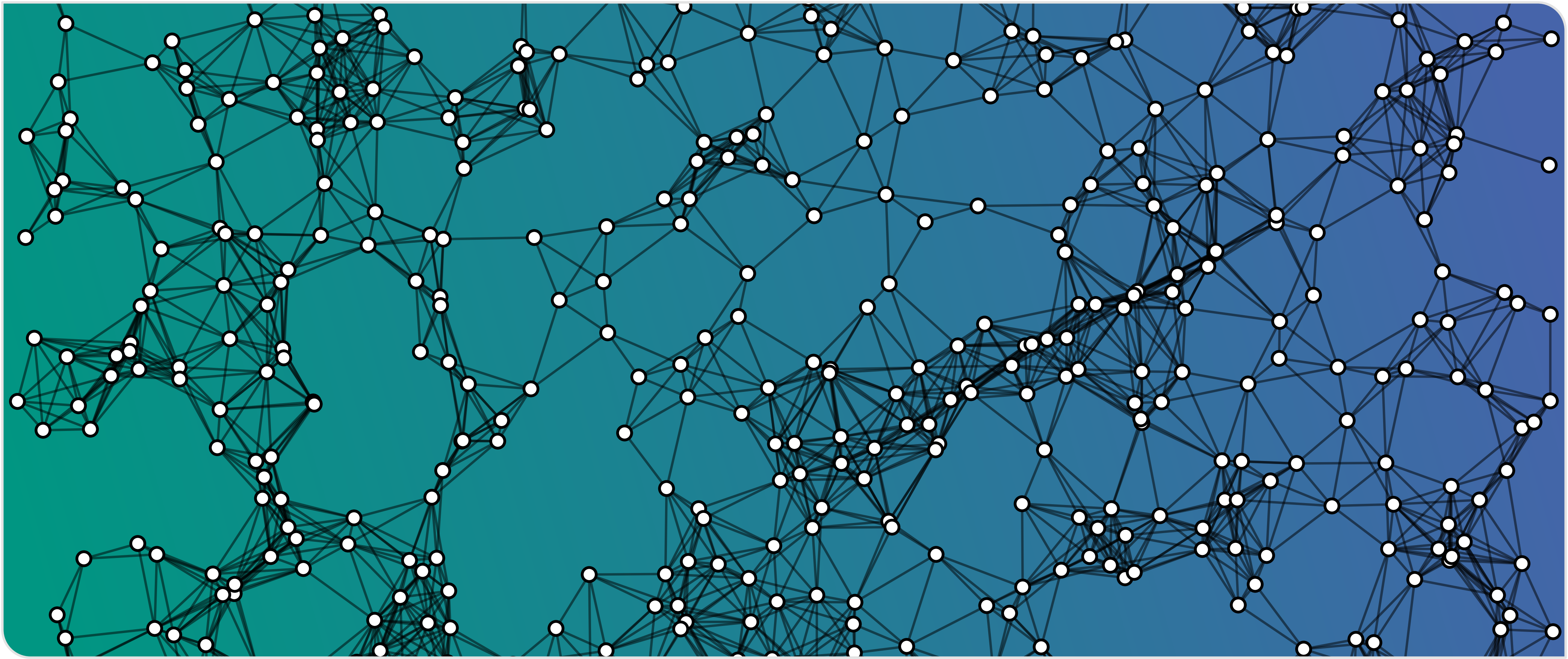
Karlsruher Institut für Technologie

Parametrisierte Algorithmen

Übung 3



Heute

Übungsblatt 2

Übungsblatt 3

Außerdem

- Lineare Programme & Lenstra
  - Aufgaben

Übungsblatt 2

Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Karlsruher Institut für Technologie

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set **10 Punkte**

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set **10 Punkte**

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching **10 Punkte**

Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1

Übungsblatt 2

Karlsruher Institut für Technologie

Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Karlsruher Institut für Technologie

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — **10 Punkte**

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

gut bearbeitet

Aufgabe 2: d-Hitting Set — **10 Punkte**

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — **10 Punkte**

Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1

Übungsblatt 2

Karlsruher Institut für Technologie

Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Karlsruher Institut für Technologie

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

gut bearbeitet

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

easy

Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

challenging

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

hard

1

Übungsblatt 2

Karlsruher Institut für Technologie

Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Karlsruher Institut für Technologie

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

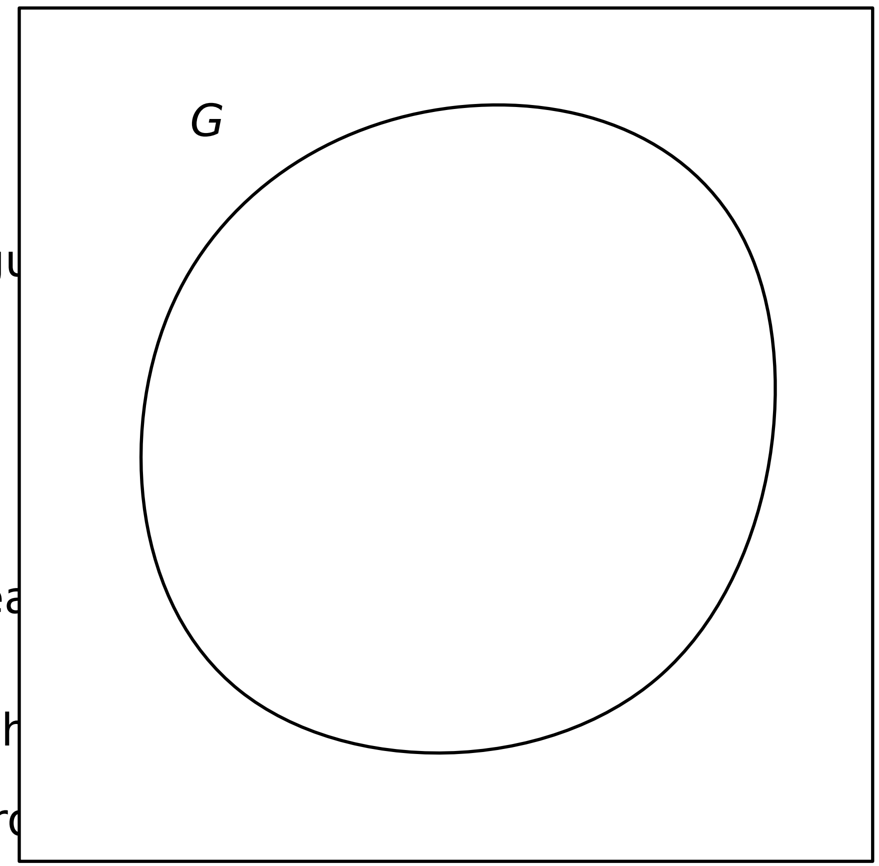
1

gu

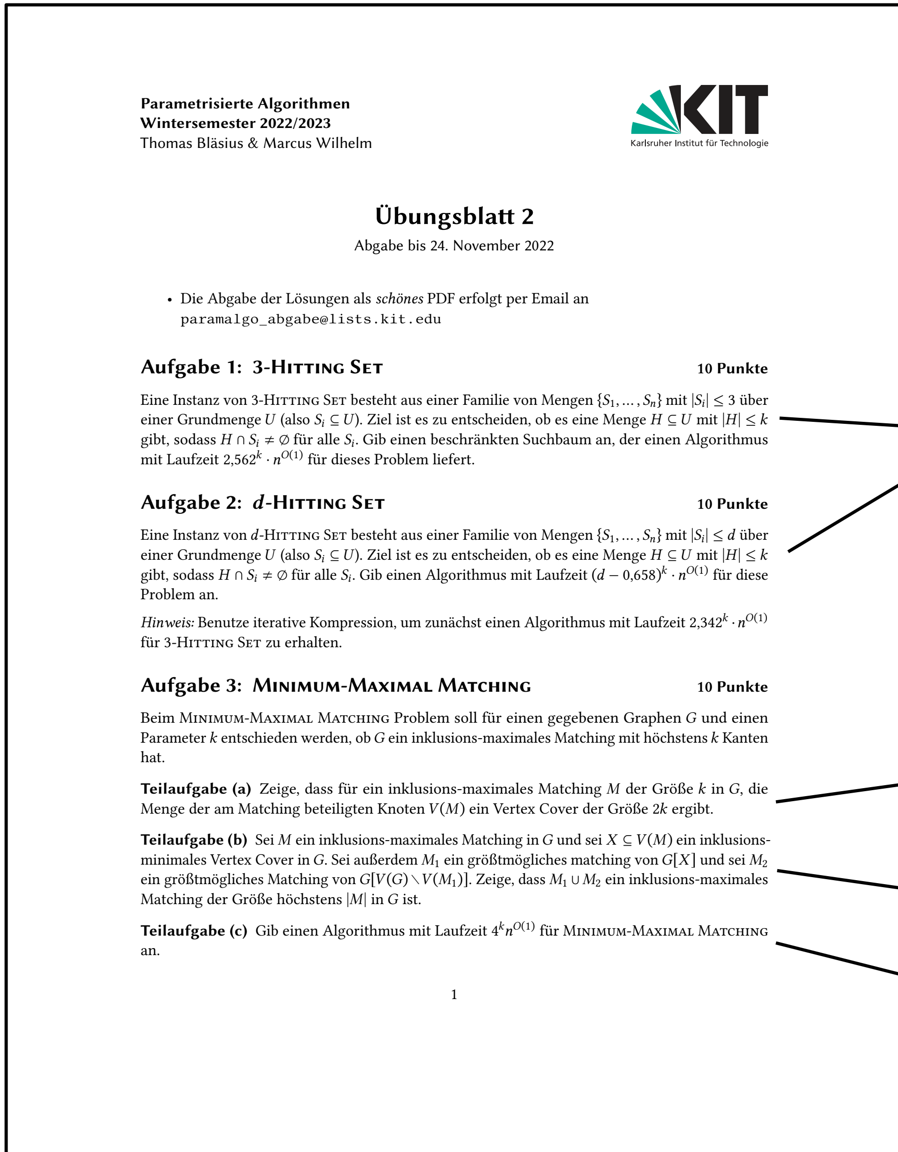
ea

ch

hard



Übungsblatt 2



Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \le 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \le k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \le d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \le k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

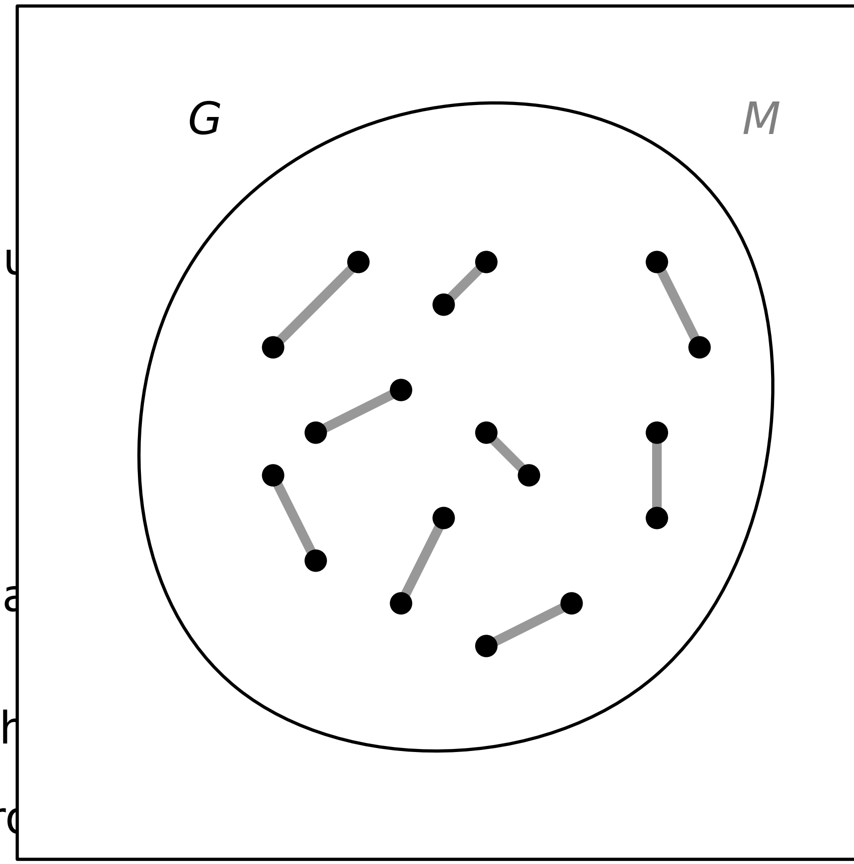
Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1



Übungsblatt 2

Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

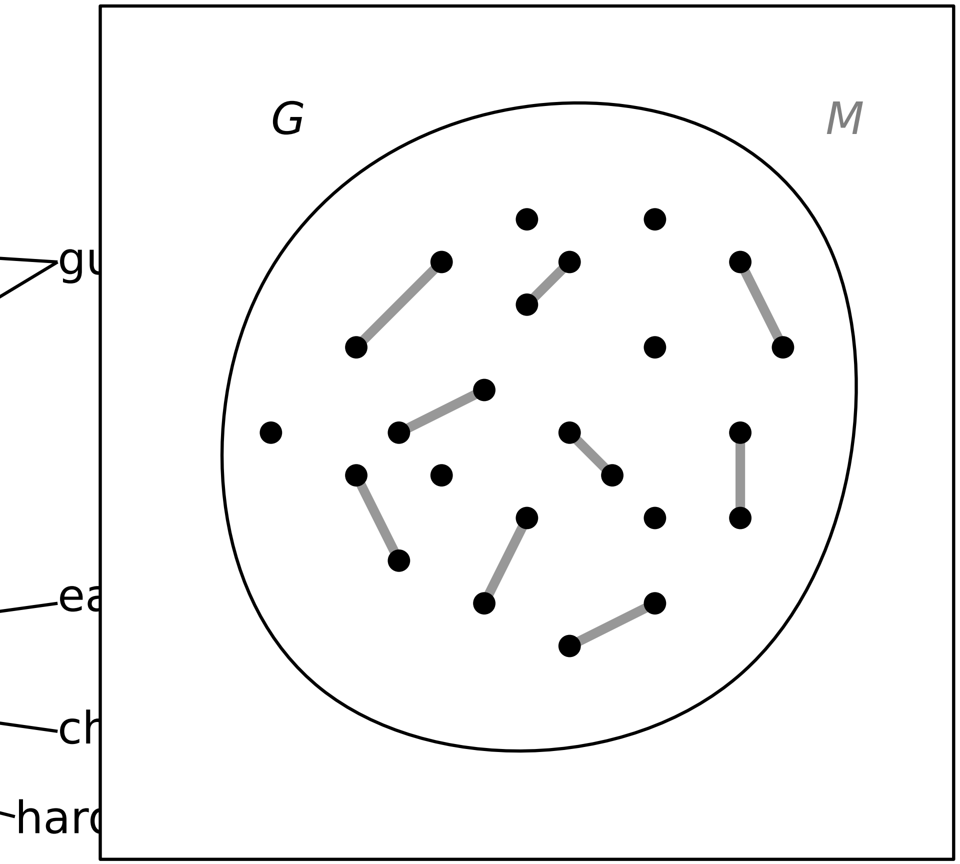
Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

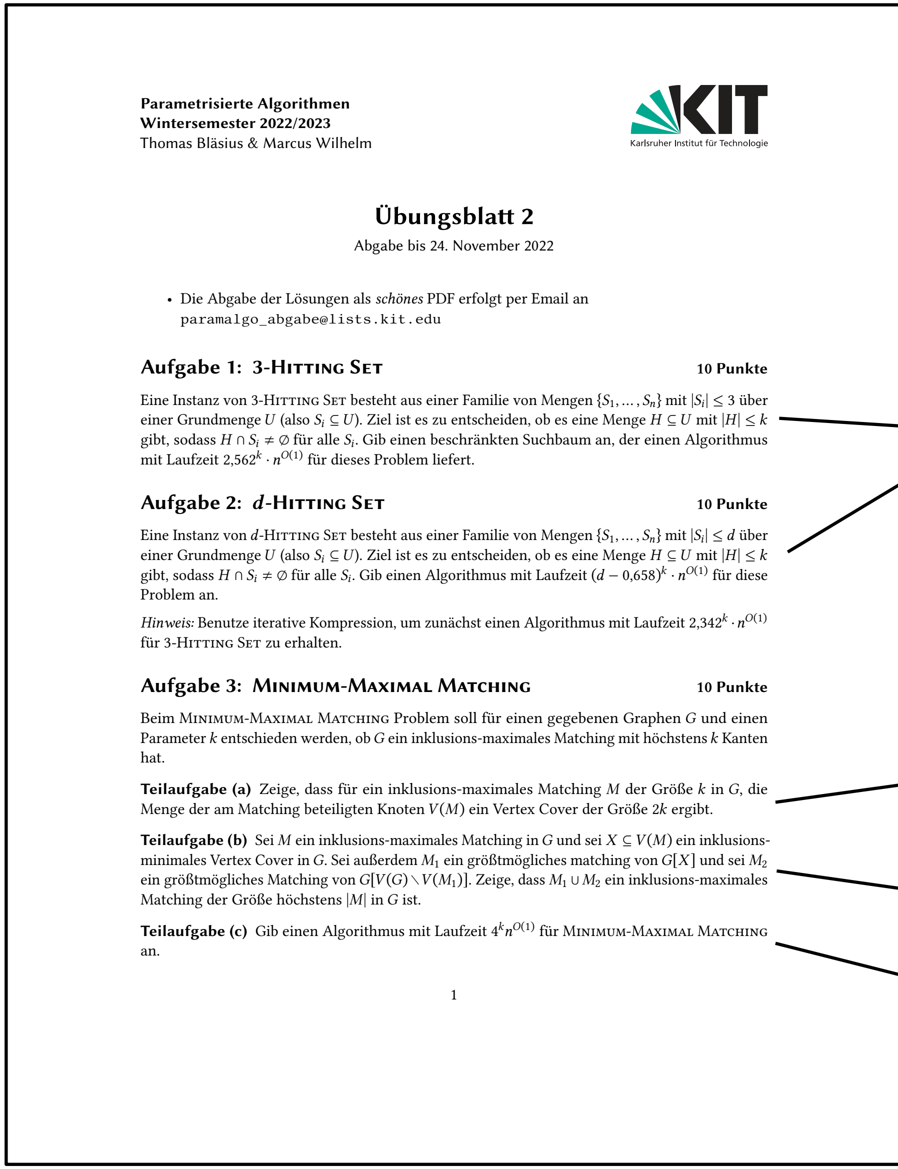
Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1



Übungsblatt 2



Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Karlsruher Institut für Technologie

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

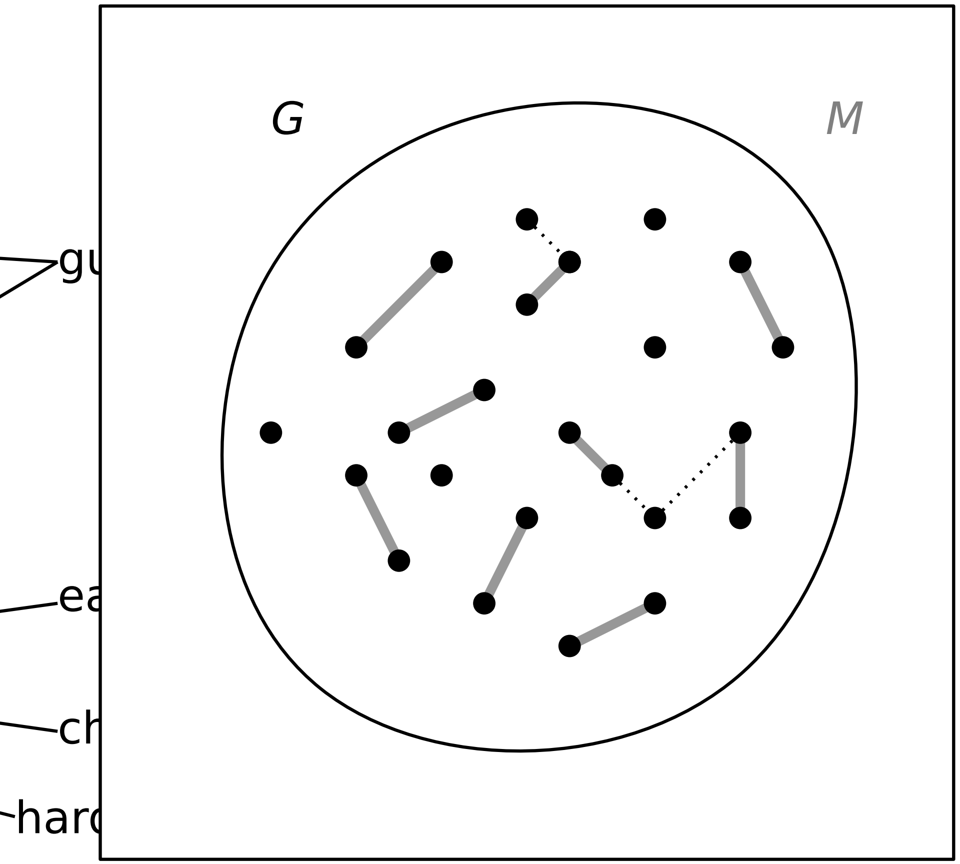
Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

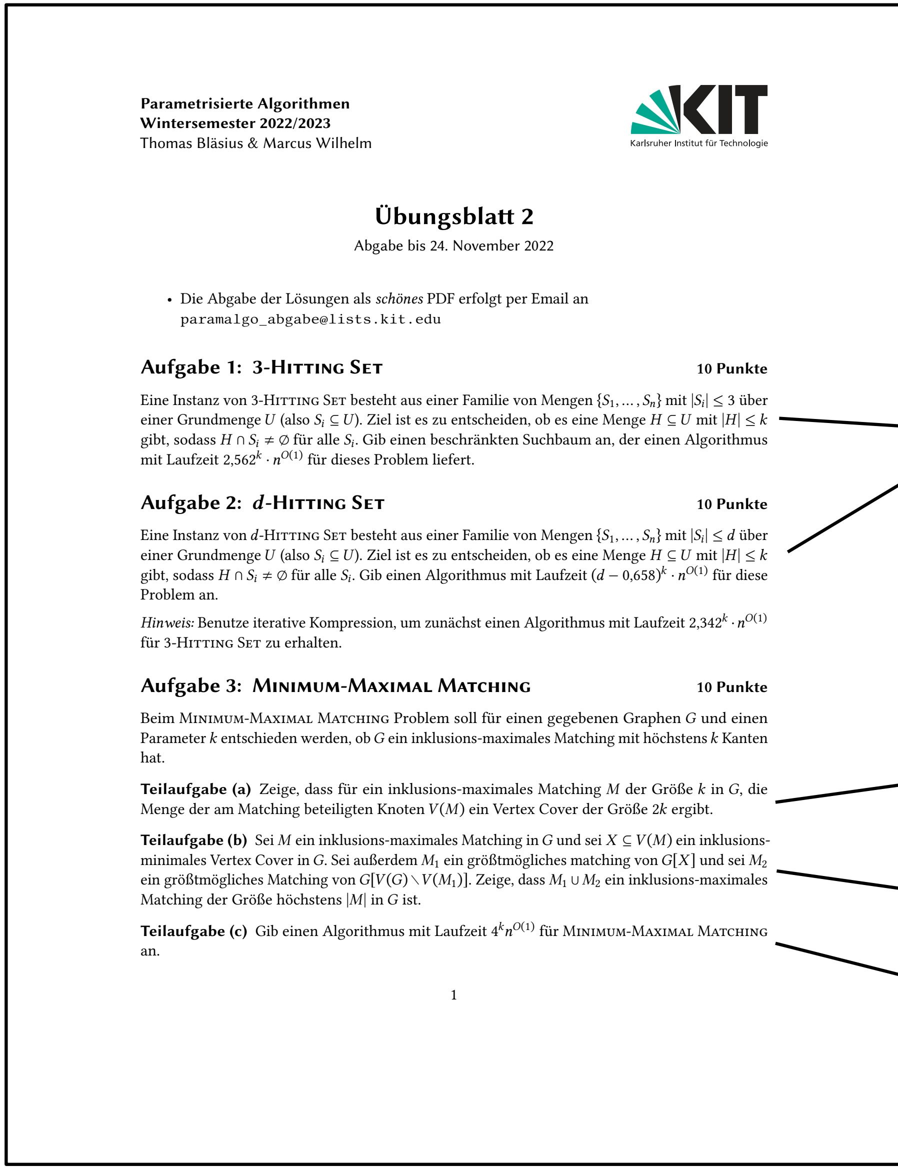
Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1



Übungsblatt 2



Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

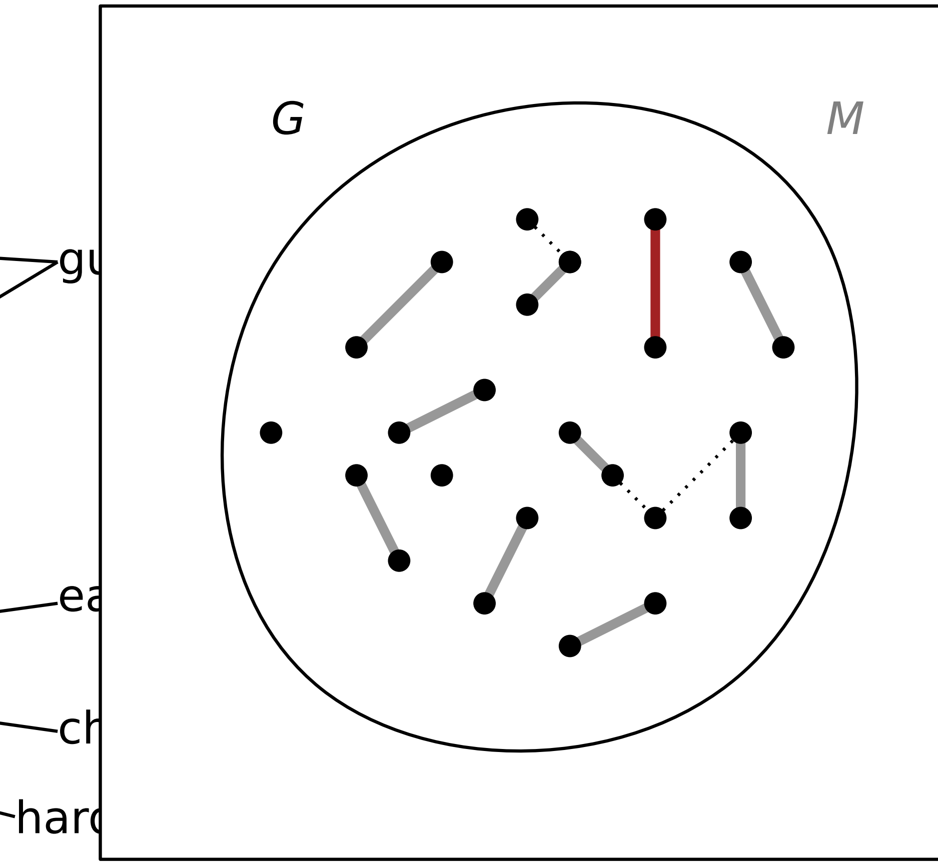
Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

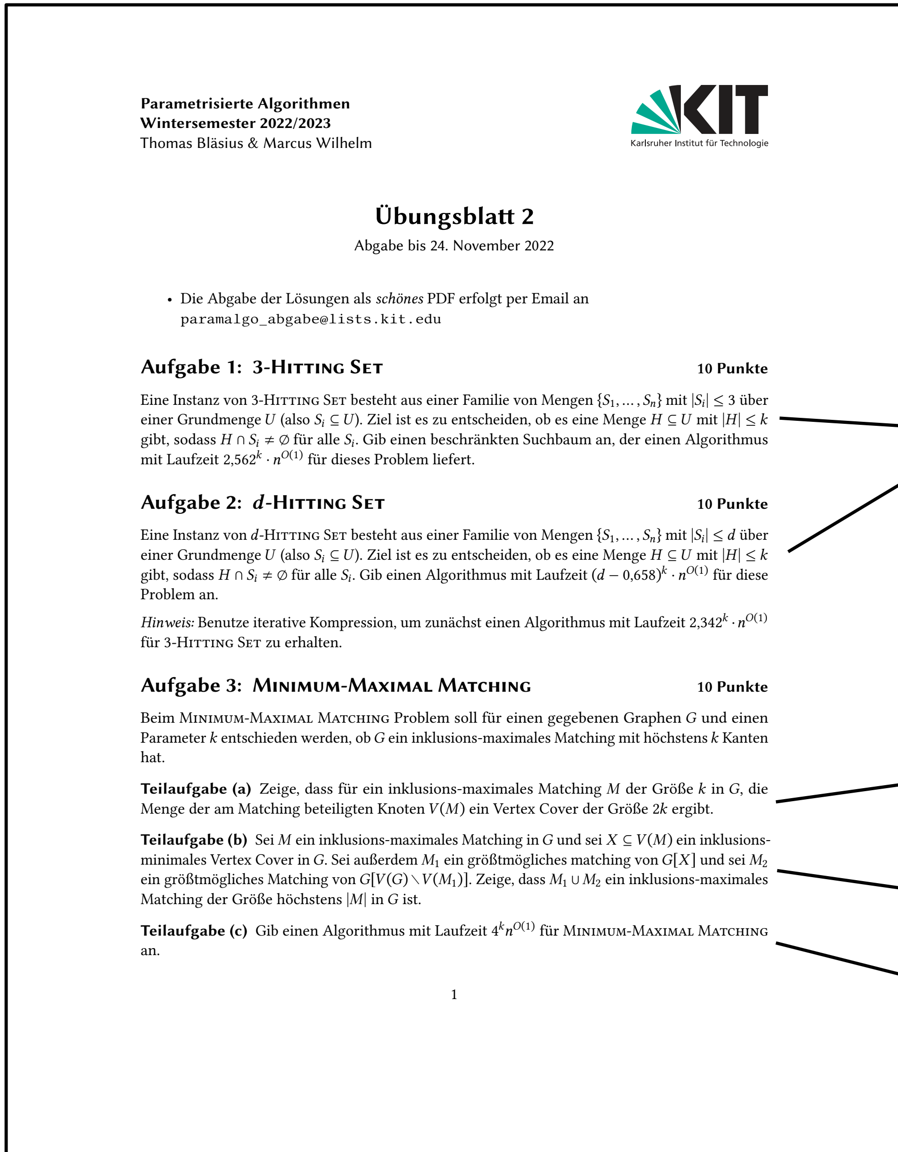
Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1



Übungsblatt 2



Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

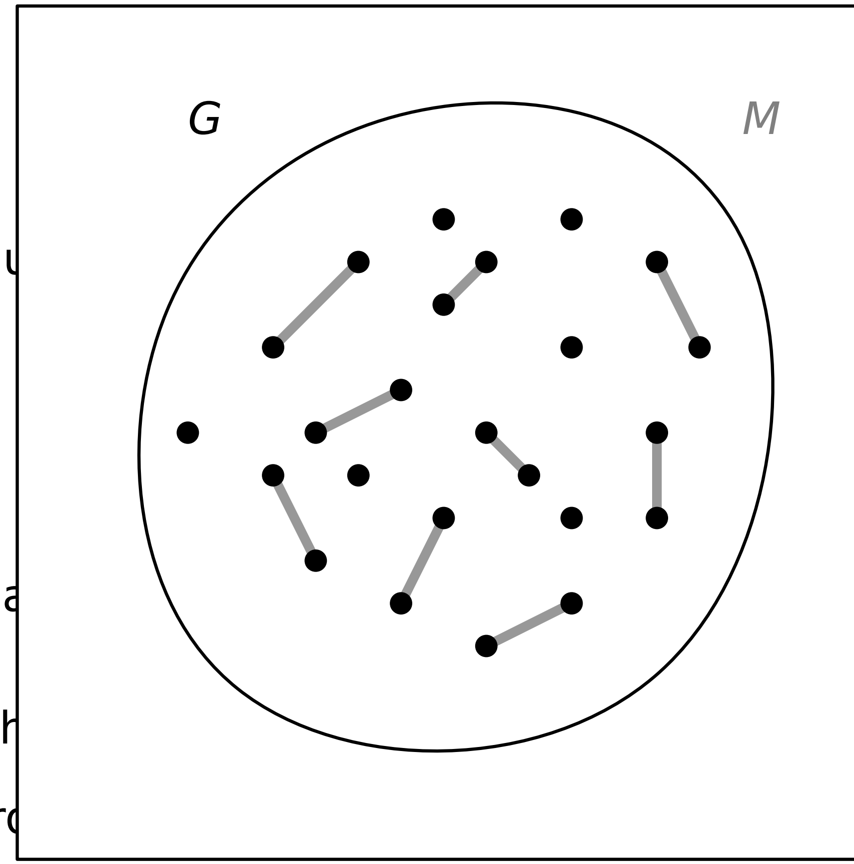
Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1



Übungsblatt 2

Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

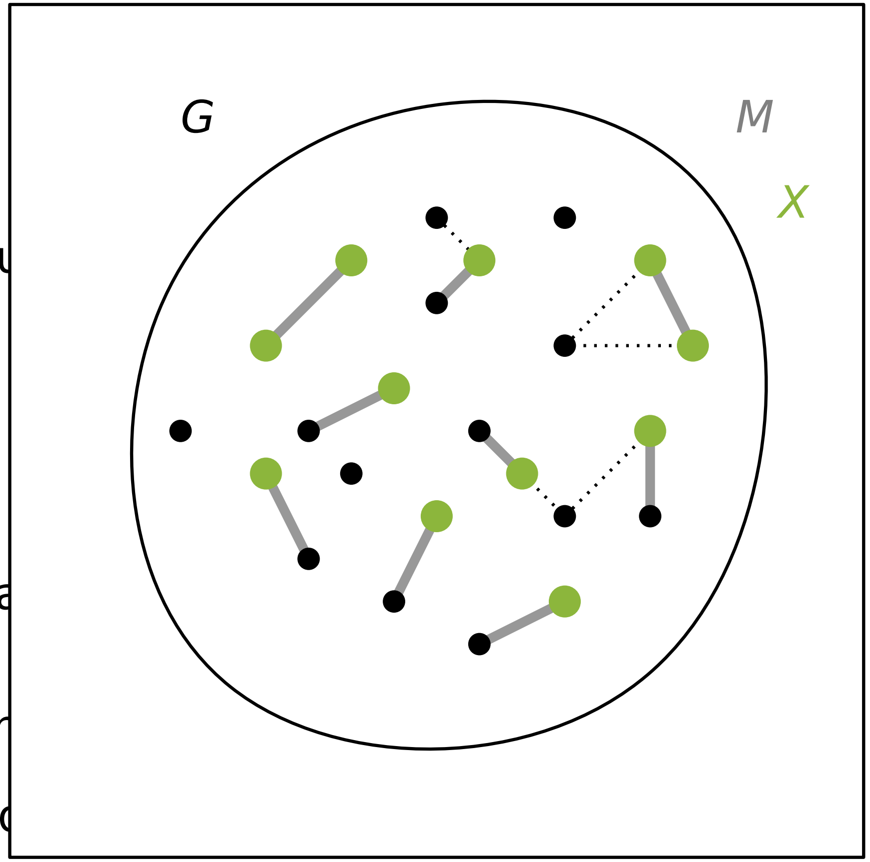
Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

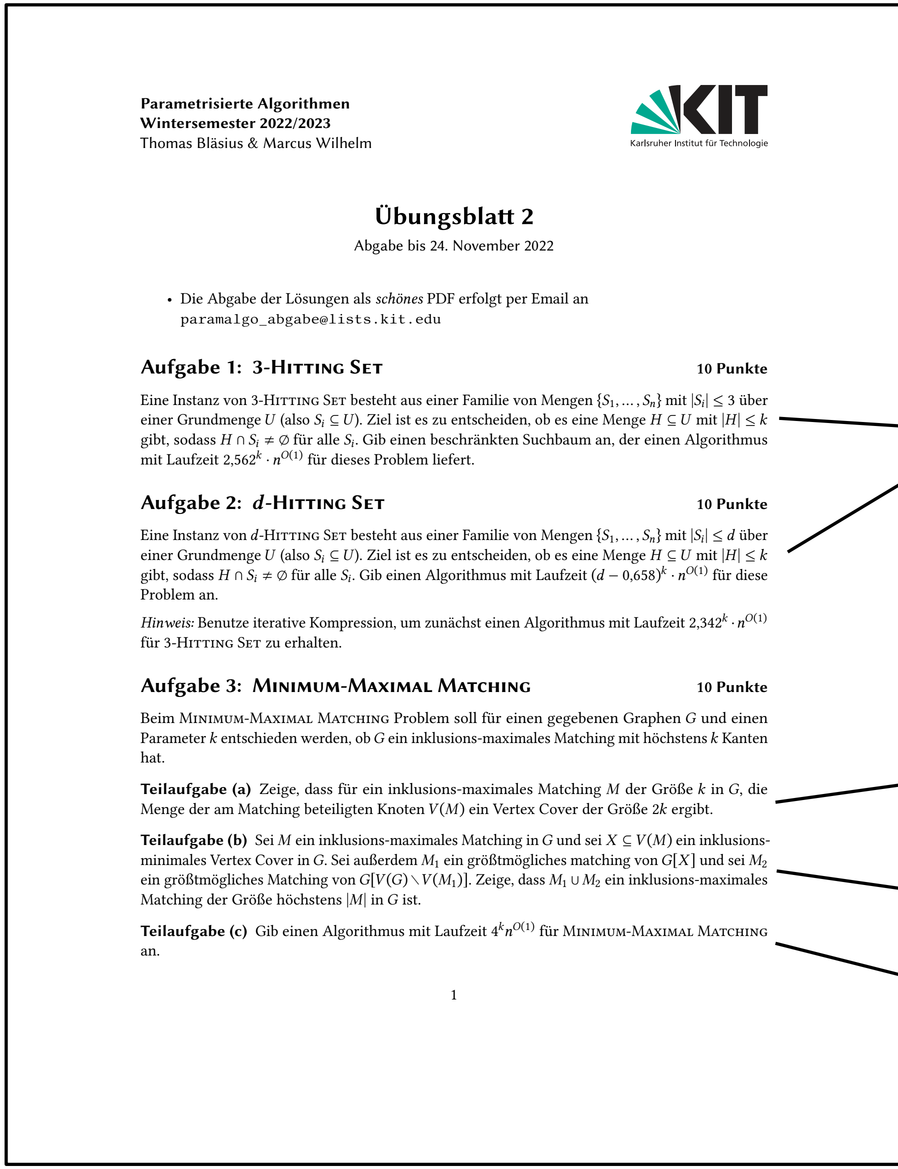
Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1



Übungsblatt 2



Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

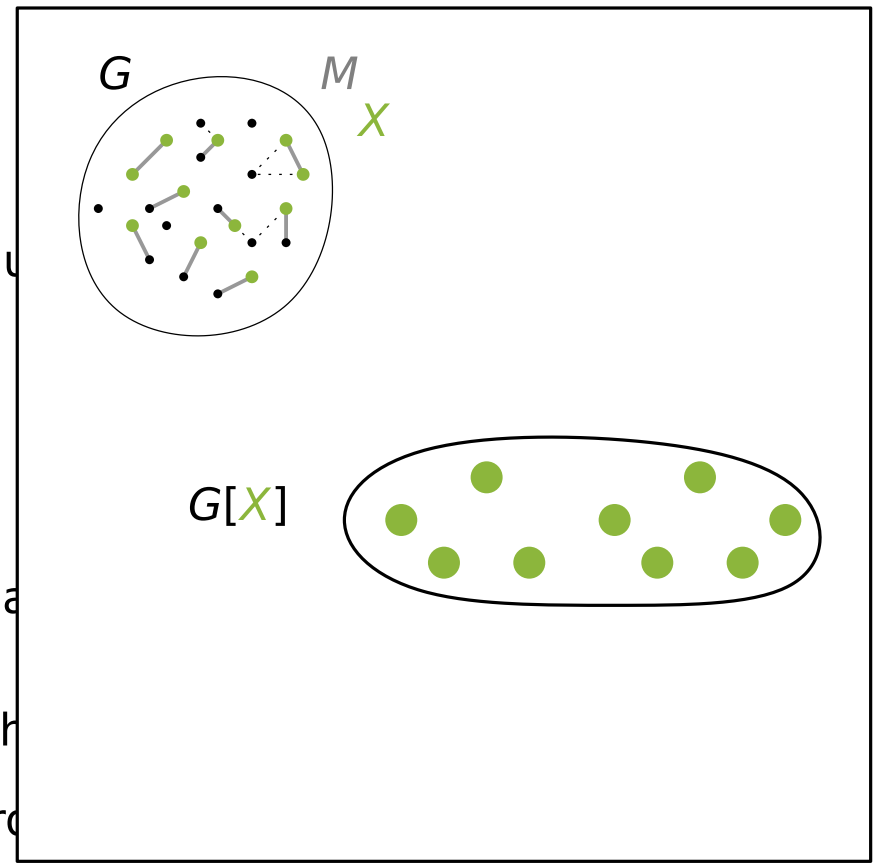
Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

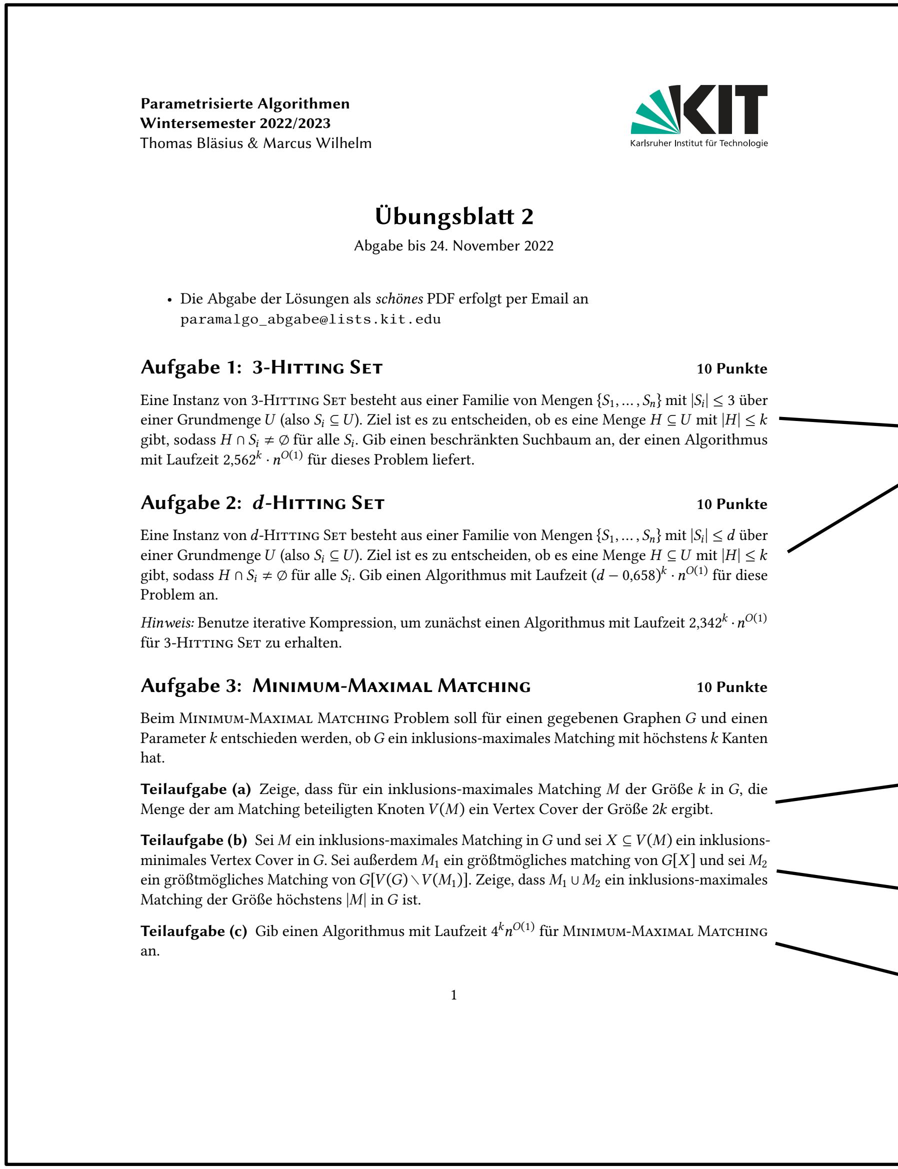
Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1



Übungsblatt 2



Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \le 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \le k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \le d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \le k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches Matching von $G[X]$ und sei M_2 ein größtmögliches matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

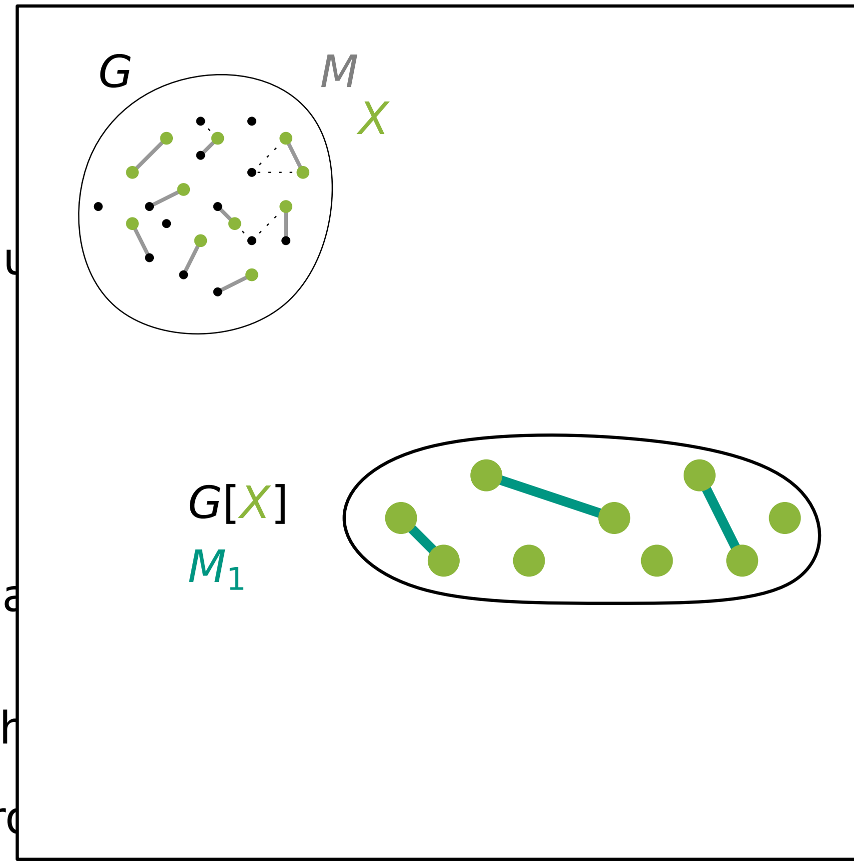
1

gu

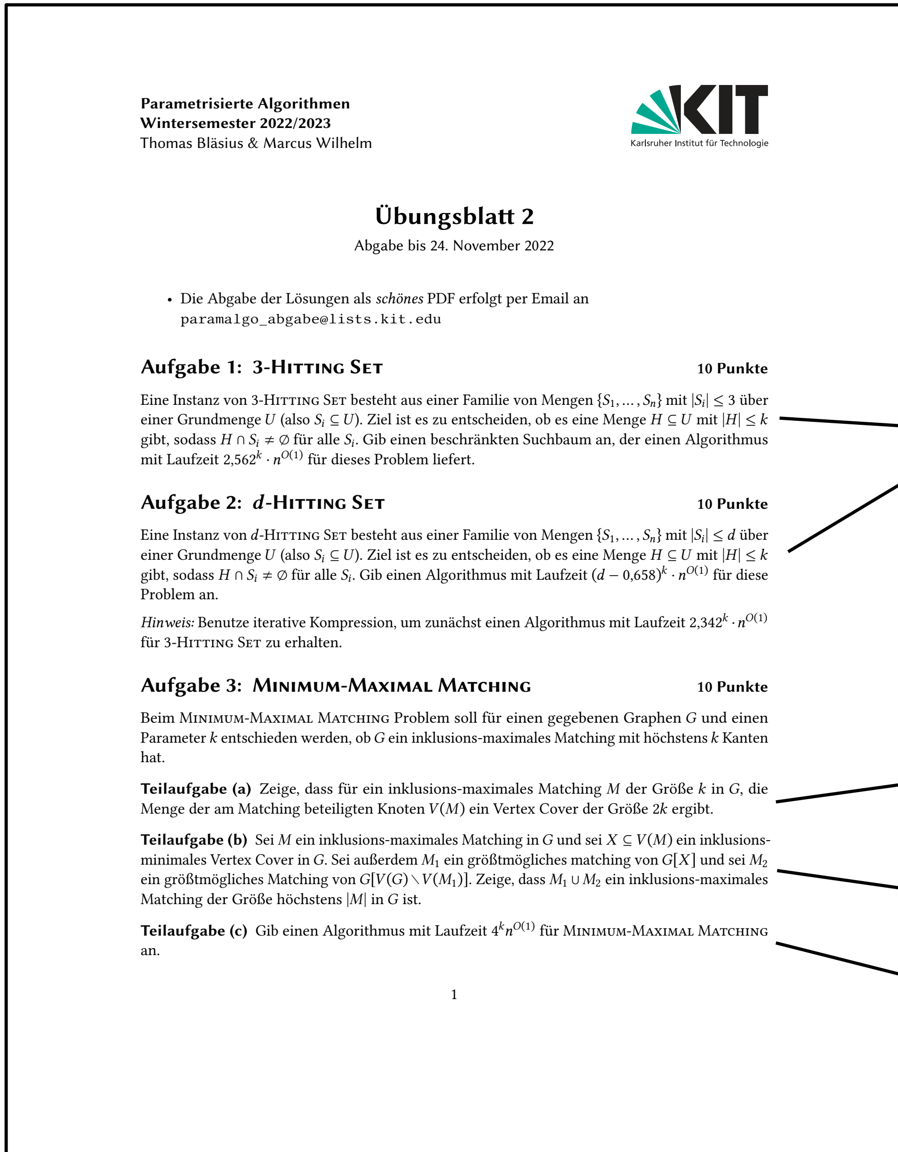
ea

ch

harc



Übungsblatt 2



Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

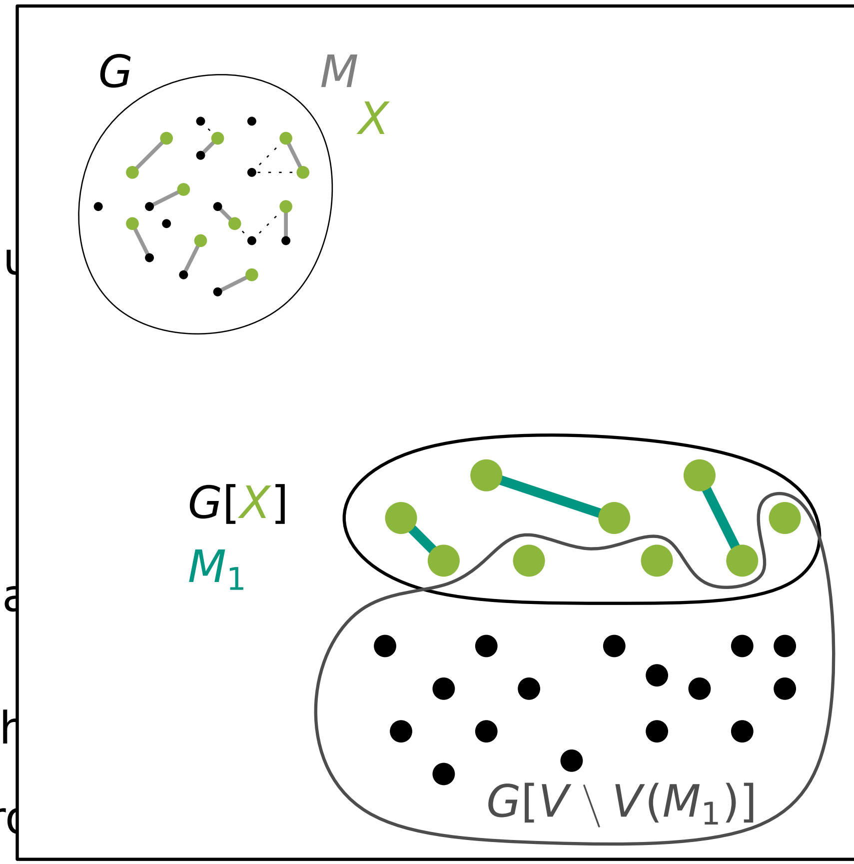
Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

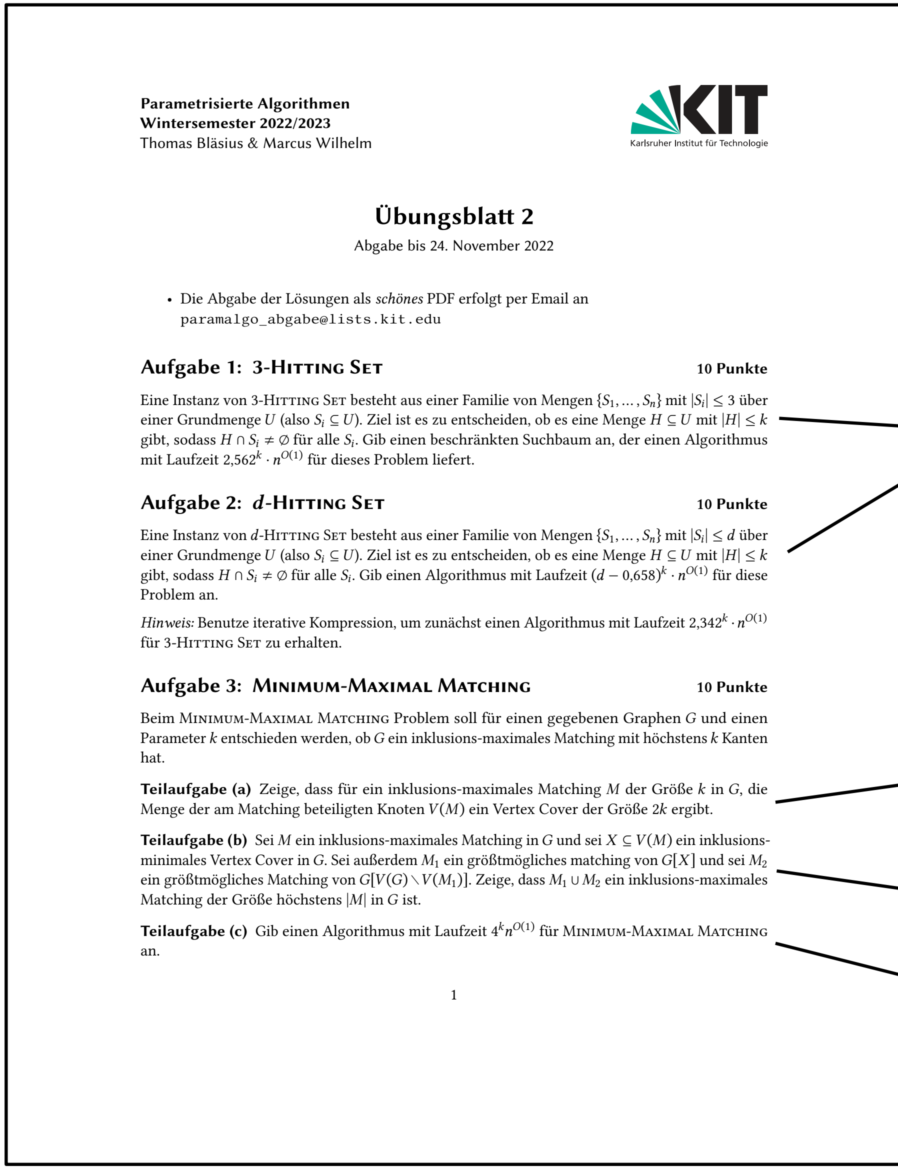
Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1



Übungsblatt 2



Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Karlsruher Institut für Technologie

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — 10 Punkte

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — 10 Punkte

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — 10 Punkte

Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

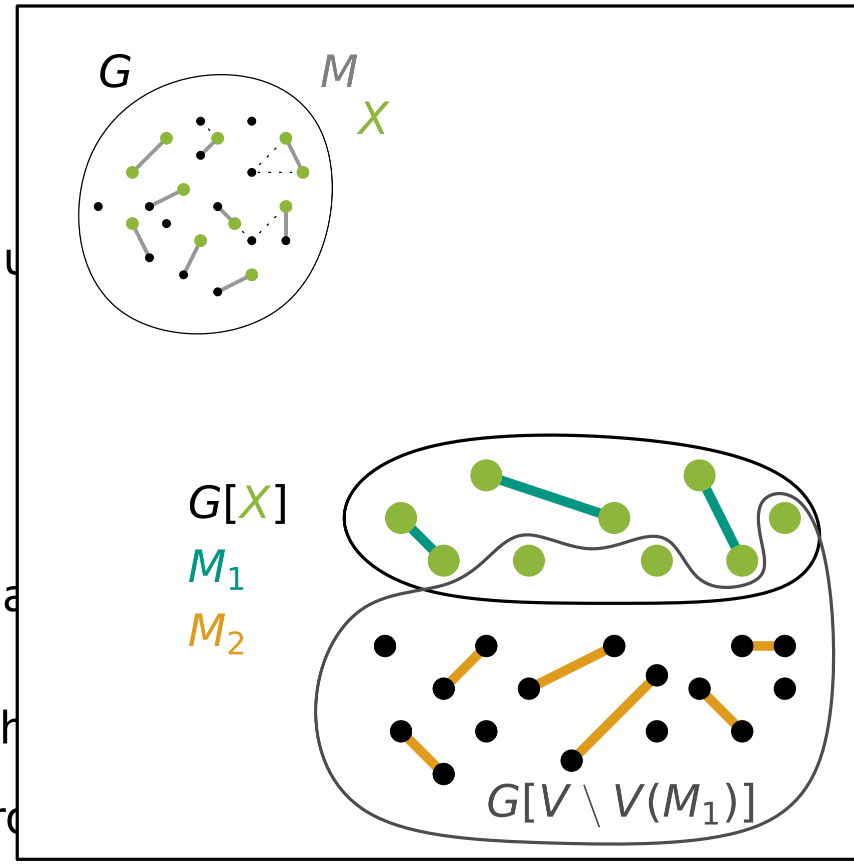
Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1

gu
ea
ch
harc



Übungsblatt 2

Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Übungsblatt 2

Abgabe bis 24. November 2022

- Die Abgabe der Lösungen als *schönes* PDF erfolgt per Email an `paramalgo_abgabe@lists.kit.edu`

Aufgabe 1: 3-Hitting Set — **10 Punkte**

Eine Instanz von 3-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq 3$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen beschränkten Suchbaum an, der einen Algorithmus mit Laufzeit $2{,}562^k \cdot n^{O(1)}$ für dieses Problem liefert.

Aufgabe 2: d-Hitting Set — **10 Punkte**

Eine Instanz von d-Hitting Set besteht aus einer Familie von Mengen $\{S_1, \dots, S_n\}$ mit $|S_i| \leq d$ über einer Grundmenge U (also $S_i \subseteq U$). Ziel ist es zu entscheiden, ob es eine Menge $H \subseteq U$ mit $|H| \leq k$ gibt, sodass $H \cap S_i \neq \emptyset$ für alle S_i. Gib einen Algorithmus mit Laufzeit $(d - 0{,}658)^k \cdot n^{O(1)}$ für diese Problem an.

Hinweis: Benutze iterative Kompression, um zunächst einen Algorithmus mit Laufzeit $2{,}342^k \cdot n^{O(1)}$ für 3-Hitting Set zu erhalten.

Aufgabe 3: Minimum-Maximal Matching — **10 Punkte**

Beim Minimum-Maximal Matching Problem soll für einen gegebenen Graphen G und einen Parameter k entschieden werden, ob G ein inklusions-maximales Matching mit höchstens k Kanten hat.

Teilaufgabe (a) Zeige, dass für ein inklusions-maximales Matching M der Größe k in G, die Menge der am Matching beteiligten Knoten $V(M)$ ein Vertex Cover der Größe $2k$ ergibt.

Teilaufgabe (b) Sei M ein inklusions-maximales Matching in G und sei $X \subseteq V(M)$ ein inklusions-minimales Vertex Cover in G. Sei außerdem M_1 ein größtmögliches matching von $G[X]$ und sei M_2 ein größtmögliches Matching von $G[V(G) \setminus V(M_1)]$. Zeige, dass $M_1 \cup M_2$ ein inklusions-maximales Matching der Größe höchstens $|M|$ in G ist.

Teilaufgabe (c) Gib einen Algorithmus mit Laufzeit $4^k n^{O(1)}$ für Minimum-Maximal Matching an.

1

Wichtigste Erkenntnis:
Bei Suchbaum Algo
kann man auch
Lösungen aufzählen

Übungsblatt 3

Parametrisierte Algorithmen
Wintersemester 2022/2023
Thomas Bläsius & Marcus Wilhelm

Karlsruher Institut für Technologie

Übungsblatt 3

Abgabe bis 8. Dezember 2022

Aufgabe 1: Kontaktvermeidung — 8 Punkte

Ein Virus geht um und muss dringend an der Ausbreitung gehindert werden. Laut Modellrechnungen kann dies bewerkstelligt werden, indem sich jeder nur noch mit (höchstens) d anderen Leuten trifft. Zum Glück können bis zu k Personen geimpft werden, was dazu führt, dass deren Kontakte sofort ignoriert werden können.

Es muss nun also entschieden werden, ob es möglich ist k Knoten aus einem gegebenen Graphen zu löschen, so dass anschließend der Maximalgrad höchstens d ist.

Gib sichere Reduktionsregeln an, die für dieses Problem einen Kern mit Größe polynomiell in $d + k$ berechnen.

Aufgabe 2: Edge Clique Cover — 10 Punkte

Das parametrisierte Problem Edge Clique Cover ist wie folgt definiert. Gegeben ist ein Graph G sowie ein Parameter k. Gesucht ist eine Menge von maximal k Cliquen, sodass jede Kante von G in mindestens einer der Cliquen enthalten ist.

Gib sichere Reduktionsregeln an, die für Edge Clique Cover einen Kern mit maximal 2^k Knoten berechnen.

Hinweis: Zwei Knoten u und v sind *echte Zwillinge*, wenn $N(u) \cup \{u\} = N(v) \cup \{v\}$ (dabei bezeichnet $N(u)$ die Menge aller Nachbarn von u). Kannst du solche echten Zwillinge loswerden?

Aufgabe 3: Anwendung von Lenstra — 4 + 8 Punkte

Erstelle eine geeignete ILP Formulierung um zu zeigen, dass folgende Probleme in FPT sind:

Variety Subset Sum: Gegeben eine Multimenge A von Zahlen in $\mathbb{N}$ ("Multi" heißt, Zahlen können mehrfach in A auftauchen) und ein Wert $b \in \mathbb{N}$. Gibt es eine Teilmultimenge $X \subseteq A$ mit $\sum_{x \in X} x = b$? Betrachte als Parameter die Anzahl unterschiedlicher Zahlen in A.

Makespan Scheduling: Gegeben $m \in \mathbb{N}$ Maschinen, n Jobs mit Bearbeitungszeiten $p_1, \dots, p_n \in \mathbb{N}$ und eine Schranke für die maximale Bearbeitungsdauer $k \in \mathbb{N}$. Gibt es eine Verteilung der Jobs auf die Maschinen, sodass keine Maschine länger als k Zeit benötigt? (Ganz formal: Gibt es eine Abbildung $f : \{1, \dots, n\} \to \{1, \dots, m\}$ sodass $\sum_{j : f(j) = i} p_j \leq k$ für alle $1 \leq i \leq m$?)

1 *bitte wenden*

Lenstras Theorem

Theorem (ohne Beweis)
Für $A \in \mathbb{Z}^{m \times n}$ und $b \in \mathbb{Z}^m$ kann die Frage, ob es ein $x \in \mathbb{N}^n$ mit $Ax \leq b$ gibt in $O(n^{2,5n}|A, b|)$ entschieden werden, wobei $|A, b|$ die Länge der Binärkodierung für die Instanz bezeichnet.

Lenstras Theorem

Theorem (ohne Beweis)
Für $A \in \mathbb{Z}^{m \times n}$ und $b \in \mathbb{Z}^m$ kann die Frage, ob es ein $x \in \mathbb{N}^n$ mit $Ax \leq b$ gibt in $O(n^{2,5n}|A, b|)$ entschieden werden, wobei $|A, b|$ die Länge der Binärkodierung für die Instanz bezeichnet.

Folgerungen

- ILP (als Entscheidungsproblem) mit Parameter n = Anzahl der Variablen, auch *Dimension* genannt, ist in FPT

Lenstras Theorem

Theorem (ohne Beweis)
Für $A \in \mathbb{Z}^{m \times n}$ und $b \in \mathbb{Z}^m$ kann die Frage, ob es ein $x \in \mathbb{N}^n$ mit $Ax \leq b$ gibt in $O(n^{2,5n}|A, b|)$ entschieden werden, wobei $|A, b|$ die Länge der Binärkodierung für die Instanz bezeichnet.

Folgerungen

- ILP (als Entscheidungsproblem) mit Parameter n = Anzahl der Variablen, auch *Dimension* genannt, ist in FPT
- Anzahl der Ungleichungen geht nur polynomiell in Laufzeit ein
- Größe der Zahlen geht nur logarithmisch in Laufzeit ein

Lenstras Theorem

Theorem (ohne Beweis)
Für $A \in \mathbb{Z}^{m \times n}$ und $b \in \mathbb{Z}^m$ kann die Frage, ob es ein $x \in \mathbb{N}^n$ mit $Ax \leq b$ gibt in $O(n^{2,5n}|A, b|)$ entschieden werden, wobei $|A, b|$ die Länge der Binärkodierung für die Instanz bezeichnet.

Folgerungen

- ILP (als Entscheidungsproblem) mit Parameter n = Anzahl der Variablen, auch *Dimension* genannt, ist in FPT
- Anzahl der Ungleichungen geht nur polynomiell in Laufzeit ein
- Größe der Zahlen geht nur logarithmisch in Laufzeit ein

Metatheorem
Ein parametrisiertes Problem mit Parameter k, dass sich als ILP mit $f(k)$ vielen Variablen darstellen lässt, ist in FPT.

Beispiel: Closest String

Problem: Closest String (*k* wird unser Parameter sein)
Gegeben k Strings $s_1, \ldots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

Beispiel: Closest String

Problem: Closest String (k wird unser Parameter sein)
Gegeben k Strings $s_1, \dots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F
s	B	B	E	F	G	D	C	G	B	B	C

Beispiel: Closest String

Problem: Closest String (*k* wird unser Parameter sein)
Gegeben k Strings $s_1, \ldots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

Beobachtung

- nur (Un)gleichheit innerhalb jeder Spalte relevant
- es gibt äquivalente Instanz mit $\Sigma = [k]$

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

Beispiel: Closest String

Problem: Closest String (k wird unser Parameter sein)
Gegeben k Strings $s_1, \dots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

Beobachtung

- nur (Un)gleichheit innerhalb jeder Spalte relevant
- es gibt äquivalente Instanz mit $\Sigma = [k]$

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

↓

s_1	1	1	1	1	1	1	1	1	1	1	1
s_2	2	1	2	2	1	1	2	1	2	2	2
s_3	2	2	3	1	2	2	1	1	2	1	3

Beispiel: Closest String

Problem: Closest String (k wird unser Parameter sein)
Gegeben k Strings $s_1, \dots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

Beobachtung

- nur (Un)gleichheit innerhalb jeder Spalte relevant
- es gibt äquivalente Instanz mit $\Sigma = [k]$
- Anzahl verschiedener Spalten nur von k abhängig ($\leq k!$)

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

s_1	1	1	1	1	1	1	1	1	1	1	1
s_2	2	1	2	2	1	1	2	1	2	2	2
s_3	2	2	3	1	2	2	1	1	2	1	3

Beispiel: Closest String

Problem: Closest String (*k* wird unser Parameter sein)
Gegeben k Strings $s_1, \dots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

Beobachtung

- nur (Un)gleichheit innerhalb jeder Spalte relevant
- es gibt äquivalente Instanz mit $\Sigma = [k]$
- Anzahl verschiedener Spalten nur von k abhängig ($\leq k!$)

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

s_1	1	1	1	1	1	1	1	1	1	1	1
s_2	2	1	2	2	1	1	2	1	2	2	2
s_3	2	2	3	1	2	2	1	1	2	1	3

Formulierung als ILP

- reduziere Lösung auf: Wie oft wurde für den Spalten-Typ t der Wert v gewählt?

Beispiel: Closest String

Problem: Closest String (*k* wird unser Parameter sein)
Gegeben k Strings $s_1, \ldots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

Beobachtung

- nur (Un)gleichheit innerhalb jeder Spalte relevant
- es gibt äquivalente Instanz mit $\Sigma = [k]$
- Anzahl verschiedener Spalten nur von k abhängig ($\leq k!$)

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

s_1	1	1	1	1	1	1	1	1	1	1	1
s_2	2	1	2	2	1	1	2	1	2	2	2
s_3	2	2	3	1	2	2	1	1	2	1	3
s	2	1	3	1	2	1	2	1	2	1	2

Formulierung als ILP

- reduziere Lösung auf: Wie oft wurde für den Spalten-Typ t der Wert v gewählt?

Beispiel: CLOSEST STRING

Problem: CLOSEST STRING (k wird unser Parameter sein)
Gegeben k Strings $s_1, \dots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

Beobachtung

- nur (Un)gleichheit innerhalb jeder Spalte relevant
- es gibt äquivalente Instanz mit $\Sigma = [k]$
- Anzahl verschiedener Spalten nur von k abhängig ($\leq k!$)

Formulierung als ILP

- reduziere Lösung auf: Wie oft wurde für den Spalten-Typ t der Wert v gewählt?

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

↓

s_1	1	1	1	1	1	1	1	1	1	1	1
s_2	2	1	2	2	1	1	2	1	2	2	2
s_3	2	2	3	1	2	2	1	1	2	1	3
s	2	1	3	1	2	1	2	1	2	1	2

$0 \times$ 1	$2 \times$ 2	$0 \times$ 3
$2 \times$ 1	$1 \times$ 2	$0 \times$ 3
$0 \times$ 1	$1 \times$ 2	$1 \times$ 3
$2 \times$ 1	$1 \times$ 2	$0 \times$ 3
$1 \times$ 1	$0 \times$ 2	$0 \times$ 3

Beispiel: Closest String

Problem: Closest String (k wird unser Parameter sein)
Gegeben k Strings $s_1, \ldots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

Beobachtung

- nur (Un)gleichheit innerhalb jeder Spalte relevant
- es gibt äquivalente Instanz mit $\Sigma = [k]$
- Anzahl verschiedener Spalten nur von k abhängig ($\leq k!$)

Formulierung als ILP

- reduziere Lösung auf: Wie oft wurde für den Spalten-Typ t der Wert v gewählt? $\rightarrow$ Variable $x_{t,v}$

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

↓

s_1	1	1	1	1	1	1	1	1	1	1	1
s_2	2	1	2	2	1	1	2	1	2	2	2
s_3	2	2	3	1	2	2	1	1	2	1	3
s	2	1	3	1	2	1	2	1	2	1	2

0×1	2×2	0×3
2×1	1×2	0×3
0×1	1×2	1×3
2×1	1×2	0×3
1×1	0×2	0×3

Beispiel: Closest String

Problem: Closest String (*k* wird unser Parameter sein)
Gegeben k Strings $s_1, \dots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

Beobachtung

- nur (Un)gleichheit innerhalb jeder Spalte relevant
- es gibt äquivalente Instanz mit $\Sigma = [k]$
- Anzahl verschiedener Spalten nur von k abhängig ($\leq k!$)

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

s_1	1	1	1	1	1	1	1	1	1	1	1
s_2	2	1	2	2	1	1	2	1	2	2	2
s_3	2	2	3	1	2	2	1	1	2	1	3
s	2	1	3	1	2	1	2	1	2	1	2

Formulierung als ILP

- reduziere Lösung auf: Wie oft wurde für den Spalten-Typ t der Wert v gewählt? $\rightarrow$ Variable $x_{t,v}$
- wähle ein Zeichen für jede Spalte von Typ t:

$$\sum_{v \in \Sigma} x_{t,v} = \text{Anzahl Spalten von Typ } t$$

0×1	2×2	0×3
2×1	1×2	0×3
0×1	1×2	1×3
2×1	1×2	0×3
1×1	0×2	0×3

Beispiel: Closest String

Problem: Closest String (*k* wird unser Parameter sein)
Gegeben k Strings $s_1, \dots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

Beobachtung

- nur (Un)gleichheit innerhalb jeder Spalte relevant
- es gibt äquivalente Instanz mit $\Sigma = [k]$
- Anzahl verschiedener Spalten nur von k abhängig ($\leq k!$)

Formulierung als ILP

- reduziere Lösung auf: Wie oft wurde für den Spalten-Typ t der Wert v gewählt? $\rightarrow$ Variable $x_{t,v}$
- wähle ein Zeichen für jede Spalte von Typ t:

$$\sum_{v \in \Sigma} x_{t,v} = \text{Anzahl Spalten von Typ } t$$

- Distanz $\leq D$ für jedes s_i: $\sum_{t \in \text{Typen}} \sum_{\substack{v \in \Sigma \\ v \neq t[i]}} x_{t,v} \leq D$

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

$\downarrow$

s_1	1	1	1	1	1	1	1	1	1	1	1
s_2	2	1	2	2	1	1	2	1	2	2	2
s_3	2	2	3	1	2	2	1	1	2	1	3
s	2	1	3	1	2	1	2	1	2	1	2

0×1	2×2	0×3
2×1	1×2	0×3
0×1	1×2	1×3
2×1	1×2	0×3
1×1	0×2	0×3

Beispiel: Closest String

Problem: Closest String (*k* wird unser Parameter sein)
Gegeben k Strings $s_1, \ldots, s_k \in \Sigma^n$ und $D \in \mathbb{N}$. Gibt es einen String $s \in \Sigma^n$, der von jedem s_i Hamming Distanz maximal D hat?

Beobachtung

- nur (Un)gleichheit innerhalb jeder Spalte relevant
- es gibt äquivalente Instanz mit $\Sigma = [k]$
- Anzahl verschiedener Spalten nur von k abhängig ($\leq k!$)

s_1	A	B	C	F	G	D	C	G	E	B	A
s_2	B	B	D	A	G	D	B	G	B	E	C
s_3	B	A	E	F	C	A	C	G	B	B	F

$\downarrow$

s_1	1	1	1	1	1	1	1	1	1	1	1
s_2	2	1	2	2	1	1	2	1	2	2	2
s_3	2	2	3	1	2	2	1	1	2	1	3
s	2	1	3	1	2	1	2	1	2	1	2

0×1	2×2	0×3
2×1	1×2	0×3
0×1	1×2	1×3
2×1	1×2	0×3
1×1	0×2	0×3

Formulierung als ILP

- reduziere Lösung auf: Wie oft wurde für den Spalten-Typ t der Wert v gewählt? $\rightarrow$ Variable $x_{t,v}$
- wähle ein Zeichen für jede Spalte von Typ t:

$$\sum_{v \in \Sigma} x_{t,v} = \text{Anzahl Spalten von Typ } t$$

- Distanz $\leq D$ für jedes s_i: $\displaystyle\sum_{t \in \text{Typen}} \sum_{\substack{v \in \Sigma \\ v \neq t[i]}} x_{t,v} \leq D$
- $\rightarrow$ ILP mit „nur“ $k! \cdot k$ Variablen

Aufgaben

Wie können die folgenden Probleme mit (I)LPs gelöst werden?

- Hitting-Set (einfach)
- Hamiltonkreis, sowie TSP
- Minimum Linear Arrangement: Given a Graph $G = (V, E)$ find an ordering $\sigma : V \mapsto \{1, \ldots, n\}$ of the vertices that minimizes $\sum_{\{u,v\} \in E} |\sigma(u) - \sigma(v)|$.

Further reading: Integer programming in parameterized complexity: Five miniatures

https://doi.org/10.1016/j.disopt.2020.100596

Integer programming in parameterized complexity: Five miniatures☆

Tomáš Gavenčiak [a], Martin Koutecký [a,1], Dušan Knop [b,*,2]

[a] Computer Science Institute of Charles University, Faculty of Mathematics and Physics, Charles University, Malostranské náměstí 25, Praha, Czech Republic
[b] Department of Theoretical Computer Science, Faculty of Information Technology, Czech Technical University in Prague, Thákurova 9, Praha, Czech Republic

ARTICLE INFO

Article history:
Received 31 December 2018
Received in revised form 14 May 2020
Accepted 25 June 2020
Available online 20 July 2020

Keywords:
Graph coloring
Sum-coloring
Parameterized complexity

ABSTRACT

Powerful results from the theory of integer programming have recently led to substantial advances in parameterized complexity. However, our perception is that, except for Lenstra's algorithm for solving integer linear programming in fixed dimension, there is still little understanding in the parameterized complexity community of the strengths and limitations of the available tools. This is understandable: it is often difficult to infer exact runtimes or even the distinction between FPT and XP algorithms, and some knowledge is simply unwritten folklore in a different community. We wish to make a step in remedying this situation.

To that end, we first provide an easy to navigate quick reference guide of integer programming algorithms from the perspective of parameterized complexity. Then, we show their applications in three case studies, obtaining FPT algorithms with runtime $f(k)\mathrm{poly}(n)$. We focus on:

- *Modeling*: since the algorithmic results follow by applying existing algorithms to new models, we shift the focus from the complexity result to the modeling result, highlighting common patterns and tricks which are used.
- *Optimality program:* after giving an FPT algorithm, we are interested in reducing the dependence on the parameter; we show which algorithms and tricks are often useful for speed-ups.
- *Minding the* $\mathrm{poly}(n)$: reducing $f(k)$ often has the unintended consequence of increasing $\mathrm{poly}(n)$; so we highlight the common trade-offs and show how to get the best of both worlds.

Specifically, we consider graphs of bounded neighborhood diversity which are in a sense the simplest of dense graphs, and we show several FPT algorithms for Capacitated Dominating Set, Sum Coloring, Max-q-Cut, and certain other coloring problems by modeling them as convex programs in fixed dimension, n-fold integer programs, bounded dual treewidth programs, indefinite quadratic programs

☆ An extended abstract presenting preliminary results appeared in proceedings of IPEC 2018 [1] (Gavenčiak et al (2018)).

* Corresponding author.
E-mail addresses: gavento@kam.mff.cuni.cz (T. Gavenčiak), koutecky@iuuk.mff.cuni.cz (M. Koutecký), dusan.knop@fit.cvut.cz (D. Knop).

1 Partially supported by Charles University project UNCE/SCI/004, by the project 19-27871X of GA ČR, and by the Israel Science Foundation grant 308/18.
2 Supported by the OP VVV MEYS funded project CZ.02.1.01/0.0/0.0/16_019/0000765 "Research Center for Informatics".

https://doi.org/10.1016/j.disopt.2020.100596