Operations Research (ORE) - Zusammenfassung

# Lineare Optimierung:

* Ziel: Lineare Zielfunktion unter Nebenbedingungen in Form von Gleichungen oder Ungleichungen minimieren oder maximieren
* Universell einsetzbar
* Nichtnegativitätsbedingungen (Variable nicht-negative ganze Zahlen)

## Definitionen/Theoretische Grundlagen:

* LOP: Aufgabe eine linerae Funktion unter der Beachtung linearer Nebenbedingungen (Restriktionen) zu maximieren oder zu minimieren.
* Zielfunktion: zu minimierende oder maximierende Funktion
* Entscheidungsvariablen: in die Zielfunktion eingehende Variablen
* Allgemeine Form: f(x) = f(x1,...,xn)= c1\*x1+c2\*x2+...+cn\*xn
* Vektorenschreibweise: f(x) = <c,x> = cT\*x
* Affinlineare Zielfunktion: f(x) = cT\*x + z0
* Nebenbedingungsaufbau: ai1 \*x1+ ai2 \* x2 + ... + ain \* xn {<=,=,>=} bi
* Nichtnegativitätsbedingungen: xi >= 0
* i = {1,2,...,m}
* zulässiger Punkt des Problems: Vektor der alle Nebenbedingungen (+NNB) eines LOPs erfüllt
* zulässige Menge(M)/zulässiger Bereich: Menge aller zulässigen Punkte
* optimaler Punkt, wenn es keinen besseren zulässigen Punkt gibt, d.h. wenn es keinen besseren Zielfunktionswert gibt
* optimaler Wert der Zielfunktion: f(x\*)
* optimaler Zielfunktionswert z\* unabhängig von opt. Punkt

## 

## Analytische Konzepte

* Standardform: P<=: max cT \* x, s.t. A\*x <= b, x >= 0
* n: Anzahl der Entscheidungsvariablen
* m: Anzahl der Nebenbedigungen
* c = (c1,...cn)T : Vektor der Zielfunktionskoeffizienten
* x = (x1,...,xn)T : Vektor der Entscheidungsvariablen
* b = (b1,...,bm)T : Vektor der Werte der rechten Seite
* Koeffizientenmatrix (mxn)-Matrix A = (aij)

## Transformation in Standardform

* Falls Zielfunktion zu minimieren ist, ersetzt f durch –f
* Jede Gleichungsrestriktion (=) kann durch die beiden Ungleichungen (<= und >=) ersetzt werden
* Jede Restriktion mit >= kann durch Miltiplikation mit -1 in eine <=-Restriktion umgewandelt werden

## Erkenntnisse aus dem Satz von Weierstraß

* Ein LOP in Standardform besitzt einen optimalen Punkt nichtleer und bechränkt ist.
* Nichtleer und beschränkt Polyeder-> konvexes Polytop (Alle Verbindungen von Punkten aus Menge sind in Menge)

## Extrempunkte konvexer Mengen

* Lassen sich nicht aus Kombination anderer Punkte darstellen
* Beispiel: alle Randpunkte einer Kreisscheibe
* Konvexe Polyeder: endliche viele Extrempunkte

## Normalform und Basen

* Alle Nebenbedingungen außer NNB als Gleichungsrestriktionen
* Geschieht durch Einführung zusätzlicher Variablen – Schlupfvariablen
* Kanonische Form: Im = (1 0 0; 0 1 0;0 0 1) -> diagonal

## Zulässige Basislösung, Basis- und Nichtbasisvariablen

* Lösung: x = (x1,...xn+m)T der Restriktion A˜\* x = b wenn n Einträge xi von x den Wert Null haben und wenn die zu den restlichen m Einträgen gehörenden Spalten a’ von A linear unabh. Sind.
* Basislösung liegt darin begründet, dass diese linear unabh. Spalten eine Basis im Rm bilden. Wenn die von null verschiedenen Einträge von x außerdem nicht negativ sind spricht man von einer zulässigen Basislösung.
* Die m lin. Unabh. Vektoren a’ einer Basislösung nennt man Basisvektoren
* m zugehörige xi Basisvariablen
* Rest: Nicht-Basisvariablen/Vektoren
* Gleichungssystem lässt sich als BXb + Nxn = b bschreiben
* Für ein LOP in Normalform sind die Ecken von M genau die zulässigen Basislösungen
* Eine zulässige Basislösung, in der mindestens eine Basisvariable den Wert null besitzt, heißt degeneriert, ansonsten heißt sie nichtdegeneriert.

## Simplex Algorithmus (SA)

* Ausgehend von einer Startecke werden Ecken (zulässige Basislösungen) mit verbesserten Zielfunktionswerten berechnet (anstatt: ungezieltes Berechnen der Zielfunktionswerte an allen Eckpunkten)
* Austauschen von Basisvektor durch Nichtbasisvektor
* Basislösungen berechnen
* Zielfunktion enthält nur aktuelle Nicht-Basis-Variablen (NBV)

## Simplex-Tableau

* Welche BV werden gegen welche NBV ausgetauscht?
* Abbruchbedingungen?
* LOP in Normalform

### Pivotelement und Austauschschritt

* PE darf nicht null sein
* PE sollte > 0 sein
* Auswahlregeln:
  + Pivotspalte s: Ist der kleinste Eintrag cj, j=1,...,n in der Zielfunktionszeile negativ, so wähle die dazugehörige Spalte als Pivotspalte aus.
  + Pivotzeile r: Wähle die Pivotzeile r so, dass unter allen i mit ais > 0 der Quotient (bi)/(ais) am kleinsten wird.
* Stoppregeln:
  + Opt. Punkt gefunden: Falls für alle j cj >= 0 gilt, so ist x ein optimaler Punkt und zo der optimale Wert des LOP.
  + Zielfunktion unbeschränkt: Ist ein cs < 0 und alle ais <= 0 ist das LOP unlösbar, da die Zielfunktion auf M unbeschränkt ist.

## Zwei-Phasen Simplex-Verfahren

* Nur 2 Schlupfvariablen?
* Und 2 Hilfsvariablen?
* Hilfszielfunktion: min. Summe der Hilfsvariablen oder max. – (Summe der Hilfsvariablen)
* Auflösen nach allen Hilfsvariablen
* Einsetzen der Ergebnisse (Hilfsvariable = (das hier)) in die Hilfszielfunktion
* Simplexverfahren mit Hilfszielfunktion
* Lösung ist die Startecke für das ursprüngliche LOP
* Phase II: Lösen des LOPs über Simplex
* Startlösung wenn nicht kanonisch?

## Die BigM-Methode

## [Guckst du hier!](http://gidf.de)

## Probleme mit degenerierter optimaler Basislösung

* Nimmt im Optimaltableau eine Basisvariable den Wert null an, d.h. die b-Spalte besitzt eine null, so kann man zum selben Schnittpunkt weitere Zustände des Tableaus mit optimalem Wert der Zielfunktion erzeugen. Dadurch besteht die Gefahr, dass der Algorithmus in einen Zyklus gerät. Dies muss ggf. beim Programmieren abgefangen werden. Deshalb: Verwendung von Anti-Zyklus-Strategien.

## Anti-Zyklus-Strategien nach Bland

Nutzt modifizierte Auswahlregeln:

* Pivotspalte s: Gibt es negative Zielfunktionskoeffizienten cj, j=1,...,n, so wähle unter ihnen denjenigen mit dem kleinsten Index s und die dazugehörige Spalte s als Pivotspalte.
* Pivotzeile r: Wähle die Pivotzeile r so, dass unter allen i mit ais > 0 der Quotient bi/as am kleinsten ist. (gleich wie bei normalem Simplex) Falls dieser Index nicht eindeutig ist, dann wähle den kleinsten Index als Zeile r.
* Stoppregeln bleiben unverändert!

# Dynamische Optimierung

## Klassifizierung und graphische Darstellung von DO

### Zeitabstände zwischen den Perioden (Stufen)

Man unterscheidet diskrete und kontinuierliche DO-Probleme. Ein diskretes Problem liegt vor, wenn Entscheidungen bzw. Zustandsänderungen zu diskreten Zeitpunkten (bzw. in diskreten Schritten) erfolgen; andernfalls spricht von Kontrollproblemen. Mit derartigen Fragestellungen beschäftigt sich die Kontrolltheorie.

### Informationsgrad über die Störgröße bk

Wie allgemein bei Entscheidungsproblemen unterscheidet man zwischen deterministischen und stochastischen Problemen. Bei deterministischen Problemen geht man davon aus, dass jede Störgröße Zufallsvariablen sind und damit verschiedene Werte mit bekannten Wahrscheinlichkeiten annehmen kann.

## Ein- oder Mehrwertigkeit der Zustands- und Entscheidungsvariablen

Anders als im „Bestellmengenproblem“ können die Zustands- bzw. Entscheidungsvariablen Vektoren sein, z.B. im Bestellmengenproblem mit mehreren Produkten.

## Endlichkeit/Unendlichkeit der Menge Xk und Uk

\_\_

Das beschriebene Bestellmengenproblem gehört zur Klasse der diskreten, deterministischen DO Probleme mit endlichen Zustands- und Entscheidungungsmengen. Derartige Probleme lassen sich anschaulich durch einen gerichteten Graphen darstellen.

## Das Lösungsprinzip der DO für diskrete, deterministische Minimierungsprobleme

Eine Folge (un,un+1,...,uk) von Entscheidungen, die ein System von einem Zustand xn-1 e Xn-1 in einen Zustand xk e Xk überführt, bezeichnet man als eine Politik. Entsprechend nennen wir eine Folge (un\*, un+1\*,...,uk\*) von Entscheidungen, die ein System unter der Minimierung der Zielfunktion von einem Zustand xn-1 e Xn-1 in einen Zustand xk e Xk überführt eine optimale Politik.

### Bellmansches Optimalitätsprinzip

Sei (u1\*,...,uk-1\*,uk\*,...,un\*) eine optimale Politik, die das System vom Anfangszustand xo = alpha in dem vorgegebenen oder einen freien aber erlaubten Endzustand xn überführt. Sei ferner xk-1\* der Zustand, dem das System in Stufe (Periode) k-1 annimmt, dann gilt:

* (uk\*,...,un\*) ist eine optimale Teilpolitik, die das System von Zustand xk-1\* in den vorgegebenen oder erlaubten Endzustand xn überführt.
* (ui\*,...,uk-1\*) ist eine optimale Teilpolitik, die das System vom gegebenen Anfangszustand (xo = alpha) in xk-1\* überführt.