

Optimierung der Ergebnisdarstellung vom ILOG-Solver im Power-LP

Anwendung der linearen Optimierung

Renate Bondar-Erni

Vildan Özgür

WS 14/15 – 14.01.2015

Agenda

1. Aktueller Stand
2. Projektbeschreibung
3. Ergebnisdarstellung anderer Solver
4. Problemstellung
5. Vorgehensweise
6. Testbeispiele
7. Versionsmanagement
8. Fazit

1. Aktueller Stand

- Power-LP in Version 0.7.3 im OR-WEB verfügbar.
- OR-Alpha hat alte Version, wird aber noch geändert.
- Power-LP in C++ programmiert.
- ILOG-Solver ist ein Produkt von IBM.
- ILOG: Im WS 13/14 als Projekt im Power-LP eingebettet.
- Funktioniert seitdem einwandfrei.
- Tool: Borland C++ Builder 6

2. Projektbeschreibung

- Ergebnisdarstellung des ILOG-Solvers zu optimieren.
- **Gründe:**
 - zu unübersichtlich
 - unnötige Informationen
- **Es soll auf das Notwendigste reduziert werden:**
 - Zielfunktionswert
 - Variablenwerte

2. Projektbeschreibung

Lösung Primal

XA | MOPS | Weidenauer | LPSolv | 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>
```

← Zielfunktionswert

↓ Variablen

3. Ergebnisdarstellung anderer Solver

```
Lösung Primal
XA | MOPS | Weidenauer | LPSolve | LOG |

Model name: lp
      x1      x2
Maximize      1      2
r_1          3      2 <=      12
r_2          1      3 <=      9
Type          Real      Real
upbo      Infinite Infinite
lowbo          0          0

Value of objective function: 6.85714

Actual values of the variables
x1          2.57143
x2          2.14286

Actual values of the constraints:
r_1          12
r_2           9

Dual values with from - till limits:
r_1          0.142857      6      27
r_2          0.571429      4      18
x1           0      -1e+024      1e+024
x2           0      -1e+024      1e+024
```

← Zielfunktionswert

← Variablen

3. Ergebnisdarstellung anderer Solver

Lösung Primal

XA **MOPS** Weidenauer | LPSolve | ILOG |

MOPS 1.7.06 (C) Uwe H. Suhl, 14.06.2004 30.10.2014 20.16

solution: (optimal)

iteration number = 2

...name... ...activity... defined as
functional 6.85714 ZF
restraints MYRHS

SECTION 1 - ROWS

NUMBER	...ROW..	AT	...ACTIVITY...	SLACK	ACTIVITY	..LOWER LIMIT.	..UPPER LIMIT.	.DUAL	ACTIVITY
3	R1	UL	12.00000		0.00000	none	12.00000	-0.14286	
4	R2	UL	9.00000		0.00000	none	9.00000	-0.57143	

SECTION 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	X2	BS	2.14286	2.00000	0.00000	none	0.00000

Lösung Primal

XA **MOPS** Weidenauer | LPSolve | ILOG |

OPTIMAL SOLUTION

Weidenauer Optimizer Ver. 3.0 vom 29.1.2000 20:16:04

Section 1 - ROWS

...ROW..	AT	...ACTIVITY...	SLACK	ACTIVITY	..LOWER LIMIT.	..UPPER LIMIT.	.DUAL	ACTIVITY
ZF	\$\$	-6.85714		NONE	NONE			
R1	UL	12.00000		0.00000	NONE	12.00000	0.14286	
R2	UL	9.00000		0.00000	NONE	9.00000	0.57143	

Section 2 - COLUMNS

.COLUMNS	AT	...ACTIVITY...	..INPUT COST..	..LOWER LIMIT.	..UPPER LIMIT.	.REDUCED COST.
X1	BS	2.57143	1.00000		4.00000	
X2	BS	2.14286	2.00000		3.00000	

4. Problemstellung

- Keine Erfahrung in der Programmiersprache C ++.
- Code vom Power-LP sehr komplex.
- Die betreffenden Code-Zeilen für ILOG beziehen sich nur auf die Funktionalität.
- Programm „CPLEX“ nicht änderbar.
- Altteam kontaktiert → konnten nicht helfen.
- Herrn Schlenker von IBM kontaktiert.

4. Problemstellung

- Parallel Herr Grütz nach Unterstützung gefragt.
- **Vorschläge von beiden erhalten:**
 - Solution Datei extrahieren, in XSLT umformatieren.
 - Parser einbauen.
- Benötigt gute Kenntnisse in C++ und ist komplex.
- Nach einfacher Lösung gesucht.

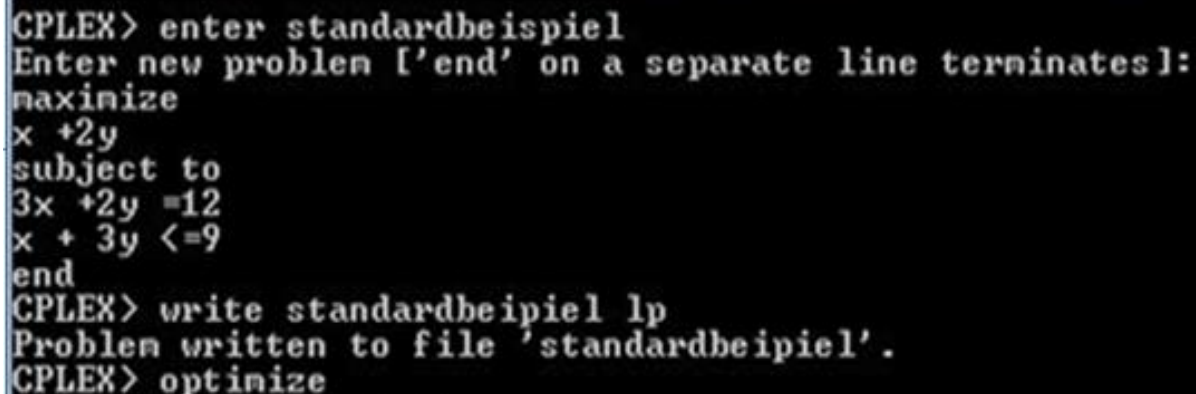
5. Vorgehensweise

- Haben uns nochmal den ILOG-Code vom Power-LP angeschaut: Klasse „solver.cpp“ mit der Methode

PutILOG_BAT(char dir)*

```
writeLF(f, "");  
writeLF(f, "\ncplex.exe -c \"read ILOG.lp \" \"optimize\" \"write ILOG.out sol\" \"quit\" ");  
close(f);
```

- Ähnliche Befehle wie in der Anwendung „cplex.exe“



```
CPLEX> enter standardbeispiel  
Enter new problem ['end' on a separate line terminates]:  
maximize  
x +2y  
subject to  
3x +2y =12  
x + 3y <=9  
end  
CPLEX> write standardbeispiel lp  
Problem written to file 'standardbeispiel'.  
CPLEX> optimize
```

5. Vorgehensweise

- Ergebnisdarstellung vom ILOG der „cplex.exe“ – Ausführung anpassen.
- Codezeile wurde dann umgeändert:

```
writeLF(f, "");  
writeLF(f, "\cplex.exe -c \"read ILOG.lp \" \"optimize\" \"display solution variables -\" \"write cplex.exe\" \"quit\" ");  
close(f);
```

- **„optimize“**: Gibt Zielfunktionswert aus.
- **„display solution variables -“**: Gibt Variablenwerte aus.
- **„write cplex.exe“**: Gibt die Ausführung aus.

5. Vorgehensweise

The image shows the ILOG CPLEX interface with the 'LOESUNG' (Solution) tab selected. The 'ILOG' sub-tab is active, displaying the solution log. A red box highlights the 'Dual simplex - Optimal: Objective = 6.8571428571e+000' line. Another red box highlights the 'Variable Name' and 'Solution Value' table at the bottom. A red arrow points from the 'Zielfunktionswert' (Objective Function Value) label to the objective value in the log. Another red arrow points from the 'Variablen' (Variables) label to the variable values in the table. A third red arrow points from the 'Zielfunktionswert' label to the 'Objective = 6.8571428571e+000' line in the command window. A fourth red arrow points from the 'Variablen' label to the 'Solution Value' column in the command window's display solution variables output.

Zielfunktionswert

Variablen

GUI Log Output:

```
Log started (V12.5.0.0) Mon Dec 22 23:40:55 2014

Problem 'ILOG.lp' read.
Read time = 0.00 sec. (0.00 ticks)
Tried aggregator 1 time.
No LP presolve or aggregator reductions.
Presolve time = 0.00 sec. (0.00 ticks)

Iteration log . . .
Iteration: 1 Dual infeasibility = 0.000000
Iteration: 2 Dual objective = 6.857143

Dual simplex - Optimal: Objective = 6.8571428571e+000
Solution time = 0.00 sec. Iterations = 2 (1)
Deterministic time = 0.00 ticks (4.05 ticks/sec)

Variable Name      Solution Value
x1                  2.571429
x2                  2.142857
```

Command Window Output:

```
CPLEX> enter standardbeispiel
Enter new problem ['end' on a separate line terminates]:
maximize
x +2y
subject to
3x +2y =12
x + 3y <=9
end
CPLEX> write standardbeispiel lp
Problem written to file 'standardbeispiel'.
CPLEX> optimize
Tried aggregator 1 time.
LP Presolve eliminated 1 rows and 1 columns.
Aggregator did 1 substitutions.
All rows and columns eliminated.
Presolve time = 0.05 sec. (0.00 ticks)

Dual simplex - Optimal: Objective = 6.8571428571e+000
Solution time = 0.13 sec. Iterations = 0 (0)
Deterministic time = 0.00 ticks (0.02 ticks/sec)

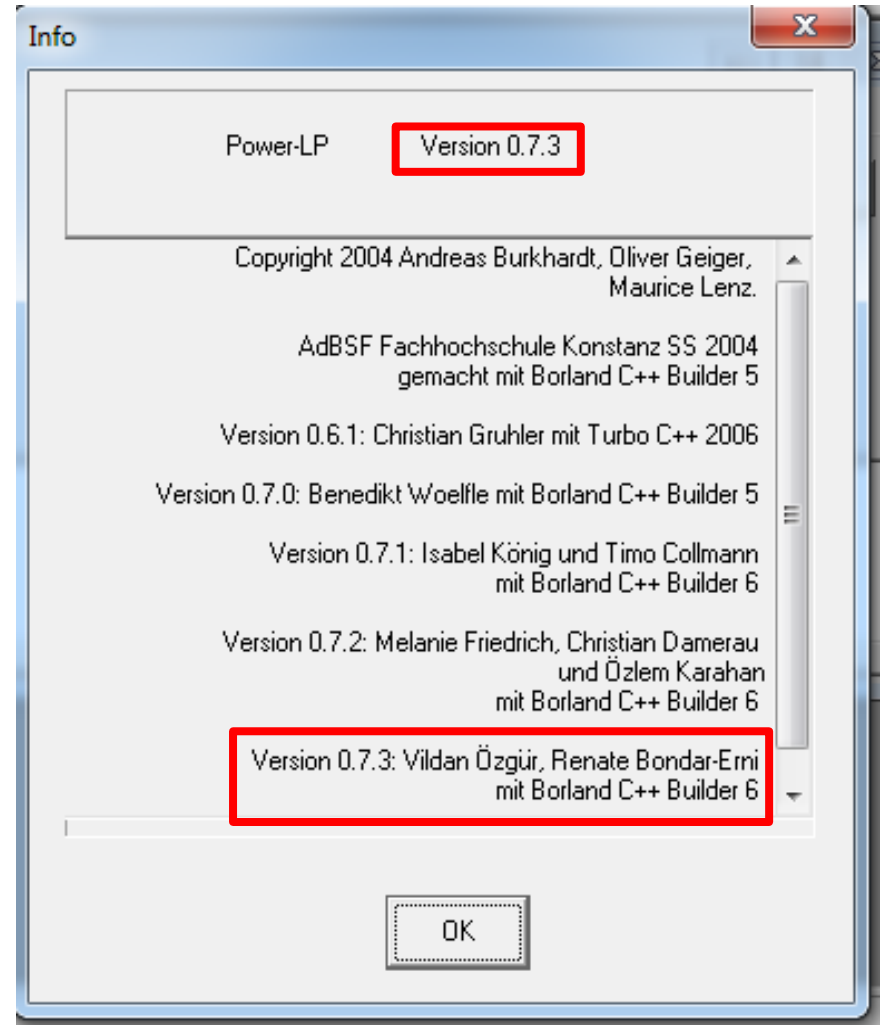
CPLEX> display solution variables -
Variable Name      Solution Value
x                   2.571429
y                   2.142857
```

6. Testbeispiele

Live-Demo im Power-LP!

7. Versionsmanagement

- Versionsnummer vom Power-LP angepasst und in die Info-Übersicht des Power-LPs eingefügt:



8. Fazit

- War spannend und lehrreich.
- Programmiersprache C++ kennen gelernt.
- Verschiedene Lösungsmöglichkeiten kennengelernt.
- Konnten unser Wissen erweitern.
- Selbstständiges Arbeiten an einem „realen“ Projekt.
- viel Stress und Probleme, aber man lernt daraus.
- Projekt erfolgreich beendet.



Herzlichen Dank
für Ihre Aufmerksamkeit!