**Design and Calculating Bounded Area**

**Introduction**

1. **Part A Working with a Given Function and Designing with Functions**
   1. **Overview**

Part A of the task requires a graphic design that is clearly described mathematically. It also asks to integrate two given functions and into the design. A good mathematical expression given in this section will be important in Part B when evaluating the area. To achieve this, a series of methods will be introduced, and based on them, a rather complex not but overly elaborate pattern is expected.

* 1. **Calculate the area of the given region**

Given a function . The second function It is created by vertically translating down by 2 units. The region bounded by the two curves is

(see Figure 1)

To calculate its area, there are 2 methods.

|  |  |  |
| --- | --- | --- |
| This area can be evaluated by the formula:  This is a general formula to calculate the region bounded by a pair of functions in a given domain. To use this formula, the equation of is required. By the definition of translation, . |  | Figure 1. Region *D0* |
| An easier way: the region between 2 curves is formed by translation. Hence, when evaluating the integral in Eq. 1, every infinitesimal rectangle of the Riemann summation has the same length , so the total area is equal to the area of a 4.5 by 2 rectangle (see the orange rectangles in Figure 2). |  | Figure 2. Put all infinitesimal rectangles together |

* 1. **A rough sketch of the final design**
  2. **Design of region *D1***
     1. **Affine transformation and parametric equation**

The task asks to integrate the region between the given functions and into the final design, and it is encouraged to use transformations including reflection, translation, or dilations, which are **affine transformations**.

A simple definition of an affine transformation is to perform a linear transformation on a vector space, and additionally, a translation. Generally, an affine transformation on vector in can be written as

where is a 2 by 2 matrix representing a linear transformation, and represents a translation. The transformation of a function is more complicated. For this specific purpose, since what is transformed is the whole shape of a function , which means the actual subject of the transformation is every point on the plane, namely

This equation naturally satisfies the restriction on the **domain.** But note that after the transformation, the relation between and will probably no longer obey the restriction of a function (one-to-one correspondence). To better describe this kind of curve, **parametric equations** are introduced. Hence, a curve can be written as . According to Eq. 2, a general affine transformation on a parametric equation is

Parameterise is simply letting , so . And , the parametric form of , is given by transformation . Now all kinds of transformation can be applied to. To adjust the given function to be a fancy frame of the design, following transformations are applied:

|  |  |  |
| --- | --- | --- |
| The region looks too long in the vertical direction. To compress vertically to 1/5 of the original size and stretch horizontally to four times the original size requires the matrix  Since there is no translation, b is the zero vector. | Figure 3. Transformation 1 |  |
| Then the wiggling end of the curves is expected to be centered and aligned with the y-axis. Hence a flip transformation and a translation are needed. The flip matrix is  The translating vector is | Figure 4. Transformation 2 |  |
| Copy a pair of curve and to make a symmetrical shape, where the same flip matrix is needed. | Figure 5. Transformation 3 |  |

An advantage of affine transformations is that the amount of the scaling of the area can be easily calculated by the **determinant** of the transformation matrix. This will be further discussed in Part B.

* + 1. **Non-linear transformation and Jacobian**

The shape of a frame is expected to be more “enclosed”, by curlingit into a circular shape. This requires a **non-linear transformation**. A non-linear transformation does not guarantee that the parallel lines in the original graph remain parallel, and hence do not have a constant matrix to describe the transformation for every point on the Cartesian plane. A general way is to establish a **map**. For this purpose, the map is:

since the new graph’s x-coordinates have a range from -18 to 18, and the curl starts at in the polar frame. What this **curl** transformation actually does is to wrap the Cartesian grid into a polar shape (see Figure 6). As a matter of convenience, the transformed curves are named .

|  |  |
| --- | --- |
| Figure 6. Curl transformation |  |

Though this transform do not have a matrix to describe it, it do have a local linear behaviour when one looks at it closely enough. This sort of linear behaviour can be measured by Jacobian. For transformation , the Jacobian is

The method to evaluate the transformed area based on the Jacobian will be further discussed in Part B.

* + 1. **Smoothness Consideration**

Generally, the smoothness can be measured by continuity , namely the nth derivatives at the two ends are equal. However, when dealing with a parametric curve, this approach will probably fail since the rate of change in position of a point with respect to t is not only related to the smoothness of the shape but also the smoothness of the motion (change in velocity). Thereby, the geometric continuity is introduced. A curve or surface can be described as having continuity, with being the increasing measure of smoothness. Consider there are two curves, and their continuity is defined as: (1) , the curves touch at the join point; (2) , the curves also share a common tangent direction at the join point, which means the derivatives are collinear; (3) , the curves also share a common centre of curvature at the join point (Wikipedia, 2023). For graph designs, is sufficiently smooth. Also, an affine transformation will not affect the original continuity, and actually, every continuous mapping can keep the original continuity.

To construct continuous joins, copy one set of curves shown in Figure 6 and rotate them 180 degrees to get curves **,** therefore,

(for full equations, see Appendix 1).

|  |  |
| --- | --- |
| Figure 7. Region *D1* | Curves give us Region *D1* as shown in Fig. 2, where there are four joins satisfying and continuity: , , , and . Take as an example. |

Since the direction of the motion is reversible in a parametric equation, Eq. 7 actually implies there is a continuity. By applying similar approaches to the other three joins, it can be found that all joins are continuous. Now, Graph A is good enough to be a part of the final design, and to form other graphs, some general methods are required.

* 1. **Design of region *D2***

.

|  |  |  |
| --- | --- | --- |
| Figure 8. g1(x) | This part of the design is the wing of a bird, which would be formed by some basic functions. Its shape is roughly like Figure 8, constituted by functions .  is simply a straight line. For convenience, it is put horizontally and assigned to be | C:\Users\Administrator\AppData\Local\Microsoft\Windows\INetCache\Content.Word\wing.png  Figure 9. D2 Rough shape |
| Figure 10. g2(x) | is winding and has an “S” shape. A very suitable function is Sigmoid, which is defined as  To fit this curve to the shape required, some parameters are added:  Where , , and control the dilations in and directions and horizontal translation. By adjusting them, it is found that the curve in Figure 10 is given when , , and are 2.6, 3.4, and 0.9. | |
| Figure 11. g3(x) | is the upper half of the left side of the wing. It is like a log function and can be modelled as  Where , , , and control the dilations in and directions and translation in and directions. The curve in Figure 10 is given when , , , and are 1, 0.1, -0.8, and 3.8, respectively. It is expected that the and connect at with continuity. The derivative of at is | |
| Figure 12. g4(x)    Figure 13. D2 no domain | is modelled by a quadratic function:  Whose derivative is . To make and connect at with continuity, the following 2 equations need to be satisfied.  Note that the equations are now hard to solve. To simplify it, let . Thus, .  3 variables and 2 equations leave 1 degree of freedom to adjust. When set , Figure 12 is given, and . | |
| Figure 14. D2 | To get the joints A, B, D, to get the **domains**, it is necessary to solve the following equations:  The solution is . Hence, the domains are , , , and (see Figure 14). | |

**Bibliography**

Wikipedia. (2023). *Smoothness*. [online] Available at: https://en.wikipedia.org/wiki/Smoothness.

3Blue1Brown (2019). *But what is a Fourier series? From heat flow to drawing with circles*. [online] YouTube. Available at: <https://www.youtube.com/watch?v=r6sGWTCMz2k&list=PL4VT47y1w7A1-T_VIcufa7mCM3XrSA5DD&index=4>.

**Appendix 1. Curves that construct *D1***

**Appendix 2. Derivative of the Bezier curves.**

*Smoothness*.

First, take the derivative of the th term of a Bezier curve without control points as coefficients:

Expand the binomial coefficient:

Notice that the two terms are typical Bezier curves of lower orders. Hence,

Now, apply the control points.

Expand this and reorder the terms by :

In this transformation, there are two eliminations, namely as the th term does not exist and the very last term with the binomial coefficient , which is usually considered to be zero. Write it in the form of a sum: