

# 2D Tolerance Analysis

Ziqi Wang

## 1 Description

Error happens everywhere in manufacture. Especially during assembling process, overlarge items are usually hard to be fitted into the assembly. One common solution would be adding tolerance within connected parts. (show in Fig.1(a), each triangles is shrunk by 90%). However, gaps or even holes appear inside the assembly (shows in Fig.1(b)). More importantly, gap accumulation is recognised as a non-linear progress. This paper imposes a methodology to analysis the gap accumulation in 2D.

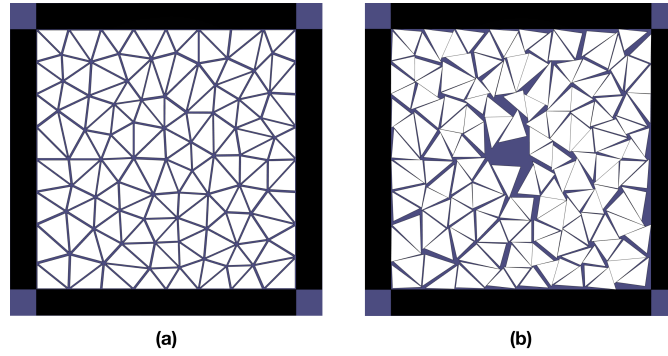


Figure 1: Triangles (white) in (a) are shrunk and restricted by the boundary (black). To enlarge the gap among the triangles, optimizing triangles' position as well as its orientation. (b) shows triangles' layout after optimization.

## 2 Methodology

### 2.1 Signed distance

Before proposing the optimization method, we firstly define the signed distance between two polygons  $P, Q$  as Fig.2. If collision happens, the signed distance  $d(P, Q)$  should be smaller than 0.

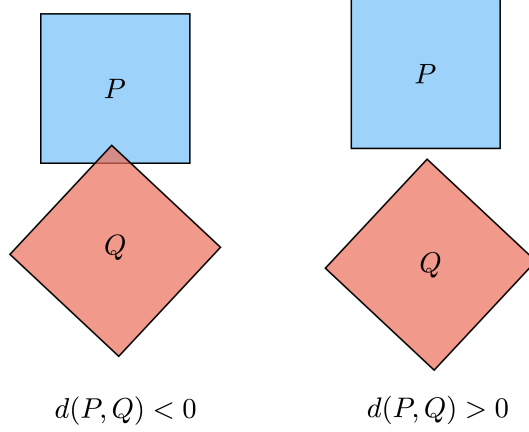


Figure 2: (a) two polygons are penetrated (b) two polygons are collisional.

Inspired by the SAT(separated axis theorem), we always can find two points  $\mathbf{p} \in P, \mathbf{q} \in Q$  and a normal  $\mathbf{n}$  which is perpendicular to one of  $P$  or  $Q$ 's edges. The signed distance can be written as (shows in Fig:3)

$$d(P, Q) = \mathbf{n}^T(\mathbf{p} - \mathbf{q}) \quad (1)$$

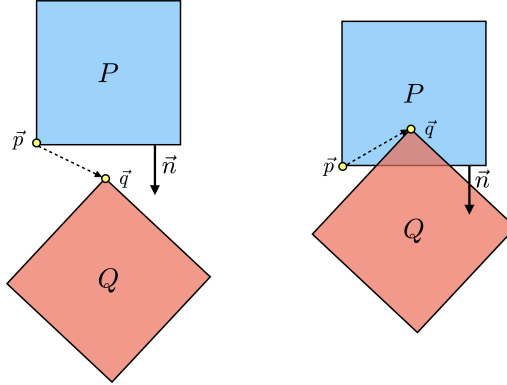


Figure 3:  $\mathbf{p}, \mathbf{q}, \mathbf{n}$  when two polygons are separated or collisional

Since  $P, Q$  are movable polygons during optimization, we have to express  $d(P, Q)$  as a function of their translation and rotation. Let us denote  $\mathbf{v}^P = (\theta^P, x^P, y^P)$  and  $\mathbf{v}^Q = (\theta^Q, x^Q, y^Q)$  are  $P, Q$ 's status variable. Then Eq.1 is transformed as:

$$d(P(\mathbf{v}^P), Q(\mathbf{v}^Q)) = \mathbf{n}(\mathbf{v}^P, \mathbf{v}^Q)^T(\mathbf{p}(\mathbf{v}^P) - \mathbf{q}(\mathbf{v}^Q)) \quad (2)$$

Pleased note that  $\mathbf{n}(\mathbf{v}^P, \mathbf{v}^Q), \mathbf{p}(\mathbf{v}^P), \mathbf{q}(\mathbf{v}^Q)$  may change rapidly due to the unsmoothness of  $P, Q$ 's face. In order to simplify the optimization, we approximate

each  $\mathbf{n}(\mathbf{v}^P, \mathbf{v}^Q)$ ,  $\mathbf{p}(\mathbf{v}^P)$ ,  $\mathbf{q}(\mathbf{v}^Q)$  locally. We assume that these variables won't change dramatically within their tiny neighbor.

Thus, the signed distance  $d(P(\mathbf{v}^P), Q(\mathbf{v}^Q))$  can be linearized as:

$$d(P(\mathbf{v}^P), Q(\mathbf{v}^Q)) \approx d(P(\mathbf{v}_0^P), Q(\mathbf{v}_0^Q)) + Jd(P(\mathbf{v}_0^P), Q(\mathbf{v}_0^Q)) \begin{bmatrix} (\mathbf{v}^P - \mathbf{v}_0^P)^T \\ (\mathbf{v}^Q - \mathbf{v}_0^Q)^T \end{bmatrix} \quad (3)$$

More specifically, Let suppose  $n$  is perpendicular to one of  $P$ 's edges. and

$$\mathbf{p}(\mathbf{v}^P) = \begin{bmatrix} \cos(\theta^P) & -\sin(\theta^P) \\ \sin(\theta^P) & \cos(\theta^P) \end{bmatrix} (\mathbf{p} - \hat{\mathbf{p}}) + \begin{bmatrix} x^P \\ y^P \end{bmatrix} + \hat{\mathbf{p}} \quad (4)$$

$$\mathbf{q}(\mathbf{v}^Q) = \begin{bmatrix} \cos(\theta^Q) & -\sin(\theta^Q) \\ \sin(\theta^Q) & \cos(\theta^Q) \end{bmatrix} (\mathbf{q} - \hat{\mathbf{q}}) + \begin{bmatrix} x^Q \\ y^Q \end{bmatrix} + \hat{\mathbf{q}} \quad (5)$$

where  $\mathbf{p}, \mathbf{q}$  is the position of  $\mathbf{p}(\mathbf{v}^P), \mathbf{q}(\mathbf{v}^Q)$  when optimization starts. And  $\hat{\mathbf{p}}, \hat{\mathbf{q}}$  are the central points of the polygons  $P, Q$ . if denote the rotation matrix in Eq.4 and 5 as  $R_{\theta^P}, R_{\theta^Q}$ .

Then the  $d(P, Q)$  can be expressed as:

$$d(P, Q) = \mathbf{n}(\mathbf{v}^P)^T (\mathbf{p}(\mathbf{v}^P) - \mathbf{q}(\mathbf{v}^Q)) \quad (6)$$

$$= \mathbf{n}(\mathbf{v}^P)^T [R_{\theta^P}(\mathbf{p} - \hat{\mathbf{p}}) + \begin{bmatrix} x^P \\ y^P \end{bmatrix} + \hat{\mathbf{p}} - R_{\theta^Q}(\mathbf{q} - \hat{\mathbf{q}}) - \begin{bmatrix} x^Q \\ y^Q \end{bmatrix} - \hat{\mathbf{q}}] \quad (7)$$

$$= f(\mathbf{v}^P) + g(\mathbf{v}^Q) \quad (8)$$

For  $f(\mathbf{v}^P)$ , using Talyor expanding series:

$$f(\mathbf{v}^P) = \mathbf{n}(\mathbf{v}^P)^T (R_{\theta^P}(\mathbf{p} - \hat{\mathbf{p}}) + \begin{bmatrix} x^P \\ y^P \end{bmatrix} + \hat{\mathbf{p}}) \quad (9)$$

$$\approx [\mathbf{n}_0 + \begin{bmatrix} -\mathbf{n}_{0y} \\ \mathbf{n}_{0x} \end{bmatrix} (\theta^P - \theta_0^P)] [\mathbf{p}_0 + R'_{\theta_0^P}(\mathbf{p} - \hat{\mathbf{p}})(\theta^P - \theta_0^P) + \begin{bmatrix} x^P - x_0^P \\ y^P - y_0^P \end{bmatrix}] \quad (10)$$

where  $\mathbf{n}_0 = \mathbf{n}(\mathbf{v}_0^P)$ ,  $\mathbf{p}_0 = \mathbf{p}(\mathbf{v}_0^P)$

Same in  $g(\mathbf{v}^Q)$ :

$$g(\mathbf{v}^Q) \approx [\mathbf{n}_0 + \begin{bmatrix} -\mathbf{n}_{0y} \\ \mathbf{n}_{0x} \end{bmatrix} (\theta^Q - \theta_0^Q)] [\mathbf{q}_0 + R'_{\theta_0^Q}(\mathbf{q} - \hat{\mathbf{q}})(\theta^Q - \theta_0^Q) + \begin{bmatrix} x^Q - x_0^Q \\ y^Q - y_0^Q \end{bmatrix}] \quad (11)$$

where  $\mathbf{q}_0 = \mathbf{q}(\mathbf{v}_0^Q)$

And combining these two function as:

$$d(P, Q) = a_0 + a_1\theta^P + a_2x^P + a_3y^P + a_4\theta^Q + a_5x^Q + a_6y^Q \quad (12)$$

The  $a_i$  can be written as:

$$a_1 = \mathbf{n}^T R'_{\theta_0^P}(\mathbf{p} - \hat{\mathbf{p}}) + \mathbf{n}_1^T(\mathbf{p}_0 - \mathbf{q}_0) \quad (13)$$

$$a_2 = \mathbf{n}_{0x} \quad (14)$$

$$a_3 = \mathbf{n}_{0y} \quad (15)$$

$$a_4 = -\mathbf{n}^T R'_{\theta_0^Q}(\mathbf{q} - \hat{\mathbf{q}}) \quad (16)$$

$$a_5 = -\mathbf{n}_{0x} \quad (17)$$

$$a_6 = -\mathbf{n}_{0y} \quad (18)$$

$$a_0 = (\mathbf{n}_0 - \mathbf{n}_1 \theta_0^P)^T(\mathbf{p}_0 - \mathbf{q}_0) - \mathbf{n}_0^T R'_{\theta_0^P}(\mathbf{p} - \hat{\mathbf{p}}) \theta_0^P \quad (19)$$

$$- \mathbf{n}_0^T R'_{\theta_0^Q}(\mathbf{q} - \hat{\mathbf{q}}) \theta_0^Q + \mathbf{n}_0^T \begin{bmatrix} x_0^Q - x_0^P \\ y_0^Q - y_0^P \end{bmatrix} \quad (20)$$

where

$$\mathbf{n}_1 = \begin{bmatrix} -\mathbf{n}_{0y} \\ \mathbf{n}_{0x} \end{bmatrix} \quad (21)$$

## 2.2 Improved signed distance

However, only using two points  $(\mathbf{p}, \mathbf{q})$  to describe the signed distance may have problems when the two polygons have face in parallel.

In Fig.4, the signed distance actually drop while our approximated function

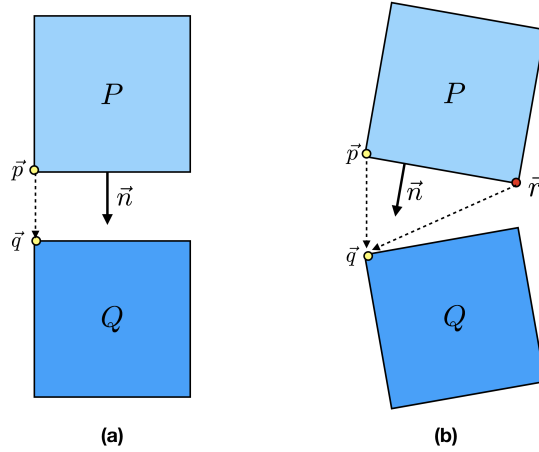


Figure 4: An unappropriate approximation of signed distance when two polygons have faces in parallel.

indicated an increase in distance. Let's denote the distance from  $(\mathbf{p}, \mathbf{q}, \mathbf{n})$  as

$d_{\mathbf{p},\mathbf{q},\mathbf{n}}(P, Q)$ . Then apparently in Fig.4:

$$d_{\mathbf{p},\mathbf{q},\mathbf{n}}(P, Q) > d_{\mathbf{r},\mathbf{q},\mathbf{n}}(P, Q) \quad (22)$$

To resolve this problem, we use two pairs of  $(\mathbf{p}, \mathbf{q}), (\mathbf{r}, \mathbf{q})$  to approximate the  $d(P, Q)$ :

$$d(P, Q) = \min(d_{\mathbf{p},\mathbf{q},\mathbf{n}}(P, Q), d_{\mathbf{r},\mathbf{q},\mathbf{n}}(P, Q)) \quad (23)$$

### 2.3 Collision resolve