Technische Universität Berlin
Fakultät IV: Elektrotechnik und Informatik
Institut für Softwaretechnik und Theoretische Informatik
Algorithmik und Komplexitätstheorie (AKT)



Algorithm Engineering für Bounded Density Edge Deletion mit halbintegraler Zieldichte

Bachelorarbeit

von Niklas Markus Keitsch

zur Erlangung des Grades „Bachelor of Science“ (B. Sc.)
im Studiengang Informatik

Erstgutachter:	Prof. Dr. Stephan Kreutzer
Zweitgutachter:	Dr. André Nichterlein
Betreuer:	T. Breitkopf und A. Herrmann

Eigenständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne Hilfe Dritter und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe. Alle Stellen, die den benutzten Quellen und Hilfsmitteln wörtlich oder inhaltlich entnommen sind, habe ich als solche kenntlich gemacht.

Sofern KI-Tools verwendet wurden, habe ich Produktnamen, Hersteller, die jeweils verwendete Softwareversion und die Art der Nutzung (z.B. sprachliche Überprüfung und Verbesserung der Texte, systematische Recherche) benannt. Ich verantworte die Auswahl, die Übernahme und sämtliche Ergebnisse des von mir verwendeten KI-generierten Outputs vollumfänglich selbst.

Die Satzung zur Sicherung guter wissenschaftlicher Praxis an der TU Berlin vom 8. März 2017 https://www.static.tu.berlin/fileadmin/www/10002457/K3-AMBl/Amtsblatt_2023/Amtliches_Mitteilungsblatt_Nr._16_vom_30.05.2023.pdf habe ich zur Kenntnis genommen.

Ich erkläre weiterhin, dass ich die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt habe.

Berlin, den _____ _____

Datum Unterschrift

Zusammenfassung

Diese Arbeit untersucht das Bounded-Density Edge Deletion-Problem, bei dem aus einem ungerichteten Graphen möglichst wenige Kanten entfernt werden sollen, sodass die maximale Subgraphendichte einen vorgegebenen Zielwert τ nicht überschreitet. Während das Problem für allgemeine rationale Zieldichten nach Bentert u. a. [Ben+26] NP-vollständig ist, wird hier gezeigt, dass es für halbintegrale Zieldichten τ mit $2\tau \in \mathbb{N}$ in polynomieller Zeit lösbar ist. Dazu wird eine Reduktion auf das Maximum Cardinality Matching-Problem vorgestellt, die in $\mathcal{O}(\tau(|V| + |E|))$ Zeit durchführbar ist. Zusätzlich werden vier Datenreduktionsregeln entwickelt und formal bewiesen, die die Probleminstanz vor der Reduktion verkleinern, wobei insbesondere eine gradbasierte Regel in linearer Zeit anwendbar ist. Die experimentelle Evaluation auf 83 realen Graphinstanzen aus dem SNAP-Datensatz zeigt, dass die Reduktionsregeln insbesondere für größere Zieldichten zu erheblichen Instanzverkleinerungen und Laufzeitverbesserungen führen.

Inhaltsverzeichnis

1 Einleitung **9**
- 1.1 Einsatz von KI-Tools . . . 12
- 1.2 Stand der Forschung . . . 12
- 1.3 Ergebnisse . . . 13
- 1.4 Grundlagen . . . 13
  - 1.4.1 Graphen . . . 13
  - 1.4.2 Fraktionale Orientierungen . . . 14

2 Reduktion auf Maximum Cardinality Matching **15**

3 Daten-Reduktionsregeln **21**
- 3.1 Knotengrad . . . 21
- 3.2 Kanten zwischen Knoten mit kleinem Grad . . . 21
- 3.3 Zusammenhangskomponenten . . . 23
- 3.4 Kreise aus Knoten mit kleinem Grad . . . 24

4 Experimentelle Evaluation **27**
- 4.1 Implementierung . . . 27
- 4.2 Tests und Validierung . . . 27
- 4.3 Testdaten . . . 28
- 4.4 Ergebnisse . . . 29
  - 4.4.1 Gesamtlaufzeit . . . 29
  - 4.4.2 Reduktionsregeln . . . 33

5 Fazit **41**
- 5.1 Ausblick . . . 42

Literatur **45**

6 Anhang **47**
- 6.1 Auflistung der Testdaten . . . 47

1 Einleitung

Graphen sind ein grundlegendes Modell zur Darstellung von Beziehungen zwischen Objekten und finden in zahlreichen Bereichen Anwendung – von sozialen Netzwerken über Kommunikationsinfrastrukturen bis hin zu biologischen Systemen. Eine wichtige Eigenschaft, die die Struktur eines Graphen charakterisiert, ist seine Dichte. Intuitiv beschreibt die Dichte, wie stark die Knoten eines Graphen untereinander vernetzt sind. Formal ist die Dichte eines Graphen G als das Verhältnis der Anzahl seiner Kanten zur Anzahl seiner Knoten definiert, also $\rho(G) = \frac{|E(G)|}{|V(G)|}$. Ein Graph mit vielen Kanten relativ zur Knotenanzahl gilt als dicht, während ein Graph mit wenigen Kanten als dünn bezeichnet wird. Die Dichte ist insbesondere deshalb relevant, weil sie Rückschlüsse auf strukturelle Eigenschaften eines Graphen erlaubt.

Neben der Dichte des gesamten Graphen ist häufig die maximale Dichte aller Subgraphen $\rho^*(G)$ von Interesse, also die höchste Dichte über alle Subgraphen von G. Während die globale Dichte eines Graphen durch Hinzufügen isolierter Knoten oder Entfernen von beliebigen Kanten gesenkt werden kann, erfasst $\rho^*(G)$ die tatsächlich dichteste Struktur innerhalb des Graphen. Es ist damit ein interessantes Maß für die lokale Vernetzung. Das Entfernen einer Kante im dichtesten Subgraphen mit n Knoten senkt dessen Dichte um $\frac{1}{n}$. Hierbei werden andere dichte Subgraphen nicht unbedingt beeinflusst, sodass $\rho^*(G)$ möglicherweise weniger oder sogar gar nicht sinkt. Straßennetze, wie die im SNAP-Datensatz [LK14], weisen typischerweise eine geringe globale Dichte sowie maximale Subgraphendichte auf. Andere geographische oder soziale Netzwerke, sowie Web-Graphen hingegen können global dünn sein, aber dichte Subgraphen enthalten.

Diese dichten Subgraphen deuten auf eng vernetzte Gruppen hin, die in vielen Anwendungen eine besondere Bedeutung haben: In sozialen Netzwerken können sie Gemeinschaften mit starker Interaktion entsprechen. Mithilfe von dichten Subgraphen können daher Betrugsmuster, wie von Hooi u. a. [Hoo+17] beschrieben, erkannt werden. Das Finden und Kontrollieren dichter Subgraphen ist daher eine wiederkehrende Fragestellung in der Graphalgorithmik.

In einigen Anwendungen ist es jedoch wünschenswert, dass die Dichte des dichtesten Subgraphen einen bestimmten Schwellenwert nicht überschreitet. Dies könnte beispielsweise in Versorgungsnetzen zur Entfernung von überschüssigen Kanten oder zur Reduktion von Netzwerkkomplexität der Fall sein. In solchen Fällen könnte es notwendig sein, Kanten zu entfernen, um die Dichte zu kontrollieren.

Diese Arbeit beschäftigt sich mit dem Problem des Entfernens von Kanten aus einem ungerichteten Graphen, sodass die Dichte, also das Verhältnis von Kanten zu Knoten, des dichtesten Subgraphen einen vorgegebenen Zielwert τ nicht überschreitet. Dies wird im Folgenden als Bounded-Density Edge Deletion (BDED)-Problem bezeichnet. In Abbildung 1.1 ist dies an einem Beispiel dargestellt. Für halbintegrale Zieldichten einen

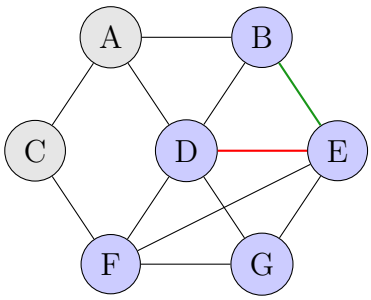


Abbildung 1.1: Beispiel für BDED mit $\tau = \frac{3}{2}$: Der dichteste Subgraph mit den Knoten $\{B, D, E, F, G\}$ hat die Dichte $\frac{8}{5}$. Durch das Entfernen der Kante $\{D, E\}$ wird die Dichte des dichtesten Subgraphen auf $\frac{7}{5}$ gesenkt, wodurch die Zieldichte τ nicht überschritten wird. Beispielsweise könnte hier auch die Kante $\{B, E\}$ entfernt werden, um die maximale Subgraphendichte auf $\frac{3}{2}$ zu senken.

Polynomialzeit-Algorithmus bereitzustellen, der aus einer hier vorgestellten Reduktion auf das in polynomieller Zeit lösbare Maximum Cardinality Matching-Problem resultiert, und diesen zu implementieren und experimentell zu evaluieren, ist das Hauptziel dieser Arbeit.

Konkrete Anwendungen ergeben sich möglicherweise in verschiedenen Bereichen: In sozialen Netzwerken könnte die Kontrolle der Dichte ähnlich wie von Hooi u. a. [Hoo+17] zur Erkennung koordinierter unechter Aktivitäten eingesetzt werden, bei denen Fake-Account-Netzwerke dichte Verbindungsstrukturen aufweisen, indem sie sich gegenseitig folgen und verstärken, um ihre Sichtbarkeit künstlich zu erhöhen. Durch das gezielte Entfernen von Verbindungen könnten solche anomalen Strukturen aufgelöst werden. Im Bereich der Betrugserkennung in Finanznetzwerken könnten ungewöhnlich dichte Transaktionsmuster durch koordinierte betrügerische Aktivitäten identifiziert werden. In Lastenverteilungs-Netzwerken könnte die Steuerung der Dichte ebenfalls eine nützliche Rolle spielen.

Die Beschränkung auf halbintegrale Zieldichten, also Werte der Form $\frac{p}{2}$ mit $p \in \mathbb{N}$, ist dabei von besonderer Bedeutung, da nach Bentert u. a. [Ben+26] für diese Werte τ-BDED-Algorithmen existieren, die in polynomieller Zeit ausführbar sind, während das Problem für allgemeine rationale Dichten NP-vollständig ist, wobei τ nicht zur Eingabe gehört. Zudem ermöglichen halbintegrale Dichten eine elegante Reduktion auf Maximum Cardinality Matching, wodurch sich enorme algorithmische Vorteile ergeben. Halbintegrale Zieldichten bieten einige intuitive Interpretationen für die Struktur des resultierenden Graphen, wenn k groß genug gewählt ist, um die maximale Subgraphendichte auf höchstens τ zu senken, wie in Abbildung 1.2 dargestellt: Für eine Zieldichte von 0 darf es keine Kante im Graph geben. Bei einer Zieldichte von $\frac{1}{2}$ darf jeder Knoten nur eine Kante haben, wie es beispielsweise in einem Matching der Fall ist. Bei einer Zieldichte von 1 darf jede Zusammenhangskomponente höchstens einen Kreis enthalten. Ab dieser Dichte können Knoten unbegrenzten Grad haben. Eine Clique mit n Knoten hat eine Dichte von $\frac{n-1}{2}$, wodurch bei einer Zieldichte von $\frac{p}{2}$ die größte erlaubte Clique K_{p+1}

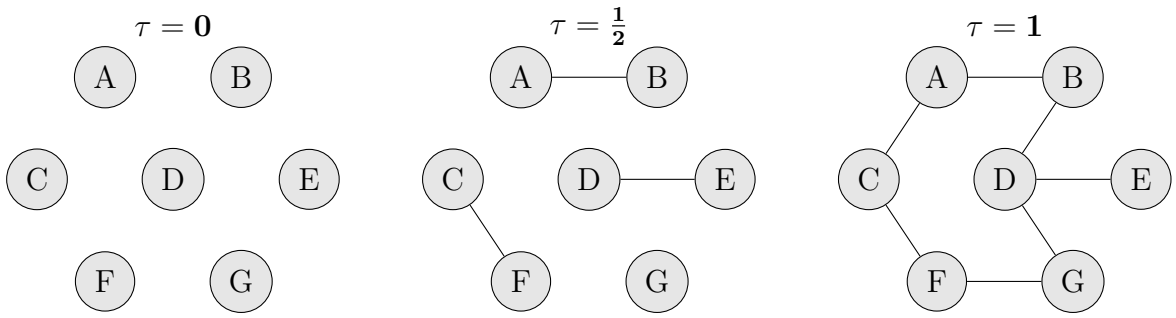


Abbildung 1.2: Beispielhafte resultierende Graphen nach dem Entfernen der Kanten aus BDED auf dem Graph aus Abbildung 1.1 für $\tau \in \left\{0, \frac{1}{2}, 1\right\}$, wenn k groß genug gewählt ist, um die Zieldichte zu erreichen.

ist.

Diese Herangehensweise profitiert davon, dass Maximum Cardinality Matching bereits ein wohluntersuchtes Problem ist. Die Wahl der Reduktion gegenüber anderen Lösungsansätzen erfolgt aus mehreren Gründen: Die Algorithmen sind bereits hochgradig optimiert und in vielen effizienten Implementierungen verfügbar. Die beste bekannte Laufzeit liegt nach Gabow [Gab17] und Vazirani [Vaz20] bei $\mathcal{O}(m\sqrt{n})$ Zeit, und wird durch den Algorithmus von Micali und Vazirani [MV80] erreicht. Für ganzzahlige Dichten wird von Bentert u. a. [Ben+26] eine Flow-basierte Reduktion mit ganzzahligen Kapazitäten vorgestellt. Indem Flüsse auf die Zieldichte beschränkt werden, können einige Knoten keinen Fluss weiterleiten, wodurch entschieden wird, ob sie entfernt werden. Diese Reduktion lässt sich nicht direkt auf halbintegrale Zieldichten übertragen, während die hier vorgestellte Reduktion auch mit diesen arbeitet.

Da die resultierende Maximum Cardinality Matching-Instanz abhängig von τ größer als der ursprüngliche Graph wird, werden Datenreduktionsregeln vorgestellt, die die Probleminstanz vor der Reduktion verkleinern, ohne die Lösbarkeit zu verändern. Diese verwenden den Grad der Knoten, um Fälle zu identifizieren, in denen Knoten entfernt werden können. Beispielsweise können alle Knoten mit einem Grad von höchstens τ entfernt werden.

Zunächst werden die notwendigen graphentheoretischen Grundlagen sowie fraktionale Orientierungen eingeführt, die eine zentrale Rolle bei der Analyse der Graphdichte spielen. Darauf aufbauend wird die Reduktion auf Maximum Cardinality Matching im Detail hergeleitet und bewiesen. Daraufhin werden die Reduktionsregeln vorgestellt, ihre Korrektheit bewiesen und auf ihre theoretische Laufzeit eingegangen. Die praktische Umsetzung dieser Reduktion sowie der Reduktionsregeln wurde in der Programmiersprache Golang geschrieben. Sie erhält einen ungerichteten Graphen und eine halbintegrale Zieldichte als Eingabe. Abschließend wird die Leistungsfähigkeit des Ansatzes durch eine experimentelle Evaluation auf realen Graphinstanzen dokumentiert, bevor die Ergebnisse zusammengefasst und ein Fazit gezogen werden.

1.1 Einsatz von KI-Tools

Im Rahmen dieser Arbeit wurden über GitHub Copilot verschiedene KI-Modelle verwendet: GPT (4.1, 5.1, 5.2 und 5.3-Codex) sowie Claude (Opus 4.5 und 4.6, Sonnet 4, 4.5 und 4.6). Der Einsatz beschränkte sich auf sprachliche und grammatische Korrekturen, grobe Recherchezwecke sowie die Generierung von Vorschlägen für Testfälle. Der sachliche Inhalt der Arbeit wurde vollständig eigenständig erstellt.

1.2 Stand der Forschung

Den dichtesten Subgraphen eines Graphen zu finden, ist bekanntlich in polynomieller Zeit mit dem Algorithmus von Goldberg [Gol84] oder einer LP-Formulierung von Charikar [Cha00] lösbar. Der Algorithmus von Goldberg [Gol84] führt eine binäre Suche über den Dichtewert durch und löst für jeden Kandidatenwert ein Maximum-Flow-Problem in einem parametrisch konstruierten Netzwerk, um zu entscheiden, ob ein Subgraph mit mindestens dieser Dichte existiert. Er läuft in $\mathcal{O}(M(n, n+m)\log(n))$ Zeit, wobei $M(n, m)$ die Laufzeit eines Algorithmus für Maximum-Flow in einem Netzwerk mit n Knoten und m Kanten bezeichnet. Goldberg [Gol84] führt für M den Algorithmus von Karzanov [Kar74], der in $\mathcal{O}(n^3)$ Zeit läuft, sowie den Algorithmus von Sleator [Sle80], der in $\mathcal{O}(nm\ \log(m))$ Zeit läuft, an.

Bounded-Density Edge Deletion wurde zunächst von Bazgan, Nichterlein und Alferez [BNA24] untersucht. Es wurde gezeigt, dass das Problem für die Dichten $\tau \leq \frac{2}{3}$ und $1 - \frac{1}{n} \leq \tau \leq 1$ mittels Maximum Cardinality Matching und Spannbäumen in polynomieller Zeit lösbar ist. Auf Bäumen kann BDED für beliebige Zieldichten in $\mathcal{O}(n^3)$ Zeit und auf Cliquen in $\mathcal{O}(n^2)$ Zeit gelöst werden.

Das τ-Bounded-Density Edge Deletion-Problem, bei dem τ konstant ist und nicht zur Eingabe gehört, wurde bereits von Bentert u. a. [Ben+26] untersucht. Dort wurde für eine ganzzahlige Zieldichte τ eine Reduktion auf eine *Maximum-Flow*-Instanz gezeigt. Außerdem wurde bewiesen, dass das Problem für halbintegrale Zieldichten oder Dichten unter $\frac{2}{3}$ polynomiell lösbar ist. Für alle anderen Werte von τ wurde gezeigt, dass das Problem NP-vollständig ist.

Das verwandte τ-Bounded-Density Vertex Deletion Problem, bei dem Knoten samt ihrer anliegenden Kanten entfernt werden, sei nach Bazgan, Nichterlein und Alferez [BNA24] für allgemeine Zieldichten NP-vollständig. Auf Bäumen sei es jedoch für alle Zieldichten in $\mathcal{O}(n)$ Zeit und auf Cliquen trivial in $\mathcal{O}(n^2)$ Zeit lösbar.

Zur Lösung von Maximum Cardinality Matching verwendet die Implementierung in dieser Arbeit den Algorithmus von Kececioglu und Pecqueur [KP98], der auf dem Algorithmus von Edmonds [Edm65] basiert. Dieser sucht iterativ nach augmentierenden Pfaden und kontrahiert dabei ungerade Kreise (sogenannte Blüten), um auch in allgemeinen, nicht bipartiten Graphen ein größtmögliches Matching zu finden. Der Algorithmus läuft in $\mathcal{O}(nm\,\alpha(m, n))$, wobei α die inverse Ackermann-Funktion bezeichnet. Er nutzt ein initiales Matching, das mit einem approximativen Greedy-Algorithmus in $\mathcal{O}(m + n)$ Zeit gefunden werden kann, sowie eine frühere Terminierung der Tiefensu-

che. Von Kececioglu und Pecqueur [KP98] wurden weitere Heuristiken vorgestellt und verglichen. In der Praxis sei die Kombination aus den zwei genannten Heuristiken für dünne Graphen am effizientesten, wobei für sehr große Graphen ein Start ohne Greedy-Lösung effizienter sein könnte. Im Vergleich zu anderen Algorithmen von Kececioglu und Pecqueur [KP98] sei dieser Algorithmus um den Faktor 4 – 350 schneller.

1.3 Ergebnisse

Die Ergebnisse dieser Arbeit lassen sich wie folgt zusammenfassen: Es wird in Theorem 2.2 gezeigt, dass BDED für halbintegrale Zieldichten τ mit $2\tau \in \mathbb{N}$ in $\mathcal{O}(\tau(|V|+|E|))$ Zeit auf Maximum Cardinality Matching reduziert werden kann. Die Reduktion liefert zusammen mit einem Maximum Cardinality Matching-Algorithmus einen polynomiellen Gesamtalgorithmus für BDED für halbintegrale Zieldichten.

Es wurden vier Datenreduktionsregeln vorgestellt und formal bewiesen, die auf dem Knotengrad, auf Kanten zwischen Knoten mit kleinem Grad, auf Zusammenhangskomponenten mit beschränktem Maximalgrad sowie auf Kreisen aus Knoten mit kleinem Grad basieren. Die erschöpfende Anwendung der Knotengrad-Regel, die wie in der experimentellen Evaluation beobachtet, besonders effektiv ist, ist in $\mathcal{O}(|V| + |E|)$ Zeit möglich.

Die experimentelle Evaluation auf 83 realen Graphinstanzen aus dem SNAP-Datensatz zeigt, dass die Reduktionsregeln insbesondere für größere Zieldichten zu erheblichen Instanzverkleinerungen führen können. Die Reduktionsregel 3.1.1 über den Knotengrad leistet dabei den mit Abstand größten Beitrag zur Instanzverkleinerung. Für größere Zieldichten wie $\tau = \frac{9}{2}$ führt die Anwendung aller Reduktionsregeln fast immer zu erheblichen Laufzeitverbesserungen, während der Effekt für kleine Zieldichten wie $\tau = \frac{3}{2}$ nahezu laufzeitneutral ist.

1.4 Grundlagen

Die folgende Notation orientiert sich an der von Bentert u. a. [Ben+26] verwendeten Notation.

Die natürlichen Zahlen $\mathbb{N} := \{0, 1, \ldots\}$ beinhalten die Null, $\mathbb{N}^+ := \mathbb{N} \setminus \{0\}$ die positiven natürlichen Zahlen. Sei $[n] := \{1, \ldots, n\}$.

1.4.1 Graphen

Alle Graphen in dieser Arbeit sind einfach und ungewichtet. Sei G ein ungerichteter Graph und D ein gerichteter Graph. Die Knotenmenge von G und D wird mit $V(G)$ und $V(D)$ und die Kantenmenge mit $E(G)$ und $A(D)$ bezeichnet. Für $H \in \{G, D\}$ ist n_H die Anzahl der Knoten und m_H die Anzahl der Kanten von H.

Der Grad eines Knotens $v \in V(G)$ wird mit $\deg_G(v) := |N(v)|$ bezeichnet, hierbei ist die Knotenmenge $N(v) := \{u \in V(G) \mid \{u, v\} \in E(G)\}$ die Nachbarschaft von v.

Den eingehenden Grad eines Knotens $v \in V(D)$ bezeichnet $\deg_D^-(v) := |N^-(v)|$, wobei $N^-(v) := \{u \in V(D) \mid (u,v) \in A(D)\}$ die eingehende Nachbarschaft von v ist. Wenn der betreffende Graph aus dem Kontext klar ist, wird der Index weggelassen. Sei $H \subseteq G$, wenn H ein Subgraph von G ist. Für eine Teilmenge $E' \subseteq E(G)$ sei $G - E'$ der Subgraph von G, der durch das Löschen der Kanten in E' entsteht.

Die Dichte von G ist definiert als $\rho(G) := \frac{m}{n}$, wobei die Dichte eines leeren Graphen als null definiert wird. Mit $\rho^*(G)$ wird die Dichte des dichtesten Subgraphen von G bezeichnet. Diese ist definiert als $\rho^*(G) := \max_{H \subseteq G} \rho(H)$.

Bounded-Density Edge Deletion (BDED)
Eingabe: Ein ungerichteter Graph G, ein $k \in \mathbb{N}$ und ein $\tau \in \mathbb{Q}^+$.
Frage: Gibt es eine Teilmenge $F \subseteq E(G)$ mit $|F| \leq k$, wobei gilt: $\rho^*(G-F) \leq \tau$?

Falls τ größer als $\frac{|V(G)|-1}{2}$ ist, dann ist die BDED-Instanz trivial lösbar, da jeder Graph mit n Knoten eine Dichte von höchstens $\frac{n(n-1)}{2n} = \frac{n-1}{2}$ haben kann und keine Kanten mehr gelöscht werden müssen.

Maximum Cardinality Matching
Eingabe: Ein ungerichteter Graph G und ein $k \in \mathbb{N}$.
Frage: Existiert eine Kantenmenge $M \subseteq E(G)$ mit $|M| \geq k$, sodass keine zwei Kanten aus M einen gemeinsamen Knoten haben?

1.4.2 Fraktionale Orientierungen

Um die Dichte des dichtesten Subgraphen zu bestimmen, können Orientierungen verwendet werden, die aus einer LP-Formulierung von Charikar [Cha00] stammen. Eine Orientierung eines ungerichteten Graphen G ordnet jeder Kante $\{u,v\}$ eine Richtung zu, also entweder (u,v) oder (v,u). Charikar [Cha00] zeigte, dass der maximale eingehende Grad über alle Knoten einer Orientierung von G eine obere Schranke für $\rho^*(G)$ ist. Dies ist ebenfalls für fraktionale Orientierungen gegeben, bei denen ein Teil der Kante $\{u,v\}$ zu u und der Rest zu v zugewiesen wird. Formell ordnet eine fraktionale Orientierung ϕ von G jeder Kante $e = \{u,v\}$ zwei nicht negative rationale Werte $\phi(e)^u$ und $\phi(e)^v$ zu, sodass $\phi(e)^u + \phi(e)^v = 1$. Sei $\deg_\phi^-(v) := \sum_{u \in N(v)} \phi(\{u,v\})^v$ der eingehende Grad des Knotens $v \in V(G)$ und $\Delta_\phi^- := \max_{v \in V(G)} \deg_\phi^-(v)$ der maximale eingehende Grad von G bezüglich der fraktionalen Orientierung ϕ.

Lemma 1.1. (Charikar [Cha00]) *Für jeden Graphen G gibt es eine fraktionale Orientierung ϕ mit $\Delta_\phi^- = \rho^*(G)$.*

Die im Folgenden eingeführten q-fraktionalen Orientierungen bilden die Grundlage für die Reduktion von BDED für halbintegrale Zieldichten auf Maximum Cardinality Matching, da durch sie die Dichteanforderungen in der resultierenden Instanz direkt abgebildet werden können.

Für ein $q \in \mathbb{N}^+$ ist eine q-fraktionale Orientierung eine fraktionale Orientierung ϕ_q eines Graphen G, die für jede Kante $e = \{u,v\} \in E(G)$, $\phi_q(e)^u$ und $\phi_q(e)^v$ einen Wert in $\{0\} \cup \left\{ \frac{x}{q} \mid x \in [q] \right\}$ zuweist. Eine 2-fraktionale Orientierung wird auch als halbintegrale Orientierung bezeichnet.

2 Reduktion auf Maximum Cardinality Matching

Ein elementarer Baustein der Reduktion von BDED auf Maximum Cardinality Matching ist das folgende Lemma, welches die Existenz von q-fraktionalen Orientierungen mit maximalem eingehendem Grad gleich der Dichte $\frac{p}{q}$ des dichtesten Subgraphen garantiert und später ermöglicht, eine Maximum Cardinality Matching-Instanz zu konstruieren, die diese Orientierung repräsentiert. Es wird im Folgenden mittels Modellierung von Orientierungen als Flussnetzwerken mit ganzzahligen Kapazitäten sowie über Skalierung und der Integrality Property [AMO93] bewiesen.

Lemma 2.1. *Sei G ein ungerichteter Graph mit $\rho^*(G) \leq \frac{p}{q}$. Dann existiert eine q-fraktionale Orientierung ϕ_q mit $\Delta^-_{\phi_q} = \frac{p}{q}$.*

Beweis. Nach Lemma 1.1 existiert eine fraktionale Orientierung ϕ auf G mit $\Delta^-_\phi \leq \frac{p}{q}$.

Der Beweis erfolgt mittels Modellierung von ϕ als Fluss in einem geeigneten Netzwerk. Die Orientierung einer Kante $e = \{u, v\}$ wird als Wahl eines Flussanteils betrachtet. Die Einschränkungen an den maximalen Fluss in einem Knoten werden als Kapazitäten zu einer Senke modelliert.

Man betrachte folgendes Flussnetzwerk $N = (V_N, A)$ mit Quelle s und Senke t. Für jede Kante $e = \{u, v\} \in E(G)$ wird ein Zwischenknoten w_e eingeführt, sodass gilt:

$$V_N = \{s, t\} \cup V \cup \{w_e \mid e \in E(G)\}.$$

Die Quelle s hat zu jedem Zwischenknoten w_e eine Kante. Jeder Knoten w_e hat zu den beiden Endknoten der Kante e eine Kante. Jeder Knoten $v \in V(G)$ hat eine Kante zur Senke t. Die Kantenmenge A ist somit definiert als:

$$A = \{(s, w_e) \mid e \in E(G)\} \cup \{(w_e, z) \mid z \in e \in E(G)\} \cup \{(v, t) \mid v \in V(G)\}.$$

Die Kapazitäten $c((x, y)) : A \to \mathbb{Q}_{\geq 0}$ sind definiert durch:

$$c((x, y)) := \begin{cases} \frac{p}{q}, & \text{falls } y = t, \\ 1, & \text{sonst.} \end{cases}$$

Für jede ursprüngliche Kante wird so ein Fluss von 1 auf die beiden Endknoten verteilt, während die maximale Summe der Flüsse dieser Knoten zu t auf $\frac{p}{q}$ beschränkt ist. Abbildung 2.1 stellt eine Kante $e = \{u, v\}$ als Flussnetzwerk dar.

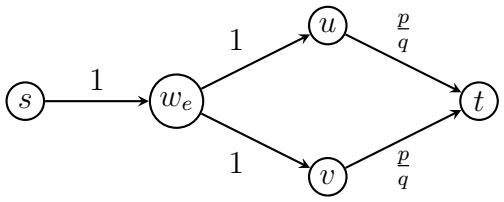


Abbildung 2.1: Flussnetzwerk für eine Kante $e = \{u, v\}$.

Ein maximaler Fluss $f_\phi : A \to \mathbb{Q}_{\geq 0}$ im Netzwerk N ist durch ϕ gegeben:

$$f_\phi((x,y)) := \begin{cases} 1, & \text{falls } x = s, \\ \phi(\{u,v\})^u, & \text{falls } (x,y) = (w_e, u) \text{ für ein } e = \{u,v\} \in E(G), \\ \deg^-_\phi(x), & \text{falls } y = t. \end{cases}$$

Dieser maximale Fluss hat den Wert $|f_\phi| = |E(G)|$. Durch Multiplikation aller Kapazitäten im Netzwerk N mit q werden alle Kapazitäten ganzzahlig. Sei qf_ϕ der mit q skalierte Fluss f_ϕ im skalierten Netzwerk. Der Wert dieses Flusses ist $|qf_\phi| = q|E(G)|$.

Nach der *Integrality Property* von Ahuja, Magnanti und Orlin [AMO93] besitzt jedes Flussnetzwerk mit ganzzahligen Kapazitäten einen optimalen Fluss, bei dem alle Flusswerte ganzzahlig sind. Somit existiert ein ganzzahliger Fluss $qf^* : A \to \mathbb{N}$ im skalierten Flussnetzwerk. Der Wert dieses Flusses ist ebenfalls $|qf^*| = |qf_\phi| = q|E(G)|$.

Sei $f^* : A \to \{0\} \cup \left\{\frac{1}{q}, \ldots, \frac{p}{q}\right\}$ der mit $\frac{1}{q}$ skalierte Fluss f^* im ursprünglichen Netzwerk. Der Wert dieses Flusses ist $|f^*| = \frac{1}{q}|qf^*| = |E(G)|$. Dieser Fluss f^* definiert wiederum eine q-fraktionale Orientierung ϕ_q von G durch:

$$\phi_q(e)^u := f^*((w_e, u)) \text{ und } \phi_q(e)^v := f^*((w_e, v)) \text{ für } e = \{u,v\} \in E(G).$$

Da $qf^* : A \to [p]$, gilt für den skalierten Fluss $f^* : A \to \{0\} \cup \left\{\frac{1}{q}, \ldots, \frac{p}{q}\right\}$. Für jede Kante $e = \{u,v\} \in E(G)$ ist $\phi_q(e)^u + \phi_q(e)^v = f^*((w_e,u)) + f^*((w_e,v)) = 1$. Für ϕ_q gilt $\Delta^-_{\phi_q} = \frac{p}{q}$, da die Kapazität jeder Kante (v,t) mit $v \in V(G)$ im ursprünglichen Netzwerk $\frac{p}{q}$ ist. □

Im Folgenden wird gezeigt, dass BDED mit halbintegralen Zieldichten in polynomieller Zeit auf Maximum Cardinality Matching reduziert werden kann. Die Grundidee besteht darin, mithilfe von Lemma 2.1 eine Maximum Cardinality Matching-Instanz zu erstellen, die die halbintegralen Orientierungen von Kanten und die Dichteanforderungen der Knoten in der BDED-Instanz modelliert. Für jede Kante werden zwei Kantenknoten eingeführt, die mit allen 2τ Kopien ihrer ursprünglichen Endknoten verbunden werden. Eine im Matching enthaltene Kante zwischen den Kantenknoten entspricht dem Löschen der Kante in der BDED-Instanz. Eine im Matching enthaltene Kante eines der Kantenknoten mit einer Kopie ihrer ursprünglichen Endknoten entspricht einer halbintegralen Orientierung der Kante. Für jeden Knoten werden 2τ Kopien erstellt. Dies stellt sicher, dass ein größtmögliches Matching einer zulässigen halbintegralen Orientierung mit maximalem eingehendem Grad τ entspricht. Ein größeres Matching bedeutet weniger gelöschte Kanten in der BDED-Instanz und umgekehrt.

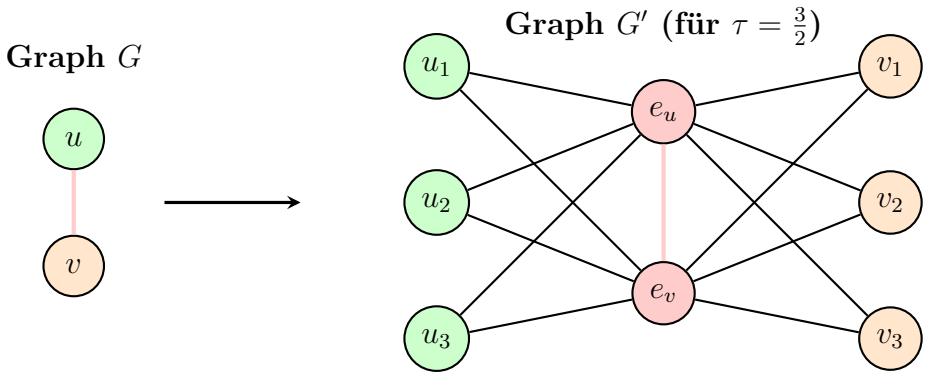


Abbildung 2.2: Konstruktion der Maximum Cardinality Matching-Instanz aus einer BDED-Instanz für eine Kante $e = \{u, v\}$ mit $\tau = \frac{3}{2}$. Jeder Knoten wird durch $2\tau = 3$ Kopien ersetzt, und für jede Kante werden zwei Kantenknoten e_u und e_v eingefügt, die mit allen Kopien der Endknoten verbunden sind.

Theorem 2.2. Bounded-Density Edge Deletion *kann für halbintegrale Zieldichten τ, mit $2\tau \in \mathbb{N}$, in $\mathcal{O}(\tau(|V| + |E|))$ Zeit auf* Maximum Cardinality Matching *reduziert werden.*

Beweis. Sei (G, k, τ) eine BDED-Instanz. Eine äquivalente Maximum Cardinality Matching-Instanz $(G', 2|E(G)| - k)$ wird wie folgt konstruiert: Für jeden Knoten $v \in V(G)$ werden 2τ Knoten (Kopien) $v_1, \ldots, v_{2\tau}$ zu G' hinzugefügt. Für jede Kante $e = \{u, v\} \in E(G)$ werden zwei neue (Kanten-)Knoten e_u und e_v, sowie eine Kante $\{e_u, e_v\}$ und außerdem die Kanten $\bigcup_{i \in [2\tau]} \{\{e_u, u_i\}, \{e_u, v_i\}, \{e_v, u_i\}, \{e_v, v_i\}\}$ zu G' hinzugefügt. Die Knotenmenge $V'(G)$ ist somit durch

$$V'(G) = \bigcup_{e \in E(G)} \{e_u, e_v\} \cup \bigcup_{\substack{v \in V(G) \\ i \in [2\tau]}} \{v_i\}$$

und die Kantenmenge $E'(G)$ durch

$$E'(G) = \bigcup_{e \in E(G)} \left(\{\{e_u, e_v\}\} \cup \bigcup_{i \in [2\tau]} \{\{e_u, u_i\}, \{e_u, v_i\}, \{e_v, u_i\}, \{e_v, v_i\}\} \right)$$

definiert. Abbildung 2.2 illustriert diese Konstruktion. Sie ist in $\mathcal{O}(\tau(|V| + |E|))$ Zeit ausführbar, da für jeden Knoten $v \in V(G)$ genau 2τ Knoten zu G' hinzugefügt werden und für jede Kante $\{u, v\} \in E(G)$ genau 2 Knoten und genau $8\tau + 1$ Kanten hinzugefügt werden. Da BDED für $\tau \geq \frac{|V(G)|-1}{2}$ trivial lösbar ist, kann dies als obere Grenze für τ angenommen werden.

Zu einer BDED-Instanz (G, k, τ) für halbintegrale Zieldichten τ mit $2\tau \in \mathbb{N}$ existiert auch eine halbintegrale Orientierung ϕ, für die gilt: $\Delta^-_\phi \leq \tau$. Bei einer halbintegralen

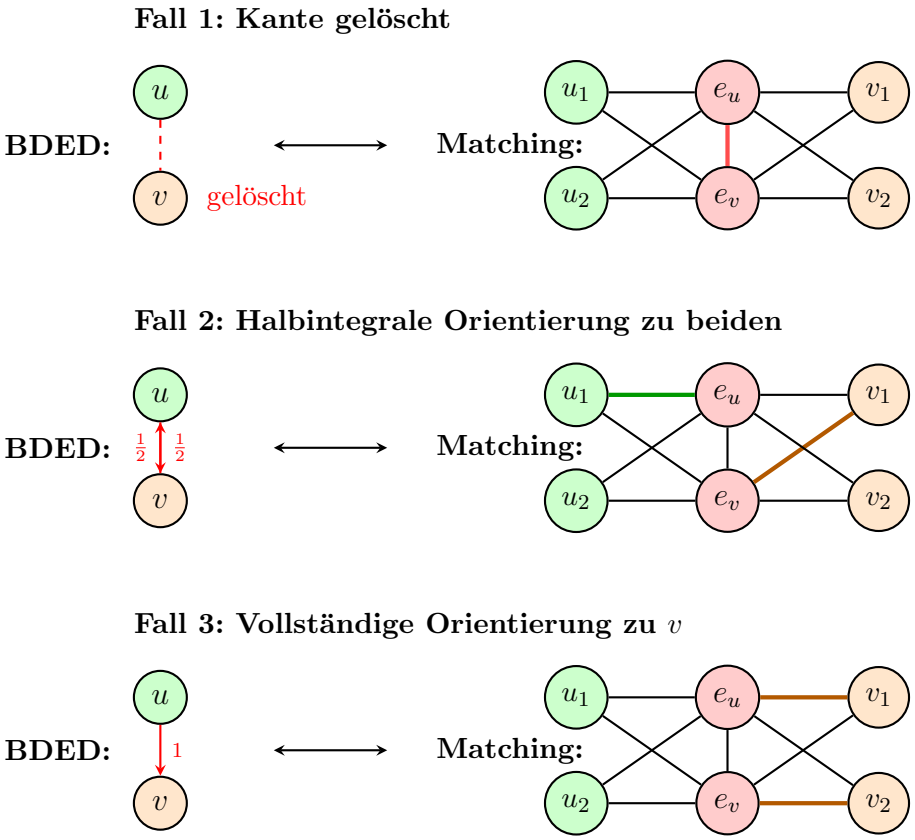


Abbildung 2.3: Interpretation der Matching-Kanten als Orientierungen bzw. Kantenlöschung in der BDED-Instanz.
Fall 1: Kante $\{e_u, e_v\}$ ist im Matching $\Leftrightarrow$ Kante wird gelöscht.
Fall 2: Kantenknoten sind mit Kopien unterschiedlicher Knoten gematcht $\Leftrightarrow$ halbintegrale Orientierung $(\frac{1}{2}, \frac{1}{2})$.
Fall 3: Kantenknoten sind mit Kopien des gleichen Knoten gematcht $\Leftrightarrow$ vollständige Orientierung zu einem Knoten.

Orientierung kann jede Kante entweder vollständig zu einem Knoten orientiert werden oder halb zu beiden Knoten. Dies wird als Matching dadurch modelliert, dass für jede Kante $\{u, v\} \in E(G)$ zwei Knoten e_u und e_v eingefügt werden, die entweder miteinander gematcht werden können (entspricht dem Löschen der Kante in BDED) oder mit den Kopien der Knoten u bzw. v gematcht werden können (entspricht je einer halben Orientierung der Kante zu u bzw. v in BDED). Ein größeres Matching in G' zu finden entspricht somit der Suche nach einer BDED-Lösung mit weniger gelöschten Kanten. Abbildung 2.3 zeigt drei mögliche Matching-Szenarien für eine Kante und deren Interpretation.

"$\Rightarrow$": Angenommen F ist eine Lösung für die BDED-Instanz (G, k, τ). Das bedeutet $|F| \leq k$ und es existiert nach Lemma 1.1 eine Orientierung ϕ für $G - F$ mit $\Delta^-_\phi \leq \tau$. Nach Lemma 2.1 existiert eine halbintegrale Orientierung ϕ' für $G - F$ mit $\Delta^-_{\phi'} \leq \tau$.

Daraus kann ein größtmögliches Matching M mit mindestens $2|E(G)| - k$ Kanten in G' konstruiert werden: Für jede Kante $\{u, v\} \in F$ wird die Kante $\{e_u, e_v\}$ in M aufgenommen. Für jede Kante $\{u, v\} \in E(G - F)$ wird M wie folgt erweitert:

Falls $\phi'(\{u, v\})^u = 1$, füge $\{e_u, u_i\}$ und $\{e_v, u_j\}$ für ungematchte u_i und u_j in M ein.
Falls $\phi'(\{u, v\})^u = \frac{1}{2}$, füge $\{e_u, u_i\}$ und $\{e_v, v_j\}$ für ungematchte u_i und v_j in M ein.
Falls $\phi'(\{u, v\})^u = 0$, füge $\{e_u, v_i\}$ und $\{e_v, v_j\}$ für ungematchte v_i und v_j in M ein.

Für einen Knoten $v \in V(G)$ gibt es hierbei stets eine ungematchte Kopie unter den 2τ Kopien von v: Da $\Delta^-_{\phi'} \leq \tau$ ist, kann v höchstens τ Orientierung zugeordnet werden. Hierdurch können höchstens τ Kantenknoten e_v oder e_u der Kanten der Form $\{u, v\} \in E(G)$, mit $u \in N(v)$, mit Kopien von v gematcht werden.

Diese Konstruktion ist die Rückrichtung in Abbildung 2.3. Das Matching M enthält somit für jede Kante in $E(G - F)$ genau zwei Kanten und für jede Kante in F genau eine Kante. Es gilt $|M| = 2|E(G - F)| + |F| = 2|E(G)| - |F| \geq 2|E(G)| - k$. Somit ist gezeigt, dass wenn eine zulässige BDED-Lösung F mit $|F| \leq k$ existiert, dann ein größtmögliches Matching M der Größe $2|E(G)| - k$ in G' konstruiert werden kann.

"⇐": Sei M ein größtmögliches Matching der Maximum Cardinality Matching-Instanz $(G', 2|E(G)| - k)$ mit $|M| \geq 2|E(G)| - k$. Hieraus kann wie folgt eine Lösung für die BDED-Instanz konstruiert werden: Jede Kante $\{e_u, e_v\} \in E(G')$ im Matching M entspricht einer Kante $\{u, v\} \in E(G)$ im ursprünglichen Graphen G.

Im Fall, dass nur eine Kante zwischen e_u oder e_v und einer Kopie eines Endknotens im Matching ist, wird diese durch die Kante $\{e_u, e_v\}$ ersetzt, ohne die Größe des Matchings zu verringern.

Somit sind alle Kantenknoten e_u und e_v im Matching M entweder miteinander oder mit Kopien ihrer ursprünglichen Endknoten gematcht. Andere Kombinationen sind nicht möglich, da M ein größtmögliches Matching ist, wodurch die Knoten e_v und e_u entweder jeweils mit einem unterschiedlichen Knoten u_i oder v_j gematcht sind, oder miteinander. Da das Matching M mindestens $2|E(G)| - k$ Kanten enthalten muss und weil jede Kante mindestens einen Endpunkt der Form e_u oder e_v hat, ist die Anzahl der Kanten der Form $\{e_u, e_v\} \in E(G')$ im Matching M höchstens k.

Sei F nun die Menge aller Kanten $\{u, v\}$ aus G, sodass $\{e_u, e_v\}$ im Matching M enthalten ist, also $F := \{\{u, v\} \in E(G) \mid \{e_u, e_v\} \in M\}$. Da das Matching M höchstens k Kanten der Form $\{e_u, e_v\} \in E(G')$ enthält, ist die Kantenlöschmenge F in der BDED-Instanz ebenfalls $|F| \leq k$.

Eine halbintegrale Orientierung ϕ für $G - F$ wird für $i, j \in [2\tau]$ wie folgt definiert:

$$\phi(\{u, v\})^u = \begin{cases} 1, & \text{falls } \{e_u, u_i\}, \{e_v, u_j\} \in M \text{ mit } i \neq j, \\ 0, & \text{falls } \{e_u, v_i\}, \{e_v, v_j\} \in M \text{ mit } i \neq j, \\ \frac{1}{2}, & \text{falls } \{e_u, v_i\}, \{e_v, u_j\} \in M \text{ oder } \{e_u, u_i\}, \{e_v, v_j\} \in M. \end{cases}$$

Zwei Kanten der Form $\{e_u, u_i\}$ oder $\{e_u, v_i\}$, sowie $\{e_v, u_i\}$ oder $\{e_v, v_i\}$ im Matching entsprechen einer halbintegralen Orientierung der Kante $\{u, v\}$ in G. Die Kante $\{e_u, e_v\}$ im Matching entspricht dem Löschen der Kante $\{u, v\}$ in G.

Für die halbintegrale Orientierung ϕ gilt: $\Delta^-_\phi \leq \tau$, da jeder Knoten $v \in V(G)$ in G' genau 2τ Kopien hat und somit maximal 2τ Kantenknoten mit diesen Kopien gematcht

werden können, sodass v höchstens τ Orientierung zugeordnet werden kann. Somit ist gezeigt, dass wenn ein größtmögliches Matching M auf G' mindestens $2|E(G)|-k$ Kanten enthält, eine zulässige BDED-Lösung F mit $|F| \leq k$ konstruiert werden kann.

Damit ist die Reduktion von BDED für halbintegrale Zieldichten auf Maximum Cardinality Matching gezeigt. □

3 Daten-Reduktionsregeln

In diesem Kapitel werden verschiedene Reduktionsregeln für das BDED-Problem mit halbintegralen Zieldichten vorgestellt. Diese Regeln können verwendet werden, um die Größe einer solchen Instanz zu reduzieren, ohne die Existenz einer zulässigen Lösung zu beeinflussen. Da die Maximum Cardinality Matching-Instanz für jeden Knoten v im ursprünglichen Graphen G 2τ Kopien erstellt, ist die Reduzierung der Knoten im ursprünglichen Graphen besonders wichtig. Die nachfolgenden Regeln verwenden Lemma 2.1 für halbintegrale Orientierungen.

Sei eine BDED-Instanz durch einen ungerichteten Graphen $G = (V, E)$, $k \in \mathbb{N}$ und eine halbintegrale Zieldichte τ mit $2\tau \in \mathbb{N}$ gegeben. Die Reduktionsregeln können auch für allgemeine $\tau \in \mathbb{Q}^+$ angewendet werden, wobei sie noch wichtiger sind, da BDED nach Bentert u. a. [Ben+26] allgemein NP-schwer ist.

3.1 Knotengrad

Die folgende Reduktionsregel entfernt Knoten, deren Grad kleiner oder gleich τ ist, da diese Knoten keine Rolle bei der Existenz einer zulässigen Lösung spielen. Sie ist in Abbildung 3.1 illustriert.

Reduktionsregel 3.1.1. *Lösche alle Knoten $v \in V(G)$ mit Grad $\deg(v) \leq \tau$ und alle an diese Knoten angrenzenden Kanten.*

Beweis. Da v höchstens τ Nachbarn hat, kann jede Kante, die an v angrenzt, vollständig zu v orientiert werden, ohne die Beschränkung $\leq \tau$ zu verletzen. □

Die erschöpfende Anwendung dieser Reduktionsregel kann in $\mathcal{O}(|V|+|E|)$ Zeit durchgeführt werden. Dies kann durch eine Breitensuche durch den Graphen erreicht werden, bei der jeder Knoten und jede Kante genau einmal besucht wird. Beim Besuch eines Knotens wird dessen Grad überprüft, und wenn dieser kleiner oder gleich τ ist, wird der Knoten entfernt und die Grade seiner Nachbarn aktualisiert. Dies kann in linearer Zeit durchgeführt werden, da jeder Knoten und jede Kante nur einmal betrachtet wird.

3.2 Kanten zwischen Knoten mit kleinem Grad

Die folgende Reduktionsregel entfernt beide Knoten einer Kante, bei der die Knoten einen Grad kleiner oder gleich $\tau + \frac{1}{2}$ haben. Sie ist in Abbildung 3.2 illustriert.

Reduktionsregel 3.2.1. *Lösche die Knoten u und v, mit $\{u, v\} \in E(G)$, sowie alle angrenzenden Kanten, falls $\deg(u) \leq \tau + \frac{1}{2}$ und $\deg(v) \leq \tau + \frac{1}{2}$ gilt.*

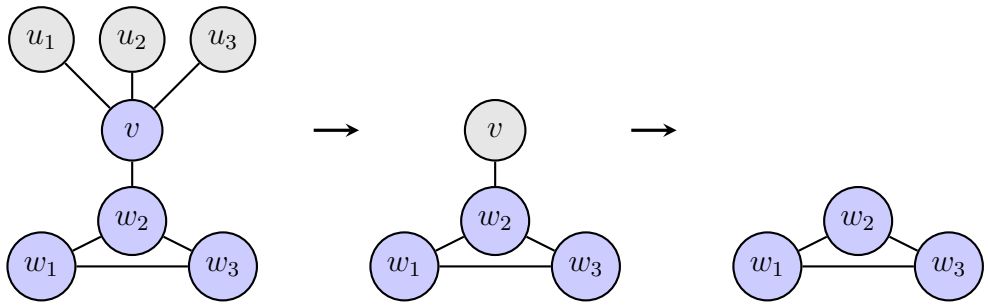


Abbildung 3.1: Reduktionsregel für Knoten mit Grad $\leq \tau$ (hier $\tau = \frac{3}{2}$). Sie wird eingesetzt, bis keine Knoten mit Grad $\leq \tau$ mehr existieren.

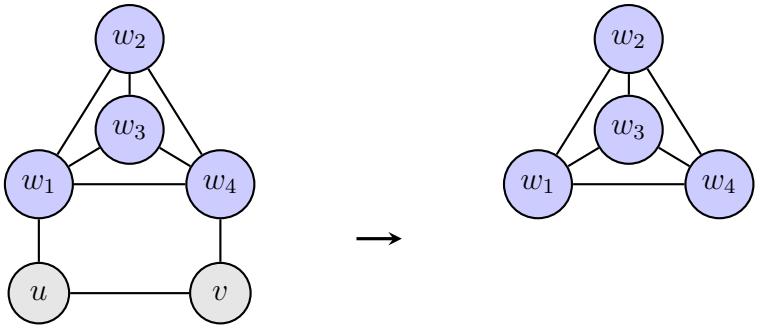


Abbildung 3.2: Reduktionsregel für Kanten zwischen Knoten mit Grad $\leq \tau + \frac{1}{2}$ (hier $\tau = \frac{3}{2}$). Die Knoten u und v haben beide Grad $2 \leq \tau + \frac{1}{2}$ und können entfernt werden.

Beweis. Da $\deg(u) \leq \tau + \frac{1}{2}$ und $\deg(v) \leq \tau + \frac{1}{2}$, kann die Kante $\{u, v\}$ mit einer $(\frac{1}{2}, \frac{1}{2})$-Orientierung orientiert werden, ohne die Beschränkung $\leq \tau$ zu verletzen. Alle anderen Kanten, die an u oder v angrenzen, können vollständig zu u bzw. v orientiert werden, ohne die Beschränkung $\leq \tau$ zu verletzen. □

Diese Reduktionsregel bietet keinen Vorteil für ganzzahlige Zieldichten τ, da in diesem Fall die vorherige Regel bereits alle Knoten mit Grad kleiner oder gleich τ entfernt. Sie sollte nur für nicht-ganzzahlige Zieldichten angewendet werden.

Die erschöpfende Anwendung dieser Reduktionsregel kann in $\mathcal{O}(|V|^2 + |V||E|)$ Zeit durchgeführt werden. Der Unterschied zu der vorherigen Regel liegt darin, dass hier Kanten mehrmals überprüft werden müssen, da die Entfernung von Knoten den Grad anderer Knoten beeinflussen kann. Es wird eine Kandidatenliste aller Knoten mit Grad kleiner oder gleich $\tau + \frac{1}{2}$ abgearbeitet, wobei für jeden Kandidaten alle angrenzenden Knoten überprüft werden. Wird eine Kante gefunden, die die Bedingung erfüllt, werden beide Knoten entfernt. Hierbei muss jedoch auch die Kandidatenliste aktualisiert werden, da die Entfernung von Knoten den Grad anderer Knoten beeinflusst. Ein Knoten kann somit mehrmals, aber höchstens $O(|V|)$-mal überprüft werden, was zu einer Gesamtzeit von $\mathcal{O}(|V|^2 + |V||E|)$ führt.

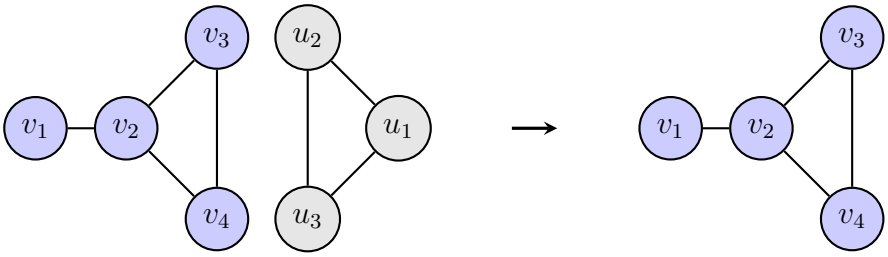


Abbildung 3.3: Reduktionsregel für Zusammenhangskomponenten mit maximalem Grad $\leq 2\tau$ (hier $\tau = 1$). Die Zusammenhangskomponente mit den Knoten u_1, u_2 und u_3 hat maximalen Grad $2 \leq 2\tau$ und kann daher entfernt werden.

3.3 Zusammenhangskomponenten

Die folgende Reduktionsregel entfernt Zusammenhangskomponenten mit maximalem Knotengrad kleiner oder gleich 2τ, da diese Komponenten keine Rolle bei der Existenz einer zulässigen Lösung spielen. Sie ist in Abbildung 3.3 illustriert.

Reduktionsregel 3.3.1. *Sei $G' = (V', E')$ eine Zusammenhangskomponente in G mit maximalem Knotengrad $\leq 2\tau$. Dann können alle $v \in V'$, sowie alle Kanten in E' entfernt werden.*

Beweis. Da G' eine Zusammenhangskomponente ist, gibt es keine Kanten zwischen Knoten in V' und Knoten in $V \setminus V'$. Somit beeinflussen die Knoten in V' die Orientierung und Kantenlöschmenge in den anderen Zusammenhangskomponenten nicht. Da der maximale Knotengrad in G' höchstens 2τ ist, kann jede Kante in E' mit einer $(\frac{1}{2}, \frac{1}{2})$-Orientierungen orientiert werden, sodass kein Knoten in V' mehr eingehende Orientierung als τ hat. □

Diese Reduktionsregel ist effektiver, wenn zuerst die Knotengradregel und Regel für Kanten zwischen Knoten mit kleinem Grad vollständig angewendet wird, da dadurch der maximale Knotengrad in den Zusammenhangskomponenten weiter reduziert werden kann. Im Graphen von Abbildung 3.1 könnte vor Anwendung dieser Regel nichts entfernt werden, da $\deg(v) = 4 > 3 = 2\tau$. Nach Anwendung der Knotengrad-Reduktionsregel kann jedoch die restliche Zusammenhangskomponente mit den Knoten w_1, w_2 und w_3 entfernt werden.

Die Anwendung der Reduktionsregel über Zusammenhangskomponenten kann mittels Knotenliste und Breitensuche in $O(|V| + |E|)$ Zeit durchgeführt werden. Hierbei werden alle Knoten mit Grad $\leq 2\tau$ in einer Liste gesammelt. Für jeden dieser Knoten wird eine Breitensuche durchgeführt, um die gesamte Zusammenhangskomponente zu finden. Während der Suche wird der maximale Knotengrad in der Zusammenhangskomponente verfolgt. Wenn der maximale Knotengrad kleiner als 2τ ist, werden alle Knoten in der Zusammenhangskomponente entfernt.

Mittels gewichteten Union-Find mit Pfadteilung kann die gesamte Reduktionsregel in $\mathcal{O}(|V|\tau \cdot \alpha(|V|))$ Zeit durchgeführt werden, wobei α die inverse Ackermann-Funktion

ist. Hierbei wird für jeden Knoten mit Grad $\leq 2\tau$ eine Union-Operation mit allen Nachbarknoten durchgeführt. Dabei wird der maximale Knotengrad in der Zusammenhangskomponente verfolgt. Nach der Durchführung aller Union-Operationen können alle Zusammenhangskomponenten mit maximalem Knotengrad $\leq 2\tau$ entfernt werden. Diese Laufzeit hat den Vorteil, dass nur über höchstens $|V| \cdot \tau$ Kanten iteriert werden muss, da jeder Knoten höchstens 2τ Nachbarn hat. Für kleinere Werte von τ und dichte Graphen ist dies in der Praxis schneller als die Breitensuche. Die Zieldichte τ ist generell durch $\frac{n-1}{2}$ nach oben beschränkt, da dies die maximale Dichte eines ungerichteten Graphen mit n Knoten ist.

Des Weiteren kann für jede nicht gelöschte Zusammenhangskomponente die Maximum Cardinality Matching-Instanz einzeln konstruiert und gelöst werden. Die resultierenden Matchings müssen dann zusammengeführt werden, um eine Lösung für die gesamte Instanz zu erhalten. Dies kann zu weiteren Laufzeitverbesserungen führen, da mehrere kleinere Instanzen, anstatt einer großen Instanz gelöst werden.

Zudem kann die Reduktionsregel optimiert werden, wenn der maximale oder minimale Knotengrad bekannt ist. Wenn der maximale Knotengrad im Graphen kleiner oder gleich 2τ ist, kann der gesamte Graph in $O(1)$ Zeit entfernt werden. Wenn der minimale Knotengrad im Graphen größer als 2τ ist, kann die Reduktionsregel übersprungen werden, da keine Zusammenhangskomponente entfernt werden kann.

3.4 Kreise aus Knoten mit kleinem Grad

Die folgende Reduktionsregel entfernt die Knoten eines Kreises, deren Grad kleiner oder gleich $\tau + 1$ ist. Sie ist in Abbildung 3.4 illustriert.

Reduktionsregel 3.4.1. *Seien* $\{v_1, ..., v_n\}$ *die Knoten und* $\bigcup_{i \in [n-1]} \{\{v_i, v_{i+1}\}\} \cup \{\{v_1, v_n\}\}$ *die Kanten eines Kreises, wobei* $\deg(v_i) \leq \tau + 1$ *für alle* $i \in [n]$. *Lösche die Knoten* $v_1, ..., v_n$ *sowie alle an diese Knoten angrenzenden Kanten.*

Beweis. Da $\deg(v_i) \leq \tau + 1$ für alle $i \in [n]$, können die Kanten $\{v_i, v_{i+1}\}$ mit einer $(\frac{1}{2}, \frac{1}{2})$-Orientierung orientiert werden, sodass jeder Knoten v_i genau eine eingehende Orientierung von diesen Kanten erhält. Es kann höchstens je $\tau - 1$ weitere Kanten zwischen Knoten außerhalb und jedem Knoten v_i geben, da sonst die Beschränkung $\deg(v_i) \leq \tau + 1$ verletzt würde. Demnach können alle diese Kanten zu dem jeweiligen Knoten auf dem Kreis orientiert werden, sodass dieser zusätzlich maximal $\tau - 1$ Orientierung erhält. Gesamt ergibt dies eine maximale Orientierung von τ für jeden Knoten auf dem Kreis. Da die Grenze von τ für die Orientierung nicht überschritten wird, und alle anderen Kanten vollständig zu diesen Knoten orientiert werden, können die Knoten $v_1, ..., v_n$ und alle angrenzenden Kanten entfernt werden, ohne die Existenz einer zulässigen Lösung zu beeinflussen. □

Die erschöpfende Anwendung dieser Reduktionsregel kann in $\mathcal{O}(|V|^2 + |V||E|)$ Zeit durchgeführt werden.

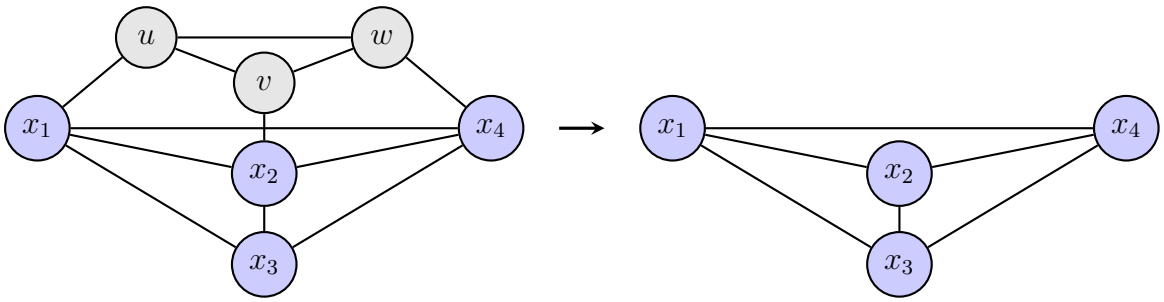


Abbildung 3.4: Reduktionsregel für Kreise mit Grad $\leq \tau+1$ (hier $\tau = 2$). Die Knoten u, v und w bilden einen Kreis mit maximalem Knotengrad $3 \leq \tau + 1$ und können entfernt werden.

Es wird eine initiale Kandidatenliste aller Knoten mit Grad kleiner oder gleich $\tau + 1$ erstellt. Dies ist in $\mathcal{O}(|V|)$ Zeit möglich.

Es wird der induzierte Subgraph G' aller Knoten mit Grad $\leq \tau + 1$ betrachtet. Dieser kann mittels der Kandidatenliste in $\mathcal{O}(|V| + |E|)$ Zeit erstellt werden.

In G' werden iterativ alle Knoten mit Grad kleiner als 2 im Bezug auf G' entfernt. Dies ist wie in der Knotengrad-Reduktionsregel beschrieben in $\mathcal{O}(|V|+|E|)$ Zeit möglich. Alle verbleibenden Knoten liegen auf einem Kreis und werden zusammen mit ihren anliegenden Kanten entfernt. Hierbei werden die Grade der Nachbarknoten in G aktualisiert, wobei eine neue Kandidatenliste durch Knoten, deren Grad auf $\tau + 1$ oder darunter gesunken ist, erstellt wird.

Nun wird aus der Kandidatenliste erneut der induzierte Subgraph aller Knoten mit Grad kleiner oder gleich $\tau + 1$ erstellt und der Prozess wiederholt, bis keine Knoten mehr entfernt werden. Dies kann höchstens $|V|$-mal wiederholt werden, da in jedem Schritt mindestens ein Knoten entfernt wird. So kann diese Reduktionsregel in $\mathcal{O}(|V|^2 + |V||E|)$ Gesamtlaufzeit erschöpfend angewendet werden.

4 Experimentelle Evaluation

4.1 Implementierung

In diesem Kapitel werden einige Aspekte der Implementierung der zuvor gezeigten Reduktion von BDED auf Maximum Cardinality Matching beschrieben, die für die experimentelle Evaluation verwendet wurde. Als Eingabe benötigt sie einen ungerichteten Graphen G in einem Dimacs-Format ähnlichen Format und eine halbintegrale Zieldichte τ. Ausgegeben wird das kleinste k, für das eine zulässige BDED-Lösung existiert, sowie die entsprechende Kantenlöschmenge F.

Die Implementierung verwendet drei der in Kapitel 3 beschriebenen Reduktionsregeln, um die Größe der Eingabeinstanz zu reduzieren, bevor die Maximum Cardinality Matching-Instanz konstruiert wird. Hierbei werden die Regeln für den Knotengrad und Kanten zwischen Knoten mit kleinem Grad so lange angewendet, bis sie nicht mehr anwendbar sind. Anschließend wird die Zusammenhangskomponenten-Regel einmalig angewendet. In dieser Implementierung werden die übrig gebliebenen Zusammenhangskomponenten nicht einzeln, sondern zusammen weiter gelöst.

Danach wird die Instanz wie in Kapitel 2 beschrieben in eine Maximum Cardinality Matching-Instanz umgewandelt. Da hierbei für jeden Knoten v im ursprünglichen Graphen G 2τ Kopien erstellt werden, ist die Reduzierung der Knoten im Graphen durch die Reduktionsregeln besonders wichtig, um die Größe nicht zu groß werden zu lassen. Anschließend wird diese mit dem in Golang implementierten Maximum Cardinality Matching-Algorithmus von Kececioglu und Pecqueur [KP98] gelöst, der auf dem Algorithmus von Edmonds [Edm65] basiert.

Verfügbar ist die in der Programmiersprache Golang geschriebene Implementierung in GitLab unter `https://git.tu-berlin.de/akt/theses/ba-niklas-keitsch`. In diesem Repository sind auch die Skripte, um die Testinstanzen im richtigen Format zu erhalten und die Ergebnisstabellen zu erstellen. Die Dokumentation zur Nutzung der Implementierung befindet sich in der README.md-Datei.

4.2 Tests und Validierung

Einerseits wurde die Implementierung mit Modultests validiert. Andererseits wurden auf den nachfolgend aufgeführten Testinstanzen, die maximale Dichte aller Subgraphen mithilfe des Algorithmus von Goldberg [Gol84] berechnet. Um diese zu validieren, wurde die maximale Dichte aller Subgraphen auch mittels der wesentlich weniger effizienten LP-Formulierung von Charikar [Cha00] für ungefähr die Hälfte der Testinstanzen berechnet. In allen Fällen stimmten die Ergebnisse vom Goldberg-Algorithmus und der

LP-Formulierung überein.

Ebenfalls wurde die maximale Dichte aller Subgraphen nach Entfernen der Kantenlöschmenge aus der BDED-Lösung auf dem reduzierten Graphen und dem initialen Graphen berechnet. Die Ergebnisse vom Goldberg-Algorithmus und dem LP stimmten in allen Fällen überein. Für τ-Werte größer als die maximale Dichte des initialen Graphen wurde stets eine leere Kantenlöschmenge zurückgegeben. Zudem waren die maximalen Dichten aller Subgraphen nach Entfernen von F in den BDED-Lösungen kleiner oder gleich dem vorgegebenen Schwellenwert τ, unabhängig davon, ob die Dichte auf dem reduzierten oder initialen Graphen berechnet wurde.

4.3 Testdaten

Verwendet wurden in der experimentellen Evaluation 83 Testinstanzen aus dem SNAP-Datensatz [LK14]. Im Anhang 6.1 sind diese Instanzen, sowie die Anzahl der Knoten, Kanten und die maximale Dichte aller Subgraphen nach Knotenanzahl sortiert aufgelistet. Diese Instanzen wurden in fünf Kategorien eingeteilt:

- **Netzwerktopologie:** Netzwerktopologien, z.B. Internet-Router-Verbindungen.

  Sie beinhaltet 27 Graphen mit zwischen $6{,}30 \cdot 10^3$ und $6{,}26 \cdot 10^4$ Knoten, sowie zwischen $2{,}08 \cdot 10^4$ und $1{,}48 \cdot 10^5$ Kanten. Der dichteste Subgraph hat eine Dichte von 4,21 bis 25,80.

- **Räumlich:** Räumliche Netzwerke, z.B. Straßennetzwerke.

  Sie beinhaltet drei Graphen mit zwischen $1{,}09 \cdot 10^6$ und $1{,}97 \cdot 10^6$ Knoten, sowie zwischen $1{,}54 \cdot 10^6$ und $2{,}77 \cdot 10^6$ Kanten. Der dichteste Subgraph hat eine Dichte von 1,88 bis 2,08.

- **Sozial:** Soziale Netzwerke, z.B. Facebook, Twitter, etc.

  Sie beinhaltet 43 Graphen mit zwischen $1{,}01 \cdot 10^3$ und $1{,}13 \cdot 10^6$ Knoten, sowie zwischen $1{,}45 \cdot 10^4$ und $2{,}99 \cdot 10^6$ Kanten. Der dichteste Subgraph hat eine Dichte von 4,80 bis 119,00.

- **Web:** Webgraphen, z.B. Hyperlinks zwischen Webseiten.

  Sie beinhaltet sechs Graphen mit zwischen $2{,}28 \cdot 10^3$ und $8{,}76 \cdot 10^5$ Knoten, sowie zwischen $3{,}14 \cdot 10^4$ und $4{,}32 \cdot 10^6$ Kanten. Der dichteste Subgraph hat eine Dichte von 28,04 bis 136,89.

- **Wissenschaftlich:** Kollaborationsnetzwerke, z.B. Co-Autorennetzwerke.

  Sie beinhaltet vier Graphen mit zwischen $2{,}78 \cdot 10^4$ und $2{,}65 \cdot 10^5$ Knoten, sowie zwischen $1{,}84 \cdot 10^5$ und $4{,}21 \cdot 10^5$ Kanten. Der dichteste Subgraph hat eine Dichte von 25,28 bis 37,34.

4.4 Ergebnisse

In diesem Abschnitt werden die Ergebnisse der experimentellen Evaluation präsentiert und diskutiert. Hierbei wird vor allem der Einfluss der verschiedenen Reduktionsregeln im Zusammenhang mit der halbintegralen Zieldichte τ auf die Gesamtlaufzeit betrachtet. Für τ werden Werte der Form $\frac{3^i}{2}$, mit $i \in \{0, 1, 2, 3, 4, 5\}$, betrachtet, da sie einen breiten Bereich abdecken, der ähnlich der maximalen Subgraphendichte der Testinstanzen ist. Sie liegen gleichzeitig im interessanteren nicht ganzzahligen Bereich, da es von Bentert u. a. [Ben+26] schon einen effizienten Algorithmus für ganzzahlige Werte von τ gibt. Außerdem werden die Unterschiede im Bezug auf die verschiedenen Kategorien von Graphen analysiert. Die Gesamtlaufzeit umfasst hierbei die Zeit für das Einlesen der Instanz, die Anwendung der Reduktionsregeln, die Konstruktion der Maximum Cardinality Matching-Instanz, sowie die Zeit für das Lösen dieser.

Die Evaluation wurde unter Windows 11 mit einem Intel Core Ultra 9-285H Prozessor, sowie 32 GB LPDDR5X-8533MT/s Arbeitsspeicher durchgeführt. Verwendet wurde die aktuellste Golang-Version 1.26.0. Es wurde kein Zeitlimit gesetzt. Alle Instanzen wurden vollständig gelöst.

4.4.1 Gesamtlaufzeit

Die Gesamtlaufzeiten für BDED mit den genannten Reduktionsregeln auf den Testinstanzen wurden in Abbildung 4.1 für die verschiedenen Graph-Kategorien sowie Zieldichten τ der Form $\frac{3^i}{2}$, mit $i \in \{0, 1, 2, 3, 4, 5\}$, dargestellt. In Abbildung 4.2 ist die Anzahl der Knoten nach Anwendung der Reduktionsregeln für dieselben Fälle visualisiert. So können die Auswirkungen der Reduktionsregeln auf die Instanzgröße und die Gesamtlaufzeit besser verstanden werden.

Allgemein ist in den Kategorien Sozial, Netzwerktopologie, Web und Wissenschaftlich zu beobachten, dass die Laufzeit mit steigendem τ zunächst ansteigt, wobei der Anstieg besonders für die Kategorie Web deutlich ist. In allen Kategorien ist ab einem gewissen Wert von τ zudem ein deutlicher Abfall der Laufzeit zu beobachten, der bei der Räumlichen Kategorie sofort eintritt. Die Anzahl der verbleibenden Knoten nach Anwendung der Reduktionsregeln nimmt wie erwartet mit steigendem τ monoton ab. Ab dem selben Wert von τ, für den die Gesamtlaufzeit stark einbricht, werden zumeist alle Knoten durch die Reduktionsregeln entfernt. Die einzelnen Kategorien werden im Folgenden genauer analysiert.

In der Kategorie **Netzwerktopologie** steigt die Laufzeit mit steigendem τ zunächst kaum an, wobei es für ein paar Instanzen bei $\tau = \frac{9}{2}$ einen deutlichen Anstieg gibt, der jedoch für die nachfolgenden Werte von τ nicht erscheint. Für die Zieldichte $\frac{27}{2}$ fällt die Laufzeit für ungefähr die Hälfte der Instanzen deutlich ab, da auf diesen bereits alle Knoten durch die Reduktionsregeln entfernt werden können. Ab der Zieldichte $\frac{81}{2}$ können für alle Instanzen bereits alle Knoten entfernt werden, wodurch alle Instanzen sehr geringe Laufzeiten aufweisen.

In der Kategorie **Räumlich** gibt es gar keinen Anstieg der Laufzeit. Hier ist die Laufzeit für die Zieldichten $\frac{1}{2}$ und $\frac{3}{2}$ ähnlich, während sie für die höheren Zieldichten

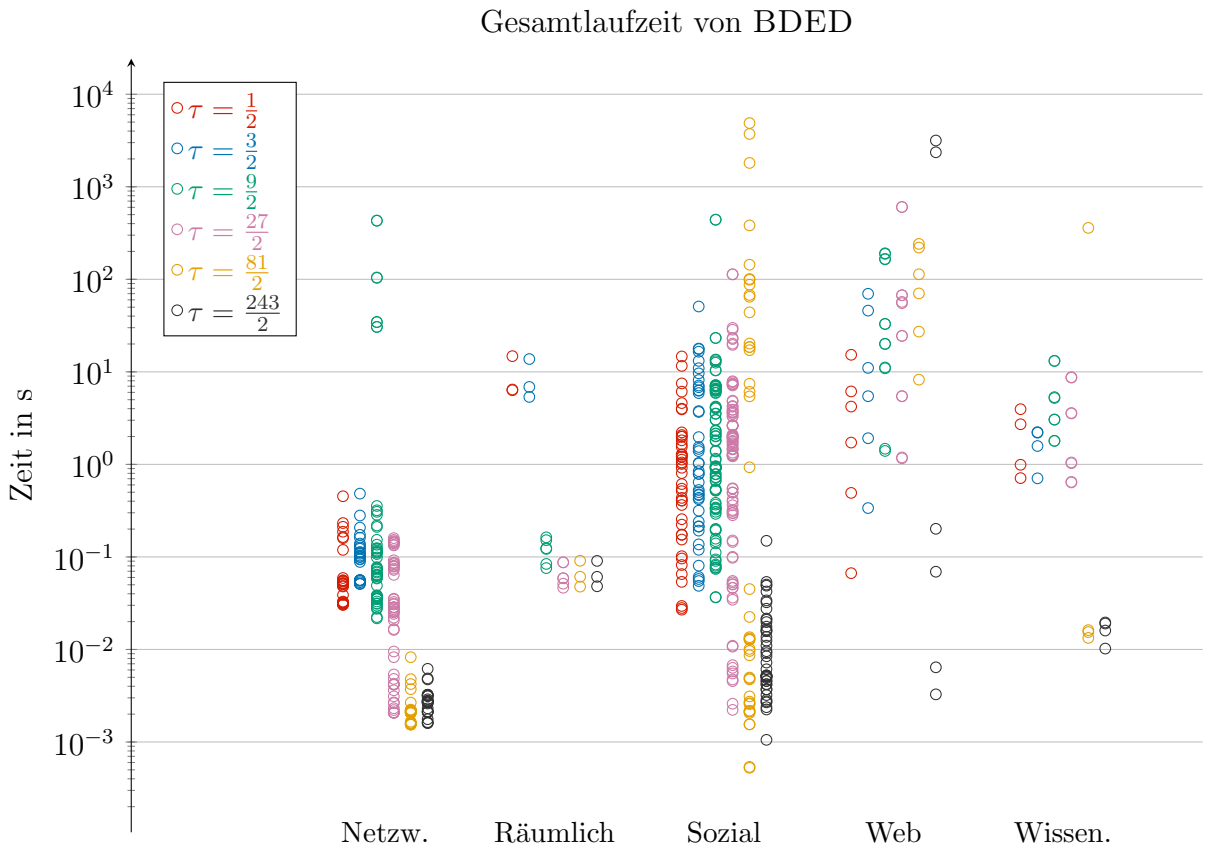


Abbildung 4.1: Gesamtlaufzeit in Sekunden von BDED mit den vorgestellten Reduktionsregeln für die verschiedenen Graph-Kategorien Netzwerktopologie, Räumlich, Sozial, Web und Wissenschaftlich und exponentiell steigenden Zieldichten τ.

deutlich niedriger ist, da alle Knoten durch die Reduktionsregeln entfernt werden können, was auf die geringen maximalen Subgraphendichten der Instanzen zurückzuführen ist.

Die Laufzeit für die Graphen der Kategorie **Sozial** teilt sich für die Zieldichten $\frac{27}{2}$ und $\frac{81}{2}$ in zwei Gruppen auf. Während eine weiter in der Laufzeit ansteigt, fällt die andere in der Laufzeit stark ab. Während die Reduktionsregeln für die eine Gruppe bereits sehr viele bis alle Knoten und Kanten entfernen können, können sie für die andere Gruppe nur wenige entfernen, da einige Instanzen noch zu dicht sind. Für die Zieldichte $\frac{243}{2}$ gibt es hingegen nur noch eine Gruppe, die sehr geringe Laufzeiten aufweist, die jedoch teilweise etwas höher sind, als die abfallende Gruppe für $\frac{81}{2}$. Hier können für alle Instanzen bereits alle Knoten und Kanten entfernt werden, wodurch die Laufzeit des Maximum Cardinality Matching-Algorithmus deutlich reduziert wird. Der teils geringe Anstieg gegenüber der abfallenden Gruppe für $\frac{81}{2}$ könnte durch einen Einfluss von τ auf die Laufzeit der Reduktionsregeln selbst erklärt werden, wie es beispielsweise bei der hier verwendeten Implementierung der Reduktionsregel 3.3.1 über Zusammenhangskomponenten der Fall ist.

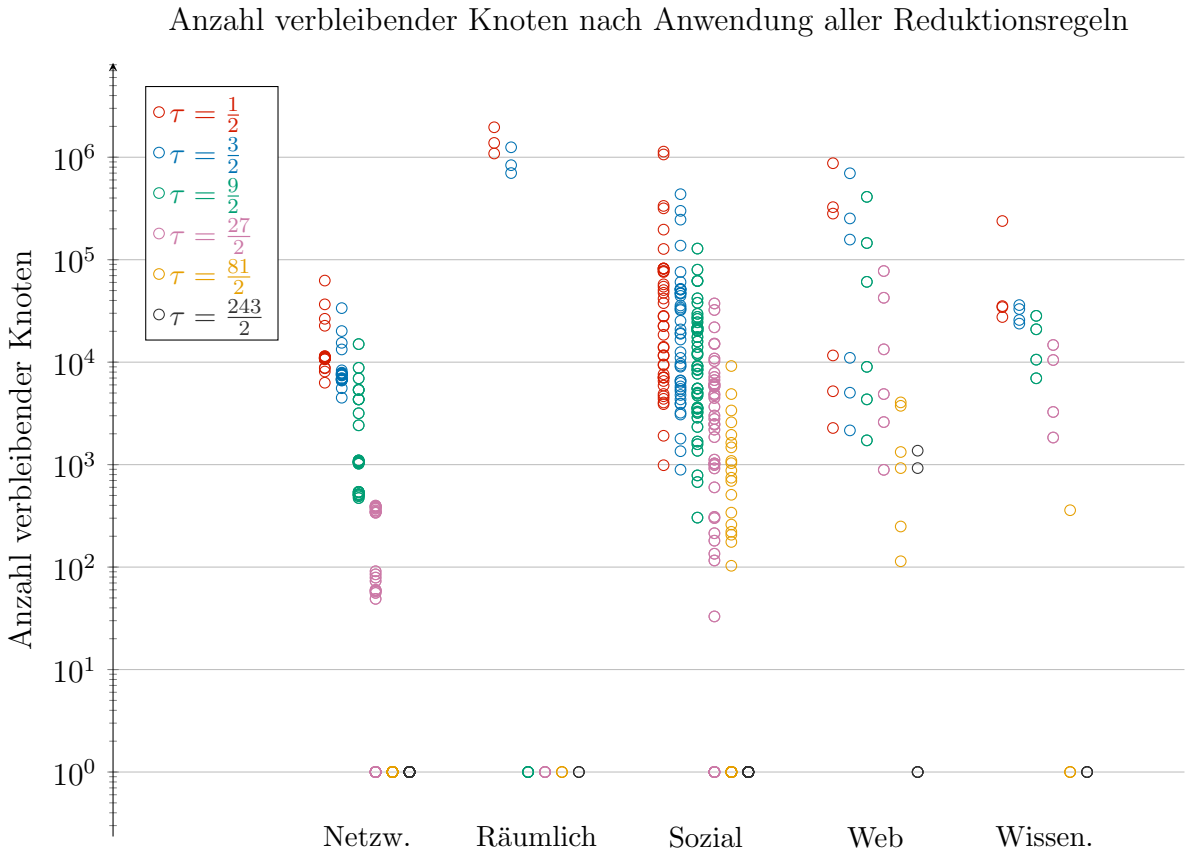


Abbildung 4.2: Anzahl der verbleibenden Knoten vor der Maximum Cardinality Matching-Konstruktion bei BDED mit den vorgestellten Reduktionsregeln für die verschiedenen Graph-Kategorien Netzwerktopologie, Räumlich, Sozial, Web und Wissenschaftlich und exponentiell steigenden Zieldichten τ. Für alle Datenpunkte, die auf der y-Achse den Wert 10^0 haben, wurden alle Knoten entfernt.

Die Graphen der Kategorie **Web** zeigen einen starken Anstieg der Laufzeit mit steigendem τ, der erst bei $\tau = \frac{243}{2}$ für bis auf zwei Instanzen wieder deutlich abfällt. Sie sind die zwei einzigen Instanzen, die auch für $\tau = \frac{243}{2}$ hohe Laufzeiten aufweisen. Es handelt sich um `web-NotreDame` mit einer maximalen Subgraphendichte von 136,89 und `wikipedia_squirrel_musae_squirrel_edges` mit einer maximalen Subgraphendichte von 78,66 und damit die beiden Instanzen mit dem dichtesten Subgraphen in der Kategorie Web. Auf diesen können die Reduktionsregeln selbst für $\tau = \frac{243}{2}$ nicht alle Knoten entfernen, sodass große Maximum Cardinality Matching-Instanzen entstehen, die eine hohe Laufzeit verursachen.

Die Kategorie **Wissenschaftlich** zeigt einen ähnlichen Verlauf, wie die Kategorie Web, wobei der Anstieg mit steigendem τ weniger stark ist und der Abfall der Laufzeit bereits früher eintritt. Hier können jedoch für die Zieldichte $\frac{243}{2}$ alle Knoten entfernt werden.

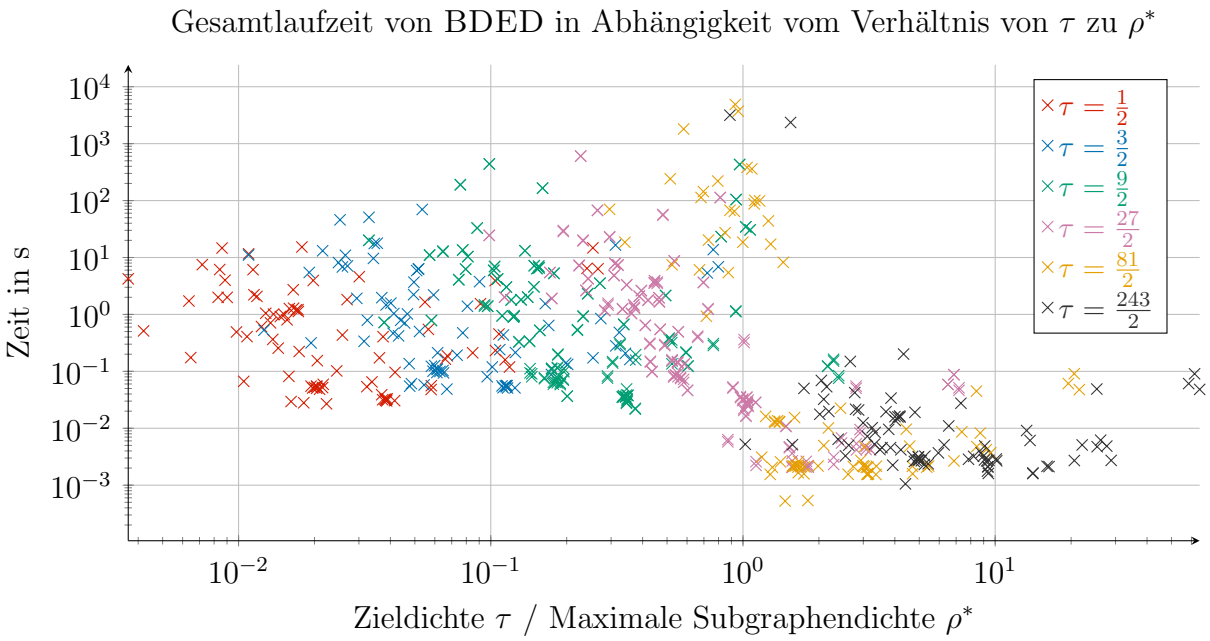


Abbildung 4.3: Gesamtlaufzeit in Sekunden von BDED mit allen vorgestellten Reduktionsregeln und exponentiell steigenden Zieldichten τ in Abhängigkeit vom Verhältnis von τ zur maximalen Subgraphendichte ρ^*.

Die beobachteten Laufzeitverläufe hängen besonders von den verwendeten Reduktionsregeln ab. Sie entfernen Knoten und Kanten aus der Instanz, wodurch sich die Laufzeit des Maximum Cardinality Matching-Algorithmus verringert, bevor die Maximum Cardinality Matching-Instanz in Abhängigkeit von τ konstruiert wird. Für kleinere Werte von τ entfernen die Reduktionsregeln jedoch weniger Knoten und Kanten, da die Regeln durch τ stärker eingeschränkt sind. Da die Matching-Instanz, abhängig von τ, in der Reduktion von BDED zu Maximum Cardinality Matching linear in der Anzahl der Knoten und Kanten im reduzierten Graphen wächst, kann die Laufzeit des Maximum Cardinality Matching-Algorithmus mit steigendem τ zunächst ansteigen. Ab einem gewissen Wert von τ entfernen die Reduktionsregeln jedoch deutlich mehr bis alle Knoten und Kanten, wodurch das Matching auf einer kleineren Instanz schneller berechnet werden kann. Hierfür ist die Verteilung der Knotengrade in den Testinstanzen entscheidend, da die Reduktionsregeln Knoten immer basierend auf ihrem Grad entfernen. Die maximale Subgraphendichte ρ^* gibt hierbei einen Anhaltspunkt, da ein Graph mit hoher maximaler Subgraphendichte zwangsläufig Knoten mit hohem Grad enthält, die von den Reduktionsregeln nicht entfernt werden können. In der Kategorie Räumlich haben die Graphen beispielweise nur eine maximale Subgraphendichte von 2,08, wodurch die Reduktionsregeln bereits für kleinere Werte von τ effektiv sind, während in der Kategorie Web die maximale Subgraphendichte bis zu 136,89 beträgt, wodurch die Reduktionsregeln erst für größere Werte von τ effektiv werden.

In der Abbildung 4.3 und Abbildung 4.4 sind die Gesamtlaufzeit von BDED und die Anzahl der verbleibenden Knoten nach Anwendung aller Reduktionsregeln abhängig

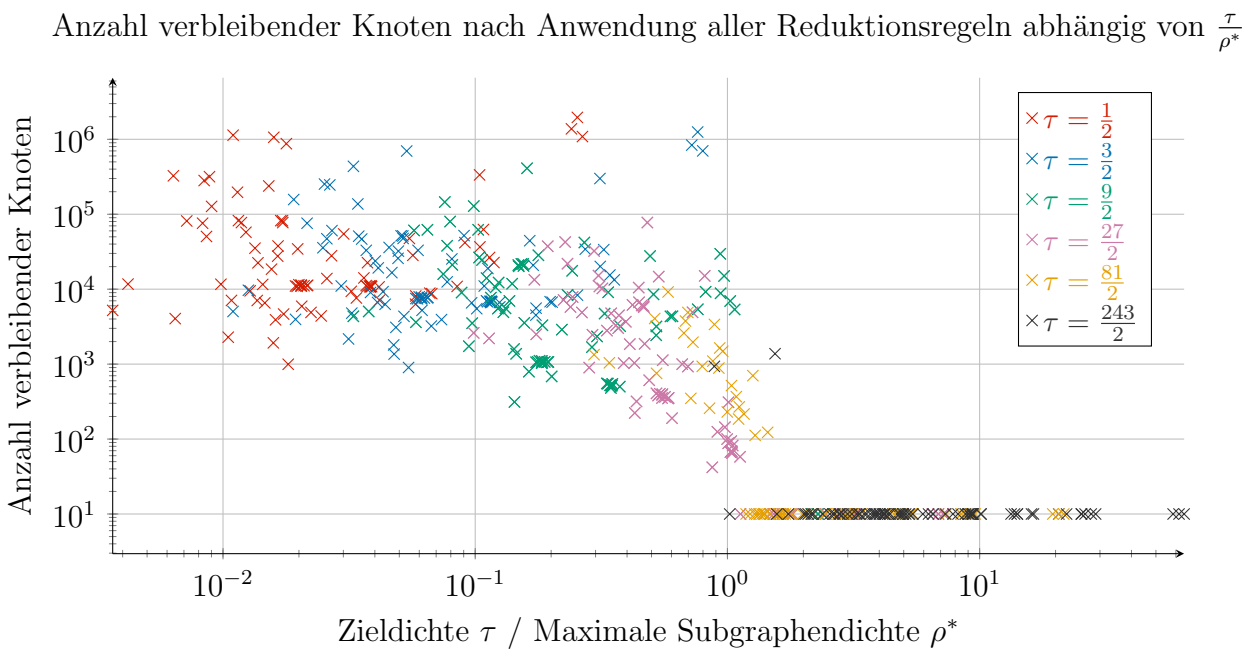


Abbildung 4.4: Anzahl der Knoten nach Anwendung aller Reduktionsregeln vor der Maximum Cardinality Matching-Konstruktion für exponentiell steigende Zieldichten τ in Abhängigkeit vom Verhältnis von τ zur maximalen Subgraphendichte ρ^*. Alle Datenpunkte, die auf der y-Achse den Wert 10^1 haben, entsprechen Instanzen, bei denen alle Knoten durch die Reduktionsregeln entfernt werden konnten. Alle diese Datenpunkte liegen bei einem Verhältnis von τ zu ρ^* von ungefähr 1 oder höher.

vom Verhältnis von der Zieldichte τ zur maximalen Subgraphendichte ρ^* dargestellt. Besonders ist zu beobachten, dass nur für $\frac{\tau}{\rho^*} \geq 1$ die Reduktionsregeln alle Knoten entfernen konnten. Passend dazu ist auch der starke Abfall der Gesamtlaufzeit von BDED für $\frac{\tau}{\rho^*} \geq 1$ zu beobachten. Jedoch gibt es auch einige Instanzen, für die trotz $\frac{\tau}{\rho^*} \geq 1$ nicht alle Knoten entfernt werden konnten. In diesen Fällen bleibt nach den Datenreduktionsregeln eine dichte Instanz übrig, wodurch die Gesamtlaufzeiten dieser hoch sind. Generell ist zu erkennen, dass für $\frac{\tau}{\rho^*} < 1$ die höchsten Gesamtlaufzeiten von BDED mit steigendem Verhältnis von τ zu ρ^* ansteigen, da hier nicht alle Knoten entfernt werden können und die Maximum Cardinality Matching-Instanz mit steigendem τ linear größer wird. Dass es keinen Fall gibt, bei dem für $\frac{\tau}{\rho^*} < 1$ alle Knoten entfernt werden konnten, muss zwangsläufig sein, da BDED sonst eine leere Kantenlöschmenge F zurückgeben würde, aber die Zieldichte τ nicht eingehalten werden würde.

4.4.2 Reduktionsregeln

In diesem Abschnitt werden die Auswirkungen der einzelnen Reduktionsregeln auf die Testinstanzen detaillierter diskutiert. In Abbildung 4.5 sind die Gesamtlaufzeiten von BDED für die verschiedenen Graph-Kategorien und die Zieldichte $\tau = \frac{9}{2}$ dargestellt,

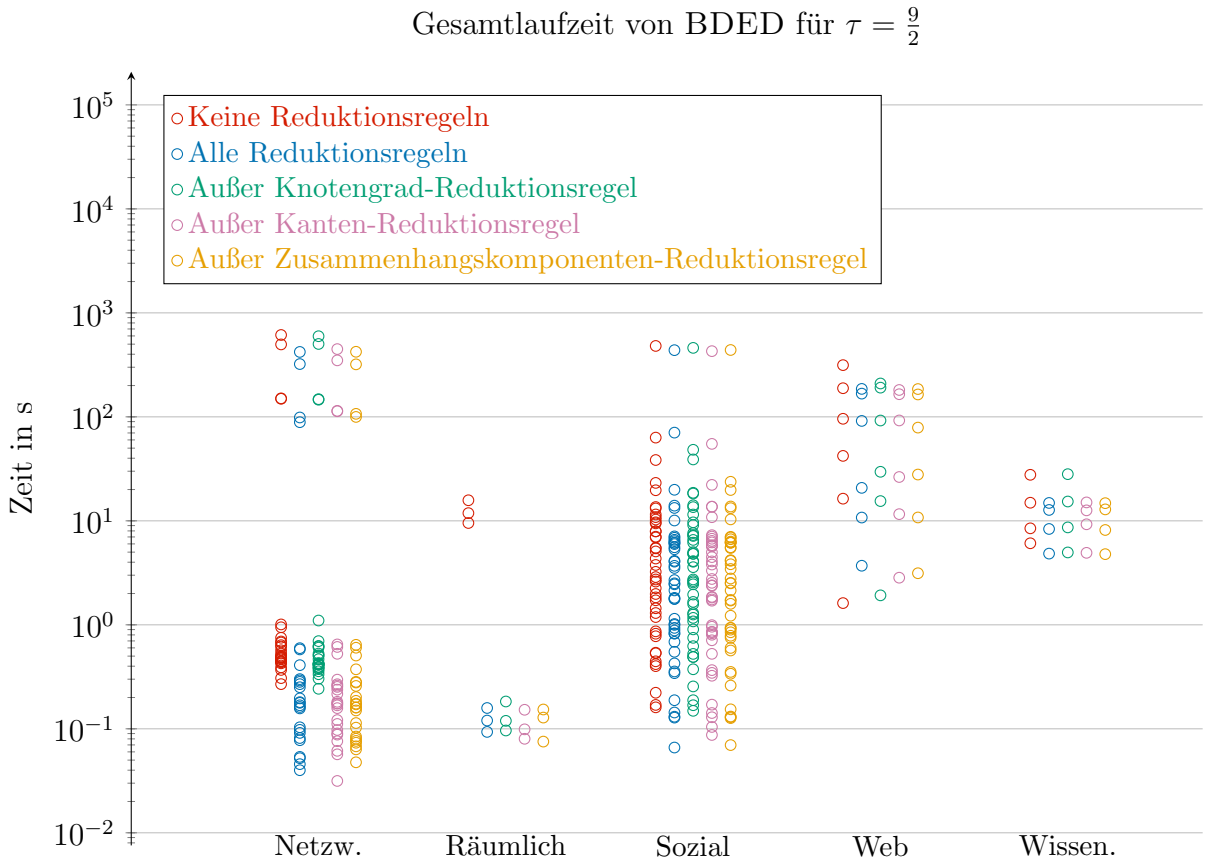


Abbildung 4.5: Gesamtlaufzeit von BDED für die verschiedenen Graph-Kategorien Netzwerktopologie, Räumlich, Sozial, Web und Wissenschaftlich und Zieldichte $\tau = \frac{9}{2}$ in Sekunden, wobei jeweils andere Kombinationen der Reduktionsregeln angewendet werden.

wobei die Auswirkungen verschiedener Kombinationen von Reduktionsregeln untersucht werden. In Abbildung 4.6 ist die Anzahl der Knoten in der resultierenden Maximum Cardinality Matching-Instanz für dieselben Fälle dargestellt. Die Wahl der Zieldichte $\tau = \frac{9}{2}$ wurde getroffen, da hier deutliche Unterschiede in der Gesamtlaufzeit zwischen den verschiedenen Reduktionsregeln zu beobachten sind. Für kleinere Werte von τ sind die Unterschiede in der Gesamtlaufzeit zwischen den verschiedenen Reduktionsregeln, wie im vorherigen Abschnitt beschrieben, weniger deutlich, da nur sehr wenige Knoten entfernt werden können. Hierbei werden die folgenden fünf Konfigurationen der Reduktionsregeln verglichen:

Wenn keine Reduktionsregeln angewendet werden, ist die Laufzeit fast immer am höchsten, da die resultierende Maximum Cardinality Matching-Instanz am größten ist. Nur in der Kategorie Web gibt es einige Fälle, in denen die Laufzeit ohne Reduktionsregeln sogar niedriger ist als mit Anwendung aller Reduktionsregeln. Dies könnte daran liegen, dass die Reduktionsregeln mehr Zeit für die Anwendung benötigen, als die Zeit, die durch das Entfernen der Knoten später eingespart wird. Da keine Knoten ent-

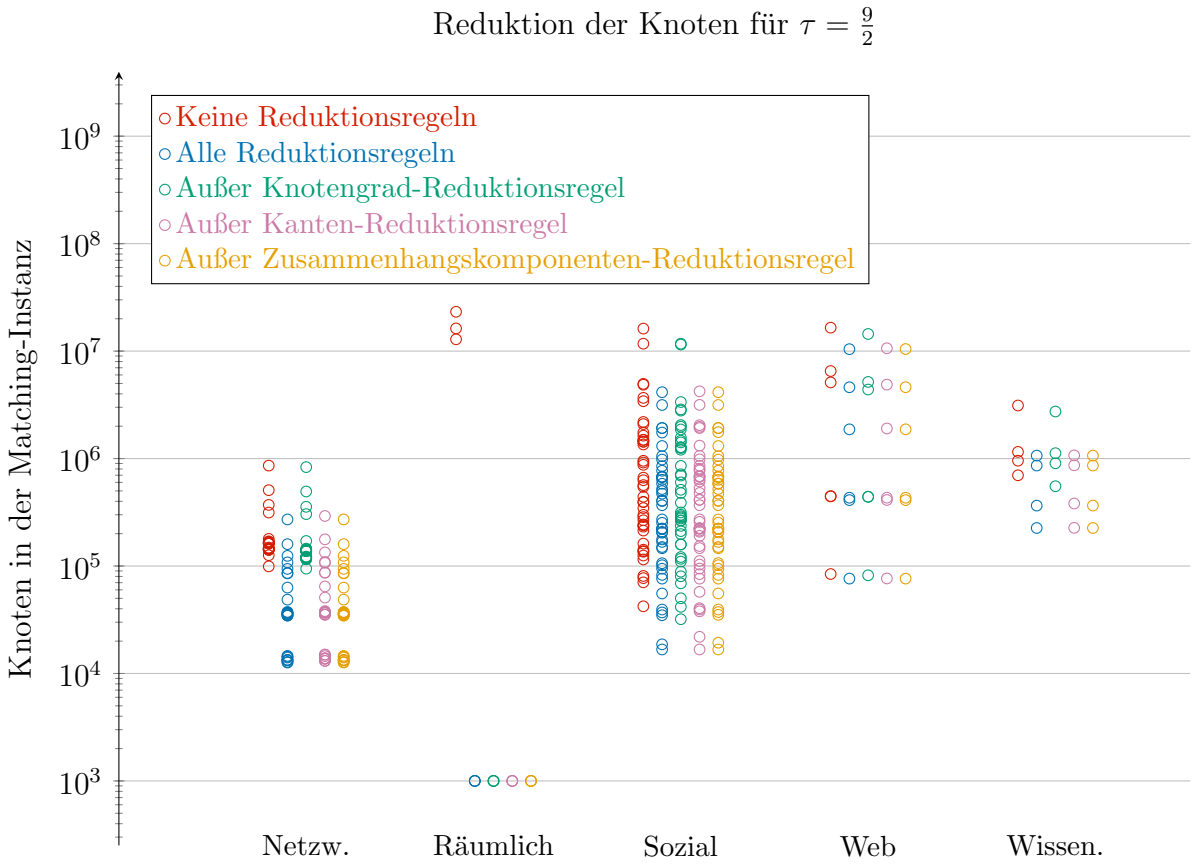


Abbildung 4.6: Knoten in der Matching-Instanz für die verschiedenen Graph-Kategorien Netzwerktopologie, Räumlich, Sozial, Web und Wissenschaftlich und Zieldichte $\tau = \frac{9}{2}$, wobei jeweils andere Kombinationen der Reduktionsregeln angewendet werden.

fernt werden, sind in der resultierenden Maximum Cardinality Matching-Instanz für jeden Knoten im ursprünglichen Graphen 2τ Kopien, sowie für jede Kante im ursprünglichen Graphen zwei Knoten vorhanden. Insgesamt sind es also $2\tau|V| + 2|E|$ Knoten. Wie erwartet sind es in jedem Fall mehr Knoten, als in den anderen Fällen, in denen Reduktionsregeln angewendet werden.

Ganz ähnlich verhält es sich, wenn außer der Reduktionsregel 3.1.1 über den Knotengrad alle Reduktionsregeln angewendet werden. Außer in der Kategorie Räumlich ist die Laufzeit sehr ähnlich zur Laufzeit ohne Reduktionsregeln. Einzig in der Kategorie Räumlich werden die dünnen Graphen bereits vollständig entfernt, wodurch die Laufzeit deutlich niedriger ist als ohne Reduktionsregeln. Die Reduktionsregeln 3.2.1 über Kanten zwischen Knoten mit kleinem Grad und 3.3.1 über Zusammenhangskomponenten entfernen also alleine meist nur wenige Knoten. Dies wird erwartet, da für die Anwendung im Gegensatz zu der Reduktionsregel 3.1.1 über den Knotengrad komplexere Bedingungen erfüllt werden müssen. Während hier ganze Zusammenhangskomponenten den Grad 2τ nicht überschreiten dürfen oder zwei verbundene Knoten einen Grad von höchstens $\tau + \frac{1}{2}$

haben müssen, kann die Reduktionsregel 3.1.1 über den Knotengrad schon für einzelne Knoten mit einem Grad von höchstens τ angewendet werden.

Wenn jedoch alle Reduktionsregeln angewendet werden, ist die Laufzeit in den meisten Fällen am niedrigsten, da hier die meisten Knoten entfernt werden können. Mit den Beobachtungen zu den beiden vorherigen Fällen, kann hieraus geschlossen werden, dass die Reduktionsregel 3.1.1 über den Knotengrad besonders effektiv ist, während die andern Reduktionsregeln weniger dazu beitragen, da sie nur wenige Knoten entfernen können. Wie bereits zuvor beschrieben, hängt dies mit der einfachen Bedingung der Reduktionsregel 3.1.1 über den Knotengrad zusammen.

Die beiden anderen Fälle, in denen außer der Reduktionsregel 3.2.1 über Kanten zwischen Knoten mit kleinem Grad oder außer der Reduktionsregel 3.3.1 über Zusammenhangskomponenten alle Reduktionsregeln angewendet werden, zeigen ähnliche Laufzeiten wie wenn alle Reduktionsregeln angewendet werden. Dies bestärkt die Interpretation, dass die Reduktionsregel 3.1.1 über den Knotengrad die größte Auswirkung auf die Laufzeit hat, da diese am meisten Knoten entfernt und schnell angewendet werden kann.

In Abbildung 4.7 werden die Gesamtlaufzeiten von BDED für die genannten Fälle noch einmal miteinander verglichen, wobei gut sichtbar ist, dass die Reduktionsregel 3.1.1 über den Knotengrad die größte Auswirkung auf die Laufzeit hat. Hier kann der etwas kleinere Unterschied in der Laufzeit zwischen keinen angewendeten Reduktionsregeln und der Anwendung aller außer der Reduktionsregel 3.1.1 über den Knotengrad besser beobachtet werden. Sehr gut ist zu erkennen, dass die Gesamtlaufzeit mit Anwendung aller Reduktionsregeln, sehr ähnlich zu denen, bei denen alle außer der Reduktionsregel 3.2.1 über Kanten zwischen Knoten mit kleinem Grad oder außer der Reduktionsregel 3.3.1 über Zusammenhangskomponenten ist.

In Abbildung 4.8 und Abbildung 4.9 werden die Gesamtlaufzeiten von BDED mit und ohne Anwendung aller Reduktionsregeln für die verschiedenen Graph-Kategorien und die Zieldichten $\tau = \frac{3}{2}$ sowie $\tau = \frac{9}{2}$ direkt miteinander verglichen. Hierbei ist zu beobachten, dass die Anwendung der Reduktionsregeln für die kleinere Zieldichte $\tau = \frac{3}{2}$ bis auf wenige Ausnahmen nur einen sehr geringfügigen Einfluss auf die Gesamtlaufzeit hat, der gleichermaßen positiv, wie negativ sein kann. Der Grund hierfür ist, dass die Laufzeit der Reduktionsregeln selbst und die Verbesserung der Laufzeit des Maximum Cardinality Matching-Algorithmus hier ungefähr gleich groß sind, da nur wenige Knoten entfernt werden können. Für die größere Zieldichte $\tau = \frac{9}{2}$ hingegen ist die Anwendung der Reduktionsregeln in den aller meisten Fällen deutlich vorteilhafter.

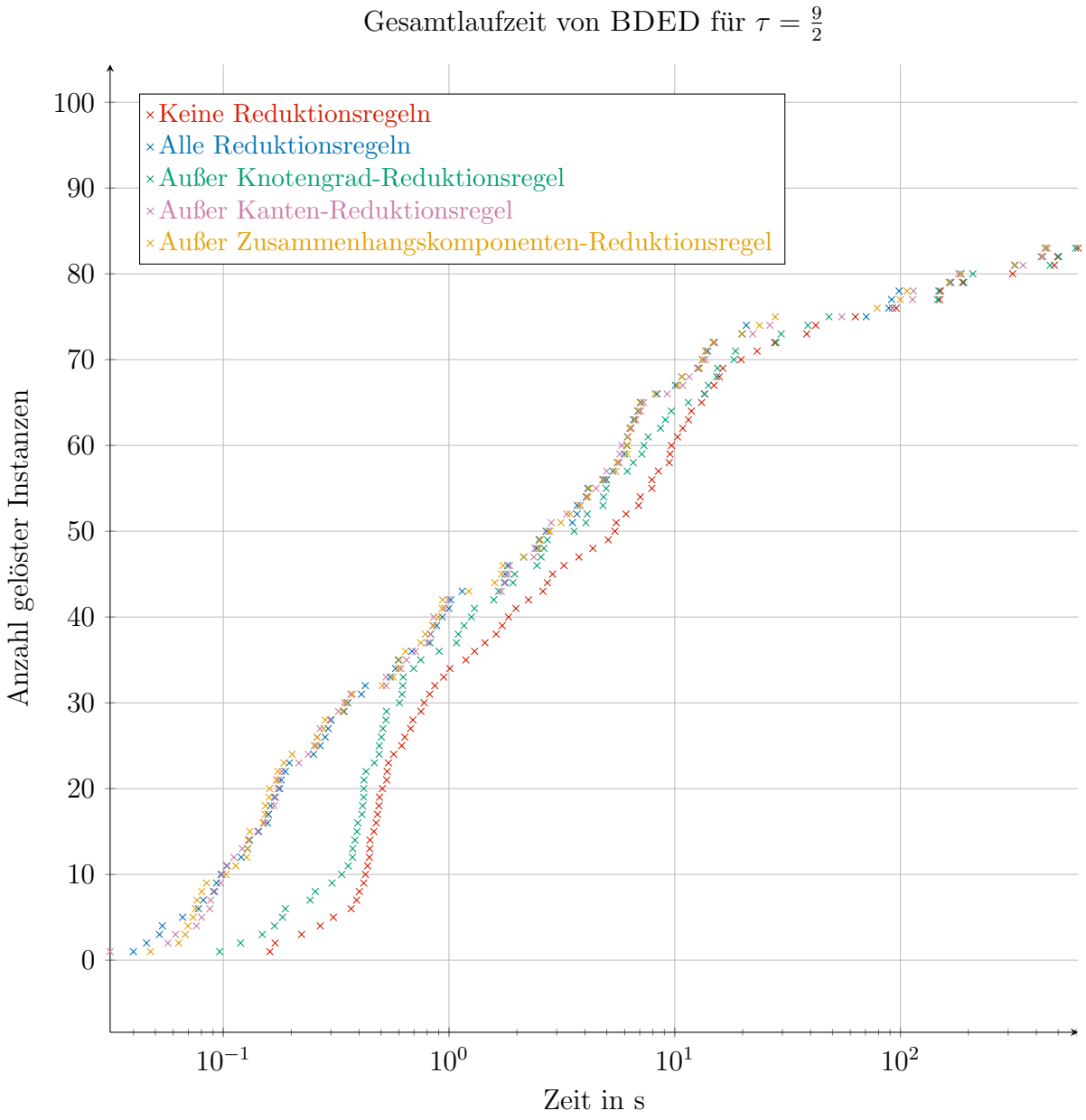


Abbildung 4.7: Anzahl der gelösten Instanzen von BDED in Abhängigkeit der Laufzeit in Sekunden für die Zieldichte $\tau = \frac{9}{2}$, wobei jeweils andere Kombinationen der Reduktionsregeln angewendet werden.

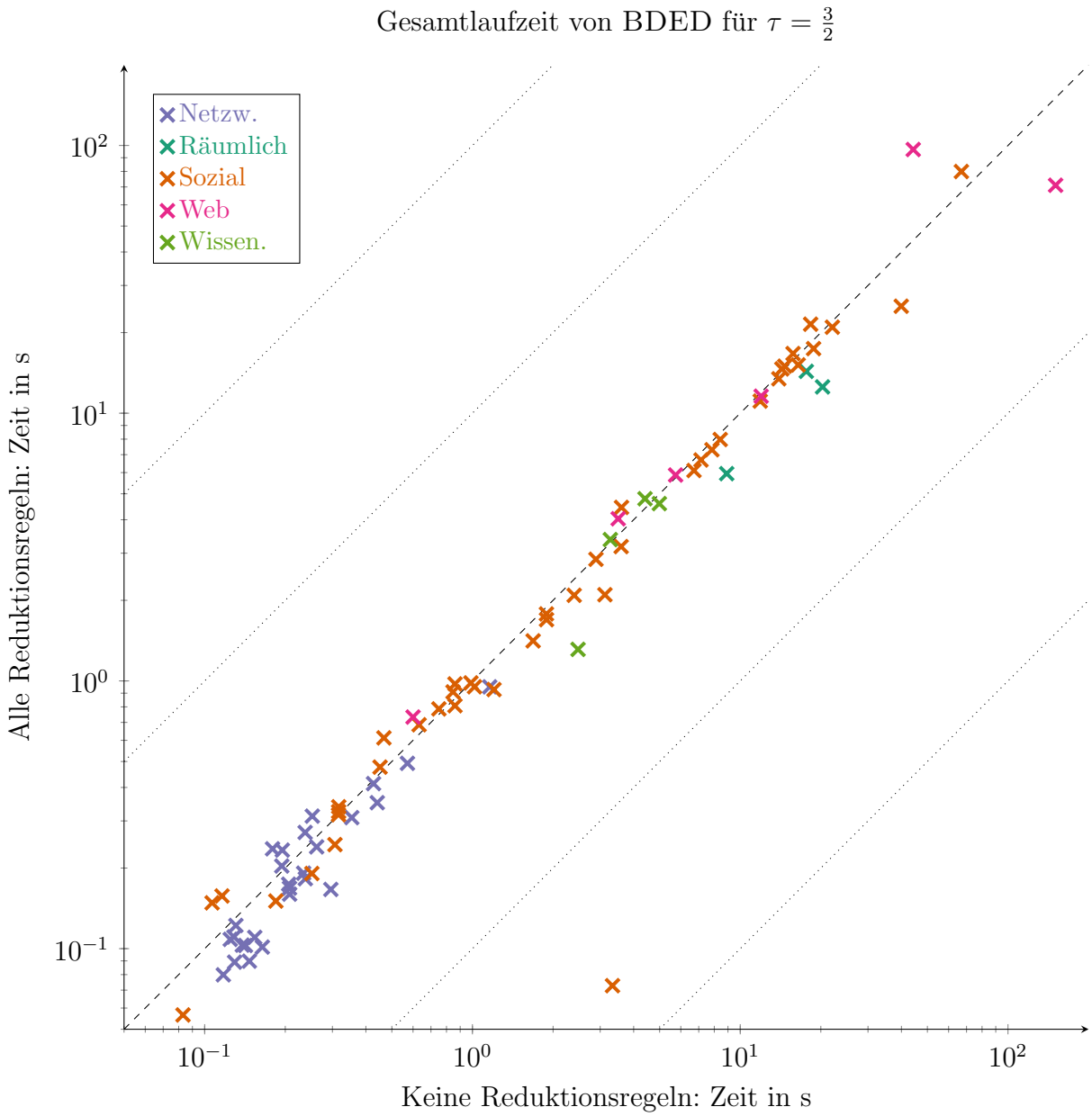


Abbildung 4.8: Vergleich der Gesamtlaufzeiten von BDED mit und ohne Anwendung der Reduktionsregeln für die verschiedenen Graph-Kategorien und Zieldichte $\tau = \frac{3}{2}$ in Sekunden.

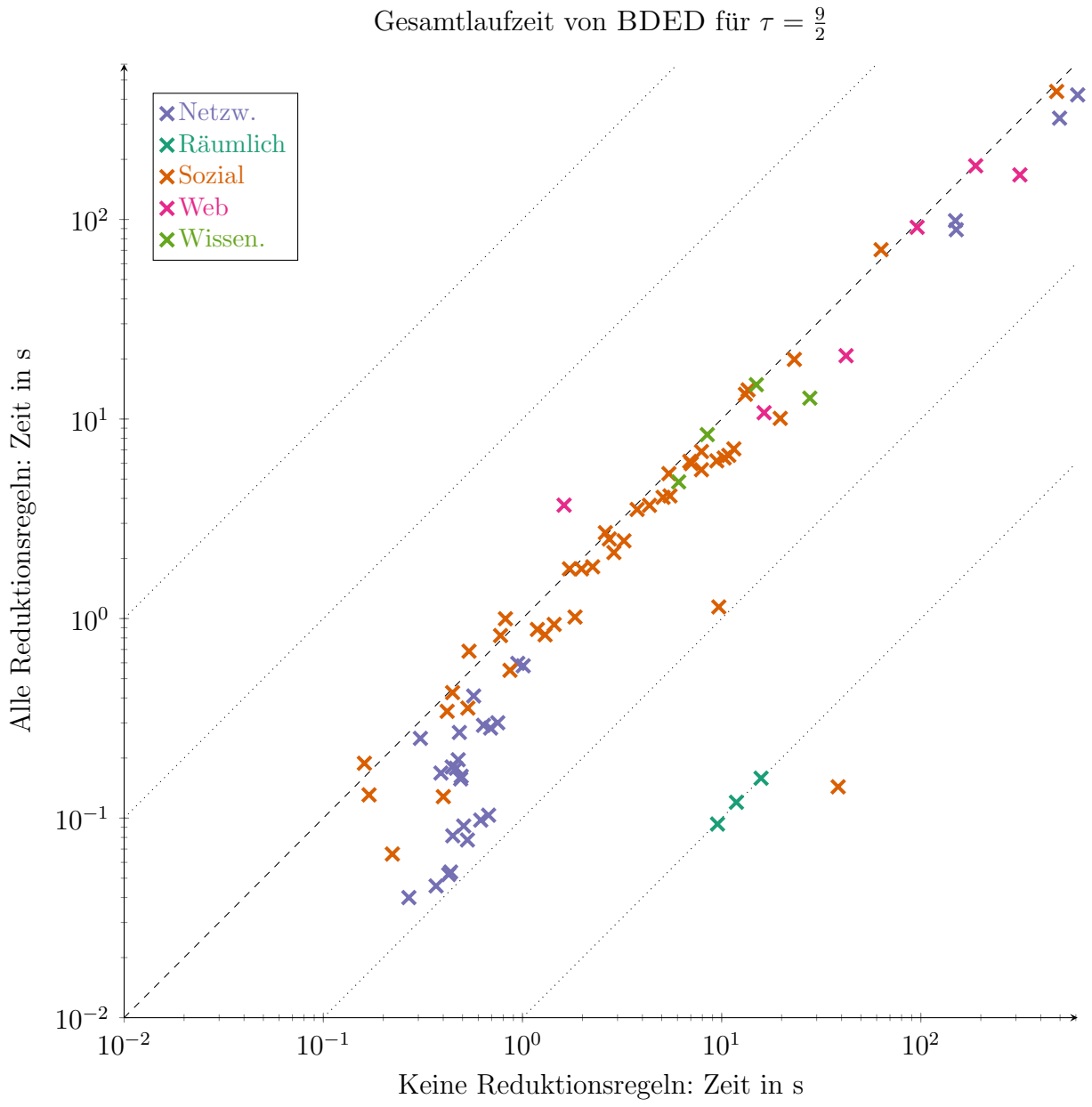


Abbildung 4.9: Vergleich der Gesamtlaufzeiten von BDED mit und ohne Anwendung der Reduktionsregeln für die verschiedenen Graph-Kategorien und Zieldichte $\tau = \frac{9}{2}$ in Sekunden.

5 Fazit

In dieser Arbeit wurde das Bounded-Density Edge Deletion-Problem für halbintegrale Zieldichten τ mit $2\tau \in \mathbb{N}$ untersucht. Ziel war es, einen Polynomialzeit-Algorithmus bereitzustellen, der auf einer Reduktion auf das Maximum Cardinality Matching-Problem basiert, diesen zu implementieren und experimentell zu evaluieren. Hierzu wurden die theoretischen Grundlagen erarbeitet, Datenreduktionsregeln entwickelt und eine Implementierung in der Programmiersprache Golang erstellt, die auf 83 realen Graphinstanzen aus dem SNAP-Datensatz evaluiert wurde.

Theoretische Ergebnisse. Als theoretisches Ergebnis wurde in Theorem 2.2 gezeigt, dass BDED für halbintegrale Zieldichten τ mit $2\tau \in \mathbb{N}$ in $\mathcal{O}(\tau(|V| + |E|))$ Zeit auf Maximum Cardinality Matching reduziert werden kann. Die Reduktion nutzt die Existenz halbintegraler Orientierungen aus Lemma 2.1, die aus der Modellierung als Flussnetzwerk mit ganzzahligen Kapazitäten und der Integrality Property folgt. Ein größtmögliches Matching entspricht einer optimalen BDED-Lösung mit minimaler Kantenlöschmenge. Da Maximum Cardinality Matching in $\mathcal{O}(m\sqrt{n})$ Zeit lösbar ist, ergibt sich ein polynomieller Gesamtalgorithmus für BDED mit halbintegralen Zieldichten. Darüber hinaus wurden vier Datenreduktionsregeln vorgestellt und formal bewiesen, die auf dem Knotengrad, auf Kanten zwischen Knoten mit kleinem Grad, auf Zusammenhangskomponenten mit beschränktem Maximalgrad sowie auf Kreisen aus Knoten mit kleinem Grad basieren. Die erschöpfende Anwendung der Reduktionsregel 3.1.1 über den Knotengrad ist in $\mathcal{O}(|V| + |E|)$ Zeit möglich, während die Reduktionsregeln 3.2.1 über Kanten und 3.4.1 über Kreise eine Laufzeit von $\mathcal{O}(|V|^2 + |V||E|)$ erfordern. Die Reduktionsregel 3.3.1 über Zusammenhangskomponenten kann mittels Breitensuche in $\mathcal{O}(|V|+|E|)$ Zeit oder mittels Union-Find in $\mathcal{O}(|V|\tau \cdot \alpha(|V|))$ Zeit angewendet werden.

Praktische Ergebnisse. Die experimentelle Evaluation auf 83 Instanzen aus fünf Kategorien – Netzwerktopologie, Räumlich, Sozial, Web und Wissenschaftlich – zeigt, dass die Effektivität der Reduktionsregeln stark von der Zieldichte τ, aber auch vom Verhältnis von τ zur maximalen Subgraphendichte ρ^* der jeweiligen Instanz abhängt. Für $\frac{\tau}{\rho^*} \geq 1$ konnten die Reduktionsregeln in nahezu allen Fällen sämtliche Knoten entfernen, sodass keine Maximum Cardinality Matching-Instanz mehr konstruiert werden musste und die Gesamtlaufzeit stark abfiel. Für $\frac{\tau}{\rho^*} < 1$ hingegen stieg die Gesamtlaufzeit mit wachsendem τ zumeist an, da die Reduktionsregeln weniger Knoten entfernen konnten und die Maximum Cardinality Matching-Instanz durch die 2τ Kopien pro Knoten linear wuchs. Die Kategorien unterscheiden sich hierbei deutlich: Räumliche Graphen, die eine sehr dünne Struktur aufweisen, konnten bereits ab $\tau = \frac{9}{2}$ vollständig reduziert wer-

den, während Web-Graphen mit maximaler Subgraphendichte von bis zu 136,89 selbst für $\tau = \frac{243}{2}$ nicht vollständig reduziert werden konnten.

Der Vergleich einzelner Reduktionsregeln ergab, dass die Reduktionsregel 3.1.1 über den Knotengrad den mit Abstand größten Beitrag zur Verkleinerung der Instanz leistet. Ohne sie ist die Gesamtlaufzeit nur gering besser als in dem Fall ohne jegliche Reduktionsregeln. Die anderen Regeln tragen nur wenig bei, da ihre Anwendungsbedingungen seltener erfüllt sind. In Kombination mit der Knotengrad-Regel erzielen sie nur eine minimale zusätzliche Verbesserung, ohne diese ist ihr eigener Einfluss etwas größer, da sie auch Knoten entfernen, die bei Anwendung aller Regeln bereits von der Knotengrad-Regel entfernt werden. Für kleine Zieldichten wie $\tau = \frac{3}{2}$ war die Anwendung der Reduktionsregeln insgesamt nahezu laufzeitneutral, da die eingesparte Zeit bei der Maximum Cardinality Matching-Lösung in etwa dem Aufwand der Regelanwendung entsprach und die Reduktionsregeln wenige Knoten entfernten. Für größere Zieldichten wie $\tau = \frac{9}{2}$ führte die Anwendung aller Reduktionsregeln hingegen fast immer zu erheblichen Laufzeitverbesserungen.

5.1 Ausblick

Die vorliegende Arbeit bietet mehrere Ansatzpunkte für zukünftige Forschung und Weiterentwicklung.

Reduktionsregeln für NP-schwere Fälle. Die vorgestellten Reduktionsregeln wurden im Kontext der polynomiell lösbaren halbintegralen Zieldichte evaluiert. Für allgemeine Zieldichten ist BDED NP-schwer [Ben+26], wodurch die Laufzeiten der Lösung deutlich höher ausfallen. In diesem Szenario ist der relative Einfluss von in Polynomialzeit ausführbaren Reduktionsregeln auf die Gesamtlaufzeit wesentlich größer, da bereits eine moderate Verkleinerung der Instanz zu erheblichen Laufzeitverbesserungen führen kann. Eine systematische Evaluation der Reduktionsregeln in Kombination mit exakten oder heuristischen Lösern für den NP-schweren Fall stellt daher eine vielversprechende Erweiterung dar.

Reduktionsregel für Cliquen bestimmter Größe. Die Reduktionsregeln 3.1.1 über den Knotengrad und 3.2.1 über Kanten zwischen Knoten mit kleinem Grad können als spezielle Fälle einer allgemeineren Regel betrachtet werden: Jede Clique der Größe n, die ausschließlich aus Knoten v mit $\deg(v) \leq \tau + \frac{n-1}{2}$ besteht, kann sicher entfernt werden. Die beiden genannten Regeln entsprechen dabei den Fällen $n = 1$ und $n = 2$. Auch die Reduktionsregel 3.4.1 über Kreise aus Knoten mit kleinem Grad überschneidet sich mit dieser allgemeinen Regel, da als Sonderfall ein Kreis aus drei Knoten, also eine Clique der Größe 3, die ausschließlich aus Knoten v mit $\deg(v) \leq \tau + 1$ besteht, sicher entfernt werden kann. Die Argumentationen der genannten Reduktionsregeln können dementsprechend auf die allgemeine Regel übertragen werden. Jeder Knoten der Clique hat $n-1$ Kanten innerhalb der Clique. In einer Clique der Größe n kann eine fraktionale Orientierung alle Kanten zwischen den Knoten der Clique mit $(\frac{1}{2}, \frac{1}{2})$ orientieren, sodass

jeder Knoten der Clique davon eine gesamte Orientierung von $\frac{n-1}{2}$ erhält. Es können jeweils maximal $\tau - \frac{n-1}{2}$ Kanten zwischen beliebigen Knoten außerhalb der Clique und jedem Knoten innerhalb existieren. Alle diese können vollständig zu dem Knoten in der Clique orientiert werden. Somit erhält jeder Knoten in der Clique insgesamt höchstens τ Orientierung und kann entfernt werden. Diese Regel experimentell zu evaluieren, sowie eine theoretische Laufzeitanalyse sind weitergehende mögliche Schritte.

Evaluation unabhängiger Lösung von Zusammenhangskomponenten. In der erstellten Implementierung werden alle nach der Reduktion verbliebenen Zusammenhangskomponenten gemeinsam in einer einzigen Maximum Cardinality Matching-Instanz gelöst. Stattdessen könnte jede Zusammenhangskomponente einzeln überführt und gelöst werden. Es wäre zu erwarten, dass das Lösen mehrerer kleinerer Instanzen in der Praxis schneller ist als das Lösen einer großen Instanz. Zudem würde sich der Speicherbedarf reduzieren, oder eine Möglichkeit zur Parallelisierung bieten, da die Instanzen unabhängig voneinander gelöst werden können.

Mögliche Ansätze zur Parallelisierung. Die aktuelle Implementierung von BDED arbeitet, bis auf den verwendeten LP-Solver [HH18], sequenziell. Es könnte evaluiert werden, wo Potential für Optimierungen durch Parallelisierung liegt. Eine Möglichkeit wäre wie zuvor beschrieben die parallele Lösung von Zusammenhangskomponenten.

Evaluation mit schnelleren Matching-Algorithmen. Der in dieser Implementierung verwendete Maximum Cardinality Matching-Algorithmus von Kececioglu und Pecqueur [KP98] hat eine Laufzeit von $\mathcal{O}(mn \cdot \alpha(m, n))$. Es wurden Algorithmen mit besserer theoretischer Laufzeitschranke vorgestellt, beispielsweise der Algorithmus von Micali und Vazirani [MV80] mit einer Laufzeit von $\mathcal{O}(m\sqrt{n})$. Eine Evaluation mit solchen Algorithmen könnte zeigen, inwieweit die Gesamtlaufzeit weiter gesenkt werden kann und ob sich das Verhältnis zwischen Reduktions- und Lösungszeit verschiebt.

Evaluation der Kreisreduktionsregel. Die vorgestellte Reduktionsregel 3.4.1 für Kreise aus Knoten mit kleinem Grad wurde nicht implementiert und somit in der experimentellen Evaluation nicht berücksichtigt. Eine zukünftige Evaluation könnte untersuchen, wie häufig diese Regel auf realen Instanzen anwendbar ist und welche zusätzliche Reduktion der Instanzgröße sie erzielen kann. Insbesondere bei Graphen mit vielen Knoten kleinen Grades, wie sie etwa in Straßennetzwerken vorkommen, könnte diese Regel einen signifikanten Beitrag leisten.

Literatur

[AMO93] Ravindra K. Ahuja, Thomas L. Magnanti und James B. Orlin. *Network flows: theory, algorithms, and applications*. USA: Prentice-Hall, Inc., 1993 (siehe S. 15, 16).

[Ben+26] Matthias Bentert, Tom-Lukas Breitkopf, Vincent Froese, Anton Herrmann und André Nichterlein. *Density Matters: A Complexity Dichotomy of Deleting Edges to Bound Subgraph Density*. 2026. arXiv: 2601.03129 [cs.DS] (siehe S. 5, 10–13, 21, 29, 42).

[BNA24] Cristina Bazgan, André Nichterlein und Sofia Vazquez Alferez. *Destroying Densest Subgraphs is Hard*. 2024. arXiv: 2404.08599 [cs.DS] (siehe S. 12).

[Cha00] Moses Charikar. *Greedy approximation algorithms for finding dense components in a graph*. In: *Proceedings of the 3rd International Workshop on Approximation Algorithms for Combinatorial Optimization (APPROX 2000)*. Bd. 1913. Lecture Notes in Computer Science. Springer, 2000, S. 84–95 (siehe S. 12, 14, 27).

[Edm65] Jack Edmonds. *Maximum Matching and a Polyhedron with* $0,1$ *Vertices*. In: *J. of Res. the Nat. Bureau of Standards* 69 B (1965), S. 125–130 (siehe S. 12, 27).

[Gab17] Harold N. Gabow. *The Weighted Matching Approach to Maximum Cardinality Matching*. In: *Fundamenta Informaticae* 154.1-4 (2017), S. 109–130. eprint: https://journals.sagepub.com/doi/pdf/10.3233/FI-2017-1555 (siehe S. 11).

[Gol84] A. V. Goldberg. *Finding a Maximum Density Subgraph*. Techn. Ber. USA, 1984 (siehe S. 12, 27).

[HH18] Q. Huangfu und J. A. J. Hall. *Parallelizing the dual revised simplex method*. In: *Mathematical Programming Computation* 10.1 (2018), S. 119–142 (siehe S. 43).

[Hoo+17] Bryan Hooi, Kijung Shin, Hyun Ah Song, Alex Beutel, Neil Shah und Christos Faloutsos. *Graph-Based Fraud Detection in the Face of Camouflage*. In: *ACM Trans. Knowl. Discov. Data* 11.4 (Juni 2017) (siehe S. 9, 10).

[Kar74] Alexander V. Karzanov. *Determining the Maximal Flow in a Network by the Method of Preflows*. In: *Soviet Mathematics Doklady* 15 (1974), S. 434–437 (siehe S. 12).

[KP98] John D. Kececioglu und Justin Pecqueur. *Computing Maximum-Cardinality Matchings in Sparse General Graphs*. In: *Workshop on Algorithm Engineering*. 1998 (siehe S. 12, 13, 27, 43).

[LK14] Jure Leskovec und Andrej Krevl. *SNAP Datasets: Stanford Large Network Dataset Collection*. `http://snap.stanford.edu/data`. Juni 2014 (siehe S. 9, 28).

[MV80] Silvio Micali und Vijay Vazirani. *An O(sqrt(—v—) —E—) Algorithm for Finding Maximum Matching in General Graphs*. In: Okt. 1980, S. 17–27 (siehe S. 11, 43).

[Sle80] Daniel D. Sleator. *An $O(nm \log n)$ Algorithm for Maximum Network Flow*. Diss. Stanford University, 1980 (siehe S. 12).

[Vaz20] Vijay Vazirani. *A Proof of the MV Matching Algorithm*. Dez. 2020 (siehe S. 11).

6 Anhang

6.1 Auflistung der Testdaten

Netzwerktopologie			
Name	$\|\mathbf{V}\|$	$\|\mathbf{E}\|$	ρ^*
p2p-Gnutella08	$6.30 \cdot 10^3$	$2.08 \cdot 10^4$	8.62
p2p-Gnutella09	$8.11 \cdot 10^3$	$2.60 \cdot 10^4$	8.61
p2p-Gnutella06	$8.72 \cdot 10^3$	$3.15 \cdot 10^4$	7.43
p2p-Gnutella05	$8.85 \cdot 10^3$	$3.18 \cdot 10^4$	7.57
oregon1_010331	$1.07 \cdot 10^4$	$2.20 \cdot 10^4$	13.04
oregon1_010407	$1.07 \cdot 10^4$	$2.20 \cdot 10^4$	12.03
oregon1_010414	$1.08 \cdot 10^4$	$2.25 \cdot 10^4$	12.87
oregon1_010421	$1.09 \cdot 10^4$	$2.27 \cdot 10^4$	13.41
p2p-Gnutella04	$1.09 \cdot 10^4$	$4.00 \cdot 10^4$	5.90
oregon1_010428	$1.09 \cdot 10^4$	$2.25 \cdot 10^4$	13.02
oregon2_010331	$1.09 \cdot 10^4$	$3.12 \cdot 10^4$	25.41
oregon1_010505	$1.09 \cdot 10^4$	$2.26 \cdot 10^4$	12.93
oregon2_010407	$1.10 \cdot 10^4$	$3.09 \cdot 10^4$	24.53
oregon1_010512	$1.10 \cdot 10^4$	$2.27 \cdot 10^4$	12.97
oregon2_010414	$1.10 \cdot 10^4$	$3.18 \cdot 10^4$	25.80
oregon1_010519	$1.11 \cdot 10^4$	$2.27 \cdot 10^4$	13.14
oregon2_010421	$1.11 \cdot 10^4$	$3.15 \cdot 10^4$	24.45
oregon2_010428	$1.11 \cdot 10^4$	$3.14 \cdot 10^4$	23.86
oregon2_010505	$1.12 \cdot 10^4$	$3.09 \cdot 10^4$	23.21
oregon1_010526	$1.12 \cdot 10^4$	$2.34 \cdot 10^4$	13.53
oregon2_010512	$1.13 \cdot 10^4$	$3.13 \cdot 10^4$	23.08
oregon2_010519	$1.14 \cdot 10^4$	$3.23 \cdot 10^4$	24.50
oregon2_010526	$1.15 \cdot 10^4$	$3.27 \cdot 10^4$	25.01
p2p-Gnutella25	$2.27 \cdot 10^4$	$5.47 \cdot 10^4$	4.21
p2p-Gnutella24	$2.65 \cdot 10^4$	$6.54 \cdot 10^4$	4.39
p2p-Gnutella30	$3.67 \cdot 10^4$	$8.83 \cdot 10^4$	4.79
p2p-Gnutella31	$6.26 \cdot 10^4$	$1.48 \cdot 10^5$	4.64
Räumlich			
Name	$\|\mathbf{V}\|$	$\|\mathbf{E}\|$	ρ^*
roadNet-PA	$1.09 \cdot 10^6$	$1.54 \cdot 10^6$	1.88
roadNet-TX	$1.38 \cdot 10^6$	$1.92 \cdot 10^6$	2.08
roadNet-CA	$1.97 \cdot 10^6$	$2.77 \cdot 10^6$	1.97

Sozial			
Name	$\lvert\mathbf{V}\rvert$	$\lvert\mathbf{E}\rvert$	ρ^*
email-Eu-core	$1.01 \cdot 10^3$	$1.61 \cdot 10^4$	27.57
twitch_PTBR_musae_PTBR_edges	$1.91 \cdot 10^3$	$3.13 \cdot 10^4$	31.58
facebook_clean_data_tvshow_edges	$3.89 \cdot 10^3$	$1.72 \cdot 10^4$	30.95
facebook_combined	$4.04 \cdot 10^3$	$8.82 \cdot 10^4$	77.35
twitch_RU_musae_RU_edges	$4.39 \cdot 10^3$	$3.73 \cdot 10^4$	20.40
twitch_ES_musae_ES_edges	$4.65 \cdot 10^3$	$5.94 \cdot 10^4$	28.75
ca-GrQc	$5.24 \cdot 10^3$	$1.45 \cdot 10^4$	22.39
facebook_clean_data_politician_edges	$5.91 \cdot 10^3$	$4.17 \cdot 10^4$	24.37
twitch_FR_musae_FR_edges	$6.55 \cdot 10^3$	$1.13 \cdot 10^5$	34.14
facebook_clean_data_government_edges	$7.06 \cdot 10^3$	$8.94 \cdot 10^4$	36.58
wiki-Vote	$7.12 \cdot 10^3$	$1.01 \cdot 10^5$	46.28
twitch_ENGB_musae_ENGB_edges	$7.13 \cdot 10^3$	$3.53 \cdot 10^4$	11.98
lasftm_asia_lastfm_asia_edges	$7.62 \cdot 10^3$	$2.78 \cdot 10^4$	14.79
twitch_DE_musae_DE_edges	$9.50 \cdot 10^3$	$1.53 \cdot 10^5$	39.02
ca-HepTh	$9.88 \cdot 10^3$	$2.60 \cdot 10^4$	15.50
facebook_clean_data_public_figure_edges	$1.16 \cdot 10^4$	$6.70 \cdot 10^4$	34.80
ca-HepPh	$1.20 \cdot 10^4$	$1.18 \cdot 10^5$	119.00
facebook_clean_data_athletes_edges	$1.39 \cdot 10^4$	$8.68 \cdot 10^4$	19.37
facebook_clean_data_company_edges	$1.41 \cdot 10^4$	$5.21 \cdot 10^4$	13.81
ca-AstroPh	$1.88 \cdot 10^4$	$1.98 \cdot 10^5$	32.11
facebook_large_musae_facebook_edges	$2.25 \cdot 10^4$	$1.71 \cdot 10^5$	36.34
ca-CondMat	$2.31 \cdot 10^4$	$9.34 \cdot 10^4$	13.37
facebook_clean_data_new_sites_edges	$2.79 \cdot 10^4$	$2.06 \cdot 10^5$	18.56
deezer_europe_deezer_europe_edges	$2.83 \cdot 10^4$	$9.28 \cdot 10^4$	8.84
git_web_ml_musae_git_edges	$3.77 \cdot 10^4$	$2.89 \cdot 10^5$	30.25
deezer_clean_data_RO_edges	$4.18 \cdot 10^4$	$1.26 \cdot 10^5$	5.49
deezer_clean_data_HU_edges	$4.75 \cdot 10^4$	$2.23 \cdot 10^5$	9.12
facebook_clean_data_artist_edges	$5.05 \cdot 10^4$	$8.19 \cdot 10^5$	58.13
deezer_clean_data_HR_edges	$5.46 \cdot 10^4$	$4.98 \cdot 10^5$	16.61
loc-brightkite_edges	$5.82 \cdot 10^4$	$2.14 \cdot 10^5$	40.56
soc-Epinions1	$7.59 \cdot 10^4$	$4.06 \cdot 10^5$	60.25
soc-sign-Slashdot081106	$7.74 \cdot 10^4$	$3.27 \cdot 10^5$	28.96
soc-Slashdot0811	$7.74 \cdot 10^4$	$4.69 \cdot 10^5$	42.27
twitter_combined	$8.13 \cdot 10^4$	$1.34 \cdot 10^6$	69.62
soc-sign-Slashdot090216	$8.19 \cdot 10^4$	$3.47 \cdot 10^5$	29.05
soc-sign-Slashdot090221	$8.21 \cdot 10^4$	$3.49 \cdot 10^5$	29.52
soc-Slashdot0902	$8.22 \cdot 10^4$	$5.04 \cdot 10^5$	43.44
soc-sign-epinions	$1.32 \cdot 10^5$	$5.01 \cdot 10^5$	55.49
loc-gowalla_edges	$1.97 \cdot 10^5$	$9.50 \cdot 10^5$	43.80
com-dblp.ungraph	$3.17 \cdot 10^5$	$1.05 \cdot 10^6$	56.57
com-amazon.ungraph	$3.35 \cdot 10^5$	$9.26 \cdot 10^5$	4.80
soc-pokec-profiles	$1.06 \cdot 10^6$	$1.07 \cdot 10^6$	31.44
com-youtube.ungraph	$1.13 \cdot 10^6$	$2.99 \cdot 10^6$	45.60

Web			
Name	$\lvert\mathbf{V}\rvert$	$\lvert\mathbf{E}\rvert$	ρ^*
wikipedia_chameleon_musae_chameleon_edges	$2.28 \cdot 10^3$	$3.14 \cdot 10^4$	47.68
wikipedia_squirrel_musae_squirrel_edges	$5.20 \cdot 10^3$	$1.98 \cdot 10^5$	136.89
wikipedia_crocodile_musae_crocodile_edges	$1.16 \cdot 10^4$	$1.71 \cdot 10^5$	50.96
web-Stanford	$2.82 \cdot 10^5$	$1.99 \cdot 10^6$	59.39
web-NotreDame	$3.26 \cdot 10^5$	$1.09 \cdot 10^6$	78.66
web-Google	$8.76 \cdot 10^5$	$4.32 \cdot 10^6$	28.04
Wissenschaftlich			
Name	$\lvert\mathbf{V}\rvert$	$\lvert\mathbf{E}\rvert$	ρ^*
cit-HepTh	$2.78 \cdot 10^4$	$3.52 \cdot 10^5$	30.34
cit-HepPh	$3.45 \cdot 10^4$	$4.21 \cdot 10^5$	25.28
email-Enron	$3.67 \cdot 10^4$	$1.84 \cdot 10^5$	37.34
email-EuAll	$2.65 \cdot 10^5$	$3.64 \cdot 10^5$	32.92