



Optimierungssysteme

Modelle, Verfahren, Software, Anwendungen

Für den Bachelor Studiengang äquivalent zum Modul
Entscheidungsunterstützung Produktion

Uwe H. Suhl und Veronika Waue

Lehrstuhl für Wirtschaftsinformatik

SS 2008

Übersicht

- Vorlesung mit integrierter Übung
- Studierende im Diplomstudiengang
 - 4 BP im Fach Wirtschaftsinformatik, Produktion oder ABWL
 - Wahlpflichtveranstaltung im Fach Wirtschaftsinformatik
 - 4 BP zusätzlich zu EUS, Logistik, betriebliche. Informationssysteme
- Bachelor Studierende
 - Veranstaltung ist äquivalent zum Modul Entscheidungsunterstützung Produktion für Bachelor Studierende (V 101053 Ü 101054)
- Download der Vorlesungs-Folien (pdf) vom Web-Server
- keine Programmierkenntnisse erforderlich, jedoch Umgang mit Softwareprodukten unter Windows, insbesondere Excel
- in der Übung wird zu Beginn ClipMOPS (Excel Addin) benutzt
- größere Modelle werden in der Modellierungssprache AMPL formuliert und mit MOPS im Rahmen von MOPS Studio gelöst
- Eine AMPL-Hausarbeit in 2-er Teams ergibt maximal 10 Punkte, die zusätzlich zu den Punkten der Klausur zur Notenbestimmung dient (optional)
- Klausur:
 - **Termin steht noch nicht fest**
 - 4 Vorfragen (20 P) und eine Hauptaufgabe die mathematische Modellierung beinhaltet (30 P)

Stoffplan (1)

Einleitung und Grundlagen

- Kurze Einführung in Unternehmensforschung / Operations Research / Management Science und ein Überblick über praktische Anwendungen

Mathematische Optimierungsmodelle (LP)

- Grundlagen, Modelle, Lösungen, Beispiele für Lineare Optimierungsmodelle, Lösung mit ClipMOPS, Ökonomische Interpretation, mathematische Grundlagen der Linearen Optimierung, Lösungsmethodik, LP-Software

Mathematische Optimierungsmodelle (IP)

- Grundlagen von Integer Optimierungsmodellen (IP), Beispiele, Prinzip des Branch-and-Bound, Lösung des IP-Beispiels mit Branch-and-Bound, prinzipielle Lösungsaspekte von IP-Modellen mit Standardsoftware

Modellierungstechniken

- Weiche Restriktionen, diskrete Werte, Indikatorvariablen, Schwellenwerte und Fixkostenprobleme, Reihenfolgerestriktionen, alternative Restriktionen, Nichtlinearitäten, Linearisierung, logische Verknüpfungen von Aussagen, Systematische Modellierung mit Aussagenlogik

Stoffplan (2)

Einführung in AMPL und MOPS Studio

- AMPL: A Mathematical Programming Language, MOPS: Mathematical Optimization System, es werden größere Modelle in der Modellierungssprache AMPL formuliert und im MOPS Studio gelöst

Praktische Anwendungsbeispiele mit Modellen

- Sortimentsoptimierung, Aktionsdisplays, Fertigungsprobleme, Standortplanung, mehrstufige Distributionssysteme, Ausbauplanung von Gasnetzen, Planung von Hub & Spoke Systemen

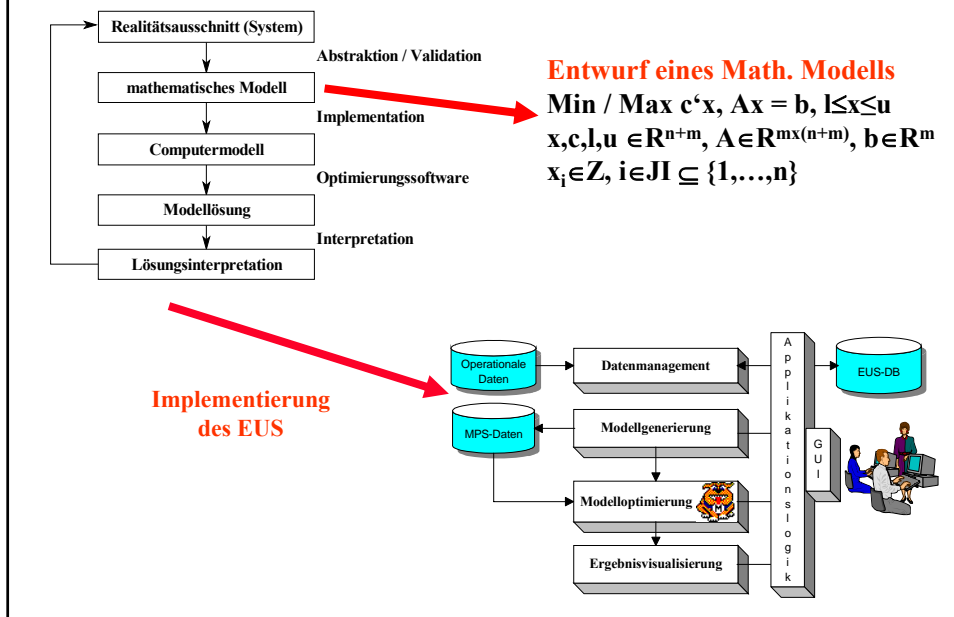
Effiziente IP-Modellformulierungen

- Schärfe der LP-Relaxierung eines IP-Modells, Fixkostenprobleme, disaggregierte Formulierungen, IP-Preprocessing in MOPS,

Einbettung von MOPS API-Funktionen in EUS Software

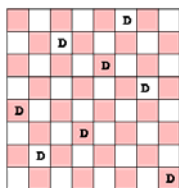
- Grundlegende MOPS API-Funktionen, Beispiel zur Lösung von Sudoku-Rätseln mit MS Excel unter Verwendung von MOPS in VBA

Entscheidungsunterstützende Systeme mit MP



MOPS Studio

- Ein interaktives graphisches Front-End zur MOPS DII und zu AMPL (A Mathematical Programming Language) [Fourer, Gay, Kernighan]
- MOPS Studio wird seit zwei Semestern an den Universitäten FU-Berlin und Paderborn in Lehrveranstaltungen „Optimierungssysteme“ eingesetzt
- Das System wird in einer größenlimitierten Version in naher Zukunft für alle Universitäten kostenlos zur Verfügung gestellt (AMPL Studentenversion)
- Beispiel „8 Queens Problem“: es sollen **alle zulässigen Lösungen** bestimmt werden; **eine** zulässige Lösung könnte über die Lösung folgenden 0-1-Problems erzeugt werden:



$$\text{Maximiere } \sum_{i=1}^8 \sum_{j=1}^8 x_{ij}, x \in \{0,1\}^{64}$$

$$\sum_{i=1}^8 x_{ij} \leq 1, j \in \{1, \dots, 8\} \text{ !in jeder Spalte höchstens ein Eintrag}$$

$$\sum_{j=1}^8 x_{ij} \leq 1, i \in \{1, \dots, 8\} \text{ !in jeder Zeile höchstens ein Eintrag}$$

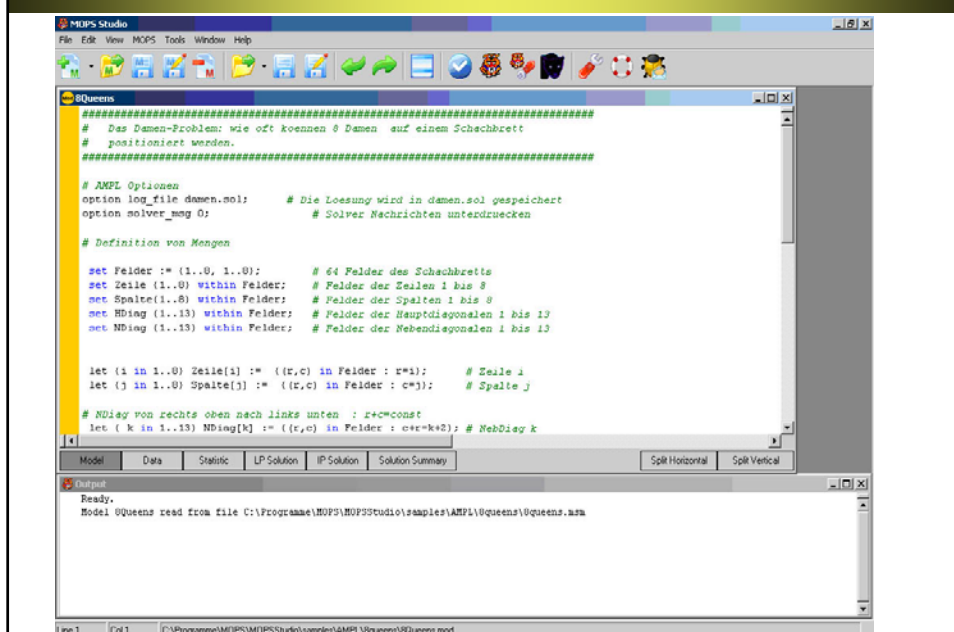
$$\sum_{j=i-k+1}^i x_{ij} \leq 1, k \in \{1, \dots, 13\} \text{ !max. Einträge in Hauptdiagonalen}$$

$$\sum_{j=i+k-2}^i x_{ij} \leq 1, k \in \{1, \dots, 13\} \text{ !max. Einträge in Nebendiagonalen}$$

Modell für 8 Queens Problem

```
# Wie oft koennen 8 Damen auf einem Schachbrett positioniert werden?
# AMPL Optionen
option log_file damen.sol;      # Die Loesung wird in damen.sol gespeichert
option solver_msg 0;           # Solver Nachrichten unterdruecken
# Definition von Mengen
set Felder := {1..8, 1..8};     # 64 Felder des Schachbretts
set Zeile {1..8} within Felder; # Felder der Zeilen 1 bis 8
set Spalte {1..8} within Felder; # Felder der Spalten 1 bis 8
set HDiag {1..13} within Felder; # Felder der Hauptdiagonalen 1 bis 13
set NDiag {1..13} within Felder; # Felder der Nebendiagonalen 1 bis 13
let {i in 1..8} Zeile[i] := {(r,c) in Felder : r=i};      # Zeile i
let {j in 1..8} Spalte[j] := {(r,c) in Felder : c=j};     # Spalte j
# NDiag von rechts oben nach links unten : r+c=const
let {k in 1..13} NDiag[k] := {(r,c) in Felder : c+r=k+2}; # NebDiag k
# HDiag von links oben nach rechts unten: c-r=const
let {k in 1..13} HDiag[k] := {(r,c) in Felder : c-r=k-7}; # HptDiag k
param act;
let act:=0;
set Soln {0..act} within Felder; # Position der Damen in den Loesungen 0 bis act
let Soln[0] := {};              # Anfangsloesung ist leer
# Modell:
var x {Felder} binary;
maximize Ziel: sum {(i,j) in Felder} x[i,j];
subject to DiagN {k in 1..13}: sum{(r,s) in NDiag[k]} x[r,s] <= 1;
subject to DiagH {k in 1..13}: sum{(r,s) in HDiag[k]} x[r,s] <= 1;
subject to Row {i in 1..8}: sum{(r,s) in Zeile[i]} x[r,s] <= 1;
subject to Col {j in 1..8}: sum{(r,s) in Spalte[j]} x[r,s] <= 1;
subject to Old {k in 1..act}: sum{(r,s) in Soln[k]} x[r,s] <= 7.1 ; # Verbot alter Loesungen
repeat
{
  solve ;
  # solve_result_num = 200 ist der MOPSSAMPL Returncode für infeasible
  if Ziel <= 7.5 or solve_result_num = 200 then break;
  display act;
  let act := act + 1;
  let Soln[act] := {(i,j) in Felder : x[i,j] >= 0.9 };
};
display Soln;
```

MOPS Studio Hauptfenster



Weiterführende Literatur

1. **Suhl, Leena / Taieb Mellouli, Optimierungssysteme – Modelle, Verfahren, Software, Anwendungen. Springer-Verlag, 2005**
2. **Fourer, Gay, Kernighan, AMPL A Modelling Language for Mathematical Programming, 2nd Edition, Thomson, Duxbury, 2003**
3. **Taylor B.W., Introduction to Management Science, Prentice Hall, 1999**
4. **Bell Peter C., Management Science / Operations Research – A Strategic Perspective. South-Western College Publishing, 1999.**
5. **Domschke, W., Drexl A., Einführung in Operations Research. Springer, 2002.**
6. **Domschke, W. et al., Übungen und Fallbeispiele zu Operations Research. Springer, 2000**
7. **Taha H., Operations Research, An Introduction. Prentice-Hall, 7th Edition, 2002.**
8. **H.P. Williams, Model Building in Mathematical Programming, John Wiley & Sons, 2002**
9. **Wolsey L., Integer Programming, John Wiley & Sons, 1998**
10. **Pochet, Y. and L. Wolsey, Production Planning by Mixed Integer Programming, Springer 2006**
11. **Feige D. und P. Klaus, Modellbasierte Entscheidungsunterstützung in der Logistik, DVV Media Group I, Deutscher Verkehrs-Verlag, 2008.**
12. **Vahrenkamp, R. und D. Mattfeld, Logistik-Netzwerke, Gabler, 2008**