Partitioning a Tile Arrangement for Optimal Construction by a Team of Simple Robots

Sándor P. Fekete*[1], Jonas Friemel*[2], Prahlad Narasimhan Kasthurirangan[3], Ramin Kosfeld*[4], Christian Scheffer*[2], and Arne Schmidt[4]

1 **Department of Computer Science, TU Braunschweig and L3S, Germany**
`s.fekete@tu-bs.de`
2 **Department of Electrical Engineering and Computer Science, Bochum University of Applied Sciences, Germany**
`{jonas.friemel, christian.scheffer}@hs-bochum.de`
3 **Applied Mathematics and Statistics Department at SUNY, Stony Brook University, New York, USA**
`prahladnarasim.kasthurirangan@stonybrook.edu`
4 **Department of Computer Science, TU Braunschweig, Germany**
`{kosfeld, aschmidt}@ibr.cs.tu-bs.de`

Abstract

We investigate the problem of optimally partitioning a given polyomino-shaped area into a disjoint set of connected subareas, such that a team of simple robots can build it in shortest possible time. Each subarea must contain its own depot from a given set, which supplies the robots with tile-sized building material. A robot can carry one tile at a time from its supply depot to a new position, while moving on existing tiles of its subconfiguration. As a result, the cost for constructing a subarea corresponds to the sum of distances from its respective depot to all of its tile sites and the total cost is the cost of the most expensive subarea. We provide hardness results for two variants of the resulting clustering problem. For given depots, it is NP-hard to decide if tiles can be assigned to the depots with a maximum cost of 3 even for rectangular instances. If the depot locations can be chosen, the problem is already hard for a maximum cost of 2. For lower costs, both variants are in P.

1 Introduction

How can a team of simple active robots [17] build and modify structures consisting of many basic components? Such questions play an important role, both in fundamental theory and in a spectrum of practical settings, with sizes ranging from very large [12] to very small dimensions [25], both of which are hard to access for direct human manipulation, e.g., in extraterrestrial space [4] or in microscopic environments [2].

As illustrated in Figure 1, it is possible to address such challenges by employing simple autonomous agents with minimal capabilities: The shown robot can move on the tile arrangement, remove individual tiles and physically relocate them to a new position by walking on the remaining configuration, which needs to remain connected at all times.

Previous work [3, 14] has described how to handle these challenges with a single active robot. Here we take an important step towards more scalable methods: How can we split the full task between multiple robots to achieve shorter overall construction times? To this end, we assume that building material (consisting of a supply of identical, pixel-shaped tiles)

* Supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 530918134.

42nd European Workshop on Computational Geometry, Hagen, Germany, March 25–27, 2026.
This is an extended abstract of a presentation given at EuroCG'26. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear eventually in more final form at a conference with formal proceedings and/or in a journal.

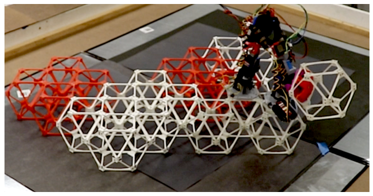
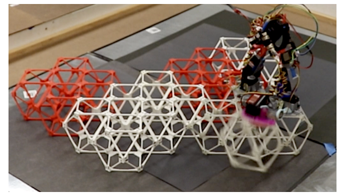

Figure 1 A simple BILL-E robot (left) carrying a light-weight tile while moving on an existing configuration and (right) placing an individual voxel for extending it. Figures by Yannuzzi [23].

is provided from a set of given depots. To ensure that robots do not collide, we partition the region into disjoint subregions, each containing a depot. Robots can pick up the necessary tiles to carry them to and place them at their destinations, while walking over previously constructed parts of their respective subregions. As a consequence, the resulting construction cost for each subregion is the sum of distances from the respective depot and the total construction cost of the system is the maximum construction cost for any subregion.

Our contributions. We investigate the problem of optimally subdividing a given polyomino-shaped area into a disjoint set of connected regions with centers, such that the maximum sum of geodesic distances of tiles assigned to a center is minimized. The cost function corresponds to the time it takes a team of simple robots to construct the shape, but the underlying clustering problem is of independent interest. We provide the following results.

(1) Deciding whether tiles in a polyomino P can be assigned to given cluster centers is NP-hard with a maximum cost of 3 per center, even if P is a rectangle (Theorem 4.2), but solvable in polynomial time if the cost is at most 2 (Theorem 4.4).
(2) If the locations of the cluster centers can be freely chosen, the problem is NP-hard for a cost of 2 (Theorem 5.1), but solvable in polynomial time for a cost of 1 (Theorem 5.2).

Due to limited space, we only provide proof sketches to theorems marked with ($\star$).

2 Preliminaries

For a set $P \subset \mathbb{Z}^2$ of N integer grid points in the plane, the graph G_P is the induced grid graph, in which two vertices $p_1, p_2 \in P$ are connected by an edge exactly if their L_1-distance is 1. The distance $d_P : P^2 \to \mathbb{N}$ is the length of a shortest path between two points $p_1, p_2 \in P$ in G_P. We call P *connected* if and only if G_P is connected. If P is connected, we consider its *polyomino* by replacing each point $p \in P$ with a unit square centered at p, which we refer to as *tile*. Throughout the paper, if the context is clear, we refer to P as the polyomino, and to G_P as the dual graph of the polyomino.

Let P be a polyomino and $C \subseteq P$ a set of *centers*. Consider an assignment function $\mathcal{A} : P \to C$ of all tiles in P to tiles in C, where $\mathcal{A}(c) = c$ for all $c \in C$. We denote the set of all tiles assigned to some $c \in C$ by $\mathcal{A}^{-1}(c)$ and for $t \in \mathcal{A}^{-1}(c)$, we say that c *covers* t. If $\mathcal{A}^{-1}(c)$ is connected for all $c \in C$, we call $\mathcal{A}$ a C*-clustering* in P. In such a C-clustering, we define the *weight* of a center $c \in C$ as the sum of distances in the subgraph $G_{\mathcal{A}^{-1}(c)}$ of G_P induced by $\mathcal{A}^{-1}(c)$ from every tile in $\mathcal{A}^{-1}(c)$ to c, i.e., $w(c) = \sum_{t \in \mathcal{A}^{-1}(c)} d_{\mathcal{A}^{-1}(c)}(t, c)$, and denote it by $w(c)$. The weight $w(\mathcal{A})$ of $\mathcal{A}$ is $\max_{c \in C} w(c)$.

In this paper, we consider the following two problems.

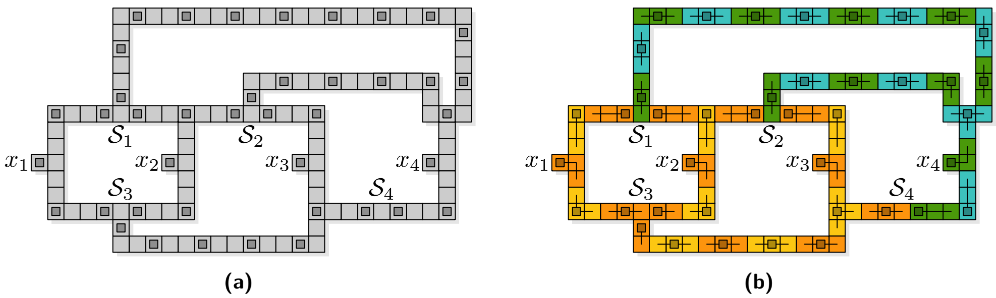

Figure 2 (a) A PCAP construction for the 3-SAT instance $\mathcal{S} = \bigwedge_{i=1}^{4} \mathcal{S}_i = (x_1 \vee x_2 \vee x_4) \wedge (x_2 \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3}) \wedge (\overline{x_3} \vee \overline{x_4})$ in the proof of Lemma 4.1. Tiles are the bigger squares and cluster centers are marked with smaller squares. **(b)** Tile assignments are visualized with lines from the centers and alternating colors (green-cyan for positive and orange-yellow for negative variables). $\mathcal{S}$ is satisfied with $(x_1, x_2, x_3, x_4) = (\texttt{false}, \texttt{false}, \texttt{false}, \texttt{true})$.

Polyomino Cluster Assignment Problem (PCAP)
Given: The dual graph G_P of a polyomino P, a set $C \subseteq P$, and a number $\sigma \in \mathbb{N}$.
Question: Is there a C-clustering of weight at most σ?

Polyomino Clustering Problem (PCP)
Given: The dual graph G_P of a polyomino P and numbers $k, \sigma \in \mathbb{N}$.
Question: Is there a set $C \subseteq P$ with $|C| \leq k$ that yields a C-clustering of weight at most σ?

Clearly, both PCP and PCAP are in NP as we can compute the weight of a C-clustering in time polynomial in the size of G_P and verify whether $|C| \leq k$ and $w(C) \leq \sigma$.

3 Related Work

Clustering is a fundamental problem in computer science; see, for example, [8, 13, 19, 20]. While work on classic variants such as k-mean and k-median spans decades [5, 7, 16, 21], our problems of interest are in the intersection of two relatively new clustering variants: connected k-clustering [10, 11, 24] and minimum load k-facility location [1]. In the language of [1] and [11], the variants we study in this work can be stated as Connected Minimum Load Facility Location with Disjoint Clusters on Grid Graphs. Carlsson and Devulapalli study a continuous variant of our problem for different metrics [6].

4 Polyomino Cluster Assignment Problem

▶ **Lemma 4.1.** PCAP *is* NP*-hard for* $\sigma = 3$.

Proof. We reduce from Planar Monotone 3-SAT [9], i.e., planar 3-SAT where each clause consists of either only positive or only negative literals. Let $\mathcal{S}$ be such a 3-SAT instance with clauses $\mathcal{S}_i$. We show how to construct a polyomino P and a set $C \subseteq P$ that has a C-clustering of weight at most 3 if and only if $\mathcal{S}$ is satisfiable.

By [18], we can embed the planar graph for $\mathcal{S}$ such that all vertices lie at integer positions and all edges are either vertical or horizontal with blowup polynomial in $|\mathcal{S}|$. Based on this embedding, we build a polyomino by placing the gadgets for variables and clauses depicted in Figure 3 accordingly. All tiles marked with a square in Figure 3 are centers, i.e., elements of C. It remains to interconnect the constructed gadgets by wires where exactly every third tile is

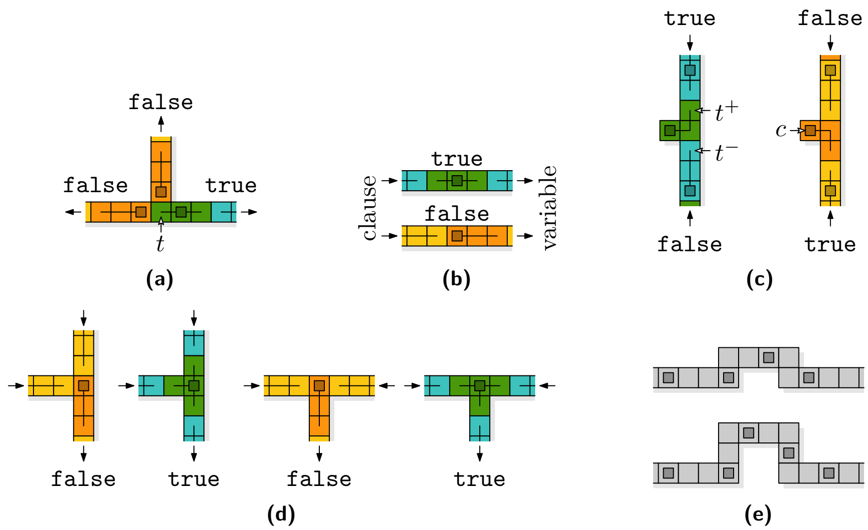

Figure 3 Gadgets: **(a)** positive clause gadget, **(b)** wire gadget from a positive clause, **(c)** variable gadgets, **(d)** splitter gadgets from positive clauses in both possible configurations, and **(e)** offset gadgets. Most gadgets are shown twice to indicate valid assignments depending on variable assignment. Gadgets from negative clauses can be constructed analogously with `true` and `false` flipped.

a center, see Figure 3(b). This construction is always possible (with enough spacing between the vertices) because we can introduce detours on straight wires to achieve every possible offset modulo 3 at the end of a wire, see Figure 3(e). If a literal occurs in multiple clauses, we place splitter gadgets along the wires, see Figure 3(d). For a full example, see Figure 2(a).

It remains to show that there exists a C-clustering of P with weight at most 3 exactly if there exists a satisfying assignment A to $\mathcal{S}$. We consider both directions separately. Given a satisfying assignment A to $\mathcal{S}$, we construct a C-clustering of weight 3 by assigning tiles as shown for every gadget in Figure 3; a complete instance is shown in Figure 2(b). Note that since there is at least one satisfying literal in every clause, at least one center in every clause gadget has enough *remaining capacity* to cover the gadget's t-tile (Figure 3(a)), where we define the remaining capacity of a center $c \in C$ as $\sigma - w(c)$.

Given a C-clustering with weight at most 3, we create a satisfying assignment A to $\mathcal{S}$ as follows. For every variable gadget, we check if tile t^+ in Figure 3(c) is assigned to the gadget's center and, exactly in this case, set the corresponding variable to `true`. We show that this assignment implies that every clause in $\mathcal{S}$ is satisfied.

By construction, the path from a clause gadget to its variable gadget follows the strict pattern of length-2 subpaths that are separated by single center tiles. We denote these subpaths as *segments*. Since segments are only connected to two centers, both of their tiles must be assigned to one of them. A segment is *mixed*, if the tile farthest away from the clause is assigned to the farther center. For example, the segments in the top wire in Figure 3(b) are mixed while those in the bottom are not. We observe that being mixed propagates to all further segments in the direction of the variable gadget: In the subsequent segment, the closer center already covers a tile of the previous segment, and therefore not all tiles of the current segment can be assigned to it, as this would exceed the center's remaining capacity. Therefore, the tile farthest away must again be assigned to the next center.

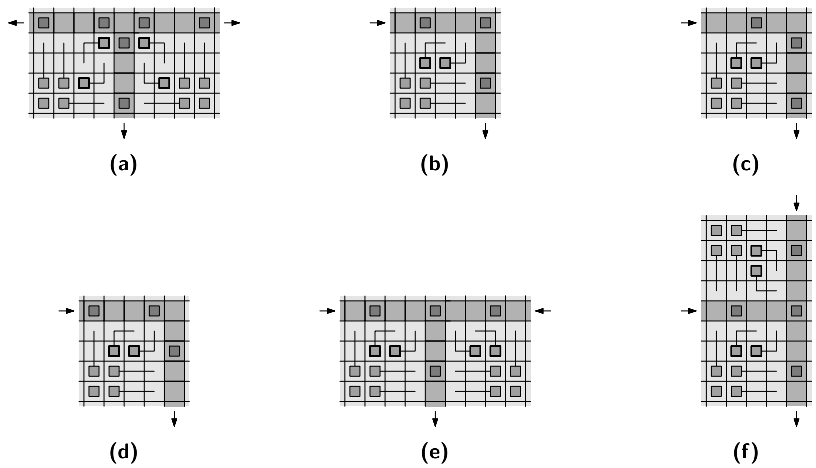

Figure 4 In convex corners, covering all newly-added tiles requires additional centers: **(a)** clause gadgets, **(b)**–**(d)** all cases of convex wire corners, and **(e)**–**(f)** splitters, up to rotation and reflection. For each setting, a valid assignment is indicated.

Now, consider a clause $\mathcal{S}_i$ of $\mathcal{S}$. In the corresponding clause gadget, the middle tile (labeled t) is assigned to one of the adjacent centers, which therefore does not have sufficient remaining capacity to be assigned all tiles on the following segment. Therefore, it and all further segments up to the respective variable gadget are mixed. Arriving in the variable gadget, there are two cases. If the corresponding literal was positive (1), the wire connects to the variable gadget from the top in Figure 3(c). Since the top center does not have sufficient remaining capacity to cover the t^+-tile, the only possibility is that t^+ gets assigned the left center (labeled c). Consequently, $\mathcal{S}_i$ is satisfied in A, because we set the corresponding variable to `true`. Analogously, if the corresponding literal was negative (2), the tile labeled t^- is assigned to the left center, and the remaining capacity does not suffice to cover t^+ as well. Thus, we set the corresponding variable to `false` and $\mathcal{S}_i$ is satisfied in A. ◀

By appropriately stretching the embedding and filling the holes in the construction, we can show that the problem remains NP-hard for $\sigma = 3$ if P is a rectangle, see Figure 5.

▶ **Theorem 4.2** ($\star$). *PCAP is NP-hard for $\sigma = 3$ even if P is a rectangle.*

Proof sketch. We stretch the polyomino P constructed in Lemma 4.1 and fill all empty positions in the bounding box around P with tiles. We then place centers at every tile with a distance of at least three from P. While these centers are too far away to influence the gadgets and cover a majority of the newly added tiles, they cannot cover all tiles in convex corners formed by gadgets. Figure 4 shows the positions of additional centers in all such corners, paired with valid assignments to cover all tiles. See Figure 5 for a full construction. As these centers are close enough to P to influence its gadgets, we analyze all possible corners to ensure that the mixed property is still propagated from clause to variable gadgets. ◀

Consequently, the smallest feasible σ for a PCAP instance cannot be approximated with a factor better than $4/3$ unless P = NP, i.e., there is no PTAS for this variant.

The NP-hardness of PCAP in Lemma 4.1 also extends to triangle numbers. We define the triangle number $T(i)$ of $i \in \mathbb{N}$ as the sum of all natural numbers up to i, i.e., $T(i) := \sum_{k=1}^{i} k$.

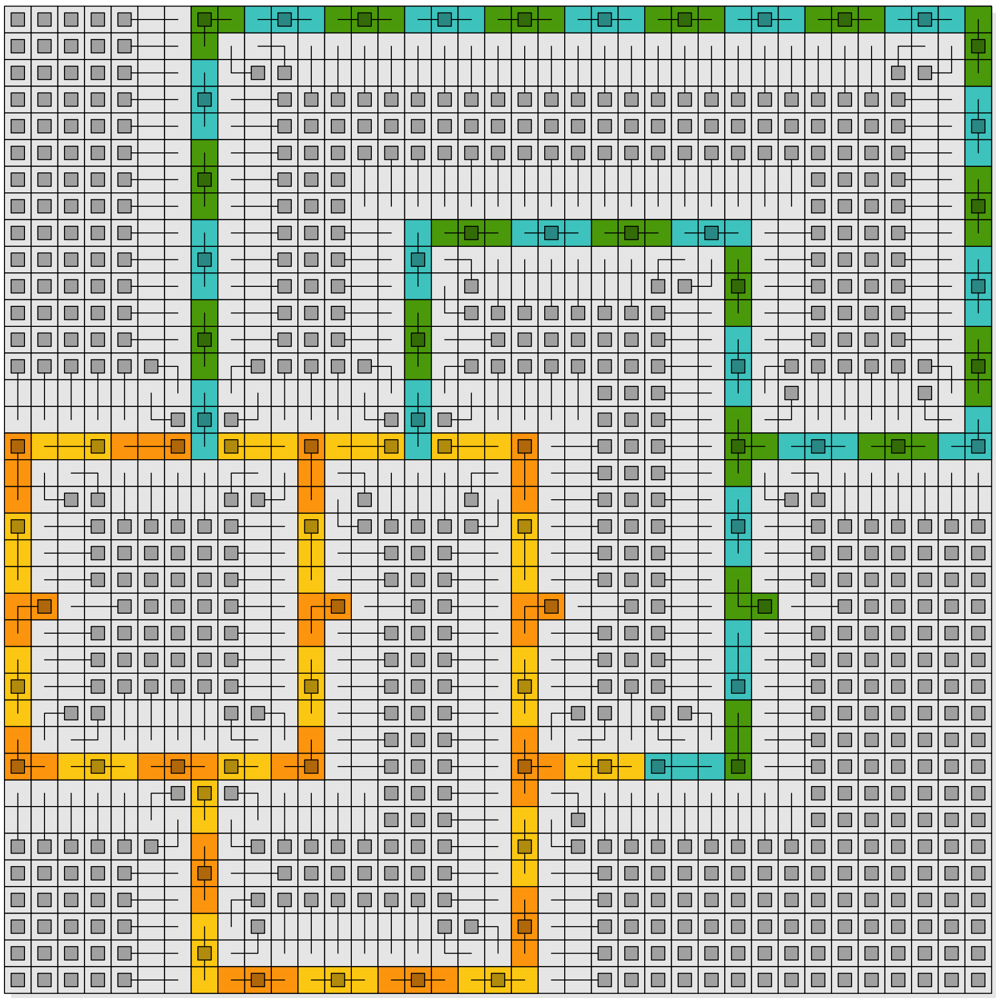

Figure 5 A rectangular PCAP instance with tile assignments for the same 3-SAT instance as in Figure 2. The colored tiles represent the embedded construction from Figure 2 (stretched), empty positions are filled with gray tiles. In the hole corners, the centers are placed as shown in Figure 4.

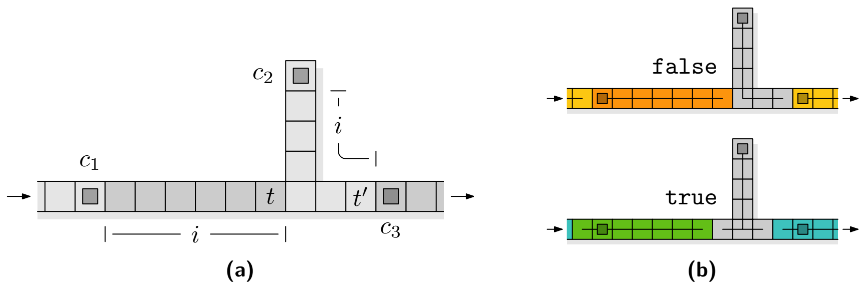

Figure 6 (a) The generalized offset gadget for $\sigma = T(i)$ is built from two paths of length i. The offset can be adapted by varying the number of tiles before the right path turns upward. **(b)** Valid assignments depend on the SAT variable assignments (shown for positive clauses).

▶ **Theorem 4.3** ($\star$). PCAP *is* NP*-hard for* $\sigma = T(i)$, $i \geq 2$.

Proof sketch. We construct a polyomino as described in Lemma 4.1, but stretch the segments to a length of i (instead of 2). As a center requires a remaining capacity of exactly $T(i)$ to cover all tiles of an adjacent segment, it cannot cover a single other tile from a different segment. Therefore, the property of being mixed is still propagated from clause gadgets to variable gadgets. The only gadget that needs to be adjusted is the offset gadget. To this end, we introduce a generalized offset gadget, see Figure 6. This gadget can be shown to equally propagate the mixed property along wires while introducing arbitrary offsets. ◀

We conjecture PCAP to be NP-hard for any constant $\sigma \geq 3$, but our proof technique cannot easily be extended beyond triangle numbers.

On the other hand, PCAP can be decided in polynomial time for $\sigma \leq 2$ with matchings.

▶ **Theorem 4.4.** PCAP *is decidable in polynomial time in the size of* G_P *for* $\sigma \leq 2$.

Proof. Let $\sigma \in \{1, 2\}$. In any valid solution, at most σ tiles can be assigned to a center $c \in C$ and they have to be adjacent to c. Otherwise, $w(c) \geq 3$. Therefore, we can restate the problem as a matching problem: For each center $c \in C$, we create up to two nodes $v_{c,1}, v_{c,\sigma}$, and for each other tile $t \in P \setminus C$, we create a node v_t. For $i \in \{1, \sigma\}$, we connect $v_{c,i}$ and v_t if the tiles of c and t are adjacent, yielding a bipartite graph G. If there is a matching in G into $V_{P \setminus C} = \{v_t \mid t \in P \setminus C\}$, i.e., any vertex in $V_{P \setminus C}$ is matched, then the PCAP instance is a yes-instance. As G has at most $2|P|$ nodes and $16|P|$ edges, any polynomial-time algorithm deciding whether there is a matching into $V_{P \setminus C}$ (e.g., Hopcroft-Karp [15]) decides PCAP. ◀

5 Polyomino Clustering Problem

▶ **Theorem 5.1.** PCP *is* NP*-hard for* $\sigma = 2$ *even if* G_P *has no vertex of degree four.*

Proof. It is NP-hard to partition a grid graph $G = (V, E)$ with maximum degree 3 into P_3 subgraphs (i.e., paths with three vertices) [22]. Because every connected grid graph is the dual graph of a polyomino, NP-hardness for PCP immediately follows by setting $k = |V|/3$ (if $|V|$ is not divisible by 3, G cannot be partitioned into P_3 subgraphs) and $\sigma = 2$ (the center will necessarily be the middle vertex of every P_3). ◀

As before, it follows that there is no PTAS for minimizing σ in PCP instances (with fixed k) and we cannot achieve an approximation factor better than $3/2$ unless P = NP.

▶ **Theorem 5.2.** PCP *is decidable in polynomial time in the size of* G_P *for* $\sigma = 1$.

Proof. Note that for each center c we have $|\mathcal{A}^{-1}(c)| \leq 2$, i.e., c itself and up to one other tile. Thus, we are searching for a maximum matching in G_P. Let M be a maximum matching in G_P, which can be found in time $O(|P|^{3/2})$ with the algorithm of Hopcroft-Karp [15]. Then, the PCP instance is a yes-instance if and only if $k \geq |M| + (|P| - 2|M|) = |P| - |M|$. ◀

6 Future Work

There are several directions for future work. On the theoretical side, most interesting is *any* constant-factor approximation algorithm; our results already indicate that a PTAS would imply P = NP. In our NP-hardness constructions, the number k of cluster centers is linear in the size of the configuration. In practical applications, k may be much smaller. As both PCAP and PCP bear resemblance to NP-hard partitioning problems even if k is constant, polynomial-time algorithms for arbitrary polyominoes seem unlikely, but the problems may be more tractable on restricted instance classes.

On the practical side, most relevant are exact methods that can solve instances of interesting size to provable optimality within reasonable time.

References

1 Sara Ahmadian, Babak Behsaz, Zachary Friggstad, Amin Jorati, Mohammad R. Salavatipour, and Chaitanya Swamy. Approximation algorithms for minimum-load k-facility location. *ACM Transactions on Algorithms*, 14(2):16:1–16:29, 2018. doi:10.1145/3173047.

2 Aaron T. Becker, Sándor P. Fekete, Phillip Keldenich, Dominik Krupke, Christian Rieck, Christian Scheffer, and Arne Schmidt. Tilt assembly: Algorithms for micro-factories that build objects with uniform external forces. *Algorithmica*, 82(2):165–187, 2020. doi:10.1007/S00453-018-0483-9.

3 Aaron T. Becker, Sándor P. Fekete, Jonas Friemel, Ramin Kosfeld, Peter Kramer, Harm Kube, Christian Rieck, Christian Scheffer, and Arne Schmidt. Efficient reconfiguration of tile arrangements by a single active robot. In *Canadian Conference on Computational Geometry (CCCG)*, pages 322–330, 2025. URL: https://cccg-wads-2025.eecs.yorku.ca/cccg-papers/7A3.pdf.

4 Mohamed Khalil Ben-Larbi, Kattia Flores Pozo, Tom Haylok, Mirue Choi, Benjamin Grzesik, Andreas Haas, Dominik Krupke, Harald Konstanski, Volker Schaus, Sándor P. Fekete, Christian Schurig, and Enrico Stoll. Towards the automated operations of large distributed satellite systems. Part 1: Review and paradigm shifts. *Advances in Space Research*, 67(11):3598–3619, 2021. doi:10.1016/j.asr.2020.08.009.

5 Pavel Berkhin. A survey of clustering data mining techniques. In *Grouping Multidimensional Data: Recent Advances in Clustering*, pages 25–71. Springer, 2006. doi:10.1007/3-540-28349-8_2.

6 John Gunnar Carlsson and Raghuveer Devulapalli. Dividing a territory among several facilities. *INFORMS Journal on Computing*, 25(4):730–742, 2013. doi:10.1287/ijoc.1120.0536.

7 Victor Chepoi, Tristan Fevat, Emmanuel Godard, and Yann Vaxès. A self-stabilizing algorithm for the median problem in partial rectangular grids and their relatives. *Algorithmica*, 62(1):146–168, 2012. doi:10.1007/s00453-010-9447-4.

8 Mark de Berg, Leyla Biabani, Morteza Monemizadeh, and Leonidas Theocharous. Clustering in polygonal domains. In *International Symposium on Algorithms and Computation (ISAAC)*, pages 23:1–23:15, 2023. doi:10.4230/LIPIcs.ISAAC.2023.23.

9 Mark de Berg and Amirali Khosravi. Optimal binary space partitions for segments in the plane. *International Journal of Computational Geometry & Applications*, 22(03):187–205, 2012. doi:10.1142/S0218195912500045.

10 Lukas Drexler, Jan Eube, Kelin Luo, Dorian Reineccius, Heiko Röglin, Melanie Schmidt, and Julian Wargalla. Connected k-center and k-diameter clustering. *Algorithmica*, 86(11):3425–3464, 2024. doi:10.1007/s00453-024-01266-9.

11 Jan Eube, Kelin Luo, Dorian Reineccius, Heiko Röglin, and Melanie Schmidt. Connected k-median with disjoint and non-disjoint clusters. In *European Symposium on Algorithms (ESA)*, pages 63:1–63:14, 2025. doi:10.4230/LIPIcs.ESA.2025.63.

12 Sándor P. Fekete. Coordinating swarms of objects at extreme dimensions. In *International Workshop on Combinatorial Algorithms (IWOCA)*, pages 3–13, 2020. doi:10.1007/978-3-030-48966-3_1.

13 Andreas Emil Feldmann and Luca Foschini. Balanced partitions of trees and applications. *Algorithmica*, 71(2):354–376, 2015. doi:10.1007/s00453-013-9802-3.

14 Javier Garcia, Jonas Friemel, Ramin Kosfeld, Michael Yannuzzi, Peter Kramer, Christian Rieck, Christian Scheffer, Arne Schmidt, Harm Kube, Dan Biediger, Sándor P. Fekete, and Aaron T. Becker. Moving matter: Using a single, simple robot to reconfigure a connected set of building blocks. In *International Conference on Automation Science and Engineering (CASE)*, pages 2260–2266, 2025. doi:10.1109/CASE58245.2025.11163739.

15 John E. Hopcroft and Richard M. Karp. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing*, 2(4):225–231, 1973. doi:10.1137/0202019.

16 Anil K. Jain and Richard C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., USA, 1988.

17 Benjamin Jenett, Amira Abdel-Rahman, Kenneth Cheung, and Neil Gershenfeld. Material–robot system for assembly of discrete cellular structures. *IEEE Robotics and Automation Letters*, 4(4):4019–4026, 2019. doi:10.1109/LRA.2019.2930486.

18 Donald E. Knuth and Arvind Raghunathan. The problem of compatible representatives. *SIAM Journal on Discrete Mathematics*, 5(3):422–427, 1992. doi:10.1137/0405033.

19 Eunjin Oh, Sang Won Bae, and Hee-Kap Ahn. Computing a geodesic two-center of points in a simple polygon. *Computational Geometry*, 82:45–59, 2019. doi:10.1016/j.comgeo.2019.05.001.

20 Eunjin Oh, Jean-Lou De Carufel, and Hee-Kap Ahn. The geodesic 2-center problem in a simple polygon. *Computational Geometry*, 74:21–37, 2018. doi:10.1016/j.comgeo.2018.02.008.

21 Josh Reese. Solution methods for the p-median problem: An annotated bibliography. *Networks*, 48(3):125–142, 2006. doi:10.1002/net.20128.

22 René van Bevern, Robert Bredereck, Laurent Bulteau, Jiehua Chen, Vincent Froese, Rolf Niedermeier, and Gerhard J. Woeginger. Partitioning perfect graphs into stars. *Journal of Graph Theory*, 85(2):297–335, 2017. doi:10.1002/jgt.22062.

23 Michael Yannuzzi. Personal communication: Figure provided by e-mail, March 2026.

24 William Chung-Kung Yen and Chien-Tsai Chen. The p-center problem with connectivity constraint. *Applied Mathematical Sciences*, 1(27):1311–1324, 2007. URL: https://www.m-hikari.com/ams/ams-password-2007/ams-password25-28-2007/yenAMS25-28-2007.pdf.

25 Yangsheng Zhu, Mingjing Qi, Zhiwei Liu, Jianmei Huang, Dawei Huang, Xiaojun Yan, and Liwei Lin. A 5-mm untethered crawling robot via self-excited electrostatic vibration. *IEEE Transactions on Robotics*, 38(2):719–730, 2022. doi:10.1109/TRO.2021.3088053.