

**Approximative Lösungen des  
Max-Cut-Problems  
mit semidefiniten Programmen**

Inaugural-Dissertation

zur

Erlangung des Doktorgrades der  
Mathematisch-Naturwissenschaftlichen Fakultät  
der Heinrich-Heine-Universität Düsseldorf

vorgelegt von

**Bernd Hirschfeld**

aus Würzburg

Düsseldorf

2004

Gedruckt mit der Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Heinrich-Heine-Universität Düsseldorf

Referent: Univ.-Prof. Dr. Florian Jarre

Koreferent: Univ.-Prof. Dr. Kristian Witsch

Tage der mündlichen Prüfung: 30. Januar 2004

# Vorwort

In jüngerer Zeit beschäftigt sich die Forschung in der stetigen Optimierung verstärkt mit semidefiniten Programmen. Hierzu gibt es mittlerweile gut entwickelte Lösungsmethoden, insbesondere sehr effiziente und polynomiell konvergente Innere-Punkte-Verfahren.

Populär wurden semidefinite Programme unter anderem durch ihren Nutzen für kombinatorische Probleme. Sie liefern häufig eine bessere Relaxierung als die zuvor gebräuchlichen linearen Programme. Man erhält mit ihnen gute obere Schranken für kombinatorische Maximierungsprobleme.

Ein herausragender Meilenstein in der Anwendung von semidefiniten Programmen auf kombinatorische Probleme ist eine Arbeit von Goemans und Williamson aus dem Jahr 1995 ([9]). In ihr wird zum einen gezeigt, dass die semidefinite Relaxierung des Max-Cut-Problems eine gute obere Schranke ermittelt. Bedeutender ist aber, dass man mit der Optimallösung der semidefiniten Relaxierung sehr gute Näherungslösungen (untere Schranken) für Max-Cut erhält (87,856 % optimal für nicht-negative Kantengewichte). Vorher hatte das beste bekannte Verfahren mit polynomieller Laufzeit eine Leistungsgarantie von im wesentlichen 50 %.

Nach einem Überblick über das Max-Cut-Problem im ersten Kapitel erläutern wir das Verfahren von Goemans und Williamson in Kapitel 2. Wir fassen auch die bisher dazu bekannten Resultate zusammen.

Im Anschluss an [9] gab es viele Arbeiten, die das Verfahren von Goemans und Williamson weiter analysierten oder (auf den Resultaten von [9] aufbauend) bessere Verfahren suchten. Bis heute gelang es aber nicht, ein beweisbar besseres Verfahren zu entwickeln. So schlugen Anjos und Wolkowicz [1] eine strengere semidefinite Relaxierung für das Max-Cut-Problem vor, konnten aber nur numerisch bessere Resultate zeigen. Auch in der Arbeit von Burer, Monteiro und Zhang [3], in der ein neues Verfahren für Näherungslösungen vorgeschlagen wird, gibt es nur numerische Anhaltspunkte für eine Verbesserung. In Kapitel 3 stellen wir zwei Verfahren vor, die auf [9] aufbauen.

Bereits bei der Analyse ihres Verfahrens entdeckten Goemans und Williamson als „Nebenprodukt“ eine neue Formulierung des Max-Cut-Problems. Diese trigonometrische Formulierung wurde bisher wenig beachtet. Nur Nesterov [24] nutzte sie für analytische Zwecke.

---

In dieser Arbeit wird erstmals die trigonometrische Formulierung näher untersucht (Kapitel 4). Darüber hinaus wird ein Verfahren entwickelt, um sie näherungsweise zu lösen. Aufbauend auf einer Näherungslösung der trigonometrische Formulierung werden Methoden aufgezeigt, um sehr gute Näherungslösungen für das Max-Cut-Problem zu ermitteln (Kapitel 6). Im letzten Kapitel werden die entwickelten Lösungsverfahren dann ausführlich numerisch untersucht.

Ganz besonders danken möchte ich meinem Doktorvater Prof. Dr. Florian Jarre für die herzliche Betreuung und die fördernde Begleitung meiner Forschungsarbeiten. Auch unserem Systemadministrator Bertold Nöckel bin ich sehr dankbar für seine stets verfügbare Hilfe.

Düsseldorf, den 30. Januar 2004

Bernd Hirschfeld

# Inhaltsverzeichnis

<b>1. Das Max-Cut-Problem</b>	<b>7</b>
1.1. Problemstellung . . . . .	7
1.1.1. Verschiedene Formulierungen . . . . .	7
1.1.2. Eigenschaften des Max-Cut-Polytops . . . . .	9
1.1.3. Affine Relaxierung . . . . .	10
1.2. Bisherige Verfahren . . . . .	11
1.3. Verwandte Probleme . . . . .	12
1.4. Anwendungen . . . . .	13
<b>2. Näherungsverfahren von Goemans und Williamson</b>	<b>16</b>
2.1. Semidefinite Programme . . . . .	16
2.1.1. Grundlagen . . . . .	16
2.1.2. Verfahren . . . . .	19
2.2. Semidefinite Relaxierung . . . . .	19
2.3. Das Verfahren von Goemans und Williamson . . . . .	21
2.3.1. Grundidee . . . . .	21
2.3.2. Güte der Näherungslösung . . . . .	22
2.3.3. Realisierung des Verfahrens und Laufzeitabschätzungen . . . . .	24
2.3.4. Lösungsraum des GW-Algorithmuses . . . . .	25
2.4. Weitere Analysen des Näherungsverfahrens . . . . .	26
2.4.1. Die Leistungsgarantie . . . . .	26
2.4.2. Das Integritätsverhältnis . . . . .	28
<b>3. Weitere Verfahren</b>	<b>30</b>
3.1. Rang-2-Relaxierung . . . . .	30
3.1.1. Idee . . . . .	30
3.1.2. Verfahren . . . . .	32
3.1.3. Analyse . . . . .	33
3.2. Branch and Bound nach Goemans . . . . .	34

<b>4. Die trigonometrische Approximation</b>	<b>36</b>
4.1. Motivation . . . . .	36
4.2. Grundlegende Eigenschaften von $\mathcal{TA}$ . . . . .	39
4.2.1. Einbeschriebene und umschreibende Kugeln . . . . .	39
4.2.2. Gemeinsame Randflächen mit $\mathcal{MC}$ . . . . .	39
4.3. Sternförmigkeit . . . . .	41
4.4. Regularität . . . . .	48
4.4.1. Ecken von $\mathcal{TA}$ . . . . .	48
4.4.2. Kanten von $\mathcal{TA}$ . . . . .	52
4.5. Lokal optimale Ecke auch global optimal . . . . .	53
<b>5. Lösung der trigonometrischen Approximation</b>	<b>55</b>
5.1. Grundidee . . . . .	55
5.2. Ein Innere-Punkte-Verfahren . . . . .	55
5.3. Runden zu einer ganzzahligen Lösung . . . . .	57
5.3.1. Kombinatorisch . . . . .	58
5.3.2. Zufallsschnitte . . . . .	58
<b>6. Numerische Ergebnisse</b>	<b>60</b>
6.1. Allgemeines . . . . .	60
6.1.1. Fallbeispiele . . . . .	60
6.1.2. Algorithmen . . . . .	61
6.1.3. Randbedingungen der numerischen Berechnungen . . . . .	65
6.2. Naiver Vergleich von GW und $\mathcal{TA}$ . . . . .	66
6.3. Die verschiedenen Varianten bei der trigonometrischen Approximation . . . . .	68
6.4. GW und $\mathcal{TA}$ mit lokaler Suche . . . . .	71
6.4.1. Überblick . . . . .	71
6.4.2. Überlegungen für die Praxis . . . . .	73
<b>Zusammenfassung</b>	<b>75</b>
<b>A. Bezeichnungen</b>	<b>77</b>

# 1. Das Max-Cut-Problem

## 1.1. Problemstellung

### 1.1.1. Verschiedene Formulierungen

Das Problem des maximalen Schnittes (Max-Cut-Problem) stammt aus der Graphentheorie.<sup>1</sup>

**Definition 1.1 (Partitionierung, Schnitt)** Sei  $G = (V, E)$  ein gewichteter, ungerichteter Graph mit Knotenmenge  $V = \{1, \dots, n\}$  und Kantenmenge  $E$  sowie den Gewichten  $g_e$  für  $e \in E$ , gegeben durch seine Adjazenzmatrix  $W = (w_{ij})$  mit  $w_{ij} = w_{ji} = g_e$  für  $e \in E, e = \{i, j\}$  und  $w_{ij} = 0$  sonst.

Dann ist für eine Menge  $U \subset V$  die **Partitionierung**  $(U, \bar{U}) := (U, V \setminus U)$ , und der **Schnitt**  $\delta(U)$  ist definiert durch  $\delta(U) := \{\{i, j\} \in E \mid i \in U, j \in \bar{U}\}$ . Das Gewicht  $w(\delta(U))$  des Schnittes  $\delta(U)$  ist die Summe der Gewichte aller enthaltenen Kanten:  $w(\delta(U)) := \sum_{\{i, j\} \in \delta(U)} w_{ij}$

**Problem 1.2 (Max-Cut)** Sei  $G = (V, E)$  ein Graph wie oben. Dann ist das zugehörige **Max-Cut-Problem**, das größte Gewicht aller möglichen Schnitte  $\delta(U)$  von  $G$  zu finden:

$$\text{mc}(G) := \max_{U \subset V} \sum_{i \in U, j \notin U} w_{ij} = \max_{U \subset V} w(\delta(U)) \quad (1.1)$$

**Bemerkung 1.3** Da ein Knoten nicht in beiden Mengen  $U$  und  $\bar{U} = V \setminus U$  liegen kann, sind etwaige Schleifen in  $G$  irrelevant. Daher wird im Folgenden stets  $w_{ii} = 0$  vorausgesetzt.

Die graphentheoretische Beschreibung (1.1) des Max-Cut-Problems ist für algorithmische Zwecke ungeeignet. Deshalb wird die Menge  $U$  für mathematische Formulierungen oft durch einen Vektor  $x \in \{-1, 1\}^n$  dargestellt:

$$x_i \in \{-1, 1\} \text{ mit } x_i = 1 \Leftrightarrow i \in U.$$

---

<sup>1</sup>Zur Einführung in die Graphentheorie sei auf das Buch von Noltemeier [26] verwiesen.

## 1. Das Max-Cut-Problem

---

Dementsprechend ist die Kante  $\{i, j\}$  genau dann im Schnitt  $\delta(U)$  enthalten, wenn  $x_i x_j = -1$  (andernfalls gilt  $x_i x_j = 1$ ). Man erhält also wegen

$$\begin{aligned} w(\delta(U)) &= \sum_{\{i,j\} \in \delta(U)} w_{ij} = \sum_{\substack{x_i x_j = -1, \\ i < j}} w_{ij} = \frac{1}{2} \sum_{\substack{x_i x_j = -1, \\ i < j}} w_{ij} (1 - x_i x_j) \\ &= \frac{1}{2} \sum_{\substack{x_i x_j = -1, \\ i < j}} w_{ij} (1 - x_i x_j) + \frac{1}{2} \sum_{\substack{x_i x_j = 1, \\ i < j}} w_{ij} (1 - x_i x_j) \\ &= \frac{1}{2} \sum_{i < j} w_{ij} (1 - x_i x_j) \end{aligned}$$

die Formulierung

$$\begin{aligned} \max \quad & \frac{1}{2} \sum_{i < j} w_{ij} (1 - x_i x_j) \\ \text{mit: } & x_i \in \{-1, 1\}. \end{aligned} \tag{1.2}$$

Diese Darstellung wird oft (vgl. etwa [12]) mit Hilfe der zum Graphen gehörigen **Laplace-Matrix**  $L$  von  $W$  umgeformt. Dabei ist die Laplace-Matrix definiert<sup>2</sup> als  $L := L(W) := \text{Diag}(We) - W$ . Mit der Bezeichnung  $L = (l_{ij})$  gilt dann

$$\begin{aligned} \frac{1}{2} \sum_{i < j} w_{ij} (1 - x_i x_j) &= \frac{1}{4} \sum_{i,j} w_{ij} (1 - x_i x_j) = \frac{1}{4} \left( \sum_{i,j} w_{ij} - \sum_{i,j} x_i x_j w_{ij} \right) \\ &= \frac{1}{4} \left( \sum_i l_{ii} + \sum_{i,j} x_i x_j l_{ij} \right) = \frac{1}{4} x^T L x. \end{aligned}$$

Man erhält so eine Formulierung des Max-Cut-Problems als quadratisches Programm:

$$\begin{aligned} \text{mc}(G) = \quad & \max \quad \frac{1}{4} x^T L x \\ & \text{mit: } x \in \{-1, 1\}^n \end{aligned} \tag{1.3}$$

---

<sup>2</sup>Mit  $e$  ist hier der Vektor  $(1, 1, \dots, 1)^T$  passender Dimension gemeint – nicht die Kante  $e \in E$ . Diese Inkonsistenz lässt sich bei kanonischer Bezeichnung leider nicht vermeiden. Aus dem Zusammenhang geht die Bedeutung jedoch hervor.

Diese Formulierung im  $\mathbb{R}^n$  kann mit  $x^T L x = \text{Spur}(L x x^T)$  noch in eine Formulierung im Raum  $\mathcal{S}^n$  der symmetrischen  $n \times n$ -Matrizen überführt werden:

$$\begin{aligned} \text{mc}(G) = & \max \frac{1}{4} L \bullet X \\ \text{mit: } & X = x x^T \\ & x \in \mathbb{R}^n \\ & \text{diag}(X) = e \end{aligned} \tag{1.4}$$

Wegen  $\text{diag}(x x^T) = \text{diag}(X) = e$  ist  $x_i^2 = 1$ , also  $x \in \{-1, 1\}^n$ .

Da die Zielfunktion linear in  $X$  ist, ändert sich der Optimalwert nicht, wenn die konvexe Hülle der zulässigen Menge betrachtet wird:

$$\begin{aligned} \text{mc}(G) = & \max \frac{1}{4} L \bullet X \\ \text{mit: } & X \in \mathcal{MC} \end{aligned} \tag{1.5}$$

mit

$$\mathcal{MC} := \text{conv}\{X \mid \exists x \in \mathbb{R}^n, x x^T = X, \text{diag}(X) = e\}. \tag{1.6}$$

Die Menge  $\mathcal{MC}$  wird als das **Max-Cut-Polytop** bezeichnet. Es gilt offensichtlich die Beziehung  $\mathcal{MC} \subset [-1, 1]^{n \times n}$ .

### 1.1.2. Eigenschaften des Max-Cut-Polytops

Das Max-Cut-Polytop hat eine sehr interessante Struktur, die zum Beispiel in [5] ausführlich beschrieben wird. Im folgenden werden wir einige wichtige Eigenschaften von  $\mathcal{MC}$  zitieren.

Das Max-Cut-Polytop „lebt“ in einem Raum, den wir in der gesamten Arbeit nicht mehr verlassen werden. Es handelt sich um einen affinen Unterraum der symmetrischen  $n \times n$ -Matrizen, der durch

$$\mathcal{L} := \{X = X^T \mid \text{diag}(X) = e\}$$

definiert ist.

**Satz 1.4 ([5])** *Das Max-Cut-Polytop  $\mathcal{MC}$  liegt im  $n(n-1)/2$ -dimensionalen Raum  $\mathcal{L}$  und hat  $2^{n-1}$  Knoten, die alle durch  $x x^T$  mit  $x \in \pm 1^n$  gegeben sind.*

*Beliebige Paare von Knoten sind jeweils benachbart, d. h. durch eine Kante – eine eindimensionale Extremalmenge – verbunden.*

Die zweite Aussage dieses Satzes ist bemerkenswert, denn es ist in den Dimensionen  $d = 1$ ,  $d = 2$  und  $d = 3$  nicht möglich, ein konvexes Polytop mit mehr als  $d + 1$  Knoten

zu konstruieren, bei dem alle Paare von Knoten benachbart sind. Für  $d = n(n-1)/2 > 3$  gilt dies aber für  $\mathcal{MC}$ , denn dann ist  $2^{n-1} > d + 1$ .

Satz 1.4 ist auch in einer weiteren Hinsicht bemerkenswert: Während zwei Optimallösungen des Max-Cut-Problems oft nicht in irgendeiner Weise benachbart sind, sind es deren Entsprechungen im Raum  $\mathcal{L}$  durchaus: Denn in  $\mathcal{L}$  sind zwei beliebige zulässige Basis-Lösungen (d. h. Ecken) des Max-Cut-Problems benachbart.

Natürlich kann man jedes Polytop durch endlich viele lineare Ungleichungen beschreiben, d. h. es handelt sich bei (1.5) um ein Lineares Programm, das man zum Beispiel mit der Simplexmethode lösen könnte. Die Schwierigkeit bei dieser Darstellung besteht jedoch in der Unzahl der Nebenbedingungen.<sup>3</sup> In jeder Ecke gibt es exponentiell viele benachbarte ( $2^{n-1} - 1$ ) Ecken und dementsprechend ebenso viele ausgehende Kanten. Somit ist Aufgabe, von einer beliebigen Ecke ausgehend, eine absteigende Kante zu bestimmen (Simplexschritt), so komplex wie das Lösen des ganzen Max-Cut-Problems.

Die Menge  $\mathcal{MC}$  ist hoch symmetrisch. So ist sie zum einen invariant gegenüber Permutationen der Knoten. Zum anderen gibt es viele Spiegel-Hyperebenen. Die Symmetriegruppe von  $\mathcal{MC}$  hat die Ordnung  $2^{n-1}n!$  (vgl. [5]).

### 1.1.3. Affine Relaxierung

Wie oben beschrieben, ist die Darstellung des Max-Cut-Polytops durch lineare Ungleichungen nicht praktikabel. Deshalb sucht man nach Möglichkeiten, die Menge  $\mathcal{MC}$  durch einfachere Beschreibungen zu approximieren. Dabei möchte man vermeiden, dass die Optimallösung des Max-Cut-Problems für die Approximation nicht mehr zulässig ist. Man wählt daher in der Regel Approximationen, die die Menge  $\mathcal{MC}$  enthalten, d. h. Relaxierungen von  $\mathcal{MC}$ .

Eine sehr gebräuchliche Relaxierung, die mit linearen Ungleichungen auskommt, ist das so genannte metrische Polytop, das wir im Folgenden kurz mit  $\mathcal{MET} := \mathcal{MET}_n$  bezeichnen. Das metrische Polytop basiert auf einer Betrachtung dreier Knoten  $i, j, k$  und der sie verbindenden Kanten. Für diese drei Kanten gilt offensichtlich, dass entweder keine oder genau zwei in einem Schnitt  $\delta(U)$  liegen können. Da  $\mathcal{MET}$  im Raum  $\mathcal{L}$  liegt, werden die Kanten wieder mit  $x_{ij}, x_{jk}, x_{ki}$  bezeichnet. Wegen  $x_{\mu\nu} = -1 \Leftrightarrow \{\mu, \nu\} \in \delta(U)$  gilt dann:

$$x_{ij} + x_{jk} + x_{ki} + 1 \geq 0 \tag{1.7a}$$

---

<sup>3</sup>Die genaue Zahl der notwendigen Hyperebenen, um  $\mathcal{MC}_n$  zu beschreiben ist nicht bekannt. Sie ist jedoch mindestens exponentiell.

(höchstens zwei Kanten) und

$$x_{ij} - x_{jk} - x_{ki} + 1 \geq 0 \quad (1.7b)$$

$$-x_{ij} + x_{jk} - x_{ki} + 1 \geq 0 \quad (1.7c)$$

$$-x_{ij} - x_{jk} + x_{ki} + 1 \geq 0 \quad (1.7d)$$

(wenn zwei Kanten nicht im Schnitt liegen, dann auch die dritte nicht). Die Bezeichnung „metrisches Polytop“ geht darauf zurück, dass obige Gleichungen als Dreiecksungleichung einer Metrik motiviert werden können (siehe [5]). Um das metrische Polytop vollständig zu beschreiben, benötigt man insgesamt  $4 \binom{n}{3}$  Ungleichungen.

Für  $3 \leq n \leq 4$  gilt  $\mathcal{MET}_n = \mathcal{MC}_n$ . Für  $n \geq 5$  ist das metrische Polytop echt größer als das Max-Cut-Polytop. Dann gilt: Da beide Polytope  $\mathcal{MC}$  und  $\mathcal{MET}$  im Würfel  $[-1, 1]^{n \times n}$  enthalten sind und jede Ecke von  $\mathcal{MC}$  auch eine Ecke von  $\mathcal{MET}$  ist, hat  $\mathcal{MET}$  mehr Ecken als das Max-Cut-Polytop. Keine dieser zusätzlichen Ecken ist aber dann ganzzahlig (siehe [5]). Somit schneiden dann die obigen Ungleichungen alle Ecken des Würfels  $[-1, 1]^{n \times n}$  ab, die nicht in  $\mathcal{MC}$  enthalten sind.

Da für  $n \geq 5$  das Polytop  $\mathcal{MC}_n$  echt größer als  $\mathcal{MET}_n$  ist, ist für Graphen, die den  $K_5$  enthalten,<sup>4</sup> das metrische Polytop keine korrekte Beschreibung.

Natürlich kann man die obige Betrachtung mit drei Knoten (Ungleichungen (1.7)) auch auf  $k > 3$  Knoten ausdehnen und entsprechende Ungleichungsrestriktionen der Problembeschreibung hinzufügen. Da die Anzahl der Restriktionen aber dann mit einem Polynom mindestens vierten Grades über  $n$  bestimmt ist, sind diese Probleme nicht mehr praktikabel zu lösen.

## 1.2. Bisherige Verfahren

Die NP-Vollständigkeit von Max-Cut wurde schon 1972 von Karp [19] gezeigt. Auch für ungewichtete Graphen ( $w_e = 1$  für alle  $e \in E$ ) ist das Max-Cut-Problem NP-vollständig. Nur für einzelne spezielle Graphen (etwa planare) kennt man polynomielle Algorithmen.

Nicht nur die exakte Lösung des Max-Cut-Problems, sondern auch die approximative Lösung, ist unter der P=NP-Frage zu betrachten. So zeigte Håstad [11], dass eine Approximation mit einer Leistungsgarantie von  $\frac{16}{17} + \varepsilon$  für jedes  $\varepsilon > 0$  NP-hart ist.

Für das allgemeine Max-Cut-Problem mit beliebigen Kantengewichten waren zunächst nur schwache Näherungsverfahren polynomieller Laufzeit bekannt:<sup>5</sup> Ausgehend von einem einfachen Verfahren ([30]) mit einer Leistungsgarantie von  $1/2$ , fand man bis 1994 nur

<sup>4</sup>Mit  $K_5$  ist der vollständige Graph mit 5 Ecken bezeichnet.

<sup>5</sup>Die Leistungsgarantien beziehen sich hier auf Graphen mit nicht-negativen Kantengewichten. Kommen negative Gewichte hinzu, so verschiebt sich der „Nullpunkt“ (kleinstes Gewicht aller Schnitte) und die Leistungsgarantien gelten für eine andere Bezugsgröße.

geringfügig bessere Verfahren bis hin zu  $\frac{1}{2} + \frac{1}{2n}$ . Auch das metrische Polytop  $\mathcal{M\mathcal{E}T}$  garantiert keine bessere Lösung.

Man beachte, dass eine Leistungsgarantie von  $1/2$  bereits mittels eines einfachen zufallsbasierenden Verfahrens erreicht werden kann: Bei der zufälligen Zuordnung aller Knoten zu  $U$  bzw.  $\bar{U}$  liegt jede Kante mit Erwartungswert  $\frac{1}{2}$  im Schnitt  $\delta(U)$  (siehe auch [28]).

Erst im Jahre 1995 gelang es Goemans und Williamson [9] einen erheblichen Fortschritt in der approximativen Lösung des Max-Cut-Problems zu erzielen: Mit ihrem randomisierten Verfahren erreichen sie eine Leistungsgarantie von etwas über 87%. Dieses polynomielle Verfahren wird in Kapitel 2 vorgestellt und analysiert.

Das Verfahren von Goemans und Williamson fand viel Beachtung. Es wurde variiert (vgl. Kapitel 3) und genauer analysiert (vgl. Abschnitt 2.4). Es gelang jedoch nicht, weitere beweisbare Verbesserungen zu erzielen.

### 1.3. Verwandte Probleme

Wie wir im Abschnitt 1.1 mit Formulierung (1.3) gesehen haben, kann man das Max-Cut-Problem als quadratisches Programm über  $\pm 1$ -Variablen formulieren. Umgekehrt kann man aber auch jedes quadratische Programm über  $\pm 1$ -Variablen auf ein Max-Cut-Problem reduzieren (vgl. etwa [28]): Sei also ein quadratisches  $\pm 1$ -Programm

$$\begin{aligned} \max x^T Q x + q^T x \\ \text{mit: } x \in \{-1, 1\}^n \end{aligned} \tag{1.8}$$

durch eine symmetrische Matrix  $Q \in \mathbb{R}^{n \times n}$  und einen Vektor  $q \in \mathbb{R}^n$  gegeben. Dann kann (1.8) in ein Max-Cut-Problem umgeformt werden über dem Graphen  $K_{n+1}$  (dem vollständigen Graphen mit  $n + 1$  Knoten). Dazu haben die Gewichte  $w_{ij}$  der Kanten  $\{i, j\}$  die Gewichte

$$w_{ij} := \begin{cases} -2Q_{ij} & \text{für } 1 \leq i, j \leq n \\ -q_i & \text{für } 1 \leq i \leq n, j = n + 1 \\ -q_j & \text{für } 1 \leq j \leq n, i = n + 1 \\ 0 & \text{für } i = j = n + 1. \end{cases}$$

Neben den quadratischen Programmen über  $\pm 1$ -Variablen ist Max-Cut auch eng verwandt mit dem Maximum-2-Satisfiability-Problem. Das Max-Sat-Problem ist ein Problem aus der Aussagenlogik. Dabei wird eine Formel  $F$  in konjunktiver Normalform betrachtet, d. h.

$$F = F(x_1, \dots, x_n) = \bigwedge_{i=1}^n \left( \bigvee_{j=1}^{m_i} l_{i,j} \right)$$

mit Literalen  $l_{i,j}$ . (Ein Literal ist eine Variable  $x_\nu$  oder ihre Negation  $\neg x_\nu$ .) Gesucht ist eine Variablenbelegung, für die möglichst viele Klauseln  $\bigvee_{j=1}^{m_i} l_{i,j}$  erfüllt sind. (Eine Klausel ist erfüllt, wenn mindestens eine ihrer Literale erfüllt ist.)

Präzise formuliert besteht das Max-Sat-Problem darin, die maximale Anzahl der erfüllbaren Klauseln zu bestimmen (über alle möglichen Variablenbelegungen). Beim Max-2-Sat-Problem ist  $m_i \leq 2$ .

Das Max-2-Sat-Problem ist NP-vollständig, wenn auch ein einfacher Algorithmus in linearer Zeit feststellen kann, ob *alle* Klauseln erfüllbar sind oder nicht.

Max-2-Sat kann ebenfalls auf Max-Cut reduziert werden (siehe zum Beispiel [9]). In [9] wird ebenfalls erläutert, wie der im Vorwort erwähnte und im nächsten Kapitel erklärte Algorithmus von Goemans und Williamson auf Max-2-Sat angewendet werden kann und eine gute Approximation an die Optimallösung (mindestens 87%) liefert. Auch wurde durch [9] die Approximation des allgemeinen Max-Sat-Problems leicht verbessert. In [7] zeigen Feige und Goemans einen Algorithmus auf, der das Max-2-Sat-Problem mit einer Leistungsgarantie von 0,931 löst. Hierbei ist bemerkenswert, dass es bereits NP-hart ist, Max-2-Sat mit mehr als  $21/22 \approx 0,9545$  zu approximieren.

## 1.4. Anwendungen

Neben der theoretischen Bedeutung des Max-Cut-Problems gibt es auch zahlreiche praktische Anwendungen (vgl. [28]).

In den Arbeiten [17], [27], [29] und [4] wird zum Beispiel die Anwendung auf Spingläser, einer Problemstellung aus dem Bereich der Ordnung/Unordnungs-Phänomene in der Physik, untersucht. Einführungen in Spingläser finden sich in [2] und [13].

Ein Spinglas (englisch *spin glass*) ist eine Legierung von magnetischen Unreinheiten in einem nichtmagnetischen Metall (z. B. das metallische Kristall AuFe). Von physikalischem Interesse ist dabei ein evtl. Phasenübergang bei extrem niedriger Temperatur.

Ein mögliches mathematisches Modell der Spingläser lautet wie folgt: Ein gegebenes Spinglas enthalte  $n$  magnetische Unreinheiten (Atome, oder allgemeiner *sites*). Jede Site  $i$  hat eine magnetische Orientierung (*spin*)  $s_i \in \mathbb{R}^3$  der Länge 1. Zwischen je zwei Sites  $i, j$  gibt es Wechselwirkungen  $J_{ij}$ , die von dem nichtmagnetischen Metall und der Entfernung zwischen den Sites abhängen. Sind nun  $i$  und  $j$  zwei Sites mit den Spins  $s_i$  und  $s_j$ , so ist die Energiewechselwirkung für das gesamte Spinglas durch

$$H = \sum_{i,j} J_{ij} s_i^T s_j$$

gegeben. Die verschiedenen Zustände, die Spins bzw. Sites annehmen können, sorgen unter Berücksichtigung der Wechselwirkungen zwischen den Sites dafür, dass eine Site bzw. das gesamte Spinglas eine gewisse Energie bzw. Temperatur hat. Von physikalischem Interesse ist es nun, den Zustand mit der niedrigsten Energie (Temperatur 0 K) eines

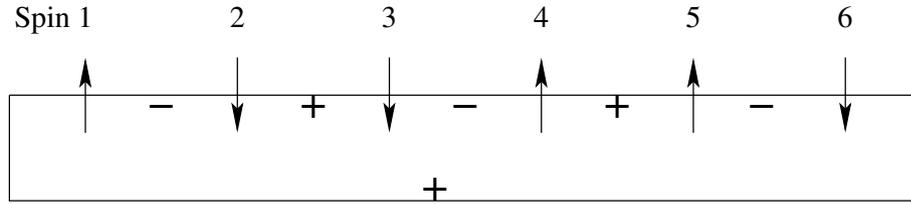


Abbildung 1.1.: Frustriertes Spinglas

Spinglases zu bestimmen (Grundzustand). Da der absolute Nullpunkt praktisch nicht erreicht werden kann, ist man auf die numerische Bestimmung an Hand von Simulationen angewiesen.

Für die numerische Simulation verwendet man oft eine Vereinfachung des obigen Modells, das so genannte Ising-Modell. Bei Ising-Spingläsers können die Sites nur zwei Orientierungen annehmen, nämlich *up* und *down*, d. h.  $s_i \in \{\pm 1\}$ . Meist stellt man sich die Anordnung der Sites als regelmäßiges, quadratisches Gitter (z. B. 2-dimensional) mit periodischem Rand (Torus) vor. Die Wechselwirkungen werden Gauß-verteilt modelliert (meist Erwartungswert 0 und Varianz  $\frac{1}{N}$ ). Die Matrix  $J$  der Wechselwirkungen ist also bis auf die Diagonale voll besetzt. Somit ist das Problem der Grundzustandssuche gegeben durch

$$\min H_S = -\frac{1}{2} \sum_{i,j} J_{ij} \sigma_i \sigma_j \quad (1.9)$$

mit:  $\sigma \in \{-1, 1\}^n$ .

Die Problematik der Bestimmung der Minimalenergie von Spingläsers lässt sich – wie auch das Max-Cut-Problem selbst – sehr schön veranschaulichen. Wir betrachten ein einfaches Spinglas mit  $N = 6$  ein-dimensional angeordneten Spins (Abbildung 1.1, die Wechselwirkungen ( $J_{ij} = \{\pm 1\}$ ) sind durch + und – gekennzeichnet). Die beiden linken Spins „möchten“ gegeneinander ausgerichtet sein, der zweite will parallel zum dritten liegen und so weiter bis zum sechsten am rechten Rand. Dieses will parallel zum ersten liegen, d. h. dieser müsste sich umdrehen, dann ist er aber nicht mehr anders als der zweite. Diese Unfähigkeit, allgemeine Zufriedenheit zu schaffen, nennt man Frustration. Durch sie wird die Bestimmung der Minimalenergie nicht-trivial: Die Bestimmung der minimalen Energie eines beliebigen Spinglases in der Formulierung (1.9) ist offenbar äquivalent zum Max-Cut-Problem und somit NP-vollständig.

Eine weitere Anwendung des Max-Cut-Problems tritt beim VLSI-Design auf. Dort geht es bei Zwei-schichtigen Schaltkreisen darum, möglichst wenig Verbindungen zwischen den beiden Schichten zu benötigen.

Auch bei Digital-Analog-Konvertern oder bei Kryptogrammen mit Vokalen und Konsonanten treten zu Max-Cut äquivalente Probleme auf (siehe [28]).

## 2. Näherungsverfahren von Goemans und Williamson

Für das Max-Cut-Problem ist seit längerem eine semidefinite Relaxierung bekannt. Wir führen zunächst in die semidefinite Programmierung ein und erläutern anschließend die semidefinite Relaxierung.

Von der Lösung dieser Relaxierung haben Goemans und Williamson in [9] nicht nur gezeigt, dass deren Optimalwert sehr nahe an dem des Ausgangsproblems liegt (kleiner als 113,9 Prozent für nicht-negative Gewichte), sondern auch, dass man mit ihr gute Näherungslösungen (im Durchschnitt  $> 87,8\%$  für nicht-negative Gewichte) in polynomieller Zeit erhalten kann. Wir stellen das Verfahren im Abschnitt 2.3 vor. Das Kapitel schließt mit Verfahrensanalysen, die über diejenigen in [9] hinausgehen.

### 2.1. Semidefinite Programme

Semidefinite Programme werden erst seit 1989 verstärkt benutzt. Dies liegt daran, dass sie zuvor mit der Ellipsoidmethode nur unpraktikabel gelöst werden konnten. Im Jahre 1989 zeigten dann Nesterov und Nemirovsky, dass Innere-Punkte-Methoden im Prinzip auf alle konvexen Optimierungsprobleme verallgemeinert werden können. Sehr ausführlich lässt sich dies in [25] nachlesen. Mit Hilfe dieses Resultats entwickelte man dann effiziente Innere-Punkte-Verfahren zur Lösung Semidefiniter Programme und erschloss neue Anwendungsgebiete.

#### 2.1.1. Grundlagen

**Definition 2.1 (positiv semidefinite Matrizen)** Eine reelle Matrix  $A$  heißt **positiv semidefinit**, in Zeichen  $A \succeq 0$ , wenn  $A$  symmetrisch ist und für alle  $x \in \mathbb{R}^n$  gilt:  $x^T A x \geq 0$ .

**Bemerkung 2.2** Da  $x^T A x \geq 0$  und  $x^T B x \geq 0$  implizieren, dass

$$x^T(\lambda A + (1 - \lambda)B)x = \lambda x^T A x + (1 - \lambda)x^T B x \geq 0 \text{ für } \lambda \in [0, 1]$$

gilt, ist die Menge der positiv semidefiniten Matrizen des  $\mathbb{R}^{n \times n}$  ein konvexer Kegel ist. Er wird im Folgenden mit  $S_+^n$  bezeichnet.

$S_+^n$  induziert eine Halbordnung für  $A, B \in S^n$ , und zwar  $A \succeq B \Leftrightarrow A - B \succeq 0$ . Diese Ordnung wird **Löwner-Halbordnung** genannt.

Nun sei noch auf eine weitere nützliche Eigenschaft positiv semidefiniter Matrizen hingewiesen (siehe z.B. [15]):

**Satz 2.3 (Schur-Komplement)** Sei

$$M = \begin{pmatrix} A & B \\ B^T & C \end{pmatrix},$$

mit  $A \in \mathcal{S}_+^n$ ,  $A$  regulär,  $C \in \mathcal{S}_+^m$ . Dann sind die beiden folgenden Aussagen äquivalent:

1.  $M$  ist positiv semidefinit,
2.  $C \succeq B^T A^{-1} B$ .

Mit Hilfe dieser Grundlagen können nun die semidefiniten Programme näher betrachtet werden:

**Definition 2.4 (Semidefinite Programme – SDP)** Seien  $A : \mathcal{S}_+^n \rightarrow \mathbb{R}^k$ ,  $B : \mathcal{S}_+^n \rightarrow \mathbb{R}^m$  zwei lineare Operatoren und  $C \in \mathcal{S}_+^n$ ,  $a \in \mathbb{R}^k$ ,  $b \in \mathbb{R}^m$ . Dann wird

$$\begin{aligned} & \max \text{Spur } CX \\ & \text{mit: } a - A(X) = 0 \\ & \quad b - B(X) \geq 0 \\ & \quad X \succeq 0. \end{aligned} \tag{2.1}$$

*semidefinites Programm (SDP) genannt.*<sup>1</sup>

Ein SDP ist also ein konvexes Optimierungsproblem, da die Restriktionen und die Zielfunktion konvex sind. Darüberhinaus sind semidefinite Programme eine Verallgemeinerung der linearen Programme, d. h. jedes LP kann als SDP geschrieben werden: Sei

$$\begin{aligned} & \max c^T x \\ & \text{mit: } Ax + b \geq 0 \\ & \quad x \geq 0 \end{aligned} \tag{2.2}$$

ein lineares Programm ( $A = (a_1, a_2, \dots, a_n)^T \in \mathbb{R}^{m \times n}$ , d. h. die  $a_i$  sind die Zeilen von  $A$ ; ferner  $b \in \mathbb{R}^m$ ,  $c, x \in \mathbb{R}^n$ ). Mit  $X := \text{Diag}(x)$ ,  $C := \text{Diag}(c)$  und dem linearen Operator  $B(X) := -(\text{Spur } A_1 X, \text{Spur } A_2 X, \dots, \text{Spur } A_m X)^T$  mit  $A_i := \text{Diag}(a_i)$  ist dann

$$\begin{aligned} & \max \text{Spur } CX \\ & \text{mit: } b - B(X) \geq 0 \\ & \quad X \succeq 0. \end{aligned}$$

ein zu (2.2) äquivalentes semidefinites Programm.

<sup>1</sup>Spur  $A := \sum_{i=1}^n a_{ii}$  für  $A = (a_{ij}) \in \mathbb{R}^{n \times n}$ . Für symmetrische Matrizen  $A, B \in \mathcal{S}^n$  gilt  $\text{Spur } AB = A \bullet B = \sum_{i,j} a_{i,j} b_{i,j} = \sum_{i,j} (A \circ B)_{i,j}$ , d. h. Spur  $AB$  kann in  $\mathcal{O}(n^2)$  Operationen berechnet werden.

### Dualität

Das zu (2.1) duale Programm ist

$$\begin{aligned} & \min a^T y + b^T t \\ & \text{mit: } A^T(y) + B^T(t) - C \succeq 0 \\ & \quad y \in \mathbb{R}^k, t \in \mathbb{R}_+^m. \end{aligned} \tag{2.3}$$

Im Gegensatz zu Dualität bei linearen Programmen, bei der die Optimalwerte des LPs und seines dualen Programms immer gleich sind (bis auf den Ausnahmefall, dass beide keine zulässige Lösung besitzen), ist die Dualität bei semidefiniten Programmen schwächer.

**Beispiel 2.5** (vgl. [31]) *Betrachte das SDP*

$$\begin{aligned} & \min x_1 \\ & \text{mit: } \begin{pmatrix} 0 & x_1 & 0 \\ x_1 & x_2 & 0 \\ 0 & 0 & x_1 + 1 \end{pmatrix} = X \\ & \quad X \succeq 0. \end{aligned} \tag{2.4}$$

Die zulässige Menge ist  $\{(x_1, x_2)^T \mid x_1 = 0, x_2 \geq 0\}$ , die Optimallösung dementsprechend  $p^* = 0$ . Das zu (2.4) duale Programm,

$$\begin{aligned} & \max -z_2 \\ & \text{mit: } \begin{pmatrix} z_1 & \frac{(1-z_2)}{2} & 0 \\ \frac{(1-z_2)}{2} & 0 & 0 \\ 0 & 0 & z_2 \end{pmatrix} = Z \\ & \quad Z \succeq 0, \end{aligned} \tag{2.5}$$

hat die zulässige Menge  $\{(z_1, z_2)^T \mid z_1 \geq 0, z_2 = 1\}$  und die Optimallösung  $d^* = -1$ .

Obwohl also beide Programme zulässig sind, gibt es also eine Dualitätslücke. Diese ist sogar endlich – ein Fall, der bei linearen Programmen gar nicht auftreten kann.

Man unterscheidet daher bei semidefiniten Programmen zwischen *schwacher* und *starker* Dualität. Die schwache Dualität ist immer erfüllt: Das duale Programm ist eine Schranke für das primale (und umgekehrt). Die starke Dualität erfordert über die Existenz zulässiger Lösungen hinaus, dass eines der beiden Programme strikt zulässige Lösungen besitzt:

**Satz 2.6 (Dualitätssatz für semidefinite Programme)** *Für ein semidefinites Programm (2.1) mit Optimallösung  $p^*$  und sein duales Programm (2.3) mit Optimallösung  $d^*$  gilt  $p^* = d^*$ , wenn eine der beiden Bedingungen erfüllt ist:*

1. Das primale Programm (2.1) ist strikt zulässig, d. h. es existiert ein positiv definites  $X$  mit  $b - B(X) > 0$ .
2. Das duale Programm (2.3) ist strikt zulässig, d. h. es existieren Vektoren  $y$  und  $t > 0$  mit  $A^T(y) + B^T(t) - C \succ 0$ .

*Beweis:* siehe zum Beispiel [25] □

### 2.1.2. Verfahren

Zur Lösung von semidefiniten Programmen (2.1) bieten sich insbesondere Innere-Punkte-Verfahren an. Für das so genannte Kurzschrittverfahren konnte der theoretische Nachweis erbracht werden, dass es im wesentlichen eine Laufzeit von  $\mathcal{O}((n^2m^2 + n^3m)\sqrt{n})$  haben (siehe zum Beispiel [31]). Praktische Implementierungen der Innere-Punkte-Verfahren verfolgen meistens einen anderen Ansatz, so dass diese theoretische Komplexitätsschranke verloren geht; sie sind aber in der Praxis erheblich schneller. Hier sei auch noch einmal auf den Übersichtsartikel [31] von Vandenberghe und Boyd hingewiesen. Er enthält eine sehr ausführliche Einführung in semidefinite Programme.

Für das Max-Cut-Problem lässt sich das primal-duale Innere-Punkte-Verfahren mit HKM-Suchrichtung sehr einfach implementieren (vgl. [12]).

## 2.2. Semidefinite Relaxierung

Das Max-Cut-Problem in seiner Formulierung (1.4) ist äquivalent ( $X = xx^T \succeq 0$  sowie  $\text{Rang } xx^T = 1$  für alle  $x \in \mathbb{R}^n$ ) zu:

$$\begin{aligned}
 & \max \frac{1}{4} \text{Spur } LX \\
 & \text{mit: } \text{diag}(X) = e \\
 & \quad \text{Rang}(X) = 1 \\
 & \quad X \succeq 0.
 \end{aligned} \tag{2.6}$$

Lässt man nun die Rang-Bedingung weg, so erhält man als Relaxierung das semidefinite Programm

$$\begin{aligned}
 & \max \frac{1}{4} \text{Spur } LX \\
 & \text{mit: } e - \text{diag}(X) = 0 \\
 & \quad X \succeq 0.
 \end{aligned} \tag{2.7}$$

Diese SDP-Relaxierung des Max-Cut-Problems wurde durch Arbeiten von S. Polyak bekannt gemacht. Die zulässige Menge von (2.7), d. h. die semidefinite Relaxierung des Max-Cut-Polytops  $\mathcal{MC}$ , bezeichnen wir mit

$$\mathcal{SR} := \mathcal{SR}_n := \{X \in \mathbb{R}^{n \times n} \mid \text{diag}(X) = e, X \succeq 0\}.$$

### Zugang von Goemans und Williamson

Die Relaxierung (2.7) wird in [9] anders motiviert: Goemans und Williamson gehen zunächst von der Formulierung (1.2) aus:

$$\begin{aligned} & \max \frac{1}{2} \sum_{i < j} w_{ij} (1 - y_i y_j) \\ & \text{mit: } y_i \in \{-1, 1\}^n \end{aligned}$$

Die Relaxierung von [9] besteht nun darin, die  $y_i$  zunächst als 1-dimensionale Vektoren der Länge 1 aufzufassen und dann diese  $y_i$  auf mehrdimensionale Vektoren  $v_i$  mit euklidischer Einheitsnorm zu verallgemeinern. Da ein linearer Raum, der von diesen  $n$  Vektoren  $v_i$  aufgespannt wird, höchstens die Dimension  $n$  hat, kann man  $v_i \in \mathbb{R}^n$  (oder  $\mathbb{R}^m$  für ein  $m \leq n$ ) annehmen. Genauer gesagt liegen die Vektoren  $v_i$  somit in der  $n$ -dimensionalen Einheitskugel  $S_n$  (oder  $S_m$  für ein  $m < n$  bei linearer Abhängigkeit der Vektoren). Nun muss noch die Zielfunktion angepasst werden. Dies geschieht sinnvoller Weise dadurch, dass man  $y_i y_j$  als Skalarprodukt  $y_i^T y_j$  interpretiert und entsprechend auf die  $v_i$  anwendet. Somit erhält man folgende Relaxierung von (1.2):

$$\begin{aligned} & \max \frac{1}{2} \sum_{i < j} w_{ij} (1 - v_i^T v_j) \\ & \text{mit: } v_i \in S_n \end{aligned} \tag{2.8}$$

Dieses Problem kann in ein semidefinites Programm überführt werden: Für eine positiv semidefinite Matrix  $Y \succeq 0$  existiert stets eine Matrix  $B \in \mathbb{R}^{m \times n}$ ,  $m \leq n$ , mit  $Y = B^T B$  ( $m = \text{Rang } Y$ , siehe zum Beispiel [21], zur Cholesky-Zerlegung siehe Unterabschnitt 2.3.3). Falls  $y_{ii} = 1$  gilt, so sind die Spalten  $(v_1, \dots, v_n)$  von  $B$  Einheitsvektoren des  $\mathbb{R}^m$ . Die übrigen Einträge von  $Y$  sind dann Skalarprodukte der  $v_i$ :  $y_{ij} = v_i^T v_j$ . Umgekehrt gibt es stets für  $n$  Vektoren  $v_i \in S_n$  eine positive semidefinite Matrix  $Y$  mit  $y_{ij} = v_i^T v_j$  (mittels  $M := (v_1 \ v_2 \ \dots \ v_n)$  und  $Y := M^T M$ ). Diese wird Gram-Matrix von  $\{v_1, \dots, v_n\}$  genannt, siehe [21, S. 110].

Somit ist also (2.8) äquivalent zu

$$\begin{aligned} \max \quad & \frac{1}{2} \sum_{i < j} w_{ij}(1 - y_{ij}) \\ \text{mit: } & e - \text{diag}(Y) = 0 \\ & Y \succeq 0. \end{aligned} \tag{2.9}$$

Wegen

$$\begin{aligned} \frac{1}{2} \sum_{i < j} w_{ij}(1 - y_{ij}) &= \frac{1}{4} \sum_{i,j} w_{ij} - \frac{1}{4} \sum_{i,j} w_{ij}y_{ij} = \frac{1}{4} \text{Spur Diag}(We)Y - \frac{1}{4} \text{Spur } WY \\ &= \frac{1}{4} \text{Spur } LY \end{aligned}$$

folgt aber die Äquivalenz von (2.9) mit (2.7).

Damit ist auch folgendes Lemma gezeigt:

**Lemma 2.7** *Sei  $G = (V, E)$  ein Graph,  $|V| = n$ , gegeben durch seine Gewichtsmatrix  $W \in \mathbb{R}^{n \times n}$ . Dann gilt für die Optimallösung  $X$  der Relaxierung (2.7): Es gibt eine Zerlegung  $X = B^T B$ ,  $\text{Rang } X = m$ ,  $B \in \mathbb{R}^{m \times n}$ , und die Spalten von  $B$  sind Vektoren der Länge 1.*

## Duales SDP

Im weiteren Verlauf wird noch das duale Programm zum Max-Cut-SDP (2.7) benötigt. Die Bildung des dualen Programms folgt unmittelbar der Darstellung des vorhergehenden Abschnitts 2.1 und lautet:

$$\begin{aligned} \min \quad & e^T y \\ \text{mit: } & \text{Diag}(y) - \frac{1}{4}L \succeq 0. \end{aligned} \tag{2.10}$$

## 2.3. Das Verfahren von Goemans und Williamson

### 2.3.1. Grundidee

Das Verfahren von Goemans und Williamson zur Bestimmung einer Näherungslösung eines Max-Cut-Problems ist eigentlich recht einfach und auf den ersten Blick unscheinbar:

1. Löse (2.7), dabei die Optimallösung als  $n$  Vektoren  $v_i \in \mathbb{R}^m$  erhaltend.
2. Sei  $r$  ein Zufallsvektor, gleichverteilt auf der Einheitssphäre  $S_m$ .
3. Setze  $U := \{i | v_i^T r \geq 0\}$ .

Mit anderen Worten: Die Optimalvektoren  $v_i$  werden von einer zufälligen Hyperebene, die durch den Ursprung verläuft und die durch  $r$  als Normale bestimmt ist, in zwei Mengen geteilt: Diejenigen Vektoren, die in den „positiven Halbraum“ (Skalarprodukt mit  $r$  nichtnegativ) zeigen, und jene Vektoren die in den „negativen Halbraum“ zeigen.

Dieses Verfahren wird dadurch motiviert, dass das Max-Cut-SDP in der Formulierung (2.8) unabhängig vom Koordinatensystem ist: Der Funktionswert ist invariant gegenüber orthonormalen Transformationen der Vektoren  $v_i$ .

Im Rest dieses Abschnittes wird nun gezeigt, dass dieses Verfahren eine polynomielle Laufzeit hat und die Näherungslösungen im Mittel mindestens 87% des Optimalwerts des Max-Cut-Problems erreichen.

### 2.3.2. Güte der Näherungslösung

Sei mit  $E[W]$  der Erwartungswert derjenigen Lösung bezeichnet, die das oben vorgestellte Verfahren liefert. Bei der Abschätzung dieses Erwartungswertes ist von zentraler Bedeutung, mit welcher Wahrscheinlichkeit einzelne Kanten im Schnitt liegen und somit einen Beitrag zum Gesamtgewicht des Schnitts leisten. Gleichbedeutend damit ist die Frage, mit welcher Wahrscheinlichkeit zwei Knoten nicht der gleichen Teilmenge zugewiesen werden. Es gilt:<sup>2</sup>

$$E[W] = \sum_{i < j} w_{ij} \Pr[i \in U, j \in \bar{U} \vee j \in U, i \in \bar{U}] \cdot 1.$$

Somit ist die Wahrscheinlichkeit  $\Pr[\text{sgn}(v_i^T r) \neq \text{sgn}(v_j^T r)]$  zu bestimmen.<sup>3</sup> Diese Wahrscheinlichkeit – dass  $r$  die Vektoren  $v_i$  und  $v_j$  separiert – hängt offensichtlich linear vom Winkel zwischen  $v_i$  und  $v_j$  ab ([9]):

$$\Pr[\text{sgn}(v_i^T r) \neq \text{sgn}(v_j^T r)] = \frac{1}{\pi} \arccos(v_i^T v_j). \quad (2.11)$$

Auf Grund der Linearität des Erwartungswertes folgt sofort der Satz:

**Satz 2.8 ([9])**

$$E[W] = \frac{1}{\pi} \sum_{i < j} w_{ij} \arccos(v_i^T v_j)$$

Der Kniff in der Analytik besteht nun darin, das Verhältnis von  $\frac{1}{\pi} \arccos(v_i^T v_j)$  und  $\frac{1}{2}(1 - v_i^T v_j)$  abzuschätzen (siehe auch Abbildung 2.1):

---

<sup>2</sup>Pr steht für „probability“.

<sup>3</sup>Hier und im Folgenden wird die Funktion  $\text{sgn}$  nicht ganz kanonisch definiert:  $\text{sgn}(x) = 1$  für  $x \geq 0$  und  $\text{sgn}(x) = -1$  sonst.

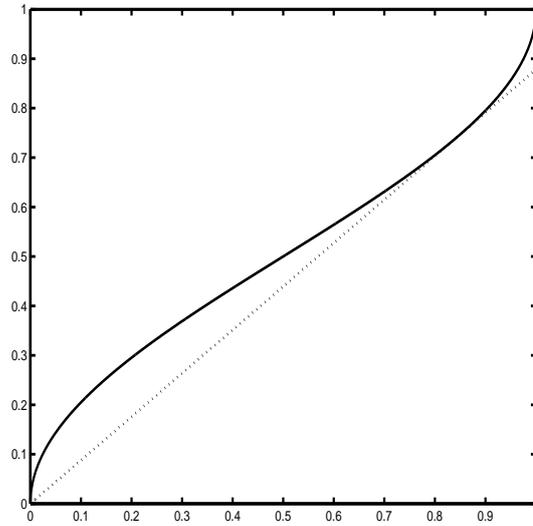


Abbildung 2.1.: Die Funktion  $\frac{1}{2}(1 - y) \mapsto \arccos(y)/\pi$  (wobei hier  $y$  mit  $v_i^T v_j$  identifiziert wird) und die Abschätzung  $\frac{1}{2}(1 - y) \mapsto \alpha \frac{1}{2}(1 - y)$

**Lemma 2.9 ([9])** Für alle  $v_i \in \mathbb{R}^n$  mit  $-1 \leq v_i^T v_j \leq 1$  ( $1 \leq i, j \leq n$ ) gilt

$$\frac{1}{\pi} \arccos(v_i^T v_j) \geq \alpha \frac{1}{2}(1 - v_i^T v_j)$$

mit

$$\alpha := \min_{0 \leq \theta \leq \pi} \frac{2}{\pi} \frac{\theta}{1 - \cos \theta} \approx 0,87856.$$

Setzt man nun die Nichtnegativität der Kantengewichte  $w_{ij}$  voraus, so ist das Hauptresultat gezeigt:

**Satz 2.10 ([9])** Sei  $E[W]$  der Erwartungswert des Verfahrens von Goemans und Williamson aus Unterabschnitt 2.3.1, sei  $Z_{SDP}^*$  der Optimalwert der semidefiniten Relaxierung (2.7) bzw. (2.8). Dann gilt:

$$E[W] \geq \alpha \frac{1}{2} \sum_{i < j} w_{ij}(1 - v_i^T v_j) = \alpha Z_{SDP}^*.$$

Bezeichne nun  $Z_{MC}^*$  den Optimalwert des Max-Cut-Problems. Dann gilt

$$Z_{SDP}^* \geq Z_{MC}^* \geq E[W] \geq \alpha Z_{SDP}^* \geq \alpha Z_{MC}^*.$$

Das Näherungsverfahren erreicht also im Durchschnitt mindestens das  $\alpha$ -fache des Optimalwerts. Umgekehrt liegt der Wert der SDP-Relaxierung (2.8) nicht über dem  $\frac{1}{\alpha}$ -fachen des Optimalwerts des Max-Cut-Problems.

### 2.3.3. Realisierung des Verfahrens und Laufzeitabschätzungen

Zunächst muss also das SDP (2.7) gelöst (Lösungsmatrix  $X$ ) werden. Hierzu bietet sich eines der im Unterabschnitt 2.1.2 erwähnten Innere-Punkte-Verfahren an, die das SDP (2.7) in polynomieller Laufzeit bis zu einer Genauigkeit  $\varepsilon$  lösen.

Anschließend muss die Zerlegung  $B^T B$  von  $X$  bestimmt werden. Da  $X$  im allgemeinen nicht positiv definit, sondern nur semidefinit ist, verwenden wir eine bekannte Erweiterung der Cholesky-Zerlegung:

#### Singuläre Cholesky-Zerlegung

Ein effektives Verfahren, wie man zu einer positiv semidefiniten Matrix  $X$  möglichst schnell eine Zerlegung in  $X = B^T B$  mit vollem Zeilenrang von  $B$  erhalten kann, wird in einer Übungsaufgabe in [10, Seite 90, P5.2-3] angedeutet. Es benötigt  $\mathcal{O}(n^3)$  Schritte.<sup>4</sup>

Sei die positiv semidefinite Matrix  $X$  wie folgt gegeben:

$$X = \begin{bmatrix} \alpha & v^T \\ v & A \end{bmatrix} \quad \text{mit } \alpha \in \mathbb{R}, v \in \mathbb{R}^{n-1}, A \in \mathbb{R}^{(n-1) \times (n-1)}$$

Dann kann man  $X$  auch schreiben als

$$X = \begin{bmatrix} 1 & 0 \\ \frac{v}{\alpha} & I_{n-1} \end{bmatrix} \begin{bmatrix} \alpha & 0 \\ 0 & A - \frac{vv^T}{\alpha} \end{bmatrix} \begin{bmatrix} 1 & \frac{v^T}{\alpha} \\ 0 & I_{n-1} \end{bmatrix}$$

Da  $A - \frac{vv^T}{\alpha}$  wiederum symmetrisch und positiv semidefinit ist, hat man mittels Pivot-suche (hier genügt es, die Diagonale zu betrachten) nach  $r = \text{Rang}(X)$  Schritten die Dekomposition

$$P^T X P = L D L^T$$

berechnet ( $P$  Permutationsmatrix,  $D \in \mathbb{R}^{r \times r}$  Diagonalmatrix und  $L \in \mathbb{R}^{n \times r}$  untere Einheits-Dreiecksmatrix). Die Matrix  $B$  ergibt sich dann aus  $D' L^T$ , wobei man aus  $D = \text{Diag}(d_1, \dots, d_r)$  die Wurzel  $D' = \text{Diag}(\sqrt{d_1}, \dots, \sqrt{d_r})$  berechnet.

#### Zufallslösungen

Nun kann man mit den in  $B$  enthaltenen Einheitsvektoren  $v_1, \dots, v_n \in \mathbb{R}^m$  die beiden folgenden Schritte durchführen, um eine zufällige Hyperebene und darauf basierend eine Näherungslösung zu erzeugen:

---

<sup>4</sup>Mit  $f(n) = \mathcal{O}(g(n))$  ist in der vorliegenden Arbeit gemeint, dass  $f$  höchstens von der Größenordnung  $g$  ist, d. h.  $(\exists c, n_0 \in \mathbb{N}) (\forall n \geq n_0) : f(n) \leq c \cdot g(n)$ .

1. Zufallsvektor  $r \in \mathbb{R}^m$  (normalverteilt) erzeugen
2. Setze  $U := \{i | v_i^T r \geq 0\}$  als eine Partition des Schnittes fest:  $x^* := \text{sgn}(B^T r)$ .

Damit erhält man auf der Grundlage der SDP-Lösung in polynomieller Laufzeit einen zufälligen Schnitt  $\delta(U)$ . Wir bezeichnen daher diese beiden Schritte zur Erzeugung einer Zufallslösung als „Erzeugung eines Zufallsschnittes basierend auf  $B$ “, oder kurz als „Zufallsschnitt-Erzeugung“.

Durch Wiederholung der Zufallsschnitt-Erzeugung, die nur geringen Rechenaufwand erfordert, kann man natürlich versuchen, die Näherungslösung  $x^*$  zu verbessern.

### Übliche Anwendung

Das Verfahren von Goemans und Williamson hat auf Grund seiner Zufallskomponente die Schwäche, nur im Mittel seine recht gute Näherungslösungen zu erzeugen. Aus diesem Grund muss man das Verfahren entweder derandomisieren (siehe [22]) oder die Zufallsschnitt-Erzeugung iteriert anwenden. Bei der wiederholten Anwendung der Zufallsschnitt-Erzeugung sind  $5n$  Iterationen gebräuchlich. Da die Berechnung eines Gewichts bereits  $\Theta(n^2)$  Schritte erfordert, liegt die Gesamtlaufzeit der Zufallsschnitt-Erzeugung somit in  $\Theta(n^3)$ .

Die so generierten Näherungslösungen sind im Allgemeinen nicht lokal optimal, so dass man üblicherweise an jede Iteration eine lokale Suche anschließt. Die lokale Suche hat jedoch keine konstante Laufzeit – schlimmstenfalls kann sie sogar exponentielle Laufzeit haben. Unter der Annahme, dass ein konstanter Anteil der Knoten noch in die andere Menge verschoben wird, ist die Anzahl der lokalen Suchschritte im Mittel aber von der Ordnung  $\Theta(n)$ .<sup>5</sup> Da je Suchschritt nur linearer Rechenaufwand besteht (das Schnittgewicht braucht nicht komplett neu berechnet werden), bleibt unter dieser Annahme die  $5n$ -fache Anwendung der Zufallsschnitt-Erzeugung auch mit lokaler Suche in der Gesamtkomplexität von  $\Theta(n^3)$ .

In numerischen Versuchen hat sich jedoch herausgestellt (siehe Abbildung 6.1 auf Seite 62), dass die obige Annahme nicht gerechtfertigt ist. Vielmehr nimmt die Qualität der Näherungslösungen des Zufallsschnitt-Verfahrens mit steigender Dimension ab, so dass die lokale Suche länger dauert und somit die Gesamtlaufzeit des Verfahrens nicht mehr in  $\mathcal{O}(n^3)$  ist.

#### 2.3.4. Lösungsraum des GW-Algorithmuses

Durch die semidefinite Relaxierung und die oben beschriebene Technik erreicht man eine erhebliche Verkleinerung des Lösungsraumes von ursprünglich  $\{\pm 1\}^n$  bzw.  $\mathbb{R}^n$  (triviales

---

<sup>5</sup>Eine exponentielle Laufzeit kann man dadurch ausschließen, dass die lokale Suche nach zum Beispiel  $1000n$  Verbesserungen abgebrochen wird.

Würfeln eines Schnittes) in den  $\mathbb{R}^m$ . Dabei ist  $m$ , also der Rang der Optimallösung von (2.7), durch  $\sqrt{2n}$  beschränkt ([9]).

Durch die Rundungstechnik mit der zufälligen Hyperebene (Zufallschnitt-Erzeugung) wird der Raum  $\mathbb{R}^m$  in verschiedene Kegel aufgeteilt, von denen jeweils genau zwei einer Lösung im Ausgangsproblem entsprechen. Da  $m$  in der Praxis oft deutlich kleiner als  $\sqrt{2n}$  ist, kann man auf die Idee kommen, alle möglichen Kegel durchzuprobieren, um dann die beste Lösung daraus zu wählen – obwohl die Anzahl der Kegel nur exponentiell beschränkt ist. Diese Idee wird dadurch bestärkt, dass man mit der Zufallstechnik in Anwendungen oft die Optimallösung des Max-Cut-Problems findet. Allerdings ist die Hoffnung, auf diese Weise vielleicht immer die Optimallösung des Max-Cut-Problems zu finden, zum Scheitern verurteilt: Es gibt Beispiele, auch für Werte von  $n \leq 5$ , bei denen *kein* Kegel die Optimallösung repräsentiert (siehe zum Beispiel [13]). Es ist in diesen Fällen also unmöglich, mit dem Verfahren von Goemans und Williamson die Optimallösung des Ausgangsproblems zu finden.

Auf der anderen Seite kann auch der Fall auftreten, dass die Lösung der semidefiniten Relaxierung (2.7) den Rang 1 hat. Dann ist die Lösung des SDPs auch eine zulässige Lösung des Max-Cut-Problems und somit Optimallösung hierfür.

## 2.4. Weitere Analysen des Näherungsverfahrens

Dieses Näherungsverfahren von Goemans und Williamson war der erste wesentliche Fortschritt bei der Approximierung von Max-Cut-Problemen nach fast 20 Jahren. Auf dessen Grundlage entwickelten Mahajan und Ramesh [22] eine deterministische Fassung, die ebenfalls mindestens 87,856% des Optimalwertes erreicht.

Da in praktischen Anwendungen und auch in pseudozufälligen Testserien oft erheblich bessere Approximationsgüten als die garantierten 87% erreicht wurden, suchte man nach Möglichkeiten die theoretischen Resultate zu verschärfen – sei es durch Negativbeispiele, sei es durch Beweise besserer Approximationsgüten.

### 2.4.1. Die Leistungsgarantie

Mit Approximationsgüte wurde oben das Verhältnis des Wertes einer Näherungslösung  $\tilde{Z}$  zum Optimalwert, d. h.  $\tilde{Z}/Z_{MC}^*$ , bezeichnet. Die Leistungsgarantie  $L_*$  eines Verfahrens ist nun die schlechteste aller mit diesem Verfahren möglichen Approximationsgüten über alle Graphen mit nicht-negativen Kantengewichten (Adjazenzmatrix  $W \geq 0$ ), d. h.

$$L_* := \min_{W \geq 0} \frac{\tilde{Z}}{Z_{MC}^*}. \quad (2.12)$$

Mit Satz 2.10 wurde nun zunächst  $L_* \geq \alpha = 0,87856$  gezeigt. Ob es nun aber überhaupt Graphen gibt, bei denen die untere Schranke  $\alpha$  angenommen wird, war längere

Zeit unklar. Der schlechteste bekannte Wert war zunächst 88,446% für den 5-Zyklus – also  $L_* \leq 0,88446$ .

Später zeigte Karloff [18], dass für jedes  $\varepsilon > 0$  ein Graph existiert, so dass für den Goemans-Williamson-Algorithmus  $E[W] \leq (\alpha + \varepsilon)Z_{MC}^*$  gilt, die Approximationsgüte *im Mittel* also beliebig nahe bei 87,856% liegen kann. Allerdings ist bei den Graphen von Karloff nicht ausgeschlossen, dass trotzdem mit großer Wahrscheinlichkeit eine zufällige Hyperebene eine sehr gute Lösung oder gar die Optimallösung findet. Bei praktischen Implementierungen, bei denen in der Regel mehrere (pseudo)zufällige Hyperebenen erzeugt werden, wäre dann die Wahrscheinlichkeit, eine (fast) optimale Lösung zu finden, dennoch sehr groß.

Die Frage, ob es Graphen gibt, bei denen auch dies ausgeschlossen werden kann, lösten Feige und Schechtman [6] erst im vergangenen Jahr. Sie zeigten, wie man Graphen konstruieren kann, so dass der beste aller mit dem Goemans-Williamson-Verfahren erzeugbaren Schnitte den Wert  $(\alpha + \varepsilon)Z_{MC}^*$  nicht überschreitet. Somit gilt also  $L_* = \alpha$ .

### Hinzunahme von Dreiecksungleichungen

Als Verbesserung des Verfahrens wird häufig vorgeschlagen, die SDP-Relaxierung durch Hinzunahme der Dreiecksungleichungen (1.7) zu verschärfen (siehe auch [12]). Wie im Unterabschnitt 1.1.3 erläutert, führt dies (bei  $n$  Knoten) zu  $4 \binom{n}{3}$  (Dreiecks-)Ungleichungen und zu einer erheblichen Vergrößerung des Aufwands zur Lösung der Relaxierung – auch wenn man sukzessive nur die (wenigen) Dreiecksungleichungen in das SDP aufnimmt, die von den jeweiligen Iterierten verletzt werden.

Wir bezeichnen nun mit  $(SDP\Delta)$  dasjenige Problem, welches entsteht, wenn man zum Problem (2.7) noch alle  $4 \binom{n}{3}$  Dreiecksungleichungen hinzunimmt.

Für den allgemeinen Fall eines beliebigen Graphen konnte bislang auch bei Hinzunahme aller Dreiecksungleichungen, d. h. für das GW-Verfahren auf Basis von  $(SDP\Delta)$ , noch keine bessere Güte als  $\alpha$  bewiesen werden ([8]). Allerdings waren zunächst keine Fälle unter 95,7% bekannt. Vor dem Hintergrund von [11], in dem gezeigt wurde, dass eine Approximation mit mindestens  $\frac{16}{17} + \varepsilon \approx 0,94117 + \varepsilon$  für jedes positive  $\varepsilon$  NP-hart ist, war die Frage nach der Möglichkeit schlechterer Fälle aber sehr bedeutsam – andernfalls wäre P ja gleich NP.

Schließlich zeigten Feige und Schechtman in [6] ebenfalls, dass man Graphen  $G$  mit der folgenden Eigenschaft konstruieren kann: Für  $G$  erzeugt das GW-Verfahren auf Basis von  $(SDP\Delta)$  nur Schnitte, deren Wert  $(\alpha + \varepsilon)Z_{MC}^*$  nicht überschreitet. Dies bedeutet also, dass die Hinzunahme von Dreiecksungleichungen die Leistungsgarantie nicht verbessert und auch hier  $L_* = \alpha$  gilt.

### Approximationsgüte bei beliebigen Gewichten

Die bisher intensiv beleuchtete Approximationsgüte von 87,856% gilt nur für nicht-negative Kantengewichte  $w_{ij}$ . Sofern die Gewichte  $w_{ij}$  der Kanten auch negative Werte annehmen, gilt folgende Abschätzung für den Erwartungswert  $E[W]$  (vgl. [9]):

$$(E[W] - W_-) \geq \alpha \left( \frac{1}{2} \sum_{i < j} w_{ij} (1 - v_i^T v_j) - W_- \right) \quad (2.13)$$

wobei

$$W_- := \frac{1}{2} \sum_{w_{ij} < 0} w_{ij}$$

die Summe aller negativen Gewichte ist.

Diese Schranke kann in speziellen Fällen verbessert werden: Falls die Laplace-Matrix  $L$  des Graphen positiv semidefinit ist, gilt folgende Abschätzung, die von Nesterov [23] bewiesen wurde.

$$E[W] \geq \frac{2}{\pi} \left( \frac{1}{4} \text{Spur } LY \right).$$

#### 2.4.2. Das Integritätsverhältnis

Die bisherigen Betrachtungen konzentrierten sich auf die Approximationsgüte (bzw. die Leistungsgarantie  $L_*$ ), d. h. auf das (schlechteste) Verhältnis des Wertes einer Nähe-

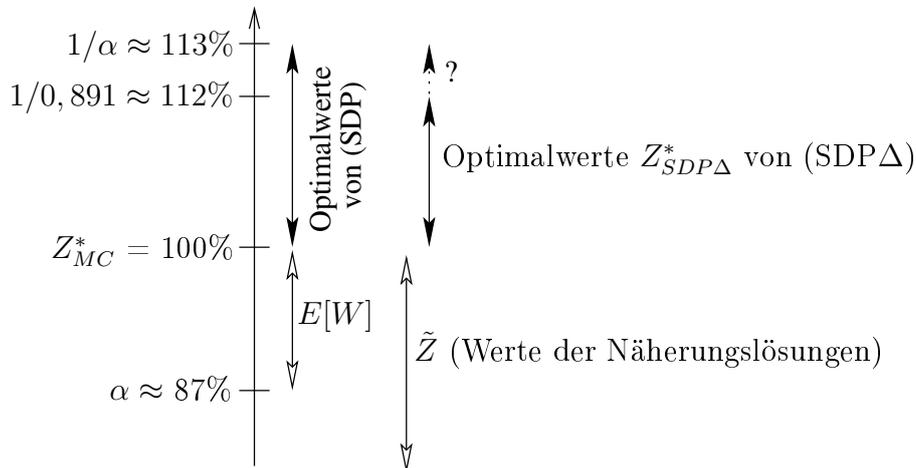


Abbildung 2.2.: Verhältnis von oberen und unteren Schranken an Max-Cut zum Optimalwert  $Z_{MC}^*$

rungslösung zur Optimallösung des Max-Cut-Problems. Ein anderes Verhältnis, das so genannte Integritätsverhältnis  $I_*$ , ist ebenfalls von großer Bedeutung. Darunter wird das schlechteste Verhältnis von Optimalwert des Max-Cut-Problems zum Optimalwert  $R^*$  einer Relaxierung unter allen Graphen mit nicht-negativen Kantengewichten (Adjazenzmatrix  $W \geq 0$ ) verstanden, d. h.  $I_* := \min_{W \geq 0} Z_{MC}^*/R^*$ . Wir definieren speziell für die SDP-Relaxierung (2.7)

$$I_*^{SDP} := \min_{W \geq 0} Z_{MC}^*/Z_{SDP}^*. \quad (2.14)$$

Mit der Arbeit von Goemans und Williamson [9] wurde zunächst nur gezeigt, dass  $I_*^{SDP} \geq \alpha$  gilt. Auch hier blieb lange Zeit die Frage offen, ob es Graphen gibt, für die  $Z_{MC}^*/Z_{SDP}^*$  tatsächlich bei 87,856 % liegt. Zunächst war nur ein Graph mit  $Z_{MC}^*/Z_{SDP}^* \approx 0,8845$  bekannt. Erst Feige und Schechtman [6] zeigten  $I_*^{SDP} \leq \alpha + \varepsilon$  für beliebiges positives  $\varepsilon$ . Also beträgt das Integritätsverhältnis genau  $\alpha$ .

### Weitere Nebenbedingungen

Auch für das Integritätsverhältnis spielen die oben erwähnten Dreiecksungleichungen eine wesentliche Rolle. Ergänzt man das semidefinite Programm (2.7) um alle Dreiecksungleichungen, erhält man die Lösung  $Z_{SDP\Delta}^* \leq Z_{SDP}^*$ . Betrachtet man hierzu wieder das Integritätsverhältnis  $I_*^{SDP\Delta} := \min_{W \geq 0} Z_{MC}^*/Z_{SDP\Delta}^*$ , so gilt auf Grund der Resultate aus [6]

$$0,87856 \leq I_*^{SDP\Delta} \leq 0,891.$$

Ergänzt man das semidefinite Programm (2.7) nicht nur um die Dreiecksungleichungen, sondern um weitere Nebenbedingungen, die alle Restriktionen mit  $k$  Knoten abdecken, erhält man für festes  $k \in \mathbb{N}$  wieder ein SDP mit polynomiell vielen Nebenbedingungen, das also prinzipiell in polynomieller Laufzeit lösbar ist.

Für diese semidefiniten Programme ist der gegenwärtige Forschungsstand allerdings noch unbefriedigend: Für wachsendes  $k$  tendiert die zur Zeit beweisbare obere Schranke an  $I_*$  gegen 1. Sollte  $I_*$  aber tatsächlich größer als  $16/17 \approx 0,94117$  sein, so wäre  $P=NP$  ([11]).

## 3. Weitere Verfahren

### 3.1. Rang-2-Relaxierung

In [3] präsentieren Burer et al. ein heuristisches Verfahren, das auf der Theorie des in Kapitel 2 vorgestellten Verfahren von Goemans und Williamson basiert. Das heuristische Verfahren von Burer et al. schneidet im numerischen Vergleich mit dem Goemans-Williamson-Verfahren ohne anschließende lokale Suche sehr gut ab. Durch eine anders gewählte Relaxierung wird eine höhere Rechengeschwindigkeit erreicht, was die Anwendung des Verfahrens für größere Probleme attraktiv macht. Allerdings geht dabei die Leistungsgarantie von Goemans und Williamson verloren.

#### 3.1.1. Idee

Die semidefinite Relaxierung des Max-Cut-Problems wurde im vorhergehenden Kapitel unter anderem als Erweiterung der Variablen  $x_i \in \pm 1$  auf  $v_i \in S_n$  motiviert (siehe (2.8)):

$$\max \frac{1}{2} \sum_{i < j} w_{ij} (1 - v_i^T v_j)$$

mit:  $v_i \in S_n$

Burer et al. ersetzen nun die Vektoren  $v_i \in S_n$  durch zweidimensionale Vektoren  $v_i \in S_2$  auf dem Einheitskreis:

$$\max \frac{1}{2} \sum_{i < j} w_{ij} (1 - v_i^T v_j) \tag{3.1}$$

mit:  $v_i \in S_2$

Dies kann dahingehend interpretiert werden, dass die Bedingung  $\text{Rang}(X) = 1$  in (2.6) durch  $\text{Rang}(X) \leq 2$  ersetzt wird. Aus diesem Grund spricht man auch von einer Rang-2-Relaxierung.

Konkret haben die Vektoren  $v_i$  unter Verwendung eines Vektors  $\theta = (\theta_1 \dots \theta_n)^T \in \mathbb{R}^n$  die Darstellung

$$v_i = \begin{pmatrix} \cos \theta_i \\ \sin \theta_i \end{pmatrix} \text{ für alle } i \in \{1 \dots n\}, \tag{3.2}$$

und man kann leicht das Skalarprodukt zweier solcher Vektoren herleiten:

$$v_i^T v_j = \cos(\theta_i - \theta_j). \tag{3.3}$$

Definiert man nun die schiefsymmetrische Matrix  $T(\theta)$  durch  $T_{ij}(\theta) := \theta_i - \theta_j$  für alle  $i, j \in \{1 \dots n\}$ , so erhält man folgende Formulierung der Rang-2-Relaxierung des Max-Cut-Problems:

$$\begin{aligned} \min \quad & \frac{1}{2} W \bullet \cos[T(\theta)] \\ \text{mit: } & \theta \in \mathbb{R}^n \end{aligned} \tag{3.4}$$

wobei der Kosinus elementweise auf die Einträge der Matrix  $T(\theta)$  angewendet wird.

Anstelle des restringierten Optimierungsproblems (2.7) über  $\mathcal{O}(n^2)$  Variablen im Verfahren von Goemans und Williamson muss hier nur ein unbeschränktes Optimierungsproblem über  $n$  Variablen gelöst werden. Andererseits ist die Zielfunktion  $f(\theta) := \frac{1}{2} W \bullet \cos[T(\theta)]$  nun nicht mehr linear (sogar nicht-konvex) und hat im allgemeinen stationäre Stellen, die keine (lokalen) Minima sind.

### Charakterisierung der Zielfunktion

Die nicht-konvexe Zielfunktion  $f$  ist offensichtlich periodisch mit Periode  $2\pi$  und invariant gegenüber Rotation ( $f(\theta) = f(\theta + \alpha e) \forall \alpha \in \mathbb{R}$ ).

Ein Vektor  $\bar{\theta} \in \mathbb{R}^n$  ist eine Winkeldarstellung eines Schnitts, falls ganze Zahlen  $k_{ij}$  existieren, so dass  $\bar{\theta}_i - \bar{\theta}_j = k_{ij}\pi$ , d. h. es existiert ein Vektor  $\bar{x} \in \{-1, 1\}^n$ , so dass

$$\cos(\bar{\theta}_i - \bar{\theta}_j) = \bar{x}_i \bar{x}_j \in \{-1, 1\} \tag{3.5}$$

In diesem Fall ist wegen  $\sin(\bar{\theta}_i - \bar{\theta}_j) = 0$  die Ableitung der Zielfunktion

$$\nabla f(\theta) = (W \circ \sin[T(\theta)])^T e. \tag{3.6}$$

gleich Null. Jede Winkeldarstellung  $\bar{\theta}$  eines Schnitts ist also ein stationärer Punkt von  $f$ .

Für das Rang-2-Verfahren ist eine Eigenschaft von  $f$  grundlegend, die im Folgenden Satz charakterisiert wird.

**Satz 3.1** *Sei  $\bar{\theta}$  eine Winkeldarstellung eines Schnitts und sei  $\bar{x} = x(\bar{\theta})$  der zugehörige Schnitt. Dann gilt:*

*Ist  $\bar{\theta}$  ein lokales Minimum (bzw. Maximum) von  $f(\theta)$ , so ist  $\bar{x}$  ein maximaler (bzw. minimaler) Schnitt.*

*Beweis:* Siehe [3]. Der Beweis nutzt die Hessematrix von  $f$ , um die Optimalität elementar herzuleiten. □

Da jeder Schnitt  $\bar{\theta}$  ein stationärer Punkt von  $f$  ist, hat  $f$  also im allgemeinen exponentiell viele Sattelpunkte ( $2^{n-1}$  Schnitte abzüglich den optimalen bzw. „anti-optimalen“ Schnitten). Diese bereiten bei der Minimierung von  $f$  Schwierigkeiten.

Im übrigen gilt die Umkehrung dieses Satzes nicht für allgemeine Graphen, d. h. die Winkeldarstellung  $\bar{\theta}$  einer Optimallösung des Max-Cut-Problems kann auch Sattelpunkt von  $f$  sein. In diesem Fall gibt es ein (globales) Minimum  $\theta'$  von  $f$ , das keine Winkeldarstellung eines Schnittes ist. Es handelt sich um das typische Problem bei Relaxierungen: Die Optimallösung der Relaxierung ist nicht zulässig für das Ausgangsproblem.

Als Beispiel hierfür wird in [3] der ungewichtete vollständige Graph  $K_3$  mit drei Ecken angegeben, für den zum Beispiel der binäre Schnitt  $\bar{x} = (1, -1, -1)^T$  maximal ist. Die Hesse-Matrix von  $f$  ist jedoch nicht positiv semidefinit:

$$\nabla^2 f(\bar{\theta}) = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 0 & 1 \\ -1 & 1 & 0 \end{pmatrix} \quad (3.7)$$

Sie hat die Eigenwerte -1, 0 und 3. Also kann  $f$  kein lokales Minimum in  $\bar{\theta}$  haben.

Als Nachweis dafür, dass Satz 3.1 über eine nicht-leere Menge eine Aussage macht, d. h. dass es Schnitte  $\bar{\theta}$  gibt, die lokal minimal sind, wird in [3] folgendes gezeigt:

Sei  $G$  ein bipartiter Graph mit nicht-negativen Gewichten. Dann gibt es einen maximalen Schnitt  $\bar{x}$ , so dass dessen Winkeldarstellung  $\bar{\theta}$  ein globales Minimum von  $f(\theta)$  ist.

Allerdings ist das Max-Cut-Problem bei bipartiten Graphen mit nicht-negativen Gewichten trivial (die Schnittmengen des Max-Cut sind die beiden Partitionen des bipartiten Graphen). Es bleibt also offen, ob Satz 3.1 für nicht-triviale Probleme relevant ist.

### 3.1.2. Verfahren

Die oben angesprochene ungünstige Situation der stationären Punkte wird noch dadurch verschärft, dass es lokale Minima von  $f$  gibt, die keine Winkeldarstellung eines Schnitts sind (d. h.  $\text{Rang } \cos[T(\theta)] = 2$ ). In [3] wird folgendes heuristische Verfahren vorgeschlagen (Details siehe unten):

1. Ausgehend von einem Startpunkt  $\theta^0$  suche ein lokales Minimum  $\theta^1$ .
2. Mittels  $\cos[T(\theta^1)]$  suche den besten Schnitt  $\bar{x}^1$ .
3. Setze  $\theta^0$  gleich der Winkeldarstellung von  $\bar{x}^1$  mit einer kleinen Störung und fange wieder von vorne an.

Die Schritte werden nun in einer Schleife durchlaufen. Dabei merkt man sich den besten bisher gefundenen Schnitt  $x^*$ . Die Schleife wird solange wiederholt, bis  $N$  Mal hintereinander  $x^*$  nicht verbessert wird.

In [3] wird auch vorgeschlagen, die Schleife ggf. mehrfach mit einem gänzlich neuen Startpunkt  $\theta^0$  zu beginnen (etwa  $M$  Mal).

Natürlich wird das Endergebnis meist besser, wenn man  $N$  und  $M$  größer wählt. In [3] werden aber bei praktischen Tests schon für  $N = 5$  und  $M = 100$  gute Ergebnisse erzielt.

### Zu den einzelnen Schritten

Im zweiten Schritt könnte man mit der Matrix  $\cos[T(\theta)]$  die Goemans-Williamson-Zufallswürfe durchführen. Da es sich aber i. a. immer um eine Rang-2-Matrix handelt (bei Rang gleich eins wäre die Optimallösung gefunden), ist es sinnvoller, deterministisch alle  $n$  möglichen Schnitte durchzuprobieren (Winkel  $\theta_i$  sortieren und mit den Zwischenwerten jeweils einen Goemans-Williamson-Schnitt bilden).

Im dritten Schritt geht man zunächst davon aus, dass bei  $\theta^0$  ein Sattelpunkt von  $f$  vorliegt. Um die lokale Minimierung wieder starten zu können, benötigt man eine kleine Störung. Man hofft auf diese Weise, ein anderes/besseres lokales Minimum von  $f$  zu finden.

### 3.1.3. Analyse

Die Grundidee des Verfahrens ist auf den ersten Blick bestechend: Reduzierung der Relaxierung von  $\text{Rang } X \leq n$  (Goemans-Williamson) auf  $\text{Rang } X \leq 2$ . Vom theoretischen Standpunkt her gesehen, geht jedoch dabei viel verloren: Das globale Minimum des resultierenden Problems (3.4) ist nicht mehr mit Methoden der lokalen Optimierung zu berechnen. Zum zweiten gilt die Leistungsgarantie von 87% des Verfahrens von Goemans und Williamson nicht mehr. Dies liegt unter anderem in Schritt Zwei des Rang-2-Verfahrens begründet: Dort ist zwar garantiert, dass  $\bar{x}^1$  ein Schnitt mit mindestens 87% des Gewichts von  $\theta^1$  ist – allerdings ist das Gewicht von  $\theta^1$  keine obere Schranke der Max-Cut-Optimallösung.

Wie zu Beginn des Abschnittes bereits erwähnt, handelt es sich also bei diesem Verfahren um eine Heuristik, d. h. es gibt keine Leistungsgarantie. Auch gibt es keine obere Schranke an die Optimallösung des Max-Cut-Problems, wie man sie durch die SDP-Relaxierung beim Goemans-Williamson-Verfahren erhält.

Auf der anderen Seite sind die in [3] aufgeführten positiven praktischen Tests zu erwähnen. Danach ist bei numerisch ähnlichen Ergebnissen die Laufzeit gegenüber dem Goemans-Williamson-Verfahren erheblich kürzer. Das Rang-2-Verfahren ist also auch für sehr große Probleme anwendbar.

Bisher sind keine theoretischen Erkenntnisse bekannt, die über die obigen Resultate hinaus gehen. Es ist also in der Max-Cut-Approximation sehr schwer, gute numerische Ergebnisse auch theoretisch zu untermauern.

### 3.2. Branch and Bound nach Goemans

In [8] skizziert Goemans, wie man die Lösungen des primalen Max-Cut-SDP (2.7) und seines dualen Pendant (2.10) in einem Branch-and-Bound-Schema verwenden kann. Dieses sehr effiziente Verfahren soll nachfolgend etwas ausführlicher erläutert werden. Dabei sei  $W = (w_{ij})_{i,j}$  wie immer die Gewichtsmatrix eines Graphen  $G$  mit  $n$  Knoten und  $L$  die Laplace-Matrix von  $W$ . Mit  $Z_{MC}^*$  bezeichnen wir wie bisher den Optimalwert des Max-Cut-Problems, mit  $Z_{SDP}^*$  den Optimalwert der semidefiniten Relaxierung (2.7) sowie mit  $Z_{DSDP}^*$  den Optimalwert des dualen SDPs (2.10).

Zunächst ist festzustellen, dass das duale Programm des Max-Cut-SDPs mittels einer Cholesky-Zerlegung  $U^T U$  von  $\text{Diag}(y) - L \succeq 0$  wie folgt umgeformt werden kann (wobei mit  $u_i$  die  $i$ -te Spalte von  $U$  bezeichnet wird):

$$(2.10) = \frac{1}{4} \min \left\{ \sum_i y_i \mid \text{Diag}(y) - L \succeq 0, y \in \mathbb{R}^n \right\} \quad (3.8)$$

$$= \frac{1}{4} \min \left\{ \sum_i y_i \mid U \in \mathbb{R}^{n \times n}, u_i^T u_j = w_{ij} \forall i \neq j, u_i^T u_i = - \sum_{j=1}^n w_{ij} + y_i \forall i \right\} \quad (3.9)$$

$$= \frac{1}{4} \min \left\{ \sum_i y_i \mid U \in \mathbb{R}^{n \times n}, u_i^T u_j = w_{ij} \forall i \neq j, \sum_{j=1}^n u_i^T u_j = y_i \forall i \right\} \quad (3.10)$$

$$= \frac{1}{4} \min \left\{ \sum_i \sum_j u_i^T u_j \mid U \in \mathbb{R}^{n \times n}, u_i^T u_j = w_{ij} \forall i \neq j \right\} \quad (3.11)$$

$$= \frac{1}{4} \min \left\{ \sum_i u_i^T \sum_j u_j \mid U \in \mathbb{R}^{n \times n}, u_i^T u_j = w_{ij} \forall i \neq j \right\} \quad (3.12)$$

$$= \frac{1}{4} \min \left\{ \left\| \sum_i u_i \right\|^2 \mid U \in \mathbb{R}^{n \times n}, u_i^T u_j = w_{ij} \forall i \neq j \right\} \quad (3.13)$$

Sei nun  $U \in \mathbb{R}^{n \times n}$  eine Matrix mit  $u_i^T u_j = w_{ij} \forall i \neq j$ . Sei  $\delta(S)$  ein beliebiger Schnitt. Dann gilt für dessen Gewicht

$$w(\delta(S)) = \sum_{i \in S} \sum_{j \notin S} w_{ij} = \left( \sum_{i \in S} u_i \right)^T \left( \sum_{j \notin S} u_j \right) = \frac{1}{4} (b+z)^T (b-z) \quad (3.14)$$

$$= \frac{1}{4} (\|b\|^2 - \|z\|^2), \quad (3.15)$$

wobei  $b := \sum_i u_i$  und  $z := \sum_{i \in S} u_i - \sum_{j \notin S} u_j$  definiert werden.

Mit der trivialen Abschätzung  $\|z\|^2 \geq 0$  hat man zunächst  $w(\delta(S)) \leq \frac{1}{4} \|b\|^2 = Z_{DSDP}^*$ . Da diese Beziehung für beliebige Schnitte  $\delta(S)$  gilt, erhält man das bekannte Resultat

$$Z_{MC}^* \leq Z_{DSDP}^*. \quad (3.16)$$

Allerdings kann man auch eine bessere Abschätzung von  $\|z\|^2$  konstruieren: Im Branch-and-Bound-Algorithmus werden in jedem BaB-Verzweigungsknoten einige Graphenknoten auf die Mengen  $S$  und  $V \setminus S$  fixiert: In einem BaB-Verzweigungsknoten  $\nu$  der Ebene  $n - l$  (die Wurzel ist die Ebene  $n$ ) werden die Knoten  $\{l + 1, l + 2, \dots, n\}$  mit einer Abbildung

$$\tau_\nu : \{l + 1, l + 2, \dots, n\} \rightarrow \{0, 1\}$$

auf  $S$  bzw.  $V \setminus S$  festgelegt. Damit gilt für alle Schnitte, die noch in  $\nu$  in Frage kommen, für die  $k$ -te Komponente  $z^{(k)}$  des Vektors  $z$  die Abschätzung

$$|z^{(k)}| = \left| \sum_{i \in S} u_i^{(k)} - \sum_{j \notin S} u_j^{(k)} \right| \geq \delta_\nu^{(k)} := \begin{cases} 0 & \text{falls } k \leq l, \\ |z_\nu^{(k)}| & \text{falls } k > l, \end{cases} \quad (3.17)$$

mit

$$z_\nu := \sum_{i > l: \tau_\nu(i)=1} u_i - \sum_{i > l: \tau_\nu(i)=0} u_i, \quad (3.18)$$

da höchstens die ersten  $i$  Komponenten  $u_i^{(1)}, \dots, u_i^{(i)}$  eines jeden Vektors  $u_i$  ungleich 0 sind, weil die  $u_i$  aus einer Cholesky-Zerlegung stammen.

Damit erhalten wir für den BaB-Verzweigungsknoten  $\nu$  die Abschätzung

$$Z_{MC_\nu}^* \leq Z_{DSDP}^* - \frac{1}{4} \|\delta_\nu\|^2, \quad (3.19)$$

wobei  $Z_{MC_\nu}^*$  den Optimalwert aller in diesem Verzweigungsknoten noch in Frage kommenden Schnitte  $\delta(S)$  bezeichnet.

Der Hauptvorteil dieses Branch-and-Bound-Verfahrens nach Goemans liegt darin, dass die Berechnung der oberen Schranke in einem Verzweigungsknoten nur sehr geringen Rechenaufwand erfordert: Wenn einmal die Lösung  $y$  des dualen Programms (2.10) und die Cholesky-Zerlegung von  $\text{Diag}(y) - L$  berechnet sind, kann man in kurzer Zeit viele BaB-Verzweigungsknoten ausloten.

Wir möchten zu diesem Verfahren noch zwei ergänzende Feststellungen machen:

**Bemerkung 3.2** *In einem Verzweigungsknoten der Ebene  $n - n = 0$  ist die Abschätzung (3.19) scharf, d. h. es gilt  $Z_{MC_\nu}^* = Z_{DSDP}^* - \frac{1}{4} \|\delta_\nu\|^2 = w(\delta(S))$  für den einen möglichen Schnitt  $\delta(S)$ .*

**Bemerkung 3.3** *Im Falle ganzzahliger Gewichte  $w_{ij}$  lässt sich die Abschätzung (3.19) noch etwas verbessern, da für die Cholesky-Zerlegung im allgemeinen nicht mehr  $U \in \mathbb{Z}^{n \times n}$  gilt, aber die Optimallösung ganzzahlig ist:*

$$Z_{MC_\nu}^* \leq \left\lfloor Z_{DSDP}^* - \frac{1}{4} \|\delta_\nu\|^2 \right\rfloor. \quad (3.20)$$

## 4. Die trigonometrische Approximation

### 4.1. Motivation

Im Rahmen der Analyse des Näherungsverfahrens von Goemans und Williamson [9] ergibt sich eine neue Formulierung des Max-Cut-Problems:

$$\max \sum_{i < j} w_{ij} \frac{\arccos(v_i^T v_j)}{\pi} \quad (4.1)$$

mit:  $v_i \in S_n$

**Satz 4.1 ([9])** Sei  $Z_{4.1}^*$  der Optimalwert von (4.1) und  $Z_{MC}^*$  der Optimalwert des Max-Cut-Problems (1.2). Dann gilt

$$Z_{4.1}^* = Z_{MC}^*. \quad (4.2)$$

*Beweis:* Zunächst einmal gilt  $Z_{4.1}^* \geq Z_{MC}^*$ , denn (4.1) ist eine Relaxierung von (1.2): Sofern  $v_i$  in einem eindimensionalen Raum liegen, reduziert sich die Zielfunktion von (4.1) zu  $\frac{1}{2} \sum_{i < j} w_{ij} (1 - v_i v_j)$ , d. h. derjenigen von (1.2).

Andererseits gilt: Seien  $\bar{v}_i$  die Optimallösungen von (4.1). Dann folgt aus Satz 2.8, dass der randomisierte Algorithmus von Goemans und Williamson mit den  $\bar{v}_i$  einen Schnitt liefert, dessen Erwartungswert  $\sum_{i < j} w_{ij} \frac{\arccos(\bar{v}_i^T \bar{v}_j)}{\pi}$  ist, also genau  $Z_{4.1}^*$ . Es muss also einen Schnitt  $\delta$  geben, für dessen Gewicht  $w(\delta)$  gilt:  $w(\delta) \geq Z_{4.1}^*$ . Also gilt  $Z_{4.1}^* \leq Z_{MC}^*$ .  $\square$

Die Formulierung (4.1) kann leicht in eine kompaktere Form gebracht werden, die wieder auf die Laplace-Matrix von  $W$  zurückgreift: Entsprechend der Argumentation im Anschluss an (2.8) auf Seite 20 ist (4.1) äquivalent zu

$$\max \sum_{i < j} w_{ij} \frac{\arccos(z_{ij})}{\pi}$$

mit:  $\text{diag}(Z) = e$   
 $Z \succeq 0$

Mit

$$\begin{aligned}
\frac{1}{\pi} \sum_{i < j} w_{ij} \arccos(x_{ij}) &= \frac{1}{2\pi} \sum_{i,j} w_{ij} \arccos(x_{ij}) = \frac{1}{2\pi} \sum_{i,j} w_{ij} \left( \frac{\pi}{2} - \arcsin(x_{ij}) \right) \\
&= \frac{1}{2\pi} \sum_{i,j} \frac{\pi}{2} w_{ij} - \frac{1}{2\pi} \sum_{i,j} w_{ij} \arcsin(x_{ij}) \\
&= \frac{1}{2\pi} \sum_i \frac{\pi}{2} l_i + \frac{1}{2\pi} \sum_{i,j} l_{ij} \arcsin(x_{ij}) \\
&= \frac{1}{2\pi} \sum_i \arcsin(x_{ii}) l_i + \frac{1}{2\pi} \sum_{i,j} l_{ij} \arcsin(x_{ij}) \\
&= \frac{1}{2\pi} \sum_{i,j} l_{ij} \arcsin(x_{ij}) = \frac{1}{2\pi} L \bullet \arcsin[Z]
\end{aligned}$$

erhält man dann

$$\max \left\{ \frac{1}{4} L \bullet \frac{2}{\pi} \arcsin[Z] \mid Z \in \mathcal{SR} \right\} \quad (4.3)$$

Die Technik, allgemeine quadratische 0-1-Programme mit Hilfe der Funktion  $\arcsin$  über semidefinite Matrizen zu formulieren, hat erstmals Nesterov [24] in allgemeiner Form angewandt. Er ergänzte dieses Resultat auch durch eine alternative Herleitung. Allerdings untersuchte Nesterov die Formulierung (4.3) nicht weiter. Er nutze sie vielmehr bei allgemeineren quadratischen 0-1-Problemen, um eine Qualitätsabschätzung von semidefiniten Relaxierungen dieser quadratischen Probleme herzuleiten.

Da lineare Zielfunktionen angenehmer sind, formulieren wir (4.3) noch um in

$$\max \left\{ \frac{1}{4} L \bullet X \mid X \in \mathcal{TA} \right\} \quad (4.4)$$

mit

$$\mathcal{TA} := \left\{ \frac{2}{\pi} \arcsin[Z] \mid Z \in \mathcal{SR} \right\}. \quad (4.5)$$

Die Abkürzung  $\mathcal{TA}$  ist motiviert durch die Feststellung, dass es sich bei der Menge  $\mathcal{TA}$  um eine trigonometrische Approximation des Max-Cut-Polytops handelt. Da die Nebenbedingungen von (4.4) eine semidefinite Ungleichung für eine nichtkonvexe Funktion der Variablen  $X$  enthalten, bezeichnen wir im folgenden (4.4) als **nichtkonvexes SDP**. Genauere Eigenschaften von  $\mathcal{TA}$  und von (4.4) werden in den weiteren Abschnitten dieses Kapitels erläutert.

Da es sich bei  $f : x \mapsto \frac{2}{\pi} \arcsin(x)$  um eine bijektive Abbildung von  $[-1, 1]$  auf  $[-1, 1]$  handelt, kann die Formulierung (4.5) durch Substitution ( $X := \frac{2}{\pi} \arcsin[Z]$ ) leicht zu

$$\mathcal{TA} = \left\{ X \mid \sin\left[\frac{\pi}{2} X\right] \succeq 0, \quad X \succeq 0, \quad \text{diag}(X) = e \right\} \quad (4.6)$$

#### 4. Die trigonometrische Approximation

---

umgeformt werden. Dabei ist  $X \in [-1, 1]^{n \times n}$  durch  $X \succeq 0$ ,  $\text{diag}(X) = e$  sichergestellt. Im übrigen gilt folgendes Lemma:

**Lemma 4.2** *Für symmetrische Matrizen  $X = (x_{ij})$  mit Einträgen  $|x_{ij}| \leq 1$  gilt:*

$$\sin\left[\frac{\pi}{2}X\right] \succeq 0 \Rightarrow X \succeq 0$$

bzw.

$$X \succeq 0 \Rightarrow \frac{2}{\pi} \arcsin[X] \succeq 0$$

*Beweis:* Die beiden Aussagen sind äquivalent, da  $f : x \mapsto \frac{2}{\pi} \arcsin(x)$  bijektiv  $[-1, 1]$  auf  $[-1, 1]$  abbildet.

Wir zeigen die zweite Aussage. Dabei betrachten wir die Potenzreihe von  $\arcsin(x)$ , die ausschließlich nicht-negative Koeffizienten besitzt. Das Hadamard-Produkt (auch Schur-Produkt) zweier positiv semidefiniter Matrizen ist wieder positiv semidefinit (so genanntes Schur-Produkt-Theorem, vgl. [14]). Insbesondere folgt somit aus  $X \succeq 0$ , dass auch die elementweisen Potenzen

$$[X]^k = X \circ X \circ \dots \circ X$$

für alle  $k \in \mathbb{N}$  wieder positiv semidefinit sind. Somit ist  $\arcsin[X]$  als Summe von positiven Vielfachen von Potenzen einer positiv semidefiniten Matrix wiederum positiv semidefinit.  $\square$

Unter Ausnutzung der obigen Definition kann die Gleichung (4.2) aus Satz 4.1 auch eingängiger geschrieben werden als

$$\max \left\{ \frac{1}{4} L \bullet X \mid X \in \mathcal{MC} \right\} = \max \left\{ \frac{1}{4} L \bullet X \mid X \in \mathcal{TA} \right\}. \quad (4.7)$$

Wir erhalten daraus folgendes Korollar:

**Korollar 4.3** *Für die Trigonometrische Approximation  $\mathcal{TA}$ , das Max-Cut-Polytop  $\mathcal{MC}$ , und die semidefinite Relaxierung  $\mathcal{SR}$  gelten die Beziehung*

$$\mathcal{TA} \subset \text{conv}(\mathcal{TA}) = \mathcal{MC} \subset \mathcal{SR}. \quad (4.8)$$

*Beweis:* Da die Zielfunktion  $C \bullet X$  in Gleichung (4.7) linear ist, gilt also

$$\max \left\{ \frac{1}{4} L \bullet X \mid X \in \mathcal{MC} \right\} = \max \left\{ \frac{1}{4} L \bullet X \mid X \in \text{conv}(\mathcal{TA}) \right\}.$$

Da diese Gleichung für jedes symmetrische  $C$  gilt, müssen die konvexen Mengen  $\mathcal{MC}$  und  $\text{conv}(\mathcal{TA})$  identisch sein.  $\square$

Insbesondere sind alle Knoten  $xx^T$  mit Komponenten  $x_i \in \{\pm 1\}$  in  $\mathcal{TA}$  enthalten.

## 4.2. Grundlegende Eigenschaften von $\mathcal{TA}$

Zunächst ist aus Korollar 4.3 nur klar, dass  $\mathcal{TA}$  das Max-Cut-Polytop von innen approximiert. Die weitere geometrische Gestalt von  $\mathcal{TA}$  ist nicht offensichtlich und soll nachfolgend näher untersucht werden.

### 4.2.1. Einbeschriebene und umschreibende Kugeln

Die Menge  $\mathcal{TA}$  kann durch eine einbeschriebene und eine umschreibende Kugel charakterisiert werden:

**Satz 4.4**  $\mathcal{TA}$  hat eine zwei-seitige zirkuläre Approximation, nämlich

$$\{X \in \mathcal{L} \mid \|X - I\|_F \leq \frac{2}{\pi}\} \subset \mathcal{TA} \subset \{X \in \mathcal{L} \mid \|X - I\|_F \leq \sqrt{n^2 - n} < n\}.$$

*Beweis:* Sei  $X \in \mathcal{L}, \|X - I\|_F \leq \frac{2}{\pi}$  gegeben. Wegen  $|\sin(\frac{\pi}{2}x)| \leq |\frac{\pi}{2}x|$  für  $x \in [-1, 1]$  gilt  $\|\sin[\frac{\pi}{2}X] - I\|_F \leq 1$ . Da  $\|\cdot\|_F$  eine obere Schranke an den betragsgrößten Eigenwert einer Matrix ist, folgt  $\sin[\frac{\pi}{2}X] \succeq 0$ .

Die umschreibende Kugel folgt sofort aus  $\mathcal{TA} \subset \mathcal{MC} \subset [-1, 1]^{n \times n}$ . □

Aus dem Satz folgt unmittelbar, dass  $\mathcal{TA}$  ein nicht-leeres Inneres hat.

### 4.2.2. Gemeinsame Randflächen mit $\mathcal{MC}$

Es gilt folgender Satz:

**Satz 4.5** (a) Für  $n = 3$  gilt

$$\mathcal{TA} = \mathcal{MC}.$$

(b) Für  $n \geq 3$  sind alle 1- und 2-dimensionalen Randflächen von  $\mathcal{MC}$  auch Randflächen von  $\mathcal{TA}$ .

*Beweis:* (a) Sei  $n = 3$ . Die vier Hyperebenen, die  $\mathcal{MC}$  charakterisieren, sind gegeben durch

$$x_1 + x_2 + x_3 \geq -1, \quad x_1 - x_2 - x_3 \geq -1, \quad -x_1 - x_2 + x_3 \geq -1, \quad -x_1 + x_2 - x_3 \geq -1,$$

wobei  $x_1, x_2, x_3$  die drei Nicht-diagonal-Elemente einer symmetrischen  $3 \times 3$ -Matrix  $X$  bezeichnen. Angenommen,  $X$  liegt auf dem Rand von  $\mathcal{MC}$ . Dann ist mindestens eine der linearen Randbedingungen aktiv. Sei zum Beispiel

$$x_1 - x_2 - x_3 = -1 \quad \text{and} \quad x_i \in [-1, 1].$$

#### 4. Die trigonometrische Approximation

---

(Die drei übrigen Fälle sind analog.) Wir zeigen

$$\sin \left[ \frac{\pi}{2} \begin{pmatrix} 1 & x_1 & x_2 \\ x_1 & 1 & x_3 \\ x_2 & x_3 & 1 \end{pmatrix} \right] \succeq 0.$$

Zunächst sind alle  $2 \times 2$  Hauptunterdeterminanten nichtnegativ. Es genügt daher, die Gesamt-Determinante zu betrachten: Sei  $\tilde{x}_i := \sin \frac{\pi}{2} x_i$ . Die Determinante ist

$$1 - \tilde{x}_1^2 - \tilde{x}_2^2 - \tilde{x}_3^2 + 2\tilde{x}_1\tilde{x}_2\tilde{x}_3. \quad (4.9)$$

Wendet man  $\sin$  auf  $x_1 = x_2 + x_3 - 1$  an, so erhält man zunächst

$$\sin \left( \frac{\pi}{2} x_1 \right) = \sin \left( \frac{\pi}{2} x_2 + \frac{\pi}{2} (x_3 - 1) \right).$$

Unter Verwendung des Additionstheorems  $\sin(\alpha + \beta) = \sin \alpha \cos \beta + \cos \alpha \sin \beta$  erhält man

$$\tilde{x}_1 = \sin \left( \frac{\pi}{2} x_2 \right) \cos \left( \frac{\pi}{2} (x_3 - 1) \right) + \cos \left( \frac{\pi}{2} x_2 \right) \sin \left( \frac{\pi}{2} (x_3 - 1) \right).$$

Die Anwendung weiterer Additionstheoreme, nämlich  $\sin(\alpha - \beta) = \sin \alpha \cos \beta - \cos \alpha \sin \beta$  und  $\cos(\alpha - \beta) = \cos \alpha \cos \beta + \sin \alpha \sin \beta$  führt zu

$$\begin{aligned} \tilde{x}_1 = & \sin \left( \frac{\pi}{2} x_2 \right) \left( \cos \left( \frac{\pi}{2} x_3 \right) \cos \frac{\pi}{2} + \sin \left( \frac{\pi}{2} x_3 \right) \sin \frac{\pi}{2} \right) \\ & + \cos \left( \frac{\pi}{2} x_2 \right) \left( \sin \left( \frac{\pi}{2} x_3 \right) \cos \frac{\pi}{2} - \cos \left( \frac{\pi}{2} x_3 \right) \sin \frac{\pi}{2} \right). \end{aligned}$$

Wendet man nun die trigonometrische Identität  $\sin^2 \alpha + \cos^2 \alpha = 1$  an, so erhält man

$$\tilde{x}_1 = \tilde{x}_2\tilde{x}_3 \pm \sqrt{1 - \tilde{x}_2^2} \sqrt{1 - \tilde{x}_3^2}.$$

Fügt man dies in (4.9) ein, so sieht man, dass die Gesamt-Determinante gleich Null ist, d. h. die Matrix ist tatsächlich positiv semidefinit.

(b) Sei nun  $n \geq 3$ . Wir zeigen zuerst, dass alle 2-dimensionalen Randflächen von  $\mathcal{MC}$  in  $\mathcal{TA}$  enthalten sind. Wegen  $\mathcal{TA} \subset \mathcal{MC}$  folgt dann, dass die 2-dimensionalen Randflächen von  $\mathcal{MC}$  auf dem Rand von  $\mathcal{TA}$  liegen.

Nehmen wir also an, dass  $X$  auf einer 2-dimensionalen Rand von  $\mathcal{MC}$  liegt. Dann ist  $X$  darstellbar als Konvexkombination von drei Ecken von  $\mathcal{MC}$  (auch wenn der 2-dimensionale Rand mehr als drei Ecken besitzen sollte), d. h.

$$X = \lambda_1 x x^T + \lambda_2 y y^T + \lambda_3 z z^T$$

mit  $\lambda_i \geq 0$  und  $x, y, z \in \{\pm 1\}^n$ . Dann sind alle  $2 \times 2$  Unterdeterminanten von  $\sin[\frac{\pi}{2} X]$  nichtnegativ, die  $3 \times 3$  Unterdeterminanten können wie im Fall  $n = 3$  behandelt werden, und alle größeren Unterdeterminanten sind singulär. Also gilt  $\sin[\frac{\pi}{2} X] \succeq 0$ .  $\square$

### Optimale Kontraktionseigenschaft

Der erste Teil des Satzes 4.5 weist darauf hin, dass  $\mathcal{TA}$  in gewissem Sinne optimal aus  $\mathcal{SR}$  gebildet wird: Es gilt die Beziehung  $X \in \mathcal{TA} \iff f[X] \in \mathcal{SR}$  mit der Funktion  $f : [-1, 1] \rightarrow [-1, 1]$  mit  $f(t) = \frac{2}{\pi} \arcsin(t)$ . Dabei hat die glatte nichtkonvexe Funktion  $f$  die kontrahierende Eigenschaft  $|f(t)| \leq |t|$ .

Bemerkenswert ist hierbei, dass die Menge  $\mathcal{SR}$  genau soweit von  $f$  zu  $\mathcal{TA}$  zusammengezogen wird, dass im Fall  $n = 3$  die beiden Mengen  $\mathcal{TA}$  und  $\mathcal{MC}$  zusammen fallen und ansonsten  $\mathcal{TA} \subset \mathcal{MC}$  gilt.

### 4.3. Sternförmigkeit

Die Menge  $\mathcal{TA}$  approximiert das Polytop  $\mathcal{MC}$  von innen und enthält auch die Ecken dieses Polytops. Es erscheint also naheliegend zu prüfen, ob  $\mathcal{TA}$  sternförmig ist.

Dazu sei  $X \in \mathcal{TA}$  und  $\lambda \in (0, 1)$  gegeben. Für die Sternförmigkeit ist

$$Y := \lambda X + (1 - \lambda)I \in \mathcal{TA}$$

zu zeigen, d. h.  $\sin[\frac{\pi}{2}Y] \succeq 0$ ,  $Y \succeq 0$  und  $\text{diag}(Y) = e$ . Letzteres ist wegen  $\text{diag}(X) = e$  sofort klar. Auch die zweite Bedingung ist evident, da  $Y$  Summe zweier positiv semidefiniter Matrizen ist. Schwieriger ist hingegen die erste Bedingung,  $\sin[\frac{\pi}{2}Y] \succeq 0$ . Zunächst gilt:

$$\text{diag}\left(\sin\left[\frac{\pi}{2}Y\right]\right) = \text{diag}\left(\sin\left[\frac{\pi}{2}((1 - \lambda)I + \lambda X)\right]\right) = e$$

Somit folgt

$$\sin\left[\frac{\pi}{2}((1 - \lambda)I + \lambda X)\right] = \sin\left[\frac{\pi}{2}\lambda X\right] + \hat{\lambda}I, \quad \text{mit } \hat{\lambda} := 1 - \sin\left(\frac{\pi}{2}\lambda\right) > 0.$$

Sofern also  $\sin[\frac{\pi}{2}\lambda X] \succeq 0$  gilt, gilt auch  $\sin[\frac{\pi}{2}Y] \succeq 0$  und  $Y \in \mathcal{TA}$ .

Die Aussage

$$\sin\left[\frac{\pi}{2}\lambda X\right] \succeq 0 \quad \text{für alle } X \in \mathcal{TA} \text{ und für alle } \lambda \in (0, 1) \quad (4.10)$$

ist äquivalent zu

$$\sin[\lambda \arcsin[Z]] \succeq 0 \quad \text{für alle } Z \in \mathcal{SR} \text{ und für alle } \lambda \in (0, 1). \quad (4.11)$$

Denn mit  $X \in \mathcal{TA}$  ist  $Z := \sin[\frac{\pi}{2}X] \in \mathcal{SR}$  und wegen der Bijektivität von  $\sin[\frac{\pi}{2}X]$  auf  $[-1, 1]^{n \times n}$  gilt

$$\sin\left[\lambda \frac{\pi}{2}X\right] = \sin\left[\frac{\pi}{2}\lambda X\right] = \sin\left[\lambda \arcsin\left[\sin\left[\frac{\pi}{2}X\right]\right]\right] = \sin\left[\lambda \arcsin[Z]\right].$$

#### 4. Die trigonometrische Approximation

---

Andererseits ist mit  $Z \in \mathcal{SR}$  auch  $X := \frac{2}{\pi} \arcsin[Z] \in \mathcal{TA}$  und es gilt

$$\sin[\lambda \arcsin[Z]] = \sin\left[\lambda \frac{\pi}{2} \frac{2}{\pi} \arcsin[Z]\right] = \sin\left[\frac{\pi}{2} \lambda X\right].$$

Um (4.11) zu zeigen, genügt es (der Argumentation im Beweis von Lemma 4.2 folgend) zu zeigen, dass die Potenzreihe der Funktion  $f_\lambda : x \mapsto \sin(\lambda \arcsin(x))$  nur nichtnegative Koeffizienten enthält.

**Lemma 4.6** Die Taylorreihe der Funktion  $f_\lambda : x \mapsto \sin(\lambda \arcsin(x))$  im Punkt 0 ist gegeben durch

$$f_\lambda(x) = \sum_{n=0}^{\infty} \left( \sum_{k=0}^n \gamma_{n,k} \lambda^{2k+1} \right) x^{2n+1}.$$

mit

$$\gamma_{n,k} := \frac{(-1)^k}{(2k+1)!} \left( \sum_{\substack{\alpha \in \mathbb{N}_0^{2k+1} \\ \sum \alpha_i = n-k}} \prod_{i=1}^{2k+1} \frac{(2j-1)^2}{(2\alpha_i+1)!} \right) \text{ für } k \leq n, \quad \gamma_{n,k} := 0 \text{ sonst.} \quad (4.12)$$

*Beweis:* Unter Benutzung der bekannten Taylorreihen von  $\sin$  und  $\arcsin$  erhält man

$$f_\lambda(x) = \sum_{k=0}^{\infty} (-1)^k \frac{(\lambda \arcsin(x))^{2k+1}}{(2k+1)!} \quad (4.13)$$

$$= \sum_{k=0}^{\infty} (-1)^k \frac{\left( \lambda \sum_{n=0}^{\infty} \frac{\prod_{j=1}^n (2j-1)}{(2n+1)!} x^{2n+1} \right)^{2k+1}}{(2k+1)!} \quad (4.14)$$

$$= \sum_{k=0}^{\infty} \frac{(-1)^k \lambda^{2k+1}}{(2k+1)!} \left( \sum_{n=0}^{\infty} \frac{\prod_{j=1}^n (2j-1)^2}{(2n+1)!} x^{2n+1} \right)^{2k+1}. \quad (4.15)$$

Nach Umformung der Potenz der inneren Reihe ergibt sich

$$f_\lambda(x) = \sum_{k=0}^{\infty} \frac{(-1)^k \lambda^{2k+1}}{(2k+1)!} \sum_{l=0}^{\infty} \left( \sum_{\substack{\alpha \in \mathbb{N}_0^{2k+1} \\ \sum \alpha_i = l}} \prod_{i=1}^{2k+1} \frac{(2j-1)^2}{(2\alpha_i+1)!} x^{2l+2k+1} \right) \quad (4.16)$$

und nach Umordnung (absolute Konvergenz!)

$$= \sum_{n=0}^{\infty} \sum_{l=0}^n \frac{(-1)^{n-l} \lambda^{2(n-l)+1}}{(2(n-l)+1)!} \left( \sum_{\substack{\alpha \in \mathbb{N}_0^{2(n-l)+1} \\ \sum \alpha_i = l}} \prod_{i=1}^{2(n-l)+1} \frac{(2j-1)^2}{(2\alpha_i+1)!} \right) x^{2n+1} \quad (4.17)$$

und Substitution ( $k := n - l$ )

$$= \sum_{n=0}^{\infty} \left( \sum_{k=0}^n \frac{(-1)^k}{(2k+1)!} \left( \sum_{\substack{\alpha \in \mathbb{N}_0^{2k+1} \\ \sum \alpha_i = n-k}} \prod_{i=1}^{2k+1} \frac{\prod_{j=1}^{\alpha_i} (2j-1)^2}{(2\alpha_i+1)!} \right) \lambda^{2k+1} \right) x^{2n+1}. \quad (4.18)$$

Mit der Definition der  $\gamma_{n,k}$  wie oben folgt die Behauptung.  $\square$

**Lemma 4.7** Die Taylorreihe der Funktion  $f_\lambda : x \mapsto \sin(\lambda \arcsin(x))$  im Punkt 0 ist gegeben durch

$$f_\lambda(x) = \sum_{n=0}^{\infty} \left( \frac{1}{n!} \sum_{i=1,3,\dots}^n c_{i,0}^{(n)} \lambda^i \right) x^{2n+1},$$

wobei die Koeffizienten  $c_{i,0}^{(n)}$  durch folgende Vorschriften rekursiv bzw. induktiv definiert sind:

$$\begin{aligned} s_{0,0}^{(0)} &:= 1 \\ s_{i,j}^{(n)} &:= -c_{i-1,j}^{(n-1)} + (j+1)s_{i,j+1}^{(n-1)} + (n-1+j-1)s_{i,j-1}^{(n-1)} && \text{für } 1 \leq i < n, 1 \leq j < n \\ c_{i,j}^{(n)} &:= s_{i-1,j}^{(n-1)} + (j+1)c_{i,j+1}^{(n-1)} + (n-1+j-1)c_{i,j-1}^{(n-1)} && \text{für } 1 \leq i < n, 1 \leq j < n \\ c_{i,0}^{(n)} &:= s_{i-1,0}^{(n-1)} + c_{i,1}^{(n-1)} && \text{für } 1 \leq i < n \\ s_{i,0}^{(n)} &:= -c_{i-1,0}^{(n-1)} + s_{i,1}^{(n-1)} && \text{für } 1 \leq i < n \\ c_{i,j}^{(n)} &:= s_{i,j}^{(n)} := 0 && \text{sonst.} \end{aligned}$$

*Beweis:* Aus der Definition der Taylorreihe,

$$f_\lambda(x) = \sum_{n=0}^{\infty} \frac{f_\lambda^{(n)}(0)}{n!} x^n, \quad (4.19)$$

und der Berechnung der Ableitungen von  $f_\lambda$  ergibt sich per Induktion

$$f_\lambda^{(n)}(x) = \sum_{j=0}^{n-1} \sum_{i=1,3,\dots}^{n-j} c_{i,j}^{(n)} \frac{\cos(\lambda \arcsin(x)) \lambda^i x^j}{(1-x^2)^{(n+j)/2}} + \sum_{j=0}^{n-2} \sum_{i=2,4,\dots}^{n-j} s_{i,j}^{(n)} \frac{\sin(\lambda \arcsin(x)) \lambda^i x^j}{(1-x^2)^{(n+j)/2}}, \quad (4.20)$$

wobei die Koeffizienten  $c_{i,j}^{(n)}$  und  $s_{i,j}^{(n)}$  wie oben definiert sind.

#### 4. Die trigonometrische Approximation

---

Da wir  $f_\lambda$  um 0 entwickeln, sind nur die Koeffizienten  $c_{i,0}^{(n)}$  relevant: Die Brüche in (4.20) verschwinden alle für  $x = 0$  außer für  $j = 0$ . Also gilt

$$f_\lambda^{(n)}(0) = \sum_{j=0}^{n-1} \sum_{i=1,3,\dots}^{n-j} c_{i,j}^{(n)} \frac{\cos(\lambda \arcsin(0)) \lambda^i 0^j}{(1-0^2)^{(n+j)/2}} = \sum_{i=1,3,\dots}^n c_{i,0}^{(n)} \lambda^i. \quad (4.21)$$

□

Auf Grund der Eindeutigkeit der Koeffizienten der Taylorreihe und von Polynomen folgt sofort folgendes Lemma:

**Lemma 4.8** Für  $n, k \in \mathbb{N}$  gilt

$$\frac{1}{n!} c_{2k+1,0}^{(n)} = \gamma_{n,k}.$$

Sofern also die Polynome  $P_n(\lambda)$ ,

$$P_n(\lambda) := \sum_{k=0}^n \gamma_{n,k} \lambda^{2k+1} = \frac{1}{n!} \sum_{i=1,3,\dots}^n c_{i,0}^{(n)} \lambda^i, \quad (4.22)$$

im Intervall  $[-1, 1]$  nicht-negativ sind, ist (4.11) gezeigt. Numerische Berechnungen der Koeffizienten  $c_{k,0}^{(n)}$  bzw.  $\gamma_{n,k}$  zeigen, dass für  $n \leq 100$  die Polynome  $P_n(\lambda)$  mit Polynomen  $\tilde{P}_n(\lambda)$  übereinstimmen, die wie folgt definiert sind:

$$\tilde{P}_n(\lambda) := \sum_{k=0}^n p_{n,k} \lambda^{2k+1} \quad (4.23)$$

mit den Koeffizienten

$$p_{n,k} := \frac{(-1)^k}{(2n+1)!} \sum_{\substack{\alpha \subseteq \{1,3,\dots,2n-1\} \\ |\alpha|=n-k}} \prod_{\nu \in \alpha} \nu^2. \quad (4.24)$$

Es gilt also  $p_{n,k} = \gamma_{n,k}$  für alle  $n \leq 100$  und alle  $k$ . Dabei hat  $\tilde{P}_n(\lambda)$  die Gestalt  $\tilde{P}_n(\lambda) = \frac{(-1)^n}{(2n+1)!} \lambda(\lambda^2-1^2)(\lambda^2-3^2) \cdots (\lambda^2-(2n-1)^2)$ , also Nullstellen bei  $x = 0, \pm 1, \pm 3, \dots, \pm(n-2)$ , wie im Beweis von Lemma 4.10 gezeigt werden wird.

Offensichtlich gilt für alle  $n \in \mathbb{N}_0$  für  $k = 0$  und  $k \geq n$  die Gleichung

$$p_{n,k} = \gamma_{n,k}. \quad (4.25)$$

Für  $k = n - 1$ ,  $n \in \mathbb{N}_0$  ist (4.25) leicht zu zeigen:

$$\gamma_{n,n-1} = \frac{(-1)^{n-1}}{(2n-1)!} \sum_{\nu=1}^{2n-1} \frac{1^2}{3!} = \frac{(-1)^{n-1}}{(2n-1)!} \frac{2n-1}{3 \cdot 2} \quad (4.26)$$

$$= \frac{(-1)^{n-1}}{(2n+1)!} \frac{2n(2n+1)(2n-1)}{3 \cdot 2} \quad (4.27)$$

$$= \frac{(-1)^{n-1}}{(2n+1)!} \frac{n(4n^2-1)}{3} = \frac{(-1)^{n-1}}{(2n+1)!} \sum_{\nu=1,3,5,\dots}^{2n-1} \nu^2 = p_{n,n-1} \quad (4.28)$$

Außerdem gilt:

$$\gamma_{n,k} = \frac{(-1)^k}{(2k+1)!} \left( \sum_{\substack{\alpha \in \mathbb{N}_0^{2k+1} \\ \sum \alpha_i = n-k}} \prod_{i=1}^{2k+1} p_{\alpha_i,0} \right) \quad (4.29)$$

Für die Koeffizienten  $p_{n,k}$  gilt folgende (einfach zu zeigende) Rekursionsgleichung:

$$(2n+1)(2n)p_{n,k} = \begin{cases} (2n-1)^2 p_{n-1,k} & \text{für } k=0, \\ -p_{n-1,k-1} + (2n-1)^2 p_{n-1,k} & \text{für } 0 < k < n, \\ -p_{n-1,k-1} & \text{für } k=n. \end{cases} \quad (4.30)$$

Für  $k = n - 1$  lässt sich auch das  $\gamma$ -Pendant der Rekursionsgleichung (4.30) noch recht einfach zeigen: Ausgehend von der zu zeigenden Gleichung in der kompakten Form (4.29) erhält man

$$\frac{(-1)^{n-1}(2n-1)}{(2n-1)!} \left( (2n+1)(2n)p_{1,0} = (2n-2)(2n-3)p_{1,0} + (2n-1)p_{0,0} \right). \quad (4.31)$$

Nach der Substitution  $3 \cdot 2 p_{1,0} = p_{0,0}$  (wegen (4.30)) erhält man eine Tautologie für beliebige  $p_{1,0} \in \mathbb{R}$  (und  $n \in \mathbb{N}$ ).

Im Fall  $k = n - 2$  erhält man nach entsprechenden Umformungen und Substitution leider keine Tautologie für beliebige  $p_{1,0} \in \mathbb{R}$ . Man muss vielmehr  $p_{1,0} = 1/6$  einsetzen, d. h. die genaue Struktur der Rekursionsgleichungen ausnutzen, um eine Tautologie (für  $n \in \mathbb{N}$ ) zu erhalten.

Auf Grund obiger Erkenntnisse und numerischer Berechnungen von  $c_{2k+1,0}^{(n)}$  und  $p_{n,k}$  für alle  $1 \leq n \leq 100$  und alle  $k$  (siehe auch Tabelle 4.1) stellen wir folgende Vermutung auf:

**Vermutung 4.9** Für die Koeffizienten aus (4.12) und (4.24) gilt

$$p_{n,k} = \gamma_{n,k} \quad \forall n \in \mathbb{N}_0, k \in \mathbb{N}_0, k \leq n. \quad (4.32)$$

#### 4. Die trigonometrische Approximation

---

$n$	$i$	$j=0$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$	$j=8$
1	0	0								
	1	1								
3	0	0	0	0						
	1	1	0	3						
	2	0	-3	0						
	3	-1	0	0						
5	0	0	0	0	0	0				
	1	9	0	90	0	105				
	2	0	-55	0	-105	0				
	3	-10	0	-45	0	0				
	4	0	10	0	0	0				
	5	1	0	0	0	0				
7	0	0	0	0	0	0	0	0		
	1	225	0	4725	0	14175	0	10395		
	2	0	-2079	0	-10710	0	-10395	0		
	3	-259	0	-3360	0	-4725	0	0		
	4	0	525	0	1260	0	0	0		
	5	35	0	210	0	0	0	0		
	6	0	-21	0	0	0	0	0		
	7	-1	0	0	0	0	0	0		
9	0	0	0	0	0	0	0	0	0	0
	1	11025	0	396900	0	2182950	0	3783780	0	2027025
	2	0	-136431	0	-1327095	0	-3108105	0	-2027025	0
	3	-12916	0	-328545	0	-1143450	0	-945945	0	0
	4	0	39564	0	242550	0	270270	0	0	0
	5	1974	0	31500	0	51975	0	0	0	0
	6	0	-2394	0	-6930	0	0	0	0	0
	7	-84	0	-630	0	0	0	0	0	0
	8	0	36	0	0	0	0	0	0	0
	9	1	0	0	0	0	0	0	0	0

Tabelle 4.1.: Koeffizienten  $c_{i,j}^{(n)}$  ( $j$  gerade) bzw.  $s_{i,j}^{(n)}$  ( $j$  ungerade) mit  $c_{2k+1,0}^{(n)} = p_{n,k}$  für  $n = 1, 3, 5, 7, 9$

Ausgehend von dieser Vermutung, kann die Sternförmigkeit gezeigt werden:

**Lemma 4.10** Für das Polynom  $\tilde{P}_n(\lambda)$  aus (4.23) gilt

$$\tilde{P}_n(\lambda) = \sum_{k=0}^n p_{n,k} \lambda^{2k+1} \geq 0 \quad \text{für alle } \lambda \in [0, 1]. \quad (4.33)$$

*Beweis:* Das Polynom  $P_n(\lambda) = \sum_{k=0}^n p_{n,k} \lambda^{2k+1}$  lässt sich darstellen als

$$P_n(\lambda) = \frac{(-1)^k}{(2n+1)!} \sum_{k=0}^n \left( \sum_{\substack{\alpha \subseteq \{1,3,\dots,2n-1\} \\ |\alpha|=n-k}} \prod_{\nu \in \alpha} \nu^2 \right) \lambda^{2k+1} \quad (4.34)$$

$$= \frac{(-1)^n}{(2n+1)!} \sum_{k=0}^n \left( \sum_{\substack{\alpha \subseteq \{1,3,\dots,2n-1\} \\ |\alpha|=n-k}} \prod_{\nu \in \alpha} -\nu^2 \right) \lambda^{2k+1} \quad (4.35)$$

$$= \frac{(-1)^n}{(2n+1)!} \lambda \prod_{k=1}^n (\lambda^2 - (2k-1)^2) \quad (4.36)$$

$$= \frac{(-1)^n}{(2n+1)!} \lambda (\lambda^2 - 1^2)(\lambda^2 - 3^2) \dots (\lambda^2 - (2n-1)^2). \quad (4.37)$$

Es besitzt also genau die Nullstellen  $0, \pm 1, \pm 3, \pm 5, \dots, \pm(2n-1)$  und ist somit wegen

$$\operatorname{sgn} \left( P_n \left( \frac{1}{2} \right) \right) = (-1)^n \prod_{k=1}^n \operatorname{sgn} \left( \left( \frac{1}{2} \right)^2 - (2k-1)^2 \right) = (-1)^n (-1)^n = 1 \quad (4.38)$$

im Intervall  $\lambda \in [0, 1]$  nichtnegativ. □

**Satz 4.11** Sofern die Gleichung (4.32) erfüllt ist, gilt für alle  $X \in \mathcal{TA}$  und  $\lambda \in (0, 1)$ :

$$\sin\left[\frac{\pi}{2}\lambda X\right] \succeq 0$$

Auf Grund der Überlegungen zu Beginn dieses Abschnitts folgt sofort der nächste Satz:

**Satz 4.12** Sofern die Gleichung (4.32) erfüllt ist, gilt:

Die trigonometrische Approximation  $\mathcal{TA}$  des Max-Cut-Polytops ist sternförmig mit Zentrum  $I$ .

## 4.4. Regularität

Für Techniken der lokalen Minimierung ist die Frage der Regularität (im Sinne von Robinson, siehe zum Beispiel [16]) eines Problems von zentraler Bedeutung. So ist die Kenntnis einer Beschreibung von  $\mathcal{TA}$ , die zumindest in den Ecken von  $\mathcal{MC}$  (die alle auch in  $\partial\mathcal{TA}$  liegen) regulär ist, hinreichende Voraussetzung, um lokale Optimalitätsbedingungen des Problems (4.4) für eine Ecke von  $\mathcal{MC}$  anzugeben. Von daher ist es von Bedeutung festzustellen, ob eine solche Beschreibung existiert.

Wir betrachten dazu eine beliebige Menge  $\mathcal{M} \subset \mathbb{R}^n$  die durch  $m$  Ungleichungen  $f_i(x) \leq 0$  ( $1 \leq i \leq m$ ) gegeben ist. Es gilt folgendes Lemma:

**Lemma 4.13** *Sei  $\bar{x} \in \partial\mathcal{M}$ . Falls der Tangentialkegel  $T_{\mathcal{M}}(\bar{x})$  von  $\mathcal{M}$  in  $\bar{x}$  leeres Inneres besitzt,  $T_{\mathcal{M}}(\bar{x})^\circ = \emptyset$ , so ist  $\mathcal{M}$  in  $\bar{x}$  nicht regulär.*

*Beweis:* Wir lassen die inaktiven Ungleichungen fort und nehmen o. B. d. A. an, dass  $f_i(\bar{x}) = 0$  für  $1 \leq i \leq m$ . Falls  $\mathcal{M}$  in  $\bar{x}$  regulär ist, so folgt

$$0 \in (\text{Bild}(DF(\bar{x})) + \mathbb{R}_+^m)^\circ \quad \text{mit} \quad F(x) := (f_1(x), \dots, f_m(x))^T.$$

Somit existiert ein  $z \in \text{Bild}(DF(\bar{x}))$  mit  $z < 0$ . (Sonst ließe sich  $\mathbb{R}_+^m$  von  $\text{Bild}(DF(\bar{x}))$  trennen, d. h. es existiert ein  $c \neq 0$  mit  $c^T x \geq 0$  für alle  $x \in \text{Bild}(DF(\bar{x}))$  und  $c^T x \leq 0$  für alle  $x \leq 0$ . Also ist  $c \geq 0$  und somit auch  $c^T x \geq 0$  für alle  $x \in \text{Bild}(DF(\bar{x})) + \mathbb{R}_+^m$  im Widerspruch dazu, dass 0 innerer Punkt dieser Menge ist.)

Also existiert ein  $w \in \mathbb{R}^n$ , mit  $DF(\bar{x})w < 0$ . Dieser Vektor  $w$  liegt aber im Inneren von  $T_{\mathcal{M}}(\bar{x})$ . □

Falls also der Tangentialkegel von  $\mathcal{TA}$  in einer Ecke  $\bar{X}$  ein leeres Inneres hätte, so ließe sich keine in  $\bar{X}$  reguläre Beschreibung von  $\mathcal{TA}$  mit Ungleichungen der Form  $f_i(X) \leq 0$  angeben.

Wir betrachten zunächst die Ecken von  $\mathcal{TA}$  und prüfen anschließend noch den Tangentialkegel bestimmter Kanten.

### 4.4.1. Ecken von $\mathcal{TA}$

Um einen ersten Eindruck von der Regularität von  $\mathcal{TA}$  zu gewinnen, wurde die Schnittmenge von  $\mathcal{TA}$  und verschiedenen ausgewählten 2-dimensionalen Mengen betrachtet. Dabei kann man leicht den Eindruck gewinnen, dass  $\mathcal{TA}$  in den Ecken sehr spitz ist und keine reguläre Beschreibung existieren kann (siehe Abbildung 4.1).

Bei näherer Betrachtung der Ecken stellt sich jedoch heraus, dass der Tangentialkegel ein nicht-leeres Inneres besitzt, sofern (4.32) erfüllt ist. Dabei spielt Satz 4.11 eine entscheidende Rolle.

Für die Betrachtung der Ecken von  $\mathcal{TA}$  benötigen wir eine Abschätzung, die wir hier als Lemma formulieren.

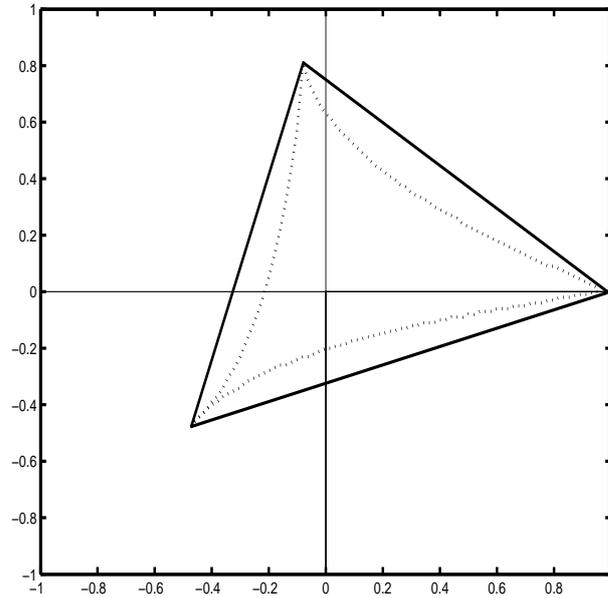


Abbildung 4.1.: Schnitt durch die Menge  $\mathcal{TA}$  (gepunktet). Als durchgezogene Linie ist das Max-Cut-Polytop  $\mathcal{MC}$  zu sehen.

**Lemma 4.14** Für  $\varepsilon \in [0, 1]$  gilt folgende Abschätzung:

$$\sin\left(\frac{\pi}{2}\right) = \sin\left(\frac{\pi}{2}(1 - \varepsilon)\right) + \frac{\pi^2}{4}(1 - e_0)\varepsilon^2 - e_1\varepsilon^3$$

mit  $0 \leq e_0 \leq \frac{1}{2}$  und  $0 \leq e_1 \leq \frac{\pi^4}{32}$ .

*Beweis:* Wir entwickeln die Funktion

$$\sin\left(\frac{\pi}{2}(1 - \varepsilon) + \frac{\pi}{2}\varepsilon\right) \quad (4.39)$$

als Taylorreihe um  $\frac{\pi}{2}(1 - \varepsilon)$ . Dann ist (4.39) für ein  $0 < \xi < \varepsilon$  gleich

$$\sin\left(\frac{\pi}{2}(1 - \varepsilon)\right) + \varepsilon \cos\left(\frac{\pi}{2}(1 - \varepsilon)\right) \frac{\pi}{2} - \frac{\varepsilon^2}{2} \sin\left(\frac{\pi}{2}(1 - \varepsilon) + \frac{\pi}{2}\xi\right) \frac{\pi^2}{4}.$$

Also gilt ( $0 \leq \sin(x) \leq 1$  für  $x \in [0, \frac{\pi}{2}]$ )

$$\sin\left(\frac{\pi}{2}(1 - \varepsilon) + \frac{\pi}{2}\varepsilon\right) = \sin\left(\frac{\pi}{2}(1 - \varepsilon)\right) + \varepsilon \cos\left(\frac{\pi}{2}(1 - \varepsilon)\right) \frac{\pi}{2} - e_0 \frac{\pi^2}{4} \varepsilon^2 \quad (4.40)$$

#### 4. Die trigonometrische Approximation

---

für ein  $e_0 \in \mathbb{R}$  mit  $0 \leq e_0 \leq \frac{1}{2}$ .

Mit der Taylorentwicklung von  $\cos\left(\frac{\pi}{2}(1-\varepsilon)\right) = \sin\left(\frac{\pi}{2}\varepsilon\right)$  um  $\varepsilon = 0$  erhält man  $\sin\left(\frac{\pi}{2}\varepsilon\right) = \frac{\pi}{2}\varepsilon - \widehat{e}_1\varepsilon^2$  für ein  $\widehat{e}_1 \in \mathbb{R}$  mit  $0 \leq \widehat{e}_1 \leq \frac{\pi^2}{8}$ . Eingesetzt in (4.40) folgt die Behauptung.  $\square$

Falls die Beziehung (4.32) erfüllt ist, lässt sich zeigen, dass in jeder Ecke von  $\mathcal{TA}$  das Innere des Tangentialkegels nicht leer ist:

**Satz 4.15** *Sei  $X \in \mathbb{R}^{n \times n}$  eine Ecke von  $\mathcal{TA}$ . Sei  $Y \in \mathbb{R}^{n \times n}$ ,  $\text{diag}(Y) = 0$ ,  $\|Y\|_\infty = 1$  eine Richtung, wobei mit  $\|\cdot\|_\infty$  die  $\text{lub}_\infty$ -Norm bezeichnet ist. Sei (4.32) erfüllt.*

*Dann existiert ein  $\tilde{\varepsilon} > 0$ , so dass für alle  $\varepsilon \in [0, \tilde{\varepsilon}]$  und alle  $\delta$  mit  $0 \leq \delta \leq \frac{\varepsilon}{2n}$  gilt:*

$$(1 - \varepsilon)X + \varepsilon I + \delta Y \in \mathcal{TA}.$$

*Beweis:* Sei eine Ecke  $X \in \mathcal{TA} \subset \mathbb{R}^{n \times n}$  ( $X = ee^T$  o. B. d. A. wegen der Symmetrie von  $\mathcal{TA}$  und  $\mathcal{MC}$ ) und sei  $Y \in \mathbb{R}^{n \times n}$ ,  $\text{diag}(Y) = 0$ ,  $\|Y\|_\infty = 1$  gegeben. Offensichtlich ist

$$\text{diag}((1 - \varepsilon)X + \varepsilon I + \delta Y) = e. \quad (4.41)$$

Es bleibt also

$$\sin\left[\frac{\pi}{2}((1 - \varepsilon)X + \varepsilon I + \delta Y)\right] \succeq 0$$

zu zeigen. Nach Lemma 4.14 gilt wegen (4.41) für ein  $0 \leq e_0 \leq \frac{1}{2}$  und ein  $0 \leq e_1 \leq \frac{\pi^4}{32}$ :

$$\sin\left[\frac{\pi}{2}((1 - \varepsilon)X + \varepsilon I + \delta Y)\right] \quad (4.42)$$

$$= \sin\left[\frac{\pi}{2}((1 - \varepsilon)X + \delta Y)\right] + \frac{\pi^2}{4}(1 - e_0)\varepsilon^2 I - e_1\varepsilon^3 I \quad (4.43)$$

und mit Taylorentwicklung des sin-Terms um  $\frac{\pi}{2}(1 - \varepsilon)X$  für Nebendiagonal-Einträge gilt für ein  $0 < \xi < \delta < \varepsilon$

$$= \sin\left[\frac{\pi}{2}(1 - \varepsilon)X\right] + \delta \cos\left[\frac{\pi}{2}(1 - \varepsilon)X\right] \circ \frac{\pi}{2}Y \quad (4.44)$$

$$- \frac{\delta^2}{2} \sin\left[\frac{\pi}{2}(1 - \varepsilon)X + \frac{\pi}{2}\xi Y\right] \circ \frac{\pi^2}{4}Y \circ Y + \frac{\pi^2}{4}(1 - e_0)\varepsilon^2 I - e_1\varepsilon^3 I$$

$$= \sin\left[\frac{\pi}{2}(1 - \varepsilon)X\right] + \cos\left[\frac{\pi}{2}(1 - \varepsilon)X\right] \circ \frac{\pi}{2}\delta Y \quad (4.45)$$

$$- e_2 M_2 + \frac{\pi^2}{4}(1 - e_0)\varepsilon^2 I - e_1\varepsilon^3 I$$

für ein  $|e_2| = \frac{\pi^2}{8}\delta^2$  und eine Matrix  $M_2 := \sin\left[\frac{\pi}{2}(1 - \varepsilon)X + \frac{\pi}{2}\xi Y\right] \circ Y \circ Y$ . Dabei gilt  $\|M_2\|_\infty \leq \|Y \circ Y\|_\infty \leq 1$ . Taylorentwicklung des cos-Terms ergibt ( $X = ee^T$ )

$$= \sin\left[\frac{\pi}{2}(1 - \varepsilon)X\right] + \frac{\pi}{2}\varepsilon \circ \frac{\pi}{2}\delta Y + \frac{\pi^2}{4}(1 - e_0)\varepsilon^2 I + e_3 M_3 - e_2 M_2 - e_1\varepsilon^3 I \quad (4.46)$$

für ein  $|e_3| \leq \frac{\pi^3}{16}\varepsilon^2\delta$  und eine Matrix  $M_3$  mit  $\|M_3\|_\infty \leq 1$ . Der erste Summand ist nach Satz 4.11 positiv semidefinit. Für die Eigenwerte  $\mu$  der Summe der beiden nächsten Summanden,  $\frac{\pi}{2}\varepsilon \circ \frac{\pi}{2}\delta Y + \frac{\pi^2}{4}(1 - e_0)\varepsilon^2 I$ , gilt nach dem Satz von Gerschgorin wegen  $\|Y\|_\infty = 1$ :

$$\left| \mu - \frac{\pi^2}{4}(1 - e_0)\varepsilon^2 \right| \leq \frac{\pi^2}{4}\varepsilon\delta.$$

Mit  $\delta := \frac{\varepsilon}{2n}$  folgt:

$$\left| \mu - \frac{\pi^2}{4}(1 - e_0)\varepsilon^2 \right| \leq \frac{\pi^2}{8n}\varepsilon^2,$$

also

$$\mu \geq \left( \frac{\pi^2}{4}(1 - e_0) - \frac{\pi^2}{8n} \right) \varepsilon^2 \geq \left( \frac{\pi^2}{8} - \frac{\pi^2}{8n} \right) \varepsilon^2 \geq \frac{\pi^2}{16}\varepsilon^2.$$

Berücksichtigt man die drei verbleibenden Fehlerterme mit Hilfe des Satzes von Bauer-Fike (siehe nachfolgendes Lemma 4.16), so ergibt sich für die Eigenwerte  $\tilde{\mu}$  von (4.46) die Abschätzung ( $\|U\|_\infty \leq n$  für unitäre Matrizen  $U \in \mathbb{C}^{n \times n}$ )

$$\tilde{\mu} \geq \frac{\pi^2}{16}\varepsilon^2 - n\varepsilon^2 \left( \frac{\pi^2}{32n^2} + \frac{\pi^3}{32n}\varepsilon + \frac{\pi^4}{32}\varepsilon \right) = \frac{\pi^2}{16}\varepsilon^2 - \varepsilon^2 \left( \frac{\pi^2}{32n} + \frac{\pi^3}{32}\varepsilon + \frac{\pi^4}{32}n\varepsilon \right)$$

bzw. für  $n \geq 2$

$$\begin{aligned} \tilde{\mu} &\geq \left( \frac{\pi^2}{8} - \frac{\pi^2}{8n} - \frac{\pi^2}{32n} - \frac{\pi^3}{32}\varepsilon - \frac{\pi^4}{32}n\varepsilon \right) \varepsilon^2 \geq \left( \frac{\pi^2}{8} - \frac{\pi^2}{16} - \frac{\pi^2}{64} - \frac{\pi^3}{32}\varepsilon - \frac{\pi^4}{32}n\varepsilon \right) \varepsilon^2 \\ &\geq (4 - 2 - 1 - \pi\varepsilon - \pi^2n\varepsilon) \varepsilon^2 \frac{\pi^2}{32} = (1 - \pi\varepsilon - \pi^2n\varepsilon) \varepsilon^2 \frac{\pi^2}{32}. \end{aligned}$$

Es existiert also ein  $\tilde{\varepsilon} > 0$ , so dass für alle  $\varepsilon \in [0, \tilde{\varepsilon}]$  gilt:

$$\tilde{\mu} \geq 0.$$

□

**Lemma 4.16 (Bauer-Fike)** *Sei  $A \in \mathbb{C}^{n \times n}$  eine hermitesche Matrix,  $E \in \mathbb{C}^{n \times n}$  eine Fehlermatrix. Wenn  $\mu$  ein Eigenwert von  $A + E \in \mathbb{C}^{n \times n}$  ist, dann existiert eine unitäre Matrix  $U \in \mathbb{C}^{n \times n}$  mit  $U^H A U = \text{Diag}(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n \times n}$  und es gilt für jede  $p$ -Norm:*

$$\min_{\lambda \in \lambda(A)} |\lambda - \mu| \leq \|U\|_p^2 \|E\|_p.$$

*Beweis:* Die Existenz von  $U$  gilt nach dem Satz über die Hauptachsentransformation (siehe zum Beispiel [20, S. 256]). Die Behauptung folgt dann aus dem Satz von Bauer-Fike (vgl. zum Beispiel [10, S. 342]). □

**Korollar 4.17** *Sofern (4.32) erfüllt ist, gilt:*

*In jeder Ecke  $X$  von  $\mathcal{TA} \subset \mathbb{R}^{n \times n}$  hat der Tangentialkegel  $T(\mathcal{TA}, X)$  einen Öffnungswinkel größer oder gleich  $2 \arctan\left(\frac{1}{2n}\right)$ .*

#### 4.4.2. Kanten von $\mathcal{TA}$

Im Gegensatz zu den Ecken kann das Innere des Tangentialkegels an den Kanten leer sein. Wir zeigen für  $n \geq 8$ , dass der Tangentialkegel von  $\mathcal{TA}$  im Mittelpunkt einer Kante (und möglicherweise in allen übrigen relativ inneren Punkten der Kante) ein leeres Inneres besitzen kann. Dies legt nahe, dass  $\mathcal{TA}$  für  $n \geq 8$  auch in den Ecken keine reguläre Darstellung besitzt.

Wir betrachten die vier Vektoren im  $\mathbb{R}^8$ :

$$x^{(1)} = \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \\ 1 \\ -1 \\ 1 \\ -1 \end{bmatrix}, \quad x^{(2)} = \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \\ 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}, \quad x^{(3)} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix}, \quad x^{(4)} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}.$$

Diese Vektoren stehen paarweise aufeinander senkrecht. Ferner sei

$$\bar{X} := \frac{1}{2} \left( x^{(1)} \left( x^{(1)} \right)^T + x^{(2)} \left( x^{(2)} \right)^T \right)$$

der Mittelpunkt einer Kante von  $\mathcal{TA}$  und

$$\bar{Z} := \frac{1}{2} \left( x^{(4)} \left( x^{(4)} \right)^T - x^{(3)} \left( x^{(3)} \right)^T \right)$$

eine Richtung. Nach Konstruktion ist  $\bar{X}\bar{Z} = 0$  und  $\bar{X}$  und  $\bar{Z}$  besitzen eine gemeinsame Basis aus orthonormalen Eigenvektoren. Um den Tangentialkegel von  $\mathcal{TA}$  in  $\bar{X}$  abzuschätzen, betrachten wir für kleine  $\varepsilon > 0$  und kleine  $\rho > 0$  die Punkte

$$X_{\pm}(\varepsilon, \rho) := (1 - \varepsilon)\bar{X} + \varepsilon I \pm \rho \bar{Z}$$

Es stellt sich heraus, dass für  $\rho = \varepsilon^2$  die Punkte  $X_{\pm}(\varepsilon, \rho)$  nicht in  $\mathcal{TA}$  liegen, und der Tangentialkegel von  $\mathcal{TA}$  in  $\bar{X}$  somit ein leeres Inneres besitzt. Wir betrachten im Folgenden die Matrix  $X(\varepsilon, \rho) := X_+(\varepsilon, \rho)$ ; die Matrix  $X_-(\varepsilon, \rho)$  lässt sich ganz analog analysieren. Es genügt zu zeigen, dass die Matrix

$$Z(\varepsilon, \rho) := \sin \left[ \frac{\pi}{2} X(\varepsilon, \rho) \right]$$

für  $\rho \geq \varepsilon^2$  nicht positiv semidefinit ist. Die Matrix  $Z(\varepsilon, \rho)$  lässt sich explizit angeben:

$$Z(\varepsilon, \rho) = \begin{bmatrix} 1 & 0 & 0 & -1 + \varepsilon_2 & 1 - \varepsilon_2 & \rho & \rho & -1 + \varepsilon_2 \\ 0 & 1 & -1 + \varepsilon_2 & 0 & \rho & 1 - \varepsilon_2 & -1 + \varepsilon_2 & \rho \\ 0 & -1 + \varepsilon_2 & 1 & 0 & \rho & -1 + \varepsilon_2 & 1 - \varepsilon_2 & \rho \\ -1 + \varepsilon_2 & 0 & 0 & 1 & -1 + \varepsilon_2 & \rho & \rho & 1 - \varepsilon_2 \\ 1 - \varepsilon_2 & \rho & \rho & -1 + \varepsilon_2 & 1 & 0 & 0 & -1 + \varepsilon_2 \\ \rho & 1 - \varepsilon_2 & -1 + \varepsilon_2 & \rho & 0 & 1 & -1 + \varepsilon_2 & 0 \\ \rho & -1 + \varepsilon_2 & 1 - \varepsilon_2 & \rho & 0 & -1 + \varepsilon_2 & 1 & 0 \\ -1 + \varepsilon_2 & \rho & \rho & 1 - \varepsilon_2 & -1 + \varepsilon_2 & 0 & 0 & 1 \end{bmatrix} + O(\varepsilon^3)$$

wobei  $\epsilon_2 := \frac{\pi^2 \epsilon^2}{8}$ . Für kleine  $\epsilon > 0$  ist daher  $(x^{(3)})^T Z(\epsilon, \rho) x^{(3)} < 0$ , was zu zeigen war.

Man erkennt auch, dass der Beweis sich in etwas komplizierterer Form wohl auf andere Punkte im Inneren der Kante von  $x^{(1)} (x^{(1)})^T$  nach  $x^{(2)} (x^{(2)})^T$  übertragen lässt; wir beschränken uns hier jedoch auf die Betrachtung des Mittelpunktes dieser Kante.

## 4.5. Lokal optimale Ecke auch global optimal

Aus den Überlegungen im Unterabschnitt 4.2.2 geht folgender Satz hervor.

**Satz 4.18** *Eine Ecke  $X = xx^T \in \mathcal{TA}$  ist eine lokal optimale Lösung von (4.4) genau dann, wenn  $x$  das Max-Cut-Problem (1.4) löst.*

*Beweis:* Nach Satz 4.5 sind alle Ecken von  $\mathcal{MC}$  in  $\mathcal{TA}$  enthalten. Insbesondere gilt: Falls  $x^*$  bzw.  $x^*(x^*)^T$  eine Optimallösung von (1.4) bzw. (1.5) ist, dann liegt die Kante  $\{\lambda xx^T + (1 - \lambda)x^*(x^*)^T \mid 0 \leq \lambda \leq 1\}$  in  $\mathcal{TA}$ , d. h.  $xx^T$  ist nur eine lokale Optimallösung von (4.4), falls  $C \bullet (xx^T) = C \bullet (x^*(x^*)^T)$ .  $\square$

Aus Satz 4.18 folgt, dass die Prüfung, ob ein gegebener Punkt  $X$  lokal optimal für (4.4) ist, gleichbedeutend mit der Frage ist, ob  $Z_{MC}^* > \frac{1}{4}L \bullet X$  gilt. Letztere Frage ist aber NP-vollständig.

Im Problem (4.4) ist die Zielfunktion linear und die zulässige Menge  $\mathcal{TA}$  ist durch eine nichtlineare semidefinite Funktion beschrieben. Die Prüfung der notwendigen und hinreichenden Optimalitätsbedingungen ist also „sehr schwer“, zumindest für die Ecken von  $\mathcal{TA}$ . Dies ist insofern bemerkenswert, als die Menge  $\mathcal{TA}$  ja zunächst durchaus angenehme Eigenschaften hat. So ist, wie im vorherigen Abschnitt gezeigt, der Tangentialkegel der Ecke unter der Voraussetzung (4.32) nicht leer.

Wenn die Menge  $\mathcal{TA}$  regulär im üblichen Sinne wäre, dann wäre die Prüfung der notwendigen und hinreichenden Optimalitätsbedingungen für ein lokales Minimum in polynomieller Zeit möglich. Würde man also eine reguläre Beschreibung der Menge  $\mathcal{TA}$  finden, so wäre  $P=NP$ . Im Abschnitt 4.4 gehen wir dieser Fragestellung nach.

Der Satz 4.18 wirft aber noch eine weitere Frage auf: Hat das Problem (4.4) weitere lokale Minima außer den Optimallösungen des Max-Cut-Problems (1.4)?

Präzise formuliert: Sei  $\bar{X}$  eine lokal optimale Ecke von (4.4), d. h. Optimallösung von (1.4). Sei  $X \in \mathcal{TA}$  ein beliebiger Punkt aus der trigonometrischen Approximation. Gilt dann die Implikation

$$X \text{ ist lokal optimale Lösung von (4.4)} \Rightarrow C \bullet X = C \bullet \bar{X} ? \quad (4.47)$$

Nachfolgend wollen wir jedes  $X \in \mathcal{TA}$ , das lokal optimale Lösung von (4.4) ist, aber nicht  $C \bullet X = C \bullet \bar{X}$  erfüllt, als **rein lokal optimale Lösung** bzw. **rein lokales Minimum** bezeichnen.

#### 4. Die trigonometrische Approximation

---

Auf Grund der linearen Zielfunktion in (4.4) kann es (rein) lokale Minima nur am Rand der Menge  $\mathcal{TA}$  geben. Da es in den Ecken von  $\mathcal{TA}$  keine reinen lokalen Minima geben kann, können auf den die Ecken verbindenden affinen Randmannigfaltigkeiten ebenfalls keine reinen lokalen Minima existieren. Insbesondere kann es wegen Satz 4.5 auf allen 1- und 2-dimensionalen Randflächen kein reines lokales Minimum geben.

Auf Grund der Kontraktionseigenschaft der Funktion  $\arcsin(\frac{\pi}{2}x)$  (wie im Unterabschnitt 4.2.2 beschrieben) und umfangreicher numerischer Ergebnisse (vgl. Kapitel 6) ist zu vermuten, dass auch auf den höherdimensionalen Randflächen keine rein lokal optimalen Lösungen existieren.

**Vermutung 4.19** *Ein Punkt  $X \in \mathcal{TA}$  ist genau dann lokales Minimum, wenn  $X$  globales Minimum von  $\mathcal{TA}$  ist.*

# 5. Lösung der trigonometrischen Approximation

## 5.1. Grundidee

Ausgehend von Resultaten aus Kapitel 4 lässt sich das Max-Cut-Problem nicht mehr nur mit Methoden der globalen Optimierung lösen. Es genügt vielmehr, für das trigonometrische semidefinite Programm (4.4), also

$$\begin{aligned} & \max \left\{ \frac{1}{4} L \bullet X \mid X \in \mathcal{TA} \right\} \\ = & \max \left\{ \frac{1}{4} L \bullet X \mid \sin\left[\frac{\pi}{2} X\right] \succeq 0, X \succeq 0, \text{diag}(X) = e \right\}, \end{aligned}$$

eine lokal maximale Ecke zu finden.

Der Vorteil dieser Formulierung gegenüber der Maximierung von  $L \bullet X$  über  $\mathcal{MC}$  ist, dass sich die Menge  $\mathcal{TA}$  durch zwei glatte Semidefinitheitsbedingungen beschreiben lässt. Hingegen ist für  $\mathcal{MC}$  keine Beschreibung von polynomiell vielen Nebenbedingungen bekannt. Wir versuchen im Folgenden, eine lokale Optimallösung von (4.4) zu berechnen. Falls die Vermutung 4.19 korrekt ist, ist diese stets eine Ecke.

Allerdings ist (4.4) nicht regulär und auch die äquivalente Formulierung (4.3) ist auf Grund der Nicht-Differenzierbarkeit in den Ecken kaum mit den konventionellen Verfahren der lokalen Optimierung zu lösen.

Wir beschreiben daher im nächsten Abschnitt ein Verfahren, dass ausgehend vom Startpunkt  $I$  mit stets zulässigen Iterierten eine Näherungslösung von (4.3) findet. Diese Näherungslösung muss noch zu einer Ecke gerundet werden. Hierfür diskutieren wir im Abschnitt 5.3 verschiedene Ansätze, die dann im Kapitel 6 numerisch getestet werden.

## 5.2. Ein Innere-Punkte-Verfahren

Da die Menge  $\mathcal{TA}$  nicht regulär ist, verwenden wir statt (4.4) die Formulierung (4.3) über der Menge  $\mathcal{SR}$ , also

$$\max \left\{ \frac{1}{4} L \bullet \frac{2}{\pi} \arcsin[Z] \mid Z \in \mathcal{SR} \right\}$$

$$= \max \left\{ \frac{1}{4}L \bullet \frac{2}{\pi} \arcsin[Z] \mid Z \succeq 0, \text{diag}(Z) = 1 \right\}. \quad (5.1)$$

Im Inneren des zulässigen Bereiches, d. h. für  $Z \succ 0$ , ist die Zielfunktion  $\frac{1}{4}L \bullet \frac{2}{\pi} \arcsin[Z]$  eine differenzierbare Funktion der Variablen  $Z_{ij}$  mit  $i \neq j$ . (Die Diagonalelemente  $Z_{ii} = 1$  werden hier nicht als Variablen behandelt.)

### Das Verfahren

Linearisiert man in (5.1) die Zielfunktion, so erhält man für einen festen Punkt  $\bar{X}$  aus dem Inneren von  $\mathcal{SR}$  ein lineares semidefinites Programm wie in (2.1):

$$\begin{aligned} & \max C(\bar{X}) \bullet X \\ & \text{mit: } \text{diag}(X) = 0 \\ & X \succeq 0, \end{aligned} \quad (5.2)$$

wobei die Einträge  $(C(\bar{X}))_{ij}$  der Matrix  $C(\bar{X})$  gegeben sind durch  $(L = (l_{ij}))$

$$(C(\bar{X}))_{ij} := \begin{cases} \frac{1}{2\pi} l_{ij} \frac{1}{\sqrt{1-(\bar{X}_{ij})^2}} & \text{für } i \neq j, \\ 0 & \text{sonst.} \end{cases} \quad (5.3)$$

Wir bilden daher in jeder Iteration unseres Innere-Punkte-Verfahrens das Barriereproblem ausgehend von (5.2). Für das Barriereproblem wird dann jeweils ein Newtonschritt mit Liniensuche durchgeführt.

Unser Algorithmus lautet somit:

**Algorithmus 5.1**  $X_0 := I$  der Startpunkt,  $C_0 := Df(X_0)$  die Ableitung im Startpunkt. Sei  $\Phi : \mathcal{SR}^\circ \rightarrow \mathbb{R}$  eine  $\theta$ -selbstkonkordante Barrierefunktion ( $\theta = n$ ) und  $\mu_0 > 0$ .

Wiederhole für  $k = 0, 1, 2 \dots$

1. Führe einen Newtonschritt aus, d. h.

$$X_{k+1} := \arg \min_X \left( \frac{C_k^T X}{\mu_k} + \Phi(X) \right).$$

2. Aktualisiere die Ableitung, d. h.

$$C_{k+1} := C(X_{k+1})$$

mit  $C(\cdot)$  wie in (5.3).

3. Verkleinere  $\mu_k$  auf  $\mu_{k+1}$ .

**Bemerkung:** Die Punkte  $X_k$  sind stets strikt zulässig ( $X_k \succ 0$ ), so dass die Ableitung  $Df(X_k)$  wohldefiniert ist.

### Konvergenz

Die Konvergenz des Verfahrens, insbesondere in polynomieller Zeit, ist nicht gesichert. Man beachte, dass die Bestimmung einer lokalen Optimallösung von (4.4) nach Vermutung 4.19 ein NP-vollständiges Problem ist.

Unsere praktische Implementierung konvergiert denn auch nicht immer gegen die gesuchte Optimallösung, sondern bleibt bei einem (relativ) inneren Punkt  $\tilde{X}$  stehen. Dies liegt zum einen in der begrenzten Rechengenauigkeit: Die Ableitung der Funktion  $f(X) := \frac{1}{4}L \bullet \frac{2}{\pi} \arcsin[X]$  geht für Matrix-Einträge  $|X_{ij}| \rightarrow 1$  gegen unendlich. Insbesondere ist sie für  $X_{ij} = \pm 1$  nicht definiert. Dabei besteht die Schwierigkeit in den Nebendiagonaleinträgen, die in recht unterschiedlicher Geschwindigkeit gegen den Betrag eins streben. Dadurch haben die Matrizen  $C_k$  Einträge von sehr unterschiedlicher Größenordnung und die übliche *double*-Arithmetik rechnet nicht mehr genau genug.

Darüberhinaus stellte sich heraus, dass die Funktion  $X \mapsto (C_k^T X) / \mu_k + \Phi(X)$  stationäre Punkte haben kann, die deutlich entfernt von der gesuchten Optimallösung liegen können.

Schließlich stellt sich hier, gerade im Hinblick auf die unterschiedliche Größenordnung der Einträge der Matrizen  $C_k$ , eine typische Eigenschaft der Innere-Punkte-Verfahren als problematisch heraus: Sie streben generell gegen das Zentrum der Optimalmenge, da sie einen bestimmten zentralen Pfad verfolgen. Sind zum Beispiel zwei Ecken optimal, so strebt ein Innere-Punkte-Verfahren gegen die Mitte der sie verbindenden Kante. Unterscheiden sich zwei Ecken nur sehr wenig in ihrem Zielfunktionswert, so strebt das Verfahren zunächst wieder gegen die Mitte der sie verbindenden Kante um erst für sehr kleine Barriereparameter gegen die optimale Ecke mit dem besseren Zielfunktionswert zu konvergieren.

Im Falle von (4.3) ist nun dieser Knick numerisch besonders schwierig zu verfolgen, da man sich in der Formulierung (4.4) in den sehr engen Tälern der Menge  $\mathcal{TA}$  bewegen muss.

## 5.3. Runden zu einer ganzzahligen Lösung

Wie oben beschrieben, endet das Verfahren zur Lösung von (4.3) nicht immer mit einer Lösung, die sehr nahe an der Optimallösung ist. Insbesondere können einzelne Einträge der Näherungslösung  $\tilde{X}$  für (4.3) sogar nahe bei Null liegen.

Wir diskutieren daher im Folgenden verschiedene Ansätze, wie man aus der Näherungslösung  $\tilde{X}$  für (4.3) eine gute Näherungslösung für das zugehörige Max-Cut-Problem findet. Dabei bezeichnet  $\bar{X} := \frac{2}{\pi} \arcsin[\tilde{X}] \in \mathcal{TA}$  die  $\tilde{X}$  entsprechende Näherungslösung der Formulierung (4.4). Mit  $\bar{Z}_{TA} := \frac{1}{4}L \bullet \bar{X}$  wird der Wert der Näherungslösung bezeichnet.

## Einfache untere Schranke

Da die Approximationslösung  $\bar{X}$  Element von  $\mathcal{TA}$  ist, gilt auch  $\bar{X} \in \mathcal{MC}$ . Zwar ist  $\bar{X}$  im allgemeinen kein Schnitt, jedoch ist  $\frac{1}{4}L \bullet \bar{X}$  eine untere Schranke an den Optimalwert des Max-Cut-Problems.

### 5.3.1. Kombinatorisch

Die Einträge der Matrix  $\bar{X}$  lassen sich sehr oft in zwei Mengen aufteilen: Einerseits gibt es Einträge, die extrem nahe bei  $\pm 1$  liegen. Die anderen liegen meist bei Null oder sind zumindest deutlich von  $\pm 1$  entfernt.

Da ein Eintrag  $\bar{X}_{ij}$  das Verhältnis der Knoten  $i$  und  $j$  im Graphen repräsentiert, kann man  $\bar{X}$  entsprechend interpretieren: Liegt ein Eintrag  $\bar{X}_{ij}$  nahe bei  $\pm 1$ , so ist das Verhältnis von  $i$  zu  $j$  schon sehr festgelegt. Ist hingegen  $\bar{X}_{ij}$  weiter entfernt von  $\pm 1$ , so ist das Verhältnis der Knoten  $i$  und  $j$  noch unklar. Diese Interpretation wird dadurch gestärkt, dass sich die Einträge, die nahe bei  $\pm 1$  liegen, zu Clustern in folgendem Sinne zusammenschließen: Sind für bestimmte (paarweise verschiedene)  $i, j, k \in \{1, 2, \dots, n\}$  die Einträge  $X_{ij}$  und  $X_{jk}$  nahe bei  $\pm 1$ , dann liegt wegen

$$\begin{pmatrix} 1 & x_{ij} & x_{jk} \\ x_{ij} & 1 & x_{ik} \\ x_{jk} & x_{ik} & 1 \end{pmatrix} \succeq 0$$

auch  $X_{ik}$  nahe bei  $\pm 1$ .

Dadurch wird folgender Ansatz motiviert: Man bestimmt die Cluster (etwa  $k$  Stück) und reduziert das Max-Cut-Problem auf ein Problem  $P'$  mit  $k$  Knoten. Nun gibt es verschiedene Möglichkeiten,  $P'$  zu lösen. Für kleines  $k$  kann man  $P'$  exakt lösen, d. h. man probiert alle Kombinationsmöglichkeiten der Cluster durch. Für größeres  $k$  bietet sich eine Näherungslösung basierend auf reinen Zufallsschnitten an. Alternativ kann man auch nochmal für  $P'$  die semidefinite Relaxierung oder die trigonometrische Approximation lösen.

### 5.3.2. Zufallsschnitte

Als weitere Möglichkeit, mittels  $\bar{X}$  einen guten Schnitt zu finden, bietet sich die Zufallsschnitt-Technik aus dem Verfahren von Goemans und Williamson an: Da  $\bar{X}$  positiv (semi)definit ist, kann man für  $\bar{X}$  mittels der Cholesky-Zerlegung eine Matrix  $R$  finden mit  $R^T R = \bar{X}$ . Die Spalten von  $R$  lassen sich dann wie in Kapitel 2 als Vektoren auffassen. Wir erzeugen dann  $R$ -basierte Zufallsschnitte.

Hier gilt zwar nicht mehr die starke Leistungsgarantie von  $\alpha \approx 87,856\%$  des Max-Cut-Optimalwerts  $Z_{MC}^*$ . Wenn aber (wie in numerischen Beispielen häufig beobachtet)  $\bar{Z}_{TA}$  nur wenig vom exakten Optimalwert  $Z_{TA}^*$  von (4.4) entfernt liegt, etwa  $\bar{Z}_{TA} \geq (1 - \varepsilon)Z_{TA}^*$ , dann hat die Zufallsschnitt-Technik auch eine Leistungsgarantie von  $(1 - \varepsilon)\alpha$ .

Die Anwendung der Zufallsschnitt-Technik hat aber noch eine andere Motivation: Da  $\bar{X} \in \mathcal{MC}$  eine Näherung an die Optimallösung des Max-Cut-Problems ist, besteht das Runden von  $\bar{X}$  zu einer ganzzahligen Lösung vor allem darin, eine Ecke von  $\mathcal{MC}$  mit kleinstem Abstand zu  $\bar{X}$  zu suchen. Die Ecken, die uns eine  $R$ -basierte Zufallsschnitt-Erzeugung liefert, sind aber gerade mit hoher Wahrscheinlichkeit Ecken aus der näheren Umgebung von  $\bar{X}$ .

## 6. Numerische Ergebnisse

### 6.1. Allgemeines

#### 6.1.1. Fallbeispiele

Für den numerischen Vergleich zwischen den Methoden GW und TA haben wir uns unter anderem an [9] orientiert.

#### Zufallsgraphen

Wir betrachten zunächst vier verschiedene Typen von zufälligen Graphen (A, B, C und D) wie folgt:

A: Jede Kante hat mit Wahrscheinlichkeit  $\frac{1}{2}$  das Gewicht 1, andernfalls das Gewicht 0.

B: Jede Kante hat ein Gewicht aus dem Intervall  $[-50, 50]$  mit gleichverteilter Wahrscheinlichkeit.

C: Für Graphen der Größe  $n \geq 10$  hat jede Kante mit Wahrscheinlichkeit  $\frac{10}{n}$  das Gewicht 1, andernfalls das Gewicht 0. Dadurch ist der Erwartungswert der Knotengrade unabhängig von der Größe des Graphen.

D: Für Graphen der Größe  $n$  haben die Kante  $(i, j)$  das Gewicht 1 mit Wahrscheinlichkeit

$$p = \begin{cases} 0,1 & \text{falls } i \leq n/2 \text{ und } j > n/2, \\ 0,05 & \text{sonst;} \end{cases}$$

andernfalls ist das Kantengewicht 0. Dies führt zu einem sog. großen Schnitt zwischen den Knoten in  $[1, \dots, n/2]$  einerseits und denjenigen in  $[n/2 + 1, \dots, n]$  andererseits.

Die Typen A, C und D haben also nichtnegative, ganzzahlige Kantengewichte (sogar binäre). Hingegen treten bei Typ B negative, nicht-ganzzahlige Kantengewichte auf; bei ihm gilt also nicht die starke 0,878-Schranke, sondern die Abschätzung (2.13).

Abk.	Algorithmus/Lösung(swert) von
SDP	semidefinite Relaxierung des Max-Cut-Problems, (2.7)
GW	Goemans-Williamson-Verfahren ohne lokale Suche
GW-ls	Goemans-Williamson-Verfahren mit lokaler Suche
TA	trigonometrisches SDP (4.4)
TA-1	TA-Näherungsverfahren, Rundungstechnik „kombinatorisch“
TA-ls	TA-Näherungsverfahren, Rundungstechnik „Zufallsschnitte“
TA-max	beste Näherungslösung von TA-1 und TA-ls
R-ls	Zufallslösungen („Monte Carlo“) mit lokaler Suche

Tabelle 6.1.: Verwendete Abkürzungen in den Auswertungen

### Betrachtete Instanzen

In [9] wurden die Typen A, B, C und D in den Dimensionen  $n \in \{50, 100, 200\}$  betrachtet mit jeweils 50, 20 bzw. 5 Instanzen. Größere Dimensionen ließen sich damals noch nicht praktikabel betrachten. So betrug die Laufzeit für ein Problem der Größe  $n = 200$  durchschnittlich ein bis drei Stunden.

Für unseren numerischen Vergleich von Methoden aus Kapitel 5 mit den herkömmlichen Techniken aus [9] haben wir sehr ausgiebige Berechnungen durchgeführt. Je Graphentyp haben wir zunächst für die Dimensionen

- $n \in \{20, 25, 30, \dots, 300\}$  jeweils 100 Instanzen,
- $n \in \{310, 320, 330, \dots, 540\}$  jeweils 50 Instanzen,

sowie für den Typ B für

- $n = 1000$  weitere 50 Instanzen,
- $n = 2000$  weitere 10 Instanzen,

erzeugt.

### 6.1.2. Algorithmen

Prinzipiell wurden alle nachfolgend beschriebenen Verfahren auf alle Probleminstanzen angewendet. Lediglich die Berechnung der exakten Lösungen wurde auf Dimensionen  $n \leq 100$  beschränkt. Die Abkürzungen der Verfahren, wie sie in den nachfolgenden Grafiken benutzt werden, sind in Tabelle 6.1 zusammengefasst.

### Lösung der semidefiniten Relaxierung

Die semidefinite Relaxierung des Max-Cut-Problems, das SDP (2.7), haben wir mittels des Verfahrens aus [12] gelöst. Das dort beschriebene Innere-Punkte-Verfahren für semidefinite Programme ist recht effizient und insbesondere für (2.7) zügig implementierbar. Da es sich um ein primal-duales Verfahren handelt, erhält man neben dem Optimalwert  $Z_{SDP}^*$  auch sofort eine primale Optimallösung  $X^*$  (von (2.7)) und eine duale Optimallösung  $y^*$  (von (2.10)). Wir haben das Verfahren beendet, wenn der primale und duale Wert sich um weniger als den Faktor  $10^{-12}$  unterschieden. Die duale Optimallösung wird für die Berechnung der exakten Lösung des Max-Cut-Problems noch benötigt.

### Goemans-Williamson-Verfahren ohne lokale Suche

Mittels der (singulären) Cholesky-Zerlegung der Optimallösung  $X^* = B^T B$  berechnen wir mit dem Verfahrens aus Abschnitt 2.3 je Problem Instanz  $5n$  zufällige Lösungen. Nur die beste Lösung wird behalten.

### Goemans-Williamson-Verfahren mit lokaler Suche

Wieder werden mittels der Cholesky-Zerlegung von  $X^*$  jeweils  $5n$  zufällige Lösungen berechnet. Für jede zufällige Lösung wird anschließend eine lokale Suche durchgeführt.

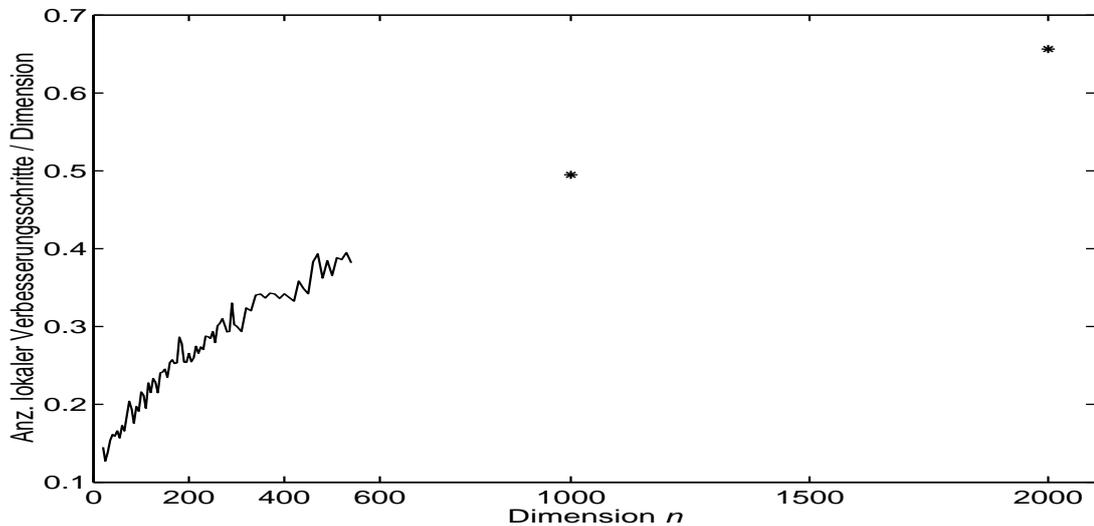


Abbildung 6.1.: Verhältnis der Anzahl der Suchschritte je lokaler Suche zur Dimension beim Goemans-Williamson-Verfahren (hier Typ B)

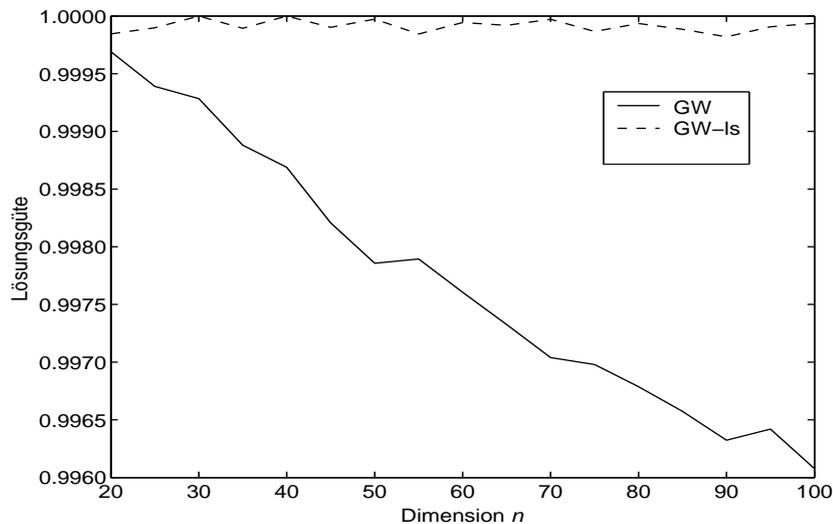


Abbildung 6.2.: Verhältnis von GW bzw. GW-ls zum exakten Lösungswert (hier Typ A)

Dabei wird für jeden Knoten einzeln geprüft, ob dessen Verschiebung in die jeweils andere Menge der Partitionierung den Optimalwert weiter verbessert. Diese Prüfung wird so lange wiederholt, bis keine Verbesserung mehr erzielt werden kann. Die gefundene Lösung ist dann in dem Sinn lokal optimal, dass sie nicht durch Verschieben eines einzelnen Knotens verbessert werden kann. Die beste der so erhaltenen  $5n$  lokal optimalen Zufallslösungen wird behalten.

Wie in Abbildung 6.1<sup>1</sup> zu sehen ist, steigt die Laufzeit je lokaler Suche mit wachsendem  $n$  superlinear an. Dies hängt mit der Qualität der Näherungslösungen beim Goemans-Williamson-Verfahren ohne lokale Suche zusammen: In Abbildung 6.2 wird gezeigt, wie die Lösungen ohne lokale Suche mit wachsender Dimension im Verhältnis zur exakten Lösung schlechter werden. Daher muss die lokale Suche mit wachsendem  $n$  eine gewisse Arbeit leisten, um an den exakten Lösungswert fast heranzukommen.

### Lösung des trigonometrischen SDP

Das trigonometrische SDP (4.4) wird mittels des Verfahrens aus Kapitel 5 gelöst. Das Verfahrens bricht ab, wenn sich der primale und der duale Lösungswert des linearisierten Problems (5.2) um weniger als den Faktor  $10^{-6}$  unterscheiden. Wie oben angedeutet, er-

<sup>1</sup>Die Werte für  $n \in [20, 540] \setminus \{20, 25, 30, \dots, 300, 310, 320, \dots, 540\}$  sind interpoliert, um eine bessere Lesbarkeit der Graphiken zu erreichen.

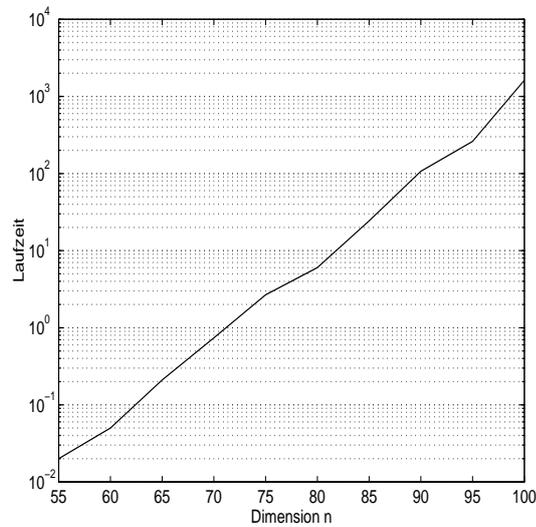


Abbildung 6.3.: Laufzeit [CPU-Sekunden] des Branch-and-Bound-Verfahrens von Goemans (hier Typ D)

reichen wir mit diesem Verfahren nicht immer die exakte Optimallösung. Wir bezeichnen die näherungsweise optimale Lösung, die wir mit diesem Verfahren ermitteln, mit  $\bar{X}$ .

### Näherungslösungen mittels des trigonometrischen SDP

Aus der näherungsweise optimalen Lösung  $\bar{X}$  von (4.4) werden, entsprechend der im Abschnitt 5.3 erläuterten Methoden, die zwei verschiedenen Näherungslösungen gewonnen.

Bei der Technik der Zufallsschnitte basierend auf der Cholesky-Zerlegung von  $\bar{X}$  erzeugen wir  $n$  Zufallsschnitte und führen jeweils anschließend eine lokale Optimierung durch. Darüber hinaus ermitteln wir einen Mittelwert für  $n/5$  Zufallsschnitte mit lokaler Suche, um die Sensibilität der Lösungswerte gegenüber der Anzahl der Zufallsschnitte zu prüfen.

### Exakte Lösung der Max-Cut-Probleme

Für kleinere Dimensionen ( $n \leq 100$ ) können wir die exakte Lösung der Max-Cut-Probleme in überschaubarer Zeit mit Hilfe der in Abschnitt 3.2 erläuterten Branch-and-Bound-Methode von Goemans bestimmen. Trotz der hohen Effizienz dieses Verfahrens bleibt das prinzipiell exponentielle Laufzeitverhalten erwartungsgemäß bestehen (siehe Abbildung 6.3).

Verfahren	A	B	$B_{1000}, B_{2000}$	C	D	gesamt
SDP	0,27	0,27	0,21	0,27	0,40	1,42
GW	0,05	0,05	0,05	0,05	0,04	0,23
GW-ls	1,20	1,50	1,82	0,95	1,08	6,55
TA	2,56	1,89	3,70	1,72	1,98	11,86
TA-1	0,95	0,95	1,01	0,80	0,52	4,24
TA-ls	0,89	0,98	1,05	0,84	0,84	4,60
exakt ( $n \leq 100$ )	4,29	6,23	–	1,50	2,34	14,36
gesamt	10,22	11,88	7,91	6,15	7,23	43,59

In den Spalten sind die 4 verschiedenen Graph-Typen angeordnet. Die Zeitangaben beziehen sich jeweils auf alle gerechnete Beispiele aller betrachteten Dimensionen (siehe „Betrachtete Instanzen“ im Unterabschnitt 6.1.1).

Tabelle 6.2.: Gesamt-Laufzeiten der numerischen Standard-Tests [CPU-Tage]

### Zufallsschnitte („Monte Carlo“ mit lokaler Suche)

Als einfachste Vergleichsmethode, die auch sehr schnell ist, haben wir das unter der Bezeichnung „Monte-Carlo-Methode“ bekannte Raten einer Lösung implementiert.<sup>2</sup> Dabei wurde, wie auch bei „GW-ls“ und „TA-ls“, eine lokale Optimierung vorgenommen.

### 6.1.3. Randbedingungen der numerischen Berechnungen

Die Berechnungen fanden auf einem Rechner mit zwei Intel-Xeon-Prozessoren statt, die mit 2,8 GHz getaktet waren. Auf Grund der so genannten Hyperthreading-Technik der Xeon-Prozessoren stehen dem System vier virtuelle Prozessoren zur Verfügung. In den Implementierungen der Algorithmen wurden keine parallelen Routinen benutzt, so dass stets nur ein Prozessor genutzt wurde. Mit 3 GB war der Computer mit ausreichend Hauptspeicher ausgerüstet.

Fast alle Algorithmen wurden in Matlab implementiert. Lediglich der Branch-and-Bound-Algorithmus von Goemans wurde in C programmiert, da hier eine Effizienzsteigerung unabdingbar war. Zur Messung der Laufzeit wurde die vom Betriebssystem (Linux, Kernel 2.4.21) ermittelte CPU-Zeit benutzt. Um mehr Fallbeispiele betrachten zu können, fanden meist zwei Berechnungen parallel statt. Beide Berechnungen wurden auf jeweils einem anderen Prozessor verankert, so dass theoretisch eine gegenseitige Beeinträchtigung ausgeschlossen werden könnte. Praktisch führten jedoch die Systemaufrufe zum Speichermanagement doch zu einer (geringfügigen) Beeinträchtigung.

<sup>2</sup>Man könnte auch, in Anlehnung an die Zufallsschnitt-Erzeugung im Goemans-Williamson-Verfahren (vgl. Seite 25) von einer  $I$ -basierten Zufallsschnitt-Erzeugung sprechen.

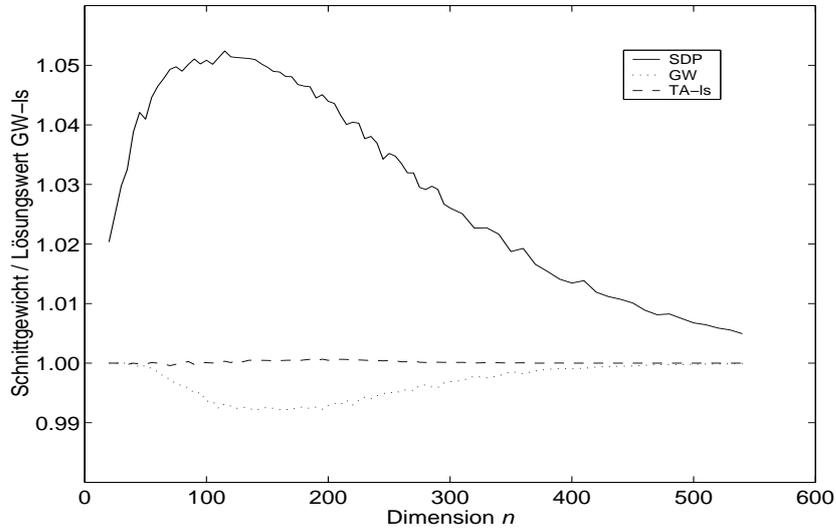


Abbildung 6.4.: Lösungswerte verschiedener Algorithmen bei Graphen des Typs D

Insgesamt haben wir sehr umfangreiche numerische Tests durchgeführt. Wie aus Tabelle 6.2 ersichtlich, benötigten allein die Standard-Tests rund 44 CPU-Tage.

### Graph-Typ D

Bei den numerischen Tests hat sich herausgestellt, dass der Graph-Typ D wenig Aussagekraft hat (vgl. Abbildung 6.4): Bei höheren Dimensionen wird die Ermittlung der Optimallösung immer leichter, da die zufälligen Störungen eine immer kleinere Rolle spielen. Selbst das Goemans-Williamson-Verfahren *ohne* lokale Suche findet bei großem  $n$  sehr oft die gleichen Lösungen wie GW-ls *mit* lokaler Suche. Da außerdem die obere Schranke (Optimalwert von SDP) gegen die gefundenen Näherungslösungen zu konvergieren scheint, verzichten wir deshalb im Folgenden auf weitere Betrachtungen dieses Graph-Typs.

## 6.2. Naiver Vergleich von GW und $\mathcal{TA}$

Wir vergleichen zunächst die jeweils einfachste Variante der Algorithmen.

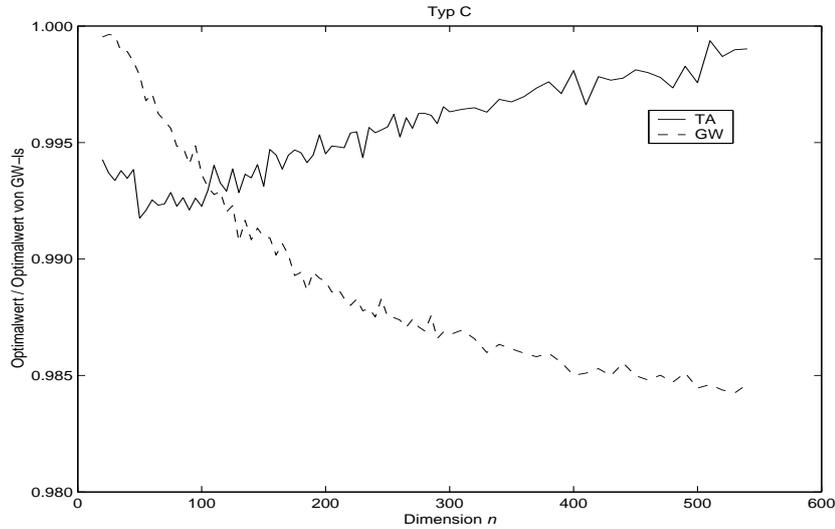


Abbildung 6.5.: Verhältnis von TA bzw. GW zum Referenzwert GW-ls (hier Typ C)

### Einfache untere Schranken

Da der Algorithmus zur Lösung der trigonometrischen Approximation des Max-Cut-Polytops (4.4) aus Kapitel 5 stets zulässige Iterierte hat, ist auch die Approximationslösung  $\bar{X}$  Element von  $\mathcal{MC}$ . Wie schon im Abschnitt 5.3 erläutert, ist somit  $\bar{Z}_{TA} = \frac{1}{4}L \bullet \bar{X}$  eine untere Schranke an den Optimalwert des Max-Cut-Problems.

Wir vergleichen diese einfache untere Schranke mit dem Goemans-Williamson-Verfahren ohne lokale Suche in Abbildung 6.5 (Graphtyp C). Bei den Graph-Typen A und B verhalten sich die beiden Kurven zueinander qualitativ ähnlich.

Der relativ starke Qualitätsverlust des Goemans-Williamson-Verfahrens erklärt sich durch den Anstieg des Ranges der Optimallösung des SDPs (2.7). Da sich dadurch der Raum, in dem die Zufallsschnitte erzeugt werden, vergrößert, müsste die Anzahl der Zufallsschnitte exponentiell mit dem Rang von der Optimallösung von (2.7) wachsen. Tatsächlich wächst sie aber nur linear ( $5n$ ).

Offensichtlich kommt die Technik der trigonometrischen Approximation besser mit größeren Dimension zurecht.

Betrachtet man das Verhältnis von TA zum *exakten* Optimalwert (für  $n \leq 100$ ), so ergibt sich je nach Graph-Typ ein unterschiedliches Bild: Bei Typ B bleibt der Wert von TA recht konstant bei 96 % des Optimalwerts, bei Typ C fällt der Wert von TA mit wachsendem  $n$  leicht von 99,4 %. Bei Typ A verbessert sich TA sogar leicht von 99,4 % ( $n = 20$ ) auf 99,7 % ( $n = 100$ )!

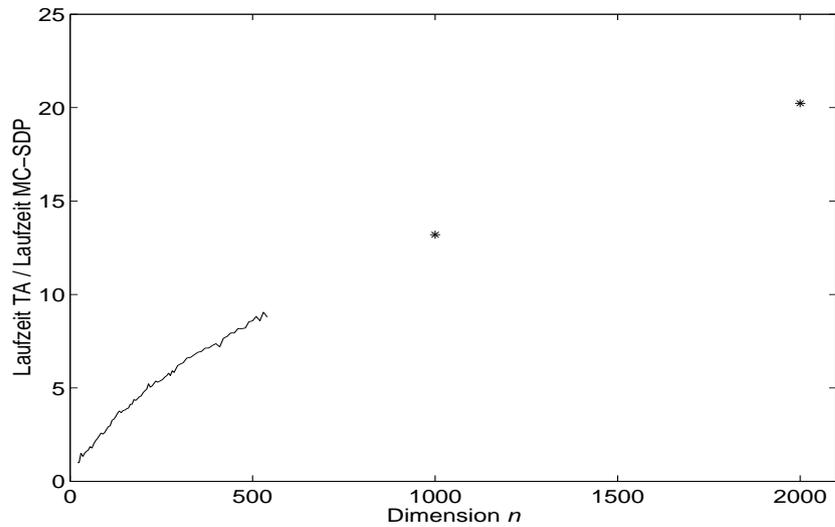


Abbildung 6.6.: Verhältnis der Laufzeiten der Algorithmen zur Lösung von (4.4) und (2.7) (hier Typ B)

### Laufzeiten

In Abbildung 6.6 werden die verwendeten Lösungsalgorithmen für die Probleme (4.4) und (2.7) bezüglich ihrer Laufzeit verglichen. Es stellt sich dabei heraus, dass die Laufzeit unseres einfachen Algorithmus' 5.1 zur Lösung der  $\mathcal{TA}$ -Approximation sublinear wächst gegenüber dem Algorithmus zur Lösung der semidefiniten Relaxierung.

Das Innere-Punkte-Verfahren zur Lösung der semidefiniten Relaxierung selbst hat eine empirische Laufzeit in etwa der Größenordnung  $n^3$ .

### 6.3. Die verschiedenen Varianten bei der trigonometrischen Approximation

Wir vergleichen nun die zwei Rundungstechniken im Anschluss an die Lösung der trigonometrischen Approximation, nämlich die kombinatorische Technik (Unterabschnitt 5.3.1) und die Zufallsschnitte (Unterabschnitt 5.3.2), untereinander und mit dem Referenzverfahren GW-ls.

6.3. Die verschiedenen Varianten bei der trigonometrischen Approximation

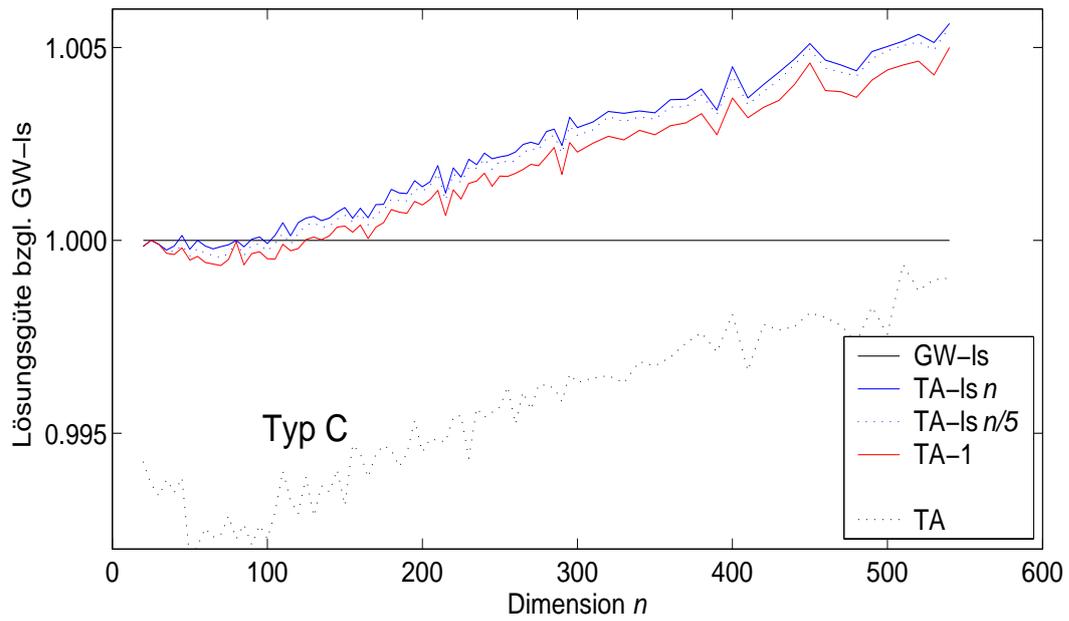
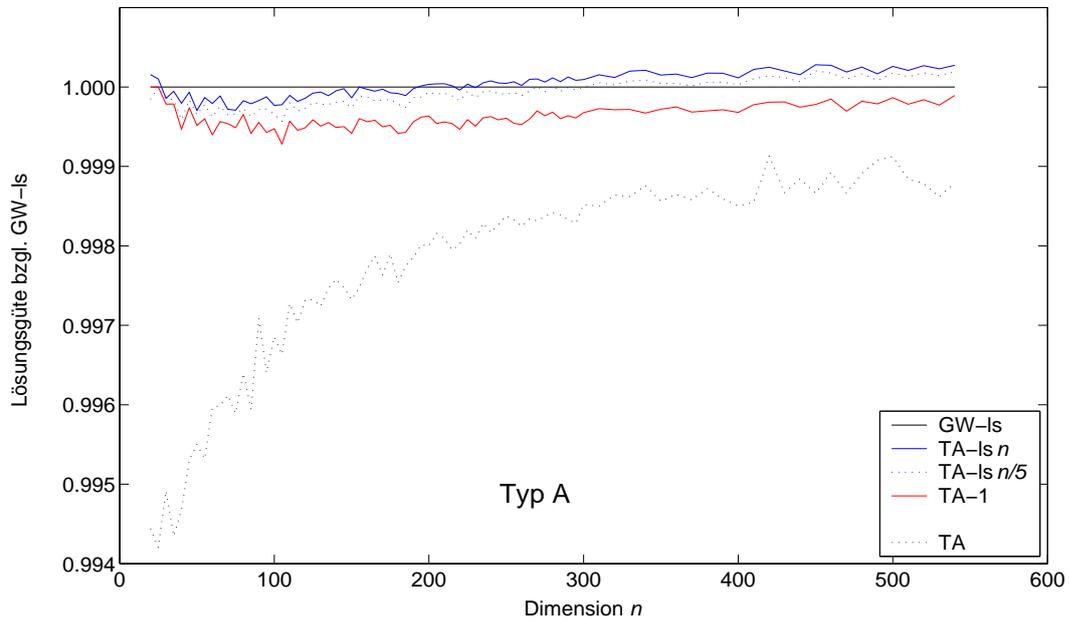


Abbildung 6.7.: Verschiedene Varianten zur Bestimmung einer Näherungslösung basierend auf der trigonometrischen Approximation  $\mathcal{TA}$

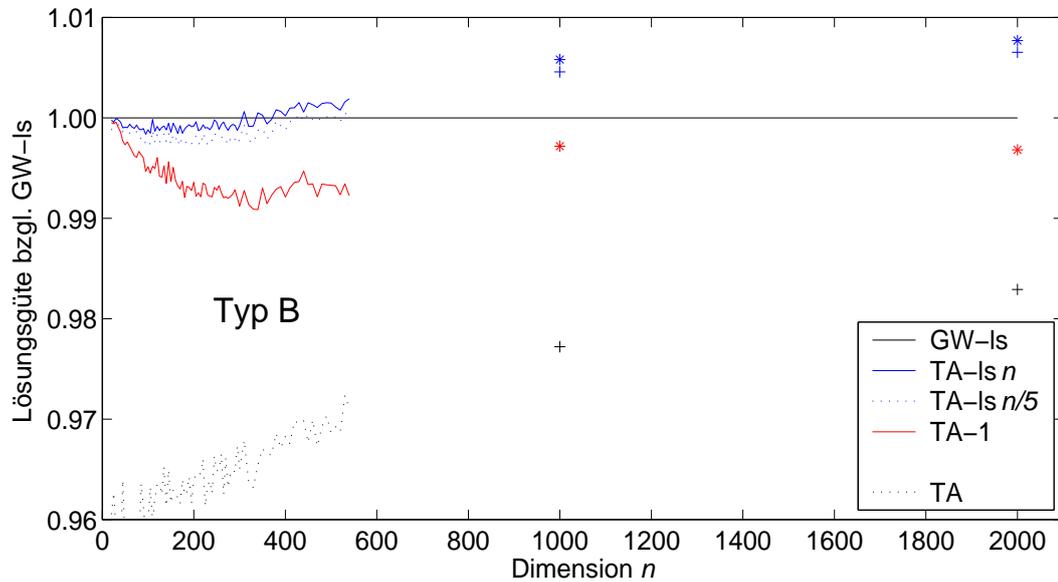


Abbildung 6.8.: Verschiedene Varianten zur Bestimmung einer Näherungslösung basierend auf der trigonometrischen Approximation  $\mathcal{TA}$

### Lösungsgüte

In den Abbildungen 6.7 und 6.8<sup>3</sup> werden die Lösungswerte der Verfahren TA-1 und TA-ls ( $n$  und  $n/5$  Zufallsschnitte) ins Verhältnis gesetzt zum Lösungswert des Verfahrens GW-ls ( $5n$  Zufallsschnitte). Wir bilden hier alle drei Graph-Typen ab, da die Ergebnisse relativ stark zwischen den Typen variieren.

Für die Rundungstechniken für die trigonometrische Approximation  $\mathcal{TA}$  lässt sich zunächst unabhängig vom Graph-Typ feststellen, dass sich die erste Rundungstechnik nicht bewährt. Das schlechte Abschneiden dieser kombinatorischen Variante lässt sich dadurch erklären, dass die Zufallsschnitt-Technik viele der kombinatorisch durchprobierten Lösungen ohnehin mit abdeckt, darüber hinaus aber noch einige weitere Lösungen findet.

Der Vergleich der TA-ls-Rundung mit dem GW-ls-Verfahren zeigt bei allen drei Graph-Typen die Überlegenheit von TA-ls für großes  $n$ . Bei kleinen Dimensionen ist GW-ls besser, da unser TA-Verfahren (Algorithmus 5.1) noch offensichtliche Schwierigkeiten hat, das trigonometrische SDP exakt zu lösen. Andererseits vergrößert sich die Abweichung der Lösung unseres Verfahrens von der Optimallösung mit wachsender Dimension  $n$  of-

<sup>3</sup>Man beachte, dass die Lösungsgüte hier nicht wie in (2.13) berechnet wird (negative Kantengewichte beim Graph-Typ B!).

fenbar nur gering oder überhaupt nicht. Man beachte, dass TA-ls hier nur mit  $n$  bzw. sogar  $n/5$  Zufallsschnitten arbeitet, GW-ls hingegen mit  $5n$ !

Je nach Graph-Typ variiert der Punkt, an dem TA-ls besser als GW-ls wird. Für Typ B haben wir deshalb noch die Dimensionen  $n = 1000$  und  $n = 2000$  betrachtet. Auch unterscheidet sich das Maß, mit dem TA-ls besser als GW-ls ist je nach Typ. Zwar sind die Unterschiede nur im Promille-Bereich, andererseits sind aber die Lösungen von TA-ls vermutlich bereits *sehr* nahe am Optimalwert. So sind für  $n \leq 100$  die Lösungswerte von TA-ls im Mittel weniger als 0,1 % vom exakten Optimalwert entfernt.

### Laufzeiten

Die Laufzeiten der drei Varianten der TA-Rundungstechniken im Vergleich zur Lösung des trigonometrischen SDPs selbst sind relativ unabhängig von der Dimension. So benötigt die Variante 1 ca. 50 % und die Variante 2 (TA-ls) ca. 10 % der Laufzeit des Algorithmus' zur Lösung von (4.4). Generell verringert sich der Anteil leicht mit steigender Dimension.

Die Laufzeiten zur Lösung der semidefiniten Programme haben wir bereits in Abbildung 6.6 verglichen.

## 6.4. GW und TA mit lokaler Suche

### 6.4.1. Überblick

Wir untersuchen nun noch die Lösungsgüte von TA-ls und GW-ls in Abhängigkeit der Anzahl der erzeugten Zufallsschnitte. Dabei vergleichen wir die beiden Verfahren noch mit der „Monte-Carlo-Methode“ (mit R-ls bezeichnet), dem zufälligen Raten von Lösungen mit anschließender lokaler Optimierung.

#### Kleine Probleme

In Abbildung 6.9 ist der Vergleich von TA-ls, GW-ls und R-ls für kleines  $n$  (hier Graph-Typ B mit  $n = 250$ ) zu sehen. Es zeigt sich, dass TA-ls mit wenigen Zufallsschnitten bereits sehr gute Lösungen findet, mit wachsender Anzahl von Zufallsschnitten sich jedoch kaum mehr verbessern kann und schließlich sogar von GW-ls überholt wird. Die Monte-Carlo-Methode braucht sehr viel länger, um die Qualität von GW-ls oder TA-ls zu erreichen.

#### Große Probleme

In Abbildung 6.10 ist nun die gleiche Untersuchung für  $n = 1000$  zu sehen. Hier fällt eine klare Hierarchie zwischen den drei betrachteten Verfahren auf. Offensichtlich würde GW-ls noch sehr viel mehr Zufallsschnitte benötigen, um die Lösungsgüte von TA-ls zu

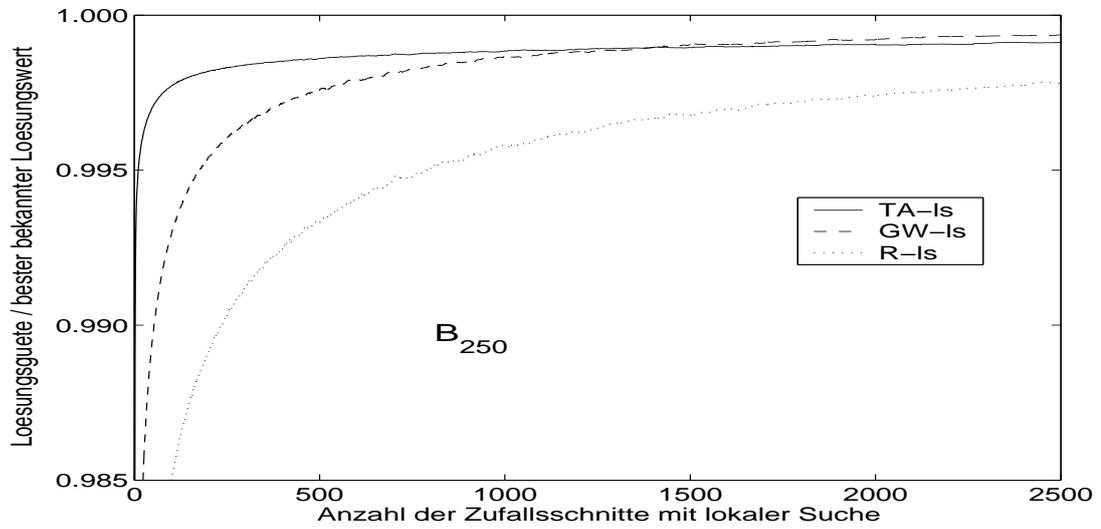


Abbildung 6.9.: Erwartete Lösungsgüte (bzgl. bester bekannter Lösung) in Abhängigkeit der Anzahl der Zufallsschnitte

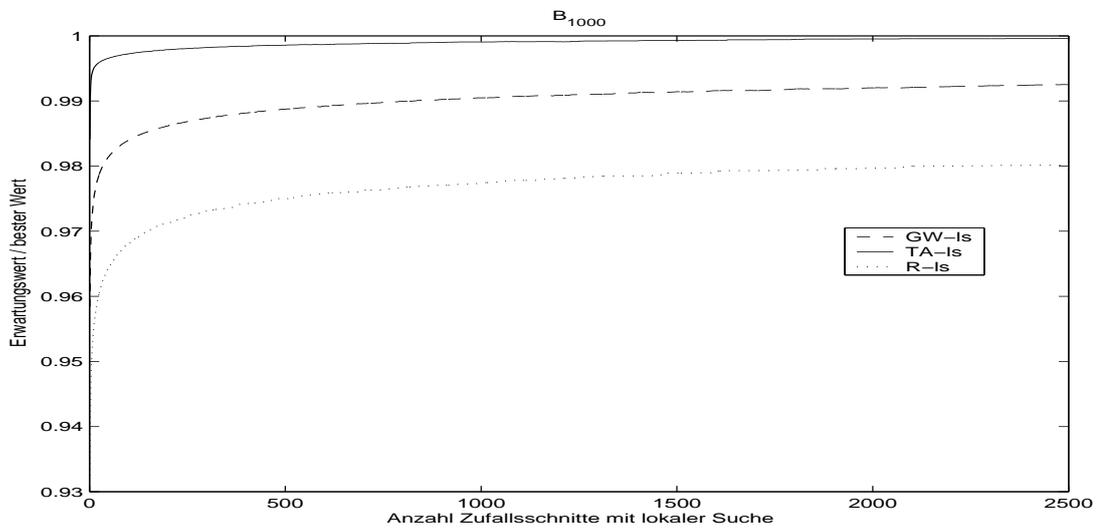


Abbildung 6.10.: Erwartete Lösungsgüte (bzgl. bester bekannter Lösung) in Abhängigkeit der Anzahl der Zufallsschnitte

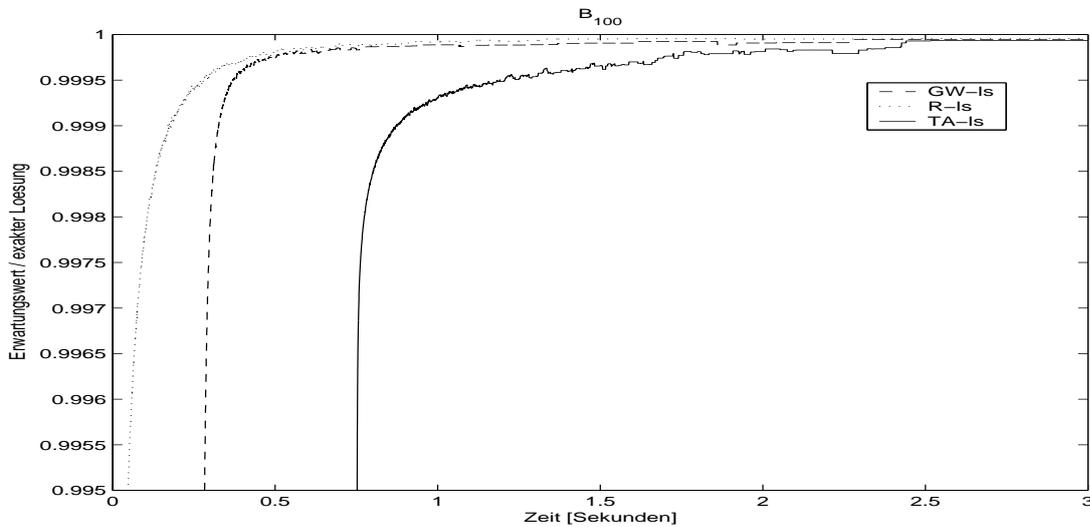


Abbildung 6.11.: Erwartete Lösungsgüte (bzgl. des exakten Optimalwerts) in Abhängigkeit der Zeit

erreichen. Wie oben angedeutet macht sich hier der Rang der Optimallösung von (2.7) bemerkbar.

### 6.4.2. Überlegungen für die Praxis

Berücksichtigt man den Zeitbedarf für die Lösung der semidefiniten Programme und für die Erzeugung der Zufallsschnitte, so zeigt sich, dass alle drei Algorithmen (GW-ls, TA-ls und R-ls) ihre Berechtigung haben – je nach Problemgröße und Graphen-Typ.

Für kleine Probleme kann sogar das „dumme“ Erzeugen von  $I$ -basierten Zufallsschnitten die effizienteste Lösung sein (siehe Abbildung 6.11 für  $n = 100$ ), um eine gute Näherungslösung von Max-Cut zu erzeugen.

Für größere Probleme hingegen lohnt sich der Aufwand zur Lösung der semidefiniten Programme (vgl. Abbildung 6.12 für  $n = 1000$ ). Wichtig ist hier, wie viel Zeit man zur Verfügung hat. Bei sehr zeitkritischen Anwendungsfällen wird man auch hier die Monte-Carlo-Methode benutzen (müssen). Hat man etwas Zeit, so ist das Goemans-Williamson-Verfahren mit lokaler Suche die ideale Wahl. Wenn hingegen eine nahezu optimale Lösung von größerer Bedeutung ist und man etwas länger Zeit hat, so liefert die trigonometrische Approximation mit Zufallsschnitt-Erzeugung (und lokaler Suche) im Mittel die beste Lösung.

Man beachte, dass man für große Probleme nicht mehr den Zeitpunkt abwarten kann,

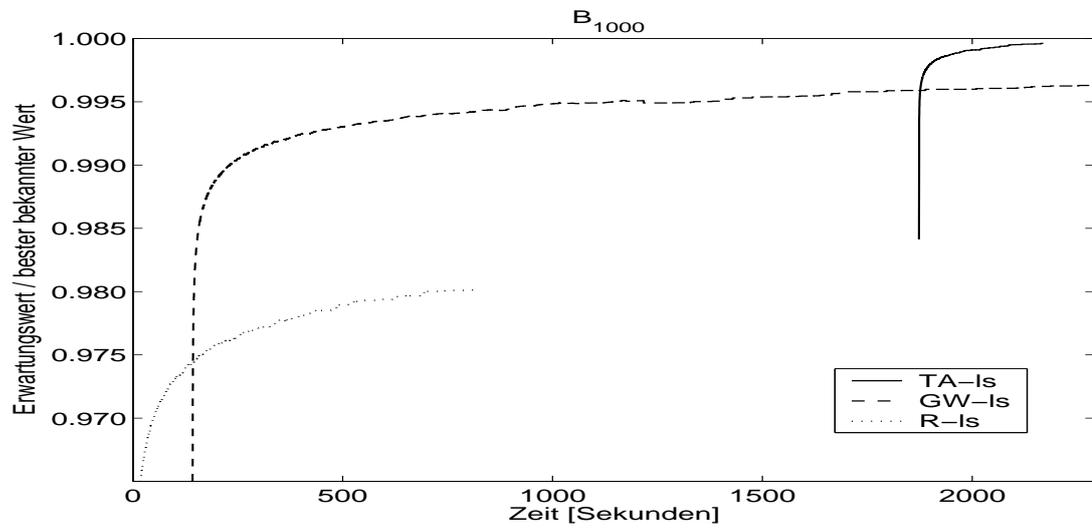


Abbildung 6.12.: Erwartete Lösungsgüte (bzgl. der besten bekannten Lösung) in Abhängigkeit der Zeit

an dem die GW-Is-Lösung besser wird als die TA-Is-Lösung: Im Beispiel der Abbildung 6.12 verbessert sich für die doppelte Anzahl an Zufallsschnitten die GW-Is-Lösung nur noch um weniger als 0,1 %-Punkte.

# Zusammenfassung

Wir haben zwei Varianten der trigonometrischen Formulierung des Max-Cut-Problems analysiert, die in [9] erstmals eingeführt wurde. Dieses trigonometrische semidefinite Programm (SDP) kann zum einen mit linearer Zielfunktion über eine nicht-konvexe Menge  $\mathcal{TA}$  formuliert werden. Eine äquivalente Formulierung besteht in einer nicht-konvexen Zielfunktion über der konvexen Menge  $\mathcal{SR}$ , der semidefiniten Relaxierung des Max-Cut-Polytops  $\mathcal{MC}$ .

Für die Menge  $\mathcal{TA}$  wurde erstmals ihre Beziehung zu  $\mathcal{MC}$  eingehend untersucht. Als innere Approximation der Menge  $\mathcal{MC}$  wird  $\mathcal{TA}$  in gewissem Sinne optimal durch eine skalare Abbildung aus  $\mathcal{SR}$  erzeugt. Zur Frage der Sternförmigkeit wurde eine Vermutung über eine Identität von gewissen Koeffizienten motiviert und plausibilisiert. Falls diese Vermutung richtig ist, folgt unter anderem, dass  $\mathcal{TA}$  sternförmig ist.

Die praktisch bedeutsame Frage der Regularität der Menge  $\mathcal{TA}$  konnte ebenfalls mit Hilfe dieser Vermutung eingegrenzt werden. Falls die Vermutung korrekt ist, folgt dass der Tangentialkegel von  $\mathcal{TA}$  in den Ecken ein nicht-leeres Inneres besitzt, eine notwendige Voraussetzung für die Regularität. Leider gibt es Punkte in Kanten von  $\mathcal{TA}$ , in denen der Tangentialkegel ein leeres Inneres hat. Da eine lokal optimale Ecke auch ein globales Optimum ist, könnte man im Falle der Regularität von  $\mathcal{TA}$  in den Ecken möglicherweise prüfen, ob eine Ecke eine Optimallösung eines Max-Cut-Problems ist.

Wir haben ein erstes Verfahren entwickelt, mit dem man das trigonometrische Formulierung des Max-Cut-Problems näherungsweise lösen kann. Für die so oder mit anderen Verfahren erhaltene Näherungslösung haben wir Strategien diskutiert, wie man eine gute zulässige Lösung für das Max-Cut-Problem erhalten kann.

Unsere ausführliche numerische Analyse des neuen Verfahrens, die sehr viele Fallbeispiele überprüfte, hat erfreuliche Resultate gezeigt. Bereits die Näherungslösungen des trigonometrischen SDP, die wir mit unserem Verfahren erhalten haben, stellen eine gute untere Schranke an die Optimallösung des Max-Cut-Problems dar.

Die Strategie, mit Zufallsschnitten, die auf den Näherungslösungen des trigonometrischen SDP basieren, Näherungslösungen für das Max-Cut-Problem zu erzeugen, zeigte hervorragende Resultate. In allen vier Kategorien von Fallbeispielen, die betrachtet wurden, war spätestens ab der Problemgröße  $n \approx 400$  unser Verfahren besser als das Goemans-Williamson-Verfahren. Somit wurde für große Problemgrößen ein Verfahren

## 6. Numerische Ergebnisse

---

eingeführt, das numerisch deutlich bessere Lösungen liefert als das bisherige Standard-Verfahren.

# A. Bezeichnungen

## Graphentheorie

$G = (V, E)$	(ungerichteter) Graph mit Knotenmenge $V$ und Kantenmenge $E$
$e \in E$	Kante eines Graphen
$e = \{i, j\}$	(ungerichtete) Kante zwischen den Knoten $i$ und $j$
$W = (w_{ij})$	Adjazenzmatrix mit Kantengewichten $w_{ij}$ der Kanten $\{i, j\}$ , hier: $w_{ii} = 0$
$\delta(U)$	Schnitt eines Graphen mit der Partitionierung $(U, \bar{U} = V \setminus U)$ , $U \subset V$
$w(\delta(U))$	Gewicht des Schnittes $\delta(U)$
$mc(G)$	Gewicht des maximalen Schnittes des Graphen $G$

## Lineare Algebra, Matrixnotationen

$\text{diag}(X)$	Diagonale der Matrix $X$ als Vektor
$\text{Diag}(x)$	Vektor $x$ als Diagonalmatrix
$e$	Vektor $(1, 1, \dots, 1)^T$ passender Dimension
$\sin[X]$	Elementweises Anwenden der Funktion $\sin$ auf die Matrix $X$
$A \circ B$	Hadamard-Produkt zweier Matrizen, $(A \circ B)_{i,j} = A_{i,j}B_{i,j}$
$A \bullet B$	Inneres Produkt zweier $n \times n$ -Matrizen, $A \bullet B = \sum_{i,j} A_{i,j}B_{i,j} = \text{Spur}(A^T B)$
$X \succeq 0$	$X$ ist positiv semidefinit und symmetrisch, $X \in \mathcal{S}_+^n$

## Mengen

$S_n$	Einheitssphäre im $\mathbb{R}^n$ : $\{x \in \mathbb{R}^n \mid \ x\ _2 = 1\}$
$\mathcal{S}^n$	Menge der symmetrischen Matrizen $X = X^T \in \mathbb{R}^{n \times n}$
$\mathcal{S}_+^n$	Menge der symmetrischen positiv semidefiniten Matrizen $X = X^T \succeq 0$
$\mathcal{L}$	affiner Unterraum der sym. Matrizen: $\{X = X^T \in \mathbb{R}^{n \times n} \mid \text{diag}(X) = e\}$
$\mathcal{MC}$	Max-Cut-Polytop
$\mathcal{TA}$	Trigonometrische Approximation von $\mathcal{MC}$
$\mathcal{SR}$	Semidefinite Relaxierung von $\mathcal{MC}$

## Literaturverzeichnis

- [1] ANJOS, M. F. ; WOLKOWICZ, H.: A strengthened SDP relaxation via a second lifting for the Max-Cut problem / University of Waterloo. Waterloo, Ontario, N2L 3G1 Canada, 1999 ( CORR-95). – Forschungsbericht. <http://citeseer.nj.nec.com/anjos99strengthened.html>
- [2] BARAHONA, F. ; GRÖTSCHEL, M. ; JÜNGER, M. ; REINELT, G.: An application of combinatorial optimization to statistical physics and circuit layout design. In: *Operations Research* 36 (1988), Nr. 3, S. 493ff
- [3] BURER, S. ; MONTEIRO, R. D. C. ; ZHANG, Y.: Rank-Two Relaxation Heuristics for Max-Cut and Other Binary Quadratic Programs. In: *SIAM Journal on Optimization* 12 (2001), Nr. 2, S. 503–521
- [4] DE SIMONE, C. ; DIEHL, M. ; JÜNGER, M. ; MUTZEL, P. ; REINELT, G. ; RINALDI, G.: Exact Ground States for Two-Dimensional  $\pm J$  Ising Spin Glasses / Institut für Informatik, Universität zu Köln. Pohligstraße 1, D-50969 Köln, 1996 ( 96.217). – Report
- [5] DEZA, M. ; LAURENT, L.: *Geometry of cuts and metrics*. Bd. 15 von Algorithms and Combinatorics. Berlin : Springer, 1997
- [6] FEIGE, Uriel ; SCHECHTMAN, Gideon: On the optimality of the random hyperplane rounding technique for MAX CUT. In: *Random Structures & Algorithms* 20 (2002), S. 403–440
- [7] FEIGE, Uriel ; GOEMANS, Michel: Approximating the value of two prover proof systems, with applications to max 2sat and max dicut. In: *Proceedings of 3rd Israel Symposium on the Theory of Computing and Systems*, 1995, S. 182–189
- [8] GOEMANS, Michel X.: Semidefinite Programming in combinatorial optimization. In: *Mathematical Programming* 79 (1997), S. 143–161
- [9] GOEMANS, Michel X. ; WILLIAMSON, David P.: Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming. In: *Journal of the Association for Computing Machinery* 42 (1995), Nr. 6, S. 1115–1145

- [10] GOLUB, G. H. ; LOAN, C. F. V.: *Matrix Computations*. Baltimore, MD : The Johns Hopkins University Press, 1983
- [11] HÅSTAD, J.: Some optimal inapproximability results. In: *Proceedings of the 29th ACM Symposium on Theory of Computing*, 1997, S. 1–10
- [12] HELMBERG, Christoph ; RENDL, Franz ; VANDERBEI, Robert J. ; WOLKOWICZ, Henry: An Interior-Point Method for Semidefinite Programming. In: *SIAM Journal on Optimization* 6 (1996), May, Nr. 2, S. 342–361. – URL: <ftp://orion.uwaterloo.ca/pub/henry/reports/sdp.ps.gz>
- [13] HIRSCHFELD, Bernd W.: *Lösung des MaxCut-Problems mittels semidefiniter Programmierung am Beispiel von Ising-Spinalgäsern*, Julius-Maximilians-Universität Würzburg, Diplomarbeit, 1999
- [14] HORN, Roger A. ; JOHNSON, Charles R.: *Matrix analysis*. Cambridge Univ. Pr., 1985
- [15] JARRE, Florian: *Convex Analysis on Symmetric Matrices*. Juli 1998. – Universität Würzburg
- [16] JARRE, Florian ; STOER, Josef: *Optimierung*. Berlin, Heidelberg : Springer, 2004
- [17] JÜNGER, Michael ; RINALDI, Giovanni: Relaxations of the Max Cut Problem and Computation of Spin Glass Ground States / Institut für Informatik, Universität Köln. Pohlstraße 1, D-50969 Köln, 1997 ( 97.300). – Report
- [18] KARLOFF, H.: How good is the Goemans-Williamson MAX CUT Algorithm? In: *SIAM J. on Computing* 29 (1999), Nr. 1, S. 336–350
- [19] KARP, R. M.: Reducibility among Combinatorial Problems. In: MILLER, R. E. (Hrsg.) ; THATCHER, J. W. (Hrsg.): *Complexity of Computer Computations*. New York : Plenum Press, 1972, S. 85–103
- [20] KOECHER, Max: *Lineare Algebra und analytische Geometrie*. 3. Springer, 1992
- [21] LANCASTER, P. ; TISMENETSKEY, M.: *The Theory of Matrices*. 2. Orlando, FL : Academic Press, 1989
- [22] MAHAJAN, Sanjeev ; RAMESH, H.: Derandomizing Semidefinite Programming Based Approximation Algorithms. In: *Proceedings of the 36th Annual Symposium on Foundations of Computer Science* IEEE, 1995, S. 162–169
- [23] NESTEROV, Jurij: *Quality of semidefinite relaxations for nonconvex quadratic optimization*. 1997. – Manuscript

- [24] NESTEROV, Jurij: Semidefinite Relaxation and Nonconvex Quadratic Optimization. In: *Optimization Methods and Software* 9 (1998), S. 141–160
- [25] NESTEROV, Jurij ; NEMIROVSKIJ, Arkadij S.: *Interior-point polynomial algorithms in convex programming*. Soc. for Industrial and Applied Math., 1994
- [26] NOLTEMEIER, Harmut: *Graphentheorie: mit Algorithmen und Anwendungen*. de Gruyter Lehrbuch, 1975
- [27] PÁL, K. F.: The Ground State Energy of the Edwards-Anderson Ising Spin Glass with a Hybrid Genetic Algorithm. In: *Physica A* (1996), S. 223–283
- [28] POLJAK, Svatopluk ; TUZA, Zsolt: Maximum cuts and large bipartite subgraphs. In: COOK, William (Hrsg.) ; LOVASZ, Laszlo (Hrsg.) ; SEYMOUR, Paul (Hrsg.): *DIMACS Series in Discrete Mathematics and Theoretical Computer Science* Bd. 20. American Mathematical Society, 1995, S. 181–244
- [29] RIEGER, H. ; SANTEN, L. ; BLASUM, U. ; DIEHL, M. ; JÜNGER, M. ; RINALDI, G.: The critical exponents of the two-dimensional Ising spin glass revisited: Exact Ground State Calculations and Monte Carlo Simulations / Institut für Informatik, Universität zu Köln. Pohligstraße 1, D-50969 Köln, 1996 ( 96.231). – Report
- [30] SAHNI, S. ; GONZALEZ, T.: P-complete approximation problems. In: *Journal of the Association for Computing Machinery* 23 (1976), Nr. 3, S. 555–565
- [31] VANDENBERGHE, Lieven ; BOYD, Stephen: Semidefinite Programming. In: *SIAM Review* (1995)