A Survey of Minimal Rectangulations of Orthogonal Polygons with Holes

Sichang Li
University of Bristol

Abstract

A polynomial-time algorithm for finding the minimal rectangulation of orthogonal polygons with holes has been found independently by several authors. This article summarizes the key ideas behind these results and integrates them into a unified and accessible explanation. Instead of comparing individual papers, the goal is to present a clearer and more detailed account of the underlying principles, making the existing algorithms easier to understand for readers new to this topic.

1. *Introduction*

The problem of partitioning an orthogonal polygon in the plane with non-degenerate holes into a minimum number of rectangles is well-studied in computational geometry (for example [1], [2], [3]). However, these approaches are often presented in distinct contexts and with varying terminology, and their analyses can be mathematically involved for readers encountering the topic for the first time. This article aims to provide an accessible and consolidated explanation of minimal rectangulation. We review the necessary definitions, analyze the central ideas in detail, and illustrate the core principles with clear examples, presenting the topic in a unified and approachable manner.

2. *Minimal Rectangulations*

Definition 1.1. Let A be a region on the plane bounded by a finite number of points and disjoint horizontal and vertical line segments (called the *border*). The coordinates of each point are integers and all line segments are horizontal or vertical (Figure 1). We call such regions ***Orthogonal Polygons*** since they can be composed of several rectangles that are disjoint except along their sides.

Definition 1.2. We define the ***non-degenerate hole*** as a ***Orthogonal Polygon*** that has a positive area, which means it cannot degenerate into either a line or a point.

Let A be an orthogonal polygon in the plane with non-degenerate holes. It can be regarded as an orthogonal polygon which contains several (possibly 0) "small" orthogonal polygons (Figure 1). We do not consider the situation where there are third-level holes inside a hole. This is because starting from such a third-level hole, it is no longer connected to the outermost region (i.e., the polygon we want to divide), so it can be treated as an independent A' and undergo the same division operations as the outer one.

We know that, in an orthogonal polygon A, every two intersecting line segments with their joint point make a corner, and there is a 90° angle on one side of this corner and a 270° angle on the other side. We use the same labeling method as Michael R. Klug [2].

Definition 2. We call a corner ***internal corner*** if the 270° angle is inside A (marked in red in Figure 1), and call a corner ***external corner*** if the 90° angle is inside A (marked in blue in Figure 1).

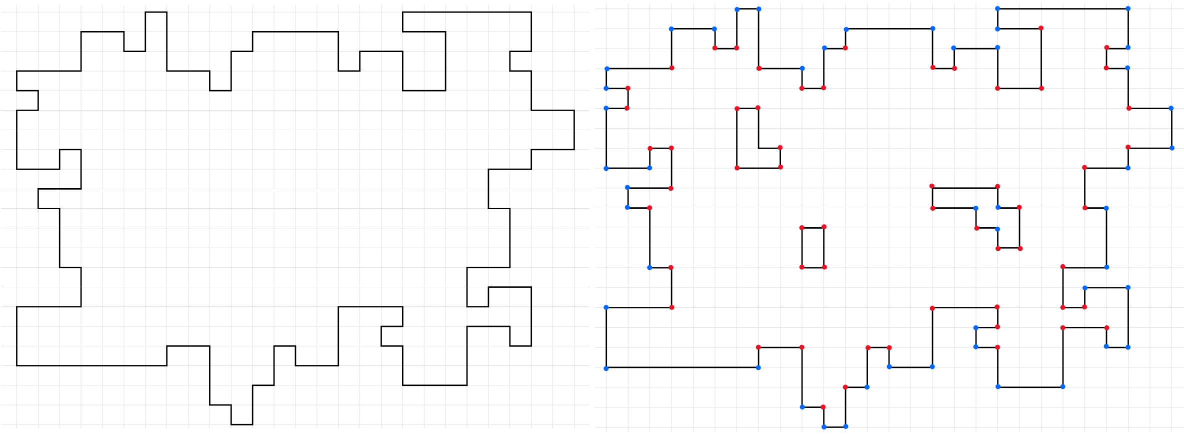

Figure 1: Example of ***Orthogonal Polygons*** (left). Example of orthogonal polygons in the plane with non-degenerate holes, with the ***internal corners*** marked in red, and the ***external corners*** marked in blue (right).

$Definition$ 3.1. Let L_u be a set of all the line segments l_u that at least 1 endpoint of l_u is ***internal corner*** (except the *border* of A). Let L be the set of $l, l \in L_u$ that all $l \in L$ cut A up into multiple rectangles (e.g. all the dashed line in Figure 2). The size of L, denoted $|L|$, is the number of $l, l \in L$.

$Definition$ 3.2. We call a line segment $l_c, l_c \in L$ the ***corner line*** if the two endpoints of l_c are both ***internal corner*** (e.g. the vertical dashed line in Figure 2). Let L_c be a set of all the ***corner lines*** in L, and the size of L_c , denoted $|L_c|$, is the number of l_c that $l_c \in L_c$.

$Definition$ 4. Let $R(L)$ be a set of all rectangles produced by the cutting lines $l, l \in L$ (called a valid cutting). The size of $R(L)$, denoted $|R(L)|$, is the number of rectangles that L cuts A into (Figure 2). A rectangulation $R(L_m)$ is *minimal* if $|R(L_m)|$ is the smallest among all rectangulations.

Symbol	Explanation	Examples
$l, l \in L_u$	Line segment that at least 1 endpoint of it is ***internal corner***	All the dashed lines in Figure 2
$l, l \in L$	$1.$ Line segments that cut A up into multiple rectangles (a valid cutting) $2.$ $\|L\|$ means the number of lines that in L	All the dashed lines in Figure 2
$l_c, l_c \in L_c$	$1.$ Line segment that the 2 endpoints of it are both ***internal corner*** $2.$ $\|L_c\|$ means the number of lines that in L_c	All the vertical dashed lines in Figure 2
$R(L)$	Rectangles that A be cutted into by L	All the single rectangles in Figure 2

$Definition$ 5. Now we introduce some quantities. n_e is the number of the ***external corners*** in A. n_g is the number of ***internal corners*** u that $\exists l_c \in L_c.\ u \in l_c$ (called *good corner*). n_b is the number of ***internal corners*** v that $\forall l_c \in L_c.\ v \notin l_c$ (called *bad corner*). n_i is the number of the ***internal corners*** in A. n is the number of all corners in A. Also, we have $n = n_e + n_i = n_e + n_g + n_b$.

One of the simplest solutions of rectangulation is connecting all the vertical ***corner lines***, and then for all the *bad corners*, draw a horizontal line which joined with *borders* or ***corner lines*** at some point. We use horizontal lines for illustration, although vertical lines work analogously. (Figure 2)

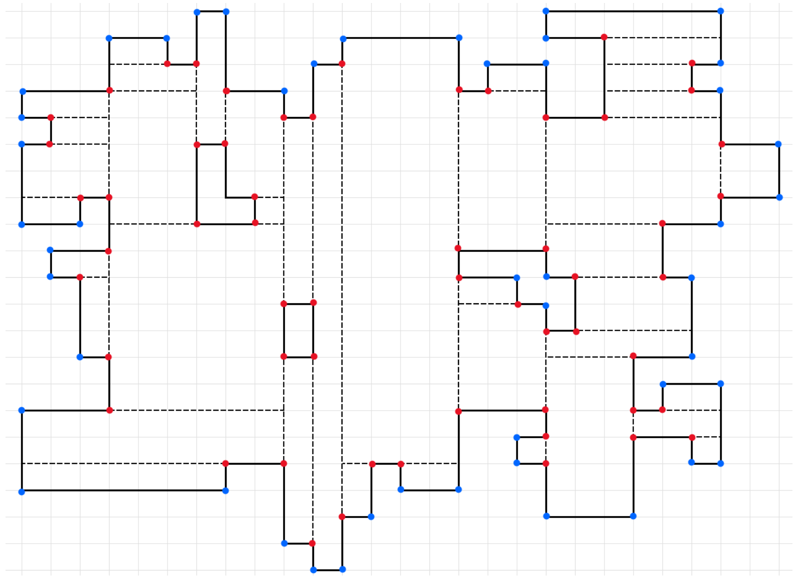

Figure 2: A example of rectangulations $R(L)$. (It may not be the smallest R, but it is a valid one.) In this example, $|L| = 41$, $|L_c| = 17$ and $|R(L)| = 39$.

Lemma 1. Let $R(L)$ be a valid cutting of A, then

$$4|R(L)| \geq 2|L_c| + 3n_b + n_e$$

Proof. We call the corner of a rectangle *rectangle corner*. Since each *rectangle corner* comes from a corner of A or the joint point of two lines (border or l, $l \in L_u$), and each joint point comes from a line drawn from an ***internal corner***, so we analyze each corner individually.

If $\forall l \in L$ satisfy that $\forall l_i, l_j \in L$. If $i \neq j$, then $l_i \cap l_j = \emptyset$ or $l_i \cap l_j = p$ & p is an end point. Then, for each l_c (represents two *good corners*), it will make 2 *rectangle corners* (Figure 3 1.1 and 1.2). For each *bad corner*, it will make 1 *rectangle corner* besides it (Figure 3 2.1), and 2 *rectangle corners* beside the joint point (Figure 3 2.2 and 2.3). For each ***external point***, it will make 1 *rectangle corner* (just itself) (Figure 3 3.1).

Since each rectangle has 4 *rectangle corners*, and these *rectangle corners* come from corners and joint point. So the lower bound of the number of *rectangle corners* rectangles have is equal to the number of that made by corners. So,

$$4|R(L)| \geq 2|L_c| + 3n_b + n_e$$

If and only if all lines $l \in L$ satisfy the restriction as mentioned, there holds

$$4|R(L)| = 2|L_c| + 3n_b + n_e$$

In this case, $|R(L)|$ is smaller, so we let all L in the following text satisfy this restriction, and the above equation always holds.

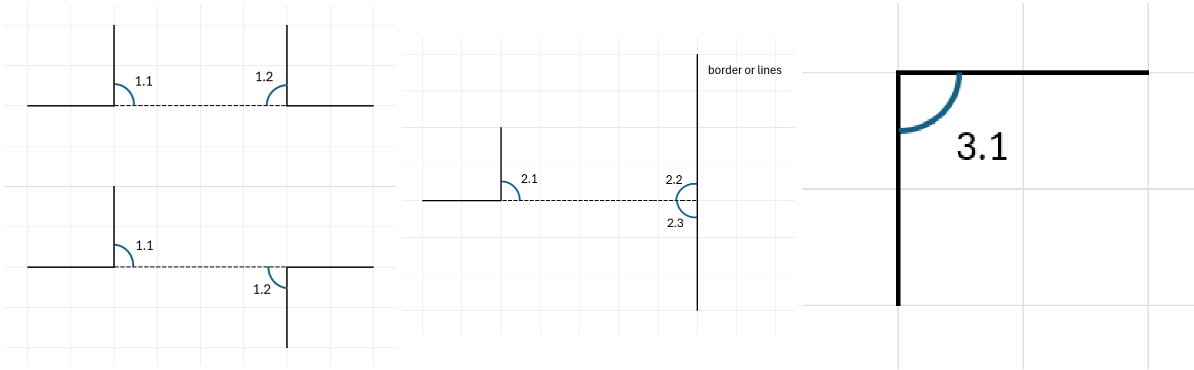

Figure 3: Examples of *rectangle corners* made by different types of corners.

$Lemma\ 2$. The number of ***internal corners*** in any orthogonal polygon with h holes is

$$n_i = \frac{n}{2} + 2h - 2$$

$Proof$. In any simple orthogonal polygon, every interior angle is either 90° (a ***external corners***) or 270° (a ***internal corners***). At each convex vertex, the polygon turns left by 90°, contributing +90° to the total turning. At each reflex vertex, it turns right by 90°, contributing -90°. Since the sum of all exterior turning angles around any simple closed curve is 360°, then

$$90^\circ n_e - 90^\circ n_i = 360^\circ$$

So,

$$n_e - n_i = 4$$

Then, for each hole, which is also a simple closed curve, $n_e - n_i = 4$ also holds. But in this case, the ***internal corners*** of holes will be the ***external corners*** of the outside orthogonal polygon and vice versa. So,

$$n_e - n_i = 4 - 4h$$

$$n_e = n_i - 4h + 4 = n - n_i$$

$$2n_i = n + 4h - 4$$

$$n_i = \frac{n}{2} + 2h - 2$$

$Lemma\ 3$. the minimum number of rectangles $|R(L_m)|$ in the partition of an orthogonal polygon with n corners and h holes is

$$|R(L_m)| = \frac{n}{2} + h - \max\{|L_c|\} - 1$$

$Proof$. Since each l_c contains 2 *good corners*, so $n_g = 2|L_c|$, then

$$\begin{aligned} 4|R(L)| &= n_g + 3n_b + n_e \\ &= (n_g + n_b + n_e) + 2(n_g + n_b) - 2n_g \\ &= n + 2n_i - 4|L_c| \end{aligned}$$

By Lemma 2, we have

$$n_i = \frac{n}{2} + 2h - 2$$

and therefore

$$
\begin{aligned}
4|R(L)| &= n + 2(\frac{n}{2} + 2h - 2) - 4|L_c| \\
&= 2n + 4h - 4|L_c| - 4
\end{aligned}
$$

$$|R(L)| = \frac{n}{2} + h - |L_c| - 1$$

Since for a given A, both n and h are constants. So the aim to find the minimal rectangulations (make $|R(L)|$ the smallest one), is equal to maximize $|L_c|$. So,

$$|R(L_m)| = \frac{n}{2} + h - \max\{|L_c|\} - 1$$

This conclusion is also mentioned in David Eppstein's literature [1].

$3.\ Maximum\ Independent\ Set$

$Definition\ 6$. Take two ***corner lines*** l_{c1}, $l_{c1} \in L_c$ and l_{c2}, $l_{c2} \in L_c$. We say that l_{c1} and l_{c2} intersect if $\exists p \in l_{c1}\,.\, p \in l_{c2}$.

Now we mark each ***corner lines*** l_{ci}, $l_{ci} \in L_c$ as a point p_i, and draw an edge connecting the points corresponding to two intersecting ***corner lines***. (Figure 4) Then, find a valid L_c is equal to find a set of points S, that there is no direct edge connection between every pair of point $p_i, p_j \in S \& i \neq j$, which is exactly the definition of an independent set in graph theory. Then find the maximum $|L_c|$ if equal to find a maximum independent set in a given graph, which is a strongly NP-hard problem ([4], [5]).

Luckily, since the horizontal ***corner lines*** can only intersect with vertical ***corner lines*** and vice versa, so the problem translates into find a maximum independent set in a bipartite graph. (Figure 4) By Kőnig's theorem [6], in any n vertex bipartite graph G, the size of the maximum independent set of G is $n - m$, where m is the size of a minimal vertex cover of G (also equal to the size of a maximum matching of G). So, according to Kőnig's theorem, we can find the maximum independent set in polynomial time, and find the minimal rectangulation in polynomial time.

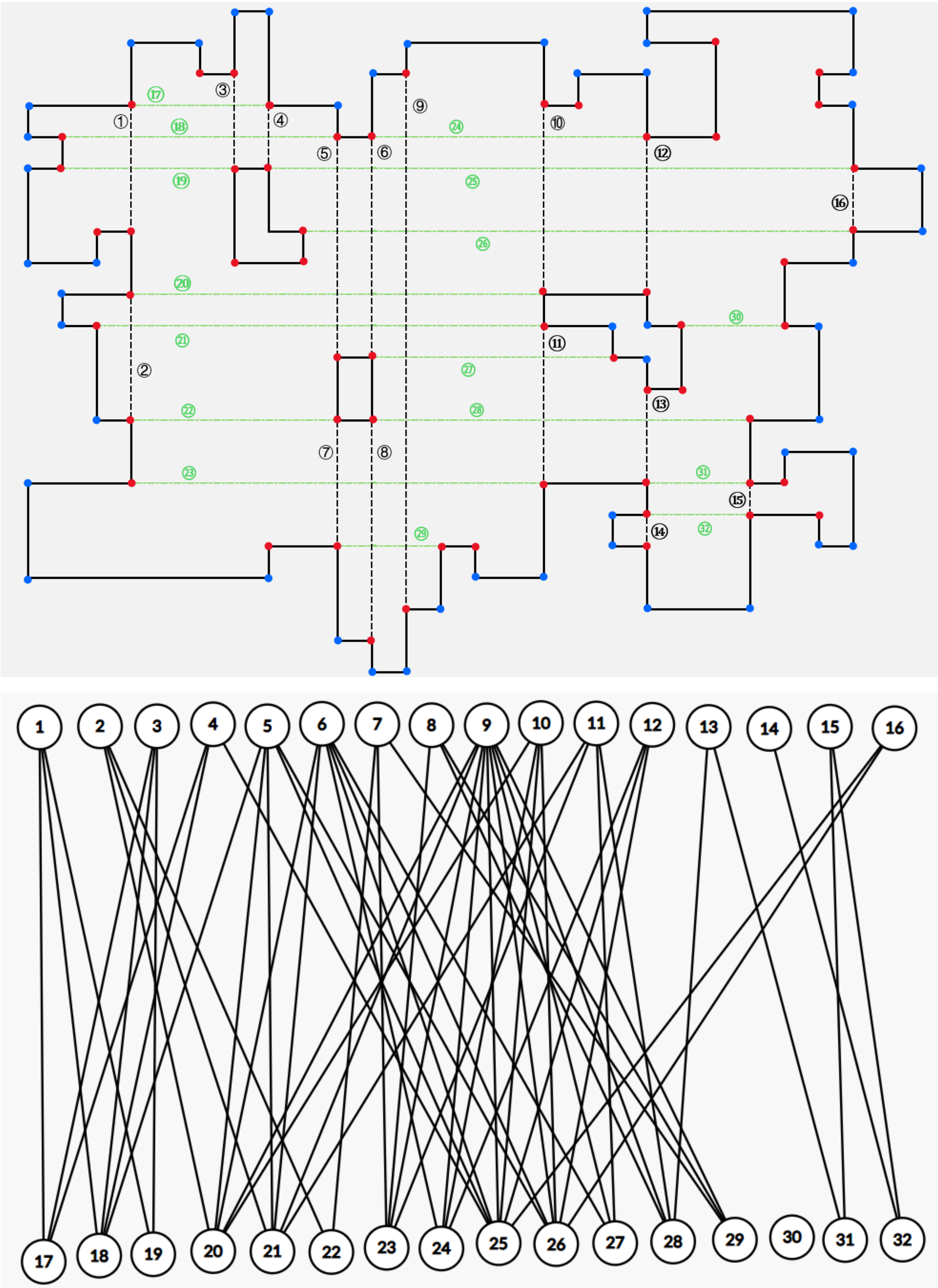

Figure 4: The First one shows an example of all ***corner lines*** in a polygon A. The horizontal ***corner lines*** are marked by green dashed lines, the vertical ***corner lines*** are marked by black dashed lines. The second one shows the corresponding bipartite graph.

4. *Time Complexity Analysis*

Suppose we have an orthogonal polygon with holes which has n corners. First, discretize the coordinates of the corners, then use queues or stacks to maintain the vertex coordinates of each column and row, which allows us to quickly obtain all the ***corner lines*** and determine whether they intersect. The time compexity of processing the input polygon and building a bipartite graph is $\Theta(n \log n)$.

Then, apply the Hopcroft–Karp matching algorithm [7] to the bipartite graph G, the time compexity of finding the maximum matching is $O(|V|^{\frac{1}{2}}|E|) = O(n^{2.5})$, since there may be $\Theta(n)$ vertices and $\Theta(n^2)$ edges. Moreover, David Eppstein mentioned that "by using geometric range searching data structures to speed up the search for alternating paths within the matching algorithm, it is possible to improve the overall running time to $O(n^{\frac{3}{2}} \log n)$" in his literature [1].

Last, connect the ***corner lines*** that are in the maximum independent set. Then, for the rest *bad corners*, draw a horizontal or vertical line which joined with *borders* or ***corner lines*** as mentioned above. The time complexity of this process is $\Theta(n \log n)$.

In summary, the overall time complexity is $O(n^{\frac{3}{2}} \log n)$, the overall memory complexity is $O(n^2)$.

4. *Open Problems and Future Directions*

4.1. **Application in Image Storage**

Normally, images are stored based on pixels, with each pixel storing a data to represent its color. We can process the image with different colors, and mark the pixels that match the target color as 'inside the polygon' and those that do not as 'outside'. In this way, the input image is transformed into multiple orthogonal polygons with holes as described in this article. Then, by cutting all the polygons, each color is eventually divided into multiple rectangles, and we can store the image by storing descriptions of these rectangles (e.g. the coordinates and/or the sizes of rectangles). This method can significantly reduce image storage memory when the resolution is high or the number of colors is relatively small.

Although rectangulation-based compression could reduce memory usage in images, it is still an open question how to generalize this method to real-world noisy images, where the number of reflex vertices may become very large. Developing some preprocessing or hierarchical rectangulation strategies maybe a possible direction.

4.2. **High-dimensional Expansion**

If possible, we can extend this algorithm or approach to higher-dimensional cuts. Like cutting an irregular solid shape into several cuboids and cubes. Predictably, it would greatly advance fields such as chemistry, physics, and architecture. Like three-dimensional molecular synthesis, structural stress analysis in construction, and the analysis of higher dimensions in physics could all potentially benefit from it.

However, since the relationships between the faces of 3D orthogonal polyhedra become much more complex, it remains open whether a polynomial-time algorithm exists for the "minimal cuboid division" of orthogonal polyhedra with and without holes. Also, proving this is an NP-hard problem is also a possible direction.

References

[1] David Eppstein. 2009. *Graph-Theoretic Solutions to Computational Geometry Problems. arXiv:0908.3916.*

[2] Michael R. Klug. 2023. *Finding Minimal Rectangulations.* (Unpublished) Retrieved from https://math.uchicago.edu/~michaelklug/rectangles.pdf.

[3] Valeriu Soltan. and Alexei Gorpinevich. 1992. *Minimum dissection of rectilinear polygon with arbitrary holes into rectangles.* In *the 8th Annual Computational Geometry* (SCG '92). ACM, 296–302. DOI: 10.1145/142675.142735.

[4] Richard M. Karp. 1972. *Reducibility among combinatorial problems*. In Complexity of computer computations.

[5] M. R. Garey. and D. S. Johnson. 1978. *"Strong" NP-Completeness Results: Motivation, Examples, and Implications*. Journal of the ACM. 25 (3): 499–508. doi:10.1145/322077.322090. ISSN 0004-5411. S2CID 18371269.

[6] Dénes Kőnig. 1931. *Gráfok és mátrixok*. Matematikai és Fizikai Lapok, **38**:116–119.

[7] John E. Hopcroft and Richard M. Karp. 1973. *An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs*. SIAM J. Comput., 2(4):225–231.