

# 1 ?”” (1) Time Parallel Algorithm for Finding 2D Convex Hull on 2 a Reconfigurable Mesh Computer Architecture

3 Omar Bouattane

4 *Received: 13 December 2020 Accepted: 4 January 2021 Published: 15 January 2021*

---

## 6 Abstract

7 In this paper we propose a parallel algorithm in image processing in ?””(1) time, intended for  
8 a parallel machine ” Reconfigurable Mesh Computer (RMC), of size  $n \times n$  Elementary  
9 Processors (PE). The algorithm consists in determining the convex envelope of a two-level 2D  
10 image with a complexity in ?””(1) time. The approach used is purely geometric. It is based  
11 solely on the projection of the coordinates of PEs retained in specific quadrants and on the  
12 application of the algorithm that determines the Min / Max in ?””(1) time. This has reduced  
13 the complexity of the algorithm for determining the convex hull at ?””(1) time.

---

15 *Index terms*— processional image, parallel processing, reconfigurable mesh computer, convex hull.

## 16 1 Introduction

17 he problem of the convex envelope (convex hull) of a plane shape has been the subject of several studies in  
18 recent years. This very important problem concerns several areas such as image processing, pattern recognition  
19 or robotics [FU97]. The convex envelope can be used, for example, to normalize a shape, to triangulate a set of  
20 points, to extract topological characteristics, to decompose a complex shape in order to facilitate its recognition.

21 Compared to the very important number of works that have been realized on the use of sequential algorithms  
22 to extract the convex hull, the number of parallel algorithms to solve this problem is significantly lower. We have  
23 to wait until the beginning of the 80’s to see the first parallel algorithms, which calculate the convex hull of a  
24 set of points [CHO81], [NAT81], [ALD83]. Since then, several authors have presented parallel algorithms to solve  
25 this problem with different levels of complexity { {

26 [KIM87], [LIN93], [PRA89] and on different input data as for example on a set of points in a plane or in a  
27 space, on bi-level images, on multi-level images with a single component or several components. On the other  
28 hand, the algorithms proposed in these works, were designed for different models of machines such as the model  
29 Mesh Connected Computer [MIL85], the model Polymorphic Torus [LI89] or the model Reconfigurable Mesh  
30 Computer [HAY98].

31 The synthesis of these different works leads us to note that the diversity of the parameters put into play  
32 complicates any attempt at global comparison between the performances of the different approaches. Table 1  
33 presents a synthesis of some bibliographical references on this problem by specifying each time the machine model  
34 used, the nature of the input data as well as the degree of complexity obtained.

35 Among the parallel algorithms for the determination of the convex hull of the connected components of a  
36 multi-level image, mention may be made of [ERR05], which describes an approach based on the structural  
37 characterization algorithm. As for the algorithm [BOU02] applies a purely geometric method. Our algorithm  
38 calculates with a complexity of ?””(1) the convex hull of an image or set of points with two levels of gray.  
39 This algorithm uses a purely geometric approach. It is based solely on the projection of the coordinates of PEs  
40 retained in specific quadrants and on the application of the algorithm that determines the Min / Max in ?””(1)  
41 time [ELMES91]. This allowed us to reduce the complexity of our solution compared to those proposed in the  
42 literature. The contribution of this document is summarized as follows: In this article, section 2 is devoted to  
43 the calculation model used for the implementation of the proposed algorithm. Section 3 presents the algorithm  
44 for determining the convex hull. Section 4 presents the result of determination of the convex hull of a two-level  
45 image matrix. This paper concludes with some remarks on future work.

## 2 II. Computational Model Architecture

The computational model of machine used in this paper is a parallel Reconfigurable Mesh Computer (RMC) [ELMES86] of size  $n \times n$  Elementary Processors (PEs) arranged in a square matrix. It is a machine which respects the SIMD (Single instruction multiple data) model, in which each PE is located in the matrix by its row and column coordinates and can be characterized by its identifier  $ID = n.i + j$ , where  $i$  and  $j$  represent the row number and the column number respectively. In this architecture, Figure ?? 1 (a) refers to a set of PEs, each one is connected to its four neighbors; if they exist; through communication channels. It has a finite number of registers; of bus length  $\log_2(n)$  bits, in which it stores data to perform arithmetic and logic operations. All PEs can perform reconfiguration operations to exchange information with other PEs in the mesh. Figure 1. (b) describes the set of possible configurations of each PE. The communication of each PE with its neighbors is implemented through four different operations:

Single Bridge (SB): A PE of the RMC is in the SB state, if it establishes links between two of its communication channels, either in transmitting mode or the receiving one. In addition, it may disconnect some of its communication channels. The algorithm for determining the convex hull consists of 7 steps: (Figure 3).

### 3 4.

Step 4: The delimitation of the area P 1 (quadrant: NW), then the determination of the PEs belonging to the convex hull forming part of this zone. The procedure will be applied to zones P 2, P 3 and P 4 .

(Figure 3). 5.

Step 5: Labeling of all PEs belonging to the zones P 1, P 2, P 3 and P 4 (Figure 8). 6.

Step 6: Elimination of all PEs that do not belong to the zones P 1, P 2, P 3 and P 4 . (Figure 8). Step 2:

Step 3: a) All the PEs of the image matrix put themselves in (SB) on their column (Figure 6 -a). The PE H sends a code to all the PEs that are on his line. So, all PEs on this line will receive this code (Figure 7: quadrant P1). This operation runs for all others PEs: PE B ,PE C, PE D ,PE E, PE F and PE G. Step 4: Delimitation of the zones P 1 P 2 P 3 and P 4 : in the area P 1 (quadrant NW), all PEs that are between A and I 1 are in configuration (SB) on their column.

The PE A gives them a code indicating that they have to block the bus that connects them with the PEs that are on their right (Figure 8.a).

In the same way ???? ?? 1 communicates to the PEs that are in its line on the left, the code for disconnect PEs from the top line (Figure 8. a). This operation will be executed in the zones P 2 , P 3 and P 4 . Step 5: Labeling of the zones P 1, P 2 , P 3 and P 4 : in zone P 1 , since the PEs I' 1 , A and I 1 are known, we proceed to the labeling all PEs belonging to P 1 . a) All PEs are configured in (SB) with respect to the line. (Figure 5). b) The ???? (?? ?? ,?? ?? ) , ???? (?? ?? ,?? ?? ) , transmit on the bus on their right a code indicating their presence. All unmarked PEs will receive this code, those to the right of the two PEs having  $(i, m)$  and  $(i, M, j, m)$ . c) The ???? (?? ?? ,?? ?? ) , ???? (?? ?? ,?? ?? ) transmit on the bus to their left a code indicating their presence. All unmarked PEs will receive this code, those to the left of the two PEs having  $(i, m, j, M)$  and  $(i, M, j, M)$  . d) The operations indicated in (a), (b) and (c) will be done on the columns. e) Result: 4 PEs will receive 2 codes one on their line and another one on the columns and will memorize them in I' 1 , I' 2 , I' 3 and I' 4 . (Figure 5). This operation will be executed in the zones P 2 , P 3 and P 4 (Figure 8).

Consequence: All PEs in these areas are identified as part of the zones P 1 , P 2 , P 3 and P 4 . Step 6: Elimination of all PEs that do not belong to the zones P 1, P 2, P 3 and P 4 . (Figure 8). ? transmits to all PEs of the image matrix the cancellation order of marking if it is a marked PE. This implies that all PEs that do not belong to zones P 1, P 2 , P 3 and P 4 will no longer be marked. Those who belong to these areas remain marked if they are.

..??

## 4 Example:

1. In the line ?? 1 , ?? 1 3 will be selected (it has the value  $(?? 3 ? ?? 1)$  which is greater than  $(?? 3 ? ?? 2)$  )

2. In the line, ?? 3 7 will be selected (it has the value  $(?? 7 ? ?? 3)$  )

which is greater than  $(?? 7 ? ?? 4) > (?? 7 ? ?? 5) > (?? 7 ? ?? 6)$ . the order to eliminate the ???? ??  $(?? 2)$  on its column. 4. L 3 , ?? 3 7 transmits through the open bus to all PEs ?? 3 4 ?? 3 5 and ?? 3 6 the order to eliminate all ???? ?? on their column. In the other lines, no???? ?? will be selected. IV.

## 5 CONCLUSION

This paper has dealt with a parallel algorithm of determining the convex hull of a two-level 2D image with a complexity  $O(n)$  time. This is executed on a parallel machine (RMC), of size  $n \times n$  Elementary Processors. The algorithm is essentially based on projections and the calculation of Min / Max in  $O(n)$  time. In a future work, we will extend this algorithm to 3D space and always in  $O(n)$  time.

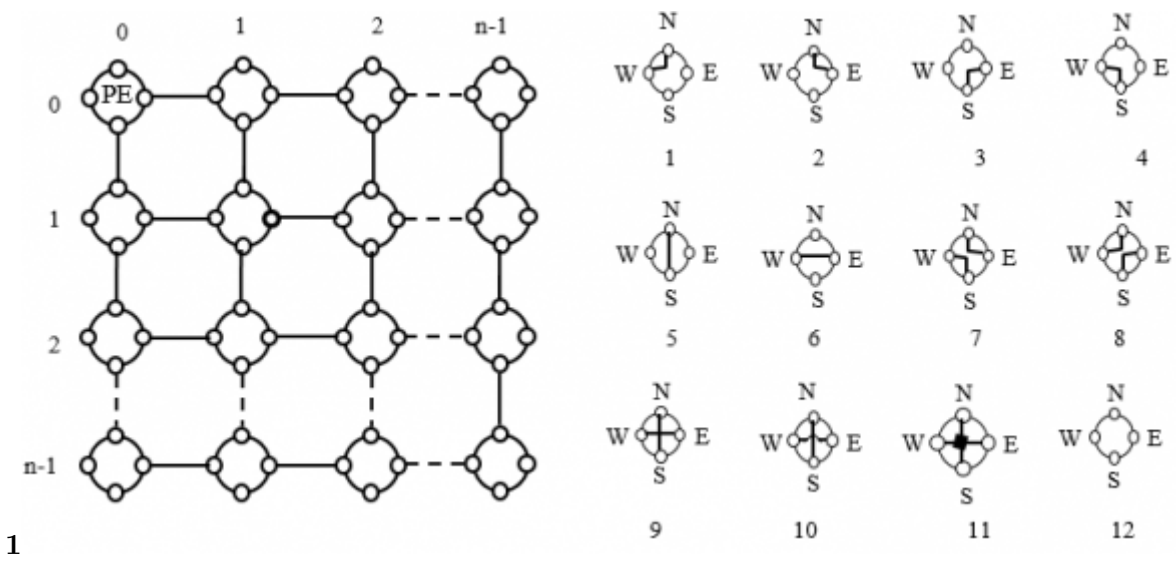


Figure 1: Figure 1 :

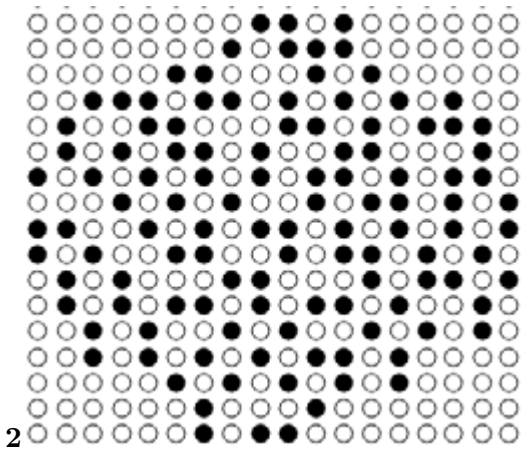


Figure 2: Figure 2 :



Figure 3: Figure 3 :



Figure 4: 1.

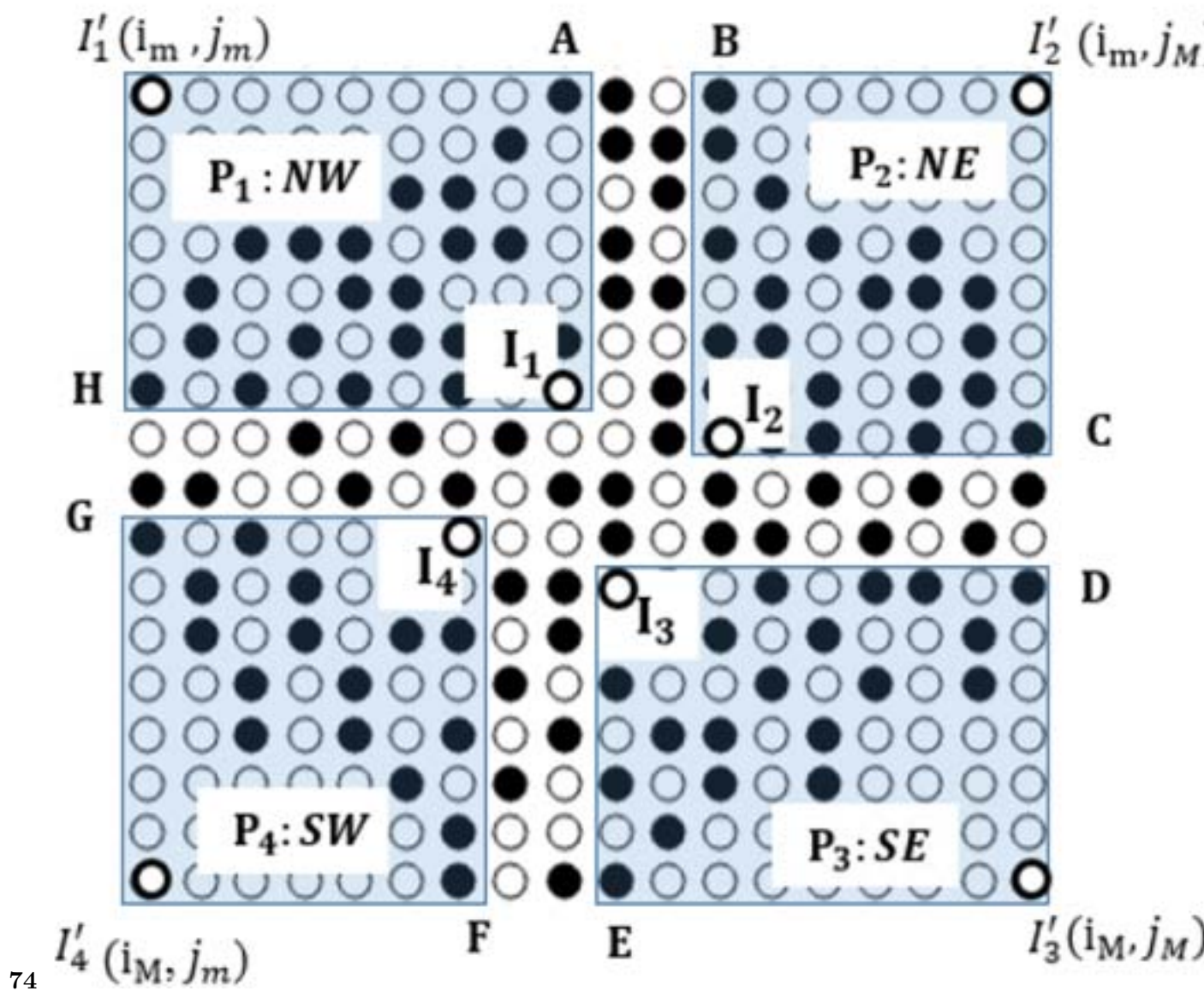
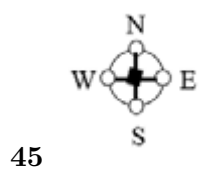
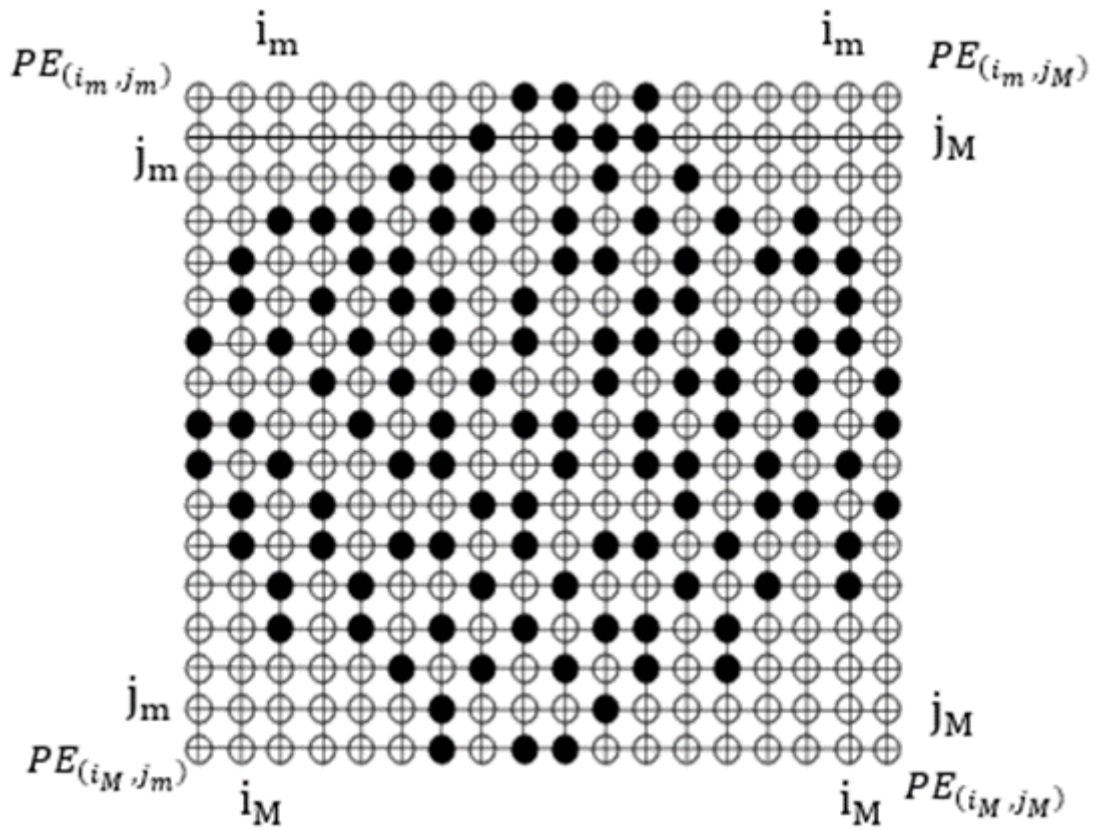


Figure 5: 7 .Figure 4



45

Figure 6: Figure 4 :Figure 5 b:



5

Figure 7: Figure 5 :

101 **6 [FU97]:**

102 FU A. M. N., and YAN H. Effective classification of planar shapes based on curve

1 2 3

<sup>1</sup>© 2021 Global Journals

<sup>2</sup>( )Year 2021

<sup>3</sup>( )Year 2021

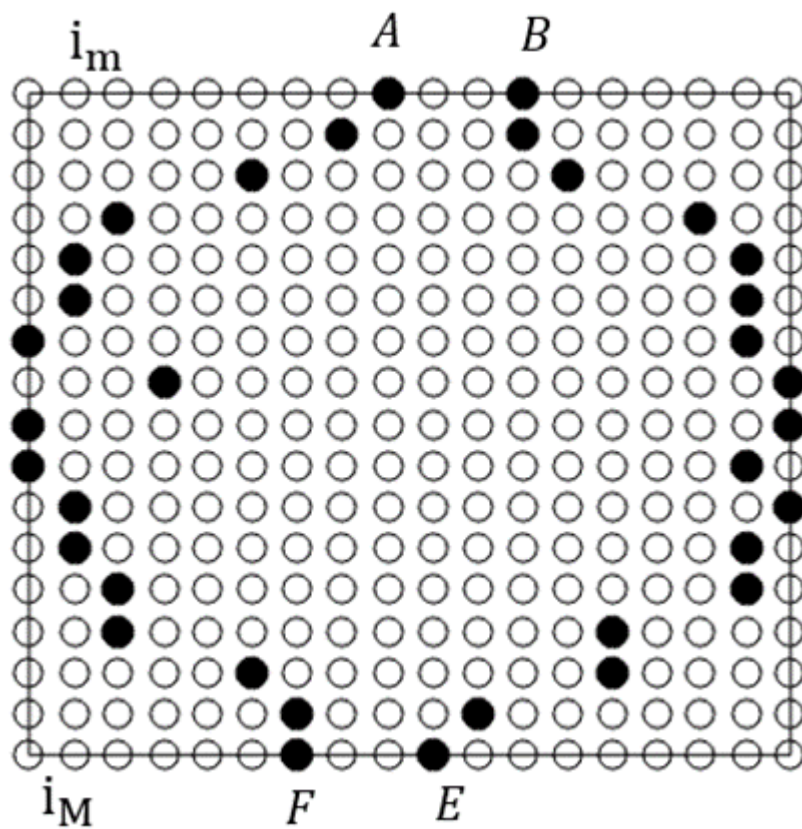
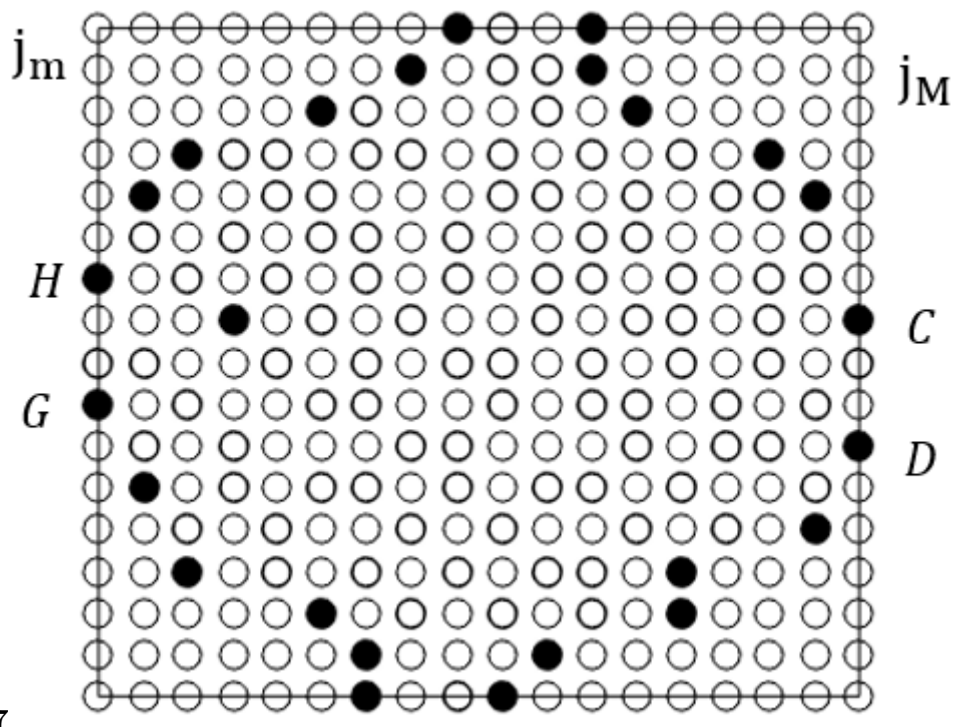
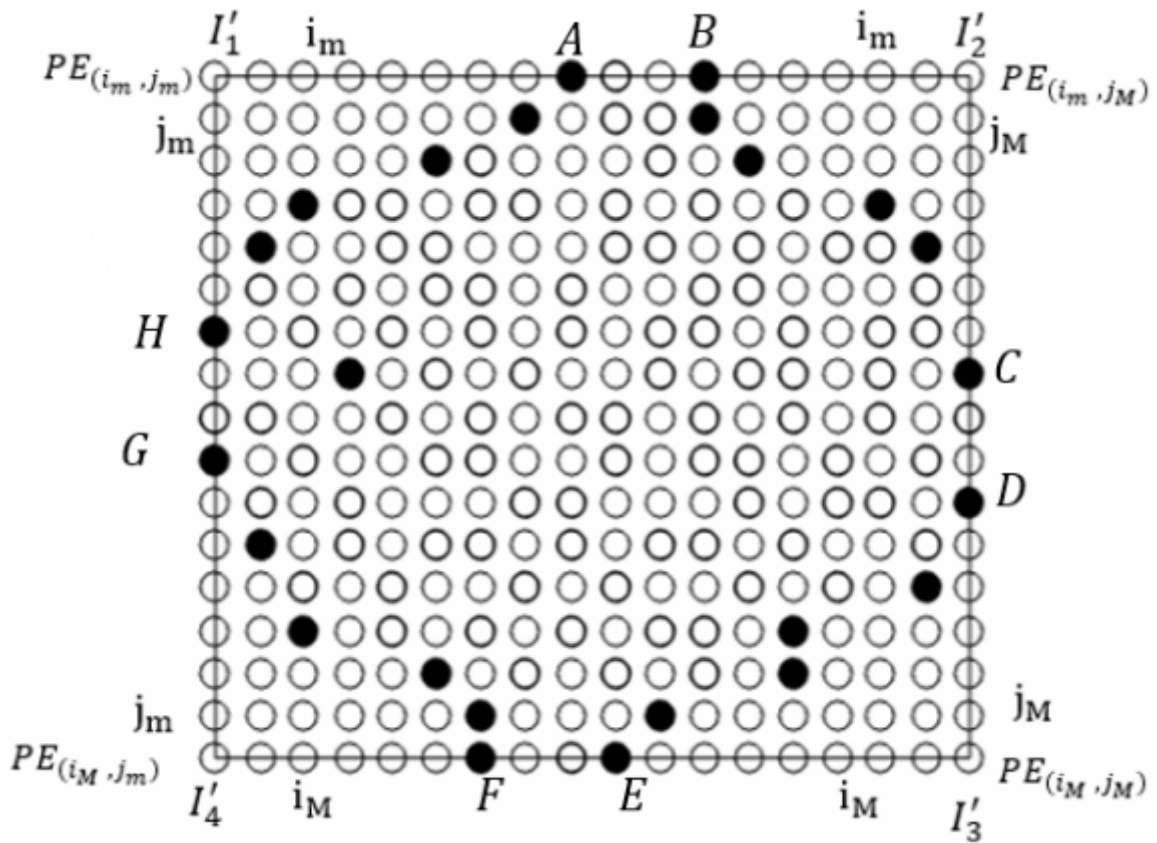


Figure 8:



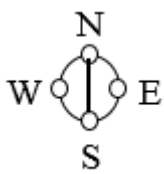
7

Figure 9: Figure 7 :



8

Figure 10: Figure 8 a:



6

Figure 11: Figure 6 :

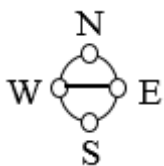
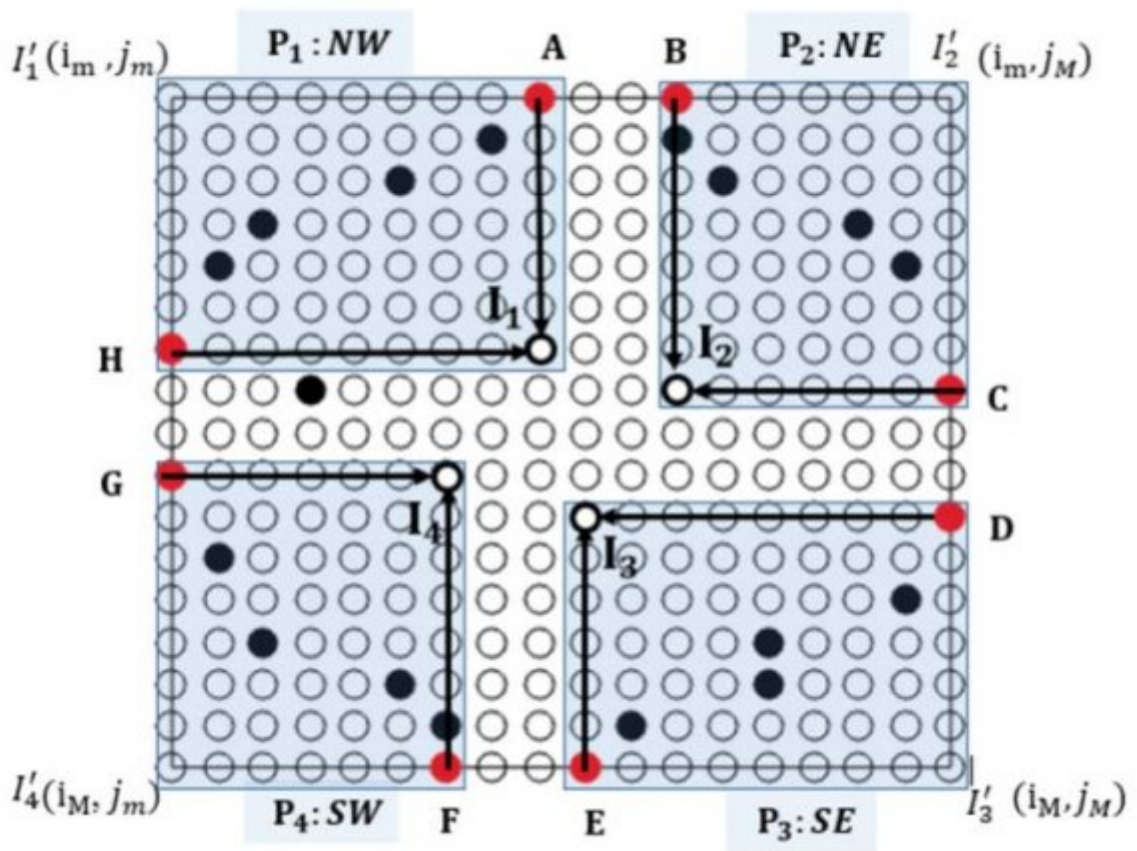


Figure 12: F



8

Figure 13: Figure 8 b



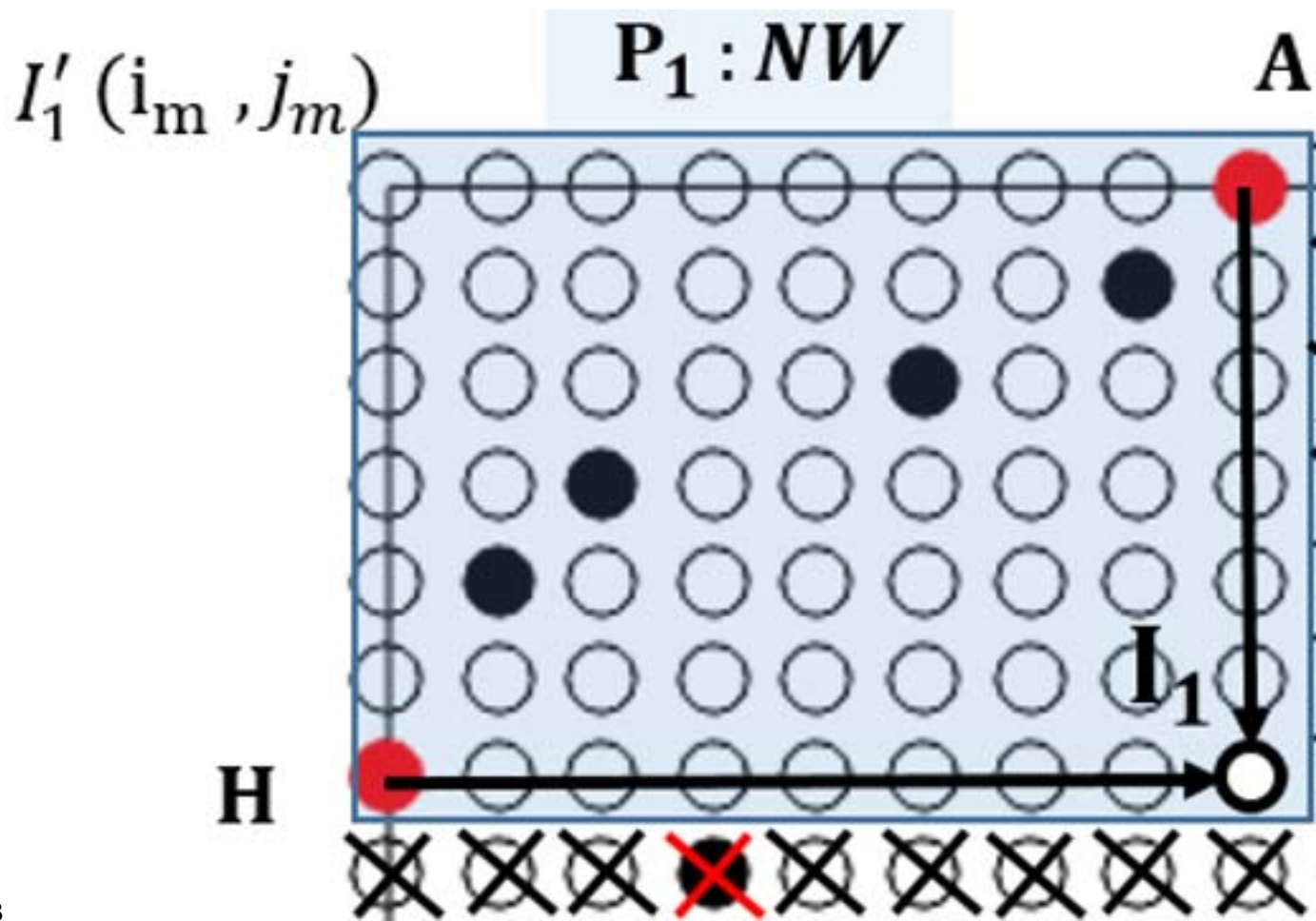


Figure 14: Figure 8 :

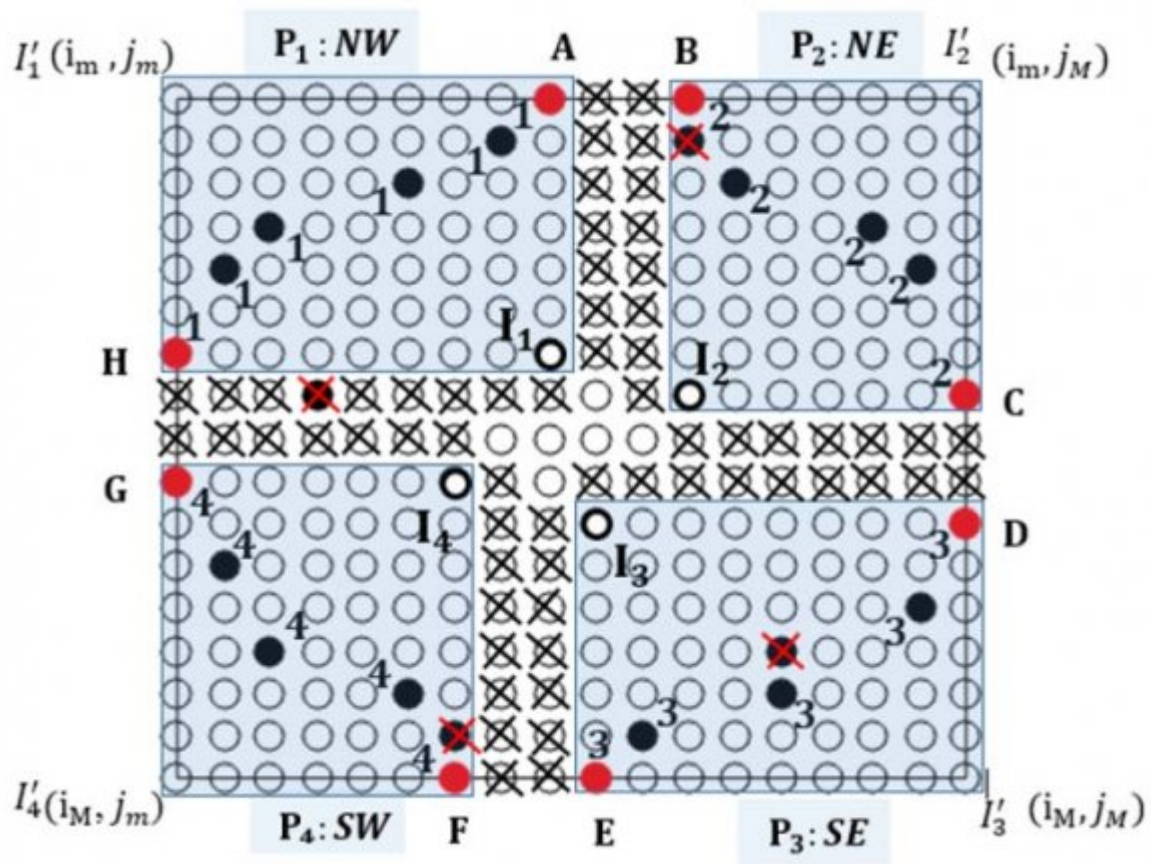


Figure 15:

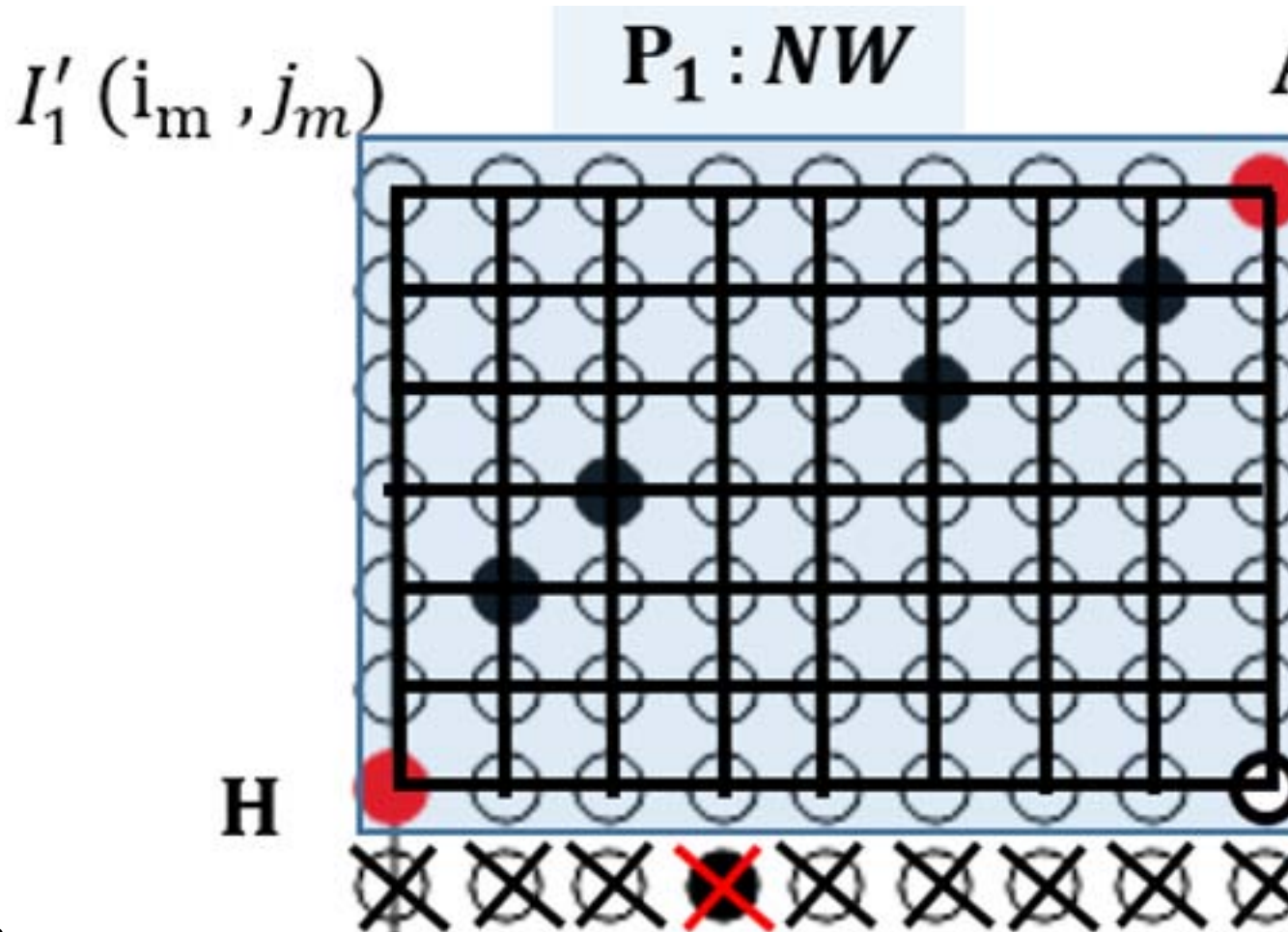


Figure 16: Step 7 :Figure 9 :Figure 10 :??

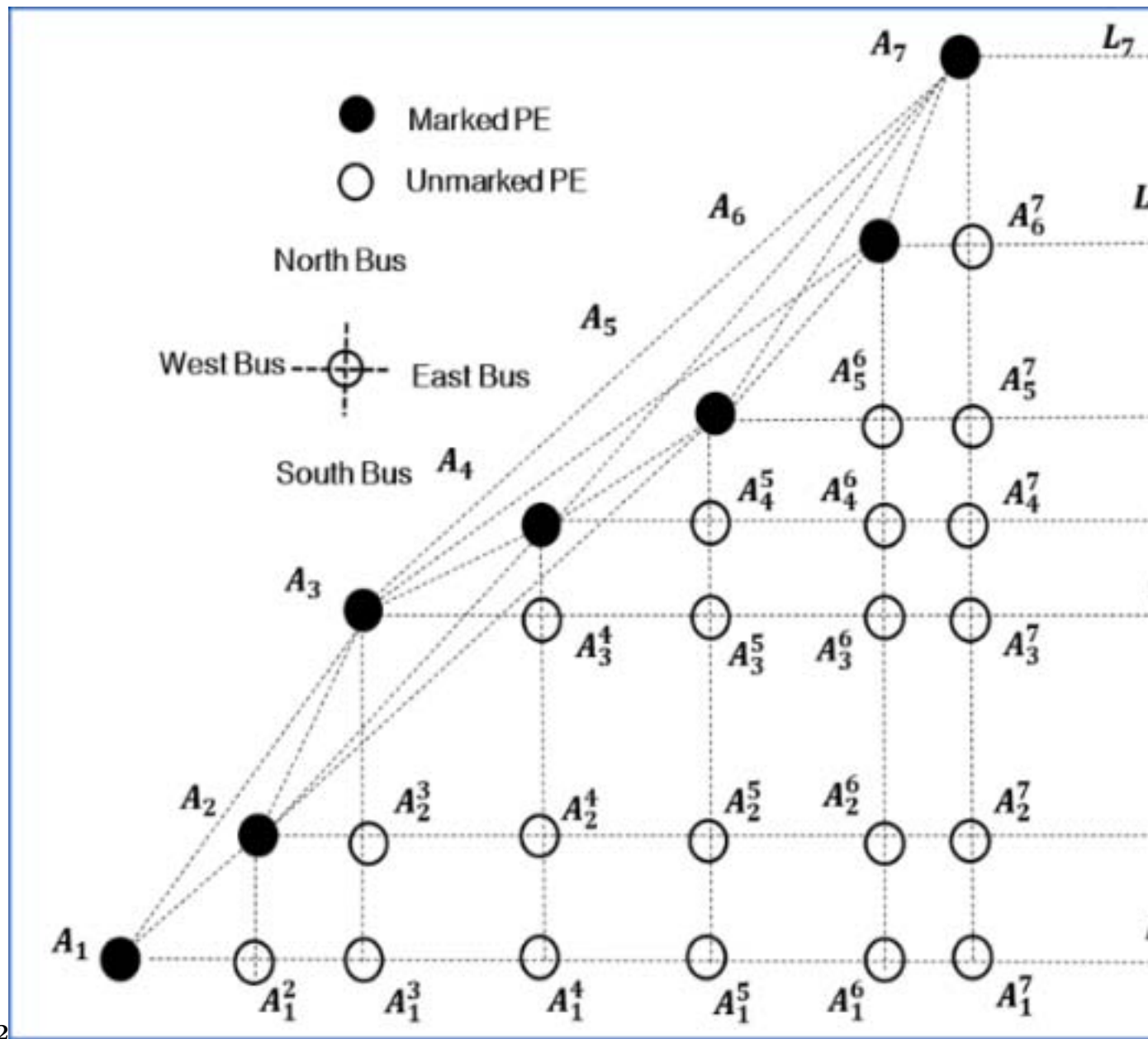


Figure 17: 2 .

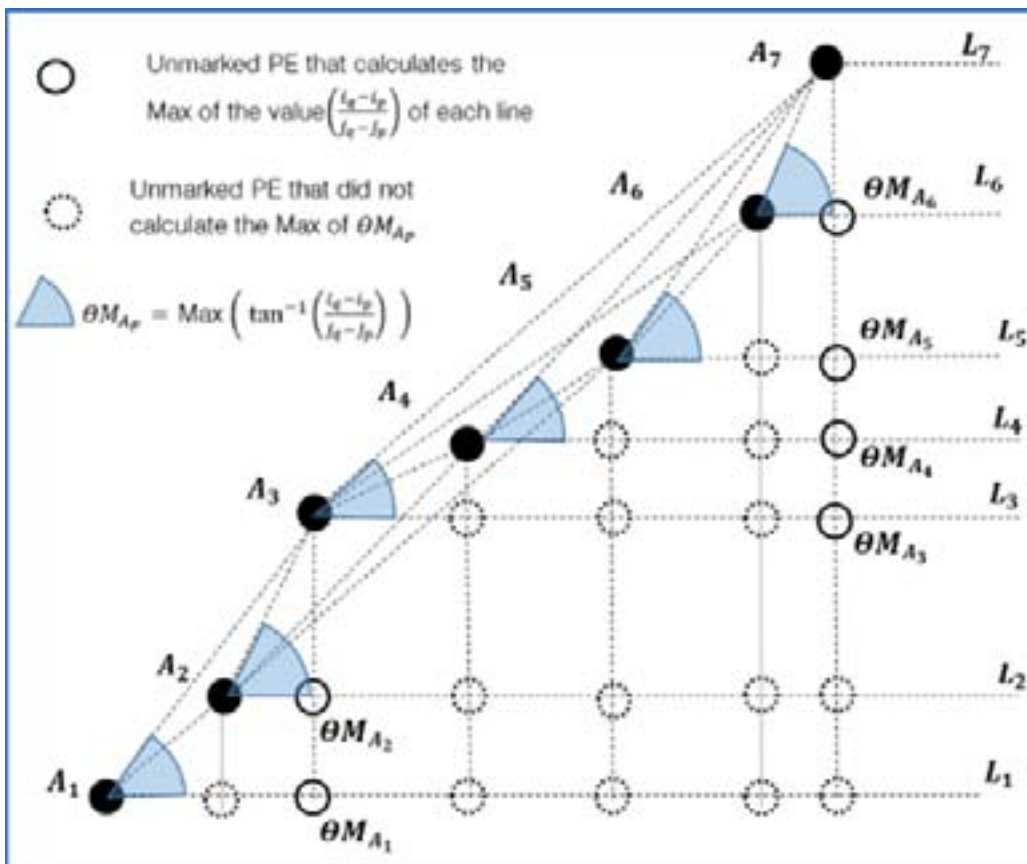


Figure 18:

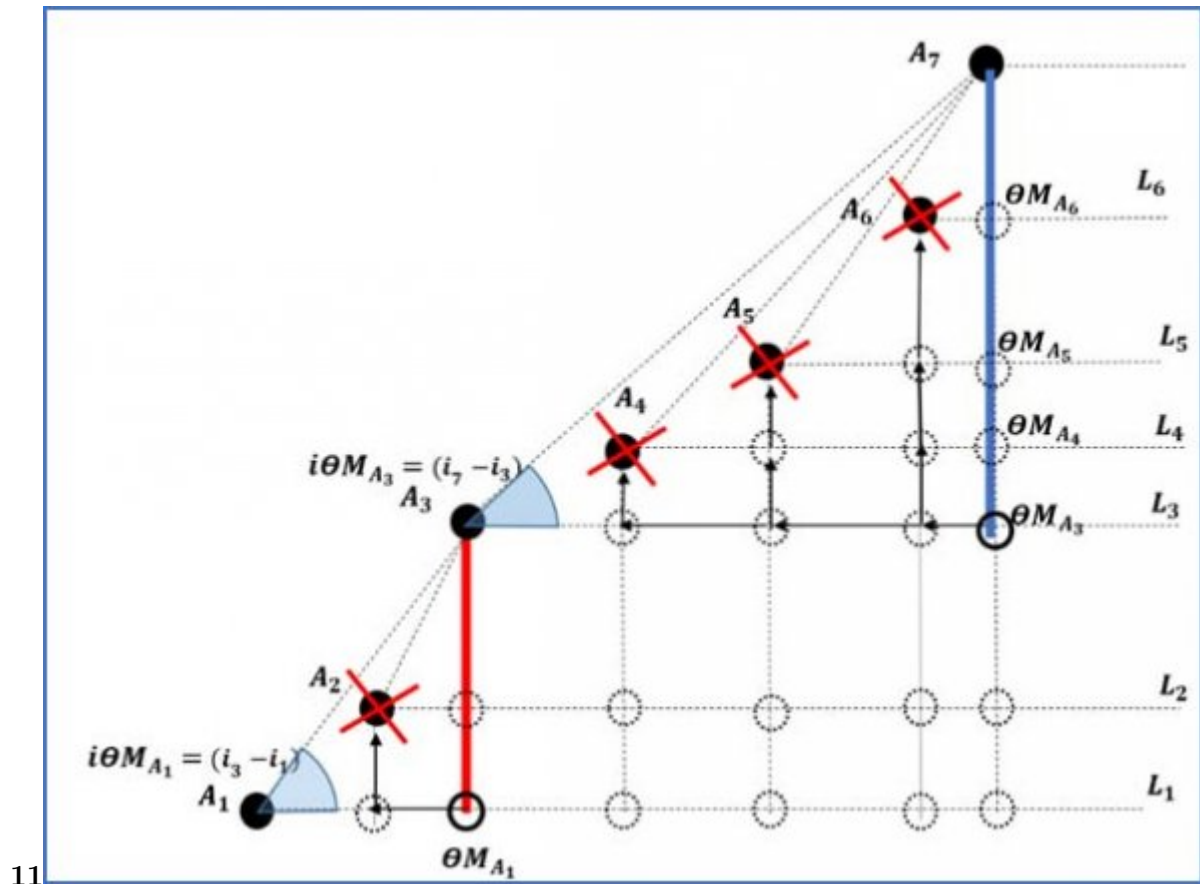


Figure 19: Figure 11 :

1

? The proposed method guarantees constant time processing because all steps are executed in ?" (1) time, while most existing algorithms are based on image processing multiple times.

? The proposed provides the convex hull not only for a set of points but also for images with two gray levels.

Figure 20: Table 1 :

---

103 [] , 10.1109/21.87089. <https://doi.org/10.1109/21.87089>

104 [J. El Mesbahi, ? ()] ‘(1) algorithm for image component labeling in a mesh connected computer’. *IEEE Trans.*  
105 *Syst. Man Cybern* J. El Mesbahi, ? (ed.) 1991. 21 p. .

106 [Errami A et al. ()] ‘? (1) time algorithm for structural characterization of multi-level images and its applications  
107 on a reconfigurable mesh computer’. Errami A , M Khaldoun , J Elmesbahi , BouattaneM . *Journal of*  
108 *Intelligent and Robotics Systems* 2005.

109 [Bouattane O et al. ()] ‘A fast parallel algorithm for convex hull problem of multi-leveled images’. Bouattane O  
110 , J Elmesbahi , Errami A . *Journal of Intelligent and Robotic Systems* 2002. 33 p. .

111 [Chow P ()] ‘A parallel algorithm for determining convex hulls of sets of points in two dimensions’. Chow P .  
112 *proc.19 th Allerton Conf. Common., Contr., Comput*, (.19 th Allerton Conf. Common., Contr., Comput)  
113 1981. p. .

114 [Hayachi T ()] ‘An  $(O(\log \log n)^2)$  time algorithm to compute the convex hull of sorted point on a reconfigurable  
115 meshes’. Hayachi T . *IEEE trans. Parallel Distrib. Sys* 1998. 9 (12) p. .

116 [LingT ()] ‘Efficient Parallel processing of image contours’. LingT . *IEEE Trans. Pattern Anal. Mach. Intelligence*  
117 1993. 15 (1) p. .

118 [Miller R ()] ‘Geometric algorithms for digitized pictures on a mesh connected computer’. Miller R . *IEEE Trans.*  
119 *Pattern Anal. Mach. Intelligence* 1985. 7 (2) .

120 [Prasanna V ()] ‘Image computation on meshes with multiple broadcast’. Reisis K D I Prasanna V . *IEEE Trans.*  
121 *Pattern Anal. Mach. Intelligence* 1989. 11 (11) p. .

122 [Ald ()] ‘Parallel algorithms for convex hulls’. M Ald . *Dep. Comput. Sci., Queens Univ* 1983.

123 [Kim C (1987)] *Parallel algorithms for digital geometry*, E Kim C . December 1987. Pullman. Washington State  
124 University (CS-87-179)

125 [Nath S et al. ()] ‘Parallel algorithms for the convex hull in two dimensions’. N Nath S , N Maheshwari , C P  
126 Bhatt P . *Proc. Conf. Anal. Problem Classes Programming Parallel Comput*, (Conf. Anal. Problem Classes  
127 Programming Parallel Comput) 1981. p. .

128 [Maresca M (March89)] ‘Polymorphic Torus Architecture for Computer Vision’. Li H Maresca M . *IEEE Trans.*  
129 *on PAMI* March89. 11 p. .

130 [References Références Referencias segment properties Pattern Recognition Lett ()] ‘References Références Ref-  
131 erencias segment properties’. *Pattern Recognition Lett* 1997. 18 p. .

132 [Elmesbahi and Cherkaoui ()] ‘Structure Analysis for Gray level Pictures on a mesh connected computer’. J  
133 Elmesbahi , J Cherkaoui . *Proc. of IEEE Internat. Conf. on SMC*, (of IEEE Internat. Conf. on SMC) 1986.  
134 10 p. .