Diplomprüfung Algorithmische Geometrie

Kurs 1840

Datum: 13.09.1999

Prüfer: Prof. Dr. Klein

Note: 1.3

Dauer: 30 min

1) Sweep im 1 -dimensionalen Raum (Währungsschwankungen) erklären
2) konvexe Hülle
   - Definition
   - mit Hilfe des Sweep erklären
   - Laufzeit?
3) Voronoi-Diagramm
   - zeichnen
   - welche Regionen sind unbeschränkt?
   - wie kann man das beweisen?
4) Sweep über Liniensegment im Raum
   - wie funktioniert es
   - was ist bei n Liniensegmenten im Raum?
   (Sind auch 7 Teilstücke)

Diplomprüfung Algorithmische Geometrie

Kurs 1840

Datum:16.12.98

Prüfer: Prof. Dr. Klein

Note:2.0

Dauer:25min

- Dichteste Punktpaar in der Ebene
- Naiver Algorithmus
- Aufwand
- Besser? Sweep!
- Algorithmus, ES , SSS, Aufwand
- Optimal?
- Konvexe Hülle
- Definition
- Lemma 4.1 Korollar 4.4
- Konstruktion? Inkrementell!
- Algorithmus
- Aufwand: Theorem 4.5
- Optimal?
- Der Durchschnitt von Halbebenen
- Dualität
- Abbildung 4.14
- Aufwand
- Voronoi-Diagramme
- Aus 5 Punkten eines malen
- Wie? Divide and Conquer!
- Algorithmus
- Aufwand
- Delaunay-Triangulation. Kap 5.4.1
- Lemma 5.18

Die Prüfung bei Prof. Klein war angenehm. Prof. Klein bohrt nicht, wenn man etwas nicht weiß wird es von Prof. Klein erklärt. Eine Prüfung in der man etwas lernt! Die Note ist wohlwollend.

Gedächtnisprotokoll der mündlichen Diplomprüfung 1840 - Algorithmische Geometrie - am

19. 8. 98 - Note: 1.3

- Wie berechnet man die maximale Teilsumme naiv?
  Jedes Objekt mit jedem vergleichen.
- Zeit?
  O(n^2).
- Wie geht es sonst noch?
  Sweep erklären.
- Zeit?
  O(n^2).
- Was ist ein Sweep allgemein? Welche Strukturen treten auf?
  Erklären mit toten, aktiven, schlafenden Objekten, ES und SSS.
- Durchschnitte von Liniensegmenten?
  Sweep erklären.
- Wie sieht die ES, wie die SSS aus?
  ES enthält die Schnitt, die Anfangs- und die Endpunkte, die SSS die Liniensegmente.
- Wie werden die Liniensegmente gespeichert.
  Es reicht die Angabe einer eindeutigen Identifizierung.
- Welche Zeit?
  O((n+k)logn).
- Warum?
  Weil die Schnittpunkte berichtet werden müssen.
- Was ist eine konvexe Hülle?
  Definition.
- Welches Verfahren?
  Linke, rechte Kontur, dann Sweep über einzelne Stücke.
- Welches noch? Darin wird O(nlogn) im Mittel erreicht.
  Inkrementelles Verfahren erklären.
- Welche Teilprobleme treten dabei auf?
  Punkt innerhalb oder außerhalb des bisherigen Polygons. Erlären mit Sektoren. Tangenten bilden.
- Welche Datenstruktur?
  Baum => Binäres Suchen => Zeit O(logn), auch für die Aktualisierung der Eckenverzeichnis
- Wie werden die Tangenten gebildet?
  Von Sektorenkante aus auf Rand entlang jede Kante untersuchen.
- Zeit?
  O(nlogn).
- Auf dem untersuchten Rand können theoretisch fast alle Punkte liegen. Wieso trotzdem 0 (nlogn)?
  Weil in diesem Fall diese Punkte gelöscht werden. D. h., über alle Punkte betrachtet ergibt sich O(nlogn).
- Was ist ein Voronoi-Diagramm?
  Erklärung.
- Zeichnen des Diagramms für vier vorgegebene Punkte.
- Welche Punkte haben unbeschränkte Regionen?
  Konvexe Hülle.
- Wieso?
  Kreis um zwei Punkte mit unbeschränkter Voronoi-Region und Mittelpunkt auf Bisektor. Kein anderer Punkt darin. Mittelpunkt -> oo - Kreis wird immer größer. Nie ein anderer Punkt darin, d. h. auf dieser Seite.
- Wie Voronoi-Diagramm erstellen?

Inkrementell, sweep, divide and conquer.

- Aussuchen.
  Sweep.
- Abgelehnt, weil sweep schon drangewesen.
  Divide and conquer. Divide-Schritt erklärt. Zeit: O(logn).
- Was ist B(L;R)? Wie findet man es?
  Im übereinandergelegten Voronoi-Diagramm suchen. Zeit: O(n).
- Wie restliche Verbindung?
  Erklären beim Zeichnen.
- Zeit?
  O(n).
- Insgesamte Zeit?
  O(nlogn).

Die Antworten sind die, die ich hätte geben müssen, und nicht unbedingt die, die ich auch gegeben habe. Professor Klein ist ein sehr verständnisvoller Prüfer, der sich alle Mühe gibt, die Prüfung streßfrei ablaufen zu lassen. Er ist keineswegs daran interessiert, jemanden fertigzumachen.

Prüfungsprotokolle Algorithmische Geometrie
Prüfer : Prof. Klein
Datum : 11.11.97

Während der Kursbetreuung zur Algorithmischen Geometrie werden in den NEWS Orientierungshilfen zum Lesen des Kurstextes und eine Sammlung möglicher Prüfungsfragen herausgegeben.
Diese sind zur Vorbereitung auf die Prüfung zu empfehlen. Auf alle Fälle sollten praktische Übungen zu den Algorithmen vorgenommen werden, da Prof. Klein diese an Beispielen vorgeführt haben möchte. Hierfür bieten sich auch die Einsendearbeiten zum Kurs an. Wird der Kurs - wie im Sommer 97 - mit einer mündlichen Scheinprüfung abgeschlossen, so ist diese als Probelauf zu empfehlen.
Prof. Klein fragt nach Zusammenhängen zwischen den Kursinhalten. Für die Prüfung darf man eine der 7 KE herausnehmen. Es bietet sich die 3. KE an, da dies am wenigsten die Zusammenhange zwischen den anderen KE stört.

Prüfung 1

Sweep - Paradigma
Kürzester Abstand zwischen Punkten in der Ebene per Sweep - Verfahren
Laufzeitverhalten und Details zu SSS und ES
Untere Schranke für Laufzeit (ε - closeness mit Beweis über lineares Modell)

Voronoi - Diagramme
Laufzeitverhalten
naiv (über Halbebenenschnitte)
für vorgestellte Algorithmen
Zeichnen eines Voronoi - Diagramms für 5 Punkte
Konstruktion des Voronoi - Diagramms per Sweep - Verfahren
mit Details zu Ereignissen und der Wellenfront

Bewegunasplanung in unbekannter Umgebung
Roboter mit Sichtsystem und unbekanntes Polygon
Baum der kürzesten Wege aufbauen und
Zusammenhang mit dem kompetitives Verfahren von m Halbgeraden erläutern

Prüfung 2

Sweev - Paradigma
Bestimmung der max. Teilsumme per Sweep - Verfahren
Einsatz beim Algorithmus zur Bestimmung des Kerns

konvexe Hülle
Def der konvexen Hülle
Verfahren zur Konstruktion der konv. Hülle
Laufzeitverhalten
Untere Schranke für Laufzeit beweisen (Zurückführen auf das Sortieren)

Bewegungsplanung in unbekannter Umgebung
Pledge - Algorithmus erklären und am Beispiel durchspielen

Gedächtnisprotokoll zur Diplomprüfung
Vertiefende Kurse der Informatik

Kurs: Algorithmische Geometrie (1840)
Datum: 6.10.97
Prüfer: Prof. Dr. R. Klein
Note: 1,3

- Wie bestimmt man naiv das kleinste Punktepaar einer Menge von Punkten?
  - Durch paarweises Vergleichen.
- Laufzeit?
  - O(n^2)
- Und mit Hilfe eines besseren Algorithmus?
  - Sweep-Verfahren erklären.
- Laufzeit?
  - O(n log n)
- Voronoi-Diagramm eignet sich auch dafür. Aufmalen eines Voronoi-Diagramms von 4 Punkten.
- Wie kann man das kleinste Punktepaar mit dem Voronoi-Diagramm finden?
  - Nächste Nachbarn eines Punktes haben benachbarte Voronoi-Regionen. Suchen des Minimums unter allen Nachbarn.
  - Laufzeit bei existierendem Voronoi-Diagramm: O(n), denn es gibt nur O(n) Voronoi-Kanten.
- Warum gibt es nur O(n) Voronoi-Kanten?
- Was kann man noch mit einem Voronoi-Diagramm machen?
  - Konvexe Hülle einer Punktmenge bestimmen.
- Beschreibung eines Verfahrens zur Konstruktion des Voronoi-Diagramms (man darf sich eins aussuchen).
- Zeitbedarf?
  - O(n log n).
- Geht es noch schneller?
  - Nein.
- Warum nicht?
  - Weil sich aus Voronoi-Diagramm in linearer Zeit die konvexe Hülle herleiten lässt, und die durch Bestimmung der konvexen Hülle in linearer Zeit eine Menge von Punkten sortiert werden kann, und das Sortieren \Omega(n log n) dauert. (Dies ist die "lange" Erklärung, es geht auch kürzer)
- Bestimmung des Kerns eines Polygons. Zuerst zeichnerisch. Danach Algorithmus erklären: Durchschnitt der inneren Halbebenen. Dies hat Laufzeit O(n log n). Mit Bestimmung der wesentlichen Kanten (B- und F-Menge, siehe Kapitel 4.4) geht dies in linearer Laufzeit.

Wichtig: Man kann sich eine Kurseinheit aussuchen, über die man nicht geprüft wird.

Die Prüfung verlief sehr nett und ruhig. Wenn man etwas nicht weiß, wird man nicht lange "gequält", sondern bekommt zuerst ein oder zwei Hilfestellungen, und kurz danach beantwortet Prof. Klein selbst die Frage.

Mögliche Prüfungsfragen zum Kurs 1840: Algorithmische Geometrie

Veröffentlicht von Elmar Langetepe in der Newsgroup feu.informatik.kurs.1840 im Sommersemester 1997.

Zu Kurseinheit 1:

- Wie lautet die Eulersche Formel für kreuzungsfreie geometrische Graphen in der Ebene? Beweis?
- Was bedeutet es, daß die Laufzeit eines Algorithmus in O(f(n)) bzw. \Omega(f(n)) liegt?
- Was versteht man unter einer Zusammenhangskomponente einer Menge?
- Warum benötigt der Elementtest bei einer Menge mit m Zusammenhangskomponenten im linearen Modell mindestens \Omega(log m) Zeit?
- Wie kann man aus der vorigen Frage schließen, daß die Bestimmung des kleinsten Abstands zwischen n Punkten in der Ebene mindestens \Omega(n log n) Zeit braucht?

Zu Kurseinheit 2:

- Wie würden Sie durch Sweep das zweitgrößte Element in einer Folge von n Zahlen bestimmen? Das drittgrößte? Das k-t-größte? Welche SSS würden Sie dabei benutzen, und wie hinge die Laufzeit dann von n und k ab?
- Erläutern Sie das Sweep-Verfahren für die Bestimmung des dichtesten Paars von n Punkten in der Ebene. Warum
  - genügt es,
  - ist es für die Laufzeit wichtig,

  daß man sich während des sweeps nur die Punkte im vertikalen Streifen links von der sweep line merkt?
- Wie würden Sie die Schnittpunkte von n nicht-senkrechten Geraden berechnen?
- Aus wievielen Segmenten kann die untere Kontur von n Parabeln bestehen, deren Mittelachsen senkrecht sind und die sich nach oben öffnen?
  Wie würden Sie diese untere Kontur effizient berechnen?
- Zu jeder endlichen Punktmenge existiert ein einfaches Polygon, das jeden Punkt aus der Menge als Eckpunkt besitzt. Wie kann man durch einen Sweep ein solches Polygon berechnen, wie hoch ist der Aufwand?

Zu Kurseinheit 3:

- Was versteht man unter einer zerlegbaren Anfrage?
- Wie läßt sich bei einer rein statischen Struktur für zerlegbare Anfragen die Operation Einfügen so implementieren, daß die amortisierten Kosten gering sind?
- Gegeben ist ein Datentyp zur Speicherung von Objekten einer geordneten Grundmenge mit den Operationen Aufbau, Löschen, Entfernen, Bericht des Minimums.
  - Welche Datenstruktur würden Sie benutzen, um das Entfernen schwach zu implementieren?
    (Hinweis: Kann man die Vorzüge von sortierten Arrays und von Listen miteinander verbinden?)
  - Welche Laufzeiten ergeben sich, wenn man das Entfernen dynamisiert?
- Konstruieren Sie einen 2-d-Baum aus einer gegebenen Punktmenge, und

beschreiben Sie die Beantwortung von Bereichsanfragen.

Zu Kurseinheit 4:

- Beschreiben Sie einen O(n log n) Algorithmus zur Konstruktion der konvexen Hülle von n Punkten in der Ebene, und begründen Sie sein Laufzeitverhalten.
- Beschreiben Sie Graham's scan zur Berechnung der konvexen Hülle von n Punkten in der Ebene und schätzen Sie die Laufzeit ab.
- Ein Kreis enthalte eine Ellipse. Wer von beiden hat den längeren Umfang?
- Kann es in einem einfachen Polygon zwei Punkte p,q geben, die dasselbe Sichtbarkeitspolygon haben? Wenn ja, können p und q sich sehen?
- Bestimmen Sie den Kern eines Beispielpolygons nach dem im Kurs angegebenen Verfahren. Wieviel Zeit benötigen Sie?
- Wenn Sie ein Polygon durchlaufen und den gesamten Rand gesehen haben, haben Sie dann bereits das gesamte Polygon gesehen? Beweisen Sie Ihre Aussage.

Zu Kurseinheit 5:

- Zeichnen Sie das Voronoi-Diagramm zu vier oder fünf vorgegebenen Punkten (euklidisch).
- Für welche Punktmengen hat das Euklidische Voronoi-Diagramm die Struktur eines Baumes?
- Warum ist der nächste Nachbar eines Punktes p in S stets ein Voronoi-Nachbar in V(S)?
- Kann es fünf Punkte in der Ebene geben, so daß jede Voronoi- Region zu jeder anderen benachbart ist? (Hinweis: Denken Sie an den Graphen K_5 aus der Aufgabe 1.3)
- Wie sehen die Einheitskreise der L_1 und L_\infty ("L-unendlich") aus? Angenommen, Sie haben einen Algorithmus zur Berechnung von L_1-Diagrammen. Können Sie ihn auch bei der Berechnung von L_\infty-Diagrammen benutzen?
- Wie kann man die Bisektoren von zwei Punkten bezüglich einer konvexen Distanzfunktion bestimmen?

Zu Kurseinheit 6:

- Beschreiben Sie eines der Verfahren
    - Inkrementelle Konstruktion
    - Sweep
    - Divide and Conquer

  zur Berechnung des Voronoi-Digrammes und skizzieren Sie die jeweilige Laufzeitabschätzung.
- Was versteht man im Sweep-Verfahren unter der Wellenfront? Wie ist ihre Komplexität und ihre Struktur?
- Wie findet man beim Verfahren Divide and Conquer die Endstücke von B(L,R)?
- Was ist ein Konfliktdreieck bezüglich eines Knotens p_i aus der Delaunay-Triangulierung DT_(i-1)? Wie groß ist die Anzahl der Konfliktdreiecke eines Knotens in bezug auf den Grad des Knotens in DT_i?

Zu Kurseinheit 7:

- Wie findet der Pledge-Algorithmus aus den Windungen eines spiralförmigen Hindernisses heraus? (Am Beispiel demonstrieren)
- Ist es möglich, daß der Pledge-Algorithmus eine Kante beliebig oft besucht?

Beweisen Sie Ihre Aussage.

- Angenommen, Sie wissen, daß die Tür in der Wand genau den Abstand k von Ihrer Startposition hat, aber nicht, auf welcher Seite sie liegt. Welchen kompetitiven Faktor können Sie erreichen?
- Beschreiben Sie allgemein das Verfahren zur Suche nach einem Zielpunkt in einem einfachen Polygon (mit Sichtsystem).
- Sei P ein einfaches Polygon und p ein Punkt in P. Wann gilt: p liegt auf dem Rand des Kerns des Sichtbarkeitspolygons von p?
- Skizzieren Sie kurz die Idee der Suchstrategie der m Halbgeraden. Welche Eigenschaft der geometrischen Reihe macht man sich generell zunutze?