



Universidade de Coimbra

Faculdade de Ciências e Tecnologia

Mestrado Integrado em Engenharia Electrotécnica e de Computadores

– Especialização em Automação –

Algoritmos de cobertura para frotas de robôs sujeitos a restrições

Diogo Lopes Ferreira

Orientador: Professor Doutor Hélder de Jesus Araújo

Júri:

Presidente: Professor Doutor Rui Pedro Duarte Cortesão

Vogais: Professor Doutor Hélder de Jesus Araújo

Professor Doutor Rui Paulo Pinto da Rocha

Coimbra, Julho de 2012

"O que sabemos é uma gota de água, o que ignoramos é um oceano."

Isaac Newton

Agradecimentos

A realização desta Dissertação de Mestrado só foi possível graças à colaboração e ao contributo, de forma directa ou indirecta, de várias pessoas, às quais gostaria de exprimir algumas palavras de agradecimento e profundo reconhecimento.

No âmbito desta dissertação gostaria de agradecer ao Professor Doutor Hélder Araújo, pela disponibilidade manifestada para orientar este trabalho, pela preciosa ajuda na definição do objectivo de estudo, pela revisão crítica do texto, pelos profícuos comentários, esclarecimentos, opiniões e sugestões, pela acessibilidade, cordialidade e simpatia demonstradas.

O maior agradecimento, como não podia deixar de ser, vai para os meus pais, pelo enorme esforço que realizaram para me proporcionar a oportunidade de estudar e por todo o incentivo que me deram durante todo o meu percurso académico.

Gostaria de agradecer do fundo do coração à minha namorada pela paciência, carinho, compreensão, incentivo e apoio demonstrado ao longo destes anos.

Um grande obrigado a toda a minha família, pelo apoio e ajuda ao longo do curso.

Por fim, gostaria de agradecer a todos os meus amigos e colegas por todos os momentos passados ao longo destes anos, em especial ao David, Leonel, Ricardo, Márcia, Joana, João.

A todos, enfim, reitero o meu apreço e a minha eterna gratidão!

Resumo

Actualmente, podem ser utilizados robôs para o cumprimento eficiente de tarefas de alto risco, até aqui realizadas por seres humanos, tais como, detecção de minas subaquáticas, mapeamento oceanográfico ou até mesmo operações de busca e salvamento em cenários de catástrofe, de modo a reduzir ao máximo o risco de perdas humanas. Na realização destas tarefas, o objectivo principal passa por efectuar um reconhecimento (ou cobertura) de um ambiente desconhecido (parcial ou totalmente) da forma mais eficiente possível com recurso à utilização de robôs.

A presente dissertação descreve quatro algoritmos de cobertura e a sua implementação, onde, foram realizadas diversas simulações de modo a avaliar o seu comportamento e desempenho. Uma vez que a cobertura pode ser efectuada por uma equipa de vários robôs e que a comunicação entre eles é um factor chave aquando da transposição dos algoritmos para o mundo real, decidiu-se incorporar este aspecto na simulação de forma a obter um comportamento mais próximo possível da realidade.

Esta dissertação foi realizada no âmbito do projecto “MORFEU – Frota Robótica Multiobjectivo para Comunicação Eficiente”, financiado por Fundação da Ciência e Tecnologia sob a referência PTDC/EEA-CRO/108348/2008.

Palavras-chave: Algoritmo de Cobertura, Boustrophedon, Backtracking Spiral Algorithm, Coordenação de múltiplos robôs, Comunicação entre robôs.

Abstract

Currently, robots can be used to realize efficiently high-risk tasks so far performed by human beings, such as, underwater mine detection, oceanographic recognition or even perform search-rescue missions in disaster scenarios to minimize the loss of human life's. During this tasks realization, the main goal is to accomplish the coverage of unknown environments (partially or fully) as efficiently as possible using robots.

This dissertation describes four coverage algorithms and their implementation, where, several simulations are done to evaluate their behavior and performance. Once the coverage can be made by a team of robots, and the communication between them represent a key factor in a real world implementation, was decided to incorporate this feature in the simulation to obtain a behavior closer to reality.

This dissertation was made in the sphere of project “MORFEU – Multi-Objective Robot Fleet for improved communication”, funded by the Foundation of Science and Technology under reference PTDC/EEA-CRO/108348/2008.

Keywords: Coverage Algorithm, boustrophedon, Backtracking Spiral Algorithm, Coordination of multiple robots, communication between robots.

Índice

Índice	6
Lista de Figuras	7
1. Introdução	10
1.1. Enquadramento do Projecto	10
1.2. Objectivos	10
1.3. Organização da Dissertação	10
2. Estado da arte	11
3. Ambiente de Desenvolvimento	12
4. Algoritmo de Decomposição de Células de <i>Boustrophedon</i>	12
4.1. Introdução	12
4.2. Descrição Algoritmo	13
4.3. Resultados Experimentais	16
4.4. Discussão	21
5. Algoritmo de Backtracking em espiral	21
5.1. Introdução	21
5.2. Descrição Algoritmo	22
5.1. Resultados Experimentais	24
5.2. Discussão	28
6. Algoritmo de exploração multi-robôs coordenados	28
6.1. Introdução	28
6.2. Descrição Algoritmo	28
6.3. Resultados Experimentais	31
6.4. Discussão	34
7. Algoritmo de exploração e mapeamento de multi-robôs coordenados	35
7.1. Introdução	35
7.2. Descrição Algoritmo	35
7.3. Resultados Experimentais	37
7.4. Discussão	43
8. Discussão e Conclusão	44
9. Trabalho Futuro	47
10. Referências Bibliográficas	48
11. Anexos	50

Lista de Figuras

Figura 1 - Sequência de movimentos do algoritmo Boustrophedon dentro de uma célula.	12
Figura 2 - Ilustração de um grafo de adjacências	13
Figura 3 - Evento de INICIO	14
Figura 4 - Comparação de movimentos necessários na cobertura do mesmo espaço para duas células e para apenas uma célula.....	14
Figura 5 - Decomposição Trapezoidal e Decomposição Boustrophedon	14
Figura 6 - Processo de decomposição de células e consequente grafo de adjacências ..	15
Figura 7 - Conjunto de mapas utilizados nas simulações.....	16
Figura 8 - Percentagem do varrimento de um robô para os vários mapas.....	16
Figura 9 - Iterações obtidas pelo robô nos vários mapas.....	17
Figura 10 - Percentagens obtidas para 1,3,5 e 8 robôs para os vários mapas.....	18
Figura 11 - Iterações obtidas para 1,3,5 e 8 robôs para os vários mapas.	18
Figura 12 - Percentagens obtidas para 1,3,5 e 8 robôs para os vários mapas.....	19
Figura 13 - Iterações obtidas para 1,3,5 e 8 robôs para os vários mapas.	19
Figura 14 - Percentagens obtidas para 1,3,5 e 8 robôs com comunicação para os vários mapas.....	20
Figura 15 - Iterações obtidas para 1,3,5 e 8 robôs com comunicação para os vários mapas.....	20
Figura 16 - Exemplo de cobertura realizado pelo algoritmo BSA.	23
Figura 17 - Percentagens obtidas com 1 robô para os vários mapas.	24
Figura 18 - Iterações obtidas com 1 robô para os vários mapas.....	24
Figura 19 - Percentagens obtidas para 1,3,5 e 8 robôs para os vários mapas.....	25
Figura 20 - Iterações obtidas para 1,3,5 e 8 robôs para os vários mapas.	25
Figura 21 - Percentagens obtidas para 1,3,5 e 8 robôs para os vários mapas.....	26
Figura 22 - Iterações obtidas para 1,3,5 e 8 robôs para os vários mapas.	26
Figura 23 - Percentagens obtidas para 1,3,5 e 8 robôs com comunicação para os vários mapas.....	27
Figura 24 - Iterações obtidas para 1,3,5 e 8 robôs com comunicação para os vários mapas.....	27
Figura 25 - Valores típicos das funções, obtidos para duas posições diferentes do robô.	30
Figura 26 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1.	32

Figura 27 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1.....	32
Figura 28 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas na métrica 2.	32
Figura 29 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas na métrica 2.....	32
Figura 30 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1 com restrições de comunicação.	33
Figura 31 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1 com restrições de comunicação.	33
Figura 32 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas na métrica 2 com restrições de comunicação.	33
Figura 33 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas na métrica 2 com restrições de comunicação.	33
Figura 34 - Ganho de Informação esperado.	37
Figura 35 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1.	38
Figura 36 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1.....	38
Figura 37 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas para a métrica 2.	39
Figura 38 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas para a métrica 2.	39
Figura 39 - Percentagens obtidas para 1,3 e 5 robôs com restrições de comunicação para os vários mapas na métrica 1.	40
Figura 40 - Iterações obtidas para 1,3 e 5 robôs com restrições de comunicação para os vários mapas na métrica 1.	41
Figura 41 - Percentagens obtidas para 1,3 e 5 robôs com restrições de comunicação para os vários mapas na métrica 2.	42
Figura 42 - Iterações obtidas para 1,3 e 5 robôs com restrições de comunicação para os vários mapas na métrica 2.	43
Figura 43 - Gráfico de Iterações com todos os algoritmos sem comunicação para 1 robô.	50
Figura 44 - Gráfico de iterações com todos os algoritmos sem comunicação para 3 robôs.	50
Figura 45 - Gráfico de iterações com todos os algoritmos sem comunicação para 5 robôs.	51
Figura 46 - Gráfico de percentagens com todos os algoritmos sem comunicação para 1 robô.	51

Figura 47 - Gráfico de percentagens com todos os algoritmos sem comunicação para 3 robôs.	51
Figura 48 - Gráfico de percentagens com todos os algoritmos sem comunicação para 5 robôs.	52

1. Introdução

1.1. Enquadramento do Projecto

A presente dissertação foi realizada no âmbito do projecto “MORFEU – Frota Robótica Multiobjectivo para Comunicação Eficiente” que visa à implementação de uma equipa de robôs com comunicação entre eles com o objectivo de efectuar cobertura num ambiente desconhecido.

Com o objectivo de estudar e implementar vários algoritmos de cobertura, foi desenvolvido um simulador com o intuito de efectuar uma análise de desempenho quer a nível do varrimento total da área simulada quer em termos do número total de movimentos efectuados pelos robôs.

1.2. Objectivos

Esta dissertação teve como objectivo o desenvolvimento de uma ferramenta com capacidade para simular e analisar o desempenho dos algoritmos de cobertura para uma equipa de robôs, sujeita a determinadas restrições tais como a comunicação entre eles.

Os algoritmos estudados foram sujeitos a modificações de forma a aumentar o seu desempenho a nível da máxima cobertura do mapa. A avaliação realizada constitui um mecanismo de apoio à decisão da escolha do algoritmo com melhor performance, a ser implementado no mundo real.

1.3. Organização da Dissertação

No primeiro capítulo são abordados os objectivos da dissertação, bem como o enquadramento do projecto e a estrutura da dissertação. Um breve resumo do estado da arte e uma descrição do ambiente de simulação é feito nos capítulos 2 e 3. Nos capítulos seguintes serão descritos os algoritmos implementados contendo uma breve introdução seguida de uma descrição do algoritmo bem como uma discussão dos resultados obtidos. Surgindo assim no capítulo 4 o algoritmo de decomposição de células de *Boustrophedon*, no capítulo 5 será explicado o algoritmo de Backtracking em espiral, já no capítulo 6 será descrito o algoritmo de exploração multi-robôs coordenados e para finalizar o que diz respeito aos algoritmos no capítulo 7 será analisado o algoritmo de exploração e mapeamento de multi-robôs coordenados. No capítulo 8 serão discutidos os resultados e enunciadas as conclusões obtidas.

2. Estado da arte

O crescente aparecimento de robôs para a realização de tarefas até aqui realizadas por humanos, em que sejam necessários efectuar varrimentos do espaço, levou a um estudo de diversos algoritmos com o intuito de realizar a tarefa de cobertura da forma mais eficiente possível e com o menor custo.

Inicialmente os algoritmos desenvolvidos foram orientados para o uso de um só robô com movimentos simples, como é o caso dos algoritmos de *Boustrophedon* [1] [2] e do algoritmo de BSA [3]. Com o surgimento da necessidade de efectuar tarefas com múltiplos robôs, começaram a surgir novos tipos de algoritmos baseados em leituras dos sensores dos robôs [4] [5] [6].

Uma grande parte dos algoritmos baseados em sensores podem ser alterados para incorporar um sistema de GPS [7] permitindo obter uma melhor estimativa da localização de robôs em ambientes exteriores. Um dos modelos utilizados para uma cobertura eficiente de múltiplos robôs faz uso de *Spanning Trees* [8] [9] para a criação de caminhos de cobertura.

No surgimento dos algoritmos de cobertura para múltiplos robôs, sentiu-se desde logo necessidade de existir comunicação entre uma base partilhada e os vários robôs [10], no entanto este tipo de algoritmos apenas é válido para espaços pequenos onde todos os robôs estejam em constante ligação com a base. De modo a suprimir esta limitação foram desenvolvidos algoritmos onde a comunicação dos robôs deixa de ser efectuada entre os robôs e a base para passar a ser efectuada apenas entre robôs [11] [12] [13] [14]. Existem vários trabalhos realizados na área que analisam o melhor sistema de comunicação entre robôs, baseados em tecnologia Bluetooth [15] ou em tecnologia Wireless [16].

3. Ambiente de Desenvolvimento

A simulação dos algoritmos foi realizada em ambiente MATLAB, devido ao facto de se tratar de uma linguagem de programação de alto nível, o que permite um rápido desenvolvimento das tarefas propostas. Por se tratar de uma poderosa ferramenta de cálculo numérico, com funções orientadas à manipulação de imagens e de matrizes e por também possuir funcionalidades para a visualização de todo o processo inerente aos algoritmos e efectuar a sua análise estatística, esta linguagem provou ser a melhor opção para o desenvolvimento deste projecto.

4. Algoritmo de Decomposição de Células de *Boustrophedon*

4.1. Introdução

Existem várias situações nas quais este algoritmo de decomposição de células pode ser utilizado tais como, aplicações marítimas para detecção de minas subaquáticas, mapeamento oceanográfico e aplicações comerciais (descontaminações, limpeza de pavimentos e inspeção de infra-estruturas).

O presente algoritmo explora uma estrutura geométrica denominada decomposição exacta de células, que não é mais que a união de regiões não interceptáveis que constituem o ambiente. Cada região é denominada como célula e a união destas preenche o ambiente livre de obstáculos. Em cada célula é necessário efectuar um trajecto que permita fazer o seu reconhecimento na totalidade, trajecto este, denominado de caminho de cobertura. O caminho de cobertura efectuado pelo algoritmo utiliza movimentos verticais realizados em direcções opostas de forma intercalada como ilustrado na Figura 1. Uma vez concluída a cobertura da célula actual, o robô desloca-se para uma célula vizinha, repetindo todo processo até ser completada a cobertura de toda a região.

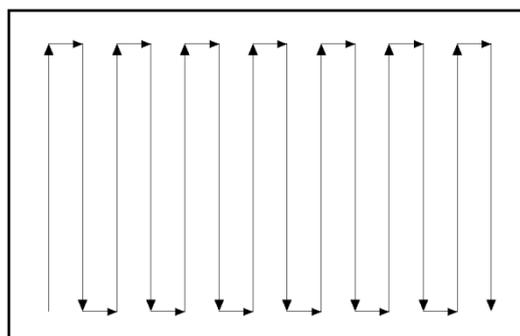


Figura 1 - Sequência de movimentos do algoritmo Boustrophedon dentro de uma célula.

4.2. Descrição Algoritmo

A aproximação usada no artigo [1] define a decomposição exacta de células como uma técnica, onde o espaço livre do robô é decomposto em células, podendo estas ser representadas como um nó de um grafo, em que as células adjacentes são interligadas. Este tipo de grafo é denominado de *grafo de adjacências* (Figura 2).

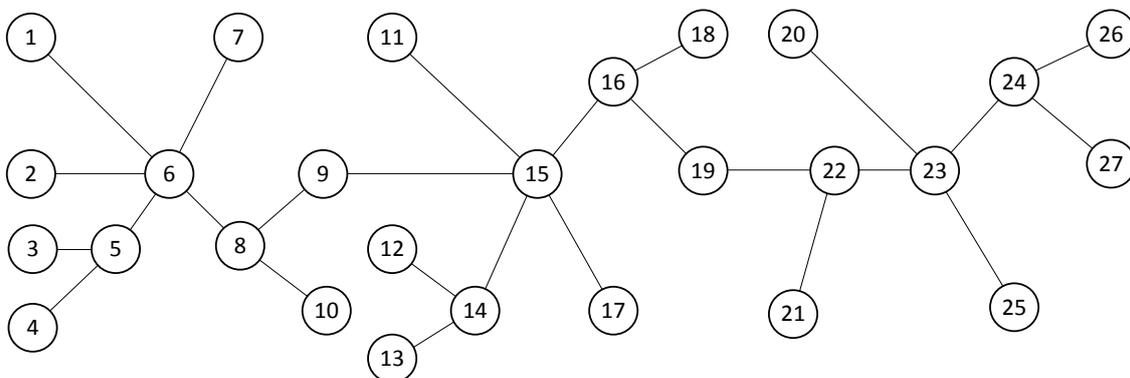


Figura 2 - Ilustração de um grafo de adjacências

O algoritmo apresentado, é baseado na decomposição trapezoidal, ilustrada na Figura 5. Na primeira etapa da decomposição trapezoidal o mapa é analisado como sendo uma só célula (podendo conter obstáculos), sob o qual é efectuado um varrimento na procura de vértices de obstáculos. A detecção do primeiro vértice (Figura 3) de um obstáculo desencadeia uma operação de abertura denominada de INICIO onde o espaço não analisado é seccionado através de uma divisão paralela ao deslocamento do varrimento dando originando uma divergência. De forma contrária, o FIM é o momento em que é detectado o último vértice do obstáculo após o qual existe a convergência das células anteriores, como demonstra a Figura 5 onde as células 9 e 10 convergem na célula 11. Entre os eventos INICIO e FIM, existe ainda o evento MEIO, onde para cada detecção de um novo vértice é terminada a célula actual e criada uma nova célula a começar nesse mesmo vértice.

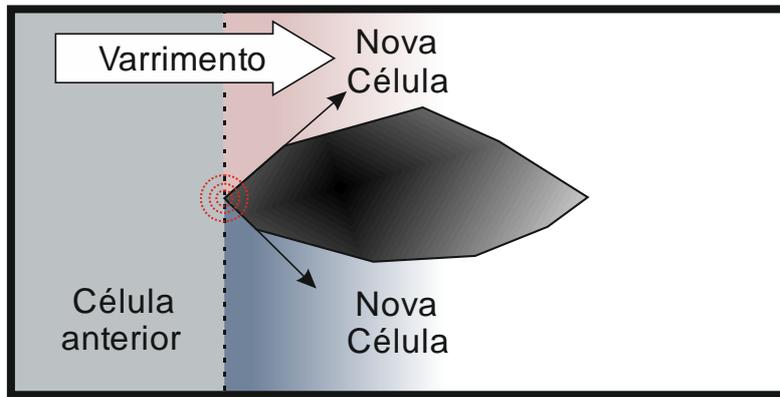


Figura 3 - Evento de INICIO

A aproximação trapezoidal divide o espaço num elevado número de células levando a um maior número de movimentos realizados pelo robô para efectuar a cobertura de todas as células dos eventos MEIO. De modo a garantir a cobertura completa do mapa de forma mais otimizada, é utilizada a decomposição *Boustrophedon*, onde todas as células que se situem no evento MEIO são agrupadas numa só célula de modo a que o número de movimentos seja menor (Figura 4).

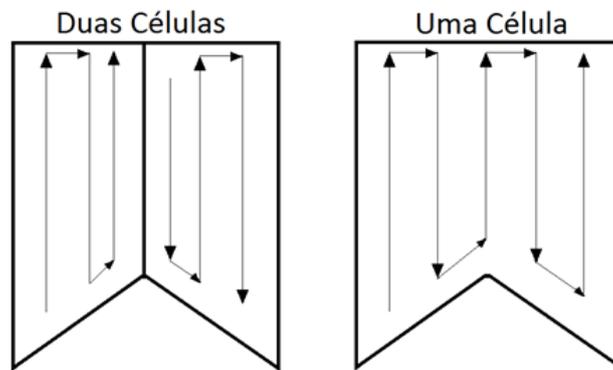


Figura 4 - Comparação de movimentos necessários na cobertura do mesmo espaço para duas células e para apenas uma célula.

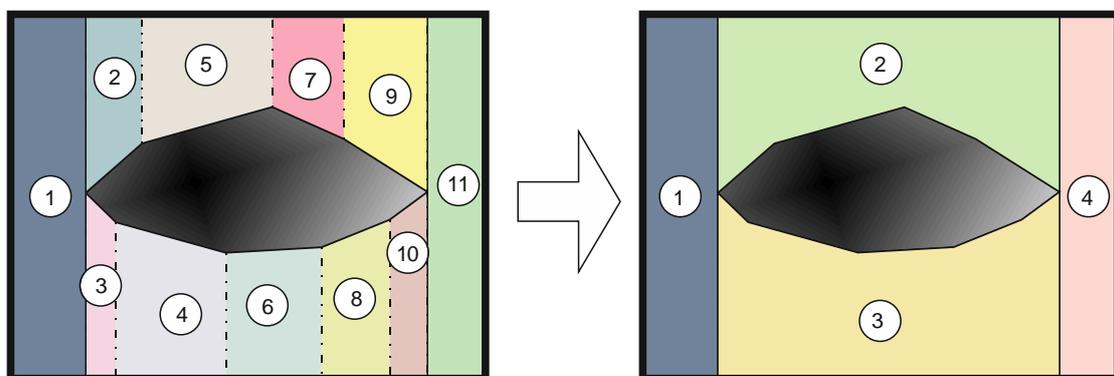


Figura 5 - Decomposição Trapezoidal e Decomposição Boustrophedon

Na Figura 7 são ilustradas as principais etapas do processo de análise de decomposição de células através da decomposição *Boustrophedon*. Tal como descrito anteriormente o processo é iniciado pela divisão do mapa em células e concluído com a criação de um grafo de adjacências (Figura 2) que efectua a ligação de todas as células adjacentes.

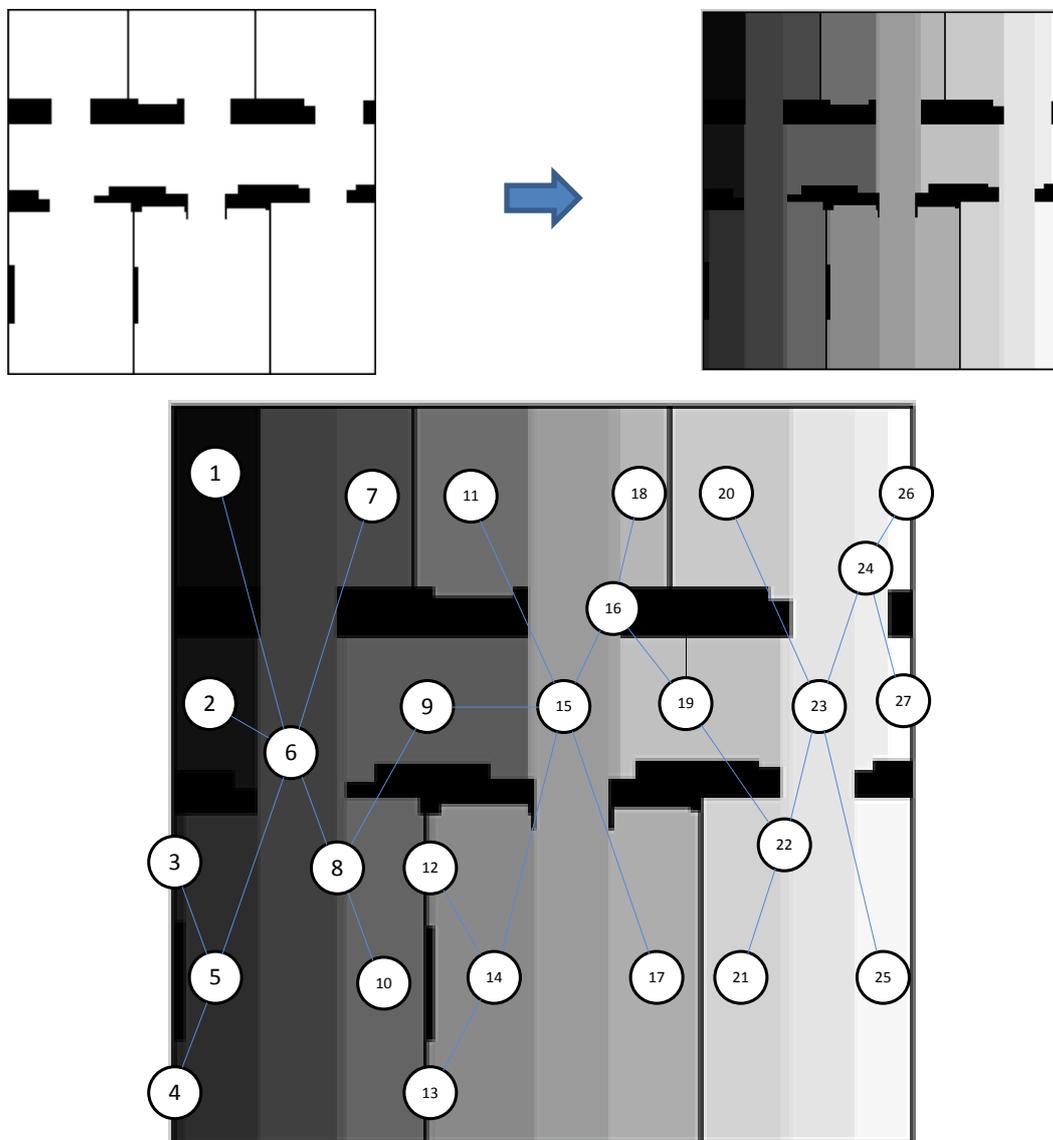


Figura 6 - Processo de decomposição de células e conseqüente grafo de adjacências

4.3. Resultados Experimentais

Numa primeira fase foi seguido o artigo original [1] que visava a implementação do algoritmo para apenas um robô. Com o intuito de analisar a performance do algoritmo para diversas situações foram criados seis mapas distintos baseados no artigo [17].

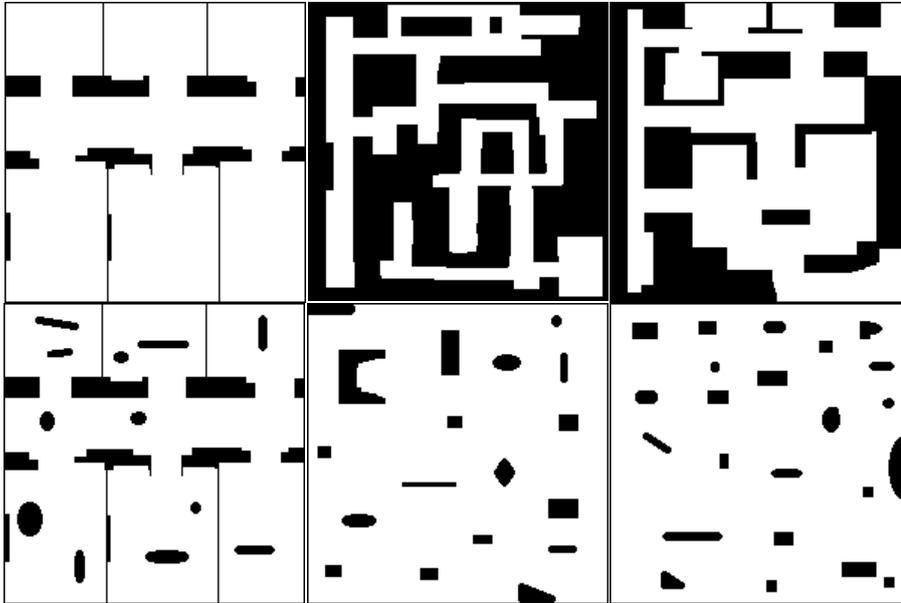


Figura 7 - Conjunto de mapas utilizados nas simulações.

Estes mapas (Figura 7) são utilizados ao longo desta dissertação para testar os algoritmos implementados com o objectivo de efectuar uma análise comparativa do desempenho dos mesmos. Para cada um dos mapas foram realizados dez testes, nos quais o robô é inicialmente posicionado em locais distintos de modo a analisar o comportamento do algoritmo.

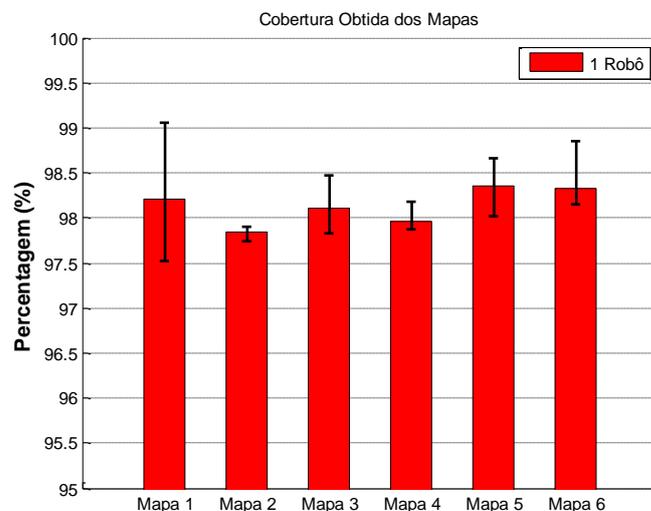


Figura 8 - Percentagem do varrimento de um robô para os vários mapas

No gráfico da Figura 8 constam as médias das percentagens da cobertura efectuada pelo robô, assim como os seus valores mínimos e máximos para todos os mapas. Importa sublinhar que nas simulações efectuadas para apenas um robô os valores de cobertura obtidos nunca foram inferiores a 97.5%.

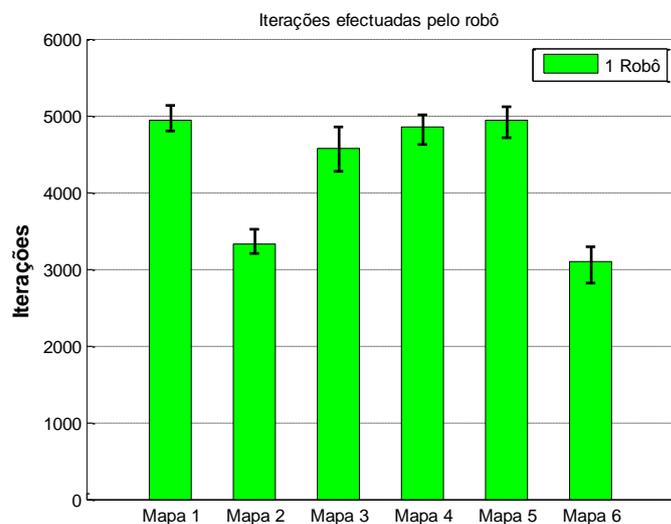


Figura 9 - Iterações obtidas pelo robô nos vários mapas

No decorrer das simulações, outro dos parâmetros avaliados foi o número total de movimentos (ou interações) efectuados pelo robô para que a pesquisa do algoritmo seja completada. Na Figura 9 verifica-se que o robô necessita de realizar um número mínimo de interações ligeiramente abaixo das 3000 para executar a cobertura do mapa.

Numa segunda fase, o objectivo passou por realizar a adaptação do algoritmo de modo a que a cobertura pudesse ser efectuada por um grupo de robôs. Uma vez que no algoritmo descrito é necessário que seja atribuída uma célula para que o robô inicie a pesquisa, no caso de múltiplos robôs serem inicialmente colocados na mesma célula, a cobertura da célula fica atribuída apenas a um destes, os outros executam uma pesquisa na vizinhança de forma a encontrar células que não tenham nenhum robô alocado de modo a dar início à execução do algoritmo.

Foi necessário também o desenvolvimento/implementação de mecanismos de detecção de colisões para que o caminho de cobertura seja realizado sem que haja embates com obstáculos. Para além deste mecanismo, um outro foi implementado de modo a evitar a colisão de elementos da frota que efectuem cobertura em células com o trajecto realizado por um robô que se desloca entre células.

Para analisar o desempenho do algoritmo em função do número de robôs foi realizado um conjunto de testes em que a performance da situação em que apenas é

usado um robô foi comparada com o desempenho obtido com equipas de três, cinco e oito robôs. No gráfico seguinte podem ser visualizados os resultados obtidos.

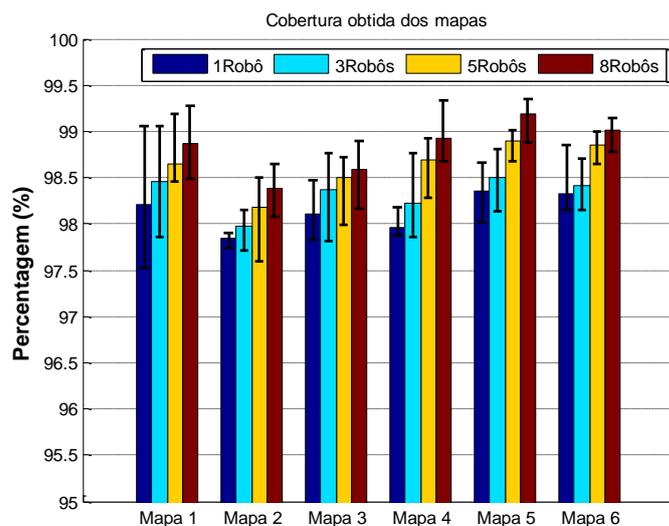


Figura 10 - Percentagens obtidas para 1,3,5 e 8 robôs para os vários mapas.

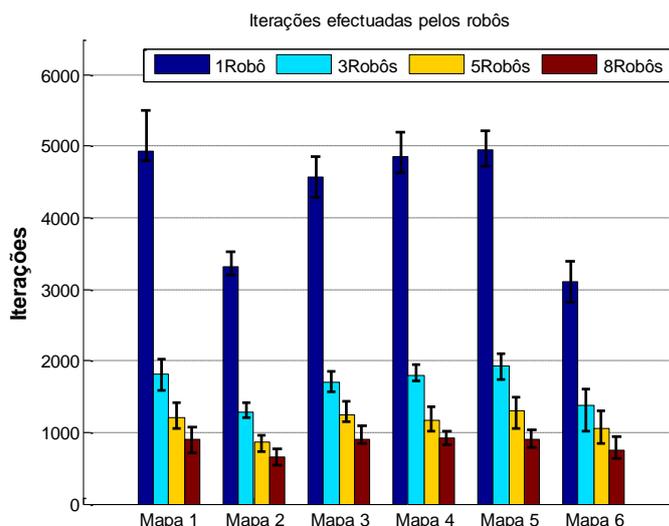


Figura 11 - Iterações obtidas para 1,3,5 e 8 robôs para os vários mapas.

Os resultados obtidos apresentam valores de percentagens, em média acima de 97.5% para qualquer um dos mapas. Apesar de os resultados serem satisfatórios verificou-se que eventualmente, dependendo da complexidade do mapa, pequenas áreas do mapa ficaram por pesquisar. Assim sendo foi implementado um novo conceito que consiste, após o término da pesquisa do algoritmo, realizar uma pesquisa no mapa em busca de zonas ainda não visitadas com um mínimo de 25 píxéis contíguos. É então criada uma lista das zonas por analisar que vão sendo distribuídas aos robôs até todas as

zonas listadas sejam atribuídas. Nos percursos entre zonas, cada robô desloca-se para a zona atribuída sem recurso ao algoritmo utilizando apenas desvio de obstáculos.

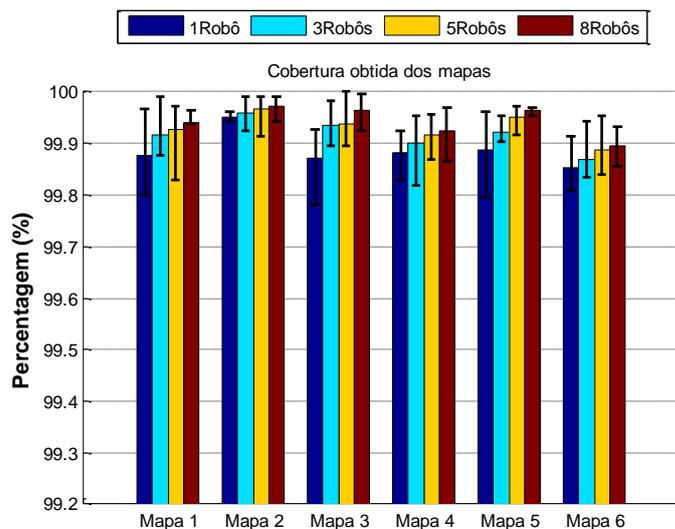


Figura 12 - Percentagens obtidas para 1,3,5 e 8 robôs para os vários mapas.

Após a inclusão do conceito descrito anteriormente verificou-se que as percentagens obtidas melhoraram significativamente, chegando a atingir em média, valores superiores a 99.8% para todos os mapas.

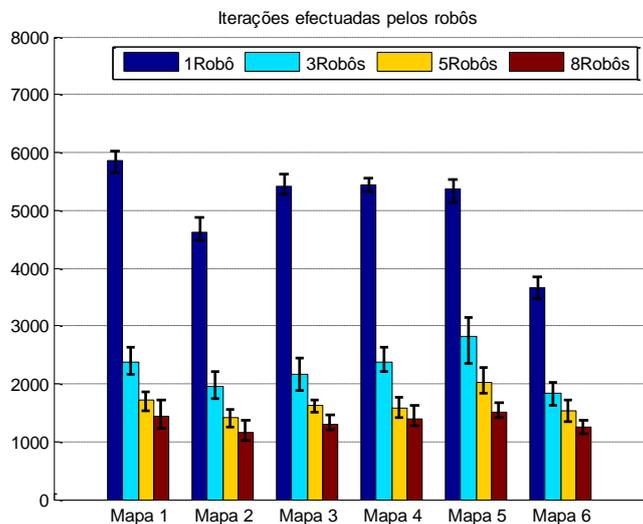


Figura 13 - Iterações obtidas para 1,3,5 e 8 robôs para os vários mapas.

Como pode ser visualizado na Figura 13, houve um ligeiro acréscimo do número de iterações, decorrente da extensão efectuada ao algoritmo, que obriga os robôs a visitar espaços com áreas (superiores a 25 pixéis contíguos) não cobertas.

Para uma melhor aproximação da realidade, a fase final de desenvolvimento deste algoritmo consistiu na inclusão da restrição referente às comunicações entre robôs.

A restrição impõe em que no instante da escolha de uma nova célula por um robô, esta seja feita de forma a que todos os robôs possam comunicar entre si, directa ou indirectamente, formando uma rede onde os nós (robôs) estão interligados. Outra característica da limitação imposta para a comunicação, especifica que em nenhum momento a distância entre dois robôs deve ser superior a 40 pixéis.

Os gráficos seguintes apresentam os resultados obtidos para os vários grupos de robôs e os vários mapas com a inclusão da restrição de comunicação.

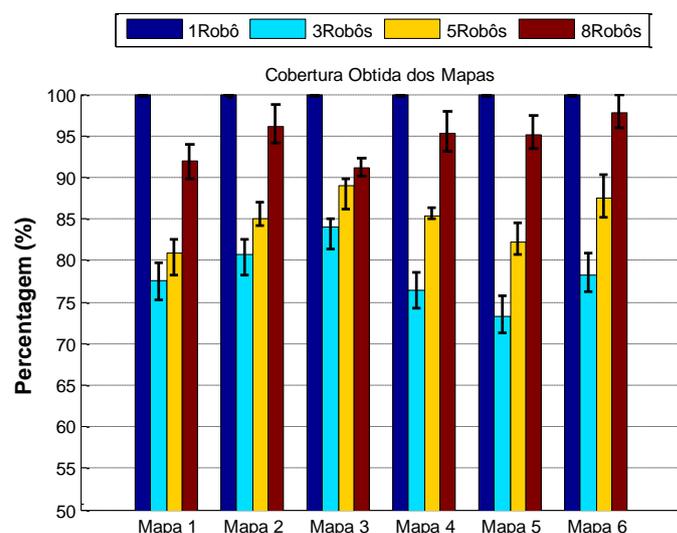


Figura 14 - Percentagens obtidas para 1,3,5 e 8 robôs com comunicação para os vários mapas.

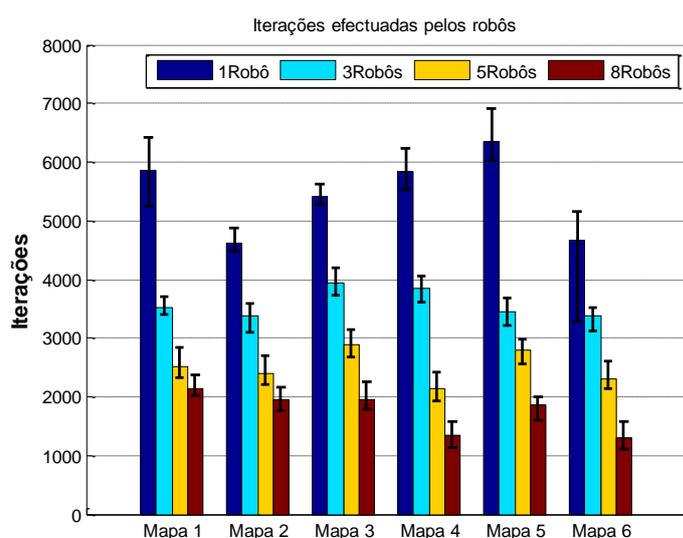


Figura 15 - Iterações obtidas para 1,3,5 e 8 robôs com comunicação para os vários mapas.

Nos resultados obtidos, a situação em que é utilizado somente um robô serve apenas como referência para avaliar a performance das restantes situações. De notar ainda que ocorreu uma quebra de percentagens de cobertura obtidas pelas equipas de robôs, o que seria de esperar uma vez que ao ser colocada a restrição, os movimentos dos robôs passam a ser limitados.

4.4. Discussão

Embora o algoritmo não tenha sido originalmente desenvolvido com o intuito da aplicação de equipas de robôs ou aspectos relativos à comunicação, estes factores foram implementados para aproximar a sua aplicabilidade ao mundo real.

Para não prejudicar a eficácia do algoritmo as limitações relativas à comunicação, foram apenas impostas no momento da escolha de uma nova célula, uma vez que se a restrição fosse aplicada numa fase inicial de pesquisa de células, cada robô deixaria as suas células por descobrir em detrimento da necessidade de se deslocar com outros robôs. Mesmo assim, ocorreu um decréscimo nos valores de percentagens de cobertura e conseqüentemente um aumento do número de iterações no caso em que são impostas as restrições de comunicação.

5. Algoritmo de Backtracking em espiral

5.1. Introdução

Num futuro próximo, um dos objectivos da robótica passará pela massificação da utilização de máquinas que colaboram com seres humanos em tarefas do dia-a-dia, quer em casa quer no trabalho. Para isto, é necessária a criação de plataformas robóticas de baixo custo e fácil usabilidade possibilitando a sua utilização por funcionários não qualificados. Entre as aplicações mais recorrentes deste tipo de plataformas, com necessidade de algoritmos de cobertura, constam aplicações como por exemplo: máquinas de limpeza autónomas, cortadores de relva, robôs agrícolas, etc.

Existem actualmente no mercado alguns produtos com as características descritas anteriormente, tais como o Roomba e Trilobite, no entanto o algoritmo usado por estes para calcular a trajectória é baseado no movimento *brownian*, que é considerado ineficiente quando comparado com o algoritmo de Backtracking em Espiral (*The Backtracking Spiral Algorithm -BSA*).

Com o intuito de reduzir o número de viragens realizadas durante o percurso, o movimento utilizado no algoritmo de Decomposição *Boustrophedon* é substituído no

BSA por um movimento em espiral, permitindo evitar o recurso a duas viragens consecutivas, baixando o número de rotações puras.

5.2. Descrição Algoritmo

O BSA [3] é baseado na execução de um caminho em espiral, em vez de utilizar movimentos *Boustrophedon*, e a sua simplicidade permite que o robô tenha uma boa performance com exigências mínimas a nível computacional. Os caminhos gerados podem ser facilmente executados por uma plataforma móvel diferencial, tornando o BSA adequado para a implementação em robôs de baixo custo. A metodologia do algoritmo permite também que não seja necessário tomar precauções relativamente à orientação inicial do robô, pois é assegurado que este encontra sempre um caminho desde que pelo menos uma das laterais ou a frente do robô esteja desocupada.

No algoritmo os pixéis podem apresentar vários estados tais como:

- Desconhecido
- Obstáculo virtual
- Obstáculo real
- BP (*Backtracking Point*)

Uma vez que neste algoritmo foi considerada como unidade base o pixel (em vez da célula), não foi necessário usar a versão alterada deste algoritmo que permite considerar células parcialmente ocupadas, pois o pixel apenas pode estar livre ou obstruído.

O algoritmo baseia-se na utilização de um caminho em espiral permitindo a cobertura de regiões simples (área que pode ser preenchida com um movimento em espiral) que por sua vez são interligadas usando um mecanismo de *backtracking*.

Um modelo do ambiente é construído à medida que o robô se move sendo a sua superfície modelada por uma grelha de ocupação onde inicialmente todas as células são marcadas como desconhecidas. À medida que o robô efectua a cobertura do mapa, as células dentro do seu raio de acção são numa primeira instância marcadas como obstáculos virtuais e as células no limiar do seu raio de detecção na direcção oposta à do obstáculo que se encontrem livres passam a ser consideradas BP's. Os obstáculos virtuais passam a ser marcados como obstáculos reais, sempre que é detectada a presença de uma obstrução pelos sensores acoplados ao robô.

O caminho é efectuado até que a espiral se feche dando assim origem a uma região simples. Após esta etapa o robô navega para um novo BP utilizando um mecanismo de *backtracking* que percorre um caminho de células já visitadas (células

marcadas como obstáculo virtual). Esta escolha pode ser baseada em diferentes critérios, mas foi utilizada a distância euclidiana entre o robô e os pontos BP's de forma a seleccionar o BP mais próximo do robô. Todo o processo é repetido para a obtenção de uma nova espiral.

No BSA, caminhos em espiral são compostos por anéis concêntricos que formam um caminho contínuo desde o limite da região até um ponto central final. Antes de começar um caminho em espiral, o robô é colocado perto de um obstáculo que esteja do seu lado referencial, denominado RLS (Reference Lateral Side), que indica a direcção relativa da localização dos obstáculos para servir de referência ao procedimento do caminho em espiral. Por sua vez o OLS (Opposite Lateral Side) é o oposto do RLS. Estas definições constituem o referencial previamente definido que permanece inalterado durante toda a execução. O seguinte pseudo-código representa as possíveis movimentações do robô usadas na construção do caminho em espiral:

```

RS1      IF (obstacles all around)
          THEN ending spiral point detected
RS2      IF (NOT obstacle in RLS)
          THEN turn to (RLS) and move forward
RS3      IF (obstacle in front)
          THEN turn to (OLS)
RS4      OTHERWISE move forward
  
```

O BSA termina quando não existirem mais BP's, o que significa que não existem mais caminhos alternativos e consequentemente toda a região acessível foi coberta. Um exemplo de cobertura efectuada pelo algoritmo e do respectivo trajecto é ilustrado na Figura 16.

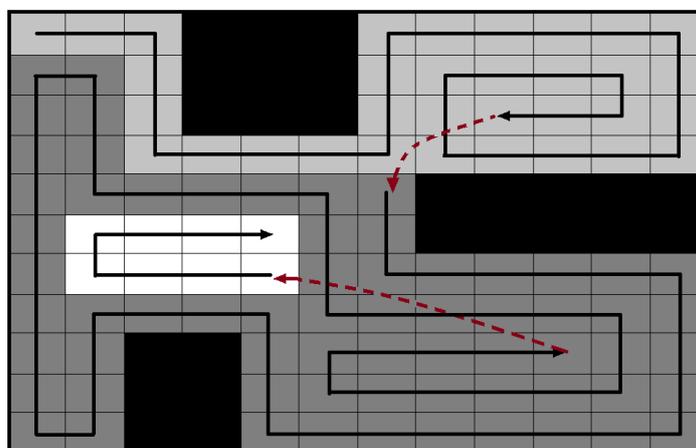


Figura 16 - Exemplo de cobertura realizado pelo algoritmo BSA.

5.1. Resultados Experimentais

À semelhança do algoritmo implementado no capítulo anterior, este foi desenvolvido para a aplicação de apenas um agente de acordo com o artigo [3].

Os gráficos seguintes apresentam os resultados obtidos com dez testes com um robô para os seis mapas ilustrados na Figura 7, tanto a nível de percentagem de mapa descoberto como em termos de iterações realizadas.

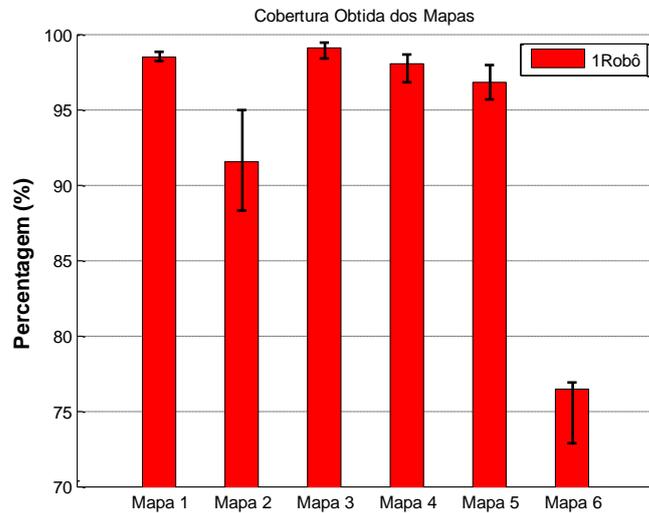


Figura 17 - Percentagens obtidas com 1 robô para os vários mapas.

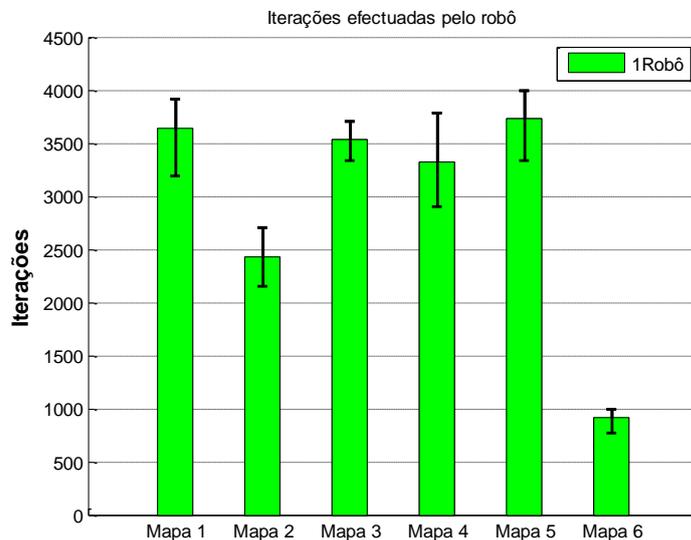


Figura 18 - Iterações obtidas com 1 robô para os vários mapas.

Dado o término da análise do algoritmo para um robô, foram efectuadas alterações no algoritmo para permitir a implementado para múltiplos robôs e foi também adicionada a capacidade dos robôs evitarem colisões entre eles e não ocorrerem disputas entre robôs nos momentos de selecção de BP's.

Os seguintes gráficos apresentam a média dos testes realizados com um e equipas de robôs de três, cinco e oito elementos seja ao nível de percentagem ou de iterações.

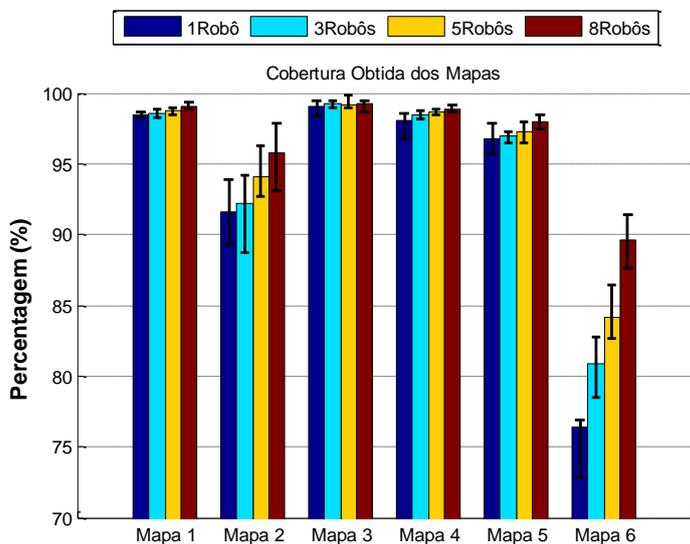


Figura 19 - Percentagens obtidas para 1,3,5 e 8 robôs para os vários mapas.

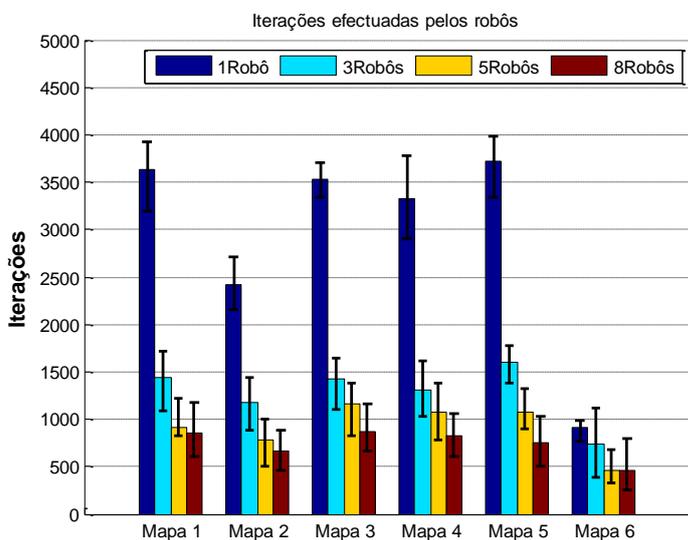


Figura 20 - Iterações obtidas para 1,3,5 e 8 robôs para os vários mapas.

Os resultados obtidos pelo presente algoritmo apresentaram valores de percentagens acima de 90%, á excepção do sexto mapa, que por ser um mapa com uma maior densidade de obstáculos leva a que o algoritmo não obtenha resultados próximos dos obtidos para os outros mapas.

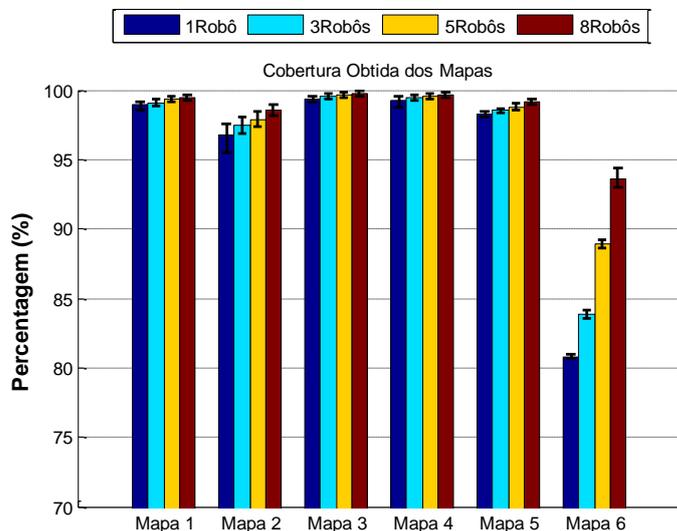


Figura 21 - Percentagens obtidas para 1,3,5 e 8 robôs para os vários mapas.

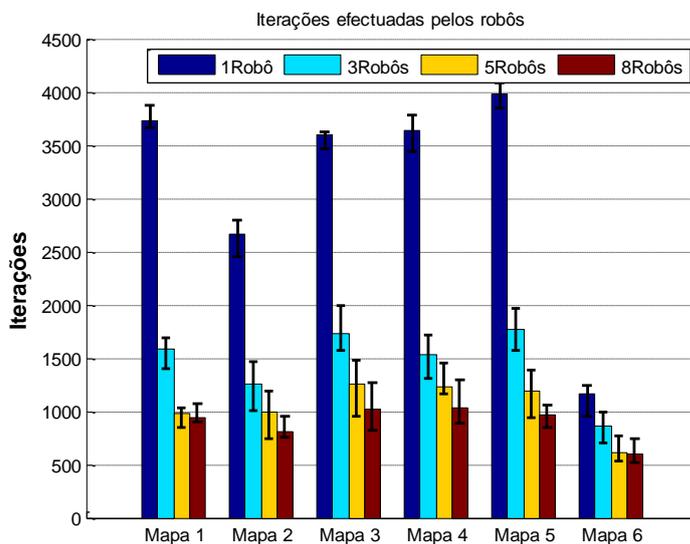


Figura 22 - Iterações obtidas para 1,3,5 e 8 robôs para os vários mapas.

Tal como no algoritmo de *Boustrophedon*, também nesta implementação ocorreram situações em que após a pesquisa terminar se verificou a existência de áreas superiores a 25 pixels ainda não cobertas. A mesma solução usada anteriormente foi aplicada neste algoritmo de modo a melhorar os níveis de percentagem de cobertura o que levou a um conseqüente aumento do número de interações realizadas.

O passo seguinte consistiu na aplicação das restrições de comunicação em tudo semelhantes às aplicadas no capítulo anterior, que apenas diferem no momento em que são aplicadas uma vez que são usadas quando o robot selecciona um novo BP pois se a restrição fosse usada de forma continua esta levaria a um cancelamento do processo de espiral cada vez que um robô se encontrasse fora do alcance de comunicação, o que terminaria com o conceito geral do algoritmo.

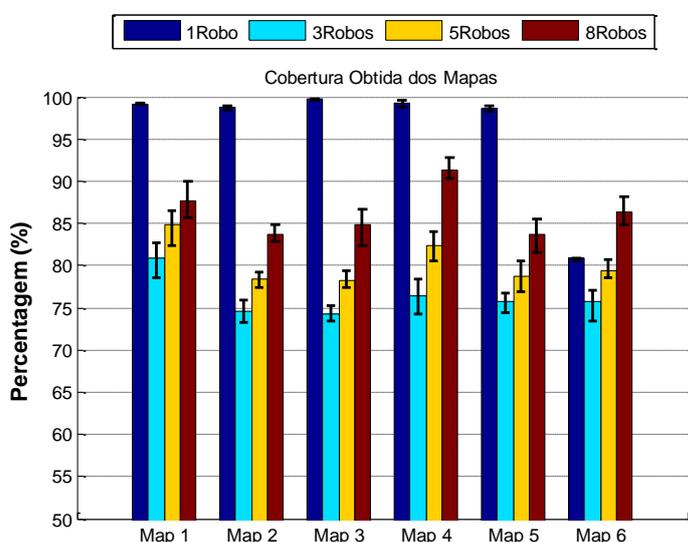


Figura 23 - Percentagens obtidas para 1,3,5 e 8 robôs com comunicação para os vários mapas.

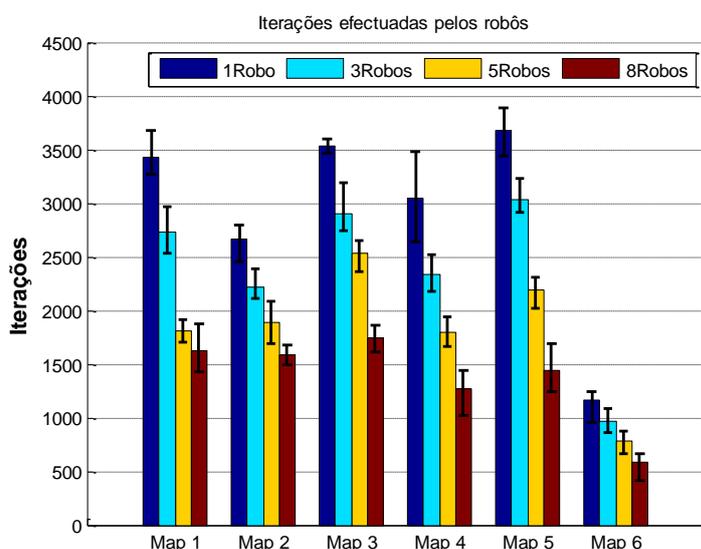


Figura 24 - Iterações obtidas para 1,3,5 e 8 robôs com comunicação para os vários mapas.

Os gráficos anteriores apresentam os resultados obtidos pelas várias equipas de robôs e apresentam também os resultados obtidos para um único robô que servem de referência. De notar ainda que as percentagens de cobertura obtidas pelas equipas de robôs sofreram uma quebra semelhante à ocorrida no algoritmo anterior, visto que a restrição de comunicação limita, em parte, a escolha livre de movimentos.

5.2. Discussão

O algoritmo descrito, à semelhança do anterior, não foi concebido com a ideia de serem colocadas restrições de comunicação, pelo que a restrição imposta é válida apenas quando cada robô termina sua espiral e necessita de procurar um novo ponto BP de modo a iniciar uma nova etapa do processo de cobertura. No que diz respeito aos resultados obtidos com as simulações do algoritmo para as várias equipas de robôs, à exceção dos resultados obtidos com a restrição de comunicação, os resultados foram satisfatórios sendo obtidos resultados superiores comparativamente aos ilustrados no artigo [3].

6. Algoritmo de exploração multi-robôs coordenados

6.1. Introdução

A exploração de um ambiente desconhecido torna-se ainda mais complexa ao usar equipas de robôs em vez de um único robô, pois com o aumento do número de agentes adiciona-se também redundância. No entanto espera-se que com vários robôs a cobertura se torne mais tolerante a falhas que possam decorrer de avarias nas plataformas. Vários trabalhos fazem referência à eficácia dos algoritmos propostos para fazer cobertura usando apenas um robô no entanto o algoritmo descrito, proposto por *Wolfram Burgard*, enuncia uma metodologia orientada à coordenação de vários agentes na exploração de um ambiente de forma eficaz e eficiente.

6.2. Descrição Algoritmo

O algoritmo usa mapas de ocupação para representar o meio ambiente, onde cada célula do mapa contém um valor numérico que corresponde à probabilidade da área ser coberta por um obstáculo. Ao explorar um ambiente desconhecido, estamos especialmente interessados nas células vizinhas de uma célula que ainda não foi descoberta, também conhecidas por células fronteira.

Ao direccionar o robô para uma célula fronteira é esperado que adquira informação sobre a área por explorar à medida que se desloca para a célula em causa. O facto de o mapa conter várias áreas por explorar gera um potencial problema na atribuição das células fronteira ao robô, o que no caso de múltiplos robôs leva a que estes se possam deslocar para a mesma célula em simultâneo. Para superar esta lacuna foi usada uma aproximação teórica de decisão onde se considera o custo de chegar á célula alvo e a utilidade dessa célula para cada robô. O custo da célula é proporcional à distância entre o robô e a célula, enquanto que a utilidade da célula fronteira para o robô depende do número de robôs que se estão a movimentar para a célula ou para uma célula nas proximidades desta.

Para determinar o custo de atingir uma célula fronteira, utiliza-se o caminho óptimo que une a posição actual do robô a todas as células fronteira. Este cálculo é efectuado com base numa variante determinística do valor de iteração onde o custo mínimo pode ser definido pelo seguinte método:

1) Inicialização. A célula que contém o robô é inicializada a 0 e as restantes a ∞ :

$$V_{xy} \leftarrow \begin{cases} 0, & \text{se } (x,y) \text{ for a posição do robot} \\ \infty, & \text{caso contrario} \end{cases}$$

2) Ciclo de Actualização. Para todas as células (x,y) fazer:

$$V_{xy} \leftarrow \min \left\{ V_{x+\Delta x, y+\Delta y} + \sqrt{\Delta x^2 + \Delta y^2} \cdot P(\text{occ}_x + \Delta x, y + \Delta y) \mid \Delta x, \Delta y \in \{-1, 0, 1\} \wedge P(\text{occ}_x + \Delta x, y + \Delta y) \in \{0, \text{occ}_{\max}\} \right\}$$

Onde occ_{\max} representa a probabilidade máxima de ocupação de uma célula que esteja ao alcance do robô. Esta técnica actualiza o valor de todas as células consoante o valor dos vizinhos com menor custo a que é adicionada a probabilidade de uma célula (x,y) estar ocupada, $P(\text{occ}_{xy})$, multiplicada pela distância a essa mesma célula.

A regra de actualização é efectuada até convergir. A convergência do algoritmo é garantida desde que o custo de alcançar uma determinada célula não seja negativo e o mapa seja limitado. O resultado da função V_{xy} pode ser usado para derivar eficientemente o caminho de mínimo custo desde a posição actual do robô até uma posição final arbitrária (x,y), tal operação é efectuada percorrendo V_{xy} elemento a elemento de forma descendente.

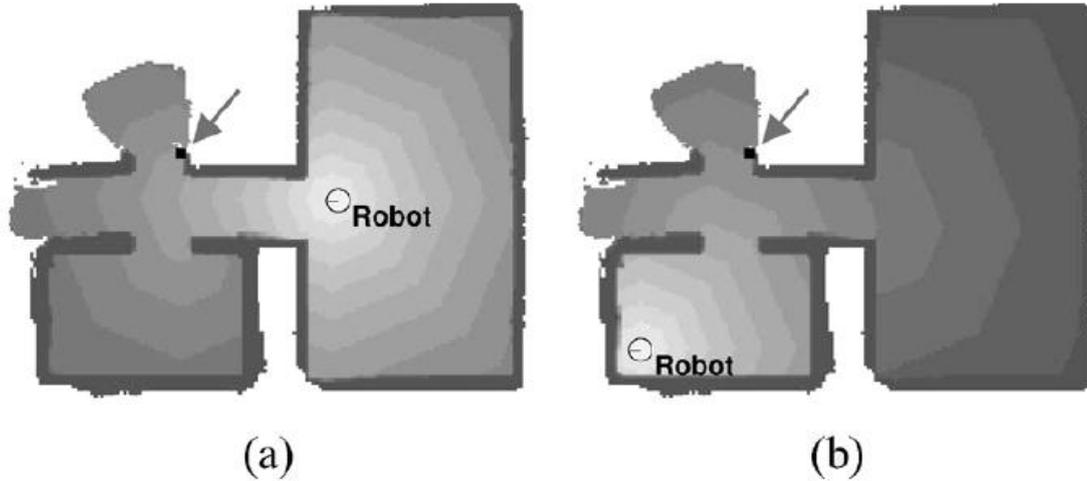


Figura 25 - Valores típicos das funções, obtidos para duas posições diferentes do robô.

Na Figura 25 o ponto a preto (que corresponde a um célula fronteira do robô) indica o ponto objectivo na área desconhecida com o caminho de menor custo. Caso não haja coordenação entre robôs, estes irão dirigir-se para o ponto em causa em simultâneo o que não seria uma solução óptima.

Após calcular o custo para este ponto é necessário estimar a utilidade das células vizinhas, para isso assume-se que todas as células fronteira t inicialmente têm uma utilidade U_t e sempre que um ponto alvo t' é seleccionado por um robô, a utilidade das células fronteira adjacente a uma distância d de t' é diminuída de acordo com a probabilidade $P(d)$ com que os sensores do robô cobrirão as células à distância d . Assim, qualquer célula t á distancia d do alvo t' irá ser coberto com uma probabilidade $P(d)$ quando o robô alcançar t' . A utilidade $U(t_n|t_1, \dots, t_{n-1})$ de uma célula t_n onde as células t_1, \dots, t_{n-1} foram atribuídas aos robôs $1, \dots, n-1$ é dada por,

$$U(t_n|t_1, \dots, t_{n-1}) = Ut_n - \sum_{i=1}^{n-1} P(\|t_n - t_i\|) \quad (1)$$

Segundo a equação anterior, quanto maior for o número de robôs a deslocarem-se para uma mesma zona onde estejam próximos vários pontos t_n , menor será a utilidade calculada para as células dessa área. Se existirem obstáculos entre duas células fronteira t e t' , a probabilidade $P(\|t - t'\|)$ é colocada a zero.

Uma vez terminada esta fase é necessário efectuar a selecção do ponto alvo. No artigo [13] o autor descreve 2 soluções para a atribuição do ponto alvo. Na implementação do algoritmo foi utilizado a segunda solução pois segundo o autor esta

apresenta melhor desempenho na cobertura. De seguida é descrita a metodologia da solução implementada.

Solução 2: Ponto alvo determinado com recurso a permutações de robôs

1. Determinar o conjunto de células fronteira.
2. Calcular, para cada robô i o custo V_t^i de atingir cada célula fronteira.
3. Determinar as localizações alvo t_1, \dots, t_n para os robôs $i=1, \dots, n$ que maximizem a seguinte função:

$$\sum_{i=1}^{n-1} [U(t_i | t_1, \dots, t_{i-1}, t_{i+1}, \dots, t_n) - \beta \cdot (V_{t_i}^i)^2]$$

A solução acima descrita determina em cada ronda a atribuição óptima dos robôs a pontos alvo de forma eficiente, no entanto, para equipas com um número elevado de robôs, o algoritmo tende a não ser aplicável pois, a aproximação do algoritmo tende para $\left(\frac{T!}{(T-n)!}\right)$, onde T representa o número de possíveis pontos alvo e n o número de robôs. Este facto leva a que o número de pontos alvo cresça exponencialmente em função do número robôs.

Importa referir que o algoritmo descrito foi implementado inicialmente pelo Mestre Filipe Gonçalves e foi por mim modificado de forma a considerar a restrição de comunicação no algoritmo. Devido a limitações temporais o algoritmo apenas foi estudado na medida necessária para que fosse possível introduzir as alterações relevantes.

6.3. Resultados Experimentais

Numa primeira fase o algoritmo foi testado, com e sem restrições de comunicação, de forma a avaliar o impacto das mesmas nos parâmetros medidos. Ao contrário dos capítulos anteriores, este algoritmo foi apenas testado para um máximo de 5 robôs pois a simulação usando 8 robôs não seria concluída em tempo útil devido ao facto de um número elevado de robôs conduzir a um aumento exponencial de pontos alvo.

A implementação original do algoritmo contém duas metodologias que diferem apenas na métrica utilizada para realizar a escolha das células alvo. A formulação das métricas é definida da seguinte forma:

- **Métrica 1:** $U(t_i|t_1, \dots, t_{i-1}, t_{i+1}, \dots, t_n) - \beta \cdot (V_{t_i}^i)^2$
- **Métrica 2:** $\frac{U(t_i|t_1, \dots, t_{i-1}, t_{i+1}, \dots, t_n)}{\beta \cdot (V_{t_i}^i)^2}$

De seguida são apresentados os resultados para ambas as métricas sem restrições de comunicação.

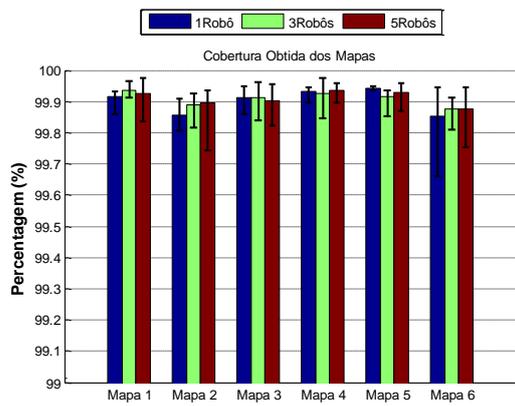


Figura 26 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1.

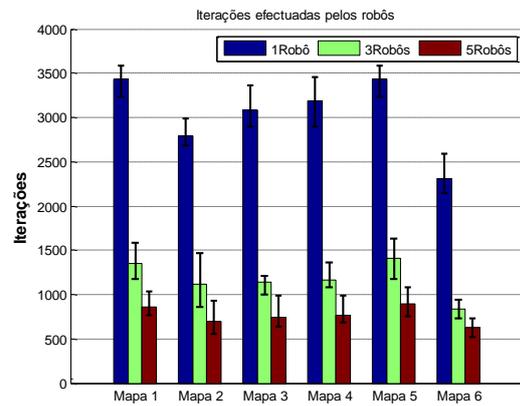


Figura 27 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1.

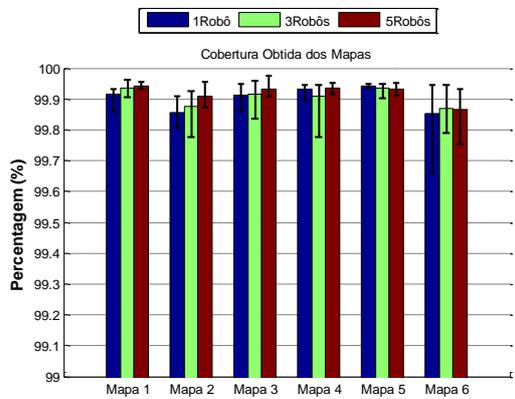


Figura 28 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas na métrica 2.

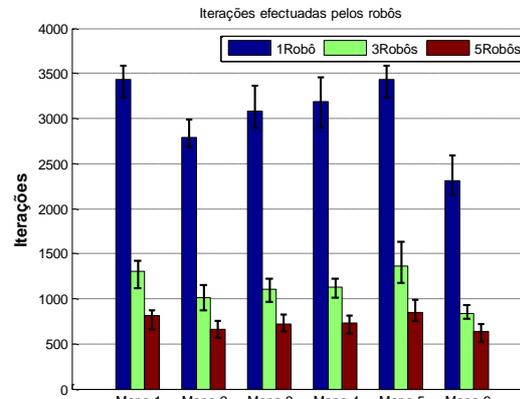


Figura 29 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas na métrica 2.

Tanto os resultados obtidos pela aplicação do algoritmo utilizando a métrica 1 e 2, demonstram uma elevada capacidade de cobertura do mapa, de forma tal que o valor médio para a cobertura dos mapas simulados está acima de 99.8%, representando assim uma situação muito próxima de cobertura total.

Pode-se concluir que para todas as situações simuladas o pior desempenho verifica-se com a aplicação de um único robô, sendo necessário um número total de iterações muito superior comparativamente a qualquer uma das outras opções simuladas (equipa de três e cinco robôs). De notar ainda, que o último mapa apresenta um número de iterações inferior a todos os outros, devido a conter uma maior percentagem de área ocupada por obstáculos.

Após serem implementadas as restrições relativamente à comunicação, os seis mapas foram de novo usados para simular novamente a configuração habitual. De seguida são apresentados os resultados para ambas as métricas com restrições de comunicação.

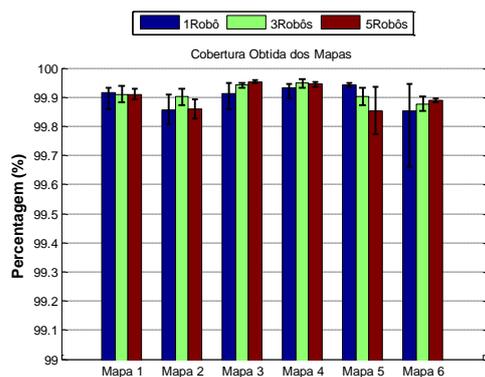


Figura 30 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1 com restrições de comunicação.

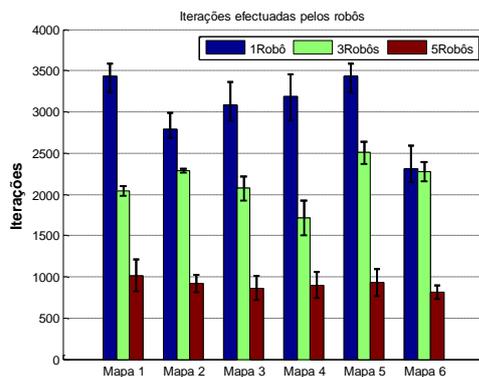


Figura 31 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1 com restrições de comunicação.

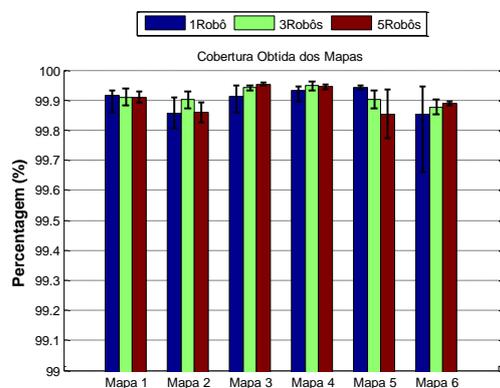


Figura 32 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas na métrica 2 com restrições de comunicação.

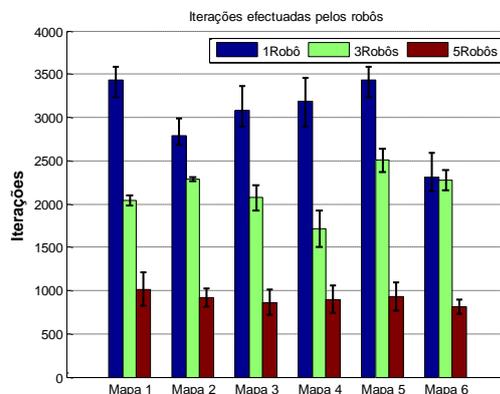


Figura 33 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas na métrica 2 com restrições de comunicação.

Os resultados obtidos foram semelhantes aos obtidos sem as restrições de comunicação. Importa salientar que ao contrário dos algoritmos anteriores as restrições não provocaram um aumento tão significativo do número de iterações.

6.4. Discussão

Comparativamente aos resultados obtidos nos algoritmos anteriores, verifica-se um aumento considerável da capacidade de cobertura do mapa e até mesmo uma redução do número de iterações. Uma das características negativas do algoritmo reside no facto deste não poder ser estendido a equipas com um maior número de robôs devido ao facto do número de pontos alvo crescer exponencialmente em relação ao número de robôs, ainda assim, para pequenas equipas de robôs este algoritmo é bastante eficaz.

7. Algoritmo de exploração e mapeamento de multi-robôs coordenados

7.1. Introdução

A criação de mapas do ambiente a explorar é um grande desafio em robótica móvel, pois requer uma boa estratégia de exploração, em particular, os robôs necessitam de ter conhecimento das áreas que vale a pena explorar e como as distribuir para que possam ser cobertas.

Grande parte do trabalho desenvolvido na criação de mapas é atribuído apenas a um robô, no entanto o algoritmo proposto visa a aplicação de múltiplos robôs, pois existem várias vantagens, entre elas o facto de com vários robôs ser possível terminar uma dada tarefa mais rapidamente e uma outra vantagem prende-se com o facto de ser possível criar mapas mais precisos devido à troca de informações entre robôs. Este algoritmo apresenta técnicas para coordenar múltiplos robôs, que visam resolver dois grandes problemas, nomeadamente, a criação de um mapa global com informação baseada nos sensores de cada robô e a tomada de decisão por parte dos robôs de forma a criar o mapa mais eficientemente.

7.2. Descrição Algoritmo

Cada robô tem o seu mapa local e um módulo central recebe todos os mapas locais provenientes dos robôs e combina-os gerando um mapa global. No início, cada robô cria o seu próprio mapa com base nos seus sensores, de seguida, incrementalmente determina três parâmetros: a probabilidade máxima para a sua própria posição, a probabilidade máxima para o mapa (localização de objectos ao seu redor) e a densidade relativamente à “verdadeira” localização.

Para criar o mapa global, o módulo central integra a informação obtida pelos robôs em tempo real e, com o auxílio de um modelo de gradiente descendente, minimiza o erro entre as leituras dos vários robôs. Uma vez que é assumido que se sabe as posições iniciais de cada robô é possível saber com precisão a localização de cada um deles relativamente aos outros.

O objectivo na coordenação da exploração com vários robôs é maximizar o ganho de informação (conhecimento do mapa) ao longo do tempo. Para isso propõe-se que cada robô construa uma “proposta” que utiliza as utilidades estimadas para o robô se deslocar para várias células fronteira (célula fronteira denomina-se por cada célula “limpa” adjacente a uma célula “desconhecida”) e que categoriza as células em três

tipos, “obstáculo” (probabilidade de ocupação acima de um determinado limite p_0), “limpa” (probabilidade de ocupação abaixo de um determinado limite p_c), e “desconhecida” (célula não analisada ou com probabilidade entre p_0 e p_c). As propostas são enviadas para um módulo central que atribui tarefas a cada robô com base em todas as ofertas recebidas e tendo em conta eventuais sobreposições na cobertura. Enquanto um robô possa “preferir” visitar uma determinada localização, o módulo pode atribuir essa mesma localização a outro robô se for esperado que o outro robô ganhe mais com a mesma informação. Os robôs enviam as suas propostas quando os seus mapas são actualizados. A exploração termina quando não existe mais informação útil a ganhar.

Para estimar o custo de visitar uma célula fronteira é calculado um caminho óptimo com base na menor distância e num movimento determinístico desde a célula onde se encontra o robô até à célula em questão.

No que diz respeito à atribuição de tarefas, após os robôs enviarem as propostas e o módulo central as analisar, o módulo central encontra a proposta com a maior utilidade (ganho menos o custo) e atribui essa tarefa ao robô que fez a proposta, de seguida “desconta” as propostas restantes dos robôs baseado nas tarefas assinaladas e escolhe a proposta com maior utilidade, este processo é continuado até todos os robôs terem tarefas atribuídas, ou não haver mais tarefas para atribuir.

A chave deste algoritmo é os “descontos”, sem eles os robôs comportar-se-iam de forma descoordenada. Os descontos são obtidos com base na informação do ganho, em específico, é estimada a percentagem de sobreposição entre as regiões de ganho de informação pela proximidade dos rectângulos de sobreposição (Figura 34) e decrementando o ganho de informação esperado por essa percentagem:

$$d_j = \frac{Area(IGR_j \cap (\cup IGR_i))}{Area(IGR_j)}, i \in R$$

$$u_j = (1 - d_j) \times i_j - c_j$$

Onde IGR_j é a aproximação rectangular da região de ganho de informação de uma célula fronteira j , o IGR_i são as regiões de ganho de informação rectangulares para os robôs R atribuídos, e i_j e c_j representam o ganho de informação esperado e o custo do caminho de ir para a célula j .

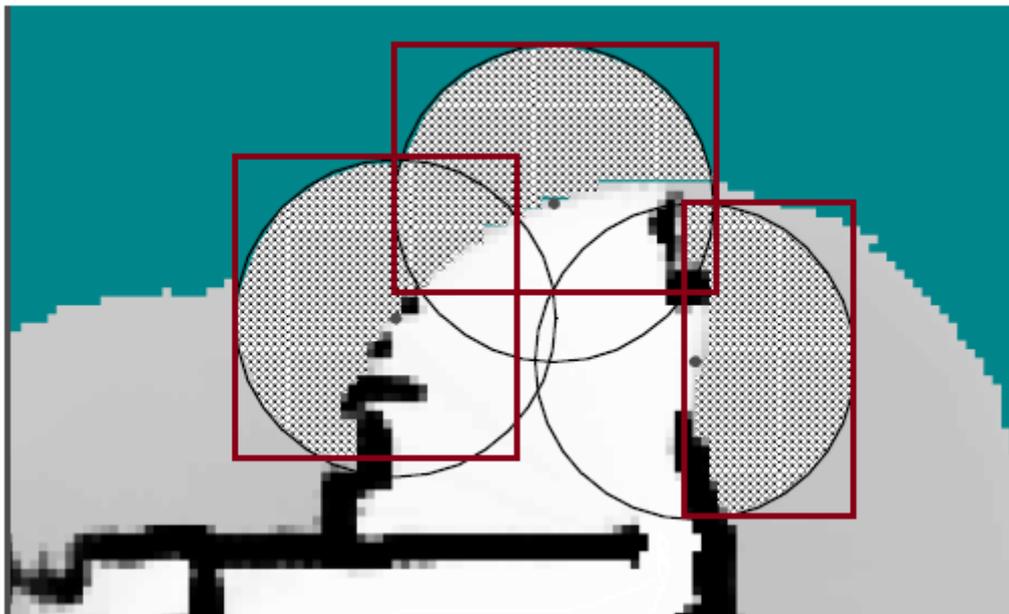


Figura 34 - Ganho de Informação esperado.

Importa referir que o algoritmo descrito foi implementado inicialmente pelo Mestre Filipe Gonçalves e foi por mim modificado de forma a considerar a restrição de comunicação no algoritmo. Devido a limitações temporais o algoritmo apenas foi estudado na medida necessária para que fosse possível introduzir as alterações relevantes.

7.3. Resultados Experimentais

À semelhança do algoritmo do capítulo anterior, realizou-se a simulação do algoritmo sem qualquer alteração, de forma a obter dados para posterior comparação com os dados obtidos após a implementação e incorporação da restrição de comunicação no presente algoritmo. Em todos os testes do algoritmo foram utilizadas as mesmas equipas de robôs que no algoritmo anterior, no entanto apesar de já ser possível a simulação de equipas com um maior número de robôs, esta ainda requer um tempo de processamento elevado, optando-se por não realizar este ponto.

A implementação original do algoritmo á semelhança do anterior contém duas versões do mesmo, onde apenas é alterada a métrica de escolha das células alvo, sendo apresentados os resultados obtidos para cada uma das métricas.

- **Métrica 1:** $(1 - d_j) \times i_j - c_j$
- **Métrica 2:** $\frac{(1-d_j) \times i_j}{-c_j}$

Na figura seguinte são apresentadas as percentagens obtidas por cada equipa de robôs para cada um dos mapas.

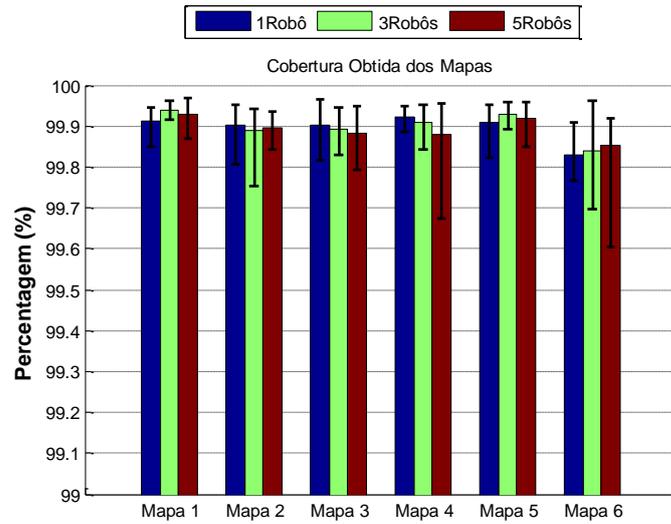


Figura 35 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1.

Os resultados obtidos pelo presente algoritmo demonstram uma enorme capacidade de cobertura do mapa perto da sua totalidade, onde em média as percentagens situam-se acima de 99.8% para todos os mapas.

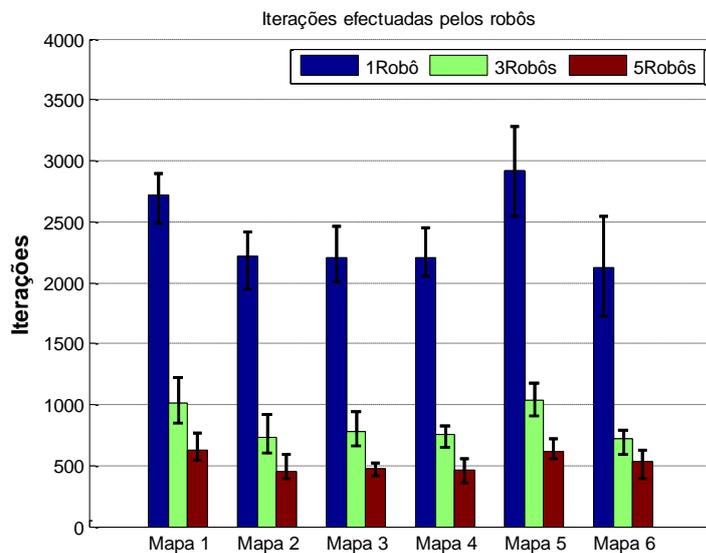


Figura 36 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas na métrica 1.

Na figura anterior verifica-se que para o caso de apenas um robô é o que necessita de mais iterações para concretizar a cobertura total em todos os mapas.

De forma semelhante, para a métrica 2, simularam-se vários testes para as mesmas equipas de robôs utilizadas anteriormente de forma a verificar as percentagens e iterações que realizariam comparativamente com a métrica anterior.

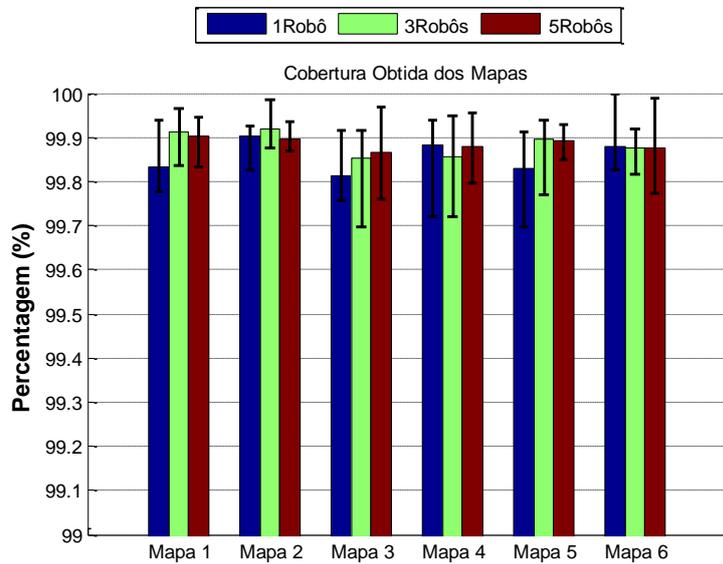


Figura 37 - Percentagens obtidas para 1,3 e 5 robôs para os vários mapas para a métrica 2.

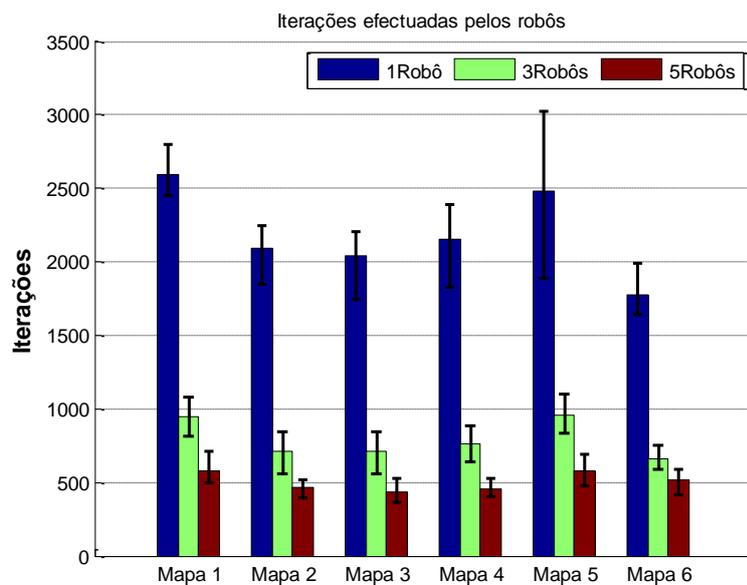


Figura 38 - Iterações obtidas para 1,3 e 5 robôs para os vários mapas para a métrica 2.

No que diz respeito às iterações realizadas pelos robôs denotou-se uma ligeira diminuição do número médio de iterações na maioria das situações para equipas de 3 e 5 robôs.

Após a obtenção dos resultados da simulação do algoritmo, foi implementada a restrição de comunicação entre robôs. Esta restrição foi incorporada no momento da selecção das células alvo por parte dos robôs, onde seriam seleccionadas apenas as células alvo dentro da distância de comunicação, definida de 40 pixels.

Uma vez concluída a implementação da restrição de comunicação entre robôs foram realizadas simulações para as métricas de escolha das células alvo.

Após a implementação no algoritmo da restrição de comunicação foram realizados dez testes para cada equipa de robôs em cada mapa e para cada uma das métricas.

- **Métrica 1:** $(1 - d_j) \times i_j - c_j$

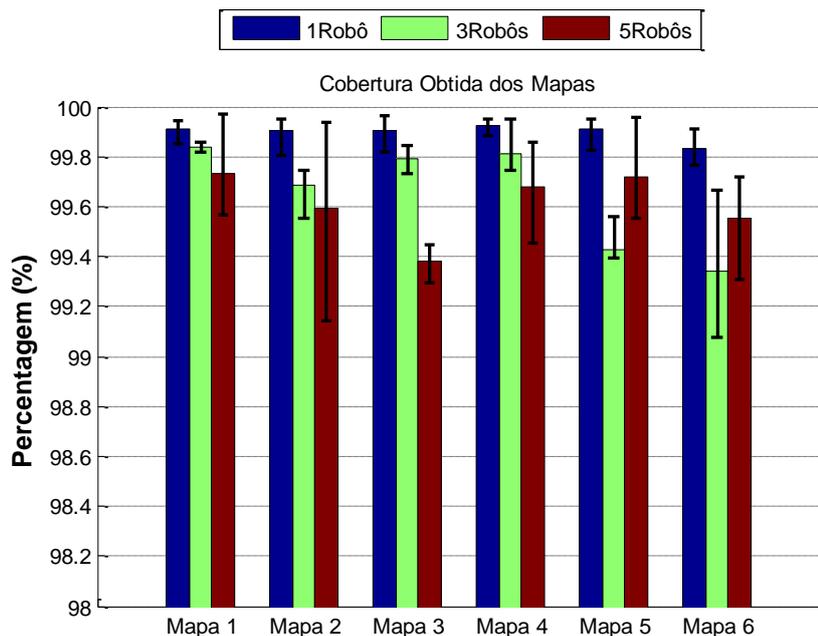


Figura 39 - Percentagens obtidas para 1,3 e 5 robôs com restrições de comunicação para os vários mapas na métrica 1.

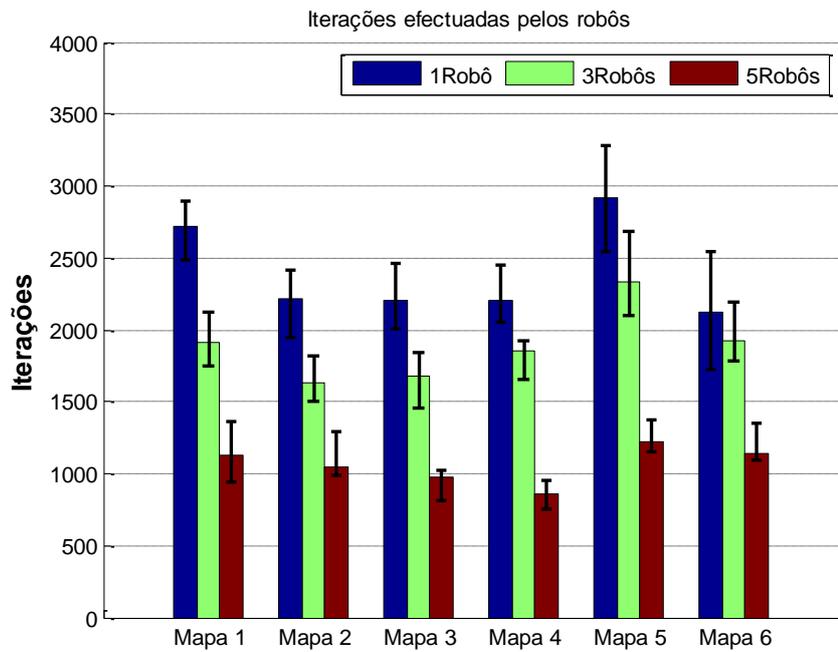


Figura 40 - Iterações obtidas para 1,3 e 5 robôs com restrições de comunicação para os vários mapas na métrica 1.

Comparativamente com o mesmo algoritmo sem a restrição de comunicação ao ser colocada a restrição ocorre uma ligeira diminuição das médias percentuais em várias situações. Quanto ao número de iterações verificou-se um aumento, comparativamente com o caso sem a restrição de comunicação para a mesma métrica.

- **Métrica 2:** $\frac{(1-d_j) \times i_j}{-c_j}$

Para a presente métrica, simularam-se vários testes para as mesmas equipas de robôs utilizadas anteriormente de forma a verificar as suas percentagens e iterações.

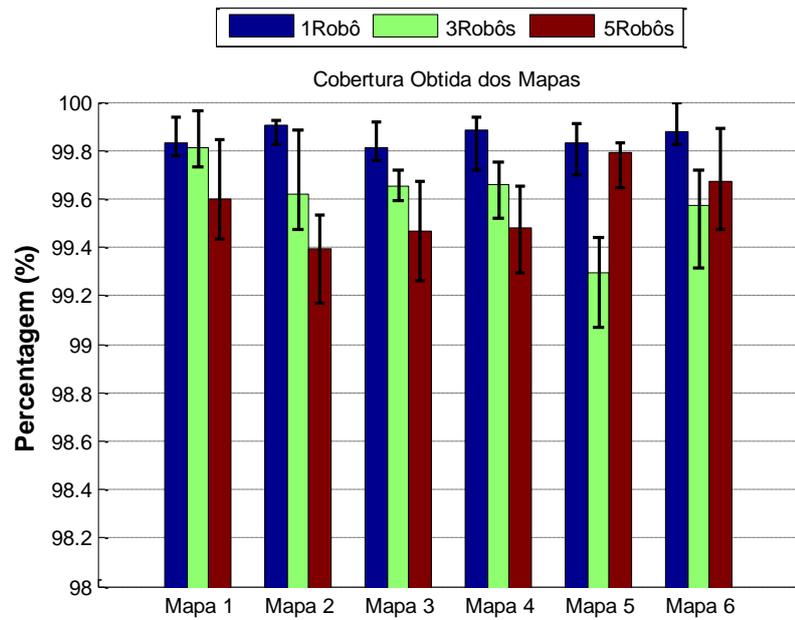


Figura 41 - Percentagens obtidas para 1,3 e 5 robôs com restrições de comunicação para os vários mapas na métrica 2.

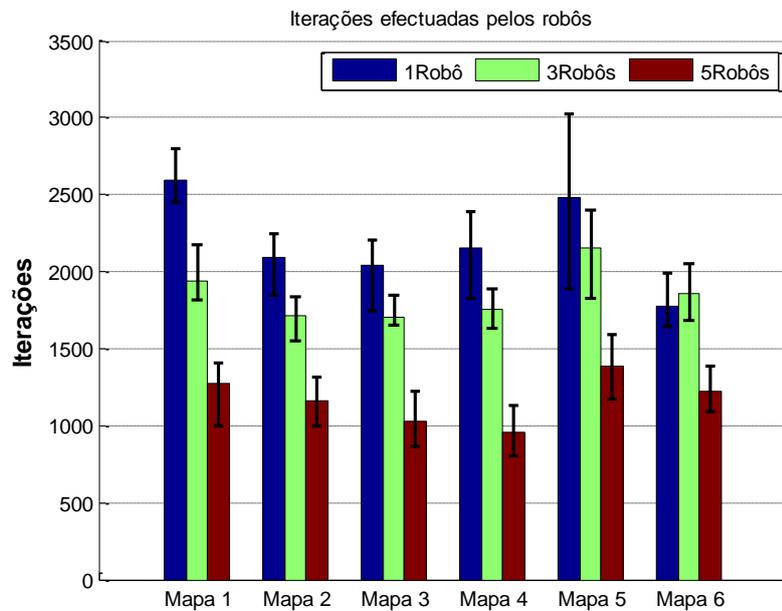


Figura 42 - Iterações obtidas para 1,3 e 5 robôs com restrições de comunicação para os vários mapas na métrica 2.

De forma semelhante ao ocorrido para a métrica anterior ocorre uma ligeira quebra nas percentagens obtidas pelos robôs em grande parte das situações e o número médio das iterações aumentou.

7.4. Discussão

Os resultados obtidos da implementação da restrição de comunicação são bastante bons, independentemente de se ter assistido a um aumento do número médio de iterações. Comparativamente com os resultados obtidos nos dois primeiros algoritmos descritos, verifica-se um aumento considerável da capacidade de cobertura do mapa e até mesmo uma redução do número de iterações. No entanto comparando com o algoritmo do capítulo anterior as diferenças entre eles já não são tão notórias.

8. Discussão e Conclusão

Uma vez terminadas as simulações de todos os algoritmos, foram incorporados os resultados obtidos de todos os algoritmos sem e com a restrição de comunicação nas tabelas seguintes para uma melhor comparação de desempenho entre eles.

Tabela 1 - Resultados obtidos sem restrição de comunicação

	1 Robô						3 Robôs						5 Robôs					
	M1	M2	M3	M4	M5	M6	M1	M2	M3	M4	M5	M6	M1	M2	M3	M4	M5	M6
Perçentagens	98,21	97,84	98,11	97,96	98,36	98,99	98,16	97,98	98,16	98,22	98,50	98,41	98,44	97,97	98,19	97,99	98,49	98,25
Boust V1	99,87	99,95	99,87	99,88	99,88	99,85	99,92	99,96	99,94	99,89	99,92	99,90	99,92	99,96	99,94	99,91	99,95	99,89
Boust V2	98,44	91,56	99,06	98,01	96,78	79,41	98,39	92,17	98,28	97,41	95,49	84,88	96,12	86,06	97,97	98,07	90,34	84,11
BSA V1	99,12	98,77	99,67	99,26	98,62	80,82	97,03	89,88	96,92	94,03	93,14	80,85	92,49	85,12	93,60	94,53	87,84	84,93
BSA V2	99,92	99,85	99,91	99,93	99,94	99,85	99,93	99,88	99,91	99,92	99,91	99,87	99,92	99,89	99,90	99,93	99,93	99,87
CMRE V1	99,92	99,85	99,91	99,93	99,94	99,85	99,93	99,87	99,92	99,91	99,93	99,86	99,94	99,90	99,93	99,93	99,93	99,86
CMRE V2	99,91	99,90	99,90	99,92	99,91	99,83	99,94	99,88	99,89	99,90	99,92	99,83	99,92	99,89	99,88	99,88	99,92	99,85
CMREM V1	99,83	99,90	99,81	99,88	99,83	99,88	99,91	99,91	99,85	99,85	99,90	99,87	99,90	99,89	99,86	99,87	99,89	99,87
CMREM V2	493,1	332,4	457,2	485,2	494,7	309,9	181,1	128,4	170,9	180,0	192,8	137,5	120,9	87,3	124,2	117,2	12,94	10,58
Boust V1	586,1	462,3	542,1	583,8	635,7	466,2	237,2	196,1	216,4	238,2	282,4	223,8	171,7	141,9	163,3	158,9	20,16	17,40
Boust V2	363,8	24,26	353,5	332,2	372,7	91,9	143,6	117,2	142,7	131,4	160,3	73,4	9,20	78,4	115,9	107,3	10,77	4,63
BSA V1	342,6	26,66	353,4	304,0	368,2	116,6	139,1	125,7	172,5	153,9	176,4	55,9	9,10	69,3	125,2	142,9	10,96	4,17
BSA V2	343,7	27,93	308,0	319,7	343,8	230,8	135,4	111,2	113,9	115,8	140,6	83,7	8,61	69,9	74,4	7,62	8,93	6,20
CMRE V1	343,7	27,93	308,0	319,7	343,8	230,8	130,1	101,4	110,0	113,1	136,1	83,7	8,07	65,9	72,4	7,26	8,50	6,36
CMRE V2	271,4	22,15	220,1	220,8	291,6	211,8	100,8	73,2	78,1	75,7	103,6	72,2	6,26	45,2	47,4	4,56	6,18	5,36
CMREM V1	258,9	20,95	203,9	215,6	248,0	177,3	94,2	71,1	70,6	75,8	95,8	65,9	5,78	4,63	4,32	4,58	5,81	5,20
CMREM V2																		

Tabela 2 - Resultados obtidos com restrição de comunicação

	3 Robôs						5 Robôs					
	M1	M2	M3	M4	M5	M6	M1	M2	M3	M4	M5	M6
Percentagens												
Boust	77,56	80,70	83,98	76,35	73,28	78,21	80,80	85,01	88,99	85,31	82,16	87,52
BSA	80,93	74,57	74,17	76,40	75,76	75,66	84,78	78,32	78,24	82,34	78,71	79,35
CMIRE V1	99,91	99,90	99,94	99,92	99,91	99,87	99,91	99,86	99,95	99,93	99,85	99,88
CMIRE V2	99,93	99,87	99,92	99,91	99,93	99,86	99,94	99,90	99,93	99,93	99,93	99,86
CMIREM V1	99,84	99,68	99,79	99,80	99,42	99,33	99,73	99,59	99,38	99,68	99,72	99,55
CMIREM V2	99,81	99,61	99,65	99,65	99,29	99,57	99,60	99,39	99,46	99,47	99,79	99,67
Iterações												
Boust	3514	3369	3904	3858	3,444	3381	2516	2402	2899	2144	2789	2309
BSA	2726	2221	2900	2333	3035	966	1810	1885	2530	1801	2185	784
CMIRE V1	2043	2287	2071	1712	2503	2275	1068	918	865	898	930	809
CMIRE V2	1913	2163	2106	1830	2360	2162	1016	1012	724	726	850	736
CMIREM V1	1908	1635	1681	1857	2336	1922	1126	1052	974	856	1218	1136
CMIREM V2	1942	1711	1706	758	2158	1859	1278	1163	1032	958	1381	1220

Na tabela 1 os algoritmos são referidos através dos seus acrónimos onde:

- Boust - Algoritmo de Boustrophedon com restrições de comunicação
- BSA - BSA com restrições de comunicação
- CMRE V1 - Algoritmo de exploração multi-robôs coordenados utilizando a 1ª métrica com restrições de comunicação.
- CMRE V2 - Algoritmo de exploração multi-robôs coordenados utilizando a 2ª métrica com restrições de comunicação.
- CMREM V1 - Algoritmo de exploração e mapeamento de multi-robôs coordenados utilizando a 1ª métrica com restrições de comunicação.

CMREM V2 - Algoritmo de exploração e mapeamento de multi-robôs coordenados utilizando a 2ª métrica com restrições de comunicação.

Na tabela 2 os algoritmos são referidos através dos seus acrónimos onde:

- Boust V1 - Algoritmo de Boustrophedon sem 2ª pesquisa
- Boust V2 – Algoritmo de Boustrophedon com 2ª pesquisa
- BSA V1 - BSA sem 2ª pesquisa
- BSA V2 – BSA com 2ª pesquisa
- CMRE V1 - Algoritmo de exploração multi-robôs coordenados 1ª métrica
- CMRE V2 - Algoritmo de exploração multi-robôs coordenados 2ª métrica
- CMREM V1 - Algoritmo de exploração e mapeamento de multi-robôs coordenados 1ª métrica
- CMREM V2 - Algoritmo de exploração e mapeamento de multi-robôs coordenados 2ª métrica

Como é possível verificar com o auxílio da tabela, os algoritmos descritos nos capítulos 6 e 7, são os que na simulação obtêm os melhores resultados quer em termos de percentagem de cobertura em todos os mapas quer nas iterações realizadas pelos robôs, isto acontece visto que ambos os algoritmos foram preparados para múltiplos robôs e para existir comunicação entre eles, o que no caso dos algoritmos de *Boustrophedon* e BSA não se verifica, dando origem a resultados inferiores a nível de percentagem de cobertura para números de iterações superiores em grande parte das situações. Para uma melhor comparação dos resultados obtidos encontram-se disponíveis na secção de anexos gráficos que agrupam os resultados obtidos sem restrições de comunicação agrupando todos os algoritmos para todos os mapas para o caso de sere utilizado 1 ou equipas de robôs de 3 e 5 agentes.

9. Trabalho Futuro

A simulação dos Algoritmos propostos foi efectuada em MATLAB que se verificou uma ferramenta poderosa no ambiente de simulação, no entanto dependendo do algoritmo é necessário um tempo computacional demasiado elevado para ser aplicado no mundo real. Assim, com o intuito de implementar um dos algoritmos simulados, será necessário importar todo o seu código implementado numa linguagem com a qual se possa obter uma melhoria significativa de performance a nível computacional.

Após a importação do código, seria implementado o algoritmo nos robôs P3DX da Pioneer pertencentes ao projecto MORFEU podendo assim obter resultados reais.

10. Referências Bibliográficas

- [1] H. Choset, “Coverage of Known Spaces: The Boustrophedon Cellular Decomposition,” *Autonomous Robots*, p. 7, 2000.
- [2] H. Choset e P. Pignon, “Coverage Path Planning: The Boustrophedon Cellular Decomposition,” p. 7.
- [3] E. González, O. Álvarez, Y. Díaz, C. Parra e C. Bustacara, “BSA: A Complete Coverage Algorithm,” *IEEE International Conference on Robotics and Automation*, p. 5, 2005.
- [4] E. U. Acar e H. Choset, “Robust Sensor-based Coverage of Unstructured Environments,” p. 8.
- [5] D. L. IV, S. Srinivasa, V. Lee-Shue, S. Sonne, H. Choset e A. Hurst, “Towards Sensor Based Coverage with Robot Teams,” p. 7.
- [6] K. Easton e J. Burdick, “A Coverage Algorithm for Multi-robot Boundary Inspection,” *IEEE International Conference on Robotics and Automation*, p. 8, 2005.
- [7] M. Bosse, N. Nourani-Vatani e J. Roberts, “Coverage Algorithms for an Under-actuated Car-Like Vehicle in an Uncertain Environment,” *IEEE International Conference on Robotics and Automation*, p. 6, 2007.
- [8] X. Zheng, S. Jain, S. Koenig e D. Kempe, “Multi-Robot Forest Coverage,” p. 6.
- [9] N. A. N. Hazon e G. A. Kaminka, “Constructing Spanning Trees for Efficient Multi-Robot Coverage,” *IEEE International Conference on Robotics and Automation*, p. 6, 2006.
- [10] P. Mukhija, R. Sawhney e K. M. Krishna, “Multi Robotic Exploration with Communication Requirement to a Fixed Base Station,” p. 2.
- [11] I. Rekleitis, V. Lee-Shue, A. P. New e H. Choset, “Limited Communication, Multi-Robot Team Based Coverage,” *IEEE International Conference on Robotics and Automation*, p. 7, 2004.
- [12] M. N. Rooker e A. Birk, “Communicative Exploration with Robot Packs,” p. 13.
- [13] W. Burgard e F. E. Schneider, “Coordinated Multi- Robot Exploration,” *IEEE transactions on robotics*, p. 11, 2005.

- [14] R. Simmons, D. Apfelbaum e W. Burgard, “Coordination for Multi-Robot Exploration and Mapping,” p. 7.
- [15] T. Kovács, A. Pásztor e Z. Istenes, “A multi-robot exploration algorithm based on a static Bluetooth communication chain,” *Robotics and Autonomous Systems*, p. 13, 2011.
- [16] M. n. Rooker e A. Birk, “Multi Robot Exploration Under The Constraints of Wireless Networking,” p. 14.
- [17] R. Sawhney, K. M. Krishna e K. Srinathan, “On Fast Exploration in 2D and 3D Terrains with Multiple Robots,” p. 8.

11. Anexos

Os gráficos apresentados de seguida visam comparar o comportamento de todos os algoritmos sem comunicação estando os gráficos divididos por 1, 3 e 5 robôs.

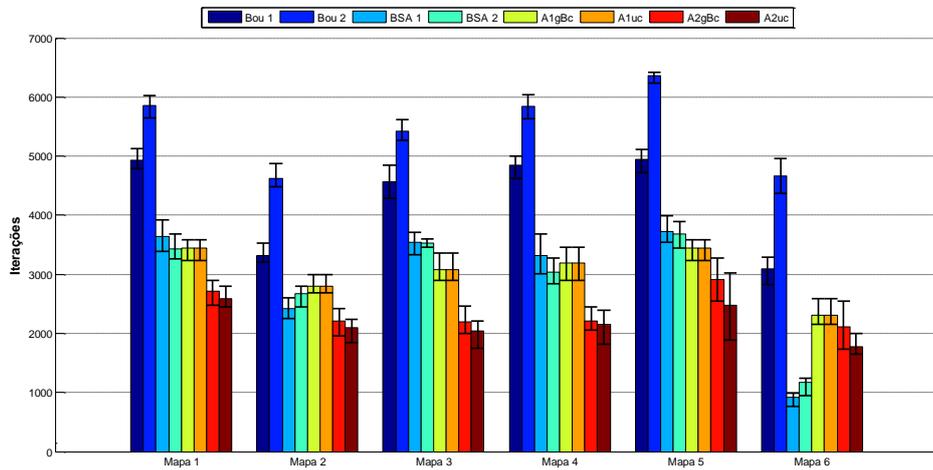


Figura 43 - Gráfico de Iterações com todos os algoritmos sem comunicação para 1 robô.

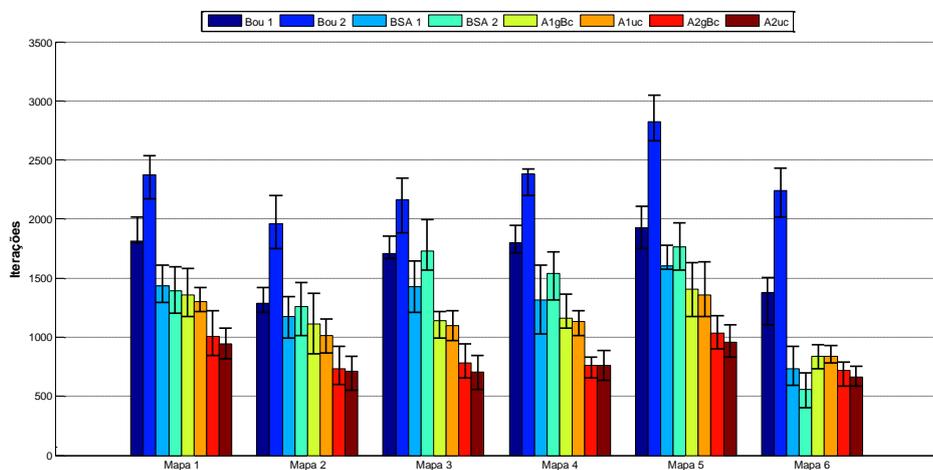


Figura 44 - Gráfico de iterações com todos os algoritmos sem comunicação para 3 robôs.

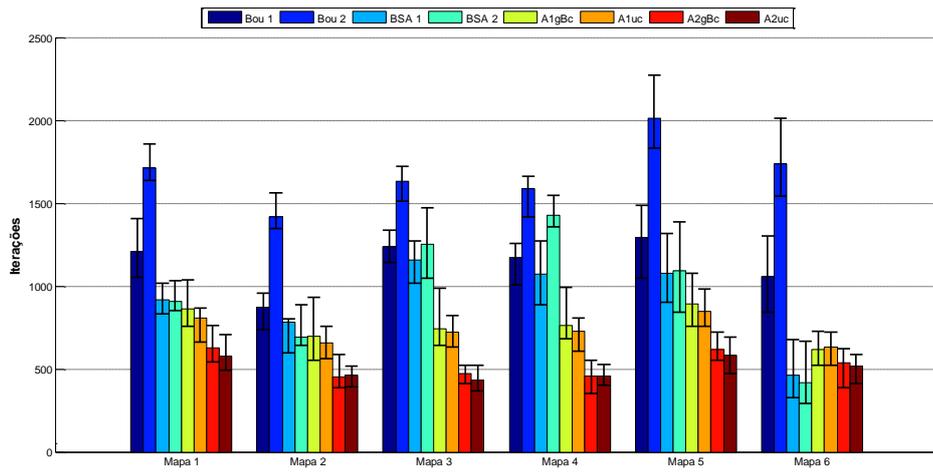


Figura 45 - Gráfico de iterações com todos os algoritmos sem comunicação para 5 robôs.

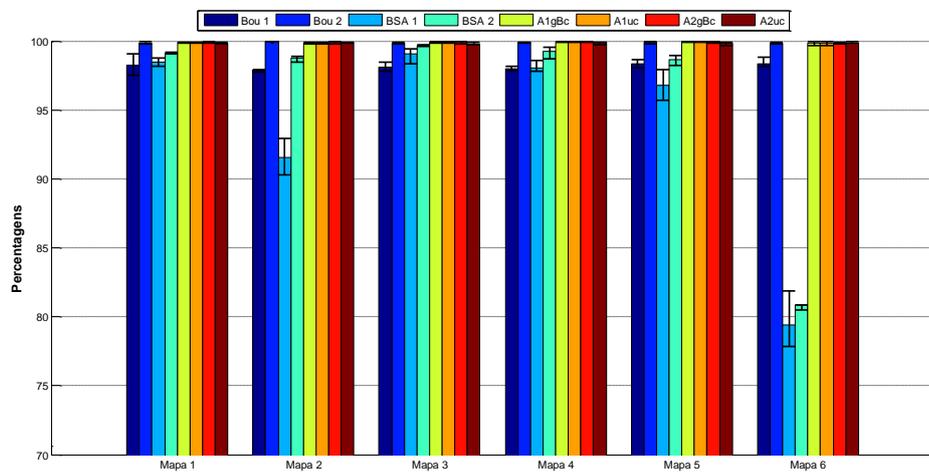


Figura 46 - Gráfico de percentagens com todos os algoritmos sem comunicação para 1 robô.

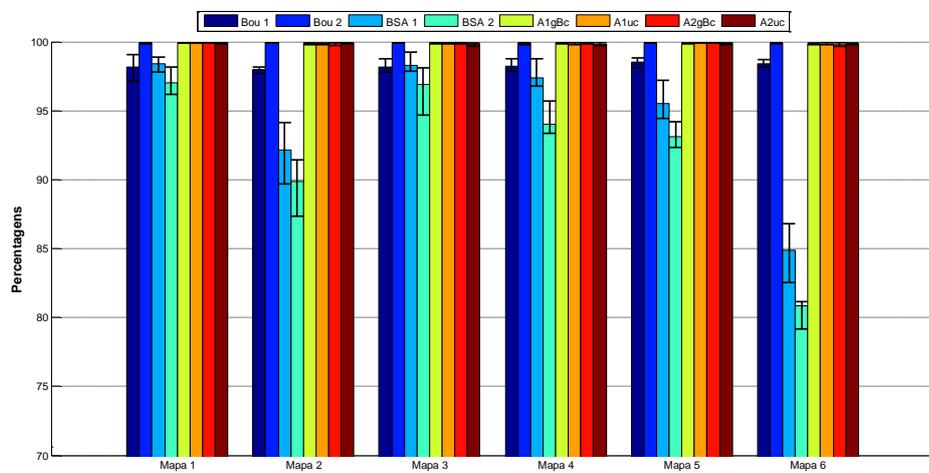


Figura 47 - Gráfico de percentagens com todos os algoritmos sem comunicação para 3 robôs.

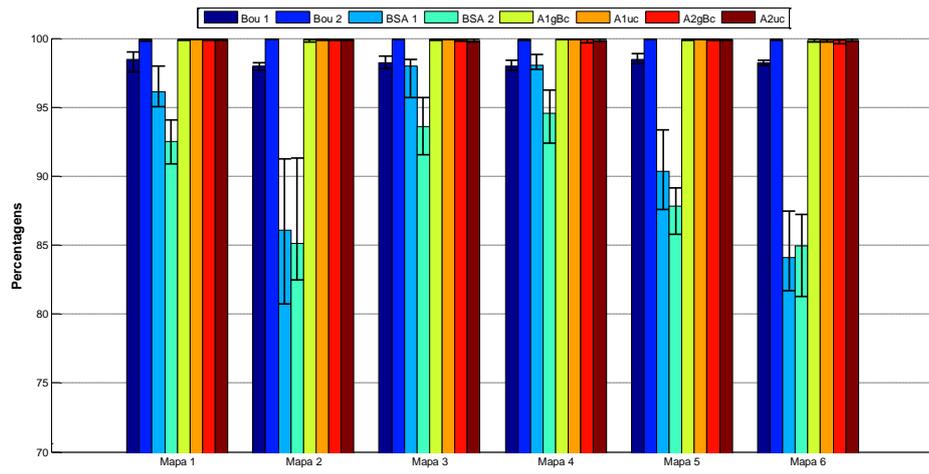


Figura 48 - Gráfico de percentagens com todos os algoritmos sem comunicação para 5 robôs.