Outer Approximation für konvexe MINLP-Probleme

im Rahmen des Seminars „Globale Optimierung“
unter Leitung von Dr. Johannes Schlöder und Dr. Ekaterina Kostina,
Sommersemester 2005, Universität Heidelberg

Hans Joachim Ferreau
ferreau@urz.uni-heidelberg.de

30.06.2005

Inhaltsverzeichnis

1 Einleitung

Ziel der globalen Optimierung ist das Auffinden von globalen Extrempunkten einer Zielfunktion unter Nebenbedingungen. Eine besondere Problemklasse stellen dabei die gemischt-ganzzahligen, nichtlinearen Programme (MINLP) dar. Bei diesen können sowohl die Zielfunktion als auch die Nebenbedingungen nichtlinear sein und darüber hinaus wird für einen Teil der Variablen Ganzzahligkeit gefordert. Abbildung 1 verdeutlicht die Unterschiede zu anderen Optimierungsproblemen.

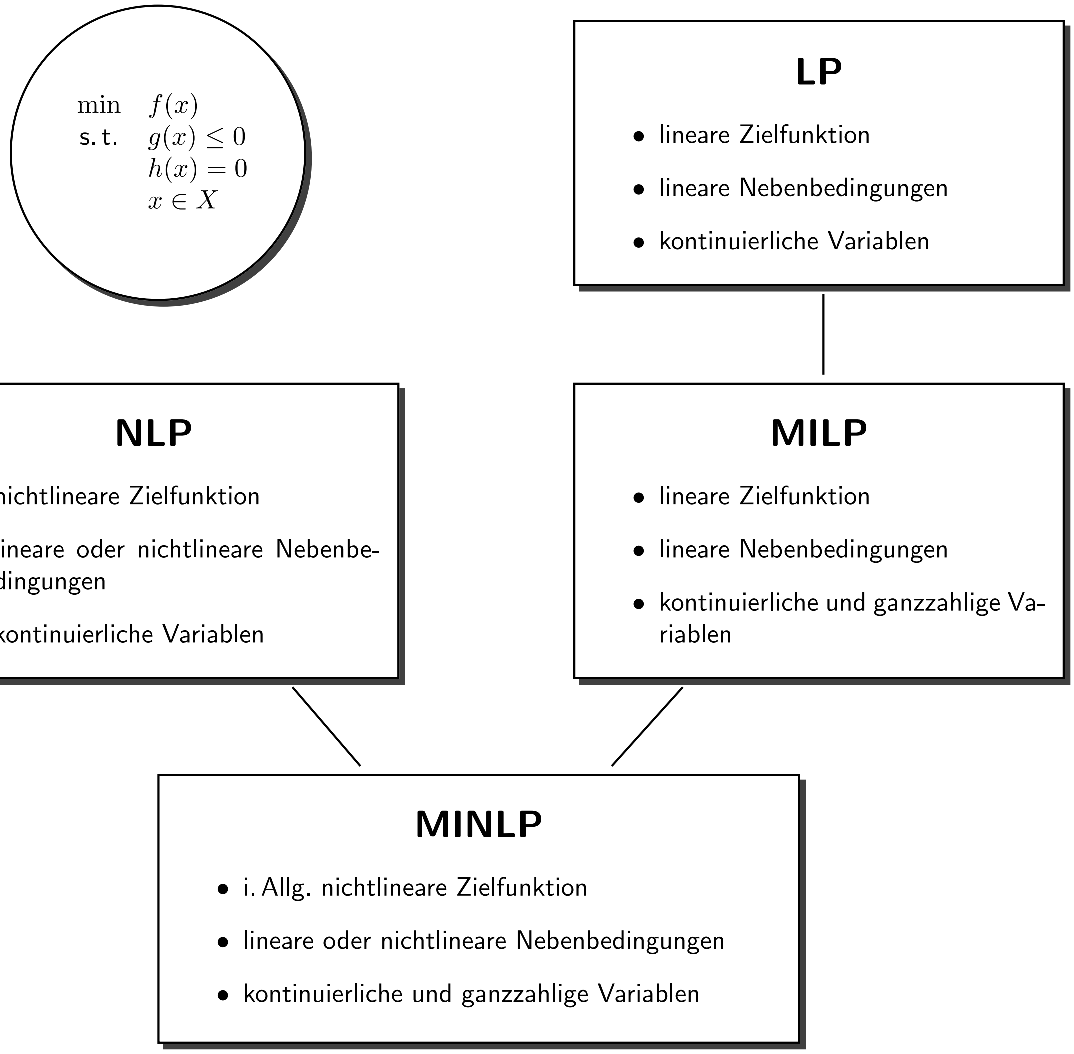

Abbildung 1: Einordnung der MINLP-Probleme in den Kontext der (globalen) Optimierung.

Bereits die globale Minimierung von kontinuierlichen, nichtlinearen Programmen ist eine sehr anspruchsvolle Aufgabe und meist nur unter bestimmten Voraussetzungen möglich. Kommen zusätzlich noch Ganzzahligkeitsbedingungen hinzu, so müssen verständlicherweise noch restriktivere Annahmen gemacht werden, um die Lösbarkeit des Problems zu garantieren. Daher beschränkt sich dieser Vortrag auf das Minimieren von *konvexen Zielfunktionen* unter *konvexen (Ungleichungs-)Nebenbedingungen*. Für derartige Probleme wurde vor knapp 20 Jahren das Konzept der Outer Approximation vorgeschlagen und seitdem mehrfach verbessert. Diese garantiert das Auffinden der optimalen Lösung nach endlich vielen Schritten. Auf Varianten für gleichungsbeschränkte Probleme werden wir ebenso wenig eingehen wie auf solche für nichtkonvexe Zielfunktionen, bei denen die globale Optimalität der Lösung nicht garantiert werden kann.

Auch zur Vereinfachung der Notation werden wir uns ferner auf ganzzahlige Binärvariablen beschränken, also auf solche, die nur die Werte 0 oder 1 annehmen können. Man macht sich aber schnell klar, dass dies keine

Einschränkung der Allgemeinheit bedeutet, da jede ganzzahlige Variable

$$z \in \{L, \dots, U\} \subset \mathbb{Z}, \quad L < U,$$

wie folgt durch Binärvariablen $y_1, \dots, y_n$ ausgedrückt werden kann:

$$z = L + \sum_{j=1}^{n} 2^{j-1} y_j, \quad \text{wobei} \quad n := 1 + \lfloor \log_2(U - L) \rfloor .$$

Außerdem sind Binärvariablen oftmals die natürliche Wahl bei der Modellierung von praktischen Problemen. So wird beispielsweise im Rahmen der Prozesssynthese versucht, gleichzeitig sowohl eine optimale *Auswahl* von Produktionsmaschinen als auch ihre optimalen *Betriebsparameter* zu bestimmen. Die Binärvariablen repräsentieren dabei die Entscheidung, ob eine bestimmte Maschine bei der Produktion eingesetzt werden soll oder nicht. Die kontinuierlichen Variablen modellieren hingegen Einstellungen wie zum Beispiel Betriebstemperatur oder -druck. Dieses Beispiel zeigt auch, weshalb Ganzzahligkeitsforderungen manchmal unverzichtbar sind: Kann man die Temperatur nur in Ein-Grad-Schritten variieren, so ist es meist kein Problem, die optimale Lösung auf den nächsten ganzen Wert zu runden. Bei den Entscheidungsvariablen geht dies hingegen nicht, da eine Maschine entweder ganz oder gar nicht eingesetzt (und vorher gekauft bzw. gebaut) werden muss.

2 Outer Approximation

2.1 Problemformulierung

Das Konzept der Outer Approximation wurde erstmals im Jahre 1986 von Duran und Grossmann in [2] vorgestellt. Es greift einige Ideen der Generalized Benders Decomposition – für eine detailierte Beschreibung siehe z. B. [4] – auf und umfasst (zunächst) folgende Klasse von MINLP-Problemen:

$$\text{OA-MINLP} \quad \left\{ \begin{array}{ll} \min\limits_{x,\,y} & f(x) + c^T y \\ \text{s.t.} & g(x) + By \le 0 \\ & x \in X := \left\{ x \in \mathbb{R}^n \;\middle|\; A_1 x \le a_1 \right\} \\ & y \in Y := \left\{ y \in \{0,1\}^p \;\middle|\; A_2 y \le a_2 \right\} \end{array} \right.$$

Dabei sind $x \in \mathbb{R}^n$, $y \in \{0,1\}^p$, $c \in \mathbb{R}^p$, $B \in \mathbb{R}^{q \times p}$ sowie $A_1 \in \mathbb{R}^{m_1 \times n}$, $A_2 \in \mathbb{R}^{m_2 \times p}$, $a_1 \in \mathbb{R}^{m_1}$, $a_2 \in \mathbb{R}^{m_2}$. Des weiteren seien folgende Annahmen* erfüllt:

(A1) Die Menge $X \subset \mathbb{R}^n$ ist nichtleer, *kompakt* und *konvex*.

(A2) Die Funktionen

$$\begin{array}{rcll} f: \mathbb{R}^n & \longrightarrow & \mathbb{R} & \text{und} \\ g: \mathbb{R}^n & \longrightarrow & \mathbb{R}^q & \end{array}$$

sind *konvex* und *stetig-differenzierbar*.

(A3) Wird der y-Vektor auf einen beliebigen Wert aus Y fixiert, so gilt im Lösungspunkt des resultierenden NLP die folgende Form der *Slater'schen Constraint Qualification*: Sei

$$V := \left\{ y \in \{0,1\}^p \;\middle|\; \exists x \in X:\ g(x) + By \le 0 \right\}, \tag{2.1}$$

dann gibt es zu jedem $y \in Y \cap V$ auch ein $\tilde{x} \in X$, so dass

$$g(\tilde{x}) + By < 0$$

gilt.

*Die Gruppierung der Annahmen weicht geringfügig von der in der zitierten Literatur üblichen ab.

Man beachte, dass die Formulierung des $\mathrm{OA\text{-}MINLP}$ bereits folgende implizite Annahmen macht:

- Die Zielfunktion ist *separierbar* in x und y;
- die y-Variablen kommen sowohl in der Zielfunktion als auch in den Nebenbedingungen nur *linear* vor.

Ferner können mit der hier vorgestellten Form der Outer Approximation keine *nichtlinearen Gleichungsnebenbedingungen* behandelt werden. Insbesondere ist eine Umformulierung der Art

$$h(x) = 0 \quad \Longleftrightarrow \quad h(x) \leq 0 \ \wedge \ -h(x) \leq 0$$

nur im linearen Fall möglich – ansonsten verletzt entweder $h(x)$ oder $-h(x)$ die Konvexitätsbedingung (A2). Gleichungsnebenbedingungen müssen daher z. B. algebraisch eliminiert werden.

2.2 Lösungsstrategien

Outer Approximation beruht im Wesentlichen auf zwei Beobachtungen: Erstens erhält man bei fester Vorgabe eines $y^j \in Y$ ein konvexes, ungleichungsbeschränktes NLP – das sogenannte *Primalprogramm* –, welches vergleichsweise leicht zu lösen ist. Eine auf diese Art gewonnene Lösung stellt eine obere Schranke für die Lösung des $\mathrm{OA\text{-}MINLP}$ dar. Zweitens ist es durch die Konvexitäts- und Differenzierbarkeitsannahmen möglich, die Funktionen f und g durch ihre Gradienten nach „unten“ bzw. nach „außen“ abzuschätzen. Die Linearisierungen werden dabei jeweils am Lösungspunkt x^j des NLP durchgeführt. Auf diese Art erhält man ein *lineares* gemischt-ganzzahliges Programm (MILP) – in diesem Zusammenhang *Masterprogramm* genannt –, welches leichter als das ursprüngliche MINLP zu lösen ist und außerdem eine untere Schranke für dessen Lösung liefert. Der optimale ganzzahlige Vektor y^{j+1} des MILP wird nun wieder als Eingabe für das nächste NLP verwendet; es werden also immer abwechselnd NLPs und MILPs gelöst. Wir werden sehen, dass die dadurch berechneten oberen und unteren Schranken für das $\mathrm{OA\text{-}MINLP}$ nach *endlich* vielen Schritten zusammenfallen. Wir erhalten unter den gemachten Annahmen (A1), (A2) und (A3) also eine optimale Lösung des $\mathrm{OA\text{-}MINLP}$.

Primalprogramm

Zu einem fest vorgegebenen Vektor $y^j \in \{0,1\}^p$ können wir den optimalen Wert der kontunierlichen Variable x durch das folgende Primalprogramm ermitteln:

$$\mathrm{OA\text{-}NLP}(y^j) \quad \left\{ \begin{array}{ll} \min\limits_x & f(x) + c^T y^j \\ \mathrm{s.\,t.} & g(x) + By^j \leq 0 \\ & x \in X \end{array} \right.$$

Dabei sind zwei Fälle zu unterscheiden:

1. Fall: $y^j \in Y \cap V$ (wobei V wie in Gleichung (2.1) definiert ist). Dann besitzt $\mathrm{OA\text{-}NLP}(y^j)$ eine optimale Lösung $x^j \in X$, denn auf Grund der Kompaktheit von X (siehe (A1)) kann das Problem nicht unbeschränkt sein. Durch $f(x^j)+c^T y^j =: UBD$ ist damit auch eine obere Schranke für den minimalen Wert des $\mathrm{OA\text{-}MINLP}$ gegeben. Des Weiteren machen wir die unter Annahme (A2) evidente Feststellung, dass für alle $x^j \in X$ gilt:

$$\begin{array}{rcll} f(x) & \geq & f(x^j) + \nabla f(x^j)^T (x - x^j) & \forall\, x \in X\,, \\ g(x) & \geq & g(x^j) + \nabla g(x^j)^T (x - x^j) & \forall\, x \in X\,, \end{array}$$

wobei $\nabla f(x^j)$ bzw. $\nabla g(x^j)$ die $n \times 1$- bzw. $n \times p$-Jacobimatrix von f bzw. g an der Stelle x^j bezeichnen. Außerdem gilt folgender

Satz 2.1 (vgl. [4, S. 148])
Unter den Annahmen (A1), (A2) und insbesondere (A3) gilt für alle $x^j \in X$:

$$\left| \begin{array}{ll} \min\limits_x & f(x) + c^T y^j \\ s.\,t. & g(x) + By^j \leq 0 \\ & x \in X \end{array} \right| = \left| \begin{array}{ll} \min\limits_x & f(x^j) + \nabla f(x^j)^T (x - x^j) + c^T y^j \\ s.\,t. & g(x^j) + \nabla g(x^j)^T (x - x^j) + By^j \leq 0 \\ & x \in X \end{array} \right|$$

Es genügt sogar, nur die Linearisierungen der im Punkt (x^j, y^j) aktiven Ungleichungen zu berücksichtigen.

Durch Einführen der Hilfsvariablen η erhält man eine äquivalente Formulierung des Programms aus Satz 2.1:

$$\left|\begin{array}{ll} \min\limits_{x} & f(x^j) + \nabla f(x^j)^T(x - x^j) + c^T y^j \\ \text{s.t.} & g(x^j) + \nabla g(x^j)^T(x - x^j) + By^j \leq 0 \\ & x \in X \end{array}\right| = \left|\begin{array}{ll} \min\limits_{x,\,\eta} & \eta + c^T y^j \\ \text{s.t.} & f(x^j) + \nabla f(x^j)^T(x - x^j) \leq \eta \\ & g(x^j) + \nabla g(x^j)^T(x - x^j) + By^j \leq 0 \\ & x \in X \\ & \eta \in \mathbb{R} \end{array}\right|$$

Eine geometrische Veranschaulichung der Linearisierungen (jeweils auf die x-Komponenten projeziert) ist in den Abbildungen 2 und 3 gegeben: Durch die geforderte Konvexität wird die Zielfunktion durch eine Tangente (bzw. eine tangentiale Hyperebene) im Punkt x^j nach unten abgeschätzt. Außerdem wird das zulässige Gebiet nun nicht mehr durch die i. Allg. nichtlinearen Nebenbedingungen g begrenzt, sondern durch den Schnitt endlich vieler Halbebenen dargestellt†. Das zulässige Gebiet bleibt dabei stets mindestens gleich groß, in der Regel wird es vergrößert. Dadurch wird die Optimallösung des OA-MINLP gleich in zweifacher Hinsicht unterschätzt.

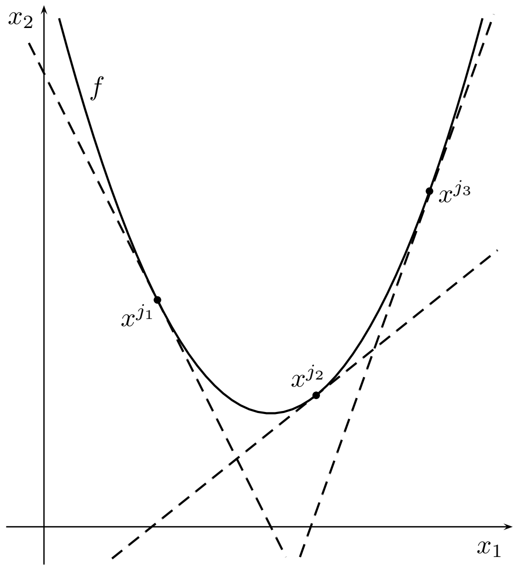

Abbildung 2: Linearisierung – und damit Abschätzung nach unten – der konvexen Zielfunktion f in verschiedenen Punkten x^{j_i}.

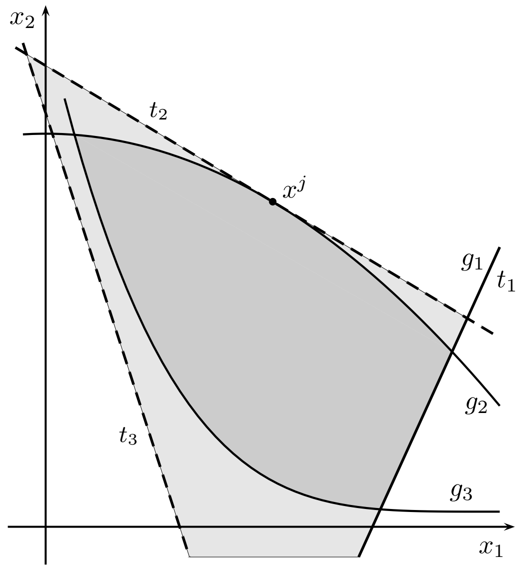

Abbildung 3: Das konvexe, durch die Nebenbedingungen g_i begrenzte, zulässige Gebiet. Durch Linearisierung der g_i im Punkt x^j erhält man die Geraden t_i, die das zulässige Gebiet nach außen abschätzen.

2. Fall: $y^j \in Y \setminus V$. Dann besitzt das OA-NLP(y^j) keine Lösung, da die Nebenbedingungen inkonsistent sind. In diesem Falle muss dafür gesorgt werden, dass die unzulässige 0-1-Belegung von y^j nicht noch einmal gewählt wird. Dazu wurde ursprünglich das Hinzufügen einer zusätzlichen Bedingung zum OA-MINLP vorgeschlagen:

Satz 2.2 (Integer Cut, vgl. [2, S. 320 f.])
Sei $y^j = (y_1^j, \ldots, y_p^j)^T \in \{0,1\}^p$ mit den Indexmengen

$$B^j := \left\{ l \mid y_l^j = 1 \right\} \quad \textit{und} \quad NB^j := \left\{ l \mid y_l^j = 0 \right\}$$

gegeben. Dann wird die Ungleichung

$$\sum_{l \in B^j} y_l - \sum_{l \in NB^j} y_l \leq \left| B^j \right| - 1$$

nur vom Vektor y^j verletzt und von allen anderen Vektoren $y \in \{0,1\}^p \setminus \left\{ y^j \right\}$ erfüllt.

Wie wir in Abschnitt 3 sehen werden, kann dieses Vorgehen zu Problemen führen. Daher wird im Rahmen der Verallgemeinerten Outer Approximation anders mit der Unzulässigkeit des NLP umgegangen.

†Man beachte, dass Linearisierungen von *nichtaktiven*, nichtlinearen Ungleichungen keine Tangenten sind.

Masterprogramm

Zum Aufstellen des Masterprogramms formulieren wir unser OA-MINLP zunächst in eine äquivalente Darstellung um:

$$\text{OA-MINLP}_{\Leftrightarrow} \quad \begin{cases} \min\limits_{y} & \text{OA-NLP}(y) \\ \text{s.t.} & y \in Y \cap V \end{cases}$$

Das Aufteilen der Minimumbildung ist dabei offenbar unproblematisch und der Ausschluss aller $y \in Y \setminus V$ ist nur eine Präzisierung des Problems: Es sei an dieser Stelle noch einmal betont, dass alle Vektoren $y \in Y$, für die es nicht mindestens einen zulässigen Vektor $x \in X$ gibt, nicht als Lösung des OA-MINLP in Frage kommen.

Daher ergibt sich folgende (ebenfalls äquivalente) Formulierung des OA-MINLP :

$$\text{OA-MINLP}_{\Leftrightarrow} \quad \begin{cases} \min\limits_{x,\,y,\,\eta} & \eta + c^T y \\ \text{s.t.} & \eta \geq f(x^j) + \nabla f(x^j)(x - x^j) \qquad \forall\, j \in F \\ & 0 \geq g(x^j) + \nabla g(x^j)(x - x^j) + By \quad \forall\, j \in F \\ & x \in X \\ & y \in Y \cap V \\ & \eta \in \mathbb{R} \end{cases}$$

wobei die Indexmenge F definiert ist als

$$F := \left\{ j \;\middle|\; \text{das Primalprogramm OA-NLP}(y^j) \text{ ist zulässig mit Optimallösung } x^j \right\} .$$

Das so umformulierte Problem ist nun wie gewünscht ein MILP; allerdings mit zwei versteckten Schwierigkeiten: Zum einem ist V nicht explizit bekannt. In [2] wird behauptet, dass man $y \in Y \cap V$ durch $y \in Y$ ersetzen kann, wenn man alle anderen Bedingungen im Programm belässt[‡]. Genau dies ist aber die zweite Schwierigkeit, denn das Programm enthält bis zu $2^p(q+1)$ Ungleichungsnebenbedigungen! Um sie überhaupt formulieren zu können, müssten alle 2^p OA-NLP(y^j) vorab gelöst werden, was dem naiven Durchprobieren aller 0-1-Kombinationen entsprechen würde.

Das Masterprogramm wird daher als Relaxierung des OA-MINLP$_{\Leftrightarrow}$ formuliert, d. h. die meisten Nebenbedingungen werden ignoriert: In jeder Iteration wird ein OA-NLP(y^j) gelöst und im Falle der Zulässigkeit um den Lösungsvektor x^j linearisiert und dem Masterprogramm die entsprechenden Nebenbedingungen hinzugefügt. Im Falle der Unzulässigkeit des OA-NLP(y^j) wird y^j mittels der in Satz 2.2 definierten Ungleichung für weitere Iterationen eliminiert. Dies ergibt folgendes (relaxierte) Masterprogramm für die i. Iteration:

$$\text{OA-MASTER}^i \quad \begin{cases} \min\limits_{x,\,y,\,\eta} & \eta + c^T y \\ \text{s.t.} & \eta \geq f(x^j) + \nabla f(x^j)(x - x^j) \qquad \forall\, j \in F^i \\ & 0 \geq g(x^j) + \nabla g(x^j)(x - x^j) + By \quad \forall\, j \in F^i \\ & \sum\limits_{l \in B^k} y_l - \sum\limits_{l \in NB^k} y_l \leq \left|B^k\right| - 1 \qquad \forall\, k \in \overline{F^i} \\ & \eta + c^T y < UBD \\ & x \in X \\ & y \in Y \\ & \eta \in \mathbb{R} \end{cases}$$

wobei die Indexmengen wie folgt definiert sind:

$$\begin{aligned} F^i &:= \left\{ j \leq i \;\middle|\; \text{das Primalprogramm OA-NLP}(y^j) \text{ ist zulässig mit Optimallösung } x^j \right\} , \\ \overline{F^i} &:= \left\{ k \leq i \;\middle|\; \text{das Primalprogramm OA-NLP}(y^k) \text{ ist unzulässig} \right\} . \end{aligned}$$

[‡]Fletcher und Leyffer präsentierten dazu in [3] ein Gegenbeispiel. Im gleichen Paper erläutern sie (mit Beweis) eine alternative Vorgehensweise, die wir in Abschnitt 3 behandeln werden.

Zusätzlich wurde die Bedingung eingefügt, dass die neue Lösung einen niedrigeren Zielfunktionswert als die aktuelle obere Schranke UBD haben muss. Das Erreichen der Optimallösung des $\mathrm{OA\text{-}MINLP}$ erkennt man dann an der Unzulässigkeit von $\mathrm{OA\text{-}MASTER}^i$.

2.3 Algorithmus

Nach der theoretischen Vorarbeit können wir nun den ersten Outer-Approximation-Algorithmus von Duran und Grossmann (siehe [2]) formulieren:

1. - Setze $i := 1$, $F^0 := \overline{F^0} := \emptyset$ und die obere Schranke $UBD := \infty$.
   - Wähle einen ganzzahligen Vektor $y^0 \in Y$ (oder $y^0 \in Y \cap V$ falls verfügbar).
2. Löse das Primalprogramm $\mathrm{OA\text{-}NLP}(y^i)$. Dabei muss einer der folgenden beiden Fälle auftreten:
   (a) $\mathrm{OA\text{-}NLP}(y^i)$ ist zulässig mit Optimallösung (x^i, y^i):
      - Falls $f(x^i) + c^T y^i < UBD$: Setze $UBD := f(x^i) + c^T y^i$, $x^* := x^i$ und $y^* := y^i$.
      - Setze $F^i := F^{i-1} \cup \{i\}$ und $\overline{F^i} := \overline{F^{i-1}}$, d. h. füge die um Punkt x^i linearisierten Ungleichungsnebenbedingungen hinzu.
      - Gehe zu Schritt 3.

   (b) $\mathrm{OA\text{-}NLP}(y^i)$ ist unzulässig: Setze $F^i := F^{i-1}$ und $\overline{F^i} := \overline{F^{i-1}} \cup \{i\}$, d. h. füge einen Integer Cut gemäß Satz 2.2 hinzu[§].
3. Löse das relaxierte Masterprogramm $\mathrm{OA\text{-}MASTER}^i$. Dabei muss einer der folgenden beiden Fälle auftreten:
   (a) $\mathrm{OA\text{-}MASTER}^i$ ist zulässig mit Optimallösung $(\hat{x}^{i+1}, \hat{y}^{i+1})$:
      - Setze $y^{i+1} := \hat{y}^{i+1}$ und $i := i + 1$.
      - Gehe zu Schritt 2.

   (b) $\mathrm{OA\text{-}MASTER}^i$ ist unzulässig: STOPP, Optimallösung (x^*, y^*) gefunden (falls $UBD < \infty$, ansonsten ist $\mathrm{OA\text{-}MINLP}$ unzulässig).

Es bleibt noch zu klären, wie man einen ganzzahligen Startvektor y^0 ermitteln kann. Eine Möglichkeit besteht darin, die *NLP-Relaxierung* zu lösen, die man durch Weglassen der Ganzzahligkeitsforderung aus $\mathrm{OA\text{-}MINLP}$ erhält. Genauer bedeutet dies, dass man die Bedingung

$$y \in \{0,1\}^p \quad \text{durch} \quad y \in [0,1]^p$$

ersetzt. In der gefundenen Lösung werden die gebrochenen Komponenten von y gerundet. Dieses Vorgehen hat ferner den Vorteil, dass man im Falle der Unzulässigkeit der NLP-Relaxierung auch die Unzulässigkeit des MINLP gezeigt hat.

Schließlich wollen wir uns noch überlegen, dass der Algorithmus tatsächlich konvergiert: Die oberen Schranken, die wir durch die Lösung der Primalprogramme erhalten, sind nach Konstruktion monoton fallend. Die unteren Schranken sind monoton steigend, da wir den Masterprogrammen sukzessive zusätzliche Bedingungen hinzufügen. Die Integer Cuts bzw. die Bedingung $\eta + c^T y < UBD$ stellen ferner sicher, dass keine unzulässige bzw. zulässige 0-1-Belegung $y^i \in Y$ zweimal untersucht wird. Wegen $|Y| \leq 2^p < \infty$ muss der Algorithmus daher nach endlich vielen Schritten terminieren. Besitzt das $\mathrm{OA\text{-}MINLP}$ zulässige Lösungen, so bedeutet Konvergenz des Algorithmus, dass die untere Schranke des Masterprogramms nicht kleiner als die obere Schranke UBD des Primalprogrammes ist. Da es sich um theoretisch beweisbare Schranken für das $\mathrm{OA\text{-}MINLP}$ handelt, hat man in diesem Falle eine Optimallösung gefunden. Somit ergibt sich folgender

Satz 2.3 (vgl. [2, S. 319])
Unter den Annahmen (A1), (A2) und (A3) terminert der Outer-Approximation-Algorithmus nach endlich vielen Schritten und berechnet eine Optimallösung des $\mathrm{OA\text{-}MINLP}$, falls dieses zulässig ist.

[§] Die in Abschnitt 3 eingeführten Linearisierungen um unzulässige Punkte wurden in [2] zwar bereits erwähnt, ohne diese Technik jedoch näher zu beschreiben.

3 Verallgemeinerte Outer Approximation

3.1 Problemformulierung

Fletcher und Leyffer stellten 1994 in [3] eine Verallgemeinerung des Outer-Approximation-Ansatzes vor. Wesentliche Verbesserungen waren:

- Die ganzzahligen Variablen y dürfen auch *nichtlinear* und *nichtseparierbar* in Zielfunktion und Ungleichungsnebenbedingungen vorkommen;
- *unzulässige* Primalprogramme werden auf theoretisch gesicherter Grundlage zur Verschärfung der Relaxierung des Masterprogramms benutzt.

Die allgemeine Problemformulierung lautet:

$$\text{GOA-MINLP} \quad \begin{cases} \min\limits_{x,\,y} & f(x,y) \\ \text{s.t.} & g(x,y) \le 0 \\ & x \in X := \left\{x \in \mathbb{R}^n \;\middle|\; A_1 x \le a_1\right\} \\ & y \in Y := \left\{y \in \{0,1\}^p \;\middle|\; A_2 y \le a_2\right\} \end{cases}$$

Dabei sind $x \in \mathbb{R}^n$, $y \in \{0,1\}^p$ sowie $A_1 \in \mathbb{R}^{m_1 \times n}$, $A_2 \in \mathbb{R}^{m_2 \times p}$, $a_1 \in \mathbb{R}^{m_1}$, $a_2 \in \mathbb{R}^{m_2}$. Außerdem werden wieder die Annahmen (A1), (A2) und (A3) wie in Abschnitt 2 gemacht werden, wobei folgende Anpassung vorzunehmen ist:

(A2) Die Funktionen

$$\begin{array}{rcl} f: \; \mathbb{R}^n \times \mathbb{R}^p & \longrightarrow & \mathbb{R} \quad \text{und} \\ g: \; \mathbb{R}^n \times \mathbb{R}^p & \longrightarrow & \mathbb{R}^q \end{array}$$

sind *konvex* und *stetig-differenzierbar*.

Es sei bemerkt, dass die Verallgemeinerte Outer Approximation auch das Lösen von ganzzahlig-nichtlinearen Programmen (INLP) ermöglicht; in diesem Kontext entspricht dies dem Spezialfall $n = 0$.

3.2 Unzulässige Primalprogramme

Das Primalprogramm

$$\text{GOA-NLP}(y^j) \quad \begin{cases} \min\limits_{x} & f(x,y^j) \\ \text{s.t.} & g(x,y^j) \le 0 \\ & x \in X \end{cases}$$

ergibt sich völlig analog zur Darstellung der „gewöhnlichen" Outer Approximation in Abschnitt 2. Ein entscheidender Unterschied besteht jedoch in der Behandlung von unzulässigen Primalprogrammen, d. h. im Falle von $y^j \in Y \setminus V$.

Die meisten NLP-Löser verwenden zum Auffinden eines zulässigen Startvektors einen Phase-I-Ansatz. Dazu versuchen sie, die Verletzungen der Ungleichungsbedingungen zu minimieren, etwa mittels

$$\begin{array}{ll} \min\limits_{x} & \sum\limits_{i=1}^{q} g_i^+(x,y^j) \\ \text{s.t.} & x \in X \end{array} \qquad \text{oder} \qquad \begin{array}{ll} \min\limits_{x} & \max\limits_{1 \le i \le q} g_i^+(x,y^j) \\ \text{s.t.} & x \in X \end{array}$$

Dabei wird die Definition $g_i^+ := \max\{0, g_i\}$ benutzt und y^j ist fest vorgegeben.

Neben der l_1- bzw. l_∞-Norm-Minimierung kann auch folgender Ansatz angewandt werden: Sobald eine Ungleichung erfüllt ist, wird sie in die Nebenbedingungen des Phase-I-Programms aufgenommen. Dadurch bleibt eine einmal erfüllte Ungleichung gültig und es werden nur noch die Ungleichungsverletzungen der übrigen (noch nicht erfüllten) Nebenbedingungen minimiert.

Gleichgültig welche Variante gewählt wurde, am Ende von Phase I wurde formal das folgende Unzulässigkeitsprogramm gelöst:

$$\text{GOA-FEAS}(y^k)\quad \begin{cases} \min\limits_{x} & \sum\limits_{i\in\overline{I}} w_i^k g_i^+(x,y^k) \\ \text{s.t.} & g_i(x,y^k)\le 0 \qquad \forall\, i\in I \\ & x\in X \end{cases}$$

Hierbei bezeichnen I bzw. $\overline{I}$ die (veränderlichen) Mengen der momentan gültigen bzw. ungültigen Ungleichungsnebenbedingungen und $w_i^k \geq 0$ sind Gewichte, die nicht alle gleich 0 sind. Der Optimalwert von $\text{GOA-FEAS}(y^k)$ ist genau dann größer als 0, wenn das zugehörige Primalprogramm unzulässig ist. Es sei aber betont, dass $\text{GOA-FEAS}(y^k)$ nicht direkt, sondern implizit mittels einer Phase-I-Methode gelöst wird. Daher sind die Gewichte im Wesentlichen ein theoretisches Konstrukt und müssen nicht explizit gewählt werden.

Der folgende Satz liefert uns eine wichtige Eigenschaft von $\text{GOA-FEAS}(y^k)$, die für die Verallgemeinerte Outer Approximation eine zentrale Bedeutung besitzt:

Satz 3.1 (vgl. [3, S. 331 f.])
Das Primalprogramm $\mathit{GOA\text{-}NLP}(y^k)$ sei unzulässig, so dass x^k das Unzulässigkeitsprogramm $\mathit{GOA\text{-}FEAS}(y^k)$ mit

$$\sum_{i\in\overline{I}} w_i^k g_i^+(x,y^k) > 0$$

löst. Dann ist $y=y^k$ für alle $x\in X$ unzulässig in den folgenden Nebenbedingungen:

$$0 \geq g_i(x^k,y^k) + \nabla g_i(x^k,y^k)^T \begin{pmatrix} x-x^k \\ y-y^k \end{pmatrix} \qquad \forall\, i\in\{1,\dots,p\}\ .$$

Auch hier genügt es, nur die Linearisierungen der im Punkt (x^k,y^k) aktiven Ungleichungen zu berücksichtigen.

3.3 Verbessertes Masterprogramm

Beim Aufstellen das Masterprogramms geht man nun wie folgt vor: Wir linearisieren wie bisher um alle Optimallösungen der zulässigen $\text{OA-NLP}(y^j)$. Im Falle der Unzulässigkeit lösen wir mittels Phase I das entsprechende Unzulässigkeitsprogramm und linearisieren dann ebenfalls, diesmal aber um die Optimallösung von $\text{GOA-FEAS}(y^k)$. Dadurch erhalten wir

$$\text{GOA-MINLP}_{\Leftrightarrow}\quad \begin{cases} \min\limits_{x,\,y,\,\eta} & \eta \\ \text{s.t.} & \eta \geq f(x^j,y^j) + \nabla f(x^j,y^j)^T \begin{pmatrix} x-x^j \\ y-y^j \end{pmatrix} \qquad \forall\, j\in F \\ & 0 \geq g(x^j,y^j) + \nabla g(x^j,y^j)^T \begin{pmatrix} x-x^j \\ y-y^j \end{pmatrix} \qquad \forall\, j\in F \\ & 0 \geq g(x^k,y^k) + \nabla g(x^k,y^k)^T \begin{pmatrix} x-x^k \\ y-y^k \end{pmatrix} \qquad \forall\, k\in \overline{F} \\ & x\in X \\ & y\in Y \\ & \eta\in\mathbb{R} \end{cases}$$

mit den Indexmengen

$$\begin{aligned} F &:= \left\{ j \;\middle|\; \text{das Primalprogramm GOA-NLP}(y^j) \text{ ist zulässig mit Optimallösung } x^j \right\}, \\ \overline{F} &:= \left\{ k \;\middle|\; \text{das Primalprogramm OA-NLP}(y^k) \text{ ist unzulässig} \right. \\ &\qquad \left. \text{und } x^k \text{ ist Optimallösung von GOA-FEAS}(y^k) \right\}. \end{aligned}$$

Man beachte, dass im Gegensatz zum $\text{OA-MINLP}_{\Leftrightarrow}$ nicht mehr $y\in Y\cap V$ gefordert wird. Wir hatten bereits erwähnt, dass im Falle der „gewöhnlichen“ Outer Approximation das Weglassen dieser Bedingung die Inäquivalenz von OA-MINLP und $\text{OA-MINLP}_{\Leftrightarrow}$ zur Folge haben kann. Dass dies im Falle der Verallgemeinerten Outer Approximation nicht geschieht, zeigt folgender

Satz 3.2 (vgl. [3, S. 336])
Unter den Annahmen (A1), (A2) und (A3) ist sind die Programme $GOA\text{-}MINLP$ und $GOA\text{-}MINLP_{\Leftrightarrow}$ äquivalent. Das soll heißen, dass (x^*, y^*) genau dann eine Lösung von $GOA\text{-}MINLP$ ist, wenn (x^*, y^*) eine Lösung von $GOA\text{-}MINLP_{\Leftrightarrow}$ ist.

Das Masterprogramm erhält man nun wieder, indem man $\text{GOA-MINLP}_{\Leftrightarrow}$ relaxiert. Auf Integer Cuts kann wegen Satz 3.2 verzichtet werden:

$$\text{GOA-MASTER}^i \left\{ \begin{array}{lll} \min\limits_{x,\,y,\,\eta} & \eta & \\ & \eta < UBD & \\ & \eta \geq f(x^j, y^j) + \nabla f(x^j, y^j)^T \begin{pmatrix} x - x^j \\ y - y^j \end{pmatrix} & \forall\, j \in F^i \\ & 0 \geq g(x^j, y^j) + \nabla g(x^j, y^j)^T \begin{pmatrix} x - x^j \\ y - y^j \end{pmatrix} & \forall\, j \in F^i \\ & 0 \geq g(x^k, y^k) + \nabla g(x^k, y^k)^T \begin{pmatrix} x - x^k \\ y - y^k \end{pmatrix} & \forall\, k \in \overline{F^i} \\ & x \in X & \\ & y \in Y & \\ & \eta \in \mathbb{R} & \end{array} \right.$$

Wieder bezeichnen F^i und $\overline{F^i}$ die Einschränkungen von F und $\overline{F}$ auf die ersten i Iterationen und es gilt jeweils $UBD := \min\limits_{j \in F^i} f(x^j, y^j)$.

Auf eine Ausformulierung des Verallgemeinerten-Outer-Approximation-Algorithmus wird an dieser Stelle verzichtet, da nur die zuvor diskutierten Anpassungen in den (unzulässigen) Primal- und Masterprogrammen vorzunehmen sind. Auch hier ergibt sich der

Satz 3.3 (vgl. [3, S. 337 f.])
Unter den Annahmen (A1), (A2) und (A3) terminert der Verallgemeinerte-Outer-Approximation-Algorithmus nach endlich vielen Schritten mit einer Optimallösung des $GOA\text{-}MINLP$ oder der Meldung, dass dieses zulässig ist.

4 Implementierungsaspekte

In den vorangegangenen Abschnitten haben wir uns primär mit der theoretischen Herleitung und Darstellung der Outer Approximation[¶] befasst. Nun sollen kurz einige Aspekte der Implementierung angesprochen werden, wobei wir auch den in [1] beschriebenen Branch&Cut-Algorithmus für konvexe 0-1-MINLP-Probleme vorstellen.

4.1 Lösen von Primal- und Masterprogramm

Der Outer-Approximation-Algorithmus löst in jeder Iteration sowohl ein NLP als auch ein MILP. Das NLP besitzt unter den gemachten Annahmen eine konvexe Zielfunktion und konvexe Ungleichungsnebenbedingungen. Es ist daher – etwa mit einem SQP-Algorithmus – relativ leicht zu lösen; Gleiches gilt auch für das korrespondierende Unzulässigkeitsprogramm. Dennoch kann das Lösen des NLP sehr aufwendig werden, wenn es eine große Zahl an Nebenbedingungen umfasst.

Für die Gesamtlaufzeit des Outer-Approximation-Algorithmus ist eine effiziente Berechnung des MILP besonders entscheidend. Kritisch ist vor allem die Tatsache, dass in *jeder* Iteration bis zu $q+1$ neue Nebenbedingungen hinzukommen – nämlich die Linearisierungen von f und g um die Punkte (x^j, y^j) bzw. (x^k, y^k). Daher kann es eine spürbare Entlastung darstellen, wenn man nur die Linearisierungen der *aktiven* Ungleichungen hinzufügt (wir haben weiter oben erwähnt, dass dies ausreicht). Die Verringerung der Zahl der Nebenbedingungen hat jedoch den Nachteil, dass die Relaxierung dadurch ggf. schwächere untere Schranken liefert und damit mehr Iterationen erforderlich werden.

Eine Standardmethode zum Lösen des Masterprogramm stellt der Branch&Bound-Ansatz dar. Hierbei wird das Problem durch Fixieren gewisser y-Komponenten auf 0 oder 1 in Teilprobleme zerlegt und diese nacheinander gelöst. Durch Berechnung von oberen und unteren Schranken hofft man, dass die meisten Teilprobleme frühzeitig verworfen werden können. Dazu wird ein zu bearbeitendes Teilproblem ausgewählt und dessen Ganzzahligkeitsforderung fallengelassen. Das resultierende LP wird mit der Simplex- oder Innere-Punkte-Methode gelöst, wobei einer der folgenden 4 Fälle eintritt:

1. Der Zielfunktionswert der optimalen Lösung liegt unter der aktuellen oberen Schranke, aber erfüllt die Ganzzahligkeitsbedingungen nicht: Dann wird eine Variable ausgewählt, die die Ganzzahligkeitsbedingung verletzt, und zwei neue Teilprobleme erzeugt, indem man ihren Wert einmal auf 0 und einmal auf 1 fixiert.

2. Der Zielfunktionswert der optimalen Lösung liegt unter der aktuellen oberen Schranke und erfüllt die Ganzzahligkeitsbedingungen: Dann hat man eine neue obere Schranke gefunden. Das Teilproblem wird jedoch nicht weiter betrachtet, da seine LP-Relaxierung auch eine untere Schranke geliefert hat, die nun nicht weiter verbessert werden kann.

3. Der Zielfunktionswert der optimalen Lösung liegt über der aktuellen oberen Schranke: Dann können zusätzliche Bedingungen wie das Fixieren von Variablen nicht zu besseren Zielfunktionswerten führen; das Teilproblem wird daher verworfen.

4. Das Teilproblem ist unzulässig: Auch hier werden zusätzliche Bedingungen die Zulässigkeit nicht wieder herstellen; das Teilproblem wird verworfen.

Das MILP ist gelöst, sobald kein Teilproblem mehr zur Bearbeitung ansteht, weil alle gemäß den Fällen 2 bis 4 verworfen wurden. Die dann aktuelle obere Schranke ist die Optimallösung.

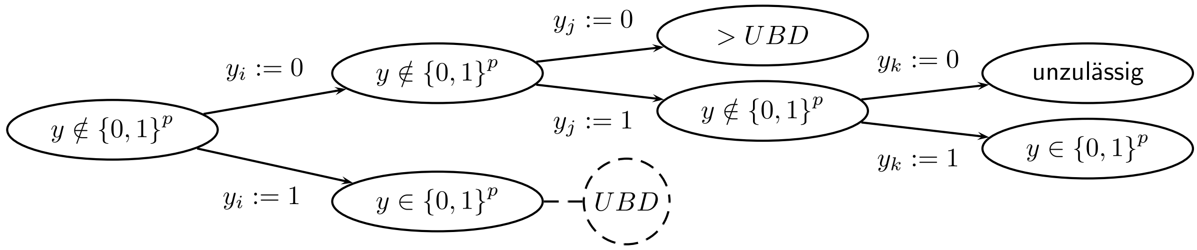

Abbildung 4: Illustration der Verzweigungen im Zuge des Branch&Bound-Verfahrens.

[¶]Der Begriff umfasst im Folgenden beide vorgestellten Varianten, wobei natürlich i. Allg. die mächtigere Verallgemeinerte Outer Approximation vorzuziehen ist.

4.2 Outer Approximation mit nur einem Masterprogramm

Um den erheblichen Aufwand zu verringern, den das Lösen eines Masterprogramms in jeder Iteration mit sich bringt, wurden in [6] die Outer Approximation mit dem Branch&Bound-Ansatz verschmolzen: Es wird *nur das anfängliche* Masterprogramm GOA-BBMASTER^0 aufgestellt und mittels eines Branch&Bound-Verfahrens gelöst. Trifft man dabei auf ein Teilproblem, dessen y-Variablen die Ganzzahligkeitsbedingungen erfüllen, so wird – anders als in Abschnitt 4.1 unter 2. beschrieben – die Betrachtung des Teilproblems nicht beendet, sondern das zugehörige Primal- bzw. Unzulässigkeitsprogramm gelöst. Um die dadurch erhaltene Lösung werden wie gewohnt Zielfunktion und Nebenbedingungen linearisiert und diese zusätzlichen Bedingungen zu *allen* noch *offenen* Teilproblemen hinzugefügt; und zwar einschließlich des Teilproblems mit dem ganzzahlig-zulässigen y-Vektors. Dessen 0-1-Belegung wird nämlich durch Aktualisieren der oberen Schranke UBD unzulässig und es kann auch in diesem Teilproblem weiter nach noch besseren Lösungen gesucht werden. Die Lösung aller Outer-Approximation-Masterprogramme (und auch der Primal- bzw. Unzulässigkeitsprogramme) erfolgt somit *implizit* während des Abarbeitens des Branch&Bound-Baumes zur Lösung des *ersten* und damit *einzigen* Masterprogramms!

Dieses Vorgehen hat jedoch einen kleinen Haken: Die impliziten Masterprogramme in den Teilproblemen des Branch&Bound-Baumes werden nun nicht mehr bis zur Optimalität gelöst, sondern abgebrochen, sobald ein ganzzahlig-zulässiger y-Vektor mit verbessertem Zielfunktionswert gefunden wurde. Da anschließend ein Primal- bzw. Unzulässigkeitsprogramm mit diesem i. Allg. suboptimalen y-Vektor gelöst wird, steigt deren Zahl gegenüber der reinen Outer Approximation an. Dieses häufigere Linearisieren führt außerdem zu einem stärkeren Anwachsen der Zahl der Nebenbedingungen in den noch offenen Teilproblemen. Daher wurde von Akrotirianakis et. al in [1] ein zusätzliches Feature eingebaut: Sie erweitern das Branch&Bound-Verfahren zu einem Branch&Cut-Verfahren, indem sie die nach ihrem Urheber benannten Gomory Cuts (erstmals in [5] beschrieben) benutzen. Wir können an dieser Stelle nur kurz zusammenfassend bemerken, dass die Gomory Cuts gebrochene Optimallösungen der LP-Relaxierung abschneiden, ohne dass dabei eine zulässige ganzzahlige Lösung verloren geht. Damit konnte die kombinierte Branch&Bound-Outer-Approximation noch weiter verbessert werden, so dass sie der reinen Outer Approximation in den meisten Fällen spürbar überlegen ist, wie numerische Tests zeigen. Wir wollen ihren Algorithmus zum Abschluss kurz darstellen.

Das anfängliche Masterprogramm GOA-BBMASTER^0 unterscheidet sich nicht vom in Abschnitt 3 vorgestellten Masterprogramm GOA-MASTER^0; ebenso bleiben $\text{GOA-NLP}(y^i)$ und $\text{GOA-FEAS}(y^i)$ unverändert. Der einzige Unterschied besteht in den nun impliziten Masterprogrammen während des Abarbeitens des Branch&Bound-Baumes:

$$\text{GOA-BBMASTER}^i \left\{ \begin{array}{lll} \min\limits_{x,\,y,\,\eta} & \eta & \\ & \eta < UBD & \\ & \eta \geq f(x^j,y^j) + \nabla f(x^j,y^j)^T \begin{pmatrix} x-x^j \\ y-y^j \end{pmatrix} & \forall\, j \in \hat{F}^i \\ & 0 \geq g(x^j,y^j) + \nabla g(x^j,y^j)^T \begin{pmatrix} x-x^j \\ y-y^j \end{pmatrix} & \forall\, j \in \hat{F}^i \\ & 0 \geq g(x^k,y^k) + \nabla g(x^k,y^k)^T \begin{pmatrix} x-x^k \\ y-y^k \end{pmatrix} & \forall\, k \in \overline{\hat{F}^i} \\ & \Gamma_1 x + \Gamma_2 y \leq \xi & \\ & x \in X & \\ & y \in Y \cap FIX^i & \\ & \eta \in \mathbb{R} & \end{array} \right.$$

Die Bedingungen $\Gamma_1 x + \Gamma_2 y \leq \xi$ repräsentieren die Gomory Cuts und die Menge FIX^i sorgt für die 0-1-Fixierungen des i. Teilproblems. Die Indexmengen wurden wie folgt angepasst:

$$\hat{F}^i := \big\{ j \leq i \;\big|\; \text{das Teilproblem } j \text{ besaß eine ganzzahlig-zulässige Lösung } y^j \text{ und das zugehörige Primalprogramm GOA-NLP}(y^j) \text{ war zulässig mit Optimallösung } x^j \big\}\,.$$

$$\overline{\hat{F}^i} := \big\{ k \leq i \;\big|\; \text{das Teilproblem } k \text{ besaß eine ganzzahlig-zulässige Lösung } y^k\text{, das zugehörige Primalprogramm GOA-NLP}(y^k) \text{ war unzulässig und GOA-FEAS}(y^k) \text{ hatte die Optimallösung } x^k \big\}\,.$$

Damit können wir nun den in [1] vorgestellten Branch&Cut-Algorithmus für konvexe MINLP-Probleme präsentieren:

1. Wähle $y^0 \in \{0,1\}^p$ und setze $\hat{F}^0 := \overline{\hat{F}^0} := \emptyset$ sowie $i := 1$.
2. Aufstellen des Start-Masterprogramms:
   (a) Falls $\text{GOA-NLP}(y^0)$ zulässig mit Optimum x^0: Setze $\hat{F}^0 := \{0\}$ und $UBD := f(x^0, y^0)$.
   (b) Falls $\text{GOA-NLP}(y^0)$ unzulässig:
      - Löse Unzulässigkeitsprogramm $\text{GOA-FEAS}(y^0)$ und erhalte Optimum x^0.
      - Setze $\overline{\hat{F}^0} := \{0\}$ und $UBD := \infty$.
   (c) Linearisiere f und g um den Punkt (x^0, y^0) und erhalte Start-Masterprogramm GOA-BBMASTER^0.
   (d) Initialisiere Teilproblemliste mit GOA-BBMASTER^0.
3. Setze $i := i + 1$ und teste ob die Teilproblemliste leer ist?
   (a) Ja: STOPP! Ist $UBD < \infty$, so ist (x^*, y^*) Optimallösung des GOA-MINLP, ansonsten ist dieses unzulässig.
   (b) Nein: Entferne ein Teilproblem aus der Liste und bearbeite es in Schritt 4 als Teilproblem i.
4. Löse die LP-Relaxierung von GOA-BBMASTER^i und erhalte Optimallösung $(\hat{x}, \hat{y}, \hat{\eta})$.
   (a) Falls $\hat{y} \in \{0,1\}^p$:
      - Setze $y^i := \hat{y}$ und löse $\text{GOA-NLP}(y^i)$ bzw. $\text{GOA-FEAS}(y^i)$ und erhalte Optimallösung x^i.
      - Linearisiere f und g um den Punkt (x^i, y^i) und setze $\hat{F}^i := \hat{F}^{i-1} \cup \{i\}$ bzw. $\overline{\hat{F}^i} := \overline{\hat{F}^{i-1}} \cup \{i\}$, d. h. füge Linearisierung zu GOA-BBMASTER^i und allen übrigen Teilproblem hinzu.
      - Füge GOA-BBMASTER^i wieder in die Teilproblemliste ein.
      - Falls $\text{GOA-NLP}(y^i)$ zulässig war mit $f(x^i, y^i) < UBD$: Setze $(x^*, y^*) := (x^i, y^i)$ und $UBD := f(x^i, y^i)$ und lösche alle Teilprobleme mit $\hat{\eta} > UBD$ aus der Liste.
      - Gehe zu Schritt 3.
   (b) Falls $\hat{y} \notin \{0,1\}^p$:
      - Verzweige an einer gebrochenen 0-1-Variable, füge zwei neue Teilprobleme in die Liste ein und gehe zu Schritt 3[‖].

        oder
      - Füge Gomory Cuts zu GOA-BBMASTER^i hinzu und gehe zu Schritt 4.

Die Entscheidung, wann ein Teilproblem in zwei neue verzweigt und wann Gomory Cuts hinzugefügt werden, wird in [1] mittels einer Heuristik entschieden.

‖ Ein offensichtlicher Fehler in [1] wurde berichtigt.

Literatur

[1] Akrotirianakis, I.; Maros, I.; Rustem, Berç.: *An Outer Approximation based Branch and Cut Algorithm for convex 0-1 MINLP problems*. Technical Report 2000-06, Imperial College of Science, Technology and Medicine, London (2000)

[2] Duran, M. A.; Grossmann, I. E.: *An outer approximation algorithm for a class of minlp problems*. Mathematical Programming, Vol. 36, S. 307-339 (1986)

[3] Fletcher, R.; Leyffer, S.: *Solving mixed integer nonlinear programs by outer approximation*. Mathematical Programming, Vol. 66, S. 327-349 (1994)

[4] Floudas, C. A.: *Nonlinear and Mixed-Integer Optimization*. Oxford University Press, Oxford (1995)

[5] Gomory, R. E.: *Outline of an algorithm for integer solutions to linear programs*. Bulletin of the American Mathematical Society, Vol. 64, S. 275-278 (1958)

[6] Quessada, I.; Grossmann, I. E.: *An lp/nlp based branch and bound algorithm for convex minlp optimization problems*. Computers and Chemical Engineering, Vol. 16, S. 937-947 (1992)

Das Handout und die Präsentationsfolien stehen unter

`http://www.mathi.uni-heidelberg.de/~ferreau/globalOpt/`

zum Download bereit.