Zrandomizowane algorytmy aproksymacyjne

Jarosław Byrka

Instytut Informatyki
Uniwersytet Wrocławski

Warszawa 07.10.2010

Aproksymacja: jakość losowego rozwiązania

1 Zrandomizowane algorytmy aproksymacyjne

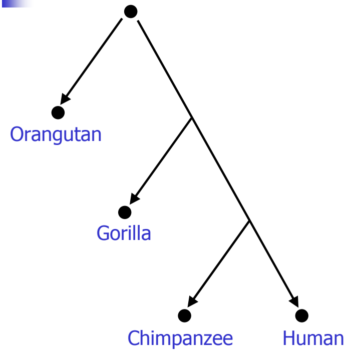
2 Drzewa filogenetyczne

3 Przybliżone equilibrium Nash'a

4 Problemy lokalizacji

5 Drzewa Steinera

Phylogenetic tree reconstruction



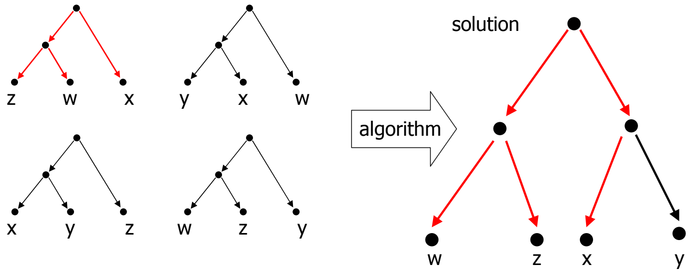
Phylogenetic tree reconstruction is essentially the science of **efficiently inferring and constructing plausible evolutionary trees** when we only have limited input data about the 'species' concerned...

At the intersection of biology, bioinformatics, computer science and mathematics.

(This tree borrowed from a presentation by Tandy Warnow)

Triplet-based methods (2)

- For example. Suppose I want to reconstruct a plausible evolution for the species set {w,x,y,z}.
- I am given a set of rooted triplets zw|x, yx|w, xy|z, wz|y. (Note zw|x = wz|x.)



Gry i equilibira

	Zdr.		Loj.	
Zdrada	4	4	0	5
Lojalność	5	0	1	1

Gra: *"dylemat więźnia"*; cyfry: ilość lat w więzieniu.

- equilibrium = stabilny przydział strategii,
  t.j. nikt nie chce samodzielnie zmieniać swojej strategii
- mixed strategy = rozkład prawdopodobieństwa po strategiach
- dla każdej gry istnieje equilirbrium [Nash'51]
- problem znajdowania equilibrium jest PPAD-zupełny nawet dla dwóch graczy [Daskalakis, Goldberg, Papadimitriou '06; Chen, Deng, Teng '06]
- zaproponowaliśmy algorytm znajdujący przybliżone equilibria

Gry i equilibira

	Zdr.		Loj.	
Zdrada	4	4	0	5
Lojalność	5	0	1	1

Gra: *"dylemat więźnia"*; cyfry: ilość lat w więzieniu.

- equilibrium $=$ stabilny przydział strategii,
  t.j. nikt nie chce samodzielnie zmieniać swojej strategii
- mixed strategy $=$ rozkład prawdopodobieństwa po strategiach
- dla każdej gry istnieje equilirbrium [Nash'51]
- problem znajdowania equilibrium jest PPAD-zupełny nawet dla dwóch graczy [Daskalakis, Goldberg, Papadimitriou '06; Chen, Deng, Teng '06]
- zaproponowaliśmy algorytm znajdujący przybliżone equilibria

Gry i equilibira

	Zdr.		Loj.	
Zdrada	4	4	0	5
Lojalność	5	0	1	1

Gra: *"dylemat więźnia"*; cyfry: ilość lat w więzieniu.

- equilibrium $=$ stabilny przydział strategii,
  t.j. nikt nie chce samodzielnie zmieniać swojej strategii
- mixed strategy $=$ rozkład prawdopodobieństwa po strategiach
- dla każdej gry istnieje equilirbrium [Nash'51]
- problem znajdowania equilibrium jest PPAD-zupełny nawet dla dwóch graczy [Daskalakis, Goldberg, Papadimitriou '06; Chen, Deng, Teng '06]
- zaproponowaliśmy algorytm znajdujący przybliżone equilibria

Outline

1 Zrandomizowane algorytmy aproksymacyjne

2 Drzewa filogenetyczne

3 Przybliżone equilibrium Nash'a

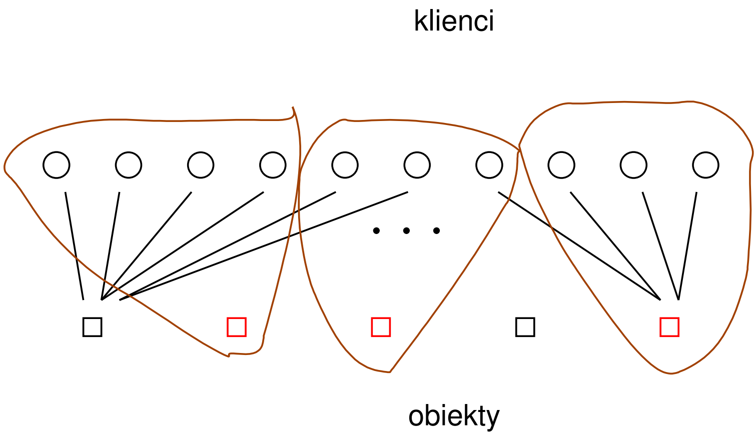
4 Problemy lokalizacji

5 Drzewa Steinera

UFLP - opis problemu

- zbiór klientów C
- zbiór lokalizacji F wraz z kosztem f_i wybudowania obiektu w $i \in F$
- koszt c_{ij} korzystania z obiektu $i \in F$ przez klienta $j \in C$
- cel: wybudować obiekty i przypisać każdemu klientowi obiekt, tak aby łączny koszt był możliwie najmniejszy
- założenie: koszty c_{ij} są metryczne,
  t.j. można o nich myśleć jak o fizycznej odległości.

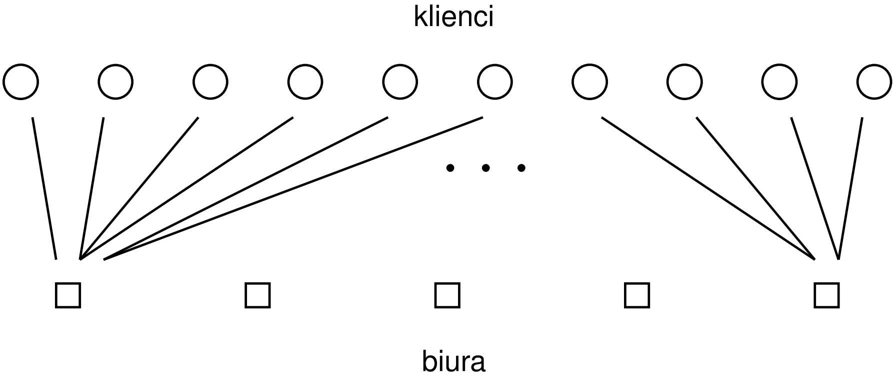
UFLP - przykład



UFLP - program całkowitoliczbowy

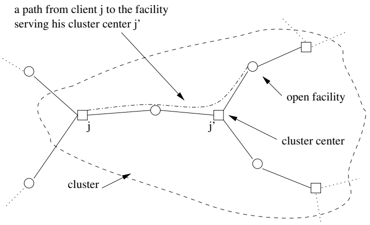
minimize	$\sum_{i \in \mathcal{F}, j \in \mathcal{C}} c_{ij} x_{ij} + \sum_{i \in \mathcal{F}} f_i y_i$		
subject to	$\sum_{i \in \mathcal{F}} x_{ij} = 1$	for all $j \in \mathcal{C}$	(i)
	$x_{ij} - y_i \leqslant 0$	for all $i \in \mathcal{F}, j \in \mathcal{C}$	(ii)
	$x_{ij}, y_i \in \{0, 1\}$	for all $i \in \mathcal{F}, j \in \mathcal{C}$	(iii)
w LP rel.	$x_{ij}, y_i \in \mathcal{R}^+$	zamiast (iii)	

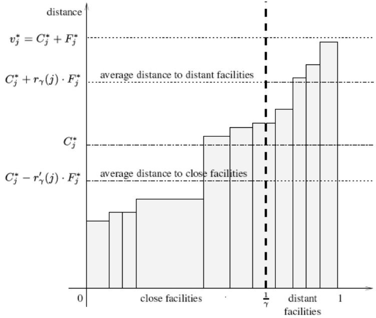
UFLP - bardziej życiowy przykład



Washington Mutual Inc. JPMorgan Chase & Co

UFLP: metody i wyniki





Nasze wyniki:

- 1.5-aproksymacja dla metrycznego UFL
- 2.492-aproksymacja dla 3-poziomowej wersji UFL
- algorytmy dla wersji *fault tolerant* i *2-stage stochastic*

Outline

1 Zrandomizowane algorytmy aproksymacyjne

2 Drzewa filogenetyczne

3 Przybliżone equilibrium Nash'a

4 Problemy lokalizacji

5 Drzewa Steinera

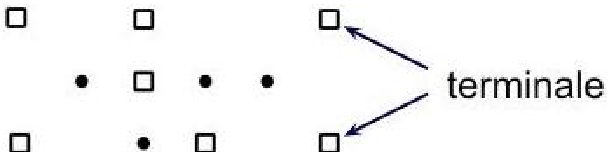
Definicja problemu

Dane:

- nieskierowany graf $G = (V, E)$
- koszt krawędzi $c : E \to Q_+$
- podzbiór wierzchołków $R \subseteq V$ zwanych terminalami

Znajdź: najtańszy podzbiór krawędzi, który łączy wszystkie terminale

$$opt := \min\{c(T) \mid T \text{ łączy } R\}$$



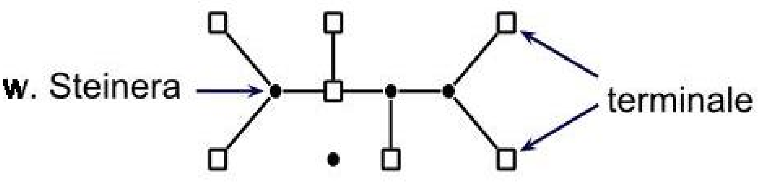
Definicja problemu

Dane:

- nieskierowany graf $G = (V, E)$
- koszt krawędzi $c : E \to Q_+$
- podzbiór wierzchołków $R \subseteq V$ zwanych terminalami

Znajdź: najtańszy podzbiór krawędzi, który łączy wszystkie terminale

$$opt := \min\{c(T) \mid T \text{ łączy } R\}$$



Wcześniejsze wyniki:

Trudność:

- **NP**-trudny nawet gdy koszty krawędzi $\in \{1, 2\}$ [Bern & Plassmann '89]
- nie aproksymowalny poniżej $\frac{96}{95}$[Chlebik & Chlebikova '02]

Algorytmy:

- 2-apr. (*minimalne drzewo spinające terminale*)
- 1.83-apr. [Zelikovsky '93]
- 1.667-apr. [Prömel & Steger '97]
- 1.644-apr. [Karpinski & Zelikovsky '97]
- 1.598-apr. [Hougardy & Prömel '99]
- 1.55-apr. [Robins & Zelikovsky '00]
- PTAS dla R^d (stałe d) [Arora '97]
- PTAS dla grafów planarnych [Borradaile et al. '07]

Wcześniejsze wyniki:

Trudność:

- **NP**-trudny nawet gdy koszty krawędzi $\in \{1, 2\}$ [Bern & Plassmann '89]
- nie aproksymowalny poniżej $\frac{96}{95}$[Chlebik & Chlebikova '02]

Algorytmy:

- 2-apr. (*minimalne drzewo spinające terminale*)
- 1.83-apr. [Zelikovsky '93]
- 1.667-apr. [Prömel & Steger '97]
- 1.644-apr. [Karpinski & Zelikovsky '97]
- 1.598-apr. [Hougardy & Prömel '99]
- 1.55-apr. [Robins & Zelikovsky '00]
- PTAS dla R^d (stałe d) [Arora '97]
- PTAS dla grafów planarnych [Borradaile et al. '07]

W pracy z F. Grandoni, T. Rothvoss i L. Sanita pokazaliśmy:

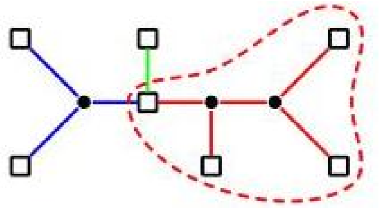
Twierdzenie

Istnieje algorytm 1.39-approksymacyjny dla problemu minimalnego drzewa Steinera.

Twierdzenie

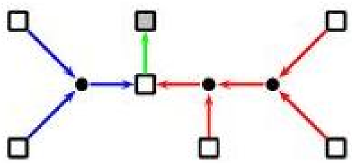
Istnieje proglam liniowy, którego oszacowanie na koszt optymalnego drzewa Steinera mija się z prawdą co najwyżej 1.55 razy.

Komponenty



Po usunięciu terminali z drzewa Steinera, zbiór jego krawędzi zostaje podzielony na komponenty (zaznaczone innymi kolorami).

Skierowane komponenty



- Wybierając jeden z terminali jako korzeń, jednoznacznie kierujemy krawędzie w strone do korzenia.
- Każdy komponent ma dokładnie jedno ujście.
- W powstałym skierowanym grafie z każdego terminala można przesłac jednostkowy przepływ do korzenia

Directed component cut relaxation

- Stwórz listę $C_1, \ldots, C_h$ możliwych komponentów zawierających co najwyżej k terminali.
- Dla każdego z nich policz koszt optymalnego drzewa Steinera na tych terminalach $c(C_j)$.

$$
\begin{aligned}
opt_k^f := \min \sum_j c(C_j) x_j & & & \qquad (k\text{-DCR}) \\
\sum_{\substack{j\,:\,\mathrm{sources}(C_j) \cap S \neq \emptyset, \\ \mathrm{sink}(C_j) \notin S}} x_j & \geqslant & 1 & \quad \forall \emptyset \subset S \subseteq R \setminus \{r\} \\
x_j & \geqslant & 0 & \quad \forall j = 1, \ldots, h.
\end{aligned}
$$

Algorytm

(1) FOR $t = 1, \ldots, \mu$ DO

- (2) Rozwiąż k-DCR $\rightarrow x^t$
- (3) Wylosuj komponent C^t z rozkładem x^t i kontraktuj go.

(4) Znajdź minimalne drzewo spinające na terminalach T^μ w pozostałym grafie.

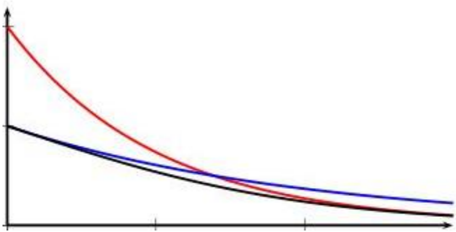
(5) Zwróć $T^\mu \cup \bigcup_{t=1}^{\mu} C^t$.

- Losowanie $C^t \in \{C_1, \ldots, C_h\}$:

  Wybierz C_j z prawdopodobieństwem $\dfrac{x_j^t}{\sum_j x_j^t}$

Analiza

Ograniczamy oczekiwany koszt rozwiązania w grafie po δ/Σ iteracjach.

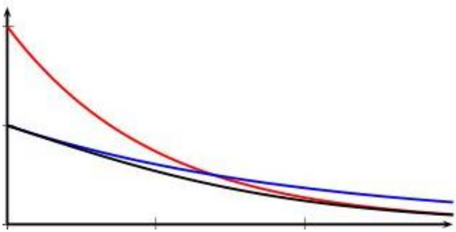


- $2e^{-\delta} \cdot opt_k^f$: oszacowanie na koszt drzewa spinającego terminale
- $e^{-\frac{\delta}{2}} \cdot opt_k$: średnio trudne oszacowanie na koszt drzewa Steinera
- $\approx (\frac{2}{e^{\delta}+1}) \cdot opt_k$: trudniejsze osz. na koszt drzewa Steinera.

O + O → $(1.5 + \epsilon)$-apr. O → $(\ln(4) + \epsilon)$-apr.

Analiza

Ograniczamy oczekiwany koszt rozwiązania w grafie po δ/Σ iteracjach.



- $2e^{-\delta} \cdot opt_k^f$: oszacowanie na koszt drzewa spinającego terminale
- $e^{-\frac{\delta}{2}} \cdot opt_k$: średnio trudne oszacowanie na koszt drzewa Steinera
- $\approx (\frac{2}{e^\delta + 1}) \cdot opt_k$: trudniejsze osz. na koszt drzewa Steinera.

O + O $\rightarrow$ $(1.5 + \epsilon)$-apr.

O $\rightarrow$ $(\ln(4) + \epsilon)$-apr.

Wyzwania:

1. Ograniczyć koszt względem opymalnego rozwiązania LP.
2. Znaleźć inne zastosowanie dla *iterative randomized rounding*.

Dziękuję za uwagę!

Wyzwania:

1. Ograniczyć koszt względem opymalnego rozwiązania LP.
2. Znaleźć inne zastosowanie dla *iterative randomized rounding*.

Dziękuję za uwagę!