

# ناحيه قابل دید نقاط روی TIN ۵,۵ بعدی و ۲,۵ بعدی

---

سپیده آقاملائی

۹۲۲۹۱۹۴۱

# قابلیت دید

- قابلیت دید (visibility):
- دو نقطه را نسبت به هم قابل دید می‌گویند اگر پاره‌خط واصل آنها توسط شیء دیگری قطع نشود.
- ناحیه قابل دید سطح (viewshed):
- بزرگترین ناحیه‌ای از یک سطح که نقاط آن از یک نقطه داده شده (ناظر) قابل دید باشند.

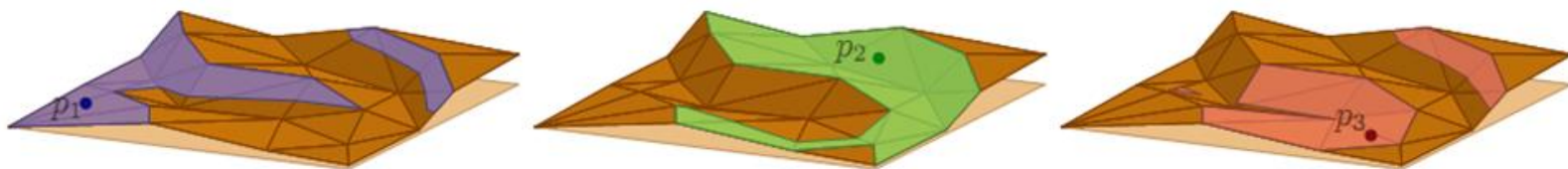
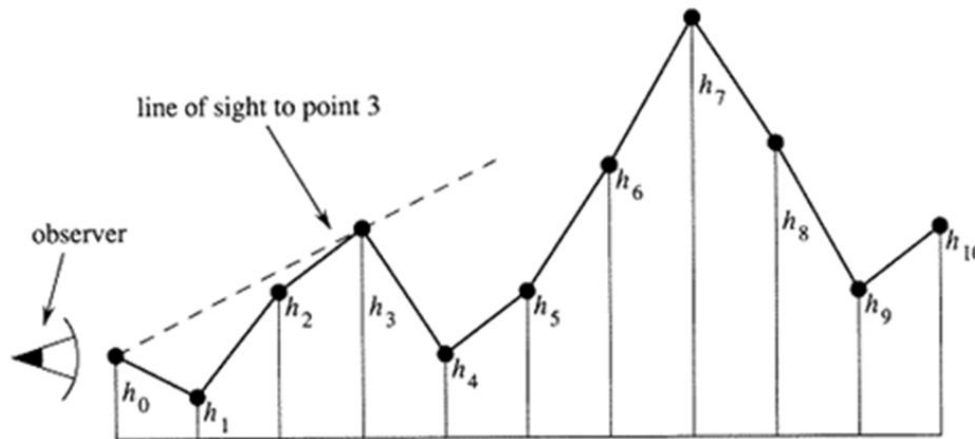


Figure 1: The viewsheds of three viewpoints on a 2.5D terrain.

# ۱,۵ بعدی و ۲,۵ بعدی

- ناحیه ۱,۵ بعدی:  
چندضلعی  $X$ -یکنوا در صفحه اعداد حقیقی است.
- ناحیه ۲,۵ بعدی:  
چندضلعی  $(X,Y)$ -یکنوا در فضای اعداد حقیقی است.



**Figure 1-121** The one-dimensional terrain for the observer in Problem 1.32. The observer is at height  $h_0$  and, in this example, can see only points 1, 2, 3, and 7.

# شبکه مثلث‌بندی شده‌ی نامنظم

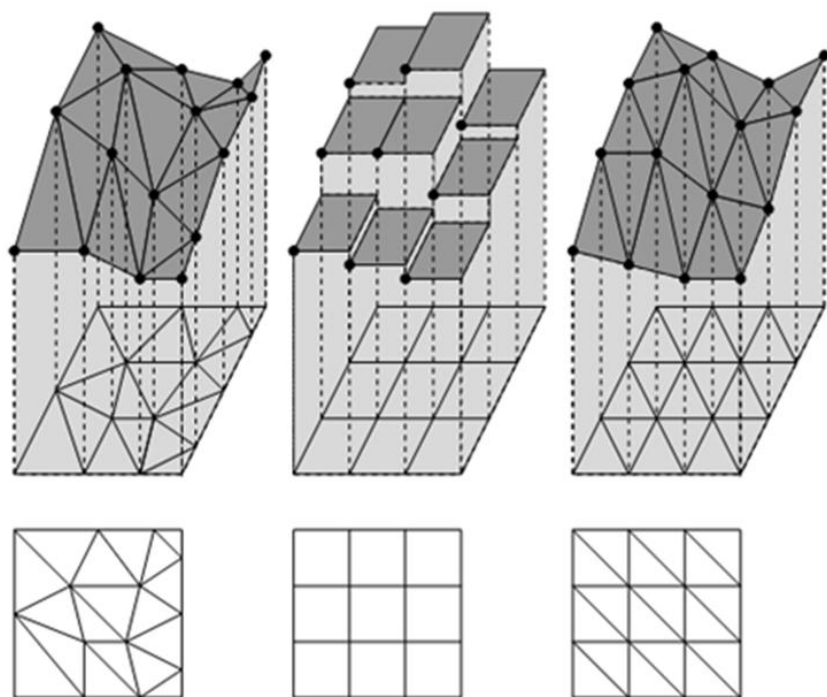


Figure 1: A TIN, a stepped RSG, and a triangulated RSG.

- شبکه مثلث‌بندی شده‌ی نامنظم
- Triangulated Irregular Network (TIN)
- دو نوع نمایش (Digital Elevation Model) در سیستم‌های اطلاعات جغرافیایی (GIS) داریم:
- نمونه‌گیری تصادفی از سطح (TIN)
- نمونه‌گیری منظم از سطح (RSG)

## نمایش سلسله مراتبی

- می‌توان بر اساس مدل نمایش، از فشردگی با انواع درخت چهارتایی (quadtree) استفاده کرد.

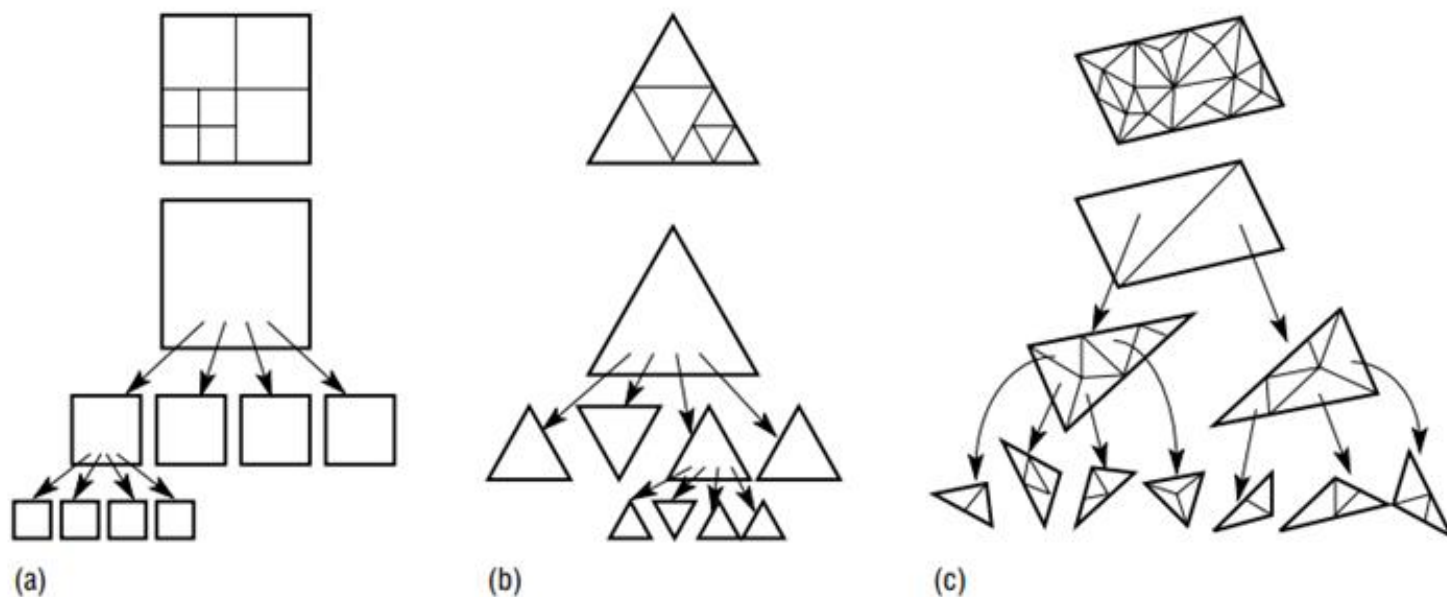
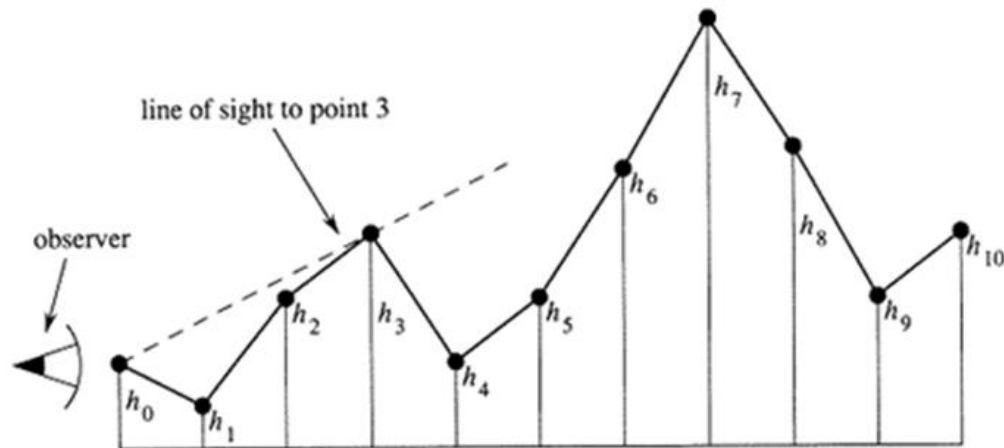


Fig 5. Hierarchical terrain models: (a) quadtree; (b) quaternary triangulation; (c) hierarchical TIN.

## محاسبه ناحیه قابل دید در حالت ۱,۵ بعدی

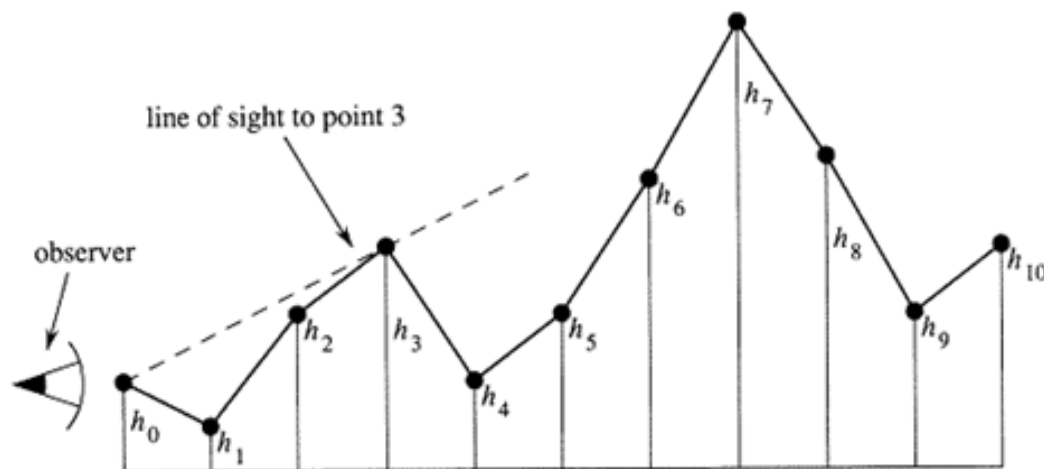
- خط دید (line of sight): خطی که از ناظر به یک رأس ناحیه رسم می‌شود.
- زاویه دید یعنی زاویه بین خط دید و محور X را حساب می‌کنیم. الگوریتم پیشوندی موازی با تابع ماکسیمم را روی آن اجرا می‌کنیم. رأسهای ماکسیمم قابل دیدی هستند. (ناحیه قابل دید گسسته)



**Figure 1-121** The one-dimensional terrain for the observer in Problem 1.32. The observer is at height  $h_0$  and, in this example, can see only points 1, 2, 3, and 7.

# الگوریتم برای حالت ۱,۵ بعدی

- تقاطع یال بین یک رأس قابل دید و غیر قابل دید را با خط دید حساب می‌کنیم و رأس ناحیه قابل دید را به دست می‌آوریم. سایر رأسهای ناحیه قابل دید رأسهای قابل دید اصلی هستند. (ناحیه قابل دید پیوسته چندضلعی با این رئوس است.)



**Figure 1-121** The one-dimensional terrain for the observer in Problem 1.32. The observer is at height  $h_0$  and, in this example, can see only points 1, 2, 3, and 7.

# پیادهسازی الگوریتم حالت ۱,۵ بعدی

- الگوریتم پیشوندی موازی PRAM را به کار می‌بریم.
- رأسهای قابل دید را در یک آرایه علامت می‌زنیم و زاویه‌های آنها را نگه می‌داریم.

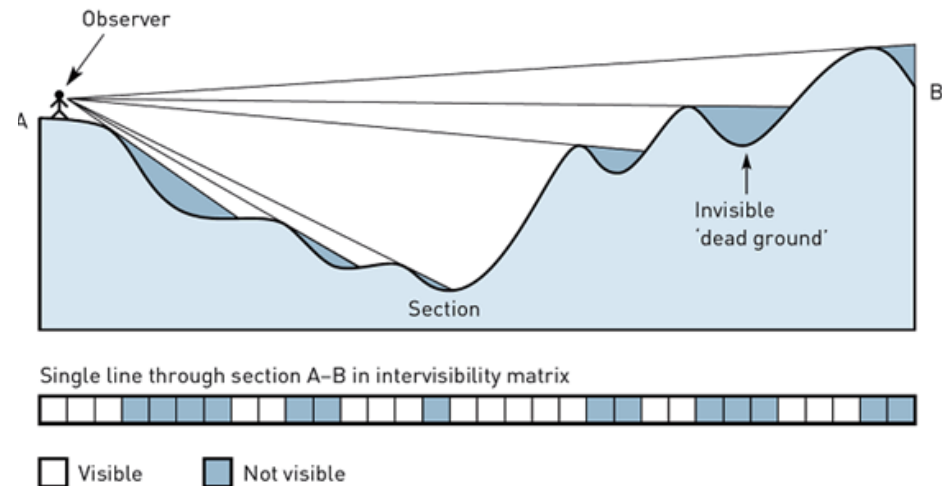
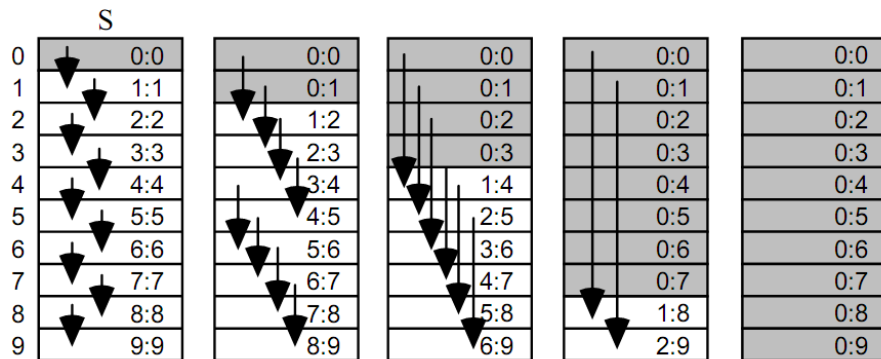


Fig. 3 Ray tracing for visibility analysis [11]



# پیشوند موازی در مدل PRAM

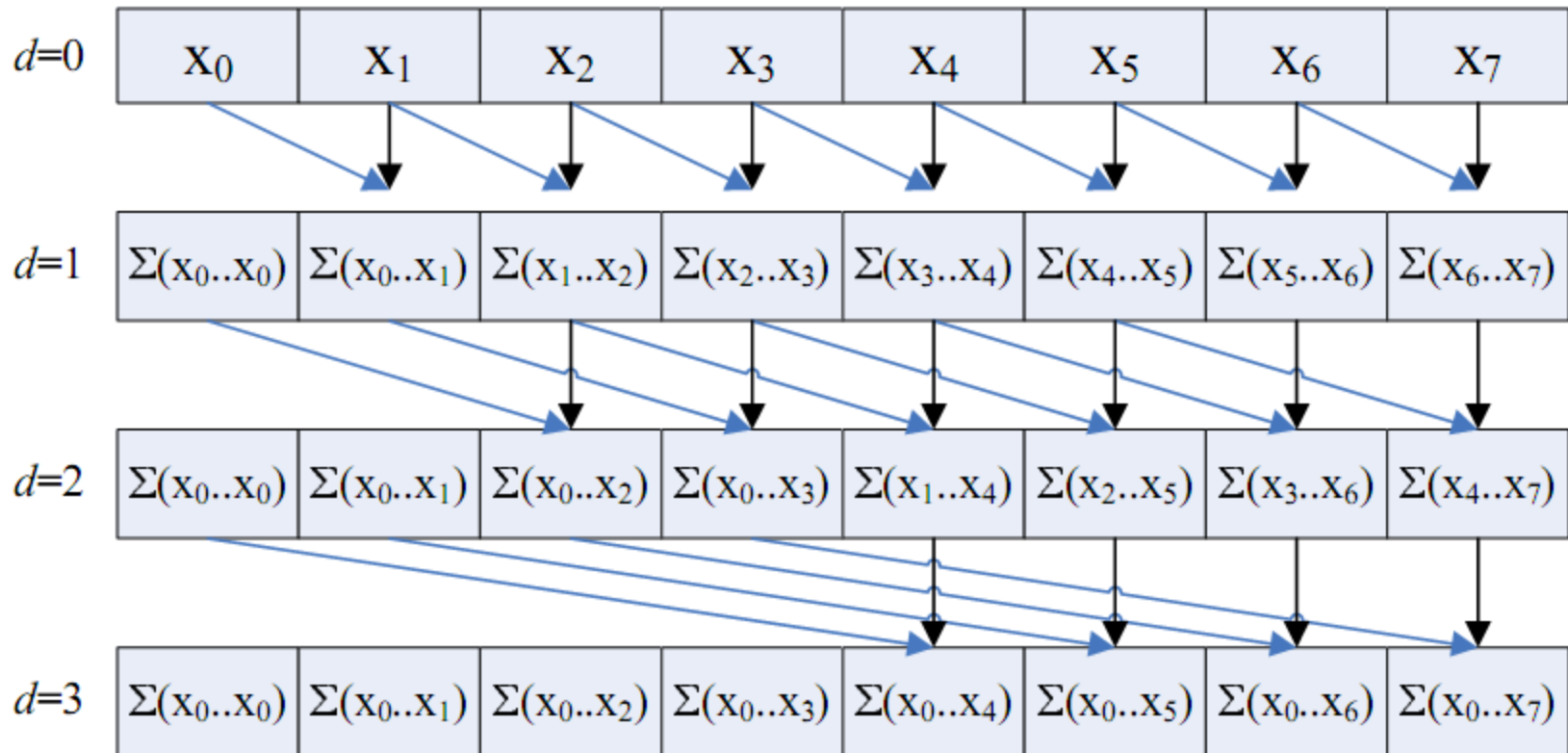


Figure 1: Computing a scan of an array of 8 elements using the naïve scan algorithm.

# مرحله‌ی اول الگوریتم بهینه پیشوند موازی

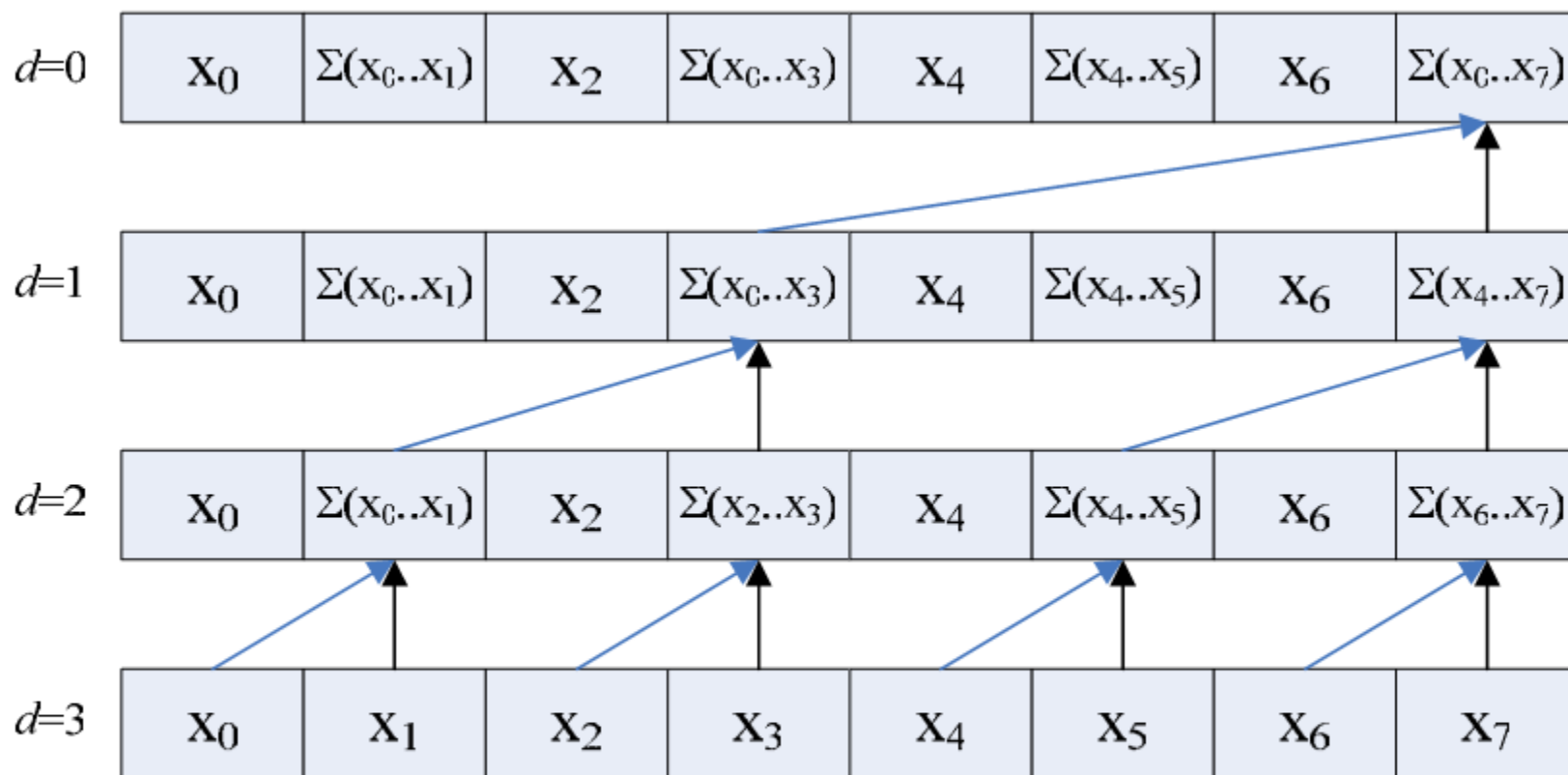


Figure 2: An illustration of the up-sweep, or reduce, phase of a work-efficient sum scan algorithm.

# مرحله‌ی دوم الگوریتم بهینه پیشوند موازی

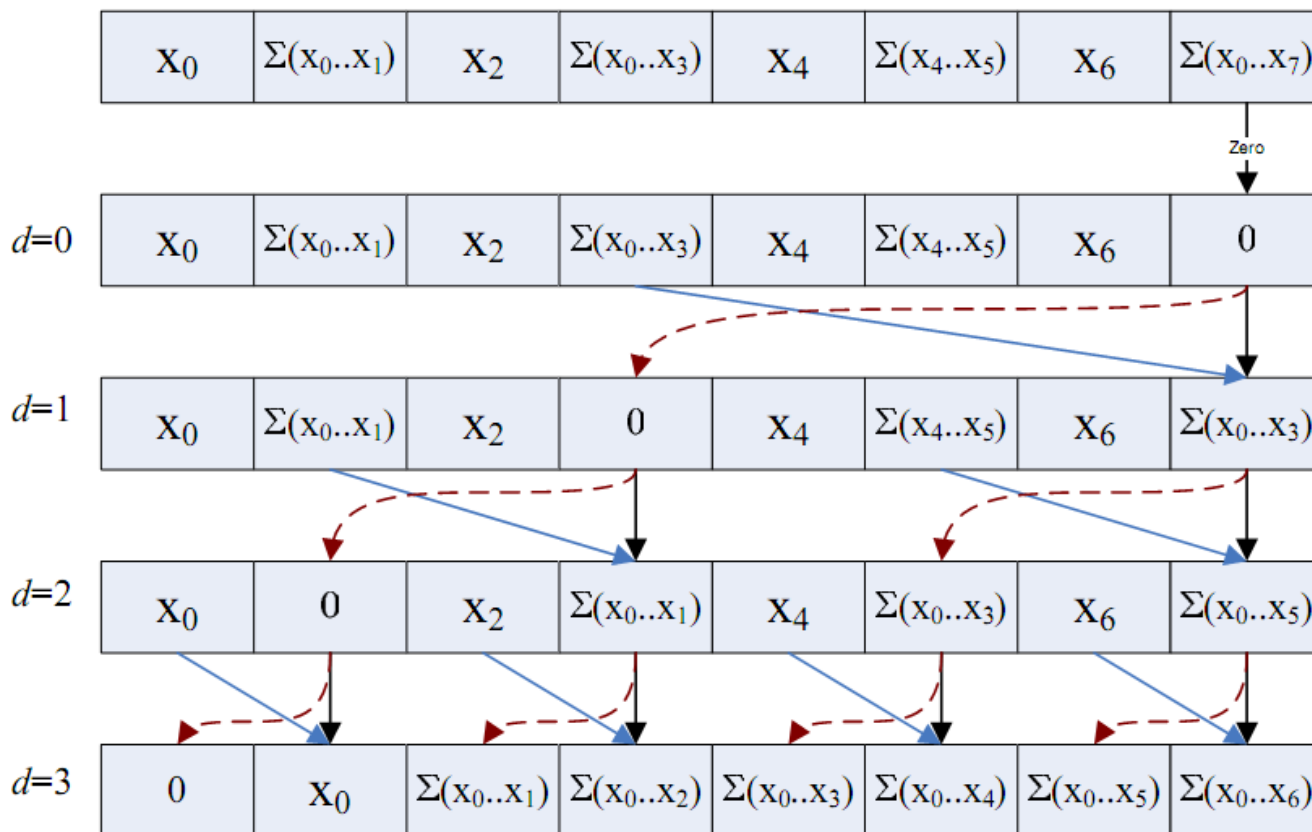


Figure 3: An illustration of the down-sweep phase of the work-efficient parallel sum scan algorithm. Notice that the first step zeros the last element of the array.

# روش‌های موجود

Approach	Authors	Input DEM	Output information
Front-to-back	De Floriani et al., 1989 (parallel version by De Floriani, Montani and Scopigno, 1994)	TIN	Continuous visibility map
Front-to-back	Lee, 1991	TIN	Discrete visibility map
Line-of-sight	Shapira, 1990; Brelloch, 1990	RSG	Discrete visibility map
Line-of-sight (parallel algorithm)	Mills, Fox and Heimach, 1992; Teng, De Menthon and Davis, 1993	RSG	Inter-visibility map
Line-of-sight	Sorensen and Lanter, 1993	RSG	Continuous visibility map
Sector-based	Stewart, 1998	RSG or TIN (just vertices)	Approximated horizon
Concentric rings (approximated algorithm)	Franklin and Ray, 1994	RSG	Discrete visibility map
Line-of-sight	Fisher, 1996	RSG	Extended viewsheds
Radial sweep	Van Kreveld, 1996	RSG	Extended viewsheds
Line-of-sight (parallel algorithm)	Rallings et al., 1998	RSG	Visibility counts

Table 1: Summary of reviewed visibility algorithms.

## محاسبه ناحیه قابل دید در حالت ۲,۵ بعدی

- اگر از نقطه دید به یک ضلع TIN وصل کنیم، ناحیه‌ی گلدانی شکلی ساخته می‌شود که زیر آن دیده نمی‌شود.
- Upper envelope این ناحیه‌ها و مثلث‌های terrain پیوسته‌ی قابل دید را می‌دهد.

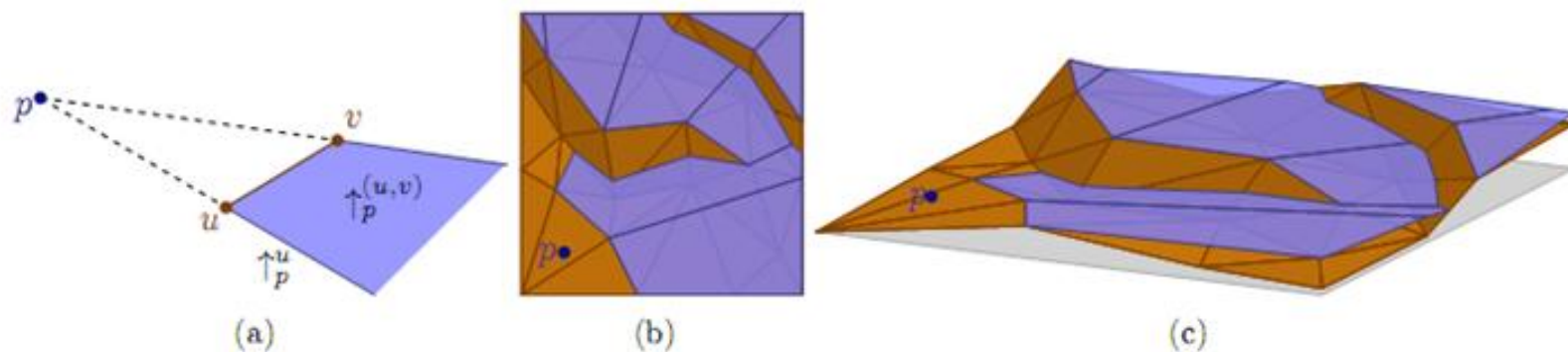


Figure 11: (a) A ray and a vase. (b) The top-down view of a terrain  $\mathcal{T}$  with a single viewpoint  $p$ . The domain is decomposed in the viewshed  $\text{Vis}(\mathcal{T}, p)$  and a collection of vases. (c) a 3D view of  $\mathcal{T}$  and the vases of  $p$ .

# الگوریتم بهینه حالت ۲,۵ بعدی

• چندضلعی ساده و lower envelope

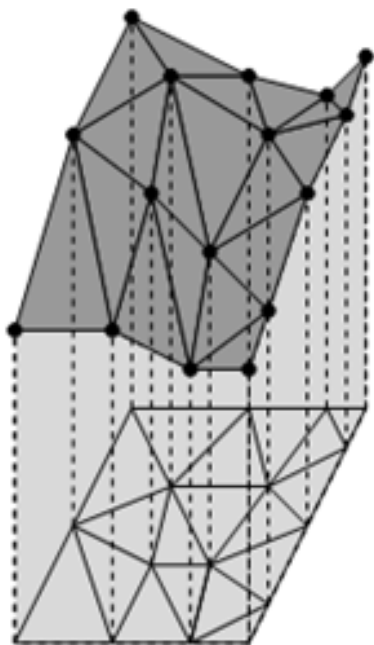
TABLE 42.6.1 Parallel visibility algorithms for a simple polygon.

PROBLEM	MODEL	TIME	WORK	REF
Kernel	EREW	$O(\log n)$	$O(n)$	[Che95]
Vis from a point	EREW	$O(\log n)$	$O(n)$	[ACW91]
Vis from an edge	CRCW	$O(\log n)$	$O(n)$	[Her92]
Vis from an edge	CREW	$O(\log n)$	$O(n \log n)$	[GSG92, GSG93]
Vis graph	CREW	$O(\log n)$	$O(n \log^2 n + m)$	[GSG92, GSG93]

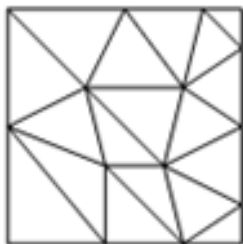
TABLE 42.6.2 General parallel visibility and enveloping algorithms.

PROBLEM	MODEL	TIME	WORK	REF
Lower env for segments	EREW	$O(\log^2 n)$	$O(n \log n)$	[Her89]
Lower env for $k$ -int curves	EREW	$O(\log^2 n)$	$O(\lambda_{k+2}(n) \log n)$	[BM87]

## الگوریتم بهینه برای حالت ۲,۵ بعدی



- الگوریتم بهینه برای TIN بدون دور است که منظور از بدون دور در آن این است که اگر شعاعی رسم شود همیشه یک وجه را قبل از وجه دیگر قطع کند.
- این الگوریتم ابتدا تصویر مثلث‌ها روی صفحه‌ی  $XY$  را بر اساس فاصله‌ی آنها نسبت به تصویر نقطه دید روی این صفحه حساب می‌کند (مرتب‌سازی زاویه‌ای) سپس ناحیه قابل دید هر مثلث را بر حسب نقطه دید به دست می‌آورد (محاسبه‌ی ناحیه دید).
- فرض می‌کنیم نقطه دید در  $(0, \infty)$  است.



# بهبودهای الگوریتم بهینه

- تقسیم داده‌ها حول نقطه دید

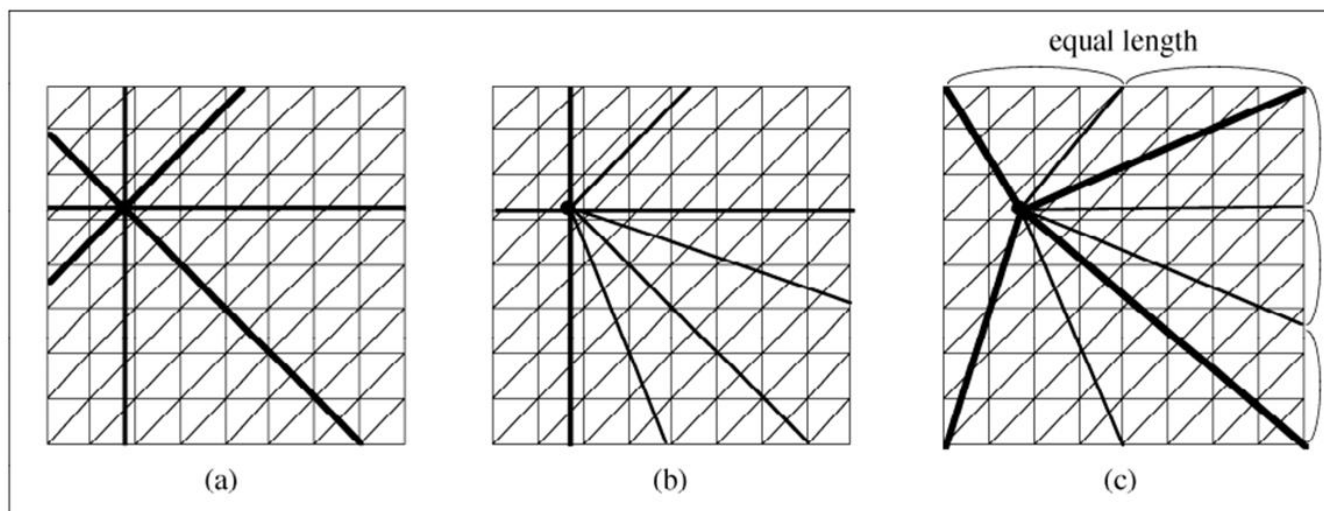


Figure 9: On the left, partition in *equal angle* sectors; in the center, *quadrant* based partition; on the right, *equal area* partition.



# بهبودهای الگوریتم بهینه

- مرتبسازی زاویه‌ای صفحات

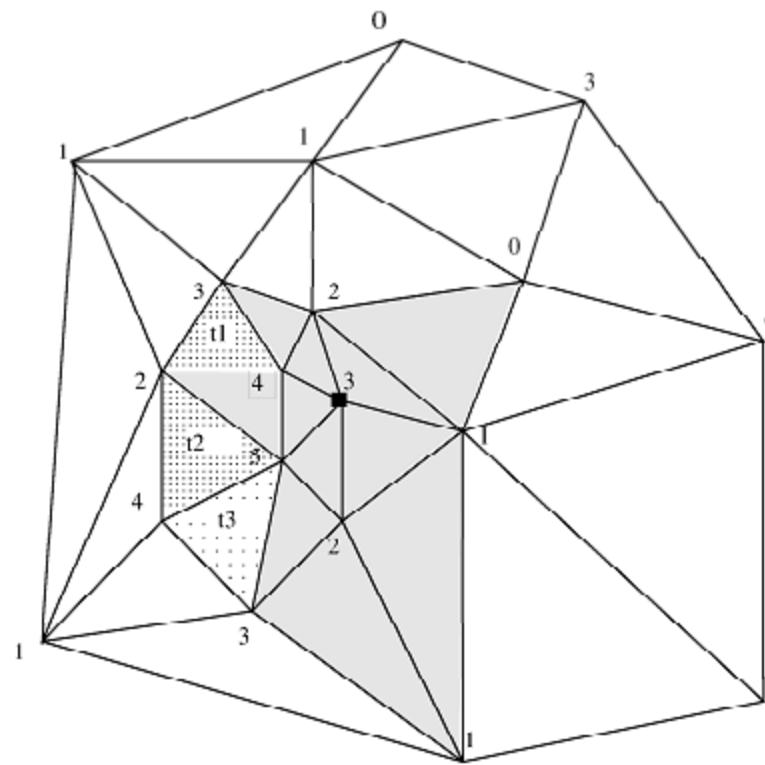
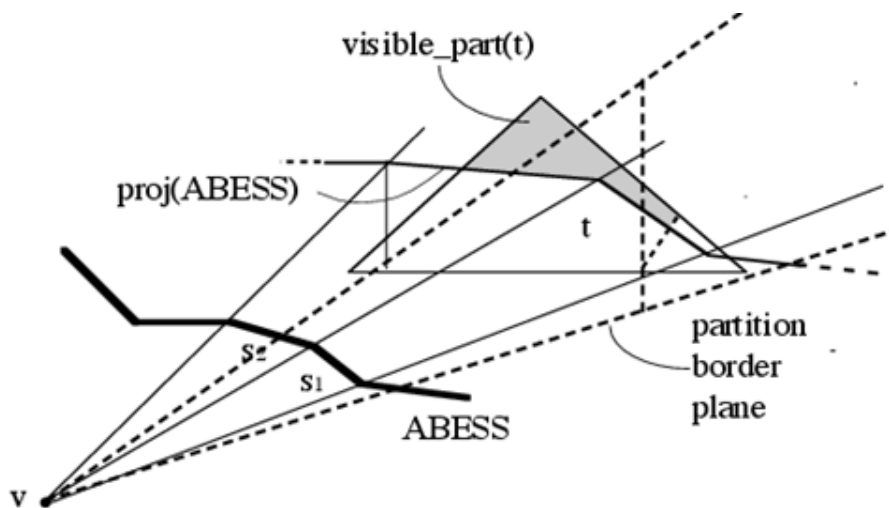


Figure 23: Test for the inclusion of a new triangle in  $\Pi$ :  $\Pi$  is the shaded polygon;  $t_1$  e  $t_2$  can be added to  $\Pi$ ,  $t_3$  cannot.

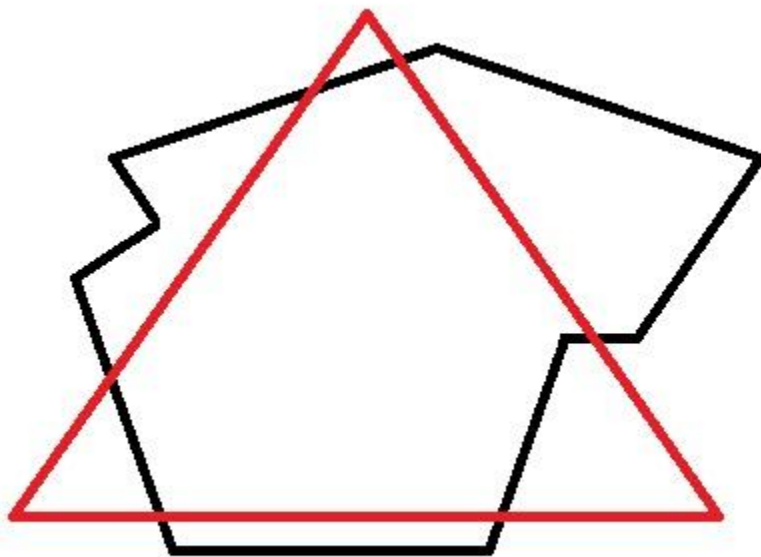
## الگوریتم حالت ۲,۵ بعدی

- ناحیه‌های مثلثی ورودی الگوریتم هستند.
- سایه‌ی هر مثلث روی ناحیه قابل دید فعلی هر مثلث دیگر حساب می‌شود.
- کار لازم برای این الگوریتم  $O(n^2\alpha(n))$  است که پیچیدگی ناحیه‌ی به دست آمده هم هست. زمان الگوریتم  $O(n\alpha(n))$  است، چون ما ناحیه‌ی هر مثلث را با یک پردازنده حساب می‌کنیم. در غیر این صورت باید تقاطع دو چند ضلعی غیرمحدب را حساب کنیم که زمان را بهتر نمی‌کند.



## محاسبه‌ی تقاطع سایه‌ی مثلث‌ها

- صفحات گذرنده از اضلاع مثلث را به دست می‌آوریم و تقاطع آنها را با صفحه‌ی مورد نظرمان پیدا می‌کنیم. با این کار یک مثلث ساخته می‌شود که با چندضلعی فعلی تقاطع آن را حساب می‌کنیم.



## سؤال؟

- [1] L. D. Floriani and P. Magillo, "Algorithms for Visibility Computation on Terrains: a Survey".
- [2] L. D. Floriani and P. Magillo, "Intervisibility on terrains," in Intervisibility on terrains.
- [3] F. T. Leighton, Introduction to Parallel Algorithms. San Mateo: Morgan Kaufman, 1992.
- [4] F. Hurtado et al., "Terrain Visibility with Multiple Viewpoints," International Symposium on Algorithms and Computation, 2014.
- [5] N. Stojanović and D. Stojanović, "Performance Improvement of Viewshed Analysis Using," TELSIKS, 2013.
- [6] Y. Xia, Y. Li, and X. Shi, "Parallel Viewshed Analysis on GPU using CUDA," Third International Joint Conference on Computational Science and Optimization, 2010.
- [7] J. Fishman, H. Haverkort, and L. Toma, "Improved Visibility Computation on Massive Grid Terrains".
- [8] C. R. Ferreira, M.V. A. Andrade, S. V. G. Magalhães, W. R. Franklin, and G. C. Pena, "A parallel algorithm for viewshed computation on grid," Journal of Information and Data Management, 2014.
- [9] M.T. Goodrich, "Parallel Algorithms in Geometry," in Handbook of Discrete and Computational Geometry.: Chapman and Hall/CRC Press, 2004, ch. 42, pp. 953-967.
- [10] M. J. Atallah and D. Z. Chen, "Optimal parallel algorithm for visibility of a simple polygon from a point," SCG '89 Proceedings of the fifth annual symposium on Computational geometry, pp. 114 - 123, 1989.
- [11] P. Bertolazzi, S. Salza, and C. Guerra, "A parallel algorithm for the visibility problem from a point," J. Parallel Distrib. Comput., vol. 9, pp. 11-14, 1990.
- [12] Mikhail J. Atallah, "Efficient Plane Sweeping in Parallel," Report Number: 85-563, 1985.
- [13] L. Boxer and R. Miller, "Parallel dynamic computational geometry," Dept. Comput. Sci., SUNY-Buffalo, Report 87-11, 1987.
- [14] L. D. Floriani and P. Magillo, "Visibility algorithms on triangulated digital terrain models," International Journal of Geographical Information Systems, vol. 8, no. 1, pp. 13-41, 1994.
- [15] L. De Floriani, C. Montani, and R. Scopigno, "Parallelizing visibility computation on triangulated terrains," International Journal of Geographic Information Systems, vol. 8, no. 6, pp. 515-531, 1994.
- [16] N. Coll, N. Madern, and J. A. Sella, "Good-visibility maps visualization".