Journal for Geometry and Graphics
Volume 19 (2015), No. 2, 283–300.

Point in Polygon Problem via Homotopy and Hopf's Degree Theorem

Shriprakash Sinha[1], **Luca Nanetti**[2], **Remco J. Renken**[2], **Gert J. ter Horst**[2]

[1] *104 – Madhurisha Heights Phase 1, Risali, 490006 Bhilai, India*
email: sinha.shriprakash@yandex.com

[2] *Neuroimaging Center, University Medical Center Groningen*
Antonious Deusinglaan 2, P.O. Box 196, 9700 AD Groningen, Netherlands
emails: {l.nanetti, r.j.renken, g.j.ter.horst}@med.umcg.nl

Abstract. The current work revisits the point-in-polygon problem by providing a novel solution that explicitly employs the properties of *epigraphs* and *hypographs*. Using concepts of epigraphs and hypographs, this manuscript provides a new definition of *inaccessibility* and *inside*, to accurately specify the meaning of inclusion of a point within or without a polygon. Via Poincaré's ideas on *homotopy* and *Hopf's Degree Theorem* from topology, a relationship between *inaccessibility* and *inside* is established and it is shown that consistent results are obtained for peculiar cases of both non-intersecting and self-intersecting polygons while investigating the point inclusion test w.r.t. a polygon. Through illustrative examples, the novel method addresses the issues of • ambiguous solutions given by the Cross Over for both non-intersecting and self-intersecting polygons and • a point being labeled as multi-ply inside a self-intersecting polygon by the Winding Number Rule, by providing an unambiguous and singular result for both kinds of polygons. The proposed solution bridges the gap between Cross Over and Winding Number Rule for complex cases.

Key Words: Point in polygon, epigraph, hypograph, homotopy, Hopf's degree theorem

MSC 2010: 65D18, 68U05

1. Introduction

Given a polygon $\mathcal{P}$ with the vertices (x_1, y_1), (x_2, y_2), …, (x_n, y_n), $(x_{n+1}, y_{n+1}) = (x_1, y_1)$, it is desired to know whether a sample point S (x_0, y_0) lies within $\mathcal{P}$. *The status of a point S with respect to a polygon $\mathcal{P}$, termed as inside, needs to be defined properly.* This appears to be crucial in order to retrieve unambiguous results not only for self-intersecting polygons but also for non-intersecting ones in a $2D$ Cartesian plane.

ISSN 1433-8157/$ 2.50 © 2015 Heldermann Verlag

In this manuscript, a new definition of *inaccessibility* and *inside* has been proposed, aiming to clarify the semantical meaning of *a point being inside (or outside) a polygon.* Later in the manuscript, it is shown that the newly proposed definitions form a bridge between the two well known definitions of Cross Over and the Winding Number Rule. This combined effect of both the existing definitions helps to resolve many of the rare issues of solving the point-in-polygon problem for simple as well as self-intersecting polygons.

Cross Over ([18], [4], [5], [6], [7]) states that if a half infinite line (ray) drawn from S cuts $\mathcal{P}$ an odd number of times, then the point is inside the polygon. However, there are a few issues regarding this definition. Depending on the *orientation* of the ray from the query point, odd or even number of intersections can be obtained, if the ray passes through vertices. This gives rise to ambiguous results for the same point with different rays at different orientation. In other words, the outcome is a *non-singular* function of the line's direction.

To resolve this issue, a prevalent solution is shifting of the ray infinitesimally ([19], [20]). Here the solution may change drastically depending on the *direction of the shift* and thus the direction of the shift is often fixed. This solution is palliative as the issue is considered to arise rarely and ambiguous results can still be found. A solution to resolve the issue could be to repeat the Cross Over multiple times until it is found that the point lies inside the polygon; however, exactly how many repetitions are necessary to prevene a reliable result is unknown leading to *non determinism.*

Winding Number Rule ([7], [3], [1]) states that the number of times one loops around S while traversing $\mathcal{P}$ before reaching the starting point on the polygon shows whether the point is inside the polygon or not. So a number ℓ greater than one can mean that the point is ℓ times inside the polygon. In this paper, this is considered as an issue because if a point lies inside a polygon once, it lies forever. Thus $\ell > 1$ depicts the idea of *redundancy.* Finally, the Cross Over and the Winding Number Rule algorithms are known to provide different solutions in specific cases of self-intersecting polygons.

The solution provided tries to address these problems by deciding upon the 'inside/outside' status of a point with respect to a polygon from a different perspective. [2] proposed to take the decision using a binary coded coordinate system and parity counting of the number of intersections of the polygon with an infinite vector. The algorithm presented here differs from the former in using S as a reference point. The location of S does not depend on the coordinate system, but it helps in forming a line (in any orientation) such that it cuts the polygon at different intersection points. For the sake of simplicity, the case where the line is horizontal is presented. The generalization only requires the rotation of the reference (i.e., horizontal-vertical) Cartesian system and the transformation of the coordinate points of the polygon with respect to the rotated system.

The first half line is used as a reference to dismember the polygon into sections. These sections (later defined as chains) are then classified as valid or invalid based on definitions of epigraphs or hypographs that may or may not contain S, respectively. A second line, orthogonal to the first, is used to sort these sections of the polygon that contain the sample point. The sections are then paired and taken out. For each pair the location of the point within the two sections, is checked. The process of checking continues until the algorithm runs out of pairs to be checked. This iterative procedure of elimination of sections that do not contain the point and final repetitive checking within the pairs of remaining sections, of the polygon, represents the core difference from [2], and is explained in detail in the following section.

A more in-depth theoretical approach to show the correctness of the proposed method follows in Section 3, accompanied by a comparison with the Cross Over (Section 3.3) and with the Winding Number Rule (Section 3.4). The conclusion follows in Section 4.

2. Algorithm implementation

The algorithm can be described in simple terms as follows. Given the sample point S with coordinates (x_0, y_0), a horizontal line $y = y_0$ (i.e., L_h) is drawn through S to cut polygon $\mathcal{P}$ at q locations $\{(x_1^{int}, y_0),\ \dots,\ (x_q^{int}, y_0)\}$. This breaks the polygon into q chains.

Definition 1. A *chain* $\mathcal{C}$ is a series of connected edges of the polygon whose starting and ending points lie on L_h, that passes through S. Mathematically, a *chain* is a function[1] $f_{\mathcal{C}}$, with a domain defined by the closed interval bounded by the x-coordinates of consecutive intersection points on L_h (here referred by starting and ending points), and a range that is the union of the coresponding y-coordinates of points of the polygon (i.e., a piecewise linear function) comprised between the starting and ending points on L_h.

Each chain is then checked for whether its two endpoints contain the test point between them; if not, the chain is discarded. Discarded chains are labeled as *invalid chains* and those kept for further consideration are referred to as *valid chains*. This classification is based on the satisfication of a criterion dependent on concepts of epigraph and hypographs. The definition of these concepts are as follows:

Definition 2. The *epigraph* of a function $f_{\mathcal{C}}\colon \mathbb{R}^n \to \mathbb{R}$ is a set of points that lie on or above the graph of the function under consideration, such that $epi(f_{\mathcal{C}}) = \{(x,t) : x \in \mathbb{R}^n,\ t \in \mathbb{R},\ f_{\mathcal{C}}(x) \leq t\}$ is a subset of $\mathbb{R}^{n+1}$.

Definition 3. The *hypograph* of a function $f_{\mathcal{C}}\colon \mathbb{R}^n \to \mathbb{R}$ is a set of points that lie on or below the graph of the function under consideration, such that $hypo(f_{\mathcal{C}}) = \{(x,t) : x \in \mathbb{R}^n,\ t \in \mathbb{R},\ f_{\mathcal{C}}(x) \geq t\}$ is a subset of $\mathbb{R}^{n+1}$.

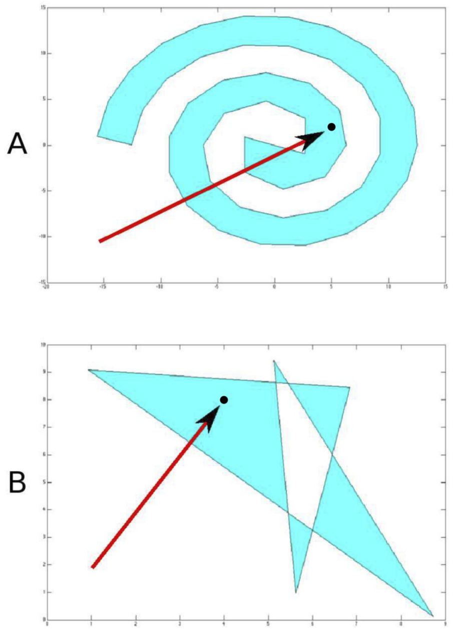
Here n equals 1 since x corresponds to a point on L_h. The remaining valid chains are then tested for intersection with a vertical line $x = x_0$ through S. The intersections found are sorted by height, and paired up. If the test point is not between a pair, it is outside. This criterion of containment is checked via the definition of a convex combination below:

Definition 4. An *affine combination* of points $x_i, x_j \in \mathbb{R}^n$ is a point of the form $\theta x_i + (1-\theta)x_j$ with $\theta \in \mathbb{R}$. This combination is called a *convex combination* if $0 \geq \theta \geq 1$.

These definitions and notations, as well as a few others, are adopted from [8]. It should be noted that the vertices of the polygon $\mathcal{P}$ are arranged in order of traversal, starting from any vertex. The traversal order can be in any one direction. Another requisite is that the edges are traversed only once. This is useful in avoiding multiple loops that may occur in cases of self-intersecting polygons.

If S lies out of the bounding box of the polygon, it is considered outside $\mathcal{P}$ and no further processing is done. Lastly, if the sample point is one of the vertices of the polygon, then it is considered to be in the polygon. This final point is assumed as the proposed algorithm would reach the same conclusion at the expense of computational time. In the theoretical proof it

[1] If any chain is not the graph of a function then in a preprocessing step we can modify the polygon accordingly by cutting edges. This has no effect on the result of the procedure, but eases our explanations.



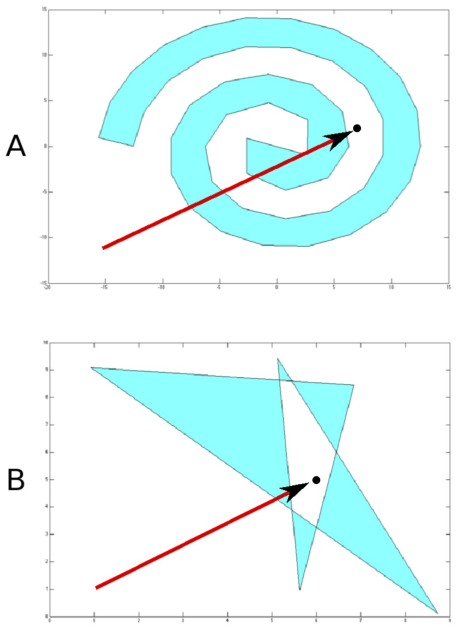


Figure 1: Query point location in (A) non-intersecting and (B) self-intersecting polygon.

Figure 2: Query point location in (A) non-intersecting and (B) self-intersecting polygon.

will be shown that this case is true and from the implementation point of view the idea holds correct.

What follows is a step-by-step explanation of how the algorithm works; pictorial representations will help in clarifying each step. The examples will regard closed polygons, both self-intersecting, and non-intersecting. Figures 1 and 2 show the polygons with the sample point being tested at different locations.

2.1. Intersecting the $\mathcal{P}$

It is known that $\mathcal{P}$ is an ordered series of vertices (x_1, y_1), (x_2, y_2), $\ldots$, (x_n, y_n), starting from (x_1, y_1) such that the ending point after the traversal has coordinates $(x_{n+1} = x_1,\ y_{n+1} = y_1)$. Given $\mathcal{P}$, the first step is to draw L_h through the sample point S (x_0, y_0), such that it intersects the polygon at certain points. As observed before, the line L_h being horizontal does not imply any loss of generality in the proposed solution for the point-in-polygon problem.

The intersection point is obtained by computing the coordinate values of the common point between L_h and a straight edge extending from (x_i, y_i) to (x_{i+1}, y_{i+1}) (henceforth $\overline{(x_i, y_i), (x_{i+1}, y_{i+1})}$). The slope and the constant of the former is 0 and y_0, and that of the latter is $m_i = \dfrac{y_{i+1} - y_i}{x_{i+1} - x_i}$ and $c_i = y_i - m_i x_i$. Here i and $i + 1$ are indices to any pair of consecutive vertices on $\mathcal{P}$. Let the coordinates of this common point be (x^{int}, y_0). Solving the algebraic equation between the two straight lines yield:

$$x^{int} = \frac{y_0 - c_i}{m_i} \tag{1}$$

In the case when the infinite line L_h is horizontal, the y-coordinate of the intersection point equals the y-coordinate of L_h, i.e., y_0. Once the intersecting point with coordinates $(x^{int},\ y_0)$ is obtained, the algorithm checks it's inclusion on $\overline{(x_i, y_i), (x_{i+1}, y_{i+1})}$ of $\mathcal{P}$, for further processing.

If the criterion of inclusion is satisfied, the intersection point is stored as a new vertex point that needs to be appended to the pre-existing list of vertices of the polygon at a later stage. This inclusion criterion is put into affect by the use of the convex combination property in Definition 4.

Three different cases arise depending on the slope of $\overline{(x_i, y_i), (x_{i+1}, y_{i+1})}$:

- $m_i = \pm\infty:$ If the edge is vertical the point (x^{int}, y_0) lies on the line. This is because $x^{int} = x_i = x_{i+1}$.
- $m_i = 0:$ If edge is horizontal then the intersection point (x^{int}, y_0) is considered to lie outside the range of (x_i, y_0) and (x_{i+1}, y_0). This is because, if $(x^{int}, y_0) \in [(x_i, y_0)$, and $(x_{i+1}, y_0)]$, then there are infinitely many values that could be assigned to x^{int}. To avoid random selection of any point within the above range, the point (x^{int}, y_0) is considered to lie outside the range. Also, if (x_0, y_0) lie on a horizontal edge it is still considered outside the range for further processing.
- $m_i \in \mathbb{R} - \{0, \pm\infty\}:$ Finally, this being the simplest case, it is easy to compute whether (x^{int}, y_0) lies on the line between the given points using Definition 4.

It is important to note that this process of finding intersection points and their subsequent inclusion into the pre-existing vertex list of $\mathcal{P}$, based on satisfaction of the convex combination property, is iterative in nature: that is, for each pair of (x_i, y_i) and (x_{i+1}, y_{i+1}) as i iterates through values 1 to n, the intersection points are computed between $\overline{(x_i, y_i), (x_{i+1}, y_{i+1})}$ and L_h. Next, for each edge $\overline{(x_i, y_i), (x_{i+1}, y_{i+1})}$ and its corresponding intersection point with L_h, it is verified whether the intersection point lies between (x_i, y_i) and (x_{i+1}, y_{i+1}). If yes, then the points are stored. The stored points are then appended to the list of existing vertices $\mathcal{P}$, such that the traversal order remains unaffected. In other words, if $(x^{int}, y_0) \in \overline{(x_i, y_i), (x_{i+1}, y_{i+1})}$ then a particular subsequence of the traversal order becomes: $(x_i, y_i) \to (x^{int}, y_0) \to (x_{i+1}, y_{i+1})$.

Figures 3 and 4 show L_h passing through S and intersecting the polygon at different edges. The location of S is indicated by the red arrow while the new vertices at the intersection of L_h with edges of $\mathcal{P}$ are pointed by the blue arrows. Some of the newly added vertices may lie very close to the pre-existing vertices, e.g., in the range of $\pm 10^{-5}$ (arbitrary units) or smaller. The algorithm removes these old vertices from the list that lie in such a close range, but stores the newly added vertices with their coordinates (x^{int}, y_0) separately. Two reasons arise for executing this step:

- To avoid further computations that may involve floating point precision of the order smaller than or equal to $\pm 10^{-5}$.
- Retention of newly appended vertices with coordinates $(x^{int},\ y_0)$ will be used later for searching chains whose epi/hypo-graph may contain S.

In this paper, the tolerance range of $\pm 10^{-5}$ is an arbitrarily assigned value.

2.2. Decomposition of polygon into valid and invalid chains

The new vertices with coordinates $(x_j^{int},\ y_0)$ (where $j \in \{1, \ldots, m\}$) and the sample point (x_0, y_0) form the basis for the next steps. From Definition 2 it is known that a point belongs to the epigraph (hypograph) if it lies on or above (below) the function under consideration. To use the mentioned properties, the polygon $\mathcal{P}$ is decomposed into chains. These chains would then be tested for convexity or concavity with respect to L_h in the following way: One

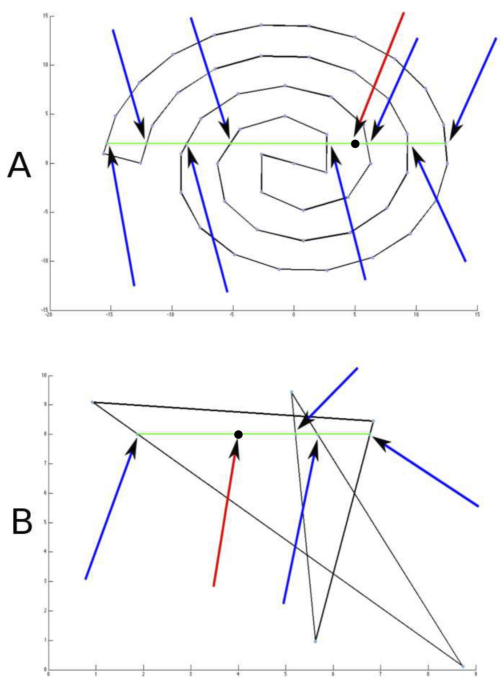


Figure 3: Finding intersection points on the polygon. Location of S indicated via red arrow in (A) non-intersecting and (B) self-intersecting polygon. Blue arrows point to newly found intersection points. The green line depicts L_h, i.e., $y = y_0$.

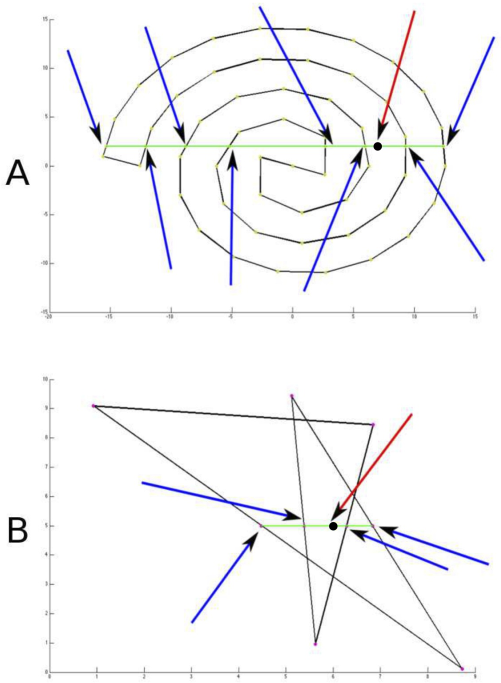


Figure 4: Finding intersection points on the polygon. Location of S indicated via red arrow in (A) non-intersecting and (B) self-intersecting polygon. Blue arrows point to newly found intersection points. The green line depicts L_h, i.e., $y = y_0$.

of the newly added vertex on $\mathcal{P}$ (with the y-coordinate y_0) is picked up as the starting vertex. A traversal order is chosen randomly and is followed until the starting point is reached again.

As the traversal is done from one intersecting vertex to the other, the polygon gets decomposed into subsets of consecutive edges, thus forming chains. Each chain may contain more than one original vertex of the polygon excluding the starting and ending vertex of the chain. These chains lie either above or below L_h, i.e., $y = y_0$. The chains are further classified as *valid* or *invalid* using the Definitions 2 and 3. In non-mathematical terms, if the starting and ending vertices of a chain are on different sides of S on L_h then the chain is a valid one: otherwise the chain is invalid. The invalid chains are discarded and the valid ones are stored with their starting and ending vertices along with coordinates of vertices on the chain.

Figures 5 and 6 show the valid and invalid chains pointed by the green and blue arrows respectively. The solid lines represent the valid chains and the dotted lines represent the invalid chains. The red arrow indicates the sample point's location in each of the figures.

2.3. Chain intersection

Hitherto, it is known that the x-coordinate of S lies in the epi/hypo-graph of the valid chains. To decide if the point lies inside or is inaccessible with respect to a polygon under consideration, what needs to be tested is whether the y-coordinate of S lies within any two nearest valid chains. The rationale behind doing these steps will be elucidated a little later, but before that it is important to define what the nearest valid chains mean:

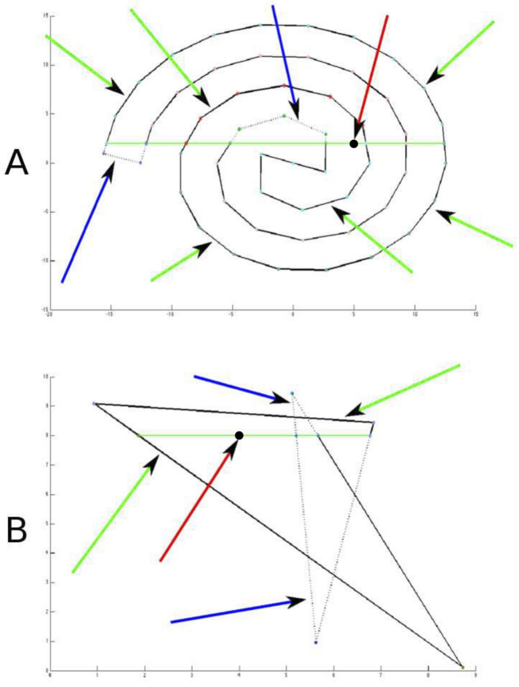


Figure 5: (A) Non-intersecting and (B) self-intersecting polygon decomposed into chains. Green and blue arrows indicate the valid and invalid chains, respectively. Location of query point w.r.t. the polygon is shown via the red arrow.

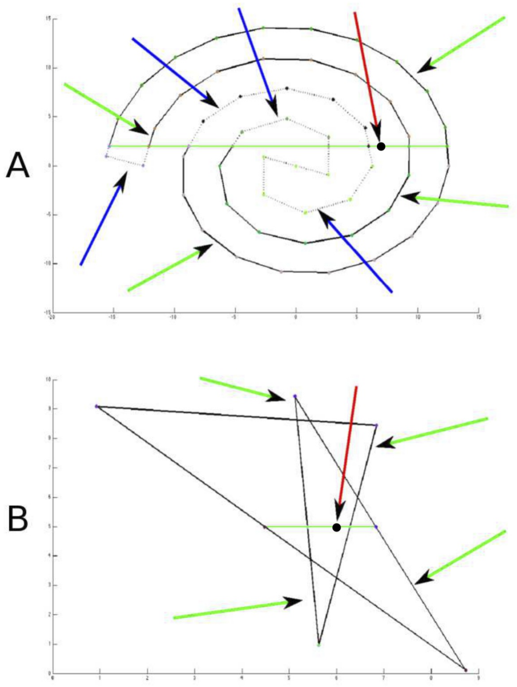


Figure 6: (A) Non-intersecting and (B) self-intersecting polygon decomposed into chains. Green and blue arrows indicate the valid and invalid chains, respectively. Location of query point w.r.t. the polygon is shown via the red arrow.

Definition 5. The chain $\mathcal{C}_i$ is a *nearest valid chain* if

- either $epi(f_{\mathcal{C}_i}) \subset epi(f_{\mathcal{C}_u})$ for all chains $\mathcal{C}_u$, $u \in \{1, \ldots, q\}$, below S, i.e., such that $(x_0, y_0) \in epi(f_{\mathcal{C}_u})$,
- or $hypo(f_{\mathcal{C}_i}) \subset hypo(f_{\mathcal{C}_v})$ for all chains $\mathcal{C}_v$, $v \in \{1, \ldots, q\}$, above S, i.e., such that $(x_0, y_0) \in hypo(f_{\mathcal{C}_v})$.

After all the valid chains have been retained, a similar procedure (as before) of intersecting the valid chains using the vertical line $x = x_0$ (L_v) that passes through S, is executed. For each valid chain, defined by a set of vertices (x_i, y_i) (such that $i \subseteq \{1, \ldots, n\}$), points of intersection are computed between the edges of the chain and L_v. The process of evaluating the intersection points follows:

The slope of the straight edge joining two vertices of the chain is $m_i = \dfrac{y_{i+1} - y_i}{x_{i+1} - x_i}$ and the constant is $c_i = y_i - m_i x_i$. Here, i and $i+1$ are consecutive vertices on the valid chain. The y-coordinate of the point of intersection is computed as follows:

$$y^{int} = m_i x_0 + c_i. \tag{2}$$

Once the coordinates of the intersection point (x_0, y^{int}) are obtained for an edge, a test is conducted to find whether the intersection point lies between the bounding points (x_i, y_i) and (x_{i+1}, y_{i+1}). This is achieved using the convex combination property in Definition 4. Three cases may arise, that need to be considered:

- $m_i = \pm\infty$, the edge is vertical. In this case the intersecting point (x_0, y^{int}) is considered to lie inside the range of (x_0, y_i) and (x_0, y_{i+1}). This is because, if the chain crosses $x = x_0$ many times before going from left of S to the right of S or vice versa then there can be infinitely many points on one chain that may be considered as intersection points. That is, it may require the algorithm to store many intersection points for just one chain. To avoid the presence of multiple intersection points on a single chain, the algorithm stores the vertex of one of the vertical edges in a chain.
- $m_i = 0$, the edge is horizontal. In this case the intersecting point (x_0, y^{int}) lies on the line. This is because $x^{int} = x_0$ and $y^{int} = y_i = y_{i+1}$.
- $m_i \in \mathbb{R} - \{0, \pm\infty\}$: This is the simplest case. It is easy to compute whether (x_0, y^{int}) lies on the line between the given points using Definition 4.

This process of finding the intersection point is repeated for all edges (i.e., for all $i \subseteq \{1, \ldots, n\}$) that constitute the valid chain under consideration. Next, for each edge $(x_i, y_i), (x_{i+1}, y_{i+1})$ and its corresponding intersection point with L_v, it is verified whether the intersection point lies within (x_i, y_i) and (x_{i+1}, y_{i+1}). It is expected that, after preprocessing of all edges on a chain, there exists only one intersection point between a chain and L_v. The reason behind this is to find one common point of intersection between the valid chain and L_v. This is done as a necessary step to sort the valid chains on the basis of y-coordinates of the newly found intersection points. The sorted valid chains will further be processed to test the inclusion of S within $\mathcal{P}$.

The pictorial representation of the line $x = x_0$ intersecting the valid chains are shown in Figures 7 and 8. The green arrows indicate the points of intersection on the valid chains. The valid chains are pointed by the blue arrows. The sample point is indicated via the red arrow. As mentioned earlier, the invalid dotted chains have been removed by the algorithm, in the final stages of the processing.

2.4. Point inclusion test

Each valid chain has a point of intersection with L_v. The algorithm sorts the series of intersection points with respect to their y-coordinates (i.e., y^{int}) that lie on L_v, thus sorting the valid chains.

It is also known that S lies either in the epigraph or in the hypograph of any valid chain. The final step to decide whether S is inside $\mathcal{P}$ is to assess whether the y-coordinate of S is compartmentalized between two nearest valid chains $\mathcal{C}_i$ and $\mathcal{C}_{i+1}$ (if $\mathcal{C}_i$ is below $\mathcal{C}_{i+1}$ after sorting), such that $S \in epi(f_{\mathcal{C}_i})$ and $S \in hypo(f_{\mathcal{C}_{i+1}})$. This is done as follows:

Considering a pair $(y_i^{int},\ y_j^{int})$ of y^{int}'s, i.e., a pair of chains, it is tested whether $S = (x_0, y_0)$ is an convex combination of (x_0, y_j^{int}) and (x_0, y_i^{int}) (for $i < j$). If such a pair is found, then the point lies inside the polygon, else outside. The only constraint is that the pair of chains or the pair of intersecting y-coordinates are mutually exclusive. Thus, if there are $m/2$ pairs, with m being an even number and $y_1^{int},\ y_2^{int},\ \ldots,\ y_m^{int}$ are the intersecting y-values in order, then the pairs $(y_1^{int},\ y_2^{int})$, $(y_3^{int},\ y_4^{int})$, $\ldots$, $(y_{m-1}^{int},\ y_m^{int})$ are mutually exclusive in the sense that the elements of one pair cannot be included in any other pair.

The intuitive idea behind this rule is that, in a pair, if the path of traversal in a chain is moving (say) from left side of S to the right side, then the traversal in the other chain must move from right side S to the left (and vice versa). Thus, any pair shall not contain chains from any other pair. These ideas can be seen in Figures 7 and 8. The green arrows mark the

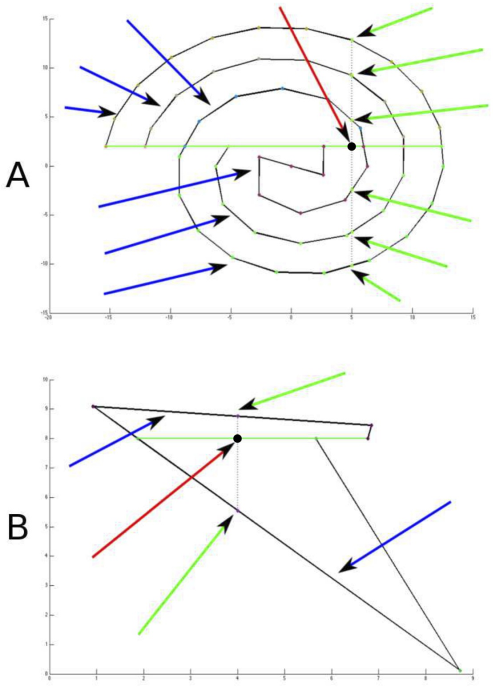


Figure 7: Inclusion test, S inside $\mathcal{P}$. Blue arrows: valid chains of (A) a non-intersecting and (B) a self-intersecting polygon. Green arrows: intersection points with the vertical line L_V. The point S (red arrow) is inside the polygon.

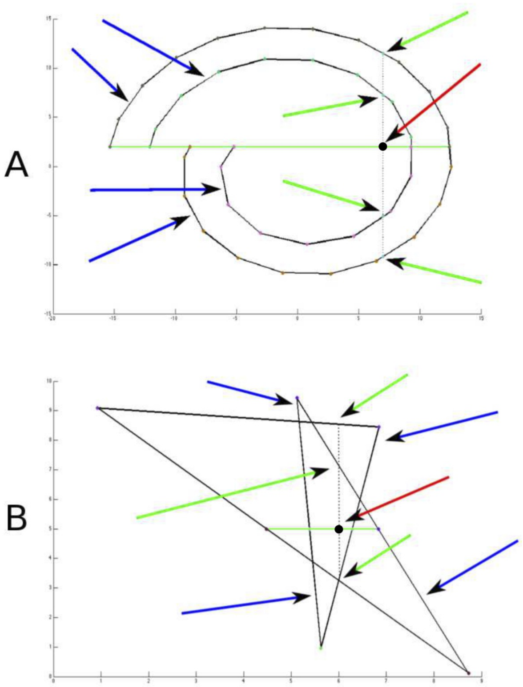


Figure 8: Inclusion test, S outside $\mathcal{P}$. Blue arrows: valid chains of (A) a non-intersecting and (B) a self-intersecting polygon. Green arrows: intersection points with the vertical line L_V. The point S (red arrow) is outside the polygon.

intersecting points on the valid chains. Testing the sample point S as an convex combination of two intersection points on L_v indicates whether the point lies inside or outside the polygon.

3. Inaccessibility-inside theorem

The meaning of inside is viewed from different perspectives via the definitions of the Cross Over and the Winding Number Rule. This gives rise to contradictory results in peculiar cases of non-intersecting and self-intersecting polygons.

This manuscript proposes new definitions of *inside* and *inaccessibility* of a point S with respect to a polygon $\mathcal{P}$. Also, a relation between inaccessibility and inside is proved. It is shown that consistent results can be obtained if the meaning of the inaccessibility and inside of a polygon is framed correctly, in an abstract sense. It must be noted that the points that lie on vertices are special cases and the definitions of *inside* and *inaccessibility* will have to be slightly modified to take them into account without changing the general, abstract meaning of inside and inaccessibility. Ultimately, two cases that are examined are: a point lying (1) on a vertex or (2) on an edge or anywhere else.

3.1. Point on vertex of polygon

Definition 6. The *inaccessibility* $\mathit{Inacc}_{\mathcal{P}}(S)$ of a point S with respect to (w.r.t. in brief) a polygon $\mathcal{P}$, is the number of valid chains that need to be *broken* by a line passing through S such that the line reaches outside the bounding box of the polygon.

$$\mathit{Inacc}_{\mathcal{P}}(S) = \begin{cases} n & \text{if } n \neq 0 \text{ valid chains need to be } \textit{broken}, \\ 0 & \text{otherwise.} \end{cases}$$

Definition 7. The status of a point S w.r.t. a polygon $\mathcal{P}$, that is $\mathit{Inside}_{\mathcal{P}}(S)$, is the existence of a chain $\mathcal{C}$ such that $S \in \mathit{epi}(f_{\mathcal{C}})$ or $S \in \mathit{hypo}(f_{\mathcal{C}})$.

$$\mathit{Inside}_{\mathcal{P}}(S) = \begin{cases} 1 & \text{if } S \in \mathit{epi}(f_{\mathcal{C}}) \text{ or } S \in \mathit{hypo}(f_{\mathcal{C}}), \\ 0 & \text{otherwise.} \end{cases}$$

These two definitions form the basis of the two theorems below.

Theorem 1. *Given a point S inside as well as inaccessible w.r.t. the polygon $\mathcal{P}$, then the following two statements*

(1) $\mathit{Inside}_{\mathcal{P}}(S) = 1$ *and*

(2) $\mathit{Inacc}_{\mathcal{P}}(S) = n$

are logically equivalent, i.e., $\mathit{Inside}_{\mathcal{P}}(S) = 1 \iff \mathit{Inacc}_{\mathcal{P}}(S) = n$.

Proof. (a) If $\mathit{Inacc}_{\mathcal{P}}(S) = n$ then $\mathit{Inside}_{\mathcal{P}}(S) = 1$.
If $\mathit{Inacc}_{\mathcal{P}}(S) = n$ then there exist n valid chains that need to be *broken* according to Definition 6. It is known that a chain is valid when either its epigraph or its hypograph contains S. This existence of n valid chains implies that $S \in \{\mathit{epi}(f_{\mathcal{C}_k}), \mathit{hypo}(f_{\mathcal{C}_k})\}$ for all $k \in \{1, \ldots, n\}$. But this is the definition of the status of S w.r.t. $\mathcal{P}$, i.e., $\mathit{Inside}_{\mathcal{P}}(S) = 1$ or $\mathit{Inside}_{\mathcal{P}}(S) \in \{1\}$.
(b) If $\mathit{Inside}_{\mathcal{P}}(S) = 1$ then $\mathit{Inacc}_{\mathcal{P}}(S) = n$.
If $\mathit{Inside}_{\mathcal{P}}(S) = 1$ then there exists a $\mathcal{C}$ with $S \in \{\mathit{epi}(f_{\mathcal{C}}), \mathit{hypo}(f_{\mathcal{C}})\}$. Thus, chain $\mathcal{C}$ is a valid chain, as it contains the point S. If S is inaccessible then there must exist at least one valid chain in $\mathcal{P}$ that needs to be *broken*. Since $\mathcal{C}$ is such a chain and the only chain that contains S, the inaccessibility order of S w.r.t. $\mathcal{P}$ is $\mathit{Inacc}_{\mathcal{P}}(S) = 1$ or $\mathit{Inacc}_{\mathcal{P}}(S) \in \{1\}$.
If S is a vertex such that it is an intersection point of two or more sides of a polygon, then all chains that have their epigraph or hypograph containing S, are valid, since it requires n chains (if n is the number of valid chains) to be *broken*. □

Pictures will help the reader to get acquainted with the practical consequences of the previous theorem. Figure 9 shows three different polygons with S as the point under consideration. The polygon in Figure 9.(A) has four chains that contain S namely (a) STU, (b) UVS, (c) SWX, and (d) XYS, which are valid. Thus, by virtue of Theorem 1, follows $\mathit{Inside}_{\mathcal{P}}(S) = 1$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 4$. Hence, S lies inside the polygon. Similarly, for Figure 9.(B) there is one chain $S$TU$S$ which is valid as it contains the point S. Thus we have $\mathit{Inside}_{\mathcal{P}}(S) = 1$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 4$. For the case of Figure 9.(C) there exists two chains that contain S, i.e., (a) $S$TU$S$ and $S$VW$S$ which are valid. So $\mathit{Inside}_{\mathcal{P}}(S) = 1$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 2$.

Since it is always true that S is inside $\mathcal{P}$ when it lies at one of its vertices, it is a reasonable strategy to check this first, avoiding further computations. Thus, in the current implementation of the algorithm, the sample point is always first checked against the vertices of the polygon in order to know if it belongs to $\mathcal{P}$.

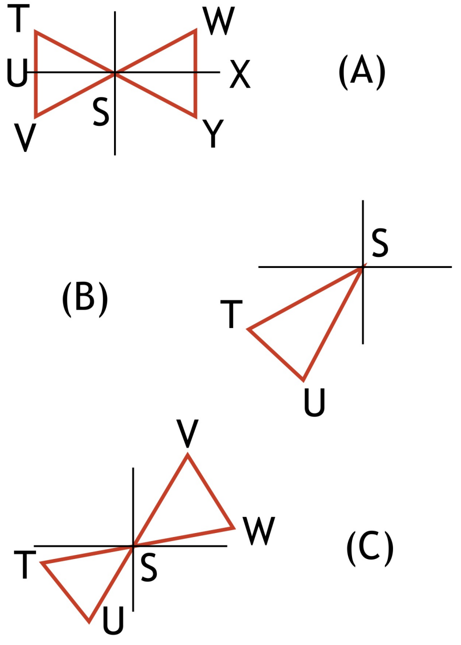


Figure 9: Polygons with locations of the point S.

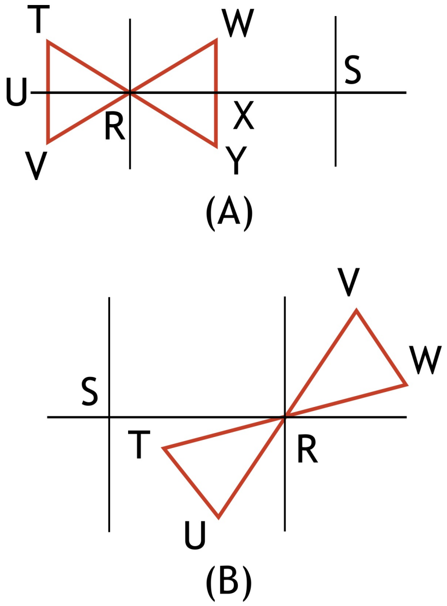


Figure 10: Polygons with locations of the point S.

Theorem 2. *Given a point S not inside as well as not inaccessible w.r.t. the polygon $\mathcal{P}$, then the following two statements*

(1) $Inside_{\mathcal{P}}(S) = 0$ and

(2) $Inacc_{\mathcal{P}}(S) = 0$

are logically equivalent, i.e., $Inside_{\mathcal{P}}(S) = 0 \iff Inacc_{\mathcal{P}}(S) = 0$.

Proof. (a) If $Inacc_{\mathcal{P}}(S) = 0$ then $Inside_{\mathcal{P}}(S) = 0$.
Given $Inacc_{\mathcal{P}}(S) = 0$, then there exist no valid chains that need to be *broken* according to Definition 6. This means that $S \notin \{epi(f_{\mathcal{C}_k}), hypo(f_{\mathcal{C}_k})\}$ for all k chains in $\mathcal{P}$. Since no chain exists whose epigraph or hypograph contains S, the status of S w.r.t. $\mathcal{P}$ is $Inside_{\mathcal{P}}(S) = 0$ or $Inside_{\mathcal{P}}(S) \in \{0\}$.

(b) If $Inside_{\mathcal{P}}(S) = 0$ then $Inacc_{\mathcal{P}}(S) = 0$.
$Inside_{\mathcal{P}}(S) = 0$ implies that $S \notin \{epi(f_{\mathcal{C}_k}), hypo(f_{\mathcal{C}_k})\}$ for all k chains in $\mathcal{P}$. This means, no valid chain exists in $\mathcal{P}$ that needs to be *broken*. Thus the inaccessibility of S related to $\mathcal{P}$ is zero, i.e., $Inacc_{\mathcal{P}}(S) \in \{0\}$, which is the desired result. □

Cases for Theorem 2 are simple and depicted in Figure 10. Figure 10 shows two different polygons with S as the point under consideration. The polygon in Figure 10.(A) has four chains that do not contain S, namely (a) RTU, (b) UVR, (c) RWX, and (d) XYR, which are invalid. Thus by Theorem 2, $Inside_{\mathcal{P}}(S) = 0$ and $Inacc_{\mathcal{P}}(S) = 0$. Hence, S lies outside the polygon.

Similarly, for Figure 10.(B) exist two chains that do not contain S, i.e., (a) RTUR and RVWR, which are invalid. So $Inside_{\mathcal{P}}(S) = 0$ and $Inacc_{\mathcal{P}}(S) = 0$.

3.2. Point not on vertex of polygon

Definition 8. The *inaccessibility* $Inacc_{\mathcal{P}}(S)$ of a point S w.r.t. a polygon $\mathcal{P}$ is the number of pairs of valid chains that need to be *broken and/or ignored* by a line passing through S such that the line reaches outside the bounding box of the polygon.

$$Inacc_{\mathcal{P}}(S) = \begin{cases} 1 & \text{if a pair of chains need to be } \textit{broken}, \\ n & \text{if } n \neq 0 \text{ pairs of chains are to be } \textit{ignored}, \\ 1+n & \text{if a pair is to be } \textit{broken} \text{ and } n \text{ pairs are to be } \textit{ignored.} \end{cases}$$

Definition 9. The status of a point S w.r.t. to a polygon $\mathcal{P}$, that is $Inside_{\mathcal{P}}(S)$, is the existence of a pair of chains $\mathcal{C}_i$ and $\mathcal{C}_j$ such that $S \in epi(f_{\mathcal{C}_i})$ and $S \in hypo(f_{\mathcal{C}_j})$, where $i < j$.

$$Inside_{\mathcal{P}}(S) = \begin{cases} 1 & \text{if a pair of chains } (\mathcal{C}_i, \mathcal{C}_j) \text{ exists such} \\ & \quad \text{that } S \in epi(f_{\mathcal{C}_i}) \text{ and } S \in hypo(f_{\mathcal{C}_j}), \\ 0 & \text{otherwise.} \end{cases}$$

The theorems revealing the relation between inaccessibility and inside of a polygon are as follows:

Theorem 3. *If a point S is inside as well as inaccessible w.r.t. a polygon $\mathcal{P}$, then the following two statements*

(1) *$Inside_{\mathcal{P}}(S) = 1$, and*

(2) *$Inacc_{\mathcal{P}}(S) = 1$ or $Inacc_{\mathcal{P}}(S) = 1 + n$,*

are logically equivalent, i.e.: $Inside_{\mathcal{P}}(S) = 1 \iff Inacc_{\mathcal{P}}(S) = 1$ or $Inacc_{\mathcal{P}}(S) = 1 + n$.

Proof. (a) If $Inacc_{\mathcal{P}}(S) = 1$ or $Inacc_{\mathcal{P}}(S) = 1 + n$ then $Inside_{\mathcal{P}}(S) = 1$.
$Inacc_{\mathcal{P}}(S) \in \{1, 1+n\}$ implies that there exists a pair of valid chains in $\mathcal{P}$ that need to be *broken* and/or n pairs of valid chains that need to be ignored. By definition, a valid chain is one whose epigraph or hypograph contains S. Taking the general case of $1+n$ (if $n = 0$, $1+n$ collapses to 1), there are $2 \times (1+n)$ valid chains such that one half lies above/on S and the other half lies below/on S. By Definition 5, the pair of nearest valid chains denoted by 1 needs to be broken while other n pairs need to be ignored.

If a vertical line L_v is drawn such that it cuts the valid chains and S, then the chains can be sorted according to the value of intersection points in $x = x_0$. Let $\mathcal{C}_1, \ldots, \mathcal{C}_{2\times(1+n)-1}, \mathcal{C}_{2\times(1+n)}$ be the sorted order of chains from bottom to top. Taking mutually exclusive consecutive pairs of these valid chains, i.e., $(\mathcal{C}_1, \mathcal{C}_2)$, $(\mathcal{C}_3, \mathcal{C}_4)$, $\ldots$, $(\mathcal{C}_i, \mathcal{C}_{i+1})$, $\ldots$, $(\mathcal{C}_{2\times(1+n)-1}, \mathcal{C}_{2\times(1+n)})$, it is easy to know whether S is an convex combination of $(x_0, y^{int}_{\mathcal{C}_k})$ and $(x_0, y^{int}_{\mathcal{C}_{k+1}})$, for all $k \in \{1, 3, 5, \ldots, 2\times(1+n)-1\}$. Here $y^{int}_{\mathcal{C}_k}$ and $y^{int}_{\mathcal{C}_{k+1}}$ are the y-coordinates of the intersection points on L_v and the chains k and $k+1$, respectively.

Since it is known that at least one pair of chains need to be *broken*, a pair of points $(x_0, y^{int}_{\mathcal{C}_i})$ and $(x_0, y^{int}_{\mathcal{C}_{i+1}})$ exists for which S has coordinates $(x_0, y_0) = (x_0, \theta y^{int}_{\mathcal{C}_i} + (1-\theta) y^{int}_{\mathcal{C}_{i+1}})$ for $0 \leq \theta \leq 1$. This implies that $\mathcal{C}_i$ is the nearest chain below/on S, and $\mathcal{C}_{i+1}$ is the nearest chain above/on S. Otherwise S won't be an convex combination of $(x_0, y^{int}_{\mathcal{C}_i})$ and $(x_0, y^{int}_{\mathcal{C}_{i+1}})$. For the rest of the n pairs, since S is not an convex combination of $(x_0, y^{int}_{\mathcal{C}_k})$ and $(x_0, y^{int}_{\mathcal{C}_{k+1}})$ for all $k \in \{1, 3, \ldots, 2\times(1+n)-1\} \setminus \{i\}$, these n pairs of chains can be *ignored* for further processing or consideration. Since these nearest chains $\mathcal{C}_i$ and $\mathcal{C}_{i+1}$ are valid also, their epigraph and hypograph contain S, respectively. This existence of a pair of a valid chains, which has to be *broken* such that $S \in epi(f_{\mathcal{C}_i})$ and $S \in hypo(f_{\mathcal{C}_{i+1}})$, implies $Inside_{\mathcal{P}}(S) = 1$, the status of S w.r.t. $\mathcal{P}$.

(b) If $\mathit{Inside}_{\mathcal{P}}(S) = 1$ then $\mathit{Inacc}_{\mathcal{P}}(S) = 1$ or $\mathit{Inacc}_{\mathcal{P}}(S) = 1 + n$.
Let $\mathcal{P}$ be a polygon such that $\mathit{Inside}_{\mathcal{P}}(S) = 1$ implies the existence of a pair of chains $(\mathcal{C}_i, \mathcal{C}_j)$ such that $S \in \mathit{epi}(f_{\mathcal{C}_i})$ and $S \in \mathit{hypo}(f_{\mathcal{C}_j})$. Given only these two chains, it is evident that both of them are nearest chains to S. Let the starting and ending points of $\mathcal{C}_i$ and $\mathcal{C}_j$ be $\{(x^{int}_{\mathcal{C}_{i_s}}, y_0), (x^{int}_{\mathcal{C}_{i_e}}, y_0)\}$ and $\{(x^{int}_{\mathcal{C}_{js}}, y_0), (x^{int}_{\mathcal{C}_{je}}, y_0)\}$, respectively. If a vertical line $x = x_0$ is drawn through (x_0, y_0) it would intersect the chains $\mathcal{C}_i$ and $\mathcal{C}_j$ at $(x_0, y^{int}_{\mathcal{C}_i})$ and $(x_0, y^{int}_{\mathcal{C}_j})$, respectively. Since $(x_0, y^{int}_{\mathcal{C}_i})$ lies below (x_0, y_0) and $(x_0, y^{int}_{\mathcal{C}_j})$ lies above (x_0, y_0), $S = (x_0, y_0)$ is an convex combination of $(x_0, y^{int}_{\mathcal{C}_i})$ and $(x_0, y^{int}_{\mathcal{C}_j})$. Thus, S lies between $\mathcal{C}_i$ and $\mathcal{C}_j$. Now, if an endpoint of $\mathcal{C}_i$ is joined with an endpoint of $\mathcal{C}_j$ and another endpoint of the former joined to the remaining endpoint of the latter, then a closed loop is formed such that traversing once from any one point, leads to the same point in the end. Let this loop be $\mathcal{P}'$. *To check for the inaccessibility, it needs to be known whether $\mathcal{P}'$ can be transformed into $\mathcal{P}$ while retaining all the geometric properties.*

Poincaré's ideas on homotopy ([9],[10], [11], and [12]) state that *geometric objects or simple cases of paths as continuous functions are homotopic if one function can continuously deform into another.* We obtain from [14]: Let $\mathbf{X}$ and $\mathbf{Y}$ be two (topological) spaces and $f, g\colon \mathbf{X} \to \mathbf{Y}$ two (continuous) maps from $\mathbf{X}$ to $\mathbf{Y}$.

Definition 10. The maps f and g are *homotopic*, if there exists a continuous map $\mathbb{F} : \mathbf{X} \times [0, 1] \to \mathbf{Y}$ such that $\mathbb{F}(x, 0) = f(x)$ and $\mathbb{F}(x, 1) = g(x)$. Such a map $\mathbb{F}$ is called a homotopy between f and g.

Simplifying the above scenario for two-dimensional Cartesian planes, f and g are functions in the 2D-space and $\mathbb{F}$ is a mapping that helps to transform f into g. In case such a mapping exists, then $\mathbb{F}$ is a homotopy between the two functions.

To retain geometric properties during the transformation, **Hopf's Degree Theorem** [13] states that *a loop $\mathbb{A}$ may be continuously deformed into another loop $\mathbb{B}$ without ever crossing the point $\mathbb{D}$ if and only if $\mathbb{A}$ and $\mathbb{B}$ have the same winding number around $\mathbb{D}$.* Formally, by virtue of [14], Hopf's degree theorem can be formulated as follows:

Theorem 4. *If $\mathbf{X}$ is a compact, connected, oriented, n-dimensional manifold without boundary and S^n the n-sphere, then the two maps $f, g\colon \mathbf{X} \to S^n$ are continuously homotopic if and only if* $\deg(f) = \deg(g)$.

Here, the *degree deg* of a map is a *homotopy invariant*, i.e., the maps respect the relation of *homotopy equivalence*. Definitions and proofs of the concept of the *degree* of a map and its *invariance against homotopies* is beyond the scope of this manuscript. Interested readers are advised to consult [14].

In the current situation f and g are the loops $\mathcal{P}'$ and $\mathcal{P}$, respectively. As long as the winding number of $\mathcal{P}'$ around S is same as that of $\mathcal{P}$ around S, $\mathcal{P}'$ and $\mathcal{P}$ are homotopic, i.e., $\mathcal{P}'$ can be deformed into $\mathcal{P}$ by Theorem 4. Without loss of generality, let us assume that $\mathcal{P}'$ can be deformed into $\mathcal{P}$.

Since $\mathcal{P}$ is formed from the two necessary chains $\mathcal{C}_i$ and $\mathcal{C}_j$ which are valid, a pair exists in polygon $\mathcal{P}$ that needs to be *broken*. Thus the minimum inaccessibility of S w.r.t. $\mathcal{P}$ is $\mathit{Inacc}_{\mathcal{P}}(S) = 1$. If there exist extra pairs of valid chains then they are *ignored* from consideration, while checking for the convex combination criteria of S with respect to the sorted pairs of chains on $x = x_0$. If n is the minimum number of pairs of valid chains that are *ignored*, then the inaccessibility of S w.r.t. $\mathcal{P}$ is $\mathit{Inacc}_{\mathcal{P}}(S) = 1 + n$. Thus $\mathit{Inacc}_{\mathcal{P}}(S) = \{1, 1 + n\}$. □

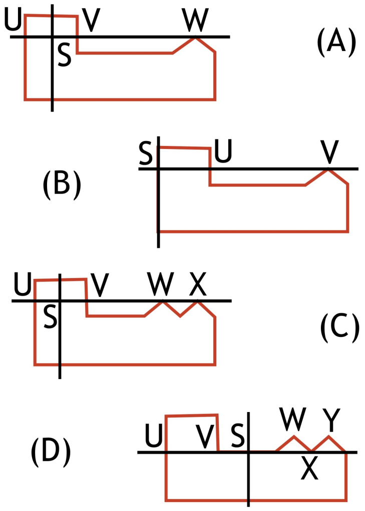


Figure 11: Polygons with locations of the point S.

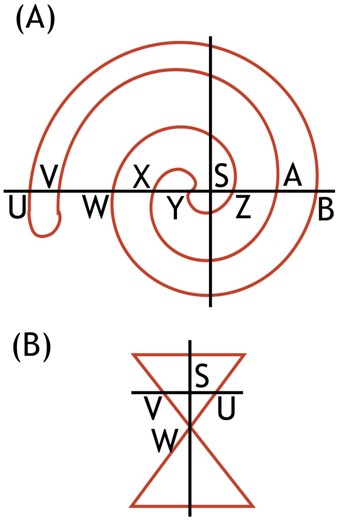


Figure 12: Polygons with locations of the point S.

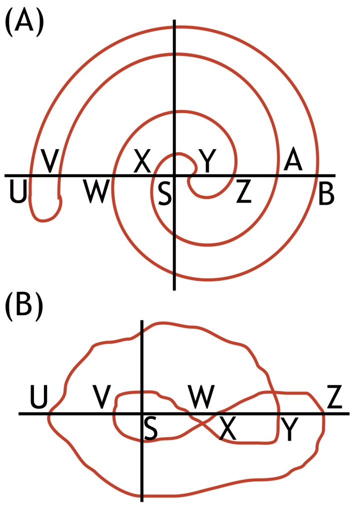


Figure 13: Polygons with S outside.

Examples for Theorem 3 are presented in the Figures 11 and 12. In Figure 11.(A), three chains exist of which two are valid. The valid chains are (a) UV and (b) WU. The chain VW is invalid. Thus $\mathit{Inside}_{\mathcal{P}}(S) = 1$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 1$. In Figure 11.(B) the point lies on the edge and the polygon can be divided into three chains of which only two contain S. These valid chains are (a) SU and (b) VS. Thus $\mathit{Inside}_{\mathcal{P}}(S) = 1$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 1$. In part (C) of Figure 11, the horizontal line cuts through two edges and touches two vertices. Thus there exist four chains of which two are valid, namely (a) UV and (b) XU, which contain S. Thus $\mathit{Inside}_{\mathcal{P}}(S) = 1$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 1$. Lastly, in part (D) of the same figure, S lies on an edge and the horizontal line passes through a vertex. In this case, there exist five chains of which two contain the sample point and are thus valid. They are (a) VW and (b) YU. Thus $\mathit{Inside}_{\mathcal{P}}(S) = 1$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 1$.

Next, in Figure 12.(A) eight chains exist, namely (a) YZ, (b) ZW, (c) WB, (d) BU, (e) UV, (f) VA, (g) AX, and (h) XY, of which six are valid except UV and XY. These two do not contain S. Now, the pair that needs to be *broken* is YZ and ZW, while the pairs (WB, AX) and (BU, VA) are to be *ignored* from consideration. Thus, $\mathit{Inside}_{\mathcal{P}}(S) = 1$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 1 + 2$ (i.e., one pair requires to be broken and two need to be ignored from consideration). Finally, in Figure 12.(B) two chains exist of which both are valid, i.e., (a) UV and (b) VWWU. Thus $\mathit{Inside}_{\mathcal{P}}(S) = 1$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 1$.

Theorem 5. *If a point S is not inside as well as inaccessible w.r.t. the polygon $\mathcal{P}$, then the following two statements*

(1) $\mathit{Inside}_{\mathcal{P}}(S) = 0$, *and*

(2) $\mathit{Inacc}_{\mathcal{P}}(S) = n$

are logically equivalent: $\mathit{Inside}_{\mathcal{P}}(S) = 0 \iff \mathit{Inacc}_{\mathcal{P}}(S) = n$.

Proof. (a) If $\mathit{Inacc}_{\mathcal{P}}(S) = n$ then $\mathit{Inside}_{\mathcal{P}}(S) = 0$.
$\mathit{Inacc}_{\mathcal{P}}(S) = n$ implies that n pairs of valid chains need to be *ignored*. Again, by definition a valid chain is one whose epigraph or hypograph contain S. If L_v, i.e., $x = x_0$, is drawn such that it cuts the valid chains and passes through S then the chains can be sorted according to the value of intersection points in $x = x_0$. Let $\mathcal{C}_1, \ldots, \mathcal{C}_{2\times n}$ be the sorted order of chains

from bottom to top. Taking consecutive pairs of these valid chains, i.e., $(\mathcal{C}_1, \mathcal{C}_2)$, $(\mathcal{C}_3, \mathcal{C}_4)$, $\dots$, $(\mathcal{C}_i, \mathcal{C}_j)$, $\dots$, $(\mathcal{C}_{2\times n-1}, \mathcal{C}_{2\times n})$, it is easy to know whether S is an convex combination of $(x_0,\, y^{int}_{\mathcal{C}_k})$ and $(x_0,\, y^{int}_{\mathcal{C}_{k+1}})$ for all $k \in \{1, 3, \dots, 2\times n-1\}$. Since n pairs need to be *ignored*, it is evident that S is not a convex combination of any of the above pairs. This suggests that there does not exist a pair such that $S \in \mathit{epi}(f_{\mathcal{C}_k})$ and $S \in \mathit{hypo}(f_{\mathcal{C}_{k+1}})$, that can be *broken*. Since no such pair exists, the status of S w.r.t. $\mathcal{P}$ is $\mathit{Inside}_{\mathcal{P}}(S) = 0$, which is the desired result.

(b) If $\mathit{Inside}_{\mathcal{P}}(S) = 0$ then $\mathit{Inacc}_{\mathcal{P}}(S) = n$.
$\mathit{Inside}_{\mathcal{P}}(S) = 0$ implies there does not exist a pair of chains $(\mathcal{C}_i, \mathcal{C}_j)$ such that $S \in \mathit{epi}(f_{\mathcal{C}_i})$ and $S \in \mathit{hypo}(f_{\mathcal{C}_j})$. Thus it is difficult to proceed with the proof. Instead, by proving its contrapositive, the above statement will hold. If $\mathit{Inacc}_{\mathcal{P}}(S) \notin \{n\}$ then $\mathit{Inside}_{\mathcal{P}}(S) \notin \{0\}$. Since $\mathit{Inacc}_{\mathcal{P}}(S) \notin \{n\}$, it follows that $\mathit{Inacc}_{\mathcal{P}}(S) \in \{1, 1+n\}$. It has been proved in part (a) above that if $\mathit{Inacc}_{\mathcal{P}}(S) \in \{1, 1+n\}$ then $\mathit{Inside}_{\mathcal{P}}(S) \in \{1\}$. But $\mathit{Inside}_{\mathcal{P}}(S) \in \{1\}$ also means that $\mathit{Inside}_{\mathcal{P}}(S) \notin \{0\}$. Thus $\mathit{Inacc}_{\mathcal{P}}(S) \notin \{n\}$ implies that $\mathit{Inside}_{\mathcal{P}}(S) \notin \{0\}$. Since the contrapositive holds, so does the original statement. □

Two examples of Theorem 5 are depicted in Figure 13. In Figure 13.(A) eight chains exist, namely, (a) UV, (b) VA, (c) AX, (d) XY, (e) YZ, (f) ZW, (g) WB, and (h) BU. Among them, three pairs exist, which are valid chains but need to be *ignored* as none of them encloses S as a convex combination of two intersection points on $x = x_0$. These pairs are (WB, AX), (XY, ZW) and (VA, BU). Thus $\mathit{Inside}_{\mathcal{P}}(S) = 0$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 3$.
For the case depicted in Figure 13.(B) six chains exist namely, (a) ZU, (b) UY, (c) YW, (d) WX, (e) VX, and (f) XZ, of which WY and XZ are not valid. The remaining pairs of valid chains need to be *ignored* and thus $\mathit{Inside}_{\mathcal{P}}(S) = 0$ and $\mathit{Inacc}_{\mathcal{P}}(S) = 2$.

For the next few sections, let EH (epi/hypo-graph method) denote the proposed method.

3.3. Crossover vs EH

Crossover (CR) states that when a line drawn from a point S in a direction, cuts the polygon $\mathcal{P}$ odd number of times, then S is inside $\mathcal{P}$, i.e.,

$$\mathit{Inside}^{CR}_{\mathcal{P}}(S) = \begin{cases} 1 & \text{if odd intersections,} \\ 0 & \text{if even intersections.} \end{cases}$$

For the case of a line passing through vertices, the problem is solved by shifting the line infinitesimally. Two issues arise in this case:

(1) There can be two solutions, if the line is not shifted slightly.

(2) If the crossover has to be repeated several times until it finds an odd number of intersections, then it is a nondeterministic problem, in case the line is shot randomly.

In view of (1), ambiguity arises on the way a ray or line is shot from S, and by (2), nondeterminism arises due to repetition of the procedure of shooting the line randomly. The following figures will illustrate these issues in detail.

In contrast to the CR, by using Theorems 1 and 3, the EH or the proposed method can easily determine deterministically, whether S lies in $\mathcal{P}$ or not. This is because in which way ever a line is drawn through S, if it cuts the polygon, then it will dismember $\mathcal{P}$ into a finite number of countable chains. If it doesn't cut the polygon and S is a vertex, then there also exists at least one chain that contains S. Searching for these valid chains and then locating which of those need to be *broken* is deterministic.

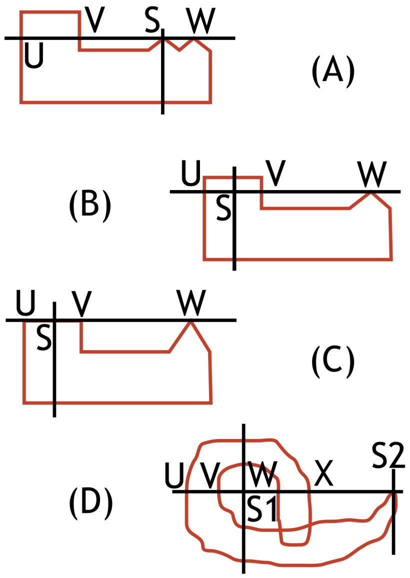


Figure 14: Cases to compare Cross Over (CR) and the proposed method (EH)

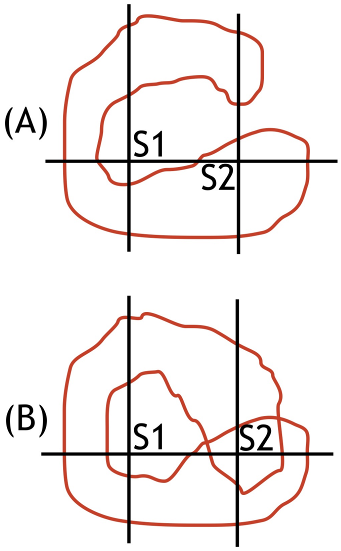


Figure 15: Cases to compare Winding Number Rule (WNR) and the proposed method (EH)

Figure 14 shows the different cases for the comparison of the CR and EH methods for the same point of investigation. In Figure 14.(A), if the a horizontal line is drawn to the left of the S then it intersects at two points U and V, and if it is drawn to the right it intersects at the point W. According to CR, when the line is drawn to the right of the S, then S is inside the polygon. If the line is drawn to the left of S, then the point is outside the polygon. This is definitely a case of ambiguity. Also, the outcome of CR depends on the direction of the ray that is shot from S. This makes the outcome of the test nondeterministic in the sense that it is not known which ray would give the correct result, if the rays are shot randomly.

The EH method overcomes this problem by segmenting the polygon into finitely countable chains. The search for an convex combination of valid chains that may contain S is deterministic as there is only limited number of chains available for checking. Thus the outcome is singular and deterministic. If two perpendicular rays with their intersection point at S are drawn at a different orientation, intersecting the polygon at different places, even then by rotating the oriented axis and the polygon to the horizontal-vertical frame, the solution remains the same. Thus randomness of the directionality of the shot rays do not affect the outcome of the point inclusion test for S.

For part (A) in Figure 14, by CR $\mathit{Inside}^{CR}_{\mathcal{P}}(S) = (0, 1)$ depending on the number of intersections that is $(2, 1)$. By the EH method, $\mathit{Inside}^{EH}_{\mathcal{P}}(S) = 1$ and $\mathit{Inacc}^{EH}_{\mathcal{P}}(S) = 3$ by Theorem 1. It must be noted that the inaccessibility of the point w.r.t. the polygon may change but the status of S w.r.t. $\mathcal{P}$ captured by the definition of *Inside* will not change if the point is inside the polygon.

Similarly, for the parts (B) and (C) in Figure 14, by CR we obtain $\mathit{Inside}^{CR}_{\mathcal{P}}(S) = (1, 0)$ depending on the number of intersections based on the direction of the ray which is $(1, 2)$. Finally, in Figure 14.(D), for point S_1 four valid chains exist namely, (a) VW, (b) XU, (c) US_2, and (d) S_2V, none of which need to be *broken* or *ignored*. Thus by Theorem 3 $\mathit{Inside}^{EH}_{\mathcal{P}}(S_1) = 0$ and $\mathit{Inacc}^{EH}_{\mathcal{P}}(S_1) = 2$. By CR, the outcome of the inclusion test changes, that is $\mathit{Inside}^{CR}_{\mathcal{P}}(S_1) = (0, 1)$ depending on the intersections obtained by the direction of the ray that is $(2, 3)$. For the point S_2, two valid chains exist, namely (a) S_2V, and (b) VWXUS_2.

Thus by Theorem 1, $\mathit{Inside}_{\mathcal{P}}^{EH}(S_2) = 1$ and $\mathit{Inacc}_{\mathcal{P}}^{EH}(S_2) = 2$. By CR follows $\mathit{Inside}_{S_2}^{EH}(S_2) = 0$, as the number of intersections is 4.

3.4. Winding Number Rule vs EH

The Winding Number Rule (WNR) states that the number of times a point tracing the polygon $\mathcal{P}$ surrounds the point S before reaching its starting point, decides whether S lies inside $\mathcal{P}$ or not. Thus

$$\mathit{Inside}_{\mathcal{P}}^{WNR}(S) = \begin{cases} n & \text{if there are } n \text{ loops around } S \\ 0 & \text{if there ar zero loops around } S \end{cases}$$

In Figure 15 an analogy to a prison wall is taken into account in order to the explain the differences. Figure 15.(A) is the initial structure of the prison, and then the final structure is shown in part (B) of the same figure. Initially, via the WNR, S_1 was lying outside and S_2 inside the prison wall. The same is the verdict by the new method. Next a portion of the prison wall is extended, and the final structure looks like that in Figure 15.(B). Note that S_1 and S_2 are still outside the new prison via the new definition, as the areas in which S_1 and S_2 lie, are not reachable from the prison's perspective. This is because two pairs of walls have to be *ignored and not broken* in each case. From this point of view both S_1 and S_2 are outside $\mathcal{P}$, in Figure 15.(B). Also, even though WNR $= 2$ for S_2 in the new prison in part (B) of the same figure, implying that the point lies twice inside, it does not make sense. It can be stated that if a point lies inside once, then it lies inside forever. There does not arise the idea of a point lying inside n times. Thus, a point lying inside n times is the same as the point is lying inside once. If it does not lie inside, then it won't lie forever. In this way the new definitions and the accompanying theorems are definitive in producing a concrete answer via means of the epigraph-hypograph method.

4. Conclusion

An old problem of containing contradictory definitions of Cross Over and Winding Rule for point in polygon is addressed. A theoretically reliable and analytically correct solution is proposed, using epi-hypo graphs, homotopy and Hopf's Degree Theorem. The proposed definitions along with the partial proofs indicate the bridge between the Cross Over and the Winding Number Rules. The method resolves the problem of ambiguity of solution due to the ray passing through vertices, the direction of the shift of the ray, and nondeterminism induced by usage of the Cross Over multiple number of times. For complex self-intersecting polygons, the presented algorithm challenges and addresses the issue of a point being multiply inside the polygons posed by the Winding Number Rule. By providing an unambiguous, singular and deterministic solution for both non-intersecting and self-intersecting polygons, the novel solution redefines concepts of inside and inaccessibility of a sample point with respect to a polygon and presents a fresh perspective of solving the old point-in-polygon problem.

5. Acknowledgement

The authors thank (1) the NeuroImaging center at UMCG Groningen, in The Netherlands, for supporting the project as a part of research in development of brain image segmentation algorithms using computational geometry algorithms, (2) anonymous reviewers for refining the manuscript, (3) H. Stachel for helping us in finalizing the manuscript, and (4) special thanks to Mr. Prabhat Sinha and Mrs. Rita Sinha for financially supporting the first author for this project.

References

[1] B. Zalik, I. Kolingerova: *A cell-based point-in-polygon algorithm suitable for large sets of points*. Comput. Geosci. **27**/10, 1135–1145 (2001).

[2] M. Chen, P. Townsend: *Efficient and consistent algorithms for determining the containment of points in polygons and polyhedra*. Eurographics, 423–437 (1987).

[3] C.W. Huang, T.Y. Shih: *On the complexity of point-in-polygon algorithms*. Comput. Geosci. **23**/1, 109–118 (1997).

[4] S. Nordbeck, B. Rystedt: *Computer cartography point in polygons program*. National Swedish Institute for Building Research Report, 19 pages, 1967.

[5] U. Manber: *Introduction to algorithms – a creative approach*. Addison-Wesley, 1989.

[6] J.D. Foley, V. Dam, A. Feiner, J.F. Hughes: *Computer Graphics – Principles and Practice*. Addison-Wesley, 1990.

[7] E. Haines: *Point in Polygon*. Strategies. Graphics Gems IV, chapter 1.4, 24–46 (1994).

[8] S. Boyd, L. Vandenberghe: *Convex Optimization*. Cambridge University Press, 2004.

[9] R.V. Eynde: *Historical evolution of the concept of homotopic paths*. Arch. Hist. Exact Sci. **45**/2, 127–188 (1992).

[10] D. Siersma: *Poincaré and Analysis Situs, the beginning of algebraic topology*. Nieuw Arch. Wiskd., Ser. 5, **13**/3, 196–200 (2012).

[11] J.M. Moller: *Homotopy theory for beginners*. `http://www.math.ku.dk/\~moller/e01/algtopI/comments.pdf`, 2012.

[12] H. Poincaré: *Papers on Topology: Analysis Situs and Its Five Supplements*. American Mathematical Society, 37, 2010.

[13] T. Needham: *Visual complex analysis*. Oxford University Press, 1999.

[14] J. Bosshardt: *The hopf degree theorem*. `http://www.math.uchicago.edu/\~may/VIGRE/VIGRE2011/REUPapers/Bosshardt.pdf`, 2011.

[15] Q. McNemar: *Note on the sampling error of the difference between correlated proportions or percentages*. Psychometrika **12**, 153–157 (1947).

[16] D. Engwirda: *Inpoly function*. `http://www.mathworks.com/matlabcentral/fileexchange/10391`, 2007.

[17] Matlab inpolygon function. `http://www.mathworks.com/access/helpdesk/help/techdoc/ref/inpolygon.html`

[18] M. Shimrat: *Algorithm 112: Position of point relative to polygon*. Commun. ACM **5**/8, p. 434 (1962).

[19] A.S. Glassner: *An introduction to ray tracing*. Morgan Kaufmann, 1989.

[20] F.P. Preparata, M.I. Shamos: *Computational Geometry*. Springer, 1985, pp. 1–35.

Received September 11, 2014; final form May 27, 2015