World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:18, No:2, 2024

The Algorithm to Solve the Extend General Malfatti's Problem in a Convex Circular Triangle

Ching-Shoei Chiang

***Abstract*—**The Malfatti's problem solves the problem of fitting three circles into a right triangle such that these three circles are tangent to each other, and each circle is also tangent to a pair of the triangle's sides. This problem has been extended to any triangle (called general Malfatti's problem). Furthermore, the problem has been extended to have 1 + 2 + … + n circles inside the triangle with special tangency properties among circles and triangle sides; it is called the extended general Malfatti's problem. In the extended general Malfatti's problem, call it Tri(T_n), where T_n is the triangle number, there are closed-form solutions for the Tri(T_1) (inscribed circle) problem and Tri(T_2) (3 Malfatti's circles) problem. These problems become more complex when n is greater than 2. In solving the Tri(T_n) problem, n > 2, algorithms have been proposed to solve these problems numerically. With a similar idea, this paper proposed an algorithm to find the radii of circles with the same tangency properties. Instead of the boundary of the triangle being a straight line, we use a convex circular arc as the boundary and try to find T_n circles inside this convex circular triangle with the same tangency properties among circles and boundary as in Tri(T_n) problems. We call these problems the Carc(T_n) problems. The algorithm is a mO(T_n) algorithm, where m is the number of iterations in the loop. It takes less than 1000 iterations and less than 1 second for the Carc(T_{16}) problem, which finds 136 circles inside a convex circular triangle with specified tangency properties. This algorithm gives a solution for circle packing problem inside convex circular triangle with arbitrarily-sized circles. Many applications concerning circle packing may come from the result of the algorithm, such as logo design, architecture design, etc.

***Keywords*—**Circle packing, computer-aided geometric design, geometric constraint solver, Malfatti's problem.

I. Introduction

IN this paper, we would like to propose an algorithm to find one, three, six, ten, …, all triangle numbers, with special tangency properties among these circles. Moreover, these circles are situated within a convex region enclosed by three circular arcs, and they are tangent to the boundary circular arc as indicated. Before we illustrate the problem, we introduce the related topics including the extended Malfatti's problem inside a triangle in this section.

Given a triangle, we want to find three circles inside the triangle, these circles tangent to each other, and every two circles tangent to one edge of the given triangle, this problem is called the Malfatti's problem. The problem generalized to any triangle is called the general Malfatti's problem. An instance of this general Malfatti's problem is shown in Fig. 1 [5]. And, these three circles are called the Malfatti circles.

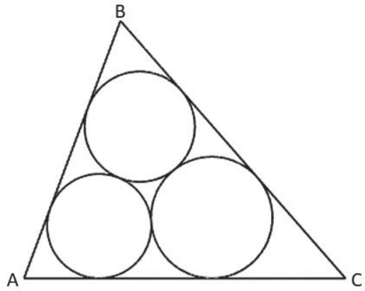


Fig. 1 Malfatti's problem

There are several known solutions of Malfatti's problem. Fukagawa and Pedoe [1] mentioned that the general Malfatti's problem on an arbitrary triangle was actually formulated and solved by Chokuen Ajima. The interested reader is referred to [2]-[4] for a history of the problem and an explanation of various solutions and generalizations.

Before extending the Malfatti's problem to involve more than three circles within the triangle, we require a more detailed specification of the tangency relationships among these circles and the sides of the triangle.

The triangle number T_n counts objects arranged in the following way: The first triangle number T_1 is one, represented by a dot, as shown in Fig. 2 (a). The n-th triangle number Tn has the form that Tn-1 on top, and following by one more row with n objects, as shown in Figs. 2 (b)-(d). T_n can be easily derived as $T_n = 1 + 2 + 3 \dots + n = n(n + 1) / 2$, where n = 1, 2, 3, …

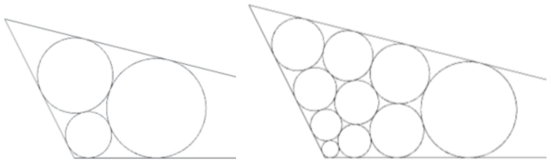
(a) $T_1 = 1$ (b)$T_2 = 3$ (c) $T_3 = 6$ (d) $T_4 = 10$

Fig. 2 Triangle number

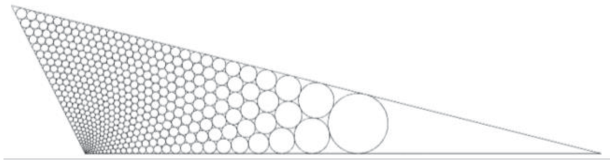
The general Malfatti's problem involves finding three circles for any given triangle. We aim to extend this problem to include more than three circles. Instead of three circles, we extend the number of circles from 3 to T_n, where n = 1, 2, 3, …, with tangency properties among circles and edge of triangles. When we consider T_n circles problem, it is called the Tri(T_n) problem, means the problem with T_n circles inside a triangle (see Fig. 3 for n = 2, 4, 31).

Ching-Shoei Chiang is with the Department of Computer Science and Information Management, Soochow University, Taipei 100, Taiwan (phone: +886-2-23111531 ext. 3801; fax: +886-2-23756878, e-mail: chiang@gm.scu.edu.tw).

The research of this paper is supported by MOST (Ministry of Science and Technology, Taiwan) grant MOST-111-2221-E-031-005.



(a) Tri(T_2) (b) Tri(T_4)

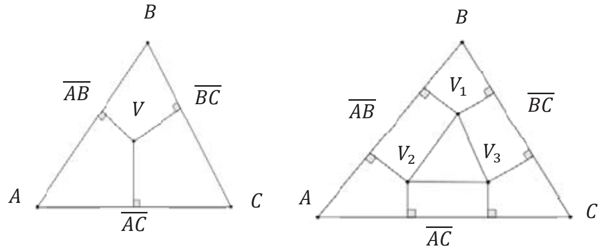


(c) Tri(T_{31})

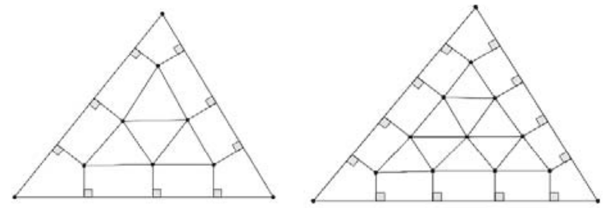
Fig. 3 Extended General Malfatti′s Problem Tri(T_n)

We consider the dots in Fig. 2 are vertices in a graph, we want to add edges to construct a special class of graph. The T_2 (n = 2) case in Fig. 2 (b) has three vertices, we add edges pairwise for these three vertices, as shown in Fig. 3 (b). There are only two rows, and one edge connect two vertices in the second row, and two edges connect the only one vertex in the first row with two vertices in the second rows. The T_3 cases in Fig. 2 have three rows, we use the same way to add edges for the first and second rows, and add edges for the second rows with the third row. There are two vertices in the second rows, and three vertices in the third rows. The first vertex at the second row connects to the first and second vertices at the third row and the second vertex at the second row connects to the second and third vertices at the third row. We can use the same approach to add the edges form T_n cases to T_{n+1} cases similarly. The graph for T_1 to T_6 cases is shown in Fig. 3. We call this graph a triangle graph for T_n. We notice that $T_n = T_{n-1}+n$, $T_1 = 1$ or $Tn = n(n+1)/2$, which is the vertices number of the triangle graph for T_n. The number of edges for the triangle graph for T_n, denoted E_n, is $E_n = E_{n-1}+3(n-1)$, $E_1 = 0$ or $En = 3n(n-1)/2$.

We consider the triangle graph for T_n, every dot represents a circle, and edge connecting two dots means these two circles are externally tangent to each other. We assume the triangle graph for T_n is inside a triangle, we want to specify the tangency properties for triangle graph for T_n and three sides of the triangle. We consider triangle graph for T_1 inside a triangle, it represents a dot(circle) inside a triangle, as shown in Fig. 4 (a). In this graph, there is a circle V inside the triangle, the edge connects V and perpendicular to $\overline{AB}$ means that circle V tangent to $\overline{AB}$ side of the triangle. So, the problem in Fig. 4 (a) indicates the problem of finding the inscribed circle of a triangle. When we change this graph from T_1 to T_2 inside the triangle, see Fig. 4 (b), we want to find three circles (represents by three dots V_1, V_2, V_3), these three circles tangency to each other represented by the edge connect these three vertices, and the circle V_1 tangents to edge $\overline{AB}$ and $\overline{BC}$, the circle V_2 tangents to edge $\overline{AB}$ and $\overline{AC}$, the circle V_3 tangents to edge $\overline{AC}$ and $\overline{BC}$. This is the malfatti's Problem. We can extend the problem, so that the graph inside the triangle from triangle graph for T_n to triangle graph for T_{n+1}, we established a problem from the circle number equal the triangle number T_n. to the circle number equal to T_{n+1}.



(a) Tri(T1) (b) Tri(T2)



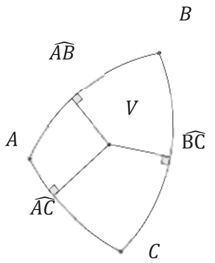
(c) Tri(T3) (d) Tri(T4)

Fig. 4 Tangency graph for Tri(Tn)

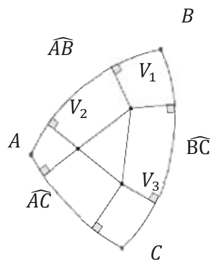
We call these graphs, which connect the triangle graph with triangle edges, a tangency graph for Tri(T_n). This tangency graph has a triangle graph for T_n inside the triangle and we need to add edges to specified the tangency property for circles, represented by dots in triangle graph for T_n, with sides of triangle. The tangency graph for Tri(T_3) and Tri(T_4) are shown in Figs. 4 (c) and (d). We classified these circles (represented by dots in a triangle) into three categories: corner circles, edge circles and inner circles. The circle at the first row (represented by dots in triangle graph), the first and last circle at the last rows are called the corner circles, and the corner circles always tangent to two side of the triangle. The first and the last circles for the second to (n-1) rows and the circles except the first and last circles in the last row are called the boundary circle, it is always tangent to one side of the triangle. The remainder circles are called the inner circles, which are not tangent to sides of the triangle. It is worth noting that all of these problems involve corner circles, and in cases where n is equal to or greater than 3, boundary circles are also included. And, the first inner circle starts from n = 4. The results [5], [6] of the Tri(T_n), n = 2,4,31 problems are presented in Figs. 3 (a)-(c), respectively.

II. Problem Statement

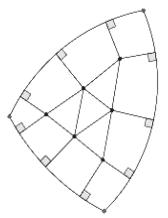
A circle divides an area into two parts, inside and outside of the circles. The inside region contains the center of the circle. For every circular arc, we call the "center side" of the arc the convex side of the circular arc (convex circular arc in short), and the other side the concave circular arc. This paper tries to find algorithms to solve similar problems for Tri(T_n), except that the area bounded the tangency graph for T_n is not a triangle but a convex circular triangle. We call these problems Carc(T_n) problems. The tangency graphs for Carc(T_n) are shown in Fig. 5. These graphs are similar to the tangency graph for Tri(T_n), as shown in Fig. 4. They have the same triangle graph for T_n, with a similar approach to connect the tangency properties between triangle graph and (convex circular) triangle edges.



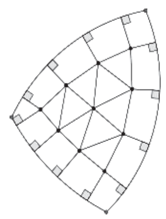
(a) Carc(T_1)



(b) Carc(T_2)



(c) Carc(T_3)



(d) Carc(T_4)

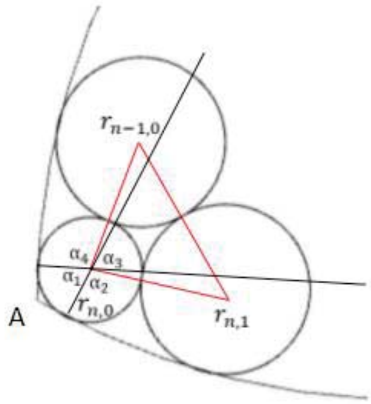
Fig. 5 The Carc(T_n) problems

The Carc(T_n) problems become more complex compared to Tri(Tn) problem, because the sides of the bounded region change from straight line to convex circular arc. We need more theorem and new algorithms to solve these tangency problems. These theorem and algorithms will be introduced in Sections III and IV. The experimental results will be shown in Section V.

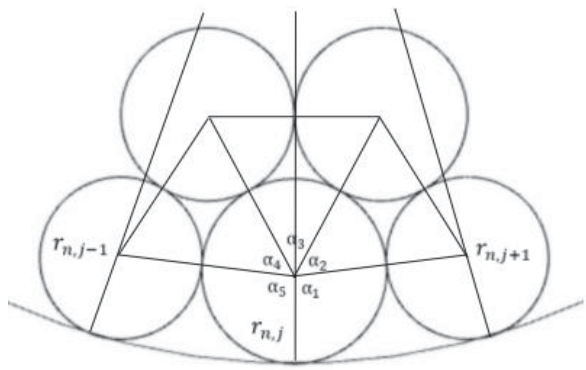
III. Theorem and Algorithm

In the algorithm we propose later, we give an initial set of radii for the circles, and enlarge/reduce its radius by its relation with its neighbors. In this process, we need to compute angles for different situation. We consider Fig. 5 (b), the center of the circle represented by dot V_1 emits four vectors to its neighbors, including the vectors to the center of circles represented by V_2, and V_3, projection point on boundary Carc $\widehat{BC}$ and boundary Carc $\widehat{AB}$. These four vectors produce four angles, and the sum of these four angles should equal to 2π. From the current radii of V_2 and V_3, and information form $\widehat{BC}$ and $\widehat{AB}$, the sum of these four angles is greater than (less than) 2π, we need to enlarge (reduce) the radius of circle V_1.

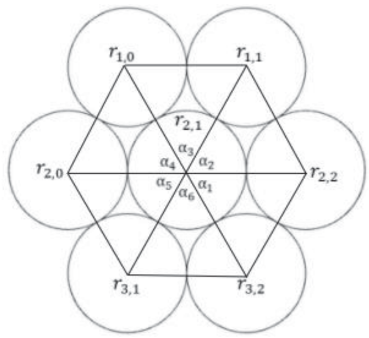
Now, we need to consider different criteria of angle computation. We consider three different kinds of circles inside three Carc, there are corner circles, boundary circles and inner circles, as shown in Fig. 6. We consider the inner circles, see Fig. 6 (c), we need to compute the angle α_i, i = 1,2,3,4,5,6. We consider the boundary circles, as shown in Fig. 6 (b), the way to compute α_2, α_3, α_4 is the same as we compute angle for inner circles, we need a way to compute α_1 and α_5 here. In the corner circle case, as shown in Fig. 6 (a), we need to find a way to find α_1.



(a) Corner Circle



(b) Boundary Circle



(c) Inner Circle

Fig. 6 Different circles in convex circular triangle

Let us denote the circle centered at v = (x,y) with radius r as C(v,r), we have the following three theorems:

Theorem1. Consider three circles $C(C_1,r_1)$, $C(C_2,r_2)$ and $C(C_3,r_3)$ tangent to each other externally, as shown in Fig. 7, then the angle $\angle C_3C_1C_2 = \varphi$ is:

$$\varphi = \cos^{-1}\frac{(r_1+r_2)r_1+(r_1-r_2)r_3}{(r_1+r_2)(r_1+r_3)} \quad (1)$$

Proof. We know the length of the triangle sides are r_1+r_2, r_1+r_3

and r_2+r_1, so the angle can be computed by the cosine theorem.

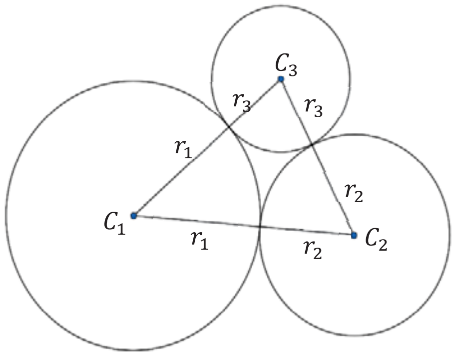


Fig. 7 Angle computation for inner circle

Theorem2. Consider two circle $C(C_1,r_1)$ and $C(C_2,r_2)$ tangent to each other externally, and these two circles' tangents to $C(v_a, r_a$ internally, as shown in Fig. 8, then the angle the angle β and γ are equal to:

$$\beta = \cos^{-1}\frac{(r_1-r_2)r_a-(r_1+r_2)r_1}{(r_1+r_2)(r_a-r_1)} \quad (2)$$

$$\gamma = \cos^{-1}\frac{{r_a}^2-(r_1+r_2)r_a-r_1r_2}{{r_a}^2-(r_1+r_2)r_a+r_1r_2} \quad (3)$$

Proof. Consider the triangle whose vertices are C_1, C_2, and C_3. The side length for the triangle is r_1+r_2, r_a-r_1, r_a-r_2, so we can find three angles for this triangle. β and γ can be computed by $\beta = \pi-\angle v_A C_1 C_2$, $\gamma = \pi-\angle v_A C_2 C_1$.

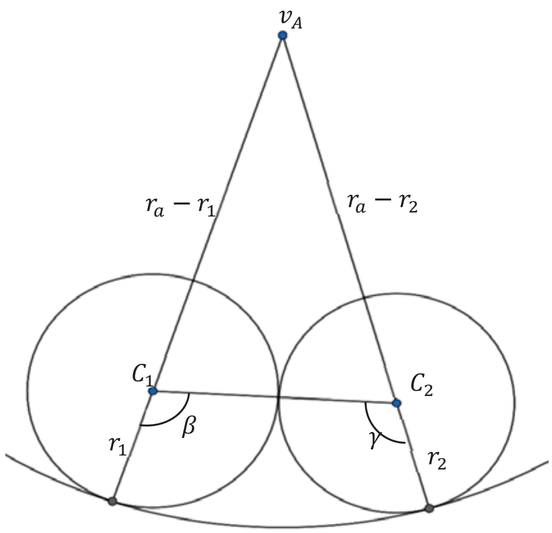


Fig. 8 Angle computation for boundary circle

Theorem3. Consider two circles $C(v_a, r_a)$ and $C(v_b,r_b)$), intersect at two points, we call one of these two points C, there is a circle tangent to these two circular Carc from the convex side center v and radius r, as shown in Fig. 9, then the angle α is equal to:

$$\alpha = \cos^{-1}\frac{(r_a-r)^2+(r_b-r)^2-\|\overline{v_a v_b}\|^2}{2(r_a-r)(r_b-r)} \quad (4)$$

Proof. Consider the triangle whose vertices are v_a, v_b and c, their side lengths are $\overline{v_a v_b}$, r_a-r, r_b-r. So, the angle $\alpha = \angle v_a v v_b$ can be found by the cosine theorem.

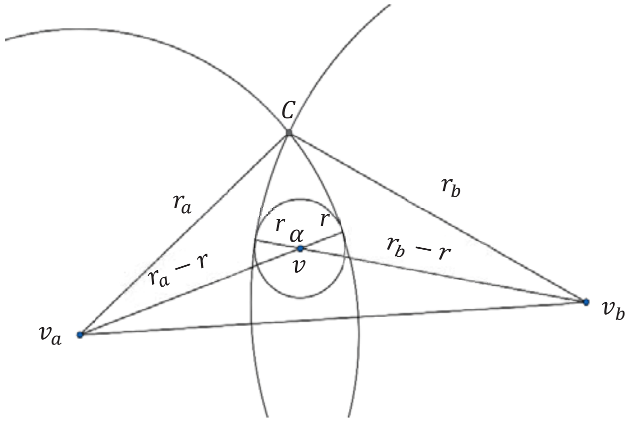


Fig. 9 Angle computation for corner circle

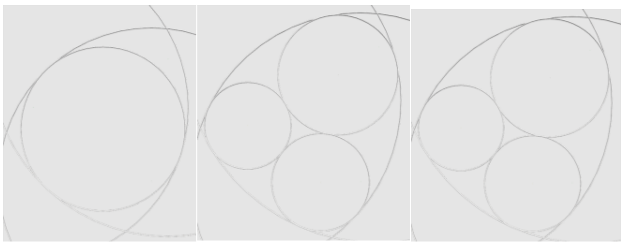
By knowing the radii for every circle C, we consider all vectors from its circle center to its neighbors' circle centers, or to the projection point on boundary circular arc. After sorting these vectors in order, we can compute every angle between two vectors via Theorem 1 to Theorem 3. We sum up all this angles, call it TA(C), and judge the enlarge/reduce the radius of this circle C. Using this idea, we have the following algorithm:

Algorithm:
1. Input information for three circles v_a, r_a, v_b, r_b, v_c, r_c, and n.
2. Enumerate circles, set initial radius $r_i = 1/(n+1)$, where i = 1,2,….,T_n.
3. Compute $TA(C_i)$ for all circles.
4. While one of $TA(C_{i,})$ is not equal 2π (within a tolerance).
   4.1 Calculate the enlarge/reduce amount for the radius of $C_{i,}$ (enlarge or reduce) if sumangle $(C_{i,j})>2\pi$ (or $<2\pi$).
   4.2 Enlarge/reduce all radii in the same time.
   4.3 Enlarge/reduce so that the result figures fit in the region bounded by three convex Carc.
   4.4 Compute $TA(C_i)$ for all circles.
5. Draw the result.

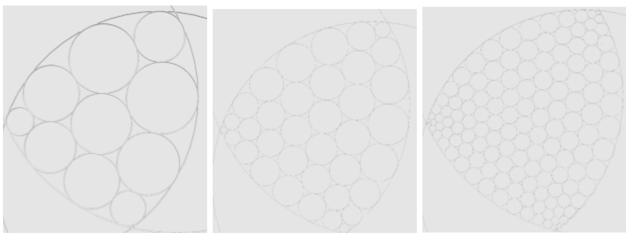
At first (step 1), we can find three convex Carc from the three circles, and find the associated three vertices from the intersection point of any pairs of circular arc. From the input integer n (n > 1), we can generate information (radii) for circles for all T_n circles (step 2). Then we compute $TA(C_i)$ (by using Theorem 1 to 3) for all circles (step 3). We use a while loop (step 3) to repeat many iterations until all circles have the properties $TA(C_i) = 2\pi$. Inside the while loop, the radii of all circles have been enlarged/reduced, depending on their $TA(C_i)$. After that, we give one more constraint to enlarge/reduce the radii of all circles with one common ratio, so that this group circles fit into the convex region bounded by three Carc (step 4.3). After the end of the while loop, the only thing left is the display of the result.

IV. Implement and Examples

We use Python 3 to implement the above algorithm in a PC (Intel Core i7-10750H CPU), and test for the problem Carc(T_n), n = 1, 2, 3, 4, 8, 16, … for the convex area inside three circles C((11,0),15.9612), C((12.72,12.69),18.5) and C((-1.3,7.65),16.1327). The results are shown in Fig. 10 and the circle information in Table I.



(a) n = 1 (b) n = 2 (c) n = 3



(d) n = 4 (e) n = 8 (f) n = 16

Fig. 10 Solution for Carc(T_n)

TABLE I
THE PERFORMANCE FOR THE PROPOSED ALGORITHM

n	Time (sec)	Iter.	Conner	Boundary	Inner
1	0.262	0	8.553	X	X
2	0.001	32	4.098, 4.611, 5.666	X	X
3	0.004	62	2.249, 2.713, 3.677	(3.534, 4.283)	X
4	0.008	104	1.373, 1.736, 2.491	(2.527, 3.523)	2.981
8	0.079	329	0.357, 0.501, 0.785	(0.791, 2.044)	(1.115,1.800)
16	0.796	915	0.083, 0.127, 0.206	(0.192, 1.114)	(0.293,1.020)

In Table I, the first column indicates the value n for Carc(n) problem, the second column is the execution time (seconds) for associated problem. The third column is the number of iterations in the while loop of the proposed algorithm. There are only corner circles for Carc(T_1) and Carc(T_2) cases, and it starts to have inner circles from Carc(T_4). There are three corner circles for all Carc(T_n) problem, and column four indicates the radii for these three circles. The fifth column to sixth column indicate the radius or radii range for boundary and inner circles in the Carc(T_n), n = 3,4,8 problems.

V. Conclusion and Future Research

The Carc(T_n), n ≥ 1, problem can be constructed in the proposed algorithm in this paper. It seems that the range of the radius for a boundary circle is wider than the range of the radius for the inner circle. The CPU time is mainly used for displaying the figure, as the computation for the Carc(T1) problem using an O(1) algorithm takes almost the same time compared to the others. We proposed the algorithm to solve Carc(T_n). The boundary convex arc is a circular arc, which represents a curve of degree 2. This problem can be easily extended to concave Carc. There are many different types and/or different degree of boundary curve, such as Bezier curve, which can be further investigated.

Acknowledgment

Thanks to Min-Hsuan Hsiung, Fan-Ming Chiu, Hung-Chieh Li, Guan Wen Wang and Hsi Wen Wang for their assistance to implement the algorithm of this paper.

References

[1] Fukagawa, H. and Pedoe, D. "The Malfatti Problem." Japanese Temple Geometry Problems (San Gaku). Winnipeg: The Charles Babbage Research Centre, pp. 28 and 103-106, 1989.
[2] O. Bottema, “The Malfatti Problem,” Forum Geometricorum 1, 43-50, 2000.
[3] M. Stefanović. “Triangel centers associated with the Malfatti circles.” Forum Geometricorum 3,83-93, 2003.
[4] Wolfram MathWorld. “Malfatti Circles,” http://mathworld.wolfram.com/MalfattiCircles.html
[5] Ching-Shoei Chiang, Christoph M. Hoffmann, Paul Rosen, “A gereralized Malfatti’s Problem”, Computational Geometry, Volume 45, Issue 8, October 2012, pages 425-435.
[6] Ching-Shoei Chiang, Hung Chieh Li, Min-Hsuan Hsiung, and Fan-Ming Chiu, “Extended General Malfatti’s Problem, The 19th International Conference on Scientific Computing, 2021/7/26-29.