

Erik Mucha

Graphenaggregation

Numerische Optimierungsverfahren
zur approximativen Komplexitätsreduktion
in Gasnetzmodellen

DIPLOMARBEIT

HOCHSCHULE MITTWEIDA

UNIVERSITY OF APPLIED SCIENCES

Fakultät für
Mathematik/Naturwissenschaften/Informatik

Dresden, 2011

Erik Mucha

Graphenaggregation

Numerische Optimierungsverfahren
zur approximativen Komplexitätsreduktion
in Gasnetzmodellen

eingereicht als

DIPLOMARBEIT

an der

HOCHSCHULE MITTWEIDA

UNIVERSITY OF APPLIED SCIENCES

Fakultät für
Mathematik/Naturwissenschaften/Informatik

Erstprüfer : Prof. Dr. Peter Tittmann

- Hochschule Mittweida (FH) -

Zweitprüfer : Dr. Armin Fügenschuh und Dr. Benjamin Hiller

- Konrad-Zuse-Zentrum für Informationstechnik Berlin -

Vorgelegte Arbeit wurde verteidigt am:

Bibliografische Beschreibung

Mucha, Erik:

Graphenaggregation - Numerische Optimierungsverfahren zur approximativen Komplexitätsreduktion in Gasnetzmodellen. 62 Seiten. Mittweida, Hochschule Mittweida (FH) - University of Applied Sciences, Fakultät MNI, Diplomarbeit, 2011

Referat

Die Diplomarbeit beschäftigt sich mit aktuell gängigen mathematischen Methoden und Verfahren numerischer Löser zur Bearbeitung großer Optimierungsprobleme der Energieversorgung, welche überwiegend zur Klasse der gemischt-ganzzahligen nichtlinearen Optimierungsprobleme gehören. Es wird ein Beispiel für solch ein Problem modelliert und dessen Ergebnisse ausgewertet.

Die Grundlage liefert ein umfangreiches Projekt des DFG-Forschungszentrums MATHEON, welches sich mit der Neugestaltung des deutschen Gasnetzes bezüglich der zukünftigen Bedarfs- und Rechtslage beschäftigt.

Inhaltsverzeichnis

Abbildungsverzeichnis	IV
Abkürzungsverzeichnis	V
Symbolverzeichnis	VI
1 Einleitung	1
2 Hintergrund des Projektes	3
2.1 Konrad-Zuse-Zentrum für Informationstechnik Berlin	3
2.2 Ferntransport von Erdgas in Deutschland	4
2.3 Verwendete Software	5
3 Modellierung des Gasnetzes	6
3.1 Graphentheoretische Begriffe	6
3.2 Begriffe des Gastransportes	7
3.3 Definitionen und Symbole	8
3.4 Zusammenhänge im Modell	10
3.5 Grundlagen der ganzzahligen Optimierung	11
4 Numerische Optimierungsverfahren	13
4.1 Gauß-Newton-Verfahren	13
4.2 Branch-and-bound	15
4.3 Schnittebenenverfahren	18
4.4 Branch-and-cut	19

4.5	Dualer Simplexalgorithmus	20
4.6	Linearisierung von Nichtlinearitäten	22
4.7	Heuristiken	23
4.8	Programmablauf in SCIP	24
5	Aggregation passiver Teilnetze	25
5.1	Erstellen eines Druckszenarios	27
5.2	Ermittlung der Rohrkonstanten über ein Gleichungssystem .	30
5.3	Ermittlung der Rohrkonstanten über ein Optimierungsproblem	32
5.3.1	Aussagen zur Konvexität	34
5.3.2	Linearisierung der Druckverlustgleichung	36
6	Auswertung der Ergebnisse	40
6.1	Selbstapproximation	40
6.2	Nachträgliche heuristische Schranken	44
6.3	Gütebetrachtung	48
7	Fazit und Ausblick	49
	Literaturverzeichnis	I

Abbildungsverzeichnis

2.1	Karte des deutschen Gasferntransportnetzes	4
3.1	Beispiel für Graphen	7
3.2	Auflistung aktiver und passiver Gasnetzbestandteile	8
3.3	Verschiedene Clustermöglichkeiten	9
4.1	Beispiel für einen Verzweigungsbaum	16
4.2	Ablaufschema von SCIP	24
5.1	Aggregation eines passiven Teilnetzes	26
5.2	Bilanzenszenario und Initialdruck	27
5.3	Druckszenario im Beispielnetz	30
5.4	Neue Rohrkonstanten	31
5.5	Epigraph	37
5.6	Spatial branching	38
6.1	Testinstanzen A, B und C	41
6.2	Testinstanz D	42
6.3	Ergebnisse der Instanzen A,B,C und D	42
6.4	Testinstanz E	46
6.5	aggregiertes Netz aus Instanz E ₁	47
6.6	Ergebnisse der Instanz E	47
7.1	sukzessive Aggregation	50

Abkürzungsverzeichnis

BFS breadth-first search, Breitensuche

CIP constraint integer program

DFS depth-first search, Tiefensuche

GNU GNU's Not Unix

GPL GNU General Public Licence

ILP integer linear program

IP integer program

LP linear program

MINLP mixed integer non-linear program

MIP mixed integer program

NLP non-linear program

SCIP Solving Constraint Integer Programs

ZFW Zielfunktionswert

ZIB Konrad-Zuse-Zentrum für Informationstechnik Berlin

Zimpl Zuse Institute Mathematical Programming Language

Symbolverzeichnis

B_V	Bilanzszenario der Knotenmenge V
C_E	Menge der Rohrkonstanten aller Kanten aus E
E	Kantenmenge
V	Knotenmenge
Z	Cluster, Knotenmenge aus V
\mathcal{S}_V	Indexmenge von Bilanzszenarios der Knotenmenge V
b_A	Gesamtbilanz der Knotenmenge A
b_v	eingespeiste bzw. abgenommene Gasmenge am Knoten v
$c_{v,w}$	Rohrkonstante der Pipe zwischen v und w
$q_{v,w}$	Volumenstrom von Knoten v zu Knoten w
v_Z	Clusterknoten

Kapitel 1

Einleitung

Unsere Gesellschaft ist durchzogen mit tausenden künstlich geschaffenen, logistischen Systemen, welche unsere immerwährenden Bedürfnisse nach Gütern, Energie und Information befriedigen sollen.

Die Fülle an Daten, die von Betreibern solcher Systeme beachtet werden muss, bevor eine richtungsweisende Entscheidung getroffen werden kann, wird von Tag zu Tag größer. Verarbeitende Simulationen und Analysen benötigen ihrerseits Modelle, welche zum Einen die reellen Zustände so präzise wie möglich abbilden und zum Anderen dafür sorgen, dass Berechnungen in einer akzeptablen Zeit zu einem Ergebnis führen. Zu große Modelle führen zu extrem langen Laufzeiten der simulierenden und analysierenden Algorithmen und machen somit eine schnelle Entscheidungsfindung unmöglich. Die effektive Vereinfachung dieser komplexen Modelle ist deshalb für jede entscheidungsorientierte Datenverarbeitung in logistischen Systemen von besonderem Interesse.

Eine Arbeitsgruppe des Konrad-Zuse-Zentrums für Informationstechnik Berlin (ZIB) ist seit längerer Zeit mit der Bearbeitung eines solchen Problems beschäftigt. Sie soll unter anderem ein Verfahren entwickeln, welches dem deutschen Gasmarkt hilft, das innerdeutsche Gasversorgungsnetz dem aktuellen Gasbedarf sowie der aktuellen Rechtslage anzupassen und dementsprechend auszubauen bzw. umzugestalten.

Es existiert im Zuge des Projektes ein Algorithmus, welcher ein *pipe network analysis problem* bearbeitet und bestimmt, ob auf einem umgestalteten Netz unter bestimmten Voraussetzungen überhaupt Gas fließen kann.

Eine Aufgabe dieser Diplomarbeit besteht darin, einen speziellen Teil der modellierten Netzstruktur zu vereinfachen, damit der Algorithmus durch weniger zu verarbeitende Daten schneller zu einer Lösung gelangen kann. Dabei soll der Unterschied zur Originallösung, der nach der Modifizierung des Netzes zu erwarten ist, so gering wie möglich ausfallen. Da der Algorithmus ein Entscheidungsproblem mit nur zwei Antwortmöglichkeiten, 'Ja' oder 'Nein', löst, muss dafür ein Fehler- bzw. Gütemaß gefunden werden. Erst dann kann geklärt werden, inwiefern eine Lösung, die auf dem modifizierten Netz gefunden wurde, im Originalnetz interpretiert werden kann.

Zur Bearbeitung dieser Aufgabe werden die mathematischen Teilgebiete der Graphentheorie sowie der diskreten Optimierung genutzt. Optimierungsprobleme der Energieversorgung und des Transports gehören überwiegend zur Klasse der gemischt-ganzzahligen nichtlinearen Optimierungsprobleme. Diese sind im Allgemeinen schwer zu lösen und für größere Datenmengen ist das Finden einer möglichst globalen Lösung sehr zeitaufwendig. Heutige numerische Löser bedienen sich einer Kombination aus zahlreichen mathematischen Methoden und Verfahren, welche, intelligent miteinander verbunden, relativ schnell globale Lösungen erreichen. In dieser Arbeit wird das Optimierungspaket des Konrad-Zuse-Zentrums für Informationstechnik Berlin zur Bearbeitung des vorher ausführlich modellierten Optimierungsproblems genutzt. Es beinhaltet eine Modellierungssprache sowie einen globalen Löser von beschränkten ganzzahligen Optimierungsproblemen.

Nachdem ein paar Informationen zum Projekt und dem deutschen Gasnetz aufgeführt und die wichtigsten Begriffe und Zusammenhänge erläutert werden, sollen im Hauptteil die derzeit gängigen genutzten numerischen Verfahren zur Lösung von speziell ganzzahligen Optimierungsaufgaben dargestellt werden. Der zweite Teil der Arbeit befasst sich mit einem spezifischen Teil des Problems, der Aggregation passiver Netzbestandteile. Abschließend werden die resultierenden Ergebnisse präsentiert und ausgewertet.

Kapitel 2

Hintergrund des Projektes

2.1 Konrad-Zuse-Zentrum für Informationstechnik Berlin

Das Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB) wurde 1984 als außeruniversitäre Forschungseinrichtung des Landes Berlin gegründet. Zusammen mit den Hochschulen und wissenschaftlichen Einrichtungen Berlins forscht und entwickelt das ZIB auf den Gebieten der anwendungsorientierten Mathematik sowie der praktischen Informatik. Zudem kann das ZIB die dafür benötigte Hochleistungsrechnerkapazität und Datenhaltung anbieten, siehe dazu auch [ZIB10]. Seit einigen Jahren arbeiten auch Studenten der Hochschule Mittweida an verschiedenen Projekten mit, um praktische Erfahrungen auf dem Gebiet Operations Research zu sammeln.

Eine Arbeitsgruppe der Abteilung Optimierung entwickelt seit 2009 im Zuge eines MATHEON¹-Projektes eine Softwareumgebung, welche die technischen Aspekte des Gastransportes sowie die wirtschaftspolitischen Interessen der Gasversorger in der Netzplanung berücksichtigt. Die neuen europaweiten Regelungen erfordern eine komplett neue Problemmodellierung, welche auch auf andere Industrieprojekte übertragen werden kann. Bis heute kam es im Zuge des Projektes bereits zu zahlreichen Publikationen. Für aktuelle Informationen sei auf die Internetseiten [Mat10] verwiesen.

¹Forschungszentrum MATHEON - Mathematik für Schlüsseltechnologien der Deutschen Forschungsgemeinschaft (DFG)

2.2 Ferntransport von Erdgas in Deutschland

Speziell in Deutschland dient der Erdgasferntransport in Hochdruckleitungen zur Abgabe an Weiterverteiler oder nachgelagerte Erdgasversorgungsunternehmen. Auf dem Gebiet der alten Bundesländer ist dafür vorrangig die E.ON Ruhrgas AG mit Hauptsitz in Essen zuständig, in den neuen Bundesländern ist in ähnlicher überragender Marktstellung die Verbundnetz Gas AG in Leipzig zuständig. Siehe dazu auch [HNR07, S. 17].

In Nordwestdeutschland, den Niederlanden und Nordfrankreich werden zwei unterschiedliche Erdgasqualitäten gehandelt: Neben dem international geläufigen H-Gas (high-calorific), welches man beispielsweise in Norwegen, Russland oder Algerien fördert, wird dort auch das aus den Niederlanden stammende L-Gas (low-calorific) gehandelt und transportiert. Das Leitungsnetz der deutschen Gaswirtschaft ist insgesamt rund 375 000 Kilometer lang. Es besteht zu rund 27 Prozent aus Leitungen mit Druckwerten von einem bis zu 100 bar und verzweigt sich bis zu den Niederdruckleitungen mit maximal 100 mbar [EON10].



Abbildung 2.1: Karte des deutschen Gasferntransportnetzes,[EON10]

2.3 Verwendete Software

Inzwischen existiert eine Vielzahl von leistungsstarken Computeralgebrasystemen und numerischen Optimierungsumgebungen. Während der Bearbeitung dieser Diplomarbeit wurde hauptsächlich auf die folgenden, für die Forschungsarbeit frei verfügbaren, Programmpakete zurückgegriffen.

GNU Octave 3.2.4

Bei Octave handelt es sich um eine in den Vereinigten Staaten entwickelte Programmiersprache für numerische Berechnungen. Octave ähnelt in seiner Struktur gewollt der proprietären Numeriksoftware Matlab[®], ist im Gegensatz zu dieser jedoch im Rahmen der GNU General Public Licence (GPL) frei erhältlich. Octave ist in weiten Teilen code-kompatibel zu Matlab.

SCIP 2.0.1

SCIP steht für Solving Constraint Integer Programs und ist derzeit einer der schnellsten nichtkommerziellen Löser für gemischt-ganzzahlige Optimierungsprobleme. Die Umgebung wird am ZIB ständig weiter entwickelt und kann sowohl zur Lösung beschränkter ganzzahliger Optimierungsaufgaben, als auch für *branch-and-cut-and-price*-Verfahren genutzt werden. Die SCIP-Bibliothek kann in C und C++ eingebunden werden und funktioniert mit dem Aufruf von verschiedenen Dateiformaten wie mps, lp und dem Zimpl-Format.

Zimpl 3.1.0

Die Zuse Institute Mathematical Programming Language (Zimpl) ist eine von Thorsten Koch [Koc04] entwickelte Programmiersprache zur Übersetzung eines mathematischen Modells in ein lineares bzw. gemischt-ganzzahliges Optimierungsproblem. Die Umgebung kann Dateien im mps- und lp-Format ausgeben, welche dann von Lösern gelesen und bearbeitet werden können.

Kapitel 3

Modellierung des Gasnetzes

Es erscheint mehr als naheliegend, ein gastransportierendes Netz als Graphenstruktur zu modellieren. Auf solch einer Struktur sind die wesentlichen Eigenschaften, wie Verbindungswege und besondere Regionen des Netzes leicht zu erkennen und zu interpretieren. Zusammen mit den entsprechenden physikalischen Eigenschaften der Graphenbestandteile ergibt sich ein universell einsetzbares Modell. Dazu werden in diesem Kapitel nicht nur die grundlegenden graphentheoretischen und physikalischen Begriffe erklärt, sondern auch spezielle Begriffe, Symbole und Zusammenhänge aufgeführt, die im Laufe der Arbeit verwendet werden.

3.1 Graphentheoretische Begriffe

Die Definitionen dieses Abschnittes sind nach [BL95] formuliert worden.

Definition 3.1. Ein geordnetes Paar $G = (V, E)$ wird *endlicher Graph* genau dann genannt, wenn V eine nichtleere Menge mit $|V| < \infty$ ist und E eine Teilmenge von $\binom{V}{2}$.

Ein geordnetes Paar $D = (V, E)$ heißt *gerichteter Graph* oder auch *Digraph* genau dann, wenn $E \subseteq V \times V - \{(x, x) | x \in V\}$. Die Elemente von E nennt man dann auch *Bögen* oder *gerichtete Kanten*. Siehe Abbildung 3.1.

Im speziellen Fall des Gasnetzes werden die Pipes als Kantenmenge E und die Rohrverzweigungen sowie die Endpunkte als Knotenmenge V inter-



Abbildung 3.1: links: ungerichteter Graph, rechts: gerichteter Graph

pretiert. Da im Gasnetz die meisten Pipes in beide Richtungen durchlaufen werden können, wird es hier als ungerichteter Graph modelliert. Zudem werden die Graphen im Laufe der Arbeit auch einfach als *Netze* bezeichnet.

Definition 3.2. $G' = (V', E')$ heißt *Teilgraph* von $G = (V, E)$ (in Zeichen: $G' \subseteq G$) genau dann, wenn die beiden mengentheoretischen Inklusionen $V' \subseteq V$ und $E' \subseteq E$ gelten. $G' = (V', E')$ heißt *Untergraph* oder *induzierter Teilgraph* genau dann, wenn $G' \subseteq G$ gilt und zudem die Aussage

$$\forall (x, y) \in V' : (x, y) \in E \Rightarrow (x, y) \in E'$$

wahr ist.

Definition 3.3. Der *Knotengrad* $\deg(v)$ eines Knotens v gibt die Anzahl der Knoten an, mit denen er über eine Kante verbunden ist.

3.2 Begriffe des Gastransportes

Da es sich im Folgenden speziell um die Betrachtung eines gastransportierenden Netzes handelt, sollen vorab zwei physikalische Begrifflichkeiten zu Gas und Gastransport genannt und erläutert werden.

Definition 3.4. Unter einem *Volumenstrom* (im Folgenden auch *Fluss*) versteht man das Volumen eines Mediums, das sich innerhalb einer Zeiteinheit durch einen Querschnitt bewegt. Es wird in m^3/s (oder auch m^3/Tag) angegeben. Der Fluss zwischen zwei Knoten v und w wird durch $q_{v,w} \in \mathbb{R}$ dargestellt. Es gilt $q_{w,v} = -q_{v,w}$.






Klasse	Name	Symbol	mögl. Durchlaufrichtung
passiv	Rohr		\longleftrightarrow
	Widerstand		\longleftrightarrow
aktiv	Ventil		\longleftrightarrow
	Kontrollventil		\longrightarrow
	Verdichter		\longrightarrow

Abbildung 3.2: Auflistung aktiver und passiver Gasnetzbestandteile

Definition 3.5. Unter dem *Gasdruck* p_v am Knoten v versteht man die Summe aller durch ein Gas oder Gasgemisch wirkenden Kräfte auf eine Gefäßwand.

Bemerkung. Der Druckunterschied zwischen zwei Pipe-Endknoten führt zum Strömen des Gases. In Pipelines, die große Gasmengen transportieren, kann ein Druck von bis zu 100 bar auftreten. Unmittelbar vor den Verbrauchsgeräten in den Haushalten beträgt der Gasdruck nur noch 20 mbar. Druckangaben betreffen stets die Druckwerte an den Knotenpunkten.

Die Bestandteile des Gasnetzmodells können in zwei Klassen eingeteilt werden: *aktive* und *passive* Bestandteile (Abbildung 3.2). In aktiven Elementen kann zum Beispiel manuell Druck aufgebaut werden, währenddessen in passiven Elementen ausschließlich die physikalischen Gesetze wirken.

3.3 Definitionen und Symbole

Im Folgenden werden die speziellen Begriffe und Symbole definiert, die im Laufe der Arbeit verwendet werden.

Definition 3.6. Sei $G = (V, E)$ mit $V = \{v_1, \dots, v_n\}$ ein ungerichteter Graph. Dann ist ein *Bilanzszenario* $B_V = (b_1, \dots, b_n)$ ein Vektor von Gasmengenwerten, der jedem Knoten $v_i \in V$ mit $i = 1, \dots, n$ eine *Bilanz*

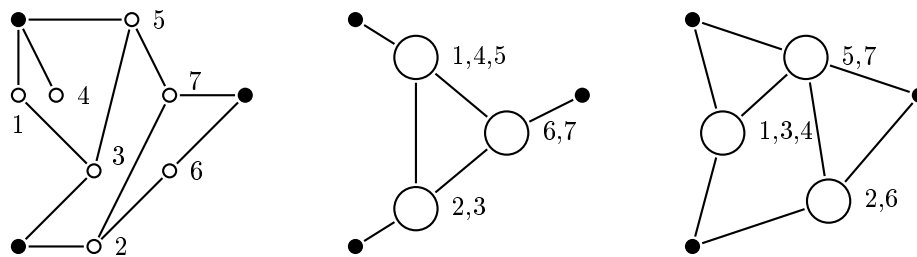


Abbildung 3.3: Verschiedene Clustermöglichkeiten

b_i zuordnet. Falls $b_i < 0$, wird v_i ein Gasbedarf zugeordnet und v_i als *Abnehmer* bezeichnet. Wenn $b_i > 0$, wird v_i ein Gasangebot zugeordnet, v_i heißt *Einspeiser*.

Bemerkung. Im Falle $b_i = 0$ handelt es sich bei v_i um einen *Transitknoten*.

Definition 3.7. Eine *Clusterung* \mathcal{P} eines Graphen $G = (V, E)$ ist eine Partition der Knotenmenge V , wobei jedes Element Z von \mathcal{P} als *Cluster* (von engl. „Traube, Bündel, Schwarm, Haufen“) bezeichnet wird.

Definition 3.8. Kanten, die zwischen zwei Knoten eines Clusters verlaufen, heißen *innere Clusterkanten*, diejenigen, die aus einem Cluster heraus- oder in ihn hineinführen, *äußere Clusterkanten*. Werden alle Knoten einer Menge Z auf einen neuen Knoten reduziert, wird von einem *Clusterknoten* v_Z gesprochen. Siehe dazu Abbildung 3.3.

Definition 3.9. Sei $G = (V, E)$ ein ungerichteter Graph, $Z \subseteq V$ eine Knotenmenge und $B_V = \{b_1, \dots, b_n\}$ ein Bilanzenszenario. Dann definiert sich die *Gesamtbilanz* b_Z als Summe aller Gasmengenwerte der Knoten aus Z :

$$b_Z = \sum_{v \in Z} b_v. \quad (3.1)$$

3.4 Zusammenhänge im Modell

Die *Druckverlustgleichung* von WEYMOUTH aus dem Jahr 1912 wird bei hohem Druck, hohem Volumenstrom und großen Rohrdurchmessern benutzt [UL02, S. 6]. Die Formel berechnet direkt den Volumenstrom durch ein Rohr für gegebene Gas- und Rohrwerte, sowie Anfangs- und Enddruck. In SI-Einheiten¹, wird die Gleichung nach [Men05, S. 61] wie folgt definiert:

$$q_{v,w} = 3.7435 \cdot 10^{-3} \cdot E \cdot \left(\frac{T_{base}}{p_{base}} \right) \sqrt{\frac{p_v^2 - e^s \cdot p_w^2}{g \cdot T_f \cdot L_{v,w} \cdot Z}} \cdot D_{v,w}^{2,667} \quad (3.2)$$

wobei:

$q_{v,w}$ - Gasvolumenstrom in m³/d

E - Pipelineeffizienz, ≤ 1

T_{base} - Basistemperatur in K

p_{base} - Basisdruck in kPa

p_v - Druck am Anfang des Stroms in kPa

p_w - Druck am Ende des Stroms in kPa

e - eulersche Zahl, Basis des natürlichen Logarithmus

s - Parameter zur Höhenverstellung, dimensionslos

T_f - durchschnittliche Gasflusstemperatur in K

g - relative Dichte des Gases ρ/ρ_0 zur Dichte von trockener Luft

$L_{v,w}$ - entsprechende Länge des Pipe-Segments in km

Z - Komprimierbarkeitsfaktor für Gas, dimensionslos

$D_{v,w}$ - Rohrdurchmesser innen in mm

Alle Eigenschaften des Rohres von Knoten v nach Knoten w und des darin beförderten Gases aus (3.2) können mit $s = 0$ zu einer Rohrkonstante $c \in \mathbb{R}^+$ zusammengefasst werden (siehe auch [WS00]):

$$c_{v,w} = \frac{1,4014 \cdot E^2 \cdot T_{base}^2 \cdot D^{5,333}}{10^5 \cdot p_{base}^2 \cdot g \cdot T_f \cdot L_e \cdot Z} \quad (3.3)$$

¹Internationales Einheitensystem (frz. *Système international d'unités*), metrisch, dezimal und kohärent

Die Gleichung (3.2) kann mittels (3.3) zu

$$q_{v,w} \cdot |q_{v,w}| = c_{v,w} \cdot (p_v^2 - p_w^2) \quad (3.4)$$

vereinfacht werden. In dieser Form wird sie in der Diplomarbeit verwendet. Desweiteren gilt in jedem Knoten v die Flusserhaltung, das heißt, die Differenz der hineinführenden und herausführenden Gasmengen muss der Bilanz am Knoten entsprechen. Es kann im Modell an keiner Stelle Gas entstehen oder verloren gehen:

$$\forall v \in V : \sum_{(v,j) \in E} q_{v,j} - \sum_{(i,v) \in E} q_{i,v} = b_v. \quad (3.5)$$

3.5 Grundlagen der ganzzahligen Optimierung

Da im Weiteren von *ganzzahliger Optimierung* die Rede ist, seien an dieser Stelle einige Besonderheiten dieses mathematischen Gegenstandes dargestellt.

Sei $z : R \rightarrow \mathbb{R}$ mit $R \subseteq \mathbb{R}^n$ eine zu minimierende Funktion und R die Menge der zulässigen Lösungen. Gesucht wird ein Element $x^* \in R$, für welches der Zielfunktionswert $z(x^*)$ minimal ist. Die Gemeinsamkeit ganzzahliger (auch: *diskreter*) Optimierungsprobleme ist der Restriktionsbereich R , welcher entweder aus endlich vielen Elementen besteht oder sich aus endlich oder unendlich vielen isolierten Punkten bzw. Teilmengen zusammensetzt [RBD88, S. 13]. Ein ganzzahliges Minimierungsproblem wird wie folgt dargestellt:

$$\begin{aligned} z &= c^T x \rightarrow \min \\ Ax &\leq y \\ x &\in \mathbb{Z}^n \end{aligned} \quad (3.6)$$

Der Restriktionbereich wird somit mit $R = \{x \in \mathbb{Z}^n \mid Ax \leq y\}$ definiert. Nach den englischsprachigen Entsprechungen ist in diesem Fall von einem *integer program* (IP) die Rede. Abhängig von der Linearität der Nebenbedingungen und der Zielfunktion spricht man von *integer linear programs* (ILP) bzw. *integer non-linear program* (INLP). Größere Probleme beinhalten ganzzahlige und nichtganzzahlige Variablen ($x \in \mathbb{Z}^p \times \mathbb{R}^{n-p}$). In diesem Fall wird das Problem als *mixed integer program* (MIP) klassifiziert. Die Modellierung des in dieser Arbeit genauer betrachteten Teilproblems findet sich in Abschnitt 5.3. Zur Komplexität sei an dieser Stelle nur erwähnt, dass es sich bei ganzzahliger Optimierung im Allgemeinen um \mathcal{NP} -*schwere* Probleme handelt (Beweis siehe [KV08, S. 386]).

Kapitel 4

Numerische Optimierungsverfahren

Die heutigen Löser, speziell die in Kapitel 2 genannte Umgebung SCIP, verwenden zur Lösung der gestellten Aufgaben die im Folgenden beschriebenen mathematischen Verfahren. Zur Lösung eines nichtlinearen Gleichungssystems wird das *Gauß-Newton-Verfahren* angewandt, zur Lösung des gemischt-ganzzahligen Optimierungsproblems die *branch-and-cut*-Methode, eine Kombination aus der *branch-and-bound*-Methode und einem *Schnittebenenverfahren*. Zusätzliche Werkzeuge können mehrere Verfahren geschickt miteinander kombinieren, um dadurch effizienter spezielle Aufgabentypen zu verarbeiten.

4.1 Gauß-Newton-Verfahren

Das *Gauß-Newton-Verfahren* [DS96], benannt nach C.F. GAUSS und I. NEWTON, bezeichnet ein numerisches Verfahren, welches die Lösungen eines nichtlinearen Gleichungssystems näherungsweise ermittelt. Es handelt sich genauer um eine Erweiterung des *Newton-Verfahrens*, bei welcher in jedem Schritt die zu minimierende Funktion durch eine quadratische Näherung ersetzt wird. Sei $F(x) = 0$, mit $F : \mathbb{R}^m \rightarrow \mathbb{R}^n$, $m \leq n$ und F differenzierbar, ein zu lösendes homogenes Gleichungssystem:

$$F(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_n(x) \end{bmatrix} = 0.$$

Im *Gauß-Newton-Verfahren* löst man dazu folgendes *Kleinste-Quadrate-Minimierungsproblem*, ausgehend von einer Startlösung x^0 :

$$\| F(x) \|_2^2 = \sum_{i=1}^n (f_i(x))^2 \rightarrow \min .$$

Im sogenannten Gauß-Newton-Schritt wird iterativ ein lineares Gleichungssystem durch

$$F(x^k) + J_F(x^k)(x - x^k) = 0 \quad (4.1)$$

aufgestellt, x ermittelt und für x^{k+1} eingesetzt. $J_F(x)$ ist hierbei die Jacobi-Matrix von F , abhängig vom aktuellen Vektor x :

$$J_F(x) = \begin{pmatrix} \frac{\partial f_1(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_2} & \cdots & \frac{\partial f_1(x)}{\partial x_m} \\ \frac{\partial f_2(x)}{\partial x_1} & \frac{\partial f_2(x)}{\partial x_2} & \cdots & \frac{\partial f_2(x)}{\partial x_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n(x)}{\partial x_1} & \frac{\partial f_n(x)}{\partial x_2} & \cdots & \frac{\partial f_n(x)}{\partial x_m} \end{pmatrix} .$$

Der Vektor x wird solange iterativ verbessert bis der Unterschied zwischen zwei aufeinanderfolgenden Iterationen $|x^{k+1} - x^k|$ unter einer vorher festgelegten Schranke liegt.

Algorithmus:

1. Wähle einen Startlösungsvektor x^0 und setze $k = 0$.
2. Berechne die Jacobi-Matrix $J_F(x^k)$.
3. Berechne die verbesserte Lösung x^{k+1} über die Gauß-Newton-Gleichung (4.1).
4. Falls $|x^{k+1} - x^k| \leq \varepsilon$, ist x^{k+1} Lösung. Sonst erhöhe k um 1 und gehe zu Schritt 2.

Eine Erweiterung des Gauß-Newton-Verfahrens ist der *Levenberg-Marquart-Algorithmus* von K. LEVENBERG und D. MARQUART. Dieser Algorithmus erzwingt in der Gauß-Newton-Gleichung (4.1) absteigende Funktionswerte durch die Zusatzbedingung $\| x - x^k \|_2^2 \leq s^k$ mit einem anzupassenden

Parameter s^k . Der Algorithmus konvergiert mit höherer Wahrscheinlichkeit, wenn auch nicht sicher. Heutige Löser kombinieren das Newton-Verfahren außerdem mit der Newton-Krylov- und der Broyden-Methode [Kel03][DS96].

4.2 Branch-and-bound

Branch-and-bound (von engl. „Verzweigen und Beschränken“) bezeichnet eine exakte Methode, die mittels einer „Teile und Herrsche“-Strategie zur Lösung von ganzzahligen Optimierungsproblemen führt. Unter einer exakten Methode versteht man ein Verfahren, für welches bewiesen werden kann, dass die ermittelte Lösung optimal ist [RBD88, S. 59]. Das *Branch-and-bound*-Verfahren wurde erstmals von A.H. LAND und A.G. DOIG im Jahr 1960 vorgestellt und unmittelbar danach von LITTLE, MURTY, SWENNY und KAREL in Algorithmen zur Lösung des Rundreiseproblems verwendet [DS06, S. 186].

In der folgenden Beschreibung der Methode wird von einem Minimierungsproblem ausgegangen. Da es sich beim allgemeinen *Branch-and-bound*-Verfahren nur um ein Meta-Verfahren handelt, muss der Ablauf für jedes Problem in Form eines speziellen Algorithmus angepasst werden.

Sei D ein gemischt-ganzzahliges Optimierungsproblem

$$\begin{aligned}
 z = c^T x &\rightarrow \min \\
 Ax &\leq y \\
 l &\leq x \leq u \\
 x &\in \mathbb{Z}^p \times \mathbb{R}^{n-p}
 \end{aligned} \tag{4.2}$$

Der Verzweigungsschritt zerlegt den gegebenen Restriktionsbereich $R_D := \{x \in \mathbb{Z}^p \times \mathbb{R}^{n-p} : Ax \leq y, l \leq x \leq u\}$ in k disjunkte Teilmengen R_{1j} , $j = 1, \dots, k$ mit $\bigcup_{j=1}^k R_{1j} = R_D$. Die Lösungen der so entstehenden *Subprobleme* D_{1j} werden dann miteinander verglichen und die beste gewählt. Da die Subprobleme nicht unbedingt einfacher sind als das Ausgangsproblem, muss gegebenenfalls erneut verzweigt werden. Der Schritt kann bis zu

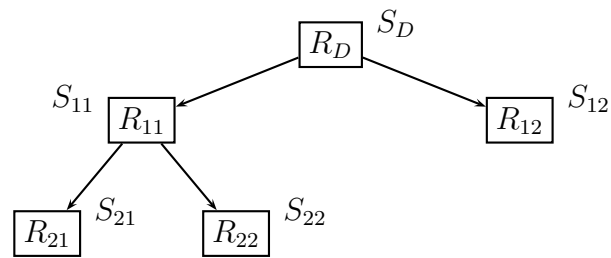


Abbildung 4.1: Beispiel für einen Verzweigungsbaum

einelementigen Teilmengen wiederholt werden, sodass eine Baumstruktur mit R_D als Wurzelknoten und den verkleinerten Restriktionsbereichen R_{ij} als Unterknoten entsteht, siehe dazu Abbildung 4.1.

Die Absicht des Beschränkungsschritts ist es, die komplette Enumeration aller potenziellen Lösungen des Ausgangsproblems zu vermeiden, denn das sind gewöhnlich exponentiell viele [ABKW08, S. 3]. Der Beschränkungsschritt ermittelt, meist mit Hilfe einer *LP-Relaxierung* des Teilproblems, eine untere Schranke S_{ij} des Zielfunktionswertes. Unter einer LP-Relaxierung versteht man die Vereinfachung der Aufgabe durch Nichtbeachten der Ganzzahligkeitsforderung. Obere Schranken f_o ergeben sich aus bereits ermittelten Funktionswerten zulässiger Lösungen. Ein Teilproblem wird genau dann aus dem Entscheidungsbaum gestrichen, wenn dessen untere Schranke S_{ij} größer oder gleich einem bereits ermittelten Funktionswert f_o ist. Durch das Streichen ganzer Teilmengen des Restriktionsbereiches kann das Verfahren auch unter *impliziter* oder *unvollständiger Enumeration* eingeordnet werden [DS06, S. 186].

Das Optimum wird dadurch gefunden, dass jedes Teilproblem R_{ij} hinsichtlich seiner Schranken untersucht und gegebenenfalls erneut zerlegt wird. Die letzte verbleibende einelementige Teilmenge ist demzufolge diejenige mit dem kleinsten Zielfunktionswert f_o .

Algorithmus [FM05, S. 38]:

1. Sei L eine Liste ungelöster Subprobleme von D (Modell 4.2). Setze $U = +\infty$ als obere Schranke für den Zielfunktionswert.
2. Wähle ein ungelöstes Problem D_{ij} aus L und lösche es dort.
3. Berechne die untere Schranke S_{ij} durch das relaxiertes Problem. Ist D_{ij} unlösbar, gehe zu Schritt 2 bis L leer ist. Andernfalls sei \tilde{x}_{ij} eine optimale Lösung und $S_{ij} = c^T \tilde{x}_{ij}$ neue untere Schranke.
4. Ist $\tilde{x}_{ij} \in \mathbb{Z}^n$, wurde eine zulässige optimale Lösung für D gefunden. Falls $S_{ij} < U$, setze $U = S_{ij}$ und lösche alle Probleme D_{ij} mit $U \leq S_{ij}$ aus L .
5. Ist $x_{ij} \notin \mathbb{Z}^n$, teile D_{ij} in Unterprobleme auf und füge sie zu L hinzu.
6. Gehe zurück zu Schritt 2, solange bis L leer ist.

Die Teilung in Unterprobleme, so wie es in Schritt 5 verlangt wird, kann unter anderen Möglichkeiten wie folgt erreicht werden: Es wird eine Variable $\tilde{x}_{ij}^k \notin \mathbb{Z}$ aus der optimalen Lösung \tilde{x}_{ij} eines relaxierten Unterproblems gewählt und zwei neue Unterprobleme mit zwei zusätzlichen Schranken $x^k \leq \lfloor \tilde{x}_{ij}^k \rfloor$ und $\lceil \tilde{x}_{ij}^k \rceil \leq x$ aufgestellt. Aus heuristischen Gründen wird oftmals diejenige Variable gewählt, die am weitesten von einer ganzen Zahl entfernt liegt. [Ach09] betrachtet diese Strategie allerdings kritisch.

Die Schnelligkeit der Lösungsfindung hängt unter anderem davon ab, wie sich der Algorithmus mittels *depth-first-search* (DFS) und *breadth-first-search* (BFS) (auf Deutsch: Tiefensuche und Breitensuche) durch den Entscheidungsbaum bewegt. DFS führt schneller zu einer zulässigen Lösung, BFS kann dafür eher neue Schranken bestimmen. Durch Kombination von DFS und BFS und die Verwendung unterschiedlicher Heuristiken (Abschnitt 4.7) kann die branch-and-bound-Methode effizient gestaltet werden. „Ein effizienter branch-and-bound-Algorithmus ist ein solcher, der das Optimum findet, ohne jeden Knoten des Entscheidungsbaumes einzeln untersuchen zu müssen“ [Fin08, S. 96].

An dieser Stelle seien zwei dieser *informierten* Suchalgorithmen erläutert [FM05, S. 40]:

Best first search: Man wählt das Unterproblem mit der niedrigsten unteren Schranke, um diese zu verbessern. Gelingt das nicht gleich zu Beginn des Entscheidungsbaumes, ist ein großer Speicherbedarf die Folge.

Best projection: Sei S_{ij} die untere Schranke eines Unterproblems, f_o die derzeit beste primale Lösung und $s(D_{ij}) = \sum_{k=1}^n \min\{\tilde{x}_k - \lfloor \tilde{x}_k \rfloor, \lceil \tilde{x}_k \rceil - \tilde{x}_k\}$ die Summe aller Unzulässigkeiten des Subproblems D_{ij} . Dann wählt diese Methode genau den Knoten des Entscheidungsbaumes mit

$$\min_{(D_{ij}) \in L} \left(S_{ij} + \frac{f_o - S_D}{s(D)} \cdot s(D_{ij}) \right)$$

wobei der Bruchterm als Maß für die Veränderung des Zielfunktionswertes pro Verminderung der Unzulässigkeit um eine „Einheit“ zu interpretieren ist.

4.3 Schnittebenenverfahren

Das *Schnittebenenverfahren* gehört, wie die *branch-and-bound*-Methode auch, zu den exakten Methoden der ganzzahligen Optimierung. Den ersten allgemein einsetzbaren Ansatz brachte R. GOMORY im Jahr 1958 zu Papier, eine allgemeine Formulierung und Begründung erfolgte 1971 durch V.A. EMELIČEV [DS06, S. 197]. Die Grundidee des Verfahrens beruht auf der Verschärfung der einfacher zu lösenden relaxierten Optimierungsaufgabe bis diese eine ganzzahlige Lösung besitzt.

Sei D eine gemischt-ganzzahlige lineare Optimierungsaufgabe gemäß (4.2). Der Restriktionsbereich $R_D := \{x \in \mathbb{Z}^p \times \mathbb{R}^{n-p} : Ax \leq y, l \leq x \leq u\}$ führt durch Ignorieren der Ganzzahligkeitsforderung zu folgender LP-Relaxierung LP_D^0 :

$$\begin{aligned} z = c^T x &\rightarrow \min \\ Ax &\leq y \\ l &\leq x \leq u \\ x &\in \mathbb{R}^n \end{aligned} \tag{4.3}$$

Das relaxierte Problem LP_D^0 wird standardmäßig mittels *Simplexmethode* (siehe Abschnitt 4.5) gelöst. Ist der Restriktionsbereich der Relaxierung $R_{LP_D^0}$ leer oder wächst die Zielfunktion über $R_{LP_D^0}$ unbeschränkt, so ist auch D nicht lösbar bzw. unbeschränkt.

Die Intention des Schnittebenenverfahrens ist es, LP_D^0 derart zu beschränken, dass eine Optimallösung die Ganzzahligkeitsforderung in D erfüllt. Geometrisch sind Schnittebenen als Hyperebenen des \mathbb{R}^n anzusehen, welche einen Teil des Restriktionsbereiches und somit einen Teil der zulässigen, nichtganzzahligen Lösungen abschneiden. Sie werden als Ungleichungen der Form $\alpha x \leq \beta$ dem Problem LP_D^0 hinzugefügt.

Algorithmus [FM05, S. 13]:

1. Sei $k = 0$ und LP_D^0 die LP-Relaxierung von D .
2. Löse LP_D^0 mittels Simplexalgorithmus. Sei \tilde{x}^k dessen optimale Lösung.
3. Liegt \tilde{x}^k in R_D ist es eine optimale Lösung von D .
Andernfalls ist eine Ungleichung aufzustellen, welche durch alle zulässigen Lösungen erfüllt wird, nur nicht von \tilde{x}^k .
4. Füge diese Ungleichung zu LP_D^k hinzu.
5. Erhöhe k um 1 und gehe zurück zu Schritt 2.

Beispiele für Schnitt-Methoden sind der *Gomory-Schnitt* und die *Knapsack-Ungleichungen* (siehe [FM05, S. 13] und [DS06, S. 201]). Schnittebenenverfahren sind generell mit Vorsicht zu verwenden, da sie oft zu schwach sind, die LP-Relaxierung merklich zu verbessern. Sie sind teilweise sehr zeitaufwändig und können zu numerischen Problemen führen. Für spezielle Optimierungsprobleme, wie das Max-Flow-Problem oder das Kürzeste-Wege-Problem existieren effektivere, kombinatorische Verfahren [FVHS10].

4.4 Branch-and-cut

In den 1980er Jahren kombinierten M. GRÖTSCHEL und M. PADBERG bei der Untersuchung des Problems des Handlungsreisenden das *branch-and-*

bound- mit dem *Schnittebenenverfahren* zum sogenannten *branch-and-cut-Verfahren*. Inzwischen ist das Verfahren zum generischen Standardwerkzeug der ganzzahligen linearen Optimierung geworden.

In branch-and-cut-Verfahren werden die LP-Relaxierungen im Verzweigungsbaum nachträglich durch eine oder mehrere Schnittebenen verbessert und erst dann neu verzweigt. Durch ein geschicktes Zusammenspiel zwischen Verzweigungs- und Schnittvorgängen sind heutige Löser in der Lage, Probleme mit einigen tausend Variablen und Nebenbedingungen zu lösen [FVHS10, S. 3].

Algorithmus [FM05, S. 39]:

1. Sei L eine Liste von ungelösten Teilproblemen. Setze $U := +\infty$ als obere Schranke.
2. Wähle ein ungelöstes Problem D_{ij} aus L und lösche es aus L .
3. Berechne die untere Schranke S_{ij} durch das relaxierte Problem. Ist D_{ij} unlösbar, gehe zu Schritt 2 bis L leer ist. Andernfalls sei \tilde{x}_{ij} eine optimale Lösung von D_{ij} und $S_{ij} := c^T \tilde{x}_{ij}$ neue untere Schranke.
4. Ist $x_{ij} \in Z^n$, wurde eine zulässige optimale Lösung für D gefunden. Falls $S_{ij} < U$, setze $U := S_{ij}$ und lösche alle Probleme D_{ij} mit $U \leq S_{ij}$ aus L .
5. Ist $x_{ij} \notin Z^n$, suche nach Schnittebenen und füge sie der Relaxierung hinzu.
6. Geh zu Schritt 3, bis keine verletzten Ungleichungen mehr gefunden werden oder verletzte Ungleichungen die untere Schranke nicht merklich verbessern können.
7. Teile D_{ij} in Unterprobleme auf und füge sie zu L hinzu.
8. Zurück zu 2., bis L leer ist.

4.5 Dualer Simplexalgorithmus

Das Simplexverfahren wurde 1947 von G. DANTZIG entwickelt [HK00, S. 11]. Es liefert eine Optimallösung, sofern eine existiert, für beliebige LPs.

Es werden mittels eines gezielten Austauschs der Basisvariablen solange die Ecken des Restriktionsbereiches abgesucht, bis der Zielfunktionswert nicht weiter verbessert werden kann. Gewöhnlich wird in einem *branch-and-cut*-Algorithmus zur Lösung der linearen Optimierungsaufgaben die *duale Simplexmethode* angewendet. Diese hat gegenüber der *primale Simplexmethode* den Vorteil, dass eine Lösung auch dann dual zulässig bleibt, wenn Schnittebenen hinzugefügt werden. Sei L eine zu lösende lineare Optimierungsaufgabe:

$$\begin{aligned} z = c^T x &\rightarrow \min \\ Ax &= y \\ x &\in \mathbb{R}_+^n \end{aligned} \tag{4.4}$$

Der Algorithmus basiert auf dem *Simplex-Tableau* der Form:

$$T(B) = \begin{array}{|c|c|} \hline \bar{c} & z \\ \hline A_B^{-1} \cdot A & y \\ \hline \end{array}$$

mit $B = (B(1), \dots, B(m))$ als Indexmenge der m Basisvariablen, A_B als die zu B gehörende reguläre $m \times m$ Teilmatrix von A und $\bar{c} = c - c_B A_B^{-1} A$ als die sogenannten *reduzierten Kosten*.

Algorithmus [HK00, S. 69]:

1. Stelle ein dual zulässiges ($\bar{c} \geq 0$) Tableau $T(B)$ auf.
2. Existiert kein $y_i < 0$, ist die Lösung optimal. Andernfalls bestimme k mit $y_k = \min_i y_i$
3. Wenn kein $a_{kj} < 0$ existiert, gibt es auch keine optimale Lösung. Andernfalls bestimme l mit $q_l = \max_{a_{kj} < 0} \frac{\bar{c}_j}{a_{kj}}$
4. Erzeuge das neue Tableau $T(B')$ indem die l -te Spalte durch elementare Zeilenoperationen zum Einheitsvektor mit $a_{kl} = 1$ transformiert wird.
5. Zurück zu Schritt 2.

Eine dual zulässige Lösung ist genau dann optimal, wenn sie auch primal zulässig ist. Der Nachteil des dualen Simplexalgorithmus ist die Deutung

einer nicht optimalen Lösung. Sie kann durch ihre primale Unzulässigkeit nicht als Näherung für die Optimallösung angenommen werden, sondern nur deren Zielfunktionswert als eine untere Schranke.

4.6 Linearisierung von Nichtlinearitäten

Optimierungsprobleme aus den Bereichen Transport- und Energieversorgungsnetzwerke führen meist auf gemischt-ganzzahlige nichtlineare Optimierungsprobleme (MINLP), da sie ganzzahlige Entscheidungsparameter und nichtlineare Beschränkungen beinhalten, welche aus physikalischen Zusammenhängen entstehen. Die Physik von Gas in Pipelines wird am besten durch Gleichungssysteme partieller Differentialgleichungen modelliert und selbst die immernoch sinnvollen Näherungsformeln sind nichtlineare Gleichungen [FVHS10, S. 1].

MINLP sind in der Regel Probleme, die für größere Instanzen nicht mehr in annehmbarer Zeit lösbar sind. Allerdings existieren heutzutage sehr effektive Löser für gemischt-ganzzahlige lineare Optimierungsprobleme (MILP), basierend auf der *branch-and-cut*-Methode (Abschnitt 4.4). Deshalb ist man daran interessiert, die Nichtlinearitäten durch lineare Zusammenhänge anzunähern.

Sei $f(x) \leq 0$ mit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ eine nichtlineare Nebenbedingung. Um eine LP-Relaxierung zu erstellen, ersetzt man f durch eine lineare Näherung

$$Lx \leq l$$

$$\text{wobei } L \in \mathbb{Q}^{q \times n}, l \in \mathbb{Q}^q \text{ und } \{x : f(x) \leq 0\} \subseteq \{x : Lx \leq l\}$$

Nach der Lösung dieser Relaxierung können zur lokalen Verbesserung Schnittebenen eingefügt werden, die den Lösungsvektor x^* der Relaxierung von der konvexen Hülle der Menge $\{x : f(x) \leq 0\}$ abtrennen. Dies ist nicht möglich, wenn die Lösung selbst nicht in $\{x : f(x) \leq 0\}$ liegt, sondern nur in deren konvexer Hülle. Dann kann die *spatial-branching*-Methode genutzt werden, welche die stetige Variable x auf die Unterprobleme mit einerseits $\{x : f(x) \leq 0, x \leq x^*\}$ und andererseits $\{x : f(x) \leq 0, x \geq x^*\}$

verzweigt und anschließend auf bessere Lösungen untersucht [FVHS10, S. 4].

4.7 Heuristiken

Heuristiken sollen ihrer Bezeichnung nach mit möglichst geringem Zeit- und Rechenaufwand zu guten, zulässigen Lösungen führen. Zur Findung einer ersten zulässigen Lösung von Optimierungsaufgaben bedient man sich zum Beispiel heuristischer Eröffnungsverfahren. Zum Teil liefern schon allein die Heuristiken akzeptable Lösungen. Heutige MIP-Löser verwenden Heuristiken auch gezielt während der Verzweigungsschritte des *branch-and-bound*-Verfahrens. Für eine genaue Beschreibung der über 20 in SCIP implementierten Heuristiken sei an dieser Stelle auf [Ber08] verwiesen. Sie können in vier Klassen eingeordnet werden:

Rundungs-Heuristiken: Heuristiken dieses Typs runden iterativ die Optimallösung einer LP-Relaxierung eines IP-Problems auf ganze Zahlen und versuchen dabei die Zulässigkeit der Lösung nicht zu verletzen.

Diving-Heuristiken: Dieser Typ löst wiederholend die LP-Relaxierung mit neu gerundeten Werten und simuliert so einen branch-and-bound-artigen DFS-Algorithmus, welcher letztlich auf eine zulässige Lösung führen kann.

Objective-Diving-Heuristiken: Ähnlich den Diving-Heuristiken, aber anstatt die Grenzen der Variablen festzulegen, werden hier die Zielfunktionskoeffizienten verändert.

Verbesserungsheuristiken: Diese Heuristiken berücksichtigen bereits gefundene primal-zulässige Lösungen und konstruieren mit diesen eine neue Lösung mit verbessertem Zielfunktionswert.

Heuristiken können im Allgemeinen keine zulässige Lösung garantieren. Sie scheitern öfter, als dass sie zum Erfolg führen.

4.8 Programmablauf in SCIP

Ein Beispiel für die Kombination der verschiedenen numerischen Optimierungsverfahren liefert der CIP-Löser SCIP des Zuse-Instituts. SCIP basiert auf einem *branch-and-bound*-Algorithmus. Dessen Ablaufschema, sei in Abbildung 4.2 (angelehnt an [BHV09, S. 2]) aufgeführt. Der Presolver versucht, vor dem Eintritt in den *branch-and-bound*-Algorithmus, das Problem zu vereinfachen, indem er Nebenbedingungen umformuliert. So versucht er vor allem, Binärvariablen zu substituieren.

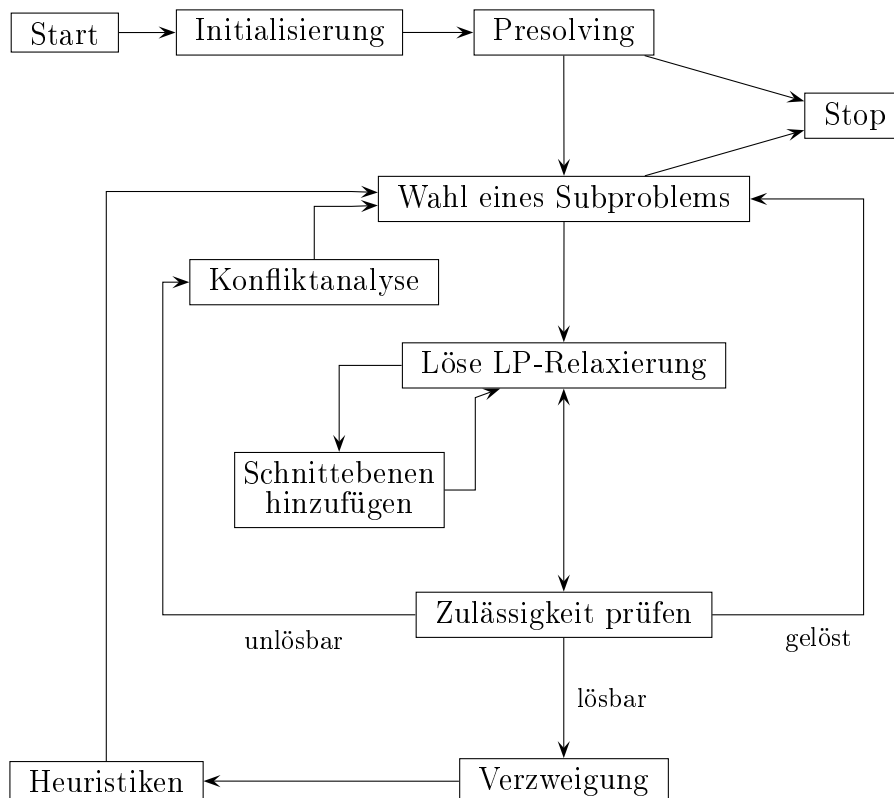


Abbildung 4.2: Ablaufschema der Hauptschleife in SCIP

Kapitel 5

Aggregation passiver Teilnetze

Einen großen Anteil des Gasnetzes nehmen *passive Teilnetze* ein. Dabei handelt es sich um die Gebiete des Netzes, in denen nur die Druckverlust-, sowie die Flusserhaltungsgleichung gelten und kein aktives Steuern an Netzbestandteilen möglich ist.

Es soll die Menge der Knoten solch eines passiven Teilnetzes V_P auf eine kleinere Menge \tilde{V}_P reduziert werden, indem günstige Vereinigungen (*Cluster*, Definition 3.7) gefunden und durch neue Knoten ersetzt werden (Abbildung 5.1). Das Teilnetz wird begrenzt durch eine Menge von definierten Randknoten V_R . Diese werden nach der Reduzierung mit bestimmten Clusterknoten verbunden. Es sei ein Netz $G = (V_P \cup V_R, E)$ welches auf ein aggregiertes Netz $\tilde{G} = (\tilde{V}_P \cup V_R, \tilde{E})$ reduziert wird. Jeder Randknoten besitzt im aggregierten Netz genau dann eine Kante zu einem Clusterknoten v_Z , wenn er im Originalnetz eine Kante zu einem Knoten aus Cluster Z besitzt:

$$\exists(v, w) \in (V_R \times Z) \cap E \iff \exists(v, v_Z) \in (V_R \times \tilde{V}_P) \cap \tilde{E}.$$

Die Beschaffenheit der gegebenen Verbindungen im Originalnetz G , sowie im aggregierten Netz \tilde{G} , werden durch die Rohrkonstanten aus Gleichung (3.3) charakterisiert. Es wird angestrebt, die neuen Verbindungen mittels der Konstanten so zu beschreiben, dass \tilde{G} in möglichst vielen möglichen Szenarien genauso reagiert wie G . Da Kanten entfernt werden, verschwinden auch mögliche Flussrichtungsalternativen und Rohreigenschaften. Verschiedene Clusterungen führen auf verschiedenen hohen Informationsverlust.

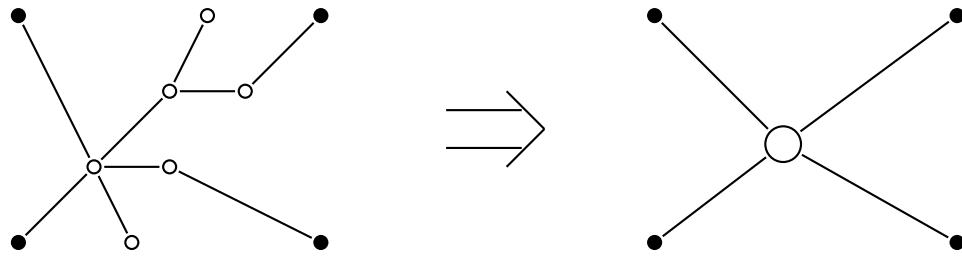
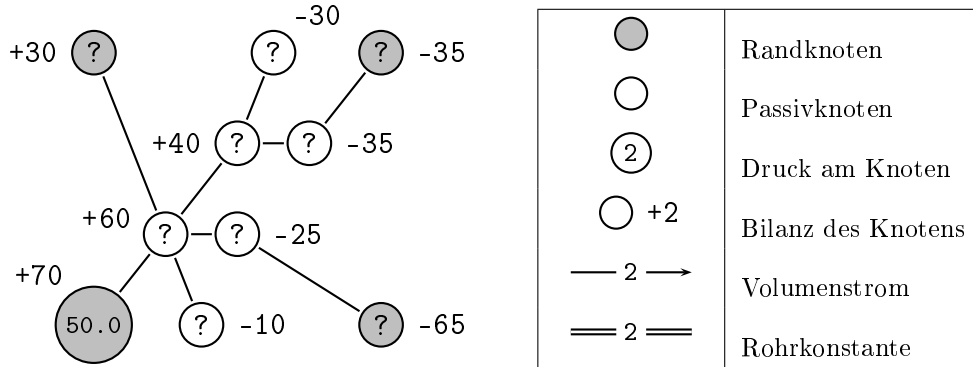


Abbildung 5.1: links: Originalnetz, rechts: aggregiertes Netz

In diesem Kapitel soll nun eine mögliche Vorgehensweise zur Lösung des Problems Schritt für Schritt ausgeführt werden. Zuerst muss ein Testnetz $G = (V, E)$ konstruiert werden und mit ihm die Menge von Rohrkonstanten C_E . Zusätzlich muss ein oder mehrere Bilanzszenarien B_V definiert werden. Das Netz wird damit beinahe vollständig charakterisiert. Es werden zusätzlich noch die zulässigen Druck- und Flussintervalle definiert, entweder einheitlich für ganz G oder für jeden Knoten bzw. jede Kante einzeln. Für die Konstruktion eines zufälligen Netzes mit gewünschten Eigenschaften wurde im Zuge dieser Arbeit ein Netzgenerator in der Programmiersprache C implementiert.

Eine Netzgröße, die sich zum Vergleich von G und \tilde{G} anbietet, ist der Druck p an den Randknoten aus V_R . Es soll erreicht werden, dass die Werte sich in einem beliebigen Szenario so wenig wie möglich voneinander unterscheiden. Dazu wird zu Beginn ein Druckszenario für G erstellt und G anschließend zu einem beliebigen Netz \tilde{G} aggregiert. Ein Optimierungsproblem soll dann zur Ermittlung der Rohrkonstanten c dienen, für die die zu erwartenden Druckunterschiede minimiert werden:

$$z = \sum_{i \in V_R} (p_i - \tilde{p}_i)^2 \longrightarrow \min. \quad (5.1)$$



(a) Beispielnetz

(b) Legende für Netzbilder

Abbildung 5.2: Bilanzenszenario und Initialdruck im Beispielnetz

5.1 Erstellen eines Druckszenarios

Um Berechnungen auf dem Originalnetz ausführen zu können, wird ein Druckszenario benötigt, welches Aussage darüber gibt, welche Druckwerte an den Netzknoten herrschen. Ausgangspunkt sei ein Netz $G = (V_P \cup V_R, E)$ mit einer Menge von passiven Knoten V_P , einer Menge von Randknoten V_R und einer Menge von Kanten $E \subseteq (V_R \times V_P \cup V_P \times V_P)$. Jeder Knoten ist mittels eines Bilanzenszenarios $B_V = (b_1, \dots, b_n), V = V_P \cup V_R$ als Einspeiser, Abnehmer oder Transitknoten definiert. Die Verbindungsstränge sind durch die gegebenen Rohrkonstanten $c_{i,j} \in C_E$ beschrieben.

In diesem Abschnitt sei folgender Beispielgraph G_B gegeben:

$$\begin{aligned}
 V_R &= \{1, 2, 3, 4\} \\
 V_P &= \{5, 6, 7, 8, 9, 10\} \\
 B_V &= (70, 30, -65, -35, 60, -10, -25, 40, -30, -35) \\
 E &= \{(1, 5), (2, 5), (3, 7), (4, 10), (5, 6), (5, 7), (5, 8), (8, 9), (8, 10)\} \\
 c_{i,j} &= 10 \quad \forall (i, j) \in E
 \end{aligned} \tag{5.2}$$

Es sollen im Folgenden alle Flusswerte sowie alle Drücke im Teilnetz über ein homogenes Gleichungssystem ermittelt werden. Die Gleichungen zur Flusserhaltung nach (3.5) für alle Knoten v aus V werden mit

$$0 = \sum_{(i,v) \in E} q_{i,v} - \sum_{(v,j) \in E} q_{v,j} + b_v \quad (5.3)$$

und die Druckverlustgleichungen nach (3.4) $\forall (v,w) \in E$ mit

$$0 = c_{v,w} \cdot (p_v^2 - p_w^2) - |q_{v,w}| \cdot q_{v,w} \quad (5.4)$$

beschrieben. Das Gleichungssystem ist durch die lineare Abhängigkeit der Flusserhaltungsgleichungen nicht eindeutig lösbar. Deshalb wird zur Ermittlung einer brauchbaren Lösung der Druck an einem der Randknoten, dieser soll v_0 heißen, festgesetzt. Im Folgenden wird er als Initialdruck p_0 bezeichnet, siehe Abbildung 5.2. Es gilt demnach

$$0 = p_{v_0} - p_0. \quad (5.5)$$

Mit dieser zusätzlichen Gleichung ist das Gleichungssystem genau dann nicht lösbar, wenn p_0 unter einer gewissen Schranke liegt, das heißt, wenn er nicht genügt, um alle Teile des Netzes bilanzengemäß zu versorgen. Außerdem sei für jeden Druckwert die Nichtnegativität gefordert:

$$p_v \geq 0 \quad \forall v \in V = V_R \cup V_P. \quad (5.6)$$

Das Gleichungssystem soll mittels *Gauß-Newton-Verfahren* (Abschnitt: 4.1) in der `fsolve`-Routine von Octave gelöst werden. Dazu wird eine oberere Dreiecksmatrix X als Datenstruktur genutzt, deren Einträge folgendermaßen besetzt seien:

$$\begin{aligned} x(i,i) &\geq 0 && \text{Druck am Knoten } i, \forall i \in V \\ x(i,j), &&& \text{Fluss von Knoten } i \text{ zu Knoten } j, \forall (i,j) \in E \end{aligned}$$

Damit bekommen die Gleichungen (5.3) die Form

$$0 = x(1, k) + \dots + x(k-1, k) - x(k, k+1) \dots - x(k, n) + b(k)$$

und (5.4) kann mit X so formuliert werden:

$$0 = c(i, j) \cdot (x(i, i)^2 - x(j, j)^2) - |x(i, j)| \cdot x(i, j).$$

Für das Beispielnetz (5.2) ergibt sich folgendes Gleichungssystem:

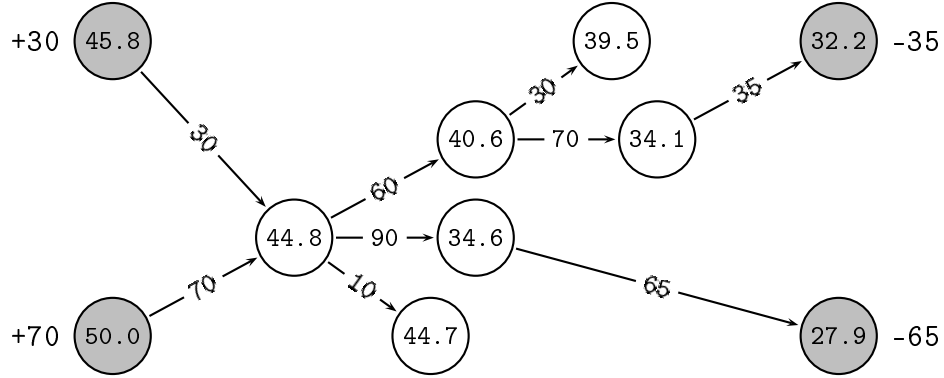
$$\begin{aligned} 0 &= x(1, 5) + 70 \\ &\vdots \\ 0 &= x(1, 5) + x(2, 5) - x(5, 6) - x(5, 7) - x(5, 8) + 60 \\ &\vdots \\ 0 &= x(4, 10) + x(8, 10) - 35 \\ 0 &= 10 \cdot (x(1, 1)^2 - x(5, 5)^2) - |x(1, 5)| \cdot x(1, 5) \\ &\vdots \\ 0 &= 10 \cdot (x(8, 8)^2 - x(10, 10)^2) - |x(8, 10)| \cdot x(8, 10) \\ 0 &= x(1, 1) - 50 \\ 0 &\leq x(i, i) \quad \forall i \in V. \end{aligned} \tag{5.7}$$

Die Einträge der Startlösungsmatrix X_0 werden mit $x(i, i) = p_0$ und $x(i, j) = b_j$, $i \leq j$ belegt. Werden die Werte des Beispielgraphen verwendet, berechnet der Löser folgende Werte für Flüsse und Drücke (siehe dazu Abbildung 5.3):

$$P_{V_R} = (50, 45.8, 27.9, 32.2)$$

$$P_{V_P} = (44.8, 44.7, 34.6, 40.6, 39.5, 34.1)$$

$$Q_E = (70, 30, -65, -35, 10, 90, 60, 30, 70).$$

Abbildung 5.3: Druckszenario im Beispielnetz mit $c_{i,j} = 10, \forall(i,j) \in E$

5.2 Ermittlung der Rohrkonstanten über ein Gleichungssystem

So wie in Abschnitt 5.1 sollen die gesuchten Größen über ein homogenes Gleichungssystem ermittelt werden. Jetzt sind die Druckwerte \tilde{p}_v an den Randknoten aus V_R bekannt und die Rohrkonstanten \tilde{c}_{v,v_Z} des modifizierten Netzes gesucht. Dieser Ansatz geht von nur einem Clusterknoten v_Z aus. Dadurch kann gesagt werden, dass alle neuen Verbindungen ein und denselben Druck an einem ihrer Enden aufweisen müssen:

$$\tilde{p}_{v_Z} = \tilde{p}_{v_Z}$$

$$\tilde{p}_i^2 - \frac{|\tilde{q}_{i,v_Z}| \cdot \tilde{q}_{i,v_Z}}{\tilde{c}_{i,v_Z}} = \tilde{p}_j^2 - \frac{|\tilde{q}_{j,v_Z}| \cdot \tilde{q}_{j,v_Z}}{\tilde{c}_{j,v_Z}} \quad \forall i, j \in V_R. \quad (5.8)$$

Diese Eigenschaft gilt für alle neuen Verbindungen zum Clusterknoten v_Z . Im Folgenden wird das homogene Gleichungssystem formuliert. Sei v_0 ein fixer Randknoten. Der Flusswert q_{k,v_Z} , $k \in V_R$ kann durch die Bilanz b_k ersetzt werden, da nur die eine Verbindung $(k, v_Z) \in \tilde{E}$ genutzt werden kann, um die Bilanz zu erfüllen. Für jedes k mit $(k, v_Z) \in \tilde{E}$ kann damit

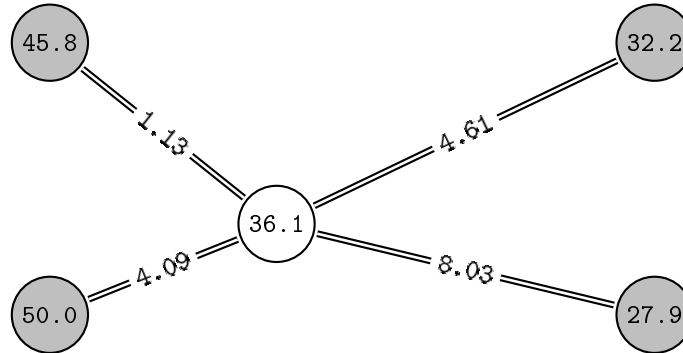


Abbildung 5.4: Neue Rohrkonstanten an den Pipes

folgende Gleichung aufgestellt werden:

$$0 = \left(p_{v_0}^2 - \frac{|b_{v_0}| \cdot b_{v_0}}{c_{v_0, v_Z}} \right) - \left(p_k^2 - \frac{|b_k| \cdot b_k}{c_{k, v_Z}} \right). \quad (5.9)$$

Auch hier soll die `fsolve`-Routine von Octave genutzt werden. Dazu sei der Lösungsvektor X mit den gesuchten Rohrkonstanten $x(i)$ und den Bilanzwerten $b(i)$, $(i, v_Z) \in \tilde{E}$. Die Gleichungen sehen dann $\forall k \in V_R$ wie folgt aus:

$$0 = p(v_0)^2 - p_k^2 - |b(v_0)| \cdot b(v_0)/x(v_0) + |b(k)| \cdot b(k)/x(k). \quad (5.10)$$

Der Ergebnisvektor X beinhaltet die Rohrkonstanten, die für ein definiertes Bilanzenszenario B_V und ein aufgestelltes Druckszenario angenommen werden müssen, um einen gemeinsamen Druck am Clusterknoten zu erreichen. Die Lösungen für das Beispielnetz (5.2) sind in Abbildung 5.4 zu finden.

Jedes Druck- bzw. Bilanzenszenario kann auf andere Rohrkonstanten führen. Möchte man mehrere Szenarien zur einer Gesamtlösung zusammenfassen, müssten die einzelnen Lösungen im Weiteren noch gemittelt werden. Zudem benötigt es einer Erweiterung für mehrere Clusterknoten. Doch der Ansatz über das Lösen von Gleichungssystemen hat einen entscheidenden Nachteil: die Nichtlösbarkeit eines ungünstigen Bilanzenszenarios.

Das Gleichungssystem ist immer dann nicht lösbar, wenn der Idealfall eines gültigen gemeinsamen Druckes am Clusterknoten nicht existiert.

5.3 Ermittlung der Rohrkonstanten über ein Optimierungsproblem

Das Problem als Optimierungsproblem darzustellen, verhindert den Nachteil des Ansatzes aus Abschnitt 5.2, da man eine mögliche Abweichung von der exakten Lösung zulässt. Im betrachteten Fall ist diese Abweichung der Unterschied zwischen den Druckwerten an den Randknoten V_R im Originalnetz $G = (V_P \cup V_R, E)$ und derer im modifizierten Netz $\tilde{G} = (\tilde{V}_P \cup V_R, \tilde{E})$. Seien zur kürzeren Schreibweise $V = V_P \cup V_R$ und $\tilde{V} = \tilde{V}_P \cup V_R$.

Es werden genau die Rohrkonstanten $\tilde{c}_{i,j}$ gesucht, für die die Summe der Quadrate der Druckdifferenzen über alle Terminalknoten und einer Indexmenge von möglichen Bilanzszenarien \mathcal{S}_V minimal wird. Zudem kann durch diesen Ansatz auch eine Lösung für eine beliebige Menge von Clusterknoten ermittelt werden.

Das Optimierungsproblem W sei wie folgt beschrieben:

$$\begin{aligned}
 z &= \sum_{s \in \mathcal{S}_V} \sum_{i \in V_R} (p_i^s - \tilde{p}_i^s)^2 \longrightarrow \min \\
 \sum_{(v,i) \in E} \tilde{q}_{v,i}^s - \sum_{(i,w) \in E} \tilde{q}_{i,w}^s - b_i^s &= 0 \quad i \in \tilde{V}, s \in \mathcal{S}_V \\
 \tilde{q}_{i,j}^s \cdot |\tilde{q}_{i,j}^s| - \tilde{c}_{i,j} \cdot \tilde{p}_i^{s2} + \tilde{c}_{i,j} \cdot \tilde{p}_j^{s2} &= 0 \quad (i,j) \in \tilde{E}, s \in \mathcal{S}_V \\
 0 &\leq \tilde{p}_i^s \leq p_{\max} \\
 0 &\leq \tilde{c}_{i,j} \leq c_{\max} \\
 -q_{\max} &\leq \tilde{q}_{i,j}^s \leq q_{\max}
 \end{aligned} \tag{5.11}$$

Die Nebenbedingungen in Gleichungsform entstehen aus den Zusammenhängen (3.2) und (3.5). Die Variablen \tilde{p}_i , $\tilde{q}_{i,j}$ und $\tilde{c}_{i,j}$ werden nach oben hin durch einen (technisch möglichen) Maximalwert beschränkt.

Um auch die Flussvariable $\tilde{q}_{v,w}$ nichtnegativ in das Modell eingehen lassen zu können und damit $|q_{v,w}|$ zu linearisieren, wird eine binäre Variable $\tau_{v,w}$ eingeführt, die Aussage darüber gibt, in welche Richtung der Fluss auf der Verbindung (v,w) fließt. Der Wert 1 steht für die Flussrichtung von v nach w , der Wert 0 für die Gegenrichtung:

$$\tau_{v,w} = \begin{cases} 0, & \text{falls } q_{v,w} \leq 0 \\ 1, & \text{falls } q_{v,w} \geq 0 \end{cases}. \quad (5.12)$$

Unter Nutzung von $\tau_{v,w}$ ergibt sich aus (5.11) das folgende Modell W' :

$$\begin{aligned} z &= \sum_{s \in \mathcal{S}_V} \sum_{i \in V_R} (p_i^s - \tilde{p}_i^s)^2 \longrightarrow \min \\ \sum_{(v,i) \in E} (2 \cdot \tau_{v,i}^s - 1) \cdot \tilde{q}_{v,i}^s - \sum_{(i,w) \in E} (2 \cdot \tau_{i,w}^s - 1) \cdot \tilde{q}_{i,w}^s - b_i^s &= 0 \quad i \in \tilde{V}, s \in \mathcal{S}_V \\ (2 \cdot \tau_{i,j}^s - 1) \cdot (\tilde{q}_{i,j}^s)^2 - \tilde{c}_{i,j} \cdot \tilde{p}_i^{s2} + \tilde{c}_{i,j} \cdot \tilde{p}_j^{s2} &= 0 \quad (i,j) \in \tilde{E}, s \in \mathcal{S}_V \\ 0 &\leq \tilde{p}_i^s \leq p_{\max} \\ 0 &\leq \tilde{q}_{i,j}^s \leq q_{\max} \\ 0 &\leq \tilde{c}_{i,j} \leq c_{\max} \\ \tau_{i,j}^s &\in \{0, 1\} \end{aligned} \quad (5.13)$$

Durch die Binärvariable $\tau_{v,w} \in \{0, 1\}$ und die stetigen Variablen $\tilde{q}_{i,j}$, $\tilde{c}_{i,j}$ sowie \tilde{p}_i , welche zudem quadratisch auftritt, gehört dieses Modell zur Klasse der *mixed integer non-linear programs* (MINLP).

Ein Spezialfall ist die Zusammenfassung aller Passivknoten zu einem einzigen Clusterknoten. Denn dann nimmt der aggregierte Graph eine Sternstruktur an. Der in der Mitte liegende Clusterknoten besitzt also einen Knotengrad entsprechend der Anzahl aller Randknoten.

Diese Struktur führt auf eine andere Klassifizierung des Modells. Die bilanzengerechte Verteilung des Gases ist nur durch eine einzige Möglichkeit zu realisieren: Alle einspeisenden Randknoten geben an den Clusterknoten

ab, alle abnehmenden Randknoten decken ihren Bedarf ebenfalls über ihre einzig bestehende Verbindung zum Clusterknoten. Durch diese Festlegung der Flussrichtungen entfallen die binären Richtungsvariablen $\tau_{v,w}$ aus den Nebenbedingungen der Optimierungsaufgabe W' und machen sie damit zu einem *non-linear program* (NLP) ohne Ganzzahligkeitsbedingungen. Der Presolver in SCIP (Abschnitt 4.8) entfernt in diesem Fall die besagten Variablen und übergibt das umformulierte Problem dann weiter an den *branch-and-bound*-Algorithmus.

Das Modell W' wurde in ein Zimpl-Modell übersetzt und dieses dann von SCIP bearbeitet. Die Ergebnisse für verschiedene Instanzen und deren Interpretation finden sich in Kapitel 6.

5.3.1 Aussagen zur Konvexität

Ist eine Optimierungsaufgabe *konvex*, so ist im Unterschied zur nichtkonvexen Optimierung jede lokale Lösung gleichzeitig auch optimal. Kann man zeigen, dass ein Problem konvex ist, vereinfacht dies die Lösungsfindung.

Definition 5.1. Sei $C \subseteq \mathbb{R}^n$ mit $C \neq \emptyset$ eine konvexe Menge und f eine Abbildung $f : C \rightarrow \mathbb{R}$. Dann heißt f *konvex* wenn gilt:

$$\begin{aligned} f(\lambda x + (1 - \lambda)y) &\leq \lambda f(x) + (1 - \lambda)f(y) \\ \forall \lambda &\in [0, 1] \\ \forall x &\in C. \end{aligned}$$

Definition 5.2. Sei $R \neq \emptyset$ eine konvexe Menge. Eine nichtlineare Optimierungsaufgabe heißt *konvex*, wenn die Zielfunktion $z : R \rightarrow \mathbb{R}$ auf R konvex ist.

Satz 5.1. Das Optimierungsproblem W ist *nicht konvex*.

Beweis. Wenn die Nebenbedingungen, welche aus der Druckverlustgleichung (3.4) stammen, nicht konvex sind, ist R und somit W ebenfalls nicht

konvex. Es bleibt zu zeigen, dass die Druckverlustgleichung nicht konvex ist. Der positive Teil der Druckverlustgleichung (3.4) mit $q_{v,w} \geq 0$ führt mit der Notation $x = (p_{v1}^2 - p_{w1}^2)$, $x \geq 0$ und $y = (p_{v2}^2 - p_{w2}^2)$, $y \geq 0$ nach Definition 5.1 auf folgende Gleichungen:

$$\begin{aligned} q(x)^2 &= cx \\ q(x) &= \sqrt{cx} \\ q(\lambda x + (1-\lambda)y) &= \sqrt{c(\lambda x + (1-\lambda)y)} \\ \lambda q(x) + (1-\lambda)q(y) &= \lambda\sqrt{cx} + (1-\lambda)\sqrt{cy}. \end{aligned}$$

Es muss also gelten:

$$\begin{aligned} \sqrt{c(\lambda x + (1-\lambda)y)} &\leq \lambda\sqrt{cx} + (1-\lambda)\sqrt{cy} && | \text{PG}^1 \\ \sqrt{c}\sqrt{\lambda x + (1-\lambda)y} &\leq \lambda\sqrt{c}\sqrt{x} + (1-\lambda)\sqrt{c}\sqrt{y} && | : \sqrt{c} \\ \sqrt{\lambda x + (1-\lambda)y} &\leq \lambda\sqrt{x} + (1-\lambda)\sqrt{y} && | ()^2 \\ \lambda x + (1-\lambda)y &\leq (\lambda\sqrt{x} + (1-\lambda)\sqrt{y})^2 && | \text{AM}^1 \\ \lambda x + (1-\lambda)y &\leq \lambda^2 x + (1-\lambda)^2 y + 2\lambda(1-\lambda)\sqrt{xy} && | \text{AM}^1 \\ \lambda x - \lambda y + y &\leq \lambda^2 x + y - 2\lambda y + \lambda^2 y + 2\lambda\sqrt{xy} - 2\lambda^2\sqrt{xy} && | \text{AK}^1 \\ \lambda(x-y) &\leq \lambda(\lambda x - 2y + \lambda y + 2\sqrt{xy} - 2\lambda\sqrt{xy}) \\ &&& \text{wahre Aussage für } \lambda = 0, \text{ sonst } | : \lambda \\ x - y &\leq \lambda x - 2y + \lambda y + 2\sqrt{xy} - 2\lambda\sqrt{xy} && | - x + y \\ 0 &\leq \lambda x - y + \lambda y + 2\sqrt{xy} - 2\lambda\sqrt{xy} - x && | \text{AK}^1 \\ 0 &\leq (\lambda - 1)(x + y - 2\sqrt{xy}) && | \text{BF}^1 \\ 0 &\leq (\lambda - 1)\underbrace{(\sqrt{x} - \sqrt{y})^2}_{\geq 0} && | \text{UA}^1 \\ 0 &\leq (\lambda - 1) && | +1 \\ 1 &\leq \lambda \\ &\implies \lambda \in \{0; 1\} \end{aligned}$$

Da $\{0;1\} \subset [0,1]$, ist $q(x)$, $x \geq 0$ nach Definition 5.1 nicht konvex. Somit ist die gesamte Druckverlustgleichung, einschließlich negativen Werten für x , nicht konvex. \square

¹PG - Potenzgesetz, AM - Ausmultiplizieren, AK - Ausklammern, BF - Binomische Formel, UA - Untere Abschätzung

Nach [Mau77] existiert eine Möglichkeit, die nicht konvexen Nebenbedingungen der Druckverlustgleichungen auf eine äquivalente konvexe Optimierungsaufgabe zu übertragen. Demnach entspricht die Lösung eines Gleichungssystems mit n Flusserhaltungsgleichungen und m Druckverlustgleichungen einem Minimierungsproblem

$$f = \sum_{(i,j) \in E} \frac{c_{i,j}}{3} |q_{i,j}| q_{i,j}^2 + \sum_{i \in V} p_i \left(\sum_{j \in \omega(i)} q_{i,j} \right) \rightarrow \min$$

sowie gleichzeitig einem Maximierungsproblem

$$g = - \sum_{(i,j) \in E} \frac{2}{3\sqrt{c_{i,j}}} |(p_j - p_i)|^{\frac{2}{3}} - \sum_{i \in V} b_i (2p_s - p_i) \rightarrow \max.$$

Daraus folgt, dass das Optimum bei $f = g$ liegt. Diese Überlegung sei an dieser Stelle rein informativ. Die Möglichkeit der Umformulierung wurde bei der Implementierung unbeachtet gelassen.

5.3.2 Linearisierung der Druckverlustgleichung

Da die heutigen Löser wie SCIP effektiv MILP lösen können, ist man bemüht, alle Nichtlinearitäten eines MINLP durch lineare Zusammenhänge zu beschreiben (siehe Abschnitt 4.6).

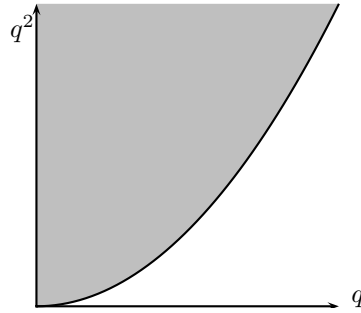
Die Nichtlinearität des Optimierungsproblems W' rührt ausschließlich aus den Nebenbedingungen mit der Druckverlustgleichung (3.4). Da die Druckwerte an den Knoten ausschließlich quadratisch auftreten, substituiert man diese durch eine neue Variable $\pi_i \in \mathbb{R}_+$, $\forall i \in V$ mit

$$q_{i,j} \cdot |q_{i,j}| = c_{i,j} \cdot (\pi_i - \pi_j) \quad (i, j) \in E. \quad (5.14)$$

Durch diesen Austausch wird die Menge der Nebenbedingungen in W' durch

$$(2 \cdot \tau_{i,j}^s - 1) \cdot (\tilde{q}_{i,j}^s)^2 - \tilde{c}_{i,j} \cdot \tilde{\pi}_i^s + \tilde{c}_{i,j} \cdot \tilde{\pi}_j^s = 0 \quad (i, j) \in \tilde{E}, s \in \mathcal{S}_V$$

ersetzt. Zudem muss durch die Substitution auch in der Zielfunktion von W' eine Änderung vorgenommen werden. Obwohl es keine äquivalente Um-

Abbildung 5.5: Epigraph von $q \mapsto q^2$ für $q \geq 0$

formung ist, bleibt das Ziel, den Unterschied zwischen den Druckwerten zu minimieren, durch die neue Zielfunktion

$$z = \sum_{s \in S} \sum_{i \in V_T} (\pi_i^s - \tilde{\pi}_i^s)^2 \rightarrow \min \quad (5.15)$$

erhalten. Es wird sogar verschärft, indem sich nun große relative Unterschiede zwischen relativ hohen Druckwerten gravierender auf den Zielfunktionswert auswirken. Das ist insofern für das Modell stimmig, da höhere Druckwerte den Knoten zugeordnet werden können, die eine wichtigere Rolle im Gesamtsystem einnehmen als jene mit relativ kleinen Druckwerten.

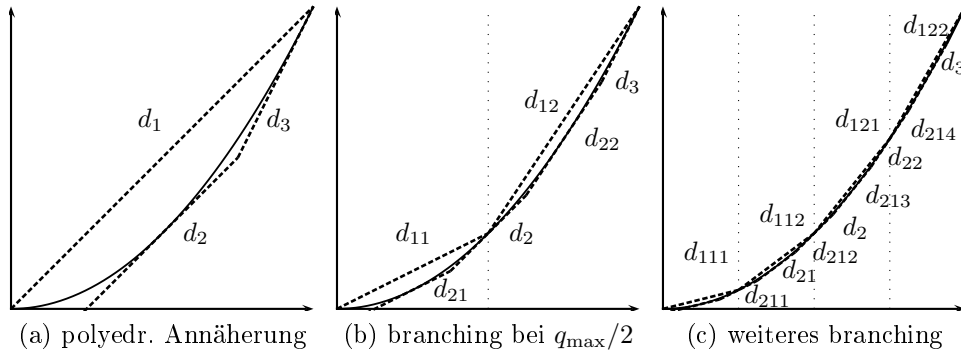
Die verbleibende Nichtlinearität ist der quadratische Flusswert $(q_{v,w})^2$ mit $q_{v,w} \geq 0$. Da $q_{v,w}$ nicht ausschließlich quadratisch vorkommt, ist eine Substitution nicht möglich. Die Funktion $q_{v,w} \mapsto (q_{v,w})^2$ wird stattdessen durch eine lineare Annäherung der konvexen Hülle an den Epigraphen der Funktion ersetzt.

Definition 5.3. Sei $C \subseteq \mathbb{R}^n$ mit $C \neq \emptyset$ eine konvexe Menge und f eine Abbildung $f : C \rightarrow \mathbb{R}$. Dann heißt die Menge

$$\text{epi} f := \{(x, z)^T \in \mathbb{R}^{n+1} \mid f(x) \leq z\} \subseteq \mathbb{R}^{n+1}$$

Epigraph oder *oberer Halbzylinder* von f , siehe auch Abbildung 5.5.

Die Annäherung kann anschließend durch Schnittebenen und beliebig

Abbildung 5.6: Beispiel für *spatial branching*

wiederholtes *spatial branching* (siehe Abschnitt 4.6) verbessert werden. Je größer die Anzahl der neu entstehenden Nebenbedingungen, desto besser wird die Nichtlinearität approximiert. Folgendes Beispiel solch einer Linearisierung ist in Abbildung 5.6 veranschaulicht.

Sei q_{\max} die obere Schranke für q . Die erste äußere polyedrische Annäherung an $q \mapsto q^2$ kann durch zwei lineare Funktionen ober- und unterhalb der Kurve sowie einer zusätzlichen Schnittebene realisiert werden:

$$\begin{aligned}
 d_1 : \quad & q_{\max} \cdot q \geq q^2 \\
 d_2 : \quad & q_{\max} \cdot q - \frac{1}{4}q_{\max}^2 \leq q^2 \\
 d_3 : \quad & 2q_{\max} \cdot q - q_{\max}^2 \leq q^2.
 \end{aligned} \tag{5.16}$$

Nach *spatial branching* bei $\frac{1}{2}q_{\max}$ ergeben sich weitere Einschränkungen:

$$\begin{aligned}
 d_{11} : \quad & \frac{1}{2}q_{\max} \cdot q \geq q^2 \quad q \leq \frac{1}{2}q_{\max} \\
 d_{12} : \quad & \frac{3}{2}q_{\max} \cdot q - \frac{1}{2}q_{\max}^2 \geq q^2 \quad q \geq \frac{1}{2}q_{\max} \\
 d_2 : \quad & q_{\max} \cdot q - \frac{1}{4}q_{\max}^2 \leq q^2 \\
 d_{21} : \quad & \frac{1}{2}q_{\max} \cdot q - \frac{1}{16}q_{\max}^2 \leq q^2 \\
 d_{22} : \quad & \frac{3}{2}q_{\max} \cdot q - \frac{9}{16}q_{\max}^2 \leq q^2 \\
 d_3 : \quad & 2q_{\max} \cdot q - q_{\max}^2 \leq q^2.
 \end{aligned} \tag{5.17}$$

Da die Funktion $q \mapsto q^2$ für alle $q \geq 0$ monoton wächst, gelten alle \leq -Bedingungen für jedes $q \geq 0$ und müssen nicht in zusätzliche Intervalle gefasst werden.

Nach einem weiteren *spatial branching* bei $\frac{1}{4}q_{\max}$ und $\frac{3}{4}q_{\max}$ ergeben sich insgesamt 12 Bedingungen für q^2 :

$$\begin{aligned}
d_{111} : \quad & \frac{1}{4}q_{\max} \cdot q \geq q^2 & q \leq \frac{1}{4}q_{\max} \\
d_{112} : \quad & \frac{3}{4}q_{\max} \cdot q - \frac{1}{8}q_{\max}^2 \geq q^2 & \frac{1}{4}q_{\max} \leq q \leq \frac{1}{2}q_{\max} \\
d_{121} : \quad & \frac{5}{4}q_{\max} \cdot q - \frac{3}{8}q_{\max}^2 \geq q^2 & \frac{1}{2}q_{\max} \leq q \leq \frac{3}{4}q_{\max} \\
d_{122} : \quad & \frac{7}{4}q_{\max} \cdot q - \frac{3}{4}q_{\max}^2 \geq q^2 & \frac{3}{4}q_{\max} \leq q \\
d_2 : \quad & q_{\max} \cdot q - \frac{1}{4}q_{\max}^2 \leq q^2 \\
d_{21} : \quad & \frac{1}{2}q_{\max} \cdot q - \frac{1}{16}q_{\max}^2 \leq q^2 \\
d_{22} : \quad & \frac{3}{2}q_{\max} \cdot q - \frac{9}{16}q_{\max}^2 \leq q^2 \\
d_{211} : \quad & \frac{1}{4}q_{\max} \cdot q - \frac{1}{64}q_{\max}^2 \leq q^2 \\
d_{212} : \quad & \frac{3}{4}q_{\max} \cdot q - \frac{9}{64}q_{\max}^2 \leq q^2 \\
d_{213} : \quad & \frac{5}{4}q_{\max} \cdot q - \frac{25}{64}q_{\max}^2 \leq q^2 \\
d_{214} : \quad & \frac{7}{4}q_{\max} \cdot q - \frac{49}{64}q_{\max}^2 \leq q^2 \\
d_3 : \quad & 2q_{\max} \cdot q - q_{\max}^2 \leq q^2.
\end{aligned} \tag{5.18}$$

Wie bereits erwähnt, wurde für die Lösungsermittlung in Kapitel 6 das Modell W' (5.13) verwendet. Numerische Modellmodifizierungen wie die Linearisierung von Nichtlinearitäten müssen nicht unbedingt manuell ausgeführt werden. Löser wie SCIP verwenden gleichartige Verfahren, um eine Aufgabe umzuformulieren, damit diese schneller und einfacher gelöst werden kann.

Kapitel 6

Auswertung der Ergebnisse

Um die Ergebnisse aussagekräftig bewerten zu können, bedarf es Antworten auf mehrere Fragen, die sowohl an das Originalnetz, als auch an das aggregierte Netz gestellt werden. Die Unterschiede und Gemeinsamkeiten in den Antworten geben Aufschluss darüber, mit welchem Informationsverlust sich die Darstellung des Originalnetzes aggregieren lässt. Im Abschnitt 6.1 soll gezeigt werden, wie gut sich ein Originalnetz über das Optimierungsproblem selbst erzeugt. Diese Selbstapproximation kann Aufschluss darüber geben, welche Parameter das Ergebnis und die Laufzeit beeinflussen, da das Ergebnis bereits bekannt ist. Im Abschnitt 6.3 werden ein paar Anmerkungen zu einer möglichen Gütebetrachtung aufgeführt.

6.1 Selbstapproximation

Eine Möglichkeit, die Lösung des Optimierungsproblems auf Genauigkeit zu untersuchen, ist, zu prüfen, ob sich ein Netz selbst approximieren wird. Man wählt als Instanz ein Netz, welches die gleiche Anzahl an Knoten und Kanten besitzt, wie das daraus resultierende aggregierte Netz. Als Gütemaß für die Lösung soll der relative Fehler über alle Rohrkonstanten,

$$\Delta = \frac{1}{|E|} \sum_{(i,j) \in E} \frac{|c_{i,j} - \tilde{c}_{i,j}|}{c_{i,j}} \quad (6.1)$$

dienen.

Die folgenden Ergebnisse wurden auf einem System mit folgenden Kenngrößen ermittelt:

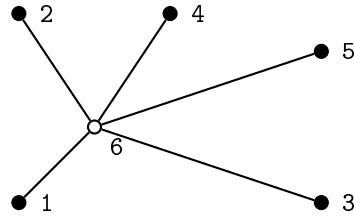
2 x 2,50 GHz Intel[®] Celeron[®]

3,9 GiB RAM

Kernel Linux 2.6.35-30-generic (64 Bit)

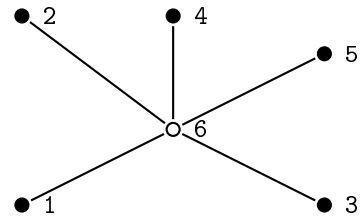
Etwaige Zeit- und Genauigkeitsunterschiede sind in jedem Fall auch auf das genutzte Betriebssystem zurückzuführen. Eine kleine Differenz in der Numerik kann große Abweichungen in der Laufzeit zur Folge haben.

A:
Vr=[1,2,3,4,5]
Vp=[6]
B1=[20;-13;5;-15;-3;6]
B2=[2;-1;1;-2;-1;1]
C=[10;10;10;10;10]



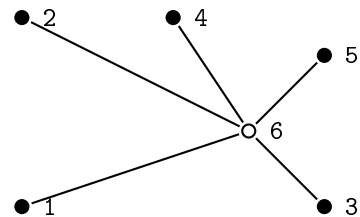
(a) Testinstanz A

B:
Vr=[1,2,3,4,5]
Vp=[6]
B3=[15;17;9;-2;-5;-34]
C=[1.4;110.8;6.4;0.5;63.6]



(b) Testinstanz B

C:
Vr=[1,2,3,4,5]
Vp=[6]
B1=[20;-13;5;-15;-3;6]
C=[9.3;359.6;12.8;28.9;0.7]



(c) Testinstanz C

Abbildung 6.1: Testinstanzen A, B und C

D:
$V_r = [1, 2, 3]$
$V_p = [4, 5, 6]$
$B_4 = [15; 17; 9; -2; -5; -34]$
$E = [(1, 4), (2, 5), (3, 6),$ $(4, 5), (4, 6), (5, 6)]$
$C = [10; 10; 10; 10; 10; 10]$

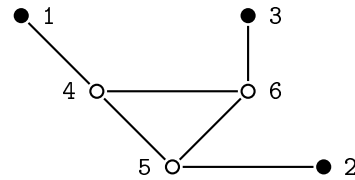


Abbildung 6.2: Testinstanz D

Name	Szenarien	p_0 in bar	Zeit in s	ZFW	Δ in %
A ₆₀	B1	60	29	0	0,03
A ₇₀	B1	70	1	0	0,01
A ₈₀	B1	80	2	0	0,003
A ₉₀	B1	90	0,2	0	0,01
A ₁₀₀	B1	100	0,07	0	0,006
A _{α}	B1, B2	90	2	0	0
B ₆₀	B3	60	236	0	0,08
B ₇₀	B3	70	> 1000	-	-
B ₈₀	B3	80	26	0	105
B ₉₀	B3	90	21.2	0	105
B ₁₀₀	B3	100	> 1000	-	-
C ₆₀	B1	60	> 1000	-	-
C ₇₀	B1	70	12	0	0,8
C ₈₀	B1	80	30	0	780
C ₉₀	B1	90	224	0	1,5
C ₁₀₀	B1	100	599	0	2363
D ₆₀	B4	60	589	0	72
D ₇₀	B4	70	111	0	58
D ₈₀	B4	80	75	0	68
D ₉₀	B4	90	0,7	0	45
D ₁₀₀	B4	100	1	0	63

Abbildung 6.3: Ergebnisse der Instanzen A,B,C und D

Die Tatsache, dass in **A** (Abbildung 6.1) alle Rohrkonstanten denselben Wert haben, welcher weder sehr groß noch sehr klein ist, scheinen dem Löser zu schnellen und guten Resultaten in kurzen Zeiten zu verhelfen (siehe für Ergebniswerte Abbildung 6.3). Die letzte Instanz A_α , die beide Bilanzszenarien einbezieht, gibt die Rohrkonstanten des Originalnetzes exakt zurück.

Beispiel **B** (Abbildung 6.1) lässt vermuten, dass es einen Zusammenhang zwischen Laufzeit und Genauigkeit des Ergebnisses gibt. Mit dem Druckscenario in B_{60} benötigt der Löser zwar länger, findet jedoch eine sehr gute Näherung. Andere Instanzen benötigen zu lange oder weichen stark vom Originalnetz ab. Durch eine zulässige Wahl des Druckes am Clusterknoten kann trotz extremer, aber zulässiger Rohrkonstanten der Originaldruck an den Randknoten erreicht werden. Die Zielfunktion wird in jedem Falle minimiert. Die Vermutung einer eventuellen Proportionalität zwischen Laufzeit und Genauigkeit des Ergebnisses wird durch Beispiel **C** (Abbildung 6.1) widerlegt. Die Güte der Ergebnisse ist nicht von der benötigten Zeit abhängig.

Beispiel **D** (Abbildung 6.2) unterscheidet sich von den anderen Beispielinstanzen dahingehend, dass die Rohrkonstanten zu drei statt zu einem Clusterknoten approximiert werden sollen. Diese sind über mehrere Kanten miteinander verbunden. Dadurch werden, anders als bei den vorherigen Beispielinstanzen, die binären Richtungsvariablen τ nicht vom Presolver substituiert (siehe Abschnitt 5.3). Es wird ein MINLP gelöst. Unterschiedliche Druckscenarien führen auch hier auf unterschiedliche Laufzeiten und Ergebnisse. Daraus lässt sich schließen, dass die Laufzeit von der Verteilung der Druck- und Bilanzwerte abhängt. Erwähnenswert bei **D** ist die relativ geringe Streuung von Δ bezüglich verschiedener Initialdruckwerte.

Die starken Abweichungen in den Beispielen **B** und **C** sind dadurch möglich, dass keine eindeutige Lösung existiert. Es wird die Optimallösung ausgegeben, die im Verzweigungsbaum des *branch-and-bound*-Algorithmus zuerst gefunden wird.

Durch die vorgegebenen Bilanzen B_V werden ausschließlich die Flusswerte q fixiert. Das heißt für die Zusammenfassung mehrerer äußerer Clusterkanten zu einem Clusterknoten v_Z unter Verwendung der Druckverlustgleichung (3.4), dass

$$\begin{aligned} \sum_{(v,i) \in E, i \in Z} q_{v,i} &= q_{v,v_Z} \\ \sum_{(v,i) \in E, i \in Z} \sqrt{c_{v,i} \cdot |p_v^2 - p_i^2|} &= \sqrt{c_{v,v_Z} \cdot |p_v^2 - p_{v_Z}^2|} \quad (6.2) \\ \frac{1}{|p_v^2 - p_{v_Z}^2|} \cdot \left(\sum_{(v,i) \in E, i \in Z} \sqrt{c_{v,i} \cdot |p_v^2 - p_i^2|} \right)^2 &= c_{v,v_Z} \end{aligned}$$

gelten muss. Im hier betrachteten Fall der Selbstapproximation wird (6.2) auf folgenden Ausdruck vereinfacht:

$$c_{v,w} \cdot \frac{p_v^2 - p_w^2}{p_v^2 - p_{v_Z}^2} = c_{v,v_Z} \quad (6.3)$$

Das Ziel ist es, $p_w = p_{v_Z}$ und somit $c_{v,w} = c_{v,v_Z}$ zu erreichen. Im Modell allerdings, sind der Druck am Clusterknoten p_{v_Z} sowie die Rohrkonstante c_{v,v_Z} variabel in ihren Grenzen. Dadurch ist die Lösung der Gleichung (6.3) nicht eindeutig. Die Optimierungsaufgabe besitzt kein einzelnes globales Optimum. Das Einsetzen von oberen oder unteren Schranken führt zwar auf einen minimalen Zielfunktionswert aber zu teilweise stark abweichenden Approximationen, die eventuell nicht auf reale Zustände übertragbar sind. Gleichzeitig führen die großen Zulässigkeitsintervalle zu langen Laufzeiten im *branch-and-bound*-Algorithmus. Um diesen Effekt zu vermeiden, bedarf es weiterer Nebenbedingungen im Modell.

6.2 Nachträgliche heuristische Schranken

Der Restriktionsbereich muss weiter eingeschränkt werden, damit der Löser ein globales Optimum wählt, welches auf reale Zustände übertragbar ist, und die Laufzeit verkürzt wird. Es sollen im Folgenden die Schranken der Rohrkonstante c_{v,v_Z} neu definiert werden, welche von einem Randknoten

$v \in V_R$ zu einem Cluster Z führt:

$$c_{v,v_Z}^{\min} \leq c_{v,v_Z} \leq c_{v,v_Z}^{\max}. \quad (6.4)$$

Dazu wird sich der Gleichung (6.2) bedient. Der Druck am Clusterknoten wird als ein Druck zwischen dem kleinsten Druck und dem größten vorkommenden Druck aller Knoten aus Z

$$p_Z^{\min} \leq p_{v_Z} \leq p_Z^{\max}$$

angenommen. Wir gehen vorerst vereinfachend von nur einem Bilanzenszenario aus. Dann liegt die zu ermittelnde Rohrkonstante zwischen den beiden Werten, die mit p_Z^{\min} und p_Z^{\max} berechnet werden. Da die Proportionalität von Druckwert und Rohrkonstante von der Durchflussrichtung abhängt, müssen für c_{v,v_Z}^{\min} und c_{v,v_Z}^{\max} jeweils beide Fälle unterschieden werden.

$$c_{v,v_Z}^{\min} = \begin{cases} \frac{q_{v,v_Z}^2}{|p_v^2 - (p_Z^{\min})^2|} & , \text{ wenn } q_{v,v_Z} \geq 0 \\ \frac{q_{v,v_Z}^2}{|p_v^2 - (p_Z^{\max})^2|} & , \text{ wenn } q_{v,v_Z} \leq 0 \end{cases} \quad (6.5)$$

$$c_{v,v_Z}^{\max} = \begin{cases} \frac{q_{v,v_Z}^2}{|p_v^2 - (p_Z^{\max})^2|} & , \text{ wenn } q_{v,v_Z} \geq 0 \text{ und } p_v \notin [p_Z^{\min}; p_Z^{\max}] \\ \frac{q_{v,v_Z}^2}{|p_v^2 - (p_Z^{\min})^2|} & , \text{ wenn } q_{v,v_Z} \leq 0 \text{ und } p_v \notin [p_Z^{\min}; p_Z^{\max}] \\ c_{\text{tech}}^{\max} & , \text{ sonst} \end{cases}$$

Für c_{v,v_Z}^{\max} muss aufgrund der Unstetigkeit der Division die Zusatzbedingung $p_v \notin [p_Z^{\min}; p_Z^{\max}]$ erfüllt sein. Ist dies nicht der Fall, wird für c_{v,v_Z}^{\max} der maximale technisch mögliche Wert eingesetzt. Im Fall der Selbstapproximation aus Abschnitt 6.1 führen die Formeln durch $p_Z^{\min} = p_Z^{\max}$ auf

$$c_{v,v_Z}^{\min} = c_{v,v_Z} = c_{v,v_Z}^{\max} \quad (6.6)$$

E:
$V_r = [1, 2, 3, 4]$
$V_p = [5, 6, 7, 8, 9]$
$B_6 = [-20; 10; 4; 16; 0; -8; -7; -5; 10]$
$B_7 = [-15; 12; 6; 11; 0; -8; -12; -4; 10]$
$B_8 = [-15; -12; 6; 11; 0; -8; 13; -5; 10]$
$B_9 = [-20; -10; 4; 16; 0; -8; 13; -5; 10]$
$E = [(1, 5), (2, 7), (3, 8), (4, 7),$ $(5, 6), (5, 8), (6, 7), (6, 9),$ $(7, 9), (8, 9)]$
$C = [10; 10; 10; 10; 10; 10; 10; 10; 10]$
techn. c_max: 100

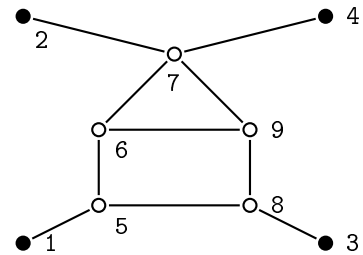


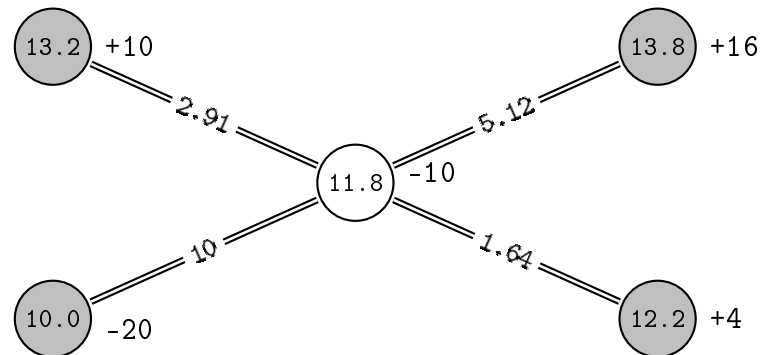
Abbildung 6.4: Testinstanz E

und damit auf eine exakte Selbstapproximation für jedes beliebige Bilanzszenario. Zudem sind die Laufzeiten nun verschwindend klein, da nur noch eine zulässige Lösung existiert. Zusammen mit der Nebenbedingung (6.5) können Testläufe zur Selbstapproximation keine interpretierbaren Aussagen zur Güte der Lösung mehr liefern.

Bemerkung. Wird zur Ermittlung eines Druckszenarios die Vorgehensweise aus Abschnitt 5.1 genutzt, ergeben sich für unterschiedliche Initialdruckwerte die selben Grenzen für die Rohrkonstanten.

Sei E (Abbildung 6.4) ein zu aggregierendes Netz. Alle passiven Knoten sollen zu einem Clusterknoten zusammengefasst werden. Für das gegebene Bilanzszenario B6 existiert ein gemeinsamer Druck am Clusterknoten. Mit den Schranken aus (6.5) wird das Problem von SCIP in 0.12 Sekunden gelöst (siehe Ergebnistabelle in Abbildung 6.6, sowie Abbildung 6.5).

Anders sieht es bei Instanzen mit nicht exakter Lösung oder mehreren Bilanzszenarien aus. Solche Instanzen benötigen sehr lange, da die Bilanzen nicht erfüllt werden können oder Flussrichtungen in den einzelnen Szenarien nicht übereinstimmen. Seien verschiedene Bilanzszenarien für die Instanz E gegeben: B6 und B7, welche jeweils einen gemeinsamen Druck am Clusterknoten für alle Verbindungen zu den Randknoten besitzen und die gleichen

Abbildung 6.5: aggregiertes Netz aus Instanz E_1

Flussrichtungen zu bzw. von den Randknoten erfordern, B8, welches ebenfalls eine exakte Lösung besitzt, aber diese mit anderen Flussrichtungen erreicht und letztlich B9, für welches kein exakter gemeinsamer Druck am Clusterknoten existiert. Die Ergebnisse in Abbildung 6.6 zeigen, dass Netze, in denen verschiedene Bilanzszenarien in die Lösung eingehen sollen, und solche, mit ungünstig gewählten Bilanzszenarios extrem lange zur Lösungsfindung benötigen. Selbst mit den nachträglich eingeführten heuristischen Schranken der Rohrkonstanten schaffte es der Löser für diese Instanzen nicht, brauchbare Ergebnisse in annehmbaren Zeiten zu finden.

Name	Szenarien	p_0	Zeit in s	ZFW	Bemerkung
E_1	B6	10	0.12	0	
E_2	B7	10	0.07	0	
E_3	B6, B7	10	>1000	-	1. zulässige Lsg bei 28s
E_4	B8	10	0.69	0	
E_5	B6, B8	10	>1000	-	bis 1000s ohne zulässige Lsg
E_6	B9	10	>1000	-	1. zulässige Lsg bei 313s
E_7	B6, B9	10	>1000	-	bis 1000s ohne zulässige Lsg

Abbildung 6.6: Ergebnisse der Instanz E

6.3 Gütebetrachtung

Werden durch einen auf dem aggregierten Netz arbeitenden Algorithmus Aussagen verschiedener Natur ermittelt, möchte man diese auch bewerten können. Angestrebt wird eine quantitative Aussage, die besagt, wann eine Antwort mit der auf dem entsprechenden Originalnetz ermittelten übereinstimmt. Im Abschnitt 6.1 war dieses Gütemaß die Abweichung der Rohrkonstanten zum Originalnetz, wie in Gleichung (6.1) definiert. Dieses Maß ist nur zu Testzwecken zu gebrauchen. Gehen wir vom eigentlichen Ziel aus, große Knotenmengen auf viel kleinere zu reduzieren, scheint die Findung eines Gütemaßes ohne statistische a-posteriori-Betrachtungen kaum umsetzbar zu sein.

Bei der Aggregation passiver Teilnetze kann man allerdings unter Beachtung der realen Zustände eine Besonderheit feststellen. Passive Teilnetze sind meist regionale Netze, welche nah an den Endversorgern liegen. Diese Knoten werden weder ihren Gas-Abnehmer-Status plötzlich ablegen, um Gas einzuspeisen, noch wird sich der Gasbedarf auf einmal verdoppeln. Das heißt, dass Flussrichtungen eventuell sogar vorausgesetzt werden können. Wenn solch ein passiver Netzbestandteil mittels dem in Kapitel 5 beschriebenen Verfahren vereinfacht werden soll, genügen wenige, in der Realität vorkommende Bilanzen- und Druckszenarien. Dadurch gehen keine unrealistischen Zustände in die Nebenbedingungen ein, die starken Einfluss auf das Ergebnis haben können.

Ein Großteil an Möglichkeiten, die Gasmengen zu verteilen, geht verloren, wenn durch die Aggregation relativ viele äußere Clusterkanten, die zu einem Randknoten führen, zu einer einzelnen Kante zusammengefasst werden. In solch einem Fall wäre es eventuell günstiger, die Clusterung anders zu erstellen, siehe dazu Abbildung 3.3.

Kapitel 7

Fazit und Ausblick

In dieser Diplomarbeit wurde anhand eines Energieversorgungsproblems auf die mathematischen Methoden und Verfahren heutiger Löser für Gleichungssysteme und MIP eingegangen. Ein Beispielproblem, welches sich mit einem speziellen Teil des Gasnetzes beschäftigt, konnte zu einem Optimierungsproblem modelliert werden. Daraus resultierende Ergebnisse wurden präsentiert und ausgewertet. Zur Lösung des Modells diente das Softwarepaket des ZIB, welches sehr gute, freie Werkzeuge zur Lösung großer Probleme bereitstellt.

Das Ziel, das Originalnetz mit dem aggregierten Netz bezüglich eines Algorithmus zur Lösung des *pipe network analysis Problems* zu vergleichen, konnte im Rahmen dieser Arbeit nur teilweise betrachtet werden. Es bedarf im Weiteren einer aufwendigen Implementierung in vorhandene Programmcodes sowie anschließenden Benchmark-Tests, um Aussagen darüber treffen zu können, welchen Laufzeitverbesserungen welche Güteverluste gegenüberstehen.

Passivnetze, so wie sie in dieser Arbeit beschrieben wurden, nehmen einen großen Teil des real vorkommenden Gasnetzes ein. Ihre Aggregation würde das reale Netz enorm ausdünnen, jedoch steht diesem Vorteil für das *pipe network analysis Problem* ein hoher Zeitaufwand und Informationsverlust entgegen. Hier stellt sich die Frage, ob sich eine Aggregation in der beschriebenen Form überhaupt lohnt.

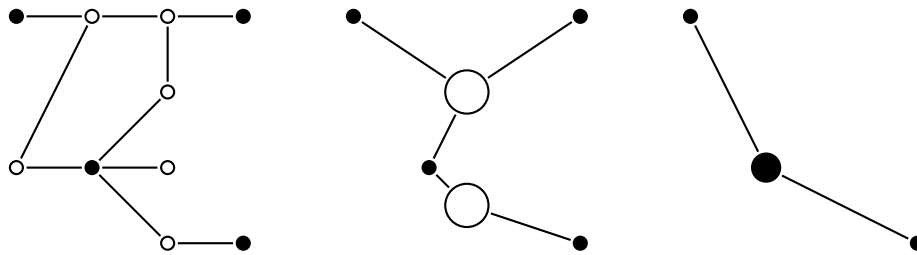


Abbildung 7.1: mögliche sukzessive Aggregation

Zudem könnte im Weiteren auch Interesse daran bestehen, aktive Elemente des Netzes zu reduzieren. Dafür werden noch tiefgreifendere Kenntnisse zu den physikalischen Zusammenhängen an aktiven Netzbestandteilen benötigt, welche in Form von Differentialgleichungen in das Modell eingehen werden. Letztendlich könnte durch eine stufenweise Aggregation ein Netz sukzessive vereinfacht werden, um verschieden starke Manipulationen miteinander zu vergleichen, siehe dazu Abbildung 7.1.

Das Gas-Projekt des MATHEON-Forschungszentrums ist ein Vorhaben, welches in vielen mathematischen Bereichen neue Erkenntnisse sammelt und versucht, diese in einer komfortablen Softwareumgebung zu verarbeiten. Die daraus resultierenden Ergebnisse sind richtungsweisend auf diesem Gebiet und können neben der Gasversorgung auch für andere gleichartige Energieversorgungs- und Transportprobleme verwendet werden.

Literaturverzeichnis

- [ABKW08] ACHTERBERG, Tobias ; BERTHOLD, Timo ; KOCH, Thorsten ; WOLTER, Kati: Constraint Integer Programming: A New Approach to Integrate CP and MIP. In: *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems, 5th International Conference, CPAIOR 2008*, 2008 (Lecture Notes in Computer Science)
- [Ach09] ACHTERBERG, Tobias: *Constraint Integer Programming*, Diss., 2009
- [Ber08] BERTHOLD, Timo: Heuristics of the Branch-Cut-and-Price-Framework SCIP. In: *Operations Research Proceedings*, Springer-Verlag, 2008, S. 31 – 36
- [BHV09] BERTHOLD, Timo ; HEINZ, Stefan ; VIGERSKE, Stefan: Extending a CIP framework to solve MIQCPs / Konrad-Zuse-Zentrum für Informationstechnik (ZIB). 2009. – Forschungsbericht
- [BL95] BODENDIEK, Rainer ; LANG, Rainer: *Lehrbuch der Graphentheorie*. Bd. 1. Spektrum Akademischer Verlag, 1995
- [DS96] DENNIS, J.E. ; SCHNABEL, R.B.: *Numerical methods for unconstrained optimization and nonlinear equations*. Society for Industrial and Applied Mathematics, 1996 (Classics in applied mathematics)

- [DS06] DEMPE, Stephan ; SCHREIER, Heiner: *Operations Research - Deterministische Modelle und Methoden*. Wiesbaden : B. G. Teubner Verlag, 2006
- [EON10] *E.ON AG: Leitungsbau*. Website. <http://www.eon-ruhrgas.com/cps/rde/xchg/SID-4D72CC1E-73009F82/er-corporate/hs.xsl/678.htm>. Version:2010, Abruf: 10.08.2011
- [Fin08] FINKE, Gerd: *Operations Research and Networks*. London : ISTE Ltd, 2008
- [FM05] FÜGENSCHUH, Armin ; MARTIN, Alexander: Computational Integer Programming and Cutting Planes. In: *Handbooks in Operations Research and Management Science, Vol. 12: Handbook on Discrete Optimization*. 2005
- [FVHS10] FÜGENSCHUH, Armin ; VIGERSKE, Stefan ; HOMFELD, Henning ; SCHÜLLDORF, Hanno: Mixed-Integer Nonlinear Problems in Transportation Applications. In: *Proceedings of the 2nd International Conference on Engineering Optimization, 2010*
- [HK00] HAMACHER, Horst W. ; KLAMROTH, Kathrin: *Lineare und Netzwerk-Optimierung*. Braunschweig/Wiesbaden : Vieweg, 2000
- [HNR07] HIRSCHHAUSEN, Christian v. ; NEUMANN, Anne ; RÜSTER, Sophia: Wettbewerb im Ferntransport von Erdgas - Technisch-ökonomische Grundlagen und Anwendung auf Deutschland / TU Dresden. 2007. – Forschungsbericht
- [Kel03] KELLEY, C.T.: *Solving Nonlinear Equations with Newton's Method*. Philadelphia, USA : Society for Industrial and Applied Mathematics (SIAM), 2003
- [Koc04] KOCH, Thorsten: *Rapid Mathematical Programming*, Diss., 2004

- [KV08] KORTE, Bernhard ; VYGEN, Jens: *Combinatorial Optimization - Theory and Algorithms*. Berlin Heidelberg : Springer-Verlag, 2008
- [Mat10] *Matheon-B20 Optimization of Gas Transport*. Website. <http://www.zib.de/de/optimierung/mip/projekte-lang/optimization-of-gas-transport/article/matheon-b20-optimierung-von-gastransport.html>. Version: 2010, Abruf: 12.08.2011
- [Mau77] MAUGIS, J.J.: Étude de réseaux de transport et de distribution de fluide. In: *Revue française d'automatique, d'informatique et de recherche opérationnelle. Recherche opérationnelle*. 1977, S. 243–248
- [Men05] MENON, E. S.: *Gas Pipeline Hydraulics*. Taylor and Francis, 2005
- [RBD88] RICHTER, Knut ; BACHMANN, Peter ; DEMPE, Stephan: *Diskrete Optimierungsmodelle - Effektive Algorithmen und näherungsweise Lösung*. Berlin : VEB Verlag Technik, 1988
- [UL02] UPP, E.Loy ; LANASA, Paul J.: *Fluid Flow Measurement - A Practical Guide to Accurate Flow Measurement*. Boston, USA : Butterworth-Heinemann, 2002
- [WS00] WOLF, Daniel de ; SMEERS, Yves: The Gas Transmission Problem Solved by and Extension of the Simplex Algorithm. In: *Management Science* Bd. 45. 2000, S. 1454–1465
- [ZIB10] *Struktur und Ziele des ZIB*. Website. <http://www.zib.de/de/institut/struktur-und-ziele-des-zib.html>. Version: 2010, Abruf: 10.08.2011

Erklärung zur selbstständigen Anfertigung der Arbeit

Ich erkläre, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Dresden, 05.09.2011

Erik Mucha