COSC 480/MATH 482 Operations Research Fall 2012 Alan C. Jamieson, PhD

Excel/Google Spreadsheet Solver Walkthrough

This document should give you a clear walkthrough of an example problem using Excel's Solver (both Windows and Mac) and Google Spreadsheets Solver. At each step, you'll see steps for Excel highlighted in green, and steps for Google Spreadsheets highlighted in red. Note that if you are using Excel 2010 or Mac Excel 2011, there's a good chance that your windows will look slightly differently than the screen shots I have in this document, but this should be good enough for you to figure things out.

Problem Statement

The WYNDOR GLASS CO. produces high-quality glass products, including windows and glass doors. It has three plants. Aluminum frames and hardware are made in Plant 1, wood frames are made in Plant 2, and Plant 3 produces the glass and assembles the products. The company is launching two new products having large sales potential:

Product 1: 8' glass door with aluminum framing (\$3000 profit/batch)

Product 2: 4' x 6' double-hung wood-framed window (\$5000 profit/batch)

Product 1 requires some of the production capacity in Plants 1 (1 hr/batch) and 3 (3 hr/batch). Product 2 requires Plants 2 (2 hr/batch) and 3 (2 hr/batch). Marketing has determined that the company could sell as many of either product as production allows. Determine what mix of products net the largest profit considering that the amount of available time for Plant 1 is 4 hours, Plant 2 is 12 hours, and Plant 3 is 18 hours per week.

Problem Formulation

Since we're going to be maximizing our profit ("Determine what mix of products net the largest profit") "Product 1: 8' glass door with aluminum framing (\$3000 profit/batch) Product 2: 4' x 6' double-hung wood-framed window (\$5000 profit/batch)" is our target function statement, since these are the statements that deal directly with profit. Our target function: Z = 3000a + 5000b

"Product 1 requires some of the production capacity in Plants 1 (1 hr/batch) and 3 (3 hr/batch). Product 2 requires Plants 2 (2 hr/batch) and 3 (2 hr/batch). Marketing has determined that the company could sell as many of either product as production allows. Determine what mix of products net the largest profit considering that the amount of available time for Plant 1 is 4 hours, Plant 2 is 12 hours, and Plant 3 is 18 hours per week." is where we see our constraints. The constraints are matched by color and form our full set of constraints:

```
a <= 4
2b <= 12 or b <= 6
3a + 2b <= 18
```

In addition, we can't allow negative production, so *a* and *b* will be non-negative.

In summary:

Maximize:	Z = 3000a + 5000b
Subject To:	a <= 4
	2b <= 12
	3a + 2b <= 18
With:	a, b >= 0

Using a Spreadsheet Solver - Setup

Now that we have our problem formulation, we need to put all of our relevant information into a spreadsheet. This is equivalent across all three potential solvers, so we only put one spreadsheet for now:

	E5							
	A	В	С	D	E	F	G	Н
1	Wyndor Gla	iss Co.						
2								
3			Doors	Windows				
4		Profit per Batch	\$3,000	\$5,000				
5					Hours		Hours	
6			Hours Used Per	Batch Produced	Used		Available	
7		Plant 1	1	0	0	<=	4	
8		Plant 2	0	2	0	<=	12	
9		Plant 3	3	2	0	<=	18	
10								
11							Total Profit	
12		Batches Produc	(0	0			0	
13								

Looking at specific cells (note that the highlights are completely optional):

G7, G8, G9 – these cells are the coefficients of the left hand sides of our three constraints

C7:C9, D7:D9 – these cells are the right hand sides of our three constraints

C4, D4 – these cells are our profit per unit for each of our products

C12, D12 – these cells we start as 0 and will be adjusted by the Solver. They represent the number of units of each product.

In addition, we'll need some cells that will do some calculations for us. Under "Hours Used":

E7 - "= C7*C12" E8 - "= D8*D12" E9 - "= C9*C12 + D9*D12" Each of these equations represent multiplying the hours used per unit for each plant (rows 7-9) by the number of units (row 12).

We'll also need one more calculation, for our total profits (our Z function):

G12 - "= C12*C4 + D12*D4"

Using a Spreadsheet Solver – Solving!

We now have our spreadsheet set up and we can move on to using the Solver to solve our linear programming problem. If you don't already have a variant of Solver installed, see the section on installation (Google's is built-in).

To start, we need to fire up Solver by clicking on <mark>Solver in Excel</mark> or <mark>Tools -> Solver in Google</mark> Spreadsheets.

To find Solver in Excel: Click on Data, then Solver.

ſ	Ca	39	• (* •) •		V	Vie no				-	-	ichant -	unit un	and	480L	ecture3Soln -	Microsoft E	xcel	unin	a the	e Coluer	to anh	-
1		Home	Insert	Page	Layout	Formulas	Data	Review	/	View												_	
							Conne	ections rties	ź↓	A Z Z A	Y	🕅 Clear	*				E ?	*	*	+	 Image: Show Detail Image: Hide Detail 	🍫 Solver	
ć	From Access	From Web	From From Text So	n Other urces *	Existing Connection	Refresh All *	🖙 Edit Li	nks	Z↓	Sort	Filter	Madvanced	Text to Columns	Remove Duplicates	Data Validation	Consolidate	What-If Analysis *	Group	Ungroup T	Subtotal			
ł.			Get Externa	l Data		C	onnection	s			Sort & Fil	ter			Data Too	ls			0	Dutline	5	Analysis	

For Mac, start the Solver app.

You should see one of the following screens (Windows, Mac, Google):

Solver Parameters	x	Solver Parameters	_	Solve			×
Set Target Cell: Solve		Set larget Cell: SAS1	Solve	Optimization type	Maximize \$	(linear only)	
Equal To: Max Min Value of: 0		Equal To: Max Min Value Of: 0	Close	Cell to optimize	Sheet1IA1		
By Changing Cells:	<u> </u>	By Changing Cells:	Options Reset All	Cells to change	Range	Restrict to >= 0	
Subject to the Constraints:	ns	Subject to the Constraints:	Help	Subject to these co	nditions		
^ <u>A</u> dd					Cell	<- \$ Cell/Value	
Change Reset	All	(Change) (Delete			add another		
- Diere Het				Solve Can	cel		

The first thing we'll set is the cell that we want to optimize, or the cell that has our Z function. We'll be setting Set Target Cell/Set Objective, or Cell to Optimize. For Google, you'll need to put the cell in manually by adjusting the "A1" portion of the Cell to Optimize box or by clicking on the textbox then clicking on the desired cell. For Office, you can click on the little button next to the text box highlighted below, which will allow you to click a specific cell.

	Solver Parameters
Solver Parameters	Solver Parameters Set Target Cell: SAS1 Solve Equal To: Max Min Value Of: O By Changing Cells: Options Reset All Subject to the Constraints: Add Change Delete Delete

In either case, you want to make sure that the cell is the right one. In our example above, we want this cell to be G12, but it may be different depending on how you have things formatted.

The next item to address is to choose what cells are to be manipulated by the software in order to maximize (or minimize as the case may be) our target cell. In Excel, this is handled by the "By Changing Cells:" textbox, in Google this is taken care of by the "Cells to Change" textbox. One special note – the Google textbox allows you to put your non-negative conditions in at the same time. Just check the box next to the textbox. Again, in the Excel version, we have a button to the right side of the textbox that allows us to choose our cells, and in the Google version, we have to manually enter that information or click on the textbox then click on the appropriate cell. The various pieces are highlighted below:

For our example, we want the cells C12 and D12 because the numbers of units that we produce are the parameters we can directly control.

Solver Parameters	Solver Parameters Set Target Cell: \$6\$12	Solve	Solve	×
Set Target Cel: 5C512 [%] Solve Equal To: @ Max @ Mig @ Yalue of: 0 Cose By Changing Cells: 5C512:SD512 [%] Guess Subject to the Constraints:	Equal To: Max Min Value Of: 0 By Changion Calls Subject to the Constraints: Add Change Delete	Close Options Reset All Help	Optimization type Kaximize (linear only) Cell to optimize Sheet11G12 Cells to change Sheet11C12D12 CR Restrict to >= 0 Subject to these conditions Cell Cell Cell/Value add another Solve Carcel	

Finally, we add in our constraints. This is done through the "Add" button on Excel variants, and by the appropriate textbox on Google Spreadsheets. The first constraint has already been added in both examples.

		Solve	×
Solver Parameters	Solver Parameters	Optimization type Maximize \$ (linear only)	
Set Target Cell: Solve	Set Target Cell: SGS12 🖼 Solve	Cell to optimize Sheet1IG12	
Equal To: Max Min Value of:	Equal To: Ax Nin Value Of: O Close	Cells to change Sheet1IC12:D12	
\$C\$12:\$D\$12	SC\$12:SD\$12 Guess Reset All	Subject to these conditions	-
Subject to the Constraints: Options	Subject to the Constraints: Help	Sheet1/E7 <= \$ Sheet1/G7	
\$E\$7 <= \$G\$7 ^ Add	SES7 <= SGS7	Cell <= \$ Cell/Value]
Change Reset All	Change	add another	
The Delete The Help	Delete	Solve Cancel	

When you click Add in Excel, you'll get some options:

Add Constraint)
Cell Reference:	Constraint:
	<= <
ОК	Cancel = Add Help
	int

<=, =, and >= are self-explanatory. int and bin are constraints that allow you to ensure that a particular cell remain an integer or a binary (0 or 1) value.

Google's solver only allows you <=, >=, and =:

0 0 0 0 0		(f)			
Optimization type	Maximize 🜩	(linear only)			
Cell to optimize	Sheet1!G12				
Cells to change	Sheet1!C12:D12	Restrict to >= 0			
Subject to these cor	ditions				
,	Cell	✓ <=	э		
,	Cell add another	✓ <= =	Э		

Now, add our constraints: E7 <= G7, E8 <= G8, and E9 <= G9. In Excel, you'll also need to add the nonnegativity constraints: C12 >= 0, D12 >= 0 (though you can do this in the options panel as well).

When finished, your solver should look like one of these three:

		Solver Parameters	
Solver Parameters Set Target Cell: SSS12 Equal To: Max By Changing Cells:	Solve Close	Set Target Cell: SC512 Equal To: Max Min Value Of: 0 By Changing Cells: SC512:SD512 Guess	Solve Close Options Reset All
$ \begin{array}{c} & & \\ & & \\ & & \\ & \\ & \\ & \\ & \\ & \\ $	Qptions Reset All	Subject to the Constraints: \$E57 <= \$C\$7	Help
	Help		-

Optimization type	Maximize \$	(linear only)	
Cell to optimize	Sheet1IG12		
Cells to change	Sheet1!C12:D12	Restrict t	D >= 0
Subject to these cor	ditions		
	Sheet1!E7	<= \$	Sheet1!G7
x	Sheet1!E8	<= \$	Sheet1!G8
х	Sheet1!E9	<= \$	Sheet1!G9
х	Cell	<= \$	Cell/Value
	add another		

Click "Solve" and magic happens! Feel free to mess around with the Options panel (only available in Excel). These options will allow you to further restrict your model and solver parameters.

Using a Spreadsheet Solver – Installation

Mac Office 2008 and 2011 – Go to <u>http://www.solver.com/mac/</u>, download the Solver application and install. You will run this at the same time as Excel.

Office 2007 and 2010 – Click on the Office button (top left), then click on Excel Options. Click on Add-Ins, then in the Manage box (bottom leftish) make sure that Excel Add-ins is selected and then click Go. In the Add-Ins available box, select Solver Add-in then hit ok and follow the prompt.

~ Fin ~