Algorithmische Geometrie

Entwurf und Analyse effizienter Algorithmen für geometrische Probleme

Warum sind geometrische Probleme bedeutsam? Reale Objekte und Problemstellungen können nicht immer auf die Maschinen-Logik herkömmlicher Architekturen abgebildet werden. Passende Algorithmen übernehmen die Umsetzung der räumlichen Parameter und machen das gestellte Problem maschinen-lösbar.

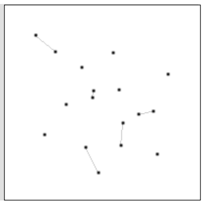
Motivation der Algorithmischen Geometrie

Die Algorithmische Geometrie ist aus Problemen der praktischen Anwendung in verschiedenen Gebieten entstanden. Wikipedia spezialisiert den Begriff der Algorithmischen Geometrie auf das Arbeiten mit geometrischen Strukturelementen wie Punkten, Linien, Kreisen, Polygonen und Körpern. Die folgenden Beispiele sollen eine Idee über die Aufgabenvielfalt der Algorithmischen Geometrie vermitteln.

Shortest Path: Mittels eines offensichtlichen Algorithmus lässt sich ein Paar mit kürzestem (euklidischem) Abstand finden – eine Lösung ist in $O(n^2)$ zu finden. Es stellt sich die Frage, ob die Laufzeit nicht verbessert werden kann.

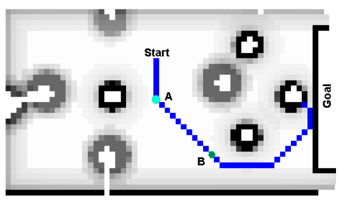
```
min := ∞;

for i:= 1 to n
     for j:= i+1 to m
            d := (x_i + x_j)^2 + (y_i + y_j)^2
            if min > d then min := d
```

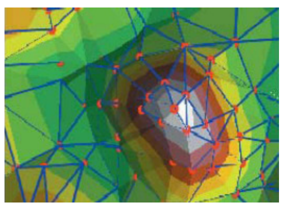


Projektion: In einer Szene überlagern/verdecken sich Objekte. Statt alle Objekte in voller Detailtreue zu berechnen, sollten nur jene Flächen und Kanten berechnet werden, die letztenendes sichtbar werden. Ziel eines geometrischen Algorithmus ist es, in der Projektion die überlappenden Flächen effizient zu bestimmen und anzupassen.

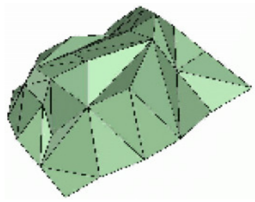
Bewegungsplanung: Ist es möglich von einem Startpunkt zum gewünschten Ziel zu gelangen? Im Besonderen ist diese Problematik für die Robotik relevant. Autonome Systeme sollen ohne Fremdeingriffe Hindernisse erkennen, Alternativrouten ermitteln und Plausibilitätprüfungen anstellen. Anleihen können hier über die Präsenz der Projektgruppe RoboCup (FU-Fighters.de) genommen werden.



Geografisches Informationssysteme (GIS): Anwendungen der Kartografie fassen mehrere Zielgebiete für Algorithmen zusammen – Verfahren zur Wegfindung unter Angabe von freien Parametern oder schnellem Skalieren auf umfangreichen topologischen Quellen. Eine Bereichsanfrage, die beispielsweise



die Anzahl der Orte zwischen zwei Längen- und Breitengraden bestimmt, sollte keinen größeren Aufwand als $O(\log n + k)$ betreiben, wobei n für die Anzahl der auszuwertenden Orte und k für die Ausgabe des Ergebnisses steht.



Computer Aided Geometric Design (CAD/CAGD): Bei der Entwicklung von Objekten im Computer stellt CAD die Planungskomponente zur computergestützten Fertigung (CAM) dar. Räumliche Körper und Flächen werden hierbei oft über Triangulierung (Dreiecksnetz) dargestellt.

Muster-/Formerkennung: Gilt es Objekten zu interpretieren, stellt sich die Frage wie nahe der Vergleich an seiner Referenz liegt und welche Abweichung noch tolerierbar ist. In Projekten wie Profi (Perceptually-relevant Retrieval Of Figurative Images) der Theoretischen Informatik an der FU Berlin oder den Biometriebemühungen des Bundes stellen sich diese Fragen. Anwendungsgebiete wie das Patentrecht sind bis heute auf die manuelle Interpretation angewiesen und werden daher in Ihrem Durchsatz durch den Faktor Mensch begrenzt. Im Beispiel einer Logogenemigung kann ein behördlicher Mitarbeiter bis zu 5000 + 3000 Logos pro Vor-/Nachmittag inspizieren – ist also in seiner Arbeitsleistung nicht weiter skalierbar.

Berechnungsmodell

Wie Anfangs erwähnt, gilt es komplexe Probleme in ein reales System zu übertragen. Da die meisten Probleme – mit Ausnahme derer die NP-Vollständigkeit erfordern – mit einer Registermaschine (RAM) lösbar sind, wird diese für alle weiteren Berechnungen vorausgesetzt. Von Vorteil ist, dass bei einer RAM mit ganzen Zahlen operiert wird und alle Grundrechenoperationen in konstanter Zeit durchgeführt werden. Das Modell ist somit unabhängig von der Länge der Zahlen (daher die Einschränkung bei NP).

Das Modell wird gelegentlich dahingehend verändert, dass
- eine „Real RAM“ verwendet wird
- beliebige reelle Zahlen in einer Speicherzelle Platz finden
- in einer Zeiteinheit algebraische Gleichungen von konstantem Grad lösbar sind (z.B. $\sqrt{\ }$)
- andere

Zu beachten ist, dass unter Anwendung entsprechender Funktionsbibliotheken Rundungsfehler auftreten. Algorithmen müssen daher entsprechend robust sein, um trotz dieser Ungenauigkeiten korrekte Antworten zu liefern. Ansätze sind zum einen konsequent exaktes Rechnen, was jedoch zu einer drastisch verringerten Laufleistung führt, oder nur hinreichend exakte Antworten zu liefern um z.B. Entscheidungsfragen auf boolscher Ebene zu beantworten.