Planejamento de Caminhos 3D: Comparação dos Algoritmos por Grade 3D e Grafo de Visibilidade

Cleiton Almeida dos Santos
Orientador: André Luiz Pires Guedes

Departamento de Informática - UFPR

20 de Setembro de 2019

Roteiro

A pesquisa

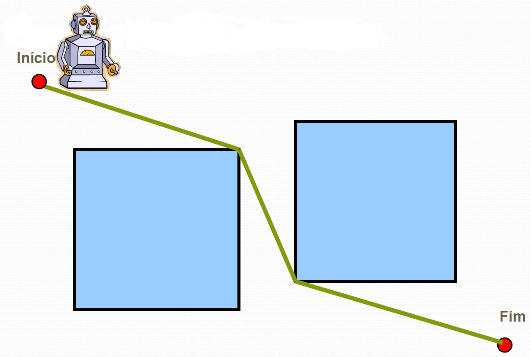
Objetivo: Encontrar algoritmos para planejamento de caminhos em ambiente 3D controlado, que gerem caminhos próximos do ótimo e com baixa complexidade computacional.

Andamento:

1. Identificar abordagens viáveis;
2. Escolher algumas abordagens;
3. Implementar escolhidas;
4. **Avaliar resultados;**
5. aplicar melhorias;
6. Avaliar algoritmos.

Em ambiente com obstáculos. Dados os pontos origem s e destino t.

- Caminho: Uma curva st, contida no espaço livre.
- Caminho ótimo: O melhor caminho. De menor custo.
- Planejamento de caminhos: Processo para encontrar.



Complexidade do algoritmo

- O conjunto de caminhos é infinito e não computável.
- Em 2D o melhor caminho passa pelos vértices dos obstáculos.
- Em 3D o melhor caminho passa pelas arestas dos obstáculos.
- Em 3D é NP-Difícil, prova em [Canny1988]. [1]

Contexto:

- Uma origem e um destino.
- O veículo está reduzido a um ponto.
- O veículo pode se movimentar em qualquer direção.
- Os obstáculos são poliedros convexos.
- Os pontos estão no $\mathbb{R}^3$.
- Ambiente controlado.

[1] Canny, J. (1988) The complexity of robot motion planning.MIT press.

Abordagens

Abordagens de algoritmos [Yang2016] [2]:

- Baseados em Amostragem
- Baseados em nós
- Baseados em Modelos Matemáticos
- Bioinspirados

[2] Yang, L., Qi, J., Song, D., Xiao, J., Han, J., e Xia, Y. (2016). Survey of robot 3D path planning algorithms. Journal of Control Science and Engineering, 2016:5.

Requisitos da solução

Os algoritmos de interesse precisam atender aos requisitos:

- Obter resultados de qualidade;
- Tempo de execução previsível;
- Ser aplicável a diferentes cenários.

Abordagens Descartadas:

- Amostragem: PRM, RRT e variações
- Bioinspirados: ACO, GA, PSO, etc.
- Modelos matemáticos: MILP, BIP
- Campos Potenciais Artificiais (APF)

Abordagens escolhidas

Discretização:

- Grafo de visibilidade
- Grade 3D

Algoritmos de busca de caminho mínimo:

- Dijkstra
- Floyd-Warshall

Grade 3D

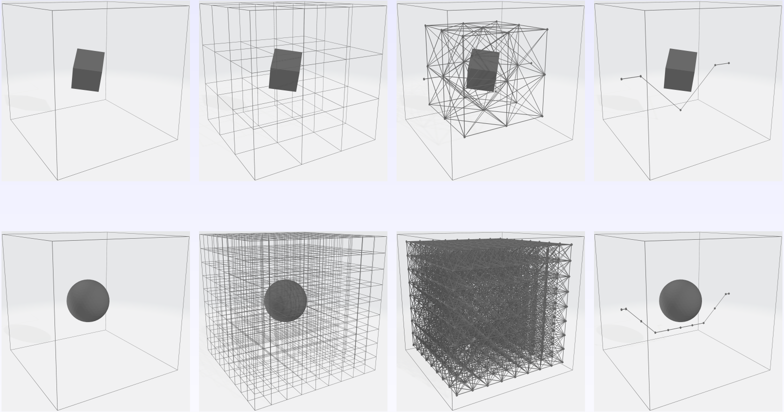
Particionar o ambiente em uma grade uniforme de pequenos volumes cúbicos, também chamados de *voxels*.

1. Marcar *voxels* como livres ou ocupados.
2. Gerar um grafo onde os *voxels* livres são vértices.
3. A vizinhança define as arestas do grafo (6 a 26)
4. Adicionar ao grafo vértices origem e destino
5. Conectar aos vértices dos *voxels* que contém.
6. Aplicar Busca de caminho mínimo em grafos.

Características:

- Caminhos com aparência serrilhada.
- O tamanho do *voxel* interfere na qualidade do caminho.
- O grafo gerado independe da complexidade dos obstáculos.
- Grau máximo do grafo conhecido.

Grade 3D



Visibilidade

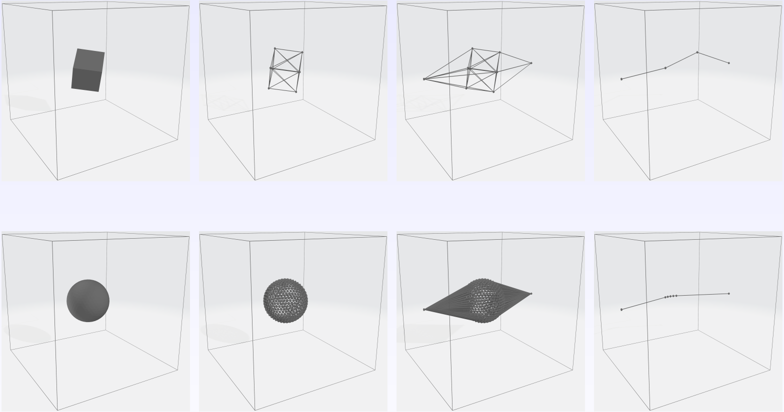
Construir um grafo a partir dos vértices dos obstáculos e da visibilidade entre eles. Dois pontos são visíveis entre si se um segmento de reta entre eles não atravessa obstáculos.

- Cada vértice do obstáculo é um vértice do grafo
- Adicionar ao grafo as arestas entre vértices visíveis
- Incluir no grafo os vértices origem e destino
- Conectar aos vértices visíveis.
- Aplicar busca de caminho mínimo em grafos.

Características:

- Caminhos mais diretos, poucas arestas, rente dos obstáculos.
- A complexidade dos obstáculos interfere no grafo.
- Grau máximo do grafo não é conhecido previamente.

Visibilidade



Comparação

Comparação entre as abordagens. Grade 3 x 3 x 3 (27 voxels, com 26 livres) e 9 x 9 x 9 (729 voxels, com 702 livres); Visibilidade Cubo (8 vértices, 18 arestas, 12 faces) e Esfera (642 vértices, 1.920 arestas, 1.280 faces)

		Grade		**Visibilidade Cubo**		**Visibilidade poliedro**	
		3x3x3	**9x9x9**	**Vértice**	**Aresta**	**Vértice**	**Aresta**
Grafo	**Vértices**	28	704	10	20	644	1.922
	Arestas	268	13.812	68	148	4.768	10.264
Caminho	**Vértices**	5	11	4	3	7	12
	Arestas	8	20	6	4	12	22
	Tamanho	1.108,5	1.025,7	906,3	895,4	912,8	912,6

Grade 3D

- Reduzir número de vértices do grafo - *Voxels* de tamanho variável. Pequenos perto dos obstáculos e grandes longe. Considerar *Octree* ou algum outro tipo de aglutinação.
- Reduzir o efeito serrilhado - Suavizar o caminho com pós-processamento, eliminar vértices intermediários e concatenar arestas adjacentes.

Visibilidade

- Construir grafo a partir das arestas dos obstáculos. aplicar um pós-processamento com uma espécie de busca binária sobre as arestas encontrar o melhor ponto de passagem.
- Pré-classificar os pares de vértices pelas normais das faces.
  - Ocluso - eclipsado por outro obstáculo.
  - Invisível - vértice(s) somente nas faces ocultas dos obstáculos.
  - Central - vértice(s) somente nas faces visíveis dos obstáculos.
  - Tangente - ambos na fronteira entre faces visíveis e ocultas.

Considerações finais

A abordagem por visibilidade parece ser mais promissora, mas a grade 3D não pode ser descartada.

Próximos passos:

- Implementar as otimizações e avaliar resultados.
- Encontrar estruturas de dados que melhorem o despenho.
- Verificar outras abordagens BSP, Delaunay, Voronoi, Fecho convexo.

Os problemas do mundo real necessitam de muitos graus de liberdade, recaindo nas dimensões mais elevadas. Em outros problemas da geometria computacional migrar do $\mathbb{R}^2$ para $\mathbb{R}^3$ é mais custoso do que migrar do $\mathbb{R}^3$ para $\mathbb{R}^d$, $d > 3$. A expectativa é que partindo de bons algoritmos para $\mathbb{R}^3$ se consiga lidar com problemas nas dimensões mais altas.