for the degree of PhD in Automation and Systems Engineering.

IRONS: INTEGRATED SIMULATION FRAMEWORK FOR COLABORATIVE ROBOTICS

Daniel Costa Ramos

February/2018

Advisor: Ubirajara Franco Moreno, PhD

Area of Concentration: Control, Automation and Systems.

Keywords: networked robotic systems; cooperative robots; communication network; simulation framework; simulation method;

Number of Pages: 189

The increasing use of mobile cooperative robots in a variety of applications also implies in an increasing research effort on cooperative strategies solutions. For this purpose, simulation is a powerful tool to quickly test algorithms, allowing to do more exhaustive tests before implementation in a real application. One way to improve accuracy in a cooperative robotic system is to incorporate elements of its communication and cooperation in the simulation, i.e., the communication network and the topology control. However, build and assess a simulation study about several simultaneous domains is challenging and requires a systematic procedure to build this simulation and the right tools to assess its results. In this work is proposed a simulation framework with a method that that englobes this procedure and define the necessary tools for this type of simulation. Its utilization and validation was made for a set of techniques, where it is exemplified how to obtain answers for questions related to cooperative strategies and communication network. After exposing the case studies and their contributions, it is conclude that the integrated simulation is capable of providing valuable information for the developer during simulation phase, and it also concluded that the framework and its method demonstrated to be adequate for the integrated simulation development, leading to more realistic simulations while improving results presentation and analysis.

LISTA DE ILUSTRAÇÕES

Figura 1 - Diagrama indicando as áreas relacionadas ao trabalho.....	7
Figura 2 - Exemplo de grafos para representação da comunicação dos robôs e a respectiva matriz adjacência com peso unitário.....	15
Figura 3 - Exemplo de topologia inicial (esquerda) e o resultado para os casos de controle de topologia simplificado (direita).....	16
Figura 4 - Exemplo utilizando menor círculo envolvente para o robô 8.	22
Figura 5 - Relação entre o modelo de referência OSI e o modelo utilizado.....	27
Figura 6 – Camada física e de enlace (OSI) definida pela IEEE 802.11.	28
Figura 7 – Exemplo de correção de atrasos utilizando RA-TDMA.	30
Figura 8 – Diagrama do método de Balci para verificação, validação e testes para modelagem e simulação.....	32
Figura 9 – Proposta de método para modelos simulados por Law, com indicação das etapas de VV&T na parte inferior.....	34
Figura 10 – Método tradicional para ABMS.....	36
Figura 11 – Diagrama com representação da proposta simplificada do método.....	36
Figura 12 - Método para análise de simulações integradas de robôs cooperativos e redes de comunicação.....	36
Figura 13- Comparação distribuição <i>t-student</i> de 2,4,12 e 50 amostras (preto) x distribuição normal (vermelho).	45
Figura 14 - Procedimentos do método com a indicação das técnicas utilizadas.	52
Figura 15- Exemplo do cabeçalho de um documento utilizado no método.....	56
Figura 16 - Estrutura da simulação e suas principais definições.	58
Figura 17 - Possíveis resultados da construção do IC.	69
Figura 18 – Topologias iniciais dos dez casos utilizados para validação.	72
Figura 19 – Resultados gráficos do ponto e tempo de convergência de cada condição inicial, para as simulações e topologias utilizadas na validação.	74

Figura 20 - Geração de referência (ponto circular) baseada em um $dmax$ que gere ao menos duas outras soluções, em robôs (R) diferentes.	76
Figura 21 – Exemplo de representação gráfica do intervalo de confiança.	78
Figura 22 - Intervalos de confiança sobre a variação dos tempos de convergência resultantes dos experimentos de $dmax$ x <i>conectividade</i> .	80
Figura 23 - Interface de programação do OMNeT++.	85
Figura 24 –Interface do OMNeT++ durante uma simulação.	86
Figura 25 - Modelo conceitual dos tópicos da simulação integrada e suas interações.	88
Figura 26 – Organização do <i>framework</i> de simulação implementado no OMNeT++.	88
Figura 27 – Resultados do primeiro estudo de caso em relação ao ponto e tempo de convergência do <i>rendez-vous</i> .	97
Figura 28 - Casos com gerador de mensagens e círculo vermelho representando o alcance da comunicação (interferência).	97
Figura 29 - Resultados de convergência dos casos com topologia fixa, onde as porcentagens indicam a probabilidade média de comunicações com sucesso.	98
Figura 30 – Intervalos de confiança do planejamento experimental para comparação entre MxMPC (e2), utilizando como fator da rede (e1) a comparação IxCM1 (a) e IxCM2 (b).	105
Figura 31 – Intervalo de confiança de 99% sobre o tempo de convergência de cada experimento de comparação.	106
Figura 32 – Intervalos de confiança de 99% entre as técnicas em ambiente com colisão (esquerda) e da mesma técnica em ambientes ideal e com colisão de mensagem (direita), para um ciclo de memória.	106
Figura 33 - Intervalos de confiança de 99% entre as técnicas em ambiente com colisão (esquerda) e da mesma técnica em ambientes ideal e com colisão de mensagem (direita), para dois ciclos de memória.	107
Figura 34 - Intervalo de confiança e indicações visuais das tendências dos resultados ($dmax$ x conectividade) ao incrementar a conectividade em cada experimento.	109
Figura 35 – Intervalos de confiança de 98% sobre a diferença no tempo de convergência para dois casos de robôs heterogêneos.	111

Figura 36 – Configuração inicial dos 20 robôs, para o caso de topologia linear (esquerda) e topologia de grade (direita).....	112
Figura 37 - Tempos de convergência de rendez-vous obtidos para consenso médio e circuncentro e para topologias linear e de grade, em função do número de robôs.	112
Figura 38 – Intervalo de confiança de 98% sobre a) o impacto de cada fator e de sua interação e b) intervalo de confiança sobre a diferença entre as técnicas de consenso no caso da topologia de grade.....	113
Figura 39 – Resultados do primeiro experimento em termos de a) intervalo de confiança de 98% de cada impacto de cada fator e sua interação, e b) tempo de convergência das amostras para cada combinação de fatores.	116
Figura 40 - Resultados do segundo experimento em termos de a) intervalo de confiança de 98% de cada impacto de cada fator e sua interação, e b) tempo de convergência das amostras para cada combinação de fatores.	117
Figura 41 - Processo de validação do consenso FK para pesos médios.	119
Figura 42 - Influência do peso nos tempos de convergência de <i>rendez-vous</i> , para topologia linear (esquerda) e topologia tri-conectada (direita).....	120
Figura 43 - Tempo de convergência em valores absolutos utilizando variação nos pesos λ (0:0,05:5), constante de tempo de 1s e topologias linear e tri-conectada.....	121
Figura 44 - Topologia de comunicação linear (esquerda) e de grade (direita) para mesmas condições iniciais.....	123
Figura 45 - Tempo de convergência da tarefa de <i>rendez-vous</i> ao utilizar diferentes pesos, para topologia linear (linhas superiores) e grade (linhas inferiores), para os constantes de tempo de 100ms (linha sólida) e 1s (linha pontilhada).	123
Figura 46 - Diferença nos resultados ao inserir uma referência falsa com duração de 5s (azul) e de 10s (verde), nos robôs 7, 6 e 10, com redução do peso do robô.....	124
Figura 47 - Diferença nos resultados ao inserir uma referência falsa e alterar os pesos de todos os robôs.	125
Figura 48 - Impacto da intensidade da referência e o desempenho de um consenso com valores limitados da diferença entre posições.....	126
Figura 49 – Página inicial do <i>site</i> para divulgação do método IRoNS.	128

LISTA DE QUADROS

Quadro 1 – Primeira parte do algoritmo de menor círculo envolvente, analisando de dois em dois robôs.....	22
Quadro 2 - Segunda parte do algoritmo de menor círculo envolvente, analisando de três em três robôs.....	23
Quadro 3 – Categorias e subcategorias do protocolo ODD.....	39
Quadro 4 – Exemplo de matriz de projeto para planejamento fatorial 2 ³	47
Quadro 5 – Exemplo de classificação da simulação padrão de acordo com indicadores de qualidade.	63
Quadro 6 – Notas para classificação da simulação padrão de acordo com indicadores de qualidade.	65
Quadro 7 - Parâmetros utilizados para validação.	71
Quadro 8 - Planejamento experimental fatorial D_{max} x Conectividade para E1.	77
Quadro 9 – Possíveis questões a serem tratadas pelo método.	94
Quadro 10 - Métricas e parâmetros utilizados na comparação entre técnicas de consenso.....	102
Quadro 11 - Planejamento experimental para comparação de técnicas de consenso e o impacto da perda de mensagens com retenção de informações.	104
Quadro 12 – Análises realizadas para estudo de caso de comparação entre estratégias de consenso e a situação da rede de comunicação.	104
Quadro 13 - Matriz experimental para o experimento de escalabilidade.	113
Quadro 14 - Matriz experimental para o experimento com comunicação indireta.....	114
Quadro 15 – Matriz experimental para amostras utilizando topologia de grade.....	115
Quadro 16 – Matriz experimental para amostras utilizando comunicação indireta.....	116
Quadro 17 - Questionamentos em relação ao consenso FK.	119
Quadro 18 - Questionamentos em relação à propagação de informações.	122

LISTA DE TABELAS

Tabela 1 – Resultados utilizados para validação estatística.	73
Tabela 2 - Valores dos intervalos de confiança de D_{max} (e^1), conectividade (e^2) e D_{max} x Conectividade (e^{12}) para cada experimento E.	79
Tabela 3 – Intervalos de confiança resultantes da validação de cada parâmetro e cada estratégia de controle.....	103
Tabela 4 - Terminologia e número de artigos encontrados na primeira busca utilizando o Google Scholar.	150
Tabela 5 - Terminologia e número de artigos encontrados na segunda busca utilizando o Google Scholar.	151
Tabela 6 - Terminologia e número de artigos encontrados na terceira busca utilizando o Google Scholar.	151
Tabela 7 - Terminologia e número de artigos encontrados quarta busca utilizando o Google Scholar.	151
Tabela 8 - Terminologia e número de artigos encontrados em outros mecanismos de busca	152
Tabela 9 – Outras terminologias utilizadas, com ferramenta de busca utilizada, número de artigos encontrados e número de artigos utilizados em abril de 2016.	153
Tabela 10 – Outras terminologias utilizadas, com ferramenta de busca utilizada, número de artigos encontrados e número de artigos utilizados em 2017.	153
Tabela 11- Incremento do uso do termo IoRT nos últimos anos.....	153
Tabela 12 – Pontos críticos $t_{v,\gamma}$ para distribuição t-student com grau de liberdade v que satisfaz a probabilidade γ	161

LISTA DE ABREVIATURAS E SIGLAS

ABM	<i>Agent Based Modeling</i>
ABS	<i>Agent Based Simulation</i>
ABMS	<i>Agent Based Modeling and Simulation</i>
CAS	<i>Complex Adaptive System</i>
CI	<i>Condições Iniciais</i>
CT	<i>Controle de Topologia</i>
DES	<i>Discrete Event Simulation</i>
EC	<i>Estudo de Caso</i>
FANET	<i>Flying Ad-Hoc Network</i>
FITA	<i>Fixed Increment Time Advance</i>
IC	<i>Intervalo de Confiança</i>
IRoNS	<i>Integrated Robotic and Network Simulation</i>
IoT	<i>Internet of Things</i>
IoRT	<i>Internet of Robotic Things</i>
LLC	<i>Logic Link Control</i>
LMI	<i>Linear Matrix Inequalities</i>
LQR	<i>Linear Quadratic Regulator</i>
M&S	<i>Modelagem e Simulação</i>
MAC	<i>Medium Access Control</i>
MANET	<i>Mobile Ad-Hoc Network</i>
MAS	<i>Multi-Agent Systems</i>
MILP	<i>Mixed Integer Linear Programming</i>
MPC	<i>Model Predictive Control</i>
MSN	<i>Mobile Sensor Network</i>
MRS	<i>Multi-Robot Systems</i>
NETA	<i>Next Event Time Advance</i>
NCS	<i>Networked Control System</i>
NRS	<i>Networked Robotic System</i>
NS	<i>Network Simulator</i>
ODD	<i>Overview, Design concepts and Details</i>
RA-TDMA	<i>Reconfigurable and Adaptive TDMA</i>
RHC	<i>Receding Horizon Control</i>
RC	<i>Robôs Cooperativos</i>
S&M	<i>Simulação e Modelagem</i>
SI	<i>Simulação Integrada</i>
SN	<i>Sensor Network</i>

SP	<i>Simulação Padrão</i>
TDMA	<i>Time Division Multiple Access</i>
UAV	<i>Unmanned Aerial Vehicle</i>
WNCS	<i>Wireless Networked Control System</i>
WNCRS	<i>Wireless Networked Cooperative Robotic System</i>
WNRS	<i>Wireless Networked Robotic System</i>
WMSN	<i>Wireless Mobile Sensor Network</i>
WSN	<i>Wireless Sensor Network</i>
VANET	<i>Vehicular Ad-Hoc Network</i>
VV&T	<i>Verification, Validation and Test</i>

TERMINOLOGIA E DEFINIÇÕES ADOTADAS

Robô: Utilizado neste trabalho para definir o dispositivo móvel com rodas, capaz de realizar de forma autônoma, sua locomoção e medição de variáveis do ambiente. Os termos *veículo* e *agente*, no contexto deste trabalho, representam este mesmo dispositivo. O termo refere-se a um robô móvel terrestre com rodas, a não ser que explicitamente esteja escrito o contrário.

Agente: é uma entidade autônoma que pode mensurar o ambiente (incluindo outros agentes) e usar estas informações para tomada de decisões. Um robô autônomo com sensores é um agente, mas um agente não é necessariamente um dispositivo físico, sendo o termo utilizado para indicar qualquer entidade que possua as características descritas.

Cooperação: ato de um robô agir junto com outro, ou outros, para um fim comum.

Tarefa: ação ou conjunto de ações a serem realizadas pelo robô para o cumprimento de determinado objetivo ou meta. O termo *missão* pode ser utilizado para definir um conjunto amplo de tarefas ou, como neste trabalho, utilizado como sinônimo de tarefa.

Sistema Cooperativo: é um sistema focado na troca de informações, compartilhamento de recursos e coordenação dos esforços de trabalho, para concluir uma determinada ação ou tarefa.

Robôs Cooperativos (RC): são robôs em um sistema cooperativo, realizando cooperação para concluir determinada tarefa. Os termos *multi-veículos*, *multi-agentes*, *multi-robôs*, *sistema multi-robôs (MRS)*, *equipe de robôs*, *equipe de veículos* e *conjunto de robôs*, são considerados sinônimos no contexto do trabalho, pois tratam sobre o mesmo grupo de robôs e os mesmos são sempre cooperativos. Por este mesmo motivo, o termo também é aplicado como sinônimo a *WNCRS (Wireless Networked Cooperative Robotic Systems)*, a não ser que explicitamente escrito ao contrário.

Estratégia de Controle Cooperativo: abordagem de controle (centralizada ou descentralizada) que possui o objetivo de direcionar os robôs cooperativos a ter um certo comportamento desejado e cumprir uma determinada tarefa. É sinônimo de *estratégia cooperativa*.

Comunicação: ato ou efeito de comunicar; ato de transmitir dados ou informações (dados tratados) entre dois ou mais robôs. As expressões *troca de mensagens* e *troca de informações*, estão relacionados à comunicação entre robôs.

Redes de comunicação: aplicado a este trabalho, refere-se à infraestrutura da comunicação entre os robôs, incluindo dentre outros aspectos, a geração de sinal, sincronismo, protocolos e encriptação. No contexto deste trabalho, a palavra *redes* refere-se a redes de comunicação, a não ser que explicitamente escrito ao contrário.

Rede de comunicação *wireless*: aplicado a este trabalho, refere-se à uma rede de comunicação que utiliza tecnologia sem fio para transmissão e troca de dados entre os robôs. Também citada como rede de *comunicação sem fio* ou *rede sem fio*.

Camadas de Rede: refere-se aos níveis (camadas) de rede definidas pelo modelo de referência OSI (*Organization for Standardization*).

SN, WSN, MSN, WMSN: uma rede de sensores (*Sensor Network* - SN) consiste de dispositivos espacialmente distribuídos que utilizam sensores para monitorar condições físicas ou do ambiente, sendo esse o seu principal objetivo. Quando utiliza uma rede de comunicação *wireless*, é denominada de rede de sensores sem fio (*Wireless Sensor Network* - WSN). De forma similar, quando os dispositivos são capazes de se deslocar, com movimento controlado ou não, são denominados de rede de sensores móveis (*Mobile Sensor Network* – MSN). Quando móveis e utilizando comunicação sem fio, são denominados de rede de sensores móveis sem fio (*Wireless Mobile Sensor Network* - WMSN).

NCS, WNCS: um sistema de controle via rede (*Networked Control System* – NCS) é um sistema de controle onde os sinais de controle e de realimentação são transmitidos entre seus componentes no formato de pacotes de informação em uma rede de comunicação. Quando utiliza uma rede de comunicação sem fio, é denominado como sistema de controle via rede sem fio (*Wireless Networked Control System* – WNCS).

NRS/WNRS: um sistema robótico via rede (*Networked Robotic System* – NRS) se trata de um caso específico do NCS, onde são considerados aspectos da rede de comunicação no sistema robótico. No caso de uma rede sem fio, pode ser denominado de WNRS (*Wireless Networked Robotic System* - WNRS). O termo NRS é adotado como padrão neste trabalho dado sua ampla utilização na literatura em trabalhos de robótica

e de redes de comunicação, no entanto, no contexto deste trabalho o termo pode ser considerado sinônimo de WNCRS.

WNCRS: de forma similar ao WNRS, mas aplicado ao caso de robôs cooperativos cuja a cooperação é realizada por uma rede de comunicação sem fio (*Wireless Networked Cooperative Robotic System – WNCRS*). Apesar desta sigla definir o escopo deste trabalho, adotou-se o termo NRS que é o termo mais utilizado na literatura.

Framework: um *framework* provê uma solução para uma família de problemas semelhantes, usando um conjunto de classes e interfaces que mostra como decompor a família de problemas. O conjunto de classes deve ser flexível e extensível para permitir a construção de várias aplicações com pouco esforço, especificando apenas as particularidades de cada aplicação.

Método: é um conjunto de regras ou critérios que servem de referência no processo de busca da explicação ou da elaboração de previsões, em relação a questões ou problemas específicos.

Metodologia: é o campo em que se estuda os melhores métodos praticados em determinada área para a produção do conhecimento.

Simulação: consiste na utilização de técnicas matemáticas e softwares, nos quais permitem imitar o funcionamento de um tipo de operação ou processo do mundo real.

Simulação de Redes: é uma simulação que utiliza modelos de redes de comunicação para emular transmissão de informações entre dispositivos. Tratam desde o aspecto de implementação de camadas de redes, quanto protocolos e transmissão de dados (com ou sem fio) no meio físico.

Simulação de Robótica Cooperativa: é simulação que utiliza modelos e técnicas para controle, sensoramento e atuação de robôs, modelos de estratégia de controle cooperativa e de troca de informações entre os robôs, com objetivo de simular uma determinada tarefa cooperativa.

Simulação Integrada: denominação utilizada neste trabalho para indicar uma simulação conjunta (simultânea no mesmo simulador) de redes de comunicação e de robótica cooperativa.

Simulação de Eventos Discretos: simula a operação de um sistema como uma sequência de eventos discretos no tempo. Cada evento ocorre em um determinado instante de tempo e determina uma mudança no sistema.

ABS/ABMS: a simulação baseada em agentes é um caso particular de uma simulação a eventos discretos, onde as entidades (agentes) interagem de forma ativa com os demais agentes e o ambiente. Quando também envolve o processo modelagem, é comum o uso do termo ABMS.

Rendez-vous: tarefa onde todos os robôs cooperativos devem concordar e atingir um determinado ponto ou região de encontro. A literatura também utiliza o termo “*Rendezvous*”. No entanto, neste trabalho foi adotada a forma separada.

Link de Comunicação: estabelecimento de conexão entre dispositivos, com finalidade de permitir a comunicação.

Topologia de Comunicação: indica entre quais robôs há um *link* de comunicação direta (direcional ou bidirecional). A topologia de comunicação pode ser física ou lógica. No caso de uma rede de comunicação sem fio, a topologia de comunicação física é determinada pelo alcance máximo do dispositivo *wireless* de cada robô e por quais robôs estão fisicamente dentro deste limite; na topologia lógica essa determinação é dada por uma lógica que não corresponde necessariamente ao alcance da comunicação.

Controle de Topologia: é a técnica na qual altera a topologia de comunicação entre os robôs por meio de mobilidade, intensidade do sinal ou omissão de mensagens, em acordo com algum critério para obter uma topologia de comunicação lógica desejada.

Overhead: custo adicional de tempo computacional, memória, largura de banda ou outro recurso necessário para atingir certo objetivo.

Validação: objetiva certificar se a transformação entrada-saída realizada pelo modelo tem precisão para representar a mesma ocorrência procedida no sistema real (ou de outra simulação de maior confiança).

Verificação: consiste em um conjunto de ações cuja meta é certificar se o modelo conceitual foi transcrito de forma adequada quando do uso de linguagem de programação.

Testbed: é uma plataforma de *hardware* para execução de experimentos, cuja as propriedades em estudo não compreendem todas as características e variáveis apresentadas no *hardware* final do dispositivo.

SUMÁRIO

LISTA DE ILUSTRAÇÕES.....	VII
LISTA DE QUADROS	XI
LISTA DE TABELAS.....	XIII
LISTA DE ABREVIATURAS E SIGLAS	XV
TERMINOLOGIA E DEFINIÇÕES ADOTADAS	XVII
1 INTRODUÇÃO.....	1
1.1 TRABALHOS RELACIONADOS	3
1.2 DEFINIÇÃO DO PROBLEMA E OBJETIVOS	7
1.3 METODOLOGIA E VALIDAÇÃO	9
1.4 ESTRUTURA DO TEXTO	11
2 CONCEITOS PRELIMINARES	13
2.1 COMUNICAÇÃO E CONTROLE DE TOPOLOGIA.....	14
2.1.1 Disseminação de Informações Falsas	17
2.2 TAREFAS COOPERATIVAS.....	18
2.2.1 O Problema de <i>Rendez-vous</i>	19
2.3 CONSENSO PARA RENDEZ-VOUS	20
2.3.1 <i>Rendez-vous</i> por Consenso Médio.....	21
2.3.2 <i>Rendez-vous</i> por Consenso Circuncentro.....	21
2.3.3 <i>Rendez-vous</i> por Consenso e RHC.....	23
2.3.4 <i>Rendez-vous</i> por Consenso e Filtro de Kalman	24
2.4 REDES DE COMUNICAÇÃO	25
2.4.1 Protocolos de Acesso ao Meio	28
2.4.2 Protocolos Adicionais de Rede.....	29
2.5 CONSIDERAÇÕES FINAIS	30
3 TÉCNICAS E METODOLOGIA.....	31
3.1 CONSTRUÇÃO DO MÉTODO IRONS	32
3.2 TÉCNICAS UTILIZADAS.....	38
3.2.1 Transição entre o Planejamento e Implementação	38

3.2.2	Processo de Validação	40
3.2.3	Análise Estatística.....	43
3.2.4	Planejamento Experimental	46
3.2.5	Análise dos Resultados	48
3.3	CONSIDERAÇÕES FINAIS.....	49
4	MÉTODO IRONS	51
4.1	PLANEJAMENTO DO ESTUDO.....	52
4.2	PLANEJAMENTO DA SIMULAÇÃO.....	57
4.2.1	Modelo Conceitual.....	57
4.2.2	Modelo Comunicativo	59
4.3	IMPLEMENTAÇÃO DA SP.....	62
4.3.1	Métricas para Avaliação	65
4.4	VALIDAÇÃO DA SP.....	67
4.4.1	Exemplo – Processo de Validação	70
4.5	ESTUDOS DE CASO.....	74
4.5.1	Exemplo – Planejamento Experimental	75
4.6	ANÁLISE DOS RESULTADOS.....	77
4.6.1	Exemplo – Análise dos Resultados	79
4.7	CONSIDERAÇÕES FINAIS.....	81
5	FRAMEWORK DE SIMULAÇÃO IRONS	83
5.1	SOFTWARE PARA SIMULAÇÃO.....	83
5.2	IMPLEMENTAÇÃO DOS MÓDULOS	87
5.3	FUNÇÕES ADICIONAIS	90
5.4	SOFTWARES AUXILIARES	90
5.5	CONSIDERAÇÕES FINAIS.....	91
6	UTILIZAÇÃO E RESULTADOS.....	93
6.1	RESULTADOS PRÉVIOS	96
6.1.1	Apresentação e Resultados.....	96
6.1.2	Contribuições para o Método IRONS.....	98
6.2	EC – PASSO A PASSO.....	99
6.2.1	Formulação do Problema.....	99

6.2.2 Escolha de Soluções	100
6.2.3 Especificação do Sistema	100
6.2.4 SP – Planejamento e Implementação	101
6.2.5 SP – Validação.....	102
6.2.6 EC – Planejamento Experimental	103
6.2.7 EC – Experimentação e Análise dos Resultados	104
6.2.8 EC – Conclusão	108
6.3 EXEMPLOS DE ESTUDOS DE CASO.....	108
6.3.1 Avaliação da Conectividade	108
6.3.2 Robôs Heterogêneos.....	110
6.3.3 Escalabilidade.....	111
6.3.4 Comunicação Multi-Hop	114
6.4 TRABALHOS CONJUNTOS.....	118
6.4.1 Comparação de Técnicas – Consenso FK	118
6.4.2 Propagação da Informação – Consenso Ponderado	122
6.5 DIVULGAÇÃO EXTERNA.....	127
6.6 CONSIDERAÇÕES FINAIS	128
7 CONCLUSÃO.....	129
7.1 CONTRIBUIÇÕES DA TESE.....	130
7.2 DIFICULDADES E PERSPECTIVAS FUTURAS.....	131
7.3 INFORMAÇÕES REFERENTES AO DOUTORADO	132
7.3.1 Doutorado Sanduíche	132
7.3.2 Lista de Publicações.....	133
REFERÊNCIAS	135
ANEXO A – REVISÃO SISTEMATIZADA	149
ANEXO B – EXEMPLO DE MODELO COMUNICATIVO	155
ANEXO C– MATERIAL ADICIONAL	161

Capítulo 1

Introdução

Para algumas tarefas robóticas, especialmente aquelas que são intrinsecamente distribuídas, como a de exploração e de busca/resgate, é perceptível as vantagens de um sistema multi-robôs em relação a um único robô (CHEN, 2015; REYES, 2015; DUCHON, 2016), possibilitando obter uma melhor distribuição de tempo e espaço (ROCHA, 2006a), a quebra de problemas complexos em problemas menores (ROCHA, 2006a), a melhoria no grau de confiança (DORIYA, 2015), a redução do custo (PARKER, 1998; DORIYA, 2015) e escalabilidade (SHINNOH, 2015). Além disso, em tarefas complexas, pode ser necessária uma quantidade considerável de recursos distintos (diferentes tipos de sensores e atuadores, por exemplo), sendo que os robôs cooperativos permitem uma maior flexibilidade por meio da distribuição do risco e da redundância, sobretudo na possibilidade de que, ao ocorrer uma falha ou dano a um robô, a missão ainda pode ser concluída (em vez de falhar completamente), embora com uma possível degradação do desempenho em relação à sua operação normal (RAC, 2009). Em outras palavras, o conjunto de robôs torna-se mais robusto e confiável (ROCHA, 2006b; QIN, 2014; CHEN, 2015; SHINNOH, 2015).

Ao utilizar um grupo de robôs espera-se que o mesmo assuma um comportamento coordenado, sendo necessário empregar estratégias de controle cooperativo que dependem diretamente do tipo de robô utilizado, a tarefa a ser realizada e o sistema de comunicação empregado na cooperação (ROCHA, 2002).

A comunicação possui um papel fundamental na cooperação entre os robôs, sendo necessária a presença de uma rede de comunicação, ou seja, uma infraestrutura para dar suporte à comunicação entre os robôs. É

desejável que a mesma seja sem fio (*wireless*), eficiente, com baixa latência e sem obstruções (OLIVEIRA, 2012). No entanto, quando se trata de uma rede sem fio, estes requisitos não podem ser garantidos, pois os dados em redes sem fio estão sujeitos a erros, largura de banda limitada e atrasos de propagação (BAER, 2008). Devido a isto, alguns trabalhos preferem idealizar a comunicação e focar apenas na estratégia de controle e na troca de informações entre os robôs (ZHANG, 2007; OLFATI-SABER, 2006; BRANDÃO, 2013), contudo, o estudo torna-se menos realista e os resultados podem apresentar disparidades ao ser implementados em robôs reais (MANFREDI, 2013; KUDELSKI, 2013; RAMOS, 2015b).

Além da estrutura de comunicação, as possíveis combinações de conectividade entre os membros do grupo de robôs, ou seja, a topologia de comunicação, pode afetar o desempenho do sistema cooperativo, especialmente quando um controle intencional sobre a topologia é utilizado. Estes aspectos devem ser cuidadosamente analisados tanto na visão da rede de comunicação quanto no aspecto da estratégia cooperativa (DAS, 2007; MONGA, 2009; MI, 2010; MEHTA, 2013; KHAN, 2014; NEVES, 2014).

Existem três principais abordagens para realizar este tipo de estudo: elaborar uma simulação detalhada com os conceitos envolvidos, utilizar um *testbed* com hardware de comunicação e/ou aplicar em robôs reais (JIMÉNEZ-GONZÁLEZ, 2013; PICKEM, 2015 e 2016). Idealmente, estas três abordagens não são mutualmente exclusivas, sendo utilizadas sequencialmente durante o ciclo de desenvolvimento. No entanto, por diversos motivos, algumas etapas acabam sendo ignoradas ou realizadas parcialmente. Dentre os motivos, pode-se destacar o tempo limitado de projeto, excessiva autoconfiança do projetista, indisponibilidade temporária dos dispositivos físicos, alto custo de obter robôs reais ou ainda, o desgaste referente ao retrabalho necessário para implementar cada etapa em tecnologias distintas (e.g.: utilização de diferentes linguagens de programação para cada etapa).

Estes problemas são potencializados quando o grau de realismo entre as três etapas é muito distante (e.g.: simulação com muitas considerações ideais), acarretando na detecção de problemas de projeto inerentes a estas considerações somente na fase de *testbed* ou na fase de implementação. Além disso, a detecção da origem do erro pode se tornar mais complexa se vários problemas de fontes distintas ocorrem simultaneamente.

Nesse contexto, quanto mais propriedades puderem ser testadas ainda na fase de simulação, onde há maior controle sobre o que será

testado, maior será a chance de possíveis problemas e/ou comportamentos serem detectados e corrigidos antes de uma implementação, ocasionando menos tempo com retrabalhos e menor custo de implementações em *hardware*.

Devido à indisponibilidade de um *testbed* e de robôs reais, à necessidade de eliminar eventos não controlados (e.g.: temperatura ambiente, vento, interferências, entre outros) e de eliminar erros de implementação, é desejada a adoção de abordagens que melhoram o realismo da simulação, facilitando a transição entre as etapas de desenvolvimento.

Para uma simulação integrada de robôs cooperativos, rede de comunicação e de controle de topologia (a qual é denominada apenas de *simulação integrada* ao longo deste trabalho), é necessário que a mesma seja capaz de lidar tanto com aspectos físicos dos robôs (movimento, sensoriamento, estratégia de cooperação e demais subsistemas) assim como com os aspectos da rede de comunicação (estrutura, protocolos, atrasos, perdas) e de controle de topologia (criar/destruir links de comunicação, topologia lógica) (LORA, 2015).

No entanto, a elaboração de uma simulação com este grau de complexidade não é um processo trivial, e seguir um método para organizar, estruturar e otimizar simulações é uma prática comum em algumas linhas de pesquisas, como por exemplo, na modelagem e simulação de sistemas (ROBINSON, 2006; LAW, 2008), para projeto de sistemas *Cyber-Physical* (CPS) (JENSEN, 2011; DELGOSHAEI, 2014), para simulações de dispositivos em redes (KRISTIANSEN, 2015; LORA, 2015; LIM, 2012) e para modelagem e simulação de agentes (ABMS)(RICHIARDI, 2006; JUNGES, 2013; SIEGFRIED, 2014; MACAL, 2014).

1.1 TRABALHOS RELACIONADOS

Em relação ao *framework* de simulação integrada, as abordagens encontradas na literatura consistem em: ou optar pela co-simulação de dois ou mais simuladores diferentes (YE, 2001; ZHANG, 2010; KUDELSKI, 2013); ou expandir as capacidades de um simulador de rede para lidar com os robôs cooperativos (LI, 2014; RAMOS, 2015a); ou expandir um simulador físico para lidar com a rede de comunicação (NEVES, 2014; GIL, 2015); ou desenvolver um novo simulador com ambas características (PIMENTEL, 2003b; LORA, 2015).

A prática da co-simulação é a abordagem mais comum e consiste em utilizar simuladores de redes de comunicação combinados com simuladores de robôs ou de eventos discretos (LORA, 2015). Pohjola (2008), por exemplo, realizou a integração entre NS2 (Network Simulator 2) (NS2, 2016) e Simulink (MATHWORKS, 2016) para simulação do controle de robôs móveis *wireless*. Kudelski (2013) elaborou um *framework* integrado denominado RoboNetSIM, resultado da integração entre o simulador de robôs ARGOS (PINCIROLI, 2012) e NS2/NS3 (NS3, 2016) aplicado à robótica cooperativa. Os autores demonstram o impacto da rede de comunicação em duas aplicações simplificadas da robótica cooperativa: o impacto de atrasos em uma aplicação de mestre e escravo; e impacto dos atrasos no tempo para a conclusão de pequenas tarefas de busca.

No entanto, há três grandes problemas na co-simulação: 1) o problema de sincronização entre diferentes simuladores (COSTA, 2015); 2) existe um *overhead* (excesso de tempo, memória, largura de banda ou outro recurso) necessário para efetivar a comunicação de diferentes ferramentas (LORA, 2015); e 3) a possibilidade da introdução de erros adicionais de projeto devido à complexidade em definir formalmente o comportamento entre diferentes domínios (LORA, 2015).

Neste sentido, alguns autores buscam abordagens alternativas para evitar estes problemas, como por exemplo, expandir as capacidades de um simulador para lidar com mais de um domínio de simulação (AL-HAMMOURI, 2008). Um exemplo é apresentado em Neves (2014), onde o MATLAB foi expandido para lidar com uma rede de comunicação com um controle de topologia lógica, aplicado ao estudo do fluxo de informações entre nós e a eficiência energética. O MATLAB também foi utilizado em Gil (2015) e em Sabattini (2016), expandindo-o para lidar com alcance limitado da comunicação e controle de topologia de robôs móveis. Por outro lado, em Pekkarinen (2014), foram expandidas as características do simulador de eventos discretos e redes de comunicação, o OMNeT++ (OMNET++, 2016), para lidar com redes de sensores sem fio (WSN) aplicado à tarefa de vigilância, visando analisar a eficiência energética.

Em relação ao método para elaborar simulações, de forma geral, é um tópico bem estabelecido na área de modelagem e simulação (M&S) de sistemas (GORE, 2013; BALCI, 1998; LAW, 2015), onde define-se um modelo conceitual inicial construído por *experts* após uma cuidadosa avaliação do problema, seguido pela implementação do código, para que, finalmente, o resultado da simulação para um conjunto de casos seja

validado utilizando dados históricos ou fontes confiáveis (ROBINSON 2006; GORE, 2013).

Mesmo sendo uma área bem estabelecida, erros de projeto ainda são comuns, como por exemplo, quando um projeto foi realizado com um modelo conceitual aparentemente completo e robusto, mas deficiente de informações, mostrando-se incoerente, incompleto ou potencialmente inválido durante a implementação na simulação (GORE, 2013). Isto pode ocorrer, por exemplo, devido à exatidão requerida pela ferramenta de simulação, criando-se uma elevada complexidade da simulação que não permite avaliar as propriedades desejadas, já que o desenvolvedor pode não saber identificar quais as modificações de parâmetros são necessárias (LAW, 2008). Uma abordagem é utilizar métodos formais de modelagem para relacionar o modelo conceitual com a implementação, mas que na prática, raramente são utilizados devido à sua sintaxe intimidante e pesada semântica matemática (GORE, 2013).

Outros autores defendem métodos menos formais, mas com acompanhamento de guias e boas práticas de simulação. Robinson (2006) por exemplo, relata boas práticas de simulação direcionadas à modelagem conceitual. Em Smith (2007) é descrito um guia de boas práticas para projeto baseado em modelo, incluindo a etapa de concepção, simulação, implementação, testes e verificação.

Contudo, os trabalhos citados não abordam as particularidades das aplicações e, por serem vagos, se tornam inadequados, incompletos ou inaplicáveis. Para preencher esta lacuna, surgiram na literatura novos trabalhos direcionados para aplicações específicas, apresentando técnicas para alguns nichos de simulação, como por exemplo, em modelagem e simulação de sistemas multi-agentes (ABMS) (KLUGL, 2008) (GRIMM, 2010) (SIEGFRIED, 2014). A ABMS recebeu mais atenção quando pesquisadores notaram uma falta de base formal e procedimental nos estudos simulados desta área (KLUGL, 2008) (SIEGFRIED, 2014) e que não havia propostas de método ou técnicas para suprir esta demanda.

Em Richiardi (2006) é discutida uma das primeiras recomendações para ABMS, sendo denominado de “protocolo para ABMS”. O autor expõe alguns pontos importantes da modelagem, estruturando-a em duas etapas: análise (subdividido em exploração, equilíbrio, investigação, calibração, análise de sensibilidade e validação) e replicabilidade. A divisão da modelagem também é proposta em Bernon (2012), onde é apresentada uma representação por um diagrama de fluxo de trabalho (*workflow*) dividido em três partes: requisitos, análise e projeto. O trabalho de Garro (2010) propõe um método com características similares

aos propostos em M&S de Balci (1998) e Law (2015), mas com foco no modelo conceitual e representação em meta-modelos da ABMS. Nesta mesma direção, Ghorbani (2013) propôs a modelagem MAIA (*Modelling Agent Systems Based on Institutional Analysis*) direcionada para ABMS sociais e que tem como base um *framework* conceitual baseado em meta-modelo.

Outros trabalhos focam em apenas um dos aspectos de um estudo simulado de ABMS, como por exemplo, especificando técnicas de validação (KLUGL, 2008), propondo um processo formal para modelagem (SIEGFRIED, 2014) ou ainda, propondo um processo somente para a documentação (GRIMM, 2013).

Estes trabalhos são uma excelente base para elaboração de um estudo simulado de robótica cooperativa. No entanto, como se deseja estudar mais de uma área de pesquisa simultaneamente (adição da rede de comunicação e a topologia de comunicação), é necessária uma abordagem mais completa para fornecer ferramentas adequadas ao desenvolvedor.

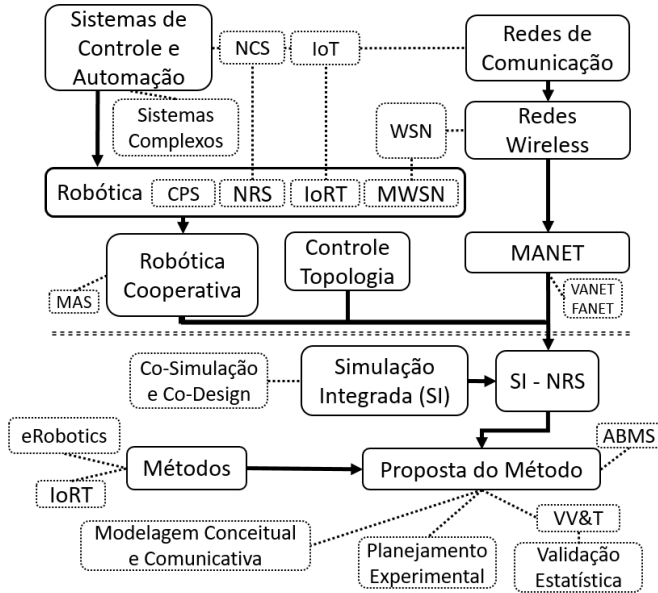
Os trabalhos realizados em *eRobotics* (KADRY, 2015), um ramo de *eSystem Engineering*, são exemplos da contribuição que um método bem elaborado pode ter em um estudo simulado. A *eRobotics* é um tópico de pesquisa que foca no desenvolvimento de métodos e conceitos, refinando processo de engenharia e utilizando técnicas semânticas de modelagem, visando fornecer os modelos de diversas áreas do conhecimento, visando uma precisa simulação física 3D de um robô.

O método proposto em Kadry (2015) acompanha um *Testbed Virtual* (*framework* de simulação), utilizando software de simulação 3D para elaborar uma complexa simulação integrada de múltiplas áreas. Já em Schuluse (2016), os autores combinam o conceito de *Testbed Virtual* com outra abordagem provinda da indústria, a noção de *Digital Twins*. Esta consiste em utilizar objetos do mundo real em conjunto com uma respectiva representação virtual, capaz de se comunicar e agir como um nó inteligente. Os autores utilizam o termo *Experimentable Digital Twins* para esta integração, utilizando-o como núcleo do processo de desenvolvimento da simulação, possibilitando simulações detalhadas em nível de sistema e a realização de sistema inteligentes, focado na modelagem 3D de aspectos físicos do robô.

Como o escopo desta tese é a robótica cooperativa, a modelagem física do robô tem menor significância nos resultados da tarefa cooperativa, onde a comunicação entre os robôs passa a ser o objeto de interesse (WEI, 2014). Contudo, não há na literatura propostas de métodos direcionados especificamente para a simulação integrada de robôs cooperativos e redes de comunicação.

Os tópicos relacionados são resumidamente listados no diagrama da Figura 1 e são parte da revisão sistemática elaborada e indicada no ANEXO A.

Figura 1- Diagrama indicando as áreas relacionadas ao trabalho.



1.2 DEFINIÇÃO DO PROBLEMA E OBJETIVOS

Motivado pela carência de um *framework* de simulação integrada de robótica cooperativa e de redes de comunicação, e de um método procedimental, objetivo e eficaz para elaborar e avaliar esse estudo simulado integrado, esta tese procura responder a seguinte pergunta de pesquisa:

Problema: *Como elaborar de forma procedural, modular e coerente, um estudo simulado integrado de robótica cooperativa, redes de comunicação e controle de topologia, onde seja possível determinar e analisar, de forma estatisticamente confiável, as possíveis interações entre técnicas destas três áreas?*

A proposta desta tese para solucionar este problema é a construção de um *framework* de simulação, aliado a um método sistemático para

elaboração de simulações, que permita construir passo a passo a simulação, de forma documentada, ordenada e modular, permitindo proporcionar respostas aos questionamentos relacionados aos três tópicos em questão (robótica cooperativa, redes de comunicação e controle de topologia). Esta solução consiste em utilizar um *framework* construído a partir de uma sequência de passos, onde esta sequência tem como objetivo guiar o desenvolvedor para uma sólida implementação, validação e experimentação da simulação integrada. Assim, o desenvolvedor poderá ser capaz de obter respostas de questionamentos em relação a interação destes tópicos ainda na fase de simulação. Portanto, o principal objetivo desta tese pode ser sucintamente descrito como:

Objetivo 1. *Desenvolver um framework aliado a um método para análise, validação e experimentação de simulações integradas de robótica cooperativa, redes de comunicação e da topologia de comunicação, cuja as ferramentas permitam responder questionamentos pertinentes à interoperabilidade destes tópicos.*

Para concluir o objetivo principal da tese, foi necessário abordar alguns objetivos específicos relacionados a etapas de desenvolvimento do método. O primeiro objetivo específico (Objetivo Específico 1.1) é criar a estrutura do método, utilizando os trabalhos disponíveis na literatura como fundamentação. Esta estrutura deverá estar em um formato de fácil interpretação e que guie o desenvolver gradativamente até atingir o seu objetivo. O desafio deste objetivo é abstrair os conceitos utilizados em outros métodos e mesclar em um único método que considere as particularidades da simulação integrada.

Objetivo Específico 1.1. *Elaborar a estrutura básica do método e do framework de simulação, abstraindo conceitos da literatura e adaptando-os para simulação de robótica cooperativa e de redes de comunicação.*

O segundo objetivo específico (Objetivo Específico 1.2) é identificar e especificar os passos críticos do método, utilizando, se necessário, técnicas conhecidas em outras áreas do conhecimento, adaptando-as para aplicação de robótica cooperativa e redes de comunicação.

Objetivo Específico 1.2. *Especificar técnicas adequadas de planejamento, validação e análise da simulação para o método proposto.*

O segundo objetivo geral deste trabalho é exemplificar a utilização do método, por meio da elaboração do *framework* de simulação, indicando quais os questionamentos que podem ser respondidos e os

resultados obtidos nesta plataforma para um conjunto de técnicas.

Objetivo 2. *Realizar estudos simulados guiados pelo método proposto no framework de simulação elaborado, indicando possíveis questionamentos e demonstrando como obter as respostas para os mesmos.*

Ao atingir estes objetivos, é esperado que o estudo tenha atendido o seu propósito inicial, auxiliando o desenvolvedor a elaborar, validar e avaliar uma simulação integrada, a identificar parâmetros importantes e principalmente, a ter estas respostas ainda na fase de simulação.

1.3 METODOLOGIA E VALIDAÇÃO

Buscou-se inspirações em trabalhos com temáticas similares, como os trabalhos de modelagem e simulação (M&S) de sistemas (ROBINSON, 2006; LAW, 2008; LAW, 2015), M&S para projeto de CPS (JENSEN, 2011; DELGOSHAEI, 2014), M&S de simulações de dispositivos em redes (KRISTIANSEN, 2015; LORA, 2015), de métodos para VV&T (Verificação, Validação e Teste) (BALCI, 1995; SARGENT, 2013) e em métodos para simulações e modelagens baseadas em agentes (ABMS) (GRIMM, 2006 e 2010; RICHIARDI, 2006; KLUGL, 2012; SIEGFRIED, 2014; MACAL, 2014).

Em especial, focou-se os trabalhos direcionados para modelagem e simulação de sistemas baseados em agentes (ABMS) e de sistemas multi-agentes (MAS), nos quais possuem fundamentos metodológicos sólidos (RICHIARDI, 2006) e são os mais próximos da aplicação de robótica cooperativa e, portanto, apresentam mais similaridades ao tema deste trabalho. Utilizou-se trabalhos recentes que almejam organizar um estudo simulado de MAS (JUNGES, 2013; SIEGFRIED, 2014; MACAL, 2014) ou que propõem algum aspecto relacionado ao planejamento, modelagem e simulação (GARRO, 2010; GOMEZ-SANZ, 2010; KUBERA, 2011; GHORBANI, 2013; HESTER, 2012).

Como existem inúmeras tarefas, tipos de robôs e técnicas para estes tópicos, as possíveis combinações tornam inviável provar, desenvolver, aplicar e validar um método para todas as combinações de técnicas e aplicações existentes. Portanto, foi determinado um escopo inicial para o mesmo, consistindo de robôs cooperativos terrestres com rodas, com trocas de mensagens, operando de forma descentralizada em uma rede Ad-Hoc sem fio móvel (MANET), executando a tarefa de *rendez-vous*.

É importante ressaltar que o método proposto não depende das

técnicas, algoritmos e/ou dos parâmetros utilizados nos exemplos. Contudo, para exemplificar a sua utilização, foi necessário utilizar um conjunto de técnicas. Os conceitos relacionados às técnicas, são brevemente apresentados no Capítulo 2 e no Capítulo 5, porém sem compromisso do autor em demonstrar que uma técnica é superior a outra.

Em relação à validação da proposta, como não há outros métodos neste escopo, não é possível realizar uma comparação direta para determinar se esta é a melhor forma para resolver o problema.

No entanto, utilizando conceitos de Engenharia de Software, uma pesquisa empírica pode contornar este problema ao analisar e mitigar as possíveis ameaças para a validação do software (FELDT, 2010; RUNESON, 2008), ou neste caso, do método. Há dois pontos a serem questionados quanto à validação: os resultados obtidos na simulação e a utilidade do método propriamente dito.

O primeiro ponto é abordado pelo próprio método, onde um correto planejamento da simulação minimiza problemas relacionados à validação a respeito da correta construção da simulação, e a utilização do planejamento experimental e da análise estatística contribuem para a validação das conclusões sobre os resultados.

Quanto ao método, na validação interna há o questionamento se realmente novos comportamentos podem ser observados e analisados utilizando este tipo de simulação integrada e o método. Este problema foi minimizado ao demonstrar, utilizando um conjunto de técnicas, a detecção de novos comportamentos resultantes da integração e operação simultânea das técnicas, comportamentos dos quais que não estavam presentes quando as técnicas foram isoladamente implementadas.

Também há um risco de que as técnicas incorporadas ao método não serem as mais adequadas a este tipo de simulação, sendo este risco minimizado ao buscar referências de áreas similares ao de robótica cooperativa.

Outro risco para a validação do método está relacionado à validação da conclusão e da confirmabilidade, pois há o risco de que o método somente favoreça o autor ou o conjunto de técnicas utilizadas para demonstração. Este risco é mitigado ao propor utilizar o método em trabalhos em conjuntos com outros membros do grupo de pesquisa, demonstrando que o método também contribuiu para outros trabalhos utilizando outros conjuntos de técnicas. Outra contramedida foi disponibilizar o método online para uso por outros desenvolvedores, sendo possível analisar em trabalhos futuros a sua utilização e o impacto gerado na comunidade científica.

1.4 ESTRUTURA DO TEXTO

Este trabalho foi estruturado da seguinte forma:

- Capítulo 2: são apresentados conceitos preliminares sobre as técnicas utilizadas para exemplificar o método. É importante lembrar que o método não depende de técnicas específicas de robótica ou de redes, e este capítulo tem apenas o propósito de fornecer um *background* para o leitor, para melhor compreensão dos resultados obtidos com os algoritmos selecionados.
- Capítulo 3: são apresentados os materiais e métodos relacionados ao método. Abrange os conceitos que o fundamentam, as técnicas incorporadas ao mesmo (modelagem, VV&T, design experimental, avaliação) e o *framework* de simulação.
- Capítulo 4: é descrito o método em detalhes e seguindo passo a passo todo o ciclo de desenvolvimento, detalhando suas características e exemplificando cada etapa.
- Capítulo 5: é descrita a implementação do *framework* utilizando o método.
- Capítulo 6: é apresentada a utilização do método aliado ao *framework*, para responder questionamentos e indicar os respectivos resultados obtidos;
- E por fim, no Capítulo 7 é apresentada a conclusão da tese, com as contribuições e propostas de continuidade do trabalho.

Capítulo 2

Conceitos Preliminares

Neste capítulo são apresentados conceitos básicos relacionados aos algoritmos da simulação integrada (estratégia de controle cooperativo, redes de comunicação e controle de topologia), visando contextualizar o leitor sobre o tema para uma melhor compreensão dos resultados obtidos.

É importante esclarecer que o método não depende de nenhuma técnica específica de algum destes três tópicos e que, inclusive, é possível o utilizar para fazer diversas comparações entre diferentes técnicas. No entanto, para exemplificar seu uso, foi necessário escolher um conjunto de técnicas e delimitar um escopo inicial de atuação.

Dada a multidisciplinaridade do trabalho e da complexidade das técnicas utilizadas nos exemplos, este capítulo foi elaborado para explanar exclusivamente estes pontos, detalhando os conceitos fundamentais envolvidos (rede de comunicação, robótica, controle cooperativo e controle de topologia) e o princípio básico das técnicas utilizadas. A discussão sobre pontos relevantes à proposta do método inicia-se no Capítulo 3.

Partindo de uma definição básica, neste trabalho, o termo *robô* é utilizado e deve ser interpretado como um dispositivo móvel com rodas, capaz de realizar de forma autônoma sua locomoção e medição de variáveis do ambiente. Por outro lado, o termo *robô cooperativo*, é um robô inserido em um sistema de múltiplos robôs, cujo foco deste sistema está na troca de informações, compartilhamento de recursos e coordenação de esforços de trabalho para cumprimento de um determinado objetivo ou tarefa.

Para atingir este objetivo, é necessária uma estratégia de controle cooperativo, ou seja, uma abordagem de controle que possui o objetivo de forçar os robôs a terem um certo comportamento desejado, visando cumprir uma determinada tarefa. Esta estratégia pode estar centralizada

em um robô / terminal de trabalho, ou ser descentralizada, onde o algoritmo é implementado em cada robô, tornando-o capaz de tomar sua própria decisão. Neste trabalho são tratados apenas casos descentralizados.

O tipo de comunicação entre os robôs é fundamental para a cooperação entre os mesmos. Foi definido como escopo inicial, trabalhar apenas com robôs capazes de ter uma comunicação explícita, ou seja, que são capazes de trocar informações diretamente entre si por meio de mensagens. Esta comunicação é sustentada por uma rede de comunicação sem fio. A definição de quais robôs se comunicam entre si é dada pela topologia de comunicação, que pode depender apenas do alcance do dispositivo de comunicação ou ser resultado de um controle de topologia.

As técnicas utilizadas para exemplificar a utilização do método, são listadas abaixo e descritas ao longo do capítulo.

- **Tarefas Cooperativas:** *rendez-vous* médio e de referência;
- **Estratégia de Controle Cooperativo para *rendez-vous*:** consenso médio, consenso por circuncentro, consenso RHC, consenso FK;
- **Rede de Comunicação:** MANET, protocolo UDP/IP, tecnologia Wi-Fi;
- **Protocolos de Rede Adicionais:** RA-TDMA e TDMA;
- **Controle de Topologia:** fixo e dinâmico;

2.1 COMUNICAÇÃO E CONTROLE DE TOPOLOGIA

A comunicação entre os robôs é representada por conceitos da teoria de grafos, onde os robôs são representados por nós e os *links* de comunicação por arestas. Portanto, obtém-se um grafo orientado $G = (V, A, w)$, onde V é o conjunto de nós (robôs), A é o conjunto de links entre os nós (comunicação) e w é um peso dado a cada link.

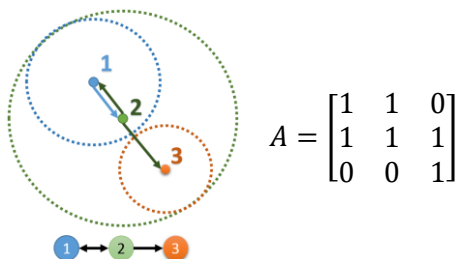
O conjunto A e w podem ser representados juntos na forma de uma matriz (matriz adjacência) de dimensão $n_r \times n_r$ relacionada ao número de total robôs n_r , onde os elementos não nulos $a_{ij} = w_{ij}$ da matriz indicam a presença do link de comunicação entre os robôs i e j , e o respectivo peso atribuído a cada um, como indicado em (2.1).

$$A = \begin{bmatrix} a_{11} & \dots & a_{1n_r} \\ \vdots & \ddots & a_{ij} \\ a_{n_r1} & a_{ij} & a_{n_r n_r} \end{bmatrix} \quad (2.1)$$

Usualmente, a determinação se há ou não presença da comunicação é dada pelo alcance físico do dispositivo utilizado para

comunicação. Na Figura 2 é apresentado um exemplo da representação por grafos e pela matriz adjacência da comunicação entre robôs, baseado somente no alcance da comunicação, onde a linha corresponde a comunicação que sai do robô e a coluna a comunicação que o robô recebe.

Figura 2 - Exemplo de grafos para representação da comunicação dos robôs e a respectiva matriz adjacência com peso unitário.



Nota-se que os robôs 1 e 2 possuem comunicação bidirecional por estarem contidos no raio de comunicação um do outro. Há uma comunicação direcional entre os robôs 2 e 3, visto que o robô 2 não está contido no raio de comunicação de 3, mas 3 está contido no raio de 2. A representação por grafo é indicada na parte inferior da figura e a respectiva matriz adjacência à direita.

A diagonal principal, comunicação do robô consigo mesmo, geralmente é adotada como nula ou com valor de 2 quando na presença de um loop interno. No caso deste trabalho, a diagonal principal é adotada com valor de 1 para facilitar a integração com a formulação do consenso, indicando que a própria informação do robô também é utilizada.

Como são considerados raios de comunicação idênticos para os robôs deste trabalho, os grafos (comunicação) serão sempre bidirecionais se os robôs não apresentarem nenhuma falha na comunicação, resultando em uma matriz adjacência simétrica.

Nas estratégias de controle apresentadas ao longo deste trabalho, são utilizados os elementos a_{ij} da matriz A em conjunto com as informações recebidas dos robôs vizinhos para determinar as ações que deverão ser realizadas pelo robô.

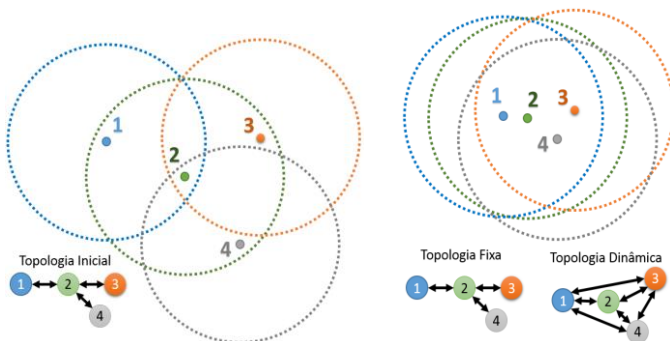
Ao se tratar de robôs com comunicação sem fio com alcance limitado e que estão constantemente se locomovendo durante uma tarefa, a topologia de comunicação é naturalmente variável. No entanto, é possível criar algoritmos para utilizar outras fontes de informações ao definir uma topologia de comunicação, podendo assim, assumir certas características desejadas.

Esta alteração na forma de se comunicar é denominada de controle de topologia (CT) e leva em consideração outros aspectos além da limitação do alcance físico da comunicação, como por exemplo, um possível roteamento de dados entre os robôs ou simplesmente a capacidade de ignorar mensagens de certos robôs. Quando existe um controle de topologia, é utilizado os termos *topologia lógica* para denominar a topologia que de fato governa os robôs e de *topologia física*, a topologia que considera apenas o alcance da comunicação.

O CT é um assunto relativamente recente na robótica, mas tem sido foco de um número crescente de estudos, principalmente direcionados para manutenção de conectividade e minimização de links (BURKHART, 2004). Pesquisas de Li (2004), Ahmadi (2006) e Wagenpfeil (2009) abordam a manutenção da conectividade, utilizando ferramentas de teoria de grafos e de redes. Quanto à minimização da rede, os trabalhos de Casteigts (2010), Abuhaiba (2011), Halldórsson (2008) apresentam soluções para redução do consumo de energia e de interferências. Por outro lado, os trabalhos de Sabattini 2015 e 2016 focam no impacto da topologia em tarefas de robôs cooperativos descentralizados, propondo sinais de controles mistos, capazes de cumprir a tarefa e manter uma conectividade desejada.

Para exemplificar a utilização de um CT no método, foram utilizados inicialmente dois algoritmos: simplificado fixo e simplificado dinâmico. O primeiro algoritmo mantém a mesma topologia de comunicação durante toda a simulação e o segundo representa o comportamento tradicional dos robôs ao permitir a criação de novos links quando os robôs estão no raio de comunicação dos demais.

Figura 3 - Exemplo de topologia inicial (esquerda) e o resultado para os casos de controle de topologia simplificado (direita).



A diferença entre ambos algoritmos é percebida no exemplo indicado na Figura 3, onde quatro robôs iniciam a tarefa com a topologia inicial baseada no alcance da comunicação, como indicado à esquerda. A medida que estes robôs se aproximam, nota-se que na topologia fixa os novos possíveis *links* são desconsiderados, mantendo a topologia inicial. Enquanto que, na topologia dinâmica, estes *links* são considerados, estabelecendo uma topologia lógica igual à topologia física.

Ambos os algoritmos também verificam o instante de tempo em que as informações foram recebidas e caso note que a informação de um *link* esteja obsoleta, o *link* é desativado, ou seja, o elemento da matriz adjacência se torna nulo e a informação relativa a este vizinho é desconsiderada.

2.1.1 Disseminação de Informações Falsas

A conectividade da rede possui uma propriedade essencial para os robôs cooperativos, a disseminação da informação. Esta, por sua vez, é caracterizada pela movimentação da informação entre os robôs e determina a disseminação das informações locais, afetando o estado global do sistema cooperativo.

Como modelar essa disseminação e como incorporar a formulação resultante no controle cooperativo, é um trabalho de doutorado em andamento na UFSC/PGEAS, cujo o objetivo é propor um algoritmo seja capaz de filtrar falsas informações transientes, sem afetar drasticamente o desempenho da tarefa.

A disseminação desta falsa informação (também denominada de rumor ou de vírus) é modelada usando modelos dinâmicos de disseminação epidemiológicas (*Epidemic*). Por meio desta, é possível entender como uma informação se espalha pela rede e como alterar a sua taxa de dispersão ao longo do tempo.

Ao utilizar uma abordagem SIS (Susceptível-Infectado-Susceptível), o sistema é caracterizado pela divisão dos nós da rede (robôs) em Suscetíveis ou Infectados. O grupo suscetível são os nós com potencial de serem infectados pela falsa informação e o segundo, o grupo já infectado, mas que possuem um potencial de serem curados.

Essa dinâmica é regida pela equação diferencial de Bernoulli (2.2), em função do grupo infectado.

$$\zeta_{i,k+1} = \zeta_{i,k} + h \cdot \left(\tau \sum_{j=1}^{n_p} \zeta_{j,k} \bar{\zeta}_{i,k} a_{i,k} - u_i \zeta_{i,k} \right) \quad i = 1, \dots, n \quad (2.2)$$

Onde ζ_i é um estado estocástico (entre 0 e 1) da probabilidade do nó i estar no grupo de infectados, $\bar{\zeta}_i = 1 - \zeta_i$ representa a probabilidade do nó i estar no grupo de suscetível, n é o total de nós, τ é a taxa de transmissão definida como probabilidade de contágio após contato (por unidade de tempo), u_i é a taxa de recuperação dos nós infectados (por unidade de tempo) e a_{ij} é o elemento da matriz adjacência entre os nós i e j . As variáveis restantes são o instante de tempo k , o passo de integração h e índice j sendo os índices dos vizinhos diretos de i , para $j = 1, \dots, n_v$.

No contexto da robótica cooperativa, a falsa informação pode consistir de uma referência falsa intencional ou resultante de erros transientes do próprio robô.

2.2 TAREFAS COOPERATIVAS

As tarefas robóticas cooperativas ou problemas robóticos cooperativos, são bem variados e dependem diretamente da finalidade dos robôs. De modo geral, considerando robôs móveis terrestres com roda, as tarefas podem ser classificadas (ORDOÑEZ, 2013) como problemas de:

- a) **Rendez-vous:** consiste no problema de projetar estratégias locais e individuais, para que todos os integrantes de um determinado grupo se encontrem em uma localização comum (ORDOÑEZ, 2013; KHAN, 2014; FENG 2016; VUNDURTHEY, 2016; SABATTINI, 2016);
- b) **Controle de Formação:** o principal objetivo no controle de formação é manter a posição relativa entre os robôs de um grupo ou então relativa a uma referência (CHEN, 2015; POONAWALA, 2015; CORREIA, 2014b);
- c) **Exploração:** são tarefas que possuem como principal objetivo o mapeamento de um ambiente (COUCEIRO, 2011; SENTHILKUMAR, 2012);
- d) **Cobertura:** é uma tarefa geralmente associada com a exploração, mas que se difere ao tratar em como navegar por uma determinada área segundo algum critério de interesse. Por exemplo, realizar buscas em um mapa tendo como meta uma melhor eficiência dos sensores ou encontrar posicionamento para que o alcance dos sensores abranja a maior área possível (DUCHON, 2016; MOHSENI, 2016);
- e) **Bandos (*flocking*):** são tarefas semelhantes ao controle de formação, mas que também mantem a sincronia da velocidade e aceleração dos robôs (REYES, 2015; SAVKIN, 2016);

- f) **Gerenciamento de recursos:** são tarefas que necessitam tomadas de decisões em conjunto pelos robôs, seja para coordenar uma ação, adquirir um recurso ou para decidir qual robô realizará a ação (KUDELSKI, 2013; JIA, 2013).

Neste trabalho, apesar do método proposto ser capaz de lidar com uma ampla gama de tarefas, optou-se em focar e exemplificar a sua utilização apenas na tarefa de *rendez-vous*. Esta tarefa é ideal para apresentar o assunto pela sua simplicidade, pela rápida conclusão sobre seus resultados e possibilita uma futura extrapolação das análises para as demais tarefas com poucas modificações.

2.2.1 O Problema de *Rendez-vous*

O problema de *rendez-vous* ocorre quando dois ou mais robôs estão espalhados em um ambiente e necessitam retornar a um mesmo ponto comum. A primeira questão em relação ao *rendez-vous* é se o mesmo é ou não solúvel. Segundo Kranakis (2005), há diversas situações onde não é possível realizar o *rendez-vous*, estando condicionado às propriedades dos robôs, forma de comunicação utilizada, topologia de comunicação inicial e, principalmente, o critério adotado para definir se há ou não solução. Este último é particularmente importante, dado que em alguns casos pode ser aceitável haver mais de um ponto de encontro entre os robôs, seja pela configuração inicial das posições ou devida às interferências presentes no ambiente.

Neste trabalho é utilizado como critério de solução a situação na qual todos os robôs convergem para um único ponto, sendo os casos restantes considerados como não-solução, ou em outros termos, um caso divergente.

Todas as condições iniciais geradas ao longo deste trabalho, foram determinadas de forma que sempre resultem em ao menos uma árvore minimamente conectada, ou seja, há ao menos um caminho para a informação percorrer de qualquer robô para outro.

O principal critério para avaliação do *rendez-vous* é a eficiência das soluções propostas, ou seja, quanto tempo os robôs levam para se agrupar em torno de um ponto (KRANAKIS 2005). A definição deste ponto, pode ser feito por definição e repasse de uma referência ou pelo comportamento natural dos robôs devido à posição inicial, aos algoritmos e as conectividades presentes no início da simulação.

a) *Rendez-vous* sem referência

Neste *rendez-vous*, a referência não é pré-estabelecida e sim calculada de acordo utilizando algum critério da técnica escolhida, combinando apenas a própria informação com a informação dos robôs vizinhos (e.g.: média das posições). Neste caso, é possível analisar se todos os robôs chegaram ao ponto de encontro (se convergiram ou divergiram), qual as coordenadas do ponto (ponto de convergência) e o tempo que levaram para alcançá-lo (tempo de convergência).

b) *Rendez-vous* de Referência

No *rendez-vous* de referência, um ou mais robôs possuem uma coordenada de referência pré-determinada ou recebida de forma externa aos demais robôs. Para que a tarefa seja executada com sucesso, todos os robôs deverão convergir para a este ponto, sendo avaliado principalmente o tempo que levaram para chegar ao mesmo (tempo de convergência).

A referência geralmente é tratada de forma direta na estratégia de controle, seja por adição de um robô virtual imóvel ou de forma explícita no cálculo das ações de controle.

Outra possível origem da referência ocorre do resultado de uma falha em um robô, no qual o mesmo ou é incapaz de se locomover ou é incapaz de receber dados e apenas envia mensagens sobre a sua posição para os robôs vizinhos, se tornando o ponto de encontro do grupo.

2.3 CONSENSO PARA RENDEZ-VOUS

O problema do consenso distribuído consiste em alcançar um acordo entre os robôs a respeito de uma única informação de forma descentralizada. Os robôs devem, de alguma forma, se comunicar uns com os outros e entrar em acordo ou convergir ao longo do tempo para um único valor a respeito desta informação.

Quando aplicado à robótica, o consenso está relacionado ao conceito de controle cooperativo, ou seja, a coordenação e controle dos robôs está associada à informação na qual se busca o consenso, definida pelo tipo de tarefa a ser realizada.

Matematicamente, o consenso é obtido quando a diferença entre as informações de todos os robôs tende a zero, como indicado em (2.3).

$$\sum_{i=1}^{n_r} \sum_{j=1}^{n_r} X_i(k) - X_j(k) \rightarrow 0 \quad (2.3)$$

Onde $X_i(k)$ e $X_j(k)$ são as informações do robô i e do robô j , para $i \neq j$ e o total de n_r robôs.

As soluções de consenso foram aplicadas nas mais diversas tarefas robóticas, incluindo *rendez-vous* (LIN, 2003; FERRARI-TRECCATE, 2009; GOWAL, 2012), controle de formação (FAX, 2004; LAWTON, 2003; GRUPTA, 2003; ROY, 2004; CORREIA, 2014), bandos (TANNER, 2003) e no processo de decisão coordenado (DARIO, 2003; ALANYALI, 2004).

O consenso aplicado para *rendez-vous* é obtido ao definir a informação na qual se deseja entrar em consenso como sendo a posição atual dos robôs, ou seja, o *rendez-vous* é obtido quando a diferença entre as posições de todos os robôs tende a zero.

A formulação do consenso para obter o *rendez-vous* é usada para denominar os diferentes algoritmos utilizados neste trabalho, sendo estes o consenso Médio, Circuncentro, RHC e FK. Estes algoritmos são brevemente descritos nos itens a seguir.

2.3.1 *Rendez-vous* por Consenso Médio

O consenso médio para *rendez-vous* é uma estratégia simples, onde o ponto de referência de cada robô é determinado pela média da posição dos robôs vizinhos e sua própria posição.

A cada instante de tempo, cada robô utiliza a informação recebida para navegar em direção ao ponto médio atualizado por meio de (2.4), onde X_{iref} é o vetor posição de referência para robô i , X_j é o vetor das posições recebidas dos vizinhos, k é o instante de tempo, a_{ij} é o elemento da matriz adjacência entre os robôs i e j , sendo j o contador auxiliar dos n robôs e n_i o número total de vizinhos à i .

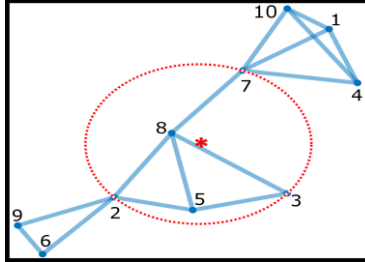
$$X_{iref}(k+1) = \frac{\sum_{j=1}^n X_j(k) \cdot a_{ij}}{n_i} \quad (2.4)$$

2.3.2 *Rendez-vous* por Consenso Circuncentro

Esta técnica foi concebida (YAMASHITA, 1999) para evitar situações de ruptura de *link* de comunicação devido à formação de *clusters* de robôs durante a tarefa, afetando desproporcionalmente os resultados de (2.4). Nesta estratégia, a referência de um robô i é o centro do menor círculo envolvente que inclua todos os seus n_i vizinhos diretos (Figura 4).

Existem diversas propostas para determinação desta solução, focadas especialmente em diminuir o custo computacional. No entanto, o impacto do algoritmo para poucos robôs é mínimo, sendo escolhido o método de força bruta para determinação do menor círculo envolvente.

Figura 4 - Exemplo utilizando menor círculo envolvente para o robô 8.



O algoritmo possui duas etapas, indicadas no Quadro 1 e no Quadro 2. Resumidamente, o primeiro algoritmo determina dentre a posição do robô atual e de seus vizinhos, qual é a maior distância entre os robôs e constrói o menor círculo neste segmento formado. Este círculo será solução se todos os robôs vizinhos estiverem contidos no mesmo, caso contrário, utiliza-se o algoritmo de 3 pontos.

Quadro 1 – Primeira parte do algoritmo de menor círculo envolvente, analisando de dois em dois robôs.

Algoritmo Circuncentro - 2 Pontos

1. Utilizando a posição do robô i e de todos seus n_i vizinhos, determina-se as distâncias entre todos robôs deste conjunto.
2. Obtém-se a maior distância do item 1 e determina-se as coordenadas do centro deste segmento;
3. Com as coordenadas do centro e raio com valor igual à metade do segmento previamente calculado, verifica-se se a distância de todos robôs em relação ao centro é menor que o raio do círculo;
4. Em caso positivo, esta é a solução.
5. Em caso negativo, os dados do maior segmento são armazenados e utiliza-se a determinação por três pontos

Na segunda formulação, utiliza-se a determinação de círculos via triangulação, ou seja, é construído um círculo utilizando três pontos equidistantes do seu centro. Para isto, é necessário calcular ponto central de dois lados do triângulo, traçar as retas ortogonais e determinar o ponto onde estas retas se encontram. Este procedimento é feito até que todos os robôs vizinhos estejam no interior do círculo. Ao final, se houver mais de uma solução, utiliza-se a de menor raio.

Quadro 2 - Segunda parte do algoritmo de menor círculo envolvente, analisando de três em três robôs.

Algoritmo Circuncentro - 3 Pontos	
1.	Utiliza-se o maior segmento entre dois pontos como base de um triângulo e traça-se uma das arestas do triângulo até um dos pontos restantes;
2.	Determina-se o ponto central deste segmento;
3.	Calcula-se a inclinação de ambas arestas;
4.	Calcula-se a inclinação ortogonal de cada uma;
5.	Utiliza-se a equação da reta para cada ortogonal, igualando ambas para calcular onde as mesmas se encontram. Este é o ponto equidistante dos três pontos e centro do círculo, sendo o raio a distância aos pontos;
6.	Deve-se verificar se todos os pontos estão contidos no círculo, ou seja, se a distância do centro do círculo em relação aos robôs restantes é menor do que o raio do círculo;
7.	Se não estiverem contidos, o ciclo de 3 pontos é refeito para cada um dos pontos restantes até encontrar uma solução;
8.	Em caso de solução, armazenam-se os dados e algoritmo continua a verificação para os pontos restantes, mas desta vez, verificando se há uma segunda solução, com um círculo de raio inferior ao encontrado;
9.	Após verificar todos os pontos vizinhos restantes, utiliza-se o dado armazenado.

2.3.3 *Rendez-vous* por Consenso e RHC

Ordoñez (2013) desenvolveu esta estratégia cooperativa utilizando *Receding Horizon Control (RHC)* para a solução do problema de *rendez-vous*. A técnica consiste em se obter o valor da variável de controle a partir da otimização de uma função de custo relacionada com as variáveis controlada e de controle, tendo em vista um modelo do sistema e a previsão de seu comportamento para um determinado número de iterações futuras, o horizonte de predição (CORREIA, 2014).

A cada iteração, essa otimização é realizada e são geradas as entradas de controle futuras, das quais apenas a primeira é de fato utilizada. Uma introdução aos aspectos teóricos e práticos das técnicas de RHC mais comuns pode ser encontrada no livro de Camacho (2007). Revisões a respeito de RHC aplicado a sistemas distribuídos são feitas por Camponogara (2002), Scattolini (2009) e Christofides (2013).

O problema de otimização para um robô i é solucionado ao obter o valor mínimo de uma função objetivo, delimitada pela saturação das velocidades, como indicado em (2.5).

$$\begin{aligned} \text{minimizar} \left(\frac{1}{2} [V_i^x \quad V_i^y] \begin{bmatrix} H_i & 0 \\ 0 & H_i \end{bmatrix} \begin{bmatrix} V_i^x \\ V_i^y \end{bmatrix} + [f_i^x \quad f_i^y] \begin{bmatrix} V_i^x \\ V_i^y \end{bmatrix} \right) \\ \text{s. t. } \quad v_{\max b}^x \leq v_i^x \leq v_{\max f}^x \\ v_{\max b}^y \leq v_i^y \leq v_{\max f}^y \end{aligned} \quad (2.5)$$

A função objetivo é composta de uma função custo auxiliar f_i , o vetor de velocidade V_i e uma matriz auxiliar H_i . As restrições do problema de otimização são aplicadas às velocidades lineares nas coordenadas x e y , v_i^x e v_i^y respectivamente, nas quais são delimitadas pelas velocidades máximas de saturação $v_{\max f}$, $v_{\max b}$ do robô.

Os elementos auxiliares são obtidos em função do consenso entre o robô i e seus respectivos j vizinhos, da referência (quando presente) e de uma penalização na ação de controle, como indicado em (2.6) e (2.7).

$$H_i = \sum_{\substack{j=1 \\ j \neq i}}^n T^T a_{ij} T + T^T a_{ir} T + T'^T \gamma_i T' \quad (2.6)$$

$$f_i = \sum_{\substack{j=1 \\ j \neq i}}^n (X_{i,0}^T - X_{j,0}^T) a_{ij} T + (X_{i,0}^T - X_{r,0}^T) a_{ir} T - V_{i,0}^T \gamma_i T' \quad (2.7)$$

Onde X_i é o vetor de posição do robô i , X_j a posição recebida do seu vizinho j , X_r a referência recebida, a_{ij} e a_{ir} são os termos relacionados aos elementos da matriz adjacência referentes ao robô vizinho e a presença de referência, γ_i é um fator de ponderação para a variação de velocidade, T é uma matriz triangular inferior com dimensão $p \times p$ composto pelo passo de amostragem h , onde p é o número de predições da janela deslizante. E, como as mesmas posições e velocidades iniciais são utilizadas para todos os passos de predição, os respectivos elementos recebem o index 0 .

2.3.4 Rendez-vous por Consenso e Filtro de Kalman

O consenso utilizando Filtro de Kalman é parte de um trabalho de mestrado em andamento no PGEAS/UFSC que visa considerar as imprecisões dos sensores e da informação dos vizinhos no cálculo de consenso para *rendez-vous* de robôs móveis. A fundamentação original para modelagem da informação, é apresentada em Battistelli (2016) e pode ser resumida em quatro etapas:

- a) **Correção Local:** consiste em cada robô i adquirir as leituras (y) dos sensores, transformando-as no par δq^i (2.8) e δo^i (2.9), que representam a informação, sendo obtidas utilizando matriz de saída C e a matriz estimada V para covariância inversa dos ruídos de medição.

$$\delta q^i = C^{iT} \cdot V^i \cdot y^i \quad (2.8)$$

$$\delta o^i = C^{iT} \cdot V^i \cdot C^i \quad (2.9)$$

- b) **Consenso:** a segunda etapa consiste em fazer o consenso entre os pares δq^i e δo^i do próprio robô e dos vizinhos, utilizando a formulação clássica de consenso, onde soma-se os termos válidos multiplicados por um determinado peso λ , originalmente relacionado ao número total de termos. Este peso é modificado conforme as propostas de método do autor, podendo ser utilizado na sua formulação original ou baseado em outros fatores, como por exemplo, a distância dos robôs. Ao final, após L interações, é obtido o par $\delta q(L)^i$ e $\delta o(L)^i$, relacionado ao resultado do consenso.
- c) **Correção:** é feita a correção dos pares δq^i e δo^i do robô, utilizando o par de termos resultante do consenso, multiplicados por um termo escalar de ponderação γ . Ao final, estima-se a posição do robô \bar{x}^i que será utilizado para predição.
- d) **Predição:** ao final, é utilizada a estimativa anterior para recalcular os valores de q e o , para serem enviados aos robôs vizinhos.

2.4 REDES DE COMUNICAÇÃO

A rede de comunicação é a infraestrutura na qual se realiza a comunicação entre os robôs, sendo esta regida por meio de protocolos de comunicação. Os protocolos de comunicação regem a sintaxe, semântica e sincronização da comunicação.

Para padronizar os protocolos utilizados em uma rede, foi elaborado pela Organização Internacional de Normas (OSI), um modelo teórico de referência de 7 camadas (TANENBAUM, 2011), sendo estas listadas abaixo.

- **Camada 1 – Física:** Define as especificações elétricas e físicas da conexão de dados e a relação entre o dispositivo e o meio de transmissão. Também pode definir o sentido de transmissão e a topologia da rede;

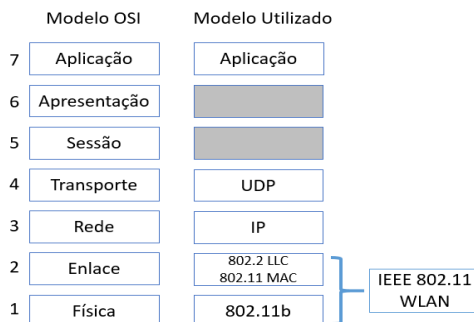
- **Camada 2 – Enlace:** A camada de enlace possibilita a transferência de dados entre dois nós distintos, possuindo meios para detecção e correção de erros que ocorrem na camada física. Essa camada pode ser dividida em controle de acesso ao meio físico (MAC) e controle de ligação lógica (LLC);
- **Camada 3 – Rede:** é a camada que fornece os meios para transferir datagramas, de um *host* de origem a um destinatário final. É responsável pelo roteamento dos dados, ou seja, determina em quais nós a informação deverá passar;
- **Camada 4 – Transporte:** é responsável por receber os dados enviados pela camada de sessão e segmentá-los para que sejam enviados a camada de rede, que por sua vez, transforma esses segmentos em pacotes. No receptor, recebe os pacotes da camada de rede e junta os segmentos para enviar à camada de sessão;
- **Camadas 5-7 –** As camadas restantes são a de Sessão, Apresentação e Aplicação. A primeira é responsável por abrir e manter sessões de comunicações e a segunda é responsável por “traduzir” os dados (e.g. decifração). A última camada é a da aplicação propriamente dita, onde estão os programas e as interações homem-máquina.

Este modelo de referência é utilizado para guiar a implementação e para apresentar de forma clara as técnicas e tecnologias utilizadas (Figura 5).

Para a rede de comunicação escolhida para os exemplos foi utilizado uma rede móvel Ad-Hoc (*Mobile Ad-Hoc Network* - MANET), que consiste em uma rede móvel sem fio que, ao contrário das redes tradicionais, não utiliza uma estação central para coordenar a troca de mensagens em uma rede, sendo os próprios nós responsáveis por encaminhar as mensagens entre si. A MANET representa um sistema distribuído complexo com nós móveis que se auto organizam dinamicamente.

Os protocolos utilizados foram UDP para camada de transporte e IP para camada de rede. Estes foram escolhidos devido a não necessidade de controle de erros (diferente do TCP, que é um protocolo orientado à conexão e com confirmação) e por serem padrões comuns, de fácil disponibilidade e compatibilidade.

Figura 5 - Relação entre o modelo de referência OSI e o modelo utilizado.

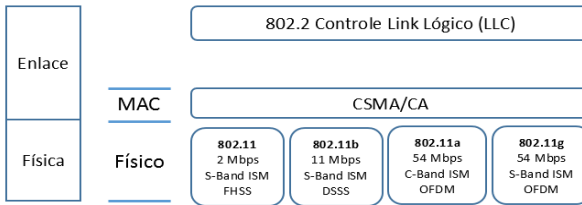


Em relação ao dispositivo físico (camada de enlace e física) foi determinado a utilização de uma comunicação sem fio entre os robôs. Essa comunicação pode ser realizada utilizando diferentes tipos de tecnologias. Para a robótica cooperativa (ZAHUGI, 2012), as tecnologias mais importantes são:

- **Bluetooth:** é um protocolo padrão de comunicação primariamente projetado para baixo consumo de energia com baixo alcance, baseado em transmissores de baixo custo. Os dispositivos usam um sistema de comunicação via rádio, não necessitando estar na linha de visão um do outro. Utiliza o padrão IEEE 802.15.
- **Wi-Fi:** Foi uma marca licenciada originalmente pela Wi-Fi Alliance para descrever a tecnologia de redes sem fio embarcadas (WLAN) baseadas no padrão IEEE 802.11. O padrão Wi-Fi opera em faixas de frequências que não necessitam de licença para instalação e/ou operação. Atualmente, trabalha principalmente nas frequências de 2.4 GHz e 5.1 GHz.
- **ZigBee:** diferencia-se por desenvolver menor consumo de energia, por possuir um alcance limitado de 100m e ter uma comunicação entre duas unidades que pode ser repetida pelas demais unidades até alcançar o destino. A tecnologia ZigBee é baseada no padrão IEEE 802.15.4.

Neste contexto, optou-se por utilizar a normatização IEEE 802.11 para as camadas de enlace e física. Esta norma define a camada de enlace e física, padronizando o LLC, utilizando o CSMA/CA para controle de acesso ao meio (MAC) e possuindo algumas versões para a camada física, que incluem a versão b, a, g (Figura 6). Estas versões afetam principalmente a largura de banda disponível.

Figura 6 – Camada física e de enlace (OSI) definida pela IEEE 802.11.



Dentre os padrões citados, foi utilizada a normatização para o IEEE 802.11b. Este padrão possui a largura de banda de 11Mb/s e aparelhos que o utilizam podem sofrer interferência de outros produtos operando na frequência de 2.4GHz, nos quais incluem o forno micro-ondas, aparelhos Bluetooth e telefones sem fio.

2.4.1 Protocolos de Acesso ao Meio

Os protocolos de acesso ao meio (MAC) são funções da camada de enlace do modelo OSI. A sua principal função é gerenciar o modo que o meio é acessado pela camada de rede.

Quando se fala em protocolos MAC, há um interesse em encontrar uma forma eficiente de compartilhar o recurso mais caro e escasso de uma rede de comunicação, o meio de transmissão, pois deseja-se evitar colisões de mensagens em dispositivos de uma rede que estão a transmitir em um mesmo espectro de frequência (CORREIA et al., 2005), podendo ser uma alocação estática ou dinâmica.

A alocação estática (síncrona) do canal divide a largura de banda disponível em N partes, alocada para cada dispositivo na rede. Cada dispositivo estará livre de colisões de sinais e de disputas pelo meio de transmissão, já que cada um transmite apenas no seu espaço alocado (slot). As principais técnicas de divisão da banda são: divisão de tempo (*Time Division Multiple Access – TDMA*), divisão de frequência (*Frequency Division Multiple Access - FDMA*) e divisão de código (*Code Division Multiple Access – CDMA*) (CORREIA, 2005).

No caso do TDMA, há necessidade de sincronização para que cada dispositivo da rede identifique seu intervalo de tempo para acesso ao meio. Para o caso de uma sincronização distribuída, cada dispositivo gera sua própria escala de operação. O uso de diferentes escalas de sincronização na rede pode causar um escorregamento de relógio (*clock drift* ou *Jitter*), levando a perda de sincronismo. Desta forma, são necessários ajustes locais e periódicos para minimizar este efeito.

Por causa dos problemas de sincronização em técnicas de alocação estática, foram desenvolvidas técnicas de alocação dinâmica do canal. Na alocação dinâmica não existe atribuição fixa de largura de banda. Os nós disputam o acesso ao meio, existindo a probabilidade de colisões no envio dos dados.

As redes Wi-Fi utilizam um protocolo CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*), baseada em quadros de controle para estabelecer um diálogo de comunicação entre os dispositivos. Neste método, os dispositivos só começam a transmissão quando acreditam que o meio estiver ocioso (*idle*) e aguardam um período de tempo para tentar transmitir novamente caso o meio esteja ocupado.

2.4.2 Protocolos Adicionais de Rede

Além dos protocolos tradicionais da rede de comunicação, é possível adicionar novas regras que afetam a forma com que a comunicação é realizada. Estas variam desde protocolos de roteamento de dados até a implementações de TDMA.

E assim como os demais protocolos de rede, estes protocolos adicionais podem ter impacto no resultado da tarefa cooperativa, sendo importante o seu estudo junto aos demais algoritmos.

Neste trabalho, são utilizados dois protocolos adicionais durante os estudos: o TDMA e o RA-TDMA (*Reconfigurable and Adaptive TDMA*).

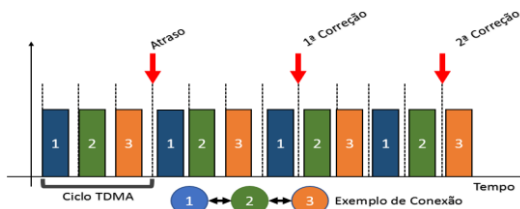
O RA-TDMA é uma forma adaptável e reconfigurável de um TDMA) (OLIVEIRA, 2015). Esta denominação tem origem na sua propriedade adaptável de sincronização dos *slots*, que não necessita sincronização de relógios.

A técnica alcança a sincronização utilizando apenas a troca de mensagens entre os robôs, que ao receber a mensagem de seus vizinhos, o robô pode estimar o atraso da mensagem e utiliza essa informação para atrasar seu *slot* de transmissão. Isto corresponde a um deslocamento de fase do ciclo do TDMA, que tende a reduzir o impacto de interferências estáticas persistentes enquanto maximiza o tempo de transmissão dos robôs.

Na Figura 7 é apresentado um exemplo do RA-TDMA para três robôs conectados sequencialmente, como é indicado na parte inferior da figura. Neste exemplo, cada robô possui um espaço no tempo para realizar sua transmissão (*slot*) e quando todos os robôs encerram sua vez, reiniciasse o ciclo de transmissões (ciclo TDMA).

Quando ocorre um pequeno atraso no robô 1 no segundo ciclo, este atraso é percebido pelo robô 2, que altera o início de seu *slot* no próximo ciclo. A correção é vista como um atraso pelo robô 3, que também altera o seu *slot* no próximo ciclo. Se não houverem mais atrasos, os robôs permanecem sincronizados entre si, mas levemente defasados em relação à posição inicial. Supõem-se que estas pequenas variações na ordem de microssegundos são aceitáveis no ponto de vista do controlador do robô e do TDMA.

Figura 7 – Exemplo de correção de atrasos utilizando RA-TDMA.



De forma geral, considerando n o número total de robôs, o tempo de *slot* (t_{slot}) para cada um é determinado pela divisão do tempo total de ciclo (t_{up}) por n , como indicado em (2.10).

$$t_{slot} = \frac{t_{up}}{n} \quad (2.10)$$

No trabalho original de Oliveira (2012), a distribuição de *slots* é dinâmica e se adapta conforme o número de robôs e aos subgrupos gerados devido à quebra da rede. Mas neste trabalho é considerado que cada robô recebe um *slot* e não muda ao longo da simulação. Essa adaptação foi mantida já que o objetivo do trabalho é manter sempre a rede conectada e qualquer falha de comunicação deve ser apenas temporária.

2.5 CONSIDERAÇÕES FINAIS

Neste capítulo foram apresentados conceitos e técnicas utilizados para exemplificar a utilização do método, também contextualizando o leitor sobre conceitos de robótica cooperativa, redes de comunicação e controle de topologia, visando melhor compreensão dos resultados obtidos.

No próximo capítulo inicia-se a formulação do método e introdução das técnicas incorporadas ao mesmo.

Capítulo 3

Técnicas e Metodologia

A proposta deste trabalho é a construção de um conjunto de instruções e procedimentos para serem utilizados para construir um estudo simulado integrado, para robótica cooperativa, redes de comunicação e controle de topologia. Para este fim, foi necessário buscar procedimentos semelhantes em outras áreas, incorporar técnicas e adaptar o procedimento resultante para esta aplicação.

Os conceitos básicos de um método direcionado para um estudo simulado podem ser generalizados em: planejamento, implementação, validação e análise de resultados (adaptado de LAW, 2008; RIBAUT, 2010).

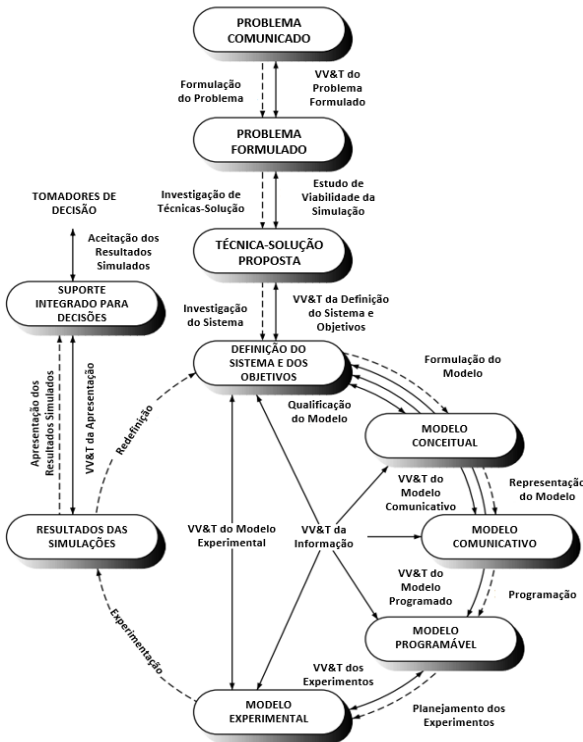
Resumidamente, o estudo simulado deve ter um correto planejamento, definindo claramente o contexto, o problema e os objetivos do estudo, facilitando a transição para a implementação da simulação. A implementação por sua vez, deve ser válida, ou seja, ser capaz de reproduzir de forma suficientemente aceitável o comportamento dos algoritmos e dos robôs envolvidos. Assim, utilizando a simulação validada, estudos de caso são planejados, simulados e seus resultados analisados, concluindo por fim, se o estudo apresentou dados ou comportamentos relevantes ao problema proposto.

Neste capítulo são abordados somente os materiais e métodos utilizados na elaboração dos passos deste método, denominado de método IRoNS (*Integrated Robotic and Network Simulation*), iniciando pela construção da formulação geral da estrutura do método, seguido pelas técnicas específicas de modelagem, validação e avaliação que foram incorporadas ao mesmo, e finalizando com a escolha do simulador para o *framework* de simulação.

3.1 CONSTRUÇÃO DO MÉTODO IRONS

Para fundamentar e estruturar o método proposto, foram pesquisados trabalhos e métodos com características semelhantes à aplicação desejada e que permitissem realizar uma adaptação para o caso de simulação de robótica cooperativa e redes de comunicação. A primeira inspiração do método veio de conceitos de verificação, validação e teste (VV&T) de modelos (BALCI, 1998)(Figura 8), onde os blocos indicam os resultados obtidos por meio de ações direcionais, indicadas pelas setas, durante o ciclo de desenvolvimento de uma simulação.

Figura 8 – Diagrama do método de Balci para verificação, validação e testes para modelagem e simulação.



Fonte: Traduzido de Balci (1998).

As ações podem ser resumidamente descritas como:

- **Formulação do Problema:** o primeiro procedimento é identificar e

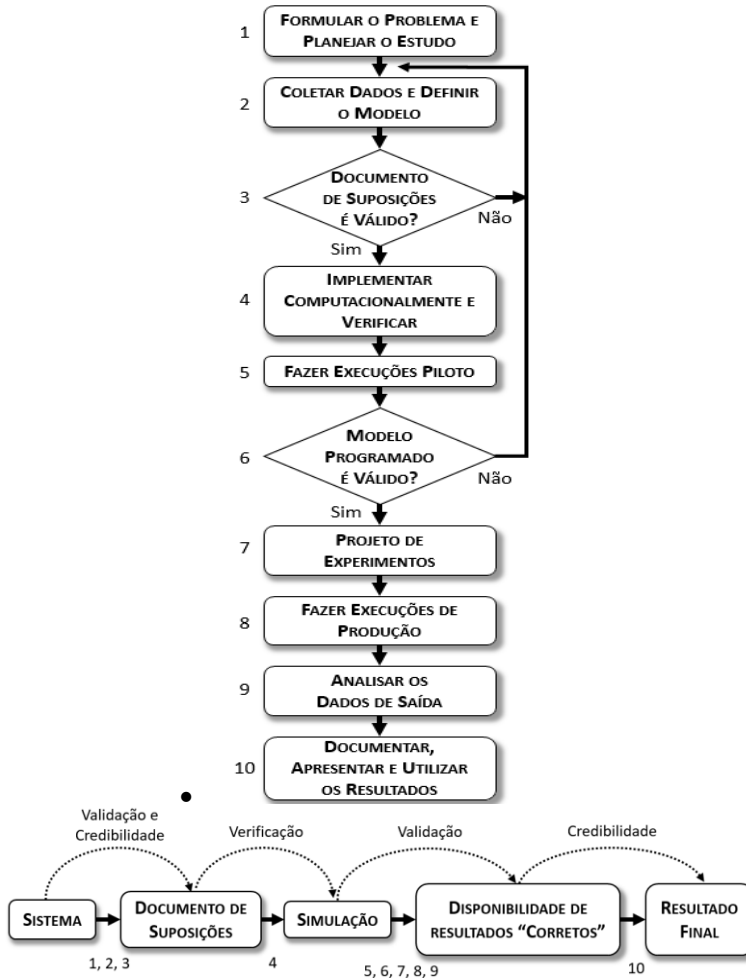
declarar o problema no qual se irá trabalhar. A partir deste ponto, inicia-se o processo de formular o problema, de forma a permitir a sua melhor compreensão;

- **Investigações de Técnicas-Solução:** com as características do problema, inicia-se a etapa de pesquisa de técnicas-soluções e da escolha dentre as soluções candidatas;
- **Definição do Sistema:** a próxima etapa é a estruturação dos objetivos e metas pretendidas, assim como as características que o sistema e que as técnicas devem possuir. Deve-se estudar se o sistema sofre mudanças previsíveis ou estocásticas, o ambiente no qual está inserido, comportamentos não intuitivos, interdependências e organização;
- **Formulação do Modelo Conceitual:** formula-se o modelo conceitual da simulação baseado nas informações das etapas anteriores. Ou seja, uma modelagem para indicar a estrutura do sistema e seu comportamento;
- **Representação do Modelo:** organiza-se o modelo conceitual de forma que seja possível sua fácil compreensão e reprodução, por meio de grafos, diagramas, pseudocódigos ou outras formas de representação;
- **Programação:** é a tradução do modelo comunicativo em linhas de código em uma ferramenta escolhida, resultando no modelo programável;
- **Planejamento dos experimentos:** após a implementação, deve-se planejar os experimentos, levando em conta quais aspectos deseja-se obter ou avaliar durante cada experimento, resultando no modelo experimental;
- **Experimentação:** é o processo de realizar os experimentos utilizando a simulação. Pode consistir de treinamento de algoritmos, análise de sensibilidade, alterações de parâmetros, alteração de função, dentre outros;
- **Avaliação final:** ao final, analisa-se os resultados e é tomada uma decisão, dado que os resultados podem ser ou não satisfatórios. Em caso negativo, o ciclo é reiniciado.

O trabalho de Balci (1998) descreve pontos importantes para o ciclo de desenvolvimento de simulações. No entanto, como é visto na Figura 8, são indicados passos de verificação, validação e teste (VV&T) para cada etapa, gerando excessivas redundâncias. Outro ponto é a proximidade de alguns tópicos, que poderiam ser integrados em um único passo (eg.: modelagem conceitual e comunicativa).

Essa integração dos passos é observada em uma proposta mais recente (LAW 2015) (Figura 9) e em materiais atuais do autor do trabalho original (BALCI, 2013). Em Law (2015), os passos de VV&C (Verificação, Validação e Credibilidade) são realizados em um número menor de etapas (parte inferior da Figura 9).

Figura 9 – Proposta de método para modelos simulados por Law, com indicação das etapas de VV&T na parte inferior.



Fonte: Traduzido de Law (2015).

Resumidamente, o procedimento proposto por Law (2015) consiste de 10 passos:

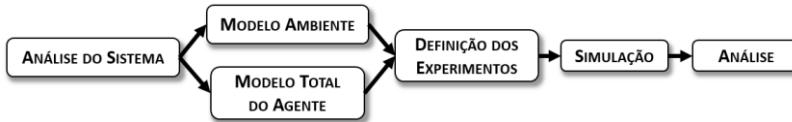
- **Formular o problema e planejar o estudo (1):** consiste em definir o problema, os objetivos do estudo, as medições de desempenho que deverão ser adquiridas, o escopo do modelo, a configuração do sistema e os recursos requeridos;
- **Coletar dados e definir o modelo (2):** coletar informação sobre o sistema e os procedimentos de operação, coletar dados para especificar os parâmetros, delimitar o escopo do estudo, criar o documento de suposições e escolher o nível de detalhamento do modelo;
- **Validade do documento de suposições (3):** realizar uma verificação estruturada do documento de suposições antes de prosseguir;
- **Implementar computacionalmente (4):** elaborar o modelo em uma linguagem de programação ou em um *software* de simulação;
- **Realizar execuções piloto (5):** realizar execuções da simulação com o propósito de validação;
- **Validação do modelo (6):** realizar a comparação entre o modelo e o sistema, revisar o modelo à procura de falhas e fazer análise de sensibilidade;
- **Projetar experimentos (7):** definir a duração de cada execução da simulação, construir várias simulações independentes utilizando números aleatórios;
- **Realizar execuções de produção (8):** realizar mais execuções da simulação para coletar dados para o próximo passo;
- **Analisar os resultados (9):** determinar o desempenho absoluto das configurações do sistema e comparar configurações alternativas do sistema;
- **Conclusão (10):** preparar a versão final do documento de suposições, do código da simulação e resultados, para futuro uso em outros projetos, apresentar os resultados.

Em relação aos métodos utilizados para aplicações semelhantes a de robótica cooperativa, destacam-se os métodos para ABMS (*Agent Based Modeling and Simulation*), como por exemplo, o método apresentado por Junges (2013) (RAUNAK, 2013 e 2014), exemplificado na Figura 10. Apesar deste exemplo ter uma estrutura similar aos métodos apresentados anteriormente, há um maior agrupamento entre os passos e requer um processo de V&V diferenciado (KLUGL, 2008).

O método IRoNS adota o principal ponto em comum de todos estes trabalhos: a necessidade de planejar, implementar e validar cada etapa.

Esta proposta é indicada na Figura 11, onde são indicadas as etapas de preparação do estudo, elaboração de uma simulação padrão e a etapa de estudos de caso com validação em relação a simulação de referência.

Figura 10 – Método tradicional para ABMS.



Fonte: Traduzido de Junges (2013).

Figura 11 – Diagrama com representação da proposta simplificada do método.

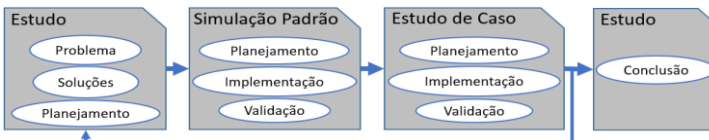
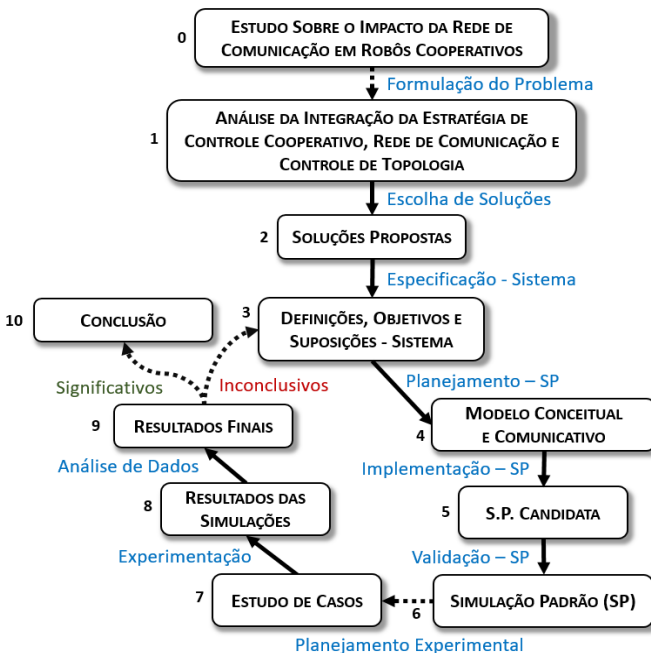


Figura 12 - Método para análise de simulações integradas de robôs cooperativos e redes de comunicação.



Este diagrama é expandido para uma estrutura de 10 passos (Figura 12) inspirada nos trabalhos de Law (2015). Neste diagrama, as setas representam ações e as caixas representam o resultado desta ação. O conjunto ação/resultado é denominado de procedimento e é representado por uma numeração de 1 a 10. O número “0” é considerada apenas o ponto de partida, ou seja, a ideia inicial do estudo que se pretende realizar.

As setas em tracejado (entre 0-1, 6-7, 9-10-3) indicam que ao prosseguir para o próximo procedimento, recomenda-se não retornar ao passo anterior.

Resumidamente, os procedimentos do método IRoNS (ação e resultado) são apresentados abaixo e detalhados nos próximos subitens.

- **Formulação do problema (1):** dado o tema do trabalho, o primeiro passo é detalhar e formalizar o problema. Ou seja, especificar qual o escopo do que será trabalhado e a categoria de técnicas que serão utilizadas;
- **Pesquisa e escolha de soluções (2):** em cada tópico escolhido, são pesquisadas as técnicas que se encaixam nas definições e dentre estas, dado um critério do autor, é escolhida uma técnica para cada tópico para serem implementadas;
- **Especificação do sistema (3):** nesta etapa, levando em conta as técnicas escolhidas, são elaborados os objetivos do estudo, definições adicionais relacionadas às técnicas e a documentação das suposições;
- **Planejamento da SP (4):** o planejamento da simulação padrão, engloba a modelagem conceitual e comunicativa, ou seja, as representações informal e descritiva respectivamente, com objetivo de auxiliar na compreensão e implementação da simulação;
- **Implementação da SP (5):** o quinto procedimento é a implementação da simulação na ferramenta escolhida. Esta simulação deverá ter parâmetros conservadores e apenas a técnica foco do estudo;
- **Validação da SP (6):** antes de utilizar a SP resultante, é necessário realizar uma etapa de validação desta SP candidata para verificar se os resultados da mesma correspondem a dados reais, dados de uma *testbed* ou de uma simulação em outro software;
- **Planejamento experimental (7):** se a SP é válida, inicia-se a etapa de planejamento experimental: implementa-se os algoritmos adicionais, planeja-se quais condições iniciais deverão ser utilizadas e quais parâmetros deverão ser variados para avaliar os resultados;

- **Experimentação (8):** a experimentação é a execução das simulações planejadas pelos estudos de caso, realizando um número adequado de simulações visando a análise de dados;
- **Análise dos dados (9):** a análise consiste primeiramente em extrair informações relevantes do conjunto de dados gerados pela experimentação, para em seguida realizar uma validação estatística dos mesmos em relação a SP para verificar se existem diferenças significativas;
- **Conclusão (10):** a última etapa é discutir os resultados e verificar se estão de acordo com os objetivos estabelecidos para o trabalho. Em caso negativo, o trabalho é reiniciado. Além disso, os resultados podem ser melhor trabalhados, visando melhor apresentação.

Estas etapas são detalhadas e exemplificadas passo a passo no próximo capítulo. No entanto, algumas etapas incorporam técnicas específicas, sendo estas, detalhadas nas próximas seções.

3.2 TÉCNICAS UTILIZADAS

Uma vez estruturado o método IRoNS, é preciso especificar técnicas para seus pontos chaves, as quais necessitam não só lidar com as particularidades da simulação integrada, mas também trabalhar em sinergia com as demais técnicas para obter os resultados esperados. A escolha das técnicas ocorreu de forma pragmática, considerando os principais autores da área e a sua utilização em aplicações similares à de robótica cooperativa.

Nos próximos itens, são brevemente discutidas as técnicas adotadas, a motivação por trás de cada escolha e os principais fundamentos envolvidos nas mesmas. A integração das técnicas ao método é discutida no próximo capítulo.

3.2.1 Transição entre o Planejamento e Implementação

A transição entre o que se deseja estudar e a implementação da simulação, ou seja, o Planejamento da SP, é o primeiro ponto crucial do método. Para esta finalidade, utilizou-se a modelagem conceitual e a modelagem comunicativa. As duas modelagens são tidas como etapas individuais em Balci (1995), no entanto, trabalhos recentes em simulações mais simples as utilizam em conjunto (LAW, 2015; SOKOŁOWSKI, 2010; ROBINSON, 2006; SIEGFRIED, 2014). Devido

à complexidade envolvida na simulação integrada, optou-se por mantê-las em um mesmo passo do método, mas executa-las individualmente.

Na modelagem conceitual, não há preocupação com o modelo computacional do sistema, ou seja, é uma ideia generalizada da interação entre seus componentes e da funcionalidade desejada (SOKOLOWSKI, 2010). Esta modelagem pode ser considerada uma representação simplificada do sistema e que deverá ser interativa e repetitiva, ou seja, o modelo deve ser constantemente revisado durante o estudo (ROBINSON, 2006). Um extenso *survey* sobre modelagem conceitual é apresentado em Robinson (2006), que inclui recomendações, métodos formais e meios para validação de modelos conceituais.

No âmbito deste trabalho, é adotada uma abordagem similar à das propostas utilizadas em Sokolowski (2010) e Siegfried (2014). Nesses trabalhos, é dito que a modelagem deve ser simples e pode ser feita por meio de diagramas, figuras, mapa de conceitos ou de outras formas visuais, visando uma simples representação da estrutura do sistema e de sua funcionalidade.

Quanto ao modelo comunicativo, este é considerado uma extensão natural e mais detalhada do modelo conceitual, tendo como foco a comunicação do problema a ser estudado e da simulação para outros membros da equipe ou para pessoas externas ao projeto. Este detalhamento pode ser feito por modelagens formais, informais, pseudocódigos, extensa documentação ou de outras formas julgadas adequadas pelos autores para a aplicação em questão.

Railsback (2012), Klugl (2012) e Macal (2014) apontam algumas técnicas de modelagem comunicativa para ABMS, destacando o protocolo ODD (*Overview, Design concepts and Details*) (GRIMM, 2006 e 2010), o qual consiste em uma série de questionamentos seccionados por áreas de interesse, agrupados em 3 grandes categorias com subdivisões, resultando em 7 categorias gerais (Quadro 3) e 11 subitens relacionados aos conceitos de projeto.

Quadro 3 – Categorias e subcategorias do protocolo ODD.

Visão Geral	Propósito
	Entidades, Variáveis de Estado e Escala
	Panorama do Processo e Agendamento
Conceitos de Projeto	Conceitos de Projeto
Detalhes	Inicialização
	Entrada
	Submodelos

Fonte: Traduzido de Grimm (2006, 2010)

Devido à sua objetividade e aplicação em ABMS, o protocolo ODD foi escolhido para a modelagem comunicativa.

A escolha destes tópicos foi ponderada levando em conta os seguintes pontos:

- Utilizar abordagens conceitual e comunicativa que não exijam linguagens complexas, possibilitando a extração de informações com uma simples visualização e/ou leitura;
- Leva-se em conta que as abordagens não precisam promover provas, códigos e nem demandar demasiado tempo de elaboração. E sim, apenas reunir as informações necessárias para guiar uma correta implementação, incentivar a convergência do estudo com o objetivo proposto (evitar divagações durante a implementação) e permitir explanar os pontos mais importantes da simulação para outros membros ou para futuras reproduções da simulação;
- Não há preocupação em determinar a melhor abordagem para estas modelagens, e sim propor a utilização de abordagens que alcancem um nível mínimo o suficiente de detalhamento que permita alcançar as metas determinadas para este passo no método proposto;
- Optou-se por abordagens citadas na literatura e que foram aplicadas em ABMS, dada a semelhança de conceitos desta aplicação com conceitos de robótica cooperativa. A adaptação destes conceitos para robótica é natural e não demanda uma completa reestruturação da técnica;
- Por último, considera-se a utilização não restritiva destas abordagens. Ou seja, se houver necessidade, modelagens adicionais podem ser consideradas e executadas após estas etapas. Esta situação pode ocorrer, por exemplo, se membros de um projeto desejam utilizar uma linguagem específica para comunicação interna do projeto.

3.2.2 Processo de Validação

O processo de validação é uma verificação de que o modelo utilizado é adequado para dada finalidade (KLUGL, 2008). Ou seja, se a simulação é válida o suficiente para ser capaz de produzir resultados confiáveis durante os experimentos. Este processo requer conhecimento

do sistema e criatividade, visto que existe atualmente mais de 100 técnicas de validação, verificação e teste (VV&T) (BALCI, 2013) de simulações.

Segundo Balci (2013), as inúmeras técnicas de VV&T podem ser classificadas conforme suas particularidades, em quatro grandes categorias:

- **Informal:** são denominadas de informais pois as ferramentas e abordagens utilizadas dependem principalmente de raciocínio humano e da subjetividade, não possuindo restrição de um formalismo matemático;
- **Formal:** são baseadas em formalismos e provas matemáticas;
- **Estática:** são técnicas que focam na verificação da precisão utilizando como base as características do projeto do modelo estático e código fonte. Ou seja, não necessitam da execução computacional do modelo;
- **Dinâmica:** são técnicas que analisam o comportamento da execução computacional, podendo exigir instrumentação adicional e módulos adicionais na programação com intenção de coleta de dados.

A utilização destas técnicas visa evitar três tipos de erros durante a modelagem e simulação, denominados de erros tipo I, II e III (SOKOLOWSKI, 2010; SARGENT, 2013). O erro tipo I ocorre quando um modelo é válido, mas é rejeitado na análise de validação. O erro tipo II ocorre quando um modelo inválido é aceito e utilizado. Por fim, o erro tipo III ocorre quando o modelo está correto e válido, mas é irrelevante para a aplicação ou para o objetivo do estudo.

Devido à complexidade envolvida na simulação integrada, foram necessárias algumas suposições para delimitar o problema, sendo adotadas algumas estratégias de validação.

A primeira estratégia adotada é a de alinhamento de modelos (KLUGL, 2008; SARGENT, 2015; LAW, 2015) e consiste na afirmação de que a validade é uma relação transitiva. Ou seja, se o modelo A reproduz validamente o sistema O e o modelo B replica os resultados de A , então B também é válido para reproduzir O . É considerado que o mesmo conceito é válido para ferramentas de simulação. Ou seja, se os resultados de uma simulação em uma ferramenta A produz resultados considerados válidos e se os mesmos resultados forem obtidos (sob as mesmas condições) em uma ferramenta B , então a simulação e o simulador são válidos para o algoritmo em questão.

Utilizando este conceito, é possível realizar validação individual dos algoritmos para verificar se os mesmos estão funcionando de forma

adequada no *framework* de simulação. Por exemplo, se uma estratégia de controle cooperativo foi desenvolvida no *software Matlab*, os dados resultantes desta simulação podem ser comparados aos obtidos no *framework* de simulação do método ao utilizar os mesmos parâmetros e condições de simulação.

A segunda estratégia para a validação é a de que independente da forma de validação, se os algoritmos forem individualmente considerados válidos, a simulação integrada dos mesmos também será suficientemente válida.

Além destas duas estratégias de validação, é necessário a escolha das técnicas pertinentes a uma simulação de robótica cooperativa. Esta aplicação, de forma semelhante à ABMS, apresenta algumas particularidades que devem ser levadas em consideração, como por exemplo, a complexidade derivada da separação entre alto/baixo nível, a falta de uma ligação intermediária, a complexidade devido a não linearidades (devido a um elevado número suposições e da parametrização excessiva) e devido à falta de dados reais, afetando a validação estatística (KLUGL, 2008).

Para lidar com estas particularidades, Klugl (2008) propõem a utilização de uma sequência de quatro técnicas (uma informal e três dinâmicas) para validação de ABMS. As quatro propostas, que também foram adotadas para o método, são:

- **Validação de Face:** utilizando todos os documentos e simulações gerados, é feita uma verificação visual à procura de inconsistências. Esta etapa pode ser dividida em três fases:
 - *Verificação da animação:* as animações, gráficos, diagramas e qualquer outro aspecto visual da simulação são verificados à procura de erros;
 - *Verificação da saída:* o resultado da simulação é avaliado quanto a sua plausibilidade, como por exemplo, se as ordens de grandeza estão corretas;
 - *Verificação imersiva:* cada agente da simulação é profundamente avaliado, verificando suas ações e reações ao longo da simulação;
- **Análise de Sensibilidade:** esta técnica consiste em alterar parâmetros da simulação, avaliando se o impacto da simulação é reproduzido de forma adequada (comparando resultados com uma simulação válida ou sistema real);

- **Calibração:** após determinar quais e como os parâmetros afetam a simulação, a calibração é utilizada para fazer ajustes finos nas simulações;
- **Validação Estatística:** a última técnica é aplicada quando todas as etapas anteriores foram concluídas e os resultados estão visualmente próximos do esperado, necessitando apenas de comprovação estatística. A técnica para este passo não foi definida em Klugl (2008), portanto, foi escolhido utilizar a validação por intervalo de confiança, proposto em Law (2015). Dada a importância da validação estatística, a mesma é detalhada separadamente na próxima seção.

Ao adotar este procedimento de validação foram considerados os seguintes pontos:

- O processo de validação desejado refere-se à reprodução das técnicas no *framework* de simulação do método e não à precisão dos resultados em relação à um sistema real.
- A validação deve conseguir lidar com diferentes categorias de algoritmos, buscando estratégias viáveis que possibilitem cogitar diferentes formas de validação;
- Considerou-se que as técnicas deveriam apresentar praticidade durante as suas execuções, terem sido utilizadas em uma aplicação de robótica ou ABMS, e que, permitisse uma integração com uma técnica de validação estatística. Portanto, a proposta de Klugl (2008) atende perfeitamente esses requisitos.

3.2.3 Análise Estatística

A validação estatística é fundamental tanto para etapa de validação quanto para análise dos resultados, sendo utilizada para comparação de resultados entre simulações, entre simulação e sistema real ou entre versões de uma mesma simulação utilizando diferentes parâmetros.

Balci (2013) cita diversas técnicas de validação estatística que podem ser utilizadas em M&S (Modelagem e Simulação), destacando-se as técnicas de construção do teste de hipótese e de intervalo de confiança.

O teste de hipótese, também mencionado em Law (2008), Lim (2012), Sokolowski (2010) e Sargent (2015), consiste em assumir uma hipótese para o erro médio entre os modelos e verificar se a hipótese é válida ou não para certo grau de confiança.

No entanto, Law (2015) e Robinson (2014) recomendam a construção de intervalo de confiança (IC) ao invés do teste de hipótese,

devido ao fato de que ambas as técnicas possuem os mesmos fundamentos, mas que, contudo, a construção do IC se diferencia porque além de indicar se há ou não uma diferença estatística significativa entre os modelos, indica também a faixa de valores válidos para um determinado grau de confiança.

A fundamentação estatística do IC é discutida em detalhes nos livros de Montgomery (1991), Stevenson (2001), Hines (2003), Barros Neto (2003) e Law (2015), sendo aqui discutidos somente os principais pontos de sua construção.

Sejam $Z_1, \dots, Z_2, Z_i, \dots, Z_n$ variáveis aleatórias discretas equiprováveis, tais que $Z_1, \dots, Z_i, \dots, Z_n \in Z$, onde n é o número de observações Z_i realizadas a respeito da população total Z , de média μ_z e variância $Var(\mu_z)$ finitas.

Para uma média amostral $\bar{Z}(n)$ definida por (3.1).

$$\bar{Z}(n) = \frac{\sum_{i=1}^n Z_i}{n} \quad (3.1)$$

E para uma variância amostral $\widehat{Var}(\bar{Z}(n))$ definida por (3.2).

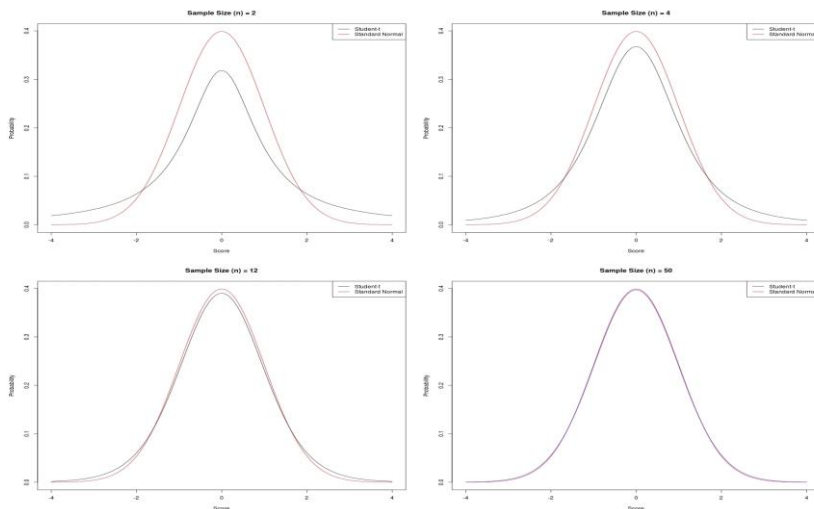
$$\widehat{Var}[\bar{Z}(n)] = \frac{\sum_{i=1}^n [Z_i - \bar{Z}(n)]^2}{n(n-1)} \quad (3.2)$$

Pode-se afirmar que para $n \rightarrow \infty$, $\bar{Z}(n)$ é um estimador imparcial de μ_z e $\widehat{Var}[\bar{Z}(n)]$ é um estimador imparcial de $Var(Z)$. No entanto, a dificuldade ao se utilizar $\bar{Z}(n)$ e $\widehat{Var}[\bar{Z}(n)]$ como estimadores para μ_z e $Var(Z)$ da população sem informações adicionais, é que não é possível saber quão perto $\bar{Z}(n)$ está de μ_z (LAW, 2015), ou seja, quão próximo a média de n observações realizadas está da média verdadeira da população, e que portanto, não permite determinar o valor necessário de n para que a expressão $\bar{Z}(n) \approx \mu_z$ seja verdadeira.

A solução é a construção de um intervalo de confiança (IC), que consiste de uma faixa de valores onde é estimado que a média verdadeira da população esteja contido com certo grau de confiança, utilizando as informações amostrais e uma curva de distribuição.

Ao se trabalhar com um número baixo de amostras, como no caso de simulações, é recomendado a utilização da distribuição *t-student* para uma melhor estimativa dos pontos críticos para o IC (LAW, 2015) (Figura 13).

Figura 13- Comparação distribuição *t-student* de 2,4,12 e 50 amostras (preto) x distribuição normal (vermelho).



Seja G_{conf} o grau de confiança desejado, determina-se um parâmetro auxiliar α , como indicado em (3.3).

$$\alpha = 2 \cdot \left(1 - \frac{G_{conf}}{100}\right) \quad (3.3)$$

Para $n-1$ amostras e α , é determinado o ponto crítico $t_{n-1, 1-\frac{\alpha}{2}}$ da distribuição *t-student* utilizando dados tabelados (ANEXO C).

O IC é definido em (3.4) em função da média amostral e do meio-comprimento L_t , sendo L_t determinado por (3.5) em termos da variância amostral e do ponto crítico.

$$IC = [\bar{Z}(n) + L_t, \bar{Z}(n) - L_t] \quad (3.4)$$

$$L_t = t_{n-1, 1-\frac{\alpha}{2}} \sqrt{\widehat{Var}[\bar{Z}(n)]} \quad (3.5)$$

Esta formulação do IC pode ser aplicada tanto no processo de validação da simulação quanto na análise do planejamento experimental, discutido nos próximos itens.

3.2.4 Planejamento Experimental

O objetivo de um planejamento experimental para simulação é a determinação de quais parâmetros possuem o maior efeito na resposta e em como fazer isto com o menor número de simulações (LAW, 2015). No entanto, neste trabalho, a prioridade não é somente descobrir o parâmetro de maior impacto e sim verificar quais parâmetros ou cenários possuem qualquer tipo de impacto no resultado da tarefa cooperativa e se o mesmo é significativo ou não.

Dentre as técnicas de planejamento experimental encontrados na literatura (MONTGOMERY, 1991), pode-se citar o uso de técnicas fatoriais para análise de impacto de parâmetros, a construção de meta-modelos para prever a resposta de modelos que não estão presentes na simulação ou a utilização de superfícies de respostas para descobrir qual a combinação de parâmetros que otimiza a resposta.

Uma solução é a utilização do planejamento experimental fatorial 2^k para determinar quais parâmetros afetam o sistema e se há interação entre os mesmos (BARROS NETO, 2003; LAW, 2015). Além disso, é possível utilizar esta técnica tanto para fatores quantitativos quanto para fatores qualitativos (MONTGOMERY, 1991). É indicada quando se deseja ter uma estimativa inicial de como os parâmetros afetam o sistema, sendo inadequada apenas quando o número de fatores simultâneos for muito alto.

Na terminologia do planejamento experimental, os parâmetros são denominados fatores e o critério de avaliação (variável numérica de indicação de desempenho do sistema) é denominado resposta.

Determina-se dois valores para cada fator, denominados pontos experimentais, correspondendo aos valores dos quais se deseja comparar. Os pontos experimentais são representados pelo sinal de negativo “-” e positivo “+”.

Portanto, para K fatores, são geradas 2^K combinações de respostas R^c , sendo $c = 1, 2, \dots, 2^K$.

Para representar a visão total do projeto experimental, é utilizada uma matriz de projeto, como exemplificado no Quadro 4 para o caso de três fatores. A matriz contém todas as informações necessárias para iniciar o planejamento experimental.

Quadro 4 – Exemplo de matriz de projeto para planejamento fatorial 2^3 .

Combinação de Fatores	Fator 1	Fator 2	Fator 3	Resposta (R^c)
1	-	-	-	R^1
2	+	-	-	R^2
3	-	+	-	R^3
4	+	+	-	R^4
5	-	-	+	R^5
6	+	-	+	R^6
7	-	+	+	R^7
8	+	+	+	R^8

Fonte: Traduzido de Law (2015).

O impacto de um fator k , indicado por e^k , é a mudança média nas respostas ao mudar o seu valor “-” para o seu valor indicado em “+”. O impacto de um fator é calculado utilizando (3.6), onde $sinal(R^c, k) = [+1, -1]$, determinado de acordo com o respectivo sinal indicado na linha da resposta R^c e coluna do fator k da matriz de projeto.

$$e^k = \frac{\sum_{c=1}^{2^K} R^c \cdot sinal(R^c, k)}{2^{K-1}} \quad (3.6)$$

Para o exemplo do Quadro 4, o impacto de cada fator e^k pode ser obtido por meio das equações indicadas em (3.7).

$$e^1 = \frac{1}{2^{3-1}} [R^1 \cdot (-1) + R^2 \cdot (+1) + R^3 \cdot (-1) + R^4 \cdot (+1) + R^5 \cdot (-1) + R^6 \cdot (+1) + R^7 \cdot (-1) + R^8 \cdot (+1)]$$

$$e^2 = \frac{-R^1 - R^2 + R^3 + R^4 - R^5 - R^6 + R^7 + R^8}{4} \quad (3.7)$$

$$e^3 = \frac{-R^1 - R^2 - R^3 - R^4 + R^5 + R^6 + R^7 + R^8}{4}$$

Para determinar se há interação $e^{k_1 k_2}$ entre 2 fatores k_1 e k_2 , é utilizada uma formulação similar como indicando em (3.8), inserindo um novo termo relacionado ao sinal do segundo fator.

$$e^{k_1 k_2} = \frac{\sum_{c=1}^{2^K} R^c \cdot sinal(R^c, k_1) \cdot sinal(R^c, k_2)}{2^{K-1}} \quad (3.8)$$

Para o exemplo anterior, a interação dos fatores é obtido por (3.9).

$$\begin{aligned}
 e^{12} &= \frac{R^1 - R^2 - R^3 + R^4 + R^5 - R^6 - R^7 + R^8}{4} \\
 e^{13} &= \frac{R^1 - R^2 + R^3 - R^4 - R^5 + R^6 - R^7 + R^8}{4} \\
 e^{23} &= \frac{R^1 + R^2 - R^3 - R^4 - R^5 - R^6 + R^7 + R^8}{4}
 \end{aligned} \tag{3.9}$$

Este procedimento pode ser expandido para qualquer combinação dos k fatores, inserindo até K termos de $\text{sinal}(R^c, k_K)$. Para o exemplo anterior, como o valor máximo de K é 3, resulta na Equação (3.10).

$$e^{123} = \frac{-R^1 + R^2 + R^3 - R^4 + R^5 - R^6 - R^7 + R^8}{4} \tag{3.10}$$

É importante mencionar que a formulação apresentada até o momento é para apenas um experimento realizado para cada combinação de fatores. No entanto, ao integrar esta técnica à análise estatística, são realizados $i = 1, 2, \dots, n$ experimentos (amostras) para a construção do intervalo de confiança, resultando em diferentes R^c e e^k para cada conjunto de amostras, onde é adicionado o índice i para diferenciá-las, resultando em R_i^c e e_i^k .

3.2.5 Análise dos Resultados

Após planejar os experimentos e adquirir todos os resultados tabelados de cada amostra i , é realizada a análise dos resultados por meio da integração do planejamento experimental e a análise estatística por intervalo de confiança.

Para a construção do IC, ao invés de utilizar a média e variância sobre erro entre simulações, como apresentado para a validação, a construção do IC é feita em relação aos impacto dos fatores e_i^k calculados no planejamento experimental.

Assim, a média amostral $\bar{e}^k(n)$ e a variância amostral $\widehat{\text{Var}}[\bar{e}^k(n)]$ são determinadas pelas equações(3.11) e (3.12), para n amostras e para os K fatores e/ou combinação de fatores utilizados.

$$\bar{e}^k(n) = \frac{\sum_{i=1}^n e_i^k}{n} \tag{3.11}$$

$$\widehat{\text{Var}}[\bar{e}^k(n)] = \frac{\sum_{i=1}^n [e_i^k - \bar{e}^k(n)]^2}{n \cdot (n - 1)} \tag{3.12}$$

A partir do grau de confiança desejado, a variável auxiliar α é determinada da mesma forma que anteriormente, indicada em (3.3).

Ao final, é construído o intervalo de confiança IC_{e^k} (3.13), utilizando α e o meio-comprimento L_{e^k} (3.14).

$$IC_{e^k} = [\bar{e}^k(n) + L_{e^k}, \bar{e}^k(n) - L_{e^k}] \quad (3.13)$$

$$L_{e^k} = t_{n-1, 1-\frac{\alpha}{2}} \sqrt{\widehat{Var}[\bar{e}^k(n)]} \quad (3.14)$$

Os intervalos de confiança resultantes estão na mesma dimensão da resposta, sendo possível uma análise direta sobre sua grandeza. Se os IC não contiverem 0, é possível concluir que o efeito do respectivo fator é estatisticamente significativo, ou seja, houve alteração significativa dos resultados ao alterar este parâmetro.

O comprimento e o valor absoluto do IC também fornecem indicadores importantes para a análise. Um IC de grande comprimento indica uma grande imprecisão, que pode ser resultante de uma má escolha de parâmetros ou de amostras, ou que o número de amostras é muito baixo. O valor absoluto indica se na prática, a diferença é significativa. Por exemplo, um IC sobre a diferença no tempo de convergência for de [1, 2] segundos, este pode ser insignificante se o tempo total de convergência foi de minutos.

3.3 CONSIDERAÇÕES FINAIS

Neste capítulo foram apresentados a construção inicial do método IRoNS e, isoladamente, os conceitos básicos das principais técnicas incorporadas ao mesmo.

O objetivo foi indicar a origem e inspirações do método, detalhando também como cada técnica funciona e quais considerações foram realizadas para sua escolha e para sua integração ao método. Não houve nesta seção a preocupação em encontrar a técnica com melhor desempenho e sim, técnicas utilizadas em aplicações semelhantes que poderiam ser adaptadas para o caso da robótica cooperativa e de redes de comunicação.

No próximo capítulo, o método é apresentado passo a passo em sua forma final, já integrado com os materiais apresentados aqui, sendo ainda exemplificada a execução de cada um dos passos.

Capítulo 4

Método IRoNS

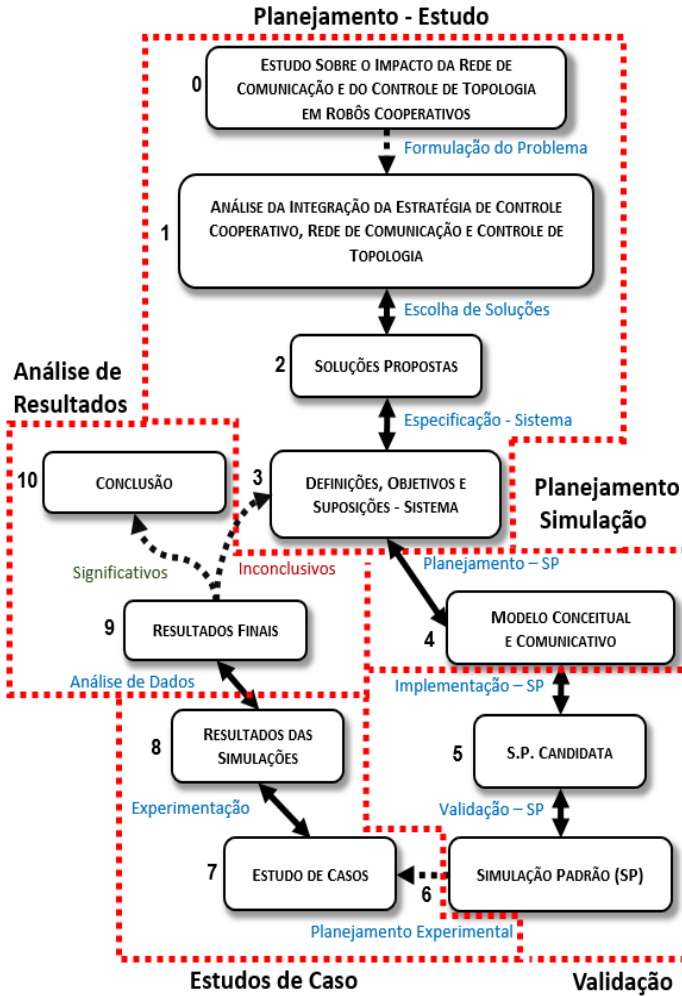
No capítulo anterior foi apresentada a base para formulação do método IRoNS, iniciando pela sua fundamentação e passando pelas principais técnicas integradas ao mesmo. O método IRoNS foi testado por meio de vários estudos de caso, onde a cada novo estudo, seus procedimentos foram refinados até ser considerado adequado e finalizado.

A versão final do método junto com sua integração com as técnicas descritas no capítulo anterior, é detalhada neste capítulo em cinco itens: planejamento do estudo, planejamento da simulação, validação, estudos de caso e análise de resultados (Caixas vermelhas na Figura 14).

Nesse diagrama, cada caixa representa um ponto de partida, resultado ou conclusão. As setas representam ações direcionais, cujos os resultados são indicados nas caixas numeradas. Utilizou-se o termo procedimento para o par de ação e resultado (seta e caixa subsequente), com exceção do primeiro e do último procedimento, nos quais possuem algumas particularidades que fogem a esta regra.

Ao longo do capítulo também são fornecidos pequenos exemplos para cada item, visando uma melhor compreensão da execução de cada procedimento. Os exemplos de estudos de caso usando o método IRoNS e suas respectivas contribuições são detalhados no próximo capítulo.

Figura 14 - Procedimentos do método com a indicação das técnicas utilizadas.



4.1 PLANEJAMENTO DO ESTUDO

Os passos iniciais (de 0 a 3) foram concebidos e direcionados para a fundamentação do estudo, focando principalmente na formulação do problema. Balci (2013) discute a importância de uma boa formulação do problema, onde o principal objetivo é evitar que o erro tipo III ocorra, ou seja, evitar que a solução (simulação) apresentada seja irrelevante ao

problema ao qual se deseja abordar. O autor cita cinco passos para formulação do problema que corroboram com os pontos levantados na elaboração do método (Seção 3.1), sendo estes:

- Adquirir dados e informações sobre o problema, baseado em uma ideia estabelecida *a priori*;
- Identificar as pessoas interessadas e os tomadores de decisão (e.g.: quais departamentos serão afetados em uma empresa);
- Especificar as necessidades e os objetivos;
- Identificar as limitações do problema;
- Especificar todas as suposições de forma clara e explícita.

Estes passos foram adaptados e incorporados ao longo do método, nos procedimentos de 0 a 3 (Figura 14).

O primeiro passo, referido como procedimento 0, não possui uma ação prévia (seta) como os demais procedimentos, é tratado como ponto inicial do método e não como um procedimento propriamente dito. Aqui, se define a ideia fundamental do que se deseja estudar e qual tópico será alvo do estudo. Por mais simples que seja, ter o assunto bem definido é parte fundamental de qualquer método para simulação (LAW, 2015), principalmente ao considerar que as simulações não devem ser mais complexas do que o necessário.

Assim, é necessário esclarecer qual dos três tópicos da simulação integrada (estratégia de controle cooperativo, redes de comunicação ou controle de topologia) será o foco do estudo e se possível, uma ideia inicial do que se deseja estudar no mesmo. Dentre as possibilidades, pode-se optar por ter:

- **Foco na Robótica Cooperativa:** estudos que visam avaliar o impacto dos demais algoritmos e de seus parâmetros no desempenho de uma tarefa cooperativa. Exemplo: Impacto de uma rede com colisões de mensagens em uma tarefa cooperativa;
- **Foco na Rede de Comunicação:** estudos que visam estudar o desempenho da rede de comunicação como um todo ou avaliar o desempenho de um protocolo. Neste caso, as particularidades da robótica e do controle de topologia são utilizadas como limitações ou desafios, mas o foco e métricas de avaliação, são direcionados para a rede/protocolo, não sendo relevante o resultado da tarefa cooperativa. Exemplo: avaliação da taxa de perda e recuperação de mensagens de um protocolo de redes de comunicação aplicado à uma tarefa de robótica cooperativa;
- **Foco no Controle de Topologia:** estudos que estão mais interessados em avaliar a manutenção da topologia de comunicação.

Não há interesse no resultado da tarefa e nem no desempenho da rede. Exemplo: avaliação da influência dos demais algoritmos na manutenção de uma topologia de comunicação desejada e se esta, é mantida durante toda simulação.

No caso desta tese, dada a restrição de tempo, os estudos foram limitados à análise focada na robótica cooperativa, ou seja, os estudos utilizando a simulação integrada são analisados em relação ao desempenho da tarefa.

Após definir este ponto de partida, segue-se para o primeiro procedimento do método, que é a formulação do problema a ser abordado (procedimento 1). Deve-se especificar a ideia inicial apresentada anteriormente, iniciando a documentação do estudo ao criar um documento de apresentação.

Este documento será complementado nos próximos passos, sendo que nesta etapa o desenvolvedor deve se preocupar apenas em contextualizar o problema, levantar a hipótese (pergunta) do estudo e delimitar o escopo do trabalho.

Exemplo 1- Formulação do problema a partir da ideia inicial.

Considerando que se deseja elaborar um estudo sobre o impacto da rede de comunicação na tarefa cooperativa. É elaborado um documento com os seguintes pontos:

- Primeiramente, tem-se a pergunta inicial do estudo: A rede de comunicação tem impacto em uma tarefa cooperativa?
- Em seguida, é dada uma contextualização sobre o tema, explicando qual é a ligação da rede de comunicação com a robótica e como essa ligação, em teoria, poderia afetar o desempenho dos robôs.
- Junto ao item anterior, visa-se responder as perguntas que surgem em sequência à pergunta inicial. Ou seja, qual o escopo do estudo? Qual a aplicação desejada? Quais os tipos de rede, robôs e tarefa que se deseja abordar? Definida a tarefa, como analisar o seu desempenho?
- O último passo é delimitar o estudo e retirar aspectos que não são desejados. Por exemplo, para este estudo não é necessário modelar, simular e analisar o consumo de energia e nem adicionar imprecisões no movimento e localização do robô.

O escopo dos estudos apresentados nesta tese, se restringe a uma aplicação com as seguintes características:

- **Geral:** foco da análise no desempenho da tarefa cooperativa, cooperação descentralizada, aplicação *indoor*, movimento em 2D;
- **Tarefa:** *rendez-vous* e suas variações;
- **Robôs:** móveis terrestres de rodas com capacidade de auto localização;
- **Comunicação:** troca explícita de mensagens por meio de uma rede sem fio e cada robô possui uma identificação única perante a rede;

- **Suposições iniciais:** localização ideal, movimento ideal, não são considerados o consumo de energia e nem a existência de obstáculos no ambiente.

O próximo procedimento (2) é buscar ou propor técnicas-soluções para os tópicos escolhidos, que satisfaçam os pontos apresentados e documentados anteriormente. Para esta etapa, assume-se que quem esteja desenvolvendo o estudo tenha um conhecimento mínimo sobre o problema, para ser capaz de propor as técnicas.

Não importa para a utilização do método quais as técnicas que serão escolhidas, no entanto, para montar a simulação e obter dados para serem analisados nesta tese, foi necessário desenvolver estudos utilizando uma seleção de técnicas. Assim, utilizou-se como critério a seleção de técnicas que nunca haviam sido combinadas (entre diferentes tópicos), que tinham sido desenvolvidas em outro *software* (para fazer a validação) e de técnicas clássicas da literatura. O resultado desta seleção foram as técnicas detalhadas no Capítulo 2, sendo brevemente listadas abaixo.

- **Tarefas Cooperativas:** *rendez-vous* médio e *rendez-vous* de referência (CORTES, 2006; ORDÓÑEZ, 2013; KHAN, 2014; FENG, 2016; VUNDURTHEY, 2016; SABATTINI, 2016);
- **Estratégia de Controle Cooperativo:** consenso médio, consenso circuncentro (YAMASHITA, 1999) e consenso RHC (ORDÓÑEZ, 2013);
- **Rede de Comunicação:** *Mobile Ad-Hoc Network* (MANET) (AKEWAR, 2012; GIL, 2015; KAHAR, 2012; KUDELSKI, 2013), protocolo UDP/IP e tecnologia Wi-Fi (TANENBAUM, 2011);
- **Protocolos de Rede Adicionais:** RA-TDMA (OLIVEIRA, 2012, 2015) e TDMA (WADUD, 2013; LEONE, 2013);
- **Controle de Topologia:** fixo e dinâmico (DAS, 2007; MI, 2010; NEVES, 2014) e eliminação de redundância;
- **Controle Individual e Deslocamento:** é utilizado o modelo ideal de controle de velocidade e para o movimento do robô é utilizado um equacionamento de primeira ordem (CHOSSET, 2005).

Por fim, no terceiro procedimento (3), são encerrados os documentos iniciados anteriormente, focando em detalhar os objetivos, em fazer as especificações restantes e, principalmente, elaborar um documento listando e descrevendo todas as suposições feitas até o momento.

Esta documentação é importante visto que os documentos deverão ser acompanhados durante toda a etapa de desenvolvimento, sendo alterados quando necessários. O formato da documentação adotado é similar ao descrito em Balci (2013), consistindo dos seguintes pontos: identificação, categoria (procedimento), nome, estudo de caso, descrição resumida e intenção de uso.

Figura 15- Exemplo do cabeçalho de um documento utilizado no método.

ID	mod comunic 03	Nome	Modelagem Comunicativa
Categoria	Planejamento SP	EC	03 – Avaliação da matriz A
Descrição	Detalhamento por meio do protocolo ODD [Adaptado de Grimm (2006, 2010)] do estudo abordado pelo método, descrevendo-o em 7 conceitos. A modelagem consiste de questionamentos e respostas, utilizando apenas linguagem textual.		
Intenção de Uso	Documentar o estudo de forma a facilitar a elaboração de artigos científicos e a replicação do mesmo por terceiros.		

O documento de suposições requer uma atenção especial, dada sua importância e impacto durante o desenvolvimento do estudo. As suposições podem ser divididas quanto às suas aplicações: em relação ao estudo ou em relação a implementação da simulação.

As suposições referentes à implementação da simulação são de conteúdo mais técnico e surgem durante a implementação, referentes ao funcionamento dos códigos, organização, classes, funções, bibliotecas, dentre outros. Neste momento de desenvolvimento, este tipo de suposição deve ser ignorado.

No entanto, as suposições mais importantes para a documentação são as que se referem ao estudo como um todo. São as suposições que adotadas nos conceitos, escolhas de abordagens e as simplificações que foram feitas. Estas ocorrem quando:

- Não há informações suficientes disponíveis a respeito de uma técnica, parâmetro ou aplicação;
- Os requerimentos do estudo são vagos;
- O autor não possui experiência ou conhecimento a respeito do sistema, técnica ou parâmetro das técnicas;
- Deseja-se simplificar algum aspecto do estudo;
- Toda situação onde escolhas são necessárias para avançar no estudo, mas não há como garantir que a escolha feita é a melhor.

Nos estudos realizados no método, foram confeccionados sempre ao menos dois documentos (durante a etapa inicial): o de apresentação e o de suposições. O documento de introdução, foi dividido em três subdivisões, descrevendo:

- **Sistema Cooperativo:** descreve o sistema como um todo, contextualizando o leitor. Descreve a aplicação, ambiente, tarefa cooperativa, contextualiza a relação entre os tópicos e outras questões não abordadas nos demais tópicos;
- **Comunicação:** são descritas as características e funcionalidades da rede de comunicação, do controle de topologia e dos protocolos de rede adicionais;
- **Controle:** é o documento que descreve as características relacionadas ao controle do robô, incluindo o controle cooperativo, o controle individual de velocidade e o modelo de movimento dos robôs;

Estes documentos são utilizados para acompanhamento da simulação e devem ser incorporados às modelagens conceitual e comunicativa (próximo procedimento).

É importante lembrar que as documentações devem ser informais, constantemente verificadas e alteradas se necessário, com intuito de auxiliar no planejamento e desenvolvimento do estudo simulado.

4.2 PLANEJAMENTO DA SIMULAÇÃO

A partir da documentação anterior, inicia-se o planejamento detalhado do *framework* de simulação. Este é a implementação da primeira simulação, denominada simulação padrão, que irá ser utilizada como referência para as comparações realizadas durante os estudos de caso e que, portanto, deve atender os requisitos e objetivos listados inicialmente. Contudo, a transição das ideias conceituais para a implementação da simulação não é algo trivial e torna-se mais complexa ao considerar que a simulação integrada engloba conceitos multidisciplinares e que pode ser implementada por múltiplos desenvolvedores.

Para auxiliar nos pontos citados, no quarto procedimento do método IRoNS (Procedimento 4), são utilizadas as modelagens conceitual e comunicativa para o planejamento e documentação do *framework* de simulação.

4.2.1 Modelo Conceitual

Na modelagem conceitual, detalhada no capítulo anterior (3.2.1), não há preocupação com o modelo computacional do sistema e sim em apresentar, de forma objetiva, os principais conceitos e técnicas utilizadas

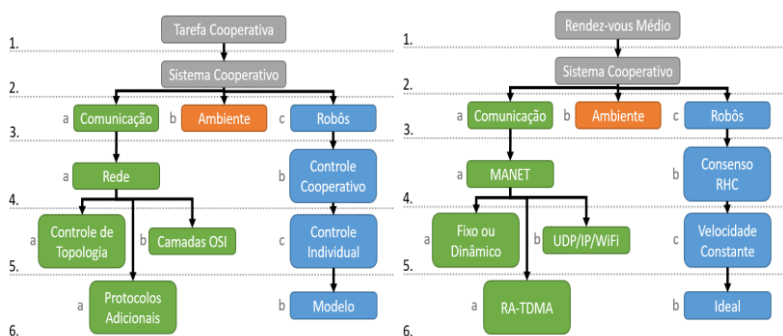
durante a simulação. Esta apresentação deve ser simples e informal, sendo realizada por meio de diagramas, figuras, mapa de conceitos ou de outras formas similares.

A extensão da modelagem conceitual varia conforme o tipo de estudo que será realizado, pois se houver a necessidade de adicionar uma variedade grande de técnicas e características, implicará em uma simulação mais complexa e com maior número de explicações.

Ao longo deste trabalho são utilizados diagramas retirados da modelagem conceitual dos estudos realizados, como por exemplo:

- **Diagrama Estrutural:** representa a estrutura da simulação, indicando quais técnicas são utilizadas e a ordem para detalhamento das mesmas (Figura 16);
- **Diagrama de Interações:** indica as relações e as trocas de informações entre os algoritmos descritos no diagrama anterior;
- **Diagrama de Estrutura de Rede:** é utilizado na representação da estrutura da rede, indicando quais técnicas são utilizadas em cada camada de rede (Figura 5);
- **Diagrama de Funcionamento do RA-TDMA:** indica o princípio de funcionamento da técnica RA-TDMA, auxiliando na sua compreensão (Figura 7);
- **Diagrama de Funcionamento do Controle de Topologia:** utilizado para indicar o funcionamento do controle de topologia simplificado aplicado ao *rendez-vous* (Figura 3).

Figura 16 - Estrutura da simulação e suas principais definições.



Cada conceito modelado deve ser acompanhado de uma breve explanação sobre o que o mesmo representa, de legendas sobre as siglas utilizadas e uma breve descrição sobre as interações com outros

diagramas. Observando o Exemplo 2, é evidente a importância do texto para melhor compreensão do diagrama.

Exemplo 2- Explicação da modelagem a respeito das técnicas e escopo da simulação.

O diagrama (Figura 16) representa a estrutura mínima da simulação integrada de robótica cooperativa e de redes de comunicação que o estudo deverá ter. Neste exemplo são apresentados dois diagramas: na esquerda é a representação genérica do que cada caixa representa e na direita os algoritmos/técnicas utilizadas no estudo em questão. Ainda, o diagrama é dividido por níveis que abordam aspectos cada vez mais específicos de cada tópico. A descrição de cada nível e de cada caixa devem estar no documento, onde seguem as seguintes orientações por nível:

- **Nível 1:** representa-se inicialmente a tarefa cooperativa escolhida. São listadas as características definidas anteriormente, como por exemplo, o tipo de tarefa, objetivos e limitações da tarefa;
- **Nível 2:** neste nível deve-se determinar as características do sistema cooperativo. Por exemplo, características como o número de robôs, posições iniciais, tipo de cooperação, aspectos desejados na simulação, dentre outras características;
- **Nível 3:** no terceiro nível são descritas as características gerais dos tópicos abordados na simulação e as informações sobre o ambiente da simulação. Neste exemplo, são descritas as características de comunicação na simulação (e.g.: tipo de comunicação e tipos de mensagem trocada), o ambiente o qual o robô estará inserido (e.g.: obstáculos, interferências, limitações) e as características dos robôs propriamente ditas (e.g.: heterogêneos / homogêneos, sensores, funcionalidades, capacidades);
- **Nível 4:** após a listagem das características gerais, listam-se as características específicas da rede de comunicação e da estratégia de controle cooperativo escolhidas;
- **Nível 5:** na quinta etapa, é utilizada a representação baseada no modelo OSI para descrever as camadas da rede, listando os protocolos utilizados. Também é especificado o algoritmo de controle de topologia e o de controle individual presente em cada robô (controle de velocidade);
- **Nível 6:** por último, são definidos os protocolos de rede adicionais (se presentes) e o modelo de deslocamento do robô.

O tamanho da modelagem conceitual e o seu grau de detalhamento é subjetiva, dependendo do autor e dependendo principalmente se há ou não outras pessoas envolvidas na implementação e seus respectivos conhecimentos sobre as técnicas utilizadas.

Levando este ponto em consideração, somado ao fato de que uma documentação detalhada é necessária para posterior reprodução dos resultados por outros pesquisadores, é recomendada a adoção e elaboração da modelagem comunicativa para estes fins, mantendo a modelagem conceitual objetiva e com o único intuito de auxiliar na implementação.

4.2.2 Modelo Comunicativo

O modelo comunicativo age como extensão do modelo conceitual, porém com o enfoque na comunicação de informações básicas sobre o

estudo e a simulação. Para definir quais são as informações necessárias e que devem necessariamente constar na documentação é utilizado o protocolo ODD (*Overview, Design concepts and Details*) (GRIMM, 2006 e 2010; RAILSBACK, 2012), detalhado na subseção 3.2.1, o qual foi originalmente utilizado em soluções de modelagem e simulação de sistemas baseados em agentes (ABMS).

Não há um consenso entre os pesquisadores sobre a definição do que é um *agente* e por consequência, do que consiste um sistema multi-agente (INGHAM, 1999). De forma geral, é dito que o agente é uma entidade capaz de controlar suas próprias ações de forma autônoma e que pode possuir características como comunicação, cooperação, aprendizagem, mobilidade, negociação e simulação.

Para fins de praticidade, é utilizada a definição de Russel e Norvig (2010) para relacionar agentes e robôs, na qual afirma que um robô pode ser considerado um caso de agente físico.

Corroborando com esta definição, o escopo abordado neste trabalho considera outras características comuns entre um sistema multi-agentes e robôs cooperativos, como a capacidade de comunicação, de cooperação e de mobilidade.

Assim, como o protocolo ODD foi desenvolvido para ABMS, apenas foi necessário ajustar terminologias e incluir os questionamentos com as particularidades da simulação integrada.

No entanto, devido a sua extensão, é inviável apresentar o seu conteúdo para cada estudo de caso realizado neste trabalho. Assim, abaixo são listadas as principais alterações realizadas nos questionamentos de cada categoria do protocolo ODD e um simples exemplo de aplicação é apresentado no ANEXO B.

- **Propósito:** utiliza-se neste ponto as informações já documentadas nos passos iniciais do método, adicionando informações a respeito do propósito específico da simulação.
- **Entidades, variáveis de estado e escala:** para o caso da robótica este item se torna mais simples, pois não há muita diversidade a ser detalhada: os robôs apresentam características semelhantes mesmo para o caso heterogêneo, o ambiente é simplificado e poucos casos possuem entidades adicionais.
- **Panorama do processo e agendamento:** nesta etapa define-se a interação entre as entidades, focando principalmente na interação entre os robôs. Apresenta-se o robô em função de submodelos, indicando a sua função e suas interações (e.g.: divisão do robô em módulos de comunicação, controle de topologia, controle

cooperativo, controle individual e mobilidade). Este item possui grande importância para a simulação integrada, visto que deve ser listada (e posteriormente descrita) a interação entre as técnicas das diferentes áreas integradas.

- **Conceitos de projeto:** o protocolo ODD define 11 conceitos de projetos que não são obrigatórios e devem ser analisados conforme o caso.
 - **Princípios básicos:** conceitos, teorias, hipóteses, abordagens, relações, dentre outras características relacionados ao uso da simulação e sua relação com o objetivo do estudo;
 - **Emergência:** é a discussão de resultados ou comportamentos que podem surgir de forma inesperada, por uma combinação complexa de parâmetros ou de outros fatores como uma condição inicial específica. Este ponto sempre deve ser analisado ao se tratar da simulação integrada, pois se deseja justamente verificar se há interação de algoritmos distintos e de seus respectivos parâmetros e qual o impacto da mesma nos resultados.
 - **Adaptação:** propriedades adaptativas são utilizadas neste trabalho, mas existem algoritmos de controle que podem incorporar alternância de comportamento dada a ocorrência de certas condições. Por exemplo, o robô pode utilizar um algoritmo mais conservador quando o mesmo é um ponto crítico na topologia de comunicação ou quando está no limite de comunicação com um de seus vizinhos.
 - **Objetivos:** a função objetivo de cada robô é determinada principalmente pelo controle cooperativo e a tarefa a ser realizada. É importante tornar explícito o objetivo dos robôs, como é medido ou determinado o sucesso deste objetivo e quais os critérios utilizados;
 - **Aprendizagem:** não há alterações neste conceito;
 - **Predição:** refere-se ao detalhamento dos processos e de parâmetros preditivos dos robôs, caso presentes;
 - **Sensoriamento:** são as características de percepção do robô, de si mesmo (sensores proprioceptivos) e do ambiente (exteroceptivos), definindo o que é medido e para qual finalidade;

- **Interação:** na interação são descritas as propriedades da estrutura de comunicação entre os robôs, detalhando os protocolos e tecnologias utilizadas, quais informações são trocadas pelo robô e onde são utilizadas;
 - **Estocasticidade:** não há alterações neste conceito;
 - **Coletividade:** a coletividade é um fator presente em todos os casos do escopo deste trabalho. Deve-se detalhar o controle cooperativo dos robôs, seus parâmetros e características;
 - **Observação:** o último item dos conceitos de projeto, é a definição de como os dados coletados da simulação integrada são utilizados para teste, compreensão e análise dos resultados;
- **Inicialização:** essa etapa caracteriza-se pela descrição dos componentes iniciais e das condições iniciais do sistema, focando principalmente na posição inicial dos robôs, seus parâmetros iniciais e em sua topologia de comunicação inicial;
 - **Dados de entrada:** não há alterações neste conceito;
 - **Submodelos:** é a descrição em detalhes dos submodelos citados no “panorama do processo e agendamento”. As divisões que não foram abordadas nos itens anteriores devem ser descritas aqui, como o controle individual do robô, o modelo de mobilidade, consumo de energia, dentre outros.

4.3 IMPLEMENTAÇÃO DA SP

Terminada a fase de documentação do estudo e da simulação, inicia-se a implementação da simulação integrada. Os aspectos técnicos da implementação realizada no OMNeT++ serão tratados no próximo capítulo. Aqui são tratados os aspectos conceituais sobre o acompanhamento da implementação e a sua relação com a análise dos estudos de caso.

A primeira implementação desta simulação em um *framework* é denominada de simulação padrão no método IRoNS, pois os resultados desta simulação serão utilizados para comparação com os resultados obtidos nos estudos de caso implementados posteriormente.

Esta simulação deverá ser construída em um *software* adequado e compatível com os objetivos e o foco do estudo. Como discutido anteriormente, neste trabalho é utilizado o simulador de eventos discretos

OMNeT++ junto ao Add-on INET (INET, 2016), focando na avaliação do impacto das técnicas utilizadas nos resultados da tarefa cooperativa.

Um conceito associado à implementação é a garantia de qualidade da Modelagem e Simulação (M&S), cujo objetivo é o desenvolvimento de uma M&S que possua uma qualidade aceitável (BALCI, 2013). A quantificação desta qualidade é verificada ao aplicar diferentes indicadores de qualidade, como por exemplo: dependência, funcionalidade, interoperabilidade, desempenho, portabilidade, suporte, usabilidade e, principalmente, pela precisão.

É recomendado que a qualidade não seja tratada de forma binária, ou seja, que não seja tratada apenas como correta/incorrecta. Deve-se buscar formas de avaliá-la em diversos níveis, sendo que a melhor classificação deve corresponder a um comportamento real da simulação (e não um valor idealizado). Balci (2013) propõe elaborar uma métrica qualitativa da M&S baseado em notas quantitativas atribuídas de acordo com a demanda/objetivo desejado (Quadro 5). Esta métrica deve ser elaborada antes de se obter valores dos indicadores e deve corresponder corretamente ao valor desejado da M&S, sendo que jamais deve ser adaptada *a posteriori*, dado que é a M&S que deve se adequar aos valores desejados.

Quadro 5 – Exemplo de classificação da simulação padrão de acordo com indicadores de qualidade.

Pontuação Nominal	Pontuação Numérica	Descrição
Excelente	[90...100]	M&S atende a todos os requerimentos para um indicador em particular ou um conjunto de indicadores.
Muito Boa	[80...90)	M&S satisfaz maioria dos requerimentos para um ou mais indicadores de qualidade.
Boa	[70...80)	M&S tem vários requerimentos para um ou mais indicadores de qualidade.
Marginal	[60...70)	M&S falha em satisfazer maioria dos requerimentos de um ou mais indicadores de qualidade
Deficiente	[40...60)	M&S é deficiente em atender os requerimentos para um ou mais indicadores de qualidade.
Não satisfatório	[0...40)	M&S é inaceitável em relação a um ou mais indicadores de qualidade.

Fonte: Traduzido de Balci (2013)

Recomenda-se também utilizar o termo “suficiente” para descrever características de qualidade e de validação da simulação, utilizando, por exemplo, a expressão “simulação é suficientemente válida”. Isto visa implicar que a validação foi julgada, a respeito de um conjunto de intenções de uso, ser apta o suficiente.

O principal indicador da qualidade da simulação é a precisão da M&S. Esta é obtida por meio do processo de verificação e validação (V&V), discutida em maiores detalhes no próximo item. Para avaliar a simulação, foram criados os seguintes indicadores de qualidade (A a F) e seus respectivos pesos na nota final:

- **A - Precisão em relação ao ponto de convergência final do *rendez-vous* (0 a 40):** quanto mais próximo de zero for a diferença entre os pontos de convergência final da estratégia de controle simulada no MATLAB e no OMNeT++, maior a pontuação obtida. São utilizados os resultados finais do processo de validação por IC para estabelecer este valor;
- **B - Precisão em relação ao tempo de simulação até o *rendez-vous* (0 a 40):** similar ao anterior, mas em relação ao tempo de simulação até o *rendez-vous* ser concluído;
- **C - Modularidade e presença dos algoritmos (0 a 5):** indica se a simulação foi elaborada de forma modular e quais algoritmos estão disponíveis para serem facilmente ativados/desativados nas categorias de controle cooperativo, controle individual, controle de topologia e protocolos adicionais de rede. É atribuída maior nota para uma melhor modularidade;
- **D - Modelagem do movimento e comunicação (0 a 5):** indica quão reais ou ideais são os algoritmos de movimentação e comunicação utilizados, considerando um robô móvel com rodas e comunicação *wireless*. Quanto mais próximas das ideais, menor a pontuação;
- **E - Observação dos parâmetros de avaliação (0 a 5):** indica se é possível observar os parâmetros de avaliação desejados (dada a tarefa de *rendez-vous* e os objetivos do estudo) e se os mesmos são obtidos diretamente ou por meio de manipulações. Quanto maior a disponibilidade e facilidade, maior a pontuação;
- **F - Modelagem do ambiente (0 a 5):** indica o modelo do ambiente utilizado, se é ideal ou possui características reais (e.g.: colisões de mensagens, colisões físicas entre robôs, obstáculos e variação no alcance de comunicação), tendo como base, um ambiente *indoor*.

O somatório destes indicadores resulta em uma nota final, na qual há uma correspondente qualitativa (Quadro 6). Espera-se que a simulação obtenha no mínimo uma pontuação “boa” para prosseguir com o método.

Quadro 6 – Notas para classificação da simulação padrão de acordo com indicadores de qualidade.

Pontuação Nominal	Pontuação Numérica	Descrição
Excelente	[90...100]	Simulação reproduz com fidelidade o comportamento de robôs cooperativos realizando <i>rendez-vous</i> por meio de uma rede de comunicação sem fio em um ambiente <i>indoor</i> .
Muito Boa	[80...90)	Simulação reproduz com boa precisão tanto o comportamento dos robôs quanto o ambiente e a integração dos algoritmos.
Boa	[70...80)	A simulação é suficientemente precisa para prever o comportamento dos robôs cooperativos para ambientes controlados.
Marginal	[60...70)	A simulação não apresenta consistência nos resultados, necessitando melhorias.
Deficiente	[40...60)	A simulação é insuficiente para prever algo.
Não satisfatório	[0...40)	A simulação é inadequada e/ou possui erros graves.

As notas foram criadas desta forma para priorizar a precisão dos resultados obtidos na simulação, sendo as demais notas indicadores do grau de realismo abordado na simulação em questão (Exemplo 3). No entanto, para obter dados relativos à precisão é necessário definir quais são as métricas de avaliação utilizadas para analisar os resultados da tarefa cooperativa (discutidas no próximo subitem).

Exemplo 3 – Classificação da simulação por indicadores de qualidade

Para a primeira simulação implementada no método, foram comparados os resultados obtidos em uma simulação do Matlab com a simulação do OMNeT++, referentes à tarefa cooperativa de *rendez-vous*. Utilizando a classificação de notas do Quadro 6, a simulação obteve classificação “Muito Boa”, devido às respectivas notas numéricas: A - 39, B - 36, C - 4, D - 0, E - 5 e F - 0, totalizando 84 pontos. Nota-se que a implementação possui uma excelente precisão, no entanto, a utilização de considerações ideais penaliza a mesma.

4.3.1 Métricas para Avaliação

A escolha das métricas para avaliação é essencial e está diretamente ligada a tarefa que os robôs estão realizando e ao que se deseja estudar (KUDELSKI, 2013). É necessário constantemente rever o objetivo do estudo e avaliar se os atuais parâmetros de análise permitem concluir algo relacionado ao mesmo. Os parâmetros de avaliação podem ser (LAW, 2015):

- **Quantitativos ou qualitativos:** se o parâmetro pode ou não ser quantificado em números. Por exemplo, o tempo total de

simulação é um parâmetro quantitativo e o parâmetro que indica o sucesso ou não da tarefa cooperativa é qualitativa;

- **Controláveis ou não controláveis:** se são parâmetros que podem ser traduzidos em ações no sistema real ou não.

Exemplo 4 - Documentação das métricas de avaliação e parâmetros.

Considerando um estudo onde será avaliada a relação entre parâmetros de conectividade da comunicação e o desempenho da tarefa cooperativa de *rendez-vous*, deseja-se observar o comportamento da matriz adjacência e de sua laplaciana, por meio dos valores de *rank* e de seus autovalores. É questionado se os autovalores da laplaciana da matriz adjacência (utilizados para determinar a conectividade da rede [CARVALHO, 2015]) possuem relação direta com o desempenho da tarefa cooperativa ou não.

Documentação

a) Métricas

- **Tarefa:** *Rendez-vous* Médio
 - *Qualitativa:* sucesso ou não da tarefa cooperativa. Define-se um *Rendez-vous* como bem-sucedido quando todos os robôs convergem para um único ponto, independentemente do tempo.
 - *Quantitativa:* ponto de convergência final (X_F, Y_F) , tempo de convergência final T_F .
- **Conectividade:** Avaliação da matriz adjacência
 - *Quantitativa:* distância média inicial dos robôs conectados, segundo menor autovalor não-nulo da laplaciana (CARVALHO, 2015), somatório do inverso de n-1 autovalores não nulos (CARVALHO, 2015).

b) Parâmetros

Parâmetros Utilizados: coordenadas de convergência (X_F, Y_F) , coordenadas das posições dos robôs (x_i, y_i) , tempo que levou para convergência (T_F) , matriz adjacência (A), laplaciana da matriz adjacência (L_A) , autovalores da laplaciana (λ_L) , diferentes controles de topologia (Fixo e Dinâmico).

c) Informações Adicionais

Validação: em relação à tarefa cooperativa, resultados obtidos em uma simulação no Matlab é comparada com os resultados obtidos no OMNeT++ sem adição de rede de comunicação. As métricas utilizadas são a diferença entre o ponto (Df) e o tempo de convergência (Tf) finais entre as simulações.

$$\text{Erro } D_F = \sqrt{([X_F^{MAT} - X_F^{OMN}]^2 + [Y_F^{MAT} - Y_F^{OMN}]^2)}$$

$$\text{Erro } T_F = T_F^{MAT} - T_F^{OMN}.$$

Para avaliação e validação geralmente são utilizados parâmetros que remetem a um resultado final da simulação, como por exemplo, o parâmetro *tempo total de duração* da tarefa do robô. Assume-se que há um evento que sinaliza o final da simulação e utiliza-se o valor do parâmetro de avaliação deste instante.

Em Law (2015) e Robinson (2014), são discutidas também outras abordagens para parâmetros de avaliação, para simulações que não possuem este evento de finalização, como por exemplo, no caso de acompanhamento de trajetórias de tempo indeterminado. Nestas simulações, é necessário manipular o parâmetro de avaliação, evitando por exemplo, incluir valores iniciais da simulação quando há presença de um transitório não relevante ao funcionamento do sistema. Estas abordagens podem ser classificadas de acordo com o procedimento utilizado:

- **Procedimentos com tamanho de amostragem fixa:** é realizada uma única execução da simulação para então utilizar uma técnica para construir a validação;
- **Procedimentos sequenciais:** a duração da execução da simulação é sequencialmente incrementada até atingir uma confiabilidade aceitável no processo de validação.

Outra possível abordagem em simulações é a utilização de técnicas de loteamento de médias de uma variável/parâmetro, das quais possuem um mesmo princípio em comum: dividir uma simulação longa em lotes e utilizar o valor médio e variância de cada lote como se fossem simulações individuais (*batched means*) (LAW, 2015; ROBINSON, 2014). Segundo Banks (2004), ao utilizar esta abordagem, recomenda-se utilizar entre 10 e 30 lotes, visto que um número inferior apresentará um resultado pouco satisfatório e acima desta faixa, os ganhos são insignificantes.

No contexto desse trabalho, como os robôs sempre estão realizando a tarefa de *rendez-vous*, o foco é escolher métricas que avaliem como a tarefa está sendo afetada. Neste sentido, é avaliado se todos os robôs convergem ou não para um determinado ponto, o tempo decorrido até esta convergência e quais as coordenadas finais deste ponto.

Todas as métricas de avaliação são documentadas (Exemplo 4) e implementadas para posterior utilização na validação e análise dos resultados.

4.4 VALIDAÇÃO DA SP

Após concluir a etapa de implementação da simulação, a mesma passa a ser denominada de simulação padrão (SP) candidata, devido a necessidade de sua validação. Como mencionado na subseção 3.2.2, para o processo de validação é considerado que cada técnica pode ser validada individualmente e que ao combinar as mesmas, a validade não será perdida.

No caso da simulação integrada deste trabalho, há necessidade de verificar e validar, ao menos, os seguintes conceitos: a rede de comunicação, o controle de topologia e o controle cooperativo. No entanto, se houver adição de outras técnicas em relação a outros conceitos, as mesmas também devem ser validadas em relação ao seu correto funcionamento no ambiente do OMNeT++.

Dentre os três conceitos mencionados, a rede de comunicação utilizada é a rede MANET padrão do OMNeT++/INET, usada em diversos trabalhos com OMNeT++ ao longo dos anos e que, por este mesmo motivo, supõem-se que é suficientemente válida.

Em relação ao controle de topologia e controle cooperativo, as técnicas são validadas quanto ao correto funcionamento, ao comparar resultados obtidos em implementações no Matlab com os obtidos na simulação do OMNeT++, sob as mesmas condições iniciais e de operação.

Esta validação ocorre em quatro passos, baseada na solução de Klugl (2010) para ABMS (detalhado nos itens 3.2.2 e 3.2.3), consistindo em:

- **Validação de Face:** é realizada visualmente por um verificador humano. Observam-se primeiramente as animações e gráficos da simulação em busca de inconsistências, para em seguida verificar se os resultados de saída da simulação em busca de grandes alterações de valores. Ao final, verifica-se de forma imersiva cada componente da simulação (e.g.: robôs), seu comportamento e os valores de seus parâmetros.
- **Análise de Sensibilidade:** a segunda etapa é alterar os valores de certos parâmetros e analisar se o comportamento da simulação é alterado da mesma forma em ambas as simulações.
- **Calibração:** esta etapa consiste em realizar ajustes finos entre as simulações para que seus resultados estejam mais próximos. Também é recomendado reavaliar os valores dos parâmetros, determinando se estes são adequados para a simulação padrão.
- **Validação Estatística:** a validação por intervalo de confiança (subseção 3.2.3) é a última validação necessária, onde é determinado se as diferenças entre as simulações são estatisticamente significativas ou não.

Para a validação estatística é utilizado a técnica por intervalo de confiança (IC), discutida na subseção 3.2.3.

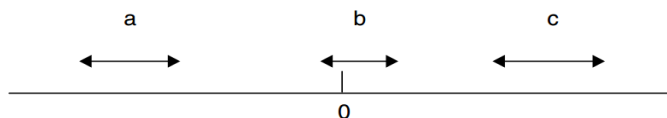
Para fins de padronização, o erro é obtido pelo resultado da simulação de referência subtraído pelo resultado da simulação a ser validada/avaliada.

Para ser considerada válida, o erro entre as simulações deve gerar um IC pequeno em torno de zero. Ou seja, é possível determinar a validade da simulação ao analisar o IC em relação a dois quesitos: quanto ao deslocamento do IC em relação ao zero e quanto a sua extensão do intervalo propriamente dito.

No primeiro quesito, há três possíveis interpretações dos resultados (ROBINSON, 2014; LAW, 2015) (Figura 17), dadas as seguintes situações:

- a) **IC está deslocado a esquerda do zero:** pode ser concluído com certo grau de confiança em que a média verdadeira do sistema 1 (simulação de referência) é menor que a média verdadeira do sistema 2 (simulação em validação);
- b) **Inclui o valor de zero:** conclui-se com certo grau de confiança que a média verdadeira do sistema 1 (simulação de referência) não é significativamente diferente da média verdadeira do sistema 2 (simulação em validação);
- c) **IC deslocado a direita do zero:** conclui-se com certo grau de confiança que a média verdadeira do sistema 1 (simulação de referência) é maior que a média verdadeira do sistema 2 (simulação em validação).

Figura 17 - Possíveis resultados da construção do IC.



Fonte: Robinson (2014)

É importante observar que a diferença estatística significativa não é o mesmo que afirmar que há uma significância prática (BANKS, 2004; ROBINSON, 2014). A significância estatística fornece resposta ao questionamento sobre a diferença na média verdadeira estimada pelos resultados.

A significância prática é obtida analisando o segundo quesito, onde é observado se os valores do intervalo são suficientemente grandes para afetar decisões práticas.

A simulação do OMNeT++ será considerada válida se o intervalo de confiança for relativamente pequeno e incluir zero. Para exemplificar numericamente como é realizado este procedimento de validação, é apresentado no próximo subitem o relato da primeira validação da simulação integrada, com foco no controle cooperativo e no controle de topologia.

4.4.1 Exemplo – Processo de Validação

A primeira simulação integrada que passou pelo processo de validação utiliza as técnicas de consenso RHC para *rendez-vous* (subseção 2.3.3), uma MANET como rede de comunicação (seção 2.4) e um controle de topologia simplificado.

Como mencionado anteriormente, a MANET utilizada é a padrão do simulador e, portanto, a mesma foi considerada válida para o estudo. Portanto, o foco da validação é o controle cooperativo e o controle de topologia. Esta validação é necessária tanto pela situação das simulações terem sido desenvolvidas por diferentes autores, quanto por utilizar diferentes linguagens de programação, bibliotecas de otimização quadrática e diferentes estruturas de execução.

A simulação utilizada como referência é a simulação desenvolvida pelo autor (ORDONEZ, 2013) no Matlab com adição do controle de topologia simplificado. A validação é realizada entre esta e a nova implementação no OMNeT++, usando como métrica de avaliação, se o *rendez-vous* foi concluído com sucesso (todos os robôs em um mesmo ponto), as coordenadas do ponto final de convergência do *rendez-vous* e o tempo total até a convergência.

Após concluir a implementação no OMNeT++, ambas simulações foram ajustadas para ter os mesmos valores, incluindo o número de robôs, alcance de comunicação, velocidade máxima de deslocamento, dentre outros parâmetros.

O processo de validação inicia-se pela comparação simples dos resultados de ambas simulações, considerando a mesma condição inicial (posição inicial dos robôs).

Nesta etapa, foram encontradas grandes diferenças entre as simulações, inicialmente devido a erros de implementação. Após as correções, os resultados ainda apresentavam diferença, requerendo uma melhor análise sobre a diferença entre as simulações.

Foi observado que na implementação no Matlab os robôs tratavam os dados simultaneamente a cada instante, diferente da implementação no OMNeT++. Constatada a fonte dos erros, as simulações foram novamente

adaptadas para tratarem as informações da mesma forma, atingindo resultados próximos.

Na segunda fase da validação, os parâmetros foram alterados para monitorar se os resultados sofriam alterações em ambas as simulações. Como os resultados foram promissores, não houve necessidade de alterações nas simulações.

Da mesma forma, a fase de calibração não necessitou de ajustes entre as simulações. No entanto, a calibração foi útil para refinar os valores dos parâmetros, no sentido de escolher valores mais conservadores para serem tratados como “padrão”.

Dentre as alterações, foi ajustado o tamanho do alcance de comunicação dado o tamanho do ambiente desejado, visando que o mesmo não seja muito curto a ponto de os robôs precisarem estar muito próximos uns dos outros, mas também nem muito longo a ponto da comunicação alcançar todos os outros robôs simultaneamente.

Também foi ajustado o número total de robôs, sendo escolhido utilizar sempre 10 para poder ter uma melhor variedade de topologias iniciais de comunicação.

Além destas alterações foram redefinidos alguns valores do consenso RHC, como os valores dos pesos utilizados na função custo, o tamanho da janela de predição e as restrições de velocidade máxima do robô. Um resumo dos principais parâmetros utilizados é apresentado no Quadro 7.

Quadro 7 - Parâmetros utilizados para validação.

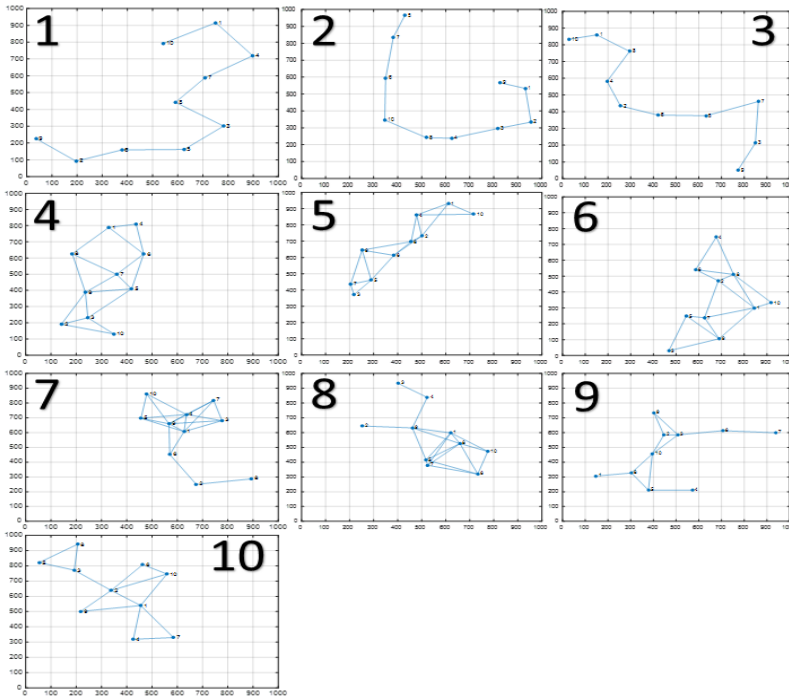
Número de Robôs (N)	10
Saturação de Velocidade	10 cm/s
Raio de Comunicação	250 cm
Janela de Predição	10
Peso do Link e de ΔV (α, λ)	1 e 0,1
Tempo de amostragem do controle e do ciclo completo do TDMA.	100 ms
Controle de Topologia	Fixo e dinâmico
Atualização do movimento	1 ms
Tempo <i>SLOT</i> TDMA	10 ms

A última etapa da validação é a análise estatística dos resultados, a qual necessita de um conjunto de amostras de ambas as simulações. As “amostras” nada mais são do que diferentes resultados da simulação ao ser submetida a diferentes condições iniciais (CIs), mas utilizando as mesmas condições de operação. No caso da tarefa cooperativa de *rendez-vous*, as amostras são geradas alterando a posição inicial dos robôs,

afetando a sua topologia de comunicação inicial e, por consequência, alterando os resultados finais de convergência do *rendez-vous*.

Quanto ao número de amostras, Law (2015) recomenda utilizar ao menos de 5 a 10 simulações diferentes para avaliar os resultados. Assim, foi escolhido criar 10 CIs, visando uma maior diversidade dos resultados sem demandar um exaustivo número de simulações.

Figura 18 – Topologias iniciais dos dez casos utilizados para validação.



Para escolher as CIs, foi elaborado um algoritmo para sorteio, onde pode ser utilizado certos critérios durante a escolha. São geradas dez CIs (Figura 18), utilizando os seguintes critérios:

- **Árvore Geradora Mínima:** é o principal critério, sendo utilizado na formulação de todas as CIs. É importante que sempre haja ao menos um caminho (neste caso, bidirecional) para que a informação flua de um robô a qualquer outro robô da rede, ou seja, os robôs devem estar conectados a todos os outros robôs de forma direta ou indireta, bidirecionalmente.

- **Somente Árvore Geradora Mínima:** foram sorteadas 3 CIs onde somente é permitida a árvore mínima (bi-direcional), ou seja, os robôs apresentam no máximo dois links de comunicação.
- **Conectividade entre 2 e 5:** foram sorteadas 3 CIs onde cada robô poderia ter no mínimo dois e no máximo cinco vizinhos diretos.
- **Sem restrições:** por fim, foram sorteadas 4 CIs que obedecem somente ao primeiro critério, mas podem ter qualquer quantidade de vizinhos diretos.

Como as amostras são independentes e a sua quantidade é idêntica para ambas simulações, é possível utilizar ambas formulações para construção do intervalo de confiança.

Os ICs foram construídos para um grau de confiança de 99%, obtendo o ponto crítico da distribuição *t-student* $t_{9, 0,995}$ de 3,250. A média e a variância do erro, foram realizadas sobre a distância entre as coordenadas finais e sobre o tempo de convergência, para o caso de topologia fixa e de topologia dinâmica separadamente. Os resultados das 10 amostras de cada simulação são comparadas graficamente na Figura 19.

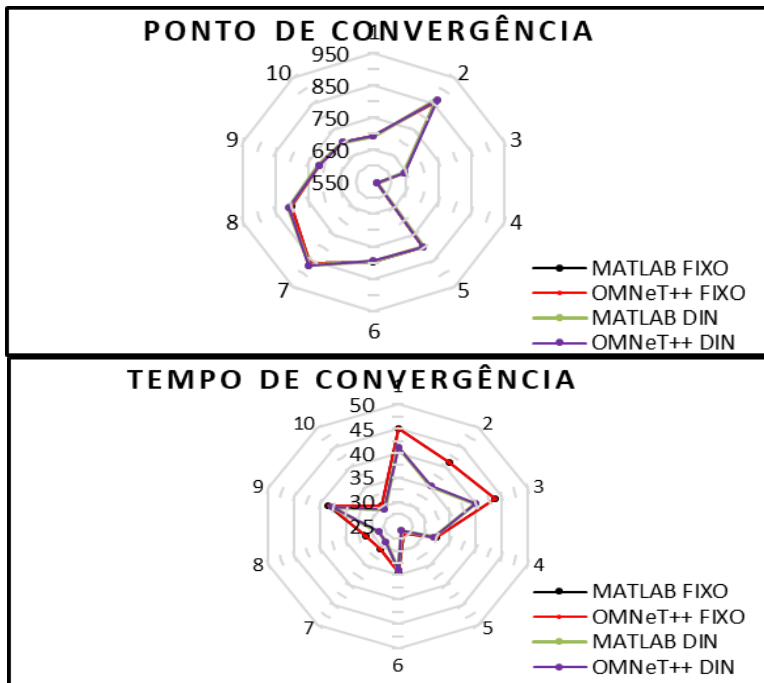
Os intervalos de confiança resultantes são indicados na Tabela 1, onde é possível observar que os intervalos para as duas variáveis, inclui zero e que possui pequenos valores, ao considerar a precisão de centímetros para a distância e que o tempo de convergência médio foi acima de 40 segundos. Pode-se concluir assim, que a simulação é suficientemente válida.

Como são considerados dois parâmetros simultaneamente, é necessário ajustar o grau de confiança total do estudo segundo a regra de Benferroni (LAW, 2015). Portanto, pode-se afirmar que este estudo de validação possui grau de confiança de aproximadamente 98%.

Tabela 1 – Resultados utilizados para validação estatística.

Topologia Fixa	
IC 99% T_f	[-0,016 ; 0,041]
IC 99% D_f	[-0,013 ; 0,0298]
Confiança no Estudo de Validação	~ 98%
Topologia Dinâmica	
IC 99% T_f	[-0,049 ; 0,049]
IC 99% D_f	[-0,128 ; 0,246]
Confiança no Estudo de Validação	~ 98%

Figura 19 – Resultados gráficos do ponto e tempo de convergência de cada condição inicial, para as simulações e topologias utilizadas na validação.



4.5 ESTUDOS DE CASO

O estudo de caso nada mais é do que o planejamento e elaboração de um experimento utilizando a simulação padrão, com objetivo de responder ao questionamento inicial (hipótese) do estudo. Como mencionado anteriormente (subseção 3.2.4), é utilizado o design experimental fatorial 2^k combinado à análise estatística por intervalo de confiança.

Para este tipo de planejamento experimental, busca-se analisar o impacto de parâmetros e algoritmos nos resultados finais da tarefa cooperativa.

O primeiro passo é verificar se as métricas de avaliação utilizadas na validação também são utilizáveis para avaliar os estudos de caso. Neste trabalho, ao focar na tarefa cooperativa de *rendez-vous*, as métricas de avaliação são as mesmas utilizadas na validação da simulação: conclusão ou não do *rendez-vous*, ponto de convergência final do *rendez-vous* e/ou o tempo total para esta convergência.

O segundo passo é determinar se o foco do estudo de caso é um parâmetro quantitativo, qualitativo ou uma mudança de técnica. Em todos os casos, o que está sendo modificado é denominado de *fator*, sendo definidos dois estados representados pelo sinal “-” e “+”. É importante a escolha de dois estados para o parâmetro onde acredita-se que haverá maior diferença nos resultados.

Por exemplo, pode-se analisar o impacto de se dobrar a velocidade máxima do robô (simulação padrão utiliza o valor de 10cm/s), ou seja, utilizar o valor de 20cm/s. Assim, a velocidade máxima é o fator escolhido, 10cm/s é o valor negativo e 20cm/s é o valor positivo.

Entretanto, ao substituir técnicas, é importante utilizar os mesmos valores para parâmetros de ambos (quando aplicado) ou valores correspondentes, a fim de se evitar outras influências nos resultados. Neste caso, cada algoritmo corresponde a um “valor”, sendo o algoritmo original (simulação padrão) considerado como o “-” e o novo como o “+”.

O número mínimo de experimentos é proporcional à quantidade de níveis utilizados para os fatores, da quantidade de fatores analisados e do número de reproduções para cada nível/fator (número de amostras). Por exemplo, em um estudo de dois níveis, três fatores e de 10 amostras, resultará em 2^3 combinações diferentes (oito respostas diferentes) e $2^3 \times 10$ (oitenta) execuções da simulação para gerar o total de amostras necessário.

Os resultados devem ser armazenados na matriz de projeto de cada amostra, como indicado no Quadro 4, determinando o impacto de cada fator e_i^k e o impacto entre fatores $e_i^{k_1 k_2}$.

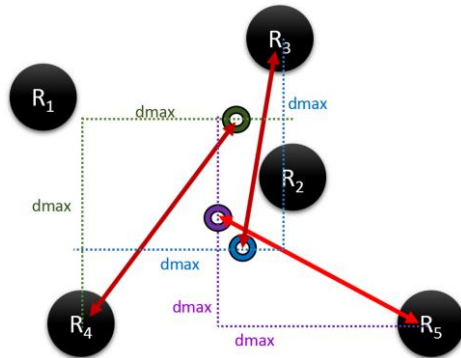
4.5.1 Exemplo – Planejamento Experimental

Neste exemplo, pretende-se estudar o impacto de parâmetros relacionados à conectividade dos robôs no resultado da tarefa cooperativa. As técnicas utilizadas são as mesmas do exemplo de validação (subseção 4.4.1), apenas com a adição da referência na estratégia cooperativa. São avaliados os parâmetros de conectividade (número médio de *links* de comunicação) e o valor da maior distância (*dmax*) entre um robô e a referência.

Para o primeiro fator, foram escolhidas as conectividades mais relevantes (*Mínima*, C2, C3, C5 e *Máxima*), sendo analisadas em pares: *Mínima*xC2, C2xC3, C3xC5, C5x*Máxima*. A conectividade *Mínima*, também conhecida como linear, é formada quando os robôs são conectados sequencialmente e as extremidades possuem apenas um link

de comunicação. As conectividades intermediárias, indicam o número médio de links entre os robôs, sendo a C2 indicação de bi-conectividade (dois caminhos entre quaisquer robôs). Por fim, a conectividade *Máxima* indica que cada robô possui o máximo de links possível, ou seja, nove links de comunicação, dado que são utilizados dez robôs. Desta forma, espera-se verificar se os resultados são afetados pelo constante incremento da conectividade.

Figura 20 - Geração de referência (ponto circular) baseada em um d_{max} que gere ao menos duas outras soluções, em robôs (R) diferentes.



O segundo parâmetro de interesse (d_{max}) é obtido na criação da referência. Os valores de d_{max} são: o maior d_{max} possível dentro do espaço de simulação (valores entre 0 e 1000, para as coordenadas x e y) que gere uma referência-solução para ao menos três robôs e o menor d_{max} possível que tenha uma referência-solução para ao menos três robôs (Figura 20).

Para gerar um número suficiente de amostras para a validação estatística, são utilizadas cinco condições iniciais (CI) distintas e para cada CI são realizados três experimentos, onde nestes é utilizado um d_{max} de mesmo valor para um robô diferente. Isto resulta em 15 amostras para cada experimento realizado.

Os resultados são avaliados em termos de tempo de convergência do *rendez-vous*, onde deseja-se obter o menor tempo de convergência, indicando que a tarefa será concluída mais rapidamente.

Os níveis dos parâmetros são denominados de “-” e “+”, conforme indicado no Quadro 8, onde R corresponde ao tempo de convergência de uma das amostras.

Ao final, são utilizadas as Equações (4.1), obtendo o impacto de d_{max} (e_1^1), impacto da conectividade (e_1^2) e a correlação entre ambos os

parâmetros (e_i^{12}) de cada amostra i . Ainda, como a conectividade está sendo dividida em pares (dois níveis por vez), é utilizada a nomenclatura de E1 para o experimento *Mínimax*C2, E2 para C2xC3, E3 para C3xC5 e E4 para C5x*Máxima*, resultando no total de 4 tabelas para cada amostra.

Quadro 8 - Planejamento experimental fatorial *Dmax* x Conectividade para E1.

Combinação de Fatores	<i>Dmax</i>	Conectividade (MínimaxC2)	Tempo Convergência (R_i^c)
1	-	-	R_i^1
2	+	-	R_i^2
3	-	+	R_i^3
4	+	+	R_i^4

$$\begin{aligned}
 e_i^1 &= \frac{-R_i^1 + R_i^2 - R_i^3 + R_i^4}{2^{2-1}} \\
 e_i^2 &= \frac{-R_i^1 - R_i^2 + R_i^3 + R_i^4}{2} \\
 e_i^{12} &= \frac{R_i^1 - R_i^2 - R_i^3 + R_i^4}{2}
 \end{aligned} \tag{4.1}$$

4.6 ANÁLISE DOS RESULTADOS

Após planejar os experimentos e adquirir todos os resultados tabelados de cada amostra, é realizada a análise dos resultados como indicado na seção 3.2.5.

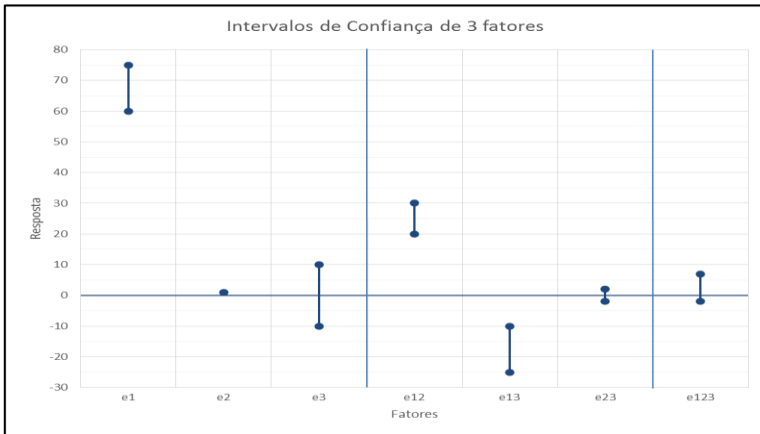
A representação gráfica do IC permite realizar as análises intuitivamente, como indicado no exemplo fictício da Figura 21. Neste exemplo, são construídos ICs a respeito de três fatores (e^1 , e^2 e e^3), seguido pelo impacto pareado dos fatores (e^{12} , e^{13} , e^{23}) e pelo impacto simultâneo dos três fatores (e^{123}). Nota-se que o parâmetro e^1 possui significância estatística (não contém zero em seu intervalo) e um valor elevado (entre 60 e 75). Conclui-se que este parâmetro exerce grande efeito na resposta ao passar de seu nível negativo para o positivo.

O segundo parâmetro está localizado levemente acima do eixo x e também não contém zero, no entanto, a grandeza e variação de seu impacto pode ser considerado insignificante em relação aos demais parâmetros. O IC do terceiro parâmetro denota que não há indicação de que o parâmetro 3 possui impacto nas respostas por si só.

Nas análises pareadas, há indicação de relação entre os fatores 1 e 2, ou seja, há uma interação significativa entre eles e uma atenuação nos

resultados quando ambos são considerados. No segundo par, entre fator 1 e 3, os resultados apontam que apesar do fator 3 não ter impacto isolado nos resultados, o mesmo possui interação significativa com o fator 1. Já o terceiro par indica que não há influência entre os parâmetros 2 e 3 e que ambos podem ser considerados isoladamente um do outro.

Figura 21 – Exemplo de representação gráfica do intervalo de confiança.



Ao final, na análise dos três fatores simultaneamente, o gráfico indica que apesar de uma variação tendendo para o lado positivo do eixo y, o resultado ainda apresenta o zero em seu intervalo e pode ser considerado como uma flutuação nos resultados. Ou seja, não há interação significativa entre os 3 parâmetros.

É importante observar o grau de confiança do estudo não necessariamente corresponde ao grau de confiança utilizado nos cálculos do IC. Ao realizar um estudo com diversos ICs, é necessário ajustar os parâmetros α individualmente, segundo a desigualdade de Benferroni (LAW, 2015).

Para se obter um estudo com determinado grau de confiança G_{total} , a soma dos α de cada IC, denotado aqui por α_s , deverá totalizar o valor de α para G_{total} . Ou seja, para um planejamento experimental de K fatores com $2^K - 1$ ICs, cada intervalo deverá ser construído utilizando (4.2).

$$\alpha_s = \frac{\alpha}{2^K - 1}. \quad (4.2)$$

Para o exemplo do Quadro 4, são construídos $2^3 - 1$ intervalos de confiança para cada impacto ($e^k, e^{k1k2}, e^{k1k2k3}$). Para elaborar um estudo com grau de confiança de 90%, implica em um $\alpha = 0,10$ (total) e por

consequência, um $\alpha_s = \frac{0,10}{7} = 0,014286$ (individual), caso seja utilizado o mesmo valor de alfa para todos ICs.

Isto significa que para construção dos IC individuais, deverá ser utilizado um grau de confiança de aproximadamente 98,57% para que o estudo tenha um grau de confiança total de 90%.

Após se obter e analisar os resultados, deve-se avaliar se os mesmos condizem com o objetivo do estudo, concluindo o ciclo do método. Um exemplo real sobre a análise dos resultados é apresentada na próxima subseção, para melhor compreensão desta análise.

4.6.1 Exemplo – Análise dos Resultados

Utilizando a formulação e os dados do exemplo anterior (subseção 4.5.1), calcula-se a média e variância entre cada um dos três “e” possíveis, de cada um dos quatro experimentos.

O IC é construído para o grau de confiança de 99,5%, resultando em uma confiança de 98,5% para cada experimento.

Os resultados finais obtidos são indicados na Tabela 2 e apresentados graficamente na Figura 22. O gráfico permite a visualização do intervalo e demonstra mais explicitamente os resultados, observando se o mesmo contém zero e o tamanho do mesmo.

Tabela 2 - Valores dos intervalos de confiança de $Dmax$ (e^1), conectividade (e^2) e $Dmax$ x Conectividade (e^{12}) para cada experimento E.

	IC - [Inferior, Superior]		IC - [Inferior, Superior]		
E1e1	59,745	74,481	E2e1	54,421	62,339
E1e2	-26,215	-12,865	E2e2	-3,789	-1,558
E1e12	-14,348	-1,412	E2e12	-2,085	0,378
E3e1	52,980	59,820	E4e1	51,800	57,774
E3e2	-3,712	-1,901	E4e2	-0,981	-0,459
E3e12	-1,964	-0,289	E4e12	-0,833	-0,140

Percebe-se que no primeiro experimento (*MínimaxC2*), tanto o valor de $dmax$ (E1e1) quanto a mudança de conectividade (E1e2) afetam os resultados significativamente e em grandes valores.

O incremento de $dmax$ do valor “negativo” para o “positivo”, afeta o tempo de convergência entre ~60 e ~75 segundos. Isto era esperado dado que ao aumentar $dmax$, aumenta também a distância percorrida pelo

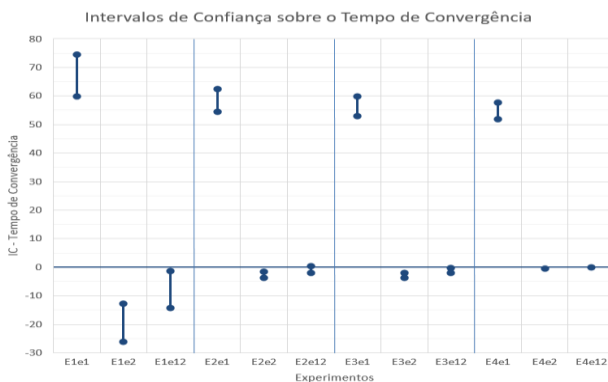
robô durante a simulação. Já o aumento de conectividade de mínima para uma biconectividade, reduz os tempos de convergência entre 13 e 26 segundos, indicando uma melhor performance. No entanto, esses valores têm grande relação entre si, dado o grande valor de E1e12, concluindo-se que os valores individuais não devem ser considerados isoladamente para os casos de baixa conectividade.

No entanto, esta dependência tende gradativamente a desaparecer ao incrementar a conectividade (experimentos E2e12, E3e12 e E4e12), diminuindo também a variação dos resultados do impacto de $dmax$ (E2e1, E3e1 e E4e1) e a significância estatística da conectividade (E2e2, E3e2 e E4e2). Isto denota que o fator conectividade tem menos efeito a cada acréscimo de *link* de comunicação (a cada aumento da conectividade), até o momento onde apenas $dmax$ afeta os resultados (e com menor variação) devido apenas ao incremento na distância percorrida pelo robô ao mudar este fator de seu valor “-” para o valor “+”.

Como os resultados foram satisfatórios, não houve necessidade de iniciar um novo ciclo do método. Conclui-se, portanto, que os resultados auxiliaram a responder o questionamento inicial sobre a relação entre a conectividade e o desempenho dos robôs.

A análise dos resultados também forneceu indicativos sobre os próximos questionamentos a serem realizados sobre o assunto, como por exemplo, o questionamento sobre a possibilidade de um controle de topologia focado em manter a biconectividade ter uma melhor influência nos resultados que os controles de topologia utilizados até o momento.

Figura 22 - Intervalos de confiança sobre a variação dos tempos de convergência resultantes dos experimentos de $dmax$ x *conectividade*.



4.7 CONSIDERAÇÕES FINAIS

Ao longo deste capítulo foi detalhado cada procedimento do método proposto para a simulação integrada de robótica cooperativa e de rede de comunicação, compreendendo desde a formulação do problema até a análise dos resultados simulados.

Foram incorporadas ao método diversas técnicas originadas em diferentes áreas do conhecimento, integrando-as e as adequando para a aplicação proposta.

O método aborda aspectos conceituais e fornece as ferramentas necessárias para construção do framework e das análises a serem realizadas. No próximo capítulo, são apresentados os aspectos de implementação da simulação desenvolvida e utilizada como referência nos estudos de caso apresentados neste documento.

Capítulo 5

Framework de Simulação IRoNS

No capítulo anterior foram analisados os aspectos conceituais e as ferramentas necessárias para uma correta documentação, análise, validação e experimentação utilizando a simulação integrada. Neste capítulo, no entanto, o foco é o framework implementado utilizando a parte inicial do método e que devido a sua modularidade, pode ser reutilizado em diversos estudos de caso.

5.1 SOFTWARE PARA SIMULAÇÃO

O método proposto neste trabalho foi direcionado para o desenvolvimento de estudos utilizando um *framework* de simulação que utilize ao menos três conceitos: 1 - consista de robôs cooperativos realizando uma tarefa robótica, junto com seus respectivos algoritmos de controle individual e cooperativo; 2 – que os robôs estejam em uma rede de comunicação com todas as suas respectivas camadas e protocolos; e 3- que haja um controle de topologia, ou seja, um algoritmo que defina como os *links* de comunicação serão estabelecidos.

Para uma simulação atender estes pontos, é necessário que a mesma seja capaz de lidar tanto com aspectos físicos dos robôs (movimento, sensoriamento, estratégia de cooperação, decisões, etc.) assim como com os aspectos da rede de comunicação (estrutura, protocolos, atrasos, perdas, etc.) e de controle de topologia (criar/destruir links de comunicação, criar topologia lógica, etc.) (LORA, 2015).

As abordagens para esse tipo de simulação consistem em: 1-

utilizar a co-simulação (simultânea) de dois simuladores diferentes (YE, 2001; POHJOLA, 2008; ZHANG, 2010; KUDELSKI, 2013; MI, 2015); 2 - expandir as capacidades de um simulador de rede para lidar com os robôs (KOPKE, 2008; GUAN, 2011; PEKKARINEN, 2014; OLIVEIRA, 2015; LI, 2014; RAMOS, 2015a); 3 - expandir um simulador físico para lidar com a rede de comunicação (NEVES, 2014; GIL, 2015; SABATTINI, 2016); ou 4 - desenvolver um novo simulador com ambas características (PIMENTEL, 2003b; LORA, 2015).

Considerando que criar um novo simulador não é algo trivial e nem recomendado (LAW, 2015), e que se deseja evitar problemas inerentes à co-simulação como os de sincronização (COSTA, 2015), de *overhead* (LORA, 2015) e de integração (LORA, 2015), optou-se em utilizar um simulador com uma rede de comunicação já estruturada e de apenas acrescentar aspectos de robótica cooperativa ao mesmo. Quanto à escolha do simulador de rede, Raj (2013) apresenta uma revisão sobre os principais simuladores de rede, incluindo:

- **NS2:** (NS2, 2016) Um dos mais famosos simuladores de eventos discretos, é um *software* aberto, baseado em OTCL e C++, sendo OTCL um interpretador usado para executar os comandos. É mais concentrado em modelos de simulação e menos na infraestrutura da mesma, possuindo também uma topologia fixa (VARGAS 2008; ZHANG 2010). O simulador carece, no entanto, de customização e não possui modelos hierárquicos (ZHANG 2010).
- **NS3:** (NS3, 2016) Apesar do nome, não é uma extensão do NS2. É um simulador reformulado em C++ e Python, direcionado para pesquisa.
- **OMNET++:** (OMNET++, 2016) É um simulador de eventos discretos com aplicação primária em redes de comunicação, mas possui uma arquitetura genérica e flexível (VARGAS 2001). É baseado em C++, possui um extenso suporte e seus componentes funcionam de forma modular. Suporta formatos customizados de pacotes, interface gráfica, modelos hierárquicos e topologias customizadas e parametrizadas (ZHANG 2010).

Existem diversos trabalhos realizados no intuito de comparar o desempenho destes (e de outros) simuladores para diversas aplicações. Em Khan (2013) é apresentada uma extensa revisão direcionada à simulação de sensores nestes simuladores, comparando desempenho, memória utilizada, tempo de processamento e utilização de CPU. O autor conclui que tanto o NS3 quanto o OMNeT++ são capazes de rodar

grandes redes sem problemas significativos, tendo NS3 um desempenho ligeiramente superior.

Em Erciyés (2010) é realizada uma comparação entre o NS2 e OMNET++, apontado que ambos são opções viáveis em relação à utilização do *software* e de sua comunidade. É mencionado que o principal ponto negativo do OMNET++ é a falta de modelos sobre a influência do ambiente na propagação do sinal e que no NS2 há maior dificuldade em se criar novos protocolos e de operação, quando comparado ao OMNET++. No entanto, o artigo conclui que não há diferenças significativas de desempenho entre os simuladores, mas que há indicativos que o OMNET++ se tornará o simulador de rede mais completo e popular devido a sua comunidade ativa.

Quanto à capacidade de nós na simulação, segundo Kudelski (2013), os simuladores NS2/NS3 conseguem operar com aproximadamente 200 nós sem apresentar problemas, quanto que o OMNET++ comporta algumas centenas de nós a mais. O *survey* realizado por Pujeri (2014) também indica uma mínima diferença entre os simuladores, recomendando o uso do NS3 ou do OMNeT++ por serem mais atuais e por terem uma melhor escalabilidade.

Com base nestes relatos, conclui-se que a escolha do simulador a ser utilizado é subjetiva, dado que não há diferença significativa entre eles.

Figura 23 - Interface de programação do OMNeT++.

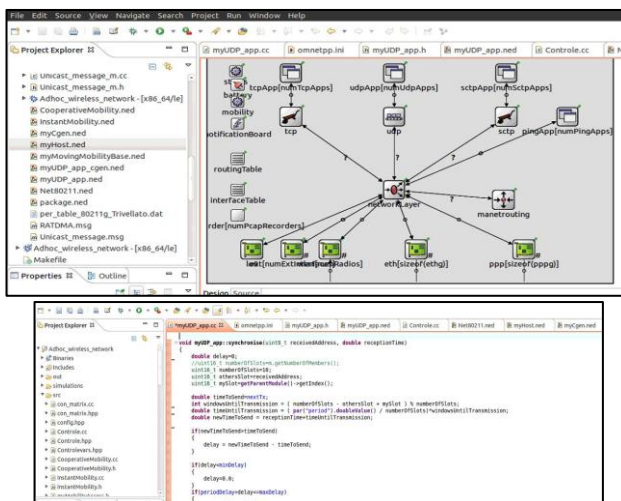
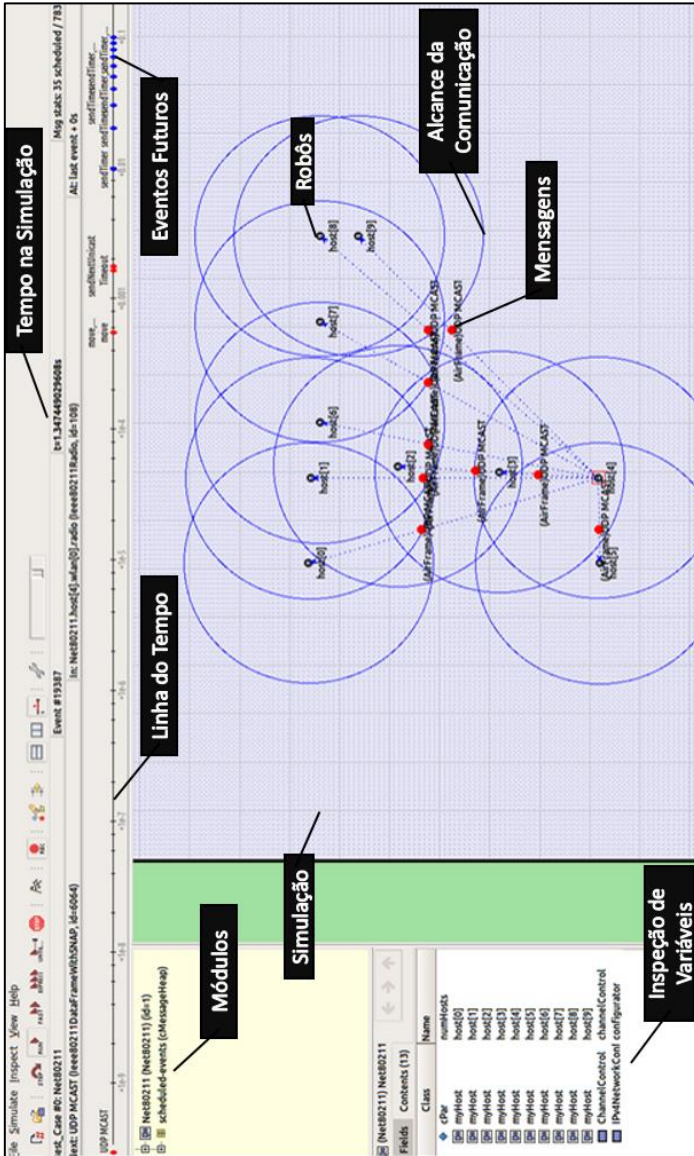


Figura 24 –Interface do OMNeT++ durante uma simulação.



Assim, o OMNeT++ foi o simulador escolhido para elaboração do *framework* de simulação no método devido às suas características

modulares (parte superior da Figura 23), a facilidade em inspecionar elementos da simulação (Figura 24) e por aceitar programação em C++ diretamente em seu código (parte inferior da Figura 23).

A modularidade permite além de uma melhor organização dos algoritmos, a fácil troca de módulos sem necessitar de alterações no restante do código. Isto é particularmente útil para uma simulação integrada que possui técnicas para distintas finalidades.

A possibilidade de usar códigos em C++ possibilitou o uso de bibliotecas de operações com matrizes, operações matemáticas e problemas otimização quadrática (QUADPROG++, 2015).

A interface gráfica do simulador possibilita visualizar o movimento dos robôs, inspecionar variáveis a qualquer instante da simulação, determinar a velocidade da simulação, visualizar a troca e conteúdo das mensagens, dentre outras facilidades que auxiliam na depuração do código.

Os *softwares* e bibliotecas utilizados neste trabalho foram mantidos na mesma versão durante todo o desenvolvimento do projeto, sendo estes listados abaixo:

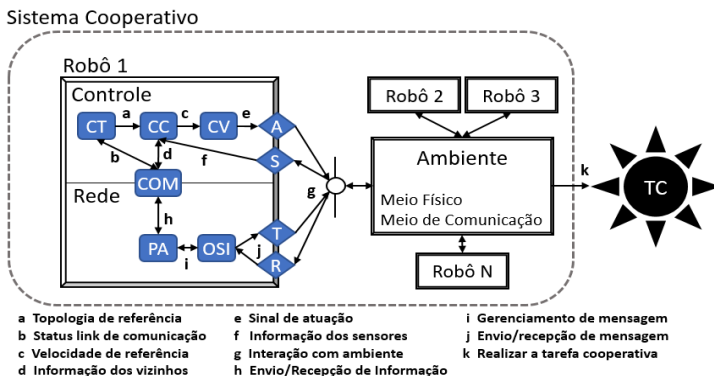
- UBUNTU 16.04
- OMNeT++ 4.6
- INET 2.5
- Quadprog++ 1.2.1

5.2 IMPLEMENTAÇÃO DOS MÓDULOS

O código do *framework* de simulação resultante da implementação consiste em mais de 5000 linhas de código divididas em 21 arquivos, tornando inviável a apresentação detalhada da mesma neste trabalho. No entanto, será abordado uma visão geral dos módulos e de suas respectivas interações.

Os módulos foram elaborados de acordo com a sua concepção inicial definida na modelagem conceitual ao utilizar o método descrito no capítulo anterior. Esse modelo conceitual (Figura 25) indica os seguintes módulos: controle de velocidade, controle cooperativo, controle de topologia, dispositivos de sensoriamento e atuação, comunicação e os aspectos de rede de comunicação (camadas de rede e dispositivo de transmissão/recepção de dados).

Figura 25 - Modelo conceitual dos tópicos da simulação integrada e suas interações.

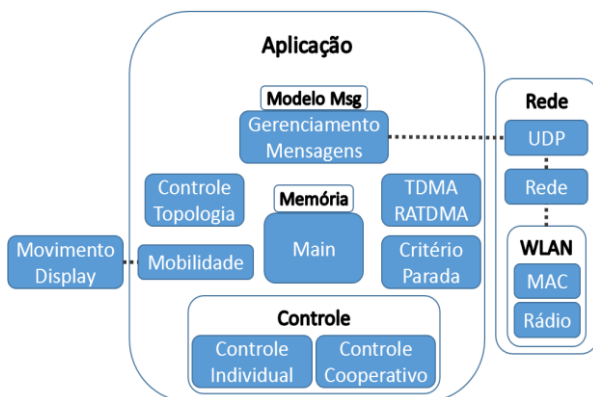


Legenda:

CT Controle Topologia	CC Controle Cooperativo
CV Controle Velocidade	COM Comunicação
PA Protocolo Adicional	OSI Camadas OSI
A Atuação	S Sensor
T Transmissão	R Recepção
TC Tarefa Cooperativa	

A implementação deste modelo conceitual resultou em alguns módulos adicionais, dispostos como indicado na Figura 26.

Figura 26 – Organização do *framework* de simulação implementado no OMNeT++.



A simulação possui uma rotina principal (“Main”), responsável pelo processamento das mensagens recebidas, pelo acionamento do cálculo de controle cooperativo e individual, e pelo armazenamento e alteração de informações na memória do robô. A execução inicial deste

módulo é realizada na inicialização do simulador e ao final de cada rotina, é criada uma auto-mensagem para iniciar novamente o ciclo deste robô em 100ms. Como há possibilidade de facilmente gerenciar e alterar este tempo de início de ciclo de cada robô, os protocolos adicionais baseados em divisão do tempo foram implementados como uma função do *main* no qual regula o acesso de cada robô conforme sua identificação.

A transmissão e recepção de mensagens é realizada pelos modelos do OMNeT++/INET (INET, 2016) para a rede de comunicação, a qual é criada de forma modular camada a camada, de forma transparente para as camadas superiores, como ocorre em implementações reais. Esta transparência e modularidade permite a troca dos módulos (camadas) de redes conforme a necessidade, sem que haja alteração nos algoritmos que estão localizados na camada de aplicação. A implementação atual consiste nos módulos UDP, Rede e a tecnologia de transmissão utilizada.

Para o armazenamento das informações foi utilizada uma classe com todas as possíveis informações do robô em formato de *struct*, ou seja, cada robô possui um vetor *struct* do tipo *robinfo*, no qual cada posição do vetor corresponde a um robô. Utilizando a sua identificação, o robô armazena e altera as informações no seu slot. Ao receber informações de outros robôs, os dados nos outros slots são preenchidos. Estas informações são periodicamente limpas, sendo utilizado a denominação de “ciclo de memória” para definir quantos ciclos a informação de outros robôs permanecem armazenadas.

O controle cooperativo utiliza as informações disponíveis na memória do robô para realizar o cálculo da ação de controle que deverá ser adotada por este robô de acordo com a ação dos demais. Esta informação é repassada ao módulo de controle individual do robô que controla a velocidade e o seguimento de referência.

OMNeT++ também possui um módulo que gerencia as informações na tela de simulação, na qual foi modificada para ser atualizada com determinado período, no qual também é realizado o movimento do robô de acordo com as velocidades lineares de referência no eixo x e y . Estas velocidades são definidas pelo sinal de controle calculado no módulo de controle individual do robô, que pode repassar as velocidades recebidas do módulo do controle cooperativo ou determinar o sinal de controle (velocidade) necessário para atingir a posição de referência recebida.

A topologia criada e utilizada no controle de topologia age de acordo com as mensagens recebidas em cada robô. Ou seja, os links de comunicação entre os robôs só são estabelecidos quando a transmissão de um robô é recebida pelo vizinho em seu alcance, sendo renovada a cada recepção de mensagem. A condição para receber ou não mensagem é estabelecido pelo modelo de rádio e de transmissão utilizado no simulador e não depende dos algoritmos de alto nível.

No entanto, estes dados sobre os links de comunicação são armazenados em estruturas matriciais, que podem ser alteradas pelo controle de topologia, sobrepondo a topologia de comunicação física.

5.3 FUNÇÕES ADICIONAIS

Para auxiliar a análise e implementação das simulações, foram utilizadas algumas propriedades adicionais do simulador. A primeira, refere-se à capacidade do mesmo em utilizar bibliotecas adicionais, onde foi utilizada a biblioteca de otimização quadrática QuadProg++ para C++ (GASPERO, 2013).

A segunda característica importante refere-se à função “WATCH” do simulador, na qual é possível marcar variáveis para que sejam observadas durante e após a simulação no OMNeT++. Ao utilizar esta função em conjunto com a função *log* e *histogram*, ao finalizar a simulação é possível obter os dados brutos e facilmente transportá-los para outra aplicação para tratamento.

Como exemplo de interação entre estes parâmetros, para os experimentos dos estudos de caso, foi criada uma variável para sinalizar o critério de parada da simulação. As coordenadas de cada robô são observadas e avaliadas durante a simulação por um pequeno módulo de observação, onde ao atingir a condição de parada pré-estabelecida a flag é acionada e são armazenados os dados do robô e da simulação. Estas informações, junto com as demais informações da simulação são gravadas no log ao final, possibilitando utilizar estes dados em tabelas de dados no Excel que já foram preparadas para extrair as informações sobre o intervalo de confiança obtido no experimento.

5.4 SOFTWARES AUXILIARES

Para concluir todas as etapas do método, foram utilizados dois softwares auxiliares. O primeiro, como exemplificado no item anterior, foi o uso do Excel para tabelamento dos dados experimentais e determinação dos intervalos de confiança de cada experimento. Como

alternativa a este software, os dados também podem ser tratados por softwares como o ESTATISTICA.

O segundo software utilizado foi o Matlab, para determinação de condições iniciais. Neste sentido, o Matlab foi utilizado devido a facilidade para implementação e execução de cálculos, onde facilmente foi possível criar condições iniciais (CIs) utilizando critérios como: árvore minimamente conectada, distância entre robôs, alcance do raio de comunicação e número de links de comunicação. Cada condição inicial representa uma amostra no experimento e deve possuir uma *seed* diferente para gerar de forma aleatória a CI. O código no Matlab além de gerar as informações necessárias de cada CI, fornece também já o código de implementação no OMNeT++ para as respectivas CIs, agilizando o processo de experimentação.

5.5 CONSIDERAÇÕES FINAIS

Ao longo deste capítulo foi detalhada a implementação e os módulos utilizados no *framework* de simulação. Dada a modularidade do simulador e a fácil adaptação de módulos, o mesmo pode ser facilmente reaproveitado sem haver necessidade de iniciar a implementação de uma nova simulação do zero.

Em conjunto com as ferramentas de análise oferecidas pelo método, é possível realizar uma infinidade de experimentos de acordo com a necessidade do pesquisador.

Para exemplificar como é possível fazer estas experimentações, no próximo capítulo são abordados estudos de caso variados, relacionado diversas características de vários domínios.

Capítulo 6

Utilização e Resultados

Nos capítulos anteriores foram apresentadas cada técnica, cada procedimento e a estrutura básica do *framework* IRoNS. Neste capítulo, o foco está nos resultados obtidos pela sua utilização, em como obter respostas aos questionamentos e em como validar que o método realmente os responde de forma adequada, visto que não existem outros métodos para realizar uma comparação.

Os questionamentos, no escopo deste trabalho, são hipóteses criadas pelo pesquisador sobre uma técnica ou conjunto de técnicas que fazem parte do contexto da simulação integrada elaborada.

É importante salientar que o método não foi criado para dar uma resposta exata e nem é afirmado que as respostas são melhores que um sistema real, e sim, que as respostas poderão fornecer indicativos importantes ao pesquisador sobre a interação destes tópicos sem necessitar de uma implementação em um sistema real para obtê-las. Estes indicativos não só podem confirmar que o pesquisador está no caminho certo durante o desenvolvimento de uma técnica, mas podem indicar também que um erro foi cometido ou um comportamento não desejado, sendo possível consertá-lo ainda na fase de simulação.

Para estabelecer um conjunto inicial das possíveis questões do que se espera que o método responda, foi natural separá-los em relação aos seus respectivos tópicos conforme indicado no Quadro 9.

As respostas destes questionamentos podem ser obtidas para um conjunto qualquer de técnicas implementadas no *framework* de simulação, onde possivelmente outros questionamentos específicos podem ser elaborados, relacionados às particularidades das técnicas envolvidas.

Quadro 9 – Possíveis questões a serem tratadas pelo método.

- a) **Controle Cooperativo**
- Qual a diferença de desempenho de diferentes controles cooperativos no resultado de uma tarefa cooperativa?
 - Como a mudança de parâmetros de uma estratégia de controle cooperativo afeta os resultados da tarefa cooperativa?
- b) **Redes de Comunicação**
- Qual o impacto da rede de comunicação em um sistema de robôs cooperativo? Há alterações nos resultados ao utilizar um protocolo de comunicação adicional?
 - Qual o impacto da perda de pacotes de mensagens no desempenho da tarefa cooperativa? Diferentes distribuições de perda afetam o resultado de que forma?
 - Como diferentes modelos de propagação de sinais de comunicação afetam os resultados da tarefa cooperativa? A tecnologia de comunicação utilizada possui algum efeito no desempenho dos robôs?
- c) **Controle de Topologia**
- Qual o impacto do controle de topologia em um sistema cooperativo?
 - Dentre N opções, qual o controle de topologia mais adequado para uma aplicação com X características?
 - Qual o impacto da topologia inicial na tarefa cooperativa?
 - Como o controle de topologia pode diluir ruídos, erros ou falhas de segurança na troca de informações?
- d) **Geral**
- Como avaliar correlação entre parâmetros de diferentes algoritmos?
 - Como analisar o desempenho de propostas de controle híbrido? (Controle cooperativo e controle de topologia)
 - Como verificar se o parâmetro de avaliação utilizado é adequado para identificar um comportamento, desempenho ou um tipo específico de falha?
- e) **Outros possíveis questionamentos sobre:**
- Consumo de energia;
 - Ambiente;
 - Controle Individual, erro sensorial e de localização;
 - Dispersão de falsas informações;
 - Robôs heterogêneos;
 - Comparação entre abordagens de tarefas com objetivo comum.

Os estudos de caso (EC) elaborados visam demonstrar a flexibilidade do método e da simulação, sendo abordados os seguintes tópicos:

- Impacto da presença da rede de comunicação na tarefa cooperativa de *rendez-vous*;
- Impacto da perda de mensagens ocasionado pela sincronização do TDMA na tarefa cooperativa de *rendez-vous*;
- Impacto da perda de mensagens por interferência de uma rede externa na tarefa cooperativa de *rendez-vous*;

- Comparação entre estratégias de controle cooperativo baseado em consenso para tarefa cooperativa de *rendez-vous*, com e sem perdas de mensagens;
- Impacto do número de links (conectividade) e do controle de topologia na tarefa cooperativa de *rendez-vous*;
- Estudo da relação entre conectividade, estratégia de controle e determinação de referências para tarefa cooperativa de *rendez-vous*;
- Impacto de uma falsa referência na tarefa cooperativa de *rendez-vous* e estudo de formas de atenuação do mesmo por meio de alteração nos pesos de consenso;
- Impacto do período de controle e de comunicação dos robôs na tarefa cooperativa de *rendez-vous*;
- Impacto do armazenamento de informações e do não cancelamento instantâneo de links de comunicação na tarefa cooperativa de *rendez-vous*;

Estes tópicos foram divididos em três categorias, conforme a abordagem adotada para sua apresentação e detalhamento neste trabalho, sendo estas:

- **Resultados Prévios:** a primeira categoria são os tópicos elaborados durante o desenvolvimento do método, onde são brevemente descritos os casos e os resultados obtidos. Estes ECs são focados apenas no *framework* de simulação e utilizam apenas alguns conceitos do método.
- **EC – Passo a Passo:** neste item é realizada a explanação passo a passo do método para um estudo de caso, visando a melhor compreensão do papel do método no estudo.
- **EC - Exemplos:** a terceira categoria são os demais casos elaborados para exemplificar a flexibilidade do método IRONS. Apesar destes ECs utilizarem todos os procedimentos do método, os casos são sucintamente apresentados, focando apenas na apresentação do problema, motivação e nos resultados obtidos.

A elaboração de estudos de caso é uma das abordagens adotadas para validar o método, dado que não há outros métodos para ser realizada uma comparação direta sobre sua eficácia. As outras duas abordagens que foram adotadas para auxiliar neste processo são: utilizar o método para outros conjuntos de técnicas e divulgar o mesmo externamente.

A primeira abordagem consiste na utilização do método em trabalhos de outros dois membros do mesmo grupo de pesquisa, demonstrando que não há dependência do desenvolvedor ou das técnicas. Já a segunda abordagem, se dá por meio da divulgação científica por artigos e a divulgação do método em um *site* próprio, incentivando sua utilização.

Nos próximos itens, são discutidos os EC nos três níveis de detalhamento mencionados anteriormente, os trabalhos em conjunto que estão em desenvolvimento e a divulgação do método.

6.1 RESULTADOS PRÉVIOS

Foram criados dois estudos de caso durante o desenvolvimento do método, com objetivo de testar o *framework* de simulação e avaliar quais os procedimentos necessários para o método proposto. Estes EC não apresentam o modelo comunicativo, planejamento experimental formal e nem validação estatística por IC. Porém, as lições aprendidas em ambos os casos contribuíram significativamente para o desenvolvimento do método.

6.1.1 Apresentação e Resultados

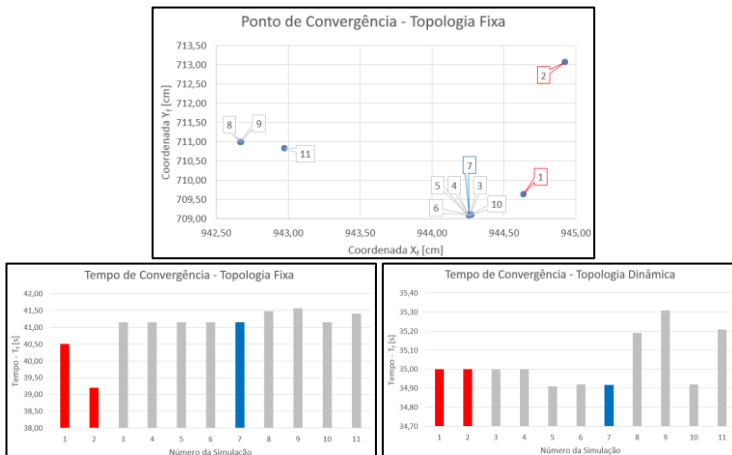
O primeiro EC surgiu de questionamento intrínseco à simulação integrada: Qual o impacto da rede de comunicação na tarefa cooperativa de *rendez-vous*?

Utilizando versão inicial da simulação integrada no OMNeT++, contendo a rede de comunicação MANET, os controles de topologia fixo e dinâmico, dois protocolos de rede adicionais TDMA e RA-TDMA e o controle cooperativo de consenso RHC, foram elaboradas 11 diferentes combinações de algoritmos, condições iniciais e parâmetros.

Apesar de não haver um rigor em sistematizar a alteração de parâmetros e nem na forma de apresentar os resultados (Figura 27), os resultados (RAMOS, 2015b) indicaram que há diferença no desempenho do *rendez-vous* entre os resultados obtidos com uma rede ideal simulada no Matlab e os resultados com presença de um rede de comunicação simulada no OMNeT++. Em especial, notou-se que quando há colisões de mensagens durante o processo de sincronização, os resultados do *rendez-vous* são afetados.

Em relação ao controle de topologia, houve uma diferença significativa nos resultados em favor da topologia dinâmica, motivando a investigação de parâmetros deste tópico.

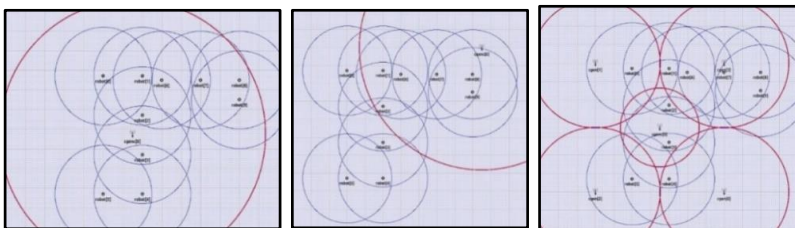
Figura 27 – Resultados do primeiro estudo de caso em relação ao ponto e tempo de convergência do *rendez-vous*.



O segundo questionamento surgiu a partir da percepção de que as colisões do primeiro caso possuíam um evidente impacto nos resultados da tarefa cooperativa, conduzindo à questão: Qual o impacto de uma rede de comunicação com colisões de mensagens na tarefa de *rendez-vous*?

Da mesma forma que o caso anterior, foram analisados o ponto e o tempo de convergência final do *rendez-vous*. No entanto, neste caso também é inserido no ambiente geradores externos de mensagens (Figura 28) que ocasionam a colisão com as mensagens trocadas entre robôs.

Figura 28 - Casos com gerador de mensagens e círculo vermelho representando o alcance da comunicação (interferência).

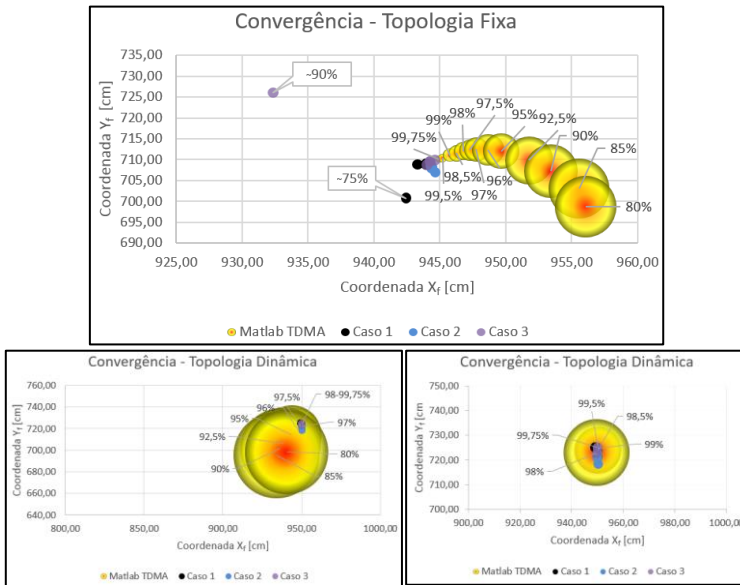


Observando os resultados obtidos para este caso (RAMOS, 2015c) (Figura 29), os pontos de convergência apresentavam um padrão quando obtidos no Matlab, convergindo em torno de um ponto, com deslocamento à direita conforme as taxas de perdas aumentavam. No OMNeT++, os resultados foram mais esparsos e não apresentavam um comportamento bem definido.

Em relação à topologia dinâmica, o ponto de convergência no Matlab apresentou uma variação muito grande, quanto no OMNeT++ foi observado o comportamento inverso, uma convergência em torno de um valor.

Em geral, os resultados da topologia dinâmica apresentam menor tempo de convergência, mesmo sob colisões de mensagens, em relação ao caso com topologia fixa. No entanto, com o aumento do número de *links* em uma rede com colisões, também houve aumento do número de informações desatualizadas em cada robô, levando a divergência de casos nos quais houve convergência utilizando topologia fixa.

Figura 29 - Resultados de convergência dos casos com topologia fixa, onde as porcentagens indicam a probabilidade média de comunicações com sucesso.



6.1.2 Contribuições para o Método IRoNS

Apesar dos estudos de caso iniciais demonstrarem que há diferença entre uma simulação de robótica cooperativa com rede de comunicação ideal e uma com rede de comunicação mais realistas, houveram pontos que não puderam ser melhor aproveitados, como por exemplo, a não quantificação desta diferença e nem a determinação de uma possível interação entre os parâmetros.

A comparação direta dos resultados de ambos os simuladores e sua apresentação em dados brutos, também não permitiu obter informações úteis para conclusão dos resultados, além da visível diferença entre os resultados.

Estes dois pontos são deficiências graves que prejudicam o estudo e podem invalidar o mesmo, dado que não há um rigor na elaboração do planejamento experimental e nem uma análise estatística dos dados.

O conjunto de ferramentas proposto no método permite utilizar um número limitado de amostras para determinar, de forma estatística, o impacto e interação de diferentes fatores presentes na simulação.

Em relação à documentação, nestes casos foram elaboradas documentações simplificadas, de forma semelhante a modelagem conceitual, para utilização durante a implementação do *framework* de simulação. Este foi um ponto positivo que foi mantido para o método, resultando na modelagem conceitual em seu formato atual. No entanto, notou-se um déficit na documentação ao repassar informações a terceiros, onde era exigido uma documentação formal mais detalhada a respeito das técnicas, considerações, suposições e resultados esperados. Esta deficiência motivou a adoção da modelagem comunicativa em conjunto com a modelagem conceitual.

O aprendizado adquirido durante a implementação destes EC, tanto em relação aos erros quanto aos acertos, aliado com a pesquisa bibliográfica em outras áreas de pesquisa, resultaram na versão atual do método IRoNS, sendo este, aplicado aos demais EC.

6.2 EC – PASSO A PASSO

Com exceção dos resultados prévios, todos os estudos realizados nesta tese utilizaram os procedimentos do método, como definido no Capítulo 4. No entanto, devido a sua extensão, foi selecionado apenas um dos estudos de caso para exemplificar detalhadamente, de forma incremental, o uso dos procedimentos do método (Figura 14).

6.2.1 Formulação do Problema

Partindo de uma ideia inicial de que se deseja trabalhar com robôs cooperativos móveis, uma rede de comunicação e com controle de topologia, deseja-se avaliar o desempenho de três técnicas de consenso aplicado à *rendez-vous*, sob a presença ou não de falhas intermitentes na comunicação. O objetivo é comparar o tempo de convergência entre três

algoritmos de *rendez-vous* e avaliar se há diferença significativa entre as técnicas.

O problema pode ser refinado ao especificar outras características que deverão estar presentes, como a utilização de 10 robôs homogêneos terrestres, compostos de duas rodas com configuração diferencial, que estão operando de forma descentralizada para realizar uma tarefa de *rendez-vous*.

Define-se como requisito a utilização da técnica de consenso RHC para *rendez-vous* (Seção 2.3.3), representado pela sigla “MPC”, a qual deverá ser comparada com quaisquer outras duas técnicas clássicas de consenso presentes na literatura.

Em relação à rede de comunicação, uma solução descentralizada e sem fio é desejada, de forma a trabalhar em sinergia com a estratégia de consenso distribuída.

6.2.2 Escolha de Soluções

O segundo procedimento do método é definir um conjunto de técnicas que satisfaçam os requisitos do estudo e que contribuam para o estudo do problema definido no procedimento anterior.

Para este caso, as duas estratégias de consenso para *rendez-vous* escolhidas foram o consenso médio (M) (subseção 2.3.1) e o consenso circuncentro (C) (subseção 2.3.2). Ambas as estratégias são aplicações clássicas do consenso e podem ser facilmente replicadas como referências em outros trabalhos, servindo de ponto de referência em futuras comparações.

A rede de comunicação adotada é uma MANET utilizando UDP/IP e tecnologia de comunicação Wi-Fi (seção 2.4). A topologia de comunicação é mantida fixa durante toda a simulação (seção 2.1) e o período de controle e de envio de mensagens foram definidos no valor de 100ms.

6.2.3 Especificação do Sistema

Nesta etapa de documentação, além das informações definidas anteriormente, são definidas as possíveis interações e as suposições a respeito das técnicas.

Um exemplo de interação esperada é a da manutenção da topologia de comunicação, pois a mesma foi fixada para que não haja influência da criação de novos links, que dependendo do momento em que são formados, afetam significativamente os resultados da tarefa.

Outra interação esperada e que é o foco do estudo, é a perda intermitente de informação na qual acarreta uma alteração momentânea no resultado do consenso. É criada a hipótese de que pode haver um melhor desempenho do conjunto de robôs se ao invés de instantaneamente cancelar o link de comunicação e descartar a informação do robô vizinho, os mesmos utilizarem informações antigas durante alguns ciclos de processamento adicionais. Neste sentido, também é esperado que o consenso RHC lide melhor com falhas intermitentes, mesmo cancelando os links instantaneamente, dado que há uma janela de predição cálculo de consenso.

Quanto às suposições sobre o sistema, como o estudo tem como objetivo apenas a comparação de técnicas de consenso e do impacto da perda de mensagens, é necessário evitar possíveis influências de outras técnicas ou variáveis. Desta forma, as principais suposições são: robôs são pontos finitos no espaço, a colisão física entre robôs não é considerada, o ambiente é um espaço aberto sem objetos, que o movimento do robô é regido por uma dinâmica de primeira ordem, que não há incerteza nas informações e que cada robô sabe sua própria posição em coordenadas globais com precisão.

6.2.4 SP – Planejamento e Implementação

O planejamento da simulação padrão (SP) consiste na modelagem conceitual e comunicativa para o *framework* de simulação, nas quais são extensas e similares ao apresentado na seção 4.2 e no ANEXO B. Portanto, as mesmas não serão detalhadas neste item.

A implementação da SP foi realizada no simulador de eventos discretos OMNeT++/INET, como discutido nas seções 5.1. Utilizando as métricas para quantificação da qualidade da simulação (seção 4.3) indicadas no Quadro 6, a simulação para este caso obteve classificação “Muito Boa”, devido às respectivas notas numéricas: A - 39, B - 39, C - 3, D - 1, E - 4 e F - 1, totalizando 87 pontos.

As métricas e parâmetros utilizados para observar e analisar os resultados, para este EC, são indicados no Quadro 10.

Quadro 10 - Métricas e parâmetros utilizados na comparação entre técnicas de consenso.

a) Métricas

- **Tarefa:** *Rendez-vous* Médio, Circuncentro e RHC.
 - *Qualitativa:* sucesso ou não da tarefa cooperativa. Define-se um *Rendez-vous* como bem-sucedido quando todos os robôs convergem para um único ponto, independentemente do tempo.
 - *Quantitativa:* Tempo de convergência final T_F .
- **Rede:** Caso com e sem perda de mensagens.
 - *Qualitativa:* casos com manutenção de informação perdida por um e dois ciclos de processamento adicionais.
- *Comparação:* entre as técnicas e entre os resultados da mesma técnica.

b) Parâmetros

Parâmetros Utilizados: coordenadas de convergência (X_F, Y_F) , coordenadas das posições dos robôs (x_i, y_i) , tempo que levou para convergência (T_F), peso do consenso (λ) e penalização da ação de controle (γ_i).

c) Informações Adicionais

Validação: em relação à tarefa cooperativa, resultados obtidos em uma simulação no Matlab é comparada com os resultados obtidos no OMNeT++ sem adição de rede de comunicação. As métricas utilizadas são a diferença entre o ponto (D_f) e o tempo de convergência (T_f) finais entre as simulações.

$$\text{Erro } D_F = \sqrt{([X_F^{MAT} - X_F^{OMN}]^2 + [Y_F^{MAT} - Y_F^{OMN}]^2)}$$

$$\text{Erro } T_F = T_F^{MAT} - T_F^{OMN}$$

6.2.5 SP – Validação

O processo de validação foi aplicado para cada uma das estratégias de consenso utilizadas neste estudo, nas quais foram originalmente simuladas no Matlab. Nesta validação foram utilizados dois critérios de avaliação: o erro no tempo de convergência (T_f) e o erro na distância de convergência (D_f).

São utilizadas as mesmas condições de simulação para cada amostra em ambos simuladores, totalizando 15 amostras. É utilizada a verificação em quatro passos e a validação estatística, onde são obtidos os intervalos de confiança sobre o erro para cada técnica, com 99% de confiança, como indicado Tabela 3.

Como todos os IC possuem intervalos pequenos, indicando uma boa precisão e todos incluem zero, indicando que o erro verdadeiro pode ser zero, pode-se concluir que o *framework* de simulação é

suficientemente válido para estas técnicas em relação as suas respectivas simulações Matlab.

Tabela 3 – Intervalos de confiança resultantes da validação de cada parâmetro e cada estratégia de controle

	IC
RHC - IC 99% - T_r	[-0,016 ; 0,041]
RHC - IC 99% - D_r	[-0,013 ; 0,0298]
Médio - IC 99% - T_r	[-0,01 ; 0,021]
Médio - IC 99% - D_r	[-0,013 ; 0,022]
Circuncentro - IC 99% - T_r	[-0,021 ; 0,016]
Circuncentro - IC 99% - D_r	[-0,023 ; 0,018]

6.2.6 EC – Planejamento Experimental

O objetivo deste experimento é comparar o desempenho das três técnicas de consenso em relação à tarefa de *rendez-vous*, com e sem a presença de falhas intermitentes na comunicação. Para isso, são utilizadas 15 amostras, nas quais consistem em 15 posições iniciais diferentes e por consequência, 15 diferentes topologias iniciais de comunicação.

Para o planejamento experimental são utilizados dois fatores: a situação da rede e a mudança da técnica de consenso. O primeiro fator é a comparação de uma rede sem perdas de mensagens, denominada aqui de ideal (I) e associado ao valor “-”, e uma rede com colisão de mensagens, onde foram selecionadas duas situações: a informação perdida é utilizada por um ciclo de controle adicional (C-M1) ou dois ciclos de controle adicionais (C-M2).

Em relação ao segundo fator, os algoritmos de consenso são comparados em pares, resultando nos seguintes experimentos de comparação: Médio (M) x MPC, Médio (M) x Circuncentro (C) e MPC x Circuncentro (C). É considerado que o primeiro algoritmo assume valor “-” do fator e o segundo o de “+”. Portanto, no caso M x MPC, por exemplo, se deseja analisar o ganho/perda de desempenho da tarefa cooperativa ao mudar a técnica de consenso médio para consenso MPC.

Os impactos dos fatores são ordenados de acordo com os próprios fatores, ou seja, e_1 corresponde a presença ou não de colisão de mensagens e e_2 corresponde a comparação entre técnicas de consenso, onde e_{12} indica a interação entre ambos fatores. O planejamento experimental generalizado para todas as comparações é indicado no Quadro 11.

Quadro 11 - Planejamento experimental para comparação de técnicas de consenso e o impacto da perda de mensagens com retenção de informações.

Combinação de Fatores	Rede	Consenso	Tempo Convergência (R^c_i)
1	-	-	R^1_i
2	+	-	R^2_i
3	-	+	R^3_i
4	+	+	R^4_i

Utilizando este planejamento experimental, foram realizadas as primeiras experimentações e análises, que indicaram uma interação entre os fatores. Em outras palavras, os resultados de comparação entre as técnicas possuem diferentes desempenhos de acordo com a situação da rede.

Para obter resultados mais precisos a respeito da comparação dos consensos em cada situação, foi elaborado um segundo planejamento experimental desacoplado, onde se realiza a comparação entre as técnicas de consenso sob uma condição de rede por vez. Como estas análises são propostas individualmente, os IC são construídos de forma similar ao da validação. Todas as análises deste segundo planejamento experimental são indicadas no Quadro 12.

Quadro 12 – Análises realizadas para estudo de caso de comparação entre estratégias de consenso e a situação da rede de comunicação.

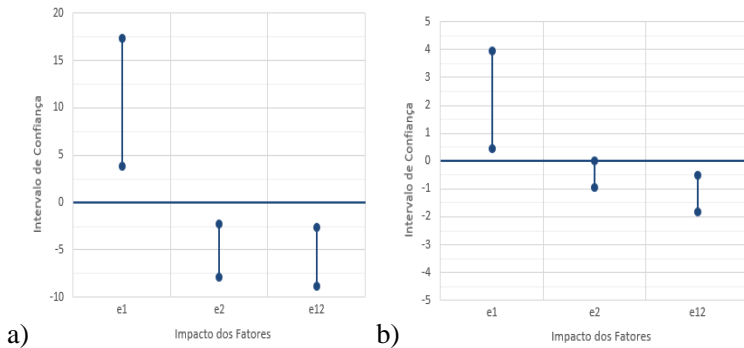
Colisão de Mensagens	Comparação - Consenso		
	M x MPC	M x C	MPC x C
Ideal	Figura 31		
Colisão – M1	Figura 32		
Colisão – M2	Figura 33		

Comparação - Colisão de Mensagem	Consenso		
	M	C	MPC
I x C-M1	Figura 32		
I x C-M2	Figura 33		

6.2.7 EC – Experimentação e Análise dos Resultados

Para o primeiro planejamento experimental, foram construídos os intervalos de confiança com grau de confiança de 99% e 15 amostras, resultando em estudos de com confiança de 97%. Os intervalos de confiança para o caso de MxMPC é indicado na Figura 30, para o caso do fator de rede com a) um ciclo de memória e b) dois ciclos de memória.

Figura 30 – Intervalos de confiança do planejamento experimental para comparação entre MxMPC (e2), utilizando como fator da rede (e1) a comparação IxCM1 (a) e IxCM2 (b)



Nota-se que o IC sobre o impacto da perda de mensagens em a) é significativo estatisticamente, com valores relativamente amplos devido a aleatoriedade das condições das amostras e da forma que as respectivas interferências afetam os resultados. Em relação à técnica de consenso, há um ganho ao utilizar o consenso MPC em relação ao médio, indicando que o mesmo é mais tolerante às falhas. Contudo, há uma interação entre os fatores, indicando que as informações sobre o consenso dependem das condições da rede.

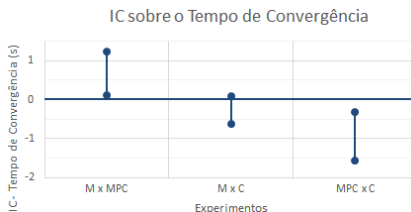
Estas conclusões também são obtidas ao utilizar dois ciclos de memória nos robôs, apesar dos resultados demonstrarem uma expressiva melhora no desempenho dos algoritmos.

Como há interesse de obter informações mais precisas a respeito da comparação entre as técnicas de consenso e observando que esta interação também ocorre para as demais comparações de consenso, o planejamento experimental foi reformulado para desacoplar os fatores. Esta mudança consiste em realizar as comparações sob o mesmo aspecto de rede ou sob a mesma técnica de consenso.

A primeira análise deste novo planejamento, é a comparação direta entre as técnicas de consenso sem a presença de colisão de mensagens. Onde são obtidos os intervalos de confiança com grau de confiança de 99% indicados na Figura 31.

Figura 31 – Intervalo de confiança de 99% sobre o tempo de convergência de cada experimento de comparação.

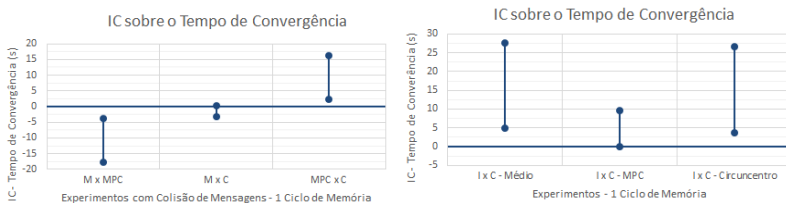
	IC - [Inferior, Superior]	
E1	0,123	1,223
E2	-0,616	0,083
E3	-1,557	-0,323



Observando o segundo intervalo de confiança é possível notar que não há diferença estatística significativa entre o consenso médio e circuncentro. Essa conclusão é confirmada ao analisar a mudança de médio para MPC, onde há um acréscimo significativo no tempo de convergência e comparar a mesma com a mudança de MPC para Circuncentro, onde a diferença é significativa e inversamente proporcional. No entanto, em termos práticos, o valor absoluto destes intervalos não é impactante, pois esta diferença de até 1,5 segundo tem pouco impacto em tempos totais da tarefa com valores entre 20 a 50 segundos.

A segunda série de experimentos consideram a colisão de mensagens e a retenção da informação por um ciclo adicional (C-M1). Adicionalmente, também é comparado o resultado de cada técnica com o resultado obtido sem a presença de colisões, resultando nos intervalos de confiança indicados na Figura 32.

Figura 32 – Intervalos de confiança de 99% entre as técnicas em ambiente com colisão (esquerda) e da mesma técnica em ambientes ideal e com colisão de mensagem (direita), para um ciclo de memória.



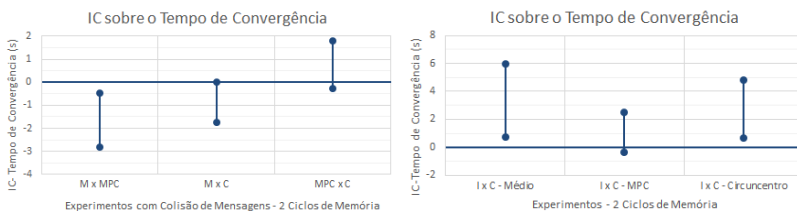
Sob estas novas condições, o desempenho das técnicas é totalmente diferente do que obtido inicialmente: o MPC passa a ter um desempenho superior em relação às demais técnicas. Isto fica evidente no gráfico à esquerda, onde ao utilizar o MPC ao invés do Médio (M x MPC), há uma melhora significativa tanto estatística quanto de valor absoluto e também ao verificar o mesmo intervalo, inversamente proporcional,

quando se utiliza o Circuncentro ao invés do MPC (MPC x C). Ainda, o gráfico comparando os resultados entre ambiente ideal e com colisões (I x C), é possível notar que tanto o Médio quanto o Circuncentro sofreram um impacto mais significativo do que o MPC. Ou seja, pode-se concluir que o MPC é mais tolerante a perda de informação do que as demais técnicas.

A terceira série de experimentos consideram outra possível questão: é possível melhorar os resultados ao aumentar o ciclo de memória dos robôs?

Para responder esta pergunta, os experimentos anteriores foram realizados modificando apenas o ciclo de memória de 1 para 2 (tolerância de duas perdas consecutivas), obtendo os resultados indicados na Figura 33.

Figura 33 - Intervalos de confiança de 99% entre as técnicas C em ambiente com colisão (esquerda) e da mesma técnica em ambientes ideal e com colisão de mensagem (direita), para dois ciclos de memória.



Ao comparar os resultados obtidos anteriormente com os resultados com dois ciclos de memórias, a primeira característica mais evidente é que a escala dos valores diminuiu drasticamente, aproximando-se dos valores originais sem colisão. No entanto, o algoritmo de Circuncentro apresentou uma melhora perceptível, com desempenho similar ao do MPC, como pode ser observado no terceiro intervalo (MPC x C) e ao verificar o menor impacto das colisões (I x C) no Circuncentro em relação ao (I x C) médio.

Estatisticamente, a diferença entre MPC e Circuncentro está no limite de ser considerada significativa ou não. Mas seguindo rigorosamente a definição adotada sobre a inclusão do zero, não é considerada significativa. Nestes casos, se essa informação for vital, o desenvolvedor pode aumentar o número de amostras para o experimento, obtendo um intervalo de confiança mais definido. Contudo, há um segundo indicativo de que não há diferença significativa entre ambos: o baixo valor absoluto do intervalo (menor que 2 segundos). Portanto, a inclusão de novas amostras não foi necessária.

6.2.8 EC – Conclusão

Com os resultados obtidos neste estudo, foi possível concluir que não há diferença significativa entre os resultados entre as três técnicas de consenso para o conjunto de parâmetros utilizados, com a condição não haver perda de mensagens durante a execução da tarefa de *rendez-vous*. Porém, quando esta condição não é atendida, o MPC apresenta melhor desempenho, dada sua capacidade em lidar com perdas esporádicas.

Uma alternativa que se mostrou mais eficiente para lidar com perda de mensagens é permitir a utilização das informações dos ciclos anteriores. Em particular, utilizar a informação de até dois ciclos aproxima os resultados ao caso ideal e possui uma melhor sinergia com o algoritmo de Circuncentro.

Obter este tipo de informação sobre o sistema, ainda na fase de simulação, somente foi possível por meio da simulação integrada e do método. Com essa informação, o desenvolvedor poderá decidir o que é mais viável para ser implementado em robôs reais, auxiliando na transição entre a simulação e implementação real, eliminando possíveis retrabalhos.

6.3 EXEMPLOS DE ESTUDOS DE CASO

Para demonstrar a flexibilidade do método para obter respostas à diferentes tipos de questionamentos, foram elaborados estudos (EC) para diversos tipos de questionamentos relacionados ao conjunto de técnicas indicadas no Capítulo 2. Ao contrário da seção anterior, estes EC são sucintamente apresentados, focando apenas na apresentação do problema, na motivação e nos resultados obtidos.

6.3.1 Avaliação da Conectividade

Este estudo de caso foi direcionado para a questão da conectividade, o qual foi utilizado em pequenos exemplos no capítulo anterior. Neste caso, os questionamentos que motivaram a construção do caso foram: Quanto o desempenho da tarefa cooperativa é afetada pela conectividade? A conectividade é a responsável pela influência ou a distância percorrida pelos robôs? Existe alguma relação entre ambas?

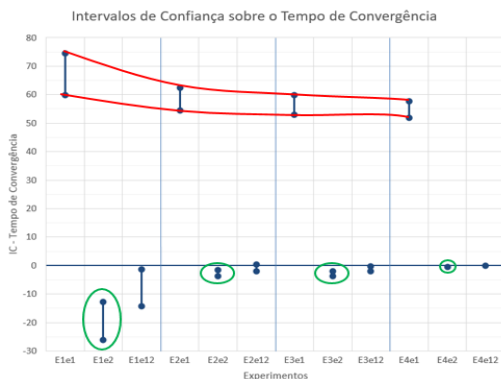
Assim foram elaboradas diferentes situações para tentar avaliar os resultados, até selecionar dois parâmetros considerados adequados: a

maior distância entre um robô (d_{max}) e a referência, e o número de links de comunicação dos robôs (C), como métrica de conectividade.

Os valores assumidos para a análise são o maior/menor d_{max} que satisfaça a criação de uma referência para ao menos três robôs e cinco valores para a conectividade: *Mínima*, C_2 , C_3 , C_5 e *Máxima*. Os resultados são apresentados de forma pareada para a conectividade: *Mínimax* C_2 (E1), $C_2 \times C_3$ (E2), $C_3 \times C_5$ (E3) e $C_5 \times$ *Máxima* (E4), sendo utilizadas 15 amostras e totalizando 150 simulações.

De acordo com os passos e dados já apresentados nos itens 4.4.1, 4.5.1, 4.6.1, os IC resultantes são indicado na Figura 34, onde também é destacada a tendência dos resultados à medida que a conectividade é incrementada.

Figura 34 - Intervalo de confiança e indicações visuais das tendências dos resultados (d_{max} x conectividade) ao incrementar a conectividade em cada experimento.



A partir do gráfico é possível destacar como d_{max} possui um efeito significativo, que pode ser agravado por uma baixa conectividade (em vermelho). Isto é visualmente perceptível no gráfico e sobretudo indicado pelo fator de correção E_{1e12} . À medida que a conectividade aumenta, a informação flui melhor entre os robôs e a mesma gradativamente tem menos influência nos resultados (em verde), se tornando praticamente insignificante a partir da conectividade 3.

Conclui-se que não é necessária uma grande conectividade entre os robôs para obter uma grande melhora nos resultados e que ao lembrar dos resultados obtidos nos *resultados prévios*, onde um maior número de *links* foi prejudicial em uma rede com perda de pacotes, pode-se afirmar que há indicativos de que um controle de topologia que mantenha uma

determinada conectividade entre os valores mínimo e máximo, pode ter efeito benéfico para uma rede de robôs cooperativos.

6.3.2 Robôs Heterogêneos

Os robôs utilizados em uma mesma tarefa podem ter diferentes características físicas, recursos, algoritmos ou capacidades de comunicação. Este caso, de robôs heterogêneos, pode ser abordado no método e no *framework* de simulação devido a flexibilidade do primeiro e a modularidade do segundo.

Para demonstrar a utilização do método para robôs heterogêneos, foram criados três cenários de utilização dos robôs, onde o desenvolvedor deseja avaliar e comparar o efeito destas configurações na tarefa cooperativa de *rendez-vous*. Os cenários foram definidos como:

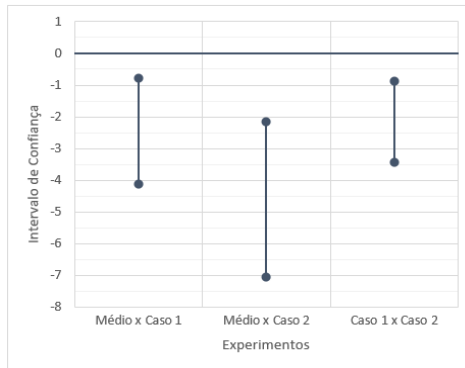
- **Médio:** dez robôs utilizando consenso médio, uma rede MANET, TDMA como protocolo adicional, período de 100ms, peso na ação de controle de 2 e controle de topologia fixa.
- **Caso 1:** três robôs de maior capacidade de processamento utilizam o consenso por MPC, peso 0,1 na variação de velocidade, 7 robôs que utilizam o consenso médio. São utilizados os mesmos parâmetros do caso anterior e controle de topologia fixa.
- **Caso 2:** três robôs com capacidade de processamento intermediária utilizando o consenso por circuncentro, com peso 2 na ação de controle. Os três robôs possuem raio de comunicação 20% maior que os 7 robôs restantes, que utilizam o consenso médio. O controle de topologia é similar aos casos anteriores, mas é fixada a partir da nova topologia resultante do aumento do raio de comunicação.

É utilizada como métrica de avaliação o tempo de convergência do *rendez-vous*, construindo o intervalo de confiança de 98% e 15 amostras (condições iniciais) para a diferença no tempo de convergência entre: 1- Caso homogêneo e o primeiro cenário (Médio x Caso 1); 2- Caso homogêneo e o segundo cenário (Médio x Caso 2); e 3 – Entres os dois cenários (Caso 1x Caso2). Os resultados são indicados na Figura 35.

Nota-se um ganho no desempenho (redução no tempo de convergência) significativo nos dois cenários heterogêneos em relação ao homogêneo. No primeiro cenário, como somente é alterada a estratégia do consenso, o intervalo é menor e mais definido, ao contrário do segundo cenário, que além de modificar a estratégia de consenso também aumenta

o raio de comunicação, apresenta maior oscilação devido à variação do número de links utilizados pelos robôs.

Figura 35 – Intervalos de confiança de 98% sobre a diferença no tempo de convergência para dois casos de robôs heterogêneos.



6.3.3 Escalabilidade

O número de robôs utilizados em uma tarefa pode variar conforme disponibilidade ou aplicação, podendo afetar o desempenho da tarefa cooperativa. Além de um maior número de informações e de interações necessárias para atingir o consenso, ao aumentar o número de robôs, também há uma maior diversidade de topologias de comunicação.

Para demonstrar um estudo relacionado ao número de robôs para tarefa de *rendez-vous*, foi elaborado um estudo onde deseja-se verificar se as diferenças nas estratégias de consenso médio e circuncentro se mantêm as mesmas ao aumentar o número de robôs e se há alteração desta diferença ao adotar diferentes topologias.

São utilizados os mesmos parâmetros para 15 configurações, cada uma com um número diferente de robôs (6 a 20), igualmente espaçados, utilizando a estratégia de controle de consenso médio (M) ou circuncentro (C), para os casos de topologia linear (L) ou em grade (G), conforme indicado na Figura 36.

Os tempos de convergência do *rendez-vous* obtidos são indicados na Figura 37, em função do número de robôs.

Figura 36 – Configuração inicial dos 20 robôs, para o caso de topologia linear (esquerda) e topologia de grade (direita).

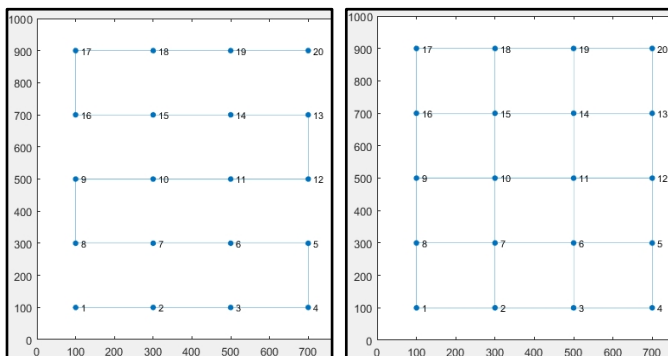
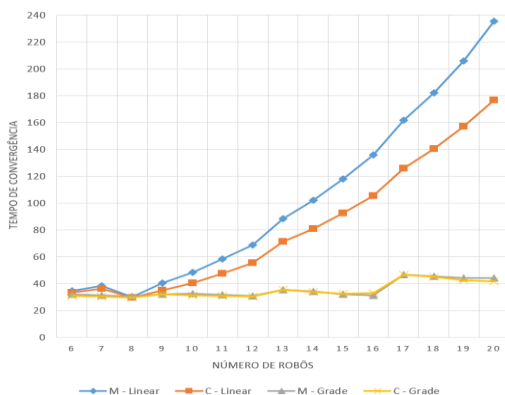


Figura 37 - Tempos de convergência de rendez-vous obtidos para consenso médio e circuncentro e para topologias linear e de grade, em função do número de robôs.



Nota-se uma dependência nas amostras para os casos de topologia linear (M – Linear e C – Linear), onde o incremento no número de robôs (acima de 9 robôs) proporciona um constante incremento nos tempos de convergência. Este comportamento inviabiliza a conclusão sobre os valores obtidos no intervalo de confiança para topologia linear.

No entanto, para demonstrar como este incremento se reflete no IC, foi dada sequência à elaboração do projeto experimental, onde foi considerado como fatores: a mudança de topologia e a mudança na técnica de consenso, resultando na matriz experimental indicada no Quadro 13.

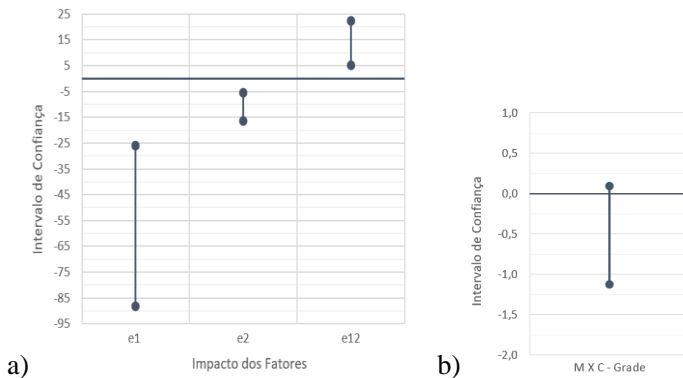
Para o primeiro fator, a topologia linear é representada pelo valor “-” e a topologia de grade pelo valor “+”. No caso do segundo fator, o consenso médio é representado como fator “-” e o circuncentro como “+”.

Quadro 13 - Matriz experimental para o experimento de escalabilidade.

Combinação de Fatores	Topologia (L x G)	Consenso (M x C)	Tempo Convergência (R^i)
1	-	-	R^1_i
2	+	-	R^2_i
3	-	+	R^3_i
4	+	+	R^4_i

São construídos os intervalos de confiança de 98% sobre o impacto de cada fator (e_1 e e_2) e sobre a interação sobre os mesmos (e_{12}), como indicado na Figura 38 – a).

Figura 38 – Intervalo de confiança de 98% sobre a) o impacto de cada fator e de sua interação e b) intervalo de confiança sobre a diferença entre as técnicas de consenso no caso da topologia de grade.



Como previsto, o intervalo de confiança sobre o impacto do primeiro fator (a topologia) é impreciso e apesar de estatisticamente significativo e ser uma boa indicação de comportamento, o seu valor dependente do número de robôs.

O segundo fator (consenso) apresenta um intervalo mais definido, no entanto, como indicado no terceiro IC (e_{12}), há interação entre os dois fatores devido a não inclusão de zero no IC e seu valor considerável. Ou seja, os resultados obtidos nas técnicas de consenso são afetados pela topologia utilizada, que por sua vez, é dependente do número de robôs.

Assim, para obter uma resposta definida, foi desconsiderando as amostras que utilizam a topologia linear, sendo construído um intervalo

de confiança sobre a diferença entre os consensos ($M \times C$) para o caso de topologia de grade, como indicado na Figura 38 – b).

O IC resultante é bem definido e contém zero, portanto, é possível concluir que para a topologia de grade com até 20 robôs, não há diferença estatística entre as estratégias de consenso médio e circuncentro.

6.3.4 Comunicação Multi-Hop

A troca de informações entre os robôs é parte fundamental dos algoritmos de consenso distribuídos e ao longo deste documento foram apresentados diversos casos baseados na troca direta de mensagens entre os robôs. No entanto, o *framework* de simulação integrado permite expandir esta comunicação ao utilizar, como exemplo, protocolos de rede com o roteamento de mensagens para repassar informação entre robôs. Esta possibilidade, permite que os robôs adquiram informações de robôs que não estão em seu alcance físico direto de comunicação.

Para exemplificar um estudo onde os robôs possuem informações de seus vizinhos indiretos, é criado um cenário onde cada robô recebe informação sobre os vizinhos de seus vizinhos diretos, ou seja, informações de robôs localizados à *2-hops* (dois vértices) de distância.

Deseja-se verificar se há melhora significativa nos resultados ao adotar este procedimento, se afeta os algoritmos de consenso de forma diferente e se há uma interação com a topologia utilizada. São utilizados os mesmos parâmetros e condições iniciais do caso anterior sobre escalabilidade (6.3.3).

O planejamento experimental é elaborado para três fatores: a topologia, estratégia de consenso e a comunicação indireta. Os valores negativos e positivos de cada fator são: topologia linear e de grade, consenso médio e circuncentro, e a comunicação direta e indireta (*2-hop*), resultando na matriz experimental indicada no Quadro 14.

Quadro 14 - Matriz experimental para o experimento com comunicação indireta.

Combinação de Fatores	Topologia (L x G)	Consenso (M x C)	2-Hop (Não x Sim)	Tempo Convergência (R_i^c)
1	-	-	-	R_i^1
2	+	-	-	R_i^2
3	-	+	-	R_i^3
4	+	+	-	R_i^4
5	-	-	+	R_i^5
6	+	-	+	R_i^6
7	-	+	+	R_i^7
8	+	+	+	R_i^8

No entanto, como visto no caso anterior, os dados para a topologia linear não são adequados para construção do intervalo de confiança, motivo pelo qual a matriz experimental típica foi substituída por dois experimentos menores com diferentes objetivos.

O primeiro experimento consiste em analisar o impacto da comunicação indireta nas técnicas de consenso e determinar uma possível interação, utilizando apenas as amostras geradas pela topologia de grade, resultando na matriz experimental indicada no Quadro 15.

Quadro 15 – Matriz experimental para amostras utilizando topologia de grade.

Combinação de Fatores	2-Hop (D x I)	Consenso (M x C)	Tempo Convergência (R_i^c)
1	-	-	R_i^1
2	+	-	R_i^2
3	-	+	R_i^3
4	+	+	R_i^4

Neste experimento, a forma de obter informação dos vizinhos é o primeiro fator, denotado como “-” quando o robô apenas utiliza a informação recebida diretamente (D) e “+” quando utiliza as informações indiretas (I) dos vizinhos de seus vizinhos. O segundo fator é a estratégia de consenso utilizada, onde o parâmetro “-” é o consenso médio e o “+” o consenso por circuncentro.

Os resultados obtidos neste experimento são indicados na Figura 39, sendo indicados em a) na forma de intervalos de confiança de 98% construídos a partir da matriz experimental do Quadro 15 e em b) para os tempos de convergência das amostras de cada combinação de fatores.

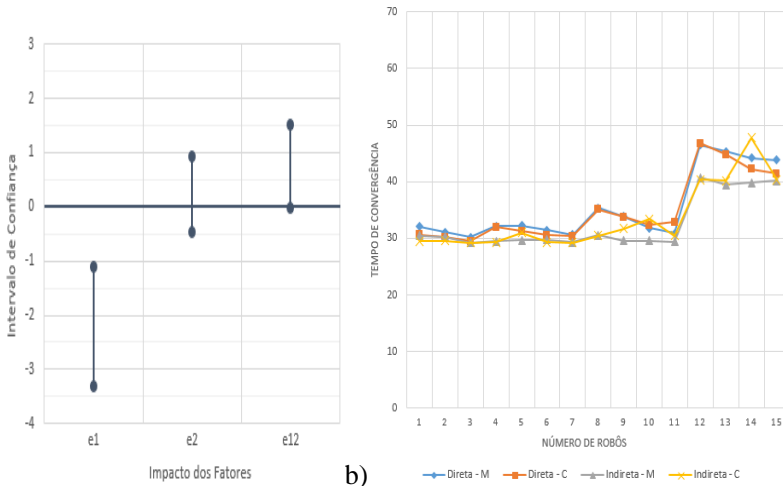
Nota-se pelo primeiro IC, e_1 - impacto da forma de comunicação, que a mesma é estatisticamente significativa, ou seja, que a sua utilização em topologias de grade é benéfica reduzindo o tempo de convergência do *rendez-vous*.

O IC resultante sobre o impacto do segundo fator (e_2) indica que é possível continuar afirmando que não há diferença estatística ao utilizar o consenso médio ou circuncentro, como indicado no caso sobre escalabilidade (Figura 38-b), mesmo ao incluir a comunicação indireta.

Esta conclusão também é corroborada pelo terceiro IC (e_{12}), que indica que não há interação significativa entre os fatores devido a inclusão de zero no IC.

Portanto, apesar da dificuldade em visualizar os ganhos utilizando a Figura 39-b, o IC da Figura 39-a fornece uma resposta clara sobre o impacto da comunicação, das estratégias e de suas interação.

Figura 39 – Resultados do primeiro experimento em termos de a) intervalo de confiança de 98% de cada impacto de cada fator e sua interação, e b) tempo de convergência das amostras para cada combinação de fatores.



a)

b)

O segundo experimento consiste em verificar se a comunicação indireta reduz os problemas detectados no exemplo de escalabilidade (subseção 6.3.3) para topologia linear.

Desta forma, é reproduzindo o experimento sobre escalabilidade com uma única alteração: os robôs utilizam a comunicação indireta, ou seja, utilizam a informação dos vizinhos de seus vizinhos diretos. A matriz experimental resultante é indicada no Quadro 16.

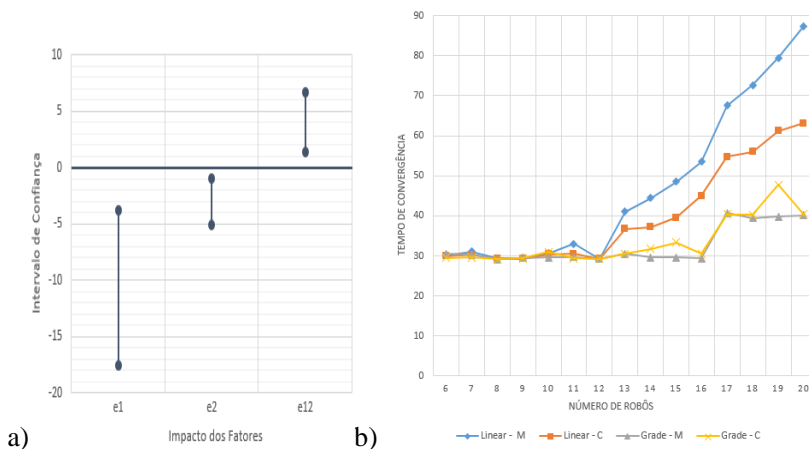
Neste experimento, o primeiro fator é a topologia de comunicação utilizada, onde a topologia linear é representada por “-” e a topologia de grade por “+”. O segundo fator é o consenso, de forma semelhante ao primeiro experimento.

Quadro 16 – Matriz experimental para amostras utilizando comunicação indireta.

Combinação de Fatores	Topologia (L x G)	Consenso (M x C)	Tempo Convergência (R_i^c)
1	-	-	R_i^1
2	+	-	R_i^2
3	-	+	R_i^3
4	+	+	R_i^4

Os resultados obtidos neste experimento são indicados na Figura 40, sendo indicados em a) na forma de intervalos de confiança de 98% construídos a partir da matriz experimental do Quadro 16 e em b) para os tempos de convergência das amostras de cada combinação de fatores.

Figura 40 - Resultados do segundo experimento em termos de a) intervalo de confiança de 98% de cada impacto de cada fator e sua interação, e b) tempo de convergência das amostras para cada combinação de fatores.



O problema de um intervalo cada vez maior, relacionado ao número de robôs, ocorre mesmo ao utilizar a comunicação indireta, com a diferença no número de robôs que o constante incremento começa a ocorrer (12 ao invés de 9) e na grande redução nos valores do tempo de convergência em relação ao caso de escalabilidade que utilizava apenas a informação dos vizinhos diretos.

Os valores obtidos apresentaram uma redução de até 147 segundos (amostra com 20 robôs, consenso médio). Seria possível elaborar um caso para comparar os resultados para comunicação direta/indireta para o caso linear, mas devido a notável redução, esta construção não é necessária para concluir que há melhora nos resultados.

Assim, respondendo aos questionamentos iniciais, a comunicação indireta melhora o desempenho da tarefa de *rendez-vous*. Esta melhora pode ser considerada por igual para ambas estratégias de consenso, sendo bem definida e limitada para o caso de topologia de grade. No caso da topologia linear, é possível afirmar que a redução pode ser elevada e significativa, mas que dependerá do número de robôs utilizados.

6.4 TRABALHOS CONJUNTOS

A segunda abordagem para validar a utilização do método, é a utilização do mesmo em outros trabalhos no mesmo grupo de pesquisa. Neste sentido, foram iniciados dois projetos onde o método é utilizado como na elaboração da simulação e na avaliação de resultados.

A proposta é demonstrar o uso do método para responder questionamentos relevantes para autores sobre suas respectivas técnicas. Desta forma, o primeiro projeto consiste em avaliar e propor melhorias para um algoritmo de consenso com imprecisão nas informações e o segundo, uma avaliação da propagação de informações em um NRS.

Como são projetos em andamento, os resultados são preliminares e não representam a sua totalidade. Sendo assim, é detalhado aqui a apresentação do estudo, os questionamentos a serem respondidos pelo método ao longo do projeto e os resultados preliminares utilizando o *framework* de simulação. Os resultados completos de cada projeto, serão apresentados no formato de artigos científicos e listados no *site* do método (RAMOS, 2017).

6.4.1 Comparação de Técnicas – Consenso FK

O consenso utilizando Filtro de Kalman é parte de um trabalho de mestrado em andamento na UFSC/PGEAS que visa lidar com as imprecisões dos sensores e da informação dos vizinhos na formulação de consenso para *rendez-vous* de robôs móveis utilizando o filtro de Kalman.

A proposta de trabalho em conjunto envolve em fornecer opções de ferramentas para o pesquisador, auxiliando-o no desenvolvimento de sua proposta. Estas ferramentas constituem um *framework* de simulação com consenso FK e nas ferramentas de validação, design experimental e avaliação de resultados.

O principal objetivo é utilizar o método para construir uma simulação integrada capaz de rodar o algoritmo de consenso FK e avaliar seu desempenho em algumas situações pontuais. Estas situações consistem de ambientes onde é possível ocorrer ruídos nos sensores dos robôs, ou seja, as informações são adquiridas pelo robô com imprecisão e repassadas aos robôs vizinhos; e de ambientes onde há perda de informação devido a problemas de comunicação.

As principais questões a serem respondidas neste projeto em conjunto são indicadas no Quadro 17.

Quadro 17 - Questionamentos em relação ao consenso FK.

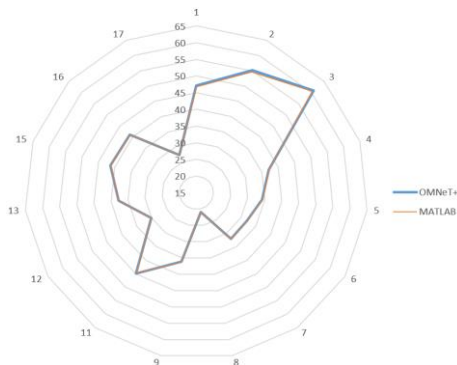
- Qual o desempenho do consenso FK em relação as demais técnicas?
- Como a alteração da formulação interna do consenso afeta os resultados?
- Como o peso γ afeta os resultados? Existe alguma forma de relacionar este parâmetro a outros fatores, como por exemplo, a distância ou intensidade do sinal de comunicação?
- Como o algoritmo se comporta na presença de ruídos? Qual o limite e tipo de ruídos toleráveis?
- O número de informações e de suas respectivas covariâncias, no consenso, afeta o resultado de forma diferente? Se sim, qual o impacto de um controle de topologia?
- Se os ruídos não forem homogêneos, qual a proporção de robôs com ruídos é tolerável?
- Qual o impacto da perda de informação devido a falhas na comunicação neste caso? Há melhora no desempenho em utilizar informações antigas, mesmo que imprecisas?

É esperado que as respostas destes questionamentos contribuam na definição dos parâmetros da técnica de consenso FK e na definição do escopo de atuação na qual a técnica possui desempenho aceitável.

Para utilizar esta técnica no *framework* de simulação no OMNeT++, foi realizada a validação da mesma, em relação à versão original desenvolvida no MATLAB, em etapas. A primeira etapa foi a versão do algoritmo que utilizava o par δq^i e $\delta \sigma^i$ dos robôs vizinhos estimados pelo próprio robô e o peso λ baseado no número de vizinhos.

A segunda validação foi elaborada utilizando o par repassado pelos outros robôs e o mesmo peso anterior, resultando na validação atual indicada na Figura 41, resultando em um intervalo de confiança (99% de confiança) sobre o erro de [0,2535, -1069].

Figura 41 - Processo de validação do consenso FK para pesos médios.

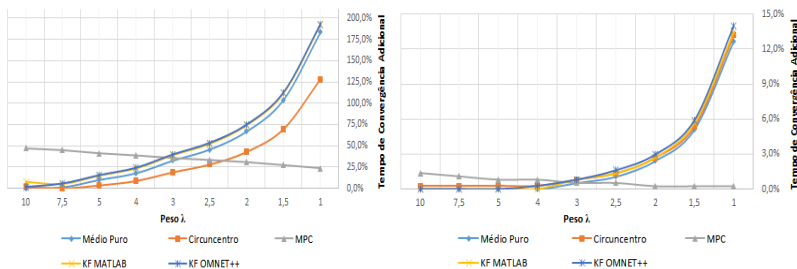


Os primeiros resultados apresentados abaixo utilizam esta versão da simulação, no entanto, o planejamento deste trabalho em conjunto contempla mais uma validação, a do consenso FK final, que utiliza a formulação dos pesos baseada na distância.

Apesar do foco deste projeto se concentrar na validação da técnica no OMNeT++, algumas análises iniciais foram realizadas para averiguar o impacto dos parâmetros no algoritmo inicial. A primeira análise foi referente ao impacto do peso no resultado da tarefa cooperativa e a comparação com os resultados obtidos utilizando as outras técnicas de *rendez-vous* (Figura 42). A simulação foi realizada utilizando controle de topologia fixa, períodos de controle de 100ms e discretização de 0,5 (em torno do valor padrão de 1,0) no peso de controle e na penalização da ação de controle (no caso do MPC), para 15 condições iniciais.

Os resultados individuais foram normalizados e apresentados em termos de porcentagem, indicando, portanto, a diferença de tempo de convergência do resultado em relação ao melhor tempo obtido nestas simulações. Foram gerados resultados para as 15 amostras, sendo apresentados aqui dois resultados (Figura 42) mais significativos: o de uma topologia linear (robôs conectados sequencialmente) e de uma topologia tri-conectada, onde com cada robô possui três links de comunicação.

Figura 42 - Influência do peso nos tempos de convergência de *rendez-vous*, para topologia linear (esquerda) e topologia tri-conectada (direita).



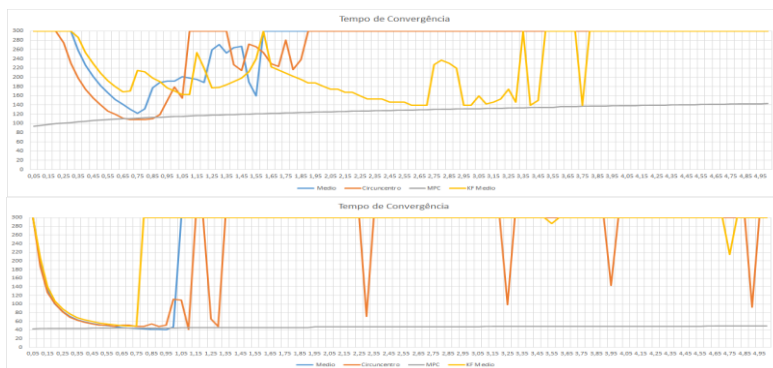
É observado que ao aumentar o peso no sinal de controle (direita para esquerda - Figura 42) a tarefa é concluída mais rapidamente (indicando um melhor resultado) e que, de forma inversa, ao aumentar a penalização do esforço de controle, na técnica MPC, o tempo de conclusão da tarefa aumenta, prejudicando o seu desempenho. Os resultados para ambas topologias apresentam curvas crescentes com a diminuição dos pesos, mas diferem significativamente no valor absoluto,

onde a topologia linear pode apresentar uma melhora de até 200% nos resultados contra 15% da topologia com mais conectividade.

Os resultados foram novamente reproduzidos com uma constante de tempo de 1s, para analisar a influência da mesma nos resultados, tanto para topologia linear quanto para a tri-conectada. Verificou-se que o comportamento não é o mesmo do que com uma menor discretização, com faixas de valores que produzem resultados melhores (Figura 43), tanto para topologia linear quanto para topologia tri-conectada.

A simulação da técnica no OMNeT++ e as análises iniciais citadas, contribuíram no trabalho do autor alvo deste trabalho em conjunto, em relação a decisão de quais parâmetros utilizar na comparação de técnicas, destacando que uma má escolha pode influenciar os resultados e conclusões. Os resultados também motivaram aprofundar estas análises em outra parceria, detalhada na próxima subseção.

Figura 43 - Tempo de convergência em valores absolutos utilizando variação nos pesos λ (0:0,05:5), constante de tempo de 1s e topologias linear e tri-conectada.



As próximas etapas planejadas para este trabalho são:

- Validação da simulação do consenso FK final, com peso baseado na distância e nas matrizes de covariação (técnica que está em desenvolvimento pelo coautor);
- Simular o Consenso FK em situações envolvendo parâmetros e falhas na rede de comunicação, sob a presença de ruídos de medição e/ou na informação dos vizinhos;
- Utilizar modelos de intensidade do sinal do WIFI no OMNeT++ e relacionar o peso do consenso FK baseado na distância com a intensidade do sinal, utilizando como referência o trabalho de Carvalho (2016);

Os resultados dos pontos listados acima serão apresentados em forma de artigo científico, planejado para primeiro semestre de 2018 e futuramente listados no *site* do projeto (RAMOS, 2017).

6.4.2 Propagação da Informação – Consenso Ponderado

A disseminação da informação no sistema cooperativo pode ser formulada conforme indicada na subseção 2.1.1. Em especial, há interesse de propor um algoritmo integrado ao consenso que seja capaz de filtrar falsas informações transientes, sem afetar drasticamente o desempenho da tarefa.

Como a formulação é semelhante ao do consenso, deseja-se estudar a disseminação da informação em um sistema de robôs cooperativos utilizando o consenso para relacionar as propriedades de ambas as áreas. Para isso, é proposto o trabalho conjunto, utilizando o método e o *framework* de simulação analisar a propagação de informações no consenso ponderado.

Primeiramente, foi estabelecido como objetivo uma análise inicial sobre o impacto na modificação destes pesos no consenso. Em seguida, o objetivo do trabalho passa a ser a fornecer ferramentas para analisar a implementação da formulação da epidemia no consenso. As principais questões a serem respondidas neste projeto em conjunto são indicadas no Quadro 18.

Quadro 18 - Questionamentos em relação à propagação de informações.

- Qual o impacto de uma falsa referência, por um tempo limitado, no sistema cooperativo?
- Qual a duração e intensidade desta falsa referência?
- As taxas de contato e de reprodução afetam o desempenho dos robôs cooperativos?
- As taxas de contato e reprodução conseguem amenizar o impacto da referência falsa? Até qual valor ou frequência é possível afirmar que o impacto não foi significativo?
- Ao utilizar uma topologia dinâmica, as conclusões anteriores ainda são verdadeiras?

A simulação deste caso é baseada na formulação do consenso médio, $\varepsilon_{i,k}$, indicada na Equação (6.1).

$$\varepsilon_{i,k} = \sum_{j=1}^n \lambda_k \cdot \frac{a_{ij,k}}{n_{v,k}} \cdot (\zeta_{i_k} - \zeta_{j_k}) \quad (6.1)$$

Onde o termo ζ_{i_k} representa a informação do robô (nó) i no atual instante k , ζ_{j_k} a última informação recebida do robô vizinho j , $a_{ij,k}$ o elemento da matriz adjacência (zero ou um), $n_{v,k}$ é o número total de vizinhos e λ_k o valor de peso utilizado. O resultado do consenso é

limitado à um valor de saturação e utilizado para atualizar a informação do próprio robô, como indicado em (6.2).

$$\zeta_{i,k+1} = \zeta_{i,k} + \text{sat}(\varepsilon_{i,k}) \cdot h \quad (6.2)$$

A validação desta simulação é trivial, visto que foi necessário apenas adaptar a formulação de *rendez-vous* médio para receber os pesos individualmente, obtendo resultados idênticos ao algoritmo anterior.

Quanto as condições iniciais do experimento, as mesmas foram padronizadas quanto a posição e quanto a sua topologia, para apenas dois casos: topologia linear e topologia de grade (Figura 44). A escolha foi baseada nas condições iniciais utilizadas por Ramachandran (2017), que em seu trabalho utiliza estas topologias e as associa com conceitos de observabilidade e rejeição a ruídos. Esta associação é de interesse ao coautor deste trabalho e uma possível comparação de resultados foi a principal motivação desta escolha.

Figura 44 - Topologia de comunicação linear (esquerda) e de grade (direita) para mesmas condições iniciais.

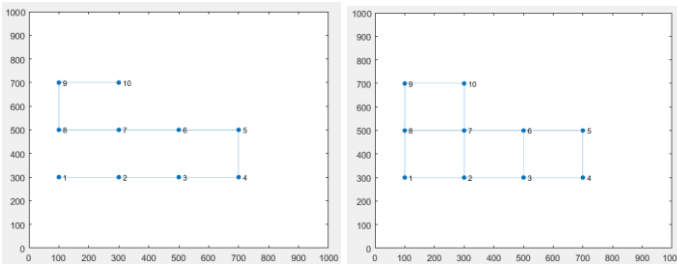
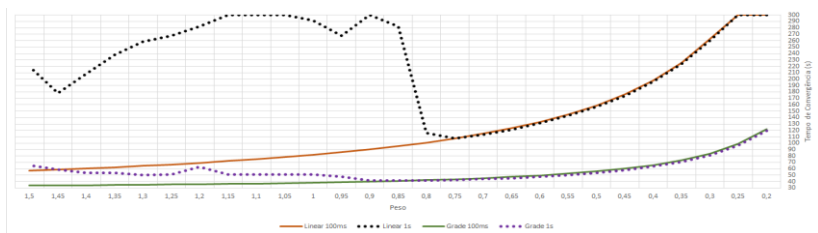


Figura 45 - Tempo de convergência da tarefa de *rendez-vous* ao utilizar diferentes pesos, para topologia linear (linhas superiores) e grade (linhas inferiores), para os constantes de tempo de 100ms (linha sólida) e 1s (linha pontilhada).



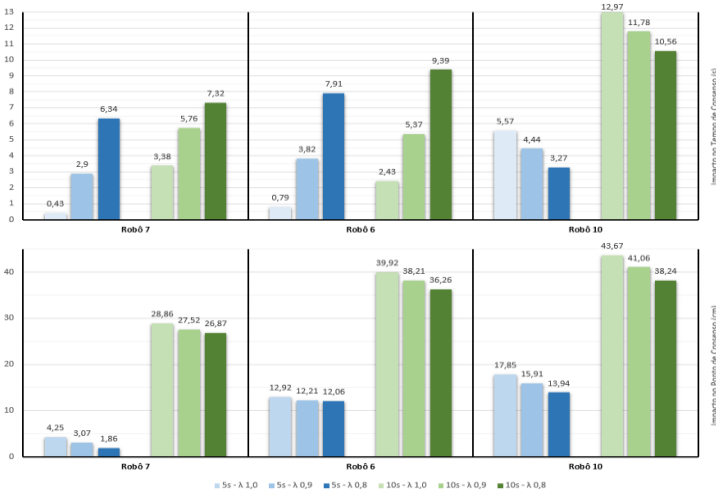
O estudo inicial concentrou-se no desempenho da tarefa cooperativa quando submetida a diferentes pesos λ no consenso, sendo verificado também como cada topologia reage a esta modificação. Este

experimento foi elaborado para uma constante de tempo de 100ms e repetido para os casos de 500ms e 1s, resultando nos gráficos indicados na Figura 45 que representam o tempo de convergência de cada caso ao utilizar determinado peso.

Os resultados indicaram que não há diferença entre as constantes de tempo de 100ms e 500ms, sendo o segundo omitido e o primeiro indicado em uma linha sólida. Há formação de uma curva crescente com a diminuição do peso, indicando perda de desempenho, principalmente para a topologia linear. No entanto, para a constante de tempo de 1s (linha pontilhada), os resultados apresentam grandes variações e também há perda de desempenho para pesos acima do valor 1.

Em seguida, foi inserida na simulação uma referência temporária falsa em um robô, com intensidade e duração controlada, verificando o seu impacto no tempo e ponto de convergência. Neste caso, é adicionado mais um termo no somatório da Equação (6.1), incrementando também o valor de $n_{a_1,k}$, ou seja, para o robô é apenas mais um membro do consenso. As simulações foram realizadas apenas para constante de tempo de 100ms.

Figura 46 - Diferença nos resultados ao inserir uma referência falsa com duração de 5s (azul) e de 10s (verde), nos robôs 7, 6 e 10, com redução do peso do robô.



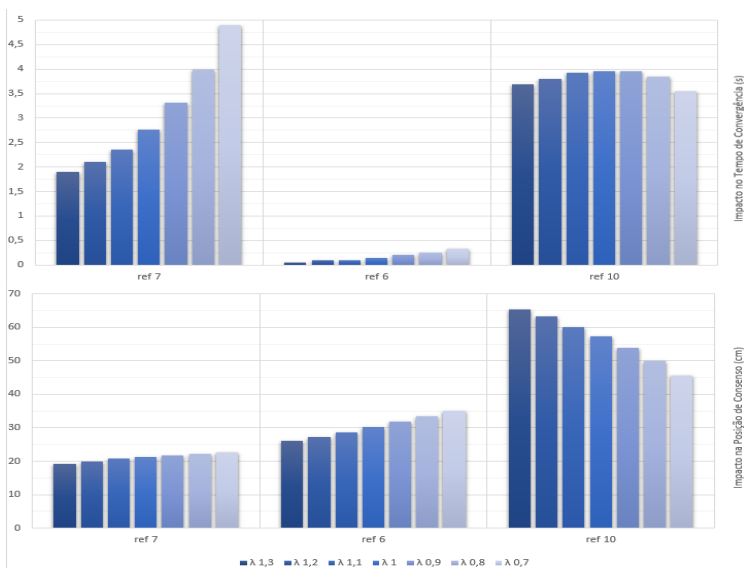
Para o experimento foi utilizada uma referência à 500 unidades do robô que recebe a recebe e com duração de 5 e 10s. Também foi verificado o efeito de diminuir o peso da informação deste robô pelos seus vizinhos e o impacto desta alteração no desempenho, sendo aqui exemplificado

para o caso da topologia linear na Figura 46. Neste gráfico são indicados os erros no tempo e na distância, para referência com duas durações distintas (5s e 10s), em relação ao caso sem a falsa referência, monitorando também os resultados à medida que os vizinhos diminuem o peso deste robô em seu consenso.

Analisando os gráficos, conclui-se que não é possível afirmar que os resultados terão melhor ou pior desempenho, visto que, por exemplo, há melhoras no tempo e no ponto de convergência para o robô 10 (duração de 5s e 10s), pior desempenho no tempo ao aplicar a redução no robô 7 (para ambas durações) e praticamente não há efeito da redução para o ponto de convergência no caso do robô 6 (duração de 5s).

Outro experimento realizado considerou apenas uma duração da falsa referência, mas considerou a redução dos pesos na rede inteira, sem distinção de qual robô possivelmente estaria contaminado, sendo exemplificado para a topologia de grade na Figura 47.

Figura 47 - Diferença nos resultados ao inserir uma referência falsa e alterar os pesos de todos os robôs.



Por fim, na análise mais recente, foi elaborada um caso para verificar se limitar o valor da referência utilizado no consenso afeta significativamente os resultados.

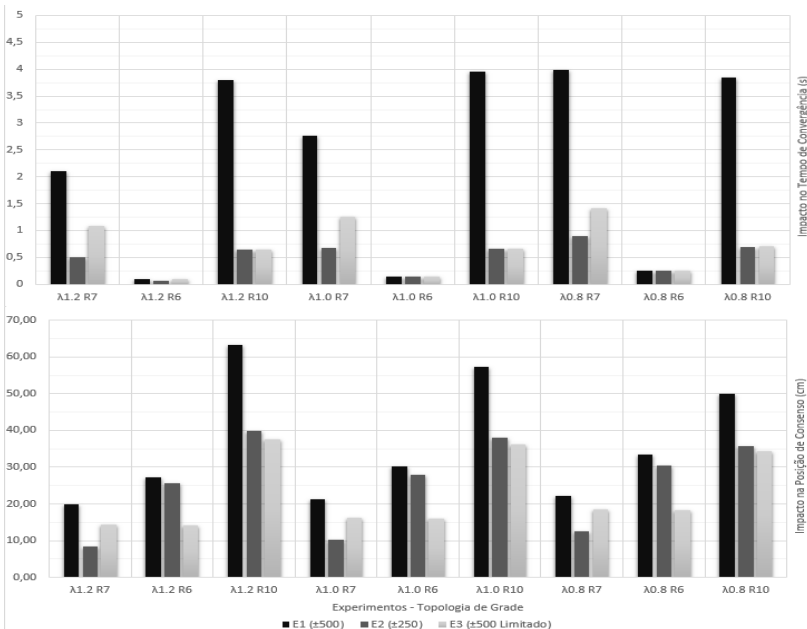
Para isto, foram utilizados dois casos com referência à 250 e 500, respectivamente, e um caso onde uma referência à 500 unidades estava sujeita a uma saturação. Esta saturação limita a diferença entre o robô e a referência baseado no alcance máximo do raio de comunicação, que para este caso, é de 250 unidades.

Os resultados obtidos são indicados na Figura 48, para três pesos em três robôs, sendo as barras escuras as referências à 500 unidades, as barras intermediárias as referências à 250 unidades e as barras mais claras o caso limitado.

Os resultados obtidos com a saturação foram satisfatórios em termos da melhora do desempenho, tanto no tempo de convergência quanto no ponto de convergência, visto que os resultados obtidos foram semelhantes ao de se utilizar uma referência de 250 unidades.

Todas as análises iniciais contribuíram para demonstrar que uma simples alteração dos pesos sem uma melhor análise da topologia e dos valores dos pesos, não garante a melhora dos resultados e nem uma rejeição à perturbação, mas que os casos pontuais indicam uma possível determinação de valores que ajudem na recuperação dos robôs.

Figura 48 - Impacto da intensidade da referência e o desempenho de um consenso com valores limitados da diferença entre posições.



Também ficou claro a importância da escolha destes parâmetros, sendo evidente o impacto de simples contramedidas adicionais, como a alteração da topologia de comunicação e a saturação da informação utilizada no consenso.

As próximas etapas planejadas para este trabalho são:

- Utilização de pesos fixos provindos da formulação da epidemia, calculados para reduzir o impacto de ruídos (técnica em desenvolvimento pelo coautor).
- Refazer as análises ao utilizar simultaneamente os pesos da epidemia do ponto anterior e outras contramedidas, como a mudança dinâmica de topologia e a limitação da informação utilizada no consenso.

Os resultados dos pontos acima serão apresentados em forma de artigo científico, planejado para primeiro semestre de 2018 e futuramente listados no *site* do projeto (RAMOS, 2017).

6.5 DIVULGAÇÃO EXTERNA

A terceira abordagem para obter respostas sobre a eficácia do método é promover a sua utilização por pessoas externas ao projeto. Para isto, primeiramente foi criado um nome para o método para melhor identificação e rastreamento do mesmo, sendo atribuído o nome de *Integrated Robotic and Network Simulation Method* (IRoNS Method), que traduzido significa “Método para Simulação Integrada de Robótica e de Redes de Comunicação”.

O segundo passo foi disponibilizar as informações básicas sobre o método em um *site* no idioma inglês (RAMOS, 2017) (Figura 49), tornando o método acessível a qualquer público interessado.

No *site*, além das informações sobre o método, estão disponíveis as informações sobre as publicações, dúvidas frequentes, autores e informações para contato.

Não é esperado que esta abordagem tenha um efeito imediato, mas sim a longo prazo, principalmente à medida que sejam elaboradas mais publicações indicando o mesmo.

A recepção das publicações tem sido positiva desde o início do projeto, sendo aceitos artigos elaborados durante a concepção do método (RAMOS, 2014), durante a sua elaboração (RAMOS, 2015a; 2015b e 2015c) e de sua aplicação após conclusão (RAMOS, 2016).

Figura 49 – Página inicial do *site* para divulgação do método IRoNS.

6.6 CONSIDERAÇÕES FINAIS

Neste capítulo foram descritos os estudos de caso criados utilizando o método para uma diversidade de problemas relacionados ao controle cooperativo, redes de comunicação e controle de topologia.

Foram adotadas três estratégias para validar e testar o método: elaboração de estudos de caso próprios, utilização em trabalhos de terceiros (projeto em conjunto) e a disponibilização externa do método em artigos científicos e na Internet.

A primeira abordagem apresentou resultados concretos, apontando comportamentos e interações interessantes entre as técnicas escolhidas para simulação. Já o trabalho em conjunto, mesmo ainda estando em andamento, apresentou resultados valiosos para os respectivos coautores, ajudando no andamento de seus projetos e no planejamento de futuras publicações. Portanto, esta abordagem também colaborou com a terceira, pois quanto mais o método for divulgado e utilizado, mais *feedback* se terá sobre a utilização do mesmo.

Capítulo 7

Conclusão

A pesquisa na área da robótica cooperativa tem se tornado a cada dia mais importante, dada a complexidade dos sistemas robóticos e de sua maior utilização nas mais diversas aplicações. Aliada a esta complexidade, a interconectividade entre robô é um fator constantemente presente nestes sistemas e que apesar de trazer ganhos significativos como menor custo e redundância, possuem certo impacto no desempenho das tarefas robóticas.

A compreensão deste impacto muitas vezes não fica claro ao desenvolvedor durante a fase de simulação e não é levado em consideração até a implementação em sistemas reais. No entanto, a detecção de comportamentos não desejáveis durante a implementação se torna mais complexa, visto que o sistema real possui uma série de outros fatores que também podem estar influenciando os resultados.

Levando este ponto em consideração e que nem sempre um sistema real está disponível ao desenvolvedor, é proposto ao longo da tese, uma abordagem diferenciada ao propor a utilização de um método para guiar a construção de simulações multidisciplinares, ou seja, simulações envolvendo os tópicos de controle cooperativo, redes de comunicação e de controle de topologia.

Ao longo deste documento foram discutidos os principais pontos relativos ao *framework* e ao método, detalhando e exemplificando cada um de seus procedimentos. Estes procedimentos abrangem tanto o planejamento inicial, com informações básicas sobre o estudo, quanto o planejamento da simulação e de sua validação, para então planejar os estudos de caso relevantes ao objetivo do estudo, que possibilitem

descobrir comportamentos não previstos nas simulações individuais de controle cooperativo. Os estudos de caso, ao final, são analisados utilizando uma ferramenta estatística para facilitar a compreensão dos resultados e relacionar o impacto entre parâmetros de diferentes algoritmos na tarefa cooperativa.

Para exemplificar a utilização do método, foi utilizado um conjunto de técnicas de outros autores que possuem simulações individuais em outros softwares, mas que não haviam sido simultaneamente simuladas e que, portanto, poderiam possuir interações que poderiam afetar os resultados de forma desconhecida.

A utilização do método nestas técnicas possibilitou verificar quais características do método apresentavam algum déficit e que necessitavam revisões, adaptações e incorporações de outras técnicas.

Dentre as técnicas adicionadas, é proposta a separação da modelagem conceitual e comunicativa, incorporando a utilização do protocolo ODD para a última. Outra adição, a validação em quatro passos aliada à análise estatística por intervalo de confiança, permite uma rápida análise e validação do algoritmo de controle cooperativo. E por fim, a elaboração planejada dos estudos de caso com a análise estatística dos resultados, permite verificar o impacto de diferentes técnicas e parâmetros.

O método em sua versão final, é também foi em aplicações além do conjunto de técnicas definidas a priori, sendo aplicado no estudo com outros autores sobre o desempenho do consenso FK e na análise da disseminação de informações no sistema cooperativo.

Em todos os estudos de casos foram descobertos novos comportamentos, não previstos anteriormente, e que, só foram possíveis de serem analisados na fase de simulação por meio da utilização do método e do *framework* de simulação integrado.

O interesse da comunidade científica neste tema é evidente pela aprovação dos artigos científicos e das avaliações recebidas, motivando a divulgação do método em inglês em um *site* próprio, para livre uso de outros desenvolvedores.

7.1 CONTRIBUIÇÕES DA TESE

As principais contribuições desta tese são listadas a seguir.

- i) *Formulação de um framework aliado a um método para elaboração de simulações integradas de controle cooperativo, controle de topologia e de redes de comunicação.*

- ii) *Demonstração de uso do método em um conjunto de técnicas, com apresentações de descobertas reais, sendo estas documentadas e publicadas.*
- iii) *Utilização do método em trabalhos de outros autores envolvendo outras técnicas, com igual obtenção de resultados documentados, publicados e relevantes para o avanço de seus projetos.*
- iv) *Determinação de um escopo inicial de questionamentos, cuja a resposta é possível de ser obtida por meio da utilização do método e de um framework de simulação integrado.*
- v) *Disponibilização do método completo de forma online, em inglês, sob o acrônimo IRONS Method.*

7.2 DIFICULDADES E PERSPECTIVAS FUTURAS

Durante o desenvolvimento deste trabalho houveram diversas dificuldades, das quais demandaram uma maior dedicação e tempo para prosseguir o trabalho e que por consequência, limitaram o número de estudos de caso elaborados.

A primeira dificuldade, logo ao início do projeto, foi o pouco material encontrado sobre métodos e sobre simulações de robôs cooperativos e redes de comunicação. Esta situação foi parcialmente superada, ao tomar conhecimento de termos e siglas específicas de algumas áreas, como NRS, NCS, MWSN e IoRT, obtendo uma boa base de referências que abordavam ao menos alguns aspectos da simulação integrada proposta. No entanto, em relação ao método, o avanço só foi possível após aquisição de material bibliográfico específico da área de VV&T, nas quais forneceram *insights* da área e melhor compreensão do que seria necessário constar no método.

A multidisciplinaridade da aplicação e dado ao fato de não haver outras pessoas auxiliando na implementação da simulação, aumentaram consideravelmente a dificuldade do projeto.

Um ponto que se havia desejo, mas que não houve progresso, foi a atualização do simulador utilizado de sua versão de 2014 para a versão de 2017. No entanto, devido ao fato de que houveram significativas alterações nos comandos e interface do simulador, percebeu-se que seria necessário dedicar um tempo a mais na atualização do código que não traria ganhos significativos para o projeto.

A partir das limitações encontradas e das oportunidades de expansão que o método possui, são propostos os seguintes pontos para trabalhos futuros:

- Utilizar o método, em sua forma atual, em outros conjuntos de técnica que possuam relevância e possam ter interações entre si por meio de um framework de simulação atualizado, otimizando os parâmetros da simulação para encontrar uma combinação de parâmetros que otimize a tarefa cooperativa.
- Utilizar o método para outras tarefas, estabelecendo critérios de desempenho e avaliação, demonstrando para quais tarefas a utilização do método ainda é satisfatória ou se há necessidade de melhorias.
- Expandir o número de tópicos abordados no método, incorporando tópicos como consumo de energia, ambientes com obstáculos e com colisão entre robôs, e outros fatores que aumentem a precisão da simulação.
- Promover e acompanhar a utilização do método por outros autores, propondo melhorias ao método e elaborando um *framework* de simulação genérico que pode ser facilmente adaptado para qualquer aplicação de robótica cooperativa.

7.3 INFORMAÇÕES REFERENTES AO DOUTORADO

O doutorado teve início em março de 2014, com período de estágio sanduíche de 8 meses concluído em 2015. No momento da entrega deste documento, foram publicados cinco artigos relacionados ao assunto da tese em congressos nacionais e internacionais, com submissão de um artigo para um periódico internacional (em revisão) e planejamento futuro de ao menos outros dois artigos para primeiro semestre de 2018.

7.3.1 Doutorado Sanduíche

O doutoramento sanduíche foi realizado em Portugal na Faculdade de Engenharia da Universidade do Porto (FEUP), especificamente no laboratório do DaRTES (Distributed and Real-Time Embedded Systems), sob responsabilidade e supervisão do prof. Dr. Luis Almeida durante o período de 05/01/2015 à 05/09/2015.

O objetivo do doutorado sanduíche foi a implementação conjunta de protocolos de comunicação e estratégias de controle cooperativo de um grupo de robôs móveis cooperativos. O principal foco neste período foi a avaliação do impacto do protocolo de comunicação na estratégia de controle, analisando o comportamento dos robôs em diversas situações, inclusive com perdas de mensagens durante a simulação.

Este período foi fundamental para implementação do primeiro *framework* de simulação, permitindo avaliar quais os principais pontos necessários para o método.

7.3.2 Lista de Publicações

Até o momento da elaboração deste documento foram realizadas as seguintes publicações relacionadas a tese:

RAMOS, Daniel C.; MORENO, U. F. (2014) **OMNET++ como Ferramenta de Simulação e Análise de Protocolos de Comunicação em Estratégias de Controle Cooperativo**. In: *1º Simpósio de Tecnologia e Ciência - UFU*, resumo expandido, ISSN: 2359-3415, v. 1. p. 43-46, Patos de Minas – MG, Brasil.

RAMOS, Daniel C.; ALMEIDA, L.; MORENO, U. F. (2015) **Cooperative Control Strategy Based on Consensus with MANET Protocols and Topology Control**. 1th Doctoral Congress in Engineering, resumo e poster em inglês, Universidade do Porto.

RAMOS, Daniel C.; ALMEIDA, L.; OLIVEIRA, L.; MORENO, U. F. (2015) **Network Interference on Cooperative Mobile Robots Consensus**. In: *Robot 2015 - Second Iberian Robotics Conference*, Advances in Robotics, Vol. 1, artigo completo em inglês, Springer, pp. 651-664, DOI: 10.1007/978-3-319-27146-0, Lisboa, Portugal.

RAMOS, Daniel C.; ALMEIDA, L.; OLIVEIRA, L.; MORENO, U. F. (2015) **Impact of Alien Networks on Consensus in a Team of Cooperative Mobile Robots**. In: *V Brazilian Symposium on Computing Systems Engineering*, artigo completo em inglês, pp. 84-89, IEEE, DOI: 10.1109/SBESC.2015.23, Foz do Iguaçu – PR, Brasil.

RAMOS, Daniel C.; ALMEIDA, L.; MORENO, U. F. (2016) **Method for Design and Performance Evaluation of Ad Hoc Networked Mobile Robotic Systems Using OMNET++**. In: *4th IFAC Symposium on Telematics Application 2016*, IFAC-PapersOnLine, Volume 49, Edição 30, pp 144-149, ISSN: 2405-8963, artigo completo em inglês, Porto Alegre - RS, Brasil.

REFERÊNCIAS

- ABUHAIBA, I. S.; Hassan, M. A. (2011) **Interference reduction in mobile ad hoc and sensor networks**. Journal of Engineering and Computer Innovations (JEI), 2:138–154.
- ACCELLERA (2016) **SystemC**. Disponível em: <<http://accellera.org/community/systemc/>>, acessado em abril de 2016.
- AHMADI, M.; Stone, P. (2006) **Keeping in Touch: Maintaining Biconnected Structure by Homogeneous Robots**. In: The Twenty-First National Conference on Artificial Intelligence (AAAI), 580–85.
- AKEWAR, M. C. et al. (2012) **A Study of Wireless Mobile Sensor Network Deployment**. IRACST – International Journal of Computer Networks and Wireless Communications, ISSN: 2250-3501. 2(4).
- AL-HAMMOURI, A. et al. (2008) **Co-simulation Tools for Networked Control Systems**. HSCC 2008, LNCS 4981, 16-29, Springer.
- ALANYALI, M. et al. (2004) **Distributed bayesian hypothesis testing in sensor networks**, in Proc. of ACC, (Boston, MA), 5369–5374.
- ANDERSON, M. et al. (2008) **Implicit cooperation strategies for multi-robot search of unknown areas**, Journal of Intelligent and Robotic Systems, 53(4):381–397.
- BAER, P. A. (2008) **Platform-Independent Development of Robot Communication Software**. Tese de doutorado, Universidade de Kassel – Alemanha.
- BANKS, J. et al. (2004) **Discrete-Event System Simulation**. 4ª Edição, Editora Prentice Hall, ISBN-10: 0131446797.
- BALCI, Osman. (1995) **Principles and Techniques of Simulation Validation, Verification and Testing**. Proceedings of the 1995 Winter Simulation Conference.
- BALCI, O. (1998) **Verification, validation, and testing**. In *The Handbook of Simulation*, Chapter 10, J. Banks, Ed. John Wiley & Sons, New York.
- BALCI, O. (2013) **Introduction to Modeling and Simulation**. Curso, Slides de Aula 1 a 22, ACM SIGSIM, disponível em: <<http://www.acm-sigsim-mskr.org/Courseware/Balci/introToMS.htm>>, acessado em junho de 2016.
- BARROS NETO, B. et al. (2003) **Como Fazer Experimentos: pesquisa e desenvolvimento na ciência e na indústria**. 2ª Edição, Capítulos 1-4, Editora UNICAMP, Brasil.
- BAUER, N. W. et al. (2014) **Exploring the Boundaries of Robust Stability under Uncertain Communication: An NCS Toolbox Applied to a Wireless Control Setup**. IEEE Control Systems, 34(4):65-86.
- BATTISTELLI, G. et al. (2016) **Stability of consensus extended Kalman filter for distributed state estimation**. In: Automatica, 68 (2016), pp. 169–178, Elsevier.
- BEKMEZCI, I. et al. (2013) **Flying Ad-Hoc Networks (FANETs): A survey**. In: Ad Hoc Networks 11 (2013), pp. 1254-1270, DOI: 10.1016/j.adhoc.2012.12.004.
- BERNON, C. et al. (2003) **Adelfe: A methodology for adaptive multi-agent systems engineering**. In: Engineering Societies in the Agents World III, LNCS, vol. 2577, pp. 70–81. Springer Berlin.

- BEURON, Razvan. (2013) **Introduction to Network Emulation**, ISBN 978-981-4364-09-6, Pan Stanford Publishing.
- BIGGS, G. et al. (2012) **And all the robots merely Players**. Artigo completo, disponível em: <https://www.willowgarage.com/sites/default/files/player_retrospective.pdf>, acessado em abril de 2016.
- BODO, Z. et al. (2015) **High-level strategy design in Robocup enviroment**. In: 16th IEEE International Symposium on Computational Intelligence and Informatics (CINTI), 129-134.
- BRANDÃO, A. S.; Pizatta, I. H. B.; Sarcinelli, M.; Carelli, R. (2013) **Controle líder-seguidor de uma formação VANT-VTNT**, SBAl, disponível em: <<http://www.sbai2013.ufc.br/pdfs/5298.pdf>>.
- BURKHART, M.; Rickenbach, P.; Wattenhofer, R.; Zollinger, A. (2004) **Does topology control reduce interference?** In: Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing - MobiHoc '04. ACM Press, ISBN 1581138490.
- CAMACHO, E. F.; Alba, C. B. (2007) **Model Predictive Control**. 2a Edição, Springer, Advanced Textbooks in Control and Signal Processing.
- CAMPONOVARA, E.; Jia, D.; Krogh, B. H.; Talukdar, S. (2002) **Distributed model predictive control**. *IEEE Control Systems Magazine*, 22(1): 44–52.
- CARVALHO, Sidney R. D. (2015) **Algoritmos para Controle de Topologia em Redes de Robôs**. Dissertação de mestrado em engenharia de automação e sistemas, UFSC.
- CARVALHO, Sidney R. D. et al (2016) **Improving robustness of robotic networks using consensus and wireless signal strength**. IFAC-PapersOnLine, v. 49, n. 30, p. 337–342.
- CASTALIA (2013) **Castalia**. Disponível em: <<https://castalia.forge.nicta.com.au/index.php/en/>>, acessado em abril de 2016.
- CASTEIGTS, A.; Nayak, A.; Stojmenovic, I. (2010) **Topology control in sensor, actuator and mobile robot networks**. In: Wireless Sensor and Actuator Networks: Algorithms and Protocols for Scalable Coordination and Data Communication, cap. 7, 185 – 206.
- CHEN, L. et al. (2015) **A Nonlinear Formation Control of Wheeled Mobile Robots with Virtual Structure Approach**. In: Proceedings of the 34th Chinese Control Conference, pp. 1080-1085, DOI: 10.1109/ChiCC.2015.7259784.
- CHOWDHURY, A. K. (2017) **IoT and Robotics: A Synergy**. PeerJ Preprints, DOI: 10.7287/peerj.preprints.2760v1.
- CORREIA, L.; Macedo, D.; Santos, A.; Nogueira, J.; Loureiro, A. (2005) **Uma Taxonomia para Protocolos de Controle de Acesso ao Meio em Redes de Sensores Sem Fio**. Relatório Técnico, RT.DCC.005/2005, UFMG.
- CORREIA, F. L. B. e Moreno, U. F. (2014a) **Controle de formação com seguimento de referência para grupos de veículos autônomos utilizando consenso e RHC**, dissertação de mestrado em engenharia de automação e sistemas, UFSC.
- CORREIA, F. L. De B.; Ordoñez, B.; Cerqueira, J.; Almeida, L.; Moreno, U. F. (2014b) **Controle de Veículos Autônomos em Formação com Seguimento de Referência Utilizando Consenso e RHC**. Congresso Brasileiro de Automática - CBA, Brasil.
- CORTES, J. et al. (2006) **Robust Rendez-vous for Mobile Autonomous Agents via Proximity Graphs in Arbitrary Dimensions**. *IEEE Transactions on Automatic Control*, 51(8).

- COSTA, L. F. S. et al. (2015) **A Distributed Simulation Platform using HLA for Complex Embedded Systems Design**. In: 19th International Symposium on Distributed Simulation and Real Time Applications, DOI 10.1109/DS-RT.2015.16.
- COUCEIRO, M. (2011) **A novel multi-robot exploration approach based on particle swarm optimization algorithms**. In Proceedings of IEEE Inter-national Symposium on Safety, Security, and Rescue Robotics (SSRR), 327–332.
- CHRISTOFIDES, P. et al. (2013) **Distributed model predictive control: a tutorial review and future research directions**. Computers and Chemical Engineering, 51(5): 21–41.
- CHOSSET, Olivier; Lynch, Kevin; Thrun, Sebastian; et al. (2005) **Principles of Robot Motion: Theory, Algorithms, and Implementations**. ISBN: 9780262033275.
- DALLE, Olivier. (2007) **The OSA Project: an Example of Component Based Software Engineering Techniques Applied to Simulation**. SCSC 2007, ISBN: 1-5655-316-0.
- DARIO, L. G. et al. (2003) **Distributed consensus protocols for coordinating buyers**” in Proc. of CDC, (Maui, Hawaii), 588– 592.
- DAS, S. R. et al. (2007) **Performance comparison of two on-demand routing protocols for ad hoc networks**, *INFOCOM*, Tel Aviv, Israel.
- DELGOSHAELI, P. et al. (2014) **A Semantic Framework for Modeling and Simulation of Cyber-Physical Systems**. International Journal on Advances in Systems and Measurements, 7(3-4).
- DEVESELEER, Louis (2013) **Implementation of consensus algorithms on mobile robots to solve cooperation problems**. Projeto final de curso: relatório de estágio supervisionado no exterior, Université de Mons / UFSC.
- DORIYA, Rajesh. et al. (2015) **A Brief Survey and Analysis of Multi-robot Communication and Coordination**. International Conference on Computing, Communication and Automation, ISBN: 978-1-4799-8890-7, 1014-1021.
- DUCHON, F. et al. (2016) **Homogenous Multi-Robot System for Mapping of Unknown Environment**. IEEE 14th International Symposium on Applied Machine Intelligence and Informatics, DOI: 10.1109/SAMI.2016.7422975, 17-22.
- EARL, M. G.; D’Andrea, R. (2005) **Multi-vehicle cooperative control using mixed integer linear programming**. In Cooperative Control of Distributed Multi-Agent Systems, J. S. Shamma ed., John Wiley & Sons.
- ERCIYES, K. et al. (2010) **Modeling and Simulation of Mobile Ad hoc Networks**. Research Gate, disponível em: <www.researchgate.net/publication/228554550_Modeling_and_Simulation_of_Mobile_Ad_hoc_Networks/links/09e4150d41b0662286000000.pdf>.
- ESPOSITO, J. M. (2010) **Decentralized Cooperative Manipulation with a Swarm of Mobile Robots: the approach problem**. American Control Conference (2010), Baltipp. 4762-4767, 10.1109/ACC.2010.5531109.
- FARINELLI, A.; Iocchi, L.; Nardi, D. (2004). **Multirobot systems: a classification focused on coordination**. IEEE Trans. Syst. Man Cybernetics B, 34(5):2015-2028.
- FAX, J. A.; Murray, R. M. (2004) **Information flow and cooperative control of vehicle formations**. IEEE Transactions on Automatic Control, 49:1465 – 1476.
- FELDT, R. et al. (2010) **Validity threats in empirical software engineering research-an initial survey**. In: Proceedings of the Conference on Software Engineering and Knowledge

Engineering (SEKE), pp. 374-379.

FENG, Zhi. et al. (2016) **Robust Connectivity Preserving *Rendez-vous* of Multi-Robot Systems under Unknown Dynamics and Disturbances**. In: IEEE Transactions on Control Network Systems, DOI:10.1109/TCNS.2016.2545869.

FERRARI-TRECCATE, G. et al. (2009) **Model predictive control schemes for consensus in multi-agent systems with single- and double-integrator dynamics**. In *IEEE Transactions on Automatic Control*, 54(11): 2560–2572.

GARRO, A. et al. (2010) **EasyABMS: A domain-expert oriented methodology for agent-based modeling and simulation**. *Simulation Modelling Practice and Theory* 18(10):1453 – 1467.

GASPERO, L. e ANGELO (2013) **Quadprog++ Project**, disponível em: < quadprog.sourceforge.net/ >, acessado em 06/2015.

GATTUPALLI, A. et al. (2013) **A simulation framework for evolution on uneven terrains for synchronous drive robot**, *Advanced Robotics* Vol. 27 , Iss. 8..

GHORBANI, A., et al. (2013) **Maia: a framework for developing agent-based social simulations**. *Journal of Artificial Societies and Social Simulation* 16(2):9.

GIL, P.; Nunes, G.; Santos, A.; Cardoso, A. (2011) **Adaptive Model Based Predictive Networked Control over WSN with Tolerance to Transmission Faults on the Forward Channel**, 2011 Fifth International Conference on Sensing Technology, IEEE, ISSN 2156-8065, 341 – 346.

GIL, A. F. A. et al. (2015) **Behavior-based connectivity control for robot networks**. *Revista Lasallista de Investigación*, ISSN: 1794-4449, 12(1).

GOMEZ-SANZ, J. J. et al. (2010) **Model driven development and simulation with the INGENIAS agent framework**. *Simulation Modelling Practice and Theory*, 18:1468–1482.

GORE, R. et al. (2013) **The Need for Usable Methods in Verification and Validation**. In: *Proceedings of the 2013 Winter Simulation Conference*, IEEE, DOI: 10.1109/WSC.2013.6721513.

GORRIERI, A. et al. (2016) **Clustering and sensing with decentralized detection in vehicular ad hoc networks**. *Ad Hoc Networks* 36(2016) 450-464, Elsevier.

GOWAL, S.; Martinoli, A. (2012) **Real-Time optimization of trajectories that guarantee the *rendez-vous* of mobile robots**. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3518–3525.

GRIMM, V. et al. (2006) **A standard protocol for describing individual-based and agent-based models**. In: *Ecological Modelling*, 198:115-126.

GRIMM, V. et al. (2010) **The ODD protocol: A review and first update**. In: *Ecological Modelling*, DOI: 10.1016/j.ecolmodel.2010.08.019, 221(23):2760-2768.

GUAN, ZhengYi. (2011) **A Reliability Evaluation of Wireless Sensor Network Simulator: Simulation vs. Testbed**. Dissertação de mestrado, UNITEC New Zealand.

GUPTA, V.; Chung, T. H.; Hassibi, B.; Murray, R. M. (2004) **On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage**, 16th IFAC World Congress.

- GUPTA, V. et al. (2003) **Stability analysis of stochastically varying formations of dynamic agents**, in Proc. of CDC,(Maui, Hawaii), 504–509.
- HALLDÓRSSON, M. M.; Tokuyama, T. (2008) **Minimizing interference of a wireless ad-hoc network in a plane**. Theoretical Computer Science, ISSN 03043975, 402(1): 29–42.
- HASAN, M. S. et al. (2008) **Overview of Wireless Networked Control Systems over Mobile Ad-hoc Network**. Proceedings of 14th International Conference on Automation & Computing, London – UK.
- HEIMLICH, O. et al. (2010) **NMLab: A Co-Simulation Framework for Matlab and NS-2**. Second International Conference on Advances in System Simulation.
- HESTER, T., et al. (2012) **Texplorer: real-time sample-efficient reinforcement learning for robots**. Machine Learning, 1–45.
- HINES, W. W. et al (2003) **Probability and Statistics in Engineering**. 4^a edição, Editora John Wiley & Sons, ISBN: 0-471-24087-7.
- I-SENSE GROUP (2016) **Robinspect – Press Release**, disponível em: <<http://i-sense.iccs.ntua.gr/news/press/item/962-robinspect-press-release>>, acessado em abril de 2016.
- INET *FRAMEWORK* (2016) **INET Framework**, disponível em <inet.omnetpp.org/>. Acessado em 02/2016.
- INGHAM, J. et al. (1999) **What is an Agent?** Relatório técnico #6/99, disponível em: <<http://www.cs.upc.edu/~bejar/aia/aia-web/ingham99what.pdf>>, acessado em julho de 2017.
- IROBOT (2016) **Roomba**, disponível em: <<http://www.irobot.com/For-the-Home/Vacuum-Cleaning/Roomba.aspx>>. Acessado em 04/2016.
- JAYAVEL, J. et al (2005) **A TDMA-based Smart Clustering Technique for VANETS**. Journal of Theoretical and Applied Information Technology, 65(2).
- JENSEN, J. C. et al. (2011) **A Model-Based Design Methodology for Cyber-Physical Systems**. Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International, (6):1666 - 1671.
- JIA, Xiao; Meng, M. (2013). **A Survey and Analysis of Task Allocation Algorithms in Multi-Robot Systems**. Proceeding of the IEEE International Conference on Robotics and Biomimetics (ROBIO), Shenzhen, China.
- JIMÉNEZ-GONZÁLEZ, A. et al. (2013) **Testbeds for ubiquitous robotics: A survey**. In: Robotics and Autonomous Systems, DOI: 10.1016/j.robot.2013.07.006, 61:1487-1501.
- JUNGES, R.; Klugl, F. (2013) **Learning Tools for Agent-Based Modeling and Simulation**. KI - Künstliche Intelligenz, DOI: 10.1007/s13218-013-0258-z, 27(3):273-280.
- KADRY, S. e Hami, A. (2015) **E-Systems for the 21st Century: Concept Developments and Applications**, Vol. 2 – Chapter 10, Apple Academic Press.
- KAHAR, Saliyah. et al. (2012) **A Review of Wireless Technology Usage for Mobile Robot Controller**. International Conference on System Engineering and Modeling (ICSEM 2012), IPCSIT, 34, Singapura.
- KHAN, A. U. R. et al. (2013) **A Performance Comparison of Network Simulators for Wireless Networks**. In *IEEE International Conference on Control System, Computing and Engineering*, DOI: 10.1109/ICCSC.2012.6487111.

- KHAN, Z. et al. (2014) **Navigation Function Based Decentralized Control of A Multi-Agent System with Network Connectivity Constraints**. ArXiv:1402.5951v1, Cornell university.
- KLAVINS, E. (2003) **A formal model of a multi-robot control and communication task**. In: 42nd IEEE Conference Control on Decision, disponível em: <klavinslab.org/publications.html>, acessado em abril de 2016.
- KLUGL, F. et al. (2008) **A Validation Methodology for Agent-Based Simulations**. Conference: Proceedings of the 2008 ACM Symposium on Applied Computing (SAC), DOI: 10.1145/1363686.1363696.
- KLUGL, F. et al. (2012) **Agent Based Modeling and Simulation**. Association for the Advancement of Artificial Intelligence, ISSN 0738-4602.
- KOPKE, A. et al. (2008) **Simulating Wireless and Mobile Networks in OMNET++: The MiXiM Vision**. In: OMNET++ Workshop'08, Marseille, França. Disponível em: <<http://www.tevp.net/papers/mixim.pdf>>, acessado em maio de 2016.
- KRANAKIS, E.; Krizanc, D.; Raisbaum, S. (2005) **Mobile Agent Rendez-vous: A Survey**. Disponível em: <http://people.scs.carleton.ca/~kranakis/Papers/rendez-vous.pdf>
- KRISTIANSEN, S. et al. (2015) **A Methodology to Model Execution of Communication Software for Accurate Network Simulation**. ACM Trans. Model. Comput. Simul., DOI: 10.1145/2746233, 26(1).
- KUBERA, Y. et al. (2011) **IODA: an interactionoriented approach for multi-agent based simulations**. Int. Journal on Autonomous Agents and Multi-agent Systems 23(3):303–343.
- KUDELSKI, L. M. et al. (2013) **RoboNetSim: An integrated framework for multi-robot and network simulation**. Robotics and Autonomous Systems 61(2013):483–496, Elsevier.
- LAW, A. M. et al. (1991) **Simulation Modeling & Analysis**. McGraw-Hill, 2a Ed., 759 páginas.
- LAW, A. M. (2008) **How to build Valid and Credible Simulation Models**. Proceedings of the 2008 Winter Simulation Conference, 39–47, DOI: 10.1109/WSC.2008.4736054.
- LAW, A. M. et al. (2015) **Simulation Modeling and Analysis**. McGraw-Hill, 5a edição, 804 páginas.
- LAWTON, J. R. et al. (2003) **A decentralized approach to formation maneuvers**, IEEE Transactions on Robotics and Automation, 19:933–941.
- LEI, C. et al. (2011) **Impact of Packet Loss on CACC String Stability Performance**. 11th International Conference on ITS Telecommunications, DOI: 10.1109/ITST.2011.6060086, 381–386.
- LEONE, P. et al. (2013) **Self-Stabilizing TDMA Algorithms for Dynamic Wireless Ad Hoc Networks**. International Journal of Distributed Sensor Networks, (2013).
- LI, Wenfeng. et al. (2011) **Swarm behavior control of mobile multi-robots with wireless sensor networks**. Journal of Network and Computer Applications 34(2011):1398-1407.
- LI, Weilin. et al. (2014) **Co-simulation platforms for co-design of networked control systems: an overview**. Control Engineering Practice 23(2014) :44-56.
- LIM, T. H. (2012) **Validation of Performance Data using Experimental Verification Process in Wireless Sensor Network**. 17th Conference on Emerging Technologies & Factory Automation (ETFA).
- LIMA, P.; Custodio, L.; Ribeiro, M. I.; Santos-Victor, J.; (1999) **The RESCUE Project:**

- Cooperative Navigation for Rescue Robots.** Instituto de Sistemas e Robótica, Instituto Superior Técnico, Lisboa – Portugal.
- LIN, J., A. S. Morse.; B. D. O. Anderson. (2003) **The multi-agent rendez-vous problem.** In IEEE Conference on Decision and Control.
- LIU, Zhe; et al. (2016) **Formation Control of Mobile Robots Using Distributed Controller With Sampled-Data and Communication Delays.** In: IEEE Transactions on Control Systems Technology, DOI: 10.1109/TCST.2016.2518618.
- LORA, M. et al. (2015) **Simulation alternatives for the verification of networked cyber-physical systems.** Microprocessors and Microsystems 39(2015):843-853.
- LYNCH, N. A. (1996) **Distributed Algorithms.** Morgan Kaufmann, ISBN 9780080504704.
- MACAL, C. et al. (2014) **Introductory Tutorial: Agent-Based Modeling and Simulation.** In: Proceedings of the 2014 Winter Simulation Conference, IEEE press, 6-20.
- MANFREDI, Sabato. (2013) **Consensus over Multi-hop Networked Systems subject to Heterogeneous Time Delays.** XXIV International Conference on Information, Communication and Automation Technologies (ICAT), Sarajevo, Bosnia and Herzegovina.
- MARTÍNEZ, S. (2007) **Practical rendez-vous through modified circumcenter algorithms.** In: Proceedings of the 46th IEEE Conference on Decision and Control, New Orleans, LA, USA.
- MATHWORKS (2016) **MATLAB.** Disponível em: <<http://www.mathworks.com/products/matlab/>>, acesso em: abril de 2016.
- MEHTA, V. K.; Arrichiello, F. (2013) **Connectivity maintenance by robotic Mobile Ad-hoc NETWORK.** ArXiv:1312.2526v1, Cornell University.
- MI, Zhenqiang (2010) **Connectivity Control of Mobile Multi-Robot Networks.** IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Montreal, Canada.
- MI, Zhenqiang; Yang, Y. Ding, H. (2012) **Fault-tolerant Dispersion of Mobile Agents based on Mobile Ad hoc Networks.** In: 2012 International Conference on Wireless Communication & Signal Processing, DOI: 10.1109/WCSP.2012.6542900.
- MI, Z.; Yang, Y.; Sun, J. (2013) **Holistic Connectivity Controller for Multirobot Dispersion,** Mathematical Problems in Engineering, (2013).
- MI, Zhenqiang. et al. (2015) **Restoring Connectivity of Mobile Robotic Sensor Networks While Avoiding Obstacles.** IEEE Sensors Journal, 15(8).
- MIR (2016) **MiR100,** disponível em: <<http://mobile-industrialrobots.com/en/mir100-2/>>, acessado em abril de 2016.
- MIXIM (2013) **Mixed Simulator,** disponível em: <<http://mixim.sourceforge.net/>>, acessado em maio de 2016.
- MOHSENI, F. et al. (2016) **Distributed Receding Horizon Coverage Control for Multiple Mobile Robots.** In: IEEE Systems Journal, Vol. 10(1), DOI: 10.1109/JSYST.2014.2325219.
- MONGA, M. et al. (2009) **On the impact of Localization Data in Wireless Sensor Networks with Malicious Nodes.** ACM SPRINGL'09, Seattle, USA.
- MONTGOMERY, D. C. (1991) **Design and Analysis of Experiments.** 3ª Edição, Capítulos 1, 2, 7-13; John Wiley & Sons, Canadá.
- MORRIS, K.M.; Mullins, B.E.; Pack, D.J.; York, G.W.P.; Baldwin, R.O. (2006) **Impact of**

- Limited Communications on a Cooperative Search Algorithm for Multiple UAVs**, in Networking, Sensing and Control, 2006. ICNSC '06. Proceedings of the 2006 IEEE International Conference, 572-577.
- MOSTEO, A. R. et al. (2008) **Multi-robot Routing under Limited Communication Range**. IEEE International Conference on Robotics and Automation.
- NAGEL, R. et al. (2008) **Efficient and Realistic Mobility and Channel Modeling for VANET Scenarios using OMNET++ and INET-Framework**. Institute of Communication Networks, ACM 978-963-9799-20-2.
- NASA (2015) **Curiosity Overview**, disponível em: <https://www.nasa.gov/mission_pages/msl/overview/index.html>, acessado em abril de 2016.
- NEVES, T. F. et al. (2014) **Topology Control in Cooperative Ad Hoc Wireless Networks**. Electronic Notes in Theoretical Computer Science 302(2014):29-51, Elsevier.
- NS2 (2016) **Network Simulator 2**. Disponível em: <http://nslam.sourceforge.net/wiki/index.php/Main_Page>, acesso em : abril de 2016.
- NS3 (2016) **Network Simulator 3**. Disponível em: <<https://www.nslam.org/>>, acesso em: abril de 2016.
- OKADA, T. et al. (2008) **Large-scale Simulation Method of Mobile Robots**, Second International Symposium on Universal Communication, DOI 10.1109/ISUC .2008.42.
- OLFATI-SABER, R. e MURRAY, R. M.. (2003) **Consensus protocols for Networks of Dynamic Agents**. American Control Conference, 2003.
- OLFATI-SABER, R. e MURRAY, R. M.. (2004) **Consensus problems in networks of agents with switching topology and time-delays**. IEEE Transactions on Automatic Control, 49:1520 – 1533, 2004.
- OLFATI-SABER, R. (2006) **Flocking for multi-agent dynamic systems: algorithms and theory**. IEEE Transactions on Automatic Control, 51:401 – 420.
- OLIVEIRA, L.; Almeida, L.; Santos, F. (2011) **A loose synchronisation protocol for managing RF ranging in mobile ad-hoc networks**. In RoboCup Symposium, Istanbul, Turkey.
- OLIVEIRA, Luis; Almeida, Luis; Santos, Frederico. (2012) **A Loose Synchronisation Protocol for Managing RF Ranging in Mobile Ad-Hoc Networks**, RoboCup 2011: Robot Soccer World Cup XV, Lecture Notes in Computer Science, 7416:574-585.
- OLIVIERA, L.; Almeida, L.; Lima, P. (2015) **Multi-hop routing within TDMA slots for teams of cooperating robots**. In: 2015 IEEE World Conference on Factory Communication Systems (WFCS), DOI: 10.1109/WFCS.2015.7160566, 1-8.
- OMNET. (2017) **OMNET and INET Framework**, disponível em: <<http://www.omnetpp.org>>.
- ORDOÑEZ, B. (2013) **Estratégia de controle cooperativo baseado em consenso para um grupo multi-veículos**. Tese de doutorado, UFSC, PGEAS.
- ORDOÑEZ, B. et al. (2014) **Generation of Trajectories Using Predictive Control for Tracking Consensus with Sensing and Connectivity Constraint**. Studies in Computational Intelligence. 1ed.: Springer Berlin Heidelberg, 2014, 19-37.
- OSA (2015) **Open Simulation Architecture**. Disponível em: <<http://osa.inria.fr/current/>>, acessado em maio de 2016.
- PARKER, L. E. (1998) **Alliance: An architecture for fault tolerant multi robot cooperation**.

IEEE Transactions on Robotics and Automation, 14:220 – 240.

PEKKARINEN, E. (2014) **Wireless Sensor Network Simulation With OMNET++ (2016) OMNeT++ Discrete Event Simulator**. Disponível em: <<https://omnetpp.org/>>, acesso em : abril de 2016.

PETROBOT (2016) **Robotic Inspection Technology**, disponível em: <<http://petrobotproject.eu/robotic-inspection-technology/>>, acessado em abril de 2016.

PICKEM, Daniel; et al. (2015) **The GRITSBot in its Natural Habitat - A Multi-Robot Testbed**. IEEE Conference on Robotics and Automation, disponível em: <<http://gritlab.gatech.edu/home/2015/11/robotarium/>>.

PICKEM, Daniel; et al. (2016) **Safe, Remote-Access Swarm Robotics Research on the Robotarium**, Cornell University Library, disponível em: <<https://arxiv.org/abs/1604.00640>>.

PIMENTEL, Bruno. et al. (2002) **NEMO – NETworked Mobile rOBots**, disponível em: <<http://homepages.dcc.ufmg.br/~brunosp/nemo/>>, acessado em abril de 2016.

PIMENTEL, Bruno. et al. (2003a) **Cooperative Communication in Ad Hoc Networked Mobile Robots**. Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems, Las Vegas, Nevada.

PIMENTEL, Bruno. et al. (2003b) **Communication Issues in Networked Exploration Robots**. VI Simpósio Brasileiro de Automação Inteligente.

PINCIROLI, C. et al. (2012) **ARGoS: a Modular, Parallel, Multi-Engine Simulator for Multi-Robot Systems**. Swarm Intelligence, volume 6(4), pp. 271-295. Springer, Germany. Disponível em: <<http://www.argos-sim.info/>>, acesso em: abril de 2016.

POHJOLA, M. et al. (2008) **Wireless control of mobile robot squad with link failure**, in: Proceedings of the 6th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 648–656.

POHJOLA, M.; Nethi, S.; Jantti, R. (2009) **Wireless control of a multihop mobile robot squad**, in Wireless Communications, IEEE , 16:1:14-20.

POONAWALA, H. A. et al. (2015) **Collision-Free Formation Control with Decentralized Connectivity Preservation for Nonholonomic-Wheeled Mobile Robots**. In: IEEE Transactions on Control of Network Systems, 2(2):122-130, DOI: 10.1109/TCNS.2014.2378876.

PUJERI, U. R. et al. (2014) **Survey of Various Open Source Network Simulators**, V. International Journal of Science and Research (IJSR), ISSN: 2319-7064.

QIN, L. et al. (2014) **A survey of fault diagnosis for swarm systems**. Systems Science & Control Engineering: An Open Access Journal, Department of Automation, TNLList, Tsinghua University, Beijing – China.

QUADPROG++ (2015) Quadprog++ Project. Disponível em: <<http://quadprog.sourceforge.net/>>, acessado em junho de 2015.

RAC - **Robótica Autônoma Cooperativa: Modelos, Algoritmos e Aplicações**. (2009) Disponível em: <http://docplayer.com.br/15282348-Rac-robotica-autonoma-cooperativa-modelos-algoritmos-e-aplicacoes.html>, acessado em abril de 2016.

RAJ, Cristhu M. R.; Chacko, N. M. (2013) **A Comprehensive Overview on Different Network Simulators**. International Journal of Engineering and Technology (IJET), ISSN: 0975-4024, 5(1).

- RAJESH, M. et al. (2015) **Energy Efficient Deployment of Wireless Sensor Network by Multiple Mobile Robots**. In: 2015 Intl. Conference on Computing Network Communications (CoCoNet'15), DOI: 10.1109/CoCoNet.2015.7411169, 72 - 78.
- RAILSBACK, S.F. et al. (2012) **Agent-based and Individualbased Modeling - A Practical Introduction**. Princeton University Press.
- RAMACHANDRAN, R. K. et al. (2017) **The Effect of Communication Topology on Scalar Field Estimation by Large Networks with Partially Accessible Measurements**. 2017 American Control Conference, AACC, 978-1-5090-5992-8.
- RAMOS, Daniel C.; Moreno, U. F. (2014) **OMNET++ como Ferramenta de Simulação e Análise de Protocolos de Comunicação em Estratégias de Controle Cooperativo**. In: *1º Simpósio de Tecnologia e Ciência - UFU*, resumo expandido, ISSN: 2359-3415, 1: 43-46, Patos de Minas – MG, Brasil.
- RAMOS, Daniel C.; Almeida, L.; Moreno, U. F. (2015a) **Cooperative Control Strategy Based on Consensus with MANET Protocols and Topology Control**. 1th Doctoral Congress in Engineering, resumo e poster, Universidade do Porto.
- RAMOS, Daniel C.; Almeida, L.; Oliveira, L.; Moreno, U. F. (2015b) **Network Interference on Cooperative Mobile Robots Consensus**. In: *ROBOT'2015 - Second Iberian Robotics Conference*, artigo completo, novembro, Lisboa, Portugal.
- RAMOS, Daniel C.; Almeida, L.; Oliveira, L.; Moreno, U. F. (2015c) **Impact of Alien Networks on Consensus in a Team of Cooperative Mobile Robots**. In: *V Brazilian Symposium on Computing Systems Engineering*, artigo completo, Foz do Iguaçu – PR, Brasil.
- RAMOS, Daniel, et al. (2016) **Method for Design and Performance Evaluation of Ad Hoc Networked Mobile Robotic Systems using OMNET++**. In: IFAC-Papers Online, v. 49, DOI: 10.1016/j.ifacol.2016.11.144.
- RAMOS, Daniel C. et al. (2017) **IRoNS Method**, disponível em: <www.ironsmethod.blogspot.com>, 2017.
- RANDAL, W. R., Ella, B. e Atkins, M. (2005) **A Survey of Consensus Problems in Multi-agent Coordination**, American Control Conference, Portland-USA.
- RAUNAK, M. et al. (2013) **A framework for simulation validation coverage**. Proceedings of the 2013 Winter Simulation Conference, DOI: 10.1109/WSC.2013.6721540.
- RAUNAK, M. et al. (2014) **Quantifying validation of discrete event simulation models**. Proceedings of the 2014 Winter Simulation Conference, pp. 628 - 639, DOI: 10.1109/WSC.2014.7019927.
- RAY, Partha P. (2017) **Internet of Robotic Things: Concept, Technologies, and Challenges**. IEEE Access (4) 2017, DOI: 10.1109/ACCESS.2017.2647747.
- REYES, L. A. V. et al. (2015) **Flocking, Formation Control, and Path Following for a Group of Mobile Robots**. In: IEEE Transactions on Control Systems Technology, DOI: 10.1109/TCST.2014.2363132, 23(4)
- RIBAUT, J. et al. (2010) **OSIF: A Framework to Instrument, Validate, and Analyze Simulations**. SimTools, DOI: 10.4108/ICST.SIMUTOOLS2010.8729.
- RICHIARDI, M. et al. (2006) **A common protocol for agent-based social simulation**. Artificial Societies and Social Simulation 9(1):15.
- RIVERBED (2016) **OPNET is now part of Riverbed**. Disponível em:

<http://br.riverbed.com/products/performance-management-control/opn_et.html>, acesso em: abril de 2016.

ROBINSON, Stewart. (2006) **Conceptual Modeling for Simulation: Issues and Research Requirements**. Proceedings of the 2006 Winter Simulation Conference, pp. 792 - 800, DOI: 10.1109/WSC.2006.323160.

ROBINSON, S. (2014) **Simulation: The Practice of Model Development and Use**, 2ª edição, ISBN 9781137328021, Palgrave Macmillan.

ROBOTLAND (2011) **Firefighting Robots in Japan**, disponível em: <http://robotland.blogspot.com.br/2011/03/firefighting-robots-in-japan.html>, acessado em abril de 2016.

ROCHA, R. (2002) **Fostering Cooperation among Intelligent Machines**, Technical Report, ISR – Universidade de Coimbra, Portugal, disponível em: <http://ap.isr.uc.pt/archive/0209_TR1_LaTeX_fostering_coop_among_intel_mach_st_of_art.pdf>, acessado em abril de 2016.

ROCHA, R. (2006a) **Building Volumetric Maps with Cooperative Mobile Robots and Useful Information Sharing: a Distributed Control Approach based on Entropy**, Ph.D. thesis, Faculty of Engineering of University of Porto, Portugal, 2006.

ROCHA, R. (2006b) **Efficient Information Sharing and Coordination in Cooperative Multi-Robot Systems**. In Proc. of II European-Latin-American Workshop on Engineering Systems (SELASI'2006), Porto, Portugal.

ROY, S. et al. (2004) **Formation and alignment of distributed sensing agents with double-integrator dynamics and actuator saturation**. Sensor Network Applications, IEEE Press.

RUNESON, P. et al. (2008) Guidelines for conducting and reporting case study research in software engineering, Empirical Software Engineering, vol. 14, no. 2, pp. 131-164.

RUSSEL, S. e NORVIG, P. (2010) **Artificial Intelligence: A Modern Approach**. Prentice Hall, 3a Edição, ISBN-13: 978-0-13-604259-4.

SABATTINI, et al. (2015) **Decentralized Estimation and Control for Preserving the Strong Connectivity of Directed Graphs**. IEEE Transactions on Cybernetics, 45(10).

SABATTINI, L. et al. (2016) **Decentralized connectivity maintenance for cooperative control of mobile robotic systems**. The International Journal of Robotics Research, 32(12):1411-1423.

SCHLUSE, M. e Rossmann, J. (2016) **From Simulation to Experimentable Digital Twins**, in IEEE International Symposium on Systems Engineering (ISSE), ISBN: 978-1-5090-0793-6.

SAMPAIO, R.C. et al. (2007) **Estudos de Revisão Sistemática: Um Guia para Síntese Críteriosa da Evidência Científica**. Rev. Brasileira de Fisioterapia, ISSN 1413-3555, São Carlos - SP.

SANFELIU, A. et al. (2008) **Network Robot Systems**. Robotic and Autonomous Systems, 56 (2008), p. 793-797, DOI: 10.1016/j.robot.2008.06.007.

SARGENT, R. G. (2013) **Verification and Validation of simulation Models**. Journal of Simulation (2013) 7:12-24.

SARGENT, R. G. (2014) **Verifying and validating simulation models**. In: Proceedings of the Winter Simulation Conference 2014, IEEE, DOI: 10.1109/WSC.2014.7019883.

SARGENT, R. G. (2015a) **An introductory tutorial on verification and validation of**

- simulation models.** In: 2015 Winter Simulation Conference (WSC), IEEE, DOI: 10.1109/WSC.2015.7408291.
- SARGENT, R. G. (2015b) **Use of the interval statistical procedure for simulation model validation.** In: 2015 Winter Simulation Conference (WSC), IEEE, DOI:10.1109/WSC.2015.7408153.
- SAVKIN, A. V. et al. (2016) **Distributed formation building algorithms for groups of wheeled mobile robots.** *Robotics and Autonomous Systems*, 75(2016) pp 463-474, Elsevier.
- SCATTOLINI, R. **Architectures for distributed and hierarchical model predictive control – a review.** *Journal of Process Control*, 19(5):723–731.
- SENTHILKUMAR, K. S. et al. (2012) **Multi-robot exploration and terrain coverage in an unknown environment.** In: *Robotics and Autonomous Systems*, Elsevier, DOI: 10.1016/j.robot.2011.09.005, 60(1):123-132
- SHINNOH, A. et al. (2015) **Communication Packet Loss Concealment for Pattern Generation with Robotic Swarms.** In: 2015 IEEE International Conference on Advanced Intelligent Mechatronics, DOI: 10.1109/AIM.2015.7222590, 537-542.
- SIEGFRIED, R. et al. (2009) **A reference model for agent-based modeling and simulation.** In: *Proceedings of the 2009 Spring Simulation Multiconference*, DOI: 10.1145/1639809.1639833.
- SIEGFRIED, R. (2014) **Modeling and Simulation of Complex Systems.** Springer Vieweg, ISBN 978-3-658-07528-6, Cap. 2, 3, 5, 8, 9.
- SMITH, P. F. et al. (2007) **Best Practices for Establishing a Model-Based Design Culture.** Mathworks, disponível em: <https://www.mathworks.com/tagteam/40538_SAE-2007-01-0777-Best-Practices-for-MBD-Culture.pdf>, acessado em maio de 2016.
- SOKOŁOWSKI, J. A.; BANKS, C. M. (2010) **Modeling and Simulation Fundamentals: Theoretical Underpinnings and Practical Domains.** Editora John Wiley & Sons, ISBN 978-0-470-48674-0.
- SOMARAKIS, C. et al. (2015) **Non-linear Flocking Networks with Collision Avoidance.** 34th Chinese Control Conference (CCC), 7633-7638.
- SOUISSI, S. et al. (2011) **Distributed Algorithms for Cooperative Mobile Robots: A survey.** *Second International Conference on Network and Computing*.
- STEVENSON, W. J. (2001) **Estatística Aplicada à Administração.** Editora Habra, ISBN: 8529400925.
- STRATULAT, B. et al. (2010) **Wireless Synchronization Protocols for Collaborative Robotic and Sensor Environments.** 19th International Workshop on Robotics in Alpe-Adria-Danube Region – RAAD, Budapest, Hungary.
- TAN, K. et al. (2010) **Comparing simulation tools and experimental testbeds for wireless mesh networks.** In: 2010 IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks, DOI:10.1109/WOWMOM.2010.5534917, 1-9.
- TANENBAUM, A. S. et al. (2011) **Computer Networks.** 5a Edição, Editora Pearson, ISBN-13: 978-0-13-212695-3.
- TANNER, H. G. et al. (2003) **Stable flocking of mobile agents, part i: Fixed topology,** in *Proc. of CDC*, (Maui, Hawaii), 2010–2015.

- TCHOUANKEM, H. et al. (2015) **Impact of Buildings on Vehicle-to-Vehicle Communication at Urban Intersections**. 12th Annual IEEE Consumer Communications and Networking Conference (CCNC).
- TIWARI, A.; Jun, M.; Jeffcoat, D. E.; Murray, R. M. (2005) **The dynamic sensor coverage problem**. In IFAC World Congress, Prague, Czech Republic.
- TONNEAU, A. S. et al. (2015) **How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities**. *Ad Hoc Networks* 30 (2015):115-127, Elsevier.
- VARGAS, A.; Hornig, R. (2008) **An overview of the OMNeT++ simulation environment**, Simutools '08: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops: 1--10.
- VAUGHAN, R. et al. (2011) **Stage**. Disponível em: <<http://rtv.github.io/Stage/>>, acessado em abril de 2016.
- VECCHIO, D. D. (2006a) **Discrete dynamic feedback for a class of hybrid systems on a lattice**. In IEEE International Symposium on Computer-Aided Control Systems Design, Munich, Germany.
- VECCHIO, D. D.; Murray, R. M.; Klavins, E. (2006b) **Discrete state estimators for a systems on a lattice**. *Automatica*, 42.
- VUNDURTHEY, B. et al. (2016) **Rendez-vous of Heterogeneous Robots amidst Obstacles with Limited Communication**. In: 2016 Indian Control Conference, DOI: 10.1109/INDIANCC.2016.7441158, 347 – 353.
- WADUD, Abdullah-Al (2013) **A TDMA Scheme for Mobile Sensor Networks**. *International Journal of Distributed Sensor Networks*, (2013), ID 907583.
- WAGENPFEIL, J.; Trachte, A.; Hatanaka, T.; Fujita, M.; Sawodny, O. (2009) **A distributed minimum restrictive connectivity maintenance algorithm**. 9th International Symposium on Robot Control, 9(1): 499–504.
- WANG, Z. et al. (2003) **Ad-hoc robot wireless communication**. In: IEEE International Conference on Systems, Man and Cybernetics, 4:4045 - 4050.
- WEI, C.; Hindriks, K. Jonker, C. M. (2014) **The Role of Communication in Coordination Protocols for Cooperative Robot Teams**, In Proceedings of the 6th International Conference on Agents and Artificial Intelligence, DOI: 10.5220/0004758700280039, 28-39.
- XIAN, X.; Shi, W.; Huang, H. (2008) **Comparison of OMNeT++ and other simulator for WSN simulation**. IEEE Conference on Industrial Electronics and Applications (ICIEA'08), 1439 – 1443.
- YAMASHITA, M. et al. (1999) **Distributed memoryless point convergence algorithm for mobile robots with limited visibility**. In: IEEE Transactions on Robotics and Automation, vol. 15(5):818–828.
- YE, W. et al. (2001) **Evaluating control strategies for wireless-networked robots using an integrated robot and network simulation**. In: Proc. of the IEEE International Conference on Robotics and Automation (ICRA), 2941–2947.
- YOSHIDA, K. et al. (2014) **Control of a Group of Mobile Robots Based on Formation Abstraction and Decentralized Locational Optimization**. In: IEEE Transactions on Robotics, DOI: 10.1109/TRO.2013.2293836, 30(3),

- YU, J. et al. (2015) **Target Assignment in Robotic Networks: Distance Optimality Guarantees and Hierarchical Strategies**. In: IEEE Transactions on Automatic Control, 60(2):327-341.
- ZAHUGI, E.M.H.; Prasad, S.V.; Prasad, T.V. (2012) **Advanced Communication Protocols for Swarm Robotics: A Survey**. International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622, 2(5):119-123.
- ZAVLANOS, M. M.; Tanner, H. G.; Jadbabaie, A.; Pappas, G. J. (2009) **Hybrid Control for Connectivity Preserving Flocking**. IEEE Transactions on Automatic Control, 54(12):2869–2875.
- ZAVLANOS, M. M. et al. (2010) **Mobility & Routing Control in Networks of Robots**. 49th IEEE Conference on Decision and Control.
- ZHANG, D.; Xie, G.; Wang, L. (2007) **Adaptive task assignment for multiple mobile robots via swarm intelligence approach**, Robots and Autonomous Systems, 55:572–588.
- ZHANG, Y. et al. (2013) **A survey on multiple unmanned vehicles formation control and coordination: Normal and fault situations**. In: 2013 International Conference on Unmanned Aircraft Systems (ICUAS), DOI: 10.1109/ICUAS.2013.6564798, 1087-1096.
- ZHANG, Z. et al. (2010) **COSMO: CO-Simulation with MATLAB and OMNeT++ for Indoor Wireless Networks**. In: Proc. Of the IEEE Globecom 2010, IEEE Communications Society.
- ZOGHBY, N. et al. (2014) **Robot Cooperation and Swarm**, Wireless Sensor and Robot Networks: From Topology Control to Communication Aspects'14, capítulo 8.

ANEXO A – REVISÃO SISTEMATIZADA

A revisão bibliográfica foi realizada por meio de um estudo sistemático, ou seja, uma revisão com objetivo de ser metódica, explícita e passível de reprodução. Foram utilizados os passos descritos no método de Sampaio (2007), os quais serão descritos aqui em termos da pesquisa realizada. A revisão é dividida em introdução, metodologia, resultados e conclusão.

a) Introdução

O principal questionamento deste trabalho é o de como estudar o impacto que uma rede de comunicação possui em robôs cooperativos. Por ser uma área multidisciplinar e complexa, os problemas da robótica cooperativa acabam sofrendo simplificações em outros aspectos, principalmente relacionado a rede de comunicação. Com isto em mente, não foi idealizado apenas o estudo da rede e robótica, mas sim em como o simular, estruturar, validar e testar. Este raciocínio leva a outros questionamentos:

- É possível adaptar métodos e técnicas existentes para outros tipos de simulação?
- Existem trabalhos focados em métodos específicos para uma classe de problemas de robótica, controle ou rede?
- Quais parâmetros e métodos de avaliação utilizados por trabalhos nas áreas de controle, robótica e redes de comunicação? É possível integrar ou relacionar estes parâmetros/avaliações?
- Como realizar experimentos sem possuir uma *testbed*?

É importante mencionar que a revisão sobre os algoritmos de cooperação e a revisão de técnicas de redes, foram realizadas anteriormente sem utilizar um método sistemático. Ambas foram consideradas suficientes, dado ao fato que as técnicas em si não são o foco deste trabalho, e sim o método para implementação, verificação, validação e testes de simulações. As técnicas são utilizadas como ferramentas para exemplificar a implementação do método, podendo ser escolhidas de acordo com uma necessidade ou situação em especial.

b) Metodologia

As bases de dados utilizadas para pesquisa foram o Google Scholar, Science Direct e o IEEE Xplore, sendo a primeira a principal fonte de pesquisa e as outras utilizadas como complementos.

A busca associada aos questionamentos foi limitada a um foco inicial, alinhada com os equipamentos disponibilizados na UFSC e com o foco da pesquisa do programa de pós-graduação da UFSC. Portanto, limitou-se a pesquisa à robótica móvel terrestre, redes de comunicação sem fio, robôs cooperando de forma descentralizada e manteve-se o foco em simulações.

Os critérios de pesquisa foram adicionados conforme o número de artigos relacionados foram encontrados, com objetivo de reduzir o total de artigos. Este processo foi necessário dado que o assunto possui termos comuns e que envolvem diversas áreas, dificultando localizar artigos relevantes ao trabalho.

São utilizados três operadores na pesquisa do Google Scholar, '+' para indicar que os resultados devem conter o termo associado, '-' para remover qualquer artigo

que contenha o termo associado e aspas (“ ”) para indicar que o termo associado deve ser idêntico ao descrito no interior das aspas, incluindo espaços e símbolos.

As principais buscas são listadas nas tabelas no item c, onde a cada linha subsequente é adicionado os termos listados. É indicado o número de artigos encontrados à direita e utiliza-se a letra *k* para indicação de milhar.

As ramificações finais de cada busca são indicadas em cores diferentes e utilizam todos os termos anteriores aos mesmos.

Ao final de cada ramificação é utilizado um protocolo de inclusão/exclusão, onde primeiramente foram lidos os títulos e resumos, selecionando os artigos relevantes à área da busca. Ainda, foram incluídos os artigos com assuntos relacionados diretamente ao trabalho, artigos que possuíam apresentação de resultados com critérios para seleção de parâmetros de simulação, artigos que possuíam critérios de avaliação e também de áreas semelhantes que poderiam fornecer *insights* sobre o trabalho.

c) Resultados

Os principais resultados das buscas no Google Scholar são apresentados na Tabela 4, Tabela 5, Tabela 6 e Tabela 7.

Tabela 4 - Terminologia e número de artigos encontrados na primeira busca utilizando o Google Scholar.

TERMOS	RESULTADOS
+ROBOT +CONTROL +NETWORK	640K
+COOPERATIVE	61K
-NEURAL	39K
REFINAR RESULTADOS: 2000 A 2016	20K
REFINAR RESULTADOS: 2006 A 2016	18,5K
-FUZZY	17K
+MOBILE	16,8K
+WIRELESS	13K
-UNDERWATER	11,5K
-AIR -SLAM	7K
-3G	6,8K
-AERIAL	4,1K
+COMMUNICATION	4,1K
+AD HOC	1,9K
REFINAR: DESCONSIDERAR CITAÇÕES	1,7K
+SIMULATION	1,3K
RAMIFICAÇÃO 1	
+MANET	351
+PERFORMANCE	311
APÓS LEITURA DOS RESUMOS	9
RAMIFICAÇÃO 2	
+DECENTRALIZED	540
APÓS LEITURA DOS RESUMOS	12

Tabela 4 - Terminologia e número de artigos encontrados na primeira busca utilizando o Google Scholar. (conclusão)

TERMOS	RESULTADOS
RAMIFICAÇÃO 3	
+OMNET++	10
APÓS LEITURA DOS RESUMOS	2
RAMIFICAÇÃO 4	
+NS2	83
APÓS LEITURA DOS RESUMOS	1
TOTAL ENCONTRADO (TODAS RAMIFICAÇÕES)	24

Tabela 5 - Terminologia e número de artigos encontrados na segunda busca utilizando o Google Scholar.

TERMOS	RESULTADOS
+MOBILE +CONTROL +NETWORK + AD HOC +COOPERATIVE +WIRELESS + COMMUNICATION – 3G –UNDERWATER –AIR –SLAM – AERIAL +VANET	3,4k
+SIMULATION	3k
+OMNET++	45
APÓS LEITURA / TOTAL ENCONTRADO	21

Tabela 6 - Terminologia e número de artigos encontrados na terceira busca utilizando o Google Scholar.

TERMOS	RESULTADOS
+VERIFICATION +VALIDATION +TECHNIQUE +“NETWORKED CONTROL SYSTEM”	1,7k
REFINAR RESULTADOS: 2006 A 2016	
+ANALYSIS +FRAMEWORK	1,1k
+DEVELOPMENT	980
+EVALUATION	776
+PERFORMANCE +STRUCTURE	650
RAMIFICAÇÃO 1	
+NCS	241
APÓS LEITURA RESUMOS	21
RAMIFICAÇÃO 2	
+VANET	22
APÓS LEITURA RESUMOS	0
TOTAL ENCONTRADO	21

Tabela 7 - Terminologia e número de artigos encontrados quarta busca utilizando o Google Scholar.

TERMOS	RESULTADOS
+VERIFICATION +VALIDATION +NETWORKED +CONTROL +SYSTEM +NCS +ANALYSIS +FRAMEWORK +DEVELOPMENT +EVALUATION +STRUCTURE +CO- SIMULATION	11
REFINAR RESULTADOS: 2006 A 2016	
LEITURA RESUMOS	3
TOTAL ENCONTRADO	3

Também foram realizadas pesquisas com algumas variações de termos, mas não obtiveram resultados significativos e foram omitidas neste anexo por razão de não prolongar demasiadamente o assunto. No entanto, houve alguns artigos encontrados em outros mecanismos de busca (Science Direct e IEEE Explore), sendo indicados os termos e número de artigos encontrados, na Tabela 8.

Tabela 8 - Terminologia e número de artigos encontrados em outros mecanismos de busca

TERMOS	RESULTADOS
VERIFICATION VALIDATION NETWORKED CONTROL SYSTEM NCS ANALYSIS FRAMEWORK DEVELOPMENT EVALUATION STRUCTURE (SCIENCE DIRECT)	640
LEITURA DOS RESUMOS	2
VERIFICATION VALIDATION NETWORKED CONTROL SYSTEM NCS ANALYSIS FRAMEWORK DEVELOPMENT EVALUATION STRUCTURE (IEEE XPLORE)	1
LEITURA DOS RESUMOS	0
NETWORKED CONTROL SYSTEM NCS CO-SIMULATION (IEEE XPLORE)	1
LEITURA DOS RESUMOS	0
NETWORKED CONTROL SYSTEM NCS CO-SIMULATION (SCIENCE DIRECT)	17,4k
CO-DESIGN	13k
ANALYSIS	12k
FRAMEWORK	5,6k
COMMUNICATION	3,2k
WIRELESS	675
REFINAR RESULTADOS: 2006 - 2016	601
MOBILE	471
AD HOC	239
LEITURA DOS RESUMOS	8
TOTAL ENCONTRADO	10

Após a etapa inicial da busca, foram escolhidos os 63 artigos mencionados nas tabelas e mais 4 encontrados em buscas menores, totalizando uma base inicial de 67 artigos.

Em seguida, foram consultadas as referências utilizadas nestes artigos e pesquisados trabalhos relacionados dos mesmos autores, somando mais 42 artigos e totalizando 109 artigos. Destes 109, 22 foram eliminados após leitura completa do artigo, reduzindo a 87 artigos.

Os artigos foram separados em 13 categorias conforme seu assunto principal: (1) MANET e MSN, (2) Controle Descentralizado, (3) OMNeT++, (4) VANET, (5) Topologia de Rede, (6) Simulação, (7) Controle e Robótica, (8) Energia, (9) Simulação vs Testbed, (10) Robótica e Redes, (11) NCS, (12) VV&T e (13) Outros.

Dentre os artigos que já haviam sido pesquisados na revisão não sistemática, 19 foram reanalisados sob esta ótica e foram incluídos nas referências bibliográficas. Desta forma, o total de artigos selecionados na primeira revisão sistemática é de 106 documentos. Outras buscas complementares realizadas ao decorrer do doutorado são apresentadas na Tabela 9 .

Tabela 9 – Outras terminologias utilizadas, com ferramenta de busca utilizada, número de artigos encontrados e número de artigos utilizados em abril de 2016.

Termos	Busca	Total	Final
Mobile Robots 2016	IEEE	229	12
Wireless Mobile Robots 2015	IEEE	295	5
Mobile Robots <i>Rendez-vous</i> 2014-2016	IEEE	25	3
Mobile Robots Formation Control 2014-2016	IEEE	289	6

Tabela 10 – Outras terminologias utilizadas, com ferramenta de busca utilizada, número de artigos encontrados e número de artigos utilizados em 2017.

TERMOS	BUSCA	TOTAL	FINAL
"NETWORKED ROBOTICS" SCHOLAR	SCHOLAR	229	12
"NETWORKED ROBOTIC" 2007-2017	SCHOLAR	112	2
"NETWORKED ROBOTIC"	SCHOLAR	65	1
"NETWORKED ROBOTICS" 2007-2017	IEEE	32	1
"NETWORKED ROBOTIC" 2007-2017	IEEE	63	11
"NETWORKED ROBOT" 2007-2017	IEEE	59	6
"INTERNET OF ROBOTIC THINGS"	SCHOLAR	3	35

Ao longo do doutorado, a base de referência utilizando 106 artigos foi ampliada, incluindo livros, *sites* e novos artigos publicados em 2017 (Tabela 10). Ao final da tese, foram contabilizadas mais de 200 referências na tese. Ao final foi realizada uma pesquisa sobre a utilização do termo IoRT, sendo adicionadas algumas referências desta nova área (Tabela 11).

Tabela 11- Incremento do uso do termo IoRT nos últimos anos.

TERMOS	BUSCA	TOTAL
INTERNET OF ROBOTIC THINGS 2006-2017	IEEE	321
INTERNET OF ROBOTIC THINGS 2016-2017	IEEE	178
Robot "Internet Of Things" 2010-2017	SCHOLAR	17300
Robot "Internet Of Things" 2013-2017	SCHOLAR	16800
Robot "Internet Of Things" 2016-2017	SCHOLAR	5380
Robot "Internet Of Things"	IEEE	305
Robot "Internet Of Things"	IEEE	465

d) Conclusão

Neste anexo foram descritas as principais buscas realizadas em respeito ao tema do trabalho. A etapa de preparar o resumo crítico sobre as obras também foi realizada e adaptada para a introdução da tese e nos capítulos relacionados.

Durante as pesquisas foram encontrados termos que auxiliaram na busca de mais trabalhos relacionados, como por exemplo, os termos “NCS”, “WNCS” e “*co-simulation*”. Estes e outros termos, foram posteriormente utilizados nas pesquisas, aumentando significativamente o acervo de referências.

Os artigos apontaram para um campo científico a ser explorado, já que apesar de simulação de controle e redes ter recebido mais atenção nos últimos anos e de ter serem encontrados estudos aplicados a robótica, ao adicionarmos o problema de formulação de um método para realizar este tipo de simulação, fica evidente a carência de pesquisas na área.

ANEXO B – EXEMPLO DE MODELO COMUNICATIVO

ID	mod_comunic_03	Nome	Modelagem Comunicativa
Categoria	Planejamento SP	EC	03 – Avaliação da matriz A
Descrição	Detalhamento por meio do protocolo ODD [Adaptado de Grimm (2006, 2010)] do estudo abordado pelo método, descrevendo-o em 7 conceitos. A modelagem consiste de questionamentos e respostas, utilizando apenas linguagem textual.		
Intenção de Uso	Documentar o estudo de forma a facilitar a elaboração de artigos científicos e a replicação do mesmo por terceiros.		

1. PROPÓSITO

➤ Qual o propósito deste estudo?

Deseja-se estudar o impacto da matriz adjacência inicial na tarefa cooperativa. Primeiramente busca-se meios para definir uma matriz aleatória que satisfaça a condição de que todos os robôs estejam conectados direta ou indiretamente e avaliar o impacto de diferentes condições iniciais. Deseja-se realizar simulações utilizando uma rede de comunicação ideal no Matlab, ideal no OMNeT++ e real sem perdas no OMNeT++.

2. ENTIDADES, VARIÁVEIS DE ESTADO E ESCALA

➤ Quais os tipos de entidades estão no estudo? Por quais variáveis de estado ou atributos, são caracterizadas estas entidades? Qual é o incremento de tempo e de espaço, e extensão do estudo?

São utilizados 10 robôs idênticos, um meio de comunicação e um ambiente para locomoção dos robôs.

Os robôs possuem um controle cooperativo, sistema de sensoriamento, controle de velocidade individual, um controle de topologia e um sistema de comunicação.

O sensoriamento é considerado ideal, ou seja, o robô sabe com precisão onde está localizado. Sensores adicionais não são necessários para este estudo.

O sistema de comunicação utiliza padrão IEEE 802.11 (Wi-Fi) para enviar informação sobre sua posição (Coordenadas X,Y) para os robôs que estão no alcance de sua comunicação (250cm) e permitidos pelo controle de topologia.

O controle de topologia consiste de uma matriz adjacência lógica de n linhas por n colunas, sendo n o número de robôs e cujo elementos consistem de 0 e 1 (1 indica presença de comunicação entre os robôs). A matriz é baseada inicialmente no alcance de comunicação física, contudo, a topologia lógica pode realizar ajustes como, por exemplo, o de ignorar certos robôs que estão no alcance de comunicação. Os algoritmos utilizados para controle de topologia são dois: fixo e dinâmico. O fixo mantém a mesma estrutura de comunicação da condição inicial, ou seja, outros robôs que entrem no alcance de comunicação, são ignorados. O dinâmico opera idêntico a topologia de comunicação física.

O controle cooperativo utiliza a informação disponível sobre a posição do robô e dos robôs vizinhos e um algoritmo de consenso com otimização e predição, para definir o ponto de encontro entre os robôs (*rendez-vous* médio). Este controle é descentralizado e para cada robô será diferente, dependendo da informação disponível de seus vizinhos diretos.

O controle de velocidade é praticamente ideal, dado que o resultado do controle cooperativo é a velocidade que o robô deve ter para alcançar o ponto desejado (velocidade em x e velocidade em y). No entanto, o robô possui uma limitação de velocidade máxima (10cm/s) e irá se deslocar em direção a mesma a cada interação.

O deslocamento dos robôs é ideal, assim como o meio onde os mesmos se deslocam. Não há obstáculos e nem colisões. O deslocamento dos robôs é regido por uma equação simples, onde a distância percorrida é dada em função da velocidade de referência (do controle de velocidade) e o tempo desde a última atualização (distância em x é independente da distância em y). A atualização do deslocamento é de aproximadamente 1/10 da interação do controle de velocidade.

O ambiente de comunicação neste estudo não sofrerá com perdas de mensagens. A informação enviada é a posição atual do robô (coordenada X, Y) dado o sensoriamento. Os robôs estão organizados em um TDMA, onde cada robô possui um slot de tempo para cálculo do consenso e transmissão de dados.

O tempo de ciclo é de 100 ms e cada robôs possui 10ms de slot. A atualização do deslocamento é feita a cada 1 ms.

3. PANORAMA DO PROJETO E AGENDAMENTO

- **Quem faz o que e em qual ordem? Quando as variáveis de estado são atualizadas? Como o tempo é modelado?**

Os robôs executam os seus algoritmos de acordo com a ordem pré-estabelecida pelo TDMA. A cada interação, o robô realiza o controle cooperativo com os dados armazenados (posição atual do robô e dos robôs vizinhos), em seguida é feito o controle de velocidade e atualização do valor de referência para deslocamento. Em seguida, repassa as informações atualizadas para controle de topologia que verifica para quais robôs será feita a transmissão dos dados. O último algoritmo executado é a transmissão dos dados propriamente dita.

Ao finalizar seu slot, o robô aguarda informações dos robôs vizinhos. Caso receba uma mensagem, esta é repassada ao controle de topologia que decidirá se armazena o dado recebido ou se irá descartá-lo.

De forma independente, o deslocamento de todos os robôs é realizado a cada 1ms de acordo com a velocidade de cada um.

4. CONCEITOS DE PROJETO

- **Descrição do estudo em onze conceitos de projeto.**

A. PRINCÍPIOS BÁSICOS

- **Quais os conceitos, teorias, hipóteses e abordagens que o estudo envolve? Explicar a relação entre: os princípios básicos, o nível de complexidade que o estudo aborda e o propósito do estudo. Como estes são considerados? São utilizados submodelos? O estudo utiliza técnicas novas ou técnicas já desenvolvidas?**

O estudo engloba as áreas de controle, robótica e redes de comunicação. As técnicas utilizadas foram desenvolvidas por outros autores e validadas separadamente, sendo elas:

- a. **Tarefa Cooperativa:** *rendez-vous* médio, dada a facilidade de parametrizar quantitativamente e qualitativamente a conclusão da tarefa (CORTES, 2006; ORDÓÑEZ, 2013; KHAN, 2014; FENG, 2016; VUNDURTHEY, 2016; SABATTINI, 2016);
- b. **Estratégia de Controle Cooperativo:** consenso RHC aplicado a *rendez-vous* (ORDÓÑEZ, 2013).
- c. **Rede de Comunicação:** MANET devido a sua ampla utilização em WSN e robótica (WANG, 2003; AKEWAR, 2012; GIL, 2015; HASAN, 2008), protocolo UDP/IP (KUDELSKI, 2013), tecnologia Wi-Fi (KAHAR, 2012);
- d. **Protocolos de Rede Adicionais:** TDMA (WADUD, 2013; LEONE, 2013);
- e. **Controle de Topologia:** fixo e dinâmico (DAS, 2007; MI, 2010; NEVES, 2014);

- f. **Software de Simulação:** OMNeT++/INET (OMNET, 2016; INET *FRAMEWORK*, 2016; NAGEL, 2008).

É feito questionamento a respeito da simplificação da rede de comunicação em algoritmos de controle cooperativo, propondo que é necessário estudar o impacto da comunicação no controle em simulações mais complexas. Em especial, neste estudo questiona-se o impacto das condições iniciais das posições dos robôs no resultado da tarefa cooperativa, avaliando a densidade e robustez da matriz adjacência ao longo da simulação e o impacto nos resultados.

O escopo do trabalho foi restrito a robôs móveis terrestres, homogêneos e em um ambiente ideal. Também foi restrito às técnicas mencionadas anteriormente. Idealizou-se no momento o movimento dos robôs, o controle de velocidade e sensoriamento, devido a necessidade de diminuir as variáveis do estudo e compreender apenas a interação entre as técnicas utilizadas.

O questionamento de sub-modelos não foi considerado aplicável a este tipo de estudo, no entanto, ressalta-se que são vários modelos trabalhando simultaneamente.

B. EMERGÊNCIA

- **Quais resultados dos modelos são esperados variar de modo complexo e imprevisível, quando há mudança de uma característica particular dos indivíduos ou do ambiente? Existem outros resultados que são mais afetados pelas regras do modelo e menos dependentes do que os indivíduos fazem?**

Espera-se impacto no ponto final de convergência e no tempo de convergência dos robôs, ou seja, quando e onde todos os robôs se reúnem ao final da tarefa cooperativa. Dado que os robôs estão em constante movimento com comunicação limitada, pode haver situações nas quais o link de comunicação é quebrado, dividindo a rede em duas ou mais sub-redes, resultando em vários pontos de *rendez-vous* e por consequência, a falha da tarefa.

Já foi mostrado em outro estudo que o controle de topologia utilizado altera os resultados, a ponto de haverem casos onde a tarefa é concluída com sucesso e no outro não. O peso dado no controle cooperativo também afeta o comportamento dos robôs, podendo a levar a quebra inesperada da rede.

C. ADAPTAÇÃO

- **Quais as características adaptativas que os indivíduos possuem? Quais as regras que eles possuem para tomar decisões? Essas características buscam aumentar a probabilidade de sucesso (em relação aos objetivos) de alguma medição individual?**

Os robôs utilizam um controle cooperativo que possui uma predição de alguns instantes a respeito das possíveis posições do robô no futuro e a relaciona com o alcance de comunicação. No entanto, esta propriedade adaptativa é inerente ao controle cooperativo escolhido e o robô não realiza nenhuma outra predição fora deste algoritmo.

D. OBJETIVOS

- **Quais objetivos específicos de cada indivíduo? Como este é mensurado? Se são realizadas escolhas pelos agentes, são baseadas em quais critérios? Qual o objetivo do conjunto de indivíduos?**

O objetivo de cada robô é calcular e chegar ao mesmo ponto que os demais, mesmo não sabendo onde todos os robôs estão. Este ponto final de convergência de todos os robôs deve ser feito de forma descentralizada, visando que todos os robôs atinjam este ponto no menor tempo

possível, observando também a diferença nos resultados utilizando diferentes modelos de comunicação (ideal Matlab, próximo ao ideal OMNeT++ e real sem perdas no OMNeT++).

O tempo de convergência é calculado a partir do início da simulação até o momento que o critério de parada é atingido. O critério de parada, assim como o ponto final, é calculado quando todos os robôs atingem um ponto e possuem desvio padrão menor que 3cm.

O objetivo do conjunto de robôs é a reunião de todos integrantes no ponto que requer menos deslocamento (no total de todos os robôs) e menor tempo possível. É vital que todos os robôs atinjam este ponto ou caso contrário, considera-se que houve falha.

E. APRENDIZAGEM

- **Indivíduos ou agentes possuem características adaptativas que mudam ao passar do tempo como consequência da experiência adquirida? Se sim, como?**

Não há aprendizagem nos algoritmos utilizados.

F. PREDIÇÃO

- **Se existem previsões nos modelos e algoritmos utilizados, como estas são realizadas?**

A única previsão utilizada está incluída no equacionamento do algoritmo de controle cooperativo.

G. SENSORIAMENTO

- **Quais as variáveis internas e as variáveis do ambiente, que são medidas por sensores? Quais as variáveis de estado que os outros indivíduos e entidades podem perceber? Sensoriamento é local, recebido por rede ou global? Se é recebido pela rede, a estrutura da rede é importa ou emergente? Os mecanismos pelos quais os agentes obtêm informações é modelado explicitamente ou é assumido que eles sabem os valores?**

O sensoriamento é considerado ideal. Os robôs podem adquirir diretamente e instantaneamente a sua atual velocidade e posição nos eixos X e Y. Não há detecção de colisão ou de outros robôs (fisicamente).

No controle de topologia, o robô sabe que está no alcance do outro robô ao receber mensagens do mesmo, ao estar no alcance de comunicação. O alcance da comunicação é binário, o robô recebe e transmite dados se estiver no raio de 250cm do robô vizinho e não esteja localizado a mais de 250cm.

O sensoriamento do grupo é dado pela comunicação direta entre os robôs, que informam sua atual posição. A estrutura de comunicação importa e pode ser pré-determinada pelas condições iniciais (topologia fixa) ou emergente (topologia dinâmica).

H. INTERAÇÃO

- **Quais os tipos de interação existente entre os módulos e os robôs? Há interações diretas nas quais indivíduos afetam uns aos outros? E indiretas? Se a interação envolve comunicação, como esta é realizada?**

Há interação entre os módulos internos do robô (comunicação, controle topologia, controle cooperativo, controle individual) e interação entre os robôs.

O módulo de comunicação é o responsável por agendar os eventos, sejam estes internos ou eventos relacionados ao recebimento/envio de mensagens para outros robôs. O módulo inicializa-se no slot do TDMA do robô em questão e agenda uma auto-mensagem com o período

total do TDMA. O módulo fica a espera de qualquer mensagem externa ou de uma requisição interna para enviar mensagem.

A auto-mensagem é sinal que se deseja calcular as ações deste instante. Portanto, é acionado o controle cooperativo, que utiliza os dados armazenados no controle de topologia e calcula com as informações disponíveis, as velocidades do robô para atingir o ponto de consenso. Essas velocidades são enviadas ao controle de velocidade, finalizando o controle cooperativo.

O controle de velocidade limita a velocidade do robô e controla o robô até atingir o ponto especificado. A informação da velocidade atual do robô é repassada ao módulo de deslocamento do robô e suas informações de posição/velocidade são preparadas para serem repassadas aos outros robôs.

Em relação ao primeiro, o módulo de deslocamento não é ligado diretamente ao robô. Este módulo atualiza o display dos robôs no simulador de acordo com a velocidade atual dos mesmos e um modelo ideal de deslocamento (sem imprecisões). Este módulo cria auto-mensagens de altíssimas frequências e a cada instante atualiza os valores internos das posições atuais que cada robô possui de si mesmo, atuando como um sensor de localização ideal.

Já em relação ao segundo, os dados retornam ao módulo de comunicação, que prepara a mensagem a ser enviada. De acordo com o controle de topologia, o módulo de comunicação envia os dados aos outros robôs em seu alcance.

Quando o robô recebe uma mensagem externa, é iniciado o módulo de comunicação e em seguida o procedimento de verificação da mensagem, enviando os dados para o módulo de controle de topologia. Neste, é verificado se os dados serão utilizados ou não, dado o controle de topologia utilizado. Em caso afirmativo, os dados são armazenados para serem utilizados no controle cooperativo, no momento adequado.

I. ESTOCASTICIDADE

- **Há algum processo modelado com aspectos aleatórios? A estocasticidade é utilizada para reproduzir variabilidade no estudo? É utilizada para causar eventos ou comportamentos com uma frequência específica?**

Não é utilizado nenhum processo aleatório até o momento.

J. COLETIVIDADE

- **Os indivíduos formam ou pertencem a uma agregação que afeta e é afetada pelos indivíduos? Como o coletivo é representado? O coletivo é uma propriedade emergente dos indivíduos ou é simplesmente uma definição do projetista?**

Os robôs formam possuem um comportamento coletivo descentralizado que surge por meio de ações individuais. Como o controle cooperativo tende a aproximar o robô de seus vizinhos, o *rendez-vous* total é alcançado ao longo de várias interações e movimentos.

A comunicação e interação, é representada por meio de um grafo orientado, onde os robôs são os nós e a possível comunicação entre os mesmos, representados por setas. Na implementação, é utilizada uma representação matricial. Dois fatores determinam os valores desta matriz: se os robôs vizinhos estão no alcance da comunicação sem fio do robô atual e o algoritmo de controle de topologia utilizado.

A matriz pode possuir um aspecto baseado na topologia física (alcance de comunicação) ou pode ser alterada pelo controle de topologia, resultando em uma topologia lógica.

K. OBSERVAÇÃO

- **Quais dados são coletados do estudo sistema cooperativo para teste, compreensão e análise? Como estes são coletados? Todos os dados são utilizados ou é realizada uma amostragem?**

São coletadas as informações sobre posição e velocidade de cada robô ao longo do trajeto e o tanto o ponto quanto o tempo de convergência final do *rendez-vous*. Para análise são utilizados apenas os dois últimos.

5. INICIALIZAÇÃO

- **Qual é o estado inicial do modelo do ambiente? Quantas entidades e de quais tipos são inicializadas, qual o valor exato de suas variáveis de estado? A inicialização é sempre igual ou pode ser alterada entre experimentos? Os valores foram escolhidos aleatoriamente?**

Os valores foram escolhidos de acordo com um robô de pequeno porte, principalmente em relação aos limites de velocidade e alcance de comunicação.

As condições iniciais utilizadas são:

Número de robôs (N)	10
Posição Inicial X	[600; 800; 800; 800; 800; 600; 900; 1100; 1300; 1300; 1300]
Posição Inicial Y	[430; 430; 630; 830; 1030; 1030; 450; 450; 450; 550; 850]
Limite de Velocidade	10cm/s and -10cm/s
Alcance de comunicação	250cm
Passos de predição (p)	10
Peso do link (a)	1
Peso da Variação de Velocidade (λ)	1
Passo de controle	100ms
Ciclo TDMA	100ms

6. ENTRADA

- **O modelo usa alguma fonte externa de dados para entrada do sistema?**

Os únicos dados de entrada são quais os algoritmos que estão habilitados e as condições iniciais.

7. SUBMODELOS

- **Detalhar os submodelos mencionados no item 3. Quais seus parâmetros, dimensões e valores de referência? Como foram projetados ou escolhidos? Como são parametrizados e testados?**

Não há submodelos.

BIBLIOGRAFIA

Bibliografia utilizada no documento segue o mesmo modelo apresentado na tese, sendo omitida neste anexo em razão da duplicidade com a bibliografia da tese.

ANEXO C– MATERIAL ADICIONAL

Tabela 12 – Pontos críticos $t_{v,\gamma}$ para distribuição t-student com grau de liberdade v que satisfaz a probabilidade γ .

v	0.6000	0.7000	0.8000	0.9000	0.9333	0.9500	0.9600	0.9667	0.9750	0.9800	0.9833	0.9875	0.9900	0.9917	0.9938	0.9950
1	0.325	0.727	1.376	3.078	4.702	6.314	7.916	9.524	12.706	15.895	19.043	25.452	31.822	38.342	51.334	63.657
2	0.289	0.617	1.061	1.886	2.456	2.920	3.320	3.679	4.303	4.849	5.334	6.205	6.965	7.665	8.897	9.925
3	0.277	0.584	0.978	1.638	2.145	2.535	2.865	3.182	3.738	4.184	4.596	4.977	5.341	5.864	6.841	7.641
4	0.271	0.569	0.941	1.533	1.970	2.332	2.633	2.902	3.476	3.899	4.284	4.695	5.047	5.466	6.328	7.007
5	0.267	0.553	0.906	1.440	1.735	2.045	2.191	2.337	2.911	3.277	3.612	3.969	4.343	4.681	5.441	6.019
6	0.265	0.549	0.889	1.415	1.698	1.895	2.044	2.170	2.365	2.517	2.640	2.841	2.998	3.130	3.541	3.999
7	0.263	0.546	0.883	1.383	1.650	1.850	2.004	2.122	2.306	2.449	2.565	2.752	2.896	3.018	3.211	3.355
8	0.262	0.543	0.879	1.372	1.634	1.832	1.948	2.058	2.238	2.359	2.465	2.634	2.764	2.872	3.043	3.169
9	0.262	0.542	0.876	1.363	1.621	1.816	1.928	2.017	2.179	2.303	2.402	2.560	2.681	2.782	2.959	3.055
10	0.262	0.540	0.873	1.356	1.610	1.796	1.906	2.002	2.160	2.282	2.379	2.533	2.650	2.748	2.900	3.012
11	0.262	0.539	0.870	1.350	1.601	1.781	1.889	1.989	2.145	2.264	2.359	2.510	2.624	2.720	2.868	2.977
12	0.262	0.538	0.868	1.345	1.593	1.763	1.867	1.968	2.120	2.235	2.327	2.473	2.583	2.675	2.817	2.921
13	0.262	0.538	0.866	1.341	1.587	1.753	1.878	1.978	2.131	2.249	2.342	2.490	2.602	2.696	2.841	2.947
14	0.262	0.536	0.865	1.337	1.581	1.746	1.869	1.968	2.110	2.228	2.315	2.458	2.567	2.657	2.796	2.898
15	0.262	0.534	0.865	1.333	1.576	1.740	1.862	1.960	2.100	2.214	2.303	2.445	2.552	2.641	2.778	2.878
16	0.262	0.533	0.864	1.330	1.572	1.734	1.855	1.953	2.101	2.214	2.302	2.443	2.550	2.639	2.776	2.876
17	0.262	0.533	0.861	1.328	1.568	1.729	1.850	1.946	2.086	2.205	2.293	2.433	2.539	2.627	2.762	2.861
18	0.262	0.533	0.860	1.325	1.564	1.725	1.844	1.940	2.086	2.199	2.285	2.423	2.528	2.614	2.748	2.845
19	0.262	0.532	0.859	1.323	1.561	1.721	1.840	1.935	2.080	2.189	2.277	2.414	2.518	2.603	2.735	2.831
20	0.262	0.532	0.858	1.321	1.558	1.717	1.835	1.930	2.074	2.183	2.269	2.405	2.508	2.593	2.724	2.819
21	0.262	0.532	0.858	1.319	1.556	1.714	1.832	1.926	2.069	2.177	2.263	2.399	2.492	2.575	2.704	2.797
22	0.262	0.531	0.857	1.318	1.553	1.711	1.828	1.922	2.064	2.172	2.257	2.391	2.485	2.568	2.695	2.787
23	0.262	0.531	0.856	1.316	1.551	1.708	1.825	1.918	2.060	2.168	2.252	2.385	2.479	2.561	2.687	2.779
24	0.262	0.531	0.856	1.315	1.549	1.706	1.822	1.915	2.056	2.162	2.246	2.379	2.473	2.554	2.680	2.771
25	0.262	0.531	0.855	1.314	1.547	1.703	1.819	1.912	2.052	2.158	2.242	2.373	2.467	2.548	2.673	2.763
26	0.262	0.530	0.855	1.313	1.546	1.701	1.817	1.909	2.048	2.154	2.237	2.368	2.462	2.543	2.667	2.756
27	0.262	0.530	0.854	1.311	1.544	1.699	1.814	1.906	2.045	2.150	2.233	2.364	2.457	2.537	2.661	2.750
28	0.262	0.530	0.854	1.310	1.543	1.694	1.812	1.904	2.042	2.147	2.230	2.360	2.452	2.532	2.656	2.745
29	0.262	0.529	0.851	1.303	1.532	1.684	1.796	1.886	2.021	2.123	2.203	2.329	2.423	2.501	2.619	2.704
30	0.262	0.528	0.849	1.299	1.526	1.676	1.787	1.875	2.009	2.109	2.188	2.311	2.403	2.479	2.594	2.678
40	0.255	0.528	0.846	1.293	1.517	1.665	1.775	1.861	1.992	2.091	2.167	2.287	2.377	2.450	2.562	2.643
50	0.254	0.527	0.845	1.290	1.513	1.660	1.769	1.855	1.984	2.081	2.157	2.276	2.364	2.436	2.547	2.626
75	0.254	0.526	0.844	1.282	1.501	1.645	1.751	1.834	1.960	2.054	2.127	2.241	2.326	2.395	2.501	2.576
100	0.253	0.524	0.842	1.282	1.501	1.645	1.751	1.834	1.960	2.054	2.127	2.241	2.326	2.395	2.501	2.576

Fonte: LAW (2015)