



OTTO VON GUERICKE
UNIVERSITÄT
MAGDEBURG

MATH

FAKULTÄT FÜR
MATHEMATIK



Deutsches Zentrum
für Luft- und Raumfahrt

Quadratische binäre Optimierung ohne Nebenbedingungen auf Chimera-Graphen

An der Fakultät für Mathematik
der Otto-von-Guericke-Universität Magdeburg
zur Erlangung des akademischen Grades
Master of Science
angefertigte

Masterarbeit

vorgelegt von
ELISABETH LOBE
geboren am 17.01.1990 in Meißen,
Studiengang Mathematik,
Studienrichtung Computermathematik.

27. April 2016

Betreut am Institut für Mathematische Optimierung
von PROF. DR. RER. NAT. HABIL. VOLKER KAIBEL
und am Deutschen Zentrum für Luft- und Raumfahrt von
DR. RER. NAT. TOBIAS STOLLENWERK und DR.-ING. ACHIM BASERMANN

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt habe. Weiterhin erkläre ich, diese Masterarbeit in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt zu haben.

Magdeburg, 27. April 2016

Inhaltsverzeichnis

Symbol- und Abkürzungsverzeichnis	III
1 Einleitung	1
2 Quadratische binäre Optimierung ohne Nebenbedingungen	3
2.1 Grundlagen	3
2.2 Komplexität und Lösungsmethoden	6
2.2.1 Klassische Einordnung und allgemeine Algorithmen	6
2.2.2 Polynomielle Lösbarkeit	6
2.2.3 Randomisierte Komplexitätsklassen	8
2.2.4 Quantenkomplexität	9
2.3 Reduktionen	10
2.3.1 Direkte Umsetzung	10
2.3.2 Lineare Nebenbedingungen an 0/1-Variablen	13
2.3.3 Ganzzahlige Probleme	18
2.3.4 Pseudoboolesche Funktionen mit beliebigem Grad	20
2.3.5 Allgemeine Graphenprobleme	23
2.3.6 Weitere Beispielprobleme	25
3 Der Chimera-Graph	33
3.1 Allgemeiner Aufbau	33
3.2 Einbettung beliebiger QUBOs	35
3.2.1 Allgemeines Prinzip	35
3.2.2 Darstellung vollständiger Graphen	36
3.2.3 Gewichte und Stärken	39
3.2.4 Allgemeines Einbettungsproblem	42
3.3 Spezielle Minoren	44
3.3.1 Vollständig bipartiter Graph	44
3.3.2 Symmetrische Kontraktionen	48
3.3.3 Variationen	64
3.3.4 Zusammenfassung	72
3.4 Beispiele	73
3.4.1 Direkte Umsetzung	73
3.4.2 Im Gittergraphen	75
4 Satellitenzeitplanung	83
4.1 Kontinuierliches Modell	83
4.1.1 Allgemeines	83
4.1.2 Umgebungsbedingungen	84

4.1.3	Gerätebedingungen	84
4.1.4	Ziele	85
4.1.5	Parameter aus technischen Voraussetzungen	86
4.1.6	Herangehensweise	87
4.2	Diskretisiertes Modell	88
4.2.1	Allgemeines	88
4.2.2	Umgebungs- und Gerätebedingungen	89
4.2.3	Ziel	90
4.2.4	Analyse	90
4.2.5	Erweiterungen	91
4.3	Vereinfachtes lineares Modell	91
4.4	Abgeleitetes quadratisches Modell ohne Nebenbedingungen	92
4.4.1	Einschränkungen für die Parameter	92
4.4.2	Nebenbedingungen	93
4.4.3	Analyse	93
4.4.4	Beispiel	94
4.5	Zusammenfassungen	95
5	Fazit	99
	Abbildungen	VI
	Literatur	VII

Symbol- und Abkürzungsverzeichnis

Matrizen/Vektoren

$M \in \mathbb{R}^{n \times m}$	reelle Matrix der Größe $n \times m$
$M_{k,j}$	(k, j) -tes Element von M
$M_{k,*}, M_{*,j}$	k -te Zeile, j -te Spalte von M
M^\top	transponierte Matrix von M
\mathbb{I}_n	Identitätsmatrix der Größe $n \times n$
$\mathbb{O}_{[n \times m]}$	Nullmatrix [der Größe $n \times m$]
$\mathbb{1}_{[n \times m]}$	Einsmatrix [der Größe $n \times m$]
\mathbb{e}_k	k -ter Einheitsvektor
$\mathfrak{2}_n = (2^0, \dots, 2^n)^\top$	$(n + 1)$ -dimensionaler Vektor der 2er-Potenzen
$\ \cdot\ _1, \ \cdot\ _2 : \mathbb{R}^n \rightarrow \mathbb{R}_{\geq 0}$	Manhattan-, euklidische Vektornorm

Graphen

$G = (V, E)$	einfacher, ungerichteter Graph mit Knotenmenge V und Kantenmenge $E \subseteq \{\{v, w\} : v, w \in V, v \neq w\}$
$D = (V, A)$	Digraph, gerichteter Graph mit Knotenmenge V und Kantenmenge $E \subseteq V \times V$
$V(G)$	Knotenmenge des Graphen G
$E(G)$	Kantenmenge des Graphen G
$G(W)$	induzierter Teilgraph mit Knoten $W \subseteq V$ und Kanten $\{\{v, w\} \in E : v, w \in W\}$
$g : V \rightarrow \mathbb{R}$	Gewichtung der Knoten
$S : E \rightarrow \mathbb{R}$	Bewertungsfunktion, Stärken der Kanten
$d(v) = \{w : \{v, w\} \in E\}$	Nachbarknoten von $v \in V$
$\delta(v) = \{\{v, w\} \in E\}$	ausgehende Kanten von $v \in V$
$\mathcal{L}(G)$	Kantengraph von G
$G^* = (F, E^*)$	Dualgraph von G mit Ländern F , dualen Kanten E^*
$\alpha(G)$	Stabilitätszahl des Graphen G
$\omega(G)$	Cliquenzahl des Graphen G

$\nu(G)$	Matchingzahl des Graphen G
K_n	vollständiger Graph mit n Knoten
$K_{n,m}$	vollständig bipartiter Graph mit $ V_0 + V_1 = n + m$ Knoten
$L_{n,m}$	Gittergraph mit n Zeilen und m Spalten

Optimierungsprobleme

IP	Integer Program
ILP	Integer Linear Program
MILP	Mixed-Integer Linear Program
QUBO	Quadratic Unconstrained Binary Optimization
SAT	Satisfiability (Erfüllbarkeitsproblem der Aussagenlogik)

Komplexitätsklassen

P	Polynomial Time
NP	Nondeterministic Polynomial Time
PP	Probabilistic Polynomial Time
PSPACE	Polynomial Space (of memory)
BPP	Bounded Error Probabilistic Polynomial Time
BQP	Bounded Error Quantum Probabilistic Polynomial Time

Anderes

$2^S = \{T : T \subseteq S\}$	Potenzmenge der Menge S
$\text{sgn}(y) = \begin{cases} +1, & y > 0 \\ 0, & y = 0 \\ -1, & y < 0 \end{cases}$	Vorzeichen von $y \in \mathbb{R}$
$\hat{\sigma}(g) = \sum_{i \in I} \frac{1}{2}(1 + \text{sgn}(g_i))g_i$	Summe der positiven Elemente von $g \in \mathbb{R}^I$ (für eine Indexmenge I)
$\check{\sigma}(g) = \sum_{i \in I} \frac{1}{2}(1 - \text{sgn}(g_i))g_i$	Summe der negativen Elemente von $g \in \mathbb{R}^I$
o.B.d.A.	ohne Beachtung der Allgemeinheit
$x \oplus y = x + y \pmod{2}$	logisches ausschließendes Oder

1 Einleitung

In der heutigen Zeit werden immer umfangreichere und komplexere mathematische Problemstellungen betrachtet. Bei der Bestimmung von Lösungen gerät oftmals selbst die modernste Computertechnik an die Grenzen ihrer Leistungsfähigkeit. Dies beruht auf dem zugrundeliegenden Rechnermodell und dessen Einschränkungen bezüglich der physikalischen Realisierbarkeit, welche sich in der Größe der Hardware und der Dauer der Berechnungen widerspiegeln. Deshalb ist es notwendig, sich mit alternativen Konzepten und deren technischer Umsetzung zu befassen. Einen vielversprechenden Ansatz bietet hierbei die Verwendung quantenmechanischer Prinzipien. Auf dieser Grundlage wurde in den 1980ern die Theorie des Quantencomputing entwickelt. Wegweisend waren hierbei die Arbeiten von P. Benioff [1], Y. Manin [2], R. Feynman [3] und D. Deutsch [4]. Der erhoffte Vorteil gegenüber klassischen Systemen beruht auf der Möglichkeit der parallelen Manipulation mehrerer Zustände, in denen sich ein Quantensystem gleichzeitig befinden kann.

Aktuell wird an vielen verschiedenen Realisierungen von Quantencomputermodellen geforscht [5]. Der erste kommerziell erhältliche Quantencomputer wird von der kanadischen Firma D-Wave Systems Inc. vertrieben. Im Jahr 2013 verkaufte der Hardwarehersteller ein Modell unter anderem an die NASA und Google. Dieser adiabatische Quantencomputer ist ein Löser für Optimierungsprobleme mit quadratischen Zielfunktionen über binären Variablen ohne Nebenbedingungen (kurz QUBOs) auf speziellen Graphen, den sogenannten Chimera-Graphen. Es wurden mehrere Studien durchgeführt, die eine mögliche Überlegenheit des D-Wave-Computers gegenüber klassischen Rechnern untersuchen. Diese sind beispielsweise in [6], [7], [8], [9] und [10] veröffentlicht worden. Deren Ergebnisse widersprechen sich teilweise und erfordern weitere Forschung auf diesem Gebiet. Die Skalierung der Laufzeit hinsichtlich der Eingabegröße ist anscheinend stark von den physikalischen Hintergründen des Rechners sowie den einzelnen Fragestellungen abhängig und kann bisher nicht konkret angegeben werden. Sollte sich der D-Wave Computer in irgendeiner Weise als überlegen herausstellen, bleibt zudem die Frage, welche bekannten Probleme sich effizient auf die eingeschränkte Struktur des Chimera-Graphen übertragen lassen. Motivation dieser Arbeit ist es daher, diese Struktur eingehender zu untersuchen und spezielle Probleme zu formulieren, die deren Restriktionen berücksichtigen.

Die Arbeit ist wie folgt aufgebaut. Zunächst wird auf die Grundlagen der diskreten Optimierung, speziell der QUBOs, eingegangen. Viele verschiedene Probleme aus der Klasse NP lassen sich mit der Graphenstruktur der QUBOs in Verbindung bringen. Daher werden im Kapitel 2 die allgemeinen Methoden aufgezeigt, mit denen diese Probleme schrittweise in ein QUBO umformuliert werden können. Anhand einiger Beispiele wird diese Vorgehensweise verdeutlicht. Anschließend wird in Kapitel 3 die spezielle Struktur des Chimera-Graphen erläutert und dessen QUBO-Formulierung aufgestellt. Um einen Zusammenhang mit anderen Graphenproblemen zu finden, werden zudem

Minoren dieses Graphen näher betrachtet. Die entsprechenden QUBOs können in den Chimera-Graphen durch Zusammenfassen von Teilgraphen eingebettet werden. Die dazu benötigten Schritte werden für jeden einzelnen Minoran ausführlich dargestellt. Am Ende dieses Kapitels werden weitere Beispiele veranschaulicht, die auf die Besonderheiten des Chimera-Graphen abgestimmt sind. Die Anwendbarkeit der in den vorherigen Kapiteln beschriebenen Methoden wird im 4. Kapitel anhand eines komplexen Anwendungsbeispiels aus der Raumfahrt demonstriert. Dazu wird ein mathematisches Modell einer Satellitenzeitplanung formuliert, vereinfacht und anschließend den entsprechenden Transformationen unterzogen. Letztendlich kann die Größe des erhaltenen QUBOs und somit die Lösbarkeit durch den Quantencomputer von D-Wave-Systems abgeschätzt werden.

2 Quadratische binäre Optimierung ohne Nebenbedingungen

In diesem Kapitel werden zunächst die Grundlagen dieses mathematisch schwierigen Optimierungsproblems beschrieben. Für einige Spezialfälle werden deren Lösungsmöglichkeiten betrachtet. Des Weiteren werden einige konkrete Beispiele für Reduktionen gezeigt sowie allgemeine Regeln für die Transformationen verschiedenster Fragestellungen auf dieses Problem aufgestellt.

2.1 Grundlagen

Heutzutage wird oft versucht, immer das Beste aus allem herauszuholen: die Kosten für ein Projekt so gering wie möglich zu halten oder den größtmöglichen Umsatz zu erzielen. Ein Teilgebiet der angewandten Mathematik, die Optimierung, befasst sich daher mit der effizientesten Bestimmung solcher Optima durch die Analyse der Problemstellung und der Erstellung von speziell darauf abgestimmten exakten oder auch approximativen Algorithmen. Es wird eine Bewertungsfunktion f über endlich vielen Variablen y_1, \dots, y_n aufgestellt, deren minimaler oder maximaler Wert gefunden werden soll. Dazu ergeben sich meist Nebenbedingungen an diese Variablen in Form von Gleichungen und Ungleichungen. Sie beschreiben eine Menge $Y \subseteq \mathbb{R}^n$, aus der die zulässigen Lösungen stammen dürfen. Daraus ergibt sich ein allgemeines Optimierungsproblem

$$\min_{\max} \{f(y) : y \in Y\} = \min_{\max_{y \in Y}} f(y) = \begin{array}{l} \min \\ \max \\ \text{s.t.} \end{array} \begin{array}{l} f(y) \\ y \in Y \end{array}$$

mit $y = (y_1, \dots, y_n)$ und $f : Y \rightarrow \mathbb{R} \cup \{-\infty, \infty\}$. Ist der Optimalwert eines Minimierungsproblems $-\infty$ oder der eines Maximierungsproblems ∞ , so heißen die Probleme unbeschränkt. Gilt hingegen für den Definitionsbereich $Y = \emptyset$, so existiert keine Optimallösung, denn das Problem ist unzulässig. Da für eine Klasse von Zielfunktionen f , die invariant unter Negierung sind, $\max\{f(y) : y \in Y\} = -\min\{-f(y) : y \in Y\}$ gilt, genügt in diesem Fall die Betrachtung von Minimierungsproblemen.

Je nach Art der Zielfunktion, Wertebereich der Variablen oder auch Art der Nebenbedingungen werden die unterschiedlichen Optimierungsprobleme klassifiziert. Wird nur gefordert, dass die Variablen alle aus dem Bereich der ganzen Zahlen stammen müssen, so wird von ganzzahliger Optimierung (engl. integer program(ming), kurz IP) gesprochen. Hierbei ist die Zulässigkeit des Lösungsraumes oft von besonderem Interesse: Existiert überhaupt eine ganzzahlige Lösung

$$z^{\min} = \arg \min_{z \in \mathbb{Z}^n} f(z)$$

zu dem Problem? Werden dazu nur lineare Zielfunktionen und Nebenbedingungen betrachtet, wird dies als ganzzahlige lineare Programmierung (kurz ILP) bezeichnet. Ein allgemeines ILP mit m Ungleichungen und r Gleichungen lässt sich dann mit $a \in \mathbb{Z}^n$, $B \in \mathbb{Z}^{m \times n}$, $C \in \mathbb{Z}^{r \times n}$, $b \in \mathbb{Z}^m$ und $c \in \mathbb{Z}^r$ wie folgt formulieren:

$$\min \{a^\top z : z \in \mathbb{Z}^n, Bz \leq b, Cz = c\}.$$

Ein anderer Spezialfall von IP ist die Beschränkung auf binäre Variablen $x \in \{0, 1\}^n$, die sogenannte 0/1-Programmierung oder auch binäre Optimierung. Wird dabei ein Minimum über einer speziellen Zielfunktion der Form

$$q(x) = x^\top Qx$$

mit einer Matrix $Q \in \mathbb{R}^{n \times n}$ ohne Nebenbedingungen gesucht, spricht man von QUBO (für engl. quadratic unconstrained binary optimization):

$$\min_{x \in \{0, 1\}^n} x^\top Qx.$$

Mit $x_j^2 = x_j$ für $x_j \in \{0, 1\}$ zeigt sich, dass dieses Matrixprodukt in lineare und quadratische Summanden zerlegbar ist und die Betrachtung oberer Dreiecksmatrizen für Q genügt:

$$q(x) = \sum_{j=1}^n Q_{j,j}x_j + \sum_{\substack{j,k=1 \\ j < k}}^n (Q_{j,k} + Q_{k,j})x_jx_k. \quad (\text{I})$$

Damit gehört q zu den pseudobooleschen Funktionen. Dies sind multilineare Polynome der Form

$$b(x) = a_\emptyset + \sum_{j=1}^n a_{\{j\}}x_j + \sum_{\substack{j,k=1 \\ j < k}}^n a_{\{j,k\}}x_jx_k + \sum_{\substack{j,k,l=1 \\ j < k < l}}^n a_{\{j,k,l\}}x_jx_kx_l + \dots = \sum_{I \subseteq \{1, \dots, n\}} a_I \prod_{j \in I} x_j \quad (\text{II})$$

mit $a_I \in \mathbb{R}$ für alle $I \subseteq \{1, \dots, n\}$ und $b : \{0, 1\}^n \rightarrow \mathbb{R}$. Sie stellen eine Verallgemeinerung der booleschen Funktionen dar, deren Wertebereich auf $\{0, 1\}$ begrenzt ist. Sollten alle $a_I \geq 0$ sein, so nennt sich dies eine Posiform. Klar ist, dass der Minimalwert einer solchen Funktion gleich a_\emptyset und $\mathbb{0}_n$ eine entsprechende Optimallösung ist. Dies gilt also insbesondere für QUBOs mit $Q \in \mathbb{R}_{\geq 0}^{n \times n}$, deren Optimalwert Null ist.

Die Schreibweise von (I) zeigt auch deutlich die Verbindung von QUBOs zur Graphentheorie: Jeder quadratischen Funktion q mit oberer Dreiecksmatrix Q liegt ein Interaktionsgraph G_q wie folgt zugrunde: $G_q = (V, E)$ ist ein einfacher, ungerichteter Graph mit Knotenmenge $V = \{1, \dots, n\}$ und Kantenmenge E bestehend aus zweielementigen Teilmengen von V . Jedem Knoten $j \in V$ ist ein Gewicht $g_j = g(j) := Q_{j,j}$ zugeordnet. Für $Q_{j,k} \neq 0$ sind die Knoten j und k gekoppelt, also $\{j, k\} \in E$. Dieser Kante ist nun die Stärke $S_{j,k} = S(j, k) := Q_{j,k}$ zugewiesen. Die Abbildung $S : E \rightarrow \mathbb{R}$ wird dann auch Bewertungsfunktion genannt. Ein Beispiel für einen gewichteten Interaktionsgraphen eines QUBOs ist in Abbildung 1(a) gezeigt. Damit lässt sich ein QUBO auch mit

$$\min_{x \in \{0, 1\}^V} \sum_{v \in V} g_v x_v + \sum_{\{v, w\} \in E} S_{v, w} x_v x_w \quad (\text{III})$$

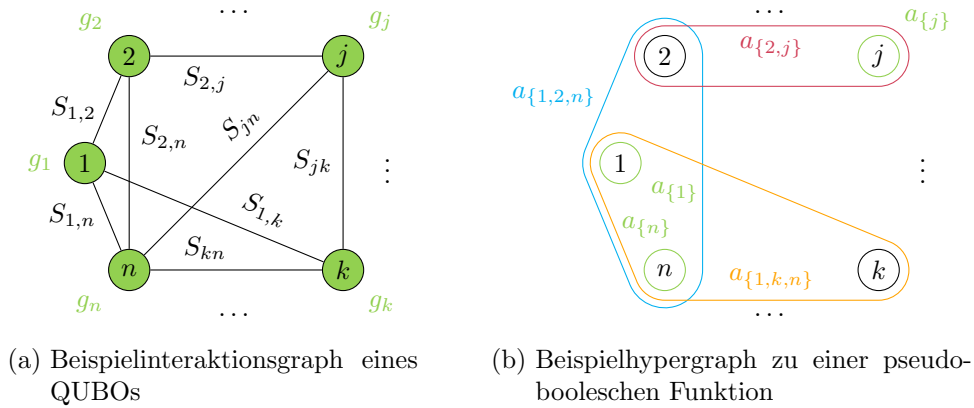


Abb. 1: Übertragung auf Graphen

über die Graphenformulierung darstellen.

Wurde durch Minimierung von q eine Lösung x^{\min} des QUBOs gefunden und ist $x_j^{\min} = 1$ für ein j , wird der Knoten j als aktiv bezeichnet. Eine Kante des Graphen $\{j, k\}$ ist genau dann aktiv, wenn $x_j^{\min} = x_k^{\min} = 1$. Die Menge der aktiven Knoten von G_q bildet damit einen bezüglich dieser Gewichte und Stärken minimalen Teilgraphen. Die Suche danach und ähnliche Probleme in Graphen werden besonders unter dem Begriff der Netzwerkoptimierung gefunden.

Analog lassen sich auch die allgemeinen pseudobooleschen Funktionen auf das Konzept der Hypergraphen übertragen. Bei diesen Graphen wird der Begriff der Kante verallgemeinert. Statt nur zwischen zwei Knoten zu bestehen, enthält eine Hyperkante eine Knotenteilmenge von V mit einer beliebigen Kardinalität. Dementsprechend bilden die a_I die Gewichtung der Hyperkante über den entsprechenden Knoten in $I \subseteq V$. Das Prinzip ist beispielhaft in Abbildung 1(b) verdeutlicht. Das Minimum der pseudobooleschen Funktion entspricht auch hier einem minimalen Teilgraphen.

Auch in der statistischen Physik sind diese Probleme häufig Gegenstand von Untersuchungen, hier in Form des Ising-Spin-Glas-Modells: Der Spin eines Atoms oder Ions bestimmt sein magnetisches Moment. In einer Anordnung mehrerer solcher Teilchen, dem Spin-Glas, beeinflussen sich deren Spins gegenseitig. Bei der Betrachtung der Gesamtmagnetisierung dieses Systems werden oft stark vereinfachte Modelle genutzt. So beschränkt man sich im Ising-Modell auf einen einzelnen Freiheitsgrad für die Drehrichtung der Spins. Damit kann dieser nur noch zwei diskrete Zustände annehmen: Ohne Beachtung der Orientierung im Raum zeigt er entweder auf- oder abwärts, also in positive oder negative Richtung. Dabei erhöhen beziehungsweise erniedrigen gleichgerichtete Teilchen die Energie je nach Vorzeichen der Wechselwirkung. Daraus ergibt sich für das System eine quadratische Energiefunktion

$$q(s_1, \dots, s_n) = \sum_{j=1}^n h_j s_j + \sum_{\substack{j,k=1 \\ j>k}}^n J_{j,k} s_j s_k$$

mit $h \in \mathbb{R}^n$ und $J \in \mathbb{R}^{n \times n}$ über den Variablen $s_j \in \{-1, +1\}$ für $j = 1, \dots, n$. Mit der Transformation $x_j = \frac{1}{2}(s_j + 1)$ beziehungsweise $s_j = 2x_j - 1$ zeigt sich, dass sich die Zielfunktionen von QUBOs und Ising-Modellen jeweils ineinander umwandeln

lassen. Da dabei ein quadratischer Term ein solcher bleibt, sind die Konzepte äquivalent und die Interaktionsgraphen beider Problemstellungen gleich. Die Verallgemeinerung des Begriffes der Nachbarschaft von mehr als zwei Teilchen, analog zum Konzept des Hypergraphen, wird hier Markov Random Field höherer Ordnung genannt.

2.2 Komplexität und Lösungsmethoden

2.2.1 Klassische Einordnung und allgemeine Algorithmen

Die Entscheidungsvariante des QUBOs, ob für ein gewähltes $r \in \mathbb{R}$ ein $x \in \{0, 1\}^n$ existiert mit $q(x) = x^\top Qx \leq r$, ist ein NP-vollständiges Problem: Es liegt in NP, da eine entsprechende Belegung der n binären Variablen direkt ein Zertifikat zur Bestätigung bildet. Zudem wird in den folgenden Abschnitten gezeigt, dass sich die Suchvarianten verschiedener NP-vollständiger Probleme in polynomieller Zeit auf QUBOs reduzieren lassen. Dies bedeutet, es gibt im Allgemeinen vermutlich keinen klassischen Algorithmus, der ein QUBO über n Variablen in einer Zeit, die polynomial von der Eingabegröße n abhängt, lösen kann.

Obwohl das Problem für $Q \in \mathbb{R}_{\geq 0}^{n \times n}$ trivial ist, ist es selbst dann NP-vollständig, wenn diese Bedingung für die Diagonalelemente von Q relativiert wird und auch nur eines davon negativ ist. Dies zeigen S. Chakradhar und M. Bushnell in [11], indem sie den Interaktionsgraphen des QUBOs in einen logischen Schaltkreis transformieren. Auch die Einschränkung auf planare Graphen, wie bei ebenen Spin-Gläsern in einem magnetischen Feld, genügt nicht, um eine bessere Lösbarkeit zu erzielen, wie F. Barahona in [12] beweist.

Dennoch existieren speziell auf QUBOs abgestimmte Lösungsverfahren, wie der von M. Carter in [13] vorgeschlagene Branch-and-Bound-Algorithmus. F. Barahona, M. Jünger und G. Reinelt stellen in [14] ihre Erfahrungen mit der Berechnung über einem Schnittebenenalgorithmus vor. Des Weiteren existiert ein polynomielles Verfahren zur Bestimmung unterer Grenzen von P. Hammer, P. Hansen und B. Simeone [15]. In [16] wird von D. Li, X. Sun und C. Liu ein neuerer Branch-and-Bound-Algorithmus vorgestellt. Eine weitere Möglichkeit stellt die Linearisierung der Zielfunktion dar. Dabei ergeben sich nach [17] jedoch für jeden quadratischen Term je eine zusätzliche reelle Variable und zwei Nebenbedingungen. Das heißt, das Problem kann auf eine Größe von bis zu $n + \frac{1}{2}n(n-1)$ Variablen und $n(n-1)$ Bedingungen wachsen. Diese gemischt-ganzzahligen linearen Probleme (kurz MILP für engl. mixed-integer linear programming) lassen sich dann mit den entsprechenden LP-Methoden wie beispielsweise durch Branch-and-Cut-Algorithmen lösen.

2.2.2 Polynomielle Lösbarkeit

Für einige Spezialfälle existieren effiziente Algorithmen. Sind beispielsweise alle Einträge von Q außer denen auf der Diagonalen nichtpositiv, so ist die polynomielle Lösbarkeit gegeben, wie J. Picard und H. Ratliff in [18] zeigen. In diesem Fall kann das Problem auf die Suche nach einem minimalem Schnitt in einem Netzwerk zurückgeführt

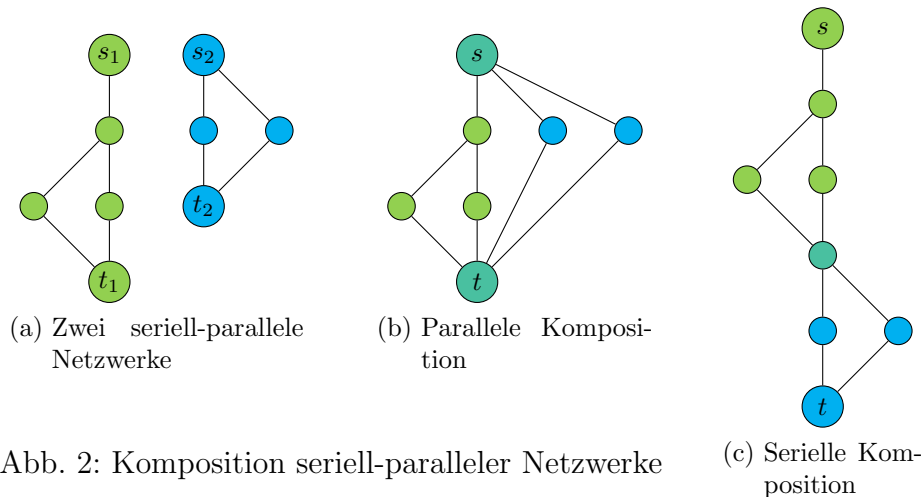


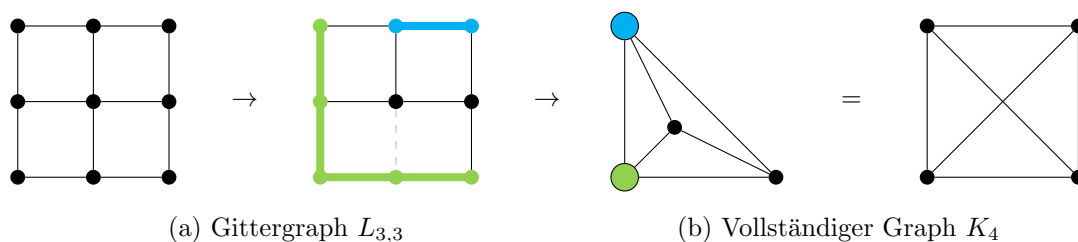
Abb. 2: Komposition seriell-paralleler Netzwerke

werden, welches im Gegensatz zum maximalen Schnitt effizient berechnet werden kann. Ist Q eine positiv semidefinite Matrix, gilt $y^\top Q y \geq 0$ für alle $y \in \mathbb{R}^n$. Damit ist das Minimum der Funktion gleich 0 und wird durch den binären Vektor \mathbb{O}_n angenommen.

Weiterhin konnte F. Barahona zeigen, dass auch für seriell-parallele Interaktionsgraphen mit beliebigen Stärken und Gewichten ein polynomieller Algorithmus existiert [19]. Es existieren verschiedene Arten diese Graphen, die besonders für die Elektrotechnik interessant sind, zu definieren. Im Folgenden wird sich im Wesentlichen auf den rekursiven Aufbau aus [20] bezogen.

Ein minimales seriell-paralleles Netzwerk (kurz SP-Netzwerk) besteht aus zwei verschiedenen Knoten, einer Quelle s und einer Senke t , die durch eine Kante verbunden sind. Aus zwei SP-Netzwerken N_1 mit s_1, t_1 und N_2 mit s_2, t_2 lässt sich ein neues SP-Netzwerk durch zwei verschiedene Kompositionsoperationen bilden. Diese sind in Abbildung 2 verdeutlicht. Zum einen können die beiden Netzwerke seriell verbunden werden, indem o.B.d.A. N_1 an N_2 angehängt wird. Dabei werden t_1 und s_2 zu einem inneren Knoten verschmolzen und s_1 und t_2 werden die neue Quelle beziehungsweise Senke des entstandenen Netzwerks. Zum anderen gibt es die parallele Komposition, bei der s_1 und s_2 zu einer neuen Quelle und t_1 und t_2 zu einer neuen Senke vereinigt werden. Ein Graph ist dann seriell-parallel, wenn zwei seiner Knoten als Quelle und Senke ausgewählt werden können und damit ein SP-Netzwerk gebildet wird. Dies kann in einer Zeit, die linear in der Zahl der Knoten ist, herausgefunden werden. Über diesen Aufbau kann niemals ein vollständiger Graph mit vier Knoten, welcher mit K_4 bezeichnet wird, gebildet werden, auch nicht durch die Kontraktion von Kanten, also dem Verschmelzen der zwei entsprechenden Knoten zu einem. Daher ist ein Graph genau dann seriell-parallel, wenn keiner seiner Minoren K_4 bildet [21]. Daran ist auch erkennbar, dass SP-Graphen immer planar sind: Nach dem Satz von Kuratowski sind die kleinsten nichtplanaren Graphen der vollständige Graph mit fünf Knoten K_5 und der vollständig bipartite Graph mit je drei Knoten $K_{3,3}$ und diese haben jeweils K_4 als Minor.

Zweidimensionale Gittergraphen mit n Zeilen und m Spalten von Knoten werden im Folgenden mit $L_{n,m}$ bezeichnet. Bei einer Breite und Länge von mehr als zwei Knoten zählen diese, wie Abbildung 3 zeigt, nicht zu den SP-Graphen. Dennoch konnte F. Barahona in [12] auch für zweidimensionale Spingläser von Teilchen, die in einer ebenen

Abb. 3: K_4 ist ein Minor des Gittergraphen $L_{3,3}$

Gitterstruktur angeordnet sind, zeigen, dass sie in polynomieller Zeit gelöst werden können.

2.2.3 Randomisierte Komplexitätsklassen

Bei einigen Problemen, wie beispielsweise dem Knacken von Verschlüsselungen, ist es oft nicht ausreichend, nur Worst-Case-Betrachtungen der Algorithmen durchzuführen. Es muss auch festgestellt werden, ob möglicherweise ein nicht exakter Algorithmus existiert, der das Problem in einer bestimmten Zeit und bei mehrfachen Durchführungen in einem signifikant großen Anteil lösen kann oder nicht. Um dies zu überprüfen, wird auf das Konzept der probabilistischen Turingmaschine (kurz PTM) zurückgegriffen. Bei dieser wird nach jedem Rechenschritt eine von zwei Übergangsfunktionen durch einen Münzwurf gewählt. Damit lässt sich die Klasse der probabilistischen Polynomialzeit (kurz PP) für Probleme einführen, für die eine PTM existiert, welche die Antwort für jede Eingabeinstanz in polynomieller Laufzeit mit einer Wahrscheinlichkeit $> \frac{1}{2}$ findet. Die Wahrscheinlichkeit, dass die Antwort falsch ist, kann dabei von der Form $\frac{1}{2} - \frac{1}{2^{p(n)}}$ sein, wobei p ein Polynom in n ist. Diese lässt sich nur durch eine exponentielle Zahl an Wiederholungen des Algorithmus beliebig klein machen.

Die Klasse PP ist in PSPACE enthalten und umfasst sowohl NP als auch coNP. Denn jedes $L \in NP$ lässt sich durch eine PTM entscheiden: Sei M eine nichtdeterministische Turingmaschine, die L entscheidet. Dann wählt die PTM im ersten Schritt gleichmäßig zufällig zwischen einer der beiden folgenden Möglichkeiten: Entweder wird ein zufälliger Pfad von M ausgeführt und das Ergebnis ausgegeben oder die Eingabe wird ohne Berechnung akzeptiert. Damit gilt für eine Eingabe $x \notin L$, dass x zu genau 50 Prozent akzeptiert wird, da die Ausgabe im ersten Fall immer nein ist. Für $x \in L$ ist die Fehlerwahrscheinlichkeit größer als $\frac{1}{2}$, da mindestens ein Pfad von M die Antwort ja liefert [22]. Damit gehören auch QUBOs, sowie alle weiteren NP-vollständigen Probleme, ebenfalls zu der Klasse PP. Ein bekanntes PP-vollständiges Problem ist MAJSAT, bei dem entschieden werden soll, ob eine aussagenlogische Formel von mehr als der Hälfte der möglichen Eingaben erfüllt wird.

Die Einschränkung der Fehlerwahrscheinlichkeit auf $\frac{1}{2} - \varepsilon$ für ein konstantes $\varepsilon \in (0, \frac{1}{2})$ ergibt dann die Klasse BPP, für bounded error probabilistic polynomial time. Dabei kann ε beliebig festgelegt werden und die Fehlerwahrscheinlichkeit lässt sich durch eine polynomielle Iterationszahl beliebig verringern. Klar ist $P \subseteq BPP \subseteq PP$. Bisher konnte kein solcher Algorithmus für NP-vollständige Probleme, also insbesondere QUBOs,

gefunden werden. Daher ist noch unklar, wie diese Komplexitätsklasse zu NP in Beziehung steht. Ebenso ist bisher kein BPP-vollständiges Problem bekannt. Es wird daher auch vermutet, dass keines existiert [22].

2.2.4 Quantenkomplexität

Mit einem Quantencomputer lassen sich im Gegensatz zum klassischen Rechner Ergebnisse nur mit einer gewissen Wahrscheinlichkeit bestimmen. 1985 führte daher D. Deutsch den Begriff der Quanten-Turingmaschine (kurz QTM) ein [4], welche einen sogenannten Quantenalgorithmus ausführen kann. Die Existenz einer effizienten universellen Quanten-Turingmaschine wurde dann 1993 gezeigt [23]. Damit kann analog zu BPP die Klasse BQP, für bounded error quantum polynomial time, über alle Probleme definiert werden, für die eine QTM existiert, die in polynomieller Zeit mit einer Wahrscheinlichkeit von $\frac{1}{2} + \varepsilon$ für ein konstantes $\varepsilon \in (0, \frac{1}{2})$ eine Lösung findet.

Da Quantenzustände komplexen Vektoren und die Transformationen Matrix-Vektor-Multiplikationen entsprechen, kann ein klassischer Rechner ein Quantensystem, also eine TM eine QTM, simulieren. Damit liegt BQP in der Klasse PSPACE. Es wird jedoch davon ausgegangen, dass es keinen polynomiellen klassischen Simulationsalgorithmus gibt. Diese klassische Berechenbarkeit impliziert aber auch, dass sich Probleme wie das Halteproblem auch nicht auf Quantencomputern lösen lassen.

Aufgrund der Eigenschaften eines Quantenbits (kurz Qubit), welches sich in den zwei Basiszuständen 0 und 1 gleichzeitig befinden kann und durch Messen völlig zufällig in einen dieser Zustände springt, kann auch eine PTM von einer QTM simuliert werden. Dabei wird nach jedem Rechenschritt ein gleichmäßig zwischen 0 und 1 verteiltes Hilfsqubit gemessen und die Übergangsfunktion entsprechend des Ergebnisses gewählt. Damit gilt also $BPP \subseteq BQP$. Dabei wird aufgrund des polynomiellen Faktorisierungsalgorithmus für Quantencomputer nach Shor eine echte Inklusion vermutet [24].

Ein anderes Konzept ist der sogenannte adiabatische Quantencomputer, der nicht auf dem konventionellen Schaltkreismodell beruht, sondern einen Optimierer für QUBOs darstellt. Dadurch lassen sich auch alle Suchvarianten der anderen NP-Probleme durch die in den folgenden Abschnitten dargestellten Reduktionen mit diesem Rechnermodell lösen. Es konnte weiterhin gezeigt werden, dass das adiabatische polynomiell äquivalent zum konventionellen Quantencomputing ist, da mit beidem effizient das jeweils andere simuliert werden kann [25] [26].

Obwohl Quantencomputer durch den Quantenparallelismus einen entscheidenden Vorteil gegenüber klassischen Rechnern bei der Berechnung NP-schwerer Probleme zu besitzen scheinen, existiert bisher keine konkrete Aussage darüber, ob NP-Probleme damit effizient lösbar sind. Insbesondere ist aufgrund der physikalischen Hintergründe die Abschätzung der Laufzeit nur sehr schwer möglich. Es ist soweit nicht im Vorhinein ersichtlich, wie groß die Erfolgswahrscheinlichkeit für das Finden der Optimallösung ist. Da aber einer der wenigen Quantenalgorithmen, der Grover-Algorithmus, für die Suche in einer unsortierten Datenbank nur einen quadratischen Vorteil gegenüber klassischen Wegen liefert [27], wird vermutet, dass $NP \not\subseteq BQP$ gilt. Die entsprechenden bekannten Inklusionen sind in Abbildung 4 verdeutlicht.

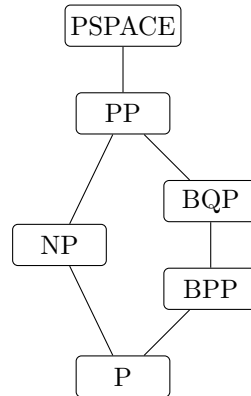


Abb. 4: Inklusionsdiagramm der Komplexitätsklassen

2.3 Reduktionen

2.3.1 Direkte Umsetzung

Die folgenden Beispiele zeigen die direkte Übertragung zweier bestimmter Graphenprobleme auf QUBOs. Es zeigt sich, dass die entstehenden quadratischen Terme der Zielfunktion hierbei genau den Kanten des Ausgangsgraphen entsprechen. Damit ist der Interaktionsgraph, der die Kopplungen der binären Variablen repräsentiert, gleich dem Ausgangsgraphen. Erfüllt dieser eine der Bedingungen aus 2.2.2, ergibt sich somit auch die polynomielle Lösbarkeit des Problems. Diese Problemstellungen stellen eine Besonderheit dar, denn im Normalfall wird der Interaktionsgraph durch die Reduktion auf ein QUBO wesentlich komplexer. Dies zeigt sich in den nächsten Abschnitten.

Stabile Menge

Eine stabile Menge in einem ungerichteten Graphen $G = (V, E)$ ist eine Teilmenge von Knoten $W \subseteq V$, die alle untereinander nicht durch Kanten verbunden sind, also für alle $v, w \in W$ gilt $\{v, w\} \notin E$. Im Komplementgraphen, dem Graphen, der nur genau die Kanten enthält, die der Ausgangsgraph nicht besitzt, entspricht dies genau einer Clique, also einem vollständigen Teilgraphen. Diese sind besonders in sozialen Netzwerken von Interesse, wo sie eine Gruppe von Menschen repräsentieren, die alle untereinander befreundet sind. Das Entscheidungsproblem, ob der Graph eine stabile Menge mit mindestens $r \in \mathbb{N}$ Knoten besitzt, ist NP-vollständig [28].

Die Suche nach der größten stabilen Menge im Graphen, deren Kardinalität die Stabilitätszahl des Graphen $\alpha(G)$ definiert, lässt sich sehr einfach als ein QUBO-Problem formulieren: Um eine maximale Lösung zu erhalten, werden die Gewichte aller Knoten auf einen negativen Wert $-A$ mit $A > 0$ gesetzt. Da bei dem gesuchten Teilgraphen keine Kante vorhanden sein darf, wird die Aktivierung jeder Kante gleichmäßig mit einer Strafe $B > 0$ belegt. Mit $x \in \{0, 1\}^V$ und $x_v = 1$ genau dann, wenn v zur Lösungsmenge gehört, ergibt sich die Zielfunktion

$$q(x) = -A \sum_{v \in V} x_v + B \sum_{\{u, v\} \in E} x_u x_v =: q_A(x) + q_B(x).$$

Dass sich nicht beliebige Werte für A und B wählen lassen, zeigt folgendes Theorem,

welches analog zu den Aussagen in [29] erstellt wurde.

Theorem 2.1 Für $B > A$ liefern Optimallösungen von $\min \{q(x) : x \in \{0, 1\}^V\}$ kardinalitätsmaximale stabile Mengen von $G = (V, E)$.

Beweis: Sei X die Menge der $x \in \{0, 1\}^V$, die eine stabile Menge von G repräsentieren. Es gilt $q_B(x) = 0$ genau dann, wenn $x \in X$. Damit minimieren die kardinalitätsmaximalen stabilen Mengen die Zielfunktion über X . Da

$$q(\mathbb{0}_V) = 0 > q(e_v) = -A$$

für jeden beliebigen Graphen G und alle $v \in V$ gilt, besitzt jede Optimallösung x^{\min} des Problems mindestens einen aktiven Knoten, welcher bereits eine stabile Menge des Graphen darstellt. Es bleibt zu zeigen, dass x^{\min} keine zwei aktiven Knoten $v, w \in V$ besitzt, die adjazent sind. Wäre dies der Fall, würde die Aktivierung der Kante mit $x_v x_w = 1$ in der Zielfunktion einen Summanden $+B$ erzeugen und es gälte

$$q(x^{\min}) \geq -A \|x^{\min}\|_1 + B,$$

wobei $\|\cdot\|_1$ die Manhattan-Norm ist, welche die Einsen eines binären Vektors zählt. Mit $B > A$ widerspricht dies jedoch der Minimalität von x^{\min} . Durch Deaktivierung von v in x^{\min} ergibt sich beispielsweise ein Optimalwert, der um $B - A$ geringer ist. Es gilt also $x^{\min} \in X$. \square

Die Strafe für die Aktivierung einer Kante muss also nur größer sein als der betragsmäßige Gewinn, der durch die Aktivierung eines weiteren inzidenten Knoten erhalten wird. Wird für die Knotengewichte -1 und für jede Kante die Stärke $B = 2$ gewählt, lässt sich aus dem Betrag der Zielfunktion über der Lösung x^{\min} direkt die Kardinalität auslesen:

Folgerung 2.2 Es gilt

$$\alpha(G) = \left| \min \left\{ - \sum_{v \in V} x_v + 2 \sum_{\{u,v\} \in E} x_u x_v : x \in \{0, 1\}^V \right\} \right|.$$

Ebenso kann man durch die analoge Aufstellung der Zielfunktion über dem Komplementgraphen \bar{G} die Cliquenzahl $\omega(G)$ des Graphen, also die größte Kardinalität einer Clique, bestimmen:

Folgerung 2.3 Es gilt

$$\omega(G) = \alpha(\bar{G}) = \left| \min \left\{ - \sum_{v \in V} x_v + 2 \sum_{\{u,v\} \notin E} x_u x_v : x \in \{0, 1\}^V \right\} \right|.$$

Analog zu diesen Folgerungen, lassen sich in gewichteten Graphen mit $g : V \rightarrow \mathbb{R}$ stabile Mengen beziehungsweise Cliques mit einem maximalen Gewicht finden. Statt daher ein Gewicht von $-A$ für die Knoten zu wählen, kann in der ersten Summe $-Ax_v$ durch $-g_v x_v$ ersetzt werden. Mit $B > \max\{0, g_v : v \in V\}$ liefert das Minimum der Zielfunktion q die gesuchte Lösung.

Maximaler Schnitt

Ein Schnitt in einem Graphen $G = (V, E)$ ist eine Partition der Knotenmenge V in zwei disjunkte Mengen $W \subset V$ und $U = V \setminus W$. Damit wird eine Menge von Schnittkanten zwischen den Partitionen eindeutig definiert, die je einen Knoten in einer der Partitionen besitzen. Dies sind die aus W ausgehenden Kanten

$$\delta(W) = \{\{w, u\} \in E : w \in W, u \in V \setminus W\}.$$

Ist der Graph mit einer Bewertungsfunktion $S : E \rightarrow \mathbb{R}$ versehen, so wird mit

$$S(\delta(W)) = \sum_{e \in \delta(W)} S(e) = \sum_{\substack{\{w, u\} \in E \\ w \in W \\ u \in V \setminus W}} S(w, u).$$

auch für den Schnitt eine Gewichtung erhalten. Das Max-Cut-Problem bezeichnet die Suche nach der Partition (W, U) mit $W \cup U = V$ und $W \cap U = \emptyset$, so dass $S(\delta(W))$ maximal wird. Die Suche nach einem Schnitt mit minimalen Gewicht ist nach dem Min-Cut-Max-Flow-Theorem in polynomieller Zeit durchführbar. Im Gegensatz dazu ist das entsprechende Entscheidungsproblem, ob für den Graphen G und ein $r \in \mathbb{R}$ ein Schnitt mit $S(\delta(W)) \geq r$ existiert, NP-vollständig. Es gehört wie das Cliquesproblem zu den 21 Problemen, die R. Karp in [28] betrachtet. In [30] wurde für den Spezialfall der Suche eines maximalen Schnittes in planaren Graphen ein polynomieller Algorithmus gefunden.

Die Darstellung von Max-Cut als QUBO wurde in [31] von E. Boros und P. Hammer gezeigt. Für $x \in \{0, 1\}^V$ mit

$$x_v^W = \begin{cases} 1, & \text{wenn } v \in W \\ 0, & \text{wenn } v \in U = V \setminus W \end{cases}$$

lässt sich das Problem folgendermaßen umformen:

$$\begin{aligned} \text{MaxCut}(G, S) &= \max_{W \subset V} S(\delta(W)) \\ &= \max_{W \subset V} \sum_{\substack{\{w, u\} \in E \\ w \in W \\ u \in V \setminus W}} S(w, u) \cdot x_w^W (1 - x_u^W) \\ \text{da } x_u^W (1 - x_w^W) &= 0 \text{ } \forall u \notin W \text{ und } \forall w \in W \\ &= \max_{W \subset V} \sum_{\{w, u\} \in E} S(w, u) (x_w^W (1 - x_u^W) + x_u^W (1 - x_w^W)) \\ &= \max_{x \in \{0, 1\}^V} \sum_{\{w, u\} \in E} S(w, u) (x_w(1 - x_u) + x_u(1 - x_w)) \\ &= - \min_{x \in \{0, 1\}^V} \sum_{\{w, u\} \in E} S(w, u) (2x_w x_u - x_u - x_w) \\ &= - \min_{x \in \{0, 1\}^V} \sum_{\{w, u\} \in E} 2S(w, u) \cdot x_w x_u - \sum_{v \in V} \sum_{w \in d(v)} S(v, w) \cdot x_v. \end{aligned}$$

Das bedeutet, der entsprechende QUBO-Graph erhält als Kantenstärken das doppelte der Stärken des Ursprungsgraphen und als Knotengewichte die negierte Summe über

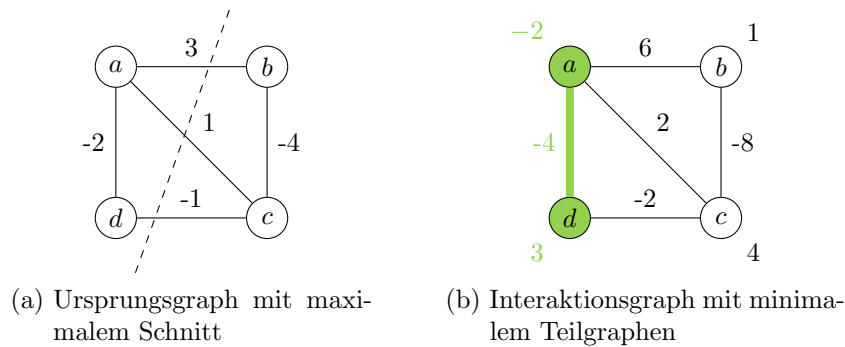


Abb. 5: Beispielgraphen zu einem Max-Cut-Problem

den Stärken der ausgehenden Kanten. Ein Beispiel dazu ist in Abbildung 5 zu sehen. Der maximale Schnitt des gezeigten Graphen beträgt 3. Die Gewichte des QUBO-Graphen in 5(b) wurden entsprechend berechnet und der markierte Teilgraph ist eine Lösung des QUBOs. Die unmarkierten Knoten ergeben jedoch ebenfalls eine Lösung, da sie denselben Schnitt definieren und damit denselben Optimalwert liefern. Demnach werden aufgrund der Symmetrie in den x_v -Variablen immer mindestens zwei Optima erhalten. Dies lässt sich aber durch vorheriges Festlegen von $x_v = 1$ oder 0 für einen einzelnen Knoten $v \in V$ vermeiden.

Umgekehrt lässt sich auch jedes QUBO als eine Max-Cut-Instanz darstellen. Dazu muss nach [31] jedoch ein weiterer Knoten zum Interaktionsgraphen hinzugefügt und mit allen bereits vorhandenen Knoten gekoppelt werden. Damit gehen die meisten Eigenschaften des ursprünglichen Graphen verloren. Daher sind beispielsweise QUBOs auf planaren Graphen nicht zwingend besser lösbar.

2.3.2 Lineare Nebenbedingungen an 0/1-Variablen

Sind in einem Optimierungsproblem zwar bereits eine lineare oder quadratische Zielfunktion über binären Variablen gegeben jedoch ebenso Nebenbedingungen hinzugefügt, so müssen diese für ein QUBO in die Zielfunktion integriert werden. Dazu wird das Konzept der Straffunktionen genutzt, welches sich auf viele verschiedene Fälle übertragen lässt: Allgemein wird für eine Bedingung β eine positive Funktion p_β für Belegungen $x \in \{0, 1\}^n$ eingeführt mit

$$p_\beta(x) \begin{cases} > 0, & x \text{ verletzt } \beta, \\ = 0, & \text{sonst.} \end{cases}$$

Damit ergibt sich für ein Optimierungsproblem mit den Bedingungen β_1, \dots, β_m

$$\min \{f(x) : x \in \{0, 1\}^n \text{ erfüllt } \beta_1, \dots, \beta_m\} = \min \left\{ f(x) + \sum_{j=1}^m P_j p_{\beta_j}(x) : x \in \{0, 1\}^n \right\}.$$

Wie die Funktionen p_{β_j} aussehen, hängt dabei, wie die Höhe der Gewichtungen $P_j > 0$, stark von den Bedingungen ab. Die Anpassung der Strafgewichte wird im letzten Abschnitt kurz näher erläutert. Im Folgenden wird gezeigt, wie die zwei Arten von linearen

Nebenbedingungen, Gleichungen und Ungleichungen, nach [32] in ein QUBO integriert werden können. Dadurch ergeben sich jedoch meist starke Unterschiede zwischen dem Ausgangsgraphen und dem Interaktionsgraphen des entstandenen QUBOs.

Gleichungen

Nebenbedingungen der Form $Cx = c$ mit $C \in \mathbb{R}^{m \times n}$ und $c \in \mathbb{R}^m$ sind lineare Gleichungen. Sie lassen sich sehr leicht durch die Bestrafungsfunktion

$$p_=(x) = \|Cx - c\|^2$$

für eine Vektornorm $\|\cdot\| : \mathbb{R}^n \rightarrow \mathbb{R}_{\geq 0}$ erzwingen. Üblicherweise wird die euklidische Norm $\|\cdot\|_2$ gewählt. Klar ist hierbei, dass $p_=(x) \geq 0$ für alle $x \in \{0, 1\}^n$ gilt und $p_=(x) = 0$ genau dann, wenn die Bedingung erfüllt ist. Möchte man die m Gleichungen mit unterschiedlichen Gewichten $P_j > 0$ für $j = 1, \dots, m$ versehen, ist dies über

$$p_=(x) = \sum_{j=1}^m P_j (C_{j,*}x - c_j)^2$$

entsprechend möglich. Durch das Hinzufügen der quadrierten Bedingungen ergeben sich neue quadratische Terme in der Zielfunktion, und zwar für alle Variablen x_k , für die $C_{j,k} \neq 0$ gilt. Diese Kopplungen erzeugen damit für jedes j je eine Clique im Interaktionsgraphen des QUBOs über all den entsprechenden Knoten. Dadurch kann sich die Zahl der Kopplungen im Graphen deutlich erhöhen.

Gilt dabei jedoch für eine Zeile j , dass $c_j \leq 0$ und dass $C_{j,k} \geq 0$ für alle $k = 1, \dots, n$, dann können die Variablen x_k , für die $C_{j,k} > 0$ gilt, direkt durch Nullsetzen aus allen Gleichungen und Ungleichungen eliminiert werden. Analoges gilt für den negierten Fall. Letztendlich ist es auch möglich, dass durch diesen Prozess nach Ausmultiplizieren und Aufsummieren aller Terme eine Konstante $a \in \mathbb{R}$ in der finalen quadratischen Zielfunktion q entsteht. Diese ist in QUBOs nicht vorgesehen, kann jedoch aus dem Optimierungsproblem herausgezogen werden:

$$\begin{aligned} & \min \{q(x) : x \in \{0, 1\}^n\} \\ & = a + \min \{q(x) - a : x \in \{0, 1\}^n\}. \end{aligned}$$

Ungleichungen

Ungleichungen werden ähnlich wie Gleichungen durch Quadrieren hinzugefügt. Jedoch müssen hierbei weitere Schwierigkeiten bedacht werden. Für eine Summe

$$\sum_{j=1}^n g_j x_j$$

ergeben sich bis zu 2^n verschiedene Summationsergebnisse. Diese sind durch die Summe der negativen g_j nach unten und die Summe der positiven g_j nach oben beschränkt mit

$$\check{\sigma}(g) = \sum_{j=1}^n \frac{1}{2}(1 - \text{sgn}(g_j))g_j = \sum_{j=1}^n \frac{1}{2}(g_j - |g_j|)$$

und

$$\hat{\sigma}(g) = \sum_{j=1}^n \frac{1}{2}(1 + \operatorname{sgn}(g_j))g_j = \sum_{j=1}^n \frac{1}{2}(g_j + |g_j|).$$

Sind nun Schranken an die Summe gegeben, muss eine Straffunktion gefunden werden, die nur die Belegungen für x bestraft, die diese Schranken verletzen, und für alle weiteren Möglichkeiten gleichermaßen 0 ergibt. Da eine Durchnummerierung von bis zu 2^n reellen Werten nicht sinnvoll ist, wird im Folgenden die Vorgehensweise bei einer Ungleichung der Form

$$l \leq \sum_{j=1}^n g_j x_j \leq h \quad (\text{IV})$$

mit ganzzahligen Gewichten g_j und ganzzahligen Schranken $l \geq \check{\sigma}(g)$ und $h \leq \hat{\sigma}(g)$ betrachtet. Aufgrund der Skalierbarkeit sind damit auch Fälle mit rationalen Parametern abgedeckt. Diese Umformungen sind zudem nur sinnvoll, wenn mindestens eine der Relationen echt ist.

Die Ungleichung (IV) lässt sich nun in eine Gleichung überführen, indem alle Möglichkeiten in einer zusätzlichen Variablen vereint werden:

$$\sum_{j=1}^n g_j x_j = z,$$

wobei für die sogenannte Schlupfvariable z gilt, $z \in \mathbb{Z}$ und $l \leq z \leq h$. Da nur 0/1-Variablen verfügbar sind, muss für z eine entsprechende Darstellung gefunden werden. Mit den binären Variablen $u_l, \dots, u_h \in \{0, 1\}$ bietet sich zum Beispiel die Möglichkeit einer Art Unärdarstellung über

$$\sum_{j=1}^n g_j x_j = \sum_{k=l}^h k u_k.$$

Wird bei der Optimierung für die linke Summe ein Wert $z^{\min} \in \{l, \dots, h\}$ angenommen, so wird $u_{z^{\min}}$ aktiviert. Da hierbei immer nur genau eine der $h - l + 1$ Schlupfvariablen u_k aktiv sein darf, muss

$$\sum_{k=l}^h u_k = 1$$

als Bedingung hinzugefügt werden. Die beiden entstandenen Gleichungen können analog zum vorherigen Abschnitt als Straffunktion formuliert werden, aber mit den zusätzlichen Variablen $u \in \{0, 1\}^{h-l+1}$:

$$p_{\leq}(x, u) = \left(\sum_{j=1}^n g_j x_j - \sum_{k=l}^h k u_k \right)^2 + \left(\sum_{k=l}^h u_k - 1 \right)^2.$$

Im Interaktionsgraphen des entsprechenden QUBOs ergeben sich somit zusätzliche $h - l + 1 \leq \hat{\sigma}(g) - \check{\sigma}(g) + 1$ Knoten. Durch das erste Quadrat in $p_{\leq}(x, u)$ bilden diese zusammen mit den x_j , für die $g_j \neq 0$ ist, eine Clique mit einer Größe von bis zu $n + h - l + 1$ Knoten.

Bedeutung der Kopplungen

Ein spezieller Fall, der besonders in den Graphenproblemen eine Rolle spielt, sind Gewichte von 1 und eine Schranke von 1 für eine Teilmenge $J \subseteq \{1, \dots, n\}$ der Indizes:

$$\sum_{j \in J} x_j \leq 1. \quad (\text{V})$$

Analog zum vorherigen Abschnitt ist das Gleichsetzen mit einer einzigen Schlupfvariable $z \in \{0, 1\}$ und so das Aufstellen der entsprechenden Straffunktion möglich:

$$p_{\leq}(x, z) = \left(\sum_{j \in J} x_j - z \right)^2 = 2 \sum_{\substack{j, k \in J \\ j < k}} x_j x_k + z + (1 - 2z) \sum_{j \in J} x_j.$$

Da die Summe in der Ungleichung (V) jedoch nur zwei mögliche Werte, 0 und 1, annehmen kann, lässt sich diese Bedingung auch folgendermaßen, abweichend von [32], erzwingen: Das eindimensionale quadratische Polynom

$$\left(y - \frac{1}{2}\right)^2 - \frac{1}{4}$$

besitzt die Nullstellen 0 und 1. Diese bilden zugleich die Minimalstellen über den ganzen Zahlen. Eine passende Straffunktion, welche genau dann minimal ist, wenn $x = \mathbb{0}_n$ oder $x = e_j$ für ein $j = 1, \dots, n$, ist demnach ebenso

$$\tilde{p}_{\leq}(x) = \left(\sum_{j \in J} x_j - \frac{1}{2} \right)^2 - \frac{1}{4}. \quad (\text{VI})$$

Diese Funktion lässt sich analog für alle Fälle aufstellen, in denen nur zwei aufeinanderfolgende ganze Zahlen als mögliche Werte zur Verfügung stehen. Eine Umformulierung ergibt hier weiterhin

$$\tilde{p}_{\leq}(x) = \left(\sum_{j \in J} x_j \right)^2 - \sum_{j \in J} x_j = 2 \sum_{\substack{j, k \in J \\ j < k}} x_j x_k.$$

Daran zeigt sich, dass keine der vorgestellten Varianten ein Alleinstellungsmerkmal besitzt und für einige spezielle Fälle durchaus einfachere Varianten gefunden werden und gegebenenfalls die Gewichte auf Knoten entfallen können. Jedoch spielen die Kopplungen unter den Variablen eine wichtige Rolle und es kann im Allgemeinen nicht davon ausgegangen werden, dass deren Zahl reduziert werden kann. Wird beispielsweise bei \tilde{p}_{\leq} auch nur eine der Kopplungen $x_j x_k$ weggelassen, so vergrößert sich die Menge der zulässigen Lösungen um den eigentlich unzulässigen Vektor $e_j + e_k$.

Reduktion der Schlupfvariablen

Die Zahl der Schlupfvariablen, wie sie in dem Abschnitt Ungleichungen erstellt werden, kann sehr schnell sehr groß werden. Daher soll hier eine in [29] angesprochene Möglichkeit dargestellt werden, diese zu reduzieren.

Mit $d := h - l$ ergibt eine Verschiebung der Grenzen in (IV) zu

$$0 \leq \sum_{j=1}^n g_j x_j - l \leq d.$$

Damit lassen sich die möglichen Werte der Summe, wie alle natürlichen Zahlen, in einer Binärdarstellung mit den zusätzlichen Variablen $y_0, \dots, y_D \in \{0, 1\}$ für $D = \lfloor \log_2(d) \rfloor$ ausdrücken:

$$\sum_{j=1}^n g_j x_j - l = \sum_{k=0}^D 2^k y_k.$$

Jedoch können die y -Variablen nur dann völlig frei mit Werten aus $\{0, 1\}$ belegt werden, wenn $d = 2^{D+1} - 1$, also $\log_2(d + 1) \in \mathbb{N}$, ist. Nur genau dann entspricht die Binärdarstellung $[d_D \dots d_1 d_0]_2$ von d einer Folge aus Einsen und es gilt

$$d - \sum_{k=0}^D 2^k y_k = \sum_{k=0}^D 2^k d_k - \sum_{k=0}^D 2^k y_k = \sum_{k=0}^D 2^k (d_k - y_k) \geq 0,$$

da $d_k - y_k = 1 - y_k \geq 0$ für alle $y_k \in \{0, 1\}$ ist. Gilt stattdessen $d < 2^{D+1} - 1$, muss die Differenz korrigiert werden. Da d aufgrund der Ganzzahligkeit maximal

$$2^{D+1} - 2 = 2(2^D - 1)$$

ist, lassen sich alle Möglichkeiten der Zahlenwerte von 0 bis d durch eine Binärzahl mit D Stellen und den aktivierbaren Rest $d - 2^D + 1$ darstellen:

$$\sum_{j=1}^n g_j x_j - l = \sum_{k=0}^{D-1} 2^k y_k + (d - 2^D + 1) y_D. \quad (\text{VII})$$

Damit lässt sich nun eine Straffunktion

$$p_{\leq}(x, y) = \left(\sum_{j=1}^n g_j x_j - l - \sum_{k=0}^{D-1} 2^k y_k - (d - 2^D + 1) y_D \right)^2$$

aufstellen, welche nur $D + 1 = \lfloor \log_2(h - l) \rfloor + 1 \leq \lfloor \log_2(\hat{\sigma}(g) - \check{\sigma}(g)) \rfloor + 1$ zusätzliche Variablen benötigt. Diese bilden dennoch zusammen mit den n ursprünglichen Variablen eine Clique im Interaktionsgraphen. Die Straffunktion ist auch gültig für den obigen Fall $d = 2^{D+1} - 1$, da dann der Koeffizient von y_D in (VII) genau 2^D ergibt. Für die anderen Fälle vergrößert sich die Zahl der Lösungen, da mehrere Belegungen der y_0, \dots, y_D den gleichen Wert in (VII) ergeben können. Dies hat jedoch keinen Einfluss auf die für die Lösung relevanten x -Variablen.

Wahl des Strafgewichtes

Durch Straffunktionen soll die Einhaltung von Bedingungen erzwungen werden. Dies setzt voraus, dass die Strafe hoch genug ist, um den bei einer Verletzung möglicherweise zu erlangenden Vorteil zu überwiegen. Klar ist, dass formal für das Strafgewicht P immer ∞ gewählt werden kann. Bei vielen Optimierern ist jedoch eine konkrete Wahl für P erforderlich. Daher muss in jedem Einzelfall abgeschätzt werden, wie groß P

mindestens gewählt werden sollte. Dies wird für viele konkrete Probleme bereits in [29] gezeigt.

Ausgangspunkt für diese Untersuchungen ist eine gegebene quadratische Zielfunktion q mit einer oberen Dreiecksmatrix Q als Abbildungsmatrix. Allgemein gilt, wird in einer Belegung der $x \in \{0, 1\}^n$ durch das Drehen einer Variable x_j von 0 auf 1 oder umgekehrt eine Bedingung β verletzt, ergibt sich eine Strafe $\Delta p_\beta(j) > 0$. Zugleich ergibt sich eine Änderung in der Zielfunktion q , welche negativ sein kann, durch Hinzukommen beziehungsweise Wegfallen des Gewichtes von x_j und der Stärken der Kopplungen, die x_j enthalten. Das heißt mit $\hat{\sigma}$ und $\check{\sigma}$, wie vorher definiert, gilt

$$\check{\sigma} (Q_{j,*} + Q_{*,j}^\top - Q_{j,j}e_j) \leq q(x) - q(x \oplus e_j) \leq \hat{\sigma} (Q_{j,*} + Q_{*,j}^\top - Q_{j,j}e_j),$$

wobei das Gewicht $Q_{j,j}$ abgezogen werden muss, da es sonst in der Summation der σ mit Faktor 2 vorkommt. Dann muss für das Gewicht P gelten

$$P\Delta p_\beta(j) > \max \{ -\check{\sigma} (Q_{j,*} + Q_{*,j}^\top), \hat{\sigma} (Q_{j,*} + Q_{*,j}^\top) \} - |Q_{j,j}|.$$

Betrifft die Bedingung β eine Teilmenge der Variablen mit Indizes $I \subseteq \{1, \dots, n\}$, muss sie für all diese Variablen erfüllt sein. Daher wird die Änderung der Zielfunktion nach oben durch

$$\Delta q := \max \{ -\check{\sigma} (Q_I), \hat{\sigma} (Q_I) \}$$

mit $Q_I = (Q_{j,k})_{j,k \in I}$ und die Änderung der Straffunktion nach unten durch

$$\Delta p_\beta := \min \{ \Delta p_\beta(j) > 0 : j \in I \}$$

abgeschätzt. Dadurch ergibt sich für P die Bedingung

$$P > \frac{\Delta q}{\Delta p_\beta}.$$

Werden mehrere Straffunktionen benötigt, müssen diese ebenso gegeneinander abgewägt und die entsprechenden Gewichte gewählt werden.

2.3.3 Ganzzahlige Probleme

Für viele Probleme, insbesondere NP-schwere, sind die Formulierungen als ILP bereits bekannt, da dies ein viel erforschtes Themengebiet ist. Im Folgenden wird daher eine Möglichkeit dargestellt, wie diese wiederum in QUBOs umgewandelt werden können. Bei einem allgemeinen ILP

$$\min \{ a^\top z : z \in \mathbb{Z}^n, Bz \leq b, Cz = c \} \tag{VIII}$$

mit $a \in \mathbb{Z}^n$, $B \in \mathbb{Z}^{m \times n}$, $C \in \mathbb{Z}^{r \times n}$, $b \in \mathbb{Z}^m$ und $c \in \mathbb{Z}^r$ müssen dafür zunächst die Variablen transformiert werden. Dies erfolgt wie im vorherigen Abschnitt zu den reduzierten Schlupfvariablen mittels einer Binärdarstellung.

Um das entsprechende QUBO über einer festen Zahl an binären Variablen formulieren zu können, müssen daher alle im ILP auftretenden ganzzahligen Variablen z_k für $k = 1, \dots, n$ begrenzt sein. Negative Werte werden durch einen Summanden ins Positive

2.3 Reduktionen

gezogen. Dementsprechend benötigt man für alle z_k eine untere Schranke $\check{z}_k \in \mathbb{Z}$ und eine obere Schranke $\hat{z}_k \in \mathbb{Z}$ mit $-\infty < \check{z}_k \leq \hat{z}_k < \infty$. Wegen

$$0 \leq z_k - \check{z}_k \leq \hat{z}_k - \check{z}_k =: d_k,$$

lässt sich z_k mit $D_k + 1 = \lfloor \log_2(d_k) \rfloor + 1$ binären Variablen $x_{k,j} \in \{0, 1\}$ für alle $k = 1, \dots, n$ ersetzen durch

$$\sum_{j=0}^{D_k} 2^j x_{k,j} + \check{z}_k,$$

und es muss gelten

$$\sum_{j=0}^{D_k} 2^j x_{k,j} \leq d_k.$$

Gilt $d_k = 2^{D_k+1} - 1$, ist diese Bedingung automatisch erfüllt, da die Grenzen 0 und d_k direkt durch die Binärdarstellung vorgegeben werden. Andernfalls muss wie bei den Schlupfvariablen im vorherigen Abschnitt eine Korrektur der Differenz stattfinden. Dabei wird z_k ersetzt durch

$$\sum_{j=0}^{D_k-1} 2^j x_{k,j} + (d_k - 2^{D_k} + 1) x_{k,D_k} + \check{z}_k.$$

Für $D = \max\{\lfloor \log_2(\hat{z}_k - \check{z}_k) \rfloor : k \in \{1, \dots, n\}\}$ lassen sich die binären Variablen in einer Matrix $X = (x_{j,k}) \in \{0, 1\}^{n \times (D+1)}$ zusammenfassen. Diese Matrix enthält zwar überflüssige Variablen, die durch Nullsetzung eliminiert werden könnten, jedoch ermöglicht sie eine kompaktere Schreibweise. Eine hohe Bestrafung der Aktivierung dieser Variablen könnte in eine Funktion $p : \{0, 1\}^{n \times (D+1)} \rightarrow \mathbb{N}^n$ entsprechend eingebaut werden mit $p(X) = p_1(X_{1,*}) + \dots + p_n(X_{n,*})$. Mit $\check{z} = (\check{z}_1, \dots, \check{z}_n)^\top$ und dem $(D+1)$ -dimensionalen Vektor der 2er-Potenzen

$$\mathfrak{2}_D = (2^0, 2^1, \dots, 2^D)^\top$$

gilt nun

$$z = X\mathfrak{2}_D + \check{z}.$$

Hierbei werden die obigen Korrekturen außer Acht gelassen, diese müssten gegebenenfalls über eine Addition des Vektors $((d_k - 2^{D_k+1} + 1) x_{k,D_k})_{k=1, \dots, n}$ erfolgen.

Damit kann die Zielfunktion ersetzt werden durch

$$a^\top z = a^\top (X\mathfrak{2}_D + \check{z}) = a^\top X\mathfrak{2}_D + a^\top \check{z} =: a^\top X\mathfrak{2}_D + \tilde{a}$$

mit $\tilde{a} \in \mathbb{Z}$. Die Gleichungen und Ungleichungen lassen sich mit $\tilde{b} \in \mathbb{Z}^m$ und $\tilde{c} \in \mathbb{Z}^r$ ebenso umformulieren zu

$$\begin{aligned} BX\mathfrak{2}_D &\leq b - B\check{z} =: \tilde{b}, \\ CX\mathfrak{2}_D &= c - C\check{z} =: \tilde{c}. \end{aligned}$$

Damit kann das Problem zunächst zu einem quadratischen binären umgeformt werden:

$$\tilde{a} + \min \left\{ a^\top X\mathfrak{2}_D + Pp(X) : X \in \{0, 1\}^{n \times (D+1)}, BX\mathfrak{2}_D \leq \tilde{b}, CX\mathfrak{2}_D = \tilde{c} \right\}.$$

Eine Gleichheit mit (VIII) ist jedoch nur erfüllt, wenn alle ganzzahligen Variablen tatsächlich begrenzt sind. Ansonsten stellt dieses Problem nur eine obere Schranke an den Optimalwert des ILPs dar. Diese Grenzen bestimmen auch stark die Zahl der bei der Umwandlung entstehenden Variablen:

$$\sum_{k=1}^n (D_k + 1) = n + \sum_{k=1}^n \lceil \log_2(\hat{z}_k - \check{z}_k) \rceil \leq n + Dn.$$

Um das Problem zu einem QUBO umzuformen, müssen die Gleichungen und Ungleichungen wie im vorherigen Abschnitt in die Zielfunktion integriert werden. Für erstere ergibt sich direkt

$$p_=(X) = \|CX\mathcal{Z}_D - \tilde{c}\|^2.$$

Um die Ungleichungen ebenso auflösen zu können, müssen hier zunächst auch Untergrenzen bestimmt werden. Als Optimalwerte $b^{\min} \in \mathbb{Z}^m$ der linearen 0/1-Probleme ohne Nebenbedingungen

$$\min \{(BX\mathcal{Z}_D)_l : X \in \{0, 1\}^{n \times (D+1)}\}$$

für $l = 1, \dots, m$ sind diese Untergrenzen leicht zu finden, indem nach dem Ausmultiplizieren von $(BX\mathcal{Z}_D)_l$ nur die Variablen auf 1 gesetzt werden, die einen negativen Koeffizienten haben. Die Zahl der benötigten Schlupfvariablen ergibt damit

$$m + \sum_{l=1}^m \lceil \log_2(\tilde{b}_l - b_l^{\min}) \rceil \leq m + Em$$

mit $E = \max\{\lceil \log_2(\tilde{b}_l - b_l^{\min}) \rceil : l \in \{1, \dots, m\}\}$. Werden diese analog zu X in einer Matrix $Y \in \{0, 1\}^{m \times (E+1)}$ zusammengefasst, kann die Straffunktion zu dem System von Ungleichungen folgendermaßen allgemein ohne Korrektur formuliert werden:

$$p_{\leq}(X, Y) = \|BX\mathcal{Z}_D - b^{\min} - Y\mathcal{Z}_E\|^2.$$

Damit lässt sich letztendlich das QUBO aufstellen. Die benötigten Strafgewichte ergeben sich dabei nach dem vorherigen Abschnitt aus den ausmultiplizierten Koeffizienten.

2.3.4 Pseudoboolesche Funktionen mit beliebigem Grad

Nicht nur lineare Nebenbedingungen wie in den vorherigen Abschnitten lassen sich in die Zielfunktion integrieren. Ebenso lassen sich analog Funktionen höheren Grades behandeln. Dabei entstehen Zielfunktionen, die nicht mehr quadratisch sind. E. Boros und P. Hammer zeigen in [17], dass sich eine allgemeine pseudoboolesche Funktion $b : \{0, 1\}^n \rightarrow \mathbb{R}$ mit einem beliebig hohem Grad, wie (II) aus 2.1, immer auf eine quadratische reduzieren lässt. Dies wird auch in [32] verwendet. Dabei werden schrittweise Produkte von Variablenpaaren durch neue binäre Variablen ersetzt. Dazu nutzt man den folgenden aussagenlogischen Zusammenhang für die booleschen Variablen $x_1, x_2, z \in \{0, 1\}$:

$$\begin{aligned} [x_1 \wedge x_2 \Leftrightarrow z] &= [[x_1 \wedge x_2 \Rightarrow z] \wedge [z \Rightarrow x_1 \wedge x_2]] \\ &= [\neg(x_1 \wedge x_2) \vee z] \wedge [\neg z \vee (x_1 \wedge x_2)] \end{aligned} \tag{IX}$$

$$= \neg [(x_1 \wedge x_2) \wedge \neg z] \wedge \neg [z \wedge \neg(x_1 \wedge x_2)],$$

wobei durch Einsetzen der äquivalenten mathematischen Formeln weiterhin gilt

$$\begin{aligned} &= (1 - x_1x_2(1 - z))(1 - z(1 - x_1x_2)) \\ &= 1 - z - x_1x_2 + 3x_1x_2z - x_1^2x_2^2z - x_1x_2z^2 + x_1^2x_2^2z^2 \quad (\text{X}) \\ &= 1 - z - x_1x_2 + 2x_1x_2z. \end{aligned}$$

Ergibt diese Funktion eins, ersetzt z genau das Produkt x_1x_2 . Bei der Minimierung der Zielfunktion müssen also die Fälle bestraft werden, in denen dies nicht gilt. Dazu könnte der Zielfunktion der negierte Strafterm

$$\neg [x_1x_2 \Leftrightarrow z] = z + x_1x_2 - 2x_1x_2z =: p_1(x_1, x_2, z)$$

mit einer entsprechenden Gewichtung hinzugefügt werden. Darin ist jedoch ein kubisches Monom enthalten. Dieses lässt sich durch Addition eines weiteren Terms beseitigen, da die Höhe der Bestrafung der einzelnen nichtoptimalen Möglichkeiten irrelevant ist. Die Addition des Terms

$$(\neg x_1)(\neg x_2)z = x_1x_2z - x_1z - x_2z + z =: p_2(x_1, x_2, z)$$

verändert den Wert der Straffunktion nicht, wenn dieser Null war. Dies zeigt sich in der Wahrheitstabelle:

x_1	x_2	z	$\neg [x_1x_2 \Leftrightarrow z]$	$(\neg x_1)(\neg x_2)z$
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	0
1	1	0	1	0
1	1	1	0	0

So ergibt sich die nur quadratische Straffunktion

$$\begin{aligned} p(x_1, x_2, z) &= p_1(x_1, x_2, z) + 2p_2(x_1, x_2, z) \\ &= 3z + x_1x_2 - 2zx_1 - 2zx_2. \end{aligned}$$

Sei $J = \{I \subseteq \{1, \dots, n\} : a_I \neq 0\} \subseteq 2^{\{1, \dots, n\}}$ die Menge der Monomindizes der booleschen Funktion b und $J_{1,2} = \{I \in J : 1, 2 \in I\}$, die Teilmenge, welche die Monome mit x_1x_2 umfasst. Die Differenz der pseudobooleschen Funktion b , die bei einer falschen Belegung von z , also bei $z \neq x_1x_2$, auftritt, ist so nach Teilabschnitt Wahl des Strafgewichtes aus 2.3.2

$$\Delta b := \max \{ -\hat{\sigma}(a_{J_{1,2}}), \hat{\sigma}(a_{J_{1,2}}) \}.$$

Mit einem Strafgewicht $P > \Delta b$ gilt dann

$$b(x) = \min_{z \in \{0,1\}} (b_{z=x_1x_2}(x, z) + Pp(x_1, x_2, z))$$

und demzufolge

$$\min_{x \in \{0,1\}^n} b(x) = \min_{\substack{x \in \{0,1\}^n \\ z \in \{0,1\}}} (b_{z=x_1x_2}(x, z) + Pp(x_1, x_2, z)).$$

Dies kann rekursiv fortgeführt werden, wobei in jedem Schritt eine Reduzierung des Grades um genau 1 erreicht werden kann. Dabei werden für ein multilineares Monom vom Grad $l \in \mathbb{N}$ genau $l - 2$ zusätzliche Variablen benötigt. Wird nach der Reihenfolge der Variablen eliminiert, ergibt sich

$$\prod_{j=1}^l x_j = \min_{z \in \{0,1\}^{l-2}} \left(z_{l-2}x_l + P \sum_{k=1}^{l-2} p(x_k, x_{k+1}, z_k) \right),$$

wobei $P > 1$ wegen $a_{\{1,\dots,l\}} = 1$ genügt. Es ist jedoch ebenso möglich die Reihenfolge zu vertauschen sowie Produkte mit neu eingeführten Variablen wieder zu substituieren. Demnach ergibt sich bei der Reduzierung einer pseudobooleschen Funktion auf quadratischen Grad eine maximale Zahl an zusätzlich benötigten Variablen von

$$\sum_{I \in \mathcal{J}, |I| > 2} (|I| - 2),$$

wenn nacheinander jedes Paar ersetzt wird. Diese Zahl lässt sich durch geschicktes Wählen mehrfach vorkommender Paare reduzieren. Dieses Problem entspricht der Suche nach dem größtem Minor eines Hypergraphen, welcher durch die Verschmelzung von Knoten innerhalb einer Hyperkante nur noch klassische Kanten, also zwischen zwei Knoten, besitzt. Beschränkt man sich dabei nur auf Hyperkanten der Kardinalität 3, ergibt sich eine Funktion mit

$$\sum_{\substack{j,k,l=1 \\ j < k < l}}^n a_{\{j,k,l\}} x_j x_k x_l = \sum_{\substack{j,k,l=1 \\ j < k < l}}^{n-1} a_{\{j,k,l\}} x_j x_k x_l + x_n \sum_{\substack{j,k=1 \\ j < k}}^{n-1} a_{\{j,k,n\}} x_j x_k.$$

Der herausgezogene n -te Summand entspricht einem Hypergraphen, dessen Hyperkanten alle den Knoten n enthalten. In [17] wird gezeigt, dass die Bestimmung der kleinsten Zahl an Reduktionen selbst für einen solchen Teilhypergraphen ein NP-schweres Problem ist, da dies auf die Suche nach der Knotenüberdeckungszahl des durch Wegnahme des n -ten Knotens entstehenden Graphen zurückgeführt werden kann. Damit ist also die Suche nach der geringsten Zahl an zusätzlichen Variablen für eine beliebige pseudoboolesche Funktion ebenfalls NP-schwer. Wie und ob dieses Problem dabei selbst als QUBO dargestellt werden kann, ist unklar.

Ähnliche Ersetzungen anderer aussagenlogischer Kombinationen ϕ von booleschen Variablen x lassen sich ebenso wie in (IX) und (X) aus $[z \Leftrightarrow \phi]$ herleiten. Die folgende Tabelle zeigt drei Beispiele für entsprechende Straffunktionen aus [32], wobei für das ausschließende Oder in der dritten Zeile eine weitere Variable $y \in \{0, 1\}$ benötigt wird:

ϕ	p_ϕ
$\neg x$	$2xz - x - z + 1$
$x_1 \vee x_2$	$x_1x_2 + (x_1 + x_2)(1 - 2z) + z$
$x_1 \oplus x_2$	$2x_1x_2 - 2(x_1 + x_2)z - 4(x_1 + x_2)y + 4yz + x_1 + x_2 + z + 4y$

Durch analoge Umformulierungen und anschließende Reduktionen einer beliebigen aussagenlogischen Formel ϕ allein kann auch zunächst geprüft werden, ob diese überhaupt erfüllbar ist. Dabei ist zu beachten, dass bei einer Minimierung die mathematische Formel der negierten Aussage zu betrachten ist. Ist deren Minimum gleich Null, so liefert eine entsprechende Optimallösung eine wahre Belegung der Atome. Ist es größer als Null, liegt eine Kontradiktion vor. Folglich ist eine SAT-Instanz ebenso als eine Entscheidungsvariante eines QUBOs darstellbar.

2.3.5 Allgemeine Graphenprobleme

Vielen Problemen, wie dem Travelling Salesman, der Graphenfärbung oder der Partitionierung, liegt bereits eine Graphenstruktur zugrunde. Es werden spezielle Knoten- oder Kantenzuordnungen gesucht, welche bestimmte Bedingungen erfüllen müssen. Im Folgenden wird zunächst für ungerichtete Graphen gezeigt, wie sich einige Problemstellungen auf QUBOs übertragen lassen. Für gerichtete Graphen kann anschließend gezeigt werden, wie diese auf ungerichtete übertragbar sind. Damit ist auch für diese Fälle eine Darstellungsmöglichkeit als QUBO gewährleistet.

Kodierung ungerichteter Graphen

Bei vielen der Graphenprobleme ist die Frage, ob ein Knoten zur gesuchten Teilmenge gehört oder nicht. Dies lässt sich mit einer 0/1-Belegung auf den Knoten mit $x \in \{0, 1\}^V$ gut kodieren. Ebenso kann hierdurch eine Unterteilung in zwei Gruppen erfolgen, was zum Beispiel bei Schnittproblemen, wie in 2.3.1, gefordert ist. Sollen die Elemente der Menge, die durch die Belegung 0 kodiert ist, angesprochen werden, wie beispielsweise in Nebenbedingungen, so kann dies analog zu den aktiven Knoten geschehen. Dabei wird nur in den Termen x_j mit $1 - x_j$ ersetzt. Ist eine Unterscheidung in mehr als zwei Gruppen gesucht, wie in die zu einer Farbe gehörenden Teilmengen bei Färbungsproblemen, wird zunächst eine Kodierung der verschiedenen Farben vorgenommen. Dies kann über eine Zuordnung einer natürlichen Zahl geschehen. Die bei deren Umsetzung benötigten binären Variablen werden Knoten zugeordnet, die stark miteinander gekoppelt werden. Sie bilden zusammen einen übergeordneten imaginären Knoten, der mit mehr als 0 oder 1 belegt werden kann. Über diesen imaginären Knoten kann das Optimierungsproblem aufgebaut werden. Dies kann ähnlich geschehen wie im Abschnitt 2.3.3. Ein Beispiel zum Graphenfärbungsproblem ist in [33] gezeigt.

Werden hingegen Kantenteilmengen gesucht und auch an diese Bedingungen gestellt, ergibt eine Knotenkodierung oft eine Zielfunktion mit einem mehr als quadratischen Grad. Hier ist eine Kante $\{u, v\} \in E$ genau dann aktiv, wenn $x_u x_v = 1$ ist. Daher kann es sinnvoll sein, direkt den Kantengraphen

$$\mathcal{L}(G) = \left(E, \{ \{e, f\} : e, f \in E, e \cap f \neq \emptyset \} \right)$$

zu betrachten. Dieser wird folgendermaßen konstruiert: Jeder Kante $\{u, v\} \in E$ wird ein Knoten zugeordnet. Mit der Variablenbelegung $y \in \{0, 1\}^E$, ist diese Kante nun aktiviert, wenn $y_{\{u,v\}} = 1$ ist. Zwei dieser Knoten sind genau dann im Kantengraphen verbunden, wenn sie einen gemeinsamen Knoten besitzen. In Abbildung 6 ist dazu ein Beispiel gezeigt. Da meist mehr Kanten als Knoten vorhanden sind, ist der Kanten-

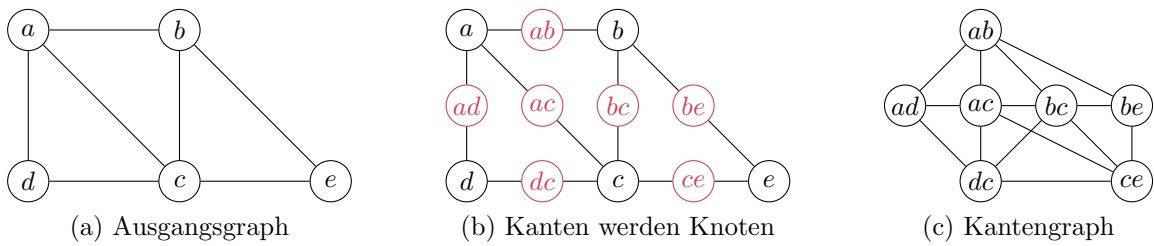


Abb. 6: Konstruktion des Kantengraphen

graph zwar größer, aber oft birgt er einen Vorteil gegenüber der Standardkodierung auf den Knoten, da keine Reduktionsvariablen benötigt werden. Sind Bedingungen für Kanten und Knoten gegeben, können beide Möglichkeiten wie bei der Reduktion pseudo-booleaner Funktionen durch das Erzwingen von $[y_{\{u,v\}} \Leftrightarrow x_u x_v]$ kombiniert werden.

Gerichtete Graphen

Einige Aufgabenstellungen, wie zu Pfaden oder Flüssen, ergeben sich erst auf gerichteten Graphen. In einem gerichteten Graphen oder auch Digraphen $D = (V, A)$ ist die Kantenmenge A eine Teilmenge der geordneten Knotenpaare aus $V \times V$. Diese sogenannten Bögen können nur in einer Richtung durchlaufen werden und es gilt $(v, w) \neq (w, v)$ für alle $(v, w) \in A$. Damit besitzt jeder Knoten eingehende und ausgehende Bögen. Dazwischen kann jedoch beim Interaktionsgraphen eines QUBOs nicht unterschieden werden. Daher wird im Folgenden eine allgemeine Möglichkeit aus [34] gezeigt, das Problem zu übertragen. Deren Anwendbarkeit muss jedoch im speziellen Fall immer geprüft werden.

Um den Digraphen D mittels eines ungerichteten Graphen G zu simulieren, muss dieser die doppelte Zahl an Knoten enthalten: Für alle $v \in V$ werden ein v_{in} und ein v_{out} benötigt. Diese Knotenpaare werden stark durch Kanten $\{v_{in}, v_{out}\}$ gekoppelt, damit sie in einer Lösung stets gemeinsam aktiviert sind oder nicht. Sind Gewichte gegeben, werden diese auf beide Knoten analog zur Vorgehensweise in 3.2.3 verteilt. Ein Bogen des Digraphen $(v, w) \in A$ wird übertragen, indem für seinen Anfangsknoten v der Knoten v_{out} mit dem Knoten w_{in} des Endknotens verbunden wird. Das Prinzip ist in Abbildung 7 verdeutlicht. Die eventuelle Stärke des Bogens wird als Stärke der Kante übernommen. Zusammengefasst ergibt sich der Graph

$$G = (V_{in} \cup V_{out}, \{\{v_{in}, v_{out}\} : v \in V\} \cup \{\{v_{out}, w_{in}\} : (v, w) \in A\})$$

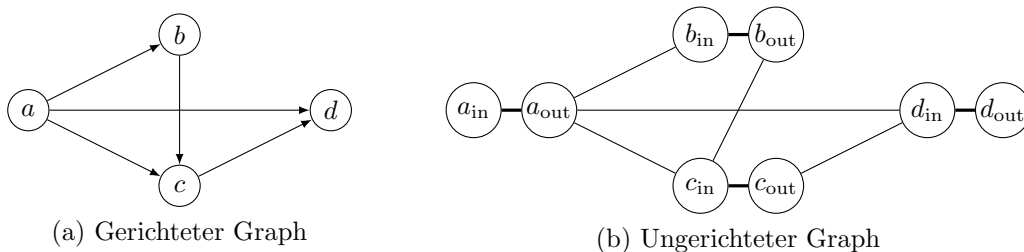


Abb. 7: Umwandlung zu ungerichteten Graphen

mit $2|V|$ Knoten und $|E|+|V|$ Kanten, über denen die Anforderungen des Problems wie in den vorherigen Abschnitten formuliert und zu einem QUBO umgewandelt werden können.

2.3.6 Weitere Beispielprobleme

In diesem Abschnitt werden beispielhaft ein paar Probleme hervorgehoben, welche in den folgenden Kapiteln eine Rolle spielen werden. Dabei wird bei den Graphenproblemen besonders auf die Komplexität des Interaktionsgraphen im Unterschied zum Ausgangsgraphen eingegangen. Es zeigt sich, dass im QUBO-Problem oft nicht nur mehr Kanten unter den bestehenden Knoten sondern auch zusätzliche Knoten mit den entsprechenden Kanten benötigt werden.

Matching

Das Matching-Problem ist eine typische Aufgabenstellung der Graphentheorie. In einer Menge von Objekten sollen Paarungen gefunden werden. Dies lässt sich in einem ungerichteten Graphen $G = (V, E)$ veranschaulichen, in dem die Objekte mit den Knoten identifiziert sind und die Kanten zeigen, welche Objekte überhaupt einander zugeordnet werden können. Ein Matching des Graphen ist definiert als eine Kantenmenge $M \subseteq E$, in der sich keine zwei Kanten einen Knoten teilen, da ein Objekt nur genau einmal gepaart werden kann. Kann einem Matching M keine Kante mehr hinzugefügt werden, so dass diese Eigenschaft eingehalten wird, heißt es nicht erweiterbar oder maximal. Ein solches Matching lässt sich sehr einfach über einen Greedy-Algorithmus finden. Ziel ist es nun aber das Matching mit der größten Kardinalität zu finden. Diese Größe entspricht dann der Matching-Zahl $\nu(G)$ des Graphen. Dieses mathematisch einfache Problem soll im ersten Schritt betrachtet werden. Anschließend folgt ein Vergleich zu der NP-schweren Variante des Problems: Die Suche nach einem maximalen Matching mit der kleinstmöglichen Kardinalität (kurz Min-Max-Matching) [35]. Beide Varianten sind beispielhaft in Abbildung 8 gezeigt.

Bei der Aufstellung des formalen Optimierungsproblems zur Suche nach der Matching-Zahl bietet sich eine Kodierung über den Kanten an, über den Variablen $x \in \{0, 1\}^E$ mit

$$x_e = \begin{cases} 1, & \text{wenn } e \text{ zum Matching gehört,} \\ 0, & \text{sonst.} \end{cases}$$

Die größtmögliche Zahl an Kanten lässt sich dann mit dem negativen Minimum der

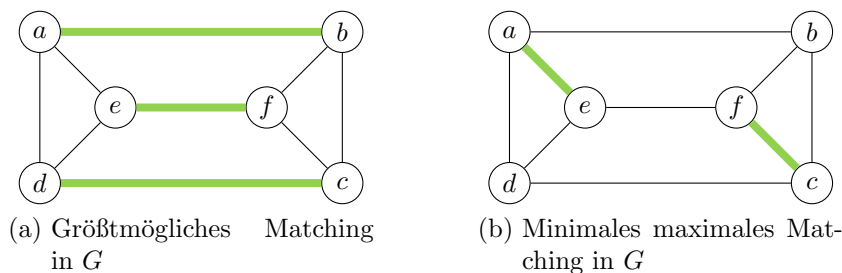


Abb. 8: Beispielgraphen zu Matching-Problemen

folgenden Zielfunktion bestimmen:

$$q(x) = - \sum_{e \in E} x_e.$$

Die Bedingung über die Zahl der aktiven ausgehenden Kanten eines Knoten ergibt dann formal die Ungleichungen

$$\sum_{e \in \delta(v)} x_e \leq 1 \tag{XI}$$

für alle $v \in V$. Diese lassen sich wie in 2.3.2 im Teilabschnitt Bedeutung der Kopplungen analog als quadratische Straffunktion formulieren und für alle Knoten aufsummieren:

$$p_{\leq}(x) = \sum_{v \in V} \sum_{\substack{e, f \in \delta(v) \\ e \neq f}} x_e x_f = 2 \sum_{\substack{\{v, w\}, \{v, u\} \in E \\ w \neq u}} x_{\{v, w\}} x_{\{v, u\}}.$$

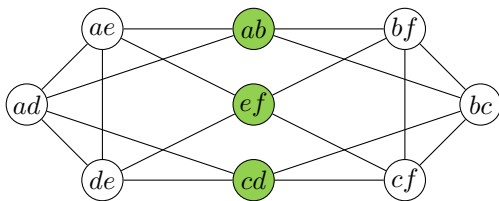
Wird die Zielfunktion entsprechend zusammengesetzt, zeigt sich, dass dies der Suche einer stabilen Menge im Kantengraph $\mathcal{L}(G)$ entspricht. Daher kann eine Gewichtung der Straffunktion nach 2.3.1 mit $P = 1$ gewählt werden.

Folgerung 2.4

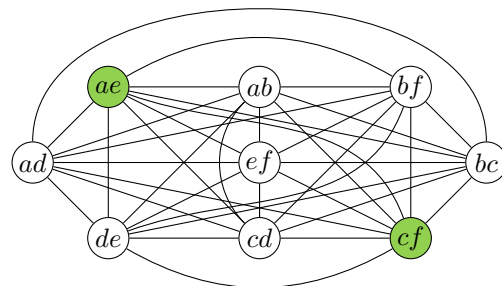
$$\begin{aligned} \nu(G) &= - \min \left\{ - \sum_{e \in E} x_e + \sum_{v \in V} \sum_{\substack{e, f \in \delta(v) \\ e \neq f}} x_e x_f : x \in \{0, 1\}^E \right\} \\ &= - \min \left\{ - \sum_{e \in E} x_e + 2 \sum_{\{e, f\} \in E(\mathcal{L}(G))} x_e x_f : x \in \{0, 1\}^E \right\} = \alpha(\mathcal{L}(G)). \end{aligned}$$

In Abbildung 9(a) ist der entsprechende Kantengraph zum Beispiel aus Abbildung 8(a) gezeigt.

Ist der Graph mit einer Bewertungsfunktion $S : E \rightarrow \mathbb{R}$ versehen, kann analog dazu auch nach einem Matching mit größtmöglichem Gewicht gesucht werden. Die Kardinalität des Matchings kann hierbei durchaus kleiner sein als $\nu(G)$. Es wird dann erhalten, wenn x_e in der Funktion q durch $S_e x_e$ für alle $e \in E$ ersetzt wird. Demzufolge muss jedoch die Gewichtung der Straffunktion neu gewählt werden.



(a) Stabile Menge im Kantengraph von G



(b) Interaktionsgraph zu Min-Max-Matching von G

Abb. 9: Interaktionsgraphen zu Matching-Problemen

Folgerung 2.5 Mit $P > \max\{0, S_e : e \in E\}$ gilt

$$\begin{aligned} \text{MaxMatch}(G, S) &= \max \{S(M) : M \subseteq E \text{ ist Matching}\} \\ &= - \min \left\{ - \sum_{e \in E} S_e x_e + P \sum_{\{e, f\} \in E(\mathcal{L}(G))} x_e x_f : x \in \{0, 1\}^E \right\}. \end{aligned}$$

Mit der konstanten Bewertungsfunktion $S_e = 1$ für alle $e \in E$ gilt so

$$\nu(G) = \text{MaxMatch}(G, S).$$

Im Min-Max-Problem, dessen QUBO-Formulierung in [29] vorgestellt wurde, wird nach einer kleinstmöglichen Kantenteilmenge gesucht. Daher gilt für die Zielfunktion

$$\tilde{q}(x) = \sum_{e \in E} x_e = -q(x).$$

Die Bedingungen (XI) und die daraus resultierende Straffunktion p_{\leq} müssen zur Einhaltung der Matching-Eigenschaften übernommen werden. Um nun aber zu verhindern, dass die Nullmenge als Lösung in Frage kommt, muss eine Bedingung für die Maximalität des Matchings ergänzt werden. Das heißt, gibt es eine Kante, deren Knoten jeweils noch nicht gepaart wurden, also jeweils keine aktive ausgehende Kante besitzen, muss diese Kante dem Matching hinzugefügt werden. Dies ließe sich wie in [29] über die Ergänzung von binären Variablen y_v über den Knoten $v \in V$ realisieren. Diese zusätzlichen Variablen zeigen an, ob ein Knoten zu einer Kante des Matchings gehört oder nicht, indem die Beziehungen $y_u y_v = x_{\{u, v\}}$ für alle $\{u, v\} \in E$ erzwungen werden.

Hier soll eine weitere Möglichkeit aufgeführt werden, bei der es nicht nötig ist, neue Variablen einzuführen: Betrachtet man eine Kante $\{u, v\}$ und ist diese aktiv, so können wegen (XI) weder die weiteren ausgehenden Kanten von u noch die von v aktiv sein. Ist hingegen $x_{\{u, v\}} = 0$, so muss nach den neuen Voraussetzungen mindestens eine dieser Kanten aktiv sein, wobei es sein kann, dass aus beiden Mengen je eine aktiv ist. Formal ergibt dies die Bedingung

$$1 \leq \sum_{\substack{e \in \delta(u) \\ e \neq \{u, v\}}} x_e + \sum_{\substack{e \in \delta(v) \\ e \neq \{u, v\}}} x_e + x_{\{u, v\}} \leq 2$$

für alle Kanten $\{u, v\} \in E$. Analog zu (VI) sind 1 und 2 die Nullstellen und die ganzzahligen Minimalstellen des Polynoms

$$\left(y - \frac{3}{2}\right)^2 - \frac{1}{4},$$

womit sich die Bedingung durch

$$p_{\geq}(x) = \sum_{\{u, v\} \in E} \left(\sum_{\substack{e \in \delta(u) \\ e \neq \{u, v\}}} x_e + \sum_{\substack{e \in \delta(v) \\ e \neq \{u, v\}}} x_e + x_{\{u, v\}} - \frac{3}{2} \right)^2 - \frac{1}{4} |E| \quad (\text{XII})$$

umsetzen lässt.

Damit kann nun das Min-Max-Matching-Problem als QUBO zusammensetzen werden. Wie der Interaktionsgraph durch die zusätzliche Bedingung verändert wurde, ist in Abbildung 9(b) gezeigt. Eine Verletzung dieser Bedingung dahingehend, dass zu viele Kanten aktiviert werden, wird durch p_{\leq} verhindert. Sind für eine inaktive Kante auch alle anschließenden Kanten deaktiviert, ist die Zielfunktion zwar um 1 verringert, jedoch ergibt die Straffunktion für diese Kante einen Wert von 2. Daher kann auch hierfür ein Strafgewicht von 1 gewählt werden.

Folgerung 2.6

$$\text{MinMaxMatch}(G) = \min \left\{ \sum_{e \in E} x_e + p_{\leq}(x) + p_{\geq}(x) : x \in \{0, 1\}^E \right\}.$$

Mit der Gewichtung 1 wird nach der Kardinalität eines solchen Min-Max-Matchings gesucht. Wie zum vorherigen Matching-Problem kann auch hierbei nach einem maximalen Matching mit dem kleinsten Gewicht gesucht werden, wenn Stärken für die Kanten gegeben sind. Analog zu Folgerung 2.5 lässt sich formulieren:

Folgerung 2.7 Mit $P > \max\{0, -S_e : e \in E\}$ gilt

$$\begin{aligned} \text{MinMaxMatch}(G, S) &= \min \{ S(M) : M \subseteq E \text{ ist maximales Matching} \} \\ &= \min \left\{ \sum_{e \in E} S_e x_e + P p_{\leq}(x) + P p_{\geq}(x) : x \in \{0, 1\}^E \right\}. \end{aligned}$$

Spannbaum

Ein Spannbaum in einem Graphen $G = (V, E)$ beschreibt eine Teilmenge der Kanten $T \subseteq E$, über welche jeder Knoten mit jedem anderen verbunden ist. Sie existieren demnach nur in zusammenhängenden Graphen. Da dabei jedoch keine geschlossenen Pfade, also Kreise, erlaubt sind, ergibt die Zahl der Kanten in T genau $|V| - 1$. Auch für jede Teilmenge der Knoten $W \subset V$ mit $W \neq \emptyset$ folgt daraus für die Zahl der Kanten zwischen diesen Knoten im Spannbaum, dass $|T(W)| \leq |W| - 1$. Mit dem Algorithmus von Prim ist die Suche nach einem Spannbaum mit den größten Gewicht in polynomieller Zeit lösbar. Eine entsprechende QUBO-Formulierung ist über eine 0/1-Kodierung auf den Kanten realisierbar, jedoch heißt das für $|W| = j$ werden für jede der $\binom{n}{j}$ Ungleichungen $\lfloor \log_2(j - 1) \rfloor$ Schlupfvariablen benötigt. Zwar hängen diese Variablen aufgrund der Zusammenhänge unter den Teilmengen ebenso voneinander ab, inwiefern die Zahl jedoch reduziert werden könnte, ist in keiner Weise ersichtlich.

Daher wird hier im Folgenden eine andere Herangehensweise an das Problem betrachtet, wie sie von J. Williams in [36] vorgestellt wurde. Sie beruht auf der erweiterten Formulierung des Spannbaum-Polytops für planare Graphen, welche die Ungleichungen über zum Teil sehr vielen Variablen auf Gleichungen über nur wenigen Variablen reduziert, wodurch lokalere Kopplungen entstehen. Sie wird wie folgt konstruiert: Die Einbettung eines planaren Graphen in eine Ebene ergibt von Kanten umschlossene Gebiete, sogenannte Länder. Diese bilden in dem sogenannten Dualgraphen $G^* = (F, E^*)$ von G die Knoten F . Dabei ist auch das äußere Umfeld des Graphen ein Land, welches mit r^* bezeichnet wird. Zwei Knoten in diesem Graphen werden nun über Kan-

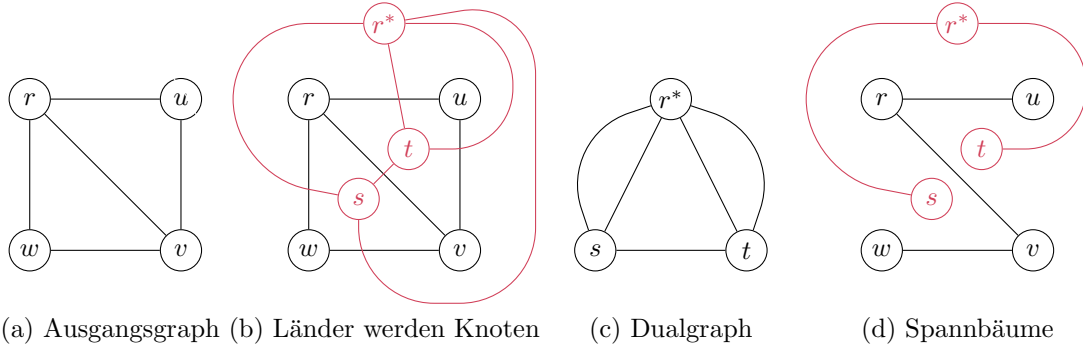


Abb. 10: Konstruktion des Dualgraphen

ten verbunden, wenn die entsprechenden Länder durch eine gemeinsame Kante e des Ursprungsgraphen voneinander getrennt werden. Dies ist die zu e duale Kante e^* . Veranschaulicht wird dies in Abbildung 10. Es gilt, dass ein Spannbaum im Dualgraphen einen Spannbaum im Ausgangsgraphen durch Entfernen der überkreuzten Kanten definiert und umgekehrt. Dies ist in Abbildung 10(d) gezeigt.

Damit lassen sich nun andere Bedingungen formulieren. Für jede Kante des Ausgangsgraphen und des Dualgraphen wird zwischen hinein- und herausgehend unterschieden und daher jeweils zwei binäre Variablen mit $x^{\text{in}}, x^{\text{out}} \in \{0, 1\}^E$ beziehungsweise $z^{\text{in}}, z^{\text{out}} \in \{0, 1\}^{E^*}$ definiert. Es ergeben sich die folgenden Anforderungen, damit x^{out} eine Belegung der Kanten darstellt, die einem Spannbaum entspricht:

$$\begin{aligned}
 x_{\{v,w\}}^{\text{in}} + x_{\{v,w\}}^{\text{out}} + z_{\{s,t\}}^{\text{in}} + z_{\{s,t\}}^{\text{out}} &= 1 & \forall \begin{array}{c} (v) \quad (s) \\ \diagdown \quad / \\ (t) \quad (w) \end{array} & \quad (\text{d.h. } \{v, w\}^* = \{s, t\}), \\
 \sum_{\{v,w\} \in E} x_{\{v,w\}}^{\text{in}} &= 1 & \forall v \in V \setminus \{r\}, \\
 \sum_{\{s,t\} \in E^*} z_{\{s,t\}}^{\text{in}} &= 1 & \forall s \in F \setminus \{r^*\}.
 \end{aligned}$$

Dabei ist r ein gewählter Knoten des Ausgangsgraphen, der im Rand des äußeren Landes liegt. Die erste Gleichung stellt einen Zusammenhang zwischen den Kanten von G und deren dualen aus G^* her. Die beiden folgenden Gleichungen beschränken den Eingangsgrad für einen Knoten auf 1, da jeder Knoten nur genau einen Vorgänger im Baum besitzen kann. Durch Aktivierung der Variablen entsprechend der Bedingungen entsteht in beiden Graphen jeweils ein Baum, dessen gerichtete Kanten von der Wurzel weg zeigen. Daher können auch $x_{\{r,v\}}^{\text{in}}$ für alle $v \in d(r)$ und $z_{\{r^*,s\}}^{\text{in}}$ für alle $s \in d(r^*)$ eliminiert werden, da in die Wurzeln keine Kanten hinein zeigen können. Der gesuchte Baum im Ausgangsgraphen wird durch die aktiven x^{out} beschrieben.

Diese Gleichungen lassen sich nun ohne weitere Schlupfvariablen in einer Straffunktion p erzwingen. Diese teilt sich auf in die Summanden

$$p = p_1 + p_2 + p_3 \tag{XIII}$$

mit

$$p_1 = \sum_{e \in E} (x_e^{\text{in}} + x_e^{\text{out}} + z_{e^*}^{\text{in}} + z_{e^*}^{\text{out}} - 1)^2,$$

$$p_2 = \sum_{v \in V \setminus \{r\}} \left(\sum_{e \in \delta(v)} x_e^{\text{in}} - 1 \right)^2,$$

$$p_2 = \sum_{s \in F \setminus \{r^*\}} \left(\sum_{e^* \in \delta(s)} x_{e^*}^{\text{in}} - 1 \right)^2.$$

Ist eine Bewertungsfunktion gegeben, kann damit ein Spannbaum mit dem größten Gewicht gesucht werden. Da ein falsche Belegung mindestens zu einer Strafe von 1 in p führt, muss das Strafgewicht den maximalen Gewinn, der in der Zielfunktion erhalten werden könnte, ausgleichen. Damit lässt sich letztendlich das QUBO mit insgesamt $4|E|$ binären Variablen aufstellen:

Folgerung 2.8 Mit $P > \max\{0, S_e : e \in E\}$ gilt

$$\text{MaxTree}(G, S) = - \min \left\{ - \sum_{e \in E} S_e x_e^{\text{out}} + Pp(x^{\text{in}}, x^{\text{out}}, z^{\text{in}}, z^{\text{out}}) : \right. \\ \left. x^{\text{in}}, x^{\text{out}} \in \{0, 1\}^E, z^{\text{in}}, z^{\text{out}} \in \{0, 1\}^{E^*} \right\}.$$

Im Gegensatz zur Standardformulierung, in der eine Vielzahl an Ungleichungen über allen Kanten aufgestellt werden, entstehen hier nur kleine lokale Cliques im Interaktionsgraphen. Dieser wird jedoch nun aus den Kantengraphen des Ausgangsgraphen und dessen Dualgraphen, mit verdoppelter Kantenzahl für die Unterscheidung zwischen ein- und ausgehenden Kanten, zusammengesetzt. Daran zeigt sich, dass die gewählte Formulierung großen Einfluss auf die letztendliche Formulierung und besonders die Größe des QUBOs hat.

In dieser Formulierung lässt sich das Problem sehr leicht erweitern. Durch Hinzufügen von Gradbeschränkungen an die eingehenden Kanten, wird es sogar ein NP-schweres Problem [35]. Mit der Bedingung

$$\sum_{e \in \delta(v)} x_e^{\text{out}} \leq c$$

für alle Knoten $v \in V$ entspricht diese Variante beispielsweise für $c = 1$ und einer konstanten Bewertungsfunktion der Suche nach einem Hamiltonpfad im Graphen. Die entsprechende Straffunktion wäre wieder

$$p_{\leq}(x^{\text{out}}) = \sum_{v \in V} \sum_{\substack{e, f \in \delta(v) \\ e \neq f}} x_e^{\text{out}} x_f^{\text{out}}. \quad (\text{XIV})$$

Folgerung 2.9 Ein Graph $G = (V, E)$ besitzt einen Hamiltonpfad, wenn gilt

$$|V| - 1 = \min \left\{ - \sum_{e \in E} x_e^{\text{out}} + 2p(x^{\text{in}}, x^{\text{out}}, z^{\text{in}}, z^{\text{out}}) + 2p_{\leq}(x^{\text{out}}) : \right. \\ \left. x^{\text{in}}, x^{\text{out}} \in \{0, 1\}^E, z^{\text{in}}, z^{\text{out}} \in \{0, 1\}^{E^*} \right\}.$$

Für $c = 2$ ist ein binärer Baum das Ziel der Optimierung. Diese Bedingung lässt sich ebenso, wie bereits gezeigt, in das QUBO integrieren, jedoch sind für $c \geq 2$ wieder weitere Schlupfvariablen nötig. Mit $C = \lfloor \log_2(c) \rfloor$ ergibt sich damit die Straffunktion

$$p_{\leq}(x^{\text{out}}, y) = \left(\sum_{\{v,w\} \in E} x_{\{v,w\}}^{\text{out}} - \sum_{k=0}^{C-1} 2^k y_k - (c - 2^C + 1) y_C \right)^2$$

über $C + 1$ neuen Variablen $y_k \in \{0, 1\}$. Damit lässt sich analog das QUBO zur Suche nach dem gradbeschränkten Baum mit dem größten Gewicht formulieren. Wird auch hier wieder eine konstante Bewertungsfunktion S gewählt, kann ebenso zunächst auf die Existenz eines gradbeschränkten Baumes getestet werden, denn in diesem Fall ergibt der Optimalwert $S(|V| - 1)$. Nur wenn ein solcher Baum existiert, liefert das folgende QUBO durch die Aktivierung von $|V| - 1$ Knoten einen gewichtsmaximalen Baum:

Folgerung 2.10 Mit $P > \max\{0, S_e : e \in E\}$ gilt

$$\text{MaxTree}(G, S, c) = - \min \left\{ - \sum_{e \in E} S_e x_e^{\text{out}} + Pp(x^{\text{in}}, x^{\text{out}}, z^{\text{in}}, z^{\text{out}}) + Pp_{\leq}(x^{\text{out}}, y) : \right. \\ \left. x^{\text{in}}, x^{\text{out}} \in \{0, 1\}^E, z^{\text{in}}, z^{\text{out}} \in \{0, 1\}^{E^*}, y \in \{0, 1\}^{\lfloor \log_2(c) \rfloor + 1} \right\}.$$

Bin-Packing

Bei diesem kombinatorischen Problem, welches auch Behälterproblem heißt, sind $n \in \mathbb{N}$ Objekte mit Gewichten $a_1, \dots, a_n \in \mathbb{N}$ gegeben. Die Objekte sollen nun auf Behälter verteilt werden. Diese haben eine Größe $d \in \mathbb{N}$, welche nicht überschritten werden darf. Damit ist klar, dass $a_1, \dots, a_n \leq d$ sein müssen. Nun ist die Frage, ob alle Objekte in nicht mehr als $r \in \mathbb{N}$ der Behälter untergebracht werden können. Dies bedeutet formal: Existiert eine Funktion $f : \{1, \dots, n\} \rightarrow \{1, \dots, r\}$, sodass für alle $k \in \{1, \dots, r\}$

$$\sum_{\substack{j=1 \\ f(j)=k}}^n a_j \leq d$$

gilt? Dieses Entscheidungsproblem ist NP-schwer [35]. Es ist besonders für die Verpackungsindustrie von großer Bedeutung, lässt sich aber auch in viele weitere Bereiche übertragen.

Eine Möglichkeit, wie dieses Problem nun als ein QUBO dargestellt werden kann, bietet sich über eine binäre Matrix X der Größe $n \times r$, für die gilt

$$X_{j,k} = \begin{cases} 1, & \text{wenn Objekt } j \text{ in Behälter } k \text{ ist,} \\ 0, & \text{sonst.} \end{cases}$$

Die Eigenschaften der Funktion f werden dabei in Bedingungen an diese Variablen verpackt. So kann jedes Objekt nur in genau einem Behälter liegen, weswegen für alle

$j \in \{1, \dots, n\}$

$$\sum_{k=1}^r X_{j,k} = 1$$

erfüllt sein muss. Die Summationsbedingungen können dann für $k = 1, \dots, r$ wie folgt formuliert werden:

$$\sum_{j=1}^n a_j X_{j,k} \leq d.$$

Die Zielfunktion des QUBOs ist damit entsprechend der in den vorherigen Abschnitten gezeigten Reduktionen mit $D = \lceil \log_2(d) \rceil$ und den Schlupfvariablen $Y \in \{0, 1\}^{r \times (D+1)}$

$$q(X, Y) = \sum_{j=1}^n \left(\sum_{k=1}^r X_{j,k} - 1 \right)^2 + \sum_{k=1}^r \left(\sum_{j=1}^n a_j X_{j,k} - \sum_{l=0}^{D-1} 2^l Y_{k,l} - (d - 2^D + 1) Y_{k,D} \right)^2$$

$$\stackrel{d=2^D-1}{=} \|X \mathbb{1}_r - \mathbb{1}_n\|^2 + \|X^\top a - Y \mathfrak{2}_D\|^2$$

mit $q : \{0, 1\}^{n \times r} \times \{0, 1\}^{r \times (D+1)} \rightarrow \mathbb{N}$. Hierbei sind keine Gewichtungen nötig, da jede Verletzung der Bedingungen zu einer Erhöhung der Zielfunktion führt. Daher existiert nur eine Lösung des Problems, also es genügen nur dann r Behälter, wenn der Optimalwert Null ist.

Um nun direkt über die Zahl der Behälter zu optimieren, können weitere r Variablen eingeführt werden. Für diese gilt

$$z_k = \begin{cases} 1, & \text{wenn Behälter } k \text{ benutzt,} \\ 0, & \text{sonst} \end{cases} = \begin{cases} 1, & \exists j \in \{1, \dots, n\} : X_{j,k} = 1, \\ 0, & \text{sonst.} \end{cases}$$

Dies entspricht den Bedingungen

$$z_k \geq X_{j,k} \quad \forall j \in \{1, \dots, n\} \quad \forall k \in \{1, \dots, r\}.$$

Da hier bei binären Variablen nur der Fall $z_k = 0$ und $X_{j,k} = 1$ bestraft werden muss, kann dies durch Addition von

$$p(X, Z) = \sum_{k=1}^r \sum_{j=1}^n (1 - z_k) X_{j,k}$$

an die Zielfunktion erzwungen werden. Damit ist die Mindestzahl an benötigten Behältern die Lösung von

$$\min \left\{ \sum_{k=1}^r z_k + P_1 q(X, Y) + P_2 p(X, z) : X \in \{0, 1\}^{n \times r}, Y \in \{0, 1\}^{r \times (D+1)}, z \in \{0, 1\}^r \right\}.$$

Da eine falsche z_k -Variable einen Unterschied von 1 ergibt, muss $P_2 > 1$ gelten. Um zu vermeiden, dass die Zuweisung der Behältervariablen in p die Bedingungen des Problems in q untergraben, muss P_1 entsprechend größer als P_2 gewählt werden.

3 Der Chimera-Graph

In diesem Kapitel wird zunächst ein spezieller Graph, der Chimera-Graph, betrachtet und das QUBO aufgestellt, welches ihn als Interaktionsgraphen besitzt. Es wird gezeigt, dass sich der vollständige Graph darin einbetten lässt. Anschließend werden einige Minoren des Chimera-Graphen und die zugehörigen QUBO-Formulierungen erstellt und ebenso deren Einbettung angegeben. Abschließend werden Beispiele gezeigt, die die Struktur der gezeigten Graphen aufgreifen.

3.1 Allgemeiner Aufbau

Der Begriff des Chimera-Graphen wurde von der Firma D-Wave-Systems eingeführt. Die Chimäre ist ein Ungeheuer der griechischen Mythologie, welches ein Mischwesen aus verschiedenen Tieren darstellt. Ebenso zeigen sich im Chimera-Graphen verschiedene Strukturen, die sich aus dem Aufbau des Prozessorchips ergeben. Der Chip ist aus sogenannten Einheitszellen aufgebaut. Sie bestehen aus Schleifen supraleitendem Materials, welche überlappend angeordnet sind. Die Flussrichtung des Stroms bedingt einen magnetischen Spin, der nach oben oder unten gerichtet sein oder sich auch in einer quantenmechanischen Überlagerung, einer sogenannten Superposition, beider Zustände befinden kann. Damit repräsentieren sie die sogenannten Quantenbits (kurz Qubits), welche die Zustände 0, 1 oder deren Superposition einnehmen können. Wo sich die Schleifen überkreuzen, beeinflussen sie sich, sind also gekoppelt. Sowohl die Schleifen als auch deren Kopplungen können gewichtet werden. Durch Anwenden eines sogenannten Energieprogramms wird der Zustand der niedrigsten Energie hergestellt. Bei der anschließenden Messung springen die Qubits entweder auf 1 oder auf 0 und das zugrundeliegende QUBO ist gelöst. Die Lösung liegt somit in klassischen Bits vor.

In Abbildung 11 ist schematisch dargestellt, wie die acht Schleifen einer Einheitszelle

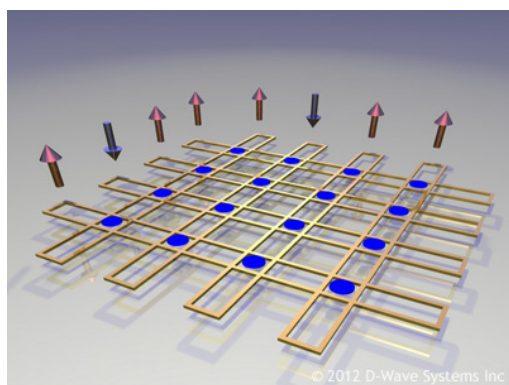


Abb. 11: Aufbau der Einheitszelle [37]

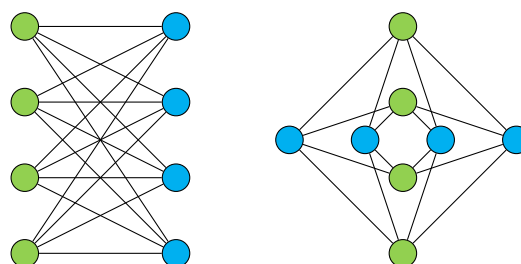


Abb. 12: Zwei Darstellungen des vollständig bipartiten Graphen $K_{4,4}$

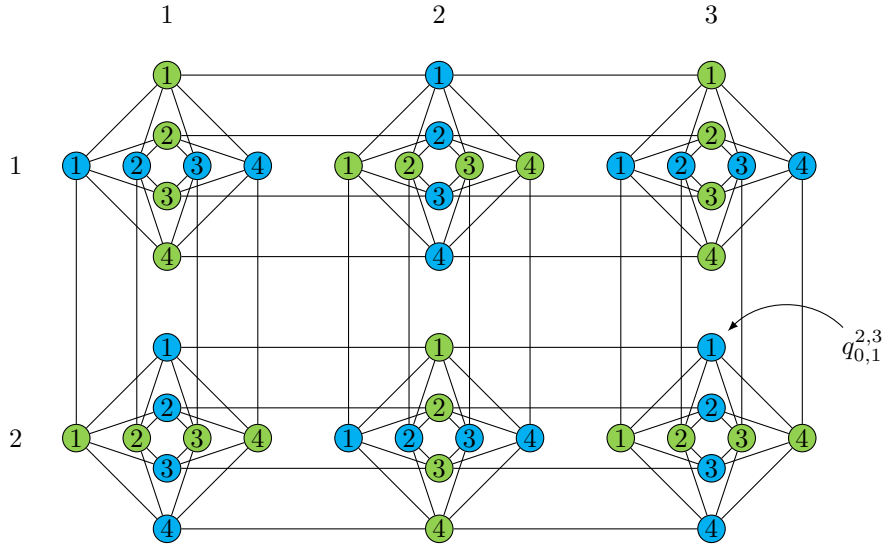


Abb. 13: Chimera-Graph $C_{2,3}$

übereinanderliegen. Es sind zwei Vierergruppen, deren Schleifen sich untereinander nicht überlappen jedoch mit allen der anderen Gruppe. Damit ergibt sich die Struktur eines vollständig bipartiten Graphen $K_{4,4}$, wie er in Abbildung 12 gezeigt ist. Diese Einheitszellen sind nun kachelartig aneinander gelegt, wobei die aneinandergrenzenden Schleifen wieder gekoppelt werden können. Der resultierende Graph ist in Abbildung 13 veranschaulicht. Im Folgenden wird ein Chimera-Graph mit n untereinander und m nebeneinander gelegten Einheitszellen mit $C_{n,m}$ bezeichnet. Theoretisch kann er bis zur Unendlichkeit in alle Richtungen fortgesetzt werden. Der dabei entstehende Graph sei C . Der aktuelle Stand der Technik ist jedoch ein Chip mit 12 mal 12 Einheitszellen also insgesamt 1152 Qubits [37]. Aufgrund von Produktionsfehlern variiert die Zahl der tatsächlich realisierten Qubits und der Kopplungen leicht nach unten.

Die Knoten des Chimera-Graphen werden anhand der Zeile r und Spalte s der Einheitszellen, der Partition p in der Einheitszelle und dem Knoten j in der Partition benannt mit

$$(r, s, p, j) \in \{1, \dots, n\} \times \{1, \dots, m\} \times \{0, 1\} \times \{1, \dots, 4\} = V(C_{n,m}).$$

Im dritten Index entspricht der Wert 0 den vertikalen Partitionen und der Wert 1 den horizontalen Partitionen, wie sie in Abbildung 13 dargestellt sind. Wie in der Abbildung weiterhin zu erkennen ist, bleibt der Graph durch das Zusammensetzen der Einheitszellen in zwei, den Farben entsprechenden, Partitionen unterteilt, deren Knoten untereinander keine Kanten besitzen. Jeder Knoten in einer Einheitszelle hat einen Ausgangsgrad von 6, bis auf die Knoten, die zum Rand liegen und denen eine Kante fehlt. Mit dieser Indizierung ergibt sich zudem, dass zwei Knoten (r_1, s_1, p_1, j_1) und (r_2, s_2, p_2, j_2) benachbart sind, wenn eine der folgenden Möglichkeiten gilt:

$$\begin{array}{lll}
 r_1 = r_2, & r_1 = r_2, & r_1 = r_2 \pm 1, \\
 s_1 = s_2, & s_1 = s_2 \pm 1, & s_1 = s_2, \\
 p_1 = 1 - p_2 & p_1 = p_2 = 0, & p_1 = p_2 = 1, \\
 & j_1 = j_2 & j_1 = j_2.
 \end{array} \quad \text{oder} \quad \text{oder} \quad \text{(XV)}$$

Die Menge der Kanten ist so $E(C_{n,m}) = E_U \cup E_H \cup E_S$ mit

$$\begin{aligned} E_U &= \{(r, s, 0, j), (r, s, 1, l)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m\}, j, l \in \{1, \dots, 4\}, \\ E_H &= \{(r, s, 0, j), (r, s+1, 0, j)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m-1\}, j \in \{1, \dots, 4\}, \\ E_S &= \{(r, s, 1, j), (r+1, s, 1, j)\} : r \in \{1, \dots, n-1\}, s \in \{1, \dots, m\}, j \in \{1, \dots, 4\}. \end{aligned}$$

Damit besitzt ein Chimera-Graph $C_{n,m}$ mit $8nm$ Knoten $16nm$ Kopplungen innerhalb der Einheitszellen, $4n(m-1)$ horizontale und $4m(n-1)$ vertikale, also gesamt genau $24nm - 4(n+m)$ Kopplungen.

Das dazugehörige Chimera-QUBO besitzt eine allgemeine, aus vier Teilen bestehende Zielfunktion $c_{n,m} : \{0, 1\}^{8mn} \rightarrow \mathbb{R}$ mit

$$\begin{aligned} c_{n,m}(q) &= c_{n,m}^g(q) + c_{n,m}^U(q) + c_{n,m}^H(q) + c_{n,m}^S(q) \quad (\text{XVI}) \\ &= \sum_{v \in V(C_{n,m})} g_v q_v + \sum_{\{v,w\} \in E_U} U_{v,w} q_v q_w + \sum_{\{v,w\} \in E_H} H_{v,w} q_v q_w + \sum_{\{v,w\} \in E_S} S_{v,w} q_v q_w \end{aligned}$$

mit $g \in \mathbb{R}^{V(C_{n,m})}$, $U \in \mathbb{R}^{E_U}$, $H \in \mathbb{R}^{E_H}$ und $S \in \mathbb{R}^{E_S}$. Werden die binären Variablen, die Qubits, ebenso mit der Zeile und Spalte der Einheitszellen, der Partition in der Einheitszelle und dem Knoten in der Partition indiziert, gilt für $q \in \{0, 1\}^{n \times m \times 2 \times 4}$, dass $q_{p,j}^{r,s} = q_{(r,s,p,j)}$ zu dem entsprechenden Knoten gehört. Damit lassen sich die entsprechenden Terme innerhalb der Einheitszellen aufstellen durch

$$c_{n,m}^g(q) + c_{n,m}^U(q) = \sum_{r=1}^n \sum_{s=1}^m \left(\sum_{p=0}^1 \sum_{j=1}^4 g_{p,j}^{r,s} q_{p,j}^{r,s} + \sum_{j=1}^4 \sum_{l=1}^4 U_{j,l}^{r,s} q_{0,j}^{r,s} q_{1,l}^{r,s} \right),$$

wobei $g \in \mathbb{R}^{n \times m \times 2 \times 4}$ die Gewichte der Knoten und $U \in \mathbb{R}^{n \times m \times 4 \times 4}$ die Stärken der Kopplungen innerhalb einer einzelnen Einheitszelle darstellen. Zu ergänzen sind die horizontalen und die vertikalen Kopplungen zwischen den Einheitszellen mit den Stärken $H \in \mathbb{R}^{n \times (m-1) \times 4}$ und $S \in \mathbb{R}^{(n-1) \times m \times 4}$:

$$\begin{aligned} c_{n,m}^H(x) &= \sum_{r=1}^n \sum_{s=1}^{m-1} \sum_{j=1}^4 H_j^{r,s} q_{0,j}^{r,s} q_{0,j}^{r,s+1}, \\ c_{n,m}^S(x) &= \sum_{r=1}^{n-1} \sum_{s=1}^m \sum_{j=1}^4 S_j^{r,s} q_{1,j}^{r,s} q_{1,j}^{r+1,s}. \end{aligned}$$

3.2 Einbettung beliebiger QUBOs

3.2.1 Allgemeines Prinzip

Wie in Abbildung 13 weiterhin zu erkennen ist, koppelt im Chimera-Graphen nicht jedes Qubit mit jedem anderen. Dadurch ist ein QUBO mit einem Chimera-Graphen als Interaktionsgraphen sehr eingeschränkt. Im Folgenden soll daher ein Weg gezeigt werden, wie ein beliebiges QUBO als ein Chimera-QUBO formuliert werden kann. Einige ähnliche Überlegungen zur Einbettungsproblematik wurden in [38] gezeigt.

Um fehlende Kopplungen zwischen Qubits zu überwinden, werden, wie bereits im vorherigen Kapitel im Teilabschnitt Kodierung ungerichteter Graphen von 2.3.5 angesprochen wurde, Qubits des Chimera-Graphen zu einem übergeordneten logischen Knoten verbunden. Durch die Verschmelzung der Qubits erhöht sich der Ausgangsgrad des entstandenen logischen Knotens gegenüber einem einzelnen Qubit, was die Kopplung zu weiteren Knoten erleichtert. Ist nun ein Ausgangs-QUBO über $x \in \{0, 1\}^V$ wie in (III) mit dem Interaktionsgraphen $G = (V, E)$ gegeben, wird jedem Knoten $v \in V$ eine Menge $I_v \subset \{1, \dots, n\} \times \{1, \dots, m\} \times \{0, 1\} \times \{1, \dots, 4\}$ mit den Indizes der zusammengefassten Qubits aus $C_{n,m}$ zugeordnet. Dabei gilt für alle $v, w \in V$ mit $v \neq w$, dass $I_v \cap I_w = \emptyset$. Für ein $J = (r, s, p, j) \in I_v$ ist $q_J = q_{(r,s,p,j)} = q_{p,j}^{r,s} \in \{0, 1\}$ die zu dem Qubit der Qubitmenge des Knotens v gehörende binäre Variable. Die Qubitmengen müssen dabei möglichst so angelegt sein, dass zwischen ihnen immer mindestens eine Kante im Chimera-Graphen gefunden werden kann, wenn die entsprechenden logischen Knoten eine Kante in E besitzen. Das heißt für alle $\{v, w\} \in E$ müssen $J \in I_v$ und $\tilde{J} \in I_w$ existieren, sodass J und \tilde{J} die Indizes benachbarter Knoten beschreiben, also (XV) erfüllen. Ist dies der Fall, ist der Ausgangsgraph ein Minor des Chimera-Graphen und die I_v zeigen die Kontraktionen an, die nötig sind, um den Chimera-Graphen in diesen umzuformen. Das heißt, $I : V \rightarrow 2^{\{1, \dots, n\} \times \{1, \dots, m\} \times \{0, 1\} \times \{1, \dots, 4\}}$ ist eine Einbettung.

Meist wird eine Kette aufeinanderfolgender Qubits zu einem sogenannten Qubitstrang zusammengefasst. Jedoch sind alle zusammenhängenden Teilgraphen des Chimera-Graphen ebenso zu einem logischen Knoten zusammenfassbar. Die Eigenschaft des Zusammenhangs ist dabei notwendig, da die Qubits durch hohe negative Stärken auf den Kanten stark gekoppelt werden müssen, damit sie alle den gleichen Wert annehmen. Denn ein logischer Knoten ist genau dann aktiv, wenn alle zusammengefassten Qubits aktiv sind, und umgekehrt nicht aktiv, wenn keines der Qubits aktiv ist. Also

$$\begin{aligned} x_v^{\min} = 1 &\Leftrightarrow q_J^{\min} = 1 \quad \forall J \in I_v, \\ x_v^{\min} = 0 &\Leftrightarrow q_J^{\min} = 0 \quad \forall J \in I_v. \end{aligned} \tag{XVII}$$

Alle weiteren Möglichkeiten geben keine Lösung des Ausgangs-QUBOs an und müssen daher ausgeschlossen werden. Dies wird bei der Belegung der Qubits mit dem gegebenen Gewicht des logischen Knotens realisiert, was in Abschnitt 3.2.3 gezeigt wird.

Graphen mit der größtmöglichen Zahl an Kanten zwischen ihren Knoten sind vollständige Graphen, welche bei N Knoten mit K_N bezeichnet werden. Damit stellt deren Einbettung in den Chimera-Graphen eine besondere Schwierigkeit dar. Über sie ist jeder beliebige Graph mit N Knoten durch Nullsetzen der Stärken fehlender Kanten auf dem Chimera-Graphen darstellbar. Deshalb wird im nächsten Abschnitt eine Möglichkeit gezeigt, wie die Einbettung vollständiger Graphen realisiert werden kann.

3.2.2 Darstellung vollständiger Graphen

Betrachtet man eine einzelne Einheitszelle des Chimera-Graphen, so stellt man fest, dass mit den acht Qubits maximal ein vollständiger Graph mit fünf logischen Knoten dargestellt werden kann, da die Qubits jeweils nur einen Ausgangsgrad von 4 haben. Dabei werden drei Paare von Knoten aus unterschiedlichen Partitionen zusammengefasst. Diese Stränge sind damit alle, sogar jeweils über zwei Kanten, untereinander gekoppelt.

Die verbleibenden beiden Knoten besitzen zueinander und zu den drei Paaren genau je eine Kopplung. Dies wird in Abbildung 14 gezeigt, wobei die verschiedenen Farben die entsprechenden logischen Knoten verdeutlichen.

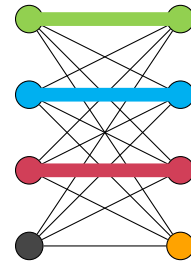


Abb. 14: K_5 in $C_{1,1}$

Wie größere vollständige Graphen erzeugt werden können, ist in [32] gezeigt. Dazu müssen die verbliebenen einzelnen Knoten der Einheitszelle zunächst ebenfalls verbunden werden, sodass die Qubits unterschiedlicher Partitionen paarweise gekoppelt sind. In einer diagonal zur ersten liegenden zweiten Einheitszelle kann ebenso ein vollständiger Graph mit vier Knoten gebildet werden. Durch Verlängern der acht entstandenen Qubitstränge über die Einheitszellen hinaus in die an beide angrenzende Einheitszelle entstehen Kanten zwischen den entsprechenden logischen Knoten. Das Prinzip kann schrittweise auf weitere diagonale Zellen ausgeweitet werden. Abbildung 15 zeigt beispielsweise, wie der Graph K_{10} über den Chimera-Graphen $C_{3,3}$ verteilt ist.

Es zeigt sich, dass diese Art der Einbettung eines vollständigen Graphen K_N mit $N \geq 4$ Knoten Qubitstränge einer Länge von mindestens $\lceil \frac{N}{4} \rceil + 1$ Qubits benötigt. Mit

$$I_{j,k} \in \{1, \dots, n\} \times \{1, \dots, m\} \times \{0, 1\} \times \{1, \dots, 4\}$$

ist

$$I_{j,*} := \left(I_{j,1}, \dots, I_{j, \lceil \frac{N}{4} \rceil + 1} \right) \tag{XVIII}$$

die geordnete Indexmenge des zum j -ten Knoten gehörenden Qubitstrangs und $q_{I_{j,k}} \in \{0, 1\}$ ist die zugehörige Variable zum k -ten Qubit dieses Strangs. Die Qubitstränge werden entlang der senkrechten Partition in der linken Spalte der Einheitszellen nach unten durchnummeriert. Wird mit der Indizierung innerhalb eines Strangs in der vertikalen Partition der linken Einheitszellen begonnen, werden zunächst weitere Qubits

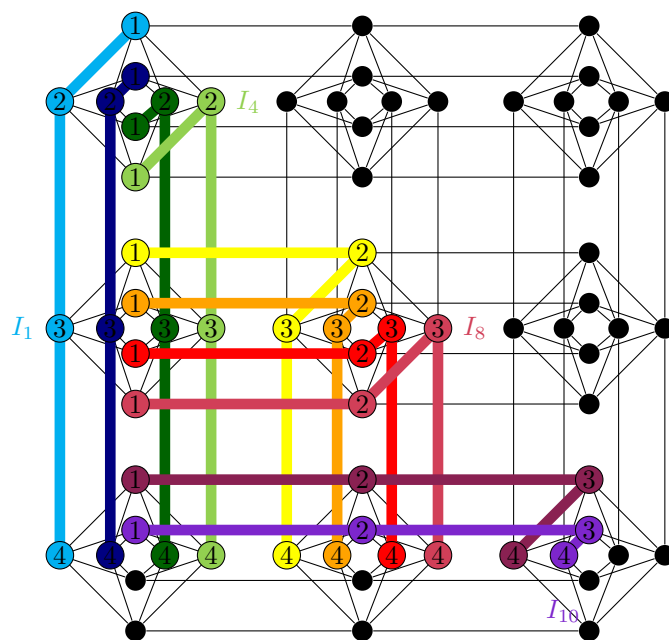


Abb. 15: Einbettung von K_{10} in den Chimera-Graphen über Qubitstränge

dieser Partition in den angrenzenden Einheitszellen durchlaufen. Dann erfolgt in der Einheitszelle auf der Diagonalen ein Wechsel in die horizontale Partition und danach ergeben sich vertikale Kopplungen zu weiteren Qubits. Damit lässt sich für $j \in \{1, \dots, N\}$ und $k \in \{1, \dots, \lceil \frac{N}{4} \rceil + 1\}$ angeben:

$$I_{j,k} = \begin{cases} \left(\left\lceil \frac{j}{4} \right\rceil, k, 0, (j-1 \bmod 4) + 1 \right) & \text{für } k \leq \left\lceil \frac{j}{4} \right\rceil, \\ \left(\left\lceil \frac{j}{4} \right\rceil, \left\lceil \frac{j}{4} \right\rceil, 1, (j-1 \bmod 4) + 1 \right) & \text{für } k = \left\lceil \frac{j}{4} \right\rceil + 1, \\ \left(k-1, \left\lceil \frac{j}{4} \right\rceil, 1, (j-1 \bmod 4) + 1 \right) & \text{für } k > \left\lceil \frac{j}{4} \right\rceil + 1. \end{cases} \quad (\text{XIX})$$

Bei dieser Einbettungsart werden durch die Qubitstränge genau $\frac{1}{2} \lceil \frac{N}{4} \rceil (\lceil \frac{N}{4} \rceil + 1) \in \mathcal{O}(N^2)$ Einheitszellen besetzt, die in einer Dreiecksform angeordnet sind. Damit wird ein Chimera-Graph mit einer Größe von mindestens $\lceil \frac{N}{4} \rceil \times \lceil \frac{N}{4} \rceil$ benötigt. Dass die quadratische Schranke auch optimal für die Qubitanzahl ist, zeigt folgendes Theorem:

Theorem 3.1 Die Zahl der benötigten Qubits zur Einbettung des vollständigen Graphen K_N in den Chimera-Graphen C liegt in $\mathcal{O}(N^2)$.

Beweis: In C haben alle Knoten einen konstanten Ausgangsgrad von 6. Damit können zwischen N Knoten maximal $3N$ Kanten in C gefunden werden und die Verschmelzung eines zusammenhängenden Teilgraphen mit einem weiteren benachbarten Qubit ergibt maximal eine Erhöhung des Ausgangsgrades um 5. Ein vollständiger Graph besitzt aber genau $\frac{1}{2}N(N+1)$ Kanten. Um diese zu erhalten, müssen daher Qubits kontrahiert werden, deren Anzahl quadratisch in N ist. \square

Umgekehrt gilt auch, dass in einen Chimera-Graphen $C_{n,m}$ mit $n, m < \infty$ maximal ein vollständiger Graph mit $N = 4 \min(n, m)$ Knoten eingebettet werden kann.

Weiterhin sind für die Einbettung auch die Kopplungen zwischen den logischen Knoten wichtig. Diese können übertragen werden, da für je zwei Qubitstränge j und l mindestens zwei Knoten existieren, die in jeweils einem der Qubitstränge liegen und zwischen denen eine Kante im Chimera-Graph existiert. Die Knoten besitzen zum Beispiel die Indizes $I_{j,o(j,l)}$ und $I_{l,i(j,l)}$ mit

$$(o(j,l), i(j,l)) = \begin{cases} (\lceil \frac{l}{4} \rceil, \lceil \frac{j}{4} \rceil + 1) & \text{für } j < l, \\ (\lceil \frac{l}{4} \rceil + 1, \lceil \frac{j}{4} \rceil) & \text{für } j > l. \end{cases}$$

Da dies symmetrisch unter Vertauschung ist, sei mit

$$O_{j,l} := I_{j,o(j,l)} = \begin{cases} I_{j, \lceil \frac{l}{4} \rceil} & \text{für } j < l, \\ I_{j, \lceil \frac{l}{4} \rceil + 1} & \text{für } j > l \end{cases} \quad (\text{XX})$$

der Index des Qubits im Strang zu j gegeben, aus dem die Kante zwischen den logischen Knoten j und l im Chimera-Graphen herausgeht.

Die Knoten des Interaktionsgraphen eines gegebenen QUBOs können nun den Qubitsträngen zugeordnet werden. Dazu seien $j_v \in \{1, \dots, |V|\}$ und $j_v \neq j_w$ für alle $v, w \in V$ mit $v \neq w$ die Durchnummerierung der Knoten entsprechend ihrer Anordnung auf den Qubitsträngen. Die Indizes des k -ten Qubits im Strang des Knotens v sind $I_{v,k} = I_{j_v,k}$ mit I wie vorher definiert. Damit das entsprechende Chimera-QUBO über den $q_{I^*,*} \in$

$\{0, 1\}^{|V| \times (\lceil \frac{|V|}{4} \rceil + 1)}$ eine Lösung des Ausgangs-QUBOs darstellt, muss also nach (XVII) bei der Einbettung über den vollständigen Graphen K_V für jeden logischen Knoten $v \in V$ gelten

$$\begin{aligned} x_v^{\min} = 1 &\Leftrightarrow q_{I_{v,k}}^{\min} = 1 \quad \forall k = 1, \dots, \left\lceil \frac{|V|}{4} \right\rceil + 1, \\ x_v^{\min} = 0 &\Leftrightarrow q_{I_{v,k}}^{\min} = 0 \quad \forall k = 1, \dots, \left\lceil \frac{|V|}{4} \right\rceil + 1. \end{aligned}$$

Die Qubits, die zwei Qubitstränge verbinden, können nun analog zu (XX) durch die Indizes $O_{v,w}$ und $O_{w,v}$ angegeben werden mit

$$O_{v,w} = O_{j_v, j_w} = \begin{cases} I_{v, \lceil \frac{j_w}{4} \rceil} & \text{für } j_v < j_w, \\ I_{v, \lceil \frac{j_w}{4} \rceil + 1} & \text{für } j_v > j_w. \end{cases}$$

3.2.3 Gewichte und Stärken

Wurde eine passende Grundstruktur aus Qubitsträngen für das QUBO zu einem Graphen $G = (V, E)$ gefunden, können die Gewichte der logischen Knoten eingebaut werden. Damit wird zugleich Bedingung (XVII) erzwungen. Besitzt der zu einem Knoten gehörige Qubitstrang eine Länge von $L > 1$, ergibt sich dessen Gewicht $g \in \mathbb{R}$ aus der Summe über den Gewichten aller L Qubits und den Stärken der $L - 1$ Kopplungen in dem Qubitstrang. Um die Symmetrie innerhalb des Strangs zu gewährleisten, erhalten die Gewichte einen einheitlichen Wert $\bar{g} \in \mathbb{R}$ ebenso wie die Stärken $\bar{S} \in \mathbb{R}$. Damit gilt

$$g = L\bar{g} + (L - 1)\bar{S} = (L - 1)(\bar{g} + \bar{S}) + \bar{g}. \quad (\text{XXI})$$

Um das auf den logischen Knoten gegebene QUBO lösen zu können, müssen die Qubits der Qubitstränge synchronisiert werden. Daher müssen auch hier die Auswirkungen der Verletzung dieser Bedingung abgeschätzt werden. Dazu wird ein anderer Ansatz als in [38] verfolgt. Eine Verletzung tritt dann ein, wenn mindestens einer der L Knoten mit einem anderen Wert als die anderen belegt wird. Um eine starke Kopplung unter ihnen zu gewährleisten, muss aufgrund der Minimierung des QUBOs stets $\bar{S} < 0$ gelten. Eine fälschliche Aktivierung eines Qubits, wenn kein anderes im Strang aktiv ist, wird durch $\bar{g} > 0$ verhindert. Auch dürfen kürzere Teilstränge nicht weniger oder gleiches Gewicht besitzen wie der gesamte Strang, damit eine Stückelung keine Verringerung der Zielfunktion bewirkt. Dabei ist irrelevant, ob g positiv oder negativ ist. Das heißt, es muss für alle $j = 1, \dots, L - 1$ gelten

$$j\bar{g} + (j - 1)\bar{S} > g.$$

Diese Bedingung ist klar erfüllt, wenn rekursiv gilt

$$j\bar{g} + (j - 1)\bar{S} > (j + 1)\bar{g} + j\bar{S} = j\bar{g} + (j - 1)\bar{S} + \bar{g} + \bar{S}.$$

Das heißt, je kürzer der Qubitstrang ist, desto höher ist dessen Gewicht. Dies ist erfüllt, wenn $\bar{g} < |\bar{S}|$ gilt. Wäre weiterhin $|\bar{S}| \geq 2\bar{g}$, würde der geringsten Fehler, mit einem Wert von \bar{g} , durch die oben genannte Aktivierung eines einzelnen Qubits im Strang erzeugt werden. Dies ist somit immer der Fall, wenn der Strang nur aus zwei Qubits

besteht und das Gewicht des logischen Knotens negativ ist. Andernfalls, wenn $|\bar{S}| < 2\bar{g}$ ist, ergibt sich die geringste Strafe von genau $|\bar{g} + \bar{S}| < \bar{g}$, wenn einer der beiden Endknoten des Strangs deaktiviert ist, obwohl alle anderen aktiv sind. Soll der Fehler für die Qubitstränge aller logischen Knoten $v \in V$ mit der Länge $L_v > 1$ auf einen Wert $P \in \mathbb{R}_{>0}$ gesetzt werden, können in diesem Fall durch Ersetzen von $-(\bar{g}_v + \bar{S}_v)$ mit P in (XXI) mit

$$\begin{aligned}\bar{g}_v &= g_v + (L_v - 1)P, \\ \bar{S}_v &= -\bar{g}_v - P = -g_v - L_v P\end{aligned}\tag{XXII}$$

die entsprechenden Gewichte \bar{g}_v und Stärken \bar{S}_v im Qubitstrang von v berechnet werden. Sinnvoll erscheint dabei die Wahl $P = 1$. Jedoch muss P dabei immer $\bar{g}_v > 0$ und $\bar{S}_v < 0$ für alle $v \in V$ einhalten. Damit müssen insbesondere negative Gewichte g_v beachtet werden und es folgt aus $(L_v - 1)P > -g_v$ die Bedingung

$$P > \max \left\{ 0, \frac{-g_v}{L_v - 1} : v \in V \right\}.\tag{XXIII}$$

Ist diese erfüllt, gilt auch $\bar{S}_v < 0$ für alle $v \in V$. Dies wurde jedoch für den Fall $-(\bar{g}_v + \bar{S}_v) < \bar{g}_v$ hergeleitet. Aus dieser Ungleichung ergibt sich $P < g_v + (L_v - 1)P$ und damit für $L_v > 2$ die weitere Bedingung

$$P > \max \left\{ 0, \frac{-g_v}{L_v - 2} : v \in V \right\},\tag{XXIV}$$

welche stärker ist, als die obige.

Im letzten Schritt werden die Stärken S_e der Kopplungen $e \in E$ des Ausgangs-QUBOs auf die bestehenden Kanten zwischen den Qubitsträngen gesetzt. Dabei kann es vorkommen, dass an ein einzelnes Qubit alle Kanten mit negativen Stärken angebunden werden. Dies würde bei einer Aktivierung dieser Kanten auch zu einer Aktivierung des entsprechenden Qubits führen, wenn dessen Gewicht zu gering ist, um eine Verringerung der Zielfunktion zu verhindern, obwohl die weiteren Qubits im Strang nicht aktiv sind. Daher muss eine weitere Bedingung an \bar{g}_v für alle $v \in V$ gestellt werden:

$$\bar{g}_v > -\check{\sigma}(S_{v,*}) = \sum_{w \in d(v)} \frac{1}{2} (\text{sgn}(S_{v,w}) - 1) S_{v,w} =: -\check{\sigma}(v),$$

mit der Funktion $\check{\sigma} : V \rightarrow \mathbb{R}_{\leq 0}$, die Summe der negativen Stärken der ausgehenden Kanten eines Knotens v zurückgibt. Mit $S_{v,w} = 0$, wenn $\{v, w\} \notin E$, kann die Summe auf alle $w \in V$ verallgemeinert werden. Sind keine negativen Stärken gegeben, entfällt diese Bedingung und kann P wie oben gewählt werden. Ansonsten muss nach (XXII) für alle $v \in V$ gelten

$$P > \max \left\{ 0, \frac{-g_v - \check{\sigma}(v)}{L_v - 1} : v \in V \right\},$$

was eine stärkere Bedingung als (XXIII) ist. Bezüglich (XXIV) ist sie für $L_v > 2$ und negative Gewichte g_v jedoch nur stärker, wenn gilt

$$g_v > (L_v - 2)\check{\sigma}(v).$$

Eine analoge Betrachtung für den Fall einer fälschlichen Deaktivierung eines Qubits

im Strang muss ebenso durchgeführt werden. Diese ist nur dann effektiv, wenn durch eingehende Kanten mit negativen Stärken der Teilstrang eine Verringerung der Zielfunktion liefert, also

$$(L - 1)\bar{g}_v + (L - 2)\bar{S}_v + \check{\sigma}(v) = g_v + P + \check{\sigma}(v) < 0$$

ist. Daher könnte $P > -g_v - \check{\sigma}(v)$ für alle $v \in V$ gefordert werden, was sehr große Werte für P zur Folge haben kann. Jedoch muss die Strafe P nur groß genug sein, um eine Aktivierung eines Qubits im Strang zu dem Knoten v entgegen den hohen Stärken auf den eingehenden Kanten zu erzwingen. Dabei spielen die positiven Stärken der ausgehenden Kopplungen eine wichtige Rolle und es muss gelten

$$P > \hat{\sigma}(S_{v,*}) = \sum_{w \in V} \frac{1}{2} (\text{sgn}(S_{v,w}) + 1) S_{v,w} =: \hat{\sigma}(v),$$

wobei $\hat{\sigma} : V \rightarrow \mathbb{R}_{\geq 0}$ die Funktion ist, die die Summe der positiven Stärken zurückgibt. In diesem Fall können Teilstränge noch immer die Zielfunktion verringern, aber der Betrag ist durch den gesamten Strang größer. Hier genügt also die Wahl der schwächeren Bedingung für einen Knoten v , womit für alle Knoten folgt:

$$P > \max \{0, \min\{-g_v - \check{\sigma}(v), \hat{\sigma}(v)\} : v \in V\}.$$

Insgesamt ergibt sich so:

Folgerung 3.2 Mit $\check{\sigma}$ und $\hat{\sigma}$ wie oben definiert, $\varepsilon > 0$ und $P_\varepsilon : \mathbb{R}^V \times \mathbb{R}^{V \times V} \times \mathbb{N}^V \rightarrow \mathbb{R}_{>0}$ mit

$$P_\varepsilon(g, S, L) := \max \left\{ 0, \min\{0, -g_v - \check{\sigma}(v), \hat{\sigma}(v)\}, \frac{-g_v}{L_v - 2}, \frac{-g_v - \check{\sigma}(v)}{L_v - 1} : v \in V \right\} + \varepsilon$$

ist (XVII) für eine Einbettung I des Graphen $G = (V, E)$ mit den Längen $L \in \mathbb{N}^V$ der Stränge, den Gewichten $g \in \mathbb{R}^V$ und den Stärken $S \in \mathbb{R}^{V \times V}$ in den Chimera-Graphen erfüllt.

Die gezeigten Schranken an P könnten noch verbessert werden, da jedes Qubit nur maximal sechs Nachbarn besitzt. Demzufolge könnten die Summen über den Stärken zu einzelnen Termen für jedes Qubit in einem Strang aufgesplittet werden und darüber der entsprechende Maximalwert bestimmt werden. Dies ist jedoch zu aufwändig, um hier genauer dargestellt zu werden. Gilt zudem eine Begrenzung der Größe der Gewichte und Stärken im Chimera-QUBO auf einen bestimmten Wertebereich, wie im Falle des Quantencomputers von D-Wave Systems auf $g, S \in [-10, 10]$, können entsprechende \bar{g} und \bar{S} über ein passendes P manchmal nur durch eine Skalierung des gesamten Ausgangs-QUBOs erhalten werden, da für ein $\mu > 1$

$$\begin{aligned} & \min \left\{ \sum_{v \in V} g_v x_v + \sum_{\{v,w\} \in E} S_{v,w} x_v x_w : x \in \{0, 1\}^V \right\} \\ &= \mu \min \left\{ \sum_{v \in V} \frac{g_v}{\mu} x_v + \sum_{\{v,w\} \in E} \frac{S_{v,w}}{\mu} x_v x_w : x \in \{0, 1\}^V \right\} \end{aligned}$$

gilt. Da $\check{\sigma}$ und $\hat{\sigma}$ linear in den Stärken sind, skaliert P ebenso mit

$$P_\varepsilon(g, S, L) = \mu P_{\frac{\varepsilon}{\mu}}\left(\frac{1}{\mu}g, \frac{1}{\mu}S, L\right).$$

Das Gleiche gilt somit für die Gewichte und Stärken im Strang.

Mit den gezeigten Umformungen entspricht nun die Lösung des Chimera-QUBOs einer korrekten Aktivierung der Qubitstränge und damit einer Lösung des ursprünglichen QUBOs. Analog sind nicht nur Stränge sondern auch andere Teilgraphen zu logischen Knoten zusammenfassbar, wobei die obige Abschätzung im konkreten Einzelfall neu durchgeführt werden müsste. Lässt man bei diesen Teilgraphen jedoch genau so viele Kanten weg, dass die Graphen nur noch minimal zusammenhängend sind, können die gezeigten Umformungen ebenso angewandt werden. Denn, da keine Kante entfernt werden kann, so dass der Graph immer noch zusammenhängend ist, ist die Zahl der Kanten genau um eins geringer als die Zahl der Knoten. Für die Einbettung eines beliebigen Graphen mit einer allgemeinen Zielfunktion wie (III) über den vollständigen Graphen kann nun Folgendes festgestellt werden:

Folgerung 3.3 Für einen Graphen $G = (V, E)$ mit Gewichten $g \in \mathbb{R}^V$ und Stärken $S \in \mathbb{R}^E$ gilt mit $P = P_\varepsilon\left(g, S, \left(\left\lceil \frac{|V|}{4} \right\rceil + 1\right) \mathbb{1}_V\right)$ für ein $\varepsilon > 0$ wie vorher definiert, ein $\mu > 1$ und I und O , wie für den vollständigen Graphen K_V angegeben, dass

$$\begin{aligned} & \min \left\{ \sum_{v \in V} g_v x_v + \sum_{\{v,w\} \in E} S_{v,w} x_v x_w : x \in \{0, 1\}^V \right\} \\ &= \min \left\{ \sum_{v \in V} \left(\left(g_v + \left\lceil \frac{|V|}{4} \right\rceil P \right) \sum_{k=1}^{\left\lceil \frac{|V|}{4} \right\rceil + 1} q_{I_{v,k}} + \left(-g_v - \left(\left\lceil \frac{|V|}{4} \right\rceil + 1 \right) P \right) \sum_{k=1}^{\left\lceil \frac{|V|}{4} \right\rceil} q_{I_{v,k}} q_{I_{v,k+1}} \right) \right. \\ & \quad \left. + \sum_{\{v,w\} \in E} S_{v,w} q_{O_{v,w}} q_{O_{w,v}} : q \in \{0, 1\}^{V \times (\left\lceil \frac{|V|}{4} \right\rceil + 1)} \right\} \\ &= \mu \min \left\{ \sum_{v \in V} \left(\left(\frac{g_v}{\mu} + \left\lceil \frac{|V|}{4} \right\rceil \frac{P}{\mu} \right) \sum_{k=1}^{\left\lceil \frac{|V|}{4} \right\rceil + 1} q_{I_{v,k}} + \left(-\frac{g_v}{\mu} - \left(\left\lceil \frac{|V|}{4} \right\rceil + 1 \right) \frac{P}{\mu} \right) \sum_{k=1}^{\left\lceil \frac{|V|}{4} \right\rceil} q_{I_{v,k}} q_{I_{v,k+1}} \right) \right. \\ & \quad \left. + \sum_{\{v,w\} \in E} \frac{S_{v,w}}{\mu} q_{O_{v,w}} q_{O_{w,v}} : q \in \{0, 1\}^{V \times (\left\lceil \frac{|V|}{4} \right\rceil + 1)} \right\}. \end{aligned}$$

3.2.4 Allgemeines Einbettungsproblem

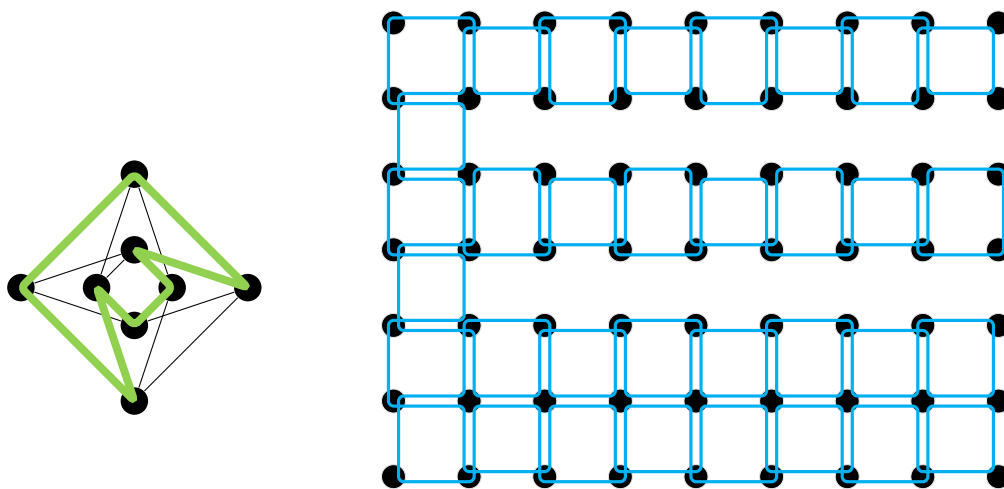
Wie in Abschnitt 3.2.2 gezeigt wurde, ist K_{4n} Minor des Chimera-Graphen $C_{n,m}$ mit $n \leq m$. Damit ist jeder Graph mit weniger oder genau $4n$ Knoten ein Minor von $C_{n,m}$. Das heißt, auch für einen Graphen mit endlich vielen Knoten können immer $n, m \in \mathbb{N}$ gefunden werden, sodass der Graph Minor von $C_{n,m}$ ist. Da im unendlich großen Chimera-Graphen C ein unendlich großer vollständiger Graph eingebettet werden kann, ist prinzipiell jeder Graph mit abzählbar vielen Knoten ein Minor von C . Jedoch stehen in der Realität nie unbegrenzt große Chips zur Verfügung. Daher ist die Frage, welche Graphen mit mehr als $4n$ Knoten Minor eines Chimera-Graphen $C_{n,m}$

mit festen n und m sind. Einige spezielle Fälle für Minoren werden daher im nächsten Abschnitt gezeigt.

Hier wird im Folgenden auf die Lösbarkeit der Fragestellung eingegangen werden, wann ein beliebiger Graph Minor eines anderen Graphen, speziell eines begrenzten Chimera-Graphen, ist. Zu entscheiden, ob ein Graph G einen Graphen H als Minor enthält, ist im Allgemeinen NP-schwer: Sei beispielsweise H ein Kreisgraph mit der gleichen Knotenzahl wie G . Dann ist die Entscheidung, ob H Minor von G ist, äquivalent zur Frage, ob G einen Hamiltonkreis enthält, und dieses Problem ist nach [28] NP-schwer. Ist hingegen H beliebig aber fest gewählt und nur G Teil der Eingabe, können N. Robertson und P. Seymour in [39] zeigen, dass das Problem in einer Zeit in $\mathcal{O}(n^3)$ lösbar ist, wobei n die Zahl der Knoten von G ist. Jedoch versteckt sich in der O-Notation eine Konstante, deren Wert superexponential von H abhängt. Das heißt, die umgekehrte Variante bleibt im Allgemeinen NP-schwer.

Der Chimera-Graph besitzt aber eine spezielle Struktur. In $C_{1,1} = K_{4,4}$ lässt sich immer ein Hamiltonkreis finden, da die Partitionen gleich groß sind und immer zwischen deren Qubits gewechselt werden kann. Ein Beispiel ist in Abbildung 16(a) gezeigt. Bricht man diesen Kreis auf, um weitere benachbarte Einheitszellen hinzunehmen zu können, geschieht das immer zwischen Knoten unterschiedlicher Partitionen. Dies zeigt, dass ein Hamiltonkreis in einem Chimera-Graphen $C_{n,m}$ für jede Einheitszelle immer genauso viele horizontale wie vertikale aus der Einheitszelle herauszeigende Kanten umfassen muss. Daher existiert für $n = 1$ und $m > 1$ kein Hamiltonkreis in $C_{n,m}$, da hier keine vertikalen Kanten vorhanden sind.

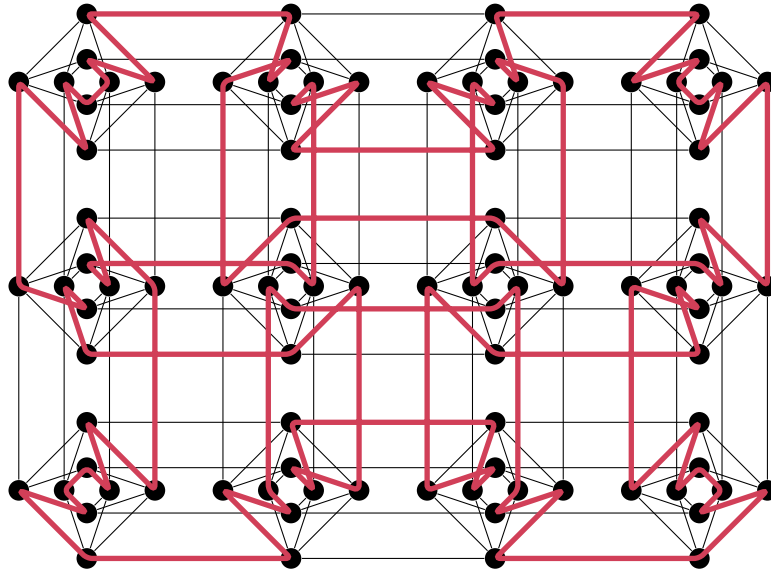
Aufgrund des Zusammenbaus aus vollständig bipartiten Graphen lässt sich jedoch für $n, m > 1$ immer ein Hamiltonkreis finden: Eine Einheitszelle kann bis zu viermal von einem Hamiltonkreis über eine horizontale Kopplung betreten und über eine vertikale wieder verlassen werden, oder umgekehrt. Dabei können immer paarweise weitere Qubits aus unterschiedlichen Partitionen der Zelle mit in den Kreis eingeschlossen werden. Für die zugrundeliegende Gitterstruktur $L_{n,m}$ kann immer ein geschlossener Pfad gefunden werden, der jeden Knoten mindestens einmal umfasst und dabei immer



(a) Hamiltonkreis in $K_{4,4}$

(b) geschlossener Pfad über alle Knoten in $L_{n,m}$

Abb. 16: Zusammensetzung eines Hamiltonkreises in $C_{n,m}$

Abb. 17: Hamiltonkreis in $C_{3,4}$

zwischen horizontalen und vertikalen Kanten wechselt. Dazu muss jeder Knoten nur maximal viermal betreten werden, wenn nach dem erweiterbaren Schema in Abbildung 16(b) vorgegangen wird. Demzufolge kann mit diesem Schema über den Einheitszellen, wie beispielsweise in Abbildung 17 für $C_{3,4}$ gezeigt, ein Hamiltonkreis aufgebaut werden.

Damit stellt die Suche danach im Chimera-Graphen kein schweres Problem dar und obiges Argument für die NP-Schwere der Minorenrelation kann nicht angewandt werden. Da auch weitere Probleme für bipartite Graphen, im Gegensatz zu beliebigen Graphen, nicht NP-schwer sind, ist es möglich, dass ein polynomieller Algorithmus existiert, der entscheidet, ob ein beliebiger Graph in den Chimera-Graphen $C_{n,m}$ eingebettet werden kann. Letzteres ist äquivalent zur Suche nach den kleinsten n und m , so dass der gegebene Graph Minor von $C_{n,m}$ ist.

3.3 Spezielle Minoren

Der Chimera-Graph ist ein sehr spezieller Graph, der nur sehr selten selbst als Interaktionsgraph der QUBO-Formulierung eines Problems auftritt. Daher werden im Folgenden Minoren des Chimera-Graphen betrachtet, die eine möglicherweise häufiger vorkommende Grundstruktur aufweisen. Des Weiteren werden die QUBOs zu diesen Graphen aufgestellt und eine Einbettungsmöglichkeit in den Chimera-Graphen gezeigt. In einer Tabelle am Ende dieses Abschnitts werden einige Kennzahlen der in den folgenden Abschnitten vorgestellten Graphen noch einmal gegenüber gestellt.

3.3.1 Vollständig bipartiter Graph

Neben vollständigen Graphen spielen auch vollständig bipartite Graphen eine große Rolle, beispielsweise bei Zuordnungsproblemen. Die Knoten einer Partition besitzen

dabei keine Kopplungen untereinander, jedoch zu allen Knoten der anderen Partition. Besitzt die eine Partition n und die andere m Knoten, wird der Graph mit $K_{n,m}$ bezeichnet. Es gilt $V(K_{n,m}) = V_0 \cup V_1$ mit $V_0 = \{1, \dots, n\}$ und $V_1 = \{1, \dots, m\}$ und $E(K_{n,m}) = \{(j, l) : j \in V_0, l \in V_1\}$. Dieser Graph besitzt dann $n + m$ Knoten und genau nm Kanten. Ein QUBO mit Interaktionsgraphen $K_{n,m}$ benötigt daher $n + m$ binäre Variablen $x^0 \in \{0, 1\}^n$, $x^1 \in \{0, 1\}^m$. Mit den Gewichten $g^0 \in \mathbb{R}^n$, $g^1 \in \mathbb{R}^m$ und den Stärken $S \in \mathbb{R}^{n \times m}$ kann die Zielfunktion über

$$k_{n,m}(x) = \sum_{j=1}^n g_j^0 x_j^0 + \sum_{l=1}^m g_l^1 x_l^1 + \sum_{j=1}^n \sum_{l=1}^m S_{j,l} x_j^0 x_l^1$$

aufgestellt werden.

Der Chimera-Graph ist bereits bipartit, wobei im Schachbrettmuster über den Einheitszellen gewechselt wird, in welche Partition die horizontalen Knoten und in welche die vertikalen gehören. Die vertikalen Knoten einer Einheitszelle gehören damit zu einer anderen Partition als die vertikalen Knoten der Nachbarzelle. Werden diese Knoten entlang der horizontalen Kopplungen miteinander verschmolzen, besitzen die daraus entstandenen Knoten weiterhin keine Kante untereinander. Verfährt man ebenso mit den horizontalen Knoten und ihren Äquivalenten in den oben und unten benachbarten Zellen, entsteht eine weitere Partition. Das Prinzip ist in Abbildung 18 veranschaulicht. Es zeigt sich, dass durch die Verschmelzung über ganze Zeilen und Spalten hinweg je zwei Knoten aus verschiedenen Partitionen über die diagonalen Kopplungen innerhalb der Einheitszellen benachbart sind. Aus $C_{n,m}$ kann also der vollständig bipartite Graph $K_{4n,4m}$ gebildet werden. Andersherum benötigt ein Graph $K_{N,M}$ für die Einbettung in den Chimera-Graphen also $\lceil \frac{N}{4} \rceil$ mal $\lceil \frac{M}{4} \rceil$ Einheitszellen, weswegen die folgende Aussage getroffen werden kann:

Theorem 3.4 Die Zahl der benötigten Qubits zur Einbettung des vollständig bipartiten Graphen $K_{N,M}$ in den Chimera-Graphen C liegt in $\mathcal{O}(NM)$.

Beweis: Analog zum Beweis von Theorem 3.1 können durch den konstanten Ausgangsgrad der Qubits nur so die benötigten NM Kanten des Graphen gebildet werden. \square

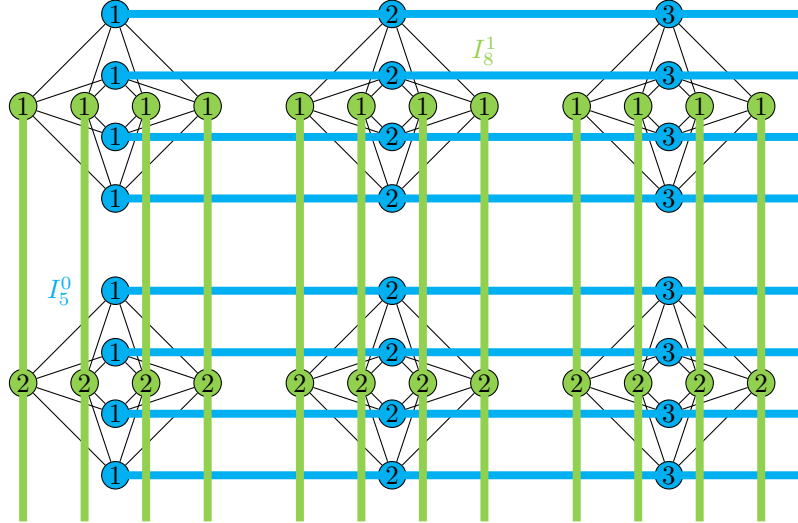
Bei der Einbettung von $K_{4n,4m}$ in $C_{n,m}$ verbinden die horizontalen Stränge der Länge m die Knoten der ersten Partition und die vertikalen Stränge der Länge n die der zweiten. Damit lassen sich die Indextmengen folgendermaßen aufstellen:

$$I_{j,k}^0 = \left(\lceil \frac{j}{4} \rceil, k, 0, (j - 1 \bmod 4) + 1 \right)$$

für $j = 1, \dots, 4n$ mit $k = 1, \dots, m$ und

$$I_{j,k}^1 = \left(k, \lceil \frac{j}{4} \rceil, 1, (j - 1 \bmod 4) + 1 \right)$$

für $j = 1, \dots, 4m$ mit $k = 1, \dots, n$. Die Kopplung zwischen zwei Knoten $j \in \{1, \dots, 4n\}$ und $l \in \{1, \dots, 4m\}$ findet damit in der Einheitszelle in der Zeile $\lceil \frac{j}{4} \rceil$ und der Spalte $\lceil \frac{l}{4} \rceil$ statt. Demzufolge wird die Kante durch das $\lceil \frac{l}{4} \rceil$ -te Element des Stranges zu j und das $\lceil \frac{j}{4} \rceil$ -te Element des Stranges zu l realisiert. Die Gewichte und Stärken in diesen Strängen können wie in Abschnitt 3.2.3 ermittelt werden. Mit analogen P wie in Folgerung 3.2 lässt sich nun das zugehörige Chimera-QUBO aufstellen:


 Abb. 18: Qubitstränge im Chimera-Graphen C zum vollständig bipartiten Graphen

Folgerung 3.5 Mit $\varepsilon > 0$, $g = (g^0, g^1) \in \mathbb{R}^{4n+4m}$, $L = (m\mathbb{1}_{4n}, n\mathbb{1}_{4m}) \in \mathbb{N}^{4n+4m}$, $\tilde{S} = \begin{pmatrix} 0 & S \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{(4n+4m) \times (4n+4m)}$ und $P = P_\varepsilon(g, \tilde{S}, L)$ gilt

$$\begin{aligned} & \min \left\{ k_{4n,4m}(x) : x \in \{0, 1\}^{4n+4m} \right\} \\ = & \min \left\{ \sum_{j=1}^{4n} \left((g_j^0 + (m-1)P) \sum_{k=1}^m q_{I_{j,k}^0} + (-g_j^0 - mP) \sum_{k=1}^{m-1} q_{I_{j,k}^0} q_{I_{j,k+1}^0} \right) \right. \\ & + \sum_{j=1}^{4m} \left((g_j^1 + (n-1)P) \sum_{k=1}^n q_{I_{j,k}^1} + (-g_j^1 - nP) \sum_{k=1}^{n-1} q_{I_{j,k}^1} q_{I_{j,k+1}^1} \right) \\ & \left. + \sum_{j=1}^{4n} \sum_{l=1}^{4m} S_{j,l} q_{I_{j, \lceil \frac{l}{4} \rceil}^0} q_{I_{l, \lceil \frac{j}{4} \rceil}^1} : q \in \{0, 1\}^{n \times m \times 2 \times 4} \right\}. \end{aligned}$$

Allgemein stellt der Chimera-Graph $C_{n,m}$ ein Gitter mit n mal m Kreuzungspunkten dar, an denen jeweils ein vollständig bipartiter Graph $K_{4,4}$ liegt. Daher wird für diesen Graphen die Notation $L_{n,m}[K_{4,4}]$ eingeführt. Nun kann eine Verallgemeinerung getroffen werden: Werden immer $k > 1$ Zeilen und $l > 1$ Spalten von Einheitszellen, wie oben gezeigt, zu einem vollständig bipartiten Graphen $K_{4k,4l}$ zusammengefasst, kann ein analoges Gitter über diesen Graphen gebildet werden, also $L_{\frac{n}{k}, \frac{m}{l}}[K_{4k,4l}]$, wenn k Teiler von n und l Teiler von m ist. Der entstandene Graph besitzt die Knotenmenge $V(L_{\frac{n}{k}, \frac{m}{l}}[K_{4k,4l}]) = V_0 \cup V_1$ mit

$$\begin{aligned} V_0 &= \{1, \dots, \frac{n}{k}\} \times \{1, \dots, \frac{m}{l}\} \times \{0\} \times \{1, \dots, 4k\}, \\ V_1 &= \{1, \dots, \frac{n}{k}\} \times \{1, \dots, \frac{m}{l}\} \times \{1\} \times \{1, \dots, 4l\} \end{aligned}$$

und die Kantenmenge $E(L_{\frac{n}{k}, \frac{m}{l}}[K_{4k,4l}]) = E_U \cup E_H \cup E_S$ mit

$$\begin{aligned} E_U &= \{ \{(r, s, 0, j), (r, s, 1, t)\} : r \in \{1, \dots, \frac{n}{k}\}, s \in \{1, \dots, \frac{m}{l}\}, j \in \{1, \dots, 4k\}, t \in \{1, \dots, 4l\} \}, \\ E_H &= \{ \{(r, s, 0, j), (r, s+1, 0, j)\} : r \in \{1, \dots, \frac{n}{k}\}, s \in \{1, \dots, \frac{m}{l} - 1\}, j \in \{1, \dots, 4k\} \}, \end{aligned}$$

3.3 Spezielle Minoren

$$E_S = \{(r, s, 1, t), (r + 1, s, 1, t)\} : r \in \{1, \dots, \frac{n}{k} - 1\}, s \in \{1, \dots, \frac{m}{l}\}, t \in \{1, \dots, 4l\}\}.$$

Er ist ebenfalls bipartit und seine Partitionen umfassen ebenso schachbrettmusterartig über die Einheitszellen verteilt $4k \lceil \frac{nm}{2kl} \rceil + 4l \lfloor \frac{nm}{2kl} \rfloor$ beziehungsweise $4l \lceil \frac{nm}{2kl} \rceil + 4k \lfloor \frac{nm}{2kl} \rfloor$ Knoten. Sobald n oder m gerade ist, sind sie damit gleich groß und umfassen jeweils $2nm \left(\frac{1}{k} + \frac{1}{l}\right)$ Knoten.

Das QUBO dieses Interaktionsgraphen verallgemeinert die Zielfunktion des Chimera-Graphen (XVI) zu

$$c_{n,m,k,l}(x) = c_{n,m,k,l}^g(x) + c_{n,m,k,l}^U(x) + c_{n,m,k,l}^H(x) + c_{n,m,k,l}^S(x)$$

über den binären Variablen $x_0 \in \{0, 1\}^{\frac{n}{k} \times \frac{m}{l} \times 4k}$ und $x_1 \in \{0, 1\}^{\frac{n}{k} \times \frac{m}{l} \times 4l}$. Die Summanden sind dabei

$$\begin{aligned} c_{n,m,k,l}^g(x) + c_{n,m,k,l}^U(x) &= \sum_{r=1}^{\frac{n}{k}} \sum_{s=1}^{\frac{m}{l}} \left(\sum_{j=1}^{4k} g_{0,j}^{r,s} x_{0,j}^{r,s} + \sum_{j=1}^{4l} g_{1,j}^{r,s} x_{1,j}^{r,s} + \sum_{j=1}^{4k} \sum_{t=1}^{4l} U_{j,t}^{r,s} x_{0,j}^{r,s} x_{1,t}^{r,s} \right), \\ c_{n,m,k,l}^H(x) &= \sum_{r=1}^{\frac{n}{k}} \sum_{s=1}^{\frac{m}{l}-1} \sum_{j=1}^{4k} H_j^{r,s} x_{0,j}^{r,s} x_{0,j}^{r,s+1}, \\ c_{n,m,k,l}^S(x) &= \sum_{r=1}^{\frac{n}{k}-1} \sum_{s=1}^{\frac{m}{l}} \sum_{j=1}^{4l} S_j^{r,s} x_{1,j}^{r,s} x_{1,j}^{r+1,s} \end{aligned}$$

für die Gewichte $g_0 \in \mathbb{R}^{\frac{n}{k} \times \frac{m}{l} \times 4k}$, $g_1 \in \mathbb{R}^{\frac{n}{k} \times \frac{m}{l} \times 4l}$ und die Stärken $U \in \mathbb{R}^{\frac{n}{k} \times \frac{m}{l} \times 4k \times 4l}$, $H \in \mathbb{R}^{\frac{n}{k} \times (\frac{m}{l}-1) \times 4k}$ und $S \in \mathbb{R}^{(\frac{n}{k}-1) \times \frac{m}{l} \times 4l}$.

Die Indexmengen $I_{0,j}^{r,s}$ der Knoten der vertikalen Partitionen mit den horizontal verbundenen Qubits besitzen die Länge l und können für $r = 1, \dots, \frac{n}{k}$, $s = 1, \dots, \frac{m}{l}$ und $j = 1, \dots, 4k$ angegeben werden mit

$$I_{0,j,u}^{r,s} = (k(r-1) + \lceil \frac{j}{4} \rceil, l(s-1) + u, 0, (j-1 \bmod 4) + 1)$$

für $u = 1, \dots, l$. Analog ergeben sich die Indexmengen $I_{1,j}^{r,s}$ für die Knoten der horizontalen Partitionen mit den vertikal verbundenen Qubits für $r = 1, \dots, \frac{n}{k}$, $s = 1, \dots, \frac{m}{l}$, $j = 1, \dots, 4l$ und $u = 1, \dots, k$ mit

$$I_{1,j,u}^{r,s} = (k(r-1) + u, l(s-1) + \lceil \frac{j}{4} \rceil, 1, (j-1 \bmod 4) + 1).$$

Daraus ergibt sich eine analoge Einbettung wie in Folgerung 3.5, jedoch ergänzt um die Zeilen- und Spaltenindizes der vollständig bipartiten Graphen und um die horizontalen und vertikalen Kopplungen unter ihnen, welche jeweils zwischen den letzten und den ersten Elementen der paarenden Stränge realisiert werden. Um die Stärken mit angeben zu können, sei im Folgenden stets für alle $e \in V(G) \times V(G) \supseteq E_U \cup E_H \cup E_S$

$$\tilde{S}_e := \begin{cases} U_e, & \text{wenn } e \in E_U, \\ H_e, & \text{wenn } e \in E_H, \\ S_e, & \text{wenn } e \in E_S, \\ 0, & \text{sonst.} \end{cases}$$

Folgerung 3.6 Mit $P = P_\varepsilon(g, \tilde{S}, L)$ für $\varepsilon > 0$, $L = (l\mathbb{1}_{\frac{n}{k}, \frac{m}{l}, 4k}, k\mathbb{1}_{\frac{n}{k}, \frac{m}{l}, 4l}) \in \mathbb{N}^{\frac{n}{k} \times \frac{m}{l} \times (4k+4l)}$, $\tilde{S} \in \mathbb{R}^{(\frac{n}{k} \times \frac{m}{l} \times (4k+4l))^2}$ und $g = (g^0, g^1) \in \mathbb{R}^{\frac{n}{k} \times \frac{m}{l} \times (4k+4l)}$ gilt

$$\begin{aligned} & \min \left\{ c_{n,m,k,l}(x) : x \in \{0, 1\}^{\frac{n}{k} \times \frac{m}{l} \times (4k+4l)} \right\} \\ &= \min \left\{ \sum_{r=1}^{\frac{n}{k}} \sum_{s=1}^{\frac{m}{l}} \left(\sum_{j=1}^{4k} \left((g_{0,j}^{r,s} + (l-1)P) \sum_{u=1}^l q_{I_{0,j,u}^{r,s}} + (-g_{0,j}^{r,s} - lP) \sum_{u=1}^{l-1} q_{I_{0,j,u}^{r,s}} q_{I_{0,j,u+1}^{r,s}} \right) \right. \right. \\ & \quad + \sum_{j=1}^{4l} \left((g_{1,j}^{r,s} + (k-1)P) \sum_{u=1}^k q_{I_{1,j,u}^{r,s}} + (-g_{1,j}^{r,s} - kP) \sum_{u=1}^{k-1} q_{I_{1,j,u}^{r,s}} q_{I_{1,j,u+1}^{r,s}} \right) \\ & \quad \left. + \sum_{j=1}^{4k} \sum_{t=1}^{4l} U_{j,t}^{r,s} q_{I_{0,j, \lceil \frac{t}{4} \rceil}^{r,s}} q_{I_{1,t, \lceil \frac{j}{4} \rceil}^{r,s}} \right) \\ & \quad \left. + \sum_{r=1}^{\frac{n}{k}} \sum_{s=1}^{\frac{m}{l}-1} \sum_{j=1}^{4k} H_j^{r,s} q_{I_{0,j,l}^{r,s}} q_{I_{0,j,1}^{r,s+1}} + \sum_{r=1}^{\frac{n}{k}-1} \sum_{s=1}^{\frac{m}{l}} \sum_{j=1}^{4l} S_j^{r,s} q_{I_{1,j,k}^{r,s}} q_{I_{1,j,1}^{r+1,s}} : q \in \{0, 1\}^{n \times m \times 2 \times 4} \right\}. \end{aligned}$$

Die Funktionen $\check{\sigma}$ und $\hat{\sigma}$, welche sich in P verstecken, werden analog zu vorher verwendet und beinhalten über \tilde{S} nun ebenso die Stärken der horizontalen und vertikalen Kopplungen. So ist beispielsweise für einen Knoten $(r, s, 0, j)$ mit $r \in \{1, \dots, \frac{n}{k}\}$, $s \in \{2, \dots, \frac{m}{l} - 1\}$ und $j \in \{1, \dots, 4k\}$

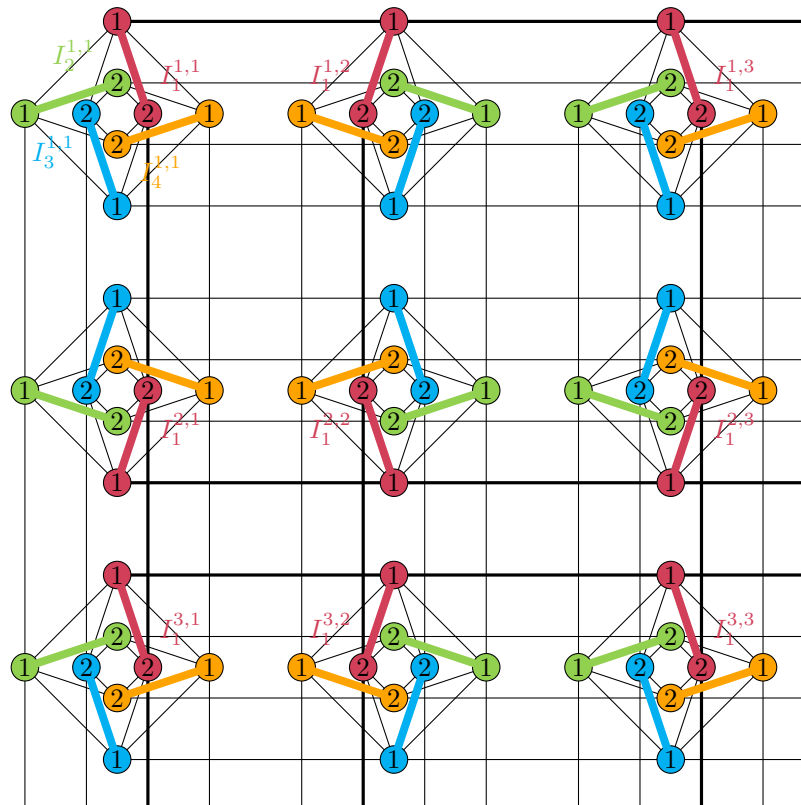
$$\check{\sigma}(r, s, 0, j) = \sum_{t=1}^{4l} \frac{1}{2} (1 - \text{sgn}(U_{j,t}^{r,s})) U_{j,t}^{r,s} + \sum_{p=0}^1 \frac{1}{2} (1 - \text{sgn}(H_{j-p}^{r,s})) H_{j-p}^{r,s}.$$

3.3.2 Symmetrische Kontraktionen

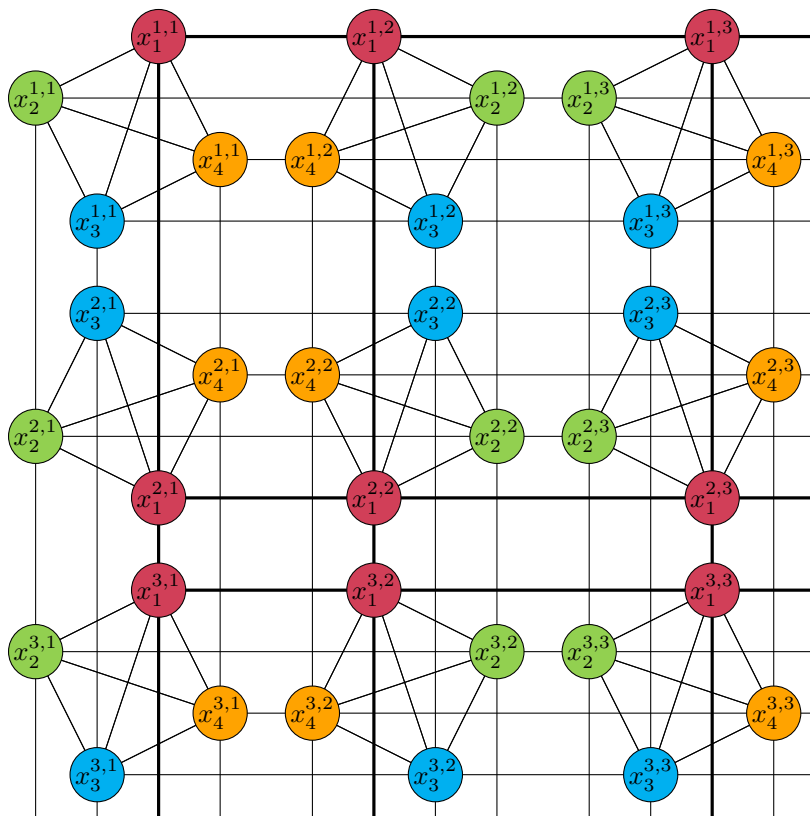
Die folgenden Minoren orientieren sich an der Struktur des Chimera-Graphen. Daher bauen sie sich immer gitterartig auf und sind stark symmetrisch. Auf den nächsten Seiten werden dazu Ausschnitte aus den Graphen zu sehen sein. Sie begrenzen sich meist auf die Knoten, die durch Kontraktion benachbarter Qubits in drei mal drei Einheitszellen des ursprünglichen Chimera-Graphen entstanden sind. Dabei zeigen die Farben immer Knoten oder Knotenstränge mit gleicher Indizierung an. Für die bessere Darstellbarkeit der Graphen wurde daher jede zweiten Spalte und Zeile gespiegelt. Dies ist möglich aufgrund der Symmetrie des Chimera-Graphen und der gebildeten Minoren.

Erste Kontraktion

Ein naheliegender Schritt zum Bilden eines Minoren des Chimera-Graphen ist die Verschmelzung je zweier Qubits aus unterschiedlichen Partitionen in jeder Einheitszelle. Dies ist in Abbildung 19(a) zu sehen. Der resultierende Graph wird mit C' bezeichnet und in 19(b) abgebildet. Entstand er aus n mal m Einheitszellen, wird er mit $C'_{n,m}$ bezeichnet. Dabei werden in jeder Einheitszelle $(r, s) \in \{1, \dots, n\} \times \{1, \dots, m\}$ vier logische Knoten $j = 1, 2, 3, 4$ gebildet, welche den Farben rot, grün, blau und orange entsprechen. Damit ist $V(C'_{n,m}) = \{1, \dots, n\} \times \{1, \dots, m\} \times \{1, \dots, 4\}$. Die Knoten in der Zelle sind nicht nur untereinander alle verbunden, sondern erhalten sowohl horizontale als auch vertikale Kopplungen zu angrenzenden Zellen, immer zu Knoten mit dem gleichen



(a) Qubitstränge im Chimera-Graphen C zu C'



(b) Minor C'

Abb. 19: Erste Kontaktktion

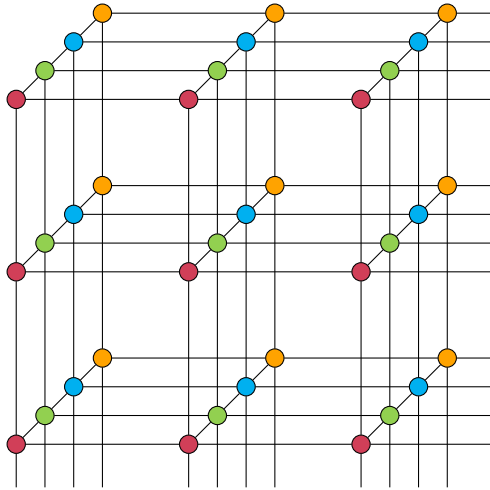


Abb. 20: Dreidimensionales Gitter $L_{n,m,4}$ ist Teilgraph von C'

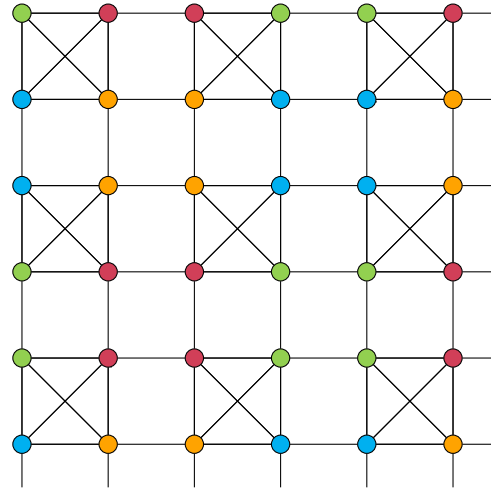


Abb. 21: $L_{2n,2m}$ ist Teilgraph von C'

Index $j \in \{1, \dots, 4\}$. Damit erhöht sich der Ausgangsgrad der verbliebenen Knoten auf 7. Für die Kantenmenge ergibt sich $E(C'_{n,m}) = E_U \cup E_H \cup E_S$ mit

$$\begin{aligned} E_U &= \{(r, s, j), (r, s, t)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m\}, j < t \in \{1, \dots, 4\}, \\ E_H &= \{(r, s, j), (r, s + 1, j)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m - 1\}, j \in \{1, \dots, 4\}, \\ E_S &= \{(r, s, j), (r + 1, s, j)\} : r \in \{1, \dots, n - 1\}, s \in \{1, \dots, m\}, j \in \{1, \dots, 4\}. \end{aligned}$$

Damit besitzt der Graph $6nm$ Kanten innerhalb der Zellen. Zusammen mit den erhalten gebliebenen $4n(m - 1)$ horizontalen und $4m(n - 1)$ vertikalen Kopplungen ergibt dies insgesamt $14nm - 4(n + m)$ Kopplungen zwischen den $4nm$ Knoten.

Der Graph C' besitzt eine starke Symmetrie: Zwei aneinandergrenzende Zeilen oder Spalten sind zueinander spiegelverkehrt. Dadurch wiederholt sich das Muster jede zweite Zeile oder Spalte. Dies ist natürlich abhängig von der Darstellung. So könnten die Zeilen oder Spalten auch als alle gleich gezeigt werden, indem jede zweite Zeile und Spalte gespiegelt würde. Jedoch hilft diese Darstellung, die weiteren Schritte deutlich zu machen. Die nm Knoten von $C'_{n,m}$, die eine gemeinsame Farbe besitzen, induzieren jeweils einen Gittergraphen der Größe n mal m . Beschränkt man die Kanten in jeder Zelle auf denselben Pfad entlang aller Knoten, beispielsweise nach den aufsteigenden Indizes, und zieht die Gitter daran in eine dritte Dimension auseinander, erhält man eine dreidimensionale Gitterstruktur der Größe $n \times m \times 4$, welche in Abbildung 20 dargestellt ist.

Durch Weglassen von einigen horizontalen und vertikalen Kanten in C' zwischen den Zellen entsteht ein Teilgraph, wie er in Abbildung 21 gezeigt ist. In diesem erkennt man deutlich eine weitere Gitterstruktur, die alle Knoten umfasst. Es zeigt sich, dass in $C'_{n,m}$ der Gittergraph $L_{2n,2m}$ eingebettet werden kann. Dieser enthält genau $(2n - 1)(2m - 1) = 4mn - 2n - 2m + 1$ Gittermaschen. Es kann festgestellt werden:

Theorem 3.7 Der Gittergraph $L_{n,m}$ mit $n, m > 2$ benötigt einen Chimera-Graphen mit einer Qubitzahl von cnm mit einer Konstanten $c > 1$.

Beweis: Da $L_{3,3}$ in $L_{n,m}$ mit $n, m > 2$ immer mindestens einmal enthalten ist, wird die Aussage zunächst für diesen Teilgraphen gezeigt. $L_{3,3}$ besitzt genau einen inneren Knoten mit vier Nachbarknoten. Je ein horizontaler und ein vertikaler Nachbarknoten dieses inneren Knotens besitzen einen gemeinsamen Nachbarn, wobei diese paarweise verschieden sind. Unter den maximal sechs Nachbarn eines Qubits lassen sich keine Paare von Qubits finden, die eine solche Eigenschaft erfüllen. Zwei der Kopplungen führen zu angrenzenden Einheitszellen, die anderen vier zu den Knoten der anderen Partition in der Einheitszelle, welche nur noch drei weitere gemeinsame Nachbarn in der Einheitszelle und keine weiteren außerhalb besitzen. Somit wird für die Einbettung von $L_{3,3}$ für mindestens einen Knoten mindestens ein weiteres Qubit benötigt.

Im Gittergraphen $L_{n,m}$ mit $n, m > 2$ lassen sich immer $\lfloor \frac{n}{3} \rfloor \lfloor \frac{m}{3} \rfloor \geq 1$ disjunkte $L_{3,3}$ -Teilgraphen finden. Bleibt zu zeigen, dass

$$nm + \lfloor \frac{n}{3} \rfloor \lfloor \frac{m}{3} \rfloor \geq cnm.$$

Dies gilt genau dann, wenn

$$\frac{\lfloor \frac{n}{3} \rfloor \lfloor \frac{m}{3} \rfloor}{nm} \geq \tilde{c}$$

für ein $\tilde{c} > 0$ ist. Damit kann $c = 1 + \tilde{c}$ gewählt werden. Wegen $z \bmod 3 \in \{0, 1, 2\}$ ist für $z \geq 3$

$$\frac{\lfloor \frac{z}{3} \rfloor}{z} = \frac{z - (z \bmod 3)}{3z} \geq \frac{1}{3} - \frac{2}{3z} \geq \frac{1}{9},$$

wodurch $\tilde{c} \geq \frac{1}{81}$ für $n, m > 2$ gilt. \square

Für symmetrische Kontraktionen mit $c \in \mathbb{N}$ ist die gezeigte Einbettung eines Gittergraphen damit sogar bestmöglich.

Das entsprechende QUBO zum Graphen $C'_{n,m}$ besitzt mit den binären Variablen $x_j^{r,s} \in \{0, 1\}$ zu den Knoten $(r, s, j) \in \{1, \dots, n\} \times \{1, \dots, m\} \times \{1, \dots, 4\}$ analog zu (XVI) die Zielfunktion

$$\bar{c}_{n,m}(x) = \bar{c}_{n,m}^g(x) + \bar{c}_{n,m}^U(x) + \bar{c}_{n,m}^H(x) + \bar{c}_{n,m}^S(x) \quad (\text{XXV})$$

mit

$$\begin{aligned} \bar{c}_{n,m}^g(x) + \bar{c}_{n,m}^U(x) &= \sum_{r=1}^n \sum_{s=1}^m \left(\sum_{j=1}^4 g_j^{r,s} x_j^{r,s} + \sum_{\substack{j,k=1 \\ k>j}}^4 U_{j,k}^{r,s} x_j^{r,s} x_k^{r,s} \right), \\ \bar{c}_{n,m}^H(x) &= \sum_{r=1}^n \sum_{s=1}^{m-1} \sum_{j=1}^4 H_j^{r,s} x_j^{r,s} x_j^{r,s+1}, \\ \bar{c}_{n,m}^S(x) &= \sum_{r=1}^{n-1} \sum_{s=1}^m \sum_{j=1}^4 S_j^{r,s} x_j^{r,s} x_j^{r+1,s}, \end{aligned}$$

wobei die $g \in \mathbb{R}^{n \times m \times 4}$ die Gewichte der Knoten und die $U \in \mathbb{R}^{n \times m \times 4 \times 4}$ die Stärken der Kopplungen innerhalb einer einzelnen Einheitszelle darstellen und $H \in \mathbb{R}^{n \times (m-1) \times 4}$ und $S \in \mathbb{R}^{(n-1) \times m \times 4}$ die Stärken der horizontalen und vertikalen Kopplungen.

Die zweielementigen Indexmengen der Qubitstränge zu dem logischen Knoten (r, s, j)

können nach Abbildung 19(a) mit

$$I_j^{r,s} = ((r, s, 0, j), (r, s, 1, \tilde{j}))$$

mit den Knotenkombinationen $(j, \tilde{j}) \in \{(1, 3), (2, 1), (4, 2), (3, 4)\}$ angegeben werden. Jedoch kann im Folgenden aufgrund der Symmetrie des Chimera-Graphen auch innerhalb der Einheitszellen von den Kombinationen (j, j) mit $j = 1, \dots, 4$ ausgegangen werden. Dabei bleibt die Belegung der Farben wie vorher erhalten. Die Qubits, die zur Kopplung der logischen Knoten $j, l \in \{1, \dots, 4\}$ innerhalb der entstandenen Zelle (r, s) beitragen, sind damit $I_{j,1}^{r,s}$ und $I_{l,2}^{r,s}$ für $j < l$ oder umgekehrt. Die horizontalen Kopplungen zwischen zwei Zellen betreffen immer die ersten Qubits im Strang und die vertikalen die zweiten. Damit gilt nun nach Abschnitt 3.2.3 für die Qubitstränge der Länge zwei:

Folgerung 3.8 Mit $P > \max\{0, -g_v - \tilde{\sigma}(v) : v \in V(C'_{n,m})\}$ für $g \in \mathbb{R}^{n \times m \times 4}$ und $\tilde{S} \in \mathbb{R}^{(n \times m \times 4)^2}$ gilt

$$\begin{aligned} & \min \left\{ \bar{c}_{n,m}(x) : x \in \{0, 1\}^{n \times m \times 4} \right\} \\ &= \min \left\{ \sum_{r=1}^n \sum_{s=1}^m \sum_{j=1}^4 \left((g_j^{r,s} + P)(q_{I_{j,1}^{r,s}} + q_{I_{j,2}^{r,s}}) + (-g_j^{r,s} - 2P)q_{I_{j,1}^{r,s}}q_{I_{j,2}^{r,s}} \right) \right. \\ & \quad + \sum_{r=1}^n \sum_{s=1}^{m-1} \sum_{j=1}^4 H_j^{r,s} q_{I_{j,1}^{r,s}} q_{I_{j,1}^{r,s+1}} + \sum_{r=1}^{n-1} \sum_{s=1}^m \sum_{j=1}^4 S_j^{r,s} q_{I_{j,2}^{r,s}} q_{I_{j,2}^{r+1,s}} \\ & \quad \left. + \sum_{r=1}^n \sum_{s=1}^m \sum_{\substack{j,l=1 \\ l>j}}^4 U_{j,l}^{r,s} q_{I_{j,1}^{r,s}} q_{I_{l,2}^{r,s}} : q \in \{0, 1\}^{n \times m \times 4 \times 2} \right\} \\ &= \min \left\{ \sum_{r=1}^n \sum_{s=1}^m \sum_{j=1}^4 \left((g_j^{r,s} + P)(q_{0,j}^{r,s} + q_{1,j}^{r,s}) + (-g_j^{r,s} - 2P)q_{0,j}^{r,s}q_{1,j}^{r,s} \right) \right. \\ & \quad + \sum_{r=1}^n \sum_{s=1}^{m-1} \sum_{j=1}^4 H_j^{r,s} q_{0,j}^{r,s} q_{0,j}^{r,s+1} + \sum_{r=1}^{n-1} \sum_{s=1}^m \sum_{j=1}^4 S_j^{r,s} q_{1,j}^{r,s} q_{1,j}^{r+1,s} \\ & \quad \left. + \sum_{r=1}^n \sum_{s=1}^m \sum_{\substack{j,l=1 \\ l>j}}^4 U_{j,l}^{r,s} q_{0,j}^{r,s} q_{1,l}^{r,s} : q \in \{0, 1\}^{n \times m \times 4 \times 2} \right\}. \end{aligned}$$

Des Weiteren kann C' aber auch folgendermaßen charakterisiert werden: Die vier Knoten aus jeder Zelle stellen je eine Clique dar. Diese vollständigen Teilgraphen K_4 sind an Kreuzungspunkten eines Gitters angeordnet, wobei statt nur einer Kante im Gitter vier Kanten vorhanden sind. Diese verbinden nur je einen Knoten einer Zelle mit seinem Äquivalent, also dem entsprechenden Knoten mit der gleichen Nummer in der gleichen Partition, aus der anderen Zelle. Wird C' auf n Zeilen und m Spalten begrenzt, erhält man ein Gitter mit n mal m K_4 's. Daher sei für $C'_{n,m}$ die Notation $L_{n,m}[K_4]$ eingeführt. Diese Struktur ist sogar erweiterbar auf größere Cliques: Dazu bildet man zunächst in zwei mal zwei Einheitszellen einen vollständigen Graphen nach der in Abschnitt 3.2.2 und in Abbildung 15 gezeigten Vorgehensweise. Diese Einbettung erweitert man über die noch freie Einheitszelle hinweg, wie in Abbildung 22 gezeigt ist. Dadurch entsteht

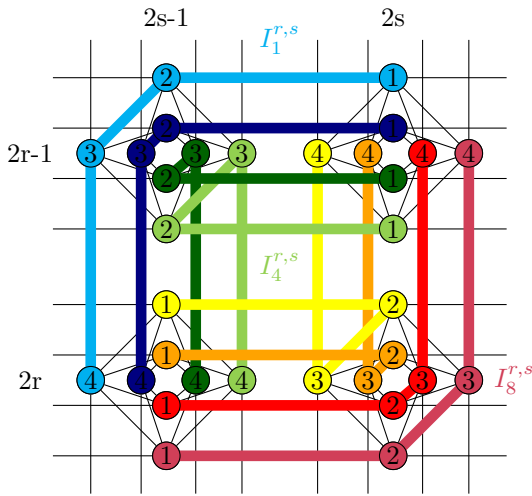


Abb. 22: K_8 erweitert

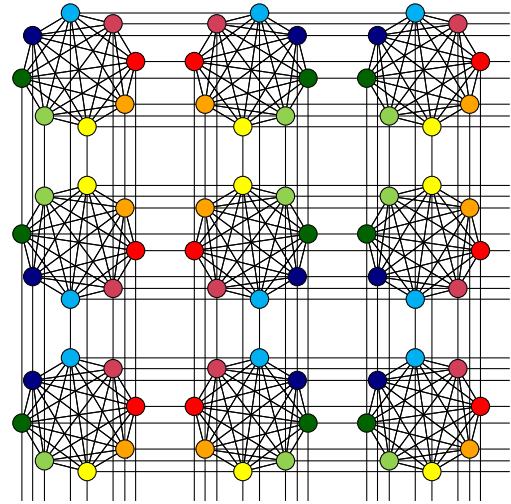


Abb. 23: Ausschnitt $L_{3,3}[K_8]$
(aus 6×6 Einheitszellen)

eine neue Zelle mit einem vollständigen Graphen über acht Knoten $j = 1, \dots, 8$, die alle jeweils zwei horizontale und zwei vertikale Kopplungen in angrenzende Zellen besitzen. Das Prinzip wird nun schachbrettartig über ganz C fortgeführt. Der dabei entstandene Graph besitzt starke Ähnlichkeit zu C' , nur befinden sich nun K_8 's an Kreuzungspunkten eines Gitters und die Kanten des Gitters entsprechen den acht Kopplungen zwischen den einzelnen Knoten. Sind n und m gerade, erhält man so in $C_{n,m}$ genau $\frac{n}{2}$ mal $\frac{m}{2}$ vollständige Graphen. Daher sei dieser Graph mit $L_{\frac{n}{2}, \frac{m}{2}}[K_8]$ bezeichnet. Er besitzt damit nur noch $2nm$ Knoten und $7nm$ Kopplungen in der Zelle sowie $4n(\frac{m}{2} - 1)$ horizontale und $4m(\frac{n}{2} - 1)$ vertikale Kopplungen zwischen den Zellen, also insgesamt $11nm - 4(n + m)$ Kopplungen. Ein Ausschnitt eines solchen Graphen mit je drei Zeilen und Spalten ist in Abbildung 23 gezeigt. Dieser ist aus sechs mal sechs Einheitszellen des Chimera-Graphen hervorgegangen.

Der Graph $L_{\frac{n}{2}, \frac{m}{2}}[K_8]$ mit geraden n und m kann dabei wie vorher C' ebenso zu einem dreidimensionalen Gitter der Größe $\frac{n}{2} \times \frac{m}{2} \times 8$ auseinander gezogen werden. Die Zellen des Graphen erhalten dabei die Indizes $(r, s) \in \{1, \dots, \frac{n}{2}\} \times \{1, \dots, \frac{m}{2}\}$. Diese entsprechen genau den halben Indizes der ursprünglichen Einheitszelle des Chimera-Graphen in der rechten unteren Ecke. Die Zielfunktion zu diesem Graphen kann analog zu (XXV) aufgestellt werden, nur erhöht sich die Knotenzahl mit Index j der $x_j^{r,s}$ auf acht und die Summation über $j = 1, \dots, 4$ muss ersetzt werden durch $j = 1, \dots, 8$. Daher sei diese Funktion mit $\bar{c}_{\frac{n}{2}, \frac{m}{2}, 8} : \{0, 1\}^{\frac{n}{2} \times \frac{m}{2} \times 8} \rightarrow \mathbb{R}$ bezeichnet. Sie besitzt die Parameter $g \in \mathbb{R}^{\frac{n}{2} \times \frac{m}{2} \times 8}$, $U \in \mathbb{R}^{\frac{n}{2} \times \frac{m}{2} \times 8 \times 8}$, $H \in \mathbb{R}^{\frac{n}{2} \times (\frac{m}{2} - 1) \times 8}$ und $S \in \mathbb{R}^{(\frac{n}{2} - 1) \times \frac{m}{2} \times 8}$.

Die Definition der Indexmengen der acht Qubitstränge der vollständigen Graphen wird analog zu I_j aus (XVIII) und (XIX) des Abschnitts 3.2.2 übernommen. Jedoch müssen zwei Ergänzungen vorgenommen werden: Einerseits muss in jedem Qubitstrang je ein Tupel hinzugefügt werden: $(1, 2, 0, j)$ für die ersten vier Stränge, also mit Knotenindex $j \leq 4$, an deren Anfang und $(1, 2, 1, j - 4)$ für die zweiten vier Stränge, mit $j \geq 5$, an deren Ende. Andererseits müssen die ersten beiden Indizes der Qubits in den Strängen durch die Verschiebung über die Einheitszellen im Chimera-Graphen geändert werden. Mit komponentenweiser Addition gilt für das k -te Qubit des j -ten Strangs in der Zelle

an Position (r, s) , dass

$$I_{j,k}^{r,s} = I_{j,k} + (2r - 1, 2s - 1, 0, 0).$$

Damit folgt für den kompletten vierelementigen Strang

$$I_{j,*}^{r,s} = ((2r - 1, 2s, 0, j), (2r - 1, 2s - 1, 0, j), (2r - 1, 2s - 1, 1, j), (2r, 2s - 1, 1, j))$$

für $j \leq 4$ und

$$I_{j,*}^{r,s} = ((2r, 2s - 1, 0, j - 4), (2r, 2s, 0, j - 4), (2r, 2s, 1, j - 4), (2r - 1, 2s, 1, j - 4))$$

für $j \geq 4$. Nach (XX) besitzen die Qubits, die zwei Stränge j und l in einer Zelle (r, s) verbinden, analog die Indizes $O_{j,l}^{s,t}$ und $O_{l,j}^{s,t}$, also für o.B.d.A. $j < l$ mit

$$O_{j,l}^{s,t} = I_{j, \lceil \frac{l}{4} \rceil}^{r,s}, \quad O_{l,j}^{s,t} = I_{l, \lceil \frac{j}{4} \rceil + 1}^{r,s}.$$

Die j -ten Qubitstränge zweier entstandener, horizontal benachbarter Zellen an (r, s) und $(r, s+1)$ sind damit über die Qubits $q_{I_{j,1}^{r,s}}$ und $q_{I_{j,2}^{r,s+1}}$ für $j \leq 4$ und $q_{I_{j,2}^{r,s}}$ und $q_{I_{j,1}^{r,s+1}}$ für $j \geq 4$ gekoppelt. In zwei vertikal benachbarten Zellen an (r, s) und $(r+1, s)$ verbinden die Qubits $q_{I_{j,4}^{r,s}}$ und $q_{I_{j,3}^{r+1,s}}$ für $j \leq 4$ und $q_{I_{j,3}^{r,s}}$ und $q_{I_{j,4}^{r+1,s}}$ für $j \geq 4$ die Stränge.

Folgerung 3.9 Mit $P = P_\varepsilon(g, \tilde{S}, L)$ für $\varepsilon > 0$, $g \in \mathbb{R}^{\frac{n}{2} \times \frac{m}{2} \times 8}$, $\tilde{S} \in \mathbb{R}^{(\frac{n}{2} \times \frac{m}{2} \times 8)^2}$ und $L = 4\mathbb{1}_{\frac{n}{2}, \frac{m}{2}, 8} \in \mathbb{N}^{\frac{n}{2} \times \frac{m}{2} \times 8}$ gilt

$$\begin{aligned} & \min \left\{ \bar{c}_{\frac{n}{2}, \frac{m}{2}, 8}(x) : x \in \{0, 1\}^{\frac{n}{2} \times \frac{m}{2} \times 8} \right\} \\ &= \min \left\{ \sum_{r=1}^{\frac{n}{2}} \sum_{s=1}^{\frac{m}{2}} \sum_{j=1}^8 \left((g_j^{r,s} + 3P) \sum_{k=1}^4 q_{I_{j,k}^{r,s}} + (-g_j^{r,s} - 4P) \sum_{k=1}^3 q_{I_{j,k}^{r,s}} q_{I_{j,k+1}^{r,s}} \right) \right. \\ & \quad + \sum_{r=1}^{\frac{n}{2}} \sum_{s=1}^{\frac{m}{2}-1} \left(\sum_{j=1}^4 H_j^{r,s} q_{I_{j,1}^{r,s}} q_{I_{j,2}^{r,s+1}} + \sum_{j=5}^8 H_j^{r,s} q_{I_{j,2}^{r,s}} q_{I_{j,1}^{r,s+1}} \right) \\ & \quad + \sum_{r=1}^{\frac{n}{2}-1} \sum_{s=1}^{\frac{m}{2}} \left(\sum_{j=1}^4 S_j^{r,s} q_{I_{j,4}^{r,s}} q_{I_{j,3}^{r+1,s}} + \sum_{j=5}^8 S_j^{r,s} q_{I_{j,3}^{r,s}} q_{I_{j,4}^{r+1,s}} \right) \\ & \quad \left. + \sum_{r=1}^{\frac{n}{2}} \sum_{s=1}^{\frac{m}{2}} \sum_{\substack{j,l=1 \\ l>j}}^8 U_{j,l}^{r,s} q_{O_{j,l}^{r,s}} q_{O_{l,j}^{r,s}} : q \in \{0, 1\}^{n \times m \times 2 \times 4} \right\}. \end{aligned}$$

Fortführend sind analoge Graphen mit größeren vollständigen Teilgraphen auf $4l$ Knoten mit $l \in \mathbb{N}$ nach dem gleichen Prinzip konstruierbar. Allgemein gilt, ist k Teiler von n und m , kann aus $L_{n,m}[K_{4l}]$ durch Zusammenfassen von k mal k K_{4l} -Graphen im Gitter der Graph $L_{\frac{n}{k}, \frac{m}{k}}[K_{4lk}]$ gebildet werden. Dieser besitzt mit $V(L_{\frac{n}{k}, \frac{m}{k}}[K_{4lk}]) = \{1, \dots, \frac{n}{k}\} \times \{1, \dots, \frac{m}{k}\} \times \{1, \dots, 4lk\}$ nur noch $4l^{\frac{nm}{k}}$ Knoten und mit der Kantenmenge $E(C'_{n,m}) = E_U \cup E_H \cup E_S$ und

$$\begin{aligned} E_U &= \{ \{(r, s, j), (r, s, t)\} : r \in \{1, \dots, \frac{n}{k}\}, s \in \{1, \dots, \frac{m}{k}\}, j, t \in \{1, \dots, 4lk\} \}, \\ E_H &= \{ \{(r, s, j), (r, s+1, j)\} : r \in \{1, \dots, \frac{n}{k}\}, s \in \{1, \dots, \frac{m}{k} - 1\}, j \in \{1, \dots, 4lk\} \}, \\ E_S &= \{ \{(r, s, j), (r+1, s, j)\} : r \in \{1, \dots, \frac{n}{k} - 1\}, s \in \{1, \dots, \frac{m}{k}\}, j \in \{1, \dots, 4lk\} \}. \end{aligned}$$

genau $2l \frac{nm}{k} (4kl - 1)$ Kanten innerhalb der Zelle und $4ln(\frac{m}{k} - 1)$ horizontale und $4lm(\frac{n}{k} - 1)$ vertikale Kopplungen. Dies ergibt insgesamt $8l^2nm + 6l \frac{nm}{k} - 4l(n + m)$.

Die zugehörigen Zielfunktionen von QUBOs über dem Graphen $L_{n,m}[K_{4l}]$ für $l \in \mathbb{N}$ können wie vorher $\bar{c}_{n,m,8}$ über $\bar{c}_{n,m,4l} : \{0,1\}^{n \times m \times 4l} \rightarrow \mathbb{R}$ mit den Parametern $g \in \mathbb{R}^{n \times m \times 4l}$, $U \in \mathbb{R}^{n \times m \times 4l \times 4l}$, $H \in \mathbb{R}^{n \times (m-1) \times 4l}$ und $S \in \mathbb{R}^{(n-1) \times m \times 4l}$ angegeben werden. Bei der Erweiterung der Einbettung dieser vollständigen Graphen in den Chimera-Graphen muss jedoch Folgendes beachtet werden: Die Spiegelung der Stränge in die andere Ecke der Einheitszellen führt bei den mittleren Einheitszellen dazu, dass die Qubitteilmenen die zu einem logischen Knoten zusammengefasst werden, keine Stränge mehr sind, sondern eine Art X-Form besitzen. Dementsprechend müsste die Struktur der Indextmenen angepasst werden. Jedoch sind die Teilgraphen minimal zusammenhängend und daher kann die Verteilung der Gewichte und Stärken wie vorher erfolgen.

Zweite Kontraktion

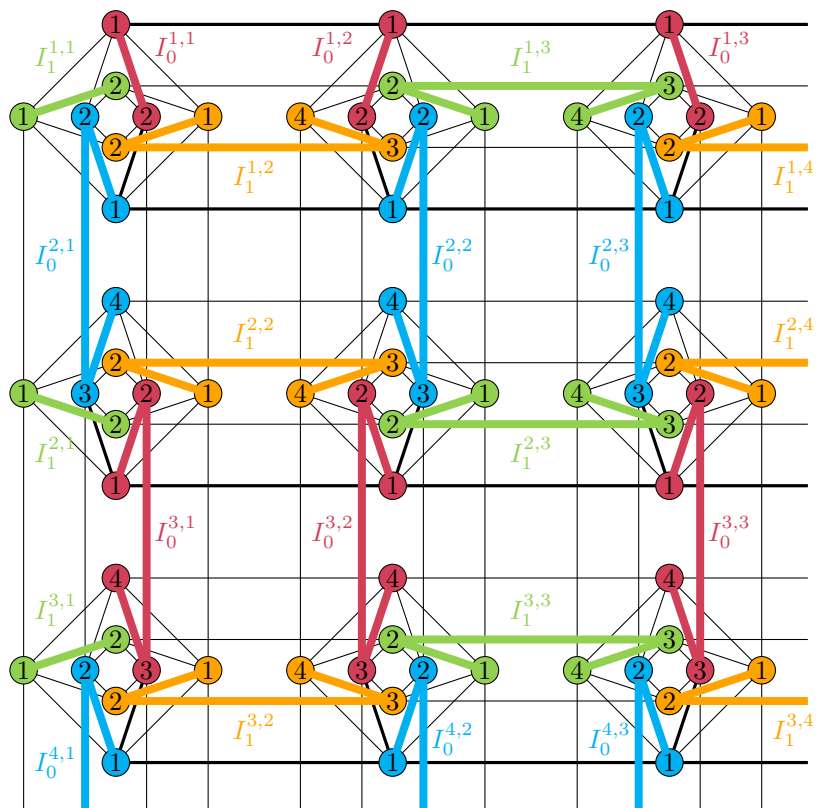
Durch Fortführen der symmetrischen Kontraktionen in C' kann der Graph C'' gebildet werden, welcher in Abbildung 24(b) dargestellt ist. Bei diesem wurden, ausgehend von C' , je zwei Knoten verschmolzen, die aus benachbarten Zellen stammen. In einer Zelle wird dabei jedem Knoten je eine der links, rechts, oben oder unten angrenzenden Einheitszellen zugeordnet. Entlang der Grenze zwischen zwei Zeilen oder Spalten betrifft dies immer Knoten der gleichen Farbe. Das heißt, formal wird ein Knoten (r, s, j) des vorigen Abschnittes mit $j \in \{1, \dots, 4\}$, $r \in \{1, \dots, n\}$ und $s \in \{1, \dots, m\}$ verschmolzen mit

- $(j, r + 1, s)$ für $j = 1$, wenn r gerade ist,
- $(j, r + 1, s)$ für $j = 3$, wenn r ungerade ist,
- $(j, r, s + 1)$ für $j = 2$, wenn s gerade ist, oder
- $(j, r, s + 1)$ für $j = 4$, wenn s ungerade ist.

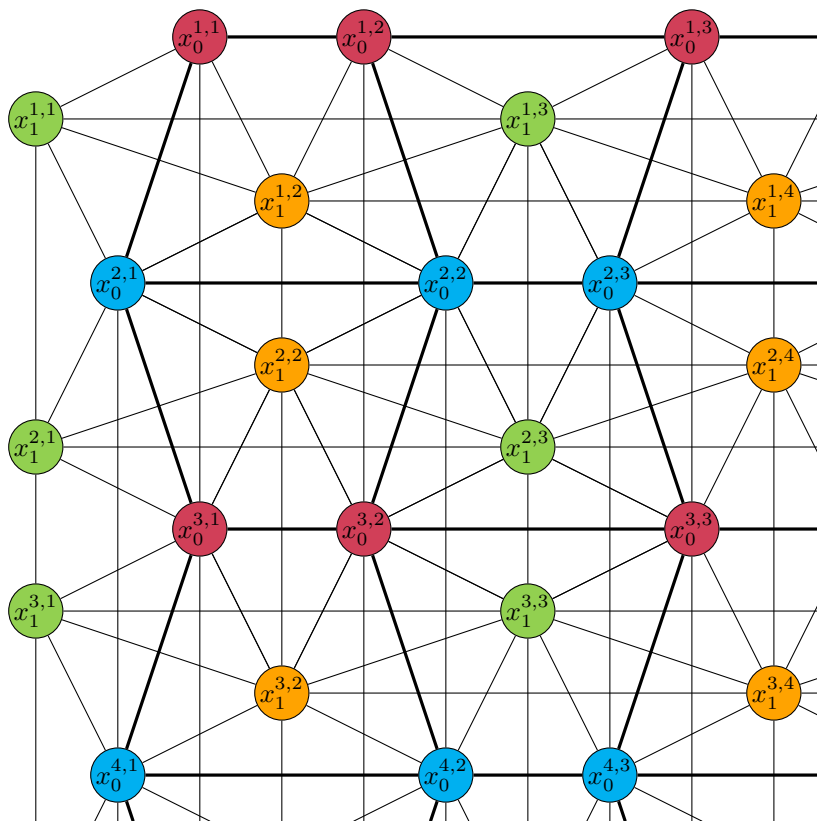
Dadurch ist ein ebenso symmetrischer, sich wiederholender Graph entstanden, dessen Knoten einen Ausgangsgrad von 10 besitzen. Nur die $2(n + m)$ Knoten am Rand von $C''_{n,m}$ sind nicht von einer Verschmelzung betroffen und verbleiben daher in $C''_{n,m}$ und behalten den Ausgangsgrad 6 beziehungsweise 5 in den Ecken. Auch ihre Qubitstränge im Chimera-Graphen besitzen noch die Länge 2. Die Zahl der restlichen inneren Knoten wird stattdessen halbiert auf $2nm - (n + m)$. Somit umfasst $C''_{n,m}$ genau $2nm + n + m$ Knoten.

Die Knoten von $C''_{n,m}$, deren Qubits senkrecht über die Zeilen der Einheitszellen hinweg verbunden wurden, im Bild mit rot und blau gekennzeichnet, und die anderen, horizontal über Spalten verlaufenden, Knoten sollen nun je einer Partition zugeordnet werden. Erstere induzieren einen Graphen, der ein Gitter der Größe $(n + 1)$ mal m enthält, und werden mit dem Partitionsindex $p = 0$ versehen. Letztere umfassen stattdessen $L_{n,m+1}$ und erhalten den Wert 1 für p . Da in diesem Graphen die Qubits über die Einheitszellen miteinander verschmolzen wurden und somit deren Struktur im entstandenen Graphen verloren gegangen ist, werden die Knoten nun anhand ihrer zugrundeliegenden Gitterstruktur indiziert und erhalten zum Partitionsindex jeweils einen Zeilen- und Spaltenindex anhand ihrer Position im Gitter der Partition. Damit ergeben sich die Knoten $(r,s,0)$ mit $r \in \{1, \dots, n + 1\}$ und $s \in \{1, \dots, m\}$ und $(r,s,1)$ mit $r \in \{1, \dots, n\}$ und $s \in \{1, \dots, m + 1\}$, also ist

$$V(C''_{n,m}) = V_0 \cup V_1 = \{1, \dots, n + 1\} \times \{1, \dots, m\} \times \{0\} \cup \{1, \dots, n\} \times \{1, \dots, m + 1\} \times \{1\}.$$



(a) Qubitstränge im Chimera-Graphen C zu C''



(b) Minor C''

Abb. 24: Zweite Kontaktion

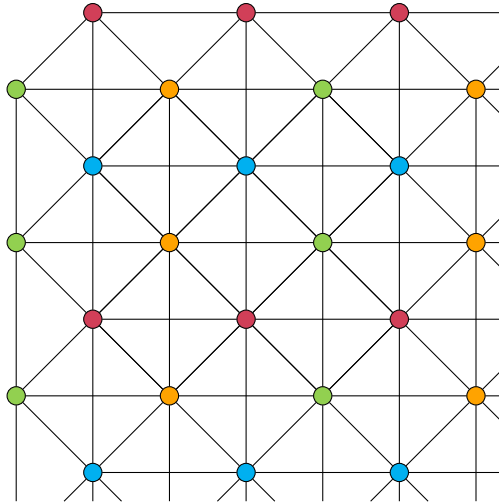


Abb. 25: Erweiterter Gittergraph \tilde{L} ist Teilgraph von C''

Deren Verteilung über den Graphen ist in Abbildung 24(b) mit dargestellt. Die Farbgebung und damit der Index der Qubits in der Partition einer Einheitszelle, ist zwar damit kein Teil der Indizierung mehr, lässt sich jedoch mit

$$j(r, s, p) = \begin{cases} 1 & \text{für } p = 0 \text{ und } r \text{ ungerade,} \\ 3 & \text{für } p = 0 \text{ und } r \text{ gerade,} \\ 2 & \text{für } p = 1 \text{ und } s \text{ ungerade,} \\ 4 & \text{für } p = 1 \text{ und } s \text{ gerade.} \end{cases} \quad (\text{XXVI})$$

wieder zurückgewinnen.

Auch zu C'' lässt sich ein Teilgraph finden, der eine Gitterstruktur besitzt. Dies ist in Abbildung 25 gezeigt. In diesem Fall ist das Gitter diagonal hineingesetzt, wodurch $C''_{n,m}$ nur $nm + (n-1)(m-1) = 2nm - n - m + 1$ Gittermaschen besitzt. Der gezeigte Graph soll im Folgenden mit $\tilde{L}_{n,m}$ bezeichnet werden, wenn er aus n mal m Einheitszellen hervorgegangen ist. Er unterscheidet sich nur durch den konstanten Faktor 2 in der Größenordnung vom Gittergraphen in $C''_{n,m}$. Im Gegensatz zu diesem existieren jedoch in $\tilde{L}_{n,m}$ in jeder Masche auch diagonale Kanten zwischen den Knoten. Dadurch kann dieser erweiterte Gittergraph eine größere Klasse an Problemen abdecken. Hier zeigt sich zudem deutlich, dass jeder Knoten, der sich nicht auf dem Rand befindet, in genau vier Cliques mit jeweils vier Knoten liegt, welche untereinander nicht mehr als zwei Knoten gemeinsam haben. Diese Eigenschaft kann in dem Beweis zu der folgenden Aussage genutzt werden:

Theorem 3.10 Die Einbettung von \tilde{L}_{nm} mit $n, m \geq 2$ in $C''_{n,m}$ ist bestmöglich. Das heißt, \tilde{L}_{nm} kann nicht in einen Chimera-Graphen eingebettet werden, der weniger als n Zeilen oder m Spalten hat.

Beweis: \tilde{L}_{nm} besitzt $n(m-1) + m(n-1)$ Knoten, der einziger gemeinsamer Knoten zweier Cliques ist. Für eine Clique mit vier Knoten benötigt man nach Abschnitt 3.2.2 mindestens sechs Qubits in einer Einheitszelle, da zwei der Knoten über je zwei Qubits dargestellt werden müssen. Damit werden für die zwei zusammenhän-

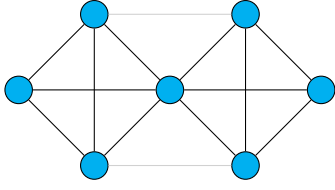


Abb. 26: Zwei Cliques mit gemeinsamen Knoten

genden Cliques mindestens 10 Qubits und so mindestens zwei Einheitszellen im Chimera-Graphen besetzt. Werden alle möglichen Belegungen der Qubits durchgegangen, kann festgestellt werden, dass der zentrale Knoten immer beide Einheitszellen überspannt, also mindestens ein Qubit seines Stranges in der einen und ein anderes in der anderen Einheitszelle liegt. Ebenso wird für die Hinzunahme einer weiteren Clique nach dem Muster wie in \tilde{L}_{nm} erneut eine weitere Einheitszelle mit einem überspannendem zentralen Knoten benötigt. Durch Fortführen dieses Aufbaus zeigt sich, dass für alle zentralen Knoten ein Chimera-Graph mit $n(m-1)$ Paaren horizontal benachbarter und $m(n-1)$ Paaren vertikal benachbarten Einheitszellen, also mit mindestens n Zeilen und m Spalten, benötigt wird. \square

Es zeigt sich, dass immer noch die vier Knoten mit unterschiedlichen Farben eine Clique bilden mit $(r, s, 0)$, $(r+1, s, 0)$, $(r, s, 1)$ und $(r, s+1, 1)$ für $r \in \{1, \dots, n\}$ und $s \in \{1, \dots, m\}$. Da sich jedoch die Farben über das Gitter hinweg spiegeln, seien die Indizes der Knoten in der Clique in den Indexmengen $J_j^{r,s} \in \{r, r+1\} \times \{s, s+1\} \times \{0, 1\}$ für $j = 1, \dots, 4$ entsprechend ihrer Farben kodiert. Die Indizes erhält man dann aus $J_j^{r,s}$ durch eine entsprechende Umkehrung von (XXVI). Damit ist $E(C''_{n,m}) = E_U \cup E_H \cup E_S$ mit

$$E_U = \{\{J_j^{r,s}, J_l^{r,s}\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m\}, j < l \in \{1, \dots, 4\}\}.$$

Zudem existieren für jeden Knoten horizontale und vertikale Kopplungen zu anderen Knoten mit der gleichen Farbe, wodurch weitere Vierercliques entstehen. Horizontal ergeben sich somit die Paarungen $(r, s, 0)$ mit $(r, s+1, 0)$ für $r = 1, \dots, n+1$ und $s = 1, \dots, m-1$ und $(r, s, 1)$ mit $(r, s+2, 1)$ für $r = 1, \dots, n$ und $s = 1, \dots, m-1$. Die senkrechten Kopplungen können analog aufgestellt werden und es folgt:

$$\begin{aligned} E_H &= \{\{(r, s, 0), (r, s+1, 0)\} : r \in \{1, \dots, n+1\}, s \in \{1, \dots, m-1\}\} \\ &\quad \cup \{\{(r, s, 1), (r, s+2, 1)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m-1\}\}, \\ E_S &= \{\{(r, s, 1), (r+1, s, 1)\} : r \in \{1, \dots, n-1\}, s \in \{1, \dots, m+1\}\} \\ &\quad \cup \{\{(r, s, 0), (r+2, s, 0)\} : r \in \{1, \dots, n-1\}, s \in \{1, \dots, m\}\}. \end{aligned}$$

Daraus ergeben sich insgesamt $10nm - m - n - 2$ Kopplungen in dem Graphen $C''_{n,m}$.

Die Zielfunktion des zu $C''_{n,m}$ gehörenden QUBOs über den dazugehörigen binären Variablen $x_0 \in \{0, 1\}^{(n+1) \times m}$ und $x_1 \in \{0, 1\}^{n \times (m+1)}$ ist

$$\bar{c}_{n,m}(x) = \bar{c}_{n,m}^g(x) + \bar{c}_{n,m}^U(x) + \bar{c}_{n,m}^H(x) + \bar{c}_{n,m}^S(x) \quad (\text{XXVII})$$

mit

$$\begin{aligned} \bar{c}_{n,m}^g(x) &= \sum_{r=1}^{n+1} \sum_{s=1}^m g_0^{r,s} x_0^{r,s} + \sum_{r=1}^n \sum_{s=1}^{m+1} g_1^{r,s} x_1^{r,s} = \sum_{p=0}^1 \sum_{r=1}^{n+1-p} \sum_{s=1}^{m+p} g_p^{r,s} x_p^{r,s}, \\ \bar{c}_{n,m}^U(x) &= \sum_{r=1}^n \sum_{s=1}^m \left(U_{j(r,s,0),j(r+1,s,0)}^{r,s} x_0^{r,s} x_0^{r+1,s} + U_{j(r,s,0),j(r,s,1)}^{r,s} x_0^{r,s} x_1^{r,s} + \dots \right) \end{aligned}$$

$$\begin{aligned}
 & + U_{j(r,s,1),j(r,s+1,1)}^{r,s} x_1^{r,s} x_1^{r,s+1} \Big) = \sum_{r=1}^n \sum_{s=1}^m \sum_{\substack{j,l=1 \\ j < l}}^4 U_{j,l}^{r,s} x_{j,l}^{r,s} x_{j,l}^{r,s}, \\
 \bar{c}_{n,m}^H(x) &= \sum_{r=1}^{n+1} \sum_{s=1}^{m-1} H_0^{r,s} x_0^{r,s} x_0^{r,s+1} + \sum_{r=1}^n \sum_{s=1}^{m-1} H_1^{r,s} x_1^{r,s} x_1^{r,s+2} = \sum_{p=0}^1 \sum_{r=1}^{n+1-p} \sum_{s=1}^{m-1} H_p^{r,s} x_p^{r,s} x_p^{r,s+1+p}, \\
 \bar{c}_{n,m}^S(x) &= \sum_{r=1}^{n-1} \sum_{s=1}^m S_0^{r,s} x_0^{r,s} x_0^{r+2,s} + \sum_{r=1}^{n-1} \sum_{s=1}^{m+1} S_1^{r,s} x_1^{r,s} x_1^{r+1,s} = \sum_{p=0}^1 \sum_{r=1}^{n-1} \sum_{s=1}^{m+p} S_p^{r,s} x_p^{r,s} x_p^{r+2-p,s}.
 \end{aligned}$$

Dabei sind $g_0 \in \mathbb{R}^{(n+1) \times m}$, $g_1 \in \mathbb{R}^{n \times (m+1)}$ und $U \in \mathbb{R}^{n \times m \times 4 \times 4}$, $H_0 \in \mathbb{R}^{(n+1) \times (m-1)}$, $H_1 \in \mathbb{R}^{n \times (m-1)}$, $S_0 \in \mathbb{R}^{(n-1) \times m}$, $S_1 \in \mathbb{R}^{(n-1) \times (m+1)}$.

Bei der Aufstellung der Indexmengen für die entsprechenden Qubitstränge muss beachtet werden, dass die Knoten, die auf dem Rand des Graphen $C''_{n,m}$ liegen, nur zwei und alle anderen Knoten vier Qubits umfassen. Damit ergibt sich die Indexmenge

$$I_p^{r,s} \in (\{1, \dots, n\} \times \{1, \dots, m\} \times \{0, 1\} \times \{1, \dots, 4\})^d$$

mit $d = 2$ bei einem Knoten (r, s, p) mit $r \in \{1, n+1\}$ und $s \in \{1, \dots, m\}$, wenn $p = 0$, oder $r \in \{1, \dots, n\}$ und $s \in \{1, m+1\}$, wenn $p = 1$, und mit $d = 4$ bei allen anderen Möglichkeiten für r und s . Bei den Strängen, die über zwei Einheitszellen verlaufen, wird mit der Indizierung in der linken beziehungsweise oberen Einheitszelle begonnen. In Abbildung 24(b) ist dies beispielhaft gezeigt. Es gilt

$$I_0^{r,s} = \begin{cases} ((r-1, s, 0, 3), (r-1, s, 1, 3), (r, s, 1, 3), (r, s, 0, 3)) & \text{für } r \text{ gerade,} \\ ((r-1, s, 0, 1), (r-1, s, 1, 1), (r, s, 1, 1), (r, s, 0, 1)) & \text{für } r \text{ ungerade,} \end{cases}$$

und

$$I_1^{r,s} = \begin{cases} ((r, s-1, 1, 4), (r, s-1, 0, 4), (r, s, 0, 4), (r, s, 1, 4)) & \text{für } s \text{ gerade,} \\ ((r, s-1, 1, 2), (r, s-1, 0, 2), (r, s, 0, 2), (r, s, 1, 2)) & \text{für } s \text{ ungerade} \end{cases}$$

mit $r \in \{2-p, \dots, n\}$ und $s \in \{1+p, \dots, m\}$. Damit entsprechen die angegebenen Stränge der Reihe nach den Farben blau, rot, orange und grün der Abbildung. Für die Knoten am linken und oberen Rand müssen dann nur die ersten beiden Elemente des entsprechenden Stranges entfernt und die beiden verbleibenden Elemente vertauscht werden, damit die ersten Elemente der Stränge für eine Gitterpartition auch in der gleichen Partition der Einheitszelle liegen. Für die Knoten am rechten oder unteren Rand genügt die Entfernung der letzten beiden Elemente. Mit $r \in \{1, \dots, n\}$ und $s \in \{1, \dots, m\}$ gilt

$$\begin{aligned}
 I_0^{1,s} &= ((1, s, 0, 1), (1, s, 1, 1)), \\
 I_1^{r,1} &= ((r, 1, 1, 2), (r, 1, 0, 2)), \\
 I_0^{n+1,s} &= \begin{cases} ((n, s, 0, 1), (n, s, 1, 1)) & \text{für } n \text{ gerade,} \\ ((n, s, 0, 3), (n, s, 1, 3)) & \text{für } n \text{ ungerade,} \end{cases} \\
 I_1^{r,m+1} &= \begin{cases} ((r, m, 1, 2), (r, m, 0, 2)) & \text{für } m \text{ gerade,} \\ ((r, m, 1, 4), (r, m, 0, 4)) & \text{für } m \text{ ungerade.} \end{cases}
 \end{aligned}$$

Es zeigt sich, dass für die horizontalen Kopplungen in Partition 0 und die vertikalen Kopplungen in Partition 1 zwischen Knoten gleicher Farbe immer die ersten Elemente der jeweiligen Stränge sorgen. In den umgekehrten Fällen, bei denen die horizontalen Kopplungen der $x_1^{r,s}$ einen Spaltenindex und die vertikalen Kopplungen der $x_0^{r,s}$ einen Zeilenindex überspringen, wird das dritte Element des ersten und das zweite Element des zweiten Qubitstrangs benötigt. Bei den Knoten des linken und des oberen Randes muss das zweite Element zur Kopplung benutzt werden. Des Weiteren treffen sich in einer Einheitszelle immer, abgesehen von den Einheitszellen am linken und oberen Rand, das Ende je eines horizontalen und vertikalen Strangs mit einem Anfang eines horizontalen und eines vertikalen. Darin ist kein sinnvolles Schema zu erkennen. Jedoch können die Indizes des gekoppelten Qubits in der Einheitszelle aufgrund der Zurückrechnung der Farbindices direkt angegeben werden.

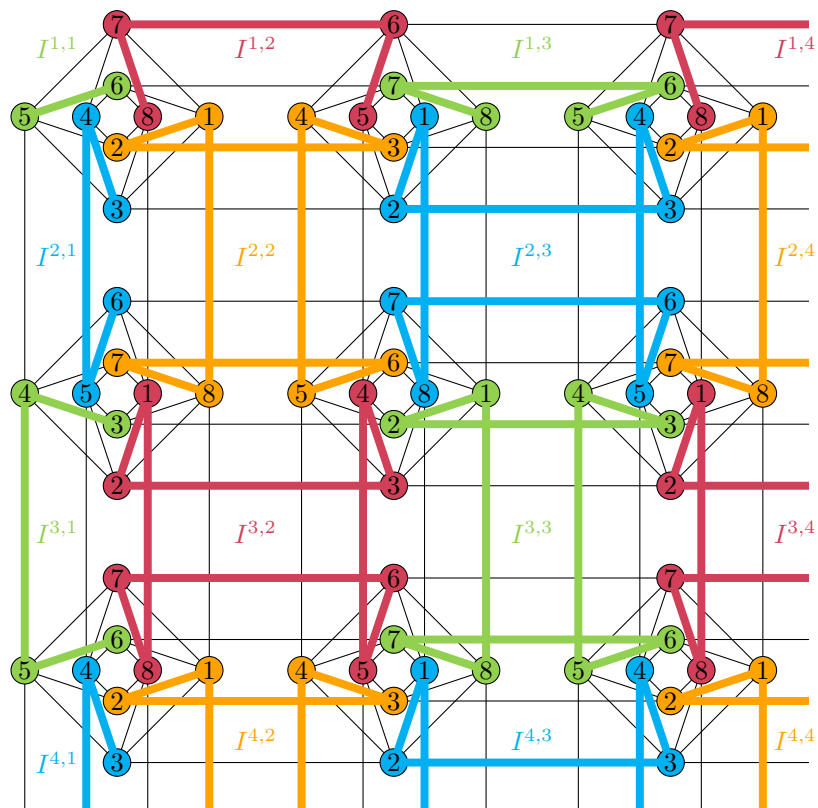
Letztendlich kann mit den Ergebnissen der vorherigen Abschnitte wie folgt zusammengefasst werden:

Folgerung 3.11 Mit $P = P_\varepsilon(g, \tilde{S}, L)$ für $\varepsilon > 0$, $g = (g_0, g_1) \in \mathbb{R}^{V_0 \cup V_1}$, $\tilde{S} \in \mathbb{R}^{(V_0 \cup V_1)^2}$ und $L_v = |I_v|$ für alle $v \in V$ ($C''_{n,m} = V_0 \cup V_1$) gilt

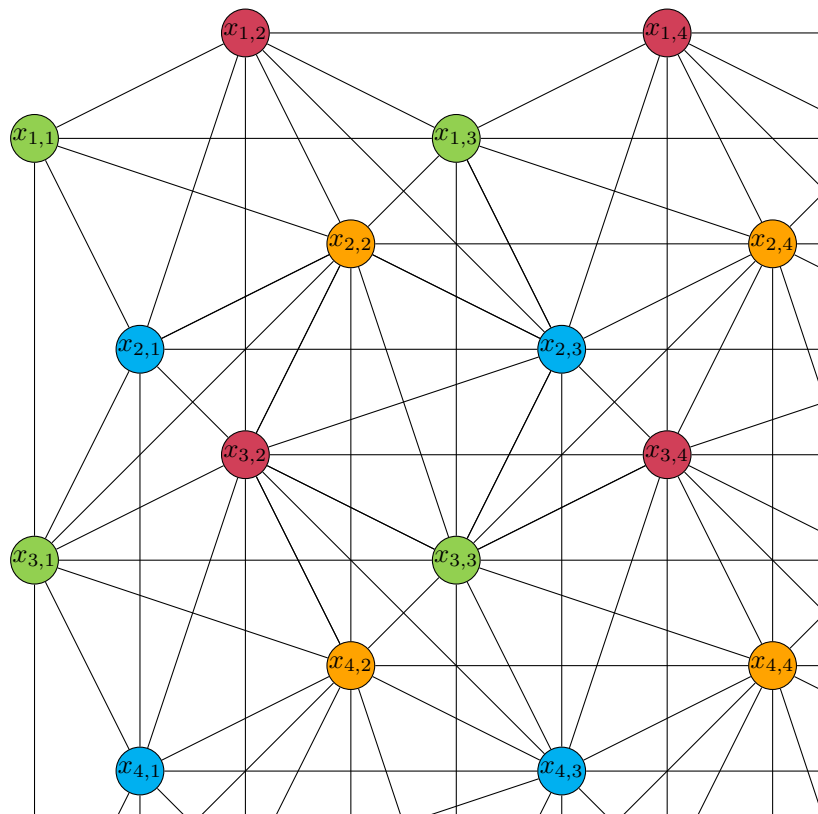
$$\begin{aligned} & \min \left\{ \bar{c}_{n,m}(x) : x \in \{0, 1\}^{2nm+n+m} \right\} \\ &= \min \left\{ \sum_{p=0}^1 \sum_{r=1}^{n+1-p} \sum_{s=1}^{m+p} \left((g_p^{r,s} + (|I_p^{r,s}| - 1)P) \sum_{k=1}^{|I_p^{r,s}|} q_{I_p^{r,s},k}^{r,s} \right. \right. \\ & \quad \left. \left. + (-g_p^{r,s} - |I_p^{r,s}|P) \sum_{k=1}^{|I_p^{r,s}|-1} q_{I_p^{r,s},k}^{r,s} q_{I_p^{r,s},k+1}^{r,s} \right) \right. \\ & \quad + \sum_{r=1}^{n+1} \sum_{s=1}^{m-1} H_0^{r,s} q_{I_{0,1}^{r,s}} q_{I_{0,1}^{r,s+1}} + \sum_{r=1}^n \sum_{s=2}^{m-1} H_1^{r,s} q_{I_{1,3}^{r,s}} q_{I_{1,2}^{r,s+2}} + \sum_{r=1}^n H_1^{r,1} q_{I_{1,2}^{r,1}} q_{I_{1,2}^{r,3}} \\ & \quad + \sum_{r=1}^{n-1} \sum_{s=1}^{m+1} S_1^{r,s} q_{I_{1,1}^{r,s}} q_{I_{1,1}^{r+1,s}} + \sum_{r=2}^{n-1} \sum_{s=1}^m S_0^{r,s} q_{I_{0,3}^{r,s}} q_{I_{0,2}^{r+2,s}} + \sum_{s=1}^m S_0^{1,s} q_{I_{0,2}^{1,s}} q_{I_{0,2}^{3,s}} \\ & \quad \left. + \sum_{r=1}^n \sum_{s=1}^m \sum_{\substack{l,k=1 \\ j < k}}^4 U_{j,k}^{r,s} q_{0,j}^{r,s} q_{1,k}^{r,s} : q \in \{0, 1\}^{n \times m \times 2 \times 4} \right\}. \end{aligned}$$

Dritte Kontraktion

Eine weitere mögliche Kontraktion findet sich in Abbildung 27(b). Wo vorher Knoten über Spalten hinweg kontrahiert wurden, geschah dies nun über Zeilen und umgekehrt. Das heißt, formal wird der Knoten (r, s, p) aus dem vorhergehenden Abschnitt mit $p \in \{0, 1\}$, $r \in \{1, \dots, n+1-p\}$ und $s \in \{1, \dots, m+p\}$ verschmolzen mit $(r, s+1, p)$ für $p=0$, wenn r und s ungerade oder wenn r und s gerade sind, dies entspricht den roten und den blauen Knoten, oder mit $(r+1, s, p)$ für $p=1$, wenn r gerade und s ungerade oder wenn r ungerade und s gerade ist, welche die grünen und orangen Knoten darstellen. Hierbei ist zu beachten, dass die Knoten in den Ecken des Graphen nicht mit einem anderen Knoten verschmolzen werden und daher ihre Qubitstränge im Chimera-Graphen die Länge 2 behalten. Die weiteren Knoten im Rand verdoppeln



(a) Qubitstränge im Chimera-Graphen C zu C'''



(b) Minor C'''

Abb. 27: Dritte Kontaktion

ihre Stranglänge auf 4 und die inneren Knoten auf 8. Wie in 27(a) zu sehen ist, fügen sich damit die Stränge der inneren Knoten zu Kreisen zusammen. Durch diese Struktur verändert sich das Verhältnis der Zahl der Kanten zu der der Knoten und Überlegungen wie im vorherigen Abschnitt müssten entsprechend angepasst werden. Jedoch ist es bei einem Kreis möglich eine Kante durch Nullsetzung ihrer Stärke herauszunehmen und für den restlichen Strang die Überlegungen von vorher zu übernehmen.

Der Graph C''' ist auch symmetrisch und seine Knoten haben einen Ausgangsgrad von 12. Ausgehend von $8mn$ Knoten in $C_{n,m}$, besitzt dieser Graph nur noch $(n+1)(m+1)$ Knoten. Er enthält über all diesen Knoten wieder einen Gittergraphen $L_{n+1,m+1}$, in dessen Gittermaschen auch die diagonalen Kanten vorhanden sind. Die Knoten besitzen zudem horizontale und vertikale Kanten, die jeweils einen Punkt im Gitter überspringen. Durch die Kontraktion haben sich nun zahlreiche Cliques über fünf Knoten gebildet, wobei sich jeder innere Knoten in genau 5 solcher Cliques befindet.

Die Partitionen des vorigen Abschnittes sind bei diesem Graphen überflüssig geworden, daher werden seine Knoten nur anhand der Gitterstruktur mit $V = \{1, \dots, n+1\} \times \{1, \dots, m+1\}$ indiziert und mit den binären Variablen $x \in \{0, 1\}^{(n+1) \times (m+1)}$ versehen. Die Kanten des Graphen sind $E(C'''_{n,m}) = E_U \cup E_H \cup E_S$ mit

$$\begin{aligned} E_U &= \{ \{(r, s), (r+1, s+1)\}, \{(r+1, s), (r, s+1)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m\} \}, \\ E_H &= \{ \{(r, s), (r, s+1)\} : r \in \{1, \dots, n+1\}, s \in \{1, \dots, m\} \\ &\quad \cup \{ \{(r, s), (r, s+2)\} : r \in \{1, \dots, n+1\}, s \in \{1, \dots, m-1\} \}, \\ E_S &= \{ \{(r, s), (r+1, s)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m+1\} \} \\ &\quad \cup \{ \{(r, s), (r+1, s)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m+1\} \}. \end{aligned}$$

Damit lässt sich das entsprechende QUBO mit der Zielfunktion

$$\bar{\bar{c}}_{n,m}(x) = \bar{\bar{c}}_{n,m}^g(x) + \bar{\bar{c}}_{n,m}^U(x) + \bar{\bar{c}}_{n,m}^H(x) + \bar{\bar{c}}_{n,m}^S(x)$$

aufstellen, wobei

$$\begin{aligned} \bar{\bar{c}}_{n,m}^g(x) + \bar{\bar{c}}_{n,m}^U(x) &= \sum_{r=1}^{n+1} \sum_{s=1}^{m+1} g_{r,s} x_{r,s} + \sum_{r=1}^n \sum_{s=1}^m (U_{r,s}^1 x_{r,s} x_{r+1,s+1} + U_{r,s}^2 x_{r+1,s} x_{r,s+1}), \\ \bar{\bar{c}}_{n,m}^H(x) &= \sum_{r=1}^{n+1} \sum_{s=1}^m H_{r,s}^1 x_{r,s} x_{r,s+1} + \sum_{r=1}^{n+1} \sum_{s=1}^{m-1} H_{r,s}^2 x_{r,s} x_{r,s+2}, \\ \bar{\bar{c}}_{n,m}^S(x) &= \sum_{r=1}^n \sum_{s=1}^{m+1} S_{r,s}^1 x_{r,s} x_{r+1,s} + \sum_{r=1}^{n-1} \sum_{s=1}^{m+1} S_{r,s}^2 x_{r,s} x_{r+2,s} \end{aligned}$$

für $g \in \mathbb{R}^{(n+1) \times (m+1)}$, $U \in \mathbb{R}^{n \times m \times 2}$, $H^1 \in \mathbb{R}^{(n+1) \times m}$, $H^2 \in \mathbb{R}^{(n+1) \times (m-1)}$, $S^1 \in \mathbb{R}^{n \times (m+1)}$ und $S^2 \in \mathbb{R}^{(n-1) \times (m+1)}$ ist. Damit ergeben sich insgesamt $6nm + m + n - 2$ Kopplungen.

Die Stränge im Chimera-Graphen des vorherigen Abschnittes können zum Teil übernommen werden. So gilt beispielsweise für die Knoten in den Ecken des Graphen

$$\begin{aligned} I^{1,1} &= ((1, 1, 0, 2), (1, 1, 1, 2)), \\ I^{1,m+1} &= ((1, m, 0, j), (1, m, 1, j)) \end{aligned}$$

mit $j = 1$, wenn m ungerade ist und $j = 2$ sonst,

$$I^{n+1,1} = ((n, 1, 0, j), (n, 1, 1, j))$$

mit $j = 2$, wenn n gerade ist und $j = 3$ sonst, und

$$I^{n+1,m+1} = ((n, m, 0, j), (n, m, 1, j))$$

mit $j = 2$, wenn n und m gerade sind, $j = 4$, wenn n und m ungerade sind, $j = 1$, wenn n gerade und m ungerade ist und $j = 3$ sonst. Für die Stränge der Länge vier am Rand des Chimea-Graphen gilt für $r \in \{2, \dots, n\}$ und $s \in \{2, \dots, m\}$

$$I^{r,1} = ((r-1, 1, 0, j), (r-1, 1, 1, j), (r, 1, 1, j), (r, 1, 0, j))$$

mit $j = 3$, wenn r gerade ist und $j = 2$ sonst,

$$I^{1,s} = ((1, s-1, 1, j), (1, s-1, 0, j), (1, s, 0, j), (1, s, 1, j))$$

mit $j = 1$, wenn r gerade und $j = 3$ sonst,

$$I^{r,m+1} = ((r-1, m, 0, j), (r-1, m, 1, j), (r, m, 1, j), (r, m, 0, j))$$

mit $j = 3$, wenn m und r gerade sind, $j = 1$, wenn m und r ungerade sind, $j = 2$, wenn m gerade und r ungerade ist und $j = 4$ sonst, und

$$I^{n+1,s} = ((n, s-1, 1, j), (n, s-1, 0, j), (n, s, 0, j), (n, s, 1, j))$$

mit $j = 1$, wenn n und s gerade sind, $j = 3$, wenn n und s ungerade sind, $j = 2$, wenn n gerade und s ungerade ist und $j = 4$ sonst. Mit

$$I^{r,s} = \left((r-1, s-1, 1, j), (r-1, s-1, 0, j), (r-1, s, 0, j), (r-1, s, 1, j), \right. \\ \left. (r, s, 1, j), (r, s, 0, j), (r, s-1, 0, j), (r, s-1, 1, j) \right)$$

und $j = 4$, wenn r und s gerade sind, $j = 2$, wenn r und s ungerade sind, $j = 1$, wenn r ungerade und s gerade ist und $j = 3$ sonst, können letztendlich die Qubitstränge der Länge acht für die inneren Knoten mit $r \in \{2, \dots, n\}$ und $s \in \{2, \dots, m\}$ aufgestellt werden.

Zur Vereinfachung der Zielfunktionen seien Indexmengen

$$O^{r,s} \in (\{1, \dots, n\} \times \{1, \dots, m\} \times \{0, 1\} \times \{1, \dots, 4\})^8$$

für alle $r \in \{1, \dots, n+1\}$ und $s \in \{1, \dots, m+1\}$ definiert. Diese können mit

$$O^{r,s} = \begin{cases} I^{r,s} & \text{für } |I^{r,s}| = 8, \\ \left(\overleftarrow{I}^{r,s}, I^{r,s} \right) & \text{für } |I^{r,s}| = 4, \\ \left(\overleftarrow{I}^{r,s}, I^{r,s}, \overleftarrow{I}^{r,s}, I^{r,s} \right) & \text{für } |I^{r,s}| = 2 \end{cases}$$

gebildet werden, wobei $\overleftarrow{I}^{r,s}$ das Tupel darstellt, bei dem die Elemente von hinten nach

vorn getauscht wurden. Da die Qubitstränge zu $I^{r,1}$ und $I^{r,m+1}$ mit $r \in \{2, \dots, n\}$ nicht wie alle anderen in der Partition 1 beginnen, werden hier noch weitere Vertauschungen eingebaut. Daher seien für diese

$$\begin{aligned} O^{r,1} &= \left((r-1, 1, 1, j), (r-1, 1, 0, j), (r-1, 1, 0, j), (r-1, 1, 1, j), \right. \\ &\quad \left. (r, 1, 1, j), (r, 1, 0, j), (r, 1, 0, j), (r, 1, 1, j) \right), \\ O^{r,m+1} &= \left((r-1, m, 1, j), (r-1, m, 0, j), (r-1, m, 0, j), (r-1, m, 1, j), \right. \\ &\quad \left. (r, m, 1, j), (r, m, 0, j), (r, m, 0, j), (r, m, 1, j) \right). \end{aligned}$$

Durch diese Vervielfältigung können nun die Elemente der Stränge, die an einer Kopp- lung der entsprechenden logischen Knoten beteiligt sind, ohne Ausnahmen am Rand bestimmt werden: So lassen sich beispielsweise die horizontalen Kopplungen zwischen Knoten gleicher Farbe über die Strangelemente 5 und 7 realisieren. Damit ergibt sich

Folgerung 3.12 Für $P = P_\varepsilon(g, \tilde{S}, L)$ mit $\varepsilon > 0$, $g \in \mathbb{R}^{(n+1) \times (m+1)}$, $\tilde{S} \in \mathbb{R}^{((n+1) \times (m+1))^2}$ und $L_v = |I^v|$ für alle $v \in V(C'''_{n,m})$ gilt

$$\begin{aligned} &\min \left\{ \bar{C}_{n,m}(x) : x \in \{0, 1\}^{(n+1) \times (m+1)} \right\} \\ &= \min \left\{ \sum_{r=1}^{n+1} \sum_{s=1}^{m+1} \left((g_{r,s} + (|I^{r,s}| - 1)P) \sum_{k=1}^{|I^{r,s}|} q_{I_k^{r,s}} + (-g_{r,s} - |I^{r,s}|P) \sum_{k=1}^{|I^{r,s}|-1} q_{I_k^{r,s}} q_{I_{k+1}^{r,s}} \right) \right. \\ &\quad + \sum_{r=1}^{n+1} \sum_{s=1}^m H_{r,s}^1 q_{O_5^{r,s}} q_{O_7^{r,s+1}} + \sum_{r=1}^{n+1} \sum_{s=1}^{m-1} H_{r,s}^2 q_{O_6^{r,s}} q_{O_7^{r,s+2}} \\ &\quad + \sum_{r=1}^n \sum_{s=1}^{m+1} S_{r,s}^1 q_{O_6^{r,s}} q_{O_4^{r+1,s}} + \sum_{r=1}^{n-1} \sum_{s=1}^{m+1} S_{r,s}^2 q_{O_5^{r,s}} q_{O_4^{r+2,s}} \\ &\quad \left. + \sum_{r=1}^n \sum_{s=1}^m \left(U_{r,s}^1 q_{O_5^{r,s}} q_{O_2^{r+1,s+1}} + U_{r,s}^2 q_{O_7^{r+1,s}} q_{O_4^{r,s+1}} \right) : q \in \{0, 1\}^{n \times m \times 2 \times 4} \right\}. \end{aligned}$$

Weitere analoge symmetrische Kontraktionen im Graphen C''' von Knoten gleicher Far- be zwischen ganzen Zeilen oder Spalten sind nicht sinnvoll. Dies führt nur dazu, dass die entsprechenden Zeilen oder Spalten komplett kollabieren. Damit tritt keine Erhöhung des Ausgangsgrades der Knoten mehr auf, denn der entstandene Graph besitzt dieselbe Struktur wie C''' . Geht man von einem begrenzten Chimera-Graphen aus, hat er je- doch weniger Knoten als vorher. Andere Kontraktionen zwischen Knoten, selbst wenn sie regelmäßig über den Graphen C''' verteilt sind, können zwar den Ausgangsgrad des entstandenen Knotens erhöhen, jedoch verringern sie die Übersichtlichkeit stark, weswegen an dieser Stelle keine weiteren Minoren von C''' gezeigt werden.

3.3.3 Variationen

Die gezeigten Kontraktionen wie von C' zu C'' und C''' lassen sich ebenso in den vorher erwähnten Graphen $L_{m,n}[K_{4k}]$ durchführen. Hier können bei der Verschmelzung über

eine Spalte oder Zeile hinweg sogar bis zu k Knoten verwendet werden, da sie nur entlang der Gitterstruktur, also in vier Richtungen, möglich sind. Die entstehenden Graphen sind jedoch kaum überschaubar und werden daher hier ebenso nicht gezeigt.

Im Folgenden soll stattdessen noch auf ein weiteres Beispiel eingegangen werden. Dieses stellt eine Variation des Graphen $L_{m,n}[K_{4k}]$ dar, mit dem Unterschied, dass zwischen den Cliques K_{4k} nicht nur die parallelen Kanten der Knoten mit gleichen Nummer- und Partitionsindizes sondern alle Kanten vorhanden sind. Das heißt, je zwei benachbarte K_{4k} -Zellen des Graphen bilden einen vollständigen Graphen K_{8k} . Dieser Graph soll mit $V_{n,m}[K_{4k}]$ bezeichnet werden. In Abbildung 28 ist als Beispiel ein Ausschnitt über einem drei mal drei Gitter der K_4 's, also $V_{3,3}[K_4]$, gezeigt. Im Gegensatz zur Darstellung von $L_{m,n}[K_4]$ in Abbildung 19(b) wurden hier für die Übersichtlichkeit die Knoten einer Zelle in einer Diagonalen dargestellt.

Dieser Graph besitzt wie $L_{m,n}[K_4]$ die $4nm$ Knoten $\{1, \dots, n\} \times \{1, \dots, m\} \times \{1, \dots, 4\}$. Die Zahl der horizontalen und vertikalen Kopplungen zwischen den Zellen erhöht sich jeweils um 12 und es gilt $E(L_{n,m}[K_4]) = E_U \cup E_H \cup E_S$ mit

$$\begin{aligned} E_U &= \{(r, s, j), (r, s, l)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m\}, j < l \in \{1, \dots, 4\}, \\ E_H &= \{(r, s, j), (r, s + 1, l)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m - 1\}, j, l \in \{1, \dots, 4\}, \\ E_S &= \{(r, s, j), (r + 1, s, l)\} : r \in \{1, \dots, n - 1\}, s \in \{1, \dots, m\}, j, l \in \{1, \dots, 4\}. \end{aligned}$$

Damit ergeben sich insgesamt $38nm - 16(n + m)$ Kanten im Graphen. Das zugehörige

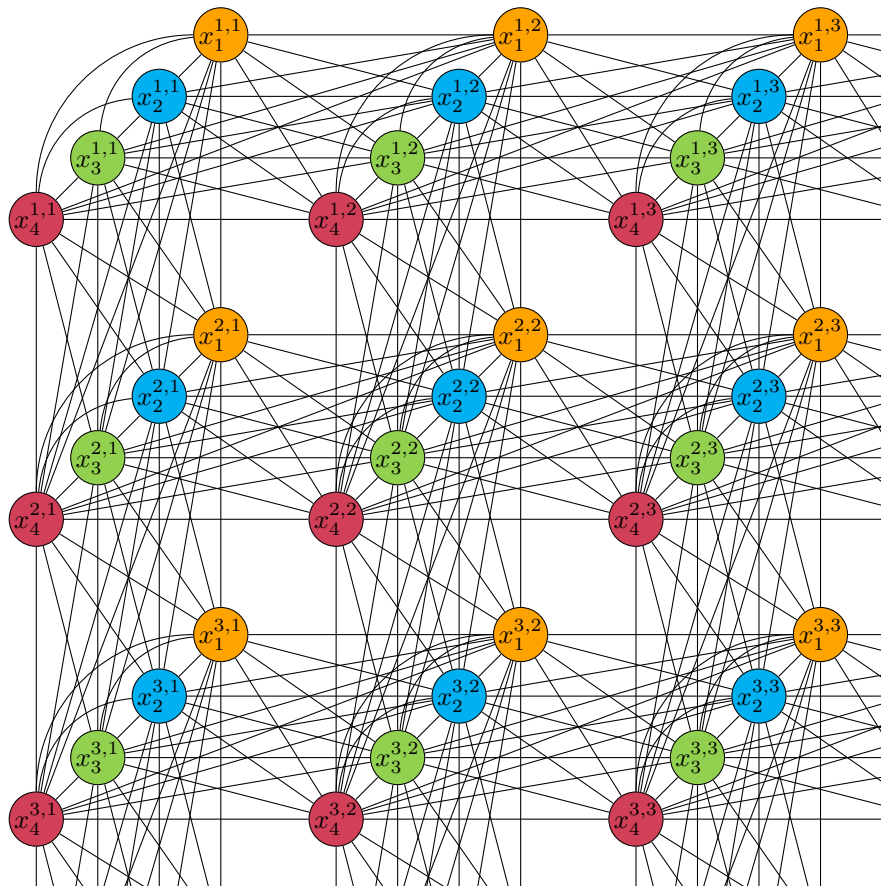


Abb. 28: $V_{n,m}[K_4]$

QUBO lässt sich analog zu $\bar{c}_{n,m}$ aus (XXV) aufstellen mit

$$v_{n,m}(x) = v_{n,m}^g(x) + v_{n,m}^U(x) + v_{n,m}^H(x) + v_{n,m}^S(x)$$

und

$$v_{n,m}^g(x) + v_{n,m}^U(x) = \sum_{r=1}^n \sum_{s=1}^m \left(\sum_{j=1}^4 g_j^{r,s} x_j^{r,s} + \sum_{\substack{j,l=1 \\ l>j}}^4 U_{j,l}^{r,s} x_j^{r,s} x_l^{r,s} \right),$$

$$v_{n,m}^H(x) = \sum_{r=1}^n \sum_{s=1}^{m-1} \sum_{j,l=1}^4 H_{j,l}^{r,s} x_j^{r,s} x_l^{r,s+1},$$

$$v_{n,m}^V(x) = \sum_{r=1}^{n-1} \sum_{s=1}^m \sum_{j,l=1}^4 S_{j,l}^{r,s} x_j^{r,s} x_l^{r+1,s},$$

wobei weiterhin $g \in \mathbb{R}^{n \times m \times 4}$ und $U \in \mathbb{R}^{n \times m \times 4 \times 4}$ sind. Nur $H \in \mathbb{R}^{n \times (m-1) \times 4 \times 4}$ und $S \in \mathbb{R}^{(n-1) \times m \times 4 \times 4}$ wurden entsprechend erweitert.

Die Symmetrie und die Gitterstruktur des Graphen $V_{n,m}[K_4]$ sprechen für eine gute Einbettungsmöglichkeit in den Chimera-Graphen. Jedoch müssen hierbei die folgenden Dinge beachtet werden. Für die Erstellung dieses Graphen über Kontraktionen im Chimera-Graphen wird zunächst von einem vollständigen Graphen mit vier Knoten ausgegangen. Dieser kann in einer Einheitszelle, wie vorher gezeigt, durch Koppeln je zweier Knoten aus unterschiedlichen Partitionen hergestellt werden. Vier weitere solcher Cliques müssen in den umliegenden Einheitszellen so platziert werden, dass

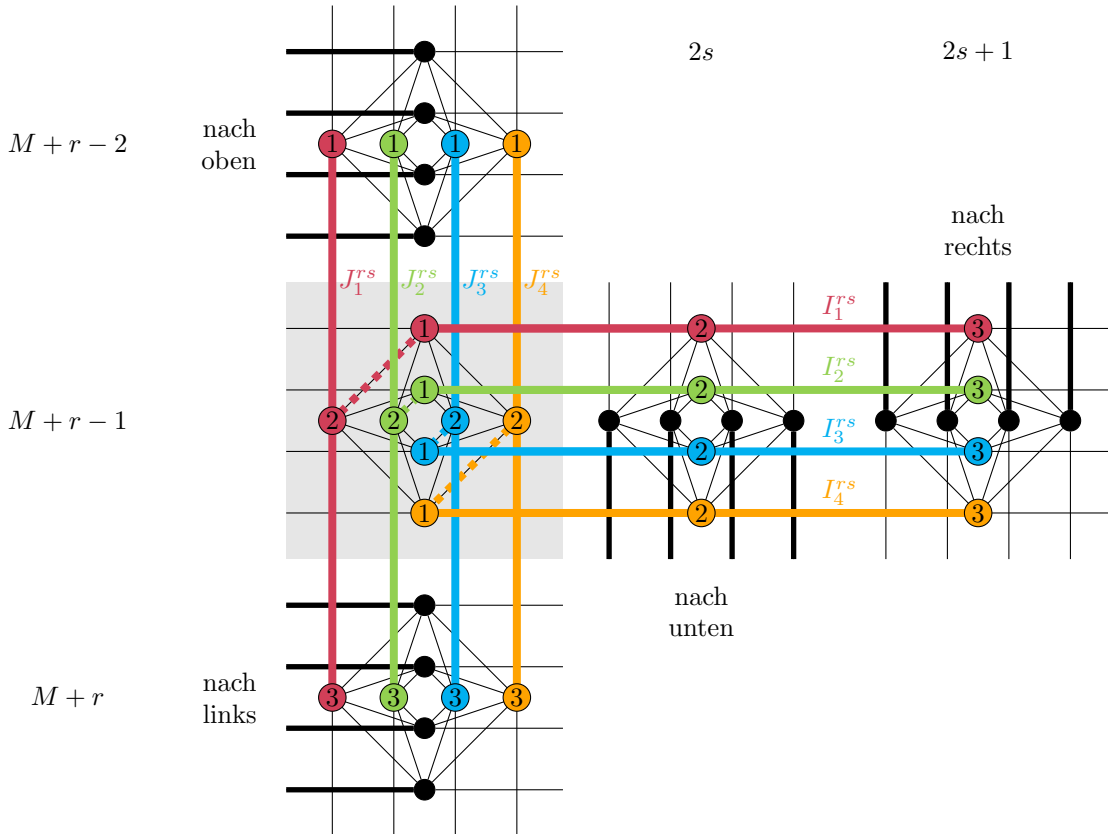


Abb. 29: Qubitstränge in T-Form

jede mit der zentralen gekoppelt werden kann. Das heißt in diesem Fall, dass zwischen zwei dieser Einheitszellen 16 Kanten entstehen müssen. Dies ist nur über eine weitere Einheitszelle, wie im Falle des vollständigen Graphen mit 8 Knoten, möglich. Dabei werden die Stränge der einen Clique in die horizontale Partition und die Stränge der anderen in die vertikale Partition dieser zusätzlichen Zelle verlängert. Dieses Prinzip muss nun für noch drei weitere Zellen mit je einer weiteren Kopplungszelle im umliegenden Graphen fortgeführt werden. Würden die Einheitszellen in einer „+“-Form zusammengefasst, könnten sie nicht entsprechend nah genug an- und aufeinander gelegt werden, um diese Bedingung zu erfüllen, ohne weitere Einheitszellen hinzunehmen zu müssen. Jedoch bietet sich beispielsweise die Möglichkeit an, die Qubitstränge in einer auf die Seite gelegten T-Form anzulegen. Dies ist in Abbildung 29 gezeigt. Dabei zeigt die graue Unterlegung die zentrale Einheitszelle an. Die verstärkten schwarzen Kopplungen zeigen die Richtungen an, an denen die angrenzenden T-Strukturen im Gitter angeordnet sind.

Wie mehrere dieser T-Strukturen im Chimera-Graphen angeordnet sind, zeigt Abbildung 30. Hier wurden die Knoten, die von einer Einheitszelle ausgehen, in der gleichen Farbe markiert. Die grau unterlegte Fläche zeigt die benötigten Einheitszellen zur Bildung eines drei mal drei Gitters an. Es sind insgesamt 23 Einheitszellen. Allgemein zeigt sich, dass für alle Zellen in $V_{n,m}[K_4]$, außer denen am rechten Rand des Gitters, jeweils genau drei horizontal benachbarte Einheitszellen im Chimera-Graphen benötigt werden, welche diagonal versetzt angeordnet sind. Für die Zellen am rechten Rand werden nur jeweils zwei weitere Einheitszellen benötigt, bis auf die Zelle in der rechten unteren Ecke, welche sogar nur noch eine weitere Einheitszelle besetzt. Damit ergeben sich

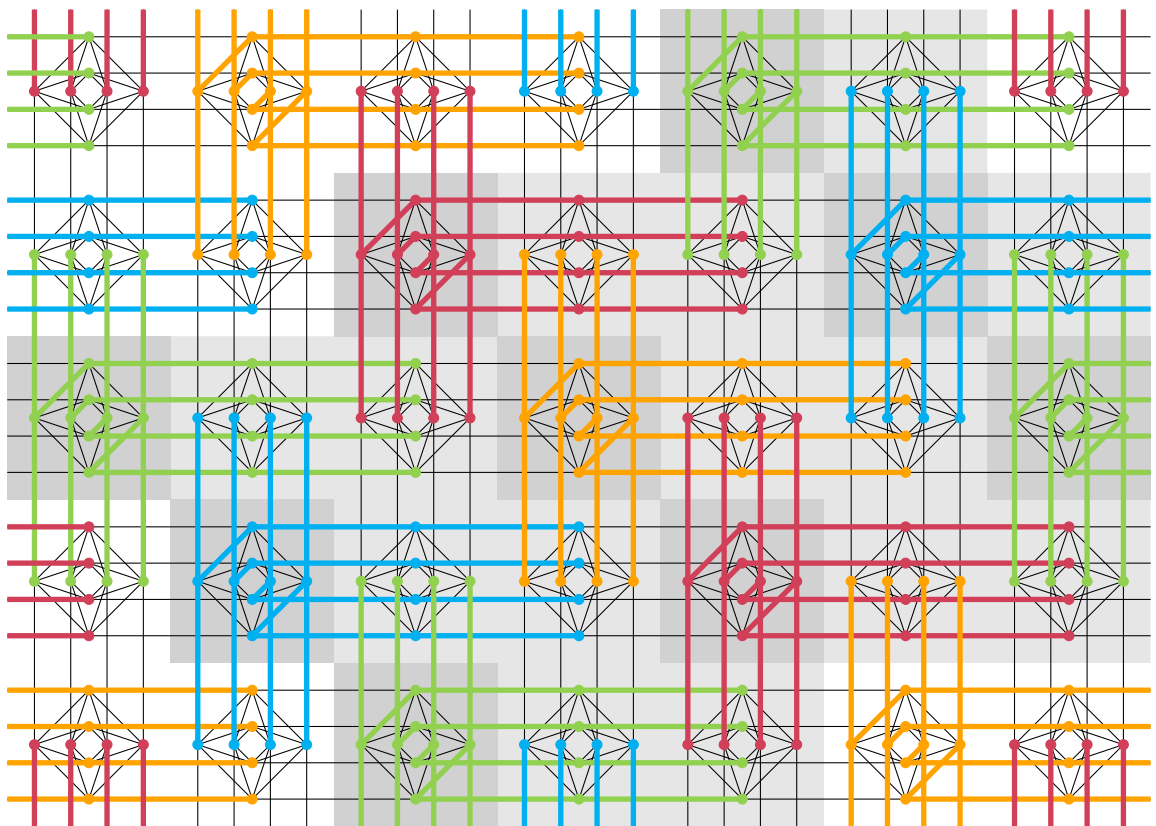


Abb. 30: Qubitstränge im Chimera-Graphen C zu $V_{n,m}[K_4]$

insgesamt $3mn - (n + 1)$ Einheitszellen. Diese liegen aufgrund des versetzten Schachbrettmusters der zentralen Einheitszellen in einem Chimera-Graphen mit mindestens $n + m - 1$ Zeilen und $n + 2(m - 1)$ Spalten. Jedoch kann hier die Gitterstruktur meist um Gitterpunkte erweitert werden, die keine ganze Zeile oder Spalte füllen.

Die Zelle $(1, 1)$ mit den vier Knoten in der linken oberen Ecke des Graphen $V_{n,m}[K_4]$ besitzt bei der Einbettung in den Chimera-Graphen $(m, 1)$ als zentrale Einheitszelle. Davon ausgehend verteilen sich die weiteren zentralen Einheitszellen in einem gleichmäßigen Muster: $(m + r - 1, 2s + r - 2)$ für $r \in \{1, \dots, n\}$ und $s \in \{1, \dots, m\}$. Sofern benötigt, also wenn die Zelle nicht am Rand von $V_{n,m}[K_4]$ liegt, schließen sich die weiteren benötigten Einheitszellen an die zentrale an. Dabei ergibt sich für jeden Knoten $(r, s, j) \in \{1, \dots, n\} \times \{1, \dots, m\} \times \{1, \dots, 4\}$ des Graphen eine Qubitmenge, die aus je zwei Qubitsträngen

$$I_j^{r,s}, J_j^{r,s} \in (\{1, \dots, n + m - 1\} \times \{1, \dots, n + 2(m - 1)\} \times \{0, 1\} \times \{1, \dots, 4\})^3$$

besteht, wobei $I_j^{r,s}$ den horizontalen Teil und $J_j^{r,s}$ den vertikalen Teil beschreibt. Damit ergibt sich

$$\begin{aligned} I_{j,k}^{r,s} &= (m + r - 1, 2s + r + k - 3, 0, j), \\ J_{j,k}^{r,s} &= (m + r + k - 3, 2s + r - 2, 1, j) \end{aligned}$$

für $k = 1, 2, 3$. Über $I_{j,1}^{r,s}$ und $J_{j,2}^{r,s}$ sind diese also zu der T-Form verbunden. Am Rand des Graphen können einige der Elemente dieser Stränge wegfallen: In der ersten Zeile des Graphen entfällt von den vertikalen Strängen der Teil oberhalb der zentralen Einheitszellen und in der ersten Spalte der unterhalb. Die horizontalen Stränge werden dagegen nur in der letzten Spalte verkürzt.

Deswegen soll mit $L^{r,s} \in \mathbb{N}$ die Anzahl der Qubits in der Qubitmenge, die einem logischen Knoten aus der Zelle (r, s) zugeordnet ist, angegeben werden. Dabei sind $L^{1,1} = L^{1,m} = L^{n,m} = 4$, $L^{r,1} = L^{1,s} = 5$ für $r \in \{2, \dots, n\}$ und $s \in \{2, \dots, m - 1\}$ und $L^{r,m} = 5$ für $r \in \{2, \dots, n - 1\}$. Für $r \in \{2, \dots, n\}$ und $s \in \{2, \dots, m - 1\}$ gilt hingegen $L^{r,s} = 6$. Damit lässt sich nun ebenfalls das in den Chimera-Graphen eingebettete QUBO formulieren. Dabei werden die obigen Definitionen für I und J verwendet, obwohl sie Einheitszellen betreffen können, die nicht direkt zur Einbettung des Graphen gehören oder sogar über den Chimera-Graphen hinausragen. Diese Indizes werden jedoch in der Summation nicht angesprochen und dienen nur dazu, die Reihenfolge der Strangenelemente zu erhalten.

Folgerung 3.13 Für $P = P_\varepsilon(g, \tilde{S}, L)$ mit $\varepsilon > 0$, $g \in \mathbb{R}^{n \times m \times 4}$, $\tilde{S} \in \mathbb{R}^{(n \times m \times 4)^2}$ und $L_v = L^{r,s}$ für alle $v = (r, s, j) \in V(V_{n,m}[K_4])$ gilt mit

$$\begin{aligned} & \min \left\{ v_{n,m}(x) : x \in \{0, 1\}^{n \times m \times 4} \right\} \\ &= \min \left\{ \sum_{r=1}^n \sum_{s=1}^m \sum_{j=1}^4 \left((g_j^{r,s} + (L^{r,s} - 1)P) \left(q_{I_{j,1}^{r,s}} + q_{J_{j,2}^{r,s}} \right) + (-g_j^{r,s} - L^{r,s}P) q_{I_{j,1}^{r,s}} q_{J_{j,2}^{r,s}} \right) \right. \\ & \quad + \sum_{r=1}^n \sum_{s=1}^{m-1} \sum_{j=1}^4 \left((g_j^{r,s} + (L^{r,s} - 1)P) q_{I_{j,3}^{r,s}} + (g_j^{r,s+1} + (L^{r,s+1} - 1)P) q_{J_{j,3}^{r,s+1}} \right. \\ & \quad \left. \left. + (-g_j^{r,s} - L^{r,s}P) q_{I_{j,2}^{r,s}} q_{I_{j,3}^{r,s}} + (-g_j^{r,s+1} - L^{r,s+1}P) q_{J_{j,2}^{r,s+1}} q_{J_{j,3}^{r,s+1}} \right) \right\} \end{aligned}$$

$$\begin{aligned}
& + \sum_{r=1}^{n-1} \sum_{s=1}^m \sum_{j=1}^4 \left((g_j^{r,s} + (L^{r,s} - 1)P) q_{I_{j,2}^{r,s}} + (g_j^{r+1,s} + (L^{r+1,s} - 1)P) q_{J_{j,1}^{r+1,s}} \right. \\
& \quad \left. + (-g_j^{r,s} - L^{r,s}P) q_{I_{j,1}^{r,s}} q_{I_{j,2}^{r,s}} + (-g_j^{r+1,s} - L^{r+1,s}P) q_{J_{j,1}^{r+1,s}} q_{J_{j,2}^{r+1,s}} \right) \\
& + \sum_{s=1}^{m-1} \sum_{j=1}^4 \left((g_j^{n,s} + (L^{n,s} - 1)P) q_{I_{j,2}^{n,s}} + (-g_j^{n,s} - L^{n,s}P) q_{I_{j,1}^{n,s}} q_{I_{j,2}^{n,s}} \right) \quad (*) \\
& + \sum_{r=1}^n \sum_{s=1}^m \sum_{\substack{j,k=1 \\ j < k}}^4 U_{j,k}^{r,s} q_{I_{j,1}^{r,s}} q_{J_{k,2}^{r,s}} + \sum_{r=1}^n \sum_{s=1}^{m-1} \sum_{j,l=1}^4 H_{j,l}^{r,s} q_{I_{j,3}^{r,s}} q_{J_{l,3}^{r,s+1}} \\
& + \left. \sum_{r=1}^{n-1} \sum_{s=1}^m \sum_{j,l=1}^4 S_{j,l}^{r,s} q_{I_{j,2}^{r,s}} q_{J_{l,1}^{r+1,s}} : q \in \{0, 1\}^{(n+m-1) \times (n+2m-2) \times 2 \times 4} \right\}.
\end{aligned}$$

Der Summand (*) wird hierbei benötigt, da die Zellen in der letzten Zeile des Graphen $V_{n,m}[K_4]$ zwar keine weiteren Kopplungen nach unten besitzen, aber die angesprochenen Einheitszellen zwischen den zentralen Einheitszellen und denen, die für die Kopplungen nach rechts verantwortlich sind, liegen und daher ebenfalls mit den entsprechenden Gewichten und Stärken der sie durchlaufenden Stränge versehen werden müssen.

Analog zu $L_{n,m}[K_{4k}]$, kann das Prinzip von $V_{n,m}[K_4]$ auch auf $V_{n,m}[K_{4k}]$ erweitert werden: Statt einer einzelnen zentralen Einheitszelle werden nun k mal k Einheitszellen zu einem vollständigen Graphen mit $4k$ logischen Knoten zusammengefasst. Deren Qubitstränge werden ebenso erweitert, wie im Teilabschnitt Erste Kontraktion von 3.3.2 gezeigt wurde. Über diese zentralen Zellen hinweg wird die T -förmige Struktur in die angrenzenden Einheitszellen ausgeweitet. Sie umfasst nun k -mal so viele Zeilen und Spalten wie bei $V_{n,m}[K_4]$. Durch die gleiche An- und Übereinanderlegung dieser T 's können die für $V_{n,m}[K_{4k}]$ benötigten Kopplungen erreicht werden. Damit benötigt dieser Graph einen Chimera-Graphen mit mindestens $k(n+m-1)$ Zeilen und $k(n+2m-2)$ Spalten.

Es existiert eine Vielzahl weiterer Varianten symmetrischer und nichtsymmetrischer Kontraktionen, die hier nicht im Einzelnen vorgestellt werden sollen und können. Denn die Zahl der Minoren des Chimera-Graphen C ist natürlich unendlich. Selbst für einen begrenzten Chimera-Graphen sind die Möglichkeiten zusammenhängende Teilgraphen zur Kontraktion zu wählen unüberschaubar. Allgemein spielen dabei jedoch Minoren eines auf n Zeilen und m Spalten begrenzten Chimera-Graphen keine Rolle, deren Knotenzahl kleiner ist als $4 \min(n, m)$. Deren Graph kann über die Einbettung des vollständigen Graphen besser realisiert werden. Für einen beliebigen Ausgangsgraphen, insbesondere für einen der keine Symmetrie oder Gitterstruktur aufweist, muss immer im Einzelfall eine mögliche Einbettung in den Chimera-Graphen gefunden werden. Ist der Graph begrenzt, so kann dies ein schweres Problem darstellen, wie bereits im vorherigen Abschnitt gezeigt wurde. Aufgrund der gezeigten Strukturen liegt es daher nahe, bei Berechnungen auf der D-Wave-Maschine QUBOs zu betrachten, die in ihrem Interaktionsgraphen eine Art Gitterstruktur aufweisen. Beispiele dazu werden im Abschnitt 3.4.2 gezeigt.

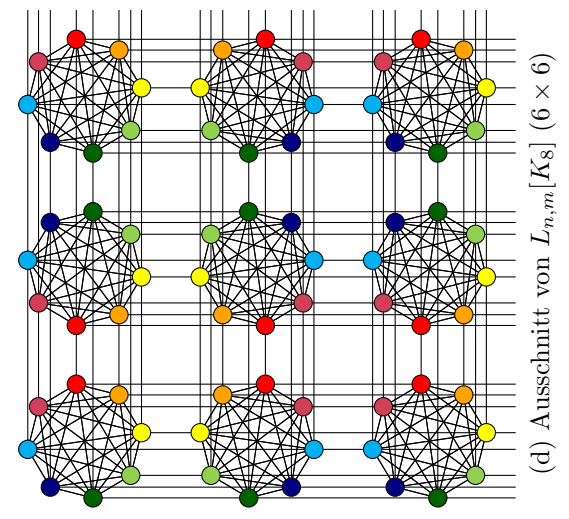
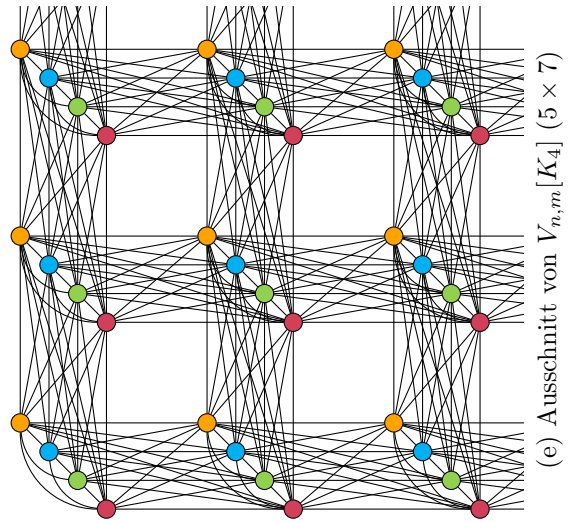
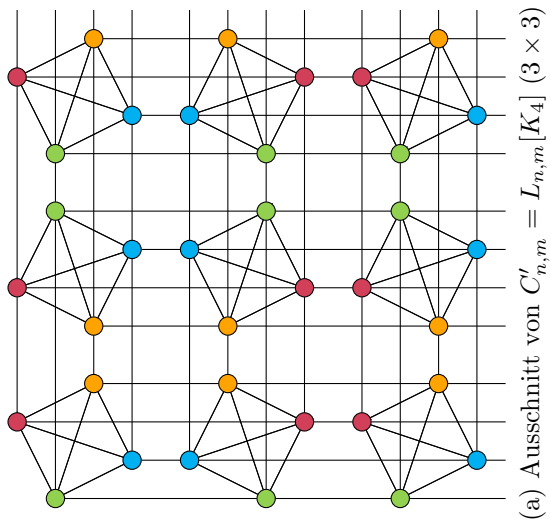
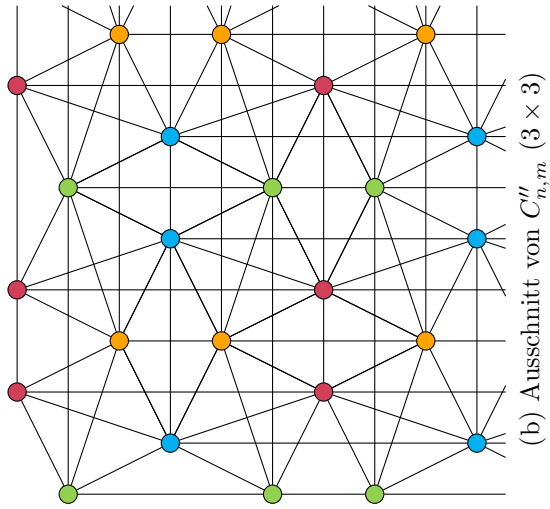
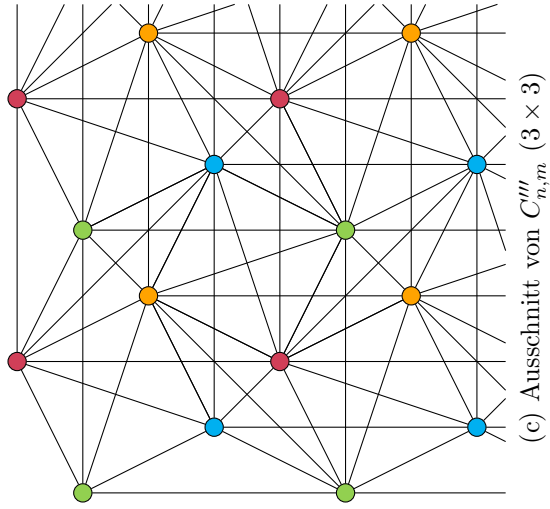


Abb. 31: Minoren des Chimera-Graphen
(hervorgegangen aus $x \times y$
Einheitszellen)

3.3 Spezielle Minoren

Graph $G = (V, E)$	Anzahl Knoten $ V $	Anzahl Kanten $ E $	max. und min. Ausgangsgrad $\max_{v \in V} d(v) $	größte Clique $\omega(G)$	Eigenschaften	Bemerkungen
$C_{n,m} = L_{n,m}[K_{4,4}]$	$8nm$	$24nm - 4(n+m)$	6 5	2	bipartit enthält $n \times m$ $K_{4,4}$'s	$n, m > 1$
$L_{\frac{n}{k}, \frac{m}{l}}[K_{4k,4l}]$	$4nm(\frac{1}{k} + \frac{1}{l})$	$16nm + 4nm(\frac{1}{k} + \frac{1}{l}) - 4(n+m)$	$4k+2$ $4l+1$	2	bipartit enthält $\frac{n}{k} \times \frac{m}{l}$ $K_{4k,4l}$'s	$k \geq l$ $k \mid n > k$ $l \mid m > l$
$K_{4n,4m}$	$4(n+m)$	$16nm$	$4n$	2	vollständig bipartit	$n \geq m$
$C'_{n,m} = L_{n,m}[K_4]$	$4nm$	$14nm - 4(n+m)$	7 5	4	enthält $n \times m$ K_4 's, $L_{2n,2m}, L_{n,m,4}$	$n, m > 1$
$L_{\frac{n}{k}, \frac{m}{k}}[K_{4k}]$	$\frac{4}{k}nm$	$(8 + \frac{6}{k})nm - 4(n+m)$	$4k+3$ $4k+1$	$4k$	enthält $\frac{n}{k} \times \frac{m}{k}$ K_{4k} 's, $L_{\frac{n}{k}, \frac{m}{k}, 4k}$	$k \mid n, m > k$
K_{4n}	$4n$	$8n^2 - 2n$	$4n-1$ $4n-1$	$4n$	vollständig	$n = m$
$C''_{n,m}$	$2nm + m + n$	$10nm - n - m - 2$	10 5	4	jeder Knoten Teil von 4 4er-Cliquen	
$C'''_{n,m}$	$nm + n + m + 1$	$6nm + m + n + 2$	12 5	5	jeder Knoten Teil von 5 5er-Cliquen	
$V_{N,M}[K_4]$	$4NM$	$38NM - 16(N+M)$	19 11	8	je zwei benachbarte K_4 bilden K_8	$N, M > 1$ $N+M \leq n+1$ $N+2M \leq m+2$
$V_{N,M}[K_{4k}]$	$4kNM$	$(40k^2 - 2k)NM - 16k^2(N+M)$	$16k+$ $(4k-1)$ $(4k-1)$	$8k$	je zwei benachbarte K_{4k} bilden K_{8k}	$N, M > 1$ $N+M \leq \frac{n}{k} + 1$ $N+2M \leq \frac{m}{k} + 2$

Abb. 32: Zusammenfassung der Minoren in $C_{n,m}$

3.3.4 Zusammenfassung

Auf der vorhergehenden Doppelseite wurden noch einmal die gefundenen Minoren und eine Tabelle mit den dazugehörigen Kennzahlen gezeigt. Es gibt einige Unterschiede bezüglich der Eigenschaften, weshalb immer je nach Problemstellung der entsprechende Graph gewählt werden muss. Die Liste der möglichen Minoren ist dabei keineswegs vollständig und könnte durch unzählige mögliche Reduktionen im Chimera-Graphen weiter ergänzt werden. Jedoch zeigt sich deutlich, dass die Klasse der Graphen, die gut, das heißt über möglichst wenige Einheitszellen, in den Chimera-Graphen einzubetten sind, doch sehr speziell ist. Daher ist es unwahrscheinlich, dass sich zu diesen Problemen direkt ein praktischer Bezug herstellen lässt.

Die Tabelle liefert zudem die Schranken an die Größe der einbettbaren Minoren. Auf dem aktuellen D-Wave-Chip mit 12 mal 12 Einheitszellen lassen sich daher nur die folgenden größtmöglichen Minoren darstellen:

$$\begin{array}{llllll}
 L_{12,12}[K_{4,4}] = C_{12,12}, & L_{12,12}[K_4], & V_{12,1}[K_4], & V_{10,2}[K_4], & V_{8,3}[K_4], & V_{6,4}[K_4]. \\
 L_{6,6}[K_{8,8}], & L_{6,6}[K_8], & V_{6,1}[K_8], & V_{4,2}[K_8], & & \\
 L_{4,4}[K_{12,12}], & L_{4,4}[K_{12}], & V_{4,1}[K_{12}], & V_{2,2}[K_{12}], & & \\
 L_{3,3}[K_{16,16}], & L_{3,3}[K_{16}], & V_{3,1}[K_{16}], & & & \\
 L_{2,2}[K_{24,24}], & L_{2,2}[K_{24}], & V_{2,1}[K_{24}], & & & \\
 K_{48,48}, & K_{48}, & & & &
 \end{array}$$

Dies würde natürlich nur für solche Chips, die keine Produktionsfehler aufweisen, gelten. Durch das Fehlen einiger Qubits und Kopplungen können diese Graphen möglicherweise nicht vollständig einbettet werden.

Die Aufstellung der QUBOs und deren Umformulierung zu Chimera-QUBOs für die gezeigten Minoren des vorigen Abschnittes verlief immer nach dem gleichen Muster. Die im Folgenden noch einmal zusammengefassten Schritte liefern einen allgemeinen Implementationsweg und können ebenso für die Betrachtung weiterer Minoren angewandt werden: Zunächst wurde eine passende Indizierung der Knoten des Minoren gesucht und damit die Knoten- und Kantenmengen formuliert. Insbesondere wenn der Graph eine Gitterstruktur enthält, erleichtert dies den nächsten Schritt: die Zuordnung von zusammenhängenden Qubitmengen, meist Strängen, zu jedem einzelnen Knoten. Dazu wurden Einbettungsfunktionen

$$I : V \rightarrow (\{1, \dots, n\} \times \{1, \dots, m\} \times \{0, 1\} \times \{1, \dots, 4\})^d$$

aufgestellt, die eine geordnete Menge an Qubitindizes einer entsprechenden Größe d liefern. Sie liefern die Relationen zwischen den Qubits und den Knotenindizes, wodurch die Einbettung des Minoren in den Chimera-Graphen realisiert werden kann.

Im nächsten Schritt wurden die Gewichte des ursprünglichen QUBOs übertragen. Anhand der Länge der zugehörigen Stränge, des Gewichtes und der Stärken der ausgehenden Kanten, wie sie für die entsprechenden Funktionen $\check{\sigma}$ und $\hat{\sigma}$ benötigt werden, können die Werte für die Qubits ermittelt werden. Nun muss weiterhin für jede Kante im Graphen eine entsprechende Kopplung zwischen den Qubits gefunden werden. Das heißt zu jedem gekoppelten Knotenpaar wird ein Paar von Qubitindizes gegeben. Diese

können zum einen direkt angegeben werden, oder wenn die Struktur der Qubitstränge es zulässt, über Strangindizes, die ein bestimmtes Element des Qubitstrangs liefern. Gezeigt wurde hierbei nur jeweils eine mögliche Indizierung und Zuordnung zwischen den Mengen.

3.4 Beispiele

3.4.1 Direkte Umsetzung

Für diese Beispiele gilt, besitzen sie als Ausgangsgraphen einen der gezeigten Minoren des Chimera-Graphen, wie zum Beispiel den Gittergraphen, dann können sie mit den entsprechenden Qubitsträngen direkt eingebettet werden. Sind keine speziellen Eigenschaften des Ausgangsgraphen bekannt, so lässt sich das Problem auf die einfachste Weise oft nur über die Einbettung eines vollständigen Graphen realisieren. Dies wird in den folgenden Abschnitten anhand der Beispiele aus Abschnitt 2.3.1 mit den Definitionen und Methoden aus 3.2.3 gezeigt. Für die Darstellung auf dem aktuellen D-Wave-Chip dürfen die Graphen dieser Beispiele nur eine Knotenzahl von maximal 48 besitzen.

Stabile Menge

Ist ein Graph $G = (V, E)$ mit $|V| = n$ gegeben und nur die Größe seiner kardinalitätsmaximalen stabilen Menge gesucht, so kann für alle Gewichte $g = -1$ und für alle Stärken $S = 2$ gewählt werden. Mit $n > 4$ werden Qubitstränge der Länge $\lceil \frac{n}{4} \rceil + 1 \geq 3$ benötigt. Da hier keine negativen Stärken vorhanden sind, gilt $\check{\sigma}(v) = 0$ für alle $v \in V$. Weiterhin ist $\hat{\sigma}(v) = 2|d(v)|$. In einem Graphen, der nur aus isolierten Knoten besteht, ist die Suche nach einer stabilen Menge trivial. Daher kann die Nachbarschaft mindestens zweier Knoten angenommen werden, womit $\hat{\sigma}(v) \geq 2$ für mindestens ein $v \in V$ und so $\max\{\min\{1, \hat{\sigma}(v)\} : v \in V\} = 1$ gilt. Somit ist nach Folgerung 3.2 $P = 2$ ausreichend groß und die entsprechenden Gewichte und Stärken der Qubits in den Qubitstränge ergeben sich nach (XXII) mit

$$\begin{aligned}\bar{g} &= -1 + 2 \lceil \frac{n}{4} \rceil, \\ \bar{S} &= -1 - 2 \lceil \frac{n}{4} \rceil\end{aligned}$$

für alle Knoten des Graphen. Die Lösung des Chimera-QUBOs entspricht damit einer stabilen Menge von G und für die Stabilitätszahl $\alpha(G)$ folgt:

Folgerung 3.14 Mit I und O , wie in Abschnitt 3.2.2 für die Einbettung des vollständigen Graphen gegeben, gilt

$$\begin{aligned}\alpha(G) = - \min \left\{ \sum_{v \in V} \left((-1 + 2 \lceil \frac{n}{4} \rceil) \sum_{k=1}^{\lceil \frac{n}{4} \rceil + 1} q_{I_{v,k}} - (1 + 2 \lceil \frac{n}{4} \rceil) \sum_{k=1}^{\lceil \frac{n}{4} \rceil} q_{I_{v,k}} q_{I_{v,k+1}} \right) \right. \\ \left. + \sum_{\{v,w\} \in E} 2q_{O_{v,w}} q_{O_{w,v}} : q \in \{0, 1\}^{n \times (\lceil \frac{n}{4} \rceil + 1)} \right\}.\end{aligned}$$

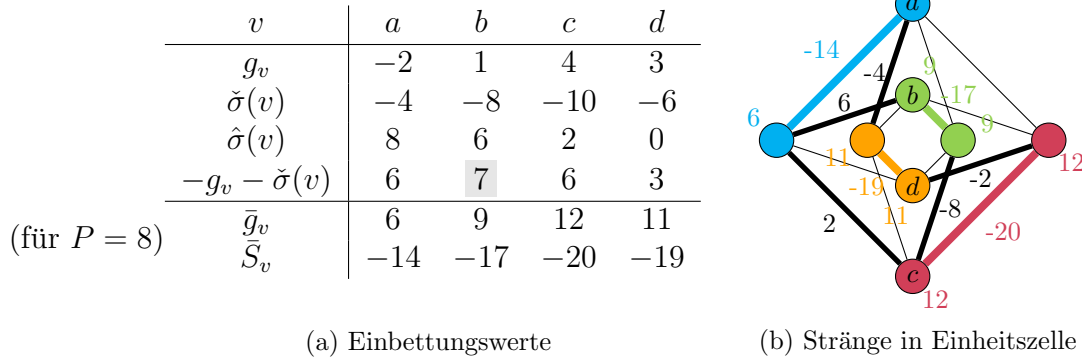


Abb. 33: Einbettung zu Beispielgraph aus Abbildung 5

Maximaler Schnitt

Gegeben ist ein ungerichteter Graph $G = (V, E)$ mit $|V| = n$ und der Bewertungsfunktion $S : E \rightarrow \mathbb{R}$. Wie im Abschnitt 2.3.1 gezeigt, ergeben sich damit für das entsprechende Max-Cut-QUBO die Gewichte

$$g_v = - \sum_{w \in d(v)} S_{v,w}$$

für alle Knoten $v \in V$ und die Stärken $2S_{v,w}$ für alle Kanten $\{v, w\}$ aus E . Damit gilt

$$\check{\sigma}(v) + \hat{\sigma}(v) = \sum_{w \in d(v)} (S_{v,w} - |S_{v,w}|) + \sum_{w \in d(v)} (S_{v,w} + |S_{v,w}|) = -2g_v$$

und g_v ist genau dann negativ, wenn $|\check{\sigma}(v)| < \hat{\sigma}(v)$ ist.

Hierfür lässt sich bei der Bestimmung von P keine einfache Regel verallgemeinern, daher wird das konkrete Beispiel aus Abbildung 5 aus Abschnitt 2.3.1 betrachtet. Der gezeigte Graph umfasst nur vier Knoten und kann damit in einer einzelnen Einheitszelle mit vier Qubitsträngen der Länge 2, welche je ein Qubit einer vertikalen und einer horizontalen Partition umfassen, eingebettet werden. In der Tabelle in Abbildung 33 sind die nötigen Werte aufgeführt, die nach Folgerung 3.2 zur Bestimmung von P benötigt werden. Es zeigt sich, dass $P > 7$ gewählt werden muss. Die entsprechende Einbettung in eine Einheitszelle ist in Abbildung 33(b) gezeigt.

Allgemein gilt für ein Max-Cut-QUBO:

Folgerung 3.15 Für $P = P_\varepsilon (S \mathbb{1}_V, 2S, (\lceil \frac{n}{4} \rceil + 1) \mathbb{1}_V)$ mit $\varepsilon > 0$ gilt für einen Graphen $G = (V, E)$ mit den Stärken $S \in \mathbb{R}^{V \times V}$ und $|V| = n$ über die Einbettung des vollständigen Graphen mit I und O , wie in Abschnitt 3.2.2 gegeben, dass

$$\begin{aligned} \text{MaxCut}(G, S) = - \min & \left\{ \sum_{v \in V} \left(- \sum_{w \in d(v)} S_{v,w} + \lceil \frac{n}{4} \rceil P \right) \sum_{k=1}^{\lceil \frac{n}{4} \rceil + 1} q_{I_{v,k}} \right. \\ & \left. + \sum_{v \in V} \left(\sum_{w \in d(v)} S_{v,w} - (\lceil \frac{n}{4} \rceil + 1) P \right) \sum_{k=1}^{\lceil \frac{n}{4} \rceil} q_{I_{v,k}} q_{I_{v,k+1}} \right\} \end{aligned}$$

$$+ \sum_{\{v,w\} \in E} 2S_{v,w} q_{O_{v,w}} q_{O_{w,v}} : q \in \{0, 1\}^{n \times (\lceil \frac{n}{4} \rceil + 1)} \Big\}.$$

Auch bei diesem Beispiel gilt, besitzt der Graph bereits eine Gitterstruktur oder entspricht sogar einem Teilgraphen der vorher gezeigten Minoren, so können direkt deren QUBO-Formulierungen verwendet werden. Hierbei werden die Stärken und Gewichte eingesetzt, P ausgewählt und eventuell überflüssige Kanten durch Nullsetzen der Stärken aus dem QUBO entfernt.

3.4.2 Im Gittergraphen

Wie die Untersuchung der Minoren des Chimera-Graphen gezeigt hat, scheinen besonders Probleme, deren QUBOs eine Gitterstruktur in ihrem Interaktionsgraphen aufweisen, interessant sein, um die Fähigkeiten des Quantencomputers von D-Wave-Systems zu evaluieren. Daher wird im Folgenden auf zwei Problemstellungen eingegangen, die genau dies vorweisen. Die Graphenprobleme werden zwar direkt in Gittergraphen betrachtet, jedoch werden durch ihre Übertragung in ein QUBO veränderte Interaktionsgraphen erhalten. Die zugrundeliegende Gitterstruktur bleibt aber erhalten, wodurch sie mit den Minoren des Chimera-Graphen in Verbindung gebracht werden können.

Matching

In 2.3.1 wurde im Abschnitt zum Matching-Problem gezeigt, dass das entsprechende QUBO als Interaktionsgraphen den Kantengraph des Ausgangsgraphen besitzt. Daher ist nun die Frage, welche Graphen einen Minor oder Teilgraphen des Chimera-Graphen als Kantengraph besitzen. Ob ein gegebener Graph ein Kantengraph ist, kann sogar in linearer Zeit herausgefunden werden. Einen solchen Algorithmus zeigte N. Rousso-poulos in [40]. L. Beineke ermittelte vorher in [41] genau neun minimale Graphen, die keine Kantengraphen darstellen. Mit diesen können Kantengraphen ebenfalls charakterisiert werden: Induziert eine Teilmenge von Knoten einen dieser neun Graphen, dass heißt, bilden sie und alle zwischen ihnen bestehenden Kanten genau diesen Graphen, kann der Ausgangsgraph kein Kantengraph sein. Einer dieser minimalen Graphen ist der vollständig bipartite Graph $K_{1,3}$, welcher in Abbildung 34 gezeigt ist. Dieser ist im Chimera-Graphen sowie in allen gezeigten Minoren $L_{\frac{n}{k}, \frac{m}{l}}[K_{4k,4l}]$, $L_{n,m}[K_{4k}]$, $C''_{n,m}$, $C'''_{n,m}$ und $V_{n,m}[K_{4k}]$ enthalten, da bei letzteren die Knoten einer Farbe immer ein Gitter bilden. In einem solchen hat ein Knoten, der nicht auf dem Rand liegt, sogar vier Nachbarn, die untereinander jeweils nicht benachbart sind. Daher bilden all diese Graphen nicht direkt den Kantengraph eines anderen Graphen.

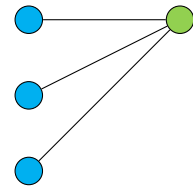


Abb. 34: $K_{1,3}$

Es könnten zwar Teilgraphen gesucht werden, in denen versucht wird die neun minimalen Graphen als induzierte Teilgraphen zu vermeiden. Jedoch ist dies, durch die Vielzahl an Möglichkeiten Kanten zu entfernen, zu aufwändig. Daher sollen nun direkt Gittergraphen und deren Kantengraphen untersucht werden.

Die inneren Knoten eines Gittergraphen $L_{n,m}$ besitzen jeweils genau vier Nachbar-knoten. Bei der Konstruktion des Kantengraphs $\mathcal{L}(L_{n,m})$ fügen sich diese Kanten zu

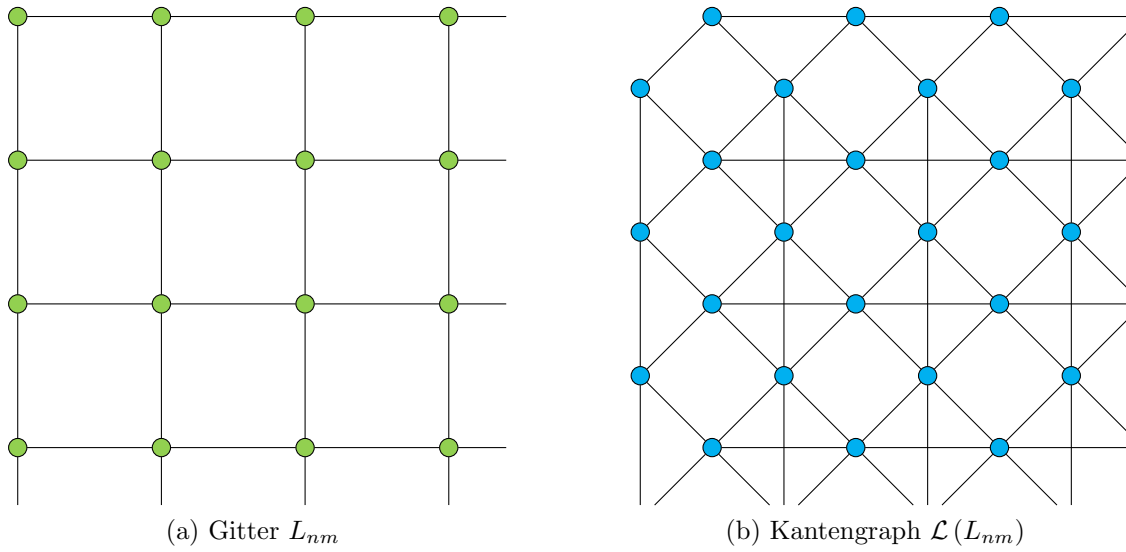


Abb. 35: Kantengraph des Gittergraphen

einer Clique zusammen. Da das gleiche ebenso für die Nachbarknoten zutrifft, sind die Cliques gitterartig verbunden. Dies ist in Abbildung 35 gezeigt. Dieser Graph weist starke Ähnlichkeiten mit dem erweiterten Teilgraphen $\tilde{L}_{n,m}$ von $C''_{n,m}$ aus Abbildung 25 auf Seite 57 auf. Daher werden die Kanten des ursprünglichen Gittergraphen analog in zwei Partitionen unterteilt: Wert 0 für die horizontalen Kanten und Wert 1 für die vertikalen. Weiterhin werden sie mit der Zeile und der Spalte indiziert, weswegen also

$$\begin{aligned} E(L_{n,m}) &= V(\mathcal{L}(L_{n,m})) \\ &= \{(r, s, p) : p \in \{0, 1\}, r \in \{1, \dots, n-p\}, s \in \{1, \dots, m-1+p\}\} \end{aligned}$$

gilt. Im Gegensatz zur Darstellung von $C''_{n,m}$ fehlen jedoch einige Kanten. Das sind zum einen jene horizontalen und vertikalen Kopplungen, die einen Index überspringen, womit gilt $E(\mathcal{L}(L_{n,m})) = E_U \cup E_H \cup E_S$ mit

$$\begin{aligned} E_H &= \{(r, s, 0), (r, s+1, 0)\} : r \in \{1, \dots, n\}, s \in \{1, \dots, m-2\}, \\ E_S &= \{(r, s, 1), (r+1, s, 1)\} : r \in \{1, \dots, n-2\}, s \in \{1, \dots, m\}. \end{aligned}$$

Zum anderen fehlen die Diagonalen der Vierercliquen, die in E_U kodiert sind. Daher gilt hierfür

$$\begin{aligned} E_U &= \{(r, s, 0), (r, s, 1)\}, \{(r, s, 0), (r, s+1, 1)\}, \{(r+1, s, 0), (r, s, 1)\}, \\ &\quad \{(r+1, s, 0), (r, s+1, 1)\} : r \in \{1, \dots, n-1\}, s \in \{1, \dots, m-1\}. \end{aligned}$$

Die Knoten werden mit den Variablen $x_0 \in \{0, 1\}^{n \times (m-1)}$ und $x_1 \in \{0, 1\}^{(n-1) \times m}$ versehen. Nun kann auch das QUBO zu diesem Kantengraphen mit einer ähnlichen Zielfunktion wie (XXVII) aufgestellt werden mit

$$\ell_{n,m}(x) = \ell_{n,m}^g(x) + \ell_{n,m}^U(x) + \ell_{n,m}^H(x) + \ell_{n,m}^S(x).$$

Dabei sind die Summanden gegeben mit $g_0 \in \mathbb{R}^{n \times (m-1)}$, $g_1 \in \mathbb{R}^{(n-1) \times m}$, $H \in \mathbb{R}^{n \times (m-2)}$

und $S \in \mathbb{R}^{(n-2) \times m}$ über

$$\begin{aligned}\ell_{n,m}^g(x) &= \sum_{p=0}^1 \sum_{r=1}^{n-p} \sum_{s=1}^{m-1+p} g_p^{r,s} x_p^{r,s}, \\ \ell_{n,m}^H(x) &= \sum_{r=1}^n \sum_{s=1}^{m-2} H^{r,s} x_0^{r,s} x_0^{r,s+1}, \\ \ell_{n,m}^S(x) &= \sum_{r=1}^{n-2} \sum_{s=1}^m S^{r,s} x_1^{r,s} x_1^{r+1,s}.\end{aligned}$$

Weiterhin ist mit $U \in \mathbb{R}^{(n-1) \times (m-1) \times 2 \times 2}$

$$\begin{aligned}\ell_{n,m}^U(x) &= \sum_{r=1}^{n-1} \sum_{s=1}^{m-1} U_{0,0}^{r,s} x_0^{r,s} x_1^{r,s} + U_{0,1}^{r,s} x_0^{r,s} x_1^{r,s+1} + U_{1,0}^{r,s} x_0^{r+1,s} x_1^{r,s} + U_{1,1}^{r,s} x_0^{r+1,s} x_1^{r,s+1} \\ &= \sum_{r=1}^{n-1} \sum_{s=1}^{m-1} \sum_{p=0}^1 \sum_{t=0}^1 U_{p,t}^{r,s} x_0^{r+p,s} x_1^{r,s+t}.\end{aligned}$$

In der QUBO-Formulierung des kardinalitätsmaximalen Matchings aus 2.3.1 sind die Gewichte der Kanten mit -1 und die Stärken der Kanten im Kantengraph mit 2 gegeben, das heißt $g_p^{r,s} = -1$ und $H^{r,s} = S^{r,s} = U_{t,p}^{r,s} = 2$ für alle entsprechenden p, r, s und t . Für das gewichtsmaximale Matching sind die Gewichte stattdessen über die Bewertungsfunktion gegeben mit $g_e = -S_e$ und die Stärken der Kanten müssen mit einem entsprechenden Straffaktor versehen werden und ergeben P . Allgemein kann damit für das Matching-Problem im Gittergraphen geschlussfolgert werden:

Folgerung 3.16 Für den Graphen $L_{n,m}$ mit der Bewertungsfunktion $S : E(L_{n,m}) \rightarrow \mathbb{R}$, gilt mit $P > \max\{0, S_e : e \in E(L_{n,m})\}$

$$\begin{aligned}\text{MaxMatch}(L_{n,m}, S) &= - \min \left\{ - \sum_{p=0}^1 \sum_{r=1}^{n-p} \sum_{s=1}^{m-1+p} S_p^{r,s} x_p^{r,s} + P \sum_{r=1}^n \sum_{s=1}^{m-2} x_0^{r,s} x_0^{r,s+1} \right. \\ &\quad + P \sum_{r=1}^{n-2} \sum_{s=1}^m x_1^{r,s} x_1^{r+1,s} + P \sum_{r=1}^{n-1} \sum_{s=1}^{m-1} \sum_{p=0}^1 \sum_{t=0}^1 x_0^{r+p,s} x_1^{r,s+t} : \\ &\quad \left. x_0 \in \{0, 1\}^{n \times (m-1)}, x_1 \in \{0, 1\}^{(n-1) \times m} \right\}.\end{aligned}$$

Aufgrund der zum Minor C''_{nm} passenden Struktur des Kantengraphen, lässt sich auch dementsprechend leicht aus Folgerung 3.11 das dazugehörige Chimera-QUBO aufstellen. Die Höhe des Strafwertes kann wieder mit $P_\varepsilon(S, P \mathbb{1}_{E(\mathcal{L}(L_{n,m}))}, L)$ für $\varepsilon > 0$ und obigem $P > \max\{0, S_e : e \in E(L_{n,m})\}$ gefunden werden. Es gilt $L_e = |I_e|$ für alle $e \in E(L_{n,m}) = V(\mathcal{L}(L_{n,m}))$ mit den entsprechenden Indexmengen I , wie für C''_{nm} definiert. Für die Knoten e mit Strängen der Länge 4 gilt daher $L_e - 2 = 2 = \frac{1}{2}|I_e|$ und für die Knoten am Rand des Kantengraphen mit Strängen der Länge 2 ebenso $L_e - 1 = 1 = \frac{1}{2}|I_e|$, womit die Unterscheidung in den Strafbedingungen entfallen kann. Da die Stärken der Kanten auch hier wieder konstant und positiv mit P gegeben sind,

gilt wie bei dem Beispiel der stabilen Menge, dass $\check{\sigma}(e) = 0$ und $\hat{\sigma}(e) \geq P$ für alle $e \in E(L_{n,m})$. Damit ist

$$\begin{aligned} & \max \left\{ 0, \min\{0, S_e, \hat{\sigma}(e)\}, \frac{S_e}{\frac{1}{2}|I_e|} : e \in E(L_{n,m}) \right\} \\ & = \max \{0, S_e : e \in E(L_{n,m})\} < P. \end{aligned}$$

Daher kann bei der Einbettung derselbe Strafwert verwendet werden und es gilt:

Folgerung 3.17 Für den Graphen $L_{n,m}$ mit der Bewertungsfunktion $S : E(L_{n,m}) \rightarrow \mathbb{R}$ gilt mit $P > \max\{0, S_e : e \in E(L_{n,m})\}$ und, dass

$$\begin{aligned} & \text{MaxMatch}(L_{n,m}, S) \\ & = - \min \left\{ \sum_{p=0}^1 \sum_{r=1}^{n-p} \sum_{s=1}^{m-1+p} \left((-S_p^{r,s} + (|I_p^{r,s}| - 1) P) \sum_{k=1}^{|I_p^{r,s}|} q_{I_{p,k}^{r,s}} \right. \right. \\ & \qquad \qquad \qquad \left. \left. + (S_p^{r,s} - |I_p^{r,s}| P) \sum_{k=1}^{|I_p^{r,s}|-1} q_{I_{p,k}^{r,s}} q_{I_{p,k+1}^{r,s}} \right) \right. \\ & \qquad + P \sum_{r=1}^n \sum_{s=1}^{m-2} q_{I_{0,1}^{r,s}} q_{I_{0,1}^{r,s+1}} + P \sum_{r=1}^{n-2} \sum_{s=1}^m q_{I_{1,1}^{r,s}} q_{I_{1,1}^{r+1,s}} \\ & \qquad \left. + P \sum_{r=1}^{n-1} \sum_{s=1}^{m-1} \sum_{p \in \{1,4\}} \sum_{t \in \{1,4\}} q_{0,p}^{r,s} q_{1,t}^{r,s} : q \in \{0,1\}^{(n-1) \times (m-1) \times 2 \times 4} \right\}. \end{aligned}$$

Diese Formulierung benötigt daher für einen Gittergraphen der Größe n mal m einen Chimera-Graphen mit mindestens $n - 1$ Zeilen und $m - 1$ Spalten. Das bedeutet, dieses Problem kann nur bis zu einer Größe von 13×13 Gitterpunkten auf der aktuellen D-Wave-Maschine gerechnet werden.

In der NP-schweren Variante dieses Problems kommt eine Bedingung und damit die Straffunktion p_{\geq} aus (XII) hinzu. Diese erzeugt für jede Kante einen quadratischen Term, der alle Kanten, die aus den Knoten der Kante herausgehen, mit einschließt. Im Kantengraphen des Gittergraphen bedeutet dies, dass über allen Nachbarknoten eines Knotens eine Clique gebildet wird. Da ein Knoten, der im Inneren dieses Graphen liegt, in zwei Vierercliquen liegt, entsteht dadurch eine Siebenerclique, wie sie in Abbildung 36 gezeigt ist. Diese sind aufgrund der Verteilung der Vierercliquen im Graphen zum Teil übereinandergelagert. Das heißt, es wird eine Struktur benötigt die ähnlich zu $V_{n,m}[K_4]$ ist. Jedoch muss dabei in jeder inneren Zelle dieses Graphen jeder Knoten mit seinem Äquivalent in einer der benachbarten Zelle verschmolzen werden, wobei jede der vier Richtungen genau einmal verwendet werden muss. Dadurch bilden diese verschmolzenen Knotenpaare die zentralen Knoten der Siebenercliquen wie in der Abbildung. Da der entstehende Graph jedoch nur wenig anschaulich ist, soll im Folgenden weder dieser gezeigt noch auf die konkrete Problemformulierung im Chimera-Graphen eingegangen werden. Jedoch kann diese bezüglich

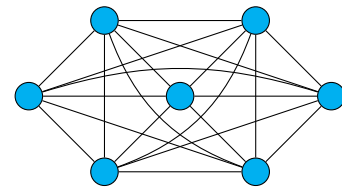


Abb. 36: Siebenerclique

der Zahl der benötigten Einheitszellen abgeschätzt werden: Jede der $n - 2$ mal $m - 2$ Vierercliquen, die im Inneren des Kantengraphen liegen, benötigen eine Zelle von $V_{n-2,m-2}[K_4]$. Das heißt es wird mindestens ein Chimera-Graph mit $n + m - 5$ Zeilen und $n + 2m - 8$ Spalten benötigt. Die fehlenden Knoten am Rand besetzen zudem weitere Qubits. Daher wird die Zahl an Einheitszellen, die für die Einbettung dieses Problems benötigt wird, in $\mathcal{O}(n^2 + m^2 + nm)$ liegen.

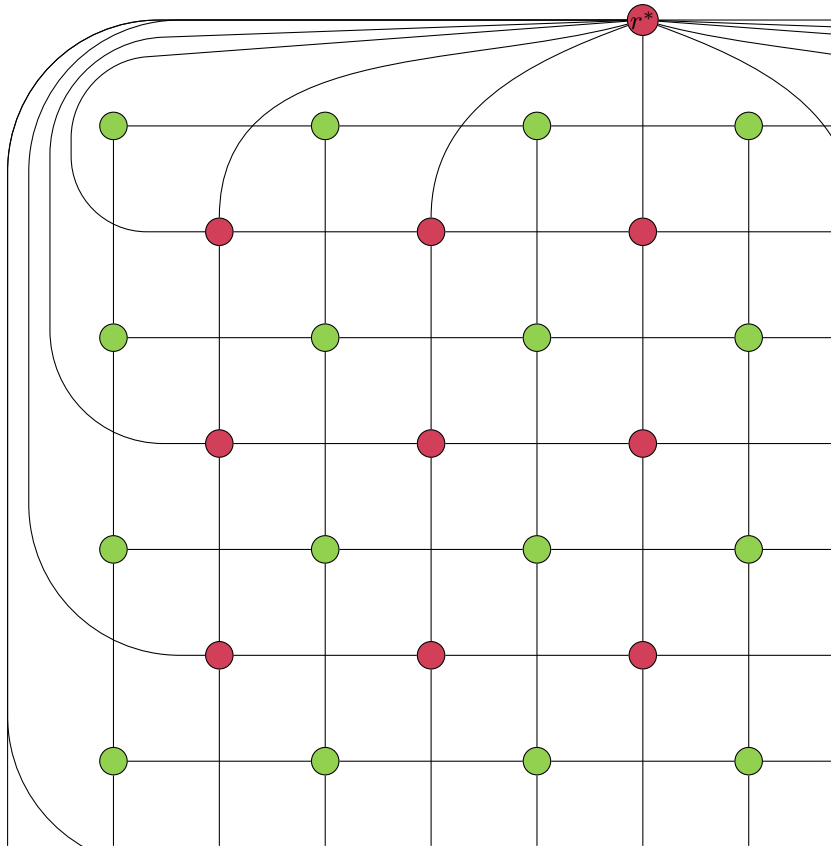
Spannbaum

Auch bei der Suche nach einem aufspannenden Baum in einem Graphen $G = (V, E)$ musste nach Abschnitt 2.3.6 ein Graph über den Kanten gebildet werden. Hinzu kamen hierbei die Kanten des Dualgraphen, wodurch das Problem über $4|E|$ Variablen aufgestellt werden konnte. Im Interaktionsgraphen entstehen dabei durch die Straffunktion (XIII) mehrere kleine Cliques: über den sich kreuzenden Kanten und über den ausgehenden Kanten jedes Knotens. Da diese entscheidend sind für die Einbettungsmöglichkeit in den Chimera-Graphen, soll im Folgenden analysiert werden, wie diese im zugehörigen Interaktionsgraphen des Spannbaumproblems in einem Gittergraphen verteilt sind. In Abbildung 37(a) auf der folgenden Seite ist dazu zunächst ein Gittergraph, in grün, und der zugehörige Dualgraph, in rot, dargestellt. Es zeigt sich, dass der Dualgraph von $L_{n,m}$ mit $n, m > 3$ neben dem zusätzlichen Wurzelknoten ebenfalls einen Gittergraphen enthält. Dieser hat die Größe $(n - 1) \times (m - 1)$.

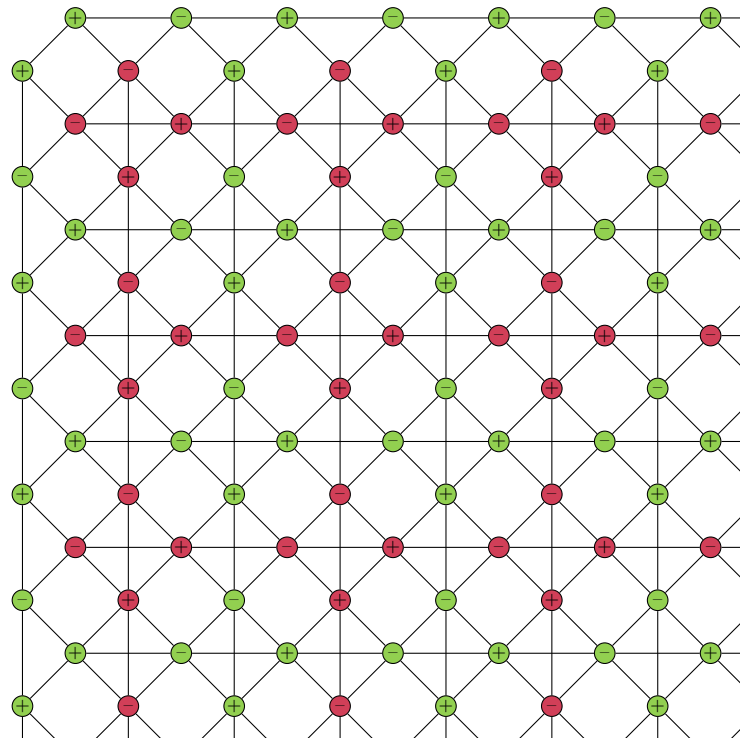
In Abbildung 37(b) wurde der Interaktionsgraph aufgestellt, der zum QUBO des einfachen Spannbaumproblems gehört. Die grünen Knoten zeigen dabei die Kanten des Ausgangsgraphen und die roten die des dualen Graphen an. Außer zum Wurzelknoten r^* ergeben sich dabei zwischen zwei Knoten immer eine „+“- und eine „-“-Kante, wobei „+“ für die vom linken zum rechten Knoten beziehungsweise vom oberen zum unteren Knoten herauszeigende Kante im Gittergraphen und „-“ für die entsprechende umgekehrte Kante steht. Die linken und oberen roten „-“-Knoten stehen für die in den Wurzelknoten zeigenden Kanten des Dualgraphen. Da aus der Wurzel keine Kanten hinauszeigen können, fehlen die entsprechenden „+“-Knoten der umgekehrten Richtung.

Insgesamt zeigt sich, dass dieser Graph wie beim Matching-Problem gitterartig Vierercliquen miteinander verbindet und damit ebenso einem Teilgraphen von C'' entspricht. Demzufolge lässt sich die Zielfunktion des QUBOs analog zum vorherigen Abschnitt aufstellen. Jedoch wird für die Suche nach einem Spannbaum im Gittergraphen $L_{n,m}$ aufgrund der Hinzunahme des Dualen der Graph $\tilde{L}_{2(n-1),2(m-1)}$ als Grundlage für den Interaktionsgraphen benötigt. Das bedeutet bei der Einbettung in den Chimera-Graphen muss dieser mindestens eine Größe von $2(n - 1)$ Zeilen und $2(m - 1)$ Spalten besitzen.

Wird nun eine NP-schwere Variante dieses Problems betrachtet, muss der gezeigte Interaktionsgraph erweitert werden. Die einzige Variante bei der keine weiteren Knoten hinzukommen, stellt die Suche nach einem Hamiltonpfad, also die zusätzliche Beschränkung des Eingangsgrades auf 1 durch die Straffunktion (XIV), dar. Dadurch entsteht für jeden Knoten eine weitere Clique im Interaktionsgraphen. Eine solche ist beispielhaft für einen Knoten in Abbildung 38 auf der Seite 81 gezeigt. Wie dabei festgestellt werden kann, überspannt diese die ursprünglich zum Knoten gehörende Clique. Werden diese Kanten für jeden Knoten des Ausgangsgraphen hinzugefügt, entsteht ein



(a) Gitter $L_{n,m}$ und dessen Duales



(b) Interaktionsgraph zu $L_{n,m}$ und seinem Dualen

Abb. 37: Spannbaumproblem im Gittergraphen

Graph mit mehreren sich überlappenden Cliques. Dies ist in Abbildung 39 dargestellt. Durch diese und die zusätzlichen Knoten der Kanten des dualen Graphen sind hierbei keine Ähnlichkeiten zu den vorher gezeigten Minoren erkennbar. Daher kann an dieser Stelle keine Einbettung in den Chimera-Graphen gezeigt werden.

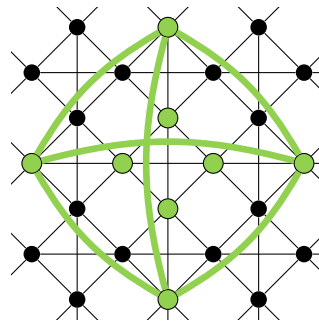


Abb. 38: Zusätzliche Kanten für einen Knoten bei Hamiltonpfadproblem

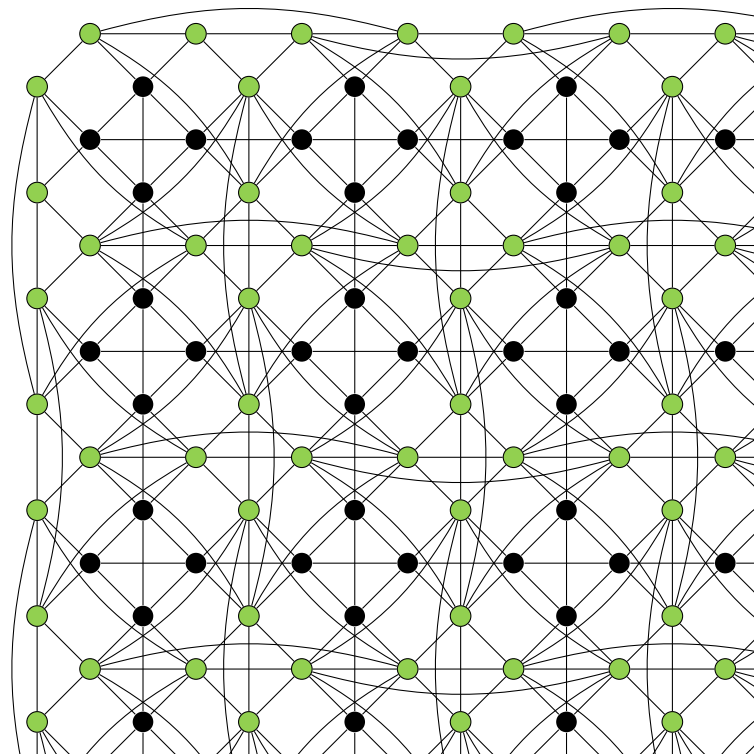


Abb. 39: Zusätzliche Kanten im Interaktionsgraphen

4 Satellitenzeitplanung

Insbesondere bei Weltraummissionen ist die Vorabplanung von essentieller Bedeutung. Der erfolgreiche Betrieb eines Raumfahrzeuges hängt stark davon ab, wie gut dessen Komponenten zusammenarbeiten. Auch die Missionsziele und deren Erreichbarkeit müssen evaluiert werden. Daher hat das DLR das Projekt „Virtueller Satellit“ geschaffen, in dem viele unterschiedliche Disziplinen zusammengeführt werden. Dabei werden unter anderem Simulationen zur geplanten Flugbahn des Satelliten durchgeführt. Bei den mehrfachen Erdumrundungen ergeben sich Randbedingungen, unter denen der Satellit dennoch seine Arbeit verrichten können muss.

Daraus lässt sich ein Optimierungsproblem aufstellen, welches im folgenden Abschnitt näher betrachtet wird. Dabei wird grundsätzlich der Problemdarstellung, wie sie in [42] vorgestellt wird, gefolgt. Die darauffolgenden Modelle stellen Möglichkeiten dar, dieses zu vereinfachen. Zunächst wird dafür die Zeit diskretisiert. Anschließend wird eine lineare Variante dieses Problems betrachtet. Zuletzt wird das Problem in ein QUBO umgewandelt und dessen Größe abgeschätzt. Damit lässt sich die Lösbarkeit mittels des eingeschränkten adiabatischen Quantenrechners von D-Wave Systems evaluieren. Am Ende dieses Kapitels ist eine Zusammenfassung der kompakten formalen Darstellungen aller Probleme zu finden.

4.1 Kontinuierliches Modell

4.1.1 Allgemeines

Die Satellitenmission ist für einen bestimmten Zeitraum angesetzt, beginnt also bei Startzeit 0 und endet zu einem festen Zeitpunkt $T \in \mathbb{R}_{>0}$. Dabei umfliegt der Satellit auf einer vorher definierten Bahn die Erde. Seine Aufgaben verrichtet der Satellit, in dem er verschiedene Zustände einnimmt. Zu diesen gehören beispielsweise das Laden der Akkus oder Senden von Daten zu einer Bodenstation. Die Daten werden durch verschiedene Experimente gesammelt, die der Satellit ausführen kann, wie zum Beispiel Sternbilder oder Sonnenflecken fotografieren. Ungünstig aber möglicherweise notwendig dabei ist auch der Leerlauf. Damit ergibt sich eine endliche Menge von Zuständen, wie $S = \{\text{Idle, Link, Charge, ExpI, ExpII}\}$. Dabei kann von jedem Zustand in einen anderen gewechselt werden. In der Realität ist es auch möglich, dass sich zwei Zustände überschneiden, wie gleichzeitiges Laden und Experimentieren. Im Folgenden wird sich auf die vereinfachte Variante, dass der Satellit zu jedem Zeitpunkt nur genau einen Zustand annehmen kann, beschränkt. Ebenso wird eine Mindestdauer für einen Zustand auf einen festen Wert $d_s \in \mathbb{R}$ für alle $s \in S$ festgelegt. Dies entspricht beispielsweise Warmlauf- oder Hochfahrzeiten und gewährleistet, dass der Satellit nicht zu viel zwischen den Zuständen springt.

Resultat der Optimierung soll am Ende ein Plan für den Satelliten sein, was zu welchem Zeitpunkt ausgeführt wird. Das heißt, gesucht ist eine Abbildung $P : S \rightarrow \mathcal{P}([0, T])$ mit

$$\bigcup_{s \in S} \bigcup_{I \in P(s)} I = [0, T]. \quad (\text{K1})$$

Dabei müssen alle Intervalle die Mindestdauer erfüllen:

$$t_1 - t_0 \geq d_s \quad \forall [t_0, t_1] \in P(s) \quad \forall s \in S. \quad (\text{K2})$$

Hierbei wird davon ausgegangen, dass der letzte Zustand vollständig abgeschlossen wird, also auch die Mindestdauer gelaufen ist. Das heißt für das $s \in S$ mit $[t_0, T] \in P(s)$ gilt, dass $t_0 \leq T - d_s$. Dies kann abgeschwächt werden, indem $t_1 \neq T$ bei (K2) gefordert wird.

4.1.2 Umgebungsbedingungen

Bestimmte Zustände können überhaupt erst angenommen werden, wenn einige Bedingungen erfüllt sind. So kann der Akku nur geladen werden, wenn die Solarpaneele ins Sonnenlicht gedreht werden können und sich der Satellit nicht im Erdschatten befindet. Begibt er sich in einen bestimmten Bereich über einer Bodenstation, so ist es ihm erst dann möglich, Daten an diese zu senden. Daher ergibt sich für jeden Zustand eine endliche Menge an disjunkten Zeitabschnitten aus dem Intervall $[0, T]$, in denen dieser Zustand angenommen werden kann:

$$B_s = \bigcup_{k=1}^{K_s} I_k \subset 2^{[0, T]}.$$

Damit muss für alle Zustände $s \in S$ für die Funktion P gelten:

$$\forall I \in P(s) \exists J \in B_s : I \subseteq J. \quad (\text{K3})$$

Dementsprechend können bereits vor der Optimierung alle Intervalle aus B_s ausgeschnitten werden, die kürzer als die Mindestdauer d_s sind.

4.1.3 Gerätebedingungen

Der Satellit besitzt zudem einen Akku mit einer nach oben und unten begrenzten Kapazität $C_{\min}, C_{\max} \in \mathbb{R}_{\geq 0}$ mit $C_{\min} < C_{\max}$ und eine Festplatte zum Speichern von Daten, deren Größe ebenso begrenzt ist durch $M_{\min}, M_{\max} \in \mathbb{R}_{\geq 0}$ mit $M_{\min} < M_{\max}$. Begonnen wird die Mission mit einem mit Kapazität $C_0 \in [C_{\min}, C_{\max}]$ geladenem Akku und einer Festplatte mit belegtem Speicher $M_0 \in [M_{\min}, M_{\max}]$. Jeder Zustand zieht oder erzeugt eine gewisse Lade- und Speicherkapazität. Während Experimente Daten auf die Festplatte laden, können durch das Senden zur Bodenstation Daten gelöscht werden. Daraus ergeben sich konstante Raten $c_s, m_s \in \mathbb{R}$ mit

$$c_s \leq 0 \quad \forall s \in S \setminus \{\text{Charge}\},$$

$$\begin{aligned}
 c_{\text{Charge}} &> 0, \\
 m_{\text{ExpI}}, m_{\text{ExpII}} &> 0, \\
 m_{\text{Link}} &< 0, \\
 m_{\text{Charge}} = m_{\text{Idle}} &= 0.
 \end{aligned}$$

Wird beispielsweise über eine Dauer $d \in \mathbb{R}_{\geq 0}$ geladen, ergibt sich eine Erhöhung des Akkustandes um $d \cdot c_{\text{Charge}}$. Sinnvollerweise sollte daher auch für jeden Zustand $s \in S$ gelten

$$\begin{aligned}
 d_s |c_s| &\leq C_{\max} - C_{\min}, \\
 d_s |m_s| &\leq M_{\max} - M_{\min}.
 \end{aligned}$$

Sollte das Systemmodell um weitere Zustände erweitert werden, können deren Kosten analog gegeben werden.

Zu jedem Zeitpunkt muss aber nun gewährleistet sein, dass die Kapazität nie unter den Minimal- und nie über den Maximalwert tritt. Für alle $t \in [0, T]$ sind also die folgenden Bedingungen einzuhalten:

$$C_{\min} \leq C_0 + \sum_{s \in S} c_s \sum_{\substack{[t_0, t_1] \in P(s) \\ t_1 \leq t}} (t_1 - t_0) \leq C_{\max}, \quad (\text{K4})$$

$$M_{\min} \leq M_0 + \sum_{s \in S} m_s \sum_{\substack{[t_0, t_1] \in P(s) \\ t_1 \leq t}} (t_1 - t_0) \leq M_{\max}. \quad (\text{K5})$$

Da eine Änderung der Summe erst bei einem Wechsel des Zustands auftritt, können die Bedingungen auf die Intervallendpunkte $t \in [0, T]$ mit $\exists s \in S \exists I \in P(s) : I = [t_0, t]$ begrenzt werden.

4.1.4 Ziele

Nun werden mögliche Missionsziele in das Optimierungsproblem eingebracht. Diese können stark variieren unter den zu beachtenden Aspekten und Ansprüchen der verschiedenen zusammenarbeitenden wissenschaftlichen Disziplinen. Im Gesamtkonzept spielen beispielsweise auch die Lebensdauer der Bauteile eine Rolle, so dass beispielsweise der Akku nicht zu oft nur kurz geladen oder die Kamera nicht zu oft in der Sonne benutzt werden soll, da diese sonst Schaden nehmen könnten. Die Wahl der Prioritäten liegt dabei bei den verantwortlichen Ingenieuren. Im Folgenden werden verschiedene Aspekte des Sammelns von Daten betrachtet.

Zum einen ist es möglich, die Anforderungen an den Satelliten in Form von zusätzlichen Bedingungen einzubringen: Sollen beispielsweise zum Experiment I nicht mehr als eine Datenmenge von $M_{\text{ExpI}} \in \mathbb{R}_{>0}$ gesammelt werden, ergibt sich die Ungleichung

$$m_{\text{ExpI}} \sum_{[t_0, t_1] \in P(\text{ExpI})} (t_1 - t_0) \leq M_{\text{ExpI}}.$$

Das Senden einer gewissen Gesamtdatenmenge $M_{\text{all}} \in \mathbb{R}_{>0}$ lässt sich folgendermaßen

erzwingen:

$$-m_{\text{Link}} \sum_{[t_0, t_1] \in P(\text{Link})} (t_1 - t_0) \geq M_{\text{all}}.$$

Dabei ist es jedoch möglich, dass keine Optimallösung existiert, also kein Zeitplan, bei dem diese Datenmenge überhaupt gesammelt werden kann. Deswegen kann auch direkt darüber optimiert und entsprechend eine Zielfunktion f , die abhängig von der Funktion P ist, mit

$$f(P) = m_{\text{Link}} \sum_{[t_0, t_1] \in P(\text{Link})} (t_1 - t_0)$$

gewählt werden. Damit lässt sich das allgemeine Optimierungsproblem folgendermaßen formulieren:

$$\min \{f(P) : P : S \rightarrow 2^{[0, T]} \text{ erfüllt (K1) - (K5)}\}.$$

Des Weiteren könnte eine Forderung lauten, dass die Laufzeit eines Zustands wie dem Leerlauf minimiert wird. Dies verursacht Kosten, ohne dass ein Ertrag gebracht wird. Weitere Ziele können mit einer entsprechenden Gewichtung hinzugefügt werden. Dann ist zum Beispiel mit $A, B \in \mathbb{R}_{>0}$

$$\tilde{f}(P) = Af(P) + B \sum_{[t_0, t_1] \in P(\text{Idle})} (t_1 - t_0)$$

die zugehörige Zielfunktion. Die hier aufgezeigten Zielanforderungen ließen sich durch lineare Funktionen beschreiben. Dennoch sind weitere und möglicherweise komplexere Variationen ebenso denkbar.

4.1.5 Parameter aus technischen Voraussetzungen

In diesem Abschnitt werden nun beispielhaft einige Parameter, wie die Raten oder Kapazitäten, die in dem Problem aufgetaucht sind, aufgeführt. Dies dient der Veranschaulichung der Größenordnungen, in denen sich die Ingenieure bei der Planung einer Mission ungefähr bewegen. Dabei wurde sich stark an einer Beispielimplementierung aus [42] orientiert, welche einen direkten Bezug zum Technologieerprobungsträger TET-1 nimmt, einem Mikrosatelliten des DLR, der seit 2012 die Erde mit einer Umlaufzeit von 95min umrundet.

$$\begin{aligned} T &= 365\text{d} = 525600 \text{ min}, & C_{\text{min}} &= 40\text{Ah} = 2400\text{Amin}, \\ C_{\text{max}} &= 240\text{Ah} = 14400\text{Amin}, & C_{\text{Start}} &= 240\text{Ah} = 14400\text{Amin}, \\ M_{\text{min}} &= 0\text{MB}, & M_{\text{max}} &= 512\text{MB}, \\ M_{\text{Start}} &= 0\text{MB}, & c_{\text{Idle}} &= -0.5\text{A}, \\ c_{\text{Link}} &= -1\text{A}, & c_{\text{Charge}} &= 10\text{A}, \\ c_{\text{ExpI}} &= -1\text{A}, & c_{\text{ExpII}} &= -1\text{A}, \\ m_{\text{Idle}} &= 0, & m_{\text{Link}} &= -2, 2 \frac{\text{Mbit}}{\text{s}} = -16, 5 \frac{\text{MB}}{\text{min}}, \\ m_{\text{Charge}} &= 0, & m_{\text{ExpI}} &= \frac{1}{6} \frac{\text{MB}}{\text{min}}, \\ m_{\text{ExpII}} &= 3 \frac{\text{MB}}{\text{min}}. \end{aligned}$$

4.1.6 Herangehensweise

Das gezeigte Problem ist kein Optimierungsproblem im klassischen Sinne. Statt über einer festen Zahl an Variablen wird über einer Funktion optimiert, die eine Zuordnung von der diskreten Zustandsmenge zu dem kontinuierlichen Zeitintervall liefern soll. Zudem ergeben sich sehr viele Nebenbedingungen, deren genaue Zahl ebenfalls von dieser Funktion abhängen. Daher muss ein anderer Weg gefunden werden, wie an das Problem herangegangen werden kann.

Eine Möglichkeit stellt die Umformulierung zu Arbeitspaketen dar, wenn für jedes Experiment $\text{Exp} \in \{\text{ExpI}, \text{ExpII}, \dots\}$ eine Mindestdatenmenge $M_{\text{Exp}} > 0$ gegeben ist. Das heißt die Bedingung

$$m_{\text{Exp}} \sum_{[t_0, t_1] \in P(\text{Exp})} (t_1 - t_0) \geq M_{\text{Exp}}$$

muss erfüllt werden. Dabei ergibt sich aus der Mindestdatenmenge eine Mindestlaufzeit $T_{\text{Exp}} > 0$ durch die gegeben konstanten Raten mit $M_{\text{Exp}} = m_{\text{Exp}} T_{\text{Exp}}$. Mit der Mindestdauer d_{Exp} , die ein Zustand laufen muss, wenn er einmal gestartet wurde, ergeben sich damit $\lceil T_{\text{Exp}} d_{\text{Exp}}^{-1} \rceil$ Arbeitspakete zu diesem Experiment, die in der Mission abgeleistet werden müssen. Solche Arbeitspakete lassen sich für jedes Experiment definieren. Sie müssen dabei nicht zwingend die gleiche Dauer haben. Damit lässt sich eine Menge A von Arbeitspaketen aufstellen, wobei $d_a > 0$ die Dauer und $c_a > 0$ der Stromverbrauch der Aufgabe $a \in A$ ist. Nun ist die Frage, ob alle diese Aufgaben von dem Satelliten während seiner Mission erfüllt werden können. Die gewünschte Datenmenge, welche gesammelt werden soll, verbirgt sich dabei in den Arbeitspaketen.

Da der Satellit bei seinem Flug um die Erde einer festen Bahn folgt, könnte von einer gewissen Periodizität der Umgebungsbedingungen ausgegangen werden. Dementsprechend wäre möglicherweise ein fester wiederkehrender Zeitpunkt wählbar, an dem der Akku wieder voll geladen wird, beispielsweise jedes Mal, wenn der Satellit wieder in der Sonne ist. Das dazwischenliegende Zeitintervall von Ende des Ladens bis zum erneuten Start habe eine feste Länge von $T_{\text{Work}} \in \mathbb{R}_{>0}$. Unter diesen Bedingungen lässt sich die Gesamtlaufzeit T der Mission in $r \in \mathbb{N}$ solcher Intervalle unterteilen. Ob die Arbeitspakete alle in den r Intervallen zwischen den Ladungen ausgeführt werden können, kann durch ein Zulässigkeitsproblem gelöst werden. Äquivalent dazu kann die Frage beantwortet werden, wie viele solcher Intervalle werden mindestens benötigt, damit die Mission ihre Ziele erreicht. Analog zum Bin-Packing-Problem aus 2.3.6 werden hierfür Variablen

$$X_{a,k} = \begin{cases} 1, & \text{wenn Arbeitspaket } a \text{ im } k\text{-ten Intervall ausgeführt wird,} \\ 0, & \text{sonst} \end{cases}$$

für $a \in A$ und $k \in \{1, \dots, r\}$ benötigt. Diese müssen die Bedingungen

$$\begin{aligned} \sum_{a \in A} d_a X_{a,k} &\leq T_{\text{Work}}, \\ \sum_{a \in A} c_a X_{a,k} &\leq C_{\text{max}} - C_{\text{min}} \end{aligned}$$

für alle $k \in \{1, \dots, r\}$ und

$$\sum_{k=1}^r X_{a,k} = 1$$

für alle $a \in A$ erfüllen. Wird das Problem allein mit diesen Nebenbedingungen betrachtet, stellt es bereits eine Erweiterung des Bin-Packing-Problems dar, wodurch es NP-schwer ist. Dabei wurden jedoch die Bedingungen an den Speicher bisher nicht beachtet. Könnte dieser ebenfalls durch den Downlink außerhalb der Arbeitszeit des Satelliten komplett geleert werden, würde die Bedingung

$$\sum_{a \in A} m_a X_{a,k} \leq M_{\max} - M_{\min}$$

hinzukommen. Dies stellt aber eine starke Vereinfachung des Problems dar, da die Zeit, in der geladen werden kann, oft nicht synchron zu der liegt, in der Kontakt zur Bodenstation aufgenommen werden kann. Daher müssten weitere Veränderungen an dieser Formulierung vorgenommen werden. Sollen zudem die Zeiten für das Laden und das Senden der Daten flexibel bleiben, bietet sich eine andere Möglichkeit an, die Diskretisierung der Zeit, welche in dem folgenden Abschnitt gezeigt wird.

4.2 Diskretisiertes Modell

4.2.1 Allgemeines

Um das Problem besser in den Griff zu bekommen, bietet sich die Möglichkeit, die Zeit zu diskretisieren. Dies bedeutet die Gesamtzeit wird in endlich viele Zeitabschnitte der gleichen Länge unterteilt und diese entsprechend durchnummeriert mit $t \in \{1, \dots, T\}$ und $T \in \mathbb{N}$. Zusammen mit der Menge der Zustände S kann das Problem nun über einer festen Zahl an Variablen aufgestellt werden. Diese seien $x \in \{0, 1\}^{S \times T}$ mit

$$x_{s,t} = \begin{cases} 1, & \text{wenn der Satellit zum Zeitpunkt } t \text{ in Zustand } s \text{ ist,} \\ 0, & \text{sonst.} \end{cases}$$

Gesucht ist dann bei der Optimierung eine Belegung dieser Variablen, die ebenfalls einem Plan für den Satelliten entspricht. Analog zu (K1) kann sich der Satellit zu jedem Zeitpunkt in nur genau einem Zustand befinden. Damit ergibt sich für alle $t \in \{1, \dots, T\}$ die Bedingung

$$\sum_{s \in S} x_{s,t} = 1. \quad (\text{D1})$$

Entsprechend (K2) kann eine Mindestdauer für jeden Zustand angegeben werden. Für diese gilt jedoch nun $d_s \in \mathbb{N} \forall s \in S$ und bezieht sich auf die Anzahl der aufeinanderfolgenden Zeitabschnitte. Die Einhaltung der Mindestdauer erfordert weitere Bedingungen an die x -Variablen: Wenn $x_{s,t-1} = 0$ und $x_{s,t} = 1$ sind, dann markiert der Zeitpunkt t den Anfang des Zustandes $s \in S$. Dementsprechend muss der Satellit die folgenden $d_s - 1$ Zeitpunkte ebenfalls im Zustand s bleiben. Daraus ergibt sich die Bedingung

$x_{s,t+1} = \dots = x_{s,t+d_s-1} = 1$, welche genau dann erfüllt ist, wenn

$$\prod_{j=t+1}^{t+d_s-1} x_{s,j} = 1.$$

Wegen

$$\neg [a \wedge b \Rightarrow c] = [(a \wedge b) \wedge (\neg c)] = ab(1 - c) \stackrel{!}{=} 0$$

folgt wie im Abschnitt 2.3.4

$$(1 - x_{s,t-1}) x_{s,t} \left(1 - \prod_{j=t+1}^{t+d_s-1} x_{s,j} \right) = 0 \quad \forall t \in \{1, \dots, T - d_s + 1\} \quad \forall s \in S. \quad (\text{D2})$$

Diese multilinearen Polynome vom Grad $d_s + 1$ schließen noch nicht die Möglichkeiten ein, ob ein Zustand das Zeitende überschreiten darf oder ob er nicht mehr gestartet werden kann, sollte dies durch die Mindestdauer erzwungen werden. Trifft ersteres zu kann (D2) einfach um

$$(1 - x_{s,t-1}) x_{s,t} \left(1 - \prod_{j=t+1}^T x_{s,j} \right) = 0 \quad \forall t \in \{T - d_s + 2, \dots, T\} \quad (\text{D2a})$$

erweitert werden. Andernfalls gilt, ist $x_{s,T-d_s+1} = 0$, dann müssen auch alle $x_{s,t} = 0$ für $t = T - d_s + 2, \dots, T$ sein. Daraus ergibt sich

$$(1 - x_{s,T-d_s+1}) \left(\sum_{t=T-d_s+2}^T x_{s,t} \right) = 0. \quad (\text{D2b})$$

Es zeigt sich also, dass diese zunächst im kontinuierlichen Fall sinnvoll erscheinende Maßnahme des Einfügens einer Mindestdauer im diskreten Fall schwierigere Nebenbedingungen hervorbringt.

4.2.2 Umgebungs- und Gerätebedingungen

Aus den feststehenden Bedingungen kann auch für die endlich vielen Zeitabschnitte festgelegt werden, ob in diesen ein Zustand einnehmbar ist oder nicht. Daher ergeben sich analoge Zeiteilmengen

$$B_s \subseteq \{1, \dots, T\}$$

für jeden Zustand $s \in S$. Entsprechend (K3) gilt für einen Zeitpunkt $t \in \{0, \dots, T\}$ mit $t \notin B_s$, dass zu diesem Zeitpunkt s nicht angenommen werden kann:

$$x_{s,t} = 0 \quad \forall t \in \{1, \dots, T\} \setminus B_s \quad \forall s \in S. \quad (\text{D3})$$

Diese Variablen können daher aus dem Optimierungsproblem eliminiert werden. Ebenso wie beim kontinuierlichen Modell können auch hier aufeinanderfolgende Zeitabschnitte, deren Zahl geringer ist als d_s , entfernt werden.

Auch bei den Gerätebedingungen zeigt sich der Vorteil der Diskretisierung. Analog zu den Akku- und Speichergrenzen wie in (K4) und (K5) lassen sich die folgenden Ungleichungen für alle Zeitabschnitte aufstellen:

$$C_{\min} \leq C_0 + \sum_{s \in S} c_s \sum_{\tau=1}^t x_{s,\tau} \leq C_{\max} \quad \forall t \in \{1, \dots, T\}, \quad (\text{D4})$$

$$M_{\min} \leq M_0 + \sum_{s \in S} m_s \sum_{\tau=1}^t x_{s,\tau} \leq M_{\max} \quad \forall t \in \{1, \dots, T\}. \quad (\text{D5})$$

Dabei müssen nur die Raten $c_s, m_s \in \mathbb{R}$ entsprechend der Länge der Zeitabschnitte angepasst werden.

4.2.3 Ziel

Wie auch im vorherigen Abschnitt können die Ziele zum einen als Bedingungen und zum anderen direkt in die Zielfunktion eingefügt werden. Diese lässt sich nun aber über eine Funktion in den x -Variablen formulieren und so das allgemeine Optimierungsproblem

$$\min \{f(x) : x \in \{0, 1\}^{S \times T} \text{ erfüllt (D1) - (D5)}\}$$

aufstellen. Dabei kann ebenso mit der entsprechenden Gewichtung $A, B > 0$ zum Beispiel der Downlink maximiert werden, während die Leerlaufzeit minimiert wird:

$$f(x) = -A \cdot m_{\text{Link}} \sum_{t=1}^T x_{\text{Link},t} + B \sum_{t=1}^T x_{\text{Idle},t}.$$

Eine Übertragung der zusätzlichen Bedingungen, wie beispielsweise die Datenobergrenze $M_{\text{ExpI}} \in [M_{\min}, M_{\max}]$ für das erste Experiment, ist ebenso unproblematisch:

$$-m_{\text{ExpI}} \sum_{t=1}^T x_{\text{ExpI},t} \leq M_{\text{ExpI}}.$$

4.2.4 Analyse

Für dieses Modell werden zunächst $|S| \cdot T$ binäre Variablen benötigt. Durch die Umgebungsbedingungen (D3) können einige dieser Variablen eliminiert werden und die Zahl sinkt auf

$$\sum_{s \in S} |B_s|.$$

Diese werden, entsprechend der angegebenen Beispiele, über einer linearen Zielfunktion optimiert. (D4) und (D5) stellen für alle $t \in \{1, \dots, T\}$ jeweils zwei lineare Nebenbedingungen an das Optimierungsproblem dar. Zusammen mit (D1) ergeben sich damit insgesamt $5T$ Nebenbedingungen. Es kommen weitere hinzu, sollten Zielanforderungen in Bedingungen gefasst werden. Des Weiteren besitzt dieses Optimierungsproblem

entweder durch (D2a) $|S| \cdot T$ oder durch (D2b)

$$\sum_{s \in S} (T - d_s + 2)$$

nichtlineare Nebenbedingungen. Bereits lineare ganzzahlige Optimierungsprobleme sind im Allgemeinen NP-schwer. Die nichtlinearen Nebenbedingungen führen zudem dazu, dass der Definitionsbereich der Variablen vermutlich nicht mehr konvex ist. Daher kann dieses Problem nur mit den sehr aufwändigen Methoden der nichtkonvexen Optimierung, möglicherweise auch nur lokal, gelöst werden.

4.2.5 Erweiterungen

Eine Möglichkeit, die sich im diskreten Modell bietet, ist die zeitliche Veränderung einiger gegebener Parameter. Dies hätte im kontinuierlichen Modell zu Bedingungen mit Integralen geführt. Der Bezug zu realen Gegebenheiten zeigt sich beispielsweise dadurch, dass Akkus nach einiger Zeit an Kapazität verlieren oder elektronische Bauteile durch kosmische Strahlung geschädigt werden. Dementsprechend kann sich für die maximale Kapazität eine positive, monoton fallende Funktion $C_{\max} : \{1, \dots, T\} \rightarrow \mathbb{R}_{>0}$ mit $C_{\max}(1) < \infty$ ergeben. Ebenso lassen sich Fehler in der Festplatte mit M_{\max} oder auch die Raten bezüglich der Zeit angeben. Für (D4) folgt damit beispielsweise

$$C_{\min} \leq C_0 + \sum_{s \in S} \sum_{\tau=1}^t c_s(\tau) x_{s,\tau} \leq C_{\max}(t) \quad \forall t \in \{1, \dots, T\}.$$

4.3 Vereinfachtes lineares Modell

Was für das kontinuierliche Modell sinnvoll war, damit nicht zu viele Sprünge zwischen den Zuständen entstehen, verkompliziert jedoch das diskretisierte Modell stark: die Mindestdauer eines Zustandes. Wählt man stattdessen eine Mindestdauer $d_s = 1$ für alle $s \in S$, so entfällt die Bedingung (D2). Da diese als einzige nichtlinear ist, wird dadurch sogar das komplette System linear.

Weiterhin bietet sich eine Möglichkeit die Zahl der Variablen, über denen optimiert wird, zu reduzieren: In den meisten Modellen kann der Satellit zu jedem Zeitpunkt einfach nichts tun. Das heißt, die Zahl der binären Variablen $x_{\text{Idle},t}$ ist genau T . Wird der Leerlaufzustand entfernt und stattdessen die Möglichkeit geschaffen, keinen Zustand einzunehmen, können diese eingespart werden. Dies bedeutet, statt (D1) ist die folgende Bedingung zu erfüllen:

$$\sum_{s \in S} x_{s,t} \leq 1. \tag{L1}$$

Mit dieser Bedingung bleibt das System auch linear. Sollte für einige Zeitpunkte jedoch mal kein Idle-Zustand verfügbar sein, lässt sich dies weiterhin durch ein Gleichheitszeichen in (L1) erzwingen. Die Bedingung (D3) kann dagegen als (L2) übernommen werden.

Um ebenso den Stromverbrauch im Leerlauf zu berücksichtigen, kann eine Grundlast $c_{\text{Base}} \in \mathbb{R}_{\geq 0}$ eingeführt werden. Mit $c_{\text{Base}} = -c_{\text{Idle}}$ und $\tilde{c}_s = c_s + c_{\text{Base}}$ für alle $s \in S$ entspricht

$$C_{\min} \leq C_0 - c_{\text{Base}} \cdot t + \sum_{s \in S} \tilde{c}_s \sum_{\tau=1}^t x_{s,\tau} \leq C_{\max} \quad \forall t \in \{1, \dots, T\} \quad (\text{L3})$$

wegen

$$\sum_{\tau=1}^t c_{\text{Idle}}(1 - x_{\text{Idle},\tau}) = \sum_{s \in S \setminus \{\text{Idle}\}} c_{\text{Idle}} \sum_{\tau=1}^t x_{s,\tau}$$

genau der Bedingung (D4). Damit die Raten weiterhin den technischen Voraussetzungen mit $\tilde{c}_s \leq 0 \quad \forall s \in S \setminus \{\text{Charge}\}$ entsprechen, muss $c_{\text{Base}} \leq |c_s|$ für alle außer dem Ladezustand gelten. Für diesen muss $c_{\text{Charge}} + c_{\text{Base}} \leq C_{\max} - C_{\min}$ gelten. Eine analoge Änderung bei der Speicherbedingung ist nicht nötig, da im Leerlauf der Speicher nicht angefasst wird und somit m_{Idle} Null ist. Damit ist (L4) gleich (D5).

4.4 Abgeleitetes quadratisches Modell ohne Nebenbedingungen

In diesem Abschnitt soll nun zu dem vorgestellten Problem ein QUBO hergeleitet werden. Dies wäre durchaus für das nichtlineare diskrete Modell möglich, in dem durch Reduktion der Grad verringert wird. Im folgenden soll sich jedoch auf die vereinfachte lineare Variante konzentriert werden. In diesem Fall können alle der vorgeschlagen Möglichkeiten für die Zielfunktion f über den Variablen $x \in \{0, 1\}^{S \times T}$ uneingeschränkt übernommen werden, da sie einen weniger als quadratischen Grad besitzen. Jedoch müssen die Freiheiten der gegebenen Parameter eingeschränkt werden, damit die Gleichungen und Ungleichungen aus den Bedingungen durch binäre Variablen realisiert werden können. Anschließend werden diese durch die bereits vorgestellten Techniken umgeformt. Letztendlich wird die Größe des entstandenen QUBOs abgeschätzt und damit die Berechenbarkeit dieses Problems auf dem D-Wave-Systems Chip evaluiert.

4.4.1 Einschränkungen für die Parameter

Aufgrund der Verwendung von binären Variablen ist es zwar möglich Gleitkommazahlen darzustellen, dies erfordert jedoch großen Aufwand und besonders eine hohe Zahl an binären Variablen. Um die QUBO-Formulierung daher so klein wie möglich zu halten, muss sich zum einen für die Kapazitäten $C_{\min}, C_{\max}, C_0, M_{\min}, M_{\max}$ und M_0 auf natürliche Zahlen beschränkt werden. Zum anderen dürfen auch die Raten c_s, m_s für alle Zustände $s \in S$ nur ganzzahlig sein, um eine endliche Zahl an Summationsergebnissen in den Bedingungen (L3) und (L4) zu erhalten und damit die entsprechende Darstellung der Bedingungen in Straffunktionen über binären Variablen zu gewährleisten.

Dementsprechend liefert dieses Problem nur noch eine starke Vereinfachung des eigentlichen Modells. Zudem verringern sich die Variationsmöglichkeiten für unterschiedliche

Zustände. Dabei eignet sich aber dennoch für eine grobe Abschätzung der Erreichbarkeit von Missionszielen des Satelliten.

4.4.2 Nebenbedingungen

Die Bedingungen (L1), (L3) und (L4) können nun wie im Kapitel 2 entsprechend umgeformt werden. Dabei ergibt sich für (L1)

$$p_{(L1)}(x) = \sum_{s \in S} \left(\sum_{t=1}^T x_{s,t} - \frac{1}{2} \right)^2 - \frac{1}{4}|S| = \sum_{s \in S} \sum_{\substack{t_1, t_2=1 \\ t_2 > t_1}}^T x_{s,t_1} x_{s,t_2}.$$

Die in (L2) auf Null gesetzten Variablen kann man komplett eliminieren, sowohl in der Zielfunktion, als auch in dieser und den folgenden Straffunktionen. Damit bleiben nur noch die $x_{s,t}$ mit $t \in B_s$. Bei (L3) und (L4) ist weiterhin zu beachten, dass für jeden Zeitpunkt t die mittleren Terme einen anderen Wert ergeben können. Das heißt, die Differenzen müssen immer durch jeweils andere Schlupfvariablen korrigiert werden. Daher werden $y \in \{0, 1\}^{T \times D+1}$ und $z \in \{0, 1\}^{T \times L+1}$ für $D = \lfloor \log_2(C_{\max} - C_{\min}) \rfloor$ und $L = \lfloor \log_2(M_{\max} - M_{\min}) \rfloor$ benötigt. Damit können dann die Bedingungen durch

$$p_{(L3)}(x, y) = \sum_{t=1}^T \left(C_0 - c_{\text{Base}}t + \sum_{s \in S} c_s \sum_{\tau=1}^t x_{s,\tau} - C_{\min} - \sum_{n=0}^{D-1} 2^n y_{t,n} - (d - 2^D + 1) y_{t,D} \right)^2,$$

$$p_{(L4)}(x, z) = \sum_{t=1}^T \left(M_0 + \sum_{s \in S} m_s \sum_{\tau=1}^t x_{s,\tau} - M_{\min} - \sum_{n=0}^{L-1} 2^n z_{t,n} - (d - 2^L + 1) z_{tL} \right)^2$$

erzwungen werden. Die Strafgewichte zu diesen Funktionen müssen je nach Wahl der Zielfunktion bestimmt werden. Da bei einer Verletzung der entsprechenden Bedingung die jeweilige Änderung für die Straffunktion genau 1 beträgt, müssen sie größer als Δf gewählt werden. Damit lässt sich das Optimierungsproblem als QUBO aufstellen mit

$$\min \left\{ f(x) + P_1 p_{(L1)}(x) + P_2 p_{(L3)}(x, y) + P_3 p_{(L4)}(x, z) : \right. \\ \left. x \in \{0, 1\}^{S \times T}, y \in \{0, 1\}^{T \times (D+1)}, z \in \{0, 1\}^{T \times (L+1)} \right\}.$$

4.4.3 Analyse

Insgesamt ergeben sich damit für dieses QUBO

$$\sum_{s \in S} |B_s| + T(D + L + 2) \\ = \sum_{s \in S} |B_s| + T(\lfloor \log_2(C_{\max} - C_{\min}) \rfloor + \lfloor \log_2(M_{\max} - M_{\min}) \rfloor + 2)$$

Variablen. Durch jedes Quadrieren einer großen Summe, wie in $p_{(L1)}$, wird jeder Summand mit jedem anderen multipliziert. Das bedeutet, die Bedingung (L1) erzeugt

für jeden Zeitpunkt einen vollständigen Teilgraphen im Interaktionsgraphen über allen zu diesem Zeitpunkt möglichen Zuständen. Bei den Akku- und Speicherbedingungen kommen zusätzlich die Schlupfvariablen aufgrund der Ungleichungen hinzu. Diese erweitern die durch (L3) und (L4) entstandenen vollständigen Teilgraphen über allen Zeitvariablen eines Zustandes zusätzlich. Damit wird ein fast vollständiger Graphen über allen vorhanden Variablen benötigt.

4.4.4 Beispiel

In diesem Abschnitt soll nun ein kleines Minimalbeispiel mit den folgenden Daten ausgewertet werden:

$$\begin{aligned}
 T &= 12 \\
 B_{\text{Charge}} &= \{2, 3, 4, 6, 7, 8, 10, 11, 12\}, \\
 B_{\text{Link}} &= \{2, 7, 12\}, \\
 B_{\text{ExpI}} &= \{1, 2, 5, 6, 9\}, \\
 B_{\text{ExpII}} &= \{1, \dots, 11\}, \\
 C_{\text{max}} &= 100, \\
 C_{\text{min}} &= 36, \\
 M_{\text{max}} &= 128, \\
 M_{\text{min}} &= 0.
 \end{aligned}$$

Nach der Formel des vorherigen Abschnitts ergeben sich damit insgesamt

$$9 + 3 + 5 + 11 + 12(\log_2(64) + \log_2(128) + 2) = 208$$

Variablen. Angenommen man würde dieses QUBO über einem vollständigen Graphen auf einem D-Wave-Chip darstellen, dann benötigte man zur Abbildung 52×52 Einheitszellen, also ungefähr 22000 Qubits. Dies steht in keinem Verhältnis zur ursprünglichen Größe des linearen Problems, zeigt aber deutlich die Schwierigkeit, ein passendes Anwendungsproblem für diesen Rechner zu finden.

4.5 Zusammenfassungen

Kontinuierliches Satellitenzeitplanungsproblem

- Gegeben:
- $T \in \mathbb{R}_{>0}$
 - $S \neq \emptyset$ mit $|S| < \infty$
 - $d_s \in \mathbb{R}_{\geq 0} \forall s \in S$
 - $B_s = \bigcup_{k=1}^{K_s} I_k \subset 2^{[0,T]}$ mit $K_s \in \mathbb{N}$, $|K_s| < \infty \forall s \in S$
 - $C_{\min}, C_{\max} \in \mathbb{R}_{\geq 0}$ mit $C_{\min} < C_{\max}$, $C_0 \in [C_{\min}, C_{\max}]$
 - $M_{\min}, M_{\max} \in \mathbb{R}_{\geq 0}$ mit $M_{\min} < M_{\max}$, $M_0 \in [M_{\min}, M_{\max}]$
 - $c_s \in \mathbb{R}$ mit $d_s |c_s| \leq C_{\max} - C_{\min} \forall s \in S$
 - $m_s \in \mathbb{R}$ mit $d_s |m_s| \leq M_{\max} - M_{\min} \forall s \in S$

Gesucht: $\min f(P)$

$$\text{s.t. (K1) } P : S \rightarrow 2^{[0,T]} \text{ mit } \bigcup_{s \in S} \bigcup_{I \in P(s)} I = [0, T]$$

$$\text{(K2) } t_1 - t_0 \geq d_s \forall [t_0, t_1] \in P(s) \text{ ggf. mit } t_1 \neq T \forall s \in S$$

$$\text{(K3) } \forall s \in S \forall I \in P(s) \exists J \in B_s : I \subseteq J$$

$$\text{(K4) } C_{\min} \leq C_0 + \sum_{s \in S} c_s \sum_{\substack{[t_0, t_1] \in P(s) \\ t_1 \leq t}} (t_1 - t_0) \leq C_{\max} \forall t \in [0, T]$$

$$\text{(K5) } M_{\min} \leq M_0 + \sum_{s \in S} m_s \sum_{\substack{[t_0, t_1] \in P(s) \\ t_1 \leq t}} (t_1 - t_0) \leq M_{\max} \forall t \in [0, T]$$

Diskretes Satellitenzeitplanungsproblem

- Gegeben:
- $T \in \mathbb{N}$
 - $S \neq \emptyset$ mit $|S| < \infty$
 - $d_s \in \mathbb{N} \forall s \in S$
 - $B_s \subseteq \{1, \dots, T\} \forall s \in S$
 - $C_{\min}, C_{\max} \in \mathbb{R}_{\geq 0}$ mit $C_{\min} < C_{\max}$, $C_0 \in [C_{\min}, C_{\max}]$
 - $M_{\min}, M_{\max} \in \mathbb{R}_{\geq 0}$ mit $M_{\min} < M_{\max}$, $M_0 \in [M_{\min}, M_{\max}]$
 - $c_s \in \mathbb{R}$ mit $d_s |c_s| \leq C_{\max} - C_{\min} \forall s \in S$
 - $m_s \in \mathbb{R}$ mit $d_s |m_s| \leq M_{\max} - M_{\min} \forall s \in S$

Gesucht: $\min f(x)$

s.t. (D1) $x \in \{0, 1\}^{S \times T}$ mit $\sum_{s \in S} x_{s,t} = 1 \forall t \in \{1, \dots, T\}$

(D2) $(1 - x_{s,t-1}) x_{s,t} \left(1 - \prod_{j=t+1}^{t+d_s-1} x_{s,j} \right) = 0 \forall t \in \{1, \dots, T - d_s + 1\}$

mit (D2a) $(1 - x_{s,t-1}) x_{s,t} \left(1 - \prod_{j=t+1}^T x_{s,j} \right) = 0$
 $\forall t \in \{T - d_s + 2, \dots, T\}$

oder (D2b) $(1 - x_{s,T-d_s+1}) \left(\sum_{t=T-d_s+2}^T x_{s,t} \right) = 0$

$\forall s \in S$

(D3) $x_{s,t} = 0 \forall t \in \{1, \dots, T\} \setminus B_s \forall s \in S$

(D4) $C_{\min} \leq C_0 + \sum_{s \in S} c_s \sum_{\tau=1}^t x_{s,\tau} \leq C_{\max} \forall t \in \{1, \dots, T\}$

(D5) $M_{\min} \leq M_0 + \sum_{s \in S} m_s \sum_{\tau=1}^t x_{s,\tau} \leq M_{\max} \forall t \in \{1, \dots, T\}$

Lineares Satellitenzeitplanungsproblem

- Gegeben:
- $T \in \mathbb{N}$
 - $S \neq \emptyset$ mit $|S| < \infty$ (ohne Leerlauf)
 - $B_s \subseteq \{1, \dots, T\} \quad \forall s \in S$
 - $C_{\min}, C_{\max} \in \mathbb{R}_{\geq 0}$ mit $C_{\min} < C_{\max}$, $C_0 \in [C_{\min}, C_{\max}]$
 - $M_{\min}, M_{\max} \in \mathbb{R}_{\geq 0}$ mit $M_{\min} < M_{\max}$, $M_0 \in [M_{\min}, M_{\max}]$
 - $c_{\text{Base}} \in \mathbb{R}_{\geq 0}$, $c_{\text{Base}} < C_{\max} - C_{\min}$
 - $c_s \in \mathbb{R}$ mit $|c_s - c_{\text{Base}}| \leq C_{\max} - C_{\min} \quad \forall s \in S$
 - $m_s \in \mathbb{R}$ mit $|m_s| \leq M_{\max} - M_{\min} \quad \forall s \in S$

Gesucht: $\min f(x)$

$$\text{s.t. (L1) } x \in \{0, 1\}^{|S| \times T} \text{ mit } \sum_{s \in S} x_{s,t} \leq 1 \quad \forall t \in \{1, \dots, T\}$$

$$\text{(L2) } x_{s,t} = 0 \quad \forall t \in \{1, \dots, T\} \setminus B_s \quad \forall s \in S$$

$$\text{(L3) } C_{\min} \leq C_0 - c_{\text{Base}} \cdot t + \sum_{s \in S} c_s \sum_{\tau=1}^t x_{s,\tau} \leq C_{\max} \quad \forall t \in \{1, \dots, T\}$$

$$\text{(L4) } M_{\min} \leq M_0 + \sum_{s \in S} m_s \sum_{\tau=1}^t x_{s,\tau} \leq M_{\max} \quad \forall t \in \{1, \dots, T\}$$

QUBO-Satellitenzeitplanungsproblem

- Gegeben:
- $T \in \mathbb{N}$
 - $S \neq \emptyset$ mit $|S| < \infty$
 - $B_s \subseteq \{1, \dots, T\} \quad \forall s \in S$
 - $C_{\min}, C_{\max} \in \mathbb{N}$ mit $C_{\min} < C_{\max}$, $C_0 \in \{C_{\min}, \dots, C_{\max}\}$
 - $M_{\min}, M_{\max} \in \mathbb{N}$ mit $M_{\min} < M_{\max}$, $M_0 \in \{M_{\min}, \dots, M_{\max}\}$
 - $c_{\text{Base}} \in \mathbb{N}$ mit $c_{\text{Base}} < C_{\max} - C_{\min}$
 - $c_s \in \mathbb{Z}$ mit $|c_s - c_{\text{Base}}| \leq C_{\max} - C_{\min} \quad \forall s \in S$
 - $m_s \in \mathbb{Z}$ mit $|m_s| \leq M_{\max} - M_{\min} \quad \forall s \in S$
 - P_1, P_2, P_3 ausreichend groß

Gesucht:

$$\begin{aligned} \min \quad & f(x) + P_1 \sum_{s \in S} \sum_{\substack{t_1, t_2=1 \\ t_2 > t_1}}^T x_{s,t_1} x_{s,t_2} \\ & + P_2 \sum_{t=1}^T \left(C_0 - c_{\text{Base}} t + \sum_{s \in S} c_s \sum_{\tau=1}^t x_{s,\tau} - C_{\min} \right. \\ & \quad \left. - \sum_{n=0}^{D-1} 2^n y_{t,n} - (d - 2^D + 1) y_{t,D} \right)^2 \\ & + P_3 \sum_{t=1}^T \left(M_0 + \sum_{s \in S} m_s \sum_{\tau=1}^t x_{s,\tau} - M_{\min} \right. \\ & \quad \left. - \sum_{n=0}^{L-1} 2^n z_{t,n} - (d - 2^L + 1) z_{t,L} \right)^2 \\ \text{s.t.} \quad & x \in \{0, 1\}^{S \times T}, y \in \{0, 1\}^{T \times (D+1)}, z \in \{0, 1\}^{T \times (L+1)} \end{aligned}$$

5 Fazit

Zielsetzung dieser Arbeit war es, die Struktur des Chimera-Graphen zu analysieren und Probleme zu finden, die speziell auf deren Einschränkungen abgestimmt sind. Zu diesem Zweck wurden unterschiedliche Minoren des Graphen betrachtet, anhand derer verschiedene Aussagen getroffen werden konnten. Da bisher kein effizienter Algorithmus existiert, der eine bestmögliche Einbettung eines allgemeinen Graphen in den Chimera-Graphen liefern kann, wurden zunächst vollständige und vollständig bipartite Graphen untersucht. Diese erlauben die Einbettung beliebiger beziehungsweise beliebiger bipartiter Graphen mit gleicher Knotenzahl, ohne dass weitere Kenntnisse über deren Struktur bekannt sind. Jedoch wurde festgestellt, dass die Zahl der hierbei benötigten Qubits aufgrund des hohen Ausgangsgrades der Knoten quadratisch in der Eingabegröße ist. Deswegen ist die maximale Größe solcher vollständigen Graphen, die auf einem beschränkten Chimera-Graphen eingebettet werden können, sehr gering.

Daraufhin wurden spezielle Minoren aus der Struktur des Chimera-Graphen abgeleitet, wobei über den gesamten Chimera-Graphen symmetrisch verteilte Kontraktionen angewandt wurden. Im ersten Schritt blieb dabei die gitterförmige Aufteilung in horizontal und vertikal aneinanderliegende und verbundene Zellen erhalten. Diese Zellen stellen vollständige oder vollständig bipartite Teilgraphen dar, deren Größe erweiterbar ist. Die Knotenzahl des gesamten Graphen ist für einen festen Erweiterungsgrad nur um einen konstanten Faktor geringer als die Zahl der Qubits des verwendeten Chimera-Graphen. Im zweiten und dritten Schritt wurden die Zellen zunehmend aufgelöst und die zugrundeliegende Gitterstruktur trat deutlicher hervor. Im Zuge dessen konnte jeweils eine bestmögliche Einbettung für den Gittergraphen und den erweiterten Gittergraphen mit diagonalen Kanten angegeben werden. Diese benötigen ebenfalls nur eine Qubitanzahl, die linear in der Zahl der Knoten ist. Somit liegt es nahe, bei Berechnungen auf der D-Wave-Maschine Probleme zu betrachten, die eine Art Gitterstruktur aufweisen.

Für die Realisierung auf der Maschine wurden für alle Minoren die dazugehörigen QUBOs formuliert. Mit einer geeigneten Indizierung der Graphen konnten zudem die für die Einbettung notwendigen Qubitstränge aufgestellt und in der Formulierung der zugehörigen Chimera-QUBOs genutzt werden. Der gezeigte Implementationsweg ist für jeden Minor anwendbar. Insbesondere die Verteilung der Gewichte und Stärken im Qubitstrang kann explizit über ein einheitliches Strafgewicht bestimmt werden. Für dessen Mindestgröße konnten geeignete Schranken angegeben werden.

Um eine Verbindung zu bekannten Problemen zu schaffen, wurden einige konkrete Beispiele der diskreten Optimierung herangezogen. Deren allgemeine Formulierungen mussten dazu meist erst in QUBOs transformiert werden. Daher wurde in Kapitel 2 zunächst die schrittweise Methodik dargestellt. Es stellte sich heraus, dass bei Graphenproblemen besonders die Umwandlung von Nebenbedingungen die Komplexität des In-

Interaktionsgraphen entscheidend beeinflusst, da diese meist zu vollständigen Teilgraphen über den beteiligten Knoten führt. Damit eignen sich Probleme mit Nebenbedingungen über vielen Variablen, wie beispielsweise Bin Packing, nur bedingt zur Berechnung auf dem D-Wave-Computer. Im Gegensatz dazu können das Problem der größten stabilen Menge und das des maximalen Schnittes eines Graphen direkt über diesem als QUBO dargestellt werden. Kann jedoch dessen Struktur mit keinem der Minore in Verbindung gebracht werden, ist die Einbettung auch nur über den vollständigen Graphen möglich.

Daraufhin wurden zwei weitere Problemstellungen, das Matching- und das Spannbaum-Problem, betrachtet. Diese wiesen in ihren linearen Formulierungen nur Nebenbedingungen über sehr wenigen Variablen auf, weshalb die Transformation zu einem QUBO nur lokal begrenzte Auswirkungen im Interaktionsgraphen hat. Um die Gitterstruktur des Chimera-Graphen aufzugreifen, wurde als Ausgangsgraph zudem ein Gittergraph gewählt. Die leichten Varianten dieser Probleme zeigten dabei einen starken Zusammenhang mit einem Minor des Chimera-Graphen, dem erweiterten Gittergraphen. Durch die Ergänzung der Nebenbedingungen, die die Probleme NP-schwer machen, wurden jedoch weitere Kopplungen hinzugefügt. Während der entstandene Interaktionsgraph des Matching-Problems annähernd einem Minor des Chimera-Graphen entsprach, konnten beim Spannbaum-Problem keinerlei Ähnlichkeiten festgestellt werden.

Abschließend folgte die Anwendung der gezeigten Methoden auf eine praktische Problemstellung der Raumfahrt: die Planung von Satellitenmissionen. Aus einem kontinuierlichen Modell konnte durch Diskretisierung und Vereinfachung ein lineares Optimierungsproblem erstellt werden. Dessen Nebenbedingungen in Form von Ungleichungen erhöhen die Komplexität des Interaktionsgraphen des QUBOs jedoch so stark, dass sich der D-Wave-Computer für deren Berechnung, insbesondere mit realistischen Missionslaufzeiten, als ungeeignet herausstellt.

Fortführend könnten zunächst die oben genannten speziellen QUBO-Formulierungen auf dem Chimera-Graphen gegenüberstellend sowohl auf klassischen Systemen als auch mit der D-Wave-Maschine gelöst werden. Durch einen Vergleich der Laufzeiten könnte eine Überlegenheit des Quantencomputers gegenüber klassischen Rechnern zumindest in absoluten Zahlen evaluiert werden. Mit der Entwicklung eines effizienten Algorithmus zur Einbettung eines beliebigen Graphen in den Chimera-Graphen, sofern dies möglich ist, ließen sich weitere Probleme, ohne spezielle Grundstruktur, ebenso testen. Eine tatsächliche Aussage über die Laufzeit in Abhängigkeit der Eingabewerte kann jedoch nur eine weitere Erforschung der physikalischen Hintergründe des Quantencomputers und der daraus resultierenden mathematischen Zusammenhänge liefern.

Abbildungen

1	Übertragung auf Graphen	5
	(a) Beispielinteraktionsgraph eines QUBOs	
	(b) Beispielhypergraph zu einer pseudobooleschen Funktion	
2	Komposition seriell-paralleler Netzwerke	7
	(a) Zwei seriell-parallele Netzwerke	
	(b) Parallele Komposition	
	(c) Serielle Komposition	
3	K_4 ist ein Minor des Gittergraphen $L_{3,3}$	8
	(a) Gittergraph $L_{3,3}$	
	(b) Vollständiger Graph K_4	
4	Inklusionsdiagramm der Komplexitätsklassen	10
5	Beispielgraphen zu einem Max-Cut-Problem	13
	(a) Ursprungsgraph mit maximalem Schnitt	
	(b) Interaktionsgraph mit minimalem Teilgraphen	
6	Konstruktion des Kantengraphen	24
	(a) Ausgangsgraph	
	(b) Kanten werden Knoten	
	(c) Kantengraph	
7	Umwandlung zu ungerichteten Graphen	24
	(a) Gerichteter Graph	
	(b) Ungerichteter Graph	
8	Beispielgraphen zu Matching-Problemen	25
	(a) Größtmögliches Matching in G	
	(b) Minimales maximales Matching in G	
9	Interaktionsgraphen zu Matching-Problemen	26
	(a) Stabile Menge im Kantengraph von G	
	(b) Interaktionsgraph zu Min-Max-Matching von G	
10	Konstruktion des Dualgraphen	29
	(a) Ausgangsgraph	
	(b) Länder werden Knoten	
	(c) Dualgraph	
	(d) Spannbäume	
11	Aufbau der Einheitszelle [37]	33
12	Zwei Darstellungen des vollständig bipartiten Graphen $K_{4,4}$	33
13	Chimera-Graph $C_{2,3}$	34
14	K_5 in $C_{1,1}$	37
15	Einbettung von K_{10} in den Chimera-Graphen über Qubitstränge	37
16	Zusammensetzung eines Hamiltonkreises in $C_{n,m}$	43

	(a) Hamiltonkreis in $K_{4,4}$	
	(b) geschlossener Pfad über alle Knoten in $L_{n,m}$	
17	Hamiltonkreis in $C_{3,4}$	44
18	Qubitstränge im Chimera-Graphen C zum vollständig bipartiten Graphen	46
19	Erste Kontaktion	49
	(a) Qubitstränge im Chimera-Graphen C zu C'	
	(b) Minor C'	
20	Dreidimensionales Gitter $L_{n,m,4}$ ist Teilgraph von C'	50
21	$L_{2n,2m}$ ist Teilgraph von C'	50
22	K_8 erweitert	53
23	Ausschnitt $L_{3,3}[K_8]$ (aus 6×6 Einheitszellen)	53
24	Zweite Kontaktion	56
	(a) Qubitstränge im Chimera-Graphen C zu C''	
	(b) Minor C''	
25	Erweiterter Gittergraph \tilde{L} ist Teilgraph von C''	57
26	Zwei Cliques mit gemeinsamen Knoten	58
27	Dritte Kontaktion	61
	(a) Qubitstränge im Chimera-Graphen C zu C'''	
	(b) Minor C'''	
28	$V_{n,m}[K_4]$	65
29	Qubitstränge in T-Form	66
30	Qubitstränge im Chimera-Graphen C zu $V_{n,m}[K_4]$	67
31	Minoren des Chimera-Graphen (hervorgegangen aus $x \times y$ Einheitszellen) 70	
	(a) Ausschnitt von $C'_{n,m} = L_{n,m}[K_4]$ (3×3)	
	(b) Ausschnitt von $C''_{n,m}$ (3×3)	
	(c) Ausschnitt von $C'''_{n,m}$ (3×3)	
	(d) Ausschnitt von $L_{n,m}[K_8]$ (6×6)	
	(e) Ausschnitt von $V_{n,m}[K_4]$ (5×7)	
32	Zusammenfassung der Minoren in $C_{n,m}$	71
33	Einbettung zu Beispielgraph aus Abbildung 5	74
	(a) Einbettungswerte	
	(b) Stränge in Einheitszelle	
34	$K_{1,3}$	75
35	Kantengraph des Gittergraphen	76
	(a) Gitter L_{nm}	
	(b) Kantengraph $\mathcal{L}(L_{nm})$	
36	Siebenerclique	78
37	Spannbaumproblem im Gittergraphen	80
	(a) Gitter $L_{n,m}$ und dessen Duales	
	(b) Interaktionsgraph zu $L_{n,m}$ und seinem Dualen	
38	Zusätzliche Kanten für einen Knoten bei Hamiltonpfadproblem	81
39	Zusätzliche Kanten im Interaktionsgraphen	81

Literatur

- [1] P. Benioff. „The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines“. In: *Journal of Statistical Physics* 22.5 (1980), S. 563–591.
- [2] Y. I. Manin. *Vychislimoe i nevychislimoe*. Sov. radio, 1980.
- [3] R. P. Feynman. „Simulating physics with computers“. In: *International Journal of Theoretical Physics* 21.6 (1982), S. 467–488.
- [4] D. Deutsch. „Quantum theory, the Church-Turing principle and the universal quantum computer“. In: *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*. Bd.400. 1818. The Royal Society. 1985, S. 97–117.
- [5] T. D. Ladd u. a. „Quantum computers“. In: *Nature* 464.7285 (2010), S. 45–53.
- [6] B. Heim u. a. „Quantum versus classical annealing of Ising spin glasses“. In: *Science* 348.6231 (2015), S. 215–217.
- [7] I. Hen u. a. „Probing for quantum speedup in spin-glass problems with planted solutions“. In: *Physical Review A* 92.4 (2015), S. 042325.
- [8] A. Perdomo-Ortiz u. a. „A performance estimator for quantum annealers: Gauge selection and parameter setting“. In: *arXiv preprint arXiv:1503.01083* (2015).
- [9] T. F. Rønnow u. a. „Defining and detecting quantum speedup“. In: *Science* 345.6195 (2014), S. 420–424.
- [10] V. S. Denchev u. a. „What is the Computational Value of Finite Range Tunneling?“ In: *arXiv preprint arXiv:1512.02206* (2015).
- [11] S. T. Chakradhar und M. L. Bushnell. „A solvable class of quadratic 0–1 programming“. In: *Discrete applied mathematics* 36.3 (1992), S. 233–251.
- [12] F. Barahona. „On the computational complexity of Ising spin glass models“. In: *Journal of Physics A: Mathematical and General* 15.10 (1982), S. 3241.
- [13] M. W. Carter. „The indefinite zero-one quadratic problem“. In: *Discrete Applied Mathematics* 7.1 (1984), S. 23–44.
- [14] F. Barahona, M. Jünger und G. Reinelt. „Experiments in quadratic 0–1 programming“. In: *Mathematical Programming* 44.1-3 (1989), S. 127–137.
- [15] P. L. Hammer, P. Hansen und B. Simeone. „Roof duality, complementation and persistency in quadratic 0–1 optimization“. In: *Mathematical programming* 28.2 (1984), S. 121–155.
- [16] D. Li, X. L. Sun und C. L. Liu. „An exact solution method for unconstrained quadratic 0–1 programming: a geometric approach“. In: *Journal of Global Optimization* 52.4 (2012), S. 797–829.

- [17] E. Boros und P. L. Hammer. „Pseudo-boolean optimization“. In: *Discrete applied mathematics* 123.1 (2002), S. 155–225.
- [18] J.-C. Picard und H. D. Ratliff. „Minimum cuts and related problems“. In: *Networks* 5.4 (1975), S. 357–370.
- [19] F. Barahona. „A solvable case of quadratic 0–1 programming“. In: *Discrete Applied Mathematics* 13.1 (1986), S. 23–26.
- [20] D. Eppstein. „Parallel recognition of series-parallel graphs“. In: *Information and Computation* 98.1 (1992), S. 41–55.
- [21] R. J. Duffin. „Topology of series-parallel networks“. In: *Journal of Mathematical Analysis and Applications* 10.2 (1965), S. 303–318.
- [22] D. P. Bovet, P. Crescenzi und D. Bovet. *Introduction to the Theory of Complexity*. Citeseer, 1994, S. 178–202.
- [23] E. Bernstein und U. Vazirani. „Quantum complexity theory“. In: *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*. ACM, 1993, S. 11–20.
- [24] P. W. Shor. „Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer“. In: *SIAM journal on computing* 26.5 (1997), S. 1484–1509.
- [25] E. Farhi u. a. „Quantum computation by adiabatic evolution“. In: *arXiv preprint quant-ph/0001106* (2000).
- [26] D. Aharonov u. a. „Adiabatic quantum computation is equivalent to standard quantum computation“. In: *SIAM review* 50.4 (2008), S. 755–787.
- [27] L. K. Grover. „A fast quantum mechanical algorithm for database search“. In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. ACM, 1996, S. 212–219.
- [28] R. Karp. „Reducibility among Combinatorial Problems“. In: *Complexity of Computer Computations*. The IBM Research Symposia Series. Springer US, 1972, S. 85–103.
- [29] A. Lucas. „Ising formulations of many NP problems“. In: *Frontiers in Physics* 2.5 (2014).
- [30] F. Hadlock. „Finding a maximum cut of a planar graph in polynomial time“. In: *SIAM Journal on Computing* 4.3 (1975), S. 221–225.
- [31] E. Boros und P. L. Hammer. „The max-cut problem and quadratic 0–1 optimization; polyhedral aspects, relaxations and bounds“. In: *Annals of Operations Research* 33.3 (1991), S. 151–180.
- [32] D-Wave Systems Inc. *Programming with QUBOs*. Techn. Ber. Release 1.5.6-1 für Windows, 09-1002A-B. 2013.
- [33] E. D. Dahl. „Programming with D-Wave: Map Coloring Problem“. In: (2013).
- [34] M. Rinaldi, G. Jünger und G. Reinelt. „The Travelling Salesman Problem“. In: (1994).
- [35] M. R. Garey und D. S. Johnson. *Computers and intractability*. Bd. 29. wh freeman New York, 2002.

- [36] J. C. Williams. „A linear-size zero—one programming model for the minimum spanning tree problem in planar graphs“. In: *Networks* 39.1 (2002), S. 53–60.
- [37] D-Wave Systems Inc. *Introduction to the D-Wave Quantum Hardware*. besucht: 04.02.2016. URL: <http://www.dwavesys.com/tutorials/background-reading-series/introduction-d-wave-quantum-hardware>.
- [38] V. Choi. „Minor-embedding in adiabatic quantum computation: I. The parameter setting problem“. In: *Quantum Information Processing* 7.5 (2008), S. 193–209.
- [39] N. Robertson und P. D. Seymour. „Graph minors. XIII. The disjoint paths problem“. In: *Journal of combinatorial theory, Series B* 63.1 (1995), S. 65–110.
- [40] N. D. Roussopoulos. „A max {m, n} algorithm for determining the graph H from its line graph G“. In: *Information Processing Letters* 2.4 (1973), S. 108–112.
- [41] L. W. Beineke. „Characterizations of derived graphs“. In: *Journal of Combinatorial Theory* 9.2 (1970), S. 129–135.
- [42] V. Schaus u. a. „A Continuous Verification Process in Concurrent Engineering“. In: *AIAA Space Conference, San Diego, CA, USA*. 2013.
- [43] V. S. Denchev u. a. „What is the Computational Value of Finite Range Tunneling?“ In: *arXiv preprint arXiv:1512.02206* (2015).
- [44] S. V. Isakov u. a. „Understanding Quantum Tunneling through Quantum Monte Carlo Simulations“. In: *arXiv preprint arXiv:1510.08057* (2015).
- [45] S. Boyd und L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

