

Um Sistema de Vigilância e Perseguição Usando Occupancy Grid e Uma Versão Modificada de Campos Potenciais

ANDRE MOTTA, Universidade Federal de Minas Gerais, Brasil

DANIEL ISHITANI, Universidade Federal de Minas Gerais, Brasil

Algoritmos de detecção e perseguição de invasores em ambientes conhecidos envolvem diversas técnicas de Robótica Móvel em conjunto. Suas aplicações variam desde uma simples câmera detectora de intrusos e reconhecimento de objetos dinâmicos em um ambiente estático até a programação de um robô perseguidor ao reconhecer um objeto estranho dentro de sua área de proteção. Usando um algoritmo de detecção por meio de um mapeamento feito com o Grid de Ocupações e uma perseguição feita com o algoritmo otimizado dos Campos Potenciais, buscamos desenvolver esta ultima aplicação.

Palavras Chave: Potential Fields, Occupancy Grid, Map Tracking, Detection, Pursuit, Evasion, Capture, Area Coverage

- Andre Motta and Daniel Ishitani. 2019. Um Sistema de Vigilância e Perseguição Usando Occupancy Grid e Uma Versão Modificada de Campos Potenciais.

1 INTRODUÇÃO

O problema proposto foi a vigilância de um ambiente totalmente conhecido pelos robôs que realizam a ronda. Os vigilantes conhecem a localização de seus aliados, então seu objetivo é perseguir e prender o intruso, caso seja visto. Para realizar a ronda determinamos uma sequência de pontos que cada robô deve seguir em seu quadrante, simplesmente seguindo reto para o próximo ponto já que sabe-se que não há obstáculos. Ao identificar um intruso, o vigilante muda o caminhamento para campos potenciais, alterando o objetivo para a posição do intruso, e volta para o último ponto da ronda caso perca o invasor de vista. O ladrão, por outro lado, tem dois objetivos. Primeiro ele precisa buscar o dinheiro e depois encontrar a saída, sem ser preso, utilizando campos potenciais para realizar o caminhamento.

2 TRABALHOS RELACIONADOS

Para este trabalho, usamos os métodos descritos por H. Moravec and A. Elfes[1] para pixelizar o mapa e adequar o posicionamento de uma coordenada x,y sobre um mapa discreto. Usamos também versões modificadas do algoritmo dos campos potenciais descrito originalmente por R. B. Tilove[2]. Para desenvolver os métodos de detecção e perseguição utilizamos técnicas que tiveram como inspiração os trabalhos de Peng Song e V. Kumar[3] e P. Kondaxakis, S. Kasderidis e P. Trahanias [4]

Authors' addresses: Andre Motta, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil, andre.motta@dcc.ufmg.br; Daniel Ishitani, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil, daniel.ishitani@dcc.ufmg.br.

© 2019 Licensa mantida pelo dono/autor.
Versao disponivel para uso pessoal. Permitida toda redistribuicao nao monetizada.

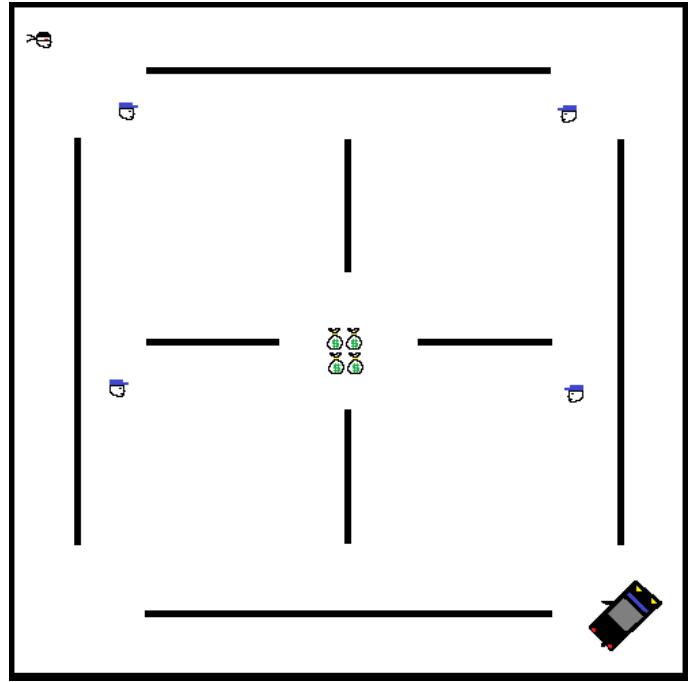


Fig. 1. Banco

3 METODOLOGIA

Em prol do uso de um código modularizado desenvolvemos um arquivo python para cada robô, sendo então cada robô uma entidade independente cada um tomando decisões em tempo real. Para a declaração de que o jogo terminou, seja pela captura do ladrão ou sua fuga, criamos um tópico para comunicação entre os robôs, o mesmo só é utilizado para o desligamento dos robôs findado o jogo.

3.1 O Ladrão

O Ladrão deve ser um robô inteligente, por premissa ele conhece o alcance de visão do policial, deve buscar o dinheiro no centro do banco e chegar ao carro de fuga como visto na figura 1. Desta forma, realizamos algumas alterações no algoritmo de campos potenciais. A primeira delas é que o ladrão possui um alcance maior de visão do que o policial. E isso reflete no fato que a repulsão causada pelo policial permite uma distância máxima 50% maior que a repulsão causada pelas paredes que é 1. A segunda das decisões é que a repulsão causada pelo policial seria mais forte e sempre sem a otimização aplicada aqui ao algoritmo dos campos potenciais.

O algoritmo de campos potenciais, usa analogamente um modelo de repulsões e atrações. Como em cargas, o robô seria uma carga

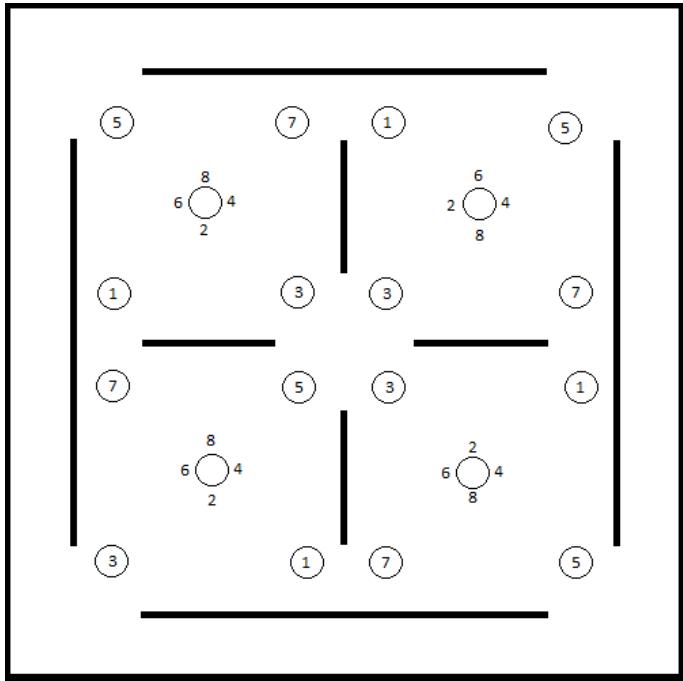


Fig. 2. Caminhamento

positiva, assim como as paredes e o policial. E o objetivo do robô seria uma carga negativa. Para o algoritmo, a força de atração ao objetivo aumenta de acordo com a distância. Já as forças de repulsão diminuem com a distância.

Este algoritmo embora simples, cria um comportamento de inteligência emergente por ser um modelo reativo. Contudo isso faz com que o mesmo não seja capaz de garantir a completude do algoritmo. Dependendo do posicionamento do robô relativo ao objetivo, mínimos locais podem ocorrer, ocasionando na parada do robô. Uma otimização para isso é a de fazer com que as paredes empurrem o robô perpendicularmente a elas, assim os mínimos locais se tornam bem mais raros.

Como descrito no algoritmo 1, podemos ver o cálculo da maneira como descrita acima, e detalhe para a otimização no cálculo da força de repulsão das paredes. Note que o algoritmo apenas referencia o cálculo da força total em X, mas a força em Y também será calculada de maneira análoga usando o seno em ambas as situações.

Explicando o referido algoritmo 1. O algoritmo de campos potenciais usa de uma fórmula matemática de cálculo para a atração e a repulsão muito similar a um sistema eletromagnético de cargas. As otimizações aqui são o uso do sensor de laser para que o ladrão possa diferenciar um policial de uma parede qualquer devido ao policial refletir o laser com mais intensidade $intensities > min_cop_intensity$. Assim permitindo sempre ser afetado pelo policial, e que essa repulsão seja aumentada em 1000 vezes.

Se a intensidade não for superior a intensidade que identifica um policial, o algoritmo dos campos potenciais escrito dita que o robô só sera afetado se $distance \leq max_distance$. Com isso fazemos uma rotação em 90 das forças para que o robô seja afetado de modo

a caminhar paralelamente a fonte da repulsão.

3.2 O Policial

O Policial, enquanto não detecta um ladrão, faz um caminhamento simples, usando apenas um cálculo de velocidades baseado na reta entre os pontos, em seu quadrante seguindo a sequência demonstrada na 2. Como descrito anteriormente, o campo de visão do policial é menor que o alcance do ladrão. Ao detectar um ladrão, o policial troca o caminhamento para campos potenciais no qual o objetivo do robô é atualizado, a cada passo, para a posição do intruso. Caso perca o invasor de vista, o policial retorna ao último ponto de sua ronda ainda usando o algoritmo dos campos potenciais e volta ao caminhamento de pontos pré-determinados.

Para identificar que um obstáculo trata-se de um invasor, o policial compara a posição (x, y) , traduzida para um mapa discretizado gerado pelo Occupancy Grid, com o mapa perfeito disponível sobre o ambiente. Caso não seja uma parede, compara a posição do obstáculo com as posições dos outros policiais, obtida através de odometria compartilhada. Se ambas as checagens forem negativas, é iniciada a perseguição, pois a detecção de um objeto estranho ao policial ocorreu.

3.3 Regras do Jogo

As regras do jogo são bem simples, o ladrão deve pegar o dinheiro no centro da sala nas coordenadas $(0, 0)$ e depois chegar ao carro de fuga nas coordenadas $(7, -7)$ enquanto evade os policiais. Se ele chegar com sucesso na posição final, ele emite uma mensagem no tópico de comunicação informando os policiais do fim do jogo.

Para os policiais existe apenas um objetivo, se detectarem um invasor devem prender este invasor a todo custo. Uma prisão é efetuada se o ladrão estiver a 0.4 de distância de um policial. Efetuando a prisão, o policial emite uma mensagem no tópico de comunicação informando os outros policiais e o ladrão do fim do jogo.

3.4 Das Especificações

3.4.1 Dos Robôs.

Robô Holônômico

Laser Field of view: 360

Robot Size = [0.2 0.2 0.2]

3.4.2 Das Rotas dos Policiais.

police0 =

[(-5.5, 0.5), (-3, 3), (-0.5, 0.5), (-3,3), (-5.5, 5.5), (-3, 3), (-0.5, 5.5), (-3,3)]

police1 =

[(0.5,5.5), (3,3), (0.5, 0.5), (3,3), (5.5, 5.5), (3,3), (5.5, 0.5), (3,3)]

police2 =

[(5.5, -0.5), (3, -3), (0.5, -0.5), (3,-3), (5.5, -5.5), (3,-3), (0.5, -5.5), (3,-3)]

police3 =

[(-0.5, -5.5), (-3,-3), (-5.5, -5.5), (-3,-3), (-0.5, -0.5), (-3,-3), (-5.5, -0.5), (-3,-3)]

3.4.3 Dos Objetivos do Ladrão.

dinheiro = $(0, 0)$

carro de fuga = $(7, -7)$

ALGORITHM 1: Campos Potenciais do Ladrão

Input: 360 Medidas do Laser (1 por grau). Cada medida contendo 1 distância e 1 intensidade. A distância ao objetivo em x e y

Output: A força total no eixo Y e X para o movimento do robô.

```

index = 0;
max_distance = 1;
min_cop_intensity = 0.5;
distance_gain = 1.5;
attraction_gain = 15;
repulsion_gain = 1.5;
total_x_repulsion = 0;
repeat
    if distances[index] <= max_distance * distance_gain and
    intensities[index] > min_cop_intensity then
        total_x_repulsion+ =
            1000 * repulsion_gain(  $\frac{1}{distances[index]} - \frac{1}{max\_distance}$  ) *
            (  $\frac{1}{distances[index]^2} * \frac{distances[index]*cos(index)}{distances[index]}$  )
    else
        if distances[index] <= max_distance then
            total_x_repulsion+ =
                repulsion_gain(  $\frac{1}{distances[index]} - \frac{1}{max\_distance}$  ) *
                (  $\frac{1}{distances[index]^2} * \frac{distances[index]*cos(index+90)}{distances[index]}$  )
        end
    end
    index ++
until index == 360;
total_x_repulsion+ = attraction_gain * (x_distance_to_goal)
Return total_x_repulsion , total_y_repulsion

```

3.5 Do Comportamento de Múltiplos Robôs

Considerando que o algoritmo de campos potenciais, por ser um algoritmo reativo, não é um algoritmo que garante o caminhamento. Eventuais batidas causadas pela repulsão do policial no ladrão e casos em que os robôs fiquem presos devido a algum mínimo local são considerados como eventos comuns ao jogo. Os mesmos são eventos caóticos e inerentes ao tipo de algoritmo. Podendo ser contornados em casos específicos por alteração de parâmetros do algoritmo, contudo a alteração de qualquer parâmetro gera outros mínimos e diferentes colisões por repulsão em outros cenários. Não havendo então um comportamento perfeito dada a própria natureza do algoritmo

4 EXPERIMENTOS

Nos experimentos realizados, os robôs vigilantes obtiveram mais sucesso que o ladrão. Em geral, quando um policial detecta um ladrão e inicia a perseguição, ele consegue prendê-lo. Existem situações nas quais a rota do policial cerca o ladrão, contra a parede ou outro policial, mesmo antes de detectar o invasor, devido ao comportamento de repulsão do ladrão para o policial.

No experimento da figura 3, o ladrão começa seu percurso seguindo reto em direção ao dinheiro, percebe um policial adiante e tenta desviar(frame 4). Seguindo sua rota, o policial detecta o ladrão e inicia a perseguição, que continua até o último frame quando o intruso é preso.

A figura 4 apresenta uma situação na qual o ladrão obteve sucesso, ele chega ao centro sem colisão com a rota do primeiro policial e

depois consegue desviar do seguinte sem ser percebido. Pode-se observar no último frame que o caminho para a saída está livre e o invasor está fora do campo de visão do vigilante.

O experimento da figura 5 demonstra a situação na qual o ladrão começou a ser cercado pelo policial mesmo antes de ser detectado, por causa da rota que o vigilante seguia. Após conseguir chegar ao centro sem ser detectado e iniciar a tentativa de fuga, o ladrão foi empurrado de volta para o centro e fica preso entre dois policiais.

4.1 Outros Experimentos Em Vídeo

Mínimo Escapável em Parede

Prisão Devido a Cercamento

Caminhamento dos Policiais

Prisão Devido a Mínimo Local

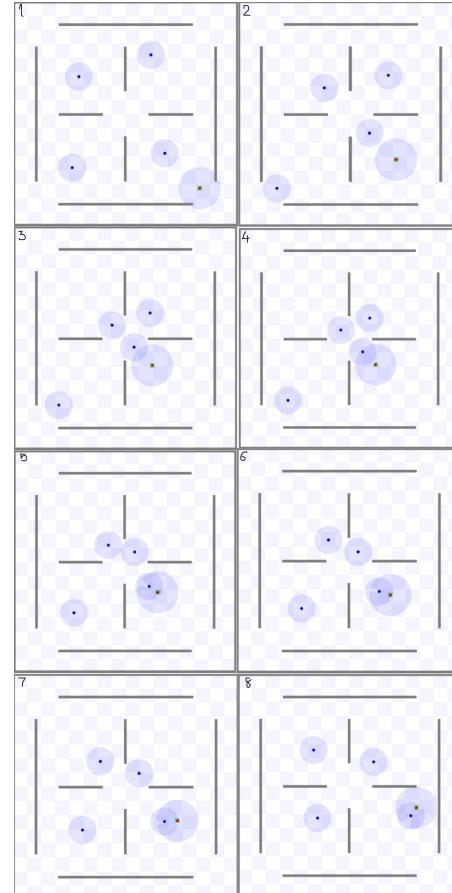


Fig. 3. Experimento 1

5 CONCLUSÃO

Ao longo do trabalho ficou clara a complexidade envolvida neste problema de perseguição e evasão. Também foi possível observar o surgimento de complexidade a partir de sistemas bastante simples. Como pudemos observar dentre os diversos experimentos realizados. A solução aqui apresentada é uma onde a captura do ladrão

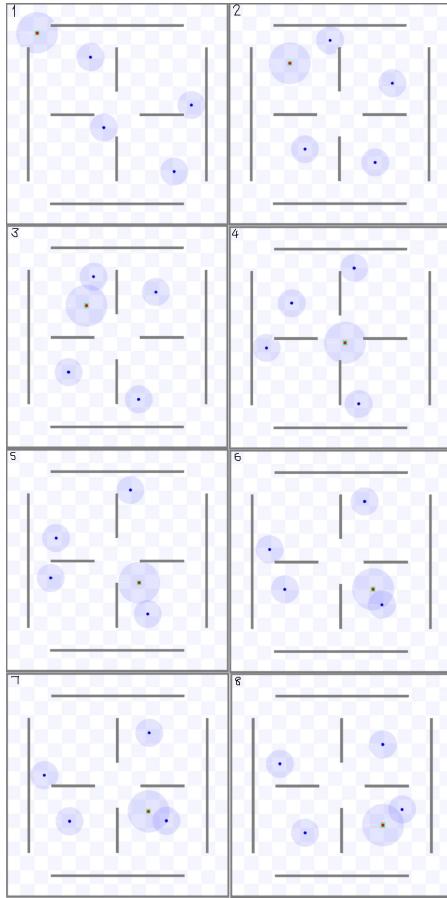


Fig. 4. Experimento 2

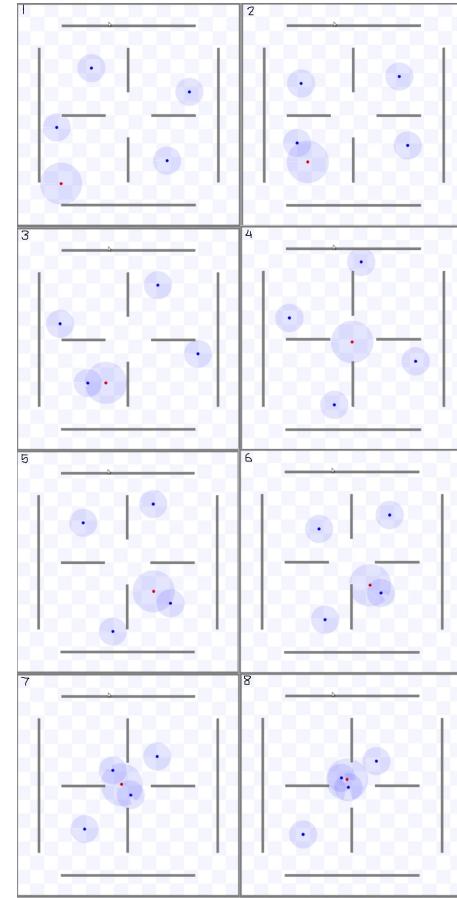


Fig. 5. Experimento 3

resulta mais do azar do que de uma decisão incorreta tomada por ele. Assim não sendo de maneira alguma um sistema completo para perseguição e vigilância. Contudo obtivemos resultados bem interessantes quanto à qualidade da perseguição pelos policiais e à capacidade de evasão à detecção pelo ladrão. Um possível trabalho futuro seria, uma vez que este trabalho foi desenvolvido de maneira completamente modularizada, adicionar mais ladrões, policiais e, talvez, mudar os cenários para entender se o comportamento observado não foi algo devido ao cenário e estejamos nos enviesando ao analisar o funcionamento do sistema.

A EXPERIMENTOS E ALTERAÇÕES POSSÍVEIS

Este artigo vem acompanhado dos códigos implementados para que possam ser executados em qualquer máquina executando o ROS (Robot Operating System).

Bastando executar o launch disponibilizado. Para alterar o ponto de início do ladrão basta pausar a execução com o comando P e alternar a posição do ladrão arrastando-o pelo mapa. Para retomar a execução basta apertar P novamente.

Os parâmetros a seguir podem ser modificados nos códigos dos robôs para obter outros comportamentos.

(1) attraction_gain

Relativo à constante multiplicativa para a atração ao objetivo (Todos)

(2) repulsion_gain

Relativo à constante multiplicativa para repulsão causada pelos obstáculos. (Todos)

(3) Repulsão aos policiais

linha 75 código de thief.py basta alterar o valor fixo 1000 por outro valor nos dois eixos x e y para obter outros comportamentos

(4) A rota dos policiais.

linha 236 de qualquer código police(0-3).py, basta descrever os pontos em ordem que definam a rota esperada.

(5) Visão dos robôs

Para alterar esses parâmetros alterações tem que ser feitas nos robôs definidos na pasta stage_models e após replicadas para a constante max_distance no caso dos policiais e no caso do ladrão max_distance = (distância definida)/1.5 como fica visível na linha 71 do código em thief.py

CONSPECTUS LIBRORUM

- [1] H. Moravec and A. Elfes (1985), "High resolution maps from wide angle sonar," *IEEE International Conference on Robotics and Automation*, St. Louis, MO, USA, 1985, pp. 116-121. doi: 10.1109/ROBOT.1985.1087316
- [2] R. B. Tilove (1990), "Local obstacle avoidance for mobile robots based on the method of artificial potentials," *IEEE International Conference on Robotics and Automation*, Cincinnati, OH, USA, 1990, pp. 566-571 vol.1. doi: 10.1109/ROBOT.1990.126041
- [3] Peng Song and V. Kumar (2002), "A potential field based approach to multi-robot manipulation," *IEEE International Conference on Robotics and Automation*, Washington, DC, USA, 2002, pp. 1217-1222 vol.2. doi: 10.1109/ROBOT.2002.1014709
- [4] P. Kondaxakis, S. Kasderidis and P. Trahanias (2008), "A multi-target tracking technique for mobile robots using a laser range scanner," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, 2008, pp. 3370-3377. doi: 10.1109/IROS.2008.4650705