

Algorithmen für schwierige Probleme

Britta Dorn

Wintersemester 2011/12

26. Oktober 2011

Test

Aufgabe 1

Aufgabe 1

Ein Graphproblem \mathcal{P} kann in Zeit $O(2.27^k \cdot n^2)$ gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist) Das heisst:

Aufgabe 1

Ein Graphproblem \mathcal{P} kann in Zeit $O(2.27^k \cdot n^2)$ gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist) Das heisst:

- 1 \mathcal{P} ist fixed-parameter tractable (bezüglich k)

Aufgabe 1

Ein Graphproblem \mathcal{P} kann in Zeit $O(2.27^k \cdot n^2)$ gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist) Das heisst:

- 1 \mathcal{P} ist fixed-parameter tractable (bezüglich k)
- 2 \mathcal{P} liegt in der Klasse FPT

Aufgabe 1

Ein Graphproblem \mathcal{P} kann in Zeit $O(2.27^k \cdot n^2)$ gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist) Das heisst:

- 1 \mathcal{P} ist fixed-parameter tractable (bezüglich k)
- 2 \mathcal{P} liegt in der Klasse FPT
- 3 Für kleine Werte von k lässt sich \mathcal{P} effizient lösen

Aufgabe 1

Ein Graphproblem \mathcal{P} kann in Zeit $O(2.27^k \cdot n^2)$ gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist) Das heisst:

- 1 \mathcal{P} ist fixed-parameter tractable (bezüglich k)
- 2 \mathcal{P} liegt in der Klasse FPT
- 3 Für kleine Werte von k lässt sich \mathcal{P} effizient lösen
- 4 Es gibt keinen besseren Algorithmus für \mathcal{P}

Lösung

Aufgabe 1

Ein Graphproblem \mathcal{P} kann in Zeit $O(2.27^k \cdot n^2)$ gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist) Das heisst:

- 1 \mathcal{P} ist fixed-parameter tractable (bezüglich k)
- 2 \mathcal{P} liegt in der Klasse FPT
- 3 Für kleine Werte von k lässt sich \mathcal{P} effizient lösen
- 4 Es gibt keinen besseren Algorithmus für \mathcal{P}

Aufgabe 2

Aufgabe 2

Ein Graphproblem Q kann in Zeit $O(n^k)$ durch Ausprobieren aller Möglichkeiten gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist). Dann gilt:

Aufgabe 2

Ein Graphproblem \mathcal{Q} kann in Zeit $O(n^k)$ durch Ausprobieren aller Möglichkeiten gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist). Dann gilt:

- 1 \mathcal{Q} ist fixed-parameter tractable (bezüglich k)

Aufgabe 2

Ein Graphproblem Q kann in Zeit $O(n^k)$ durch Ausprobieren aller Möglichkeiten gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist). Dann gilt:

- 1 Q ist fixed-parameter tractable (bezüglich k)
- 2 Q liegt in der Klasse FPT

Aufgabe 2

Ein Graphproblem Q kann in Zeit $O(n^k)$ durch Ausprobieren aller Möglichkeiten gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist). Dann gilt:

- 1 Q ist fixed-parameter tractable (bezüglich k)
- 2 Q liegt in der Klasse FPT
- 3 Q könnte in der Klasse FPT liegen (dazu muss man aber einen besseren Algorithmus finden)

Aufgabe 2

Ein Graphproblem Q kann in Zeit $O(n^k)$ durch Ausprobieren aller Möglichkeiten gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist). Dann gilt:

- 1 Q ist fixed-parameter tractable (bezüglich k)
- 2 Q liegt in der Klasse FPT
- 3 Q könnte in der Klasse FPT liegen (dazu muss man aber einen besseren Algorithmus finden)
- 4 Es gibt keinen besseren Algorithmus für Q

Lösung

Aufgabe 2

Ein Graphproblem Q kann in Zeit $O(n^k)$ durch Ausprobieren aller Möglichkeiten gelöst werden (wobei n die Anzahl der Knoten und k die Größe der Lösungsmenge des Problems ist). Dann gilt:

- 1 Q ist fixed-parameter tractable (bezüglich k)
- 2 Q liegt in der Klasse FPT
- 3 Q könnte in der Klasse FPT liegen (dazu muss man aber einen besseren Algorithmus finden)
- 4 Es gibt keinen besseren Algorithmus für Q

Aufgabe 3

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse `FPT` (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- 1 MAST kann immer (durch geschickte Algorithmen) in Polynomzeit gelöst werden.

Aufgabe 3

Das Problem **MAXIMUM AGREEMENT SUBTREE (MAST)** ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- 1 MAST kann immer (durch geschickte Algorithmen) in Polynomzeit gelöst werden.
- 2 Es kann sein, dass bestimmte Instanzen von MAST in Polynomzeit gelöst werden können.

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- 1 MAST kann immer (durch geschickte Algorithmen) in Polynomzeit gelöst werden.
- 2 Es kann sein, dass bestimmte Instanzen von MAST in Polynomzeit gelöst werden können.
- 3 MAST kann in einer Laufzeit gelöst werden, in der im Exponenten nur k auftritt.

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- (4) Instanzen von `MAST` mit sehr wenigen Arten, die gelöscht werden müssen, können effizient gelöst werden.

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- (4) Instanzen von MAST mit sehr wenigen Arten, die gelöscht werden müssen, können effizient gelöst werden.
- (5) Instanzen von MAST mit sehr vielen Arten, die gelöscht werden müssen, können nicht gelöst werden.

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- (4) Instanzen von MAST mit sehr wenigen Arten, die gelöscht werden müssen, können effizient gelöst werden.
- (5) Instanzen von MAST mit sehr vielen Arten, die gelöscht werden müssen, können nicht gelöst werden.
- (6) Es gibt einen parametrisierten Algorithmus, der MAST immer schnell löst.

Lösung

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- 1 MAST kann immer (durch geschickte Algorithmen) in Polynomzeit gelöst werden.
- 2 Es kann sein, dass bestimmte Instanzen von MAST in Polynomzeit gelöst werden können.
- 3 MAST kann in einer Laufzeit gelöst werden, in der im Exponenten nur k auftritt.

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- 1 MAST kann immer (durch geschickte Algorithmen) in Polynomzeit gelöst werden.
- 2 Es kann sein, dass bestimmte Instanzen von MAST in Polynomzeit gelöst werden können.
- 3 MAST kann in einer Laufzeit gelöst werden, in der im Exponenten nur k auftritt.

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- (4) Instanzen von MAST mit sehr wenigen Arten, die gelöscht werden müssen, können effizient gelöst werden.

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- (4) Instanzen von `MAST` mit sehr wenigen Arten, die gelöscht werden müssen, können effizient gelöst werden.
- (5) Instanzen von `MAST` mit sehr vielen Arten, die gelöscht werden müssen, können nicht gelöst werden.

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- (4) Instanzen von MAST mit sehr wenigen Arten, die gelöscht werden müssen, können effizient gelöst werden.
- (5) Instanzen von MAST mit sehr vielen Arten, die gelöscht werden müssen, können nicht gelöst werden.
- (6) Es gibt einen parametrisierten Algorithmus, der MAST immer schnell löst.

Aufgabe 3

Das Problem `MAXIMUM AGREEMENT SUBTREE (MAST)` ist NP-vollständig und liegt in der Klasse FPT (Parameter k hier Anzahl der Arten, die gelöscht werden müssen). Das bedeutet:

- (4) Instanzen von `MAST` mit sehr wenigen Arten, die gelöscht werden müssen, können effizient gelöst werden.
- (5) Instanzen von `MAST` mit sehr vielen Arten, die gelöscht werden müssen, können nicht gelöst werden.
- (6) Es gibt einen parametrisierten Algorithmus, der `MAST` immer schnell löst.

Datenreduktion und Problemkerne

