On finding the guard that sees most and the shop that sells most

Ignaz Rutter

Universität Karlsruhe

17. Juni 2005

Überblick

Es geht um zwei Optimierungsprobleme. Ich werde:

- Beide Probleme kurz vorstellen
- Die Probleme einzeln näher betrachten
- Algorithmen zur Lösung entwickeln
- Die gefunden Algorithmen bewerten
- Einen Ausblick auf mögliche Verbesserungen geben

Zwei Probleme

Probleme haben ähnliche Struktur:

- Einem Punkt x wird eine Fläche $V(x)$ zugeordnet.
- Es wird ein Punkt x_{opt} gesucht, der den Flächeninhalt von $V(x_{\text{opt}})$ maximiert.

Zwei Probleme

Probleme haben ähnliche Struktur:

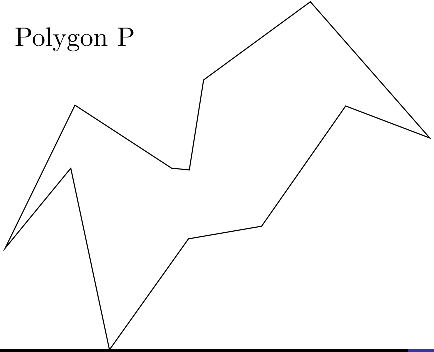
- Einem Punkt x wird eine Fläche $V(x)$ zugeordnet.
- Es wird ein Punkt x_{opt} gesucht, der den Flächeninhalt von $V(x_{\mathrm{opt}})$ maximiert.

Für beide Probleme habe ich keinen exakten Algorithmus finden können.

Finding the guard that sees most

Gegeben: einfaches Polygon P.
Gesucht: Punkt im Inneren von P, der am meisten sieht.

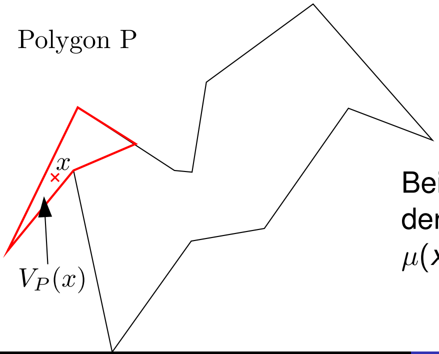
$$\mu(P) = 1$$



Finding the guard that sees most

Gegeben: einfaches Polygon P.
Gesucht: Punkt im Inneren von P, der am meisten sieht.

$$\mu(P) = 1$$

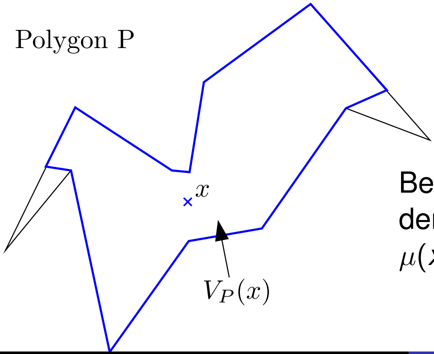


Beispiel für einen Punkt, der wenig sieht:
$\mu(x) = \mu(V_P(x))$ klein

Finding the guard that sees most

Gegeben: einfaches Polygon P.
Gesucht: Punkt im Inneren von P, der am meisten sieht.

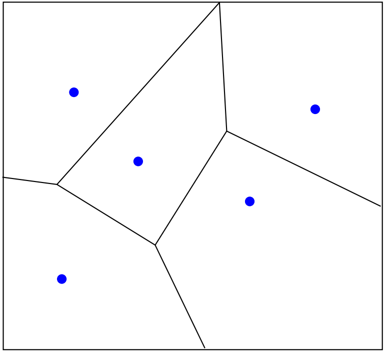
$$\mu(P) = 1$$



Beispiel für einen Punkt, der viel sieht:
$\mu(x) = \mu(V_P(x))$ groß

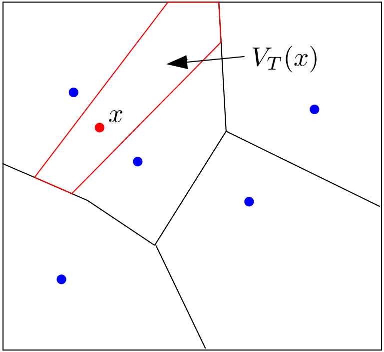
Finding the shop that sells most

Gegeben: Voronoi-Diagramm einer Punktmenge $T \subseteq [0,1]^2$
Gesucht: Punkt, der nach dem Einfügen ins Voronoi-Diagramm ein möglichst großes Voronoi-Gebiet hat



Finding the shop that sells most

Gegeben: Voronoi-Diagramm einer Punktmenge $T \subseteq [0,1]^2$
Gesucht: Punkt, der nach dem Einfügen ins Voronoi-Diagramm ein möglichst großes Voronoi-Gebiet hat

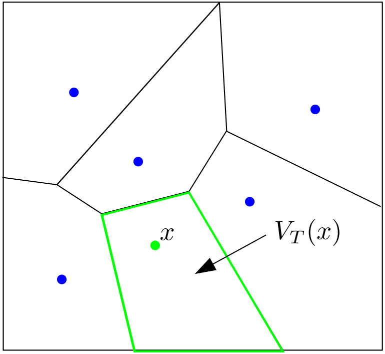


Punkt x mit kleinem Voronoi-Gebiet:
$\mu(x) = \mu(V_T(x))$ klein.

Finding the shop that sells most

Gegeben: Voronoi-Diagramm einer Punktmenge $T \subseteq [0,1]^2$

Gesucht: Punkt, der nach dem Einfügen ins Voronoi-Diagramm ein möglichst großes Voronoi-Gebiet hat



Punkt x mit großem Voronoi-Gebiet:

$\mu(x) = \mu(V_T(x))$ groß.

klassische Optimierung

Vorgehensweise:

klassische Optimierung

Vorgehensweise:

- Zielfunktion aufstellen
- Nebenbedingungen formulieren
- Optimierungsverfahren anwenden (Lagrange, Gradientenverfahren, cg-Verfahren)

klassische Optimierung

Vorgehensweise:

- Zielfunktion aufstellen
- Nebenbedingungen formulieren
- Optimierungsverfahren anwenden (Lagrange, Gradientenverfahren, cg-Verfahren)

Das funktioniert nicht!
Die Zielfunktion ist nicht geschlossen darstellbar.

Approximation

Idee: Diskretisiere das Problem!

Approximation

Idee: Diskretisiere das Problem!

- Taste Gebiet randomisiert ab (Sampling)
- finde Punkt, der bezüglich abgetasteter Punkte optimal

Approximation

Idee: Diskretisiere das Problem!

- Taste Gebiet randomisiert ab (Sampling)
- finde Punkt, der bezüglich abgetasteter Punkte optimal

Fragen:

- Funktioniert das wirklich?

Approximation

Idee: Diskretisiere das Problem!

- Taste Gebiet randomisiert ab (Sampling)
- finde Punkt, der bezüglich abgetasteter Punkte optimal

Fragen:

- Funktioniert das wirklich?
- Wie viele Punkte muß man samplen?

Approximation

Idee: Diskretisiere das Problem!

- Taste Gebiet randomisiert ab (Sampling)
- finde Punkt, der bezüglich abgetasteter Punkte optimal

Fragen:

- Funktioniert das wirklich?
- Wie viele Punkte muß man samplen?
- Wie groß ist der Aufwand für die Lösung der beiden Probleme?
- Genaues Vorgehen?

Finding the guard that sees most

Schreibweisen

Voraussetzungen:

1. *P* einfaches Polygon
2. *n* Anzahl der Eckpunkte von P
3. Vereinfachende Annahme: $\mu(P) = 1$

Schreibweisen

Voraussetzungen:

1. P einfaches Polygon
2. n Anzahl der Eckpunkte von P
3. Vereinfachende Annahme: $\mu(P) = 1$

Für $x \in P$ sei $V_P(x)$ das Sichtbarkeitspolygon von x.

$\mu(x)$ ist Flächeninhalt von $V_P(x)$

Schreibweisen

Voraussetzungen:

1. *P* einfaches Polygon
2. *n* Anzahl der Eckpunkte von P
3. Vereinfachende Annahme: $\mu(P) = 1$

Für $x \in P$ sei $V_P(x)$ das Sichtbarkeitspolygon von x.

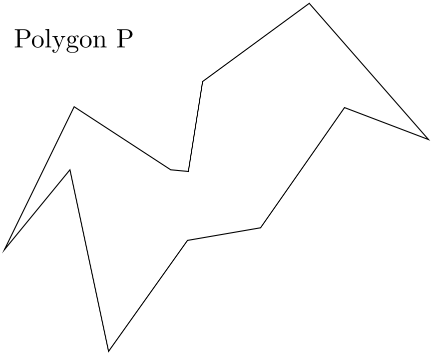
$\mu(x)$ ist Flächeninhalt von $V_P(x)$

$\mu_{\text{opt}} := \max_{x \in P} \mu(x)$

Eine Schätzfunktion für $\mu(x)$

$S \subseteq P$ endlich, $x \in P$

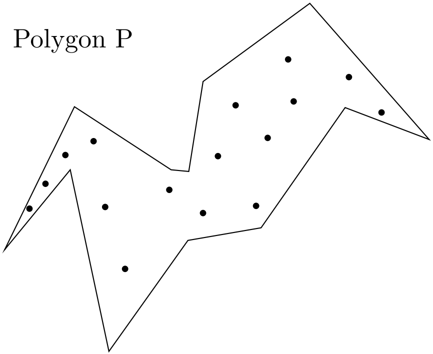
$e_S(x) = \frac{|V_P(x) \cap S|}{|S|}$ schätzt Fläche die von x aus sichtbar ist.



Eine Schätzfunktion für $\mu(x)$

$S \subseteq P$ endlich, $x \in P$

$e_S(x) = \frac{|V_P(x) \cap S|}{|S|}$ schätzt Fläche die von x aus sichtbar ist.

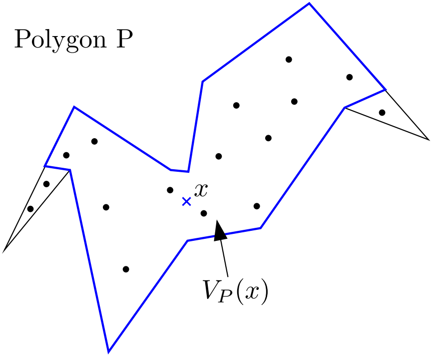


$|S| = 16$

Eine Schätzfunktion für $\mu(x)$

$S \subseteq P$ endlich, $x \in P$

$e_S(x) = \frac{|V_P(x) \cap S|}{|S|}$ schätzt Fläche die von x aus sichtbar ist.



$|S| = 16$

$|V_P(x) \cap S| = 13$

$\Rightarrow e_S(x) = \frac{13}{16}$

Bereichsraum und ϵ-Approximation

Definition

Sei $\mathcal{V} := \{V_P(x) \mid x \in P\}$,
dann heißt $(P, \mathcal{V})$ *Bereichsraum*.

Bereichsraum und ϵ-Approximation

Definition

Sei $\mathcal{V} := \{V_P(x) \mid x \in P\}$,
dann heißt $(P, \mathcal{V})$ *Bereichsraum*.

Definition

Eine Samplemenge S ist eine ϵ-*Approximation* für diesen Bereichsraum, wenn für jedes $x \in P$ gilt:

$$|e_S(x) - \mu(x)| \leq \epsilon$$

ϵ-Approximationssatz

Satz (ϵ-Approximationssatz)

Eine zufälliges gleichverteiltes Sample S aus $O((d/\delta^2)\log(d/\epsilon\delta))$ Punkten aus einem Bereichsraum mit VC-Dimension d ist mit Wahrscheinlichkeit $\geq 1-\delta$ eine ϵ-Approximation für diesen Bereichsraum.

Beweis.

[N. Alon, J.H. Spencer, The probabilistic method, 2000] $\square$

Die VC-Dimension

Bereichsraum: (X, R) mit X Menge, $\emptyset \neq R \subseteq 2^X$.

Die VC-Dimension

Bereichsraum: (X, R) mit X Menge, $\emptyset \neq R \subseteq 2^X$.

$A \subseteq X$, $\Pi_R(A) := \{A \cap r \mid r \in R\}$

Die VC-Dimension

Bereichsraum: (X, R) mit X Menge, $\emptyset \neq R \subseteq 2^X$.

$A \subseteq X$, $\Pi_R(A) := \{A \cap r \mid r \in R\}$

R zerlegt A, wenn $\Pi_R(A) = 2^A$.

Die VC-Dimension

Bereichsraum: (X, R) mit X Menge, $\emptyset \neq R \subseteq 2^X$.

$A \subseteq X$, $\Pi_R(A) := \{A \cap r \mid r \in R\}$

R zerlegt A, wenn $\Pi_R(A) = 2^A$.

Die VC-Dimension

Bereichsraum: (X, R) mit X Menge, $\emptyset \neq R \subseteq 2^X$.

$A \subseteq X$, $\Pi_R(A) := \{A \cap r \mid r \in R\}$

R zerlegt A, wenn $\Pi_R(A) = 2^A$.

$VCDim(X, R)$ ist der Betrag einer größten endlichen Menge A, die von R zerlegt wird.

Die VC-Dimension

Bereichsraum: (X, R) mit X Menge, $\emptyset \neq R \subseteq 2^X$.

$A \subseteq X$, $\Pi_R(A) := \{A \cap r \mid r \in R\}$

R zerlegt A, wenn $\Pi_R(A) = 2^A$.

VCDim(X, R) ist der Betrag einer größten endlichen Menge A, die von R zerlegt wird.

Beispiele:

- $\mathit{VCDim}(\mathbb{R}, \{[a, \infty) \mid a \in \mathbb{R}\}) = 1$

Die VC-Dimension

Bereichsraum: (X, R) mit X Menge, $\emptyset \neq R \subseteq 2^X$.

$A \subseteq X$, $\Pi_R(A) := \{A \cap r \mid r \in R\}$

R zerlegt A, wenn $\Pi_R(A) = 2^A$.

VCDim(X, R) ist der Betrag einer größten endlichen Menge A, die von R zerlegt wird.

Beispiele:

- $VCDim(\mathbb{R}, \{[a, \infty) \mid a \in \mathbb{R}\}) = 1$
- $VCDim(\mathbb{R}, \{H \mid H \text{ abgeschlossener Halbraum im } \mathbb{R}^2\}) = 3$

Abschätzung der VC-Dimension

Satz

Die VC-Dimension des Bereichsraums $(P, \mathcal{V})$ ist durch 23 beschränkt.

Beweis.

[Pavel Valtr, Guarding galleries where no point sees a small area, 1998] □

Ein Lemma

Lemma

Ist S ϵ-Approximation für $\epsilon = \delta/2n$ und $x_{\mathrm{app}} \in P$ ein Punkt, der $e_S(x_{\mathrm{app}})$ maximiert, dann gilt:

$$\mu(x_{\mathrm{app}}) \geq (1-\delta)\mu_{\mathrm{opt}}$$

Ein Lemma

Lemma

Ist S ϵ-Approximation für $\epsilon = \delta/2n$ und $x_{\text{app}} \in P$ ein Punkt, der $e_S(x_{\text{app}})$ maximiert, dann gilt:

$$\mu(x_{\text{app}}) \geq (1-\delta)\mu_{\text{opt}}$$

$\Rightarrow$ genügt Punkt zu finden, der $e_S(x_{\text{app}})$ maximiert.
Dieser Punkt ist dann $(1-\delta)$-Approximation des Punktes in P, der am meisten sieht.

1. Idee

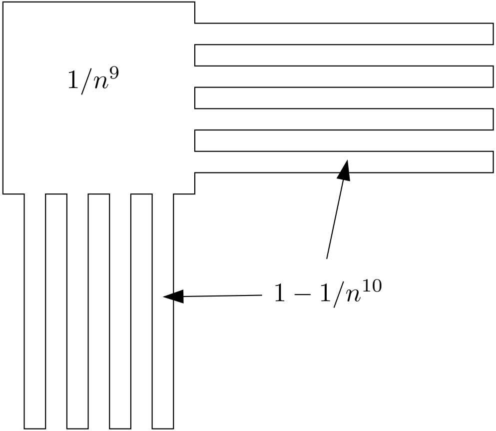
- müssen Punkt x_{app} finden, der $e_S(x_{\text{app}})$ maximiert.

1. Idee

- müssen Punkt x_{app} finden, der $e_S(x_{\mathrm{app}})$ maximiert.
- Wähle einfach den besten Punkt aus dem Sample S!

1. Idee

- müssen Punkt x_{app} finden, der $e_S(x_{\mathrm{app}})$ maximiert.
- Wähle einfach den besten Punkt aus dem Sample S!
- Das funktioniert nicht, Gegenbeispiel:



Abschätzungen für den Beweis

Es gelten folgende Eigenschaften:

1. $\mu_{\text{opt}} \geq 1/(n-2)$
2. $|\mu(x) - e_S(x)| \leq \epsilon (= \frac{\delta}{2n}) \ \forall x \in P$

Abschätzungen für den Beweis

Es gelten folgende Eigenschaften:

1. $\mu_{\text{opt}} \geq 1/(n-2)$
2. $|\mu(x) - e_S(x)| \leq \epsilon(= \frac{\delta}{2n})\ \forall x \in P$

Beweis.
Triangulation,
größtes Dreieck ist untere Schranke für größten Sichtbereich! □

Beweis des Lemmas

Beweis.

Sei $x_{\text{opt}} \in P$ ein Punkt mit $\mu(x_{\text{opt}}) = \mu_{\text{opt}}$.

$$\begin{aligned}\mu(x_{\text{app}}) &\geq e_S(x_{\text{app}}) - \frac{\delta}{2n} \\ &\geq e_S(x_{\text{opt}}) - \frac{\delta}{2n} \\ &\geq \mu(x_{\text{opt}}) - \frac{\delta}{n} \geq (1-\delta)\mu_{\text{opt}}.\end{aligned}$$

$\square$

2. Idee

- Suche x_{app} in ganz P, nicht nur in S!

2. Idee

- Suche x_{app} in ganz P, nicht nur in S!
- Idee: $s \in V_P(x) \iff x \in V_P(s)$

2. Idee

- Suche x_{app} in ganz P, nicht nur in S!
- Idee: $s \in V_P(x) \iff x \in V_P(s)$
- Setze $\mathcal{W}_S = \{V_P(s) \mid s \in S\}$.
  Suche Punkt in P, der in der größten Anzahl von Polygonen in $\mathcal{W}_S$ enthalten ist.

2. Idee

- Suche x_{app} in ganz P, nicht nur in S!
- Idee: $s \in V_P(x) \iff x \in V_P(s)$
- Setze $\mathcal{W}_S = \{V_P(s) \mid s \in S\}$.
  Suche Punkt in P, der in der größten Anzahl von Polygonen in $\mathcal{W}_S$ enthalten ist.

$|S| = M$, dann: Komplexität des Arrangements $\mathcal{A}(\mathcal{W}_S)$ in $O(nM^2)$.

Denn:
Strecke in P kann Rand eines Sichtbarkeitspolygons in maximal zwei Punkten schneiden
Sichtbarkeitspolygon hat höchstens n Ecken

Berechnung von $\mathcal{A}(\mathcal{W}_S)$

$\epsilon = \delta/2n$, $|S| = M$

- Berechne für jedes $s \in S$ $V_P(s)$ mit Sweeping.
- $M = O(1/\epsilon^2 \log(1/\epsilon)) = O((n^2/\delta^2)\log(n/\delta))$
- Komplexität von $\mathcal{A}(\mathcal{W}_S)$ ist $O(nM^2) = O((n^5/\delta^4)\log^2(n/\delta))$
- Läßt sich in $O((n^5/\delta^4)\log^3(n/\delta))$ berechnen

Resultat

Lemma

Für ein einfaches Polygon P und einen Parameter $\delta > 0$ kann in $O((n^5/\delta^4)\log^3(n/\delta))$ Zeit ein Punkt $x \in P$ bestimmt werden, sodaß $\mu(x) \geq (1-\delta)\mu_{\text{opt}}$. Dieser Algorithmus funktioniert mit Wahrscheinlichkeit $\geq 1-\delta$.

Beweis

Beweis.

$\epsilon = \delta/2n$, S zufälliges gleichverteiles Sample der Größe

$$M = O(1/\epsilon^2 \log(1/\epsilon)) = O((n^2/\delta^2)\log(n/\delta))$$

aus P, ist ϵ-Approximation mit hoher Wahrscheinlichkeit.
Berechne das Arrangement $\mathcal{A}(\mathcal{W}_S)$ in $O((n^5/\delta^4)\log^3(n/\delta))$ Zeit.

Für jede Fläche des Arrangements berechne Anzahl der Polygone in $\mathcal{W}_S$ die sie enthalten.
Wähle beliebigen Punkt aus der Fläche mit der größten Anzahl.

Finding the shop that sells most

Einige Definitionen

T Menge von Punkten in $[0,1]^2$, $|T| = n$.
Für einen Punkt x bezeichne $V_T(x)$ die Voronoi-Region von x im Voronoi-Diagramm von $T \cup x$.

Wir suchen einen Punkt x_{opt} der $\mu_{\text{opt}} = \mu(x_{\text{opt}})$ maximiert.

Lege Torus-Topologie zugrunde.

Ziel:
Wir wollen einen Punkt x_{app} finden mit $\mu(x_{\text{app}}) \geq (1-\delta)\mu_{\text{opt}}$, wobei $\delta > 0$ ein Parameter ist.

Was haben wir vorhin getan?

Was haben wir vorhin getan?

Wir haben

- eine Schätzfunktion definiert, die recht genau war

Was haben wir vorhin getan?

Wir haben

- eine Schätzfunktion definiert, die recht genau war
- eine untere Schranke für $\mu(x)$ gesucht

Was haben wir vorhin getan?

Wir haben

- eine Schätzfunktion definiert, die recht genau war
- eine untere Schranke für $\mu(x)$ gesucht
- gezeigt, daß Optimum der Schätzfunktion $(1-\delta)$-Approximation des optimalen Punktes ist

Was haben wir vorhin getan?

Wir haben

- eine Schätzfunktion definiert, die recht genau war
- eine untere Schranke für $\mu(x)$ gesucht
- gezeigt, daß Optimum der Schätzfunktion $(1-\delta)$-Approximation des optimalen Punktes ist
- uns überlegt, wie wir so ein Optimum finden

Was haben wir vorhin getan?

Wir haben

- eine Schätzfunktion definiert, die recht genau war
- eine untere Schranke für $\mu(x)$ gesucht
- gezeigt, daß Optimum der Schätzfunktion $(1 - \delta)$-Approximation des optimalen Punktes ist
- uns überlegt, wie wir so ein Optimum finden

Hier geht's genauso!

Reichweite einer Voronoi-Region

Definition

Die *Reichweite* einer Voronoi-Region $V_T(x)$ ist der Abstand zwischen x und dem von x am weitesten entfernten Punkt in $V_T(x)$.
Das ist der Radius des kleinsten Kreises um x, der $V_T(x)$ enthält.

Reichweite einer Voronoi-Region

Definition

Die *Reichweite* einer Voronoi-Region $V_T(x)$ ist der Abstand zwischen x und dem von x am weitesten entfernten Punkt in $V_T(x)$.
Das ist der Radius des kleinsten Kreises um x, der $V_T(x)$ enthält.

Lemma

Sei ℓ die größte Reichweite aller Voronoi-Regionen $V_T(t)$ mit $t \in T$. Dann gilt:

$$\frac{1}{4}\pi\ell^2 \leq \mu_{\text{opt}} \leq \pi\ell^2$$

Vorgehen

1. Zerlege das Einheitsquadrat in ein Gitter von Quadraten mit Seitenlänge ℓ.
2. Verwende in jeder Gitterzelle $\mathcal{Q}$ eine ϵ-Approximation.

Vorgehen

1. Zerlege das Einheitsquadrat in ein Gitter von Quadraten mit Seitenlänge ℓ.
2. Verwende in jeder Gitterzelle $\mathcal{Q}$ eine ϵ-Approximation.
3. Definiere dadurch Schätzfunktion e_S, sodaß für jedes $x \in \mathcal{Q}$ gilt:

$$|e_S(x) - \mu(x)| \leq \frac{\delta \pi \ell^2}{8}$$

Vorgehen

1. Zerlege das Einheitsquadrat in ein Gitter von Quadraten mit Seitenlänge ℓ.
2. Verwende in jeder Gitterzelle $\mathcal{Q}$ eine ϵ-Approximation.
3. Definiere dadurch Schätzfunktion e_S, sodaß für jedes $x \in \mathcal{Q}$ gilt:

$$|e_S(x) - \mu(x)| \leq \frac{\delta\pi\ell^2}{8}$$

4. Wähle Punkt $x_\mathcal{Q} \in \mathcal{Q}$, der $e_S(x_\mathcal{Q})$ maximiert.

Vorgehen

1. Zerlege das Einheitsquadrat in ein Gitter von Quadraten mit Seitenlänge ℓ.
2. Verwende in jeder Gitterzelle $\mathcal{Q}$ eine ϵ-Approximation.
3. Definiere dadurch Schätzfunktion e_S, sodaß für jedes $x \in \mathcal{Q}$ gilt:

$$|e_S(x) - \mu(x)| \leq \frac{\delta\pi\ell^2}{8}$$

4. Wähle Punkt $x_\mathcal{Q} \in \mathcal{Q}$, der $e_S(x_\mathcal{Q})$ maximiert.
5. Nimm als x_{app} den Punkt $x_\mathcal{Q}$, der $e_S(x_\mathcal{Q})$ maximiert.

Vorgehen

1. Zerlege das Einheitsquadrat in ein Gitter von Quadraten mit Seitenlänge ℓ.
2. Verwende in jeder Gitterzelle $\mathcal{Q}$ eine ϵ-Approximation.
3. Definiere dadurch Schätzfunktion e_S, sodaß für jedes $x \in \mathcal{Q}$ gilt:

$$|e_S(x) - \mu(x)| \leq \frac{\delta \pi \ell^2}{8}$$

4. Wähle Punkt $x_\mathcal{Q} \in \mathcal{Q}$, der $e_S(x_\mathcal{Q})$ maximiert.
5. Nimm als x_{app} den Punkt $x_\mathcal{Q}$, der $e_S(x_\mathcal{Q})$ maximiert.

Bringt uns das was?

Rechtfertigung

Ja das liefert uns tatsächlich ein ausreichend gutes x_{app}.

Beweis.

Rechtfertigung

Ja das liefert uns tatsächlich ein ausreichend gutes x_{app}.

Beweis.

$$\begin{aligned}\mu(x_{\text{app}}) &\geq e_S(x_{\text{app}}) - \frac{\delta\pi\ell^2}{8} \geq e_S(x_{\text{opt}}) - \frac{\delta\pi\ell^2}{8} \\ &\geq \mu_{\text{opt}} - \frac{\delta\pi\ell^2}{4} \geq \mu_{\text{opt}} - \delta\mu_{\text{opt}} = (1-\delta)\mu_{\text{opt}}\end{aligned}$$

□

Rechtfertigung

Ja das liefert uns tatsächlich ein ausreichend gutes x_{app}.

Beweis.

$$\begin{aligned}\mu(x_{\text{app}}) &\geq e_S(x_{\text{app}}) - \frac{\delta\pi\ell^2}{8} \geq e_S(x_{\text{opt}}) - \frac{\delta\pi\ell^2}{8} \\ &\geq \mu_{\text{opt}} - \frac{\delta\pi\ell^2}{4} \geq \mu_{\text{opt}} - \delta\mu_{\text{opt}} = (1-\delta)\mu_{\text{opt}}\end{aligned}$$

$\square$

offene Fragen:

- Wie definiert man e_S?

Rechtfertigung

Ja das liefert uns tatsächlich ein ausreichend gutes x_{app}.

Beweis.

$$\mu(x_{\text{app}}) \geq e_S(x_{\text{app}}) - \frac{\delta\pi\ell^2}{8} \geq e_S(x_{\text{opt}}) - \frac{\delta\pi\ell^2}{8}$$
$$\geq \mu_{\text{opt}} - \frac{\delta\pi\ell^2}{4} \geq \mu_{\text{opt}} - \delta\mu_{\text{opt}} = (1-\delta)\mu_{\text{opt}}$$

□

offene Fragen:

- Wie definiert man e_S?
- Wie findet man die x_Q?

Lokalität der Optimierung

Sei $\mathcal{Q}$ feste Gitterzelle, $x \in \mathcal{Q}$.

1. Reichweite von $V_T(x)$ ist höchstens ℓ

Lokalität der Optimierung

Sei $\mathcal{Q}$ feste Gitterzelle, $x \in \mathcal{Q}$.

1. Reichweite von $V_T(x)$ ist höchstens ℓ
2. $V_T(x)$ schneidet also nur $\mathcal{Q}$ und die acht benachbarten Zellen

Lokalität der Optimierung

Sei $\mathcal{Q}$ feste Gitterzelle, $x \in \mathcal{Q}$.

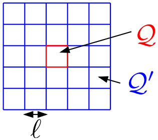
1. Reichweite von $V_T(x)$ ist höchstens ℓ
2. $V_T(x)$ schneidet also nur $\mathcal{Q}$ und die acht benachbarten Zellen
3. alle Punkte von T die an Definition von $V_T(x)$ beteiligt, liegen in $\mathcal{Q}$ und den 24 Gitterzellen mit Abstand höchstens 2ℓ

Lokalität der Optimierung

Sei $\mathcal{Q}$ feste Gitterzelle, $x \in \mathcal{Q}$.

1. Reichweite von $V_T(x)$ ist höchstens ℓ
2. $V_T(x)$ schneidet also nur $\mathcal{Q}$ und die acht benachbarten Zellen
3. alle Punkte von T die an Definition von $V_T(x)$ beteiligt, liegen in $\mathcal{Q}$ und den 24 Gitterzellen mit Abstand höchstens 2ℓ

$\mathcal{Q}'$:= Vereinigung dieser 25 Zellen, $T_{\mathcal{Q}} := T \cap \mathcal{Q}'$



Lemma

S sei quadratisches Gitter mit Dichte ϵ, C konvexer Körper mit Durchmesser höchstens D. Dann gilt:

$$|\mu(C) - \epsilon^2 |C \cap S|| \leq 4D\epsilon$$

Lemma

S sei quadratisches Gitter mit Dichte ϵ, C konvexer Körper mit Durchmesser höchstens D. Dann gilt:

$$|\mu(C) - \epsilon^2 |C \cap S|| \leq 4D\epsilon$$

Beweis.

Zerlege die Ebene in kleine Quadrate mit Seitenlänge ϵ, sodaß jeder Punkte von S im Zentrum eines kleinen Quadrats liegt. Der Rand von C schneidet höchstens $4D/\epsilon$ kleine Quadrate.

$\square$

ϵ-Approximation

Diesmal benutzen wir ein Gitter und keine randomisierte Abtastung.

ϵ-Approximation

Diesmal benutzen wir ein Gitter und keine randomisierte Abtastung.
$\epsilon := \delta\pi\ell/64$
S quadratisches Gitter mit Seitenlänge ϵ, das $\mathcal{Q}'$ überdeckt

ϵ-Approximation

Diesmal benutzen wir ein Gitter und keine randomisierte Abtastung.
$\epsilon := \delta\pi\ell/64$
S quadratisches Gitter mit Seitenlänge ϵ, das $\mathcal{Q}'$ überdeckt
Für $x \in \mathcal{Q}$ sei

$$e_S(x) = \epsilon^2 |V_T(x) \cap S|$$

Schätzung der Voronoi-Region von x

ϵ-Approximation

Diesmal benutzen wir ein Gitter und keine randomisierte Abtastung.
$\epsilon := \delta\pi\ell/64$
S quadratisches Gitter mit Seitenlänge ϵ, das $\mathcal{Q}'$ überdeckt
Für $x \in \mathcal{Q}$ sei

$$e_S(x) = \epsilon^2 |V_T(x) \cap S|$$

Schätzung der Voronoi-Region von x

$$|e_S(x) - \mu(x)| \leq 8\ell\epsilon \leq \delta\pi\ell^2/8$$

ϵ-Approximation

Diesmal benutzen wir ein Gitter und keine randomisierte Abtastung.
$\epsilon := \delta\pi\ell/64$
S quadratisches Gitter mit Seitenlänge ϵ, das $\mathcal{Q}'$ überdeckt
Für $x \in \mathcal{Q}$ sei

$$e_S(x) = \epsilon^2 |V_T(x) \cap S|$$

Schätzung der Voronoi-Region von x

$$|e_S(x) - \mu(x)| \leq 8\ell\epsilon \leq \delta\pi\ell^2/8$$

Denn: Durchmesser $V_T(x)$ ist höchstens 2ℓ

ϵ-Approximation

Diesmal benutzen wir ein Gitter und keine randomisierte Abtastung.
$\epsilon := \delta\pi\ell/64$
S quadratisches Gitter mit Seitenlänge ϵ, das $\mathcal{Q}'$ überdeckt
Für $x \in \mathcal{Q}$ sei

$$e_S(x) = \epsilon^2 |V_T(x) \cap S|$$

Schätzung der Voronoi-Region von x

$$|e_S(x) - \mu(x)| \leq 8\ell\epsilon \leq \delta\pi\ell^2/8$$

Denn: Durchmesser $V_T(x)$ ist höchstens 2ℓ

Jetzt fehlt uns nur noch $x_{\mathcal{Q}}$!

Arrangement berechnen

Für $s \in S$:
$W(s) = \{x \in \mathcal{Q} \mid s \in V_T(x)\}$

$W(s)$ ist größte Kreisscheibe mit Zentrum s, die keinen Punkt aus T enthält, geschnitten mit $\mathcal{Q}$

Arrangement berechnen

Für $s \in S$:
$W(s) = \{x \in \mathcal{Q} \mid s \in V_T(x)\}$

$W(s)$ ist größte Kreisscheibe mit Zentrum s, die keinen Punkt aus T enthält, geschnitten mit $\mathcal{Q}$

$\mathcal{W}_S = \{W(s) \mid s \in S\}$

Arrangement berechnen

Für $s \in S$:
$W(s) = \{x \in \mathcal{Q} \mid s \in V_T(x)\}$

$W(s)$ ist größte Kreisscheibe mit Zentrum s, die keinen Punkt aus T enthält, geschnitten mit $\mathcal{Q}$

$\mathcal{W}_S = \{W(s) \mid s \in S\}$

Betrachte Arrangement $\mathcal{A}(\mathcal{W}_S)$.

Wie zuvor:
suche Punkt, der in der größten Anzahl Mengen in $\mathcal{W}_S$ enthalten ist.

Resultat

Satz

Gegeben eine Menge T von n Punkten in der Ebene und einen Parameter $\delta > 0$, kann man in $O(n/\delta^4 + n\log(n))$ Zeit deterministisch einen Punkt x_{app} berechnen, sodaß $\mu(x_{\text{app}}) \geq (1-\delta)\mu_{\text{opt}}$.

Resultat

Satz

Gegeben eine Menge T von n Punkten in der Ebene und einen Parameter $\delta > 0$, kann man in $O(n/\delta^4 + n\log(n))$ Zeit deterministisch einen Punkt x_{app} berechnen, sodaß $\mu(x_{\text{app}}) \geq (1-\delta)\mu_{\text{opt}}$.

Beweis.

- Voronoi-Diagramm, Reichweite $\ell \Rightarrow O(n\log(n))$

Resultat

Satz

Gegeben eine Menge T von n Punkten in der Ebene und einen Parameter $\delta > 0$, kann man in $O(n/\delta^4 + n\log(n))$ Zeit deterministisch einen Punkt x_{app} berechnen, sodaß $\mu(x_{\text{app}}) \geq (1-\delta)\mu_{\text{opt}}$.

Beweis.

- Voronoi-Diagramm, Reichweite $\ell \Rightarrow O(n\log(n))$
- für jede Gitterzelle $\mathcal{Q}$: ϵ-Gitter, $\epsilon = \delta\pi\ell/64 \Rightarrow$ $M = 25\ell^2/\epsilon^2 = O(1/\delta^2)$ Punkte

Resultat

Satz

Gegeben eine Menge T von n Punkten in der Ebene und einen Parameter $\delta > 0$, kann man in $O(n/\delta^4 + n\log(n))$ Zeit deterministisch einen Punkt x_{app} berechnen, sodaß $\mu(x_{\text{app}}) \geq (1-\delta)\mu_{\text{opt}}$.

Beweis.

- Voronoi-Diagramm, Reichweite $\ell \Rightarrow O(n\log(n))$
- für jede Gitterzelle $\mathcal{Q}$: ϵ-Gitter, $\epsilon = \delta\pi\ell/64 \Rightarrow$ $M = 25\ell^2/\epsilon^2 = O(1/\delta^2)$ Punkte
- Für $s \in S$ berechne $W(s)$: finde nächsten Nachbarn in $T_{\mathcal{Q}}$

Resultat

Satz

Gegeben eine Menge T von n Punkten in der Ebene und einen Parameter $\delta > 0$, kann man in $O(n/\delta^4 + n\log(n))$ Zeit deterministisch einen Punkt x_{app} berechnen, sodaß $\mu(x_{\text{app}}) \geq (1-\delta)\mu_{\text{opt}}$.

Beweis.

- Voronoi-Diagramm, Reichweite $\ell \Rightarrow O(n\log(n))$
- für jede Gitterzelle $\mathcal{Q}$: ϵ-Gitter, $\epsilon = \delta\pi\ell/64 \Rightarrow$ $M = 25\ell^2/\epsilon^2 = O(1/\delta^2)$ Punkte
- Für $s \in S$ berechne $W(s)$: finde nächsten Nachbarn in $T_{\mathcal{Q}}$
- Berechne Arrangement mit Sweepline-Algorithmus: $O(M^2)$

Resultat

Satz

Gegeben eine Menge T von n Punkten in der Ebene und einen Parameter $\delta > 0$, kann man in $O(n/\delta^4 + n\log(n))$ Zeit deterministisch einen Punkt x_{app} berechnen, sodaß $\mu(x_{\text{app}}) \geq (1-\delta)\mu_{\text{opt}}$.

Beweis.

- Voronoi-Diagramm, Reichweite $\ell \Rightarrow O(n\log(n))$
- für jede Gitterzelle $\mathcal{Q}$: ϵ-Gitter, $\epsilon = \delta\pi\ell/64 \Rightarrow$ $M = 25\ell^2/\epsilon^2 = O(1/\delta^2)$ Punkte
- Für $s \in S$ berechne $W(s)$: finde nächsten Nachbarn in $T_{\mathcal{Q}}$
- Berechne Arrangement mit Sweepline-Algorithmus: $O(M^2)$
- Jede Gitterzelle hat Abstand höchsten 2ℓ zu Punkt von T. $\Rightarrow$ Anzahl Gitterzellen: O(n).

Bewertung

Konstruktion funktioniert sehr gut. Laufzeit von $O(n/\delta^4 + nlog(n))$ ist vermutlich nahezu optimal.

Bewertung

Konstruktion funktioniert sehr gut. Laufzeit von $O(n/\delta^4 + n log(n))$ ist vermutlich nahezu optimal.

Warum ist das Verfahren hier viel schneller?

Bewertung

Konstruktion funktioniert sehr gut. Laufzeit von $O(n/\delta^4 + n\log(n))$ ist vermutlich nahezu optimal.

Warum ist das Verfahren hier viel schneller?

- haben gute obere Schranke für die Größe
- konnten also die Suche gut einschränken

Bewertung des Algorithmus für das Guard-Problem

Haben Algorithmus, der in $O((n^5/\delta^4)\log^3(n/\delta))$ mit hoher Wahrscheinlichkeit eine gute Lösung findet.

Bewertung des Algorithmus für das Guard-Problem

Haben Algorithmus, der in $O((n^5/\delta^4)\log^3(n/\delta))$ mit hoher Wahrscheinlichkeit eine gute Lösung findet.

- Algorithmus relativ langsam, da Sample groß
- Sample groß, da ϵ klein genug für Extremfall $\mu_{\mathrm{opt}} \approx 1/n$

Bewertung des Algorithmus für das Guard-Problem

Haben Algorithmus, der in $O((n^5/\delta^4)\log^3(n/\delta))$ mit hoher Wahrscheinlichkeit eine gute Lösung findet.

- Algorithmus relativ langsam, da Sample groß
- Sample groß, da ϵ klein genug für Extremfall $\mu_{\text{opt}} \approx 1/n$

Ausblick:
$\epsilon -$ *Approximation* nicht wirklich nötig:

- müssen nicht jeden Sichtbereich gut abschätzen können
- würde genügen dort exakt zu arbeiten wo die Lösung liegt, an den anderen Stellen nicht überschätzen

Bewertung des Algorithmus für das Guard-Problem

Haben Algorithmus, der in $O((n^5/\delta^4)\log^3(n/\delta))$ mit hoher Wahrscheinlichkeit eine gute Lösung findet.

- Algorithmus relativ langsam, da Sample groß
- Sample groß, da ϵ klein genug für Extremfall $\mu_{\mathrm{opt}} \approx 1/n$

Ausblick:
$\epsilon -$ *Approximation* nicht wirklich nötig:

- müssen nicht jeden Sichtbereich gut abschätzen können
- würde genügen dort exakt zu arbeiten wo die Lösung liegt, an den anderen Stellen nicht überschätzen

Im Paper wird $O(\frac{n}{\mu_{\mathrm{opt}}\delta^4}\log^3\frac{n}{\delta})$ erreicht.

Verfahren

- Beide Probleme lassen sich ähnlich lösen
- Legt Anwendbarkeit auf weitere Probleme nahe
- Im Paper weiterhin bearbeitet: Shape matching, Gallery guarding

Verfahren

- Beide Probleme lassen sich ähnlich lösen
- Legt Anwendbarkeit auf weitere Probleme nahe
- Im Paper weiterhin bearbeitet: Shape matching, Gallery guarding

Was hat das alles eigentlich mit
„Randomisierte Algorithmen in der algorithmischen Geometrie“
zu tun?

Verfahren

- Beide Probleme lassen sich ähnlich lösen
- Legt Anwendbarkeit auf weitere Probleme nahe
- Im Paper weiterhin bearbeitet: Shape matching, Gallery guarding

Was hat das alles eigentlich mit
„Randomisierte Algorithmen in der algorithmischen Geometrie“
zu tun?

Im ersten Moment nicht sehr viel: Allgemeines Verfahren kommt ohne Randomisierte Konstruktionen aus.

Verfahren

- Beide Probleme lassen sich ähnlich lösen
- Legt Anwendbarkeit auf weitere Probleme nahe
- Im Paper weiterhin bearbeitet: Shape matching, Gallery guarding

Was hat das alles eigentlich mit
„Randomisierte Algorithmen in der algorithmischen Geometrie“
zu tun?

Im ersten Moment nicht sehr viel: Allgemeines Verfahren kommt ohne Randomisierte Konstruktionen aus.

Aber: Vernünftige Schätzer zu finden ist schwer.
Oft einfacher zu zeigen: Eine Konstruktion liefert mit hoher Wahrscheinlichkeit guten Schätzer

Fragen?

Vielen Dank
für die Aufmerksamkeit