

Pfadgenerierung/Polygone/Polygonoffsetting

Jan Stenzel

17. Juni 2015

Gliederung I

- 1 Polygone
 - Definition
 - konkav, konvex und überschlagen

- 2 Clipping
 - Was kann passieren?
 - Ablauf des Algorithmus
 - Ergebnisse

- 3 Polygonoffsetting durch Ausrechnen der Umlaufzahlen
 - Warum sollte man diesen Algorithmus verwenden?
 - Wie funktioniert der Algorithmus?
 - Umlaufzahlen
 - äußeres, inneres Offset
 - Ablauf für inneres/äußeres Offset

Polygone I

Definition (Polygon)

Ein Polygon ist eine ebene Figur, die aus verbundenen Liniensegmenten besteht. Ein Polygon hat eine endliche Anzahl von Ecken und Kanten, sodass genau 2 Kanten an einer Ecke aufeinander treffen. (Die Liniensegmente sind also geschlossen.)

Ein Polygon kann konvex, konkav oder überschlagen sein.

Polygone I

Was bedeutet konkav/konvex und überschlagen?

Definition (konvex)

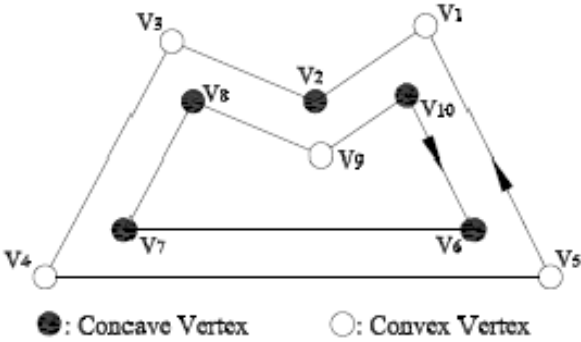
Eine Ecke heißt konvex, wenn eine linke Drehung gemacht wird, während man die Kontur entlang läuft. Ein Polygon, das nur konvexe Ecken hat, nennt man konvexes Polygon.

Definition (konkav)

Eine Ecke heißt konkav, wenn eine rechte Drehung gemacht wird, während man die Kontur entlang läuft.

Polygone I

Beispiel:



Polygone I

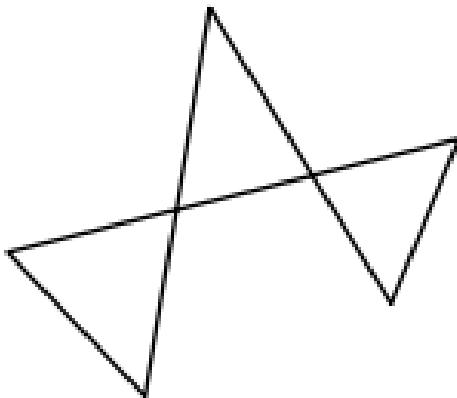
Was bedeutet konkav/konvex und überschlagen?

Definition (überschlagen)

Ein Polygon heißt überschlagen, wenn sich die Kanten an Punkten schneiden, die keine Eckpunkte des Polygons sind.

Polygone I

Beispiel:



Clipping I

zur Einleitung...

- Wozu braucht man Clipping?
- Was ist Clipping?

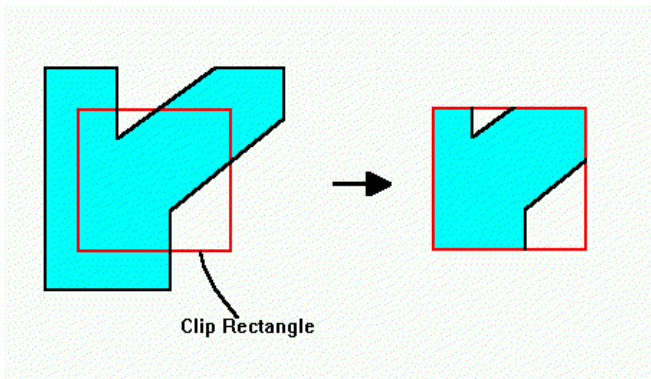
Clipping I

Was ist Clipping? / Wozu braucht man Clipping?

- Schneiden bezüglich eines Rechteckfensters oder eines beliebigen Polygons
- alles andere fällt weg
- 2 Polygone: Subjekt und Clip Polygon
- entweder sehr zeitaufwendig oder nur für eine bestimmte Art von Polygonen
- Oberflächen mit hoher Qualität produzieren

Clipping I

Beispiel:



Clipping I

Was kann passieren?

- 1** Kanten müssen:
 - 1.1 vollständig übernommen werden
 - 1.2 teilweise übernommen werden (innen nach außen)
 - 1.3 teilweise übernommen werden (außen nach innen)
- 2** Kanten müssen gar nicht übernommen werden. Es kann dabei sein, dass das Polygon komplett verschwindet.
- 3** Neue Kanten müssen aufgenommen werden (entlang des Clip Polygons)

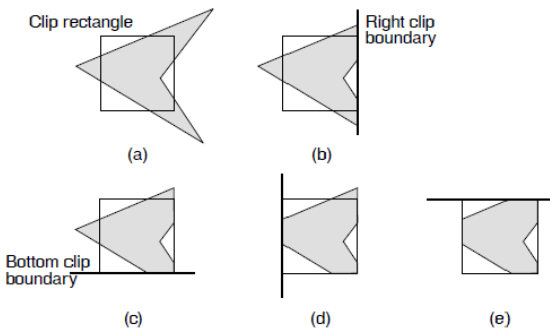
Clipping I

Ablauf des Algorithmus

- nur für konvexe Clip Polygone geeignet
- Polygon von oben nach unten einscannen
- AET (Active Edge Table) sammelt Informationen
- findet lokale Minima/Maxima und zwischen liegende Ecken
- Kanten definiert durch zwei Buchstaben

Clipping I

Der Ablauf:



Clipping I

Ergebnisse

- Trapeze als Output beim Clipping
- keine konkaven Polygone zugelassen
- schneller als nur den Füllalgorithmus anzuwenden

Polygonoffsetting I

Auch zu diesem Thema zur Einleitung:

- Wozu braucht man Offsetting?
- Was ist Offsetting?

Was ist Offsetting und wozu braucht man es? I

Wozu braucht man Offsetting? / Was ist Offsetting?

- Input: beliebig viele Polygone, die sich nicht überlappen
- Output: Konturen des Offset-Polygons
- Absetzen von den äußeren Konturen
- Offset kann mehrfach bestimmt werden
- einen Pfad generieren (zur Befüllung des Inneren)
- Herstellung von Mulden / Löchern

Polygonoffsetting I

Warum dieser Algorithmus?

Der vorgestellte Algorithmus nutzt positive Umlaufzahlen, um nicht-überlappende Polygone abzusetzen. Bei diesem Algorithmus sind die Löcher frei wählbar.

Polygonoffsetting I

- Eingabe: sich nicht überlappende 2D-Polygone
- haben äußere und innere Konturen
- Anwenden der Rechte-Hand-Regel-Konvention

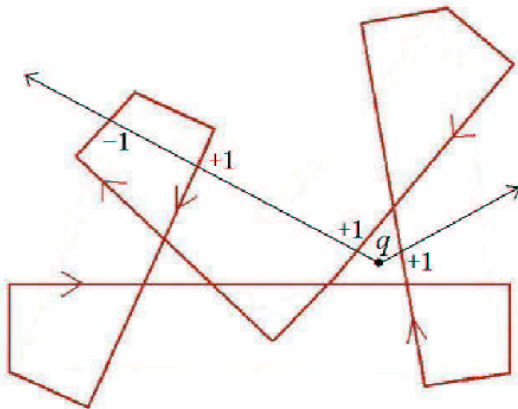
Polygonoffsetting I

Umlaufzahlen (Definition):

- Gegeben sei ein zweidimensionaler euklidischer Raum (\mathbb{R}^2). Es sei P ein Polygon in diesem Raum und q ein Punkt in diesem Raum. Es sei R ein Strahl von q ins Unendliche, der keine Ecke eines Polygons trifft.
- $\omega(R, P) = \sum_{e_i \in P} \psi(R, e_i)$
- $\psi(R, e_i) = \begin{cases} 0 & , \text{ wenn } R \text{ } e_i \text{ nicht unterbricht} \\ 1 & , \text{ wenn } e_i \text{ aus Sicht von } q \text{ CCW gekreuzt wird} \\ -1 & , \text{ wenn } e_i \text{ aus Sicht von } q \text{ CW gekreuzt wird} \end{cases}$

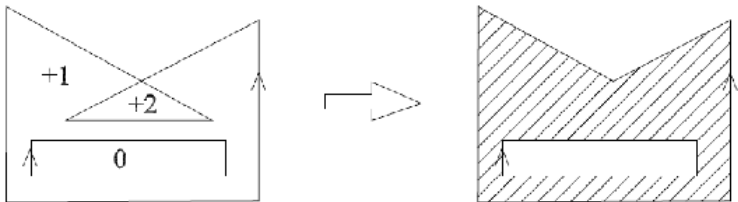
Polygonoffsetting I

Beispiel für Berechnung der Umlaufzahlen:



Polygonoffsetting I

Umgang mit überschlagenen Polygonen:



Polygonoffsetting I

Definition (äußeres, inneres Offset)

Bei dem äußeren Offset wird ein Polygon nach Außen abgesetzt.
Beim inneren Offset wird ein Polygon nach Innen abgesetzt.

Polygonoffsetting I

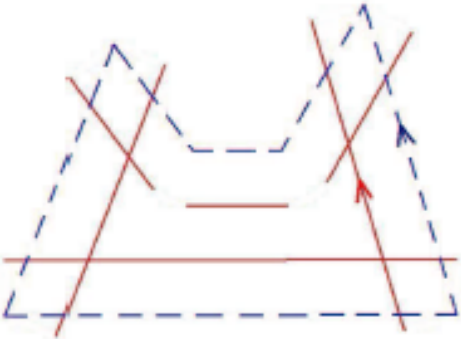
Ablauf für inneres Offset

- 1 Jede Kante nach innen absetzen.
- 2 Verbinden der Liniensegmente zu einer Kurve.
- 3 Berechnung der Umlaufzahlen
- 4 Entfernen von ungültigen Stellen. Der gültige Teil wird beibehalten.

möglicher Fall: Kanten können verschwinden, wenn mehrere Offsets durchgeführt werden.

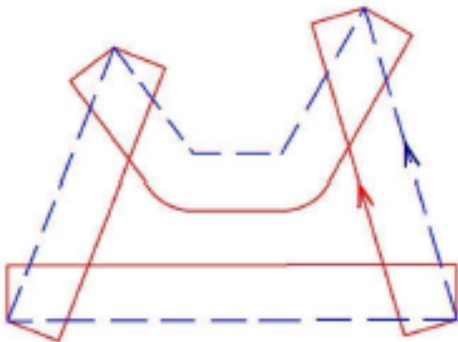
Polygonoffsetting I

1. Jede Kante nach innen absetzen:



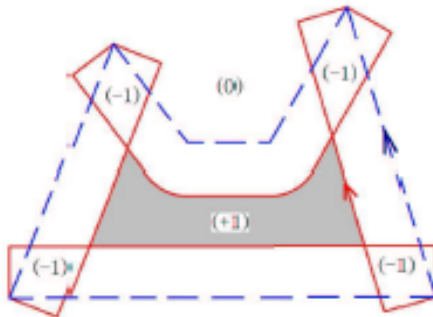
Polygonoffsetting I

2. Verbinden der Linien zu einer Kurve:



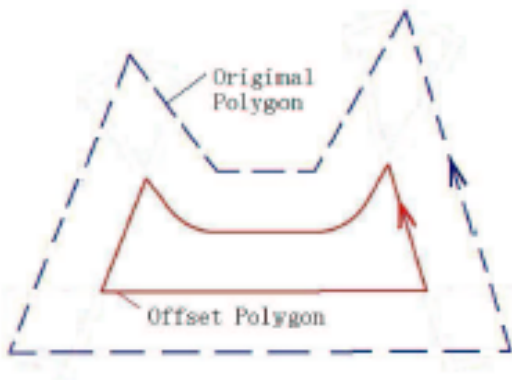
Polygonoffsetting I

3. Berechnung der Umlaufzahlen:



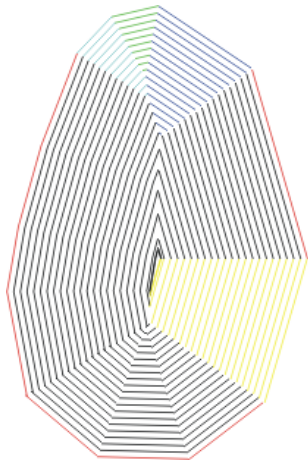
Polygonoffsetting I

4. Entfernen von ungültigen Stellen:



Polygonoffsetting I

Mehrfaches Anwenden:



Polygonoffsetting I

Ablauf für äußeres Offset

- 1 Jede Kante nach außen absetzen.
- 2 Verbinden der konvexen Ecken durch eine runden Bogen.
- 3 Berechnung der Umlaufzahlen.
- 4 Entfernen von ungültigen Stellen.

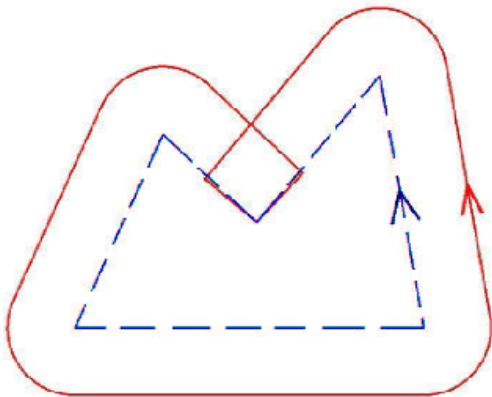
Polygonoffsetting I

1. Jede Kante nach außen absetzen:



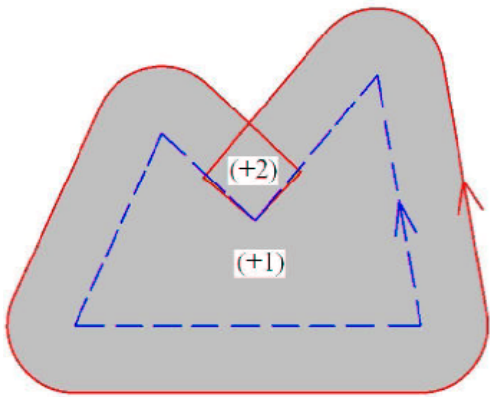
Polygonoffsetting I

2. Verbinden der konvexen Ecken durch einen runden Bogen:



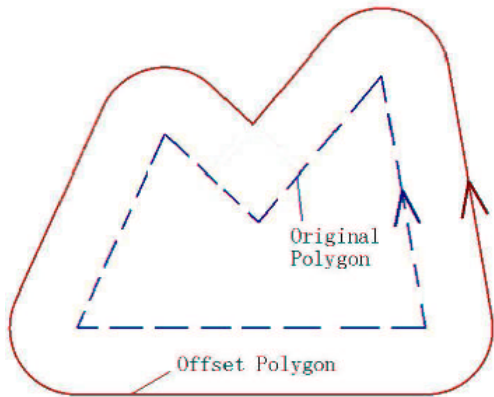
Polygonoffsetting I

3. Berechnung der Umlaufzahlen:



Polygonoffsetting I

4. Entfernen von ungültigen Stellen:



Polygonoffsetting I

Vereinfachung der Offset-Kurve

- Eliminierung der Kanten, die gleiche Geometrie, aber unterschiedliche Richtungen haben

Theorem 1:

Wenn zwei Kanten in einem Polygon die gleiche Geometrie, aber unterschiedliche Richtungen haben, wird die Umlaufzahl nicht geändert, wenn die beiden Kanten weggelassen werden.

Polygonoffsetting I

Beweis Beweis von Theorem 1: Angenommen, es gäbe zwei Kanten e_1 und e_2 in einem Polygon P , die diese Voraussetzungen erfüllen. Nach dem Entfernen der zwei Kanten erhält man ein neues Polygon P' . Die beiden Polygone haben die gleichen Ecken. Für einen Strahl R mit dem Endpunkt q , der keine Ecke von P und auch keine von P' enthält, trifft entweder auf keine der beiden Kanten oder auf beide Kanten. Wenn keine der Kanten getroffen wird, ist die Umlaufzahl unverändert. Wenn beide Kanten getroffen werden, ist ihre Verteilung entweder $+1$ und -1 oder -1 und $+1$. ■

Polygonoffsetting

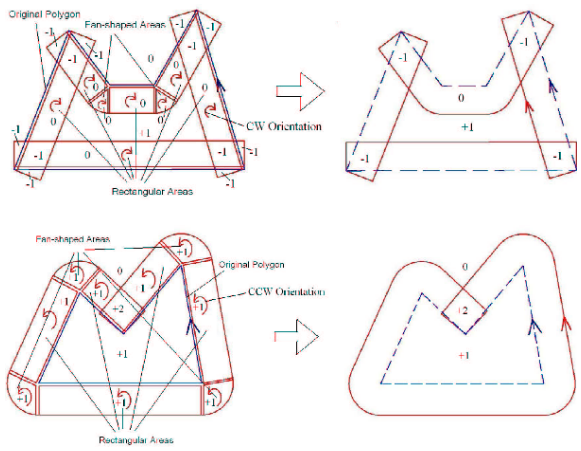
Beweis Beweis von Theorem 1 (Fortsetzung): Das ergibt:

$$\begin{aligned}\omega(q, P) &= \sum_{e_i}^P \psi(R_q, e_i) \\ &= \sum_{e_i \in P \setminus \{e_1, e_2\}} \psi(R_q, e_i) + \psi(R_q, e_1) + \psi(R_q, e_2) \\ &= \sum_{e_i \in P \setminus \{e_1, e_2\}} \psi(R_q, e_i) + 1 - 1 \\ &= \sum_{e_i \in P \setminus \{e_1, e_2\}} \psi(R_q, e_i) \\ &= \omega(q, P')\end{aligned}$$



Polygonoffsetting I

Der gesamte Algorithmus:



Polygonoffsetting I

Komplexität des Algorithmus

- zwei Schritte: Offset-Kurve bestimmen und Berechnung der Umlaufzahlen
- Es sei n die Anzahl der Liniensegmente und Bogensegmente in der Offset-Kurve
- Offset-Kurve: $O(n)$ Zeit und $O(n)$ Raum
- Umlaufzahlen bestimmen: $O((n + k) \log(n))$ Zeit und $O(n + k)$ Raum

Polygonoffsetting I

Testergebnisse

- 30 bis 150-mal schneller als andere Algorithmen
- korrekte Ergebnisse
- konsistente Ergebnisse

Ende! |

**Vielen Dank für eure Aufmerksamkeit.
Habt ihr noch Fragen?**

Quellen I

Quellen

Papers:

1. VATTI, B.R: A generic solution to polygon clipping
Communications of the ACM, Vol 35, Issue 7 (July 1992 pp
56-63)
2. CHEN, X & McMAINS, S:: Polygon offsetting by
Computing winding umbers, Proceedings of IDETC/CIE 2005
Published in: Robotics and Biomimetics (ROBIO), 2013 IEEE
International Conference on Date of Conference: 12-14 Dec
2013 Pages 74 - 78 INSPEC Accession Number: 14161647

Quellen II

Quellen

Internetquellen, hauptsächlich für die Bilder

- 1 Deussen O., “Clipping“, unter: https://www.inf.tu-dresden.de/content/institutes/smt/cg/teaching/lectures/CG1WS0203/secure/clipping_script.pdf (abgerufen am 12.06.2015)
- 2 Haowei H., “Computer Graphics - Clipping“, unter: <http://www.cc.gatech.edu/grads/h/Haowei.Hsieh/Haowei.Hsieh/mm.html> (abgerufen am 12.06.2015)
- 3 unter: [“http://www.math.com/tables/geometry/polygons.htm“](http://www.math.com/tables/geometry/polygons.htm) (abgerufen am 12.06.2015)