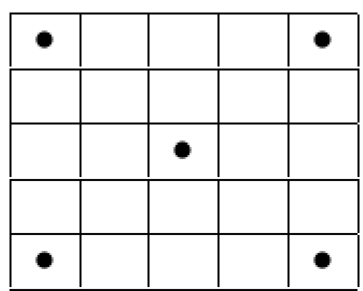
Optimal Separation on Two-Dimensional Arrays

By Jim Tao (Mentor: Dr. Wen-Qing Xu)

(i) Mathematics has always been important to me. When I was little, I liked to do mathematical puzzles out of a book I had bought from a catalog. I would think and ponder about these puzzles and discuss them with my family and friends. The answers were not straightforward, and I found the solutions interesting to read. Doing the puzzles, I discovered that mathematics is more than just a set of drills to memorize. It is a subject full of interesting, clever ideas. Ultimately, I grew up and became one of those "math nerds," distinguishing myself in various mathematical contests. I took the USA Math Olympiad, and I experienced firsthand the challenge of writing rigorous mathematical proofs.

Gradually, I became more interested in proof-based mathematics, so I pursued it further in a research internship at CSULB. My mentor, Dr. Wen-Qing Xu, had published several articles in the field of error-correction coding theory. Since that field of mathematics does not require the use of calculus and other collegiate mathematical preparations, but it does involve serious mathematical arguments, so I decided to pursue research in this area. I studied the separation of symbols on two-dimensional arrays, and came up with formulas for the maximum separating distance in various cases. I wrote proofs of my results, and spent many, many days discussing them with my mentor and revising them over and over again. It was intense, grueling work, but in the end it paid off. I entered my research into the Siemens Competition and Intel Science Talent Search, and I became a semifinalist in both competitions. I learned that I can indeed do mathematical research, even though it is sometimes tedious; research, after all, requires patience and hard work. My project made math more alive for me because I was able to experience the working day of a mathematician. As Paul Erdös once said, “A mathematician is a device for turning coffee into theorems.”

If you are a high school student and wish to undertake a project in math and science, I would advise you to start early, as soon as possible. I started my research about five months before the deadline of the Intel Science Talent Search, and that was not early enough. Sure, I got a semifinalist standing, but I know I could have done better. People who did better than me had started their projects much earlier than I did. Also, I would advise you to do a project in a subject you are interested in. I picked mathematics because I was, since childhood, fascinated with mathematics. As a result, it was not challenging for me to spend months and months doing mathematics. On the other hand, I saw people who gave up on their projects because they could not maintain a serious interest in the subject they were researching. If you would rather play flash games all day, you should not waste time pretending to do research.



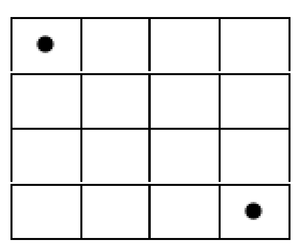
Optimal separation of 5 symbols on a 5×5 array with maximum distance of separation 4

(ii) How do you like this piece of art? It is dots on grids. It could potentially sell for billions upon billions of dollars. Now, we’ll get straight to the point. These dots are antisocial, and they want to be as far apart as possible. If we imagine them as pieces or symbols of the same data, we can see that, naturally, we do not want them near each other. Data corruption is localized, damaging data symbols only in its area of effect. If we separate the data out, we can make it so data corruption damages only one symbol at a time. That way, we facilitate fast and easy error correction through error-correcting codes [1], [3].

In this project, we study the maximum separating distance $D(m, n; l)$ of any number l of data symbols on any $m \times n$ array. By separating distance we mean the value of the smallest distance between any two symbols in a fixed configuration; if we make that smallest distance as large as possible, we end up spreading them out rather evenly on the array. Instead of the Euclidean distance as determined by the Pythagorean Theorem, we use the Manhattan distance. We measure the Manhattan distance between two points by adding together the absolute differences of their coordinates; the Manhattan distance between, say, (1, 2) and (5, 7) is (5 – 1) + (7 – 2) = 9. We may assume that the number of rows is less than or equal to the number of columns on the array. We prove an exact formula for the cases in which the number of symbols was two, three, four, or five, an exact formula for when there are too many data symbols, an exact formula for when there is a small number of rows, and upper and lower bounds for the general case.

We prove the following results in the paper.

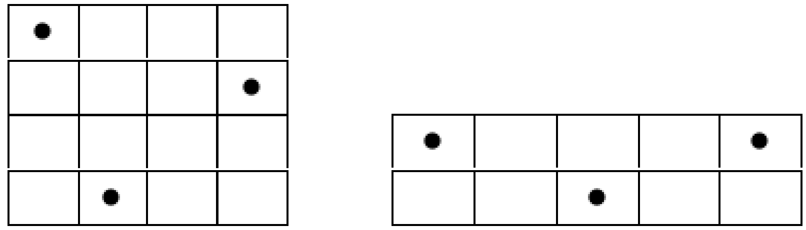
The formula for $l = 2$:



$$\mathcal{D}(m, n; 2) = m + n - 2$$

This case is intuitively obvious. If we are to separate two things as far as possible from each other in a rectangle, we place them at opposite corners of the rectangle. Though it may be trivial, as mathematicians call it, not mentioning it would constitute a serious omission.

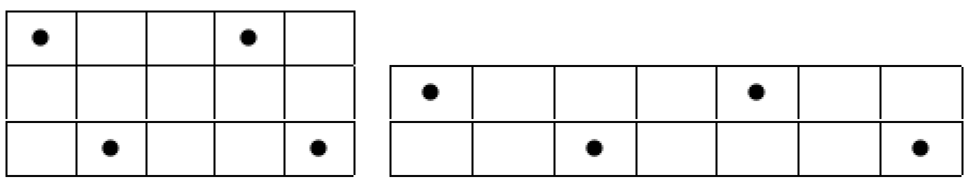
The formula for $l = 3$:



$$\mathcal{D}(m,n;3) = \begin{cases} \lfloor 2(m+n-2)/3 \rfloor & \text{if } n \le 2m-1 \\ m-1+\lfloor (n-1)/2 \rfloor & \text{if } n \ge 2m-1 \end{cases}$$

This case is much less obvious unless we draw the picture. As we can tell, the best arrangement is in a (roughly) equilateral triangle. When the array gets too flat or skinny, our arrangement degenerates into isosceles triangles.

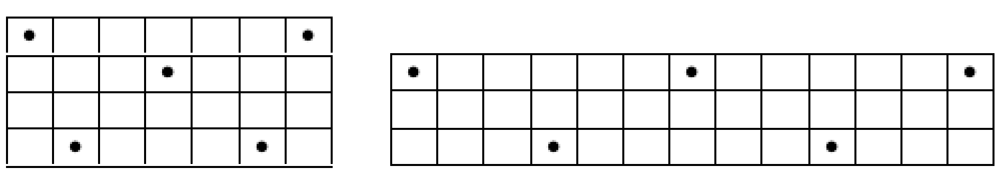
The formula for $l = 4$:



$$\mathcal{D}(m,n;4) = \begin{cases} \lfloor (m+n-2)/2 \rfloor & \text{if } n \le 3m-2 \\ m-1+\lfloor (n-1)/3 \rfloor & \text{if } n \ge 3m-2 \end{cases}$$

This case is much like the last one in that we must place them in a (roughly) equilateral quadrilateral or rhombus. When the array gets too flat or skinny, our arrangement degenerates into parallelograms.

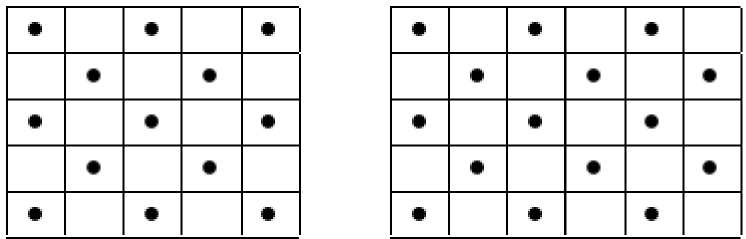
The formula for $l = 5$:



$$\mathcal{D}(m,n;5) = \begin{cases} \lfloor 2(m+\lceil n/2 \rceil-2)/3 \rfloor & \text{if } n \le 4m-3 \\ (m-1)+\lfloor (n-1)/4 \rfloor & \text{if } n \ge 4m-3 \end{cases}$$

This case is different from the previous ones. Now we start to see the use of concave polygonal arrangements, although they remain (roughly) equilateral. If the array gets too flat or skinny, our arrangement degenerates into isosceles trapezoids.

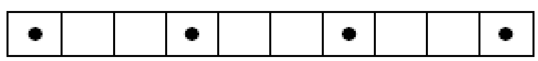
The formula for too many symbols:



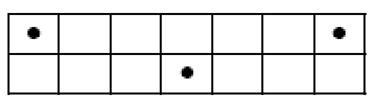
$$\text{Let } l > \lceil mn/2 \rceil. \text{ Then } \mathcal{D}(m,n;l) = 1.$$

In the above configurations, adding one more symbol forces $D(m, n; l)=1$.

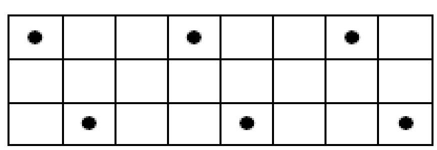
Formulae for small m (numbers of rows):



$$\mathcal{D}(1,n;l) = \lfloor (n-1)/(l-1) \rfloor$$



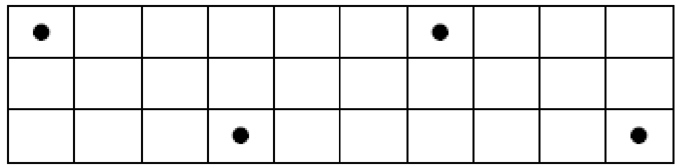
$$\mathcal{D}(2,n;l) = 1 + \lfloor (n-1)/(l-1) \rfloor$$



$$\mathcal{D}(3,n;l) = \begin{cases} 1 & \text{if } l > \lceil 3n/2 \rceil \\ 2 & \text{if } \lceil 2n/3 \rceil < l \leq \lceil 3n/2 \rceil \\ 3 & \text{if } \lceil n/2 \rceil < l \leq \lceil 2n/3 \rceil \\ 2 + \lfloor (n-1)/(l-1) \rfloor & \text{if } l \leq \lceil n/2 \rceil \end{cases}$$

For small numbers of rows, the patterns are simple, so we may easily find the formula for separating distance in those cases.

The zigzag bound:



$$\mathcal{D}(m,n;l) \leq (m-1) + \lfloor (n-1)/(l-1) \rfloor$$

This upper bound on separating distance comes from zigzagging through the array. This bound can actually be achieved when

$$n - 1 \geq (l-1)(m-1)$$

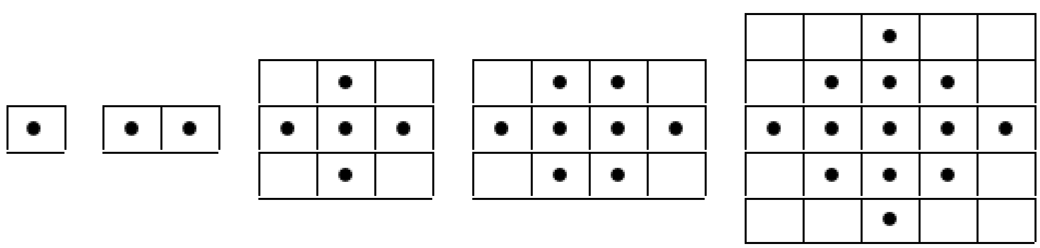
When that happens, we have

$$\mathcal{D}(m,n;l) = (m-1) + \lfloor (n-1)/(l-1) \rfloor$$

The sphere-packing bound:

$$\mathcal{D}(m,n;l) \geq d \text{ for any } d \in \mathbb{N} \text{ satisfying } (l-1)|\mathcal{S}_{2,d}| < mn.$$

This lower bound on separating distance comes from packing discrete 2-D spheres in the array.



2-D spheres $\mathcal{S}_{2,d}$ with $d = 1, 2, 3, 4, 5$

These are what we mean by discrete 2-D spheres. They represent areas damaged by data corruption. The formula

$$|\mathcal{S}_{2,d}| = \lceil d^2/2 \rceil$$

gives the number of cells covered by a discrete 2-D sphere [1], [2], [3].

This concludes our overview of the paper's results.

References

[1] M. Blaum, J. Bruck and A. Vardy, "Interleaving schemes for multidimensional cluster errors," IEEE Trans. Inform. Theory, vol. 44, pp. 730-743, 1998.

[2] S. W. Golomb and L. R. Welch, "Perfect codes in the Lee metric and the packing of polyominoes," SIAM J. Appl. Math., vol. 18, pp. 302-317, 1970.

[3] W.-Q. Xu and S.W. Golomb, "Optimal interleaving schemes for correcting two-dimensional cluster errors," Discrete Appl. Math., vol. 155, pp. 1200-1212, 2007.