

PROBLEMA DE PERSEGUIÇÃO-EVASÃO BASEADO EM *RANDOM WALK*

ANTÔNIO GONÇALVES¹, ELYSON CARVALHO¹, LUCAS MOLINA¹, EDUARDO FREIRE¹

1. Grupo de Pesquisa em Robótica da UFS
Departamento de Engenharia Elétrica, Universidade Federal de Sergipe
Av. Marechal Rondon, São Cristóvão, SE, Brasil

E-mails: renatonhocg@gmail.com, ecarvalho@ufs.br, lmolina@ufs.br, efreire@ufs.br

Abstract— *One of the main reasons for the use of robotic systems at the expense of human beings is to avoid risky situations, such as those found in activities of: search, surveillance and rescue. The problem of pursuit-evasion is a fundamental theoretical basis for the application of robotics in these cases. This paper presents an approach to solve the problem of pursuit-evasion without a priori knowledge of the map by coordinating a group of multiple mobile robots. The algorithm proposed here is based on the random walk in order to ensure system convergence, being contemplated planar environments without obstacles, which will be handled discreetly by an occupancy grid. To validate the proposed approach, various experiments were performed, which showed convergence in all cases.*

Keywords— Random Walk, Multi Robot System, Pursuit-Evasion Game without Map, Occupancy Grid

Resumo— Um dos principais motivos do uso de sistemas robóticos em detrimento de seres humanos é evitar situações de risco, como as encontradas atividades de busca, vigilância e resgate. O problema de perseguição-evasão é uma base teórica fundamental para a aplicação da robótica nestes casos. Este artigo apresenta uma abordagem para solução do problema de perseguição-evasão sem um conhecimento a priori do mapa através da coordenação de um grupo de múltiplos robôs. O algoritmo aqui proposto é baseado na random walk com o intuito de garantir a convergência do sistema, sendo contemplados ambientes planos e sem obstáculos, que serão tratados discretamente por meio de uma grade de ocupação. Para a validação da abordagem proposta foram realizados vários experimentos, que apresentaram convergência em todos os casos.

Palavras-chave — Caminhada Aleatória, Múltiplos Robôs, Perseguição-Evasão sem Mapa, Grade de Ocupação.

1 Introdução

Uma das principais motivações do uso de sistemas robóticos é a substituição de humanos em funções que lhes sejam particularmente perigosas. Atividades como busca e resgate costumam oferecer riscos à integridade física humana, por serem realizadas em ambientes hostis. Por muitas vezes, estas tarefas exigem a presença de vários agentes para que sejam cumpridas, no caso aqui proposto, um grupo de robôs. Porém, a coordenação deste grupo para atividades de busca, resgate e perseguição não é um encargo trivial e para ter êxito necessita de um planejamento que tem uma fundamentação teórica no problema de perseguição-evasão.

O problema de Perseguição-Evasão foi originalmente definido por Parsons (1976), onde foi levantada a seguinte questão: “Suponha que um homem está perdido em uma caverna escura. Um conjunto de buscadores que conhecem a estrutura dessa caverna é enviado para resgatá-lo. Qual é o número mínimo de buscadores necessários para encontrar o homem perdido na caverna, independentemente de como ele se comporte?”. Para a solução deste problema, Parsons transformou seu ambiente em um grafo e, a partir deste grafo, calculou qual o número mínimo de buscadores necessário para encontrar o homem na caverna.

O estudo desta área resultou na criação de várias abordagens que foram divididas da seguinte maneira por Kolling (2009): Perseguição-Evasão em Grafos, Perseguição-Evasão baseada em Visibilidade, Busca Probabilística em grafos, e Perseguição-Evasão sem Conhecimento do Mapa.

A Perseguição-Evasão em grafos (Megiddo, 1988; Barrière, 2002; Fomin, 2003; Kolling, 2007) é a área mais explorada dentre estas supracitadas. Nela, é dado um grafo G representando o ambiente, do qual o evasor tem total conhecimento e se move com velocidade ilimitada pelos seus nós. O evasor é capturado no momento em que se encontrar no mesmo nó que um perseguidor. O maior desafio desse tipo de abordagem é o cálculo do número mínimo de perseguidores necessários para a garantia de captura do evasor, este número é geralmente referido na literatura como *search number* ($s(G)$). A principal desvantagem desta abordagem é a necessidade prévia do mapa para a extração do grafo.

O problema de perseguição-evasão baseada em visibilidade (Suzuki, 1992; LaValle, 1997) é inspirada na proposta de Sugihara *et al.* (1990) onde os agentes buscadores eram fixos e tinham suas visões limitadas, simulando um farol ou um holofote. Posteriormente, Suzuki e Yamashita (1992) propuseram que os agentes fixos fossem trocados por agentes móveis, o que atualmente é considerado perseguição-evasão baseada em visibilidade. Nesse artigo é proposto que o robô que fará a busca tenha um número

' k ' de sensores, ou holofotes, que emitirão raios em direções discretas, estabelecendo o conceito de *k-searcher*. Caso o perseguidor possua sensores cobrindo todos os ângulos ele é chamado *∞-searcher*. Já LaValle *et al.* (1997) propuseram um algoritmo que coordenasse um *∞-searcher* de modo a garantir a captura dos invasores. A principal desvantagem desta abordagem é que os holofotes são abordados como sensores ideais de alcance infinito, o que não é compatível com a realidade.

Na abordagem de busca probabilística em grafos, Isler *et al.* publicaram três importantes artigos nesta área: (Isler *et al.*, 2004a), (Isler *et al.*, 2004b) e (Isler *et al.*, 2005). No primeiro deles (Isler *et al.*, 2004a) eram considerados um ou mais perseguidores procurando um evasor em um determinado grafo. Este jogo é dividido em turnos, no qual, em cada turno, a caça tenta obter informações sobre a localização do caçador, porém, eles só percebem a presença de um perseguidor a um nó de distância. A partir dessa premissa, é provado que bastam dois perseguidores para garantir a captura da presa. Nos outros dois artigos, são mostradas abordagens de busca probabilística para um problema de perseguição-evasão em grafos baseado em visibilidade. Assim como no problema de perseguição-evasão em grafos, ainda é necessário o conhecimento prévio do mapa.

Finalmente, em perseguição-evasão sem conhecimento de mapas, Kolling e Carpin (2010) propuseram uma solução para o problema de perseguição-evasão onde os perseguidores são bastante limitados, o que o torna bastante importante. Os robôs só possuem estratégia de seguir parede, formar linha de exploração e lidar com obstáculos. Os resultados dos seus experimentos mostram que, apesar de garantir a limpeza do ambiente, ele não cumpre esta tarefa com o número mínimo de robôs necessário, por não possuir uma maneira ótima de lidar com obstáculos. Além dessa desvantagem citada anteriormente, por ser um algoritmo em desenvolvimento, não são apresentadas definições e provas formais que comprovem o cumprimento da limpeza de qualquer ambiente fechado.

Este artigo é focado em apresentar uma solução para o problema de perseguição-evasão sem conhecimento dos mapas para o pior dos casos, no qual o evasor é onisciente e tem velocidade ilimitada. Esta suposição faz com que todo espaço no qual os evasores podem estar presentes seja considerado contaminado. A tarefa será descontaminar (ou limpar) o ambiente por completo e, ao realizar esta tarefa, garantir a captura do evasor, que se dá no instante da visualização do mesmo.

A abordagem aqui proposta será baseada na técnica denominada *random walk* (em português, caminhada aleatória), por esta ser uma formalização matemática que garante que um objeto móvel parte de um ponto inicial e , por meio de sucessivos passos aleatórios, chega a seu destino em um tempo $t < \infty$ (Papoulis, 1965). Baseado nesta premissa e com algumas restrições locais, a serem definidas nas próxi-

mas seções, é suposto que o conjunto de robôs capture o evasor em um tempo finito.

Este artigo está distribuído da seguinte maneira: Na seção 2 é apresentada uma introdução de *random walk* e detalhes da abordagem aqui proposta. Os algoritmos utilizados para simulação são contemplados na seção 3. Na seção 4 são mostrados os resultados obtidos nas simulações. Por fim, na seção 5 são apresentadas as conclusões e expectativas de desenvolvimento do corrente trabalho e em seguida as referências bibliográficas.

2 *Random Walk* e a Abordagem Proposta

O problema da *random walk* foi originalmente proposto por Karl Pearson (1905), onde ele buscava encontrar a probabilidade de um homem, partindo de um ponto O e dando sucessivos n passos aleatórios, se encontrar em uma distância entre r e $r + \delta \cdot r$ do ponto de origem O .

A formalização matemática da *random walk* prova que, caso exista um caminho possível para que este homem chegue a um ponto desejado, haverá uma probabilidade não nula para que o mesmo chegue ao seu destino final em um tempo finito através de sucessivos passos aleatórios.

Neste artigo, a *random walk* é aplicada aos agentes perseguidores e, com isso, a convergência do sistema é garantida, caso haja o número mínimo de robôs necessário para cumprir a tarefa. Porém, para resolução do problema de perseguição-evasão, os robôs perseguidores devem percorrer seus possíveis caminhos de maneira sincronizada não permitindo a recontaminação do ambiente. Como a probabilidade de que isso ocorra utilizando apenas a *random walk* é muito baixa, serão impostas algumas restrições estratégicas locais, que serão discutidas mais a frente, a fim de aumentar as probabilidades e, conseqüentemente, diminuir o tempo de convergência do sistema. O objetivo dessas restrições é fazer com que os robôs comportem-se como um grupo que se moverá em conjunto.

O problema de perseguição-evasão aqui proposto se dará em um mapa discreto, não conhecido previamente pelos perseguidores, que será representado por uma grade regular dividida com uma amostra s . Quanto menor o valor atribuído a s , melhor será a definição da grade e, portanto, apresentará mais detalhes do mapa original. Em uma grade regular, toda célula que contenha ao menos um ponto representando um obstáculo será considerada como uma célula ocupada.

Os robôs perseguidores serão considerados pontos que ocuparão o centro de uma célula, já os evasores terão o tamanho de uma célula, garantindo que os mesmos não se escondam na imperfeição do processo de discretização do ambiente.

Para o início do cenário, os robôs começarão em quaisquer células não ocupadas, por um obstáculo, ou por outro robô. Na sua movimentação, os robôs terão uma restrição dinâmica de só poderem se mover para

as quatro células imediatamente vizinhas ou permanecerem onde estão. Além das células imediatamente vizinhas, os robôs possuirão uma visão de todas as células que estejam totalmente cobertas por um círculo de raio r_{sensor} . A representação gráfica do robô e suas limitações dinâmicas e sensoriais são dadas na Figura 1.

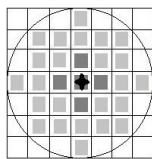


Figura 1 - Representação de um robô na grade. A área cinza escuro representa a área de atuação do mesmo. A área cinza claro representa o campo de visão, delimitado por um círculo de raio círculo de raio r_{sensor} .

Além das restrições dinâmicas, ao moverem-se, os robôs deverão obedecer algumas restrições físicas e estratégicas impostas. A restrição física determina que um robô não poderá ir a uma célula caso a mesma já esteja ocupada por um obstáculo do ambiente, ou por outro robô. As restrições estratégicas serão com a finalidade de evitar a recontaminação por entre o grupo de perseguidores.

Antes das restrições estratégicas serem impostas, é importante a definição do conceito de ligação, que será utilizado neste trabalho. Uma ligação é uma reta virtual traçada entre dois elementos em questão. As ligações poderão ser de dois tipos: robô-parede ou robô-robô. A função de uma ligação é garantir que o espaço, entre os dois elementos em questão, está monitorado, não permitindo a passagem despercebida de um evasor por entre os mesmos.

Caso a ligação seja entre um robô “A” e uma parede, a distância máxima permitida entre estes dois elementos será r_{sensor} . Caso a ligação seja entre dois robôs (robô B e robô C), a distância máxima permitida entre os mesmos será de duas vezes r_{sensor} , pois esta distância é o somatório do raio de monitoramento do robô B com o do robô C.

Para que os robôs possam realizar ligações do tipo robô-robô, será necessário o conhecimento da posição do robô com o qual ele está se ligando. Portanto, é preciso que haja um raio de comunicação entre os robôs de pelo menos duas vezes r_{sensor} e que cada robô tenha conhecimento de sua posição global.

Feitas as definições necessárias, serão estabelecidas agora as restrições estratégicas:

- O grupo de robô se ligará formando uma espécie de linha (Figura 2), onde cada robô deverá possuir somente duas ligações;
- Dois, e apenas dois, robôs deste grupo deverão possuir uma ligação do tipo robô-parede cada. Esta restrição evitará a formação de ciclos por parte do grupo de robôs;
- Os demais robôs deverão manter duas ligações robô-robô;

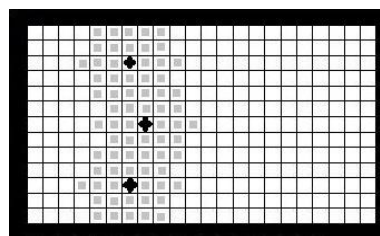


Figura 2 - Exemplo de uma configuração onde todos os robôs perseguidores, representados em preto, possuem duas ligações e não permitem recontaminação.

Analisando a Figura 2, pode-se observar que o robô mais acima da formação possui uma ligação com a parede de que limita o ambiente superiormente e outra ligação do tipo robô-robô com o robô mais central. De maneira análoga, o robô mais abaixo da figura possui também uma ligação robô-parede e outra do tipo robô-robô. O robô mais central possui duas ligações do tipo robô-robô. A partir dessas constatações sobre a figura, pode-se afirmar que esta configuração cumpre a todas as restrições.

A movimentação dos robôs se dará por turnos onde serão sorteadas aleatoriamente células candidatas a destino de cada um dos robôs perseguidores, que deverão atender às restrições dinâmicas, físicas e estratégicas impostas neste trabalho.

3 Implementação do sistema

Nessa seção serão abordados alguns detalhes importantes para a criação do ambiente de simulação, que foi utilizado para testar o algoritmo aqui proposto em um conjunto de robôs. Como o algoritmo foi proposto para um problema de perseguição-evasão em uma grade de ocupação, algumas adaptações devem ser feitas para uma melhor semelhança com a realidade.

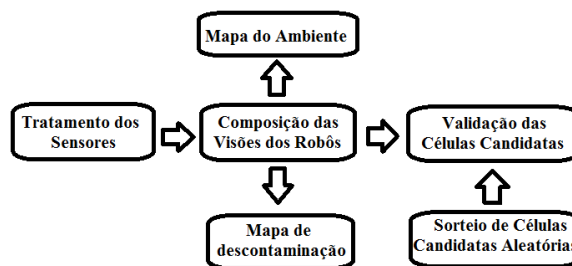


Figura 3 - Diagrama de blocos do Sistema.

O diagrama de blocos dado na Figura 3 representa um esquema do ambiente de simulação. Maiores detalhes de cada bloco serão dados nas subseções a seguir.

3.1 Tratamento dos Sensores

Para determinar a visão do robô, é considerado um círculo de raio r_{sensor} com o seu centro coincidente com o centro da célula onde o robô se encontra. A partir deste círculo, podem-se determinar todas as células que forem totalmente internas ao círculo. Este conjunto de células foi chamado de máscara do sen-

sor e delimita a visão dos robôs na grade de ocupação (Figura 1). É importante notar que a máscara é diretamente dependente do raio do sensor.

A máscara do sensor servirá de filtro para selecionar a área do mapa discreto que será vista pelo robô, porém, ao aplicar somente esta máscara no mapa, obtém-se uma visão em raios-X dos obstáculos (Figura 4.a). Para simular uma visão que seja mais compatível com a realidade, é feita uma propagação do bloqueio da visão por parte dos obstáculos do mapa, como é possível analisar na Figura 4.

Para verificar as células que possuem sua visão bloqueada por um obstáculo, devem-se traçar retas entre o centro da célula ocupada pelo robô e os quatro vértices da célula ocupada pelo obstáculo. As duas retas que formarem o maior ângulo serão usadas como retas de base para bloquear a visão das demais células.

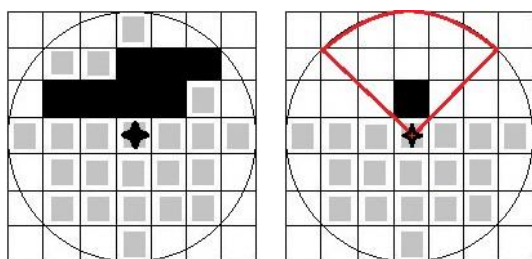


Figura 4 - (a) Visão do Robô através dos obstáculos. (b) Visão do Robô após a propagação do bloqueio.

Serão bloqueadas todas as células que possuem uma distância, entre o seu centro e o centro do círculo, maior do que a distância entre a célula ocupada e o centro do círculo e, além disso, possuem ao menos um vértice interno à fatia de círculo delimitada pelas retas de base. Estas células que tiveram sua visão bloqueada gerarão uma incerteza de ocupação (não se pode deliberar acerca de sua ocupação) e deverão propagar essa incerteza para as células que elas bloqueariam caso estivessem ocupadas. O resultado da propagação do bloqueio da visão simula com mais realidade o conhecimento do robô sobre o mapa e pode ser visualizado na Figura 4.b.

3.2 Composição de Visão e Mapeamento

Após o resultado do tratamento dos sensores para cada um dos robôs, será necessária uma composição dessas visões, que será a união das visões de cada um dos robôs. Essa operação só é possível porque os robôs tem conhecimento de sua localização global e podem se comunicar com os robôs que estão a uma distância de duas vezes o raio do sensor.

Após a composição das visões, todos os obstáculos encontrados são armazenados em uma matriz, que representará o ambiente de acordo com a visão do grupo de robôs. Ao mesmo tempo é formado um mapa de contaminação indicando a situação atual de descontaminação, no qual, a cada iteração, é feita uma tentativa de recontaminação das células descontaminadas.

Esta tentativa de recontaminação é feita a partir da situação atual do mapa de contaminação, onde são verificadas todas as vizinhanças imediatas das células descontaminadas. Caso ao menos uma das células vizinhas esteja contaminada, a célula analisada no início do processo será recontaminada, gerando uma nova configuração para o mapa. A cada vez que este sofrer uma alteração de configuração, a tentativa de recontaminação deverá executada novamente de maneira recursiva, até não haver mais possíveis recontaminações.

3.3 Random Walk Paralela e Condição de Parada

Para a movimentação dos robôs, foi realizado um algoritmo simulando um paralelismo de movimentação dos mesmos. Para gerar uma *random walk* é utilizado uma função geradora de números pseudoaleatórios que sorteará uma célula candidata para cada um dos robôs perseguidores.

Após a composição das células candidatas, obtém-se um cenário candidato, onde os robôs estarão dispostos em suas novas posições. Caso este cenário candidato obedeça às restrições impostas na seção anterior, as células candidatas serão aceitas e tomadas como destino dos robôs na próxima iteração. Caso o cenário candidato não cumpra as restrições exigidas, serão realizados novos sorteios de células candidatas, que irão compor um novo cenário candidato até que o mesmo se enquadre nas restrições.

A cada iteração, o grupo de perseguidores realizará este processo de sorteio e testes, armazenando seu histórico de descontaminação. O processo estará finalizado quando o ambiente estiver totalmente descontaminado, o que garantirá a captura do evasor.

Por ser omnisciente e possuir velocidade ilimitada, o evasor não necessita ser implementado, pois é assumido que ele estará sempre no último ponto a ser descontaminado do mapa.

3.4 Estabilidade do Sistema

As restrições impostas ao grupo de robôs fazem com que o mesmo cumpra a tarefa mais rapidamente, através da exclusão de caminhos que não atingem o objetivo final. Essas restrições são em nível de grupo, não limitando a movimentação do robô individualmente dentro do cenário. Como só são excluídos os caminhos que não levam à convergência, existirá ao menos um caminho no qual o grupo de robôs cumpre a tarefa proposta. A existência de ao menos uma possibilidade de convergência é suficiente para sustentar que a argumentação de Papoulis (1965) se aplica à situação do algoritmo proposto.

4 Resultados

Para análise dos resultados obtidos com os experimentos realizados, foram escolhidas duas métricas: evolução da descontaminação e número de iterações para conclusão da descontaminação do ambiente. A

métrica de evolução da descontaminação é uma função no domínio das iterações e quantifica as células contaminadas por iteração. A outra métrica utilizada é o número de iterações que o grupo de robôs necessitou para descontaminar todo o ambiente.

Os experimentos foram realizados em mapas de 800x600 cm, que foram discretizados com uma taxa de amostragem $s=25$. Todos os robôs terão o mesmo raio de alcance de sensor $r_{sensor}=150$ cm e terão um alcance de comunicação de 300 cm. Para que possa ser garantida a convergência em todos os experimentos, a quantidade mínima de robôs necessários foram quatro.

O primeiro experimento foi realizado em um ambiente retangular, no qual os robôs conseguiram descontaminar toda a área em todas as simulações realizadas, variando apenas o número de iterações para a conclusão da tarefa. Uma imagem representativa de uma sequencia de situações que ocorrerão para a descontaminação total do ambiente é apresentada na Figura 5.

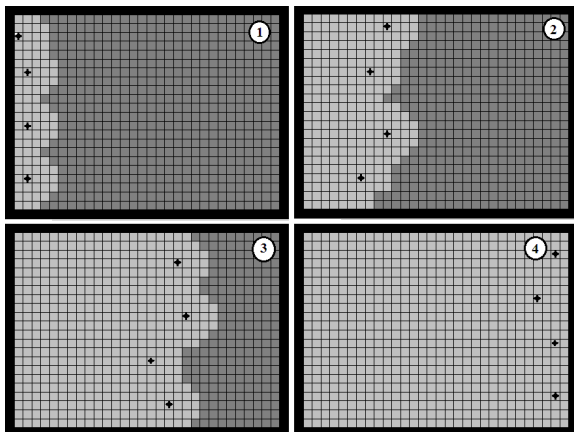


Figura 5 - Sequência que resultou na descontaminação do ambiente do primeiro experimento, em uma das simulações. A cor clara representa as células descontaminadas, já a cor escura representa as células contaminadas.

Este experimento foi repetido cinco vezes e os resultados obtidos tiveram uma média de convergência de 56 mil iterações, com o resultado mínimo em 43274 iterações e o máximo em 62543 iterações. No gráfico do histórico da descontaminação do ambiente em uma das repetições (Figura 6), pode-se notar a atuação da aleatoriedade criando uma forte oscilação no progresso da convergência do sistema. Em vários instantes atinge-se metade da descontaminação do ambiente, e, no entanto, o grupo de robôs executa uma série de passos que faz com que o número de células contaminadas volte a aumentar. Pode-se notar, pela leitura do gráfico da Figura 6, que a descontaminação total do ambiente ocorre em algum instante próximo à 51000ª iteração.

Para situações semelhantes à deste primeiro experimento, os resultados de heurísticas encontradas na literatura, como a de Kolling (2010), serão mais eficientes, normalmente efetuando a descontaminação de um ambiente deste porte em um tempo menor. Porém, por se tratarem de heurísticas, não possuem provas de convergências, tornando-os sistemas não

robustos, podendo apresentar falhas em alguns cenários.

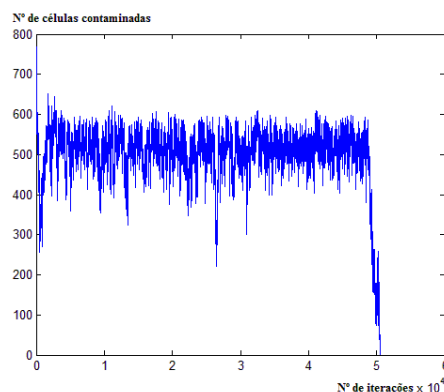


Figura 6 - Histórico de descontaminação de uma simulação no primeiro ambiente proposto.

No segundo experimento, foi utilizado um mapa com as mesmas dimensões do primeiro, porém com configuração diferente. Para este mapa, foram feitas três simulações que tiveram uma média de 58 mil iterações para que ocorresse a descontaminação total. Na segunda simulação, a configuração dos robôs no início e término do problema são apresentadas na Figura 7, onde também é possível notar a configuração do mapa.

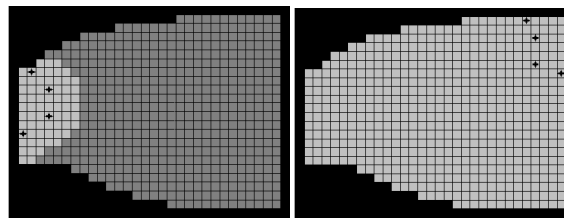


Figura 7 - Condições iniciais das posições dos robôs e contaminação do ambiente, representada pelas células escuras.

Em todas as três simulações, o algoritmo proposto obteve êxito. O resultado mais representativo do histórico de descontaminação é apresentado na Figura 8, onde é interessante notar que o experimento esteve muito próximo da descontaminação total do ambiente em um instante próximo à 38000ª iteração, o que mostra novamente que a ação da componente aleatória é muito influente no sistema. Nesta simulação em específico, a tarefa foi concluída após 63 mil iterações, aproximadamente.

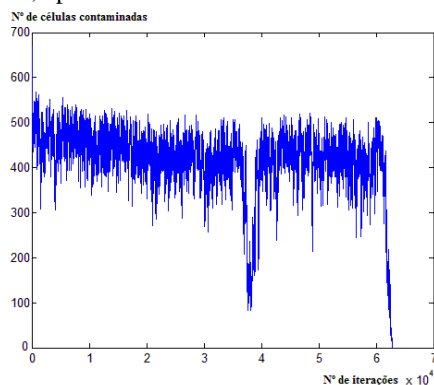


Figura 8 - Histórico de descontaminação do Mapa do segundo experimento.

Por fim, foi realizado um último experimento com o mapa apresentado na Figura 9. Para este ambiente foram realizadas cinco simulações com posições iniciais diferentes para os robôs e obteve-se êxito em 100% delas.

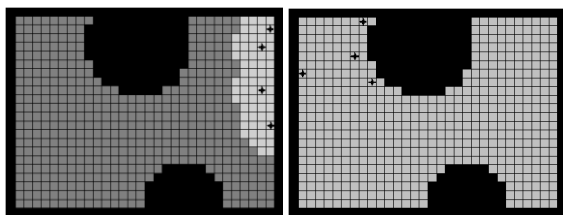


Figura 9 - (a) Mapa original do terceiro experimento. (b) Representação discreta do mesmo mapa.

A média de número de iterações para convergência foi a maior dentre os três mapas neste trabalho (6 mil iterações). Uma das simulações obteve um resultado muito eficiente, convergindo em 3958 iterações. O histórico de descontaminação dessa simulação em particular é apresentado na Figura 10.

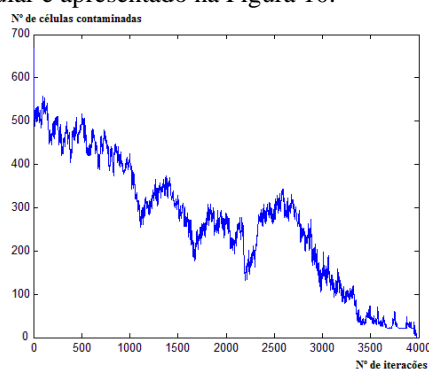


Figura 10 - Histórico de descontaminação do Mapa do último experimento.

5 Conclusão

Neste artigo foi apresentada uma abordagem baseada em *random walk* para solução do problema de perseguição-evasão em uma grade de ocupação, utilizando robôs perseguidores com visão limitada.

Apesar de apresentar resultados que exigem muitas iterações, o algoritmo proposto possui convergência garantida devido à *random walk*. Por se tratar de um sorteio aleatório de células candidatas, nas quais serão realizados apenas alguns testes de restrições, o algoritmo se mostra simples e de baixo custo computacional.

Após a simulação de vários experimentos, esta abordagem baseada em *random walk* se mostrou robusto, atingindo êxito em todas as ocasiões. A principal característica contra a abordagem proposta é o alto tempo de convergência.

Como forma de melhorar este tempo de convergência vários trabalhos podem ser realizados no sentido de ponderar as probabilidades do sorteio aleatório das células candidatas. O parâmetro mais importante a ser levado em conta é a taxa de descontaminação em relação à célula candidata.

Além desta sugestão de melhoria no tempo de convergência, os trabalhos futuros podem contemplar

ambientes com obstáculos, que não foram considerados no presente trabalho.

Agradecimentos

Agradecemos a CAPES pelo apoio financeiro, fundamental para a realização deste trabalho.

Referências Bibliográficas

- Fomin, F. V.; Kratsch, D. and Müller, H. (2003). "On the domination search number." *Discrete Applied Mathematics*, 127(3):565–580.
- Isler, V.; Kannan, S. and Khanna, S. (2004a). "Locating and capturing an evader in a polygonal environment." In *Proceedings of the Workshop on Algorithmic Foundations of Robotics*, pp. 351–367.
- Isler, V.; Kannan, S. and Khanna, S. (2004b). "Randomized pursuit-evasion with limited visibility." In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, pp. 1060–1069, Philadelphia, PA, USA. Society for Industrial and Applied Mathematics.
- Isler, V.; Kannan, S. and Khanna, S. (2005). "Randomized pursuit-evasion in a polygonal environment." *IEEE Transactions on Robotics*, 21(5):875–884.
- Kolling, A. and Carpin, S. (2007). "Detecting intruders in complex environments with limited range mobile sensors." In K. Kozłowski, editor, *Robot Motion and Control, LNCIS 360*, pp. 417–426. Springer-Verlag London Limited.
- Kolling, A. (2009). *Multi-Robot Pursuit-Evasion*. Ph.D thesis, University of California.
- Kolling, A. and Carpin, S., "Multi-robot pursuit-evasion without maps," *Robotics and Automation (ICRA)*, 2010 IEEE International Conference on , vol., no., pp.3045,3051.
- L. Barrière, L.; Flocchini, P.; Fraigniaud, P. and Santoro, N. (2002). "Capture of an intruder by mobile agents." In *Proceedings of the Fourteenth Annual ACM Symposium on Parallel Algorithms and Architectures*, pp. 200–209, New York, NY, USA. ACM Press.
- LaValle, S. M.; Lin, D.; Guibas, L.; Latombe, J.C.; and Motwani, R. (1997). "Finding an unpredictable target in a workspace with obstacles." In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 737–742.
- Megiddo, N. ; Hakimi, S. L.; Garey, M. R.; Johnson, D. S. and Papadimitriou, C. H. (1988). The complexity of searching a graph. *Journal of the ACM*, 35(1):18–44.
- Papoulis, A. (1965). *Probability, random variables, and stochastic processes*. McGraw-Hill, 1965.
- Parsons, T.D. (1976). "Pursuit-evasion in a graph." In Y. Alavi and D. R. Lick, editors, *Theory and Applications of Graphs*, volume 642, pp. 426–441. Springer Berlin / Heidelberg.
- Pearson, K. (1905). The Problem of the Random Walk. *Nature* 72, 294; 318; 342.
- Sugihara, K.; Suzuki, I.; and Yamashita, M. (1990). "The searchlight scheduling problem." *SIAM Journal on Computing*, 19(6):1024–1040.
- Suzuki, I. and Yamashita, M. (1992). "Searching for a mobile intruder in a polygonal region." *SIAM Journal on Computing*, 21(5):863–888.