

Fortgeschrittene Methoden der Bioinformatik - Prüfungsskript

Inhaltsverzeichnis

1	Grundlagen	1
2	Generische Repräsentation	2
3	Pebble Game	3
4	Rigidity in 3D	3
4.1	Computation Complexity:	6
4.2	Zusammenfassung sparse graphs	6
4.3	Pebble Collection	7
5	von 2D zu 3D	7
6	Abbie Algorithmus	9

1 Grundlagen

Geg.:

X von Punkten in R^3 , $\forall x, y \in X$

$d(x, y)$: euklidische Distanz bekannt

Ges.:

Finde 3D-Repräsentation mit $\Phi : x \rightarrow R^3$ (Abbildung von x in 3D-Raum)

$$\underbrace{\|\Phi(x) - \Phi(y)\|}_{ges} = \sqrt{\underbrace{\sum_{i=1}^3 (\Phi_i(x) - \Phi_i(y))^2}_{geg}}$$

Kongruenztransformation:

- Verschiebung: $\Phi' = \Phi + \tau$
- Rotation: $\Phi' = R \cdot \Phi$ (Spiegelung des Koordinatensystems)

rechnerei...

- mit 7 Eigenwerten genau 7 Dimensionen, bei Eigenwerten = 0 \rightarrow Dimension überflüssig
- negative Eigenwerten nicht im euklidischen Raum darstellbar
 - **Lösung:** Distanz nicht exakt einbettbar \rightarrow negative Eigenwerte weglassen

Was machen wenn nur Teile der Distanzen vorhanden sind?

Qualität prüfen:
$$\sum_{\underbrace{k,l}_{\text{ueber alle bekannten Distanzen } k,l}} (\sqrt{(x^k - x^l)^2} - d_{kl})^2 \rightarrow \min (= 0)$$

Wann sind genug Informationen bekannt?

- Graph $G(V, E)$, $|V| = n$, $|E| = m$
- Konfiguration $p : V \rightarrow R^d$ für $d = 2$
- Framework (G, p)

Definition: Ein Framework ist flexibel wenn es eine stetige Deformation $p \rightarrow p'$ gibt, sodass alle Abstände in G (=Kantenlängen) erhalten bleiben. Andernfalls ist (G, p) rigid.

minimal rigid:

- rigid
- entfernen einer Kante führt zu einem Teilgraphen der flexibel ist

Gibt es eine mögliche Bewegungsfreiheit?

$(p^i - p^j)(p^i - p^j) = d_{ij}^2$ in Abhängigkeit der Zeit darstellen:

$$\begin{aligned}\frac{\partial}{\partial t}(p^i - p^j)(p^i - p^j) &= \left(\frac{\partial}{\partial t}p^i - \frac{\partial}{\partial t}p^j\right)(p^i - p^j) + \left(\frac{\partial}{\partial t}p^i - \frac{\partial}{\partial t}p^j\right)(p^i - p^j) \\ \frac{\partial}{\partial t}p^i &= v^i, \quad \frac{\partial}{\partial t}p^j = v^j \\ &= 2(v^i - v^j)(p^i - p^j) \\ 2 \cdot \underbrace{(v^i - v^j)}_* (p^i - p^j) &= \frac{\partial}{\partial t}d_{ij}^2 = 0 \quad \forall \{i, j\} \in E\end{aligned}$$

* wenn Verschiebung bzw. Bewegungsfreiheit existiert, dann existiert hier eine Lösung

Definition: G mit Präsentation p ist "infinitesimal flexibel" wenn es eine Lösung $(v^1, v^2, v^3, \dots, v^n) \neq \vec{0}$ von $(p^i - p^j)(v_i - v_j) = 0$ gibt ($\forall \{i, j\} \in E$)

2 Generische Repräsentation

Eigenschaften: rigid \Leftrightarrow infinitesimal rigid \Rightarrow generisch rigid

Benötigte Eigenschaften für rigid:

- pro Vertex 2 Verbindungen
- $m=2n-3$ (externe FG)
 - m = Anzahl der Kanten
 - n = interne Freiheitsgrade pro Knoten
 - externe FG: 2 Translation + 1 Rotation

2 Punkte fix = 4 Freiheitsgrade

- 1 Distanz = 3 Freiheitsgrade übrig

$$\left. \begin{array}{l} rigid < 2n - 3 \\ minimal\ rigid = 2n - 3 \end{array} \right\} + \text{sinnvolle Verteilung der Kanten}$$

Laman Theorem 1: G ist generisch minimal rigid in 2D \Leftrightarrow

- $m = 2n-3$
- Laman-Bedingung: G enthält keine Teilgraphen mit k Knoten und $m' > 2k-3$ Kanten (unabhängiges System von Kanten)

Graphen, die diese Bedingungen erfüllen, heißen Laman-Graphen

Beispiel Henneberg-Konstruktion

Start mit einer Kante, dann Iteration: Addition von 1 Knoten x

- Typ 1: verbinde x mit vorhandenen Knoten mittels zwei neuer Kanten
- Typ 2: finde 3 Knoten u, v, w mit mindestens 1 Kante in $G(u, v, w)$ induziert ist. Verbinde x mit u, v, w , lösche eine Kante aus $G(u, v, w)$

Laman Theorem 2: Die Kanten von G sind unabhängig in 2D \Leftrightarrow
Für jede Kante (a,b) in G hat der Multigraph G_{4l} der durch Vervierfachung von (a,b) entsteht, keinen induzierten Teilgraphen mit $m' > 2n'$ Kanten und n' Knoten

3 Pebble Game

4 Endresultate:

- wellconstraint (minimal rigid): Anzahl pebbles = l , alle Kanten eingesetzt
- overconstraint (redundant): Anzahl pebbles = l , nicht alle Kanten eingesetzt
- underconstraint (flexible): Anzahl pebbles $> l$, alle Kanten eingesetzt
- other: Anzahl pebbles $> l$, nicht alle Kanten eingesetzt

4 Rigidity in 3D

jetzt Laman-Bedingungen analog?

2D:

- $m=2n-3$ (Vollständigkeit)
- \forall Teilgraphen $m' \leq 2n'-3$ (Unabhängigkeit)
 - $2n$ Freiheitsgrade für n Punkte (2 Translationen)
 - 3 Freiheitsgrade eines allgemeinen starren Körpers in 2D (Dimensionen der Symmetriegruppe)

jetzt 3D:

- 3 Freiheitsgrade pro Punkt (3 Translationen)
- 6 Freiheitsgrade eines allgemeinen starren Körpers in 3D (3 Translation + 3 Rotation)

Hoffnung das gilt:

- $m=3n-6$
- \forall Teilgraphen $m' \leq 3n'-6$ für $n' \geq 3$ (notwendige Bedingung)

jedoch weitere Bedingungen notwendig!!! (siehe Beispiel Doppelbanane)

bei Molekülen:

- body-hinge-framework
- Interpretation: 1 hinge = 5 joints \rightarrow so bleibt 1 Freiheitsgrad offen

Multigraph:

- $V \leftrightarrow$ Bodies
- $E \leftrightarrow$ hinges \leftrightarrow chemische Einfachbindung \leftrightarrow 5-fach Kanten = joints

Beschreibung der rigidity von body-joint-frameworks

bei Doppelbindungen werden zusätzliche Kanten eingefügt, da in der Ebene fixiert werden muss

(k,l)-sparse graphs: Verallgemeinerung der Laman-Graphen ((2,3)-sparse-graphs (rigid graph))

Ein Graph (V,E) ist (k,l) -sparse wenn:

1. Unabhängigkeitsbedingung: jede Teilmenge V' von V spannt höchstens $|E'| \leq k \cdot |V'| - l$ Kanten auf (benötigt für genügend nicht redundante Kanten)

Ein Graph (V,E) ist (k,l) -tight wenn:

1. Unabhängigkeitsbedingung **UND**
2. Vollständigkeitsbedingung: $|E| = k \cdot |V| - l$ (benötigt für Verbrauch aller entsprechend für rigidity nicht gebrauchten Freiheitsgrade)

Ein Graph H ist $(k,l)=(V,F)$ steif, genau dann wenn er einen spannenden (k,l) -sparse Teilgraphen $G=(V,E)$ $E \subseteq F$ enthält (spannend: alle Knoten, nicht alle Kanten)

Die Kanten $F \setminus E$ sind "redundant"

Gegeben ein beliebiger Graph G , nennen wir eine Kantenmenge E (k,l) -sparsity-unabhängig wenn für jede Teilmenge $B \subseteq E$ gilt:

$|B| \leq k|V(B)| - l$ mit $|V(B)|$ =Knotenmenge die von B aufgespannt wird

(k,l)-sparsity-Unabhängigkeit für $1 \leq l \leq 2k$ definiert einen Matroiden
 Einschränkung: mindestens 2 Knoten da sonst keine Kante aufgespannt werden kann

Maxwell-Unabhängigkeit:

Definition $\underbrace{\binom{d}{k}}_k, \underbrace{\binom{d+1}{2}}_l$ ähnlich sparsity-Definition: $\underbrace{\binom{d+1}{2}}_k, \underbrace{\binom{d+1}{2}}_l$

mit k =Anzahl Raumdimensionen, l =Anzahl Freiheitsgrade im \mathbb{R}^d

$|E'| \leq d|V(E')| - \frac{d(d+1)}{2}$ für alle E' mit $|V(E')| \geq d$

→ macht Matroideigenschaft kaputt

→ Bar-Joint nicht in 3D mit Pebble-Game lösbar!

Matroid $M=(X,I)$: Grundmenge X und unabhängige Menge I , wenn Teilmenge $A \subseteq X$ auch $A \in I \rightarrow A$ unabhängig

Eigenschaften:

1. $\emptyset \in I$
2. $A \in I$ und $B \subseteq A \Rightarrow B \in I$
3. Austauschaxiom: $A, B \in I, |A| < |B|, \exists x \in B \setminus A \mid A \cup \{x\} \in I$

Greedy: ...

Wegen 2. haben alle Lösungen B die gleiche Anzahl von Elementen unabhängig von der Reihenfolge in der X durchlaufen wird

nochmal Pebble Game

gerichteter Graph D , Knotenmenge V , Knoten v , Teilgraph $V' \subset V$

Funktionen:

- $\text{peb}(v)$: Anzahl der Pebbles an Knoten v
- $\text{span}(v)$: Anzahl loops an v
- $\text{out}(v)$: outdegree exklusive Loops
- $\text{peb}(V')$: Anzahl der Knoten in Teilgraphen V'
- $\text{span}(V')$: Anzahl der Kanten, die von V' aufgespannt sind, inklusive Loops
- $\text{out}(v')$: Anzahl der Kanten die von V' nach $V \setminus V'$ zeigen

Lemma: Invarianten des Pebble Games mit Graph G , Knoten v , Teilgraph $V' \subset V$

(Invarianten: Eigenschaften, die über das Spiel gleich bleiben)

1. $\text{peb}(v) + \text{span}(v) + \text{out}(v) = k$
2. $\text{peb}(V') + \text{span}(V') + \text{out}(V') = k \cdot |V'|$
3. $\text{peb}(V') + \text{out}(V') \geq l$ (am Ende des Spiels bleiben mind. l Pebbles im Spielfeld)
4. $\text{span}(V') \leq k|V'| - l$

Definition Block: ein Teilgraph G' mit

- $|E'| = k|V'| - l$
 - $E(G')$ ist (k, l) -sparsity-independent
- $\Rightarrow (k, l)$ -sparsity-rigid Teilgraph

Sparsity-Komponente: Ein maximaler Block bzgl. Mengen-Inklusion
 V' spannt einen Block auf $\Leftrightarrow \text{peb}(V') + \text{out}(V') = l$

- l Pebbles am Ende \Rightarrow steif
- Kanten zurückgewiesen, $|E| > |E(D)| \Rightarrow$ überbestimmt bzw. Redundanzen enthalten

Lemma*: wenn $e(u, v) \cup E(D)$ unabhängig, $\text{peb}(u) + \text{peb}(v) < l + 1 \Rightarrow$ dann gibt es ein Pebble in $\text{Reach}(u) \cup \text{Reach}(v)$ das nach u oder v transportiert werden kann

Zusammenfassung: Eine Kante e wird in D eingesetzt $\Leftrightarrow e \cup E(D)$ unabhängig
 Beweis: Lemma* anwenden bis genug Pebbles in u, v gesammelt \Rightarrow unabhängig
 \Rightarrow kann eingesetzt werden

Invariante 4 \Leftarrow eingesetzt \Rightarrow unabhängig

Theorem: Das (k, l) pebble game erkennt korrekt (k, l) -sparsity Unabhängigkeit

4.1 Computation Complexity:

Laufzeit: $\mathcal{O}(k \cdot l \cdot |V| \cdot |E|)$

Speicher: $\mathcal{O}(|V| + |E|)$

4.2 Zusammenfassung sparse graphs

(k, l) -sparse graphs: $|E'| \leq k|V'| - l$

(k, l) -tight/rigid $\Leftrightarrow (k, l)$ -sparse und $|E'| = k|V'| - l$

blocks:

(k,l) -sparsity-block: Teilgraph $G'(V',E')$ eines (k,l) -sparse graphs $|E'| = k|V'| - l$
 $\Leftrightarrow (k,l)$ -pebble game

- Konstruktion eines DiGraphen D' mit orientierten unabhängigen Kanten
 $\forall v \in V: \text{peb}(v)$
- genau l Pebbles in $D \Leftrightarrow$ steif
- minimal steif \Rightarrow keine Kanten wurden als redundant verworfen

Definition component:

$V' \subseteq V$ ist eine (k,l) -sparsity componente von G wenn

1. V' ein Block ist
2. V' maximal bezüglich Knotenmenge \rightarrow also maximal erweiterter Block

Eigenschaften von component:

$B_1(V_1, E_1), B_2(V_2, E_2)$ zwei (k,l) -sparse blocks

- für $0 \leq l \leq k$ und $V_1 \cap V_2 \neq \emptyset$ gilt: $V_1 \cup V_2$ und $V_1 \cap V_2$ sind ebenfalls Blocks
- für $k < l < 2k$ und $|V_1 \cap V_2| \geq 2$ gilt: $V_1 \cup V_2$ und $V_1 \cap V_2$ sind ebenfalls Blocks

Lemma: Für $l > 0$: jede (k,l) -sparsity component ist zusammenhängend

V' ist eine (k,l) -sparsity-component \Rightarrow in V' gibt es genau l pebbles ($\text{peb}(u) + \text{peb}(v) = l$) nach einsetzen von e

Nach dem Einsetzen von $e=(u,v)$ gibt es in der component, die von u und v aufgespannt wird, keine weiteren pebbles

4.3 Pebble Collection

zwei grundlegende Möglichkeiten (Algorithmen)

5 von 2D zu 3D

bisher 2D bar+joint frameworks:

- \Leftrightarrow Laman graphs
- $\Leftrightarrow (2,3)$ -sparse graphs

jetzt in 3D: (3,6)-sparse graphs
2 Probleme:

1. (3,6)-sparsity ist kein Matroid \rightarrow pebble game verwendbar um unabhängige Mengen von Kanten zu bekommen aber maximal unabhängige Menge nicht eindeutig
2. rigid in 3D \Rightarrow (3,6)-sparse \rightarrow Umkehr ist aber falsch (siehe Maxwell: Gegenbeispiel Doppelbanane)

Daher nun body + bar frameworks

Übersetzung in Multigraphen: Verwandschaft zu body + hinge framework
- 5 generische bars = 1 hinge

Moleküle: bond bending frameworks (Molekülgraphen)

- steife Stäbe, fixe Längen
- fixe Bindungswinkel

Für Graph G wenn Graph H existiert sodass $G=H^2$

Quadrat des Graphen: $e=(u,v) \in H^2$ wenn $x \exists ux$ und xv in H oder $(u,v) \in H$
Verbindung von rigid-Clustern (components)

1. pivot-joint: $B_1 \cap B_2 = \{v\}$ (zwei Graphen B_1 und B_2 über genau einen Knoten v verbunden)
2. edge-joint: zwei Graphen B_1 und B_2 über genau eine Kante (u,v) verbunden
3. implied-hinge-joint: $(u,v) \in E$ aber (u,v) in Block $\rightarrow u,v$ liegen auf Achse, die jedoch keine Kante ist (siehe Doppelbanane)

Lemma: Wenn $G=H^2$ dann gibt es weder pivot-joints noch implied-hinge-joints

Theorem: Die Kanten eines Graphen $G=H^2$ sind unabhängig bzgl. rigidity in 3D \Leftrightarrow für alle Teilgraphen mit $|V'| \geq 3$ Knoten $|E'| \leq 3|V'|-6$ gilt $G=H^2$ ist minimal generisch steif in 3D \Leftrightarrow die Kanten Maxwell-unabhängig sind und $|E|=3|V|-6$

dies stimmt für Moleküle! jedoch keine Matroid-Eigenschaften!!!

\rightarrow keine Garantie, dass durch einsetzen der Kanten rigid-Basis erzeugt wird

6 Abbie Algorithmus

Positionsproblem lösen:

set of vertices V und set of edges E , $e_{ij} \in E$, falls es bekannte Distanz i,j gibt

Realisation: mapping $P: (v \rightarrow \mathbb{R}^3)$

molecule problem: sei $p_i = P(v_i \in V)$

Problem minimieren: $F(P) = \sum_{e_{ij} \in E} (\underbrace{|p_i - p_j|^2}_{\text{Distanzquadrate}} - \underbrace{d_{ij}^2}_{\text{bekannte Distanz}})^2$

$\rightarrow \mathcal{O}(n^P)$: lange Laufzeit, daher optimieren

Divide and Conquer Algorithmus

Abbie (1)

1. finde maximale, einzigartig realisierbare Teilgraphen
 \rightarrow für jeden Teilgraphen: klein genug für $\mathcal{O}(n^P)$?
2. JA: setze Position via globaler Optimierung
3. NEIN: zerlege in kleinere Teilprobleme, rufe 1. rekursiv auf
4. kombiniere via globaler Optimierung

notwendige Bedingungen für eindeutig einbettbar (siehe 1.):

- Vertex $(d+1)$ -connected
- redundant rigid

max unique (2)

1. wenn $g = K_{5,5} \rightarrow$ kein einzigartiger Teilgraph
2. wenn g nicht 4-vertex-connected \rightarrow rekursiv 4-vertex-connected Teilgraphen lösen
3. wenn g nicht redundant steif \rightarrow rekursiv auf redundant steifen Teilgraphen lösen
4. wenn auf hinreichende Bedingungen positiv getestet $\rightarrow g$ einzigartig
5. sonst ??? \rightarrow interessanter Graph

hinreichende Bedingungen für eindeutig einbettbar (siehe 4.):

- Stressmatrix mit Nullity $\geq d+1$
 - Nullity: $\text{Nullity}(G) = n - r(A) \dots$ mit Graph G mit n Knoten, A =Adjazenzmatrix, $r(A)$: Rang der Adjazenzmatrix

QR-Faktorisierung: redundant steife Komponenten (3)

- unabhängiges Set von redundanten Kanten finden (in Q)
- um Basis für verbliebene Flexe zu finden
- \forall 3-Cliquen x,y,z im induzierten Graph $\rightarrow \forall v \neq x,y,z$ falls v induzierte Kante zu x,y,z hat \rightarrow füge v zum wachsenden Teilgraphen von x,y,z hinzu

kombinatorische Positionierung

1. 2 Chunks teilen sich 4 Knoten \rightarrow dadurch nur eine Variante zur Vereinigung (Knoten 1...4 vereinen)
2. 1 Knoten mit 4 Kanten in Chunk \rightarrow verbinden
3. 1 Knote mit nur 3 Kanten (valent) in Chunk \rightarrow Spiegelung möglich \rightarrow ausprobieren welche die bessere Variante ist (Heuristik verwenden)
4. 2 valente Knoten zu Chunk + 1 Kante zwischen v und $w \rightarrow$ kann verbunden werden (valent: 3 Kanten zum Basechunk)
5. 2 valente Chunks zum Basechunk verbinden/hinzufügen
6. 1 valenter Chunk und 1 valenter Knoten zum Basechunk verbinden/verbinden

Optimierung von F' als Minimierungsfunktion

- vertex: 3 Freiheitsgrade (Position)
- chunk: 6 Freiheitsgrade (Position, Rotation)
- für alle Spiegelungen S: 2^{S-1} Möglichkeiten
- einen chunk fixieren als "base"
- gemeinsame Knoten mit "base"
 - (3): nur Spiegelung
 - (2): Rotation (Hinge)
 - (1): 3 Rotationsmöglichkeiten
 - (0): alle 6 Freiheitsgrade