Turf Optimization for Artificial Grass Installation: A Geometric and Cost-Efficient Approach

Jakob Emil Pálmason

Reykjavik University, Iceland

jakobp20@ru.com

Abstract

Installing artificial turf in irregularly shaped areas can result in significant material waste if not optimally planned. This paper addresses this challenge by developing a Python-based optimization system for turf roll layout. The system computes a material-efficient cutting and placement plan for a defined irregular area. Using geometric heuristics, we have a centroid-based rotation search to minimize waste. By rotating the target area and slicing it into parallel strips corresponding to the width of the turf roll, the algorithm finds the orientation that best fits the area with minimal off-cuts. The method uses Shapely and NumPy for geometric processing and waste calculation and employs Matplotlib to visualize the layout. The output is an installation map with detailed roll placements and seam locations, delivered via an interactive front-end map interface. This automated approach significantly improves manual planning efficiency and material waste by exploring many layout options quickly and unbiasedly. The results demonstrate that optimizing roll orientation can substantially reduce wasted turf while balancing practical considerations such as installation complications. This work relates to existing research in construction layout planning and irregular packing. The proposed solution shows promise in saving material and cost for turf installations and paves the way for more innovative construction layout tools.

Keywords

Artificial turf, layout optimization, irregular packing, material utilization, waste reduction, heuristic algorithm, geometric planning

1 Introduction

Artificial turf, also known as artificial grass, synthetic turf, or synthetic grass, has become a common surface choice for sports fields and landscaping, with thousands of installations worldwide each year. Contractors installed more than 6,000 fields in North America in 2001, and the region continues to add approximately 1,000 to 1,500 new installations annually [1]. Such widespread adoption leads to substantial material consumption, as each project uses factory-produced turf strips of fixed width and significant length, where installers must precisely cut to match specific field geometries. A well-known practical issue is material waste, which is prevalent when dealing with irregularly shaped areas.

When covering an irregular plot, the rectangle turf strips inevitably produce off-cuts. To manage this uncertainty, planners often budget additional materials, resulting in increased project costs and substantial environmental impact due to plastic waste generated by discarded off-cuts. This issue underscores the importance of optimizing turf layouts to reduce material waste and environmental burdens associated with disposal.

Planning layouts for turf installation is a manual or semi-manual task performed by installation teams or project planners. Workers typically rely on previous experience or CAD tools to visualize strip placements, then manually adjust and trim turf on site. Although conventional methods function for rectangular areas, for more complex shapes, this manual approach can be time-consuming for installers and planners alike and often fails to produce optimal results, leading to material shortages or excessive off-cuts. Although standard CAD programs help planners measure areas and align turf strips, these tools do not inherently optimize the layout for minimal waste; that task is up to the planner's intuition. Although there are comparable layout optimization problems in the construction and flooring industries, such as efficiently cutting floorboards or sheet materials, specialized solutions tailored specifically to optimize artificial turf layouts remain scarce.

The task of laying artificial turf aligns closely with the two-dimensional strip-covering variant of the cutting stock problem, where a polygonal area must be entirely covered using parallel strips of fixed width and effectively unlimited length while minimizing waste. Researchers categorize such problems as NP-hard spatial optimization problems; therefore, it is impractical to brute-force all possible cutting configurations [2]. The turf variant admits two simplifications: the strips are uniform and infinitely long, and the search space can be limited to a discrete set of rotation angles and evenly spaced starting offsets. This structure makes a heuristic optimization approach both practical and effective.

This study introduces an optimization algorithm to determine the turf strips' optimal orientation and offset position relative to the target area. The algorithm rotates the polygonal target shape around its centroid in increments of 0 to 179°. At each rotation angle around the polygon's centroid, the system places a set of parallel turf strips against the lower boundary, then iteratively shifts the layout through ten offset positions in equal increments. Sampling those different starting positions closes stray gaps and cuts down on waste, especially when the field outline is irregular; after evaluating every configuration, the algorithm picks the rotation-offset pair that yields the least waste. The developed solution leverages computational geometry techniques implemented via the Shapely library to perform accurate geometric computations where strips lay virtually across the target polygon, and computations determine how much of each strip lies within or outside the designated area. The algorithm identifies the angle that yields the highest material utilization. The results, including the arrangement of each turf roll and the total waste area, are visualized for the user, providing a clear installation plan followed on-site.

This paper thus bridges the gap between theoretical optimization techniques for layout problems and the practical challenges turf installers face. It brings proven methodologies from construction layout planning and irregular shape packing into a user-friendly application tailored to artificial turf installation projects. The following sections review related work on construction site planning and packing problems, describe the methodology and system architecture, analyze results from illustrative test cases, and conclude by identifying potential areas for future work.

2 Metatron and the Challenge

Metatron, founded in 1969, is Iceland's leading installer of artificial turf for sports fields and public recreational areas. Over decades, the company has expanded its scope from laying artificial turf for football fields and playgrounds to providing end-to-end infrastructure such as stadium lighting and irrigation systems. Metatron recently signed an exclusive partnership with SYNLawn, a global artificial turf manufacturer. Metatron and SYNLawn can offer top-quality artificial turf for public spaces, sports fields, home gardens, and commercial landscapes. The collaboration marks a big step forward, establishing Metatron as the leading provider of premium turf solutions for private and public clients; however, it also carries new demands.

Despite its expertise, Metatron's recent development brings new challenges in material estimation for irregular areas. For simple rectangular fields, installation planning is straightforward, but new landscaping projects involve irregularly shaped plots, and planning the strip layout becomes more complex. Companies worldwide that resemble Metatron quote such installations by manually sketching layouts. Estimators must create a drawing and sketch placement lines for each turf strip to visualize how the area will be covered. This hands-on approach is labor-intensive and inherently imprecise. The method is prone to human error and can miss opportunities to reduce waste, especially for unconventional shapes with curves or oblique angles, resulting in suboptimal layout plans.

Inefficiencies in manual layout planning have significant operational, economic, and environmental implications for Metatron. Operationally skilled staff spend considerable time iterating on layout sketches, and any imprecision can lead to inaccurate bids or lost profits. An estimator must often add generous waste margins to avoid running short of turf, so bids overestimate material needs. Economically, this results in higher costs and unnecessary wasted material. The company frequently orders more artificial turf than is necessary to ensure full coverage, only to trim excess pieces during installation. Those off-cuts directly represent money paid for material that never gets used. Over numerous projects, the cost of such material overage can erode profit margins and make Metatron's offers less competitive. Environmentally, excess turf off-cuts made from plastic and nylon pose a waste disposal problem and extra cost, affecting the company's image as a sustainable enterprise. Larger pieces can be used for patching small areas, but most scraps are too small to reuse and, therefore, discarded as trash.

Metatron's quoting dilemma illustrates a strong need for an optimization-based approach to improve labor efficiency, reduce material waste, and produce more accurate bids by solving the turf layout problem through rigorous geometric optimization.

The academic literature on turf optimization layouts remains scarce, but studies on artificial turf usage and costs highlight why an efficient layout is valuable. McNitt [3] documented the rapid growth of artificial turf adoption in the early 2000s, driven by improved products and lower costs for turf systems. Even with these advances, installing an artificial field remains a substantial investment. A lifecycle cost analysis by Daviscourt et al. [4] found that over 8 years, the total cost of a field with artificial turf was higher than that of a natural grass field, largely due to the high material and installation expenses incurred upfront. This economic reality motivates the search for ways to economize the installation process.

3 Background

Spatial layout optimization tasks (e.g., cutting and packing) are fundamental in many industrial applications where efficient space utilization is critical. In manufacturing and material processing, these problems arise when parts or shapes are arranged on raw material stock to maximize utilization, for example, cutting patterns in sheet metal, textiles, or glass. Similar challenges occur in construction planning, for instance, when determining the layout of structural components or floor tiles and resource allocation, where objects must be packed optimally for storage or transport.

Spatial layout problems are notoriously difficult to solve. They belong to a class of optimization challenges that grow exponentially with the size of the problem. Even the most simplified versions are NP-hard. For example, the classic bin packing problem, items in the fewest bins, and the two-dimensional strip packing variant, where items must fit into a fixed width strip of minimal height. The difficulty persists even for geometrically simple cases. For example, when covering an orthogonal polygon with the minimum number of rectangles, as shown by Culberson and Reckhow [5]. These tasks are often NP-hard in the strong sense, meaning no pseudo-polynomial time algorithms exist for exact solutions in general. Optimal solutions to packing problems require searching an exponentially large space, making exact solutions computationally infeasible for large problems. Consequently, practitioners must resort to approximate and heuristic solution strategies. It is well-recognized that heuristic and metaheuristic algorithms are best for these layout problems in real-world scenarios. Solution strategies span greedy constructive heuristics, linear programming relaxations, evolutionary computation, and other metaheuristics. These approaches efficiently navigate the vast search space and accept near-optimal results in exchange for practical runtimes.

A core objective in layout optimization is minimizing material waste. In cutting and packing contexts, waste corresponds to the unused material, trim scraps or empty space, after the pieces are cut or placed. In industries such as paper, fabric, wood, plastic, metal, and other materials, efficient layouts can significantly decrease the percentage of material discarded. Even modest improvements in utilization can lead to substantial economic savings over large production volumes and decrease manufacturing processes' environmental footprint. In summary, the connection between combinatorial layout optimization theory and real-world tasks such as turf installation is strong: improvements in algorithmic solutions for abstract packing problems directly enable more economical, efficient, and sustainable outcomes. This background motivates the study, which examines strip-packing and layout planning methodologies, enhancing material utilization in turf installation projects.

3.1 Two-Dimensional Irregular Packing and Cutting Problems

Two-dimensional cutting, packing, and covering problems involve arranging geometric pieces within a region without overlap, aiming to maximize material utilization and minimize waste. These problems are inherently combinatorial: even the simplest variants (e.g., packing rectangles into a bin or covering an area with given shapes) can be NP-hard, meaning no efficient algorithm is known for finding the perfect solution [6, 7].

A special class of packing problems is the strip-cutting or strip-packing problem, sometimes described as a 2D cutting-stock problem. In a strip-packing scenario, one gets a roll of material of fixed width, unlimited length, and a set of items to cut out. The goal is to place all the items on the strip without overlapping so that the total length of the material used is minimized [7]. Thus minimizing the wasted area of raw material, as using less length of the fixed width roll reduces unwanted trim [8]. A fundamental performance benchmark in such problems is the area-based length bound. In an ideal waste-free layout, the length of the material needed cannot be less than the total area of items divided by the strip width. In practice, irregular item shapes and additional constraints usually mean more length, making waste unavoidable.

When the items cut from the strip collectively form a single target shape, the problem can be viewed as a covering problem where the target polygon is enclosed by pieces or strips cut from the roll. A tangible example is the carpet cutting problem, where pieces of carpet are cut from a fixed-width roll to cover a floor area. The objective is to minimize the total carpet used, directly leading to minimizing waste material [9]. This problem is modeled as a 2D open-dimension packing problem, with fixed container width and the length dimension to cut [7]. If the target shape is simple, for example, an orthogonal polygon, and the covering pieces are restricted, for example, fixed-width rectangles, then determining the minimum waste cover is NP-hard [5]. In practice, heuristic algorithms, e.g., greedy stripe placement or metaheuristic search, generate near-optimal strip cover layouts for irregular shapes.

3.2 Construction Layout Planning and Facility Placement

The challenge of optimally arranging elements on a site is not unique to turf installation—the Construction Site Layout Planning (CSLP) problem shares similar combinatorial characteristics. In CSLP, the task is to allocate multiple facilities, such as material storage, workshops, equipment, etc., within the boundaries of a construction site, respecting geometric and safety constraints while optimizing objectives like material handling cost, safety, or travel time. Essentially, this refers to a spatial packing/allocation problem with additional project-specific constraints [10]. CSLP can be static, planned once for the entire project, or dynamic, replanned at different phases as space requirements change [10]. Modern approaches allow the site boundary to be an irregular polygon and represent facilities with more realistic shapes rather than simplifying the area to a grid or rectangle [11]. This introduction of free-form geometry in CSLP models, as demonstrated by Abotaleb et al., increases the possibilities of the layout but also the complexity of the search space [11].

CSLP is a complicated optimization challenge that current research tackles with genetic algorithms, particle swarm optimization, simulated annealing, and other heuristic or metaheuristic techniques rather than exact algorithms [7, 11]. Comprehensive reviews of CSLP methods conclude that due to the NP-hard nature of the problem and the many practical constraints, tailored heuristic approaches and even interactive decision support systems are necessary to obtain good solutions in reasonable time [10].

3.3 Turf Strip Optimization as a Simplified Packing/Cutting Problem

In this project, we must cover a simple polygonal area by laying parallel turf strips of fixed width and unlimited length, minimizing the excess strip area outside the polygon. We seek an arrangement of identical fixed-width and unlimited-length rectangles whose union covers the target region. This formulation is analogous to the fixed-width carpet cutting problem, for example, Shutt et al. model covering a floor area by cutting shapes from a fixed-width carpet roll [9]. In our case, the shapes are entire strips rather than arbitrary pieces. The primary decision variables are a global rotation angle for the strip orientation and a one-dimensional offset of the strip pattern relative to the polygon. With the angle and offset in place, the strips cover the polygon, and waste is the total strip area minus the polygon area used.

We note the key simplifications relative to general layout/packing problems:

- **Single facility:** Only one polygonal region is covered (no multiple disconnected areas or "facility" choices).
- **No obstacles/holes:** The polygon has no internal holes or forbidden regions; we need not decompose the domain.
- **Regular item shapes:** All covering items are uniform rectangles (strips) of the same width. There are no irregular or varying shapes to place.
- **Limited decision variables:** We optimize only the common strip orientation (rotation angle) and a single lateral offset. There is no combinatorial placement sequence or subset selection beyond choosing these continuous parameters.

These restrictions eliminate many complexities of typical layout optimization (such as selecting among multiple facilities, fitting irregular parts, or handling guillotine cutting patterns). In particular, we do not consider arbitrary polygon sub-parts or variable strip widths. Despite these simplifications, the problem remains computationally unfeasable. Even simple rectangular layout problems are NP-hard; for instance, arranging facilities on a plant floor, the Facility Layout Problem, is known to be NP-hard [12], and covering a polygon by rectangles is also NP-hard in general. Exact methods, e.g., constraint programming, can solve only small instances; for example, Schutt et al. report proving optimal solutions for 106 of 150 real-world instances of a related carpet-cutting problem, using specialized exact algorithm [9], and it still relied on heuristics for the rest. Thus, we adopt a heuristic search strategy; we discretely sample candidate rotation angles, and for each angle, we slide the strip pattern throug possible offsets to evaluate waste. This rotation and offset search is aligned with standard approaches in the cutting/packing literature, where one often fixes an orientation or a small set of orientations and then solves a one-dimensional covering problem [9, 12]. Such heuristic sacrifice guaranteed optimality but is necessary given the NP-hard nature of the problem, and it has been shown to yield high-quality, low-waste layouts in practice.

4 Methodology and System Design

This section presents the approach developed to optimize the placement and orientations of artificial turf rolls within an irregular area. It describes the geometric processing steps, the rotation-based alignment strategy, and the criteria used to evaluate the optimal solution. The method is implemented in Python and incorporates spatial analysis libraries for polygon operations and visualizations.

4.1 Problem Formulation

The turf strip layout problem is formally a constrained geometric optimization task. Given a simple polygon $P \subset \mathbb{R}^2$ representing the target area and a fixed roll width w, the objective is to determine an optimal arrangement of rectangular strips that completely covers P while minimizing excess material. The decision variables consist of a rotation angle $\theta \in [0°, 360°)$ applied to the polygon and the placement configuration of horizontal strips.

The objective function seeks to minimize waste, defined as:

$$\min f(\theta) = A_{\text{placed}}(\theta) - A_{\text{used}}(\theta) \tag{1}$$

Where $A_{\text{placed}}(\theta)$ represents the total area of all turf strips required for coverage at rotation angle θ, and $A_{\text{used}}(\theta)$ denotes the actual area of turf intersecting with the target polygon. The placed area encompasses all rectangular strip areas, while the used area equals the sum of intersections between strips and the polygon interior.

The problem operates under several constraints. First, the strip width remains constant at w meters, matching the manufactured roll width. Second, all strips must be oriented parallel to each other, reflecting the practical constraint of unrolling turf from continuous rolls. Third, the union of all strips must completely contain the target polygon: $P \subseteq \bigcup_i S_i$, where S_i represents the i-th strip. Finally, strips are positioned horizontally after rotation, with edges aligned perpendicular to the unrolling direction.

4.2 Geometric Processing Framework

The geometric processing pipeline transforms input coordinates into a normalized representation suitable for optimization. The system accepts polygon vertices as Cartesian coordinates in meters or GPS coordinates in latitude-longitude format. A local planar projection uses the polygon centroid as the reference point for GPS inputs. The projection employs the standard transformation:

$$(x, y) = ((\lambda - \lambda_0) \cdot \cos(\phi_0) \cdot R,\ (\phi - \phi_0) \cdot R) \tag{2}$$

where ϕ and λ represent latitude and longitude in radians, the subscript 0 denotes the centroid coordinates, and $R = 6{,}371{,}000$ meters is the Earth's radius

The polygon is normalized by moving its first vertex to the origin and rotating it so the edge to the second vertex lies along the positive x-axis. This consistent frame makes the optimization repeatable and reliable for any shape. The core geometric operation involves slicing the rotated polygon into horizontal strips of height w. For a polygon rotated by angle θ around its centroid, the bounding box determines the vertical extent $[y_{\min}, y_{\max}]$. Strips are generated sequentially from bottom to top, with the i-th strip defined as:

$$S_i = \{(x, y) : x_{\min} - \varepsilon \leq x \leq x_{\max} + \varepsilon, \\ y_{\min} + i \cdot w \leq y \leq y_{\min} + (i+1) \cdot w\} \tag{3}$$

where ε represents a small extension beyond the polygon bounds to ensure complete coverage. The intersection between each strip and the rotated polygon is found using robust geometric predicates implemented in the Shapely library [11]. The library handles numerical precision issues inherent in computational geometry operations, particularly when dealing with near-degenerate cases or vertices lying exactly on strip boundaries. For each strip S_i, the intersection $S_i \cap P$ yields either a simple polygon, a multipolygon for non-convex shapes, or an empty set for strips beyond the polygon.

4.3 Optimization Algorithm

Optimization employs an exhaustive angular search to identify the rotation angle yielding minimum waste. The algorithm evaluates discrete rotation angles from 0° to 359° in one-degree increments, balancing solution quality and efficiency. For each candidate angle θ, the polygon undergoes rotation around its centroid, preserving shape properties while exploring different strip orientations. The centroid is the rotation pivot to maintain spatial locality and prevent excessive translation during the search process.Given polygon vertices $V = \{v_1, v_2, \ldots, v_n\}$, the centroid $\mathbf{c}$ is computed as:

$$\mathbf{c} = \frac{1}{A} \sum_{i=1}^{n-1} \frac{(x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i)}{2} \tag{4}$$

where A represents the polygon area calculated using the shoelace formula. The algorithm computes placed and used areas for each rotation angle through systematic strip generation and intersection operations. The placed area accumulates as: The placed area is computed as:

$$A_{\text{placed}} = \sum_i |S_i| = \sum_i w \cdot (x_{i,\max} - x_{i,\min}) \tag{5}$$

where $x_{i,\min}$ and $x_{i,\max}$ represent the horizontal bounds of the intersection between strip i and the rotated polygon.

The used area calculation sums the actual coverage within the polygon boundary:

$$A_{\text{used}} = \sum_i |S_i \cap P| \tag{6}$$

The waste metric for each configuration equals $A_{\text{placed}} - A_{\text{used}}$, quantifying the excess turf material extending beyond the target area. This formulation aligns with established construction site layout planning, where material utilization efficiency is a primary optimization criterion [10].

The algorithm maintains the configuration, yielding minimum waste across all evaluated angles. By exploring the complete rotation space at discrete intervals, the approach guarantees finding the best solution within the discretization resolution, providing a practical balance between optimality and computational tractability for real-world turf installation planning.

Polygon areas are normalized prior to optimization to enhance computational efficiency and solution consistency. Initially, vertices $V = \{v_1, v_2, \ldots, v_n\}$ undergo translation such that the first vertex aligns with the coordinate origin. Subsequently, the polygon is rotated to position its longest edge parallel to the horizontal axis, normalizing the orientation. The required rotation angle θ is derived from the endpoints of the longest edge and calculated by:

$$\theta = -\tan^{-1}\left(\frac{y_{\max} - y_{\min}}{x_{\max} - x_{\min}}\right) \tag{7}$$

Rotations occur around the polygon's centroid $\mathbf{c}$, ensuring spatial locality and consistent comparative conditions across evaluations. The centroid is determined from vertices V by:

$$\mathbf{c} = \frac{1}{A} \sum_{i=1}^{n-1} \frac{(x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i)}{2} \tag{8}$$

Consequently, due to this geometric normalization, optimal solutions typically align closely with fundamental angles such as 0° or 90°, exploiting intrinsic polygon symmetries to reduce material waste.

4.4 Multi-Objective Optimization and Pareto Frontier

The turf layout problem exhibits inherent trade-offs between minimizing waste and reducing installation complexity, measured by the number of strips required. Following principles from multi-objective optimization [12], the algorithm identifies Pareto-optimal solutions where no improvement in one objective is possible without degrading another.

The Pareto frontier computation proceeds in two phases. First, the algorithm records the configuration yielding minimum waste for each unique strip count observed during the exhaustive search. This produces a set of candidate solutions $\{(n_1, w_1), (n_2, w_2), \ldots, (n_k, w_k)\}$, where n_i represents strip count and w_i represents waste. Second, the algorithm filters this set to remove dominated solutions, where solution i dominates solution j if $n_i \leq n_j$ and $w_i \leq w_j$ with at least one strict inequality.

4.5 Implementation and Computational Considerations

The implementation leverages the Shapely library for geometric operations, particularly polygon intersection and rotation transformations. These operations handle numerical precision issues inherent in computational geometry, especially with GPS coordinates converted to planar projections. The algorithm employs appropriate tolerance values and validates polygon validity after transformations.

The computational complexity scales as $O(\alpha\beta nm)$, where α represents the number of rotation angles tested, β the number of offsets per angle, n the number of polygon vertices, and m the average number of strips per configuration. While this represents substantial computation for large polygons, the discrete nature of the problem and bounded search space make exhaustive evaluation feasible for typical residential installations.

The algorithm's design aligns with established approaches in construction layout optimization [10], adapting general principles to the specific constraints of turf installation. Unlike generic packing problems that allow arbitrary piece placement [2], the turf optimization problem restricts solutions to parallel strips of uniform width, significantly reducing the search space while maintaining practical relevance.

4.6 Output Generation and Visualization

The optimization runs and produces both detailed diagrams and a cost report. It shows exactly where to place each strip, where cuts are needed, and the key measurements. It also calculates how much turf and glue, and any other materials, are required. By combining geometric layout with real installation needs, this tool delivers practical, field-ready instructions rather than just theoretical results.

4.6.1 Data

The system uses structured data stored in a Google Sheets database. This includes:

- It stores prices for various turf types and consumables, such as glue and sand, which the program uses for cost calculations.
- Project data. Names, dimensions, and location coordinates are stored in the same database and linked to each layout.

Using Google Sheets ensures that pricing data is always up-to-date, enabling accurate and flexible quoting. An example of the pricing data used is shown in Figure 1.

Gervigras nafn	Breidd (m)	kr/fm án VSK
SYN Berlin	4	6694
SYNNatural 50L	2	5516
SYNNatural 35L	4	4177
Precision Putt	3.66	11456
SYNFringe	2	12986
Tee Strike+	2	28554
SYNPlay 15mm	4	5680

Annað nafn	Eining	kr/ein án VSK
Sandur, þveginn og þurrkaður	kg	110
Lím	kg	2203
Límborði	m	122

Figure 1. Price list used in calculations. Prices are pulled from a live Google Sheets file and may change over time.

4.7 Error Analysis and Precision Considerations

Spatial precision in turf layout optimization is somewhat limited by several factors. First, user-drawn boundaries on a web map are not exact. Users click or draw freehand, so manual digitization can introduce offsets up to half a meter. Second, map APIs provide coordinates with finite precision. For example, Google Maps API uses lat/lon to size decimal places ≈0.1m and will round finer digits. These coordinate quantization effects add minor positional errors. Third, the geometry computations themselves use floating-point arithmetic; modern double precision yields sub-millimeter rounding error at residential scales, so numerical precision is effectively sufficient.

A final source is map projection distortion. In practice, the layout is computed in a local planar coordinate system. We project latitude/longitude onto a local tangent plane (East-North-Up) centered on the garden. This frame has axes aligned with true north and east [13], so it matches intuitive directions. At the plane's origin, there is no distortion, and over tens of meters, any distortion is negligible. Geospatial references note that map projections cause "minute" distortion on large-scale, small-area maps [14]. In practice, the combined effect of these errors is on the order of centimeters, which is a small relative total area.

5 Results and Analysis

This section presents the empirical evaluation of the turf strip layout algorithm across a series of representative plot geometries. Four categories of plots—rectangular, L-shaped, U-shaped, and irregular—were used to assess performance concerning material waste, layout completeness, and runtime efficiency. Each scenario reflects increasing geometric complexity and realism, emulating conditions encountered in actual construction sites. Results are supported by quantitative metrics and visualized in Figures 2 through 12. All data referenced in the subsections below from the layouts generated using the implemented heuristic described in Section 3.

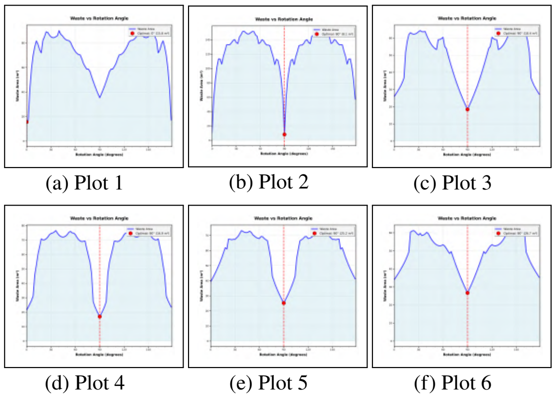


Figure 3. Waste analysis results for six rectangle-shaped plot configurations

5.1 Rectangle-shaped Plot

Rectangular plots are the baseline scenario to validate the algorithm under ideal geometric conditions. These simple convex shapes allow for intuitive layout strategies, making them a natural benchmark for evaluating solution quality and computational speed.

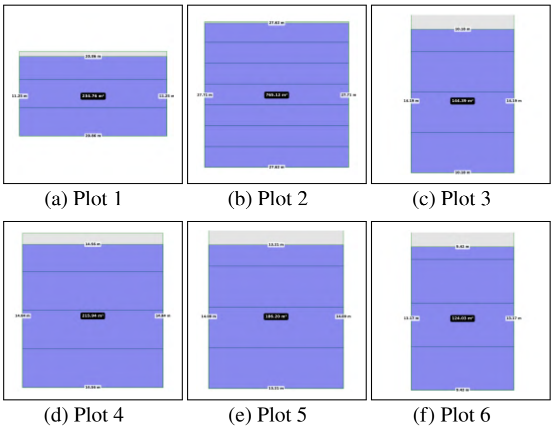


Figure 2. Six rectangle-shaped plot configurations used for layout testing

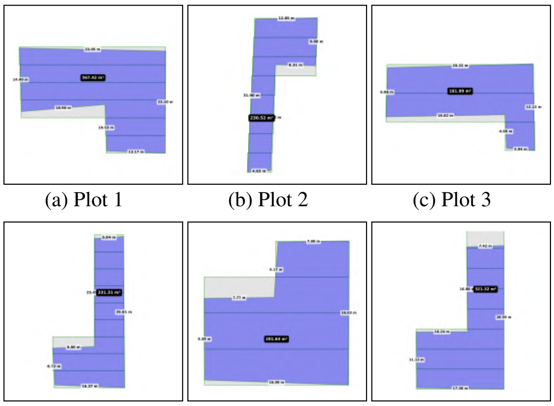
The algorithm achieved high efficiency on simple rectangular plots. As shown in Figure 2, six rectangular test plots of varying areas were each optimally covered by parallel turf strips aligned with the plot's edges. The resulting waste percentages (Figure 3) ranged from as low as 1.06% up to 21.5%, with larger, well-proportioned rectangles yielding minimal waste (e.g., Plot **b** with only 8.11 m^2 waste, 1.06%, Table 1) and smaller or irregularly dimensioned rectangles incurring higher relative waste (e.g., Plot **f** with 26.69 m^2 waste, 21.53%) due to leftover trim along the boundaries. Overall, the rectangle scenarios averaged about 10% of material waste. In all cases, the optimal strip orientation was either 0° or 90° (Table 1), indicating that aligning strips with the rectangular geometry maximizes coverage. The layout performance was consistent—every rectangular plot was covered with straight strips with no gaps—and the algorithm's runtime was negligible for these simple convex shapes (on the order of only a few seconds per layout). This validates that the method finds near-optimal layouts for orthogonal plots quickly, matching the intuitive strategy of covering a rectangle with parallel strips.

Table 1. Rectangle Scenario Summary

Plot	Area (m^2)	Vertices	Angle (°)	Waste (m^2)	Waste (%)	Pareto	Strips/100m^2
a	234,76	4	90	15,57	6,65	1	1,28
b	765,12	4	90	8,11	1,06	1	0,91
c	144,39	4	90	18,43	12,8	2	2,77
d	215,94	4	90	16,94	7,88	1	1,85
e	186,2	4	0	25,2	13,55	1	2,15
f	124,03	4	90	26,69	21,53	2	3,22

5.2 L-shaped Plot

L-shaped plots introduce geometric concavity, representing moderately complex site conditions. These scenarios test the algorithm's ability to adapt layout orientation and preserve low waste despite corner intrusions and non-trivial edge interactions.



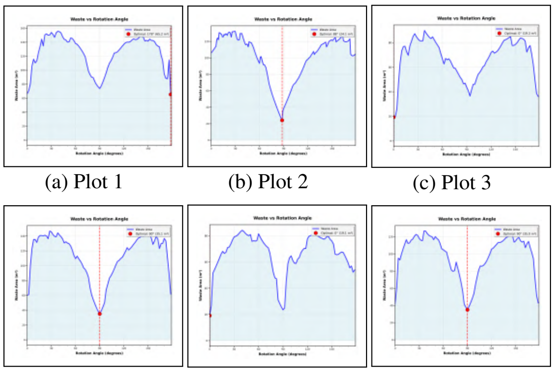
(a) Plot 1 (b) Plot 2 (c) Plot 3

(d) Plot 4 (e) Plot 5 (f) Plot 6

Figure 4. Six L-shaped plot configurations used for layout testing

Table 2. L-Shaped Scenario Summary

Plot	Area (m^2)	Vertices	Angle (°)	Waste (m^2)	Waste (%)	Pareto	Strips/100m^2
a	567,42	6	92	64,9	11,45	1	1,06
b	230,52	6	88	26,07	11,34	2	3,47
c	161,99	6	1	19,64	12,2	1	1,85
d	331,31	6	90	35,49	10,72	2	2,72
e	201,64	6	1	23,58	11,73	1	1,98
f	321,32	6	2	35,08	10,93	2	2,49



(a) Plot 1 (b) Plot 2 (c) Plot 3

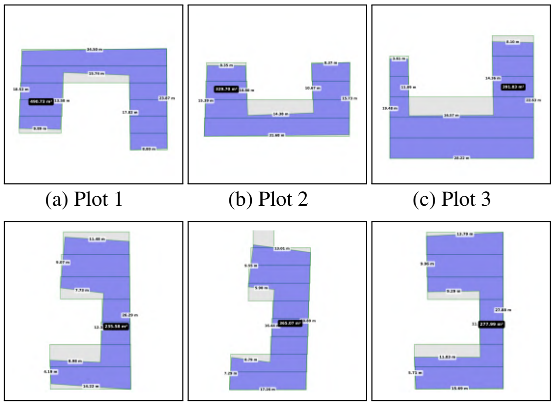
(d) Plot 4 (e) Plot 5 (f) Plot 6

Figure 5. Waste analysis results for six L-shaped plot configurations

For L-shaped (rectangular concave) plots, the algorithm demonstrated robust performance with relatively uniform waste levels. Figure 5 illustrates six L-shaped test configurations, each composed of two perpendicular rectangular sections. The optimization consistently found strip layouts that conformed well to the L geometry, filling both arms of the shape with parallel strips. Waste percentages for all L-shaped cases clustered around 11% (Table 2), with a narrow range of approximately 10.7–12.2% across plots. The concave corner inherent to an L-shape imposes a modest but steady waste liability—primarily small triangular off-cuts where strips in one wing overshoot or cannot fill the corner. The algorithm identified an optimal orientation for each plot (e.g., slight rotations of 1–2° for some cases, Table 2) to best align strips along both segments of the L, thereby minimizing these off-cuts. All six L-shaped layouts were covered entirely by the strip arrangements (Figure 4) with no uncovered areas, and the waste remained low and consistent regardless of plot size. Computationally, solving the L-shaped layouts remained efficient: the average runtime per L-shaped instance was only a few seconds, comparable to the rectangular cases. This suggests that the method scales well to slightly more complex (concave) geometries, maintaining low waste and quick solution times for L-shaped sites.

5.3 U-Shaped Plot

The U-shaped category also includes concavity, incorporating two inward-facing arms that create tighter constraints on strip alignment. These cases illustrate how increased shape complexity affects layout and material efficiency.



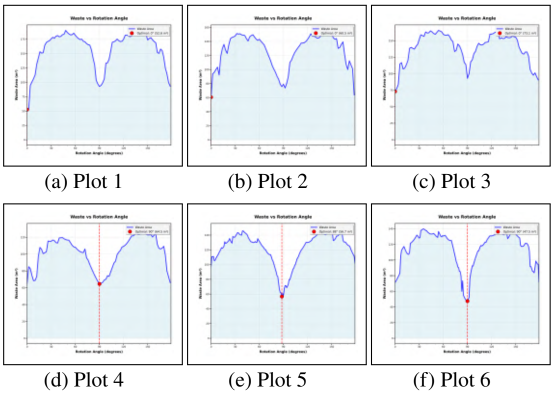
(a) Plot 1 (b) Plot 2 (c) Plot 3

(d) Plot 4 (e) Plot 5 (f) Plot 6

Figure 6. Six U-shaped plot configurations used for layout testing

Table 3. U-Shaped Scenario Summary

Plot	Area (m^2)	Vertices	Angle (°)	Waste (m^2)	Waste (%)	Pareto	Strips/100m^2
a	490,73	8	87	52,89	10,79	1	1,83
b	329,7	8	2	60,17	18,29	1	1,82
c	391,83	8	0	73,12	18,7	1	2,04
d	235,58	8	85	64,75	27,55	2	2,97
e	365,07	8	88	56,67	15,53	3	2,47
f	277,99	8	1	47,56	17,17	2	2,52



(a) Plot 1 (b) Plot 2 (c) Plot 3

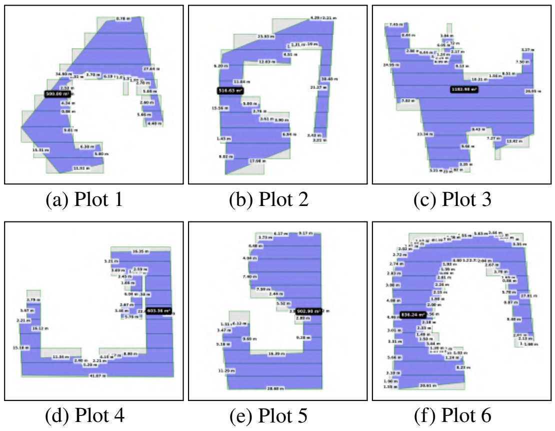
(d) Plot 4 (e) Plot 5 (f) Plot 6

Figure 7. Waste analysis results for six U-shaped plot configurations

The U-shaped plots (which include two concave corners) posed a more complex scenario, yet the algorithm effectively generated feasible low-waste layouts. Six U-shaped configurations (Figure 7) exhibited greater variation in waste outcomes than the L-shapes. As summarized in Table 3, waste ranged from about 10.8% in the largest U plot to 27.5% in the smallest, more acute U-shaped plot. In general, larger U-shaped areas achieved waste levels on par with the L-shapes (around 10–15%), while smaller or more irregular U configurations saw elevated waste. The highest waste occurred in Plot d (27.55% waste), which had the smallest area and pronounced concave recesses, leaving significant unused strips (Figure 6). This trend underscores that more turf off-cuts are inevitable as concave complexity increases in a confined area. Nonetheless, the algorithm minimized waste wherever possible by adjusting strip orientation and starting offset for each U-shape; optimal orientations varied (e.g., 85°, 87°, etc., in Table 3) to align strips with the U's axis and its wings. All U-shaped cases were covered entirely by strips, and the number of strips per area was reasonable (approximately 1.8–3.0 strips per 100 m^2, Table 3), indicating efficient use of material despite the shape complexity. The solution runtime remained within practical limits for these cases as well—on the order of tens of seconds or less for each U-shaped layout—showing that the added geometric complexity had a modest impact on computational performance.

5.4 Irregular Plot

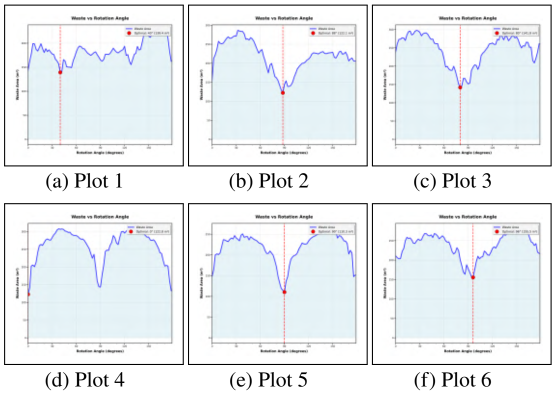
Irregular plots represent the most realistic and challenging geometries, including arbitrary polygonal boundaries and multiple concave regions. These scenarios demonstrate the algorithm's full potential and adaptability in handling free-form construction sites with unpredictable outlines.



(a) Plot 1 (b) Plot 2 (c) Plot 3

(d) Plot 4 (e) Plot 5 (f) Plot 6

Figure 8. Six irregular plot configurations used for layout testing



(a) Plot 1 (b) Plot 2 (c) Plot 3

(d) Plot 4 (e) Plot 5 (f) Plot 6

Figure 9. Waste analysis results for six irregular plot configurations

Table 4. Unorthodox Scenario Summary

Plot	Area (m^2)	Vertices	Angle (°)	Waste (m^2)	Waste (%)	Pareto	Strips/100m^2
a	500	23	102	150,58	30,12	3	2,8
b	516,63	23	168	131,79	25,54	2	3,1
c	1182,98	30	4	151,69	12,83	5	1,61
d	603,36	28	1	120,42	19,97	4	2,82
e	902,9	21	96	109,4	12,13	2	1,66
f	836,24	68	178	155,74	18,63	3	2,03

The algorithm's advantages became most evident in the irregular plot scenarios. Figure 9 illustrates six highly irregular plot shapes (with many sides and arbitrary angles) used to challenge the layout optimizer. Despite the complex geometries, the algorithm successfully generated turf strip layouts for all cases (Figure 8), covering each irregular region completely. Material waste outcomes, summarized in Table 4, varied more widely for these irregular plots compared to simpler shapes. Waste percentages ranged from a low of about 12.1% up to roughly 30.1%. Notably, the largest and relatively more convex irregular plot (Plot c, area $\approx$ 1183 m^2) achieved one of the lowest waste fractions (12.8%), comparable to the L- and U-shaped best cases.

In contrast, smaller plots with highly convoluted boundaries saw higher waste—e.g., Plot a (500 m^2) and Plot b (517 m^2) incurred about 30.1% and 25.5% waste, respectively, primarily due to numerous small wedges of turf trimmed off around their jagged perimeters. These results indicate that irregular polygons can be efficiently covered when their overall area is large enough to accommodate full strips, but very irregular edges will inevitably create some waste. The algorithm was able to adapt to each unique shape by exploring different strip orientations; the optimal layout orientations for these cases were often non-intuitive angles (e.g., 96°, 168°, etc., as listed in Table 4) that best fit the polygon's global geometry. In several irregular cases, the optimizer found multiple near-equivalent solutions (up to five Pareto-optimal layouts for a given plot, see Table 4), each offering a trade-off between slightly lower waste and fewer strips. Even for the most complex shape tested (Plot f, with 68 vertices), the algorithm produced a practical layout with 18.6% waste and did so in a reasonable time. On average, the irregular plot scenarios required an order of tens of seconds of computation each—a slightly longer runtime than simpler shapes, yet still efficient given the complexity.

5.5 Large-Scale Industrial Complex Case Study

The proposed optimization algorithm was evaluated on a challenging real-world scenario involving a 16,293.57 m^2 plot with highly irregular geometry featuring 43 vertices and a perimeter of 697.38 meters.

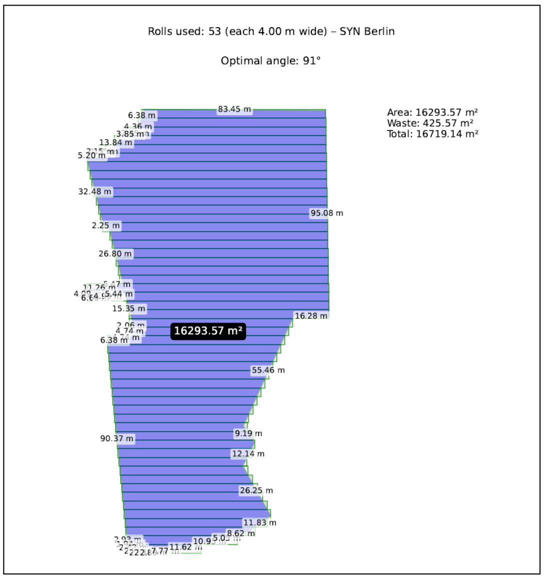
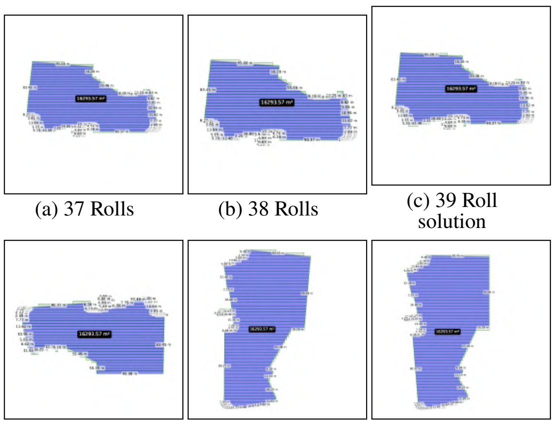


Figure 10. Optimized layout result for the 16,293.57 m^2 plot

Despite the geometric complexity, the algorithm achieved remarkable material efficiency with only 432.12 m^2 of waste, representing 2.65% of the total area. The optimal configuration required 53 turf strips oriented at 91°, effectively aligning perpendicular to the structure's primary axis. This orientation minimized waste by exploiting the natural alignment of the polygon's major features while accommodating irregular boundaries through strategic strip placement (Figure 10).



(a) 37 Rolls (b) 38 Rolls (c) 39 Roll solution

(d) 41 Roll solution (e) 51 Roll solution (f) 52 Roll solution

Figure 11. Pareto-optimal layout alternatives for the university plot excluding the most material-efficient configuration

Table 5. Cost Analysis

Description	Amount	Unit Price	Total Price
SYN Berlin	16,719.14 m^2	6,694 kr	111,917,921 kr
Sandur	163,302.00 kg	110 kr	17,963,220 kr
Glue	4,028.64 m	625 kr	2,517,898 kr
Glue-Tape	4,028.64 m	122 kr	491,494 kr
Total Cost			**132,890,532 kr**

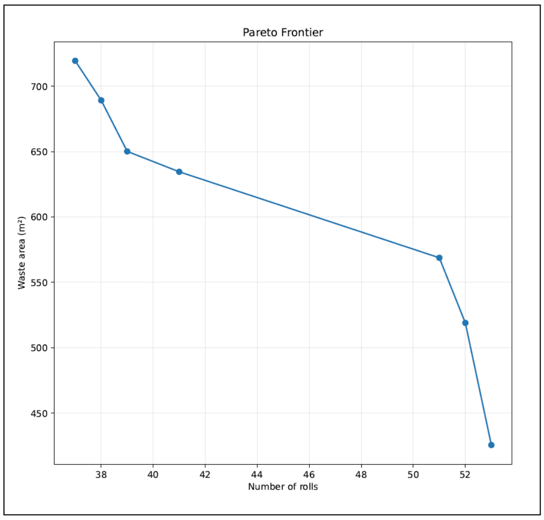


Figure 12. Pareto front showing trade-offs between waste and roll count for the university plot

Table 6. Pareto-optimal layout solutions for the university plot

Rolls	Angle (°)	Waste (m^2)	Glue (m)
37	175	719.49	4046.45
38	175	689.25	4040.03
39	176	650.20	4028.51
41	0	634.60	4026.22
51	85	568.82	4064.88
52	90	519.00	4053.38
53	91	425.57	4028.64

6 Discussion

Across all scenarios, clear trends and trade-offs can be seen in turf layout optimization. Increased geometric complexity correlates with higher waste percentages; simple rectangular sites can be covered with minimal waste, especially when their dimensions align well with standard strip widths (yielding as low as $\sim 1\%$ waste in our tests). Introducing concave corners (L- and U-shapes) leads to a baseline waste of roughly 10–15%, as some turf must be trimmed to accommodate the indented corners. Further increasing irregularity (multiple concave regions or angled edges) can drive waste higher, as seen in the most convoluted U-shaped and irregular plots (with peak waste around 25–30%). These findings are consistent with general principles in irregular packing problems—the more geometrically irregular the region, the harder it is to achieve perfect coverage, and the greater the fraction of unusable off-cuts tends to be higher [2]. In other words, highly non-convex polygons inherently leave voids when filled with rectangles, a phenomenon also noted in covering tasks (e.g., carpet installation for complex floor plans [9]). Despite this inherent difficulty, the algorithm's performance was strong in all cases, suggesting a significant advantage for automated optimization, especially for irregular geometries. Notably, for complex shapes, the waste remained within manageable levels (approximately 12–18% in several irregular plots), and in one large irregular case, the waste was only 12.8%, rivaling the efficiency achieved on much simpler L- or U-forms. Thus, the proposed method can exploit geometry opportunities (such as broad convex sections of an otherwise irregular boundary) to place full-length strips and minimize waste. Such performance on arbitrary shapes is a marked improvement over manual layout approaches, likely resulting in substantially higher waste for the same irregular sites. In practical terms, reducing waste from, say, 25–30% down to $\sim$ 15% on an odd-shaped field can translate to significant cost savings and material reduction—an important benefit given the high material cost of synthetic turf and its infill components [4]. Furthermore, from a sustainability and lifecycle perspective, minimizing waste supports better resource efficiency, aligning with industry trends emphasizing cost-effective and eco-conscious practices in synthetic turf installation [3][15].

The results also highlight the presence of trade-offs between competing objectives: minimizing material waste and the number of strips (which relates to installation labor and seam count). Our algorithm treated turf layout as a bi-objective optimization, and in many scenarios, the outcome was a set of Pareto-optimal solutions. This is especially evident in the irregular plots and some U-shaped cases, where multiple layouts offered balances between waste and strip count. For instance, an irregular plot might have one solution using a few large strips (reducing joins and labor) at the cost of slightly higher waste and an alternative solution with numerous narrower strips that fit the edges more tightly (yielding lower waste but more seams). All such solutions are valuable, as they allow a project manager to choose based on priorities—whether material savings or reduced installation effort is more critical in a given project. The prevalence of dual-objective solutions aligns with other multi-objective layout problems in construction and manufacturing. By presenting a Pareto frontier of solutions for complex shapes, the methodology gives decision-makers flexibility to accommodate practical considerations (like unusual site constraints or installer preferences) without straying far from optimality on either objective.

7 Future Work

While the proposed algorithm effectively optimizes turf layout for a single region, its usage can expanded by addressing practical site-specific constraints that have not yet been addressed. One important extension is to incorporate entrance-facing layout constraints. In many real installations, the turf orients toward a main entrance or focal point at a specified angle (for visual alignment or functional reasons). We plan to allow the user to define an entrance vector or anchor direction, e.g., aligning strips at 45° toward the entrance, and integrate this preference into the optimization. The algorithm would then prioritize strip orientations that adhere to the desired angle by restricting the search around that angle or by introducing a secondary objective that penalizes deviation. Crucially, this must be balanced against material efficiency—ensuring that honoring an orientation constraint does not excessively increase waste. By accommodating user-defined orientation targets, the system would address aesthetic and practical preferences often mandated in the field without straying far from optimal material usage.

Another key improvement is to support internal obstacles within the target area. Real-world turf plots frequently contain excluded regions such as trees, planters, or playground equipment that should not be covered. To handle this, the model will treat such features as internal polygonal obstacles (holes) in the input region, excluding them from coverage. The layout algorithm must then adjust strip generation and waste calculation to respect these no-cover zones. This entails detecting and clipping any turf strips overlapping an obstacle region, either splitting the strip into separate segments around the obstacle or removing that portion as unusable off-cut. Any turf area covering an obstacle would count as waste, prompting the algorithm to seek orientations and offsets that minimize such overlaps. Incorporating internal obstacles will require extending the current geometric computations (which assume a single, hole-free polygon) to handle complex polygons with multiple components. Although this adds computational complexity, it increases the realism and applicability of the tool. By accounting for typical site constraints like orientation alignments and interior no-go zones, future enhancements will make the system more robust and directly applicable to complex installation scenarios, bridging the gap between an idealized layout optimization and the practical challenges of real-world turf installations.

References

[1] James R. Jastifer, Andrew S. McNitt, Robert B. Anderson, Christina D. Mack, Richard W. Kent, Kirk A. McCullough, and Michael J. Coughlin. Synthetic turf: History, design, maintenance, and athlete safety. *Sports Health*, 11(1):84–90, 2018. doi:10.1177/1941738118793378.

[2] Baosu Guo, Yu Zhang, Jingwen Hu, Jinrui Li, Fenghe Wu, Qingjin Peng, and Quan Zhang. Two-dimensional irregular packing problems: A review. *Frontiers in Mechanical Engineering*, 8:966691, 2022. doi:10.3389/fmech.2022.966691.

[3] Andrew S. McNitt. Synthetic turf in the usa–trends and issues. *International Turfgrass Society Research Journal*, 10:27–33, 2005.

[4] B. Daviscourt, A. R. Kowalewski, B. McDonald, J. Lambrinos, and B. Eleveld. A life-cycle cost analysis of synthetic infill and natural grass systems. In *Proceedings of the International Turfgrass Society Conference*, page 373–379, New Brunswick, NJ, USA, 2017.

[5] Joseph C. Culberson and R. A. Reckhow. Covering polygons is hard. *Journal of Algorithms*, 17(1):2–44, 1994. doi:10.1006/jagm.1994.1002.

[6] Jie Fang, Xubing Chen, Yunqing Rao, Yili Peng, and Kuan Yan. A new approach for bin packing problem using knowledge reuse and improved heuristic. *Scientific Reports*, 14:31708, 2024. doi:10.1038/s41598-024-81749-5.

[7] Edgar Oviedo-Salas, Jesús David Terán-Villanueva, Salvador Ibarra-Martínez, Alejandro Santiago-Pineda, Mirna Patricia Ponce-Flores, Julio Laria-Menchaca, José Antonio Castán-Rocha, and Mayra Guadalupe Treviño-Berrones. Grasp optimization for the strip packing problem with flags, waste functions, and an improved restricted candidate list. *Applied Sciences*, 12(4):1965, 2022. doi:10.3390/app12041965.

[8] Germán Pantoja-Benavides, David Álvarez Martínez, and Francisco Parreño Torres. The normalized direct trigonometry model for the two-dimensional irregular strip packing problem. *Mathematics*, 12(15):2414, 2024. doi:10.3390/math12152414.

[9] Andreas Schutt, Peter J. Stuckey, and Andrew R. Verden. Optimal carpet cutting. In *Principles and Practice of Constraint Programming – CP 2011*, volume 6876 of *Lecture Notes in Computer Science*, pages 69–84, Berlin, Heidelberg, 2011. doi:10.1007/978-3-642-23786-7_8.

[10] Alaa Al Hawarneh, Salaheddine Bendak, and Firas Ghanim. Construction site layout planning problem: Past, present and future. *Expert Systems with Applications*, 168:114247, 2021. doi:10.1016/j.eswa.2020.114247.

[11] Ibrahim Abotaleb, Khaled Nassar, and Ossama Hosny. Layout optimization of construction site facilities with dynamic freeform geometric representations. *Automation in Construction*, 66:15–28, 2016. doi:10.1016/j.autcon.2016.02.007.

[12] Jingfa Liu, Jun Liu, Xueming Yan, and Bitao Peng. A heuristic algorithm combining pareto optimization and niche technology for multi-objective unequal area facility layout problem. *Engineering Applications of Artificial Intelligence*, 89:103453, 2020. doi:10.1016/j.engappai.2019.103453.

[13] James A. Farrell and Michael J. Barth. *The Global Positioning System & Inertial Navigation*. McGraw-Hill, New York, NY, USA, 1999. ISBN 9780070220454.

[14] Jackson School of Geosciences. Geo327g/386g: Gis & gps applications in earth sciences – map projections and coordinates. In *Lecture Slides, University of Texas at Austin*, Austin, TX, USA, 2021. `https://www.jsg.utexas.edu/`.

[15] Kai Cao, Chenghu Zhou, Richard Church, Xia Li, and Wenwen Li. Revisiting spatial optimization in the era of geospatial big data and geoai. *International Journal of Applied Earth Observation and Geoinformation*, 129:103832, 2024. doi:10.1016/j.jag.2024.103832.