

# PCB Component Copper Landing Pad Design Optimization

Hsiao-Chieh Ma, Yi-Yu Liu

Department of Computer Science and Information Engineering,  
National Taiwan University of Science and Technology, Taipei City, Taiwan, R.O.C.  
b10715041@mail.ntust.edu.tw, yyliu@mail.ntust.edu.tw

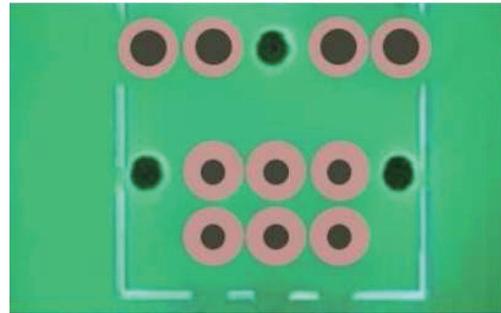
## ABSTRACT

**Abstract**— As the density of electronic components increases in modern PCB designs, the adjustment of copper landing pads has become a complex and essential issue during PCB layout design stage. Common copper landing pad adjustment strategies are optimized by experienced PCB layout engineers. In this paper, we propose an optimization framework to legalize copper landing pads via pad offset, pad cutting, and pad shrinking operations, with minimal pad distortion. The experimental results demonstrate the effectiveness to significantly reduce the manual task of PCB layout engineers for time and effort saving.

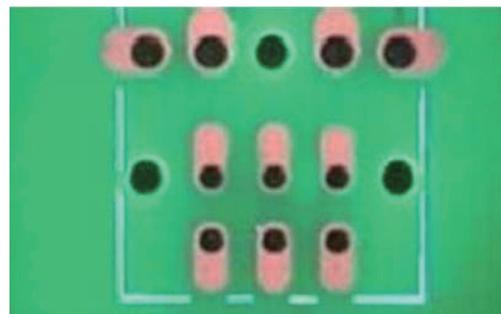
## I. INTRODUCTION

With the advancement of PCB and semiconductor manufacturing technology, the integration capability of both printed circuit boards (PCB) and electronic components continue to scale up. Therefore, the density of electronic components as well as copper landing pads have become a complex design issue during PCB layout design stage [1]. Owing to various design parameters limited by different PCB manufacturers, the copper landing pads are required to be adjusted accordingly. Common copper landing pad adjustment strategies include pad offset, pad cutting, and pad shrinking. Figure 1 depicts the adjustment of PCB copper landing pads. These methods involve in a wide range of manufacturability and reliability considerations and are therefore optimized by experienced PCB layout engineers. However, manual design requires a lot of engineering time and effort. The resultant layout is error-prone and may suffer reliability degradation. In this work, we propose a two-stage optimization framework to adjust copper landing pads. The first stage analyzes the design-rule violations of a PCB layout and constructs a corresponding conflict graph. The second stage develops pad offset, pad cutting, and pad shrinking operations to legalize all copper landing pads. To the best of our knowledge, this is the first systematic approach to tackle the copper landing pad adjustment problem with minimal landing pad distortion.

The rest of this paper is organized as follows. Section II introduces the design issues of printed circuit boards. Section III presents our design optimization framework.



(a) Original copper landing pads with design-rule violations.



(b) Optimized copper landing pads.

Fig. 1. Copper landing pad optimizations. (Images courtesy of Footprintku Inc.)

The experimental results are illustrated in Section IV. Finally, Section V concludes our work.

## II. PRELIMINARY

A PCB is a carrier to assemble multiple electronic components and to provide reliable electrical connections between component landing pads [4]. A PCB is made of laminated insulating material and with conducting material (e.g., copper) upon/between the insulating layers. The primary function of PCBs is to affix all electronic components in the predefined locations by soldering and to interconnect signals and power/ground between these components.

There are two typical mounting methods: the through-hole mounting and the surface mounting [3]. Components using through-hole mounting technology are mounted on the board by passing their leads through drilled via holes

in PCB. The through hole indicates a hole that runs from the top side of a PCB to the bottom side of a PCB, where all intermediate layers are vertically drilled by using either drilling machines or lasers. A plated-through hole (PTH) is surrounded by copper to interconnect desired layers. Most of the PTH are in circular and oblong shapes. Rectangular holes, with a relatively expensive manufacturing cost, are less commonly used. The surface-mount technology (SMT) is a technique of soldering electronic components to the surface of a PCB [2]. The main difference between the two technologies is that the SMT does not need to reserve corresponding through holes for the component pins. In addition, the component form factor using SMT technology is much smaller than that of using the through-hole technology. The electronic components using surface-mount technology are called surface-mount devices (SMD).

In order to maintain the PCB manufacturability, two design-rule constraints, the minimum gap between two coppers landing pads and the minimum width of a PTH copper landing pad to its corresponding enclosed drill, must be satisfied. To meet the design-rule constraints from different PCB manufacturing suppliers, we need to adjust some copper landing pads. Owing to the small form factor of SMDs, the adjustments of SMD copper landing pads could reduce the adhesion force of the electronic components. Therefore, all SMD copper landing pads are kept unchanged.

The copper landing pad optimization problem is that given a PCB design consisting of PTH copper landing pads with drills (i.e., the through hole in the middle of PTH), SMD copper landing pads, and design-rule constraints, the PTH copper landing pads are adjusted to avoid design-rule violations. In addition to the design rules, there are some important metrics related to manufacturability and reliability. The outline shapes of legalized copper landing pads should be similar to the original ones for better optical and visual recognition. Similarly, the overlay regions of copper landing pads before and after legalization should be kept as large as possible. Finally, the area of legalized copper landing pads should be kept as same as possible to maintain enough adhesion force.

### III. LAYOUT DESIGN OPTIMIZATION

Our design optimization algorithm consists of two stages, the pre-processing stage and the legalization stage. The pre-processing stage verifies design-rule violations between neighboring landing pads and constructs a conflict graph for the PCB design [7]. In the legalization stage, we develop three operations to legalize the design. Figure 2 illustrates the concepts of pad offset, pad cutting, and pad shrinking operations. Figure 2(a) shows two PTHs that violate the minimum gap constraint. The circles colored in *blue* represent the copper landing pads of PTHs, and the circles colored in *green* within the landing

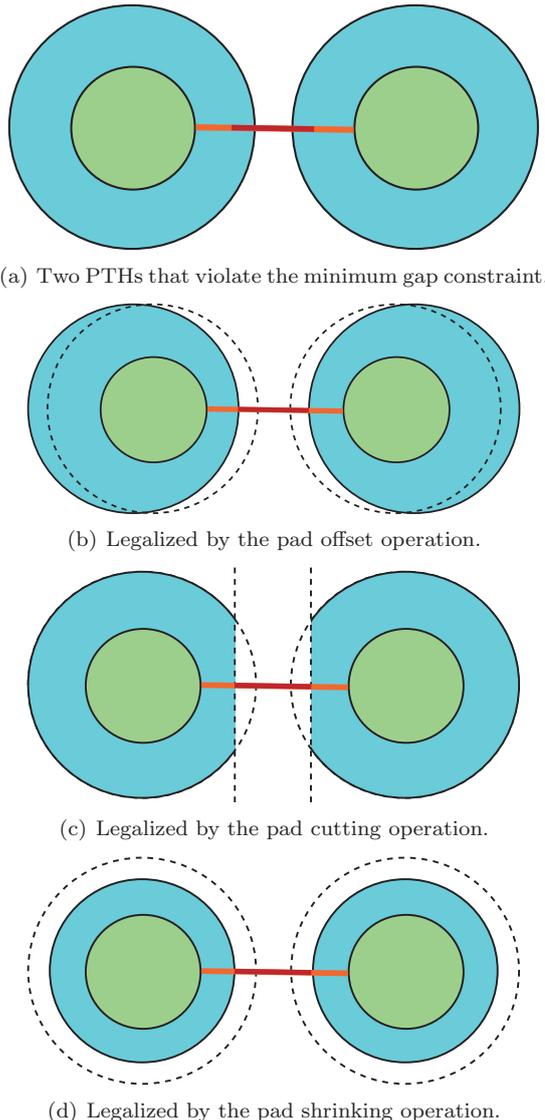


Fig. 2. Legalization operations. (Images courtesy of Footprintku Inc.)

pads are through-hole drills. The *red* and *orange* lines represent the minimum gap and minimum width constraints, respectively.

#### A. Pre-processing Stage

First, we have to pair all PTH copper landing pads to their corresponding enclosed drills, respectively. That is, we identify which drill is inside a copper landing pad and then verify the minimum width constraint for all enclosed drills. To do this, we only need to check if any point of the drill is inside the copper landing pad since the whole drill should be inside the pad if they are paired [5]. According to Jordan Curve Theorem, a point is said to be inside a polygon if, for any ray from this point, there is an odd number of crossings of the ray with the polygon's edge [6]. This rule still holds even if we have arcs in our polygon.

After that, we find out all nearby neighbors, which might violate the minimum gap constraint, of each landing pad so that we don't need to traverse all landing pads in follow-up optimization operations. We can verify the minimum gap constraint of two PTH neighbors by finding the shortest distance between the respective boundary lines. Similarly, we can verify the minimum width constraint by finding the shortest distance between the a PTH and its corresponding drill.

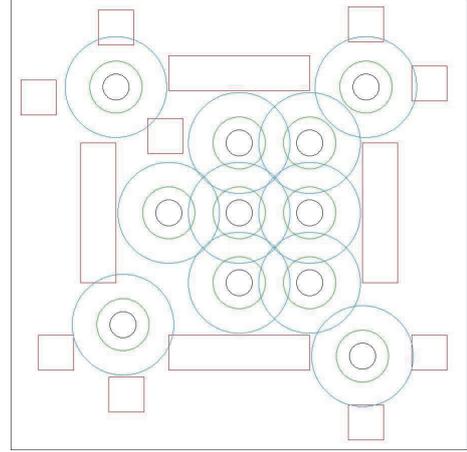
$$AdjList(v) = \{u : u \in V \wedge Distance(u, v) < min_{gap} \wedge u \neq v\} \quad (1)$$

Lastly, we construct a conflict graph that each node represents a component, and the edge representing the minimum gap constraint is violated by the adjacent PTHs. Equation 1 describes the conflict of a vertex  $v$ , where  $V$  represents the set of all vertices. Now we transform the problem into a graph problem which we are familiar with. Our objective is to eliminate all conflict edges in the graph by adjusting the copper landing pads. Note that although we are unable to adjust the landing pads of SMDs, all violated edges are still taking into account in the conflict graph. Figure 3(a) depicts a PCB layout example (Case 1). The rectangles drawn in *red* lines are SMDs. The *green* circles are the copper landing pads of PTHs, and the *black* circles inside the pads are the drills. The *blue* circles are the shapes after expanding the PTH copper landing pads by minimum gap distance. From Figure 3(a), minimum gap violation occurs when a *blue* circle intersects with another *green* circle or *red* lines. Figure 3(b) sketches the conflict graph of Figure 3(a), where PTHs are represented in *white* nodes and SMDs are represented in *gray* nodes. The weight of a node represents the number of incident edges to that node.

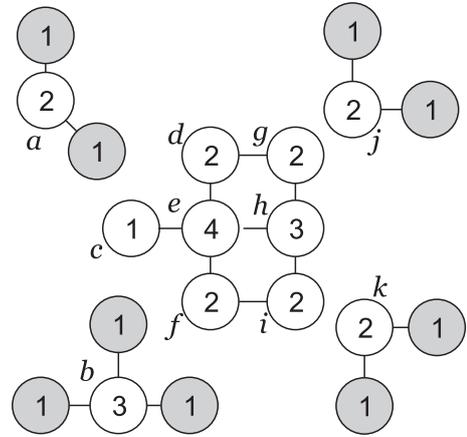
### B. Legalization Stage

After pre-processing, we legalize the design by eliminating the edges in conflict graphs. We develop three operations, the offset operation, the pad cutting operation, and the pad shrinking operation, to legalize the violations.

The offset operation legalizes two conflict landing pads, which violates the minimum gap constraint, by shifting landing pads. This is the most preferable operation since both the area and contour of the landing pad are kept unchanged. In each conflict edge, we select movable landing pads (i.e., non-SMD landing pads) as candidates and evaluate the offset vectors of all landing pads by force-directed. If both components are PTHs, we add half of the distance of violation in the opposite direction to the force. If one side is an SMD, then the force added to the PTH should be the whole distance of violation. If the design-rule violations could be eliminated by shifting conflict landing pads, we apply the corresponding offset



(a) A PCB layout example (Case 1) with minimum gap constraint violation.



(b) The conflict graph of Case 1

Fig. 3. The conflict graph of Case 1.

operations to those pads and move them to designate locations. Otherwise, we keep the conflict landing pads unchanged.

$$D = \min(d), \quad d = \{Distance(v, AdjList(v))\} \quad (2)$$

The pad shrinking operation takes a PTH copper landing pad and a shrinking offset as inputs. The operation shrinks the copper landing pad by a specified offset inward [9]. Equation 2 derives the shrinking offset  $D$ . The shrinking offset  $D$  is evaluated by comparing the distance between the input node and all its adjacent nodes and getting the one that violates the constraint the most. The pad shrinking operation is capable of simultaneously removing multiple edge conflicts from adjacent landing pads at the expense of sacrificing overlay region as well as electronic component adhesion force.

$$\pi r_1^2 = r_2^2 \times \frac{\theta}{\pi} + Distance(P_0, P_3) \times x \quad (3)$$

$$P_3 = P_0 + \sqrt{r_2^2 - x^2} \times \frac{P_1 - P_0}{Distance(P_1, P_0)} \quad (4)$$

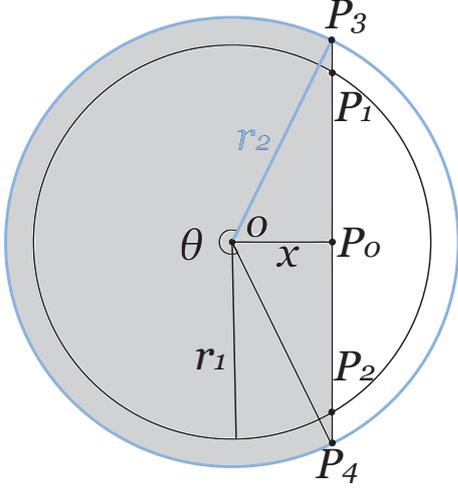


Fig. 4. Cut area compensation for landing pad.

$$\theta = |\arctan(P_3 - O) - \arctan(P_4 - O) - \pi| + \pi \quad (5)$$

The pad cutting operation takes a PTH copper landing pad as input. As illustrated in Figure 2(c), we cut the shape of a landing pad to eliminate the design-rule violation when there is a conflict edge incident to the landing pad. However, to maintain enough adhesion force to electronic component, we try to compensate the cut area of the landing pad by enlarging the pad. In order to avoid new design-rule violations, after enlarging the pad, we are required to verify the distance to neighboring landing pads. Figure 4 illustrates the cut area compensation for the landing pad. Assume that the original landing pad is centered at point  $O$  and the collinear points  $P_0$ ,  $P_1$ ,  $P_2$ ,  $P_3$ , and  $P_4$  form the cut line. We expect to enlarge the cut landing pad from radius  $r_1$  to  $r_2$  so that the cut area could be compensated. Let  $P_0$  be the closest point to  $O$  on the cut line.  $x$  is the distance between  $O$  and  $P_0$ . The enlarged radius  $r_2$  can be derived by Equations 3-5. Equation 3 indicates that the uncut area of the original landing pad with radius  $r_1$  is the same to the cut area of the enlarged landing pad with radius  $r_2$  cut by  $\overline{P_3P_4}$  (i.e., the region painted in gray). Equation 4 derives the locations of  $P_3$  and  $P_4$ . Equation 5 derives the angle of the sector.

After the pre-processing stage, the conflict graph may contain several disconnected conflict subgraphs, each consists of nearby copper landing pads. By iterating through these disconnected conflict subgraphs and applying the legalization operations to the graph nodes independently, the violations could be fully removed. In each conflict subgraph, we first try to legalize the design using only offset operation by finding the component in the center and then pushing the others outward. To identify the center landing pad, we select the node with the highest weight (i.e., with the most incident conflict edges) as the starting node in each connected component of the conflict graph. From the starting node, we use a depth-first search

TABLE I  
TESTCASE SCORES

	$S_1$ (30%)	$S_2$ (30%)	$S_3$ (40%)	Total
Case 1	25.64	26.96	38.51	91.10
Case 2	30.00	29.19	39.79	98.98
Case 3	30.00	24.08	32.57	86.65

to perform the offset operation with the current node and one of its unvisited adjacent nodes. The adjacent node will then be marked visited and selected as the next node to be processed. The algorithm will keep repeating the step until there is no unvisited node in the conflict graph. If there are still illegal PTHs after the offset operation, we perform both pad shrinking and pad cutting operations simultaneously and select the one with better result (i.e., higher score), which will be discussed in Section IV. For each of the remaining nodes in the conflict graph, we make a copy of the node, then perform the pad cutting operation on the copied node and the shrinking operation on the original node. If the result of the shrinking operation is better, assign the result back to the original node. Actually, we also tried to perform the pad cutting and shrinking operations at the offset operation phase, but the results were not very satisfactory.

#### IV. EXPERIMENT RESULTS

We use the testcases and score functions from ICCAD 2021 CAD Contest Problem E to evaluate our proposed optimization algorithm [10]. There are 6 problems in the contest, and Problem D, E, and F is for domestic students only. Our results won the first place in ICCAD 2021 CAD Contest Problem E. The score functions  $S_1$ - $S_3$  are summarized in Equations 7-9. The shape difference  $Shape_{diff}$  represents the outline shape difference between the original and the legalized copper landing pads in Equation 6. Figure 5 illustrates outline shape difference. In Figure 5(a), the number of lines changes from 0 to 4 while the number of arcs changes from 1 to 0. Therefore, the  $Shape_{diff}$  is  $1 + 4 = 5$ . In Figure 5(b), the number of lines changes from 0 to 2 while the number of arcs changes from 1 to 2. Thus, the  $Shape_{diff}$  is 3. The  $S_1$  formula evaluates the shape similarity of the adjusted copper landing pads as compared to the original ones. The  $Diff_{limit}$  is a predetermined constant. If we use a lot of pad cutting operations, the  $S_1$  score will be very low. The  $S_2$  formula calculates the overlay region of the copper landing pads before and after adjustment. All three operations could impact the  $S_2$  score. The  $S_3$  expects the total area of copper landing pads before and after legalization could be the same in order to maintain the adhesion force. Both pad cutting and pad shrinking operations affect the  $S_3$  score.

Table I lists individual scores of all testcases. We achieved the best result in the contest. Figure 6 depicts

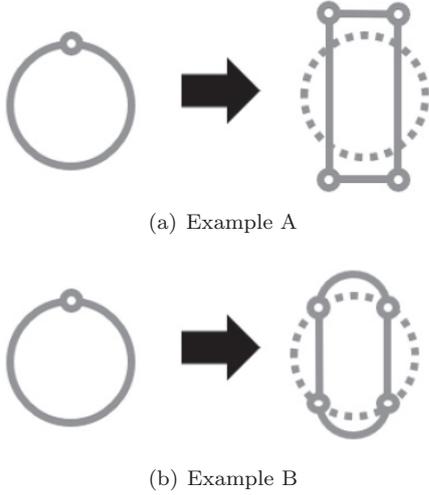


Fig. 5. Examples of the shape difference. (Images courtesy of Footprintku Inc.)

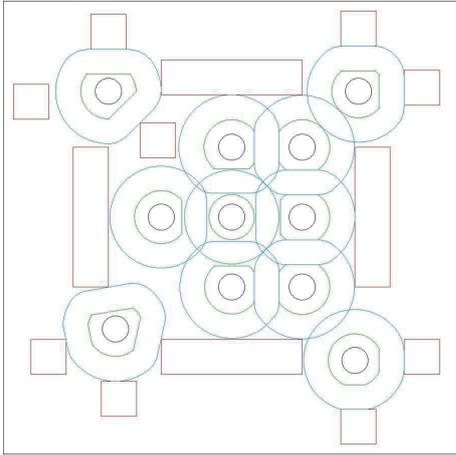


Fig. 6. Legalized layout of Case 1.

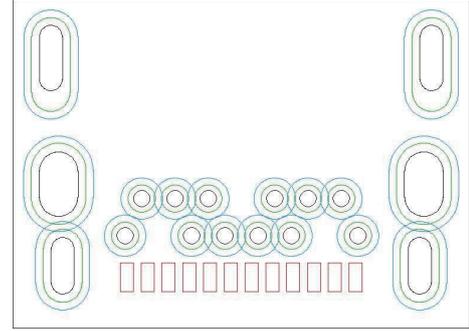
the legalized layout of Case 1. Figure 7 and Figure 8 demonstrate the PCB layouts of Case 2 and Case 3, respectively. Since our framework successfully avoids using pad cutting operation in Cases 2 and 3, there is no  $S_1$  score degradation in Table I.

$$Shape_{diff} = |Line_{orig} - Line_{adj}| + |Arc_{orig} - Arc_{adj}| \quad (6)$$

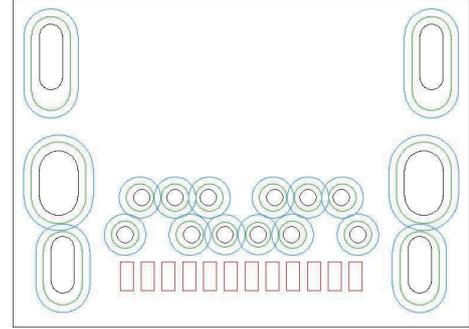
$$S_1 = 1 - \frac{Shape_{diff}}{Diff_{limit}} \quad (7)$$

$$S_2 = \frac{Area_{overlap}}{Area_{orig}} \quad (8)$$

$$S_3 = 1 - \frac{|Area_{orig} - Area_{adj}|}{Area_{orig}} \quad (9)$$



(a) The original layout of Case 2 with violations.



(b) The legalized layout of Case 2.

Fig. 7. PCB layouts of Case 2.

## V. CONCLUSION

We have proposed an optimization framework, which is the first systematic approach in this field, to adjust the PTH copper landing pads automatically using three operations, the offset operation, the pad cutting operation, and the pad shrinking operation. The experimental results demonstrate the effectiveness of our framework. Our proposed framework significantly reduces the manual task of PCB layout engineers for time and effort saving. In the future, we plan to validate the robustness and the limitation of our optimization framework by using more stringent industrial testcases.

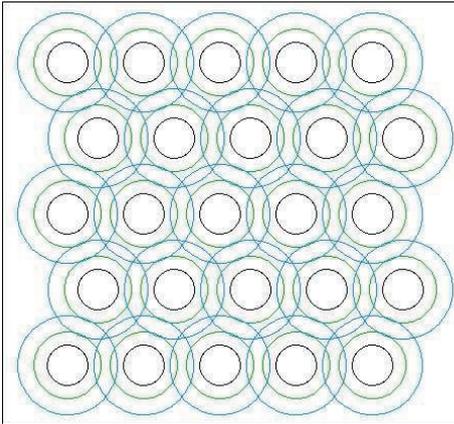
## Acknowledgement

The authors would like to thank Mr. Sean Kuo from Footprintku Inc. for providing this interesting research topic and valuable discussions.

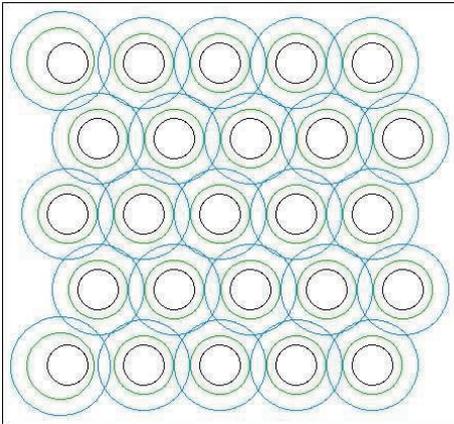
## REFERENCES

- [1] Shiraishi, H. and Ishii, M. and Kurita, S. and Nagamine, M. *ICAD/PCB: Integrated Computer Aided Design System for Printed Circuit Boards*, 19th DAC, pp. 727-732, 1982.
- [2] Masri Ayoba and Graham Kendall, *The optimisation of the single surface mount device placement*

[10] 2021 ICCAD Contest, Available: [http://iccad-contest.org/2021/tw/03\\_problems.html](http://iccad-contest.org/2021/tw/03_problems.html)



(a) The original layout of Case 3 with violations.



(b) The legalized layout of Case 3.

Fig. 8. PCB layouts of Case 3.

*machine in printed circuit board assembly: a survey*, IJSS, pp. 553-569, 2009.

- [3] Jonathan F. Bard, Raymond W. Clayton and Thomas A. Feo, *Machine Setup and Component Placement in Printed Circuit Board Assembly*, IJFMS 6, pp. 5-31, 1994.
- [4] *Printed circuit board*, Available: [https://en.wikipedia.org/wiki/Printed\\_circuit\\_board](https://en.wikipedia.org/wiki/Printed_circuit_board)
- [5] *Point in Polygon Strategies*, Available: <http://erich.realtimerendering.com/ptinpoly/>
- [6] *Jordan Curve Theorem*, Available: [https://en.wikipedia.org/wiki/Jordan\\_curve\\_theorem](https://en.wikipedia.org/wiki/Jordan_curve_theorem)
- [7] *Conflict Graph*, Available: <https://www.math.cmu.edu/~bkell/21110-2010s/conflict-graphs.html>
- [8] *Computational Geometry*, Available: <https://web.ntnu.edu.tw/~algo/Point.html>
- [9] *A Survey of Polygon Offsetting Strategies*, Available: <http://fcacciola.50webs.com/Offsetting%20Methods.htm>