Algorithmen für schwierige Probleme

Britta Dorn

Wintersemester 2011/12

30. November 2011

Wiederholung

Baumzerlegung

$G = (V, E)$

Eine Baumzerlegung von G ist ein Paar $\langle \{X_i \mid i \in V_T\}, T \rangle$, wobei T Baum mit Knotenmenge V_T, $X_i \subseteq V$ Bags oder Taschen, so dass gilt:

1. Jeder Knoten von G in mindestens einem Bag
2. Endknoten jeder Kante zusammen in mindestens einem Bag
3. Für jeden Knoten v bilden Taschen, die v enthalten, einen Unterbaum.

Weite einer Zerlegung: Maximale Anzahl Knoten in einem Bag - 1

Baumweite eines Graphen: Weite der besten Zerlegung (so klein wie möglich)

Test

Test

Aufgabe 1

Die Baumweite eines Graphen

1. ist $\leq$ der Weite einer Zerlegung
2. ist $\leq n$ (Anzahl der Knoten des Graphen)
3. kann nie größer als eine Konstante werden
4. kann nie kleiner als 1 werden
5. ist 1, wenn der Graph ein zusammenhängender Baum mit mindestens 2 Knoten ist
6. kann auch für Bäume 2 oder größer sein
7. ist einfach zu bestimmen
8. ist für bestimmte Arten von Graphen einfach zu bestimmen (Bäume, Gitter, Räder, ...)

Lösung

Test

Aufgabe 1

Die Baumweite eines Graphen

1. ist $\leq$ der Weite einer Zerlegung
2. ist $\leq n$ (Anzahl der Knoten des Graphen)
3. kann nie größer als eine Konstante werden
4. kann nie kleiner als 1 werden
5. ist 1, wenn der Graph ein zusammenhängender Baum mit mindestens 2 Knoten ist
6. kann auch für Bäume 2 oder größer sein
7. ist einfach zu bestimmen
8. ist für bestimmte Arten von Graphen einfach zu bestimmen (Bäume, Gitter, Räder, ...)

Test

Aufgabe 2

1. Eine Baumzerlegung eines Graphen ist immer eindeutig
2. Aus einer Baumzerlegung kann man den ursprünglichen Graphen eindeutig konstruieren
3. Bei einer Baumzerlegung müssen in jedem Bag die gleiche Anzahl Knoten sein
4. Bei einer Baumzerlegung für einen zusammenhängenden Graphen kann nie ein Bag nur einen Knoten enthalten
5. Bei einer Baumzerlegung für einen zusammenhängenden Graphen mit mindestens 2 Knoten können nie alle Bags nur einen Knoten enthalten

Lösung

Test

Aufgabe 2

1. Eine Baumzerlegung eines Graphen ist immer eindeutig
2. Aus einer Baumzerlegung kann man den ursprünglichen Graphen eindeutig konstruieren
3. Bei einer Baumzerlegung müssen in jedem Bag die gleiche Anzahl Knoten sein
4. Bei einer Baumzerlegung für einen zusammenhängenden Graphen kann nie ein Bag nur einen Knoten enthalten
5. Bei einer Baumzerlegung für einen zusammenhängenden Graphen mit mindestens 2 Knoten können nie alle Bags nur einen Knoten enthalten

Baumzerlegungen

Typischer Ansatz für FPT-Algorithmen

1. Ermittle Baumzerlegung des gegebenen Graphen (wie? später)
2. Wende Polynomzeit-Algorithmus an, der auf Baumzerlegung beruht (meistens Dynamisches Programmieren)

Laufzeit für das Dynamische Programmieren wächst exponentiell mit der Baumweite!

Baumzerlegung für Vertex Cover

Korrektheit des Algorithmus

(a) Bedingung 1 der Definition 8.2 einer Baumzerlegung ($V = \bigcup_{i \in V_T} X_i$) stellt sicher, dass jeder Knoten in der Berechnung berücksichtigt wurde.

(b) Bedingung 2 (die beiden Endknoten einer Kante müssen zusammen in einem Bag auftauchen) stellt sicher, dass nach der Entfernung der ungültigen Färbungen im Verlauf der Berechnung nur noch tatsächliche Vertex Cover berücksichtigt werden.

Korrektheit des Algorithmus

(c) Bedingung 3 einer Baumzerlegung (Unterbaum...) garantiert die Konsistenz des Dynamischen Programmierens:

Tritt ein Knoten $v \in V$ in zwei verschiedenen Bags X_{i_1} und X_{i_2} auf, so ist ausgeschlossen, dass für das errechnete optimale VC dieser Knoten in den Tabellen A_{i_1} bzw. A_{i_2} zwei unterschiedliche Farben zugewiesen bekommt. klar: Wenn die beiden Bags Vater/Sohn sind, dann kann das nicht passieren. Der Algo färbt Knoten aus dem "Schnitt" gleich und ändert dann nur die Erweiterungen der Färbung des gemeinsamen Teils durch. Aber könnte es ein Problem sein, wenn zwei andere Bags so sind (nicht Vater/Sohn)? nein: Dieser Konflikt hätte sich spätestens im Bag X_{i_0} eines kleinsten gemeinsamen Vorfahren i_0 von i_1 und i_2 in T aufgelöst. Denn laut Bedingung 3 der Baumzerlegung muss v auch in X_{i_0} auftauchen.

Baumzerlegung für Vertex Cover

Laufzeit

Aktualisieren einer Tabelle A_j durch Tabelle A_i (bei geschicktem Umsortieren der Tabellen) in

$$O((\#\text{Zeilen von } A_i + \#\text{Zeilen von } A_j) \cdot |X_j|)$$
$$= O((2^{|X_i|} + 2^{|X_j|}) \cdot |X_j|)$$
$$= O(2^w \cdot w) \text{ (} w \text{ war Weite der Baumzerlegung, d.h. } |X_i|, |X_j| \leq w\text{)}$$

Für jede Kante $e \in E_T$ im Baum T muss ein Abgleich zweier Tabellen stattfinden, erhalte insgesamt $O(2^w \cdot w \cdot |E_T|) = O(2^w \cdot w \cdot |V_T|)$.