

Heurística para o posicionamento de observadores em terrenos armazenados em memória externa

Salles Viana Gomes de Magalhães¹ and Marcus V. A. Andrade¹

Universidade Federal de Viçosa, Viçosa, Brasil
{smagalhaes,marcus}@dpi.ufv.br

Resumo The huge volume of more precise geographic data available has created great challenges for GIS community and it has demanded the development of more efficient algorithms that usually requires the processing of data in external memory. An important GIS application related to terrain processing is the facilities positioning such as mobile phone towers, fire observation towers, vigilance systems, etc. This paper presents an efficient method to site “observers” in terrains represented by a huge digital elevation matrix stored in external memory. As shown by the practical results, the proposed method can solve the problem using, in average, 15% less observers than other ones.

Key words: siting, terrain visibility, facilities positioning.

1 Introdução

Os avanços tecnológicos no processo de coleta de dados sobre a superfície terrestre têm disponibilizado um enorme volume de dados geográficos, o que tem exigido o desenvolvimento de técnicas mais avançadas para o processamento desses dados utilizando os SIG (Sistemas de Informações Geográficas) [1].

Normalmente esses dados são armazenados sob a forma de modelos digitais de elevação (MDE), que são representações aproximadas do terreno real. O MDE é representado por uma matriz que armazena a elevação de pontos do terreno amostrados de forma regular (isto é, com um espaçamento horizontal e vertical uniforme). Neste caso, a elevação dos pontos que não pertencem à amostra é obtida através de um processo de interpolação [2,3]. Este método de representação é bastante simples, porém requer um grande espaço de armazenamento e, muitas vezes, eles não podem ser processados na memória interna dos computadores. Assim, é importante desenvolver aplicações eficientes para o processamento em memória externa.

É importante observar que o projeto de algoritmos que manipulam dados em memória externa deve ser realizado privilegiando o menor número de acessos a esta memória externa, pois o custo de acesso a esta memória é muito maior do que o da memória interna. Mais precisamente, o objetivo principal no projeto destes algoritmos é minimizar o número de acessos à memória externa visto que este é o fator determinante no tempo de processamento do algoritmo.

Dentre as várias aplicações em SIG envolvendo os dados de um terreno, há um grupo importante de aplicações relacionadas ao conceito de visibilidade que consiste em determinar quais as “partes” (os pontos) de um terreno são visíveis a partir de um observador posicionado em um determinado local. Essas aplicações relacionadas à visibilidade são de grande utilidade prática em áreas como: telecomunicações, planejamento ambiental, navegação de veículos autônomos, monitoramento militar, etc [4,3,5] e um problema importante nessas áreas é o posicionamento ótimo de um número limitado de recursos de modo a “cobrir” da melhor forma possível um terreno. Esses recursos podem ser antenas de transmissão de rádio, TV, internet e telefonia celular, torres de observação de incêndios florestais, vigilantes, entre outros [6,7].

A proposta deste trabalho é descrever o desenvolvimento e implementação de um método para posicionamento de recursos em terrenos representados por grandes matrizes de elevação armazenadas em memória externa. Este trabalho é baseado no método proposto por Franklin et al. [8,5] para o posicionamento de observadores em terrenos representados por matrizes de elevação armazenadas em memória interna.

2 Problemas Envolvendo Visibilidade em Terrenos

2.1 Determinação de Visibilidade

Dada uma região M representada por uma matriz de elevação e dados dois pontos P e T pertencentes a M , seja A um alvo posicionado no ponto T e seja O um observador situado em P cujo raio de visão é R (ou seja, o alcance máximo da visão é R)¹. O alvo A é dito visível a partir de O se a distância entre O e A for menor ou igual a R e se não existir nenhum ponto do terreno entre P e T tal que a altura desse ponto seja maior do que a altura do segmento de reta que liga O e A . O segmento ligando os pontos O e A é chamado Linha de Visão e é denotado por LOS (*Line of Sight*). Veja a figura 1. Nessa figura, o ponto p_1 é considerado visível a partir do observador O posicionado em p_0 pois não existe nenhum ponto entre p_0 e p_1 que intercepta a LOS Op_1 . Já p_2 não é visível por haver um ponto entre p_0 e p_2 com elevação maior do que a elevação da LOS Op_2 .

2.2 Índice de Visibilidade

O índice de visibilidade, denotado por vis (de *visibility index*) de um observador O com raio de visão R posicionado num ponto P indica a porcentagem da área de um círculo de raio R , centrado em P , que pode ser vista por O . O valor do vis pode ser usado para determinar os melhores candidatos para se posicionar observadores de modo que eles cubram uma grande área visível.

¹ O alvo A e o observador O podem estar localizados a uma certa altura acima do terreno e estas alturas podem ser diferentes.

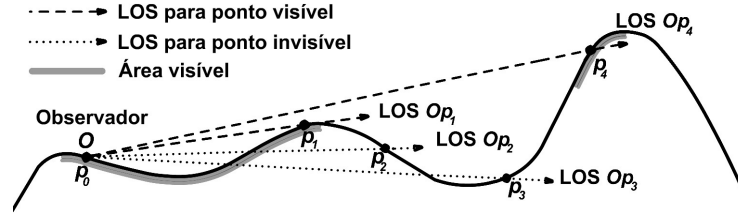


Figura 1. Determinação de visibilidade com o uso de linhas de visão

2.3 Viewshed

Dado um ponto p e um disco de raio r centrado em p , o *viewshed* de p com raio de visão r , denotado por $VS(p, r)$, corresponde ao conjunto de pontos do disco que são visíveis por um observador posicionado em p .

O *viewshed* de um ponto pertencente a um terreno T de tamanho $n \times n$ normalmente é representado por uma matriz cujos elementos possuem valores 0 (para pontos não visíveis) ou 1 (para pontos visíveis).

2.4 Posicionamento de Recursos

Um problema prático importante relacionado à visibilidade é a otimização do posicionamento de recursos como torres de telefonia celular, postos de observação de incêndio, sistemas de radar, etc que podem ser interpretados como o posicionamento de observadores representados por seus respectivos *viewsheds*.

Como demonstrado em [4], este problema é NP-Completo e se reduz a um problema clássico na área de otimização que é a cobertura de conjuntos [9].

Neste trabalho, vamos considerar a seguinte versão do problema: dada uma taxa de cobertura a ser alcançada, desejamos utilizar o número mínimo de observadores para atingir esta cobertura. Como o problema é NP-Completo, vamos obter uma solução aproximada, onde vamos posicionar um número de observadores próximo do mínimo para alcançar a cobertura desejada.

Vale mencionar que a heurística proposta pode ser adaptada para resolver uma variação desse problema que consiste em cobrir a maior² área possível utilizando, no máximo, uma quantidade pré-definida de observadores.

3 O método Site para posicionamento de recursos

Visto que o problema de posicionamento de recursos é NP-Completo então uma estratégia interessante é utilizar uma heurística para obter uma solução aproximada para o problema. Com base nisso, Franklin e Ray [5] desenvolveram o método Site adaptando a estratégia gulosa utilizada na solução do problema de

² Também neste caso, a solução obtida seria uma aproximação.

cobertura de conjuntos para utilizá-la na solução do problema de posicionamento de observadores. A idéia básica deste método é a seguinte: dado um conjunto S de pontos do terreno candidatos a receberem observadores, calcule o *viewshed* de todos os pontos de S e, então, selecione de forma gulosa o ponto de S cujo *viewshed* mais contribui para melhorar a visibilidade atingida pela solução atual.

Mais especificamente, o método calcula o *viewshed* de todos os candidatos a observadores e, então, cria uma matriz de visibilidade V que representará a visibilidade do terreno obtida pela solução atual, sendo que essa matriz é inicializada com todos os pontos marcados como não visíveis. Então, o processo realiza uma iteração na qual, em cada passo, é selecionado o observador que mais contribui para o aumento da área visível de V . Este processo é repetido até se atingir o número máximo de observadores desejados ou até que a adição de novos observadores não aumente a área visível representada por V .

Para determinar qual observador mais contribui para o aumento da área visível de V , é utilizada uma matriz de visibilidade temporária V' correspondente à união entre V e o *viewshed* de cada ponto candidato a observador. Assim, obtendo os valores de V' para todos os observadores ainda não utilizados, será escolhido o observador que produza o V' com a maior área.

É importante salientar que o *viewshed* de cada candidato a observador é mantido em memória principal e, portanto, é inviável utilizar todos os pontos do terreno como candidatos a observador, pois o *viewshed* de um observador em um terreno de dimensões 1201×1201 , por exemplo, ocuparia 176KB de memória e o armazenamento em memória dos *viewsheds* de todos pontos desse terreno ocuparia 242GB de memória (o que é inviável na maioria dos computadores). Além do problema do espaço em memória ocupado pelos *viewsheds*, o cálculo dessas estruturas e o posicionamento dos recursos seria muito ineficiente pois todos os pontos do terreno teriam que ser avaliados. Para contornar esse problema, o Site seleciona um número pré-definido de pontos que têm a maior probabilidade de serem bons candidatos a receberem observadores e estes pontos são inseridos no conjunto S .

A escolha desses pontos para formar o conjunto S é feita com base no índice de visibilidade (*vi x*) dos pontos do terreno. Isto é, o Site calcula o *vi x* aproximado de todos os pontos do terreno e, então, seleciona os pontos que possuem os maiores *vi x* . Para evitar a concentração de muitos pontos em áreas particulares do terreno (áreas com alta taxa de visibilidade) e a consequente ausência de pontos em outras áreas, o terreno é subdividido em m terrenos menores e, então, são selecionados os K/m pontos que possuem os maiores índices de visibilidade em cada subdivisão do terreno.

3.1 Uso do Site para posicionamento de observadores em grandes terrenos

Uma maneira simples de se adaptar o Site para aplicá-lo a terrenos que não cabem na memória principal seria subdividir o a matriz de elevação em matrizes menores que possam ser processadas em memória interna e, então, realizar o posicionamento utilizando o Site nessas matrizes.

Mais especificamente, seja T um terreno representado por uma matriz de elevação de dimensão $n \times n$. Suponha que seja possível realizar, na memória interna, o posicionamento de recursos em uma matriz de dimensão $m \times m$ (sendo $m < n$). Assim, a matriz T é subdividida em k submatrizes menores T_1, T_2, \dots, T_k de dimensões $m \times m$. Daí, o Site pode ser aplicado a cada matriz individualmente.

Essa abordagem geralmente não gera resultados satisfatórios pois, ao processar um terreno T_i , não é considerada a influência da visibilidade de observadores posicionados próximos às bordas de um terreno T_j vizinho a T_i . Ao desconsiderar essa influência, são geradas redundâncias em áreas próximas às bordas e, além disso, o algoritmo guloso pode deixar de posicionar algum observador em uma área próxima às bordas de um terreno T_d pois parte do *viewshed* desse observador estaria fora de T_d e, portanto, não adicionaria muita área visível à subdivisão atual do terreno (apesar dele poder adicionar área visível a outro terreno T_c , vizinho a T_d).

Para ilustrar este problema, veja a figura 2. Nela, o terreno foi dividido em quatro terrenos T_1, T_2, T_3 e T_4 e um observador foi posicionado próximo às bordas de T_2 . Note que o *viewshed* desse observador influencia a visibilidade de todos os terrenos. Se o posicionamento de observadores for feito em cada terreno individualmente, apenas a visibilidade adicionada a T_2 seria considerada.

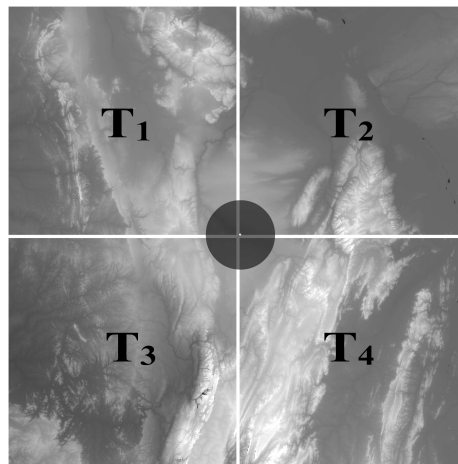


Figura 2. Divisão de um terreno em quatro terrenos menores

Um inconveniente ainda mais importante dessa solução é que em cada subdivisão seria necessário atingir a taxa de cobertura desejada para que ao final o terreno como um todo seja coberto como desejado. Isso pode levar a soluções muito ruins (ou até mesmo impedir a obtenção da solução) caso alguma subdivisão seja de difícil visibilidade.

4 O algoritmo proposto

O algoritmo proposto neste trabalho, chamado de *EMSite* (*External Memory Site*) estende a idéia descrita na Seção 3.1 de modo a considerar a influência dos observadores posicionados próximos às bordas dos terrenos subdivididos.

A idéia utilizada consiste em dividir os terrenos em regiões menores do que a maior região que pode ser processada em memória interna e adicionar a essas regiões uma borda do tamanho do raio de visão (a submatriz aumentada com a borda deve possuir tamanho suficiente para processamento em memória interna). Os observadores somente serão posicionados na área do terreno que não inclui a borda, mas a visibilidade deles é considerada tanto na área que não pertence às bordas quanto nas próprias bordas. Veja a figura 3. A região indicada por T_2 corresponde a uma subdivisão do terreno e a área indicada por A corresponde à borda adicionada a T_2 . Note que observadores só podem ser posicionados em T_2 , mas a visibilidade deles pode atingir a área A . Com isso, toda visibilidade do observador posicionado nesse terreno é considerada durante o posicionamento.

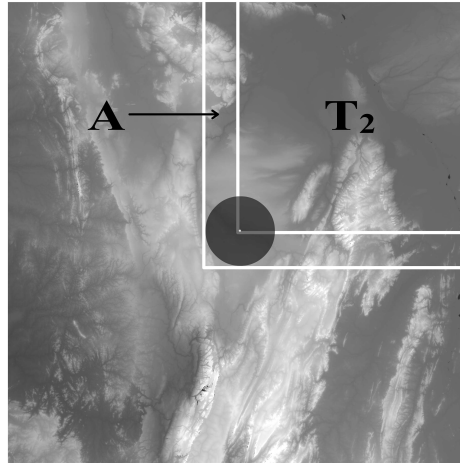


Figura 3. Adição de borda extra a T_2

Para representar a visibilidade do terreno, uma matriz V é mantida em disco sendo que esta matriz é inicializada com todos os pontos indicados como não visíveis (situação inicial do terreno). Daí, o método *Site* é aplicado a cada região T_i (acrescida da respectiva borda) e a matriz de visibilidade gerada pelo *Site* é sobreposta à matriz V que está armazenada em disco. É importante observar que ao final do processamento de cada região a matriz V é atualizada e, no início do processamento de uma região T_i , o método *EMSite* utiliza a região da matriz V corrente correspondente à região T_i (acrescida da borda).

Como critério de parada, o algoritmo avalia duas características da solução corrente: o número de observadores posicionados e a área de cobertura atingida.

Assim, o algoritmo termina quando é atingido o número máximo de observadores ou quando a área coberta do terreno é maior ou igual à taxa de cobertura desejada.

Para tentar alcançar a cobertura desejada usando um menor número de observadores, uma boa estratégia gulosa é iniciar o posicionamento dos observadores nas regiões do terrenos onde o posicionamento gere melhor resultado. Para determinar os terrenos mais promissores, num primeiro passo é feito o posicionamento de k observadores em cada subdivisão do terreno. Com essa amostragem, é definida a ordem em que essas subdivisões serão processadas, ou seja, as subdivisões onde a adição dos observadores gerou as maiores taxas de cobertura serão as primeiras onde novos observadores serão posicionados.

Além disso, a distribuição dos observadores em cada subdivisão do terreno também leva em consideração uma variável δ cujo valor define a contribuição mínima que cada observador deve adicionar à região visível. Assim, no primeiro passo, a heurística adiciona observadores numa região enquanto não forem adicionados k observadores e enquanto a contribuição de cada observador for maior do que δ . No primeiro passo, o valor de δ é pré-definido com base na área de um *viewshed* completo (ou seja, um *viewshed* totalmente visível).

Se ao final do primeiro passo, o EMSite não atingir o critério de parada (isto é, se não atingir a cobertura desejada e se não houver posicionado o número máximo permitido de observadores), então o método é repetido até que esse critério seja satisfeito. Nesse caso, o valor de δ é reduzido para que o método adicione mais observadores em cada região. Isto é, em cada iteração, δ é multiplicado por um valor menor do que 1, diminuindo, assim, a exigência de área visível mínima para os observadores adicionados. Além disso, em cada iteração, a ordem de processamento das subdivisões do terreno é recalculada considerando o tamanho da área visível adicionada em cada região pelo último observador posicionado naquela região na iteração anterior. Dessa forma, as subdivisões mais promissoras são sempre atualizadas e são as primeiras a serem processadas.

5 Modificações propostas no método Site

Para melhorar a eficiência do algoritmo Site original, foram realizadas algumas modificações neste método para diminuir o espaço necessário para se armazenar os *viewsheds* dos observadores. A modificação proposta consiste em armazenar apenas o *bounding-box* do *viewshed*. Por motivos de simplicidade, considera-se como *bounding-box* o quadrado B centrado no observador e com lado $2r - 1$, onde r é o raio de visão do observador. Para facilitar o processamento do *viewshed*, a matriz B é codificada em bits de forma a armazenar um conjunto de 8 pontos em 1 byte.

Além disso, para facilitar a sobreposição de um *viewshed* ao mapa de visibilidade durante o posicionamento, foi proposto um formato de arquivo onde o *viewshed* é armazenado. Esse formato inclui um cabeçalho que descreve o número de bytes ocupados por cada linha, as coordenadas do *bounding-box* que contém o

viewshed, o raio de visão do observador, as coordenadas do observador relativas ao *bounding-box* e as coordenadas do observador relativas ao terreno original.

Com tais modificações, o método requer uma menor quantidade de memória para realizar o posicionamento e, além disso, este posicionamento pode ser feito de forma mais eficiente, pois uma menor quantidade de dados será processada.

6 Resultados

Para avaliar os resultados obtidos pelo EMSite, ele foi implementado em C++ e testado utilizando-se terrenos pertencentes ao nordeste do Brasil. De acordo com os nossos conhecimentos, não há um outro método (já implementado) para posicionamento de observadores em terrenos capaz de processar dados em memória externa. Assim, o EMSite foi comparado com uma versão modificada do Site original similar à descrita na seção 3.1.

Os terrenos utilizados foram obtidos na página do SRTM [10]:

- Terreno 1: Terreno pertencente à região nordeste do Brasil, possui dimensão 4804×4804 .
- Terreno 2: Terreno pertencente à região nordeste do Brasil, possui dimensão 4804×4804 sendo que a elevação dos pontos no centro desse terreno foi modificada artificialmente para produzir uma região onde a elevação possui valor 0. Isso foi feito para simular um terreno com uma planície no centro, pois supomos que esta situação pode trazer dificuldades para que a cobertura desejada seja alcançada.

As tabelas 1 e 2 mostram os resultados obtidos pelos métodos EMSite e Site considerando diferentes taxas de coberturas desejadas com os observadores posicionados respectivamente a $15m$ e $1m$ acima do solo. Foram considerados raios de visão com valores de 100 e 250 pontos (a coluna *RV*). A coluna *# Obs* mostra o número de observadores posicionados pelo respectivo método para atingir a taxa de cobertura. Os métodos foram configurados para dividir os terrenos em 16 regiões e vale mencionar que, apesar dos terrenos originais possuírem 44 MB, eles não podem ser processados em memória interna pelo Site original porque, durante o processamento, esse método armazena muitos *viewsheds* na memória interna.

Como os resultados sugerem, o EMSite foi bem mais eficiente do que o Site adaptado pois conseguiu atingir a cobertura desejada utilizando menos observadores - em média, cerca de 15% menos - principalmente com maiores taxas de cobertura desejadas. Em particular, veja a tabela 2: para 88% de cobertura desejada, o EMSite alcançou esta cobertura utilizando menos da metade dos observadores utilizados pelo Site.

Uma outra vantagem importante do EMSite é que, em determinadas situações (como por exemplo, 90% de cobertura desejada na tabela 2), o método Site não pode alcançar a cobertura desejada.

Terreno	RV	Cobertura Desejada	Site	EmSite
			# Obs	# Obs
Terreno 1	100	25%	202	197
		50%	428	408
		75%	754	831
		85%	1338	1037
Terreno 2	100	25%	200	195
		50%	416	401
		75%	791	729
		85%	1239	1017
Terreno 1	250	75%	166	137
		90%	293	237
Terreno 2	250	75%	208	168
		90%	257	210

Tabela 1. Comparação entre número de observadores posicionados pelos métodos EMSite e Site para atingir a taxa de cobertura definida na coluna Cobertura Desejada considerando observadores posicionados a 15m acima do solo.

Terreno	RV	Cobertura Desejada	Site	EmSite
			# Obs	# Obs
Terreno 1	250	25%	113	84
		50%	338	291
		80%	1964	1574
		88%	11866	5424
		90%	—	13259

Tabela 2. Comparação entre número de observadores posicionados pelos métodos EMSite e Site para atingir a taxa de cobertura definida na coluna Cobertura Desejada considerando observadores posicionados a 1m acima do solo.

7 Conclusão e trabalhos futuros

Este trabalho apresenta a descrição e implementação do método EMSite que é capaz de realizar o posicionamento de recursos em terrenos representados por matrizes de elevação cujos tamanhos impedem que o processamento seja realizado em memória interma. Com base nos testes realizados, podemos concluir que o EMSite apresenta bons resultados e é capaz de processar grandes terrenos que não poderiam ser processados utilizando os métodos convencionais. Além disso, se comparado ao método Site adaptado para processar terrenos em memória externa utilizando a subdivisão (convencional) do terreno, o EMSite é capaz de alcançar uma determinada taxa de cobertura utilizando cerca de 15% menos observadores. Esta economia, em diversos casos, pode representar um ganho considerável principalmente quando os recursos a serem posicionados envolvem altos valores financeiros (como por exemplo, torres de telefonia celular).

O código fonte do EMSite está disponível em <http://www.dpi.ufv.br/marcus/projects/EMSite.htm>.

Como trabalhos futuros, o próximo passo é adequar o método EMSite de modo a permitir que o método seja capaz de processar observadores com grandes raios de interesse que geram *viewsheds* de grandes tamanhos que não podem ser armazenados em memória interna.

Agradecimentos

Este trabalho foi parcialmente financiado com recursos da FAPEMIG - Fundação de Amparo à Pesquisa do Estado de Minas Gerais através do projeto APQ-00329-08 - HidroExt - Hidrografia em terrenos armazenados em memória externa e também pelo CNPq - Conselho Nacional de Pesquisa.

Referências

1. R. Laurini and D. Thompson, *Fundamentals of Spatial Information Systems*, Academic Press, 1992.
2. C. A. Felgueiras, “Modelagem numérica de terreno,” in *Introdução à Ciência da Geoinformação*, A. M. V. M. In G. Câmara, C. Davis, ed., **1**, INPE, 2001.
3. Z. Li, Q. Zhu, and C. Gold, *Digital Terrain Modeling - principles and methodology*, CRC Press, 2005.
4. G. Nagy, “Terrain visibility,” *Computers and Graphics* **18**, pp. 763–773, 1994.
5. W. R. Franklin and C. Ray, “Higher isn’t necessarily better - visibility algorithms and experiments,” in *6th Symposium on Spatial Data Handling*, (Edinburgh, Scotland), 1994.
6. D. G. Tarboton and D. P. Ames, “Advances in the mapping of flow networks from digital elevation data,” in *World Water and Environmental Resources Congress*, pp. 107–111, 2001.
7. M. A. Chaves, *Modelos digitais de elevação hidrológicamente consistentes para a bacia amazônica*. Phd thesis, Universidade Federal de Viçosa, Departamento de Engenharia Florestal, 2002.
8. W. R. Franklin and C. Vogt, “Tradeoffs when multiple observer siting on large terrain cells,” in *12th International Symposium on Spatial Data Handling*, 2006.
9. N. V. de Oliveira., “Problemas de coberturas de conjuntos, uma comparação numérica de algoritmos heurísticos,” tese de mestrado em engenharia de produção, Universidade Federal de Santa Catarina - UFSC, 1999.
10. The Shuttle Radar Topography Mission (SRTM), “www2.jpl.nasa.gov/srtm,” (acessado em maio de 2009).