Maximales Quadrat in Polygon

Marco Romanutti[1,2]

[1] *Fachhochschule Nordwestschweiz FHNW, Brugg*
[2] `Effiziente Algorithmen, Klasse 5Id`

Gegeben ist ein geschlossenes Polygon in der Ebene, welches sich selbst nicht schneidet. Gesucht ist das grösste Quadrat welches vollständig innerhalb des Polygons liegt. Das Programm soll alle Polygone aus der Datei polygons.in einlesen und für jedes eingelesene Polygon das maximale Quadrat bestimmen und ausgeben.

1 Ansatz

Es wird versucht, möglichst grosse Quadrate entlang der Kanten eines Polygons zu konstruieren. Eine Lösung ist dann gültig, wenn **alle Eckpunkte und Kanten des Quadrats innerhalb oder auf den Eckpunkten/Kanten des Polygons liegen**. In den nachfolgenden Abschnitten werden die Schritte beschrieben, die ausgehend von einer Kante des Polygons das grösste Quadrat finden. Die Schritte werden für jede Kante des Polygons durchgeführt. Visualisierungen zu den einzelnen Schritten befinden sich auf der letzten Seite.

Schritt 1

Ein initiales Quadrat wird auf Basis der jeweiligen Kantenlänge des Polygons erzeugt (vgl. Abbildung 3 auf der letzten Seite). Dieses Quadrat wird mit einer fixen Schrittweite solange verkleinert, bis das Quadrat vollständig im Polygon liegt (vgl. Abbildung 4). Dies ist der Fall, wenn folgende Bedingungen erfüllt sind:

- Keine der Kanten des Quadrats schneidet eine der Kanten des Polygons
- Kein Eckpunkt des Polygons liegt in der Fläche des Quadrats[1]

Der Fall, dass es sich beim Polygon um ein Dreieck handelt und dessen Eckpunkte drei Punkte des Quadrats umfassen, muss zusätzlich abgefangen werden. Abbildung 1 zeigt ein solches ungültiges Beispiel.

[1] Wobei Eckpunkte des Quadrats, welche auf Eckpunkten/Kanten des Polygons liegen, gültig sind

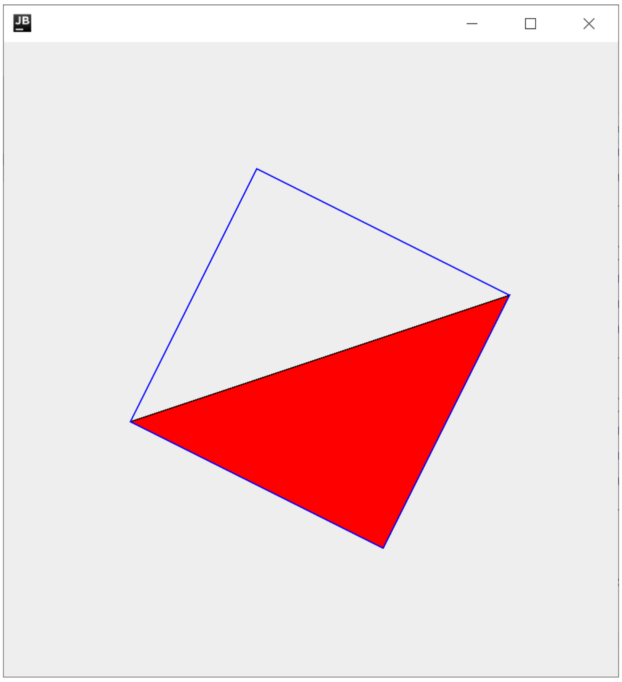

Figure 1: *Ungültiges Quadrat aus Schritt 1 obwohl sich keine der Kanten schneiden und alle Punkte des Polygons auf den Eckpunkten des Quadrats liegen*

Schritt 2

Anschliessend wird das Quadrat mit einer variablen Schrittweite soweit vergrössert, bis eine Kante oder ein Eckpunkt des Quadrats auf einer Kante oder einem Eckpunkt des Polygons liegt (vgl. Abbildung 5). Falls zu grosse Schrittweiten gewählt wurden, werden diese bis zu einem gegebenen Epsilon verkleinert.

Schritt 3

Bisher wurde für alle Transformationen des Quadrats angenommen, dass diese in der Mitte der jeweiligen Kante des Polygons liegt. In einem weiteren Schritt wird das Polygon auf beide Seiten verschoben und vom neuen Standort aus jeweils geprüft, ob es vergrössert werden kann (vgl. Schritt 2 und Abbilldung 6).

Eine Optimierung könnte mit der Heuristik erreicht werden, dass jeweils mit der längsten Kante des Polygons gestartet wird. Der Ansatz liefert für viele Beispiele gute Resultate. Ein möglicher Schwachpunkt ist jedoch die Abhängigkeit der Kanten: Bei einem sternförmigen Polygon mit grosser Fläche in der Mitte und spitzen Zacken würde das Programm keine guten Resultate liefern.

2 Anleitung Software

Voraussetzung für richtige Resultate sind im **Gegenuhrzeigersinn** aufgeführte Eckpunkte im Input-File, z.B:

```
10.5
2
3 105,125 170,130 150,300
4 100,100 500,100 500,300 100,300
```

Das Programm kann mit der `main`-Methode der Klasse `Solver` gestartet werden und **die Optimierung läuft anschliessend maximal solange wie das Zeitlimit aus dem Input-File**[2]. Mit der Methode `render` wird anschliessend für jedes Polygon die beste Lösung visualisiert[3]. Die Nummer des zu visualisierenden Polygons kann in der Konsole eingegeben werden. Die Korrektheit wurde mittels verschiedener konvexer und nicht-konvexer Polygone überprüft. Die Test-Inputs liegen im Verzeichnis `resources`. Abbildung 2 zeigt die Visualisierung der besten Lösung von Polygon 6 aus der Input-Datei.

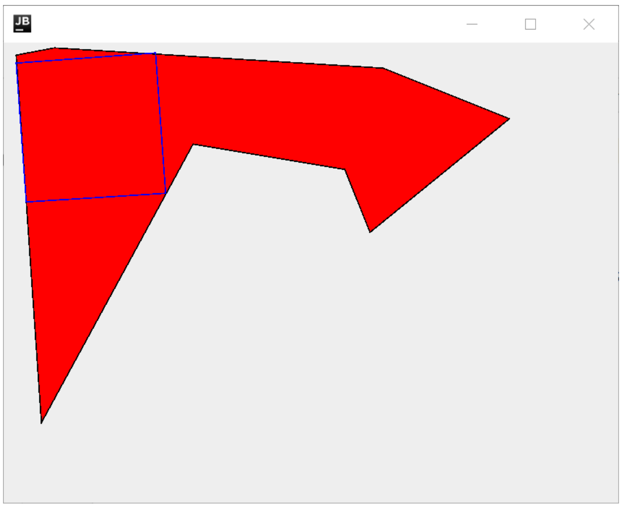

Figure 2: *Maximales Quadrat für das Polygon 6*

[2] Messbeginn ist Start der Optimierungsfunktion

[3] Durch die Verwendung von Integer-Zahlen sind die Visualisierungen teilweise etwas ungenau!

References

[1] Mathoverflow: Largest inscribed rectangle, `https://mathoverflow.net/questions/105837/`

[2] Stackoverflow: Point inside rectangle, `https://stackoverflow.com/questions/17136084/`

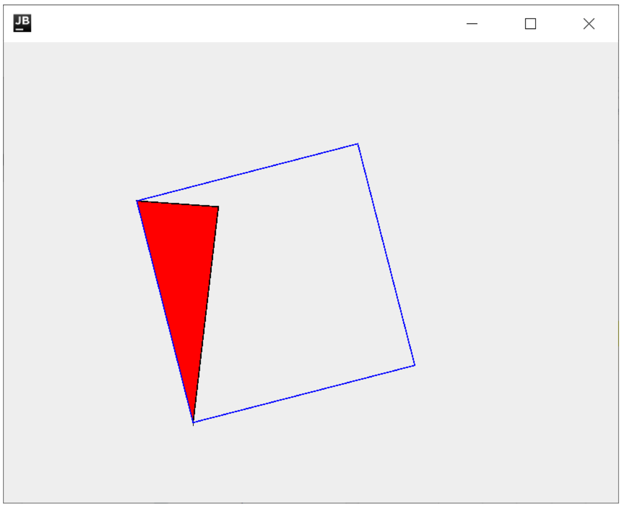

Figure 3: *Schritt 1, Initiales Quadrat das verkleinert werden muss für eine gültige Lösung*

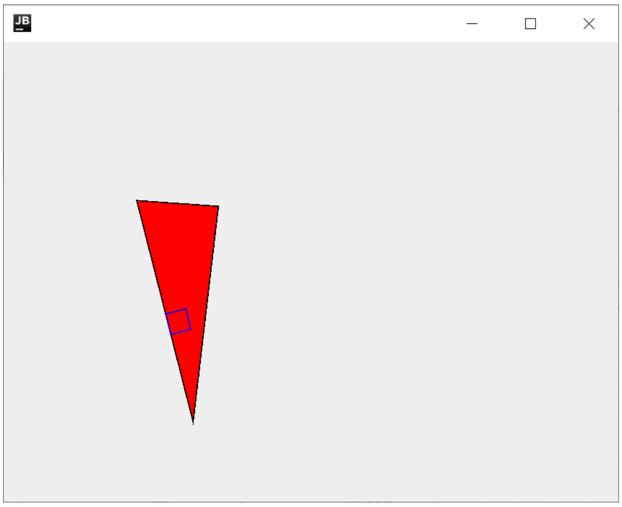

Figure 4: *Schritt 1, Verkleinertes Quadrat - gültig aber noch nicht das grösstmögliche Quadrat*

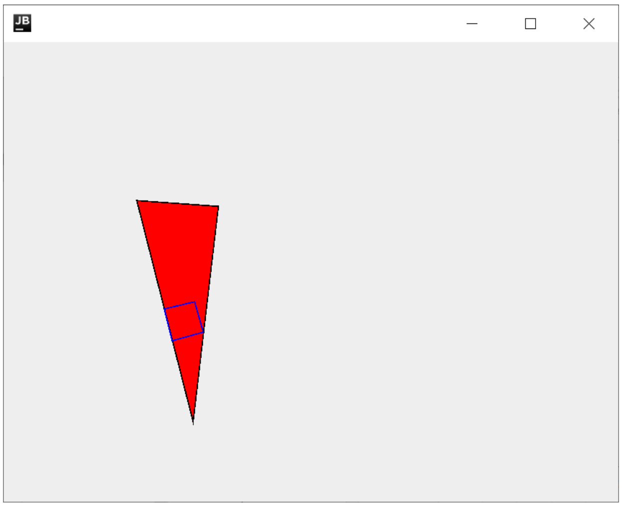

Figure 5: *Schritt 2, Vergrössertes aber immer noch zentriertes Quadrat*

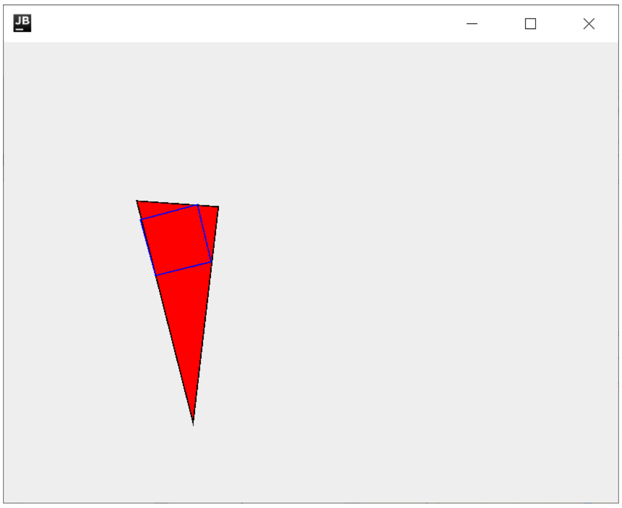

Figure 6: *Schritt 3, Verschobenes und maximiertes Quadrat für die aktuelle Kante*