Procedia Computer Science

Volume 51, 2015, Pages 356–365

ICCS 2015 International Conference On Computational Science

Siting Multiple Observers for Maximum Coverage: An Accurate Approach

A.R. Cervilla, S. Tabik, and L.F. Romero

Dpt. Computer Architecture, University of Malaga
29071 Malaga, Spain
amrocer@uma.es stabik@uma.es felipe@uma.es

Abstract

The selection of the minimal number of observers that ensures the maximum visual coverage over an area represented by a digital elevation model (DEM) have great interest in many fields, e.g., telecommunications, environment planning, among others. However, this problem is complex and intractable when the number of points of the DEM is relatively high. This complexity is due to three issues: 1) the difficulty in determining the visibility of the terrain from one point, 2) the need to know the visibility at all points of the terrain and 3) the combinatorial complexity of the selection of observers.

The recent progress in total-viewshed maps computation not only provides an efficient solution to the first two problems, but also opens other ways to new solutions that were unthinkable previously. This paper presents a new type of cartography, called the masked total viewshed map, and provides optimal solutions for both sequential and simultaneous observers location.

Keywords: maximun coverage, optimal located of observes, total visibility model

1 Introduction

The last decade has know the continuous creation of large numbers of DEMs (Digital Elevation Models) of terrains and surfaces generated by photogrammetry and LIDAR (Laser Imaging Detection and Ranging) clouds of points. This data allows the characterization of the surface heterogeneity with very high spatial resolutions, on the order of 1 cm^2. Indeed, there is an increasing public and private demand for higher resolution DEMs of very large volumes. This fact has produced a high interest for developing new efficient algorithms able to extract useful information from these grids. For instance, the viewshed model [11], solar irradiation model [12, 10], win model, horizon model [13] and so forth. The existence of these algorithms allows the opportunity to improve and create new useful real life applications which were not possible before. Examples include, the determination of the minimum number of surveillance cameras for early fire detection in industrial areas, the determination of surveillance towers in natural areas and national parks. However, there is no algorithms or tools able to determine

 Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2015

doi:10.1016/j.procs.2015.05.255

the optimal location of the minimum number of observers to allow the maximum coverage.The design of algorithms to solve the problem of maximum converge on DEMs with high resolution is complex due to the huge volume of the data to be processed, which make it a *Big data* problem. It was demonstrated that it is NP-complete problem [8] and it can be reduced to a maximum coverage optimization problem [2].

As far as we know, there exist no tool in the market and literature to determine with reliability an optimal solution to this problem. There are heuristics that search the optimal set of observers for the commonly called *multi-observer siting problem*: i) local search methods that reduce the areas around the observers in a small radius in order to overcome the computational requirements [9, 5] ii) greedy search methods that initially select a small set of candidates and successively incorporate new points to that set until they cover the whole area of interest [9, 4], iii) genetic algorithms [7] and iv) *simulated annealing* methods[3]. In addition, there are a number of works that use either random or exhaustive search strategies [1].

Our algorithm, and also its accuracy, in comparison with all related works, relies on considering that all the points of the area of interest in a DEM are potential observers. We exhaustively selects the set of observers with the highest total-viewshed and iteratively finds among that set a sub-set that provides the maximum coverage. The proposed method can be used to improve the surveillance in fire-fighting, anti-theft, environment control, occluded path calculation, mobile phone coverage and radio or television communications among other examples. The main objective of this work is to provide a tool that finds the optimal placement of a set of observers that give the maximum coverage.

2 Mask based Total-Viewshed

The proposed algorithm for the multi-observer siting problem is a variation of the algorithm to calculate the total viewshed map developed by the authors in a previous work [11]. A total-viewshed map indicates the number of visible points at each and all the points of terrain.

Total viewshed map may be useful in many fields, however in many other cases, there are parts of the terrain in which there is few or null interest for the observers. In mobile cell planning, for example, there are areas without population. For instance, in fire–fighting, there is no need to have visual coverage of the sea. For theses cases, we propose in this work a new cartography: the mask based total-viewshed map, which can be easily calculated by including few additional operations in the total viewshed algorithm.

2.1 Total-Viewshed Algorithm

The total–viewshed algorithm developed by the authors and recently published in [11] calculates the visibility of every point of the map in a discrete set of line of sights (usually, in 360 equally–distributed azimuthal directions). For a given point of view (POV) and a given direction S, the algorithm determines the visibility of point POV by means of a sequential analysis of every point P in the line of sight determined by S, until the bounds of the DEM. For each encountered point P, the *analyzePoint* function is performed, see Algorithm 1.

This function determine if point P is visible or not, in order to detect visible stretches in the line of sight. Then, it computes the total visible surface $totalSurface$ from every point POV in the DEM as a function of the stretch location.

Algorithm 1: visibility kernel

```
point POV;
float visible_stretch;
float distanceStar, maxangle;

analyzePoint(point P){
        float distance = P.d − POV.d;
        float height = P.h − POV.h;
        float angle =height / distance;
        bool this_visible = angle > maxangle

        bool startStretch = this_visible && !visible_stretch
            //First point of a visible segment found
        if startStretch then
            storeStartStretch(P) // Save stretch geometry
            distanceStart=distance;
        end

        bool endStretch = !this_visible && visible_stretch
            //First point of a non visible segment found
        if endStretch then
            storeEndStretch(P); // Save stretch geometry
            totalSurface += evalSurface(distanceStart, distance);
        end

        visible_stretch = this_visible
        maxangle = max(angle,maxangle)
        }
```

2.2 Masked Total-Viewshed Algorithm

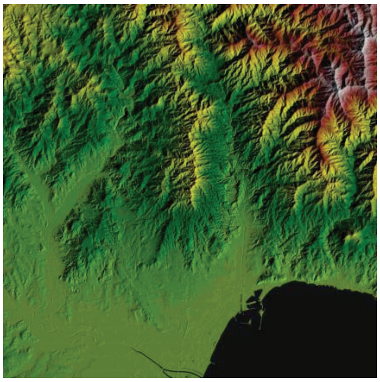
We propose to add a new line in *analyzePoint* (as a last line) for every point P in a visible stretch, so that, if the point does not belong to the area of interest in the landscape (we call it an *unmasked* point), then we subtract the surface related of that point from $totalSurface$.

if(P.unmasked&&this_visible) $totalSurface- = evalSurface(distance)$

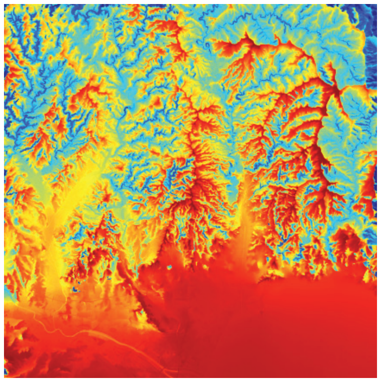
Note that these additional floating point operations added to the function would increase the cost of the algorithm. But there is another operation (a division) in every call to *analyzePoint* function which is very expensive in CPU cycles, and which will be performed in parallel with the new operations, by considering the separate functional units and out-of-order execution in the core microarchitecture employed in our experiments (intel Haswell) [6]. Experiments have also shown that the new lines is performed without any increase of the CPU time.

Moreover, the boolean mask can be easily replaced by a byte–size integer which could also indicate a graded interest of the point. In fact, a negative value for the parameter can also indicate that the point has special relevance for the observer.

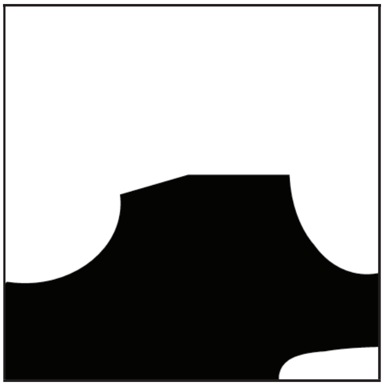
As a result, a masked total viewshed map is built, which indicates the surface *of interest* that is seen from every point. In Figure 1(d)) we show the masked total viewshed map for a sample DEM (Figure 1(a))), in which the mask Figure 1(c)) has been applied. Note the differences with the original total viewshed map (Figure 1(b)))



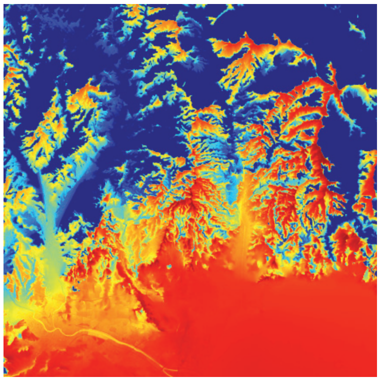
(a) DEM of the city of Malaga



(b) Total-viewshed map



(c) Mask (#0) for areas of interest, abour 39%



(d) Masked Total Viewshed Map

Figure 1

The masked total viewshed map shown in the figure provides an excellent tool to calculate the place for a single observer: we put it in the absolute maximum of the map.

3 The algorithm for Siting Observers

The algorithm proposed in this section iteratively determines the optimal set of observers that guarantees the required coverage with a low computational cost.

In a first iteration, the total-viewshed is calculated on an initial area of interest, called mask, then the observer that produces the highest coverage is selected. Successively the same approach is performed on an updated mask until the maximum coverage is obtained.

The input of this method are 1) the DEM that we want to analyze, an example is shown in Figure 1(a)). 2) An initial mask (MASK_0) is provided, which indicates the areas of interest, i.e., where coverage is necessary. The mask is an useful tool to exclude the regions where the interest is null or very low, because the coverage in that regions was guaranteed by a previous existing observer or because simply it isn't a region of interest. The coverage mask can be of type boolean or based on a graded scale depending on the interest. An example is depicted in Figure 1(c)).

The algorithm proceeds as follows. In the first iteration, the masked total viewshed map (MAP_1) is computed on the considered mask. Then, the observer that provides the best coverage is found, OBS_1, by considering the absolute maximum in MAP_1. It is worthwhile to mention that the observer may be sited either outside or inside the masked area. Finally, the viewshed from OBS_1 (VIEW_1) is subtracted from MASK_0 to obtain MASK_1 (MASK_1=MASK_0-VIEW_1)

In the following iterations, the masked total viewshed map (MAP_i) is again calculated on the current DEM, thus providing new observers (OBS_i) and updated masks (MASK_i). Successively, the processes continues until the total coverage is guaranteed or until the desired number of observers is reached.

Algorithm 2: The siting observers algorithm

```
input mask MASK_0
while(condition) //either n==max_n or cover > max_cover
  {
  MAP_i= computeMaskedMap(DEM, MASK_i-1);
  point OBS_i= max(MAP_i);
  VIEW_i= viewshed(OBS_i, DEM);
  MASK_i= MASK_i-1 - VIEW_i;
  }
```

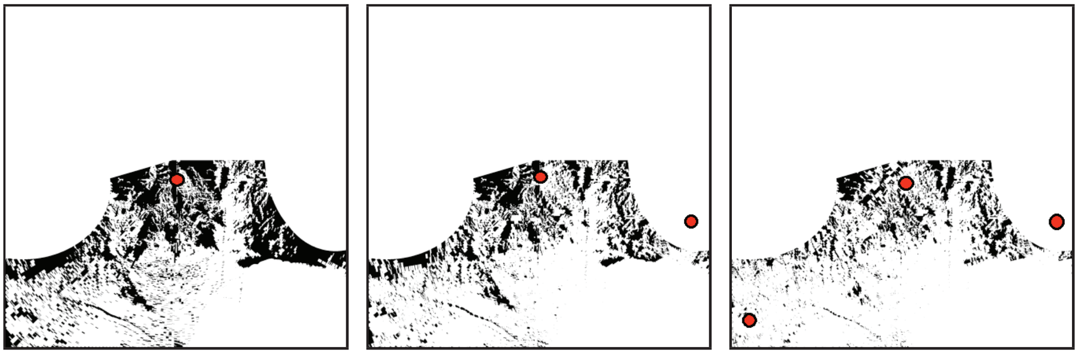
4 Evaluation and Results

For evaluation and comparison purposes we selected a DEM (already shown in the previous section as Figure 1(a))) of a terrain that guarantee obtaining valid results. The city of Malaga, in the south of Spain, is located between the Mediterranean sea and the highest mountains of the peninsula, i.e., Sierra Nevada. Three types of terrain are included in this region, sea, valley and mountains. The criteria to decide whether to provide coverage in specific zones can be either highly populated areas or the areas situated near from the costs.

The black area in Figure 1(c) depicts the regions where the population is situated and where the maximum coverage is needed, Figure 1(b) shows the total-viewshed map considering that all the points of the terrain are observers situated at 10 m above the ground.

In a first experiment, we limit to three the number of towers. The masks used in iteration 1, 2 and 3 to find tower #1, #2 and #3 are shown in Figures 1 (a), 2 (a) and (b) respectively. Figures 2 shows the resulting masks 1, 2 and 3 after siting each one of the three towers.

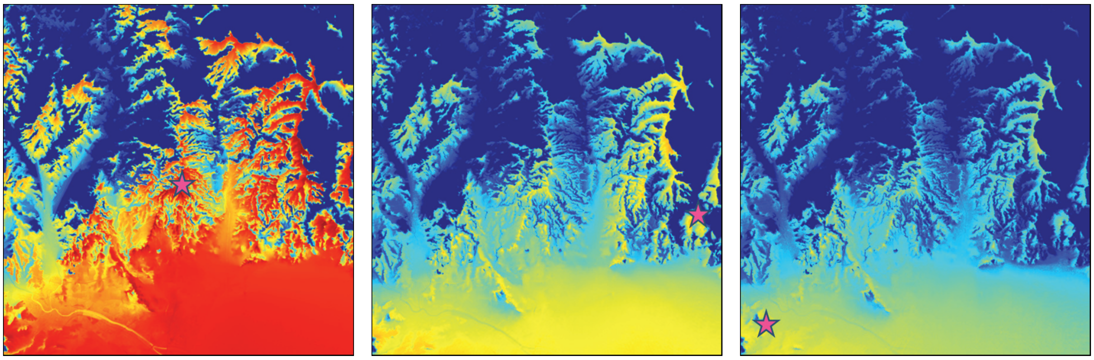
Figure 3(a), (b) and (c) show the masked total-viewshed maps that have been used in the computation. Figure 3(a) shows the total-viewshed computed in the area of interest, the point with the maximum viewshed is the location of the first observer, tower #1. Figure 3(b) shows the total-viewshed computed in the updated area of interest, where the areas covered by the tower #1 is eliminated; the point with the maximum viewshed is the location for the second



(a) MASK 1 (b) MASK 2 (c) MASK 3

Figure 2: Masks (covered areas) resulting after the sequential siting of towers #1, #2 and #3.

observer, tower #2. Finally, tower #3 is situated in the maximum of the area that was not yet covered by tower # 1 and #2.



(a) MAP 1 (b) MAP 2 (c) MAP 3

Figure 3: Masked TVS maps #1, #2 and #3, using masks #0, #1 and #2 as input, respectively. The star symbols (in red color) indicates the observer (tower) positions.

To find three towers that provide the maximum coverage on the DEM shown in Figure 3(b), we obtained the results shown in Table 1. Where, column 1 indicates the size of the used masks expressed in number of points. Column 2, indicates the percentage occupied by the mask in the terrain. Column 3 indicates the coordinates of the tower that gives the maximum coverage in the area indicated by the mask. Columns 4 and 5 indicate the number of additional points that covered by each observer (or tower) and the resulting coverage.

4.1 A discussion: Our method versus other methods

Let us use the analogy between terrain surveillance and radio-signals emission. Let P be the set of potential locations for broadcasting towers and Q the set of potential locations for receptor towers. Most methods in literature consider exhaustively searching the observers, $Q_{continuous}$, among all the points of the terrain $P_{continuous}$ as impossible. Note that *continuous* means,

Table 1: Tower location and coverage

Mask	Mask cover	Max location (row,col)	Covered points	Tower coverage
none	0%	958 / 1028		
0 (2416436)	60,41%	1016 / 1006 (tower 1)	+1085474	87,54% (#1)
1 (3501910)	87,54%	1240 /1839 (tower 2)	+166092	91,70% (#1,2)
2 (3668002)	91,70%	1696/21 (tower 3)	+102416 (3770418)	94,26% (#1-3)

all the points in the DEM at the highest resolution. Network planning methods in the field of mobile phone communications is based on the demand node concept, which consists in providing coverage to a finite number of points that represents the terrain $Q_{discrete}$. Some of these methods search for the potential observers among all the terrain ($P_{continuous} \wedge Q_{discrete}$), while others use discrete sets of locations ($P_{discrete} \wedge Q_{discrete}$). There are also methods that delimit only the set of possible observers ($P_{discrete} \wedge Q_{continuous}$). It is easy to demonstrate that the last group of methods has the same complexity as a search in $P_{continuous} \wedge Q_{discrete}$ since it is just a change of roles between observers and observed points. Nevertheless, the method that uses the entire possible points in both sets: $P_{continuous}$ and $Q_{continuous}$ is, obviously, the unique approach that guarantee that the found locations are the best solution.

4.2 Simultaneous versus sequential inclusion of towers

Finding the best n locations for siting n observers to give the maximum coverage can be performed simultaneously or sequentially. In the sequential approach, once the first tower is found and added, the remaining $n-1$ locations are selected in similar fashion. The methodology presented in the previous section is based on this approach.

The simultaneous addition of multiple towers is more complex than the sequential one. The main idea consists of selecting multiple possible locations without any consideration on the position of one point versus the other. The reason behind this approach is that the observer situated in the best place of a terrain (for an unique tower) is not necessarily the best choice (if more towers will be placed).

Some papers [1] demonstrated that the simultaneous addition does not show substantial improvement in comparison with the serial addition. We implemented an experiment to verify this conclusion using total-viewshed maps. The experiment were designed for the example of two towers, although it can bee extended to a higher number.In this case, three steps has been used to determine the location of these towers.

4.2.1 Step 1. Seed set composition

The first step of the proposed methodology consists of selecting a first set of candidates that determine an approximate location of the towers, i.e., seeds of towers. We generated three sets: the first set of candidates (A) is selected among the locations that have the highest visibility of their surrounded points, in a radius of 500 m, in the terrain of interest. Set B is selected among the highest points of the terrain with respect to the points around them. The third set, set (C) are points distributed homogeneously in all the terrain with a distance of 500 m between each two points. The size of sets A, B and C are 442, 669 and 1378 points respectively. We excluded the points that represent the sea area.

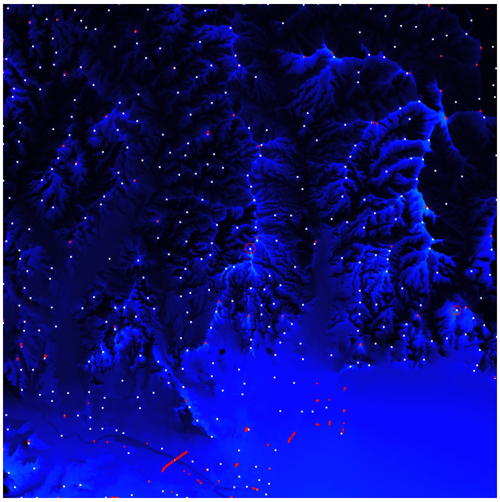


Figure 4: Candidates in sets A (white points, with higher visibility) and B (red points, on relative maximum heights)

4.2.2 Step 2. Seed selection

Given the three sets of candidates, four pairs of seeds are selected via an exhaustive search among candidates and by comparing all the possible pairs of points in sets A, B, C and D=B+C. Notice that the exhaustive search implies the simultaneous computation of the visibility of 2539320 pairs, which would be intractable if we didn't have full visibility data. The Table 2 shows the resulting position of the towers.

Table 2: Seed location in simultaneous addition

Pair	Pos #1	Pos #2	Pair coverage
A	1001/992	1837/1240	91,48%
B	1964/1025	825/1716	89,87%
C	1900/50	800/1700	88,15%
D	1964/1025	825/1716	89,87% (same as B)

Note that the set D does not improve the results of set B, and that set A clearly provides better candidates.

4.2.3 Step 3. Pair optimization

Once chosen the four pairs of candidates, we have tried to improve its location using two optimization strategies (steepest descent, implicit filtering and Nelder-Mead). None of these approaches have shown significant improvement. (lower than 0.1% in coverage, with horizontal displacement from the seeds on less than 100 meters). Using the Nelder-Mead method, the improvements are somewhat higher (about 0.5%), although they could be considered a random result, due to the extremely noise and non-smooth nature of the optimization problem.

Moreover, the Nelder-Mead method, applied to pair A, moved the towers to exactly the same position determined with the sequential addition of towers (table 1).

4.2.4 Performance

The results shown in this paper have been performed on a Dual Xeon E5-2698 v3, with 32 cores. This means that we can execute up to 6 sectors per core. Each one of these maps takes approximately 90 seconds to be calculated. The number of floating point operations per sector in our algorithm is

$$flop\ count = n \times ((4 + 4m)\frac{\sqrt{n}}{2} + 4nrs)$$

Variables m and nrs stands for the ratio of unmasked points and the total number of visible segments per direction and per sector, respectively. nrs is two orders of magnitude smaller than $\sqrt(n)$, and its mean value for the case study analyzed in this work is around 6 (in each direction). On the other hand, $m > 0.75$. This results in 28Gflop per sector, and a peak performance of 60Gflops on the Dual Xeon platform.

The most related works to our own work do not utilize specific algorithms for viewshed computation. Instead, they use the functions implemented in commercial software. For example, GRASS and ArcGis plugins, or modules, require several seconds to calculate the viewshed at each single point of the terrain, while our algorithm takes 0.00003 sec. As results, the use of these software may take hours or even days of CPU time to obtain less accurate results. The main reason behind this runtime differences is that our total-viewshed algorithm reutilizes visibility data between neighbor points and is based on an efficient data structure that practically eliminates L1 caches misses (about one line every 7048 flop).

5 Conclusions

In this work, we provide a suboptimal solution for the problem of siting observers to give the maximum coverage to a terrain represented by a DEM. The proposed algorithm is based on a new tool, the total-viewshed algorithm, which has been recently published by the authors. By modifying the algorithm with a single line, the areas of interest have been considered and the calculated map takes into account the coverage of existing observers. An intuitive technique for the location of observers is presented, and it is shown to be an optimal solution for a sequential addition. We also provided a heuristic for simultaneous location, which is based on a previous calculation of candidates among the places with higher visibility. Experiments show that both sequential and simultaneous placement gives nearly identical results.

Acknowledgements

This work was supported by Ministerio de Ciencia e Innovación, Spain, under Projects TIN2010-16144, TIN2013-42253-P and by Junta de Andalusia under Projects TIC-8260 and P11-TIC-7176.

References

[1] Mohan R Akella, Eric Delmelle, Rajan Batta, Peter Rogerson, and Alan Blatt. Adaptive cell tower location using geostatistics. *Geographical Analysis*, 42(3):227–244, 2010.

[2] TH Cormen, CE Leiserson, RL Rivest, and C Stein. Introduction to algorithms mit press. *Cambridge, MA,*, 2003.

[3] Lawrence Davis. Genetic algorithms and simulated annealing. *G*, 1987.

[4] Salles VG de Magalhaes, Marcus VA Andrade, and Chaulio Ferreira. Heuristics to site observers in a terrain represented by a digital elevation matrix. In *XI GEOINFO*, 2010.

[5] Wm Randolph Franklin and Clark Ray. Higher isnt necessarily better: Visibility algorithms and experiments. In *Advances in GIS research: sixth international symposium on spatial data handling*, volume 2, pages 751–770. Taylor & Francis, Edinburgh, 1994.

[6] R Intel. Intel 64 and ia-32 architectures optimization reference manual. *Intel Corporation,Sept*, 2014.

[7] Melanie Mitchell. *An introduction to genetic algorithms*. MIT press, 1998.

[8] George Nagy. Terrain visibility. *Computers and graphics*, 18(6):763–773, 1994.

[9] Mauricio GC Resende. Greedy randomized adaptive search procedures (grasp). *Encyclopedia of optimization*, 2:373–382, 2001.

[10] Luis F Romero, Siham Tabik, Jesús M Vías, and Emilio L Zapata. Fast clear-sky solar irradiation computation for very large digital elevation models. *Computer Physics Communications*, 178(11):800–808, 2008.

[11] S. Tabik, A. Cervilla, E. Zapata, and L. Romero. Efficient data structure and highly scalable algorithm for total-viewshed computation. *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of*, PP(99):1–7, 2014.

[12] Siham Tabik, A Villegas, Emilio L Zapata, and Luis F Romero. A fast gis-tool to compute the maximum solar energy on very large terrains. *Procedia Computer Science*, 9:364–372, 2012.

[13] Siham Tabik, Emilio L Zapata, and Luis F Romero. Simultaneous computation of total viewshed on large high resolution grids. *International Journal of Geographical Information Science*, 27(4):804–814, 2013.