Acompanhamento

Incorporação de Sistemas Multiagente de Uso de Solo e Transporte na Geração de Cidades Artificiais

06/07/2022

Formalização: tempo médio de viagem

Tempo médio de viagem para dada célula c (acessibilidade).

Notação

- M: a rede viária, representada por um grafo orientado fortemente conexo
- C: conjunto de todas as células da grade
- E: conjunto de todas as arestas da rede viária
- $c_1, c_2, ..., c_{|C|} \in C$: células
- $e_1, e_2, ..., e_{|E|} \in E$: arestas
- V: conjunto de vértices
- $e = (e_s, e_t);\ e_s, e_t \in V$: nós de origem (source) e destino (target) de dado vértice e
- $C_e \subseteq C;\ e \in E$: conjunto de células que tem a aresta e como a mais próxima
- $e^c \in E$: aresta cuja célula mais próxima é c
- $v_e \in \mathbb{R}$: velocidade de deslocamento na aresta e
- v_0: velocidade de deslocamento base/fora da malha (definida pelo usuário, esperada menor que v_e para todo e)

Definições

Distância célula-aresta:

- Geometricamente, a aresta e é um segmento de reta entre as coordenadas de e_s e e_t
- $d(c, e)$ distância usual reta-ponto (métrica euclidiana)
- $p_c \in \mathbb{R}^2$ é o ponto mais próximo de c no segmento de reta que caracteriza e

Distância célula-malha:

- $d_M(c) = min_e\{d(c, e)\}$

- Calcula-se a distância entre a célula e todas as arestas e usa a menor como sua distância até a malha

Tempo de viagem célula-malha:

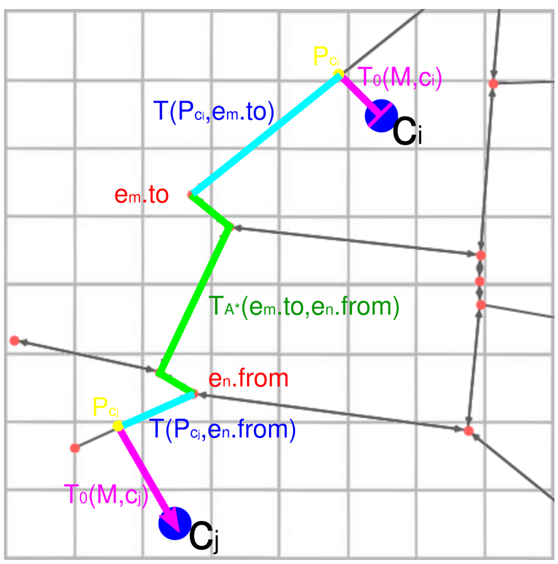
- $t_M(c) = \frac{d_M(c)}{v_0}$

Tempo de viagem célula-célula:

- Composto pelo tempo de viagem da célula de origem c_i até a malha, dentro da malha até chegar no ponto mais próximo da célula de destino (p_{c_j}) e da malha até a célula de destino c_j

$$t(c_i, c_j) = \frac{d_M(c_i)}{v_0} + \frac{d(p_{c_i}, e_t^{c_i})}{v_{e^{c_i}}} + Dijkstra(e_t^{c_i}, e_s^{c_j}) + \frac{d(p_{c_j}, e_s^{c_j})}{v_{e^{c_j}}} + \frac{d_M(c_j)}{v_0}$$

- Dijkstra já é feito na rede com tempos de viagem com pesos e pode ser substituído por outro algoritmo

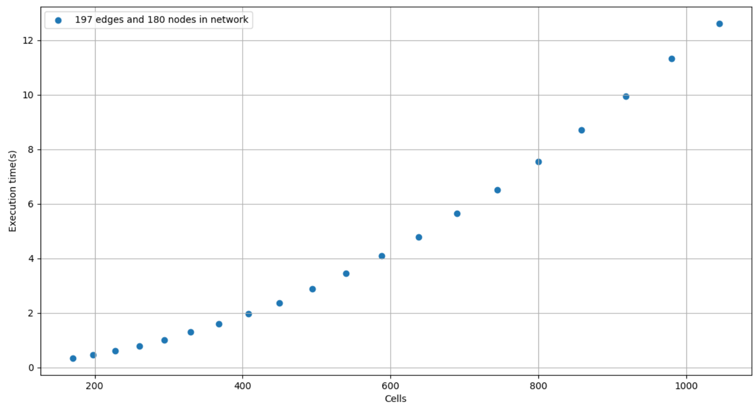


Derivação

Dada uma célula c, o tempo médio de viagem a partir de c para as demais células é dado por:

$$T(c) = \frac{1}{|C|} \sum_i t(c, c_i)$$

Como $t(c_i, c_j) \neq t(c_j, c_i)$, calcular o tempo médio de viagem para todas as células é $O(|C|^2)$.



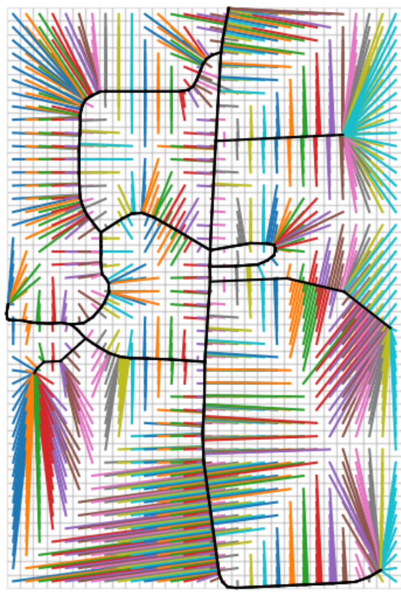
Constantes também são grandes já que como armazenamento na memória é inviável, muito deve ser recalculado em cada iteração

Reescrevendo:

$$T(c) = \frac{1}{|C|}\sum_i \left(\frac{d_M(c)}{v_0} + \frac{d(p_c, e_t^c)}{v_{e^c}} + Dijkstra(e_t^c, e_s^{c_i}) + \frac{d(p_{c_i}, e_s^{c_i})}{v_{e^{c_i}}} + \frac{d_M(c_i)}{v_0}\right)$$

Como dito anteriormente, o conjunto de células $C_e \subseteq C$ têm a mesma aresta e como a mais próxima.

Dada essa definição, temos que $\cup_e C_e = C$ e $C_{e_i} \cap C_{e_j} = \emptyset$ para todo $e_i \neq e_j$.



Faremos a seguinte distinção:

- Caso c_i e c_j pertençam à mesma vizinhança (ambos dividem a mesma aresta como mais próxima), então

$$t(c_i, c_j) = \frac{d_M(c_i) + d_M(c_j)}{v_0} + \frac{d(p_{c_i}, p_{c_j})}{v_e}$$

onde $v_{e^{c_i}} = v_{e^{c_j}} = v_e$.

Caso contrário, calculamos a distância como definido anteriormente.

Reescrevemos assim o tempo de viagem total como:

$$|C|T(c) = \sum_{i, c_i \notin C_{e^c}} \left(\frac{d_M(c)}{v_0} + \frac{d(p_c, e_t^c)}{v_{e^c}} + Dijkstra(e_t^c, e_s^{c_i}) + \frac{d(p_{c_i}, e_s^{c_i})}{v_{e^{c_i}}} + \frac{d_M(c_i)}{v_0}\right)$$

$$+ \sum_{i, c_i \in C_{e^c}} \left(\frac{d_M(c) + d_M(c_i)}{v_0} + \frac{d(p_c, p_{c_i})}{v_{e^c}}\right)$$

Tirando os termos constantes das somatórias:

$$|C|T(c) = (|C| - |C_{e^c}|)\left(\frac{d_M(c)}{v_0} + \frac{d(p_c, e_t^c)}{v_{e^c}}\right) + \sum_{i, c_i \notin C_{e^c}} \left(Dijkstra(e_t^c, e_s^{c_i}) + \frac{d(p_{c_i}, e_s^{c_i})}{v_{e^{c_i}}} + \frac{d_M(c_i)}{v_0}\right)$$

$$+|C_{e^c}|\frac{d_M(c)}{v_0} + \sum_{i,c_i \in C_{e^c}} (\frac{d_M(c_i)}{v_0} + \frac{d(p_c, p_{c_i})}{v_{e^c}})$$

$$= |C|\frac{d_M(c)}{v_0} + (|C| - |C_{e^c}|)\frac{d(p_c, e_t^c)}{v_{e^c}} + \sum_{i,c_i \notin C_{e^c}} (Dijkstra(e_t^c, e_s^{c_i}) + \frac{d(p_{c_i}, e_s^{c_i})}{v_{e^{c_i}}} + \frac{d_M(c_i)}{v_0})$$

$$+ \sum_{i,c_i \in C_{e^c}} (\frac{d_M(c_i)}{v_0} + \frac{d(p_c, p_{c_i})}{v_{e^c}})$$

Entre os termos, podemos notar que alguns independem da célula de origem. Em especial, podemos definir:

$$T_e = \sum_{c \in C_e} (\frac{d(p_c, e_s)}{v_e} + \frac{d_M(c)}{v_0})$$

de tal forma que

$$\sum_i (\frac{d(p_{c_i}, e_s^{c_i})}{v_{e^{c_i}}} + \frac{d_M(c_i)}{v_0}) = \sum_{e \in E} T_e$$

Além disso, o tempo de viagem dado por $Dijkstra(e_t^c, e_s^{c_i})$ só precisa ser calculado uma vez para cada par de vértices, uma vez que ele se repete para todas as células que compartilham a mesma aresta.

Assim reescrevemos a equação:

$$|C|T(c) = |C|\frac{d_M(c)}{v_0} + (|C| - |C_{e^c}|)\frac{d(p_c, e_t^c)}{v_{e^c}} + \sum_{e \in E, e \neq e^c} (T_e + |C_e| Dijkstra(e_t^c, e_s^{c_i}))$$

$$+ \sum_{i,c_i \in C_{e^c}} (\frac{d_M(c_i)}{v_0} + \frac{d(p_c, p_{c_i})}{v_{e^c}})$$

Como uma última simplificação, fazemos a seguinte manipulação:

$$\sum_{i,c_i \in C_{e^c}} (\frac{d_M(c_i)}{v_0} + \frac{d(p_c, p_{c_i})}{v_e^c}) = \sum_{i,c_i \in C_{e^c}} (\frac{d_M(c_i)}{v_0} + \frac{d(p_{c_i}, e_s)}{v_e^c} - \frac{d(p_{c_i}, e_s)}{v_e^c} + \frac{d(p_c, p_{c_i})}{v_e^c})$$

$$= T_{e^c} + \sum_{i,c_i \in C_{e^c}} (\frac{d(p_c, p_{c_i})}{v_{e^c}} - \frac{d(p_{c_i}, e_s^c)}{v_{e^c}})$$

Com isso podemos fazer as substituições na equação original afim de deixar a implementação mais direta:

$$|C|T(c) = |C|\frac{d_M(c)}{v_0} + (|C| - |C_{e^c}|)\frac{d(p_c, e_t^c)}{v_{e^c}} + \sum_{e \in E}(T_e + |C_e| Dijkstra(e_t^c, e_s^{c_i}))$$

$$+ \sum_{i, c_i \in C_{e^c}} (\frac{d(p_c, p_{c_i})}{v_{e^c}} - \frac{d(p_{c_i}, e_s^c)}{v_{e^c}}) - |C_{e^c}| Dijkstra(e_t^c, e_s^c)$$

O cálculo continua sendo $O(|C|^2)$, entretanto com constantes muito menores uma vez que o processo de cálculo da distância entre as células individuais só precisa ser feito entre células que videm a mesma aresta mais próxima.

O processo assume que a malha viária é um grafo é orientado e fortemente conexo.

Otimizações podem ser feitas, mas a princípio o processo deve ser repetido para cada atualização da rede viária.

Detalhes de implementação

Rede viária

- Rede de entrada é um arquivo da biblioteca *SUMO*, gerado a partir de um mapa *OpenStreetMap* (`.osm`)
  - Grafo orientado
  - Arestas possuem *geometria* (são um conjunto de segmentos de reta)
  - Velocidade constante nas arestas
- Abordagens para tratamento de geometria:
  1. Acrescentar vértices e arestas geométricas para considerar esses pontos (atual implementação,; mais simples e menos eficiente)
  2. Tratar a geometria das arestas na medida de distância célula-aresta (operações mais eficientes no grafo; menos vértices e arestas; vizinhanças de células maiores)
- Abordagem para tratamento de topologia (grafo **deve** ser fortemente conexo): CONTRAMÃO
  - Garantir que no mapa original todo vértice é origem ou destino de ao menos uma aresta
  - Para todo par de vértices v_a, v_b, caso haja uma aresta $e_m = (v_a, v_b) \in E$ (isto é, saindo de v_a e indo até v_b) mas $e_n = (v_b, v_a) \notin E$, introduzir aresta e_n
  - Velocidade em e_n pode ser determinada como menor que velocidade em e_m como estímulo à percorrer um caminho alternativo na mão correta, caso ele exista