algorithms

MDPI

Article

Extended General Malfatti's Problem

Ching-Shoei Chiang

Department of Computer and Information Science, Soochow University Taipei, Taiwan 100, China; chiang@gm.scu.edu.tw

Abstract: Malfatti's problem involves three circles (called Malfatti circles) that are tangent to each other and two sides of a triangle. In this study, our objective is to extend the problem to find 6, 10, ... $\sum_1^n i$ (n > 2) circles inside the triangle so that the three corner circles are tangent to two sides of the triangle, the boundary circles are tangent to one side of the triangle, and four other circles (at least two of them being boundary or corner circles) and the inner circles are tangent to six other circles. We call this problem the extended general Malfatti's problem, or the Tri(T_n) problem, where Tri means that the boundary of these circles is a triangle, and T_n is the number of circles inside the triangle. In this paper, we propose an algorithm to solve the Tri(T_n) problem.

Keywords: Malfatti's problem; geometric constraint solver; computer-aided geometric design; circle packing; data visualization

1. Introduction

Malfatti's problem, as originally stated in 1803, was used to carve three circular columns from a triangular prism of marble to minimize waste. Equally, the problem poses the fitting of three circles into a given triangle such that the sum of their areas is the maximum possible. Malfatti erroneously believed that the solution was provided by three inscribed circles that are tangent to each other and tangent to the sides of the triangle. An instance of this problem is shown in Figure 1.

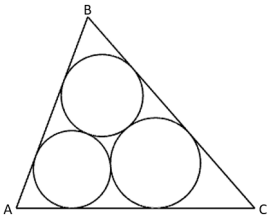


Figure 1. Malfatti's problem.

Since then, this second problem (circle tangency problem), which Malfatti believed maximizes the area (maximum area problem), has become understood as Malfatti's problem in the literature. There are several known solutions to Malfatti's problem. Fukagawa and Pedoe [1] mentioned that the general Malfatti's problem for triangles was actually formulated and solved by Chokuen Ajima. Readers should refer to [2–4] for a history of the problem and an explanation of its various solutions and generalizations.

Malfatti's problem involves finding three circles for an arbitrary triangle, and our objective is to extend the number of circles from three circles to $T_n = \sum_1^n i$, $n = 1, 2, \ldots$. circles inside the triangle, with tangency properties among the circles and three sides of the triangles. We refer to the problem of T_n circles as the Tri(T_n) problem. Under this definition, the general Malfatti's problem is the Tri(T_2) problem, and the Tri(T_1) problem asks us to find the inscribed circle of a triangle.

There are four sections in this paper. Sections 1 and 2 illustrate Malfatti's problem as well as the extended Malfatti's problem. Section 3 provides a definition and theorem to

check for updates

Citation: Chiang, C.-S. Extended General Malfatti's Problem. *Algorithms* **2024**, *17*, 374. https://doi.org/10.3390/a17080374

Academic Editor: Binhai Zhu

Received: 29 June 2024
Revised: 20 August 2024
Accepted: 20 August 2024
Published: 22 August 2024

Copyright: © 2024 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

support the proposed algorithms to solve the Tri(T_n) problem. The algorithm and execution result are illustrated in Section 4. Section 5 describes the implementation of this problem in more detail, as well as its potential applications. Section 6 presents our conclusion.

2. Problem Description and Algorithm to Solve Tri(T_n) Problem

In this section, we provide some terms to simplify the description of this problem (Section 2.1) and analyze the problem algebraically (Section 2.2).

2.1. Problem Description

In this paper, we define a triangle with three vertices, A, B, and C, and its associated angles are ∠A, ∠B, ∠C. The side lengths, opposite to the vertices A, B, and C, are a, b, and c, respectively. Furthermore, we define the coordinates for these three vertices, A = (x_1,y_1), B = (x_2,y_2), and C = (x_3,y_3), in a Cartesian coordinates system.

We aim to find T_n circles inside this triangle with tangency properties, as shown in Figure 2. In this figure, $r_{i,j}$ denotes the radius of the circle, and its associated circle is $C_{i,j}$. In the Tri(T_1) problem, our objective is to find a circle inside a triangle and tangent to three sides of the triangle, which involves finding the incircle of the triangle. The Tri(T_2) problem is identical to Malfatti's problem. In the Tri(T_3) problem, the aim is to find six circles; their tangency properties are shown in Figure 13b. We can similarly define the Tri(T_4) (as shown in Figure 13c) and the Tri(T_n) problems (as shown in Figure 2).

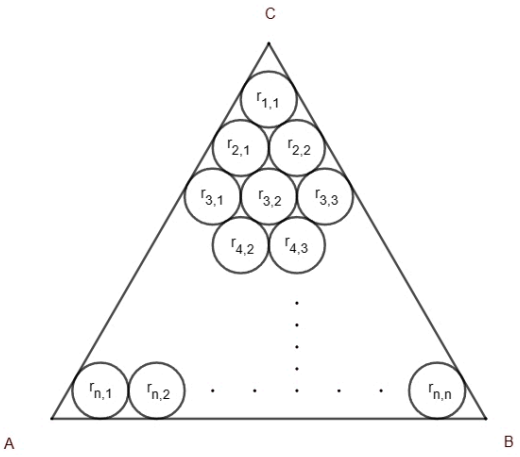
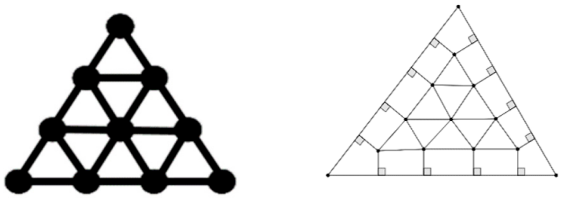


Figure 2. The tangency property of the Tri(T_n) problem.

In Figure 2, among these $T_n = \sum_1^n i$ circles, we define the circles that are tangent to two sides of the triangle as the corner circles and the circle that is tangent to one side of the triangle as the boundary circle. The circles that are not tangent to any side of the triangle are called the inner circles. Note that all problems except Tri(T_1) have three corner circles ($C_{1,1}$, $C_{n,1}$, $C_{n,n}$) and have 3(n − 2), n > 1 edge circles. There are (n − 2)(n − 3)/2, n > 2 inner circles. It should also be noted that there is no inner circle when n = 1, 2, 3 and no boundary circle when n = 1, 2.

There are tangency relations among all circles inside the triangle, as shown in the tangency graph for all circles (see Figure 3a). In the graph for the circles in the Tri(T_n) problem, the vertices represent circles, and the edge connecting two vertices represents the external tangency relation between these two circles. The circle $C_{i,j}$, $i = 2, \ldots, n$, $j = 1, \ldots, i-1$, (with radius $r_{i,j}$) is tangent to the next circles $C_{i,j+1}$ in the same row, and tangent to the circle in the next row $C_{i+1,j}$ and $C_{i+1,j+1}$, for $i = 1, \ldots, n-1$, $j = 1, \ldots, i$ in the next row. Figure 3a is an example of the tangency relations for $T_4 = 10$ circles inside the triangle. Besides the tangency property among circles, some circles are also tangent to the sides of the triangle. To add the tangency relation between the circle and the side of the triangle, we added a triangle to surround these circles and an edge to connect the vertex that represents a circle to the sides of the triangle; an example for T_4 is shown in Figure 3b.

Note that the tangency graph is a helpful tool for knowing the tangency relation among the circles and sides of the triangle, but it is not a solution to our problem because we do not know the circle centers and radii for all circles yet. Therefore, the Figure 3b is a tangency graph for the Tri(T_4) problem. And, given the coordinates of the three vertices of a triangle, the associated solution is shown in Figure 13c.



(**a**) (**b**)

Figure 3. The tangency graph among circles and triangle edge. (**a**) Inside of the triangle, (**b**) With the triangle edge.

2.2. Algebraic Analysis of the Problem

We can approach Malfatti's problem (see Figure 1) in the following way:

1. Let $C_i = (x_i,y_i,r_i)$ represent a circle whose center and radius are (x_i,y_i) and r_i, respectively; $i = 1, 2, 3$. L_1, L_2, L_3 are three lines for the triangle. Circle C_1 is tangent to lines L_2 and L_3. Circle C_2 is tangent to lines L_1 and L_3, and Circle C_3 is tangent to lines L_1 and, L_2.
2. Generate the equations for the "circle/side" tangency relation carefully. For example, circle C_3 is tangent to line L_1. The equation can be generated by computing the distance from (x_3,y_3) to line L_1 is equal to r_3. The line equation we generate, $ax + by + c = 0$, has the properties that all points (x_i,y_i) inside the triangle have the property that $ax_i + by_i + c > 0$, $i = 1, 2, 3$ (detailed explained below). Therefore, we generate six equations because every circle is tangent to two sides of the triangle. Note that the equations are degree-one equations.
3. Generate the equations for the "circle/circle" tangency relation. For example, circle C_1 is tangent to C_2. The equation can be generated by computing the distance between two centers, which is equal to the sum of their radii. There are three equations generated in this case. Note that when we want to eliminate the square root of the distance function, it becomes a degree-two polynomial.

We generate six degree-one (details explained below) and three degree-two polynomials. It probably can be solved by a Computer Algebra package, such as Maple [5] or SymPy [6], in this case. However, When n increases, the number of equations grows quadratically. Using SymPy to implement the above approach, it is easy to solve the Tri(T_1) problem because we generate three linear equations with three variables. However, when we go to the Tri(T_2) problem, we generate six linear equations and three quadratic equations. The program runs for a long time without a solution or any error message. Therefore, we stop using a symbolic approach to solve the Tri(T_n) problem when $n > 1$.

Regarding how to generate the degree-one equation in step two, we use a directed line, or ray, to generate the equation. Let $v_1 = (x_1,y_1)$, $v_2 = (x_2,y_2)$, $v_3 = (x_3,y_3)$ be the vertices of the triangle, rotated counterclockwise. Consider the ray from v_1 to v_2, the equation for this ray is $ax + by + c = 0$, where $a = y_1 - y_2$, $b = x_2 - x_1$, $c = x_1y_2 - x_2y_1$. Using this formula, the ray from v_2 to v_1 will be $-ax - by - c = 0$. Furthermore, the points (x_3,y_3) on the left part of the ray have the property $ax_3 + by_3 + c > 0$. Both rays from v_1 to v_2 and from v_2 to v_1 represent the same line. The equation for the constraint with the circle center (x_3,y_3) and radius r_3, which is tangent to the ray $ax + by + c=0$, is $ax_3 + by_3 + c - r_3\sqrt{a^2 + b^2} = 0$. This is because a, b, c are real numbers; this is a degree-one equation with variables x_3, y_3, and r_3.

In the Tri(T_n) problem, we have $3T_n$ variables. There are 3n equations generated by the "circle/side" tangency relation and the $3T_{n-1}$ "circle/circle" tangency relation. If we want

to solve the Tri(T_3), Tri(T_4), Tri(T_8), and Tri($_{30}$) problems, for which the solutions are shown in Figure 13, the system of equations will have 18, 30, 108, and 1395 equations. Based on the quadratic growth of the number of equations and our experience regarding the Tri(T_2) problem, this is not an effective way to solve the problem. We searched for a mathematical solution from other researchers for Tri(T_1) and Tri(T_2) and proposed an algorithm to solve the Tri(T_n) problem Tri(T_n) for n > 2.

3. Definition and Theorem for the Tri(T_n) Problem

Let us start with the Tri(T_1) and Tri(T_2) problems. These two problems can be solved directly using a mathematic equation, as described in Sections 3.1 and 3.2.

3.1. The Tri(T_1) Problem

In this problem, we aim to find a circle tangent to three sides of a triangle. We can use the formula for the inscribed circle to solve the problem. The formula is

Theorem 1 ([7]). *The inscribed center is at (x,y) = ((aA + bB + cC)/(a + b + c) with radius* $r = \sqrt{\frac{(s-a)(s-b)(s-c)}{s}}$, *where s = (a + b + c)/2.*

Considering a = b = c or a = b, it is easy to establish the following corollaries (evidence in Theorem 1):

Corollary 1. *Consider an equilateral triangle with side length a, its inscribed circle centered at (A + B + C)/3 with radius* $\frac{a}{2\sqrt{3}}$.

An isosceles triangle is a triangle that has at least two sides of equal length. Let us call these two sides the legs of the isosceles and the other side the base of the triangle, then

Corollary 2. *Consider an isosceles triangle whose leg and base lengths are a and c, respectively. Its inscribed circle is centered at (x,y) = [a(A + B) + cC]/(2a + c) with radius* $r = \frac{c}{2}\sqrt{\frac{2a-c}{2a+c}}$.

3.2. The Tri(T_2) Problem

We aim to find three circles inside a triangle that are tangent to each other pairwise, and every circle is tangent to two sides of the triangle.

There are formulae to solve this problem. These formulae were discovered by Malfatti and published in 1811 [8].

Theorem 2 ([3,8]). *The radii of the Malfatti circles are*

$$r_1 = \frac{r}{2(s-a)}(s - r + d - e - f) \quad (1)$$

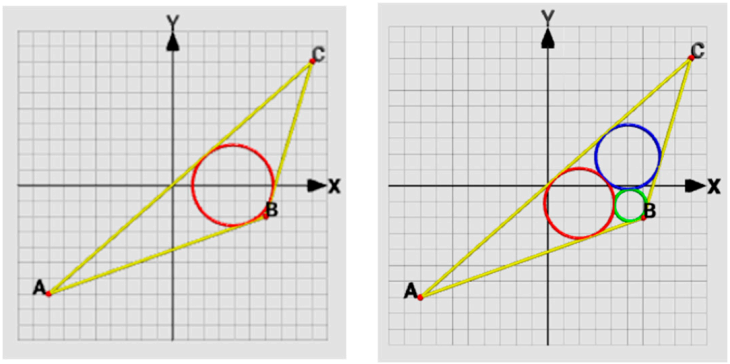
$$r_2 = \frac{r}{2(s-b)}(s - r - d + e - f) \quad (2)$$

$$r_3 = \frac{r}{2(s-c)}(s - r - d - e + f) \quad (3)$$

where s = (a + b + c)/2, $r = \sqrt{\frac{(s-a)(s-b)(s-c)}{s}}$ *is the radius of the inscribed circle, and d, e, and f is the distance from the center of the inscribed circle to A, B, and C, respectively.*

The values r_1, r_2, and r_3 are the radii of the circles near vertices A, B, and C, respectively. Notice that when we find the radii of these three circles (for example, r_1), the center of these circles is easy to find from the location of the vertex (A), angle bisectors (of ∠A), and radius (r_1). We provide a simple example for finding the inscribed circle and Malfatti's circle.

Example 1. *Consider a triangle whose vertices are A = (−8,−7), B = (6,−2), and C = (9,8). From theorem 1, we find the inscribed center (x,y) = (3.88, 0.01), with a radius of 2.61, as shown in Figure 4a. The centers of three Malfatti circles located at (1.99, −1.01), (5.18,−1.22), and (5.03, 1.80) with radii 2.29, 1.01, and 2.02, respectively, are shown in Figure 4b.*



(**a**) (**b**)

Figure 4. Solution for the Tri(T_1) and Tri(T_2) problems. (**a**) Inscribed circle. (**b**) Malfatti circle.

Using a = b = c or a = b, we establish the following corollaries (proof by Theorem 2):

Corollary 3. *The radii of the three Malfatti circles for the equilateral triangle with side length a have the same value* $r_1 = r_2 = r_3 = \frac{3-\sqrt{3}}{2} r = \frac{\sqrt{3}-1}{4} a$.

Corollary 4. *Consider an isosceles triangle whose leg and base lengths are a and c, respectively. Its inscribed circle is centered at (x,y) = [a(A + B) + cC]/(2a + c) with radius* $r = \frac{c}{2} w$, *where* $w = \sqrt{\frac{2a-c}{2a+c}}$, *The three Malfatti circles for this triangle have radii* $r_1 = r_2 = (2a + c)w(1 - w)/4$ $r_3 = c[\frac{1}{4w} + \frac{w^2}{4} - \frac{c}{2a+c}\sqrt{\frac{a}{2a-c}}]$.

Note that the variables r, d, e, f can be represented by the constant w where $r = \frac{c}{2} w$, $d = e = \sqrt{\frac{a}{2a-c}} cw$, f = aw, and when a = c, the result for Corollary 4 is the same as that for Corollary 3.

When the angle opposite to the base is 2θ, we have sin(θ) = c/2a. Assume that $T = \tan(\frac{\theta}{2})$, Corollary 4 can be rewritten as

Corollary 5. *Consider an isosceles triangle whose leg and base lengths are a and c, respectively. The angle opposite to the side with length c is 2θ. The three Malfatti circles for this triangle have radii* $r_1 = r_2 = s\frac{2T(1-T)}{1+T^2}$, $r_3 = \frac{c}{2(1+T)^2(1-T)}\left(3T^2 - 2\sqrt{2}T\sqrt{1+T^2} + 1\right)$ *where s = (2a + c)/2,* $T = tan(\frac{\theta}{2})$.

Let us give an example of finding the Malfatti circles of an isosceles triangle and observe the result.

Example 2. *Consider an isosceles triangle whose leg and base lengths are l = 10 and b = 12, respectively. The radii for the Malfatti circles are* $r_1 = r_2 = 2.0$, $r_3 = 1.718$, *as shown in Figure 5a. Let (l,b) = (10, 8), the radii for the Malfatti circles are* $r_1 = r_2 = 1.583$, $r_3 = 1.827$, *as shown in Figure 5b. We observe that when the length of leg* $l = \sqrt{\left(\frac{b}{2}\right)^2 + h^2} < b$ *(as in Figure 5a),* $r_1 = r_2 < r_3$. *Additionally, when l > b,* $r_1 = r_2 > r_3$. *It is easy to guess that when l = b, the isosceles triangle becomes an equilateral triangle, so that* $r_1 = r_2 = r_3$. *Let l = b = 10,* $r_1 = r_2 = r_3 = 1.830$.

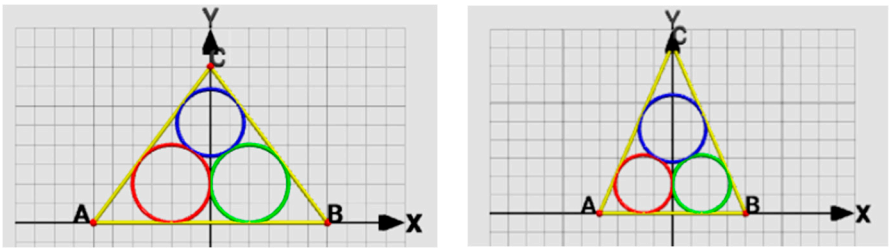


Figure 5. Malfatti circle for an isosceles triangle. (**a**) (*l*,*b*) = (10,12) (**b**) (*l*,*b*) = (10,8).

In the Tri(T_3) problem, we aim to find six circles inside the triangle satisfied by constraints specified by its tangency graph. The three circles tangent to two sides of the triangle are known as corner circles. Furthermore, each of the other three circles is tangent to each other, and it is also tangent to one side of the triangle and the two corner circles. We did not find any simple mathematical solution for the Tri(T_3) problem. However, this problem has been solved using an algorithm with excellent performance. Chiang, Hoffmann, and Rosen [9] proposed algorithms to solve the classical Malfatti's problem and the Tri(T_3) problem. In the Tri(T_3) problem, the isosceles case can also use a depth buffer to extract the solution, and for other triangles, the idea of an inverse construction and angle locking has been introduced in the algorithm to solve the Tri(T_3) problem. Please see Reference [9] for details.

3.3. The Tri(T_n) Problem, n > 2

In this paper, we propose an algorithm to solve the Tri(T_n), n > 2 problems. Before illustrating the algorithm, we will provide terms and theorems to support it.

There are some theorems we need to derive before we propose the algorithm.

Theorem 3. *Three circles C, C_0, C_1 are tangent to each other with radii r, r_0, and r_1. The angle α between two vectors, from the center of C to center of C_0, and from center of C to center of C_1, as shown in Figure 6, is equal to* $cos^{-1}\frac{r^2+(r_0+r_1)r-r_0r_1}{(r+r_0)(r+r_1)}$.

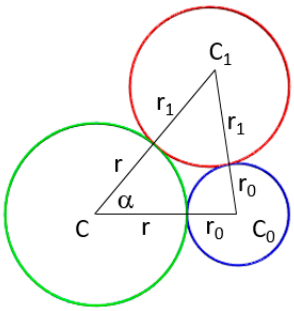


Figure 6. The angle connecting the centers of three pairwise tangent circles.

Proof of Theorem 3. Consider the triangle whose sides are the line segments connecting the centers of three circles. We know the length of the triangle sides are r + r_0, r + r_1, and r_1 + r_2. From the cosine theorem, we have (r_0 + r_1) = (r + r_0) + (r + r_1) − 2(r + r_0)(r + r_1)cos(α). Therefore, the angle α is equal to cos $\frac{r^2+(r_0+r_1)r-r_0r_1}{(r+r_0)(r+r_1)}$. □

Theorem 4. *Two circles, C_1 and C_2, are tangent to each other with radii r_1, r_2. Furthermore, these two circles are tangent to a line, as shown in Figure 7. Then,*

$$\alpha = \frac{\pi}{2} + sin^{-1}\frac{r_2 - r_1}{r_2 + r_1}.$$

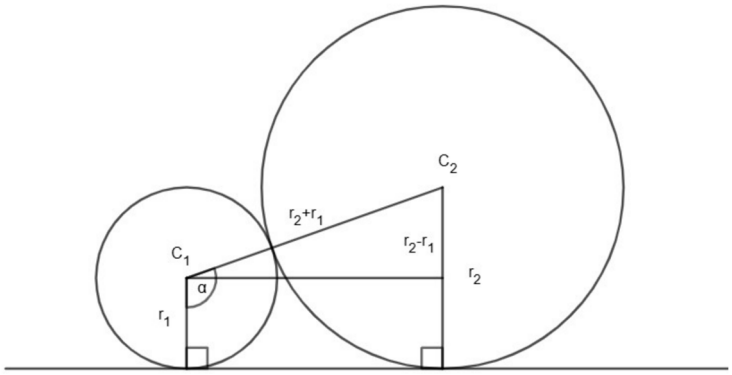


Figure 7. Angle computation for Theorem 4.

Proof of Theorem 4. Assume $r_2 > r_1$, as shown in Figure 7; then, the angle α is greater than $\frac{\pi}{2}$ and $\alpha - \frac{\pi}{2} = \sin^{-1}\frac{r_2-r_1}{r_2+r_1}$. □

Note that if $r_2 < r_1$, $\alpha < \frac{\pi}{2}$, it is because the value of $\sin^{-1}\frac{r_2-r_1}{r_2+r_1}$ is negative.

In our algorithm for solving the Tri(T_n) problem, we provide a criterion to modify the radius for each circle. Consider the tangency graph for the Tri(T_n) problem. We define three different kinds of circles: corner circles, boundary circles, and inner circles. The corner circle is tangent to two other circles and two sides of the triangle. The boundary circle is tangent to four other circles and one side of the triangle. The inner circle is tangent to the other six circles (see Figures 2 and 3). Here, we illustrate the criterion for the inner circle, boundary circle, and corner circle, respectively.

Assume that n circles $C_1, C_2, \ldots, C_n$ with radii $r_1, r_2, \ldots, r_n$, are tangent to one circle C and its neighbor circles, and C_n is tangent to C_1, as shown in Figure 8c. In this situation, the circle ring from C_1 to C_n (n = 9 in Figure 8c, is explicitly tangent to the center circle C. Alternatively, circle C is explicitly tangent to circle rings C_1 to C_n. The size of the circle ring is n. When α_i is the angle between two vectors that connect the center of C_i and C and C and $C_{i+1 \bmod n}$, we define $SA(C) = \sum_1^n \alpha_i$.

Assume that the circle ring $C_1, C_2, \ldots, C_n$ is explicitly tangent to C. If we know the radii for all circles in the circle ring, how do we find the radius r for C? Assume the current radius of C is r, and the circle ring $C_1, C_2, \ldots, C_n$ is explicitly tangent to C without the constraint that C_n is tangent to C_1. We would like to change the value r so that C_n is also tangent to C_1. Assume that n = 9, with a current radius r of C, we can compute $SA(C) = \sum_1^9 \alpha_i$ using Theorem 3 (see Figure 8). There are three different cases for a SA(C): $SA(C) > 2\pi$, $SA(C) = 2\pi$, and $SA(C) < 2\pi$. If $SA(C) = 2\pi$ (Figure 8c), we find the radius for C so that the circle rings are explicitly tangent to circle C. If $SA(C) < 2\pi$, as shown in Figure 8a, we need to reduce the value r so that the circle ring is explicitly tangent to circle C. On the other hand, if $SA(C) > 2\pi$, as shown in Figure 8b, we need to increase the value r so that the circle ring is explicitly tangent to circle C. With this property, the radius of C can be easily found using a binary search algorithm. In fact, from the arc length computation for the circle C, it $(2r) \times \pi =$ (current r) $\times$ SA(C) can be derived, and thus we find that r = (current r) $\times$ (SA(C)/2π).

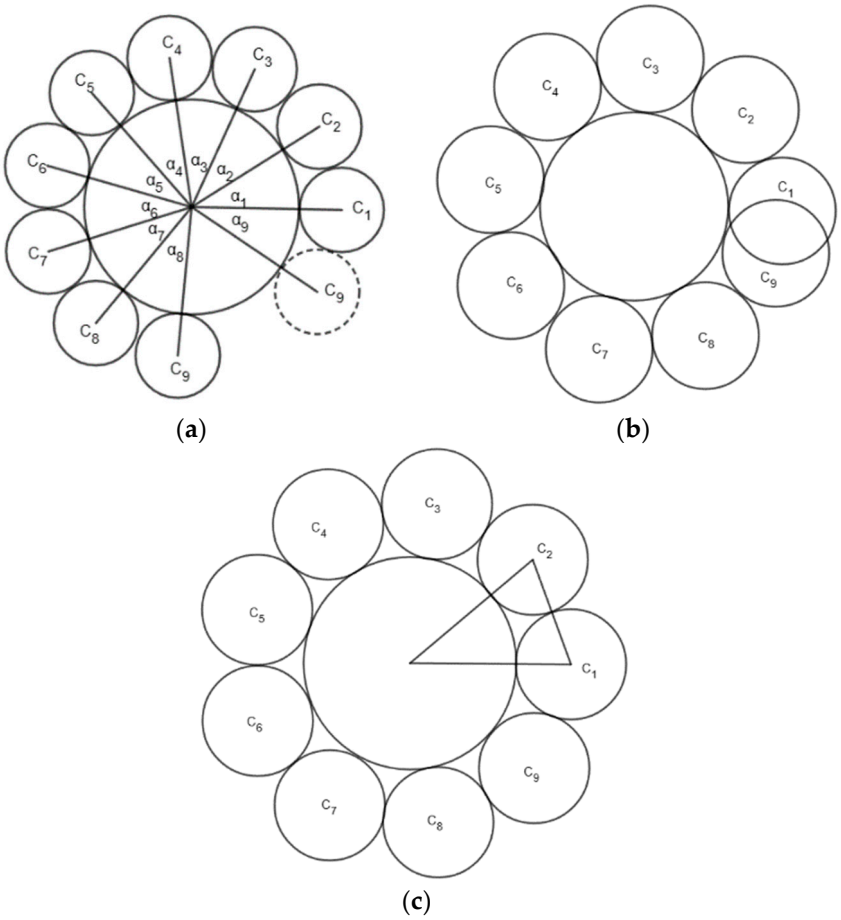


Figure 8. The criterion for the enlarge/reduce radius for center circle. (**a**) SA(C) < 2π (**b**) SA(C) > 2π. (**c**) SA(C) = 2π.

In the solution of the Tri(T_n), n > 3 problem, the inner circle is explicitly tangent to the circle ring, and the size of the circle ring is always six (See Figure 9c). Therefore, with an estimated radius r, it is easy to construct an O(1) algorithm to find the radius of C, which satisfies the constraint SA(C) = 2π. This algorithm (See Algorithm 1) has the following steps:

Algorithm 1. The radius for center circle tangents to a circle ring

1: Input the radii for the circle ring $r_1, r_2, \ldots, r_n$, and an initial radius (r = 1) of C.
2: Using this radius r, compute SA(C).
3: Output the result r × (SA(C)/2π).

Consider the inner circle for the Tri(T_n) problem, as shown in Figure 9c. The angles α_1, α_2, α_3, α_4, α_5, α_6 can be computed using Theorem 3, and the radius of the inner circle can be computed by Algorithm 1.

Now consider the boundary circle (see Figure 9b); the angles α_2, α_3, α_4 can be computed using Theorem 3, and α_1 and α_5 can be computed using Theorem 4. When C is a boundary circle, we define SA(C) as the sum of these five angles. Note that we use the same term SA(C) regardless of whether C is an inner circle or boundary circle. Therefore, if C is a corner circle, SA(C) is equal to the sum of the four angles, as shown in Figure 9a. The angle α_1 is equal to $\pi - \angle A$; α_2 and α_4 can be computed using Theorem 4, and α_3 can be computed using Theorem 3. From the above description, we know that SA(C) can be computed regardless of whether C is a corner circle, a boundary circle, or an inner circle.

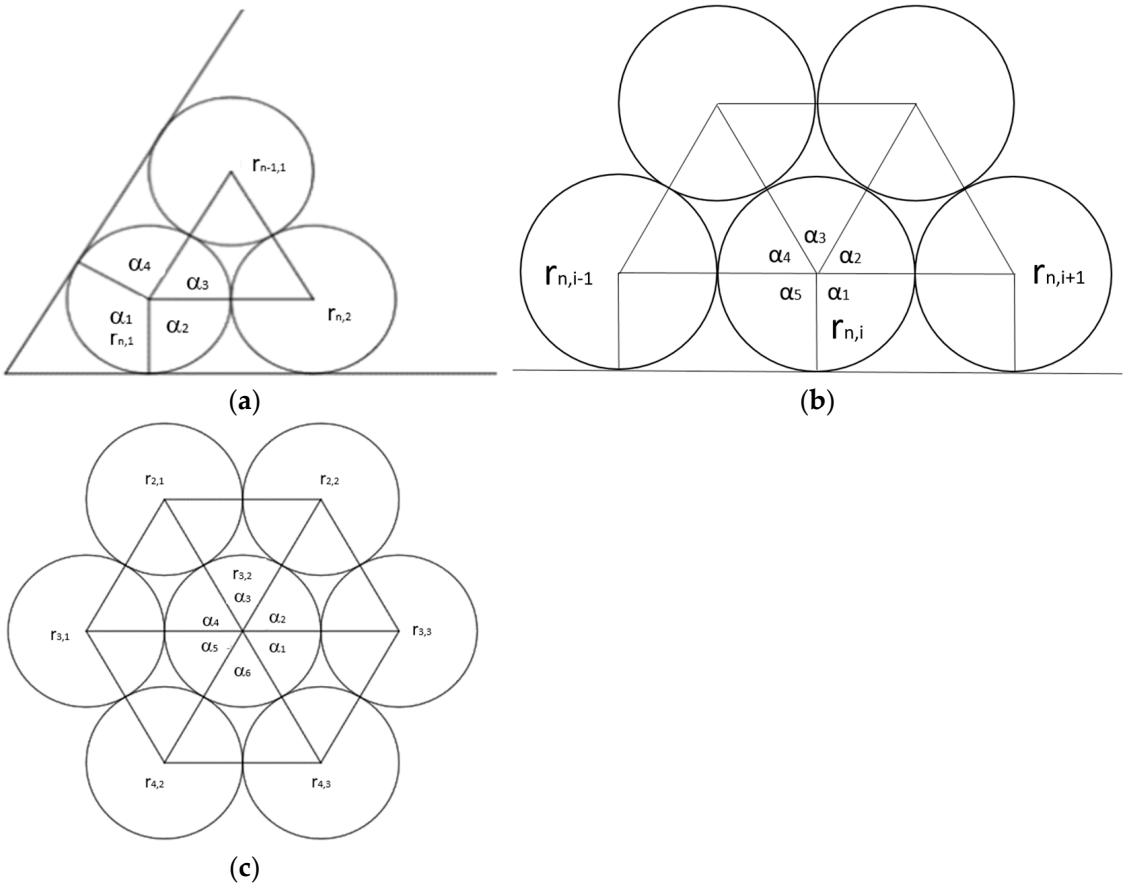
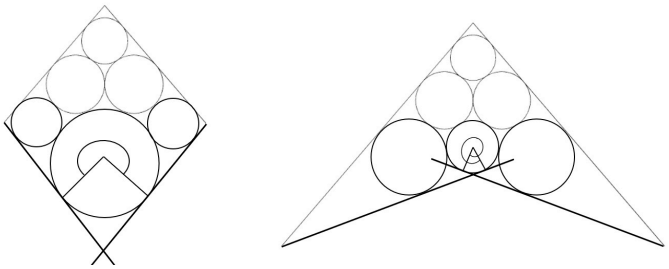


Figure 9. Angles for corner, inner, and boundary circles. (**a**) Corner circle. (**b**) Boundary circle. (**c**) Inner circle.

We know that when the circle radii in the circle ring are fixed, we can find one (inner) circle explicitly tangent to the circle ring. With a similar approach, if the boundary circle with a variable radius is tangent to four circles and one line, we can also find the radius of the boundary circle. Consider the boundary circle in Figure 10; the SA(C) can be computed for the fixed radii of the four circles. When SA(C) < 2π (Figure 10a), we need to reduce the radius of C, and when SA(C) > 2π (Figure 10b), we need to enlarge the radius of C until SA(C) = 2π (Figure 9b). This scenario also works when C is a corner circle (see Figure 11).

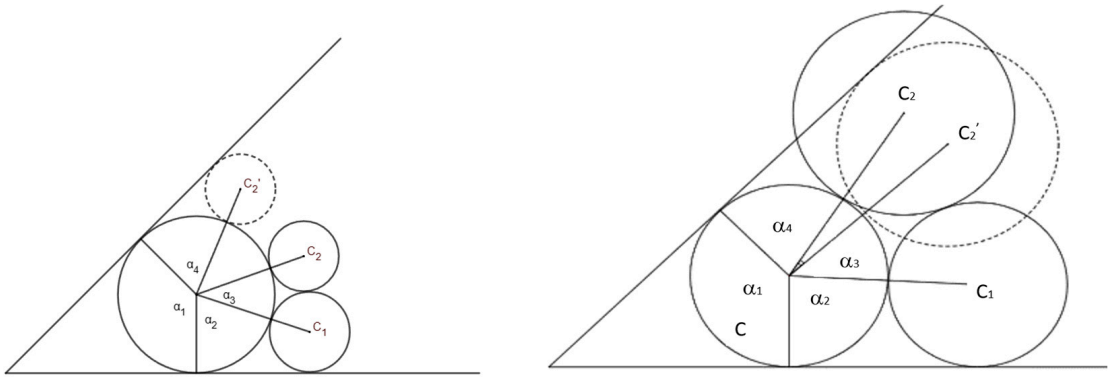


(**a**) SA(C) < 2π (**b**) SA(C) > 2π

Figure 10. The criterion for enlarging/reducing the radius of the boundary circle.

We conclude the following: if SA(C) > 2π, enlarge the radius of C, and if SA(C) < 2π, reduce the radius of C. The circle C can be an inner circle, boundary circle, or corner circle.

We describe the criterion used to modify all circles in the Tri(T_n) problem and propose an algorithm in the next section.



(**a**) SA(C) < 2π (**b**) SA(C) > 2π

Figure 11. The criterion for enlarging/reducing the radius of the corner circle.

4. The Proposed Algorithm to Solve the Tri(T_n) Problem

We present the algorithm and its results in the following section.

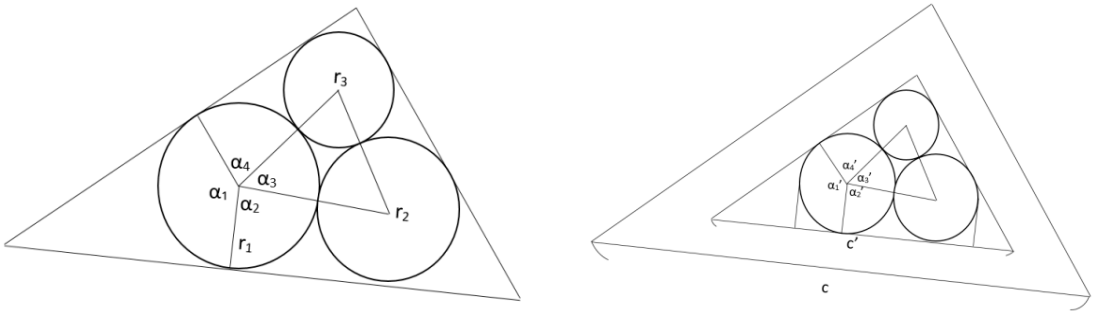
4.1. The Algorithm Used to Solve the Tri(Tn) Problem

The proposed algorithm has the following steps.

1. Input A, B, and C vertices for triangles and n for Tri(T_n) problem.
2. Calculate the initial radii for all T_n circles.
3. Use a while loop to modify the radii of all circles until all circles C satisfy SA(C) = 2π within a small tolerance.
4. Display the results, including the triangle A, B, and C and all circles.

We implemented the above idea and stepped into an infinite loop. Why did this happen? Essentially, we used angle computation to compute SA(C) for all circles with other circles. If we found one solution for the Tri(T_n) problem for triangle ABC, this solution would also satisfy the constraint for any similar triangles of ABC. Conversely, if we found r_m, where $1 \leq m \leq T_n$ was one solution satisfied for SA(C) = 2π, we multiplied all radii to a constant k, $k \times r_m$, where $1 \leq m \leq T_n$ was also satisfied for SA(C) = 2π. There are infinitely many solutions, so we need more constraints.

Using Tri(T_2) as an example, we attempted to add more constraints. In Tri(T_2), we need to find Malfatti circles using the algorithm we proposed; however, no mathematical solution for this exists. As shown in Figure 12, there are three corner circles that we need to find. With the angle name in Figure 12, $\alpha_1 = \pi - \angle A$, $\alpha_2 = \frac{\pi}{2} + \sin^{-1}\frac{r_2 - r_1}{r_2 + r_1}$, $\alpha_3 = \cos^{-1}\frac{r_1^2 + r_1 r_2 + r_1 r_3 - r_2 r_3}{(r_2 + r_1)(r_3 + r_1)}$ and $\alpha_4 = \frac{\pi}{2} + \sin^{-1}\frac{r_3 - r_1}{r_3 + r_1}$. Note that if we enlarge the radii k times ($R_i = k \times r_i$), we find the same values for α_1, α_2, α_3, and α_4. We require one more constraint, such as the length of one side of the triangle (See Figure 12).



(**a**) Surrounding angle for the corner circle (**b**) Change ratio= c/c′.

Figure 12. All circles in the triangle should also be tangent to the triangle.

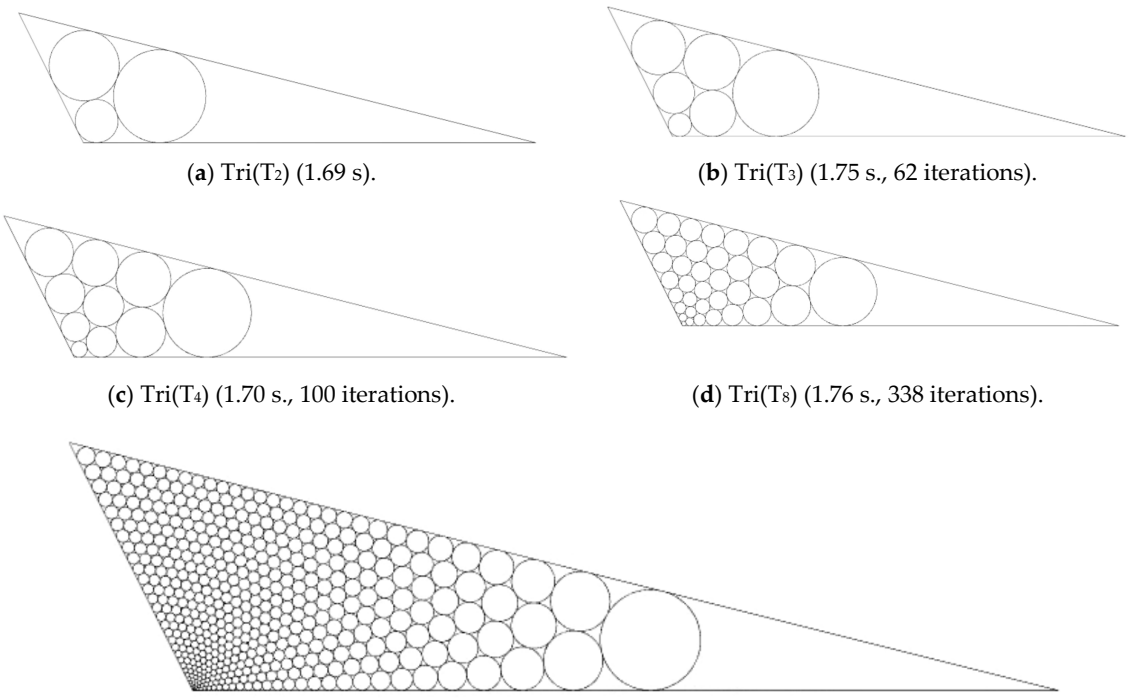
Now, the proposed algorithm (See Algorithm 2) is as follows.

Algorithm 2. The Tri(T*n*) problem

1: **Input** the coordinate of three vertices A, B, C, and n; calculate the angle A, B, C, and the edge length c
2: Initial radius list $r_{old} = [r_i \mid \text{where } 1 \leq i \leq T_n]$. eps = 0.01. Calculate the new radius list; call it r_{new} from $SA(C_i) = \theta_i$ using the radii in the old radius list r_{old}.
3: **While** one of $|\theta_i - 2\pi| >$ eps: # one of the circle is two large or too small, then 3.1–3.3
3.1: Compute change ratio k from c, the side length of the triangle, and new radius list r_{new}.
3.2: Modify the radius list at the same time. $r_{old} = k \times r_{new}$
3.3: Calculate new radius list r_{new} f from SA(Ci) = θi using the radii in r_{old}.
4: Compute the change ratio k; the solution radii is $k \times r_{new}$.
5: Display the result.

4.2. Example of the Tri(T_n) Problem

Given the coordinates of A = (0,0), B = (70,0), and C = (−10,20), we obtain the following result (see Figure 13) for the Tri(T_n), n = 2, 3, 4, 8, 30 problem. This lasted 1.69, 1.75, 1.70, 1.76, and 13.15 s, respectively. The radii for Tri(T_2) can be directly computed from a mathematical equation. The number of iterations in the while loop is 62, 100, 338, and 3533 for n = 3, 4, 9, 31, respectively.



(**a**) Tri(T_2) (1.69 s).

(**b**) Tri(T_3) (1.75 s., 62 iterations).

(**c**) Tri(T_4) (1.70 s., 100 iterations).

(**d**) Tri(T_8) (1.76 s., 338 iterations).

(**e**) Tri(T_{30}) (13.15 s., 3533 iterations).

Figure 13. The Tri(T_n) problem.

5. Implementation Detail and Potential Applications

There are two things that we need to consider when implementing Algorithm 2. The first was how to enlarge/reduce the radius, and the second was how to illustrate all radii once they were found.

To begin with, we set a step value (=0.01), and when the radius r needed to be enlarged, we added the value r = r + step, similarly reducing the radius (r = r − step). This approach made the algorithm very inefficient. For example, it took 20,391 iterations (3 s) to solve the Tri(T_3) problem, and thus, we required a more effective method of enlarging/reducing the radius. This enlarge/reduce strategy had to correspond to the surrounding angle of

the circle. The new radius had to have a circumference equal to the circular arc length, whose radius was r and angle was θ, i.e., $r' = \frac{\theta}{2\pi}r$. Note that when $\theta > 2\pi$, we enlarged the radius ($r' > r$), and when $\theta < 2\pi$, we reduced the radius ($r' < r$). Using this approach to enlarge/reduce the radii, the same example took 62 iterations (less than 2 s).

The main task for the drawing process was to find the center of each circle. Consider three cases as follows (See Figure 2):

1. The center of $C_{1,1}$: This circle is a corner circle, and we want to determine the center of $C_{1,1}$ from two edges and a radius value ($r_{1,1}$) for the corner circle.
2. The center of $C_{2,1}$: This circle is an edge circle when $n > 1$. From the above step, we can determine the center and radius of $C_{1,1}$. With the one side of the triangle $\overline{AC}$, we can find the center of $C_{2,1}$. With this approach, we can find the center of $C_{i,1}$ for all circle circles $1 < I < n$.
3. The center of $C_{2,2}$: If we determine the center and radii of $C_{1,1}$, $C_{2,1}$, from the above two steps and from the radius of $C_{2,2}$, we can find the center of $C_{2,2}$.

The above three problems can easily be solved using simple geometry. Furthermore, if we solve the center position for circles row-by-row (from left to right in Figure 2), it falls in one of the above three questions so that every center of the circles can be easily found. After we find the center of all circles with their associated radii, the drawing process becomes simple.

This problem can be applied to several different applications, such as the following:

1. Logo design: the design of logos for different products.
2. Product design: for a triangle prism comprising six bottles of different sizes and radii containing different liquids, the radii can be calculated.
3. Radar location design: In a triangle area, a radar system can be established consisting of several radars, with each radar having a different radius working in a circular area. This working circular area is much smaller than the area of the triangle. Our algorithm can determine the center of each radar, and the radii of the circular area should be greater than the radii we find from the algorithm so that every object that passes this triangle area will be detected by one of the radars, i.e., nothing can pass this triangle area.
4. Wireless transmitter location design: Similar to radar location design, the application of the algorithm can determine the center of a wireless transmitter location and what kind of wireless transmitter we should use. Larger radii are used so that the union of the circular area covers the triangle, and any location inside the area can receive wireless signals.
5. Mouse pad design: Suppose that there is a triangle pad, and we want to cut this pad into different sizes of circular mouse pads. The algorithm proposed in this paper can suggest a method for cutting the pad. Notice that a different initial value r can produce a different number of circular pads.

6. Conclusions and Future Research

The Tri(T_n) problem can be solved by the proposed O(T_n) algorithm. This algorithm can solve the Tri(T_{46}) (1128 circles) problem in less than one minute without a display.

We know that the Tri(T_1) and Tri(T_2) problems have unique solutions. One solution can be found using the algorithm we proposed, but it is unclear whether the solution is unique for Tri(T_n), $n > 2$. Therefore, the uniqueness property or its counter-example for the Tri(T_n) problem is a topic for future research.

Funding: This research was funded by National Science and Technology Council in Taiwan grant number NSTC-112-2221-E-013-003.

Data Availability Statement: This paper did not need big data for the algorithm.

Acknowledgments: I would like to extend my sincere gratitude to Hung-Chieh Li, Min-Hsuan Hsiung, and Fan-Ming Chiu for their invaluable contributions to the programming aspect of this research.

Conflicts of Interest: The author declare no conflict of interest.

References

1. Fukagawa, H.; Pedoe, D. *"The Malfatti Problem". Japanese Temple Geometry Problems (San GaKu)*; The Charles Babbage Research Centre: Winnipeg, MB, Canada, 1989; pp. pp. 28+103–106.
2. Bottema, O. The Malfatti Problem. *Forum Geom.* **2000**, *1*, 43–50.
3. Stefanović, M. Triangle centers associated with the Malfatti circles. *Forum Geom.* **2003**, *3*, 83–93.
4. Wolfram MathWorld. Malfatti Circles. Available online: http://mathworld.wolfram.com/MalfattiCircles.html (accessed on 15 August 2024).
5. Bernardin, L.; Chin, P.; DeMarco, P.; Geddes, K.O.; Hare, D.E.G.; Heal, K.M.; Labahn, G.; May, J.P.; McCarron, J.; Monagan, M.B.; et al. *Maple Programming Guide*; Maplesoft, a Division of Waterloo Maple Inc.: Waterloo, ON, Canada, 1996.
6. Meurer, A.; Smith, C.P.; Paprocki, M.; Čertík, O.; Kirpichev, S.B.; Rocklin, M.; Kumar, A.; Ivanov, S.; Moore, J.K.; Singh, S.; et al. SymPy: Symbolic computing in Python. *PeerJ Comput. Sci.* **2017**, *3*, e103. [CrossRef]
7. Kay, D.C. *College Geometry*; Holt, Rinehart and Winston: New York, NY, USA, 1969.
8. Solution du Dernier des Deux Problèmes Proposés à la Page 196 de ce Volume. Annales de Mathématiques Pures et Appliquées, Tome 1 (1810–1811); pp. 343–348. Available online: http://www.numdam.org/item/AMPA_1810-1811__1__343_0/ (accessed on 15 August 2024).
9. Chiang, C.-S.; Hoffmann, C.M.; Rosen, P. A generalized Malfatti problem. *Comput. Geom.* **2012**, *45*, 425–435. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.