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

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

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 Mathematical Programming Tools

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 sover 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 procedimets 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
Unitats	(€/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**

	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

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 unaTm producte $p \in \mathcal{P}$
- $u = [6000\ 4000\ 3500]'$, capacitat mercat.
- $l = [1000\ 500\ 750]'$, comandes setmanals.

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

Model matemàtic:

$$
(\text{PL}) \begin{cases} \max z = & 25x_1 + 30x_2 + 29x_3 \\ s.a.: & \\ & \dfrac{1}{200}x_1 + \dfrac{1}{200}x_2 + \dfrac{1}{200}x_3 \le 35 \\ & \dfrac{1}{200}x_1 + \dfrac{1}{140}x_2 + \dfrac{1}{160}x_3 \le 40 \\ & [1000\ 500\ 750] \le x \le [6000\ 4000\ 3500] \end{cases}
$$

es maximitza el benefici total

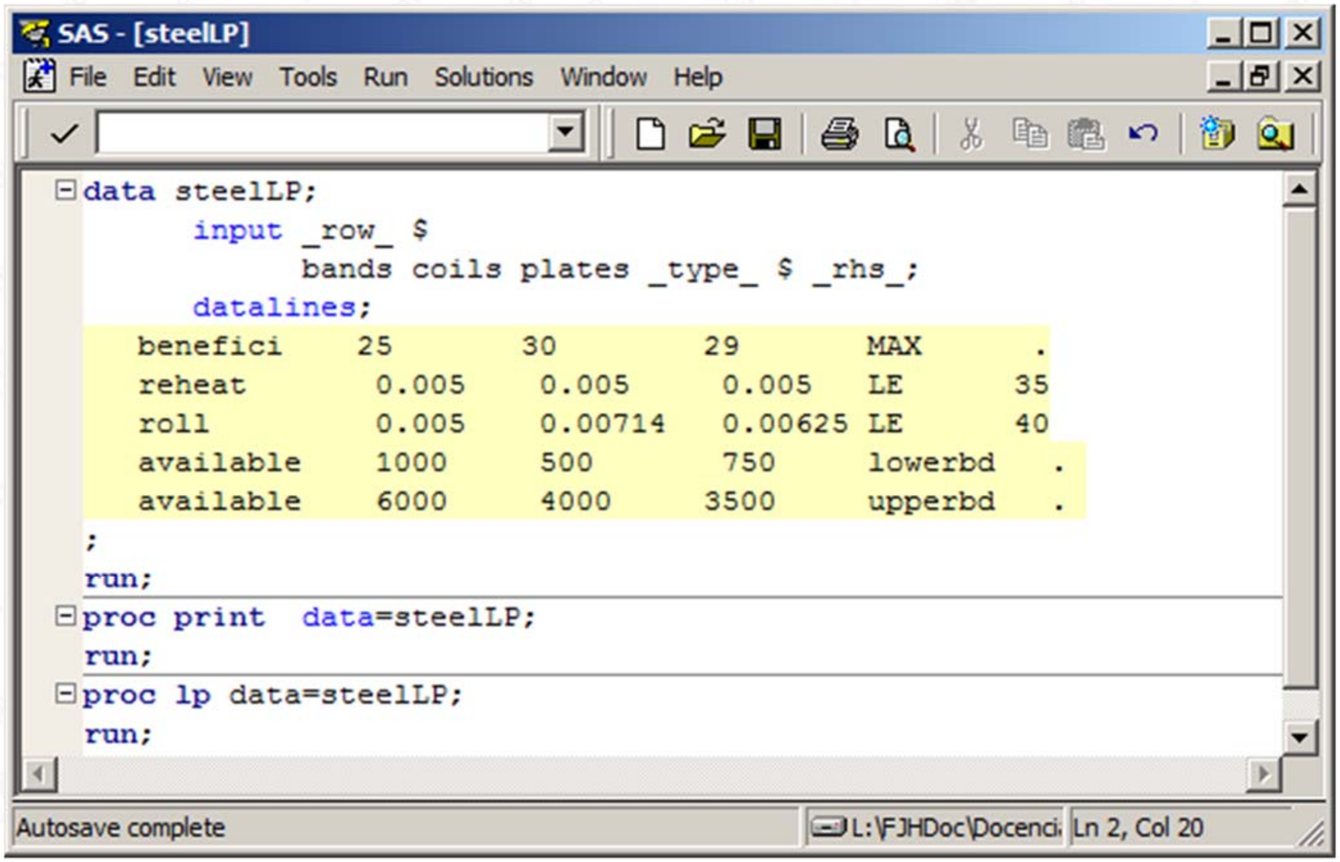
no es supera el nre. d'hores setmanals disponibles.

capacitat mercat i comandes

Resolució de `steel` amb `LP: steelLP.sas`

- **Implementació i resolució:**

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



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	OBJECTVE	.	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

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH
Departament d'Estadística i Investigació Operativa

Resolució del prob. `steel` amb OPTMODEL

	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

Paràmetres:

- $\mathcal{P} = \{bands, coils, plates\}$, **conjunt de prod. a fabricar.**
- $\mathcal{S} = \{reheat, roll\}$, **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 unaTm producte $p \in \mathcal{P}$
- $u = [6000\ 4000\ 3500]'$, capacitat mercat.
- $l = [1000\ 500\ 750]'$, comandes setmanals.

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

Model matemàtic:

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

- **Paràmetres del model:**

SAS - [steel.sas PROC OPTMODEL running]

File Edit View Tools Run Solutions Window Help

```
proc optmodel;

/* Parameters of the model */

set    PROD = {'bands', 'coils', 'plate'};
set STAGE = {'reheat', 'roll'};
number profit{PROD}        = [ 25 30 29 ];
number market{PROD}        = [ 6000 4000 3500 ];
number commit{PROD}        = [1000 500 750 ];
number rate{STAGE, PROD} = [ 200 200 200
                             200 140 160 ];
number avail{STAGE}        = [ 35 40 ];
```

$\mathcal{P}$ $\mathcal{S}$ c u l T b

File saved successfully. C:\Documer 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 fucntion */

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];
```

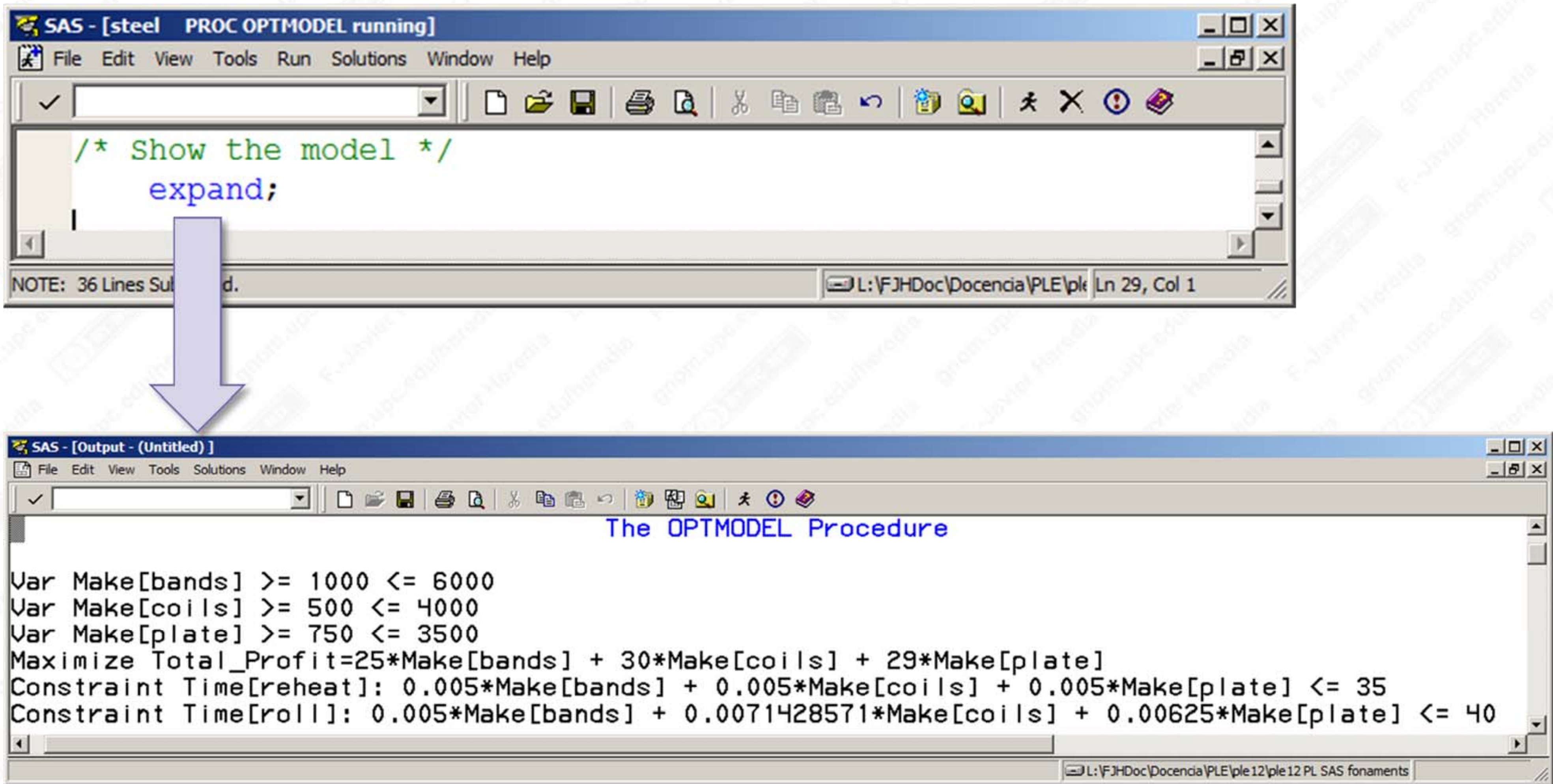
File saved successfully. C:\Documents a Ln 25, Col 62

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

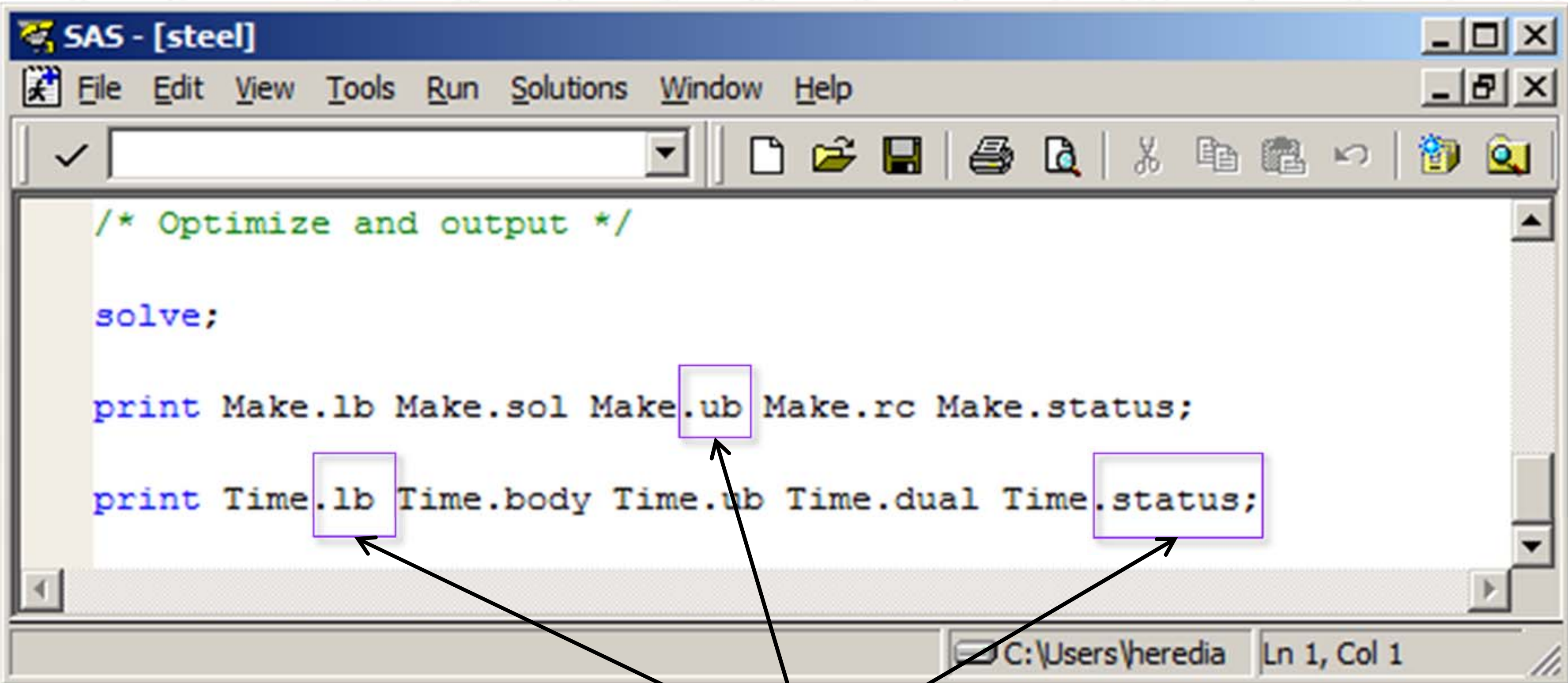
$$\max z = \sum_{p \in \mathcal{P}} c_p x_p$$

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

- **Presentació del model:**



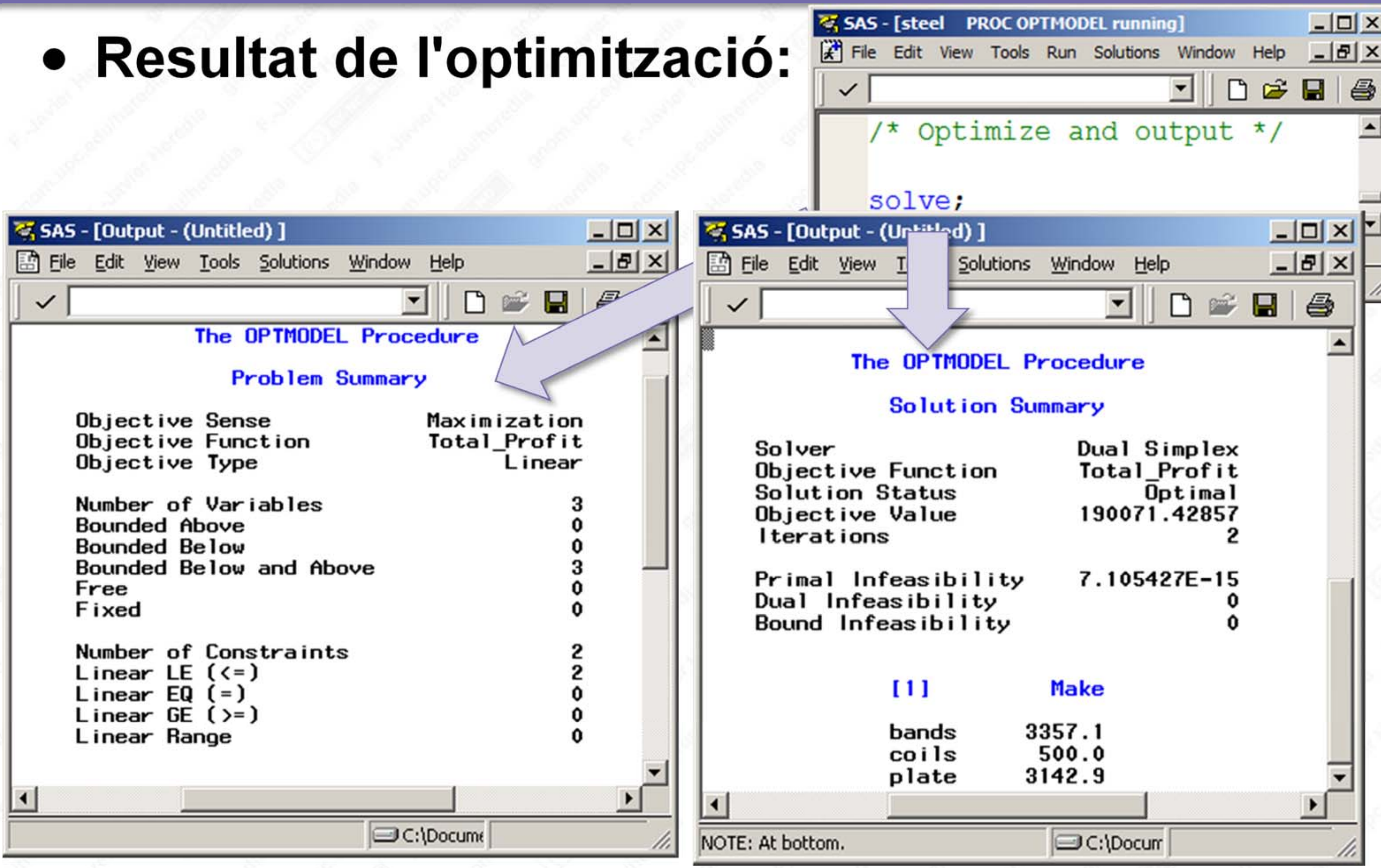
- **Optimització i sortida de resultats:**



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:**



Resolució del prob. steel: **Output**

SAS - [steel PROC OPTMODEL running]

File Edit View Tools Run Solutions Window Help

```
/* output */

print Make.lb Make.sol Make.ub Make.rc Make.status;

print Time.lb Time.body Time.ub Time.dual Time.status;
```

Autosave complete | L:\FJHDoc\Docencia\PLE | Ln 34, Col 13

SAS - [Output - (Untitled)]

File Edit View Tools Solu

[1] ($p \in \mathcal{P}$)	Make.LB (l_p)	Make.SOL (x_p^*)	Make.UB (u_p)	Make.RC (r_p)	Make.STATUS ($\mathcal{B}^*, \mathcal{N}^*$)
bands	1000	3357.1	6000	0.0000	B
coils	500	500.0	4000	-1.8571	L
plate	750	3142.9	3500	0.0000	B

[1] ($s \in S$)	Time.LB ($b_s = -\infty$)	Time.BODY ($\sum_{p\in\mathcal{P}} a_{ps}x_p^*$)	Time.UB (b_s)	Time.DUAL (λ_s^*)	Time.STATUS ($\mathcal{B}^*, \mathcal{N}^*$ (folgues))
reheat	-1.7977E308	35	35	1800	L
roll	-1.7977E308	40	40	3200	L

NOTE: At right side. | C:\Documents and S

Sufixos (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
any	.name	No	Name text for any non-dummy symbol