

# **Schnittpunkte von Strecken**

Dokumentation zur ersten Aufgabe im Modul  
„Computational Geometry“

Markus Hüttner, Michael Röder

## Inhalt

|   |   |
|---|---|
| 1. Beschreibung des Problems.....                   | 3 |
| 2. Lösungsansatz „CCW“ .....                        | 3 |
| 3. Abstraktion des Problems.....                    | 3 |
| 3.1. Der Standard Fall.....                         | 3 |
| 3.2. Sonderfall: Es existiert ein CCW von Null..... | 4 |
| 3.3. Sonderfall: Eine Strecke der Länge Null.....   | 4 |
| 4. Lösungsansatz.....                               | 5 |
| 4.1. Abdeckung der Sonderfälle.....                 | 5 |
| 4.2. Abdeckung des Standardfalls.....               | 5 |
| 4.3. Test der Implementierung.....                  | 5 |
| 5. Ergebnis.....                                    | 6 |

# 1. Beschreibung des Problems

In drei verschiedenen großen Dateien sind jeweils Zeile für Zeile Strecken in Form von Punkten angegeben. Es gilt zu ermitteln, wie viele Strecken sich pro Datei schneiden. Zudem ist dabei die für die Vergleiche benötigte Zeit zu erfassen.

## 2. Lösungsansatz „CCW“

In der Vorlesung wurde ein Konzept besprochen, mit dem man überprüfen kann, auf welcher Seite einer Strecke (im Sinne eines Halbstrahls) ein weiterer Punkt liegt. Die sogenannte CCW-Funktion liefert je nach Beschaffenheit der Koordinaten ein Ergebnis mit einem anderen Vorzeichen.

Mögliche Ergebnisse sind:

- Positives Vorzeichen: Der Punkt liegt links von der Strecke
- Null: Der Punkt ist kollinear zu der Strecke
- Negatives Vorzeichen: Der Punkt liegt rechts von der Geraden

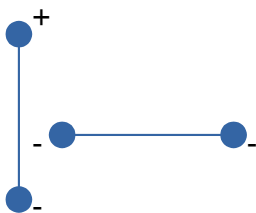
Die Idee ist, sich dieses Verhalten zu nutze zu machen, um dadurch Zeit zu sparen, da man auf diese Weise nicht das Gleichungssystem für die Geraden lösen muss.

## 3. Abstraktion des Problems

In diesem Abschnitt soll das Ausgangsproblem abstrakt dargestellt werden, um eine Lösung modellieren zu können.

### 3.1. Der Standard Fall

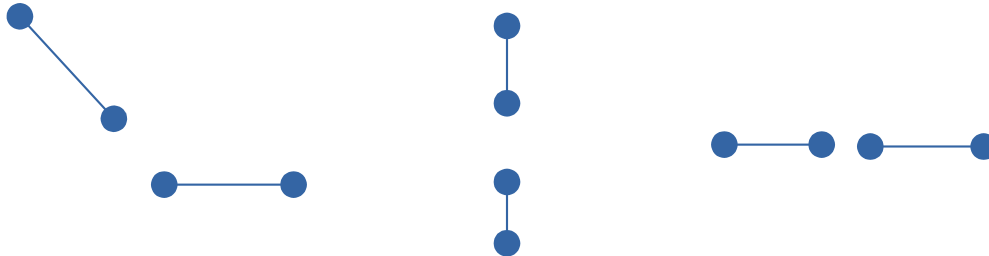
Existiert für keinen Punkt ein CCW von Null, so lässt sich das Schnittverhalten aus den Vorzeichen der Ergebnisse ableiten. Gilt für beide Strecken, dass deren Punkte im Vergleich mit der anderen Strecke unterschiedliche Vorzeichen besitzen, so ist ein Schnittpunkt gegeben. Da sich ein Beispiel konstruieren lässt, in dem die beiden Punkte der ersten Strecke ein unterschiedliches Vorzeichen im CCW besitzen, umgekehrt aber zwei gleiche Vorzeichen entstehen, kann man nicht von einer symmetrischen Relation ausgehen. Dies hat zu Folge, dass beide Seiten geprüft werden müssen:



(Abbildung 3.1.1. Asymmetrie des CCW)

### 3.2. Sonderfall: Es existiert ein CCW von Null

Ein CCW von Null bedeutet, dass der zugehörige Punkt kollinear zur anderen Strecke liegt. Ist dies der Fall, ist noch keine eindeutige Aussage möglich, ob die zwei Strecken einen Schnittpunkt besitzen.



( Abbildung 3.2.1. Kollinearität ohne Schnittpunkt)

Wie in Abbildung 3.2.1. zu sehen gibt es viele Möglichkeiten, wie ein CCW von Null entstehen kann, ohne dass ein Schnittpunkt gegeben ist. Darum muss man diejenigen Fälle, in denen ein Schnittpunkt möglich ist, einzeln abdecken. Diese sind:

#### Ein Punkt liegt auf der andern Strecke

Liegt ein Punkt auf der andern Strecke, so ist dieser automatisch der Schnittpunkt.



(Abbildung 3.2.2. Ein Punkt auf der anderen Strecke)

#### Überlappung

Überlappen sich die beiden Strecken, so existieren unendlich viele Schnittpunkte.



(Abbildung 3.2.3. Überlappung)

### 3.3. Sonderfall: Eine Strecke der Länge Null

Berechnet man das CCW gegen eine Strecke der Länge Null, so ist für alle nur denkbaren Punkte das Resultat ebenfalls Null, da alle Punkte kollinear zu dieser Strecke liegen. Um also einen Schnittpunkt zu erhalten, muss entweder die Strecke der Länge Null vollständig auf der anderen Strecke liegen (siehe Überlappung) oder ein Punkt der anderen Strecke ebenfalls identisch mit den beiden gleichen Punkten sein. Damit lässt sich die Abstraktion aus 3.2. auch auf diesen Sonderfall anwenden.

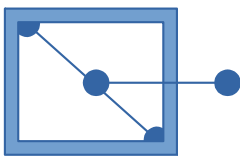
## 4. Lösungsansatz

In diesem Kapitel wird der implementierte Lösungsansatz näher beschrieben.

### 4.1. Abdeckung der Sonderfälle

Alle untersuchten Sonderfälle belaufen sich auf ein CCW von Null. Daraus ergibt sich, dass das Existieren eines CCWS von Null bedeutet, dass die Sonderfälle geprüft werden müssen. Generalisiert man diesen Ansatz, so erhält man folgende Aussage: Existiert ein CCW von Null, so muss mindestens ein Punkt auf der jeweils anderen Strecke liegen, sonst gibt es keinen Schnittpunkt.

Um jedoch die Gleichung der Strecke zu vermeiden, muss man nun das Problem „Liegt ein Punkt auf einer Strecke“ anhand der gegebenen Informationen modellieren. Gegeben ist demnach das CCW und die Koordinaten der Punkte. Ein CCW von Null sagt aus, ob der Punkt kollinear liegt. Befindet sich der Punkt zusätzlich innerhalb der Koordinaten (beide Dimensionen) der Punkte der anderen Strecke, so erhält man die Aussage: Der Punkt ist innerhalb der Koordinaten der Strecke kollinear zu der Strecke, liegt also auf der Geraden, die durch diese Strecke definiert wird. Trifft diese Aussage zu, so liegt der Punkt innerhalb der Grenzen der Strecke auf der Geraden und somit auf der Strecke selbst. Damit ist dieser Punkt ein Schnittpunkt.



(Abbildung 4.1.1. Kollinear und in den Grenzen)

Hieraus lässt sich die folgende Lösung für die Sonderfälle ableiten:

Wenn mindestens ein CCW von Null gefunden wurde, sind alle Punkte mit einem CCW von Null zu überprüfen, ob sie auf der anderen Strecke liegen. Ist dies für mindestens einen Punkt gegeben, so existiert ein Schnittpunkt, sonst ist kein Schnittpunkt mehr möglich.

### 4.2. Abdeckung des Standardfalls

Wenn kein CCW von Null gegeben ist, so lässt sich der Test wie in 3.1. beschrieben regulär durchführen und erbringt das erwartete Ergebnis.

### 4.3. Test der Implementierung

Um eine möglichst fehlerfreie Softwarelösung zu erreichen, wurde die Implementierung zusätzlich durch Unit-Tests mit dem Framework „google test“ überprüft, die Tests liefen fehlerfrei.

## 5. Ergebnis

Im Durchschnitt wurde durch die Implementierung folgendes Ergebnis erzielt:

```
{  
  "s_1000_1.dat": {"duration": "9ms", "count": 11},  
  "s_10000_1.dat": {"duration": "883ms", "count": 732},  
  "s_100000_1.dat": {"duration": "1min 39sec 483ms", "count": 77126}  
}
```