Modified One-Point-and-Area Algorithm for Sub-division of Irregular Parcels

Amrit Karmacharya[1], Janak Parajuli[1], Shrisha Makaju Shrestha[1]
akarmacharya8@gmail.com, janak.parajuli1@gmail.com, makajushrisha@gmail.com
[1]Survey Department

KEYWORDS

Polygon Sub-division, Irregular-shaped parcel, One-point-and-area, Land parcels, Cadastre, Concave parcel

ABSTRACT

Parcel sub-division is a crucial part of the Land Administration, and its techniques depend on the user requirements and other constraints. Splitting of a parcel based on a given area is one of such constraints whose subdivision has been popularly carried out using the trial-and-error method which is less accurate and time-consuming. For more accurate and direct subdivision, Habib developed algorithms based on popular eight cases. Among them, when the area to be split is known, a point or line is selected as a reference. In case if it has to be a point, then a one-point-and-area method is used for subdivision while a move-line-and-area method is used if line segment is used as reference. It was found that Habib Algorithm worked well for convex shapes, however it could not account for the concavity of irregular-shaped parcels. It was found that the approach for the calculation of the area considered only the cases of convex shapes. Therefore, the researchers observed the issues of clinging and retracting were identified. Thus, this study proposes an effective algorithm for the division of an irregular parcel using an improved version of One-Point-and-Area method used in Habib Algorithm. An additional stage of generating a convex hull was introduced to guide the iteration process, the formula to calculate area was adjusted to compute only the interior area and finally, the determination of the partition polygon was also done through an iterative improvement process rather than the existing one-step process. Experiments showed that the proposed algorithm accurately divided both irregular-shaped parcels and convex parcels. In the case of convex parcels, the result was found to be the same as Habib's method. Due to the addition of an extra stage of generating a convex hull and extra iterations in the final stage, the proposed algorithm consumed more time. However, it was found negligible in regard to the volume of parcels involved in splitting at a time. Overall, the proposed algorithm accurately solved the issues of clinging and retracting irregular polygons and without changing the results in convex polygons. So, it was concluded to be a suitable replacement of Habib Algorithm, and its implementation will provide more accurate results in parcel sub-division problems.

1. INTRODUCTION

Land partition involves dividing large pieces of land into smaller parcels based on specific requirements. Ideally, the parcel is staked out on the ground (Habib, 2020) and land professionals draw plans according to the measurements taken from the ground which are provided to legal offices (land revenue office and survey office) for authorization. However, practically, landowners first split the parcel shape in maps and stake out them on the ground accordingly. This splitting is done commonly using the following guidelines and tools for land subdivisions. (Easa, 2008)

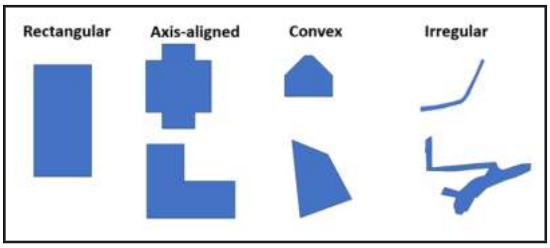
1) Partition line goes in a direction through a specific point (One point and Line method).
2) The partition line goes through two specific points (Two points/Join Points method).
3) The partition line runs through a determined vertex and cuts a specified area (One-Point-and-Area method).
4) The partition line starts in a supposed direction and cuts a fixed area (Move line cut area).
5) The partition line starts in a supposed direction and fixed distance from an edge (Offset distance from edge method)

While using each one or a combination of the above methods, technical professionals generally carry out either the trial-and-error method or the direct method. The trial-and-error procedure assumes a tentative partition line and then establishes the exact one by relying on an iterative approach (Habib, 2020). It is a time-consuming approach susceptible to human errors. Direct methods determine partition lines using geometric and trigonometric characteristics where points are considered to have 0-dimensionality, lines have 1-dimensionality, and polygons have 2-dimensionality. This method has a variety of tools and approaches for subdivision although it is not always guaranteed to give the required results. The cases of parcel sub-division where the partition line (1-dimensionality) is determinable using points (0-dimensionality), angles (1-dimensionality), or their combination (e.g. One point and line, Two Points, offset distance from edge) can be handled by simple trigonometric calculations. However, the sub-division where a required area is given (2-dimensionality) and the partition line (1-dimensionality) is to be determined such that one of the split areas equals the input area is tricky, although in some cases, it might be the only method applicable. Take a parcel to be split into two halves for inheritance purposes for example. Here, the original and required areas are known and a common point is selected to provide road access to before splitting the parcel. Now, the task is to find a point on the other side which will divide the original parcel in half. Unlike tools such as Join Points, One point and a line, and offset distance where desired output can be achieved in a single operation, the One point and area and move line and area tools require some hit and trial operations before finding the desired output.

1.1 Background

Parcel sub-division is a spatial operation and the shape of the polygon is an important factor impacting its efficiency (Zhang et al., 2021 In the real world, parcels with arbitrary shapes such as being (1) (Habib, 2020), highly irregular, complex, concave, or convex, self-intersecting, or even hollows can be found. As seen in Figure 1, categorically, the parcels can be distinguished into four groups (Cogo et al., 2023) rectangular, axis-aligned, convex, and irregular.

The rectangular shapes have exactly four edges and all their angles are 90°. Axis-aligned shapes, as the name suggests, are seemingly aligned with either of the axes (X, Y, or Z) and can have angles dividable by 90°. All angles in convex polygons are always smaller than or equal to 180°. Other shapes not belonging to the above categories are irregular-shaped parcels. In this research, a concave-shaped parcel is treated as an irregular one.



Source: Cogo et al., 2023

Figure 1: Categorization of Shape

The suggested algorithm determines the location of vertices at various partition levels. It is a straightforward, precise process and only requires the coordinaters for the peoperty corners. Among the eight scenarious, one is where the partition line runs through a determined vertex and cuts a specified area. This scenario is also knbown as the One-Point-and-Area mehtod.

As shown in Figure 2, coordinates P_1, P_2…. Pn represent a tract of land for which the coordinates of each point are known. Line P_1P' passes through point P_1 and intersects the polygon at point P' to separate the tract into two parts. It is required to compute the coordinates of the point, where the clipped part (shaded polygon) has the required area, A_S. In this case, the remaining triangular area A_1 is solved by subtracting the area of a portion A_2 from the area of the shaded part, which was specified only by the known vertices.

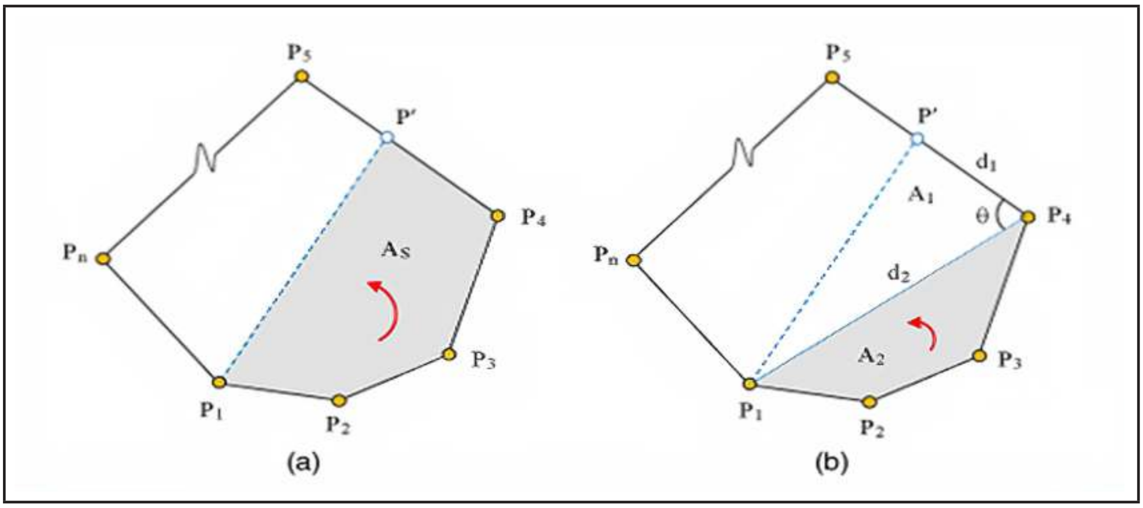
$$A_1 = A_S - A_2 \qquad \text{(i)}$$

For $A_1 > 0$, using their coordinates, the distance between the points P_1 and P_4 is computed by (Habib, 2020)

$$d_1 = \sqrt{(x_{p_1} - x_{p_4})^2 + (y_{p_1} - y_{p_4})^2} \qquad \text{(ii)}$$

Then distance d_1 is obtained from equation 3. Once, d_1 and bearing of P_4P_5 is computed Coordinates of P' can be determined.

$$A_1 = \frac{1}{2}(d_1 d_2)\sin\theta \Rightarrow d_1 = \frac{2A_1}{d_2 \sin\theta} \qquad \text{(iii)}$$



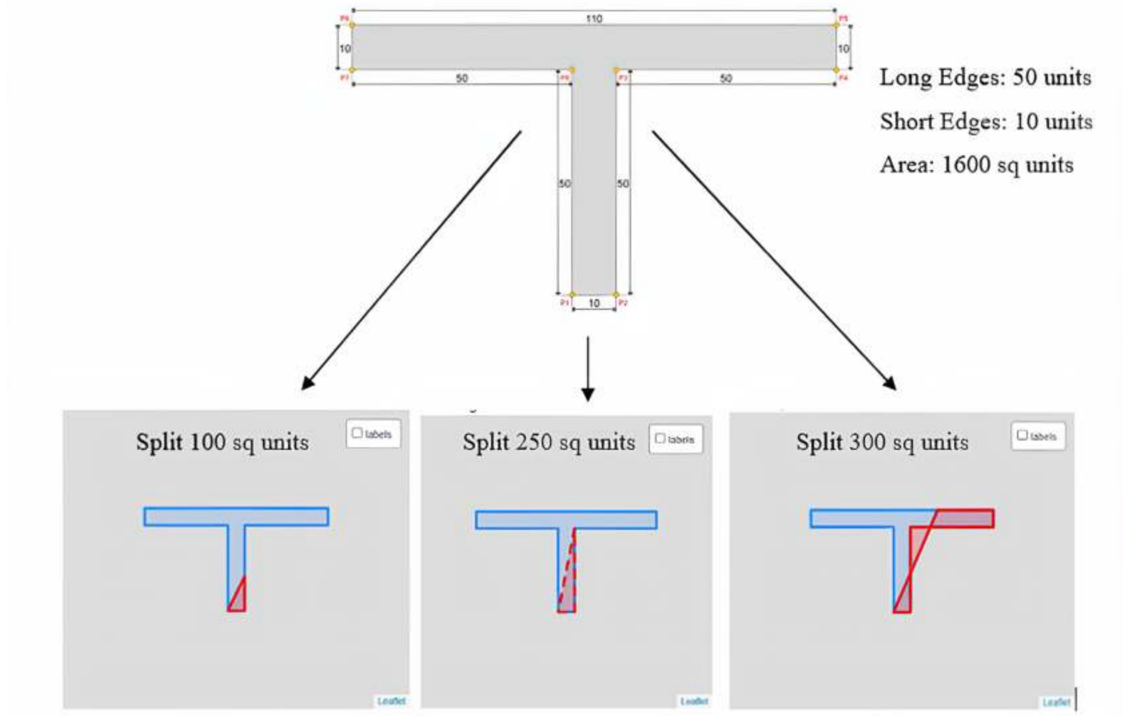
Source: Habib, 2020

Figure 2 Steps in One point and area method

1.2 Statement of the problem

In the One-Point-and-Area method, the Habib algorithm uses the SAS (Side-Angle-Side) formula for area calculation of convex-shaped polygons and modifies it to determine the distance d_1 from the vertex. However, when the partitioning line is passed through a concave section, the Habib algorithm does not give the required result. Splitting of the concave section in parcels was found to be suffering two issues, namely the issue of clinging and the issue of retracting.

Consider a clinging case of an axis-aligned parcel for example as in Figure 3. The triangle $P_1P_2P_3$ covers 250 sq. units., when it must be split by areas less than 250 sq. units such as 100, the resulting polygon was found to be clinging along line P_2P_3 as observed in Figure 3a. This might be the desired result in some cases, although the solution can be as in Figure 3d in other cases. On the other hand, for split areas greater than 250 sq. units such as 300, the resulting polygon was found to be retracting along line P_3P_4. This result was found incorrect for two reasons, first, the result was self-intersecting. Second, the result included a portion outside the original polygon. The expected solution for this case should be as presented in Figure 3e.



a) Clinging to the Concave section b) Inflection Polygon c) Retracting in the Concave section

d) Required Solution in Clinging e) Required Solution in Retracting

Figure 3: Problems in the Concave Section

1.3. Research objectives

The main objective of this project is to formulate an algorithm for the One-Point-and-Area method to accurately split an irregular-shaped polygon given a required area. In addition to this, the research is attempting to address the following research questions: -

1) How does the existing algorithm perform parcel subdivision in cases of irregular-shaped parcels, particularly the concave-shaped ones?
2) How can existing algorithms be improved to address the issues raised by irregular-shaped parcels, namely the clinging and retracting cases?

1.4 Significance of the study

In practice, while splitting parcels either using trial and error or direct method, two issues of high concern were found. When the trial-and-error method was followed, the professionals lacked the necessary tools due to the specific nature of splitting tasks. As already mentioned, this was normally time-consuming and susceptible to human errors. On the other hand, the direct method was also found to have its shortcomings. This method which employs trigonometric concepts, was found fast and accurate on normal convex parcels, it could not produce accurate results when the shape of the parcel was irregular. Apart from this, irregular-shaped parcels were also observed to bear legal and practical complexities such that the time and effort required in the trial-and-error method seemed justifiable. Hence, an alternative approach for sub-dividing irregular-shaped parcels was not found to be prioritized. In addition, to the knowledge of the author, very limited, if not no research has been done which could accurately split all the cases of irregular-shaped parcels. The authors believe that there is a necessity for such a method which can generate more accurate results efficiently than existing algorithms.

As such, the application of this research is expected to be significant for the following.

1) The proposed algorithm can accurately split both regular and irregularly shaped parcels.
2) In some specialized cases of irregular parcels where the retracting edge caused an error, the proposed algorithm removes those errors.

2. LITERATURE REVIEW

There a few research on parcel sub-division one of which is the "Proposed Algorithm of Land Parcel Sub-division" by Maan Habib (Habib, 2020). It focuses on the algorithm and steps for parcel subdivisions in land surveying and the legal change of property boundaries. The proposed algorithm provides a simple and relatively accurate solution for land partitioning using closed-form and direct procedures. The mathematical model is based on the coordinates of parcel vertices and specific constraints to fulfill site requirements. Another study, Zlatanova et al. (2014) related to land parcel sub-division is "SpaceSub-divisionTestbed: A Graphical Tool for Arbitrary Shaped 2D Polygon Sub-division" which provides methods and graphical tools for sub-division of both regular and arbitrary shapes (Cogo et al, 2023). The tool provides support for indoor applications and offers a practical approach to handling sub-division problems.

To the knowledge of the authors. scientific studies dealing with the automated partitioning of the process of parcel sub-divisions using vector data format are scarce. In an early work, Wakchaure (2001) created a standalone GIS tool to create a sub-division layout at the single parcel level for build-out analysis. The tool partitioned a parcel into subplots recommending a possible pattern for development, but it was not automatic, and the accuracy was also unreliable. In another research, Stevens et al. (2007) hinted at the development of an algorithm for automatically creating small land parcels which could be integrated into the vector-based City model of urban growth. An agent-based model of urban growth called Agent City, embedded a land sub-division module into its upgraded version of model (Jjumba & Dragićević, 2012). They stated that the module would initially divide the larger land parcel into city blocks and then the blocks into cadastral lots. Land Parceling System (LandParcelS) designed as a GIS-based software module during land consolidation integrated support system (LACONISS) for land planning and decision making (Demetriou et al, 2012). LandParcelS automates the process of parcelization, generating a set of new parcels that represent an alternative plan for land reallocation. The system generates new parcels based on optimization of their shape, size, land value, and road access. However, it is not capable of creating roads and is also inappropriate for subdividing parcels into city blocks and then city blocks into housing lots. Wickramasuriya et al. (2011) developed a GIS-based sub-division tool capable of generating urban sub-division layouts including both streets and lots. The tool carries out the partition of both rectangular and irregularly shaped parcels aiming to optimize the output by creating the highest number of lots and the lowest number of streets possible. Yet the model does not offer options for different sub-division styles and performs poorly in terms of shape and size of resultant lots adjacent to the boundary of irregular parent parcel. The tool cannot extend the road network if the candidate parcel is disjointed from the existing roads.

In addition, the authors found a few other procedural modeling tools that perform automatic polygon sub-division. These tools apply sub-division algorithms on a variety of input spaces. A circumscribed rectangle structure was used for diagonal polygon sub-division by Nikanorova and Romanovsky (Nikanorova & Romanovsky, 2020). Mkrttchian et al. (2023) proposed a trigonometric algorithm for subdividing non-convex polygons into sets of convex polygons. Vanegas et al. (2009) performed parcel sub-divisions using oriented bounding boxes. Dahal and Chow (2014) introduced a GIS toolset for the automatic subdivision of parcels. The toolset performs sub-division by using bounding boxes and contains special divisions for irregular T- and L-shapes. Adao et al. applied a rule-based approach using bounding boxes for the automatic sub-division of building interiors (Adao et al., 2014). They later extended this approach to irregular shapes in a tool by performing the fake-concave technique, which allows the user to specify parts of convex shapes as disposable parts for deletion after the sub-division is complete (Adao et al., 2014).

3. METHODOLOGY

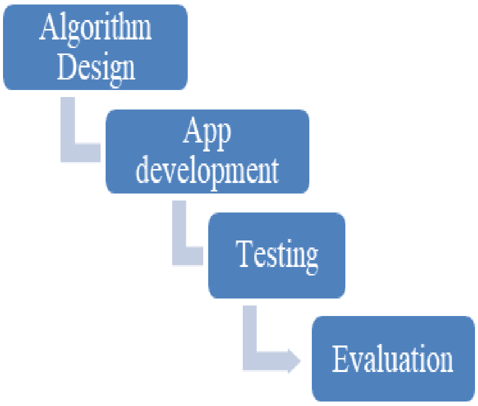


Figure 4: Workflow Diagram

This study modifies the existing Habib One-Point-and-Area algorithm to address the issues arising in subdividing irregular-shaped parcels. We add a guiding layer in the algorithm and upgrade the One-Point-and-Area metho, addressing the problems with clinging, retracting, and interior area calculation during the sub-division process.

3.1 Working model

The existing Habib algorithm was acquired from and reconstructed using HTML/CSS/Javascript.

3.2. Data

A dataset was created by collecting existing examples from the literature and creating new data. The data was formatted in GeoJSON format since it is the popular human readable industry standard (Fosci & Psaila, 2023).

3.3 Algorithm design

The Habib algorithm was found to be lacking a method to handle concave sections of parcels. It used two formulae namely SAS and Shoelace or Gauss to calculate areas which do not correctly account for retracting cases of concave section. SAS formula was found to include exterior area i.e. the area falls inside the triangle but outside the polygon whereas, Shoelace considered exterior area as negative and hence over-compensated the value while determining a new point. This is why the Habib Algorithm could not compute areas accurately upon using both formulae.

Hence, this research is focused on modifying Habib's One-Point-and-Area algorithm to accurately handle concave sections. The algorithm could be divided into three stages:

1) Collect input and Initialization by rearranging the vertices such that the input point was at the start of the polygon vertices list and the verticeswere arranged anticlockwise.
2) Iteratively refining to Find the Last Partition Polygon and determining the pair of vertices in which the required point lies.
3) Finally calculate the exact partition point and generate the desired polygon result.

The mentioned issues arose in the second stage as the iteration was guided by each edge respectively. The algorithm did not have any way to detect concave sections, and the issues were exacerbated as the said formulae for area calculation did not handle concave sections as well.

To solve this problem, three modifications were made. An additional stage to generate a convex hull was added after the first stage which was used during iteration to find the pair of vertices where the new point should lie. Another modification made was in the approach of calculating the area by improving it such that it only gave the interior area, thereby rectifying the errors made by SAS and Shoelace or Gauss formulae, finally the last stage to generate the desired polygon was done iteratively. Thus, we propose an improved version of the Habib algorithm that consists of four stages.

1) Same as Habib Algorithm

2) Generating a convex hull which guides the third process. If the start point did not lie on the convex hull, then it was added at the start of the convex hull.

3) Determining the pair of vertices upon which the required point was supposed to lie. For this, an iterative approach was used to form an initial partition polygon using the first three vertices of the convex hull. The intersection of the formed polygon and the original polygon gave internal area. If the area of the intersection was less than the required area, another vertex from the convex hull was appended to the end of the partition polygon, and the process was repeated.

4) Finally, the partition polygon was improved using the same technique as in the Habib algorithm. With the information on the LastPartitionPolygon and the area difference, the Habib algorithm was used to find a new vertex. The SAS formula included the exterior area, although only the interior area was necessary which made the exterior area greater than the interior area. As the area and distance were inversely proportional, the computed distance was minimized which meant the new point was still not correct. Hence, an iterative approach was employed to minimize the difference by appending the new vertex to LastPartitionPolygon and re-computing the interior area until the area was accurate.

3.4 Overview of algorithms

Habib One-Point-and-Area Algorithm	Proposed One-Point-and-Area Algorithm
1. Get input and rearrange vertices. 2. Iterate over vertices to find the last partition polygon. 3. Derive partition polygon	1. Get input and rearrange vertices. 2. Generate convex hull and identify vertices on convex hull. 3. Iterate over convex hull to find last partition polygon. 4. Iterate to derive partition polygon

Table 1 Overview of Algorithms

```
Algorithm: Proposed One-Point-and Area
Input:   {polygon: [list of coordinates of
         corners $P_0P_1P_2\ldots P_n$],
         $A_s$: Required Area,
         $P_s$: Start Point}
Output:  Partition Polygon
1:  Initialize
    1a.  Polygon = rearranges vertices such
         that $P_s$ is at start and vertices are
         anticlockwise,
2:  Make Convex Polygon
    2a.  Generate Convex Hull [$C_0C_1C_2\ldots C_n$]
    2b.  If $P_s$ not in Convex Hull, add $P_s$ in
         convex hull
    2c.  LastPartitionPolygon = [polygon of
         convex hull $P_sC_1C_2$],
         I = Intersection of
         LastPartitionPolygon and original
         Polygon
         $A_2$ = interiorarea of I,
         $P_{Last} = C_{2,}$
         $P_{Next} = C_3$
3:  Find Last Partition Polygon
    3a.  $A_1 = A_s - A_2$
    3b.  If $A_1 = 0$, I is required polygon, goto
         4g
           $A_1 < 0$, add next vertex from convex
         hull to LastPartitionPolygon. Update
         $A_2$, $P_{Last}$ $P_{Next}$. goto 3a
           $A_1 > 0$, goto 4
4:  Derive Partition Polygon
    4a.  $d_2$ = distance ($Ps$ , $P_{Last}$)
    4b.  $\Theta$ = angle $P_sP_{Last}P_{Next}$
    4c.  $d_1 = 2 * A_1 / (d_2 * \sin\Theta)$
    4d.  $X_{p'} = X_{plast} + (X_{pnext} - X_{plast})*d_1/$ distance
         $(P_{Next} - P_{Last})$
         $Y_{p'} = Y_{plast} + (Y_{pnext} - Y_{plast})*d_1/$ distance
         $(P_{Next} - P_{Last})$
    4e.  $P' = [X_{p'}, Y_{p'}]$
         PartitionPolygon = [$P_sC_1C_{2\ldots}$ $P_{Last}$ $P'$]
         I = Intersection of PartitionPolygon
         and original Polygon
         $A_3$ = interiorarea of I,
    4f.  $A_1 = A_s - A_3$
    4g.  If $A_1 = 0$, I  is required polygon
           $A_1 < 0$, goto 4c
```

Algorithm 1 Proposed One-Point-and-Area Algorithm

3.5 Software interface design

Both Habib and the proposed algorithm were designed and implemented in an HTML/CSS/Javascript application using Leafletjs and Turfjs on local server. The software had three map views: first view for presenting results from Habib algorithm, second one for results from proposed algorithm, the final one for overlapping results from both algorithms for visualization and differentiation. Test polygons were preloaded in a GeoJSON file which could be selected from the "Select Polygon" dropdown. After the selection of a polygon its total area was displayed, and the polygon vertices populated the "select point" dropdown menu. Selecting Point and entering required area (less than polygon area) and clicking "Perform Split" displayed the results in the map views. The map views were synchronized allowing the zoom or pan tasks on one of the views were synchronized to other views respectively.

3.6 Testing and analysis

As mentioned previously, the application was tested using a test polygon as a sample. For simplicity in area calculation and verification, a T-shaped axis aligned figure was used as a case of an irregular polygon. Being a geometrical figure, it was easy to verify the area and polygon as well as it also provided the concave sections. The shorter edges were 10m and the longer edges were 50m long. The WKT representation of the polygon was:

Polygon ((0 0, 0 50, -50 50, -50 60, 60 60, 60 50, 10 50, 10 0, 0 0))

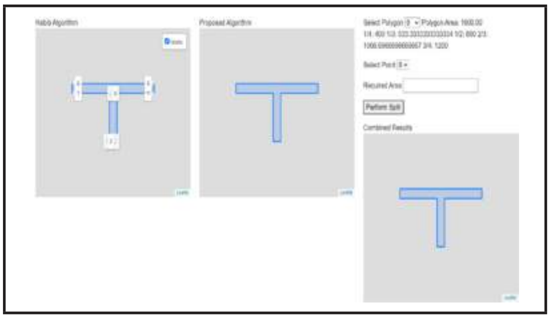


Figure 5: Software Interface

It consisted of 8 vertices and 8 edges with a total area of 1600 sq. m. and a perimeter of 340 m.

Case Study

Case 1: No clinging and no retracting

Taking the lower left point (1) as the selected point and the required area to be 40 sq. m., the outputs from the two algorithms were found to be the same and accurate (Figure 6) as concave section in this case only affected areas more than 47 sq. m

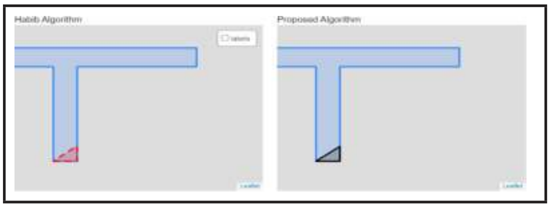


Figure 6 Output at Required Area 40 sq.m.

Case 2: Clinging Parcels

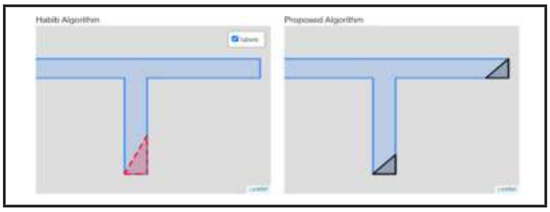


Figure 7: Output at Required Area 100 sq. m.

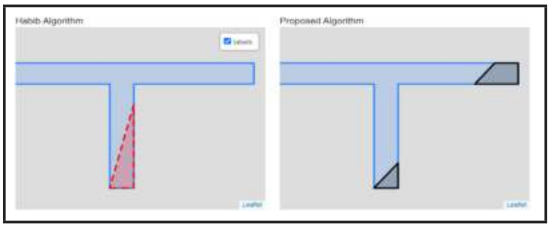


Figure 8: Output at Required Area 200 sq. m.

Again, taking the lower left point (1) as the starting point and this time area is 100 sq. m., the area computed by both algorithms was found to be different (Figure 7). The Habib algorithm could not remove the clinging, while the proposed algorithm successfully removed it.

In addition, if the required area was changed to 200 sq.m., then the proposed algorithm just followed correct the polygon edges.

Case 3: Retracting Parcels

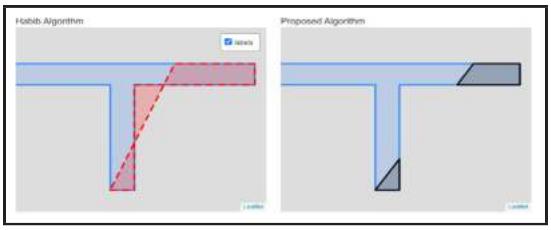


Figure 9: Output at Required Area 300 sq. m.

In this case, to split 300 sq.m, the Habib Algorithm could not acknowledge the concave section in contrast to the proposed algorithm (Figure 9). The output from the Habib Algorithm was found to be inaccurate since the polygon was intersecting with itself and part of it lay outside the original polygon. Since the result of the proposed algorithm was accurate, this case depicts the advantage of the proposed algorithm over the Habib Algorithm.

4. RESULT AND DISCUSSION

4. 1 Experiments and observations

Experiments were carried out on the sample polygons from the dataset with the required area for subdivision taken as 1/4th, 1/3rd, 1/2, 2/3rd, 3/4th of the parcel area as they were the most common in inheritance cases. Points were selected randomly, and consideration was made such that the sub-divisions were roughly equitable. To evaluate the difference in result, the angle made by the partition line with the parcel was computed. The base of the angle was the line joining selected point and the second vertex (first vertex is the selected point itself) on the convex hull of the parcel. Although the Habib algorithm did not require a convex hull, it was considered here to make it comparable with the proposed algorithm. All the angles were considered in degrees.

1) Selected point: 1, Selected Polygon: 2

Figure 10 displays clinging issue in a parcel with an inner cavity along vertices 15-21.

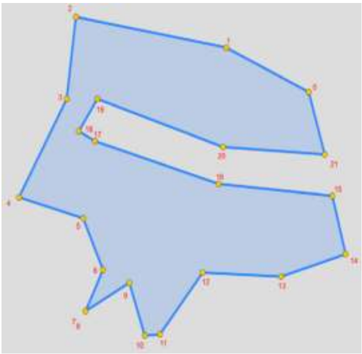
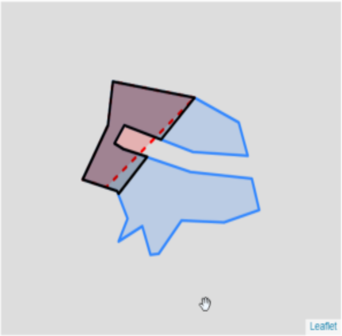
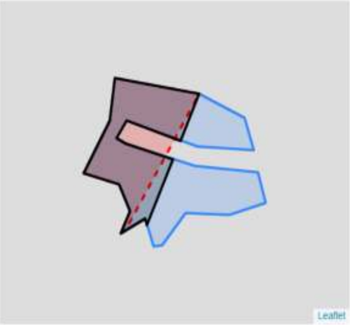
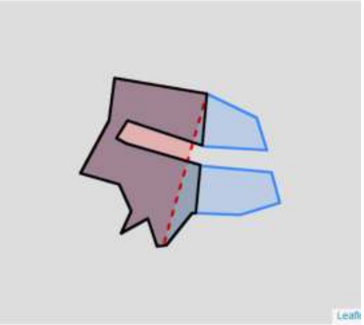
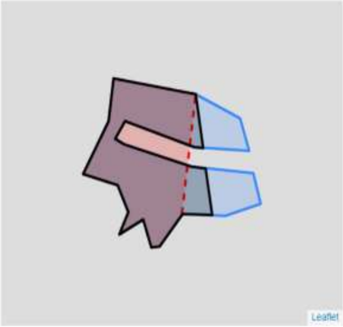
Original (1265.01 sq.m.)	One fourth (316.25)	One third (421.67)
		
Half (632.50)	Two third (843.34)	Three fourth (948.75)
		

Figure 10 : Subdivision on First Irregular Parcel

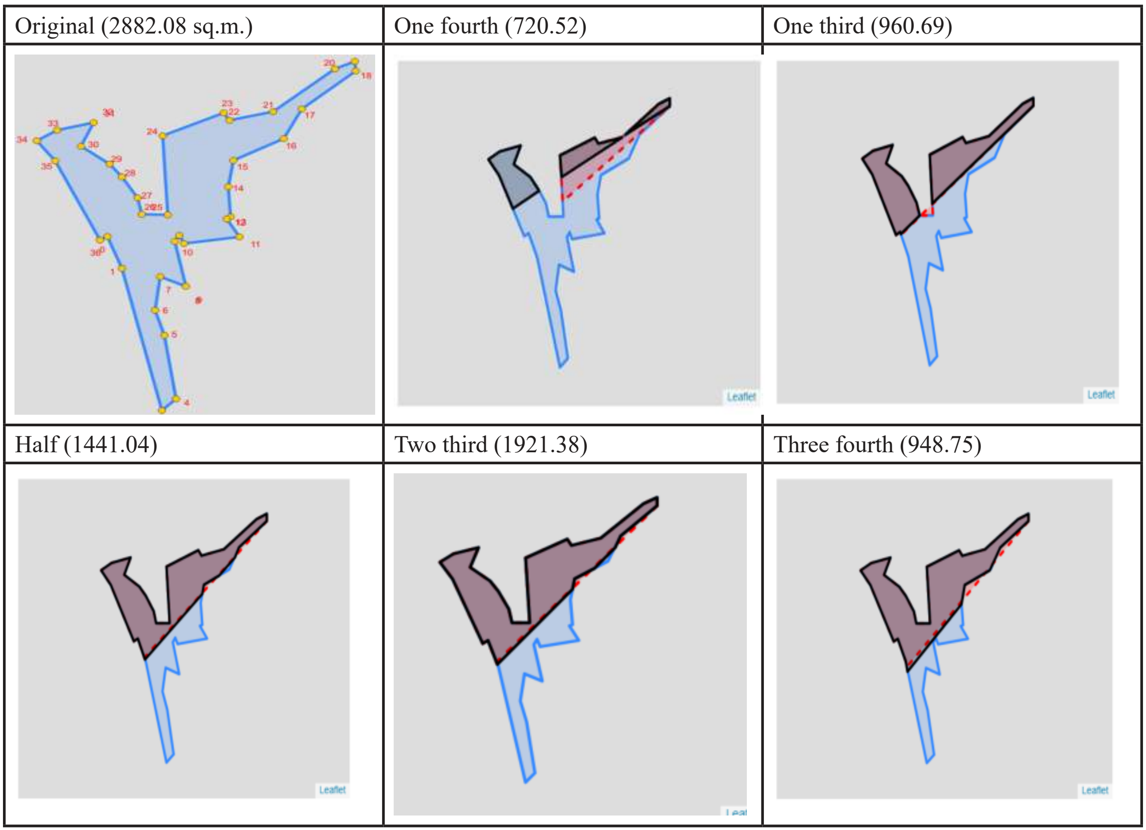
Area computation using Habib algorithm included portion inside this cavity in contrast to the proposed algorithm which successfully excluded it. This can be observed visually.

3) Selected point: 17, Selected Polygon:6

Parcel in Figure 11 contained much irregularity. The point for sub-division was selected such that both cases of clinging and retracting could be observed. On splitting 1/4th and 1/3rd of the parcel area respectively, clinging was observed because of which the partition line in Habib Algorithm moved forward. The retracting issue was seen in case of splitting half area of the parcel which is discussed in the section to come. Clinging along convex section were observed while splitting the 2/3rd and 3/4th area were, the partition line in Habib Algorithm remained backward since it included the exterior area as well. In contrast, the proposed algorithm compensated for this by moving forward.

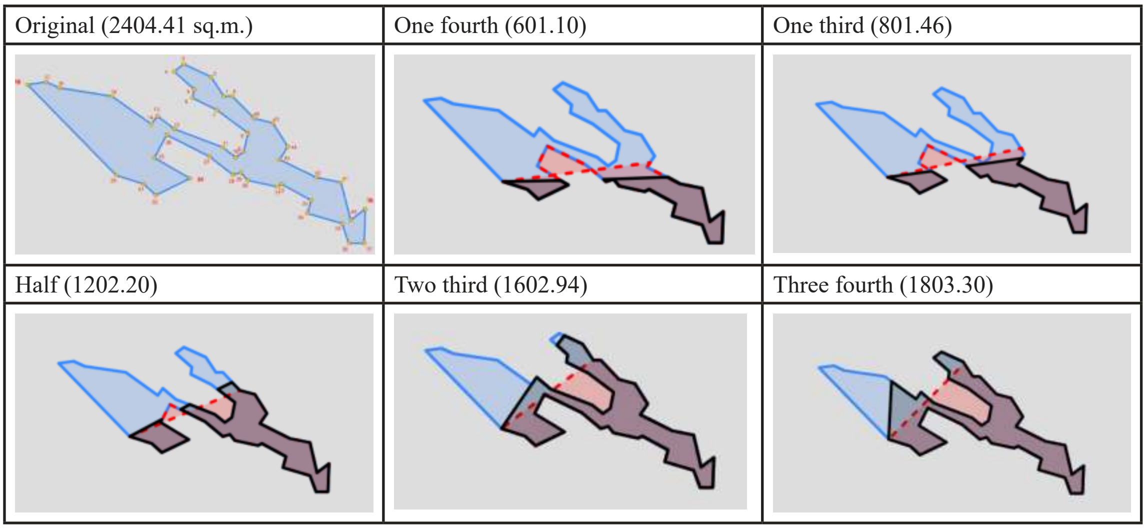
4) Selected point: 10, Selected Polygon: 8

In Figure 12, while splitting 1/4th and 1/3rd of parcel area, retracting issue was noticed. In these retracting cases, Habib algorithm considered the exterior area as negative since they were self-intersecting. This negative area was then compensated by forward moving partition point. In the case of splitting half of the parcel, both cases of retracting and clinging on convex section were detected. The exterior area of clinging was greater than the exterior area of retraction. This made partition point move forward making angle difference positive. Overall, it can be observed that the proposed algorithm addressed the clinging issue in both concave and convex sections as well as the retracting issue. It provided accurate results in all cases while Habib Algorithm could not accurately address the above issues



Original (2882.08 sq.m.)
One fourth (720.52)
One third (960.69)
Half (1441.04)
Two third (1921.38)
Three fourth (948.75)

Figure 11: Sub-division on Second Irregular Parcel



Original (2404.41 sq.m.)
One fourth (601.10)
One third (801.46)
Half (1202.20)
Two third (1602.94)
Three fourth (1803.30)

Figure 12: Sub-division on Third Irregular Parcel

4.2 Performance

For evaluating the performance, the Microsoft Edge browser (Version 120.0.2210.144 (Official build) (64-bit)) was used. All the dependencies were made local and other programs were terminated. For each reading, the browser was hard reset so that the variables did not stay in memory.

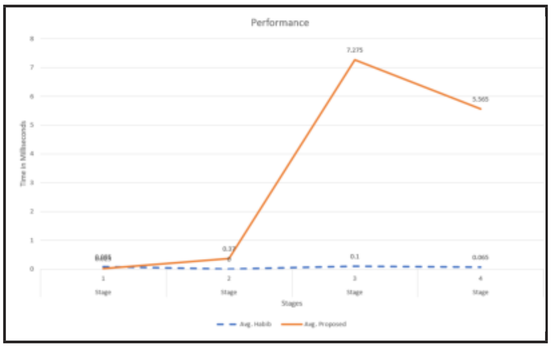


Figure 13 Average time taken in the operating stages for both algorithms.

It can be observed from the Figure 13 that the proposed algorithm, overall, was found slower than its counterpart. The time taken for the first stage of collecting input was found to be nearly similar in both cases. The second stage was not application to Habib algorithm as it completely lacked this stage. In the case of third and fourth stages, time taken by proposed algorithm was found significantly higher than that by Habib Algorithm since the former included an expensive intersection operation. Although, it was expected that the additional stage (2nd stage) would have much impact to the performance of proposed algorithm, it was observed that the highest effect was put forth by 3rd and 4th stage. In the fourth stage, Habib Algorithm performed singular execution of the "derive partition polygon" function whereas proposed algorithm executed it multiple times. Due to this, it was expected that this stage would have largest impact on increasing the computation time, however it was not the case. Overall, the third stage was detected to have the highest performance lag which was counter intuitive.

4.3 Limitation

This study has focused on solving the issues that arose while using the Habib algorithm for parcel splitting. Sub-division of multi-polygonal parcels (parcels with hollow) although has not been tested thoroughly in this research, we expect the concepts to apply to those issues as well. This research has used visualization as a method for evaluation. Other measures of polygon features like concavity, amplitude of vibration, and roundness have not been useful. Those measures compare singular polygons, but the results of the proposed algorithm are mostly multi-polygons. So, the comparisons are inconclusive.

5. CONCLUSION AND RECOMMENDATION

This research presents a modified version of the one-point-and-area method for parcel subdivision, particularly applicable for splitting irregular-shaped parcels. We have proposed to add an extra stage to generate a convex hull and use it to guide the iteration process. The area calculation formula computes the interior area and excludes the exterior area. The single-step process of deriving the partition point is kept in a loop because of the change in the formula of area.

The results from the experiments show that the final partition polygons generated are devoid of clinging and retracting issues. The algorithm handles clinging, retracting, and a combination of both issues. Although the proposed algorithm consumed more time, the quantitative increase in processing time was in milliseconds which is negligible. Normally, these algorithms are not used in bulk but are carried out on a single parcel. Due to this, compounding of increased computation time is not applicable. Hence, the algorithm can be s suitable for the One-Point-and-Area subdivision method.

While this study has limitations in performance, there is room for improvement. As a future work, in stage 3 for finding the "Last Partition Polygon", the initial guess of the last partition polygon can be improved based on the ratio of the required area to the original parcel area. Since stage 3 had taken the most time, authors believe that it can be improved significantly. Another is depending on the area difference if the required area is higher than half of the parcel area. The parcel vertices can be reversed, and the process be started from another direction. This trick has the potential to reduce computation by half in the worst cases. Multi-processing can be leveraged for the iteration of the third stage.

In the fourth stage for deriving a partition polygon, if iteration is to be done then the partition point derived in the last iteration can be removed as it has no effect on accuracy, but it takes computation resources. Intersection operation can also be limited to smaller parts of the original parcel, instead of the whole original parcel. This reduces the number of vertices that participate in the computation of intersection and reduces processing time.

Since this study is based on geometric properties and a Cartesian coordinate reference frame, this study can be implemented elsewhere relevant. Based on the results of the experiment and the discussions, the proposed algorithm is found to be an effective solution for parcel sub-division. It can handle all cases of parcel splitting using one point and including irregular parcels. The same concept can also apply to move-line-and-method as well.

REFERENCES

Adão, T., Magalhães, L., Peres, E., & Pereira, F. (2014). Procedural generation of traversable buildings outlined by arbitrary convex shapes. *Procedia Technology*, *16*, 310–321.

Cogo, E., Krupalija, E., & Rizvic, S. (2023). SpaceSubdivisionTestbed: A Graphical Tool for Arbitrary Shaped 2D Polygon Subdivision. *Proceedings of the 7th International Conference on Computer Science and Application Engineering* (pp. 1–5). doi:10.1145/3627915.3629605

Dahal, K. R., & Chow, T. E. (2014). A GIS toolset for automated partitioning of urban lands. *Environmental Modelling & Software*, *55*, 222–234. doi:10.1016/j.envsoft.2014.01.024

Demetriou, D., Stillwell, J., & See, L. M. (2012). *LandParcelS: A module for automated land partitioning*. School of Geography, University of Leeds.

Easa, S. M. (2008). Unified direct method for land subdivision: circular sides permitted. *Journal of Surveying Engineering*, *134*(2), 55–60. doi:https://doi.org/10.1061/(ASCE)0733-9453(2008)134:2(55)

Fosci, P., & Psaila, G. (2023). Soft Querying Features in GeoJSON Documents: The GeoSoft Proposal. *International Journal of Computational Intelligence Systems*. doi:10.1007/s44196-023-00325-3

Habib, M. (2020). Proposed algorithm of land parcel subdivision. *Journal of Surveying Engineering*, *146*(3), 04020012. doi:10.1061/(ASCE)SU.1943-5428.0000322

Jjumba, A., & Dragićević, S. (2012). High Resolution Urban Land-use Change Modeling: Agent iCity Approach. *Applied Spatial Analysis and Policy*, *5*(4), 291–315. doi:10.1007/s12061-011-9071-y

Mkrttchian, V., Revazyan, N. x, & Khachatryan, V. (2023). Sub division of the object into solid polygons and its applied and economic aspects. *Alternative*. Retrieved from https://api.semanticscholar.org/CorpusID:259034257

Nikanorova, M. Y., & Romanovsky, Y. R. (2020). On Subdivisions of Polygons. *Journal of Mathematical Sciences*, *251*, 524–530. Retrieved from https://api.semanticscholar.org/CorpusID:228884101

Stevens, D., Dragicevic, S., & Rothley, K. (2007). iCity: A GIS–CA modelling tool for urban planning and decision making. *Environmental Modelling & Software*, *22*(6), 761–773. doi:https://doi.org/10.1016/j.envsoft.2006.02.004

Vanegas, C. A., Aliaga, D. G., Benes, B., & Waddell, P. (2009). Visualization of simulated urban spaces: Inferring parameterized generation of streets, parcels, and aerial imagery. *IEEE Transactions on Visualization and Computer Graphics*, *15*(3), 424–435.

Wakchaure, A. S. (2001). An ArcView tool for simulating land subdivision for build out analysis.

Wickramasuriya, R., Chisholm, L. A., Puotinen, M., Gill, N., & Klepeis, P. (2011). An automated land subdivision tool for urban and regional planning: Concepts, implementation and testing. *Environmental Modelling & Software*, *26*(12), 1675–1684.

Zhang, P., Fan, J., Zhang, P., Zhang, Z., Chen, Z., & Han, L. (2021). Comparative Study on the Effect of Shape Complexity on the Efficiency of Different Overlay Analysis Algorithms. *IEEE Access*, *9*, 144179–144194. Retrieved from https://ieeexplore.ieee.org/abstract/document/9583235

Zlatanova, S., Liu, L., Sithole, G. & Zhao, J. (2014). Space subdivision for indoor applications. *GISt Report* 66. Doi. 10.13140/2.1.2914. 2081

Author's Information

Name	:	Amrit Karmacharya
Academic Qualification	:	Master of Science in Geospatial Technologies
Organization	:	Survey Department
Current Designation	:	Survey Officer
Work Experience	:	10 yrs.
No. Published paper/article	:	3