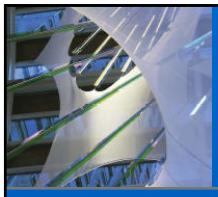
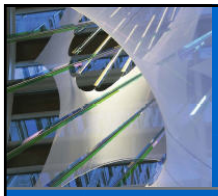


Tutorial 10

Using Solver For Complex Problems



Review

- 1 variable data tables:
  1 input cell, many results
- 2 variable data table:
  2 inputs, only one result
- Scenarios:
  Set up “what if” situations to compare what impact they have

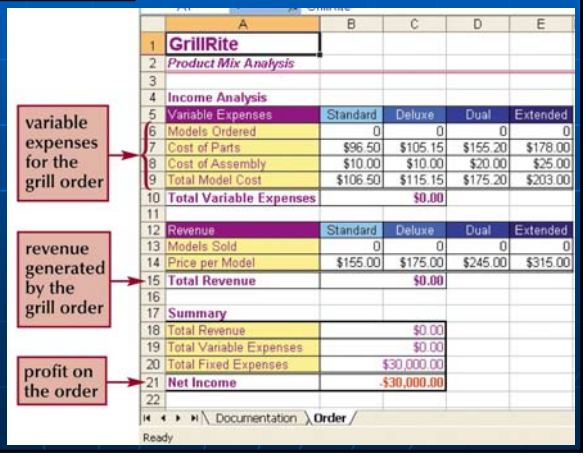
Formulate a problem

- *Solver* is an Excel tools that can help us solve complex, multi-variable problems.
- A typical multi-variable problem is optimizing the mix of several different products to be manufactured, stocked, and/or sold by a factory or business.
- In order to solve such a problem in Excel, it is first necessary to create an Excel workbook showing all of the interrelated facts.

Income Analysis portion for GrillRite

This figure shows an income analysis worksheet for a barbecue grill manufacturer.

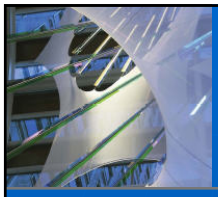
This worksheet will be referenced in other slides in this presentation to illustrate solving complex problems with Excel.



	A	B	C	D	E
1	GrillRite				
2	*Product Mix Analysis*				
3					
4	Income Analysis				
5	Variable Expenses	Standard	Deluxe	Dual	Extended
6	Models Ordered	0	0	0	0
7	Cost of Parts	$96.50	$105.15	$155.20	$178.00
8	Cost of Assembly	$10.00	$10.00	$20.00	$25.00
9	Total Model Cost	$106.50	$115.15	$175.20	$203.00
10	Total Variable Expenses		$0.00		
11					
12	Revenue	Standard	Deluxe	Dual	Extended
13	Models Sold	0	0	0	0
14	Price per Model	$155.00	$175.00	$245.00	$315.00
15	Total Revenue		$0.00		
16					
17	Summary				
18	Total Revenue		$0.00		
19	Total Variable Expenses		$0.00		
20	Total Fixed Expenses		$30,000.00		
21	Net Income		-$30,000.00		
22					

Documentation | Order

Ready

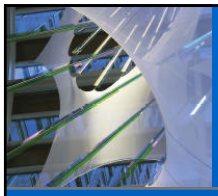


Grill Parts Inventory

This figure shows a grill parts inventory worksheet that will also be referenced in other slides in this presentation.

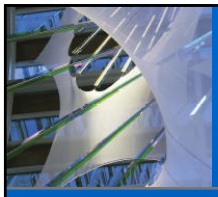
Parts Inventory

Description	Available	Ordered	Remaining	Cost
Burner (Main)	1411	0	1411	$12.15
Burner (Side)	957	0	957	$8.25
Chassis (Extended)	781	0	781	$65.25
Chassis (Standard)	877	0	877	$35.50
Control Dial	2671	0	2671	$2.50
Gas Tubing	2321	0	2321	$3.50
Igniter	2217	0	2217	$2.25
Igniter Button	1012	0	1012	$5.20
Rack (Main)	1417	0	1417	$10.30
Rack (Side)	723	0	723	$6.30
Rack (Top)	710	0	710	$6.15
Support Stand	1388	0	1388	$25.10



An example problem

- The problem addressed in this example is to optimize the profit that a grill manufacturing plant can make, by balancing the number of each kind of grill against inventory and revenue.
- Note the following factors :
  - The cost, both in parts and labor, to produce each kind of grill
  - The selling price of each grill
  - The total income
  - The current inventory of all parts
  - A chart of how many of each part are needed by each grill type

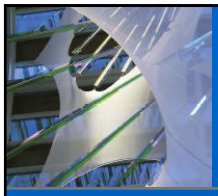


Mix-optimizing problem considerations

In every mix-optimizing problem, the constraints or requirements must be defined.

In our example, the solution must meet the following requirements:

1. There must be enough of each kind of grill to fill the current orders
2. No further parts can be ordered or manufactured, so no more grills can be manufactured once a part type is exhausted
3. The company wants a mix that will maximize profits



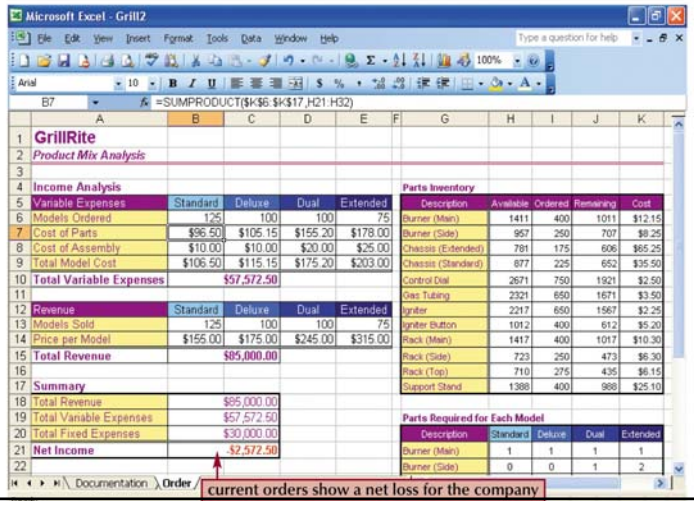
Perform what-if analyses

(the hard way - using trial and error)

- Once you have a worksheet set up that defines the problem, you can begin to do a what-if analysis.
- A good way to begin is to enter some reasonable numbers, and see what happens.
- This will also be a check to see if the worksheet is set up properly.

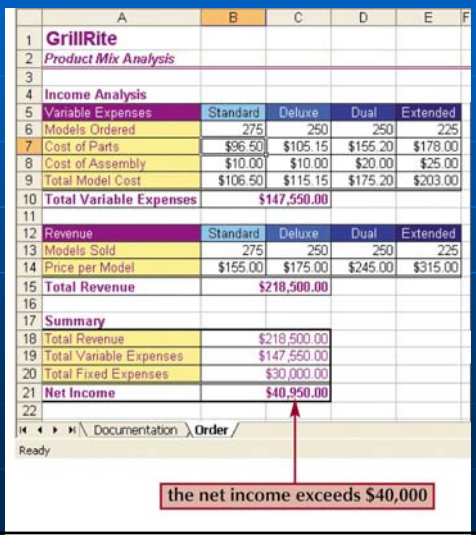
An income analysis for an order

Ordering enough grills to cover existing orders — Figure 10-3

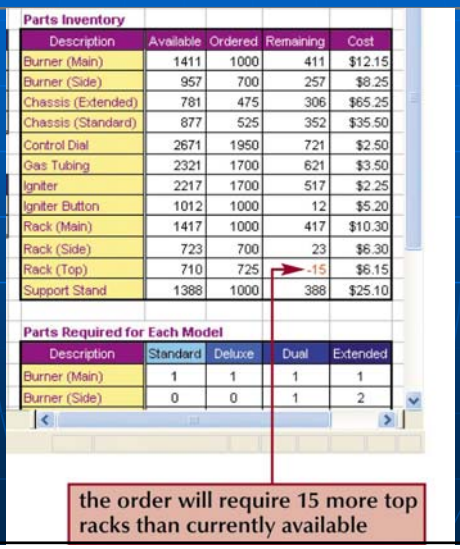


Now what-if? The order continued

Now the question is: How many more grills of each kind can be manufactured? What if we add 150 to the count of each model and order that many? This figure shows what happens with a guess—a what-if of 150 more of each model. The net income looks very good.

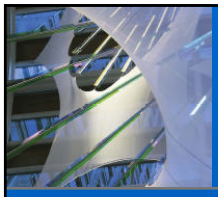


Recognizing a problem



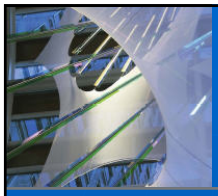
Explore other order possibilities

- If you increase the number of Extended grills by 15 and decrease the number of Deluxe grills by 15, the solution is better, but is it the best?
- We could keep on trying different entries for the count of different models of grill, trying to come up with the best solution, but continuing to use this manual trial-and-error method is tedious, and could take a long time.
- Fortunately, Excel has a tool that can help in trial-and-error problem solving.



Goal Seek - automates the trial-and-error process

- When you use Goal Seek, you specify the result you want, and Goal Seek changes the value in an input cell to arrive at that result.
- With Goal Seek, we specify the net income we want, and tell Goal Seek to change one of the inputs until it arrives at that goal.

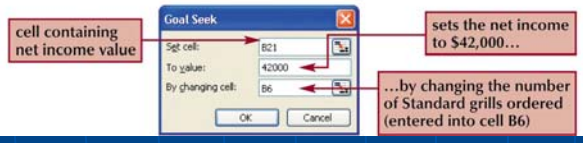


Using Goal Seek

- Suppose we want a net income of 42,000.
- From the *Tools* menu, click *Goal Seek*.
- In the Goal Seek dialog box, specify:
  - The cell to be set to some value
  - The value you want to be in that cell
  - The cell to change to accomplish the goal

The Goal Seek dialog box

Completed Goal Seek dialog box Figure 10-7



Understand the Goal Seek dialog box

- Goal Seek only allows you to enter one input cell to change.
- If you read the text of the entire Goal Seek dialog box as a sentence, it says, *Set cell B21 to value 42000 by changing cell B6*.
- Keeping that sentence in mind will help you know what Goal Seek is going to try to do.
- When it finds a solution, Goal Seek reports success in a Goal Seek Status box.

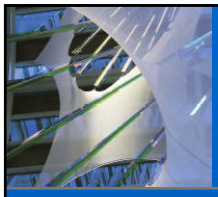
Problems

There are a couple of problems with this solution seen in the previous slide.:

1. you cannot order 331.2887 Standard grills.
2. there is a negative quantity in the *Remaining* column for the part called *Igniter Button*, which means you have exceeded the inventory for that part.

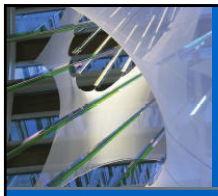
Fixing problems

- The first problem is caused by the fact that Goal Seek does not differentiate between integer and real numbers – manually change the number to the appropriate integer.
- To fix the second problem, you can run Goal Seek again, this time specifying that cell J13 (the remaining igniter buttons) should be set to 0, by changing cell B4.
- When you run Goal Seek with these values, the value displayed in B6 is 302. If you click on cell B6, you will see that the real value is 302.000000000002. You should fix this by manually entering 302 in cell B6.



The final Goal Seek solution

- After making the changes shown in the previous slide, check the inventory status, to see if there are any negative numbers.
- The number of igniter buttons remaining is zero; this product mix uses them all.
- Now, check the net income. It has fallen from $42,000 to just a little over $40,500, but we do have a workable solution that satisfies the first two constraints:
  - All current orders can be filled
  - The supply of parts has not been exceeded



Did Goal Seek provide the final answer?

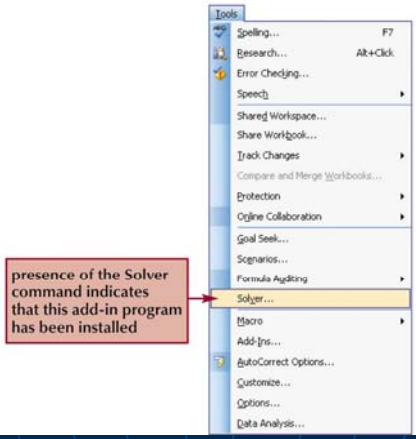
- Goal Seek helped find a solution that worked, but remember that our problem is to find the best solution—the one that yields the maximum net income.
- We could do several trials using Goal Seek, and see which one of them is better, but we won't know that we have found the best one.
- The only way to be absolutely sure that you have found the best solution using the trial-and-error method is to create all the solutions and then pick the best one.
- In the current problem, there are thousands of different combinations, certainly more than we want to run.

Use Solver to find the best solution

- Excel has an add-in program called Solver.
- Check the *Tools* menu to see if Solver is installed and activated on your machine.
- If it is, the word *Solver* will appear on the tools menu.
- You can activate it by clicking *Add-Ins* on the *Tools* menu.
  - This will bring up a list of the add-in programs available to activate
- Find Solver on the list, check its box, and then click *OK*.

Look for Solver on the Tools menu

Figure 10-10 Contents of the Tools menu



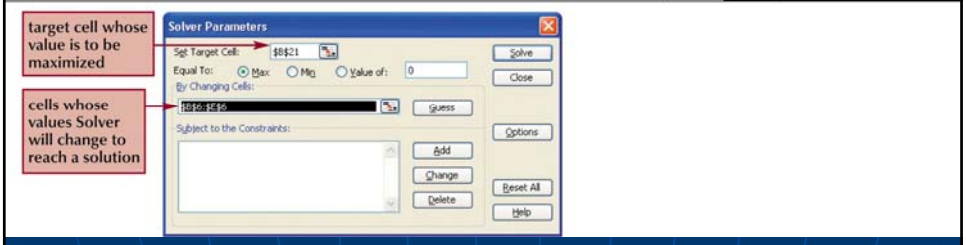
Information required by Solver

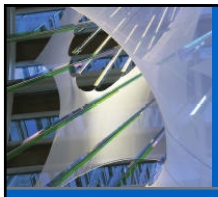
Solver needs four types of values that you will enter into the Solver Parameter dialog box:

1. It needs the address of the target cell.
2. It needs to know what you want to do with the target cell. You can ask Solver to set the target cell's value to the:
   - greatest possible value
   - the least possible value
   - to a certain value that you enter
3. Solver needs to know which cells it can change to arrive at the desired result.
4. Solver needs to know what constraints have to be applied to the solution.

The Solver Parameters dialog box

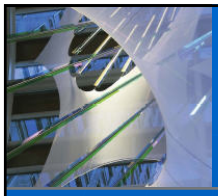
Solver Parameters dialog box **Figure 10-12**





Add Solver constraints

- The first requirement is that the number of each part cannot exceed the available supply of each part.
- Since the counts of available parts are in column H, and the count of the parts used in the order are in corresponding places in column I, the constraint is that the value in cell I6 must be less than or equal to the value in cell H6, the value in cell I7 must be less than or equal to the value in cell H7, and so on.



Constraint dialog box

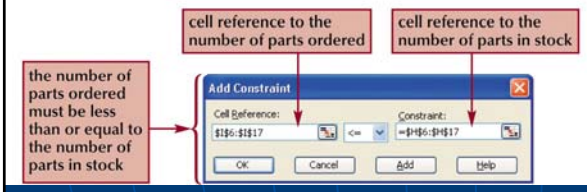
- Click *Add* in the *Constraints* section of the Solver Parameters dialog box.
- This brings up the Add Constraint dialog box, which has three boxes to enter values in.
  - The first box asks for references to the cells in question, in our case I6:I17
  - The second box lets you choose from several comparison options; we will choose "="
  - The third box asks you for the address of the constraining cells, in our case H6:H17
- Add additional constraints as needed for the problem being solved.

View the constraints

Figure 10-16 Adding new constraints to Solver

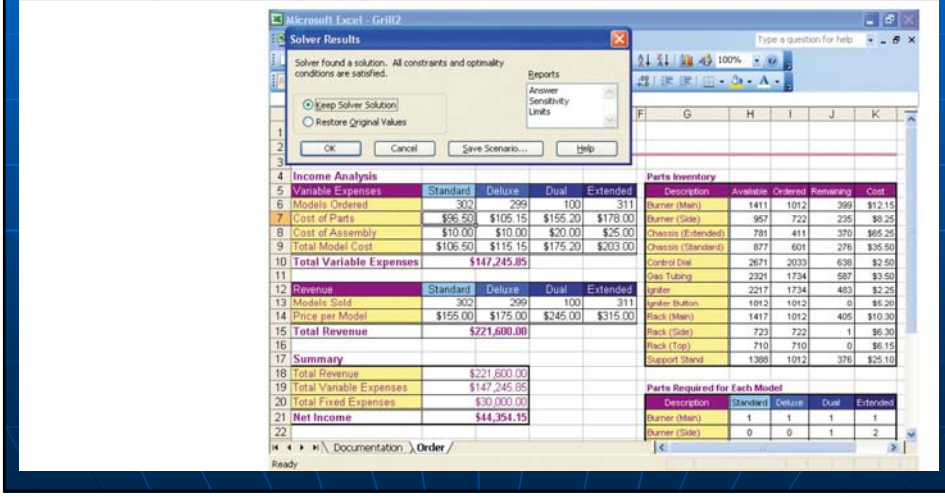
The Add Constraint dialog box

Add Constraint dialog box Figure 10-13



The Solver Results box

Figure 10-19 Final Solver solution

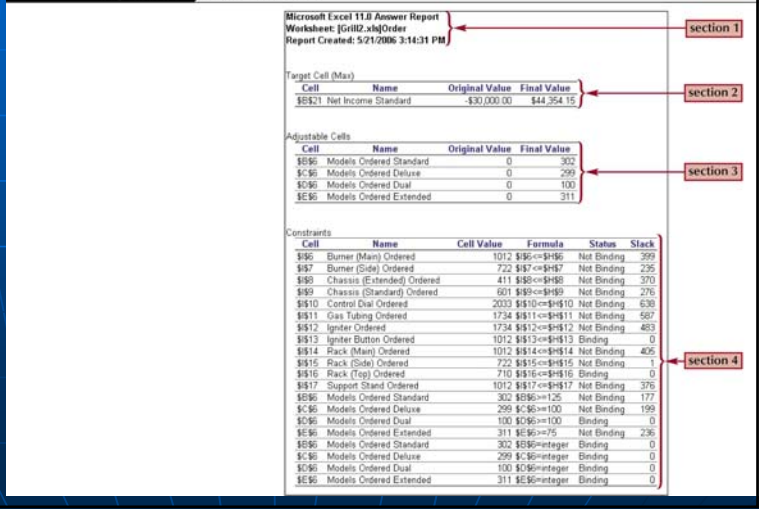


Create an answer report

- Create an answer report from the Solver Results window:
  - Set the changing cells (B6:E6) to zero
  - Bring up Solver and notice that the parameters, changing cells, and constraints are still just like you set them
  - Click the *Solve* button and wait for the Solver Results window to appear
  - Check the Keep Solver Solution check box, and click on *Answer* in the Reports box
- Solver creates the answer report and stores it in a separate worksheet called "Answer Report".

A Solver Answer Report

Figure 10-21 The answer report



Microsoft Excel 11.0 Answer Report
Worksheet: [Grill2.xls]Order
Report Created: 5/21/2006 3:14:31 PM

section 1

Target Cell (Max)

Cell	Name	Original Value	Final Value
$B$21	Net Income Standard	-$30,000.00	$44,354.15

section 2

Adjustable Cells

Cell	Name	Original Value	Final Value
$B$6	Models Ordered Standard	0	302
$C$6	Models Ordered Deluxe	0	299
$D$6	Models Ordered Dual	0	100
$E$6	Models Ordered Extended	0	311

section 3

Constraints

Cell	Name	Cell Value	Formula	Status	Slack
$I$6	Burner (Main) Ordered	1012	$I$6<=$H$6	Not Binding	399
$I$7	Burner (Side) Ordered	722	$I$7<=$H$7	Not Binding	235
$I$8	Chassis (Extended) Ordered	411	$I$8<=$H$8	Not Binding	370
$I$9	Chassis (Standard) Ordered	601	$I$9<=$H$9	Not Binding	276
$I$10	Control Dial Ordered	2033	$I$10<=$H$10	Not Binding	638
$I$11	Gas Tubing Ordered	1734	$I$11<=$H$11	Not Binding	587
$I$12	Igniter Ordered	1734	$I$12<=$H$12	Not Binding	483
$I$13	Igniter Button Ordered	1012	$I$13<=$H$13	Binding	0
$I$14	Rack (Main) Ordered	1012	$I$14<=$H$14	Not Binding	405
$I$15	Rack (Side) Ordered	722	$I$15<=$H$15	Not Binding	1
$I$16	Rack (Top) Ordered	710	$I$16<=$H$16	Binding	0
$I$17	Support Stand Ordered	1012	$I$17<=$H$17	Not Binding	376
$B$6	Models Ordered Standard	302	$B$6>=125	Not Binding	177
$C$6	Models Ordered Deluxe	299	$C$6>=100	Not Binding	199
$D$6	Models Ordered Dual	100	$D$6>=100	Binding	0
$E$6	Models Ordered Extended	311	$E$6>=75	Not Binding	236
$B$6	Models Ordered Standard	302	$B$6=integer	Binding	0
$C$6	Models Ordered Deluxe	299	$C$6=integer	Binding	0
$D$6	Models Ordered Dual	100	$D$6=integer	Binding	0
$E$6	Models Ordered Extended	311	$E$6=integer	Binding	0

section 4

Solver is designed to use an iterative process:

- It starts with an initial solution, and then does the problem over and over, using different values in the changing variables
- When a change to a variable results in a better solution, Solver makes another change to the same variable in the same direction
- When a change results in a worse solution, Solver does not make any more changes to that variable in that direction
- Solver continues to make changes and re-run the problem, until it arrives at a solution that is not significantly better than the previous one.
- At that point, Solver reports success.