

Grau d'Estadística UB-UPC

Programació Lineal

Laboratori 1

SAS/OR i PROC OPTMODEL

F.-Javier Heredia

<http://gnom.upc.edu/heredia>



**UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH**

**Departament d'Estadística
i Investigació Operativa**



This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/3.0/>

Resolució de problemes de PL amb SAS/OR

1. Introducció a SAS/OR.

2. Formulació de problemes de (PL) amb PROC OPTMODEL⁽¹⁾.

- Presentació del problema steel.
- Resolució amb PROC LP.
- Resolució amb PROC OPTMODEL.

(1) SAS/OR® 9.3 User's Guide Mathematical Programming, cap. 8

HTML:

http://support.sas.com/documentation/cdl/en/ormpug/65554/HTML/default/viewer.htm#ormpug_optmodel_toc.htm

PDF: <http://support.sas.com/documentation/cdl/en/ormpug/65554/PDF/default/ormpug.pdf>

SAS/OR® Software (font: <http://support.sas.com/rnd/app/or.html>)

SAS/OR software integrates essential optimization, scheduling, simulation, and related modeling and solution capabilities in an adaptable environment. Its **powerful set of management science solutions provide companies the knowledge they need to identify and optimize business processes and management challenges. SAS/OR software is designed for people with operations research/management science or similar training who are seeking to build and solve decision guidance models** that use one or more of the following operations research techniques:

- **mathematical programming**
- discrete event simulation
- project and resource scheduling
- local search optimization
- decision analysis
- bill-of-material (BOM) processing

SAS/OR® Software

Mathematical Programming Tools (font: <http://support.sas.com/rnd/app/or/MP.html>)

Linear Programming

- The OPTMODEL procedure solves linear programs. The syntax of the procedure's modeling language enables you to express the problem in a form that very closely resembles the symbolic form. PROC OPTMODEL provides three LP solvers: **primal simplex**, **dual simplex**, and **interior-point**. The simplex solvers implement a **two-phase simplex method**. (...).

Mixed-Integer Linear Programming

- The OPTMODEL procedure solves mixed-integer linear programming problems using the MILP solver. The MILP solver implements an **LP-based branch-and-bound algorithm**. The algorithm also implements advanced techniques including presolvers, **cutting planes**, and **primal heuristics**. The resulting improvements in efficiency enable you to use PROC OPTMODEL to solve larger and more complex optimization problems.

S'estudia a PLE

S'estudia al màster d'EIO UPC-UB (MEIO)

Procedure OPTMODEL

- PROC OPTMODEL

- Llenguatge de modelització per a Programació Matemàtica: permet implementar i resoldre computacionalment problemes d'optimització **en una notació similar a la matemàtica**.
- Representa un avenç importantíssim respecte dels procediments usats a IIO (LP)
- Permet usar gran part dels optimitzadors de SAS/OR sota una interfície comuna:
 - ❖ LP , OPTLP – linear programming
 - ❖ MILP – mixed integer linear programming
 - ❖ NLPU – unconstrained nonlinear programming
 - ❖ IPNLP – interior point nonlinear programming
 - ❖ QP – quadratic programming (experimental)
 - ❖ SQP – sequential quadratic programming



Planificació de la producció: steel

- Una empresa de manufactura d'acer ha de programar la producció setmanal d'un taller de laminat que transforma planxes d'acer en tres tipus de peces, **bandes**, **espirals** i **plaques**, d'acord amb les següents dades:

Steel	Benefici unit.	Comandes setmanals	Capacitat mercat	Capacitat de la fase d'escalfat	Capacitat del fase de laminat
<i>Unitats</i>	(€/Tm)	(Tm)	(Tm)	(Tm/h)	(Tm/h)
Bandes	25	1.000	6.000	200	200
Espirals	30	500	4.000	200	140
Plaques	29	750	3.500	200	160
Hores setmanals disponibles				35	40

Resolució del prob. steel amb LP

Paràmetres:

- $n = 3$, nre. de productes a fabricar.
- $m = 2$, nre. de fases producció (recursos).
- $c = [25 \ 30 \ 29]'$, cost productes.
- $b = [35 \ 40]'$, màxim hores setmanals disponibles.
- $T = \begin{bmatrix} 200 & 200 & 200 \\ 200 & 140 & 160 \end{bmatrix}$, t_{ps} , hores necessàries fase $s \in \mathcal{S}$ per produir una Tm producte $p \in \mathcal{P}$
- $u = [6000 \ 4000 \ 3500]'$, capacitat mercat.
- $l = [1000 \ 500 \ 750]'$, comandes setmanals.

	Ben. unit.	Com. set.	Cap. mercat	reheat	roll
Unitats	(€/Tm)	(Tm)	(Tm)	(Tm/h)	(Tm/h)
bands	25	1.000	6.000	200	200
coils	30	500	4.000	200	140
plates	29	750	3.500	200	160
Hores setmanals disponibles				35	40

Variables: x_p , quantitat a fabricar producte $p \in \mathcal{P}$

Model matemàtic:

$$\begin{aligned}
 \text{(PL)} \quad & \left\{ \begin{array}{l} \max z = 25x_1 + 30x_2 + 29x_3 \\ \text{s. a. :} \\ \frac{1}{200}x_1 + \frac{1}{200}x_2 + \frac{1}{200}x_3 \leq 35 \\ \frac{1}{200}x_1 + \frac{1}{140}x_2 + \frac{1}{160}x_3 \leq 40 \\ [1000 \ 500 \ 750] \leq x \leq [6000 \ 4000 \ 3500] \end{array} \right.
 \end{aligned}$$

es maximitza el benefici total

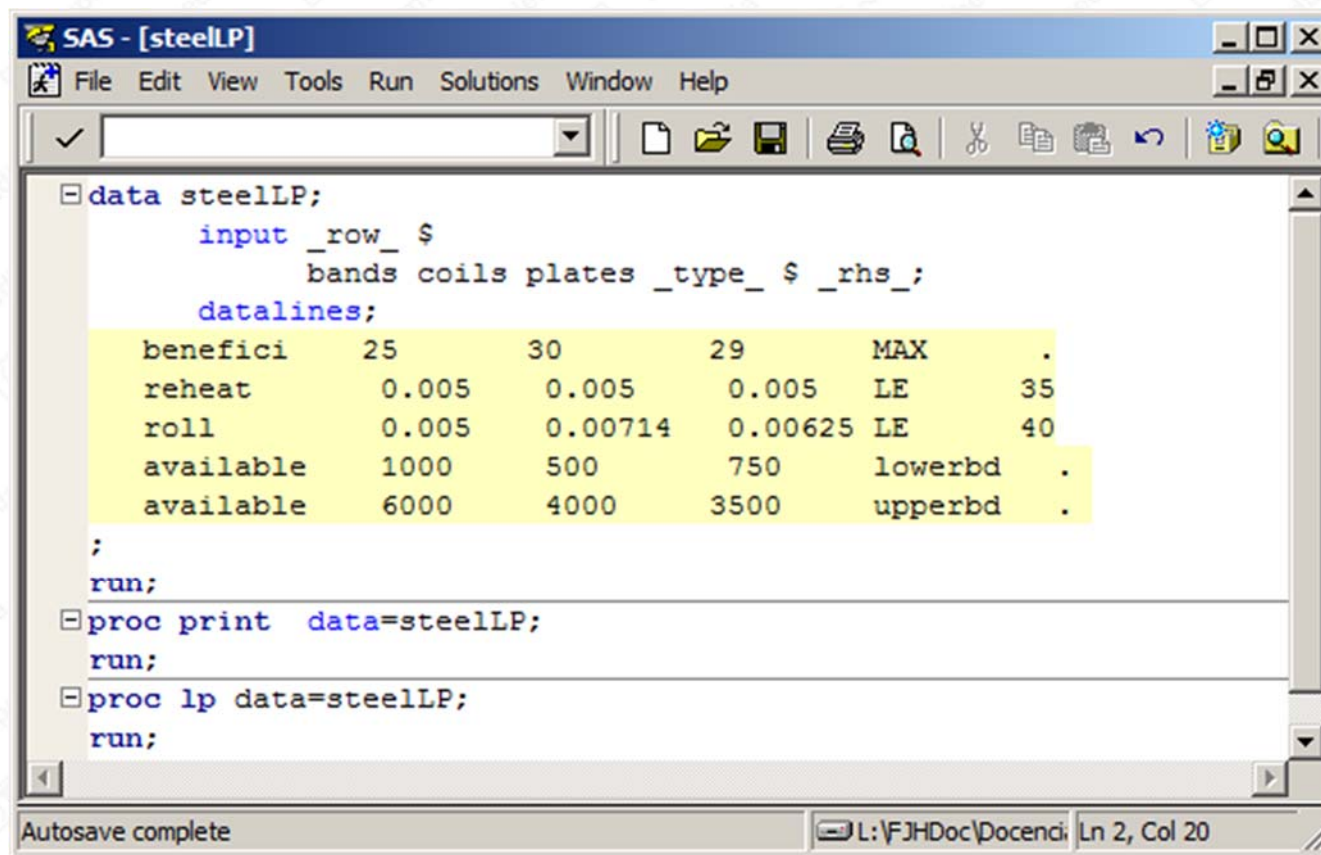
no es supera el nre. d'hores setmanals disponibles.

capacitat mercat i comandes

Resolució de steel amb LP: steelLP.sas

- Implementació i resolució:

$$(PL) \begin{cases} \max z = & 25x_1 + 30x_2 + 29x_3 \\ s. a.: & \\ & 0.005x_1 + 0.005x_2 + 0.005x_3 \leq 35 \\ & 0.005x_1 + 0.00714x_2 + 0.00625x_3 \leq 40 \\ & [1000 \ 500 \ 750] \leq x \leq [6000 \ 4000 \ 3500] \end{cases}$$



```
SAS - [steelLP]
File Edit View Tools Run Solutions Window Help

data steelLP;
  input _row_ $
        bands coils plates _type_ $ _rhs_;
  datalines;
benefici 25      30      29      MAX      .
reheat   0.005   0.005   0.005   LE      35
roll     0.005   0.00714 0.00625 LE      40
available 1000   500     750    lowerbd .
available 6000   4000    3500   upperbd .
;
run;

proc print data=steelLP;
run;

proc lp data=steelLP;
run;
```

Autosave complete L:\FJHDoc\Docenci Ln 2, Col 20

Resolució de steel amb LP: steelLP.sas

- Solució òptima

SAS - [Output - (Untitled)]

File Edit View Tools Solutions Window Help

Variable Summary

Col	Variable Name	Status	Type	Price	Activity	Reduced Cost
1	bands	BASIC	UPLOWBD	25	3356	0
2	coils	LOWBD	UPLOWBD	30	500	-1.848
3	plates	BASIC	UPLOWBD	29	3144	0
4	reheat		SLACK	0	0	-1800
5	roll		SLACK	0	0	-3200

The SAS System 18:27 Wednesday, February 6, 2013 15

The LP Procedure

Constraint Summary

Row	Constraint Name	Type	S/S Col	Rhs	Activity	Dual Activity
1	benefici	OBJECTIVE	.	0	190076	.
2	reheat	LE	4	35	35	1800
3	roll	LE	5	40	40	3200

NOTE: At left side.

L:\FJHDoc\Docencia\PLE\ple12\ple

Resolució del prob. steel amb OPTMODEL

Paràmetres:

- $\mathcal{P} = \{\text{bands}, \text{coils}, \text{plates}\}$, conjunt de prod. a fabricar.
- $\mathcal{S} = \{\text{reheat}, \text{roll}\}$, conjunt de fases producció.
- $c = [25 \ 30 \ 29]'$, cost productes.
- $b = [35 \ 40]'$, màxim hores setmanals disponibles.
- $T = \begin{bmatrix} 200 & 200 & 200 \\ 200 & 140 & 160 \end{bmatrix}$, t_{ps} , hores necessàries fase $s \in \mathcal{S}$ per produir una Tm producte $p \in \mathcal{P}$
- $u = [6000 \ 4000 \ 3500]'$, capacitat mercat.
- $l = [1000 \ 500 \ 750]'$, comandes setmanals.

	Ben. unit.	Com. set.	Cap. mercat	reheat	roll
Unitats	(€/Tm)	(Tm)	(Tm)	(Tm/h)	(Tm/h)
bands	25	1.000	6.000	200	200
coils	30	500	4.000	200	140
plates	29	750	3.500	200	160
Hores setmanals disponibles				35	40

Variables: x_p , quantitat a fabricar producte $p \in \mathcal{P}$

Model matemàtic:

$$\begin{aligned}
 \text{(PL)} \quad & \left\{ \begin{array}{ll} \max z = & \sum_{p \in \mathcal{P}} c_p x_p \\ \text{s. a.:} & \end{array} \right. \quad \begin{array}{l} \text{es maximitza el benefici total} \\ \\ \sum_{p \in \mathcal{P}} \frac{1}{t_{ps}} x_p \leq b_s, s \in \mathcal{S} & \text{no es supera el nre. d'hores setmanals disponibles.} \\ l_p \leq x_p \leq u_p, p \in \mathcal{P} & \text{capacitat mercat i comandes} \end{array}
 \end{aligned}$$

Implementació del problema: steel.sas (1/2)

- Paràmetres del model:

The screenshot shows the SAS PROC OPTMODEL window for the steel.sas problem. The code defines the following parameters:

Symbol	SAS Code
S	<code>set PROD = {'bands', 'coils', 'plate'};</code>
u	<code>set STAGE = {'reheat', 'roll'};</code>
T	<code>number profit(PROD) = [25 30 29];</code>
b	<code>number market(PROD) = [6000 4000 3500];</code>
P	<code>number commit(PROD) = [1000 500 750];</code>
c	<code>number rate(STAGE, PROD) = [200 200 200 200 140 160];</code>
l	<code>number avail(STAGE) = [35 40];</code>

File saved successfully. C:\Docume Ln 11, Col 49

Implementació del problema: steel.sas (1/2)

- Model matemàtic:

```
SAS - [steel.sas  PROC OPTMODEL running]
File Edit View Tools Run Solutions Window Help

/* Decision variables */
var Make{p in PROD} >= commit[p] <= market[p];

/* Objective function */
max Total_Profit= sum {p in PROD} profit[p] * Make[p];

/* Constraints */
constraint Time {s in STAGE}:
    sum {p in PROD} (1/rate[s,p]) * Make[p] <= avail[s];

File saved successfully. C:\Documents a Ln 25, Col 62
```

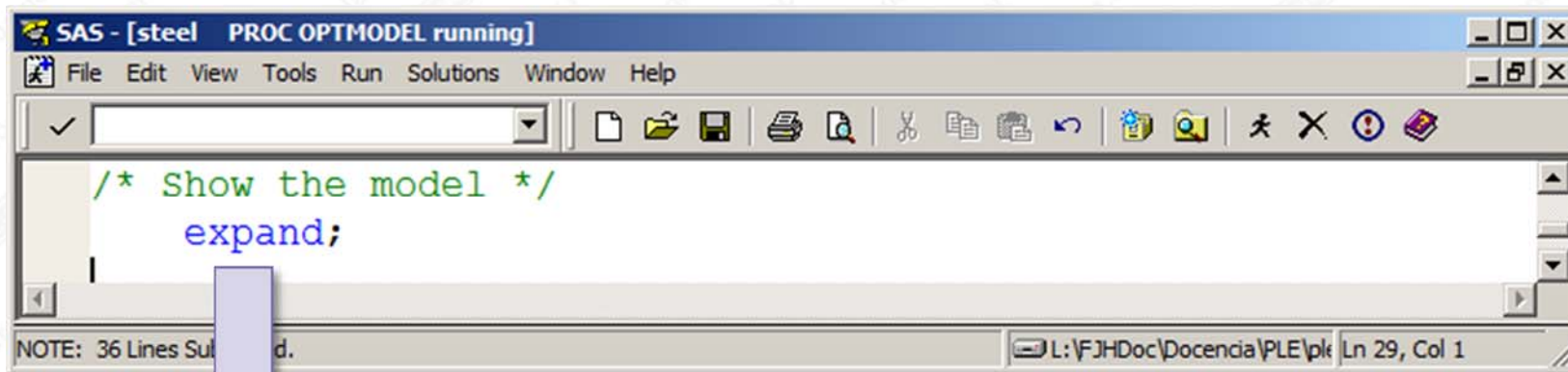
$$l_p \leq x_p \leq u_p \quad p \in \mathcal{P}$$

$$\max z = \sum_{p \in \mathcal{P}} c_p x_p$$

$$\sum_{p \in \mathcal{P}} a_{ps} x_p \leq b_s \quad s \in \mathcal{S}$$

Implementació del problema: steel.sas (1/2)

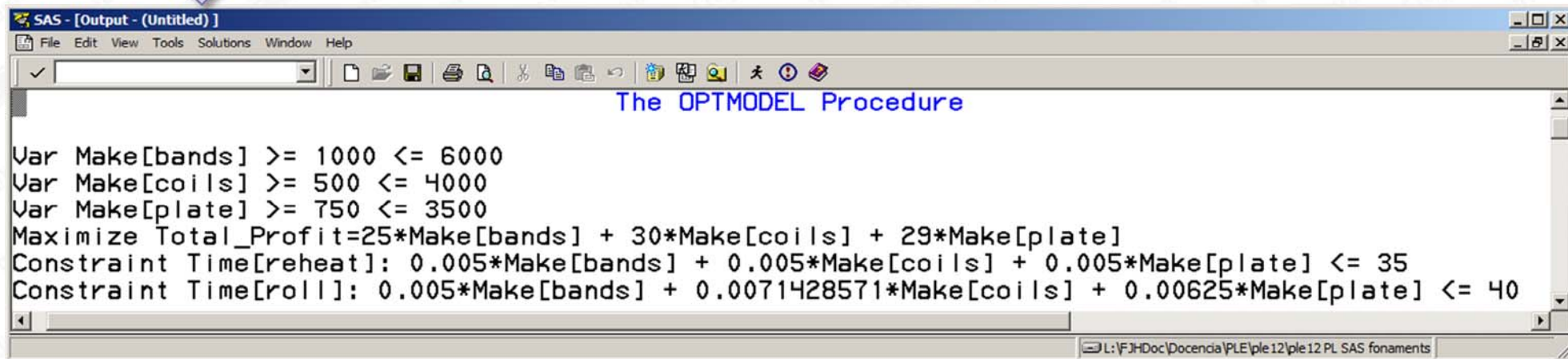
- Presentació del model:



```
/* Show the model */
expand;
```

NOTE: 36 Lines Submitted.

L:\FJHDoc\Docencia\PLE\ple\Ln 29, Col 1

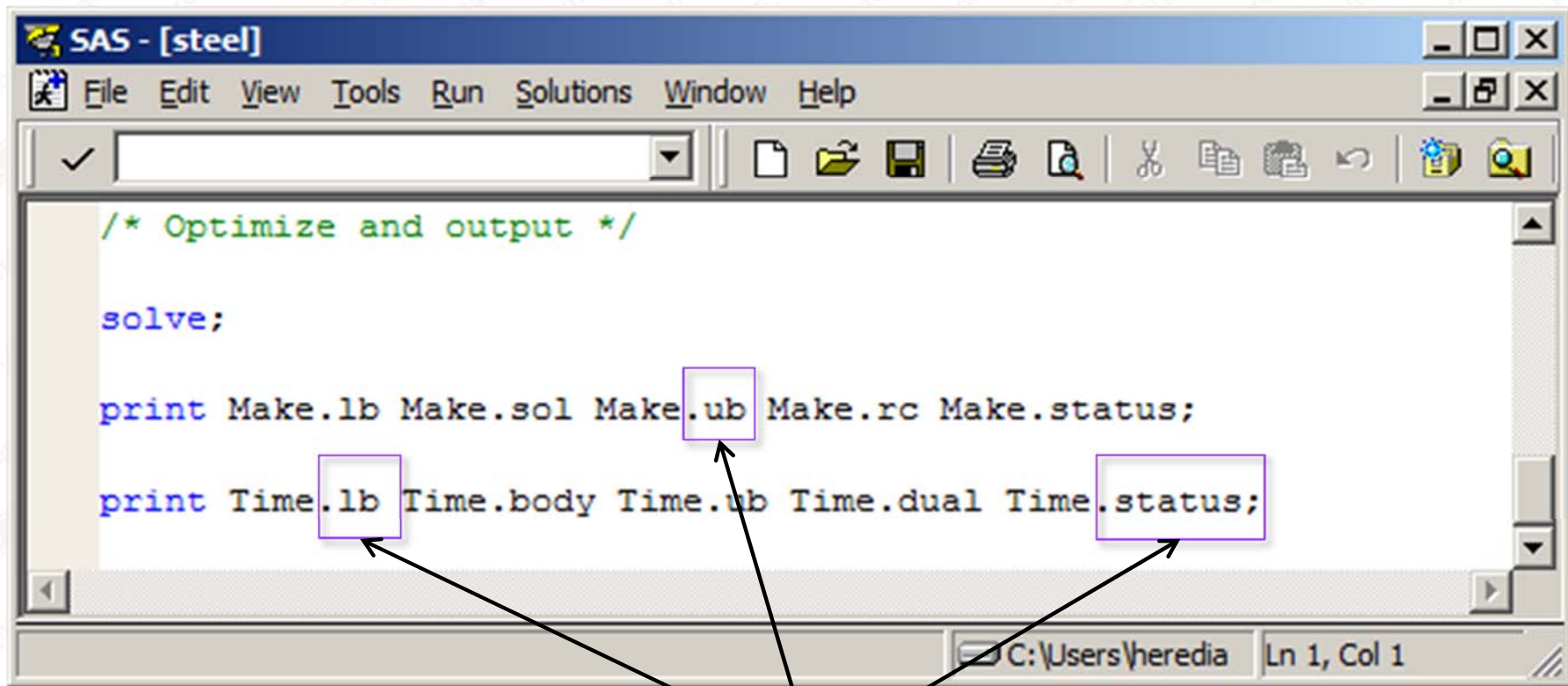


The OPTMODEL Procedure

```
Var Make[bands] >= 1000 <= 6000
Var Make[coils] >= 500 <= 4000
Var Make[plate] >= 750 <= 3500
Maximize Total_Profit=25*Make[bands] + 30*Make[coils] + 29*Make[plate]
Constraint Time[reheat]: 0.005*Make[bands] + 0.005*Make[coils] + 0.005*Make[plate] <= 35
Constraint Time[roll]: 0.005*Make[bands] + 0.0071428571*Make[coils] + 0.00625*Make[plate] <= 40
```

L:\FJHDoc\Docencia\PLE\ple\ple 12 PL SAS fonaments

- Optimització i sortida de resultats:



```
/* Optimize and output */  
  
solve;  
  
print Make.lb Make.sol Make.ub Make.rc Make.status;  
  
print Time.lb Time.body Time.ub Time.dual Time.status;
```

Sufixos: permeten accedir a informació detallada sobre l'estat de variables, f.o. i constriccions a l'òptim (s'estudiaran més endavant).

Resolució del prob. steel: Output

- Resultat de l'optimització:

The screenshot shows the SAS PROC OPTMODEL running interface. The main window displays the code: `/* Optimize and output */` and `solve;`. Below this, two output windows are shown. The left window, titled 'The OPTMODEL Procedure Problem Summary', displays the problem details. The right window, titled 'The OPTMODEL Procedure Solution Summary', displays the solution results. A large arrow points from the PROC OPTMODEL running window to the Problem Summary window.

The OPTMODEL Procedure Problem Summary

Objective Sense	Maximization
Objective Function	Total_Profit
Objective Type	Linear
Number of Variables	3
Bounded Above	0
Bounded Below	0
Bounded Below and Above	3
Free	0
Fixed	0
Number of Constraints	2
Linear LE (<=)	2
Linear EQ (=)	0
Linear GE (>=)	0
Linear Range	0

The OPTMODEL Procedure Solution Summary

Solver	Dual Simplex
Objective Function	Total_Profit
Solution Status	Optimal
Objective Value	190071.42857
Iterations	2
Primal Infeasibility	7.105427E-15
Dual Infeasibility	0
Bound Infeasibility	0

Make

[1]	
bands	3357.1
coils	500.0
plate	3142.9

NOTE: At bottom.

Resolució del prob. steel: Output

SAS - [steel] PROC OPTMODEL running

```

/* output */
print Make.lb Make.sol Make.ub Make.rc Make.status;
print Time.lb Time.body Time.ub Time.dual Time.status;
    
```

SAS - [Output - (Untitled)]

Table 1: Make Variables

[1]	Make.LB	Make.SOL	Make.UB	Make.RC	Make.STATUS
bands	1000	3357.1	6000	0.0000	B
coils	500	500.0	4000	-1.8571	L
plate	750	3142.9	3500	0.0000	B

Table 2: Time Variables

[1]	Time.LB	Time.BODY	Time.UB	Time.DUAL	Time.STATUS
reheat	-1.7977E308	35	35	1800	L
roll	-1.7977E308	40	40	3200	L

Mathematical Annotations:

- $p \in \mathcal{P}$ (rows of Table 1)
- l_p (Make.LB)
- x_p^* (Make.SOL)
- u_p (Make.UB)
- r_p (Make.RC)
- B^*, \mathcal{N}^* (Make.STATUS)
- $s \in \mathcal{S}$ (rows of Table 2)
- $\underline{b}_s = -\infty$ (Time.LB)
- $\sum_{p \in \mathcal{P}} a_{ps} x_p^*$ (Time.BODY)
- b_s (Time.UB)
- λ_s^* (Time.DUAL)
- B^*, \mathcal{N}^* (folgues) (Time.STATUS)

NOTE: At right side.

Sufijos (1/2)

- **Informació detallada sobre variables:**

Table 8.10 Suffix Names

Name\Kind	Suffix	Modifiable	Description
Variable	.init	No	Initial value for the solver
Variable	.lb	Yes	Lower bound
Variable	.ub	Yes	Upper bound
Variable	.sol	No	Current solution value
Variable	.rc	No	Reduced cost (LP) / gradient of Lagrangian function
Variable	.dual	No	Reduced cost (LP) / gradient of Lagrangian function
Variable	.relax	Yes	Relaxation of integrality restriction
Variable	.priority	Yes	Branching priority
Variable	.direction	Yes	Branching direction
Variable	.status	Yes	Status information from solver
Variable	.label	Yes	Label text for the solver

Sufixos (2/2)

- Informació detallada sobre f.o. i constriccions:

Table 8.10 Suffix Names

Name\Kind	Suffix	Modifiable	Description
Objective	.sol	No	Current objective value
Objective	.label	Yes	Label text for the solver
Constraint	.body	No	Current constraint body value
Constraint	.dual	No	Dual value from the solver
Constraint	.lb	Yes	Current lower bound
Constraint	.ub	Yes	Current upper bound
Constraint	.status	Yes	Status information from solver
Constraint	.label	Yes	Label text for the solver
Implicit Variable	.sol	No	Current solution value
Problem	.label	Yes	Label text for the solver
<i>any</i>	.name	No	Name text for any non-dummy symbol

