

# Level set methods for solving isoperimetric problem and plateau problem

자유전공학부  
2011-13285  
문승용

## 1 Introduction

미분기하학의 고전적인 주제 중 하나인 isoperimetric problem은 surface 위의 simple closed curve의 둘레와 curve가 둘러싸고 있는 interior의 넓이 사이의 관계를 찾는 문제이다. 주어진 surface가  $\mathbb{R}^2$ 일 경우 simple closed curve의 둘레  $l$ 와 넓이  $A$  사이의 관계가 널리 알려져 있는데 이는 다음과 같다.

$$4\pi A \leq l^2$$

(= holds for circle)

둘레가 최대가 되는 경우에 curve 위의 각 점에서의 curvature가 같아짐을 알 수 있는데 따라서 넓이가  $2\pi r^2$ 인 plane simple closed curve  $c$ 의 둘레의 길이가 최대가 되는 조건은 다음과 같다.

$$c : \text{maximal length} \longleftrightarrow \kappa(p) = \frac{1}{r} \quad \forall p \in c$$

( $\kappa(p)$  : curvature of  $c$  at  $p$ )

마찬가지로 plateau problem 역시 미분기하학에서 중요한 주제 중 하나인데,  $\mathbb{R}^n$ 에 주어진 closed curve를 경계로 하는 minimal surface를 찾는 문제이다. minimal surface란 넓이를 locally minimizing하는 surface를 의미하는데 변분법을 이용하면 다음의 관계를 얻을 수 있다.

$$A : \text{minimal surface} \longleftrightarrow H(p) = 0 \quad \forall p \in A$$

( $H(p)$  : mean curvature of  $A$  at  $p$ )

isoperimetric problem의 경우 각 점에서의 curvature 값의 변동이 적을수록 넓이 대비 둘레가 줄어들음을 알 수 있다. 따라서 어떤 plane curve가 주어져 있을 때 각 점에서의 curvature 값이 균일하도록 만들어 준다면 넓이 대비 둘레를 최소화시킬 수 있다. curve 위의 각 점에 inward normal vector와 평행하고 크기가 signed curvature와 같은 vector를 대응시킬 수 있는데, 이 벡터들의 모임을 curvature flow라 부른다. Gage와 Hamilton은 [1] curvature flow에서 모든 convex curve는 circle로 수렴하게 된다는 것을 증명하였다. 또한 Grayson은 [2] 모든 plane curve에 대해 위의 정리가 성립함을 증명하였다. 마찬가지로 plateau problem에서는 surface의 mean curvature 값이 surface의 넓이를 결정하는데, 위와 마찬가지로 surface 위의 각 점에 inward normal vector와 평행하고 크기가 mean curvature와 같은 벡터를 대응시킨 후, 이에 대한 surface의 local flow를 생각해보면 기존의 surface보다 넓이가 줄어들게 된다.

이번 보고서에서는 level set method를 이용하여  $\mathbb{R}^2$ 에 주어진 simple closed curve를 curvature flow를 이용하여 넓이를 유지시키면서 최소의 둘레를 갖는 곡선으로 변화시키는 알고리즘을 만들어보았다. 또한  $\mathbb{R}^3$ 에 두 개의 square curve가 일정한 거리를 두고 있을 때, 두 curve를 경계로 하는 minimal surface를 계산해보았다.

## 2 Level set method

closed curve  $c_0 : S^1 \rightarrow \mathbb{R}^2$ 가 주어져 있다고 하자. 곡선은 시간이 지남에 따라서 변화하고, 시간  $t$ 에서의 곡선의 image를  $c_t$ 라 하자. 시간  $t$ 에서 curve 위의 각 점은 outward normal vector 방향으로  $F(c_t)$ 의 속력으로 움직인다고 하자. 즉,  $F$ 는 곡선의 모양  $c_t$ 에 의존한다.  $c_t$ 를 구하기 위해서 먼저  $c_0$ 를 zero level set으로 갖는, 거의 모든 점에서 미분가능한 함수  $\varphi : \mathbb{R}^2 \rightarrow \mathbb{R}$ 를 찾자.  $\varphi$ 의 대표적인 예로는 signed distance function이 있다.

$$\varphi(\vec{x}) = \begin{cases} -\inf\{\text{dist}(\vec{x}, p) \mid p \in c_0\} & \text{if } \vec{x} \in \text{int}(c_0) \\ \inf\{\text{dist}(\vec{x}, p) \mid p \in c_0\} & \text{otherwise.} \end{cases}$$

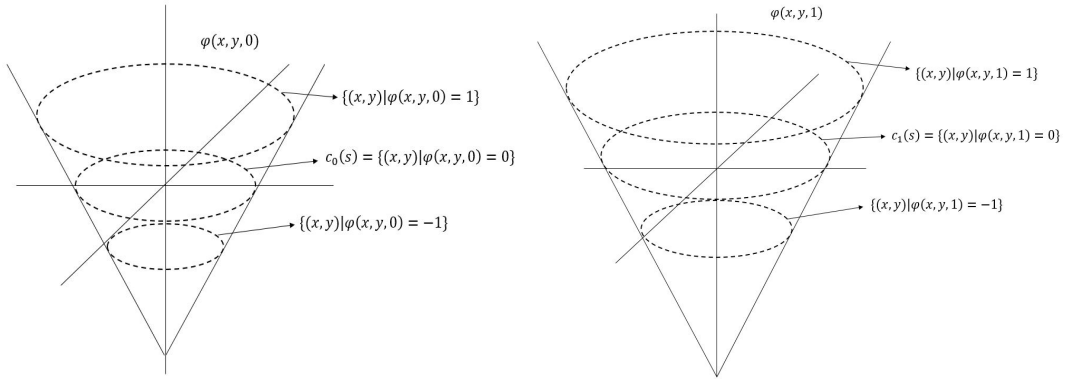


Figure 1: Moving circle with constant speed

$\varphi$ 를 확장하여  $\varphi(x, y, t) : \mathbb{R}^2 \times [0, \infty) \rightarrow \mathbb{R}$ ,  $\varphi(x, y, 0) = \varphi(x, y)$ 를 생각하자. 이제  $t > 0$ 일때  $\varphi(x, y, t)$ 의 함수값을 결정해주어야 하는데,  $\varphi(x, y, t)$ 의 zero level set이  $c_t$ 이라고 가정하자.  $c_0$  위의 임의의 점  $p_0 = (x_0, y_0)$ 를 택하자. 그러면  $p_0$ 는 시간이 흐름에 따라 움직이게 될텐데,  $p_0$ 의 integral curve를  $p(t) = (x(t), y(t))$ 라 하자. 그러면 가정에 의해  $p(t)$ 는 항상  $\varphi(\vec{x}, t)$ 의 zero level set에 속하게 된다. 따라서 다음의 식을 만족시킨다.

$$\varphi(x(t), y(t), t) = 0 \quad \text{for all } t \in [0, \infty)$$

위의 식을  $t$ 에 대해 미분하면 다음의 식을 얻는다.

$$(\varphi_x, \varphi_y) \cdot (x'(t), y'(t)) + \varphi_t = 0$$

$(\varphi_x, \varphi_y)$ 와  $(x'(t), y'(t))$ 는  $\varphi(\vec{x}, t) = c_t(\vec{x})$ 의  $p(t)$ 에서의 normal vector이고,  $(x'(t), y'(t))$ 의 크기는  $F$ 이므로 다음의 식을 얻는다.

$$\begin{aligned} F \times |(\varphi_x, \varphi_y)| + \varphi_t &= 0 \\ \varphi(x, y, 0) &= \varphi(x, y) \end{aligned}$$

### 3 Numerical approximation

#### 3.1 Extension of speed function

위의 미분방정식을 numerical하게 풀기 위해서는 Euler method를 이용하면 된다.

$$\begin{aligned} \frac{\varphi^{n+1}(x, y) - \varphi^n(x, y)}{\Delta t} + F^n \times |(\varphi_x^n(x, y), \varphi_y^n(x, y))| &= 0 \\ \varphi(x, y, 0) &= \varphi(x, y) \\ (\varphi^n(x, y) = \varphi(x, y, n\Delta t), \quad F^n(x, y) = F(x, y, n\Delta t)) \end{aligned}$$

하지만  $F$ 의 정의역은 시간이  $t$ 일때  $\{(x, y) | \varphi(x, y, t) = 0\}$ 이다. 따라서  $F$ 를 정의역을  $\mathbb{R}^2$ 로 확장해줘야 위의 미분방정식을 풀 수 있다.  $t = n\Delta t$ 일 때  $F_{ext}^n$ 를 다음과 같이 정의하자.

$$\begin{aligned} F_{ext}^n : \mathbb{R}^2 &\rightarrow \mathbb{R} \\ F_{ext}^n(x, y) &= F^n(x, y) \quad \text{if } \varphi(x, y, n\Delta t) = 0 \end{aligned}$$

Adalsteinsson과 Sethian에 의하면 [3]  $\varphi(x, y, n\Delta t) \neq 0$ 인  $(x, y)$ 에 대해서  $F_{ext}^n$ 의 함수값을 다음 식을 만족하도록 정해준 후, 위의 차분방정식에  $F^n$  대신  $F_{ext}^n$ 를 대입하면 우리가 원하는 해를 얻을 수 있다.

$$\nabla F_{ext}^n \cdot \nabla \varphi^n = 0$$

또한 Sethian에 의하면 [4]  $F$ 가 curvature(or mean curvature)에 의존하는 함수일 때  $F_{ext}^n$ 를 다음과 같이 정의하면 우리가 원하는 해를 얻을 수 있다.

$$F_{ext}^n(x, y) = f(\kappa(x, y))$$

( $\kappa(x, y)$  : the curvature(or mean curvature) of the level set curve(or surface) of  $\varphi^n$  passing  $(x, y)$ )

#### 3.2 Narrow band method

위의 차분방정식을 풀기 위해서는  $\varphi^n$ 의 zero level set을 포함하는 정사각형 모양의 subset을 잡고 이를 정사각형의 mesh로 잘라야 한다. 그리고 각 gridpoint  $(x_i, y_j)$ 에서 위의 차분방정식을 풀면 된다.

$$\frac{\varphi^{n+1}(x_i, y_j) - \varphi^n(x_i, y_j)}{\Delta t} + F_{ext}^n \times |(\varphi_x^n(x_i, y_j), \varphi_y^n(x_i, y_j))| = 0$$

연산 횟수를 줄이기 위해서  $\varphi^n$ 의 정의역을  $\varphi^n$ 의 zero level set의 tubular neighborhood에 제한시키고, 이 영역들에서만 차분방정식을 푸는데 이를 Narrow band method라 부른다.

Narrow band method를 이용하는 경우,  $t = 0$ 일 때 주어진 곡선  $c_0$ 으로부터 반지름이  $r$ 인 tubular neighborhood( $B_r$ )를 구한 다음  $\varphi^0$ 을 정의하게 되는데, tubular neighborhood 안에 들어있는  $(x_i, y_j)$ 에 대해서만 signed distance를 계산해주면 된다. 따라서  $\varphi^0$ 은 다음과 같이 정의된다.

$$\varphi^0(x_i, y_j) = \begin{cases} -\inf\{dist((x_i, y_j), p) | p \in c_0\} & \text{if } (x_i, y_j) \in \text{int}(c_s) \cap B_r \\ -r & \text{if } (x_i, y_j) \in \text{int}(c_s) \cap B_r^c \\ \inf\{dist((x_i, y_j), p) | p \in c_0\} & \text{if } (x_i, y_j) \in \text{int}(c_s)^c \cap B_r \\ r & \text{if } (x_i, y_j) \in \text{int}(c_s)^c \cap B_r^c \end{cases}$$

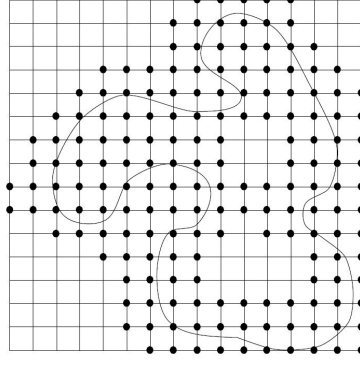


Figure 2: Narrow band method

마찬가지로 tubular neighborhood 안에서  $F_{ext}^0$ 를 구해준 후 위의 차분방정식을 풀면  $\varphi^1$ 을 구할 수 있다. tubular neighborhood를 고정시킨 후 위의 차분방정식을 계속해서 풀면  $\varphi^n$ 을 구할 수 있다. 만약 어떤 자연수  $N$ 이 존재하여  $\varphi^N$ 의 zero level set이 tubular neighborhood를 벗어나게 된다면  $\varphi^N$ 의 zero level set을 이용하여 새로운 tubular neighborhood를 잡은 후  $\varphi^N$ 을 재정의 해줘야 하는데, 위에서  $\varphi^0$ 을 정의할 때와 마찬가지로 방법으로 하면 된다.

정사각형 grid에서 차분방정식을 풀게 되면 연산 횟수는  $O(N^2)$ 이 된다( $N$ : the number of meshes in  $x$  direction). 하지만 Narrow band method의 연산 횟수는  $O(kN)$ 이 된다. 여기서  $k$ 는 tubular neighborhood의 반지름에 비례하는 값이다.

## 4 Algorithm

### 4.1 Isoperimetric problem

Isoperimetric problem을 풀기 위해 simple closed curve의 넓이를 유지하면서 둘레를 최소화하는 알고리즘을 만들었다. curve  $c_t$ 가 시간에 따라 변화하게 되면 그 넓이  $A$ 도 변화하게 되는데, Green's theorem과 간단한 미적분학을 통해 변화율을 계산해보면 다음과 같다.

$$\frac{d}{dt}A(c_t) = \int F ds \quad (F : \text{speed function})$$

$F = -\kappa$ 인 경우 넓이의 변화는 다음과 같다.

$$\frac{d}{dt}A(c_t) = - \int \kappa ds = -2\pi$$

따라서 curvature flow에서 곡선의 넓이는 감소하게 된다. 따라서 곡선의 모양을 유지시키면서 넓이를 증가시켜야하는데, 이를 위해 곡선 위의 각 점에 outward normal 방향으로 constant speed를 추가로 더해주면 된다.  $F = -\kappa + \frac{2\pi}{l}$ 인 경우 넓이의 변화는 다음과 같다.

$$\frac{d}{dt}A(c_t) = - \int \kappa + \frac{2\pi}{l} ds = 0$$

이제  $\kappa$ 의 값을 구해야하는데, curve  $c_t$ 의 explicit한 form을 알지 못하고  $c_t$ 를 zero level set으로 갖는  $\varphi(\vec{x}, t)$ 에 대한 정보만 존재한다. Riemannian geometry의 기초적인 성질을 이용하면  $\kappa$ 를  $\varphi$ 의 식으로 표현할 수 있다.

$$\kappa = \nabla \cdot \left( \frac{\nabla \varphi}{|\nabla \varphi|} \right)$$

따라서 isoperimetric problem 을 풀기 위한 차분방정식을 표현하면 다음과 같다.

$$\frac{\varphi^{n+1}(x_i, y_j) - \varphi^n(x_i, y_j)}{\Delta t} + \left( -\nabla \cdot \left( \frac{\nabla \varphi^n(x_i, y_j)}{|\nabla \varphi^n(x_i, y_j)|} \right) + \frac{2\pi}{l} \right) \times |(\varphi_x^n(x_i, y_j), \varphi_y^n(x_i, y_j))| = 0$$

$\kappa(x_i, y_j)$  를 수치적으로 구하기 위해서 central difference approximation 을 이용한다.

$$\varphi_x^n(x_i, y_j) = \frac{\varphi^n(x_{i+1}, y_j) - \varphi^n(x_{i-1}, y_j)}{2h} \quad \varphi_y^n(x_i, y_j) = \frac{\varphi^n(x_i, y_{j+1}) - \varphi^n(x_i, y_{j-1})}{2h}$$

$$\varphi_{xx}^n(x_i, y_j) = \frac{\varphi^n(x_{i+1}, y_j) - 2\varphi^n(x_i, y_j) + \varphi^n(x_{i-1}, y_j)}{h^2}$$

$$\varphi_{xy}^n(x_i, y_j) = \frac{\varphi^n(x_{i+1}, y_{j+1}) + \varphi^n(x_{i-1}, y_{j-1}) + \varphi^n(x_{i+1}, y_{j-1}) - \varphi^n(x_{i-1}, y_{j+1})}{4h^2}$$

$$\varphi_{yy}^n(x_i, y_j) = \frac{\varphi^n(x_i, y_{j+1}) - 2\varphi^n(x_i, y_j) + \varphi^n(x_i, y_{j-1})}{h^2}$$

$$\kappa = \frac{\varphi_{xx}^n(\varphi_y^n)^2 - 2\varphi_x^n\varphi_y^n\varphi_{xy}^n + \varphi_{yy}^n(\varphi_x^n)^2}{((\varphi_x^n)^2 + (\varphi_y^n)^2)^{3/2}}$$

( $h$  : the size of a mesh)

마찬가지로  $|(\varphi_x^n(x_i, y_j), \varphi_y^n(x_i, y_j))|$  는 다음과 같이 근사한다.

$$|(\varphi_x^n, \varphi_y^n)| = ((\varphi_x^n)^2 + (\varphi_y^n)^2)^{1/2}$$

이를 종합하면 다음과 같은 식이 나온다.

$$\frac{\varphi^{n+1} - \varphi^n}{\Delta t} + \left( -\frac{\varphi_{xx}^n(\varphi_y^n)^2 - 2\varphi_x^n\varphi_y^n\varphi_{xy}^n + \varphi_{yy}^n(\varphi_x^n)^2}{((\varphi_x^n)^2 + (\varphi_y^n)^2)^{3/2}} + \frac{2\pi}{l} \right) \times ((\varphi_x^n)^2 + (\varphi_y^n)^2)^{1/2} = 0$$

이제 전체적인 알고리즘을 설명하면 다음과 같다.

1.  $\mathbb{R}^2$  위에 임의의 simple closed curve 를 그린다.
2. simple closed curve 를 포함하는 임의의 정사각형 subset 을 잡는다. 이를 다시 정사각형의 mesh 로 자른다.
3. tubular neighborhood 를 잡고  $\varphi^0$  를 계산한다.
4. curve 의 둘레를 계산한다. curve 위의 임의의 점들을 택한 후  $l = \int |c'(s)|ds$  을 이용하여 근사치를 구한다.
5. 위의 차분방정식을 풀어 tubular neighborhood 내의 점들에 대해  $\varphi^1$  을 계산한다. tubular neighborhood 이외의 점에서는  $\varphi^1$  의 함수값을  $\varphi^0$  의 함수값과 같도록 만들어준다.

6. 4와 5를 반복한다. curve가 tubular neighborhood를 넘어가게 되면 3번으로 돌아간다. 또한 50회의 iteration 동안 tubular neighborhood의 업데이트가 진행되지 않았다면 3번으로 돌아간다.

## 4.2 Plateau problem

Plateau problem을 풀기 위하여 두 개의 정사각형 모양의 curve를 포함하는 minimal surface를 찾는 알고리즘을 만들었다. Plateau problem에서는 3차원의 공간에서 미분방정식을 풀어야 한다. 따라서 위의 미분방정식에  $z$  항을 추가해야 한다. 따라서 우리가 풀어야 할 차분방정식은 다음과 같다.

$$\frac{\varphi^{n+1}(x_i, y_j, z_k) - \varphi^n(x_i, y_j, z_k)}{\Delta t} + F_{ext}^n \times |(\varphi_x^n(x_i, y_j, z_k), \varphi_y^n(x_i, y_j, z_k))| = 0$$

$$F_{ext}^n = \pm H \text{ (depends on the orientation of the surface)}$$

Chopp에 의하면 [5]  $H$ 를  $\varphi^n$ 에 관하여 나타낸 식은 다음과 같다.

$$H = \frac{\varphi_{xx}^n((\varphi_y^n)^2 + (\varphi_z^n)^2) + \varphi_{yy}^n((\varphi_x^n)^2 + (\varphi_z^n)^2) + \varphi_{zz}^n((\varphi_x^n)^2 + (\varphi_y^n)^2) - 2\varphi_{xy}^n\varphi_x^n\varphi_y^n - 2\varphi_{yz}^n\varphi_y^n\varphi_z^n - 2\varphi_{xz}^n\varphi_x^n\varphi_z^n}{2((\varphi_x^n)^2 + (\varphi_y^n)^2 + (\varphi_z^n)^2)^{3/2}}$$

이제 위의 식들에 central approximation method를 통해 구한  $\varphi$ 의 도함수들의 값을 대입하면 차분방정식을 풀 수 있다. plateau problem의 알고리즘은 isoperimetric problem과 크게 다르지 않으나, surface가 항상 주어진 두 curve를 포함해야하기 때문에 제약조건이 발생하게 된다. 따라서  $\Delta t$ 의 시간동안 surface를 mean curvature flow에 따라 이동시킨 후, surface를 주어진 두 curve에 다시 붙여주는 작업이 반드시 필요하다. 따라서 전체적인 알고리즘을 설명하면 다음과 같다.

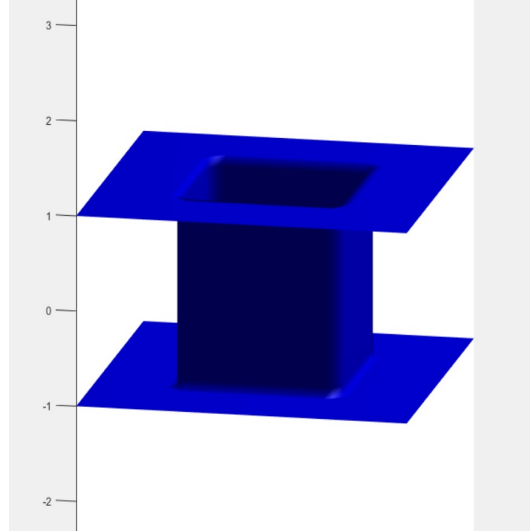


Figure 3: a surface containing two squares

1.  $\mathbb{R}^3$  위에 한 변의 길이가 2인 정사각형 2개를 그린다.
2. 두 정사각형을 포함하는 임의의 직육면체 subset을 잡는다. 이를 다시 정사각형의 mesh로 자른다.

3. 두 정사각형을 포함하는 surface을 Fig. 3과 같이 그린다. Fig. 3에서는 일단 두 정사각형을 잇는 기둥 모양의 surface를 그린 후, 두 정사각형과 평행하면서  $z$ 축과 수직인 두 평면을 그려주었다.

4. tubular neighborhood를 잡고  $\varphi^0$ 를 계산한다. 3에서 정의한 surface는 정의역을 두 connected component로 나누는데, 기둥의 안쪽과 연결되어있는 component들의 점에는 양수의 signed distance를 주고, 다른 component의 점들에는 음수의 signed distance를 준다.

5. 위의 차분방정식을 풀어 tubular neighborhood 내의 점들에 대해  $\varphi^1$ 을 계산한다. Fig 3을 살펴보면  $|z| = 1$ 이면서 정사각형 경계에 포함되지 않는 점에 대해서는 mean curvature가 0이므로 mean curvature flow에 영향을 받지 않음을 알 수 있다. 하지만 정사각형 경계 위에 있는 점들이나,  $|z| < 1$ 인 점들에 대해서는 mean curvature flow에 의해 이동함을 알 수 있다. 따라서  $\varphi^1$ 의 zero level set은 정사각형 경계를 벗어나게 된다. 새로운 곡면을 다시 정사각형 경계에 붙여줘야 하는데, 새로운 zero level set을 다음과 같이 정의한다.

$$A = \{(x, y, z) \mid \varphi^1(x, y, z) = 0, |z| < 1\} \cup \{(x, y, z) \mid \varphi^0(x, y, z) = 0, |z| = 1\}$$

6. 새로운 zero level set의 tubular neighborhood를 잡고 새로운  $\varphi^1$  값을 계산한다.

7. 평형에 이를 때까지 5를 반복한다.

## 5 Results

### 5.1 Isoperimetric problem

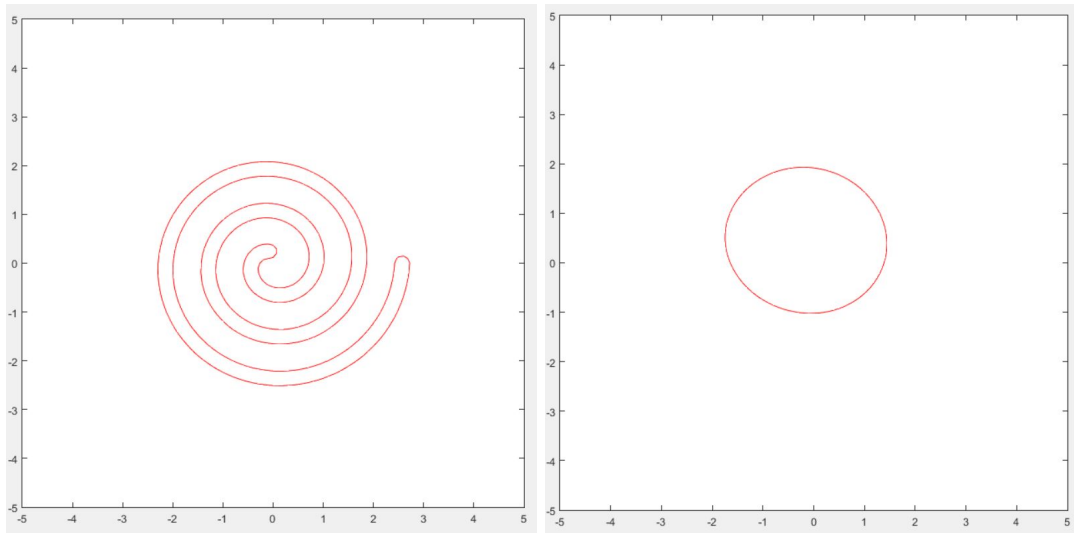


Figure 4: a simple closed curve converging to a circle

$[-5, 5] \times [-5, 5]$ 를  $100 \times 100$ 의 mesh로 자른 후 Fig. 4 왼쪽의 곡선에 대해 위의 알고리즘을 적용하였다.  $\Delta t = 0.5 \times (\frac{10}{100})^2$ 이었다. 총 500번 iteration한 결과, 원으로 수렴하는 것을 확인할

수 있었다. 넓이는 7.25에서 7.37로 1.6%의 변화를 보였다. 이를 통해 수치적으로 isoperimetric inequality가 성립함을 확인할 수 있었다.

## 5.2 Plateau problem

$[-2, 2] \times [-2, 2] \times [-5, 5]$ 를  $60 \times 60 \times 100$ 로 자른 후 한 변의 길이가 2인 정사각형 두 개를 포함하는 minimal surface를 구해보았다.  $\Delta t = 0.2 \times (\frac{10}{100})^2$ 이었다. 총 500번 iteration한 결과, 두 정사각형 사이의 거리가 2, 3일 때는 catenoid와 같은 꼴의 minimal surface가 형성됨을 확인할 수 있었다. 하지만 두 정사각형 사이의 거리가 4일 때는 두 정사각형을 연결하는 어떠한 surface도 형성되지 않음을 확인할 수 있었다.

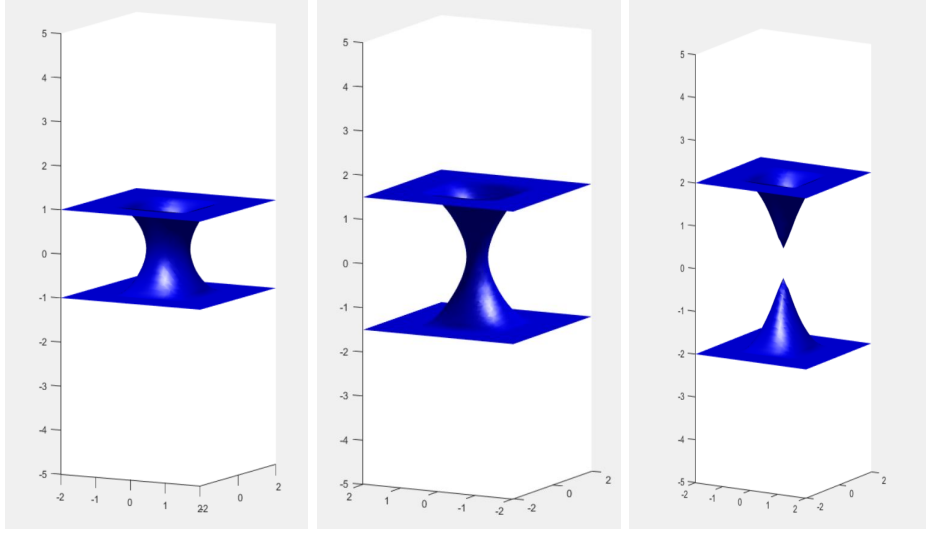


Figure 5: minimal surfaces between square grids

## 6 Discussion

이번 보고서에서는 level set method를 이용하여 curvature flow에 따른 curve 또는 surface의 움직임을 모델링하였다. 하지만 level set method를 이용해서 구한 해의 수렴성에 대해서는 아직 명확히 증명되지 않았다. level set method 이외에도 minimal surface를 찾는 여러 가지 방법들이 존재한다. Dziuk와 Hutchinson은 [6] minimal surface를 Dirichlet energy를 최소화하는 곡면으로 정의하고, finite element method를 이용하여 minimal surface를 구하였다. 또한 해의 수렴성까지 증명하였다.

$$u : D \rightarrow \mathbb{R}^3 \text{ minimal} \longleftrightarrow D(u) = \frac{1}{2} \int_D |\nabla u|^2 \text{ has a critical value}$$

이러한 점에서 level set method는 finite element method와 같은 기존의 수치적 방법에 비해 수학적으로 엄밀하지 않다고 할 수 있다. 하지만 임의의 n차원의 hypersurface에 대해 방법을 적용할 수 있기 때문에, 물리나 생물, 컴퓨터 공학과 같은 다양한 분야에 응용될 수 있다는 점은 level set method의 장점이라고 할 수 있다.



## References

- [1] M. Gage, R. S. Hamilton, *The heat equation shrinking convex plane curves*, Journal of Differential geometry 23 (1986), 69–96.
- [2] M. A. Grayson, *The heat equation shrinks embedded plane curves to round points*, Journal of Differential geometry 26 (1987), 285–314.
- [3] D. Adalsteinsson, J. A. Sethian, *The fast construction of extension velocities in level set methods*, Journal of computational physics 148 (1999), 2–22.
- [4] J. A. Sethian, *Fast marching methods and level set methods for propagating interfaces*, von Karman institute lecture series (1998).
- [5] D. L. Chopp, *Computing Minimal surfaces via level set curvature flow*, Journal of computational physics 106 (1993), 77–91.
- [6] G. Dziuk, J. E. Hutchinson *The discrete Plateau problem: Algorithm and Numerics*, Mathematics of computation 68 (1999), 1–23.