

# Commitment ILOG-Verbesserung

## **Ergebnisdarstellung vom ILOG** im Power-LP optimieren

Renate Bondar-Erni und Vildan Özgür 30.10.2014

Version 1.2

#### **Historie:**

Version	Datum	Bearbeiter	Beschreibung
1.0	21.10.2014	Renate Bondar-Erni	Erstellung des
		und Vildan Özgür	Commitments
1.1	28.10.2014	Renate Bondar-Erni	Änderungen
		und Vildan Özgür	hinzugefügt
1.2	30.10.2014	Renate Bondar-Erni	Änderungen
		und Vildan Özgür	hinzugefügt

#### **Projektorganisation**

Im Rahmen der Veranstaltung "Anwendung der linearen Optimierung" findet an der HTWG Konstanz im Wintersemester 14/15 das Projekt zur ILOG-Verbesserung statt.

Zeitlicher Rahmen: WS 14/15

Projektbeginn: 29.10.2014

Projektende: 14.01.2015

Das Projekt endet mit einer Präsentation der ILOG-Verbesserung im Programm "Power-LP", inklusive der gesamten Dokumentation durch die Abnahme von Herrn Prof. Dr. Grütz und Herrn Kane.

#### Projektverantwortlicher

Herr Prof. Dr. Grütz

#### **Technischer Verantwortlicher**

• Herr Kane

### Projektmitglieder

Nachname, Vorname	Matrikelnummer
Bondar-Erni, Renate	286507
Özgür, Vildan	286106

#### Ziel:

Die Ergebnisdarstellung vom ILOG im Power-LP ist momentan sehr unübersichtlich und knapp bemessen.

Deswegen soll sie so optimiert werden, dass sie die Ergebnisse in der 1. Priorität ähnlich wie in der Ergebnismaske vom MOPS oder Weidenauer im Power-LP darstellt. Alternativ würde

man die Darstellung vom ILOG als 2. Priorität dem LP-Solve anpassen, falls die erstgenannte Darstellung wie MOPS oder Weidenauer nicht funktionieren sollte.

#### **Beispiel:**



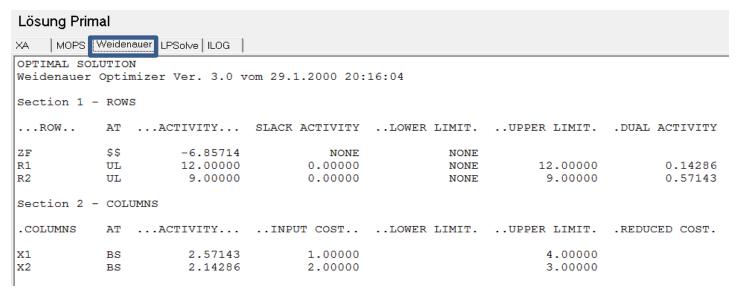
#### Ergebnisausgabe bei ILOG (momentan):

```
Lösung Primal
XA MOPS Weidenauer LPSolve ILOG
<?xml version = "1.0" encoding="UTF-8" standalone="yes"?>
<CPLEXSolution version="1.2">
 <header
  problemName="ILOG.lp"
   objectiveValue="6.85714285714286"
  solutionTypeValue="1"
   solutionTypeString="basic"
  solutionStatusValue="1"
  solutionStatusString="optimal"
  solutionMethodString="dual"
  primalFeasible="1"
  dualFeasible="1"
  simplexIterations="2"
  writeLevel="1"/>
 <quality
   epRHS="1e-06"
   epOpt="1e-06"
  maxPrimalInfeas="0"
  maxDualInfeas="0"
  maxPrimalResidual="8.88178419700125e-16"
  maxDualResidual="0"
  maxX="2.57142857142857"
  maxPi="0.571428571428571"
  maxSlack="0"
  maxRedCost="0"
  kappa="3.14285714285714"/>
 <linearConstraints>
  <constraint name="c1" index="0" status="LL" slack="0" dual="0.142857142857143"/>
  <constraint name="c2" index="1" status="LL" slack="0" dual="0.571428571428571"/>
 </linearConstraints>
 <variables>
  <variable name="x1" index="0" status="BS" value="2.57142857142857" reducedCost="-0"/>
  <variable name="x2" index="1" status="BS" value="2.14285714285714" reducedCost="-0"/>
 </variables>
```

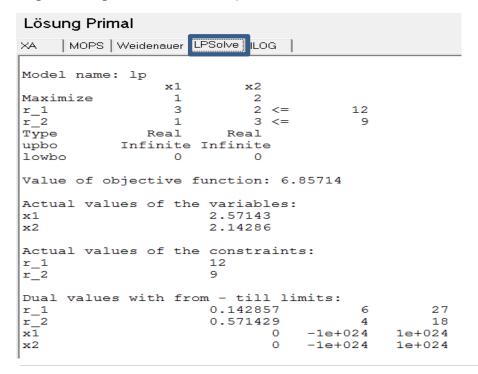
#### Ergebnisausgabe beim MOPS (so ähnlich soll es beim ILOG aussehen):

#### Lösung Primal MOPS | Weidenauer | LPSolve | ILOG | MOPS V7.06, (C) Uwe H. Suhl, 14.06.2004 30.10.2014 20.16 solution: (optimal) iteration number = ...activity... ...name... defined as functional 6.85714 ZF MYRHS restraints SECTION 1 - ROWS ...ACTIVITY... SLACK ACTIVITY ..LOWER LIMIT. ..UPPER LIMIT. NUMBER ...ROW.. AT .DUAL ACTIVITY 12.00000 0.00000 12.00000 3 R1 TJT. -0.14286none 4 R2 UL9.00000 0.00000 none 9.00000 -0.57143SECTION 2 - COLUMNS NUMBER .COLUMNS AT ...ACTIVITY... .. INPUT COST.. ..LOWER LIMIT. ..UPPER LIMIT. .REDUCED COST. 1 X1 BS 2.57143 1.00000 0.00000 none 0.00000 2 BS 2.14286 2.00000 0.00000 0.00000 X2 none

#### Ergebnisausgabe beim Weidenauer (so ähnlich soll es beim ILOG aussehen):



#### Ergebnisausgabe beim LP-Solve (alternativ könnte der ILOG auch so ähnlich aussehen):



#### Folgende Bearbeitungsfälle innerhalb der ILOG-Verbesserung sind möglich:

#### 1. Fall → Was mindestens erreicht bzw. bearbeitet werden soll:

Die Ergebnisdarstellung vom ILOG soll so bearbeitet werden, dass die Ergebnisse ähnlich wie beim LP-Solve dargestellt werden. Damit wäre die Mindestanforderung bzw. die 2. Priorität von Prof. Grütz erfüllt.

2. Fall → Wenn der Mindestfall erfüllt ist, kann dann der nächste Schritt erfolgen:

Die Ergebnisdarstellung vom ILOG soll jetzt so verfeinert werden, dass die Ergebnisse ähnlich wie bei MOPS oder Weidenauer dargestellt werden. Damit wäre die 1. Priorität von Prof. Grütz als Anforderung erfüllt.

#### Wenn dann noch Zeit vorhanden ist, könnten folgende Fälle bearbeitet werden:

#### 3. Fall → Informationen über die quadratische Programmierung sammeln:

Das Ziel ist es, dass der ILOG auch die quadratische Programmierung beherrschen soll. Dafür würde man dann Informationen zu deren Umsetzung sammeln.

#### 4. Fall → Quadratische Programmierung umsetzen:

Hierbei würde man dann an Hand der Informationen die quadratische Programmierung im ILOG umsetzen.

#### **Funktionale Anforderungen**

- Die Ergebnisausgabe vom ILOG im Power-LP soll mindestens ähnlich der LP-Solve-Darstellung sein.
- Im weiteren Schritt wird dann versucht, die Ergebnisse ähnlich der MOPS- bzw. Weidenauer-Darstellung anzuzeigen.
- Falls dann noch Zeit wäre, würde man sich mit der quadratischen Programmierung auseinandersetzen.
- Dabei soll aber die Funktionalität bestehen bleiben.
- Es soll Windows 7 lauffähig bleiben.

#### Nichtfunktionale Anforderungen

- Dokumentation aller Änderungen durch Kommentare bzw. farblicher Markierung im Code des Programmes und durch eine schriftliche Dokumentation.
- Testen der Ergebnisausgabe.
- Bereitstellung der optimierten Ergebnisdarstellung vom ILOG an Herr Kane.

#### Rahmenbedingungen

- Design und Funktionalität vom Power-LP soll bestehen bleiben. Nur Ergebnisdarstellung von ILOG wird optimiert.
- Es soll Windows 7 lauffähig bleiben.
- Entwicklung in C++ mit dem Programm "Borland C++ Builder".