Bundling Three Convex Polygons to Minimize Area or Perimeter*

Dongwoo Park[†] Sang Won Bae[‡] Helmut Alt[§] Hee-Kap Ahn[†]

31[st] May, 2014

Abstract

Given a set $\mathcal{P} = \{P_0, \ldots, P_{k-1}\}$ of k convex polygons having n vertices in total in the plane, we consider the problem of finding k translations $\tau_0, \ldots, \tau_{k-1}$ of $P_0, \ldots, P_{k-1}$ such that the translated copies $\tau_i P_i$ are pairwise disjoint and the area or the perimeter of the convex hull of $\bigcup_{i=0}^{k-1} \tau_i P_i$ is minimized. When $k = 2$, the problem can be solved in linear time but no previous work is known for larger k except a hardness result: it is NP-hard if k is part of input. We show that for $k = 3$ the translation space of input polygons can be decomposed into $O(n^2)$ cells in each of which the combinatorial structure of the convex hull remains the same. Moreover, we show that the description of the objective function for each cell can be fully described using constant space and the function description for a neighboring cell can be obtained in constant time by using coherence. Based on this decomposition, we present a first $O(n^2)$-time algorithm that finds optimal translations of input polygons minimizing the area or the perimeter of the corresponding convex hull.

1 Introduction

We consider the problem of finding translations of k convex polygons such that they are contained in a smallest possible convex region while their interiors are disjoint. This problem can be modelled as follows: given a set $\mathcal{P} = \{P_0, \ldots, P_{k-1}\}$ of k convex polygons in the plane with n vertices in total, find k translations $\tau_0, \ldots, \tau_{k-1}$ of $P_0, \ldots, P_{k-1}$ such that the translated copies $\tau_i P_i$'s, for $0 \leqslant i \leqslant k-1$, do not overlap each other and the area or the perimeter of the convex hull of $\bigcup_{i=0}^{k-1} \tau_i P_i$ is minimized.

This problem can be seen as a generalization of a *packing problem* of finding a smallest region, called a *container*, of a given shape (such as a disk, a square, or a rectangle) that packs the input objects under translations. Packing problems have received significant attention in a number of disciplines. For instance, it goes back to Kepler's conjecture (1611) on sphere packing in three-dimensional Euclidean space [8]. Sugihara et al. [11] considered a related problem of minimizing the disk bundling a set of disks with applications to minimizing the sizes of holes through which sets of electric wires are to pass. They proposed a heuristic method that makes use of the Voronoi diagram of circles. Milenkovic studied the packing of a set of polygons into another

*The research by H.-K. Ahn and D. Park was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2011-0030044) The research by S.W.Bae was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning(2013R1A1A1A05006927).

[†]Department of Computer Science and Engineering, POSTECH, Pohang, Korea. {dwpark,heekap}@postech.ac.kr

[‡]Department of Computer Science, Kyonggi University, Suwon, Korea. swbae@kgu.ac.kr

[§]Freie Universität Berlin, Germany. alt@mi.fu-berlin.de

polygon container with applications in the apparel industry [10]. He gave a $O(n^{k-1}\log n)$ time algorithm for packing k convex n-gons under translations into a minimum area axis-parallel rectangle container. Later, Alt and Hurtado [3] presented a near-linear time algorithm that packs two convex polygons into a minimum area or perimeter rectangle. Recently, Egeblad et al. [6] presented an efficient method for packing polytopes into another polytope container under translation in arbitrary dimension.

Much less is known about the case when the container has no restriction on its shape. For $k = 2$, Lee and Woo [9] presented a linear time algorithm for finding a translation τ_1 of P_1 that minimizes the convex hull area of $P_0 \cup \tau_1 P_1$. It is not difficult to see that their algorithm also works for minimizing the perimeter of the convex hull. When k is part of input, the problem is shown to be NP-hard even if the polygons are rectangles [4], which was done by reducing the partition problem [7] into this problem.

Later, Tang et al. [12] considered the case the the objects can be reoriented freely and gave an $O(n^3)$-time algorithm for finding a rigid motion that minimizes the area of the convex hull. Very recently, Ahn and Cheong [1, 2] presented a near-linear time approximation algorithm for finding a rigid motion that minimizes either the perimeter or the area of the convex hull.

Our results. We study the problem of bundling three convex polygons under translations where we restrict their interiors to remain disjoint and their orientations to be fixed while we allow translating them. Without loss of generality, we assume that P_0 is stationary. We show that the translation space of P_1 and P_2 can be decomposed into $O(n^2)$ cells in each of which the combinatorial structure of the convex hull remains the same. Moreover, we show that the description of the objective function for each cell can be fully described using constant space and the function description for a neighboring cell can be obtained in constant time by coherence. This was done by a careful analysis on all event configurations at which the combinatorial structure of the convex hull changes. We then present an $O(n^2)$-time algorithm that returns an optimal pair of translations that minimizes the area or the perimeter of the convex hull of the union.

2 Preliminaries

Let $P_0, \ldots, P_{k-1}$ be k convex polygons in $\mathbb{R}^2$ with n vertices in total. For a vector $\tau \in \mathbb{R}^{2k}$, we write $\tau = (\tau_0, \ldots, \tau_{k-1})$, where $\tau_i \in \mathbb{R}^2$. The *translate* of P_i by τ_i, denoted by $\tau_i P_i$, is $\{a + \tau_i \mid a \in P_i\}$. We let $U(\tau) = \bigcup_{i=0}^{k-1} \tau_i P_i$ and let $\mathrm{conv}(\tau) := \mathrm{conv}(U(\tau))$.

Ahn and Cheong [2] studied the area and perimeter functions and observed the following.

Lemma 1 (Ahn and Cheong [2]) *The function $f : \mathbb{R}^{2k} \to \mathbb{R}$ with $f(\tau) = |\mathrm{conv}(\tau)|$ is convex for any $k \geqslant 2$. The function $g : \mathbb{R}^{2k} \to \mathbb{R}$ with $g(\tau) = \|\mathrm{conv}(\tau)\|$ is convex and piecewise linear for $k = 2$, but this is not necessarily the case for $k > 2$.*

Our problem can be viewed as an optimization problem of minimizing the area $\|\mathrm{conv}(\tau)\|$ or the perimeter $|\mathrm{conv}(\tau)|$ over $\tau \in \mathbb{R}^{2k}$ subject to $\tau_i P_i \cap \tau_j P_j = \emptyset$ for all $0 \leqslant i < j \leqslant k-1$. One can reduce the search space by a simple observation.

Lemma 2 *For the bundling problem with respect to either area or perimeter, there is an optimal translation vector $\tau^* \in \mathbb{R}^{2k}$ such that the union $U(\tau^*)$ is connected, that is, every translate touches another translate under τ^*.*

Proof. If $U(\tau^*)$ is connected, we are done. Suppose that $U(\tau^*)$ consists of more than one connected components and C is any one of them. If C appears on the boundary of the convex hull $\mathrm{conv}(\tau^*)$, then one can clearly translate C to have less area and perimeter. Thus, C is completely contained in the interior of $\mathrm{conv}(\tau^*)$. We then translate it freely to make it touch with any other component of $U(\tau^*)$, keeping it inside $\text{conv}(\tau^*)$. This translation process makes no change of their convex hull but decrease the number of connected components by one. We can repeat this process for any connected component of $U(\tau^*)$ until we finally have a single connected component. □

We can thus concentrate only on the cases where the k polygons are connected. We shall call $\tau \in \mathbb{R}^{2k}$ a *configuration* if $U(\tau)$ is connected. A configuration τ is *feasible* if and only if the interiors of the translates are disjoint under τ. Thus, our goal is to find an optimal feasible configuration with respect to area or perimeter.

Let $\mathcal{K}$ be the set of configurations for given k polygons $P_0, \ldots, P_{k-1}$. Each configuration $\tau \in \mathcal{K}$ is associated with several properties describing the structure of the convex hull $\text{conv}(\tau)$. If $\tau_i P_i$ and $\tau_j P_j$ are in contact, then a vertex v of P_i lies on an edge e of P_j under τ, or vice versa. We call the pair (v, e) a *contact* induced by τ. Let $C(\tau)$ be the set of contacts induced by a configuration $\tau \in \mathcal{K}$. Note that Lemma 2 implies that $|C(\tau)| \geqslant k - 1$.

The convex hull $\text{conv}(\tau)$ is bounded by a closed polygonal curve consisting of some edges of the translated polygons $\tau_i P_i$'s and line segments connecting two vertices of the $\tau_i P_i$'s. We call such a segment of the latter type a *bridge*. More specifically, there is a bridge (u, v) connecting a vertex u of a polygon and a vertex v of another polygon if these polygons appear consecutively along the boundary of the convex hull. Note that a bridge (u, v) is a degenerate edge of length 0 if u and v are in contact. Let $H(\tau)$ be the set of those pairs of vertices induced by $\tau \in \mathcal{K}$. Two configurations $\tau, \tau' \in \mathcal{K}$ are said to have the same combinatorial structure if both $C(\tau) = C(\tau')$ and $H(\tau) = H(\tau')$ hold.

In the following sections, we will show that the configuration space can be decomposed into a number of cells in which the configurations have the same combinatorial structure, so that the area or the perimeter function is described and minimized. For convenience of elaboration, we make a general position assumption on input polygons in the sense that no two polygon edges are parallel.

3 The Configuration Space for Three Polygons

In this section, we study the configuration space of three convex polygons P_0, P_1 and P_2 under translation. We first investigate the configuration space by introducing a parametrization of configurations. Then we define events and event curves in the configuration space from which the combinatorial structure of the convex hull or the motion of the input polygons changes, and analyze their complexity.

3.1 Parametrization of configurations

As a warm-up exercise, consider the case of $k = 2$ where two convex polygons P_0 and P_1 are given. By Lemma 2, any configuration $\tau \in \mathcal{K}$ requires P_1 to touch P_0. Imagine that P_0 is stationary and P_1 translates around P_0 in the counter-clockwise direction, keeping them touching each other, until P_1 then reaches back to the initial position. The set $\mathcal{K}$ of configurations thus forms a space homeomorphic to a unit circle. This motion of P_1 around P_0 is piecewise affine, and the total distance that P_1 travels is exactly $|P_0| + |P_1|$. Therefore, letting $L := |P_0| + |P_1|$, the interval $[0, L)$ fully describes the configuration space $\mathcal{K}$: For any $\lambda \in [0, L)$, let $\tau(\lambda)$ be the configuration whose corresponding translated copy of P_1 is a snapshot at a moment when P_1 travels a distance of exactly λ around P_0 from its initial position.

We now turn to the case of $k = 3$, where three convex polygons P_0, P_1, and P_2 are given as input. Lemma 2 implies that in any configuration $\tau \in \mathcal{K}$, at least one of the three polygons

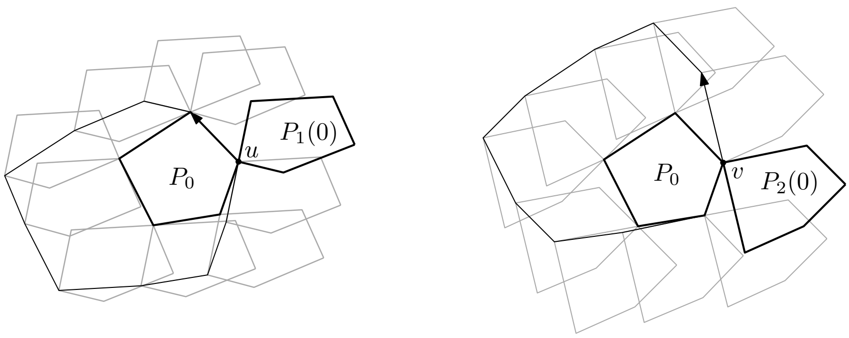

Figure 1: Sliding P_1 and P_2 around P_0: we parameterize the configuration space $\mathcal{K}_0$ by a pair of parameters (λ_1, λ_2) for $\lambda_1 \in [0, L_1)$ and $\lambda_2 \in [0, L_2)$.

touches the other two, simultaneously. Without loss of generality, we assume that both P_1 and P_2 translate around P_0 in the counter-clockwise direction keeping touching P_0, while P_0 remains stationary. Let $\mathcal{K}_0 \subset \mathcal{K}$ be the space of configurations in which $\tau_0 = (0, 0)$ and P_0 touches both of P_1 and P_2. As discussed above for $k = 2$, the distance that each of P_1 and P_2 travels around P_0 is exactly L_1 and L_2, respectively, where $L_1 := |P_0| + |P_1|$ and $L_2 := |P_0| + |P_2|$. See Figure 1. Then, any pair $(\lambda_1, \lambda_2) \in [0, L_1) \times [0, L_2)$ corresponds to a configuration $\tau(\lambda_1, \lambda_2)$ when P_1 and P_2 travel around P_0 by distance exactly λ_1 and λ_2, respectively, from their initial positions.

Notice that the definition of configurations do not prevent P_1 and P_2 from overlapping each other; rather, the translates of P_1 and P_2 around P_0 are independent, and are determined independently by two different parameters λ_1 and λ_2, respectively. We denote by $P_1(\lambda_1)$ and $P_2(\lambda_2)$ the translated copy of P_1 and P_2, respectively, corresponding to the parameters λ_1 and λ_2, respectively. By abuse of notation, we shall call a pair (λ_1, λ_2) a configuration in $\mathcal{K}_0$ and regard $\mathcal{K}_0$ to be $[0, L_1) \times [0, L_2)$.

3.2 Events and event curves

Recall that any configuration $\tau \in \mathcal{K}_0$ is associated with the set $C(\tau)$ of contacts and the set $H(\tau)$ of bridges of the corresponding convex hull $\text{conv}(\tau)$. These two combinatorial associates determine the structure of the convex container and the motion of the polygons, thus being helpful in describing the objective function on the configuration space as will be shown in next sections. One natural approach would decompose the configuration space, $\mathcal{K}_0$ into cells in each of which $C(\tau)$ and $H(\tau)$ remain the same for all configurations τ in the cell.

We call a configuration $\tau = (\lambda_1, \lambda_2) \in \mathcal{K}_0$ an *event* if it is one of the following cases:

C0 event A vertex of $P_1(\lambda_1)$ or $P_2(\lambda_2)$ reaches a vertex of P_0; that is, a vertex-vertex contact

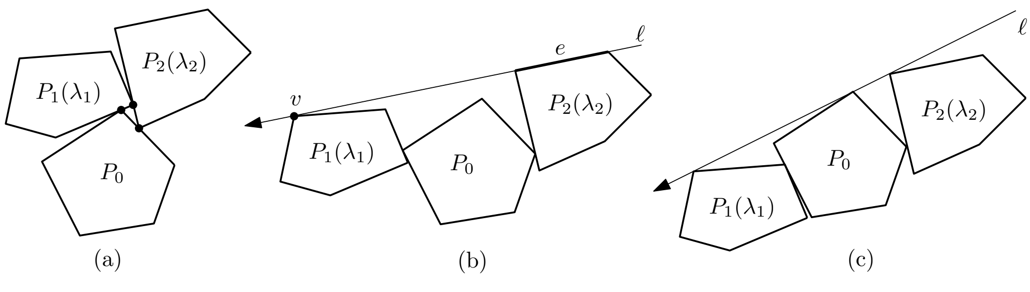

Figure 2: Corresponding translates of the three polygons at events of different types: (a) C2 event, (b) H1 event, and (c) H2 event.

occurs between P_0 and one of the others.

C1 event $P_1(\lambda_1)$ and $P_2(\lambda_2)$ touch each other at a vertex of $P_1(\lambda_1)$ and a vertex of $P_2(\lambda_2)$; that is, a vertex-vertex contact occurs between P_1 and P_2.

C2 event $P_1(\lambda_1)$ and $P_2(\lambda_2)$ touch each other; that is, the three polygons are pairwise touching and it holds that $|C(\tau)| = 3$. See Figure 2(a).

H0 event $P_i(\lambda_i)$, for $i = 1$ or 2, is tangent to the supporting line of an edge of P_0 from the side containing P_0, or vice versa.

H1 event $P_1(\lambda_1)$ is tangent to the supporting line of an edge of $P_2(\lambda_2)$ from the side containing $P_2(\lambda_2)$, or vice versa. See Figure 2(b).

H2 event The three polygons P_0, $P_1(\lambda_1)$, and $P_2(\lambda_2)$ have a common tangent line ℓ and the three lie in the same side of ℓ. See Figure 2(c).

Remark that $\mathcal{K}_0$ includes configurations whose corresponding translates of P_1 and P_2 may overlap each other; the set of C2 events indeed form the borderline between configurations causing overlap and those not causing overlap. Note, however, that all the changes of $C(\tau)$ and $H(\tau)$ can be captured by a series of the events, when τ continuously moves inside $\mathcal{K}_0$ while it avoids overlap between P_1 and P_2. In particular, although some portions of H1 events indeed imply an overlap between P_1 and P_2, it suffices to track all the changes of $H(\tau)$ by H0, H1, and H2 events if τ continuously moves without any overlap. On the other hand, events of type C1 and C2 by definition imply no overlap between P_1 and P_2. Also, any C1 event is a C2 event by definition.

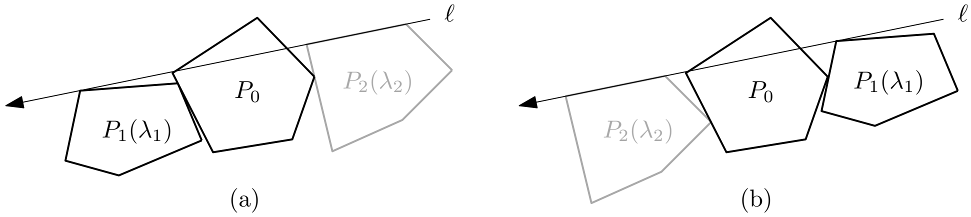

Figure 3: Illustration of decomposing the set of H1 events. There are two different types of translations for P_1 around P_0 for a directed line ℓ tangent to P_1; (a) P_1 is ahead of P_0, or (b) P_0 is ahead of P_1 along ℓ. The analogue also holds for P_2.

The set of all events forms a set of curves in the configuration space $\mathcal{K}_0 = [0, L_1) \times [0, L_2)$, and thus decomposes the space into cells. To see this more precisely, we partition the set of events into subsets as follows:

Curves of C0 events Any C0 event corresponds to a vertex-vertex contact, involving a pair (v, v') of vertices, exactly one of which belongs to P_0. We denote by $\gamma^{\mathrm{C0}}_{vv'} = \gamma^{\mathrm{C0}}_{v'v}$ the set of all C0 events with the involved pair (v, v').

Curves of C2 events For any C2 event $\tau = (\lambda_1, \lambda_2)$, $P_1(\lambda_1)$ and $P_2(\lambda_2)$ touch each other. We have two cases: either $P_1(\lambda_1)$ is ahead of $P_2(\lambda_2)$ (in the sense that $P_2(\lambda_2 + \epsilon)$ overlaps $P_1(\lambda_1)$ for arbitrarily small $\epsilon > 0$) as depicted in Figure 2(a), or $P_2(\lambda_2)$ is ahead of $P_1(\lambda_1)$. We denote the set of C2 events corresponding to the former by γ^{C2}_1 and the set of C2 events corresponding to the latter by γ^{C2}_2. Note that every C1 event coincides with a C2 event by definition, and thus all the C1 events are included in $\gamma^{\mathrm{C2}}_1 \cup \gamma^{\mathrm{C2}}_2$.

Curves of H0 events Any H0 event τ corresponds to a collinearity of an edge e and a vertex v, one of which belongs to P_0. Let ℓ be the supporting line of e and assume that ℓ is directed so that the two polygons that each of v and e belongs to lie on its left side. There are two cases: the vertex v is ahead of e or behind e, along the directed line ℓ. We denote the set of all H0 events corresponding to the former by $\gamma^{\mathrm{H0}}_{ve}$ and the set of all H0 events corresponding to the latter by $\gamma^{\mathrm{H0}}_{ev}$.

Curves of H1 events Any H1 event $\tau = (\lambda_1, \lambda_2)$ corresponds to a collinearity of an edge e and a vertex v, each of which belongs mutually to P_1 or P_2. Let ℓ be the supporting line of e and assume that ℓ is directed so that both $P_1(\lambda_1)$ and $P_2(\lambda_2)$ lie on its left side. Note that ℓ translates as e (and the polygon containing e) translates. Observe that there are two different types of translations for P_1 around P_0 such that ℓ keeps being tangent to the translate of P_1; P_1 is ahead of P_0 along the directed line ℓ (Figure 3(a)) or vice versa (Figure 3(b)). The analogue also holds for P_2. Thus, $\tau = (\lambda_1, \lambda_2)$ falls into one of the four cases. We denote by $\gamma^{\mathrm{H1}}_{ve,11}$ the set of H1 events (λ_1, λ_2) defined by (v, e) such that $P_1(\lambda_1)$ is ahead of P_0 and $P_2(\lambda_2)$ is also ahead of P_0. Similarly, define the other three $\gamma^{\mathrm{H1}}_{ve,12}$, $\gamma^{\mathrm{H1}}_{ve,21}$, and $\gamma^{\mathrm{H1}}_{ve,22}$. Figure 2(b) shows a H1 event in the set $\gamma^{\mathrm{H1}}_{ve,12}$.

Curves of H2 events Any H2 event $\tau = (\lambda_1, \lambda_2)$ is associated with a line ℓ commonly tangent to the three polygons P_0, $P_1(\lambda_1)$, and $P_2(\lambda_2)$. We assume that ℓ is always directed so that P_0, together with the other two, lies on its left side. We have again four cases as we have for H1 event curves; either $P_1(\lambda_1)$ (or $P_2(\lambda_2)$) is ahead of P_0 along ℓ or is behind P_0. We divide the set of H2 events into four subsets as we did for H1 events and denote them by $\gamma^{\mathrm{H2}}_{11}$, $\gamma^{\mathrm{H2}}_{12}$, $\gamma^{\mathrm{H2}}_{21}$, and $\gamma^{\mathrm{H2}}_{22}$, respectively. Figure 2(c) shows a H2 event in the set $\gamma^{\mathrm{H2}}_{12}$.

We let Γ be the family of those nonempty subsets of events defined above. We show in the following that every $\gamma \in \Gamma$ forms a monotone curve (or a set of monotone curves) in $\mathcal{K}_0$ with several nice behaviors.

Lemma 3 *Any set $\gamma \in \Gamma$ is monotone in both the λ_1-axis and the λ_2-axis, and consists of at most three curves on the configuration space $\mathcal{K}_0$. In addition, γ has following properties according to its type. The asterisks below mean "any."*

- *$\gamma = \gamma^{\mathrm{C0}}_*$ or γ^{H0}_*: γ is a line parallel to the λ_1-axis or the λ_2-axis.*
- *$\gamma = \gamma^{\mathrm{C2}}_*$: γ is non-decreasing and piecewise linear, each of whose breakpoints coincides with a C0 or C1 event.*
- *$\gamma = \gamma^{\mathrm{H1}}_*$: γ is monotone and piecewise linear, each of whose breakpoints coincides with a C0 event.*
- *$\gamma = \gamma^{\mathrm{H2}}_*$: γ is non-decreasing and piecewise hyperbolic, each of whose breakpoints coincides with a C0 or H0 event.*

Proof. We consider each type separately.

- *(C0 events) When $\gamma = \gamma^{\mathrm{C0}}_{vv'}$.*
  Without loss of generality, assume that v belongs to P_0 and v' belongs to P_1. By definition, such a C0 event implies a contact between v and v'. This fixes the position of P_1, and thus the value of λ_1. On the other hand, P_2 is free to slide around P_0. We thus have $\gamma = \{(c, \lambda_2) \mid \lambda_2 \in [0, L_2)\}$ for some constant $c \in [0, L_1)$. Therefore, γ is parallel to the λ_2-axis. Analogously, if v' belongs to P_2, then γ is parallel to the λ_1-axis.

- *(H0 events) When $\gamma = \gamma^{\mathrm{H0}}_{ve}$ or $\gamma^{\mathrm{H0}}_{ev}$.*
  This case is similar to C0 events; all the events in γ has a fixed λ_1- or λ_2-coordinate if P_1 or P_2 is involved in the event, respectively. Thus, γ is parallel to the λ_1-axis or to the λ_2-axis.

- *(C2 events) When $\gamma = \gamma^{\mathrm{C2}}_1$ or γ^{C2}_2.*
  Assume that $\gamma = \gamma^{\mathrm{C2}}_1$. The other case can be shown analogously. Recall that γ consists of

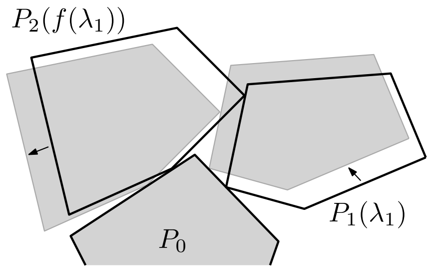

Figure 4: Illustration to the proof of Lemma 3 for C2 events.

all configurations $(\lambda_1, \lambda_2) \in \mathcal{K}_0$ such that the three polygons P_0, $P_1(\lambda_1)$, and $P_2(\lambda_2)$ are pairwise touching, and $P_2(\lambda_2)$ is ahead of $P_1(\lambda_1)$. Let us fix λ_1 to be a value in $[0, L_1)$. Then, we have two cases: (1) there exists a unique $\lambda_2 \in [0, L_2)$ such that the three polygons P_0, $P_1(\lambda_1)$, and $P_2(\lambda_2)$ are pairwise touching and $P_2(\lambda_2)$ is ahead of $P_1(\lambda_1)$, or (2) there are more than one such values of λ_2. The latter case in fact can happen when both P_0 and $P_1(\lambda_1)$ touches a common edge of $P_2(\lambda_2)$, so P_2 is able to slide while keeping in contact to both. This implies that γ is monotone because we can switch the role of λ_1 and λ_2 by symmetry, while γ may contain a *vertical* line segment in $\mathcal{K}_0$.

We now walk along γ by increasing λ_1 from 0 to L_1, and observe the behavior of the three polygons corresponding to the current point (λ_1, λ_2) on γ. Since $P_2(\lambda_2)$ should touch $P_1(\lambda_1)$ and be ahead of $P_1(\lambda_1)$, it also has to slide around P_0 in the counter-clockwise direction, as λ_1 increases. This implies that γ is continuous as λ_1 increases, except at the limits of $\mathcal{K}_0$ (that is, when $\lambda_1 = 0$ or L_1, or $\lambda_2 = 0$ or L_2), and γ has a non-negative slope at every point, so non-decreasing. In addition, one can easily see that the motion of $P_1(\lambda_1)$ and $P_2(\lambda_2)$ is linear while the contact $C(\tau)$ remains the same.

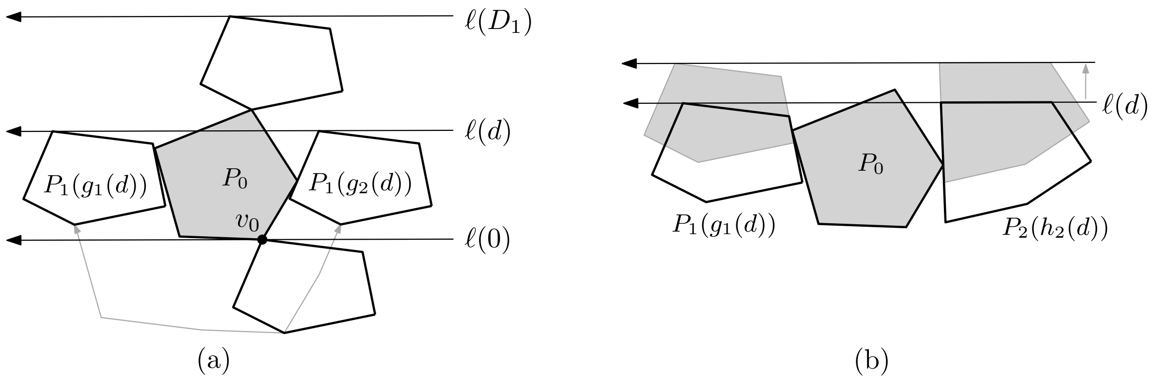

Figure 5: Illustration to the proof of Lemma 3 for H1 events.

- *(H1 events) When $\gamma = \gamma^{\mathrm{H1}}_{ve,ij}$ for some $1 \leqslant i, j \leqslant 2$.*
  We consider all the four subsets for fixed v and e, simultaneously. Without loss of generality, assume that e is horizontal and the polygon that e belongs to lies below e in $\mathbb{R}^2$. We also assume that the bottommost vertex v_0 of P_0 lies on the line $y = 0$. Define $\ell(d)$ for $d \geqslant 0$ to be the line $\{y = d\}$, directed to the left.

  Let t_0 be the value such that the topmost vertex of $P_1(t_0)$ touches v_0 of P_0. Let t_1 be the value such that $P_1(t_1)$ is the highest translate of P_1 that touches P_0, and let D_1 be such that $\ell(D_1)$ passes through the topmost vertex of $P_1(t_1)$. Observe then that for any $0 < d < D_1$, there are two translates of P_1 each of which touches P_0 and $\ell(d)$, and P_1

lies to the left of $\ell(d)$. See Figure 5(a). We denote these two translates by $P_1(g_1(d))$ and $P_1(g_2(d))$; we have $g_1(0) = g_2(0) = t_0$ and $g_1(D_1) = g_2(D_1) = t_1$, and $P_1(g_1(d))$ is to the left of $P_1(g_2(d))$ for any $0 < d < D_1$ in $\mathbb{R}^2$. Analogously, we define D_2 for P_2, and also define $h_1(d)$ and $h_2(d)$ for P_2 that are analogous to $g_1(d)$ and $g_2(d)$, respectively. Then, observe that $(g_i(d), h_j(d)) \in \gamma^{\mathrm{H1}}_{ve,ij}$ for any $0 \leqslant d \leqslant \min\{D_1, D_2\}$.

By adjusting the coordinate system of $\mathcal{K}_0$, we assume that $g_1(0) = g_2(0) = 0$ and $h_1(0) = h_2(0) = 0$. Then, both g_1 and h_1 are decreasing from L_1 and L_2, respectively, while both g_2 and h_2 are increasing from 0 because $P_1(g_1(d))$ and $P_2(h_1(d))$ slide around P_0 in the clockwise direction while $P_1(g_2(d))$ and $P_2(h_2(d))$ slide in the counter-clockwise direction as d increases. This also implies the continuity of g_i and h_j for any $1 \leqslant i, j \leqslant 2$. Putting it all together, we conclude that $\gamma^{\mathrm{H1}}_{ve,ij}$ is the graph of a partial function of λ_1 that is either increasing or decreasing; it is increasing if $i = j$, or decreasing otherwise. For example, Figure 5(b) illustrates the case of $(i, j) = (1, 2)$, where one observes that $\gamma^{\mathrm{H1}}_{ve,12}$ is the graph of a decreasing function. Hence, in the general case where $g_1(0) = g_2(0) \neq 0$ and $h_1(0) = h_2(0) \neq 0$, each $\gamma^{\mathrm{H1}}_{ve,ij}$ consists of at most three monotone curves whose endpoints lie at the limit of $\mathcal{K}_0$.

Finally, we show that f is piecewise linear with breakpoints at C0 events. For the purpose, consider the grid G on $\mathcal{K}_0$ generated by all C0 event curves and pick any grid cell σ intersected by γ. By definition, any configuration $\tau \in \sigma \cup \gamma$ has a fixed contact set $C(\tau)$. Then, it is not difficult to see that the function f is linear. (See Figure 5(b).) Therefore, each $\gamma^{\mathrm{H1}}_{ve,ij}$ is the graph of a monotone and piecewise linear function whose breakpoints coincide with C0 events. Note that all the endpoints of the curves $\gamma^{\mathrm{H1}}_{ve,ij}$ for any $1 \leqslant i, j \leqslant 2$ also coincide with C0 events.

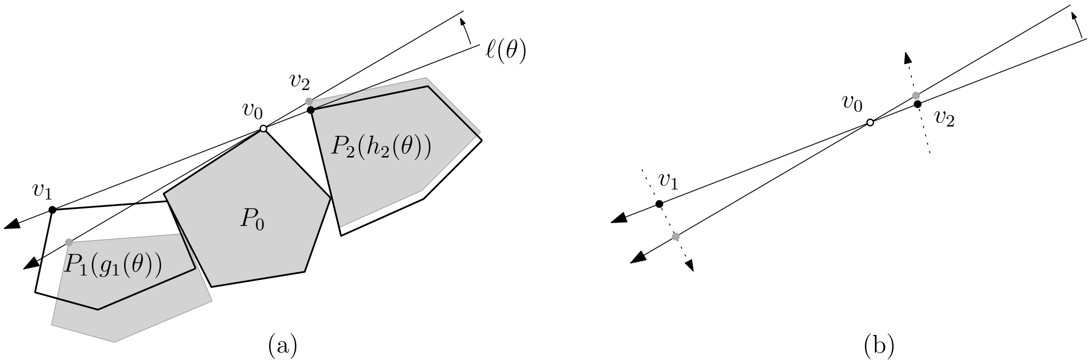

Figure 6: Illustration to the proof of Lemma 3 for H2 events.

- *(H2 events) When $\gamma = \gamma^{\mathrm{H2}}_{ij}$ for some $1 \leqslant i, j \leqslant 2$.*
  For any H2 event (λ_1, λ_2), there is a directed line that is commonly tangent to the three polygons, as discussed above. For any $\theta \in [0, 2\pi)$, consider the directed line $\ell(\theta)$ oriented in θ and tangent to P_0 such that P_0 lies on the left side of $\ell(\theta)$. Then, there are exactly two different values of $\lambda_1 \in [0, L1)$ such that $P_1(\lambda_1)$ is tangent to $\ell(\theta)$ from the left of $\ell(\theta)$; either $P_1(\lambda_1)$ is ahead of P_0 along $\ell(\theta)$ or not. We let $g_1(\theta)$ and $g_2(\theta)$ be the values of λ_1 for the former and the latter cases, respectively. Analogously, we define $h_1(\theta)$ and $h_2(\theta)$ for $P_2(\lambda_2)$. We then observe that $(g_i(\theta), h_j(\theta)) \in \mathcal{K}_0$ is an H2 event for any $\theta \in [0, 2\pi)$ and any $1 \leqslant i, j \leqslant 2$. Moreover, we have $(g_i(\theta), h_j(\theta)) \in \gamma^{\mathrm{H2}}_{ij}$.

  Observe that g_i and h_j increase continuously as θ increases from 0, unless $g_i(\theta) = 0$ or $g_i(\theta) = L_1$, or $h_j(\theta) = 0$ or $h_j(\theta) = L_2$. The inverse functions of g_i and h_j are also well defined: there is a unique $\theta \in [0, 2\pi)$ such that $g_i(\theta) = \lambda_1$ for any fixed $\lambda_1 \in [0, L_1)$. This implies that each set $\gamma_{ij}^{\mathrm{H2}}$ is monotone in both axes, and is the graph of an increasing function f_{ij} from $[0, L_1)$ to $[0, L_2)$. Figure 6(a) illustrates the case of $(i, j) = (1, 2)$.

In order to see that f_{ij} is piecewise hyperbolic, consider the grid G on $\mathcal{K}_0$ generated by all C0 and H0 curves. Recall that any event curve of type C0 or H0 is an axis-parallel line in $\mathcal{K}_0$. In any cell σ of G that is intersected by γ, we have a constant contact set C and bridges. Since γ is monotone, $\gamma \cap \sigma$ is connected and thus it gives us an open interval $I = (\theta_1, \theta_2)$ such that $\gamma \cap \sigma = \{(g_i(\theta), h_j(\theta)) \mid \theta_1 < \theta < \theta_2\}$. For any $\theta \in I$, observe that $\ell(\theta)$ always passes through three fixed vertices v_0, v_1 and v_2 such that v_a belongs to P_a for $a = 0, 1, 2$. (See Figure 6(a) for an illustration.) Also, $\ell(\theta)$ rotates at v_0 as $\theta \in I$ increases. Each of v_1 and v_2 translates in a fixed direction as θ increases in the interval I. Observe that the direction of v_1 and v_2 is determined by the contacts C. This constrains the motion of P_1 and P_2 to be all linear along γ inside σ. A careful analysis using basic trigonometry[1] concludes that f_{ij} is hyperbolic in σ, that is, of the form $f_{ij}(\lambda_1) = 1/(c_1 + c_2\lambda_1) + c_3$ for some constants c_1, c_2, c_3, and f_{ij} is increasing unless γ reaches the boundary of $\mathcal{K}_0$ in σ. This shows the final case of the lemma.

This completes the proof. ◻

Each $\gamma \in \Gamma$ thus consists of one, two, or three curves unless it is axis-parallel. As shown in the proof of Lemma 3, the endpoints of γ occur when $\lambda_1 \in \{0, L_1\}$ or $\lambda_2 \in \{0, L_2\}$, except the endpoints of H1 event curves that lie on a C2 event curve. This discontinuity is because the configuration space $\mathcal{K}_0$ is indeed periodic; if we extend $P_1(\lambda_1)$ and $P_2(\lambda_2)$ for $\lambda_1 > L_1$ and $\lambda_2 > L_2$, then we have $P_1(\lambda_1 + L_1) = P_1(\lambda_1)$ and $P_2(\lambda_2 + L_2) = P_2(\lambda_2)$, and therefore γ becomes connected. We thus call each $\gamma \in \Gamma$ an *event curve* of type C0, C2, H0, H1, or H2 according to its type.

3.3 Complexity of event curves

We now discuss the complexity of event curves and of their arrangement $\mathcal{A}(\Gamma)$.

Lemma 4 *The family Γ consists of $O(n)$ event curves and the number of C1 events is bounded by $O(n)$. Also, each event curve in Γ consists of either $O(n)$ line segments or $O(n)$ hyperbolic segments.*

Proof. We take event curves of each type into account. Note that there are exactly two C2 event curves and four H2 event curves as defined above. Let n_0, n_1, and n_2 be the number of vertices of P_0, P_1, and P_2, respectively.

A C0 event curve $\gamma_{vv'}^{\mathrm{C0}}$ is associated with a pair of vertices (v, v'), where v is a vertex of P_0 v' is a vertex of P_1 or P_2. The number such pairs (v, v') with $\gamma_{vv'}^{\mathrm{C0}} \in \Gamma$ is exactly $n_0 + n_1$ if v' is

[1]Let $\lambda_2 = f_{ij}(\lambda_1)$. In Figure 6, let d_1 and d_2 be the Euclidean distance from v_0 to the lines supporting the motion of v_1 and v_2 locally. Then, the pair (λ_1, λ_2) can be parameterized as $(d_1 \tan(\theta + \alpha_1) + l_1, d_2 \tan(\theta + \alpha_2) + l_2)$, where c_1, c_2, l_1, l_2 are constants and θ represents the orientation of the line $\ell(\theta)$. Eliminating the parameter θ yields

$$\lambda_2 = d_2 \tan\left(\tan^{-1}\left(\frac{\lambda_1 - l_1}{d_1}\right) + \alpha_2 - \alpha_1\right) + l_2.$$

This equation is simplified as follows by applying the "addition formula for tangent": $\tan(u + v) = (\tan u + \tan v)/(1 - \tan u \tan v)$.

$$\left(\lambda_1 - l_1 - \frac{d_1}{\tan(\alpha_2 - \alpha_1)}\right)\left(\lambda_2 - l_2 + \frac{d_2}{\tan(\alpha_2 - \alpha_1)}\right) = d_1 d_2 \cdot \frac{\tan^2(\alpha_2 - \alpha_1) + 1}{\tan^2(\alpha_2 - \alpha_1)}.$$

This represents an equation of a hyperbolic curve of the form $\lambda_2 = f_{ij}(\lambda_1) = 1/(c_1 + c_2\lambda_1) + c_3$.

a vertex of P_1, and $n_0 + n_2$ if v' is a vertex of P_2 [9]. Thus, there are exactly $2n_0 + n_1 + n_2$ C0 event curves in Γ.

For any edge e of P_0, there are exactly two translates of P_1 such that P_1 is tangent to the supporting line of e, and thus a vertex of P_1 lies on the line. Equivalently, for any edge e' of P_1, there are exactly two translates of P_1 such that P_0 is tangent to the supporting line of e'. This implies that the number of H0 event curves defined by P_0 and P_1 is exactly $2n_0 + 2n_1$. Analogously, the number of H0 event curves defined by P_0 and P_2 is exactly $2n_0 + 2n_2$. On the other hand, the number of H1 event curves is exactly $8n_1 + 8n_2$ since we have four event curves for each such pair of a vertex and an edge. Hence, the number of H0 and H1 event curves is exactly $4n_0 + 10n_1 + 10n_2$.

For the number of C1 events, recall that each C1 event lies on a C2 event curve. Walking along a C2 event curve γ, we have a continuous motion of P_1 and P_2 where the three polygons are pairwise touching, and P_1 and P_2 slide around P_0 in the counter-clockwise direction. During this walk, observe that P_1 indeed slides around P_2. Hence, the number of C1 events along a C2 event curve is exactly $n_1 + n_2$. Since we have two C2 event curves, the number of C1 events in total is $2n_1 + 2n_2$.

To show the second statement, we bound the number of breakpoints on each event curve of type H1, H2, and C2. Recall that any C0 or H0 event curve is axis-parallel by Lemma 3. Consider an H1 event curve $\gamma \in \Gamma$. Let m be the number of C0 events on γ; that is, the number of C0 event curves intersected by γ. Lemma 3 implies that γ consists of m line segments. Again by Lemma 3, γ is monotone and therefore $m = O(n)$ since there are $O(n)$ C0 event curves that are axis-parallel. Similarly, any C2 event curve consists of $O(n)$ line segments and any H2 event curve consists of $O(n)$ hyperbolic segments. □

We now consider the arrangement $\mathcal{A}(\Gamma)$ of the event curves in Γ.

Lemma 5 *The complexity of the arrangement $\mathcal{A}(\Gamma)$ is $O(n^3)$, and each of its edges is either a line segment or a hyperbolic arc. More specifically, the number of crossings between any two event curves in Γ is $O(n)$.*

Proof. We first show that the number of crossings between any two event curves in Γ is bounded by $O(n)$, which implies that the combinatorial complexity of the arrangement $\mathcal{A}(\Gamma)$ is bounded by $O(n^3)$ since Γ consists of $O(n)$ event curves by Lemma 4. By Lemma 3, any C0 or H0 event curve is axis-parallel and any $\gamma \in \Gamma$ is monotone in both axes. Thus, any C0 or H0 event curve intersects any other event curve at most once.

Consider two event curves $\gamma_1, \gamma_2 \in \Gamma$ of type C2, H1, or H2. By Lemmas 3 and 4, each γ_i is monotone and has $O(n)$ breakpoints. And γ_i is either linear or hyperbolic on any interval of $[0, L_1)$ between two consecutive breakpoints of γ_i This implies that there are at most two values of λ_1 in each of the intervals such that $f_1(\lambda_1) = f_2(\lambda_1)$. Therefore, there are at most $O(n)$ crossings between γ_1 and γ_2. Since there are only $O(n)$ event curves of type C2, H1, or H2, there are at most $O(n^2)$ such combinations of (γ_1, γ_2). We thus have at most $O(n^3)$ crossings in this case. □

Note that the complexity of $\mathcal{A}(\Gamma)$ can be $\Omega(n^3)$ by a concrete construction of input polygons, so the bound of Lemma 5 is shown to be tight. Nonetheless, we prove a better bound if we focus on the feasible configurations, which imply no overlap between P_1 and P_2. We can easily see that the $O(n^3)$ complexity of $\mathcal{A}(\Gamma)$ is completely due to crossings among H1 event curves. By Lemma 3, any C0 or H0 curve crosses any other curve at most once. Since there are only six curves of type C2 and H2, the number of combinations (γ_1, γ_2) of any two curves of type C2, H1, or H2 but not both of H1 is $O(n)$, which implies that the total number of crossings between such combinations of curves is at most $O(n^2)$ by Lemma 5. Fortunately, the number of H1–H1 crossings that are feasible is shown to be much smaller.

Recall that the two C2 curves divide $\mathcal{K}_0$ into two regions, one consisting of all feasible configurations and the other of all infeasible configurations in $\mathcal{K}_0$. We denote by $\mathcal{F} \subset \mathcal{K}_0$ the former region. Since we want to find an optimal feasible configuration, we are mostly interested in the feasible region $\mathcal{F}$ and how it is decomposed.

Lemma 6 *Any two H1 event curves cross at most twice in $\mathcal{F}$. Therefore, the arrangement $\mathcal{A}(\Gamma)$ consists of $O(n^2)$ vertices and edges in $\mathcal{F}$.*

Proof. Consider two distinct H1 curves $\gamma_1, \gamma_2 \in \Gamma$. If one of the two is non-increasing and the other non-decreasing, then they cross at most once and we are done.

Without loss of generality, assume that both γ_1 and γ_2 are non-decreasing such that γ_1 is defined by a collinearity of a vertex v of P_1 and an edge e of P_2 and γ_2 is by a vertex v' and an edge e' whichever of P_1 and P_2 they belong to. Suppose that γ_1 crosses γ_2 at (λ_1, λ_2) in $\mathcal{F}$. Let ℓ be the line supporting e of $P_2(\lambda_2)$ and ℓ' be the line supporting e' at this configuration. Then, we have $v \in \ell$ and $v' \in \ell'$. Also, let d be the distance between v and the closer endpoint of e along ℓ. We then observe that for any crossing in $\gamma_1 \cap \gamma_2$ the distance between v and the closer endpoint of e must be exactly d. This can be seen by simple geometry: Imagine that P_1 moves along ℓ towards e of P_2 from infinity, and see the distance between the line supporting e' and the vertex v'. There is at most one instance where e' and v' are aligned, and the distance between v and the closer endpoint of e is exactly d at the moment.

Now, consider the location of P_1 and P_2 as above. Since $(\lambda_1, \lambda_2) \in \mathcal{F}$, they do not overlap each other. We then have at most two possible position of P_0 that touches both P_1 and P_2. This means that there are at most two such coordinates (λ_1, λ_2), and thus two H1 curves can cross at most twice in $\mathcal{F}$. Since there are $O(n)$ H1 curves in Γ, this suffices to show that the number of crossings in $\mathcal{F}$ among all H1 curves is $O(n^2)$. ◻

Figure 7 shows the arrangement $\mathcal{A}(\Gamma)$ of the event curves for the three input polygons depicted in Figure 1. Although we insist to decompose $\mathcal{K}_0$ into cells in each of which the contacts $C(\tau)$ and the bridges $H(\tau)$ stay constant, remark that some cells of $\mathcal{A}(\Gamma)$ are in fact not the case. For our purpose, however, it suffices to well decompose the feasible region $\mathcal{F}$, which imply no overlap between P_1 and P_2.

Lemma 7 *The arrangement $\mathcal{A}(\Gamma)$ of the event curves decomposes the feasible region $\mathcal{F} \subset \mathcal{K}_0$ into cells σ such that both $C(\tau)$ and $H(\tau)$ remain constant over all $\tau \in \sigma$.*

Recall that all configurations in $\mathcal{K}_0$ assume P_0 to keep contact with both P_1 and P_2. Alternating the role of P_0 by P_1 or P_2, we achieve a complete description of the configuration space $\mathcal{K}$. Letting $\mathcal{K}_1$ and $\mathcal{K}_2$ be the analogous configuration space for P_1 and P_2, respectively, we have $\mathcal{K} = \mathcal{K}_0 \cup \mathcal{K}_1 \cup \mathcal{K}_2$.

4 Algorithms

In this section, we present an algorithm that computes an optimal feasible configuration that minimizes the area or the perimeter of the convex hull of the three convex polygons under translation. The arrangement $\mathcal{A}(\Gamma)$ of the event curves is indeed sufficient to deal with the area or perimeter function in each feasible cell. Note that for any feasible configuration $\tau \in \mathcal{F}$, we have $2 \leqslant |H(\tau)| \leqslant 4$.

Lemma 8 *Let σ be any cell of $\mathcal{A}(\Gamma)$ with $\sigma \subset \mathcal{F}$. The area function is hyperbolic paraboloidal on σ if $|H(\tau)| = 3$ for $\tau \in \sigma$, or linear otherwise; the perimeter function is unimodal on σ and*

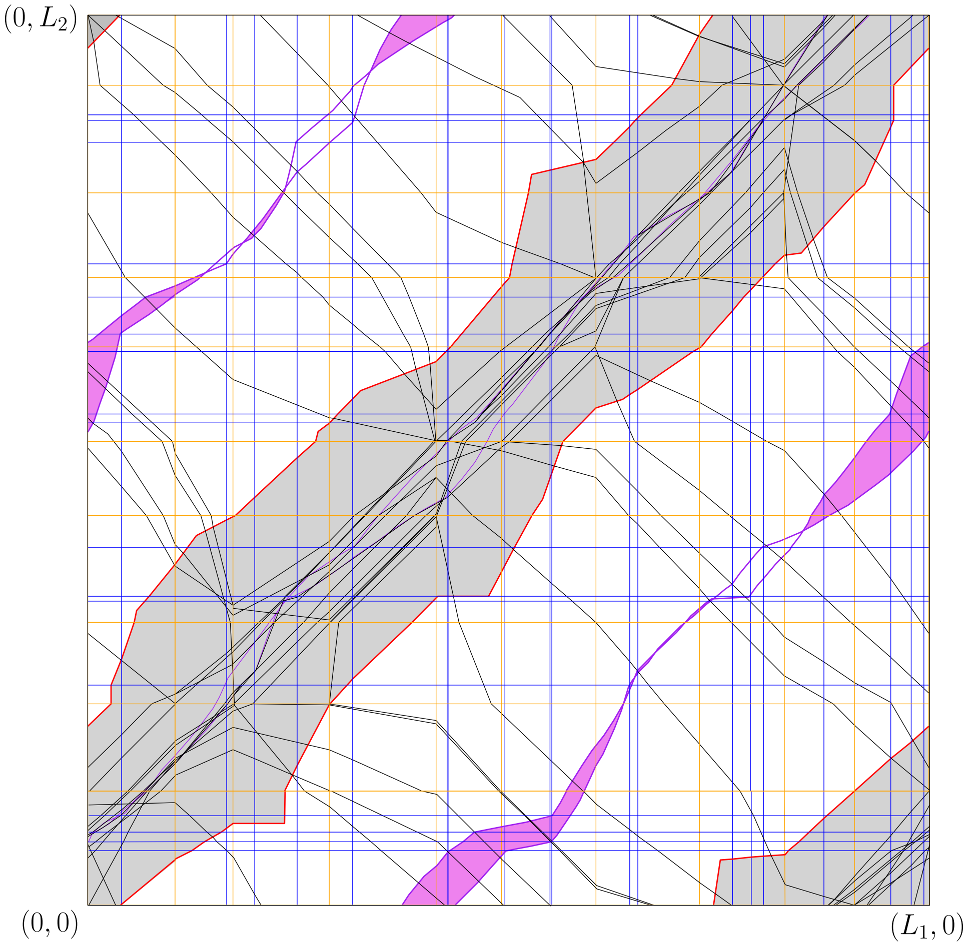

Figure 7: The arrangement $\mathcal{A}(\Gamma)$ of the event curves in the configuration space $\mathcal{K}_0$: The placements of $P_1(0)$ and $P_2(0)$ correspond to the left and right figures of Figure 1, respectively. Event curves of type C0 (blue), H0 (orange), H1 (black), C2 (red), and H2 (purple). Any configuration in the gray region is infeasible, so the feasible region $\mathcal{F}$ is the complement of the gray region. For any configuration τ in the purple region enclosed by H2 event curves, we have $|H(\tau)| = 4$.

on any edge incident to σ, and is of $O(1)$ descriptive complexity.

Proof. We analyze the area or perimeter function restricted in σ. By Lemma 7, the contacts $C = C(\tau)$ and the bridges $H = H(\tau)$ stay constant over $\tau \in \sigma$.

Each bridge in H is an edge of $\mathrm{conv}(\tau)$ between two vertices of the translated polygons. Let $(v_1, v_2), \ldots, (v_{2|H|-1}, v_{2|H|})$ denote the bridges in H, and x_i and y_i be the x- and y-coordinates of v_i in $\mathbb{R}^2$ under the translation by any $(\lambda_1, \lambda_2) \in \sigma$. Since the translations of $P_1(\lambda_1)$ and $P_2(\lambda_2)$ are linear inside σ, we can view both of x_i and y_i as functions linear on either λ_1 or λ_2.

We first consider the area function restricted in σ. Let f_σ be the function mapping $\tau \in \sigma$ to the area $\|\mathrm{conv}(\tau)\|$ of the convex hull of the translates by τ. Note that $2 \leqslant |H| \leqslant 4$ since $\sigma \subset \mathcal{F}$. We handle each of the three cases according the cardinality of H.

Case of $|H| = 2$. In this case, $\mathrm{conv}(\tau)$ contains one of the three translates in its interior by any $\tau \in \sigma$. This implies that only two of them appear on the boundary of $\mathrm{conv}(\tau)$ and they are connected by two bridges (v_1, v_2) and (v_3, v_4). We assume that v_1 and v_4 belong to one polygon and v_2 and v_3 to the other, as shown in Figure 8(a). Observe then that $x_4 - x_1$ and $y_4 - y_1$ are constant for any $(\lambda_1, \lambda_2) \in \sigma$ since v_1 and v_4 belong to the same polygon.

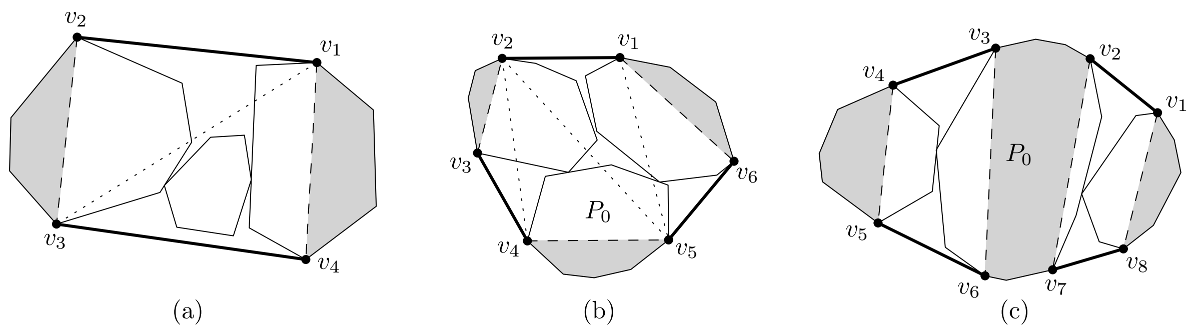

Figure 8: Illustration to the proof of Lemma 8 according to the number of bridges on a cell σ: (a) $|H| = 2$, (b) $|H| = 3$, (c) $|H| = 4$. The thick segments depict the bridges on the boundary of the corresponding convex hull and the gray regions are invariable components over all $\tau = (\lambda_1, \lambda_2) \in \sigma$.

Analogously, both $x_3 - x_2$ and $y_3 - y_2$ are constant. We let a_1, b_1, a_2, b_2 be those constants such that $a_1 = x_4 - x_1$, $b_1 = y_4 - y_1$, $a_2 = x_3 - x_2$, and $b_2 = y_3 - y_2$.
The convex hull $\mathrm{conv}(\tau)$ for $\tau \in \sigma$ consists of two invariable components and a quadrilateral $Q(\tau)$ formed by v_1, v_2, v_3, v_4. Thus, $f_\sigma(\tau)$ is described as the area of the quadrilateral $Q(\tau)$ plus a constant. We now focus on the area of the quadrilateral $Q(\tau)$. Since $Q(\tau)$ is the union of two triangles $\triangle v_1v_2v_3$ and $\triangle v_1v_3v_4$, its area $\|Q(\tau)\|$ is explicitly formulated as follows:

$$\begin{aligned}
\|Q(\tau)\| &= \frac{1}{2}\begin{vmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \end{vmatrix} + \frac{1}{2}\begin{vmatrix} x_3 - x_1 & y_3 - y_1 \\ x_4 - x_1 & y_4 - y_1 \end{vmatrix} \\
&= \frac{1}{2}\begin{vmatrix} x_2 - x_1 & y_2 - y_1 \\ x_2 - x_1 + a_2 & y_2 - y_1 + b_2 \end{vmatrix} + \frac{1}{2}\begin{vmatrix} x_2 - x_1 + a_2 & y_2 - y_1 + b_2 \\ a_1 & b_1 \end{vmatrix} \\
&= \frac{1}{2}|b_2(x_2 - x_1) - a_2(y_2 - y_1)| + \frac{1}{2}|b_1(x_2 - x_1 - a_2) - a_1(y_2 - y_1 - b_2)|,
\end{aligned}$$

where $|\cdot|$ denotes the determinant of a square matrix or the absolute value of a real number by abuse of its usage. Thus, the area function f_σ is shown to be linear to $\tau = (\lambda_1, \lambda_2) \in \sigma$ since all x_i and y_i are linear in either λ_1 or λ_2.

Case of $|H| = 3$. In this case, the convex hull $\mathrm{conv}(\tau)$ for any $\tau \in \sigma$ consists of three invariable components and the hexagon formed by the six vertices involved in H. Let $v_1, \ldots, v_6$ be the vertices involved in H indexed in the counter-clockwise order along the boundary of $\mathrm{conv}(\tau)$ in such a way that v_4 and v_5 belong to P_0 as shown in Figure 8(b). Thus, f_σ is represented by the area of the hexagon plus a constant. The hexagon is further divided into four triangles $\triangle v_2v_3v_4$, $\triangle v_1v_5v_6$, $\triangle v_2v_4v_5$, and $\triangle v_1v_2v_5$. Observe that the area of all triangles but $\triangle v_1v_2v_5$ is linear in σ since v_2 and v_3 belong to a common polygon, v_1 and v_6 belong to another common polygon, and v_4 and v_5 are stationary. The area of $\triangle v_1v_2v_5$ is represented as follows:

$$\|\triangle v_1v_2v_5\| = \frac{1}{2}\begin{vmatrix} x_1 - x_5 & y_1 - y_5 \\ x_2 - x_5 & y_2 - y_5 \end{vmatrix}.$$

Note that x_5 and y_5 are constant since v_5 is stationary. Plugging this into f_σ, hence, we obtain that $f_\sigma(\lambda_1, \lambda_2)$ is of the form $c_1\lambda_1\lambda_2 + c_2\lambda_1 + c_3\lambda_2 + c_4$ for some constants $c_1, \ldots, c_4$, which is reformulated into $c'_1(\lambda_1 + c'_2)(\lambda_2 + c'_3) + c'_4$. This equation indeed describes a transformed copy of a hyperbolic paraboloid.

Case of $|H| = 4$. Let $v_1, \ldots, v_8$ be the vertices involved in H indexed in the counter-clockwise order along the boundary of $\mathrm{conv}(\tau)$ in such a way that v_2, v_3, v_6, and v_7 belong to P_0, as shown in Figure 8(c). Note that P_0 must have four vertices in H in this case. Then, the convex hull $\mathrm{conv}(\tau)$ for $\tau \in \sigma$ consists of three invariable components and two quadrilaterals $Q_1(\tau)$ and $Q_2(\tau)$, formed by v_1, v_2, v_7, v_8 and by v_3, v_4, v_5, v_6, respectively. Thus, the function f_σ can be represented by $\|Q_1(\tau)\| + \|Q_2(\tau)\|$ plus a constant. Without loss of generality, we assume that v_1 and v_8 belong to P_1 while v_4 and v_5 belong to P_2. We then observe that $\|Q_i(\tau)\|$ is dependent only on λ_i for $i = 1, 2$ and, moreover, is linear in λ_i. Hence, we conclude that f_σ is linear on σ in this case.

Summarizing the above argument, we obtain the lemma about the area function.

We now turn to the perimeter function. Considering each case of $|H|$ as done above for the area function, the perimeter $|\mathrm{conv}(\tau)|$ of the convex hull $\mathrm{conv}(\tau)$ for $\tau = (\lambda_1, \lambda_2) \in \sigma$ is described as follows:

$$|\mathrm{conv}(\tau)| = c + \sum_{j=1}^{|H|} \sqrt{(x_{2j} - x_{2j-1})^2 + (y_{2j} - y_{2j-1})^2},$$

where c denotes a constant in $\mathbb{R}$ and x_i and y_i are the x- and y-coordinates of v_i as defined above. Since $|H| \leqslant 4$ and both x_i and y_i are linear in either λ_1 or λ_2, this simply shows that the perimeter function is of $O(1)$ complexity. In addition, the convexity, and thus the unimodality of the perimeter function directly follows from Lemma 1: Any cell σ of the arrangement $\mathcal{A}(\Gamma)$ with $\sigma \subset \mathcal{F}$ corresponds to a set of translation vectors $\tau \in \mathbb{R}^4$ that lie in a common 2-dimensional affine subspace of $\mathbb{R}^4$ since the configurations $\tau = (\lambda_1, \lambda_2) \in \sigma$ translate each of P_1 and P_2 along a line in $\mathbb{R}^2$. By the same argument, restricted on any edge e of $\mathcal{A}(\Gamma)$ with $e \subset \mathcal{F}$, the convexity of the perimeter function can be shown if e appears to be a line segment, since a line segment lying in the closure of a cell of $\mathcal{A}(\Gamma)$ corresponds to a line segment in $\mathbb{R}^4$.

What is left is the case where e is a hyperbolic arc. In this case, e is of type H2 and is described by an equation of the form $\lambda_2 = h(\lambda_1) = 1/(c_1 + c_2\lambda_1) + c_3$ for some constants c_1, c_2, and c_3 by Lemma 3 and its proof. As discussed above, the x and y-coordinates x_i and y_i of v_i are described as linear functions in λ_1 and λ_2. So, substituting λ_2 by $h(\lambda_1)$ in the above equation for $|\mathrm{conv}(\tau)|$ yields a function $g(\lambda_1)$ that maps λ_1 to the perimeter of the corresponding convex hull of the three polygons along the edge e. A careful analysis of the function g shows that its derivative has at most one real zero. We thus conclude that g has at most one local minimum except two limits of λ_1 corresponding to the endpoints of e. This implies the unimodality of the perimeter function along the hyperbolic edge e. $\square$

Our algorithm consists of two phases. First, it computes the arrangement $\mathcal{A}(\Gamma)$ in the feasible region $\mathcal{F}$ only, and second it traverses each cell of the arrangement and computes an optimal translation for the area or the perimeter function restricted to the cell.

By Lemma 8, the second phase is relatively easy once the cells and the edges of $\mathcal{A}(\Gamma)$ lying in $\mathcal{F}$ are fully specified. At this phase, we visit every cell in $\mathcal{F}$ by crossing over an incident edge and thus moving to a neighboring cell. Then, by coherence, the description of the objective function restricted in the next cell can be obtained in constant time. Lemma 8 guarantees that the area or the perimeter function can be minimized in $O(1)$ time in a cell or on each of its bounding edges. Hence, the total time complexity of the second phase is bounded by $O(n^2)$ time by Lemma 6.

The arrangement can be easily computed in $O(n^2 \log n)$ time by a typical plane-sweep algorithm. In the following, we focus on improving the time bound to $O(n^2)$ for the task.

4.1 Computing the arrangement $\mathcal{A}(\Gamma)$ in $\mathcal{F}$

In order to compute the arrangement $\mathcal{A}(\Gamma)$, we first compute all the event curves in Γ with full description, and then identify all the intersections among them that lie in $\mathcal{F}$.

Preprocessing. As a preprocessing, we take any two polygons P_i and P_j for $0 \leqslant i < j \leqslant 2$ and move P_j around P_i keeping a contact to P_i in the counterclockwise direction. During this motion, we gather all occurrences of vertex-vertex contact in order and store them into a sorted list C_{ij} with the corresponding pairs of vertices. In addition, we maintain two external common tangents of P_i and P_j and gather all occurrences at which one of the two tangents supports an edge of P_i or P_j. We also store them into a sorted list H_{ij} with the corresponding pairs of vertex and edge. Let us make each of C_{ij} and H_{ij} to be a circular list for later use. This preprocessing can be handled in $O(n)$ time as done in [9]. Observe that each member of C_{01} and C_{02} describes a C0 event curve in $\mathcal{K}_0$, and each of H_{01} and H_{02} describes an H0 event curve. We thus find all C0 and H0 event curves by traversing these lists.

Computing the event curves. Let G be the grid on $\mathcal{K}_0$ induced by all the C0 and H0 event curves. The other event curves can be obtained by tracing each across the grid cells of G. Consider the four H2 event curves. By Lemma 3, each H2 event curve γ appears to be a hyperbolic segment in each grid cell σ intersected by itself, and the equation of each segment in σ can be described with help of the lists C_{ij} and H_{ij}. We first locate its starting point at $\lambda_1 = 0$ in $O(n)$ time from the lists C_{ij} and H_{ij}, and then we trace γ cell by cell. As we walk along γ and move to the neighboring grid cell σ', we immediately tell the change of the contacts or the bridges from the lists C_{ij} and H_{ij} so that the equation of γ in σ' can be updated in $O(1)$ time. Hence, tracing γ spends time proportional to the number of grid cells of G that are intersected by γ. Lemma 4 tells us that the number of such grid cells, and thus the cost of tracing an H2 curve is $O(n)$.

The other event curves of different types can be traced in the same fashion, taking $O(n)$ time for each. While tracing a C2 event curve, we can also specify all C1 events: this can be done by looking up the list C_{12} with a pointer that indicates the current contact between P_1 and P_2. Tracing an H1 event curve needs to look up the list H_{12}; in fact, only the members of H_{12} can determine an H1 event curve. We hence can compute all the event curves in Γ with their full description in $O(n^2)$ time.

Specifying all necessary crossings. We then compute the arrangement $\mathcal{A}(\Gamma)$ in $\mathcal{F}$ by specifying all necessary crossings among the event curves in Γ.

Note that for any two event curves $\gamma_1, \gamma_2 \in \Gamma$, all the crossings between them can be computed in $O(n)$ time by Lemmas 3 and 5. For all pairs (γ_1, γ_2) of event curves such that γ_1 is of type C2 or H2 and γ_2 is of type C2, H1, or H2, we are thus able to specify all the crossings between γ_1 and γ_2 in $O(n^2)$ time, since the number of such pairs (γ_1, γ_2) is $O(n)$. What remains is to specify the crossings among the H1 event curves.

For the last task, we take only feasible portions of every event curve into account. Let $\Gamma_{\mathcal{F}} := \{\gamma \cap \mathcal{F} \mid \gamma \in \Gamma\}$. Computing $\Gamma_{\mathcal{F}}$ can be done by cutting each $\gamma \in \Gamma$ by the C2 curves and discarding its infeasible portions. Each feasible portion γ' of an event curve γ inherits its type from γ. Fortunately, this cutting does not increase the number of curves much, especially, H1 event curves.

Lemma 9 *The number of H1 event curves in* $\Gamma_{\mathcal{F}}$ *is* $O(n)$.

Proof. We show that any H1 event curve in Γ crosses a C2 event curve at most twice, which directly implies the lemma.

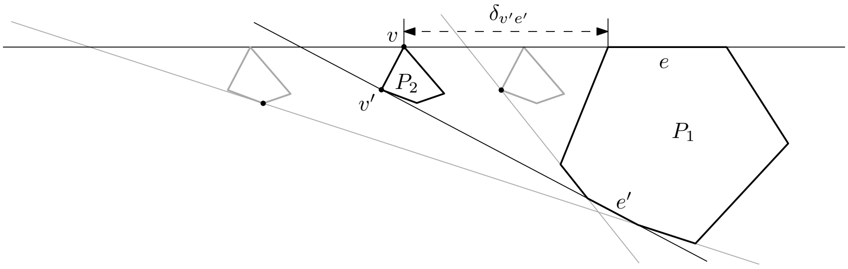

Figure 9: Illustration to the list Δ.

Let $\gamma \in \Gamma$ be an H1 event curve defined by (v, e). For any configuration $(\lambda_1, \lambda_2) \in \gamma$, $P_1(\lambda_1)$ and $P_2(\lambda_2)$ have a common external tangent that supports e. If it is also a C2 event, in addition $P_1(\lambda_1)$ and $P_2(\lambda_2)$ must touch each other. In such a scene, P_0 must touch both $P_1(\lambda_1)$ and $P_2(\lambda_2)$; there are at most two such possibilities, implying at most two coordinates $(\lambda_1, \lambda_2) \in \gamma$. $\square$

Lemma 6 implies that the number of crossings in $\mathcal{F}$ between a fixed H1 curve γ and all the other H1 curves is $O(n)$. In the following, we show that all such crossings can be specified in $O(n)$ time. By Lemma 9, it suffices to conclude the total $O(n^2)$ time.

For the purpose, we need some more observations. Let $\gamma \in \Gamma_{\mathcal{F}}$ be an H1 event curve defined by a pair $(v, e) \in H_{12}$ of vertex v and edge e of P_1 and P_2. For any configuration $(\lambda_1, \lambda_2) \in \gamma$, $P_1(\lambda_1)$ and $P_2(\lambda_2)$ have a common external tangent that supports e. Let g be a function partially defined on $[0, L_1)$ whose graph is γ (see the proof of Lemma 3.) Define $d_\gamma(\lambda_1)$ to be the distance between v and the endpoint of e that are the closer to v in the corresponding translates $P_1(\lambda_1)$ and $P_2(g(\lambda_1))$. Observe that d_γ is linear in a grid cell σ of G since the translations of P_1 and P_2 are linear in σ along γ.

On the other hand, we consider the other external common tangent $\ell(\lambda_1)$ of $P_1(\lambda_1)$ and $P_2(g(\lambda_1))$. When an edge e' of P_1 or P_2 lies on $\ell(\lambda_1)$, we have a crossing between γ and another H1 event curve γ' defined by (v', e') for the vertex v' lying on $\ell(\lambda_1)$; we let $\delta_{v'e'} := d_\gamma(\lambda_1)$ at such a value of λ_1. By a geometric observation, at such a crossing, $P_1(\lambda_1)$ and $P_2(g(\lambda_1))$ have two external common tangents, one supporting e and the other supporting e'; this fixes a unique value of $d_\gamma(\lambda_1)$ to be $\delta_{v'e'}$. This implies that for any λ_1, γ' crosses γ at $(\lambda_1, g(\lambda_1))$ if and only if $d_\gamma(\lambda_1) = \delta_{v'e'}$. See Figure 9 for an illustration.

We thus perform the following procedure. We compute the value $\delta_{v'e'}$ for all $(v', e') \in H_{12}$ and store them into a sorted array Δ with corresponding label (v', e'). Then, we walk along γ cell by cell to find all occurrences such that $d_\gamma(\lambda_1) = \delta$ for some $\delta \in \Delta$. This completely specifies all crossings between γ and all other H1 event curves in $\Gamma_{\mathcal{F}}$. To compute Δ, initially make P_1 and P_2 touch each other, keeping a common tangent going through v and e, and consider the other common tangent line ℓ. If we move P_2 in the direction parallel to e and away from P_1, then the tangent line ℓ will rotate monotonously in one direction. This implies that the order of $\delta_{v'e'}$ follows from the order of (v', e') in the list H_{12}. Thus, we can compute Δ in $O(n)$ time.

Let $\delta_1, \ldots, \delta_m$ be the members of Δ in the order. Once we compute Δ, we walk along γ by increasing λ_1 to find all λ_1 such that $d_\gamma(\lambda_1) = \delta$ holds for some $\delta \in \Delta$. Since d_γ is linear in each cell σ of G intersected by γ, the task is not difficult if we maintain a variable a such that $\delta_a \leqslant d_\gamma(\lambda_1) < \delta_{a+1}$ for the current value of λ_1. Hence, we can find all crossings on $\gamma \cap \sigma$ with the other H1 event curves in time $O(1 + c)$, where c is the number of the reported crossings in σ. If we sum up this over all grid cells intersected by γ, we obtain $O(n)$ time bound.

Putting it all together, we can specify all intersections among curves in $\Gamma_{\mathcal{F}}$ in $O(n^2)$ time, which are the vertices of $\mathcal{A}(\Gamma_{\mathcal{F}})$. We then cut each curve $\gamma \in \Gamma_{\mathcal{F}}$ by the crossings on γ to obtain the edges of $\mathcal{A}(\Gamma_{\mathcal{F}})$. As a result, we can build the underlying graph of the arrangement $\mathcal{A}(\Gamma_{\mathcal{F}})$, and then the arrangement $\mathcal{A}(\Gamma_{\mathcal{F}})$ can be built in the same time bound $O(n^2)$.

We finally conclude our main result.

Theorem 1 *Given three convex polygons P_0, P_1, and P_2 having a total of n vertices, one can find in $O(n^2)$ time using $O(n^2)$ space an optimal pair (τ_1, τ_2) of translation vectors such that the interiors of P_0, $\tau_1 P_1$ and $\tau_2 P_2$ are disjoint, and the area $\|\mathrm{conv}(P_0 \cup \tau_1 P_1 \cup \tau_2 P_2)\|$ or the perimeter $|\mathrm{conv}(P_0 \cup \tau_1 P_1 \cup \tau_2 P_2)|$ is minimized.*

5 Conclusion

We study the bundling problem for three convex polygons in the plane and present an efficient algorithm for the problem with quadratic running time and space. We believe that our approach in this paper can naturally be extended to the cases of $k > 3$ convex polygons. It would be interesting to investigate their combinatorial and algorithmic complexity for small k. One can also consider the bundling problem in higher dimensions.

Another direction of future study would be to study the case of non-convex polygons. It is well known that for two polygons, their Minkowski sum [5] (or Minkowski difference) provides the translation space of one polygon over the other. It is, however, unclear how to extend the idea for the case of more than two polygons.

References

[1] H.-K. Ahn and O. Cheong. Stacking and bundling two convex polygons. In *Proc. 16th Internat. Sympos. Algo. Comput. (ISAAC 2005)*, volume 3827 of *LNCS*, pages 40–49, 2005.

[2] H.-K. Ahn and O. Cheong. Aligning two convex figures to minimize area or perimeter. *Algorithmica*, 62:464–479, 2012.

[3] H. Alt and F. Hurtado. Packing convex polygons into rectangular boxes. In *Proc. 3rd Japanese Conf. Discrete Comput. Geom. (JCDCG 2000)*, volume 2098 of *LNCS*, pages 67–80, 2001.

[4] K. Daniels and V. Milenkovic. Multiple translational containment, part i: An approximation algorithm. *Algorithmica*, 19:148–182, 1997.

[5] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars. *Comput. Geom.: Algo. and Appl.* Springer-Verlag, 3rd edition, 2008.

[6] J. Egeblad, B. Nielsen, and M. Brazil. Translational packing of arbitrary polytopes. *Comput. Geom.: Theory and Appl.*, 42:269–288, 2009.

[7] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H. Freeman and Co., San Francisco, CA, 1979.

[8] J. Kepler. *Vom sechseckigen Schnee*, volume 273 of *Ostwalds Klassiker der Exakten Wissenschaften.* Akademische Verlagsgesellschaft Geest & Portig K.-G., Leipzig, 1987. Strena seu de Nive sexangula, Translated from the Latin and with an introduction and notes by Dorothea Goetz.

[9] H. Lee and T. Woo. Determining in linear time the minimum area convex hull of two polygons. *IIE Trans.*, 20:338–345, 1988.

[10] V. Milenkovic. Translational polygon containment and minimum enclosure using linear programming based restriction. In *Proc. 28th Annu. ACM Sympos. Theory Comput. (STOC 1996)*, pages 109–118, 1996.

[11] K. Sugihara, M. Sawai, H. Sano, D.-S. Kim, and D. Kim. Disk packing for the estimation of the size of a wire bundle. *Japan J. Industrial and Applied Math.*, 21:259–278, 2004.

[12] K. Tang, C. Wang, and D. Chen. Minimum area convex packing of two convex polygons. *Internat. J. Comput. Geom. Appl.*, 16:41–74, 2006.