12-1-2007

**Vertex Pops and Popturns**

Greg Aloupis  
*Université Libre de Bruxelles*

Brad Ballinger  
*Davis School for Independent Study*

Prosenjit Bose  
*Carleton University*

Mirela Damian  
*Villanova University*

Erik D. Demaine  
*MIT Computer Science & Artificial Intelligence Laboratory*

See next page for additional authors

Follow this and additional works at: https://scholarworks.smith.edu/csc_facpubs

Part of the Computer Sciences Commons

**Recommended Citation**

Aloupis, Greg; Ballinger, Brad; Bose, Prosenjit; Damian, Mirela; Demaine, Erik D.; Demaine, Martin L.; Flatland, Robin; Hurtado, Ferran; Langerman, Stefan; O'Rourke, Joseph; Taslakian, Perouz; and Toussaint, Godfried, "Vertex Pops and Popturns" (2007). Computer Science: Faculty Publications, Smith College, Northampton, MA.  
https://scholarworks.smith.edu/csc_facpubs/195

This Conference Proceeding has been accepted for inclusion in Computer Science: Faculty Publications by an authorized administrator of Smith ScholarWorks. For more information, please contact scholarworks@smith.edu
1 Introduction

This paper considers transformations of a planar polygon $P$ according to two types of operations. A vertex pop (or a pop) reflects a vertex $v_i$, $i \in \{1, \ldots, n\}$, across the line through the two adjacent vertices $v_{i-1}$ and $v_{i+1}$ (where index arithmetic is modulo $n$). A popturn rotates $v_i$ in the plane by $180^\circ$ about the midpoint of the line segment $v_{i-1}v_{i+1}$. Pops and popturns are polygon reconfiguration moves similar to “Erdős pocket flips” and “flipturns” [5, 3] in that they preserve the lengths of the polygon edges.

Our goal in this paper is to study which polygons can be convexified by a series of pops or popturns, under various intersection restrictions and definitional variants.

We distinguish between three types of polygons. A simple polygon is non-self-intersecting, in that edges intersect only at common endpoints. A polygon is weakly simple if its boundary does not “properly cross” itself. Finally, a general polygon may be self-intersecting with proper crossings. Pops and popturns can easily introduce weak or proper crossings, so the latter two classes are often more natural to study.

We also focus on two subclasses of polygons. In an orthogonal polygon, adjacent edges meet at right angles. In an equilateral or unit polygon, all edge lengths are equal, say, to 1. In unit polygons, pops and popturns become identical operations.

We will see that a vertex pop can create a hairpin vertex (or a pin): a vertex $v_i$ whose incident edges overlap collinearly. If also $v_{i-1}=v_{i+1}$ (which arises naturally in unit polygons), then the reflection line for a pop of $v_i$ is not determined. Whether to allow a pop of such a pin, and if so, how to define it, leads to many interesting variations, detailed in Section 3 below.

Table 1: Summary of our results.

<table>
<thead>
<tr>
<th>Polygons</th>
<th>Moves</th>
<th>Convexifiable?</th>
</tr>
</thead>
<tbody>
<tr>
<td>arbitrary</td>
<td>popturns</td>
<td>yes, always</td>
</tr>
<tr>
<td>simple</td>
<td>popturns</td>
<td>yes iff no purse</td>
</tr>
<tr>
<td>weakly simple, unit, orthogonal</td>
<td>pops+180° rot. or pops+untwists</td>
<td>yes, always</td>
</tr>
</tbody>
</table>

**Our results.** Table 1 lists our results. If crossings are permitted, it remains unresolved whether every polygon can be convexified via vertex pops, but we show that popturns suffice. Restricting to simple polygons, it is known that every star-shaped polygon can be convexified by popturns [1, Thm. 3.2]. We characterize precisely the class of polygons that can be convexified by simple popturns: those without a “purse.” Our final result is specialized to unit orthogonal polygons, which can be reconfigured under various hairpin move restrictions.

2 Popturns

The polygon $P$ with clockwise vertices $(v_1, v_2, \ldots, v_n)$ can be seen as a cyclic sequence of rooted vectors $(e_1, \ldots, e_n)$, where $e_i = (v_{i-1}, v_i)$. A sequence of vectors is simple if they form a simple polygon, and clockwise if each vector has the interior of $P$ on its right side. In the following, we will use the terms sequence of vectors and polygon interchangeably. A popturn then corresponds to swapping two adjacent vectors in their cyclic ordering. We call the popturn crossing-free if the resulting polygon is simple. The two vectors to be popturned and their images form a parallelogram. We call the popturn simple if it is crossing-free and this parallelogram does not contain $P$. This is the case if and only if the resulting polygon is simple and clockwise. A crossing-free popturn can turn a polygon “inside-out” whereas the more restrictive simple popturn cannot.

If we permit crossings, popturns can convexify by simulating bubble sort on edge directions, where each adjacent swap corresponds to a popturn; see [2, p. 32].

**Theorem 1** Any polygon of $n$ vertices can be convexified (permitting crossings) by a sequence of at most $\frac{1}{2} \binom{n}{2}$ popturns.
In the remainder of this section, we concentrate on simple polygons and simple popturns. The turning angle $\tau_i = \tau_{v_i}$ at vertex $v_i$ is the clockwise angle between the vectors $e_i$ and $e_{i+1}$ ($-\pi < \tau_i < \pi$), and the total turning angle $\tau_{i,j} = \tau_{e_i,e_j}$ between edges $e_i$ and $e_j$ is $\sum_{l=1}^{j-1} \tau_l$. Because the polygon is closed, simple, and clockwise, $\tau_{i,j} + \tau_{j,i} = 2\pi$. Notice that the total turning angle $\tau_{e_i,e_j}$ between two edges, $e_i$ and $e_j$ does not change after a simple popturn unless $e_i$ and $e_j$ are adjacent, i.e., $j = i + 1$ or $j = i - 1$, and the popturn is performed at their common vertex. Consider, for example, Fig. 1(a), in which a popturn at $v_3$ reorders the sequence of vectors $\{\ldots, e_1, e_2, e_3, e_4, \ldots\}$ to $\{\ldots, e_1, e_3, e_2, e_4, \ldots\}$. Only two edge-turning angles change: $\tau_{e_2,e_3} = \frac{1}{4}\pi$ becomes $\tau'_{e_2,e_3} = (2 + \frac{1}{4})\pi$, wrapping around the entire polygon; meanwhile, $\tau_{e_3,e_2} = \frac{1}{3}\pi$ becomes $\tau'_{e_3,e_2} = -\frac{1}{4}\pi$.

Figure 1: (a) Popturn at $v_3$ of $\{e_2,e_3\}$. (b) Purse $e_1,\ldots,e_j$.

A purse is a (cyclic) subsequence $e_i,\ldots,e_j$ such that $\tau_{i,j} \leq -\pi$; see Fig. 1(b). We show the following:

**Lemma 2** If $e_i,\ldots,e_j$ is a purse, then $e_i$ and $e_j$ can never be made adjacent by any sequence of simple popturns; and so $\tau_{i,j}$ is constant.

**Proof:** As stated previously, $\tau_{e_i,e_j}$ will be affected by a popturn only if $e_i$ and $e_j$ are adjacent, i.e., $j = i + 1$ or $j = i - 1$. In the first case, $\tau_{i,j} = \tau_{i,i+1}$ must be strictly between $-\pi$ and $\pi$. In the second case, $\tau_{i,j} = \tau_{j+1,j} = 2\pi - \tau_{j+1,j+1}$, which is strictly between $\pi$ and $3\pi$. However, purse $e_i,\ldots,e_j$ has $\tau_{i,j} \leq -\pi$, meeting neither case. Before $e_i$ and $e_j$ become adjacent, $\tau_{i,j}$ must change, but before $\tau_{i,j}$ can change, $e_i$ and $e_j$ must become adjacent. Thus $e_i$ and $e_j$ will never become adjacent.

A vertex $v_i$ is reflex if $\tau_i < 0$. A popturn at a reflex vertex is called a reflex popturn.

**Lemma 3** Given a simple clockwise polygon, if the popturn at a reflex vertex $v_i$ is not simple, then the polygon has a purse.

**Proof:** Let $v'_i$ be the position of $v_i$ after the popturn. If the popturn at $v_i$ is not simple, then the parallelogram $v_{i-1}v_iv_{i+1}v'_i$ intersects $P$. Suppose that $P$ has no purse. It follows that edge $e_{i+2}$ is outside of the parallelogram. Then by the Jordan curve theorem, there is a proper intersection between the boundary of $P$ and $v_{i+1}v'_i$ or $v'_iv_{i-1}$. Assume by symmetry that there is such an intersection on the edge $v_{i+1}v'_i$ and let $q$ be the first proper intersection encountered while walking from $v_{i+1}$ to $v'_i$; see Fig. 2.

![Figure 2: Proof of Lem. 3.](image)

Let $P'$ be a counterclockwise polygon formed by taking the portion of $P$ between $v_{i+1}$ and $q$ and a vector $e'$ from $q$ to $v_{i+1}$. Let $e^*$ be the vector preceding $e'$ in $P'$. The polygon $P'$ is closed, simple, and counterclockwise. Thus the total turning angle $\tau_{e',e^*} + \tau_{e^*,e'} = -2\pi$. But the vector $e^*$ is part of a vector of $P$, say $e_j$, and $e_j$ is parallel to $e'$; thus, $\tau_{i,j} = \tau_{e',e^*}$. Finally, $e^*$ and $e'$ are adjacent, so $\tau_{e^*,e'}$ must be strictly between $-\pi$ and $\pi$, and $\tau_{i,j} = \tau_{e',e^*} = -\tau_{e^*,e'} - 2\pi < -\pi$. Thus $e_i,\ldots,e_j$ is a purse. \qed

**Theorem 4** A simple polygon $P$ can be convexified by a finite sequence of simple popturns if and only if $P$ contains no purse.

**Proof:** If $P$ contains a purse $e_i,\ldots,e_j$, then by definition, $\tau_{i,j} < -\pi$ and by Lem. 2, $e_i$ and $e_j$ will never become adjacent, which implies that the value of $\tau_{i,j}$ will remain the same after any sequence of simple popturns. In a clockwise convex polygon, the total turning angle between every pair of edges is non-negative. This implies $P$ can never become convex after any sequence of simple popturns.

Note that applying any sequence of popturns to $P$ will result in a polygon which is a permutation of the original vectors. If $P$ contains no purse, then by Lem. 3, the popturn at any reflex vertex is simple. Such a popturn will increase the area of the polygon, so the same permutation of vectors will never be repeated. Since the number of different permutations of vectors is finite, any sequence of reflex popturns will have to be finite as well. At the end of such a maximal sequence, no reflex vertex remains and the polygon is convex. \qed

Now we can more precisely bound the number of popturns needed to convexify a polygon:
Lemma 5 Let P be a polygon that has no purse. Any maximal sequence of reflex popturns will convexify an n-gon P after exactly $\sum_{i,j} |a_{i,j} < 0| \leq \binom{n}{2}$ popturns.

The situation is significantly more complex in the case of crossing-free popturns. In the full version we prove:

Theorem 6 Deciding if a polygon can be convexified by a sequence of crossing-free popturns is NP-Hard.

3 Unit Orthogonal Polygons

When restricted to simple pops, even the 12-vertex polygon in Fig. 3(a) cannot be convexified. Here we loosen that restriction and allow hairpin vertices. A hairpin vertex $v_i$ in a unit polygon has $v_{i-1} = v_{i+1}$, which leaves a pop of $v_i$ undefined. We feel it is natural to define the

![Figure 3: A unit polygon that cannot be convexified by pure pops.](image)

pop of a pin $v_i$ when $v_{i-1} = v_{i+1}$ as the reflection across the line $L$ perpendicular to the pin edges and through their common endpoint. This permits convexifying the previous example; see Fig. 3(b-d). Through an extension of the argument in Thm. 4, we can show that pops together with pin pops still do not suffice to convexify all unit polygons while remaining weakly simple. But rather than detail this argument, we turn instead to positive results.

3.1 Pin-move Extensions

There are three natural pin-move extensions: rotating a pin 90°, rotating 180°, or “untwisting” a pin. The first is related to the work of Dumitrescu and Pach [4], in that their “coin moves” can be simulated in certain contexts with the help of 90° pin rotations. However, we do not pursue this connection, and only observe that 90° rotations are subsumed by 180° rotations. We next show that the second two pin movements, and therefore the first, permit convexifying any unit orthogonal polygon while remaining weakly simple.

Let $P$ be a unit orthogonal polygon. We define a U-shaped boundary piece $(v_i, v_{i+1}, ..., v_j)$ to be a cup if $v_{i+1}$ and $v_{j-1}$ are both reflex or both convex and $v_{i+1}, v_{j-1}$ are collinear. The line segment $v_i v_j$ is the cup lid. A cup is open if no piece of $\partial P$ lies along its lid. A horizontal cup (or H-cup) is an upright or upside down U-shape; a vertical cup (or V-cup) is a U-shape on its side.

Our reconfiguration algorithm converts $P$ to a canonical form by moving pins around $\partial P$. If $[v_{i-1}, v_i, v_{i+1}]$ is a pin, call $v_i$ its tip and $v_{i-1} = v_{i+1}$ its base. We distinguish two types of pins. A flat pin has the tip vertex $v_i$ coincident with either $v_{i-2}$ or $v_{i+2}$; see Figs. 4(a,b). A barb pin has a tip vertex $v_i$ distinct from both $v_{i-2}$ and $v_{i+2}$; see Fig. 4(c,d).

![Figure 4: (a,b) Flat pins (c,d) Barb pins.](image)

Now we relax the condition that pops preserve simplicity of the polygon, and allow for simple pin “twists” in a small neighborhood around their base point. A twisted pin (e.g. Fig. 5(b)) is the result of pop($v_i$) applied in the following two conditions: (i) $[v_{i-1}, v_i, v_{i+1}]$ is a simple (un twisted) flat pin, and (ii) $v_i v_{i+1}$ and $v_{i+1} v_{i+2}$ are orthogonal (cf. Figs. 4(b), 5(a)). Once a pin becomes twisted, we immediately untwist it (cf. Fig. 5(c)). Note that our pop operations apply on the simple polygon obtained by separating the pin base into two points within an epsilon-disk of the base, as illustrated in the pin drawings. Although pin untwisting may seem like “cheating,” in fact the operation is quite natural, for the coincidence of $v_{i-1}$ with $v_{i+1}$ means that Fig. 5(b,c) are geometrically identical. Although

![Figure 5: (a) Initial pin $[v_{i-1}, v_i, v_{i+1}]$ (b) Pin twisted after pop($v_i$) (c) Untwisted pin.](image)

it may appear from Fig. 5(a,c) that the result of popping/untwisting a pin is the same as rotating the pin 180° about its reflex base point, the pop and the 180°-turn operations are not always identical. Nevertheless, we show that they are equivalent in the sense that the composite operations (see Sec 3.3) used by our algorithm can be defined in terms of either pop/untwist pin operations or pop/180°-turn pin rotations.

3.2 Canonical Form

Let $P$ be a polygon with $2x$ horizontal edges and $2y$ vertical edges. The canonical form of $P$ is a rectangle of length $x$ and height $y$. It is used as an intermediate stage in reconfiguring $P$ into another polygon with a same number of horizontal and vertical edges.
3.3 Composite Operations

We define three composite operations used by the reconfiguration algorithm. Each can be implemented using pop/untwist operations or pop/180°-turn operations.

**SLIDE(II):** Moves the pin II one lattice edge cw around the boundary. See Fig. 6.

**WALK(II, c):** Applies a sequence of SLIDE operations to walk the pin II cw along ∂P until its base coincides with corner point c.

**POP-SWEEP(II, c):** Here II is an outward pointing barb pin whose base vertex b is connected to vertex c by a straight boundary segment. This operation pops all vertices on the boundary segment, starting with b.

3.4 Converting P To Canonical Form

Let T = ℓr be the leftmost among the topmost maximal horizontal sections of ∂P, with ℓ (r) the left (right) endpoint of T. The algorithm uses the composite operations to convert P into a canonical rectangle R that has its lower-right corner at r (see Fig. 7(d)). Initially, R is degenerate and coincides with line segment [ℓ, r].

The algorithm repeatedly creates a pin II and walks it around ∂P to the top left corner tℓ of R (initially tℓ = ℓ), where it uses a POP-SWEEP operation to expand II into a new (top) row or (left) column of R. A pin is created by popping all base vertices of an open cup, which always exists (Lem. 7). E.g., in Fig. 7(d), popping base vertex b2 of cup (a2, b2, c2, d2) creates a pin (see Fig. 7(e)). The cup must be open, for otherwise popping the base vertices results in ∂P touching along non-pin edges. In the first iteration, the algorithm uses a pin II corresponding to an upright open H-cup; this ensures that, once it reaches ℓ, II expands into a row extending from ℓ to r, turning R into a one-row rectangle (ℓ = bℓ, tℓ, r). Figs. 7(a-g) show this for two pins.

**Lemma 7** If P is not a rectangle, it has at least one open cup in the halfplane H bounded above by T.

Figure 6: (a, b) Sliding a barb pin (c, d, e) Sliding a flat pin.

Figure 7: (a) P (b) H-cup (a1, b1, c1, d1) turned into pin II by pop(b1) (c) Pin II after WALK (d) Rectangle R after POP-SWEEP (e-g) Same steps for next pin.

**Proof:** If P is not a rectangle, then the reconfiguration is not complete and some part of ∂P lies in the interior of H. Therefore, P has at least one upright horizontal cup in H, namely the H-cup with a lowest horizontal edge as base. Of all upright horizontal cups in H, let C = (v1, v1+1, ..., vj−1, vj) be one with a highest base. Assume for the sake of contradiction that C is not open. Then its lid contains some (maximal) horizontal section u1, ..., uk+s of ∂P. Assume w.l.o.g. that k ≥ j. If vk and vj coincide, then (vj−1, vj−2, ..., vj−1, vj = vj, vj+1) is an open V-cup in H and the proof is finished, and similarly if vj+k and vi coincide. So assume k ≠ j and k + s ≠ i. Then v1, ..., vj+k must be the base of an upright H-cup, call it D. Simple arguments show that D lies in H and is higher than C, a contradiction. □

**Theorem 8** The described algorithm transforms P into a rectangle in Θ(n²) pop operations.

**Acknowledgments.** We thank the other participants of the 2007 Workshop on Reconfiguration at the Bellairs Research Institute of McGill University.

**References**


