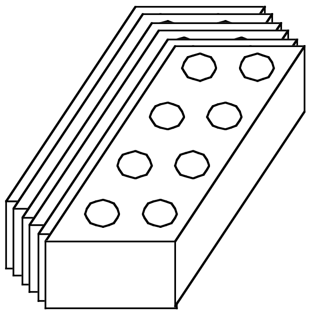
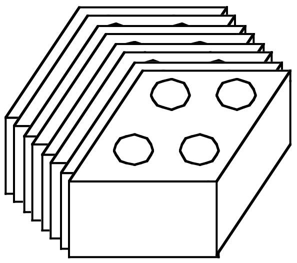
A Production Problem

Weekly supply of raw materials:



6 Large Bricks



8 Small Bricks

Products:

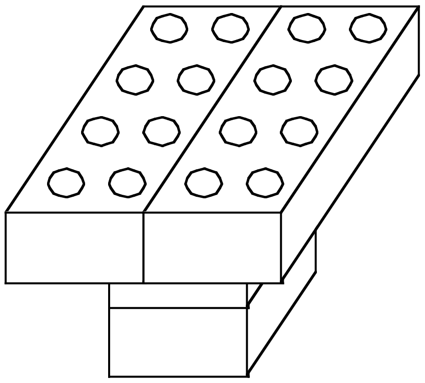
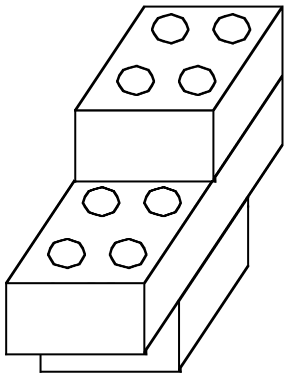


Table
Profit = $20/Table



Chair
Profit = $15/Chair

Linear Programming

Linear programming uses a mathematical model to find the best allocation of scarce resources to various activities so as to maximize profit or minimize cost.

Developing a Spreadsheet Model (Data Cells and Changing Cells)

Enter all of the data for the problem on the spreadsheet in a readable, easy to understand way. In this case, this consists of the profit data, the bill of materials, and the available raw materials. In a "real-world" problem, this data may already exist on a spreadsheet. It is a good idea to color-code all of these "data cells" (e.g., shade them blue), as these data are subject to change in a dynamic environment. Color-coding allows you to quickly see what data were used to find a solution, and change these later if necessary.

	B	C	D	E	F	G
3		Tables	Chairs			
4	Profit	$20.00	$15.00			
5						
6		Bill of Materials				Available
7	Large Bricks	2	1			6
8	Small Bricks	2	2			8

Next, add a cell in the spreadsheet for every decision that needs to be made—in this case, the production quantities. These decision variables are referred to as "changing cells", since these are the cells that change when making a decision. Again, it is a good idea to color code these "changing cells" (e.g., shade them yellow). If you don't have any particular initial values you want to enter for the changing cells, you can start by just entering a value of 0 in each cell.

	B	C	D	E	F	G
3		Tables	Chairs			
4	Profit	$20.00	$15.00			
5						
6		Bill of Materials				Available
7	Large Bricks	2	1			6
8	Small Bricks	2	2			8
9						
10		Tables	Chairs			
11	Production Quantity	0	0			

Developing a Spreadsheet Model (Target Cell)

Next, develop an equation which defines the objective of the model—i.e., the quantity that you want to maximize or minimize. Typically this equation involves both the data cells and the changing cells, in order to determine the quantity of interest (e.g., total profit or total cost). This cell is called the "target cell". Again it is a good idea to color code this cell (e.g., shade it orange).

	B	C	D	E	F	G
3		Tables	Chairs			
4	Profit	$20.00	$15.00			
5						
6		Bill of Materials				Available
7	Large Bricks	2	1			6
8	Small Bricks	2	2			8
9						
10		Tables	Chairs			Total Profit
11	Production Quantity	0	0			$0.00

The equation in the target cell (G11) is as follows:

	G
10	Total Profit
11	=SUMPRODUCT(C4:D4,C11:D11)

SUMPRODUCT sums the products of individual cells in two ranges. For example, SUMPRODUCT(C4:D4, C11:D11) sums the products C4*C11 plus D4*D11. The two specified ranges must be the same shape (same number of rows *and* columns). For *linear programming* you should *try to always use* the SUMPRODUCT function (or SUM) for the objective function, as this helps guarantee that the equation will be linear.

Developing a Spreadsheet Model (Constraints)

Finally, any constraints of the model need to be added to the spreadsheet. A constraint is of the form (Quantity A ≤ Quantity B). Also acceptable are = or ≥ type constraints. In this case, we must assure that the (total number of each type of brick used) ≤ (bricks available). Both sides of the constraint should be included on the spreadsheet. The number of available bricks is already included on the spreadsheet (G7 and G8), but we must calculate the number of bricks used (a function of the changing cells and bill of materials). The resulting spreadsheet model is as follows. The changing cells (Production Quantity) were changed to 1 to check the equations.

	B	C	D	E	F	G
3		Tables	Chairs			
4	Profit	$20.00	$15.00			
5						
6		Bill of Materials		Total Used		Available
7	Large Bricks	2	1	3	<=	6
8	Small Bricks	2	2	4	<=	8
9						
10		Tables	Chairs			Total Profit
11	Production Quantity	1	1			$35.00

The equations used for column E are shown below.

	E
6	Total Used
7	=SUMPRODUCT(C7:D7,$C$11:$D$11)
8	=SUMPRODUCT(C8:D8,$C$11:$D$11)

Absolute addressing (with a $ sign in front of the letter and number) is used for the production quantities. This way, the formula in cell E7 can be easily copied to cell E8. The cell addresses referring to the bill of materials (C7:D7) use relative addressing (without dollar signs) and so will adjust for the new row (C8:D8) when copied to cell E8, continuing to refer to the cells at the same relative position (the two cells immediately to the left). The cell addresses referring to the production quantities ($C$11:$D$11) continue to refer to the same row when copied to cell E8.

The <= entered in column F are entered to remind us that we want the cells in E7:E8 to be less than or equal to the cells in G7:G8.

Before calling on the Solver to find the optimal solution to the model, make sure there is a cell in your spreadsheet for each of the following:

- the quantity you wish to maximize or minimize
- every decision variable
- every quantity that you might want to constrain (include both sides of each constraint)

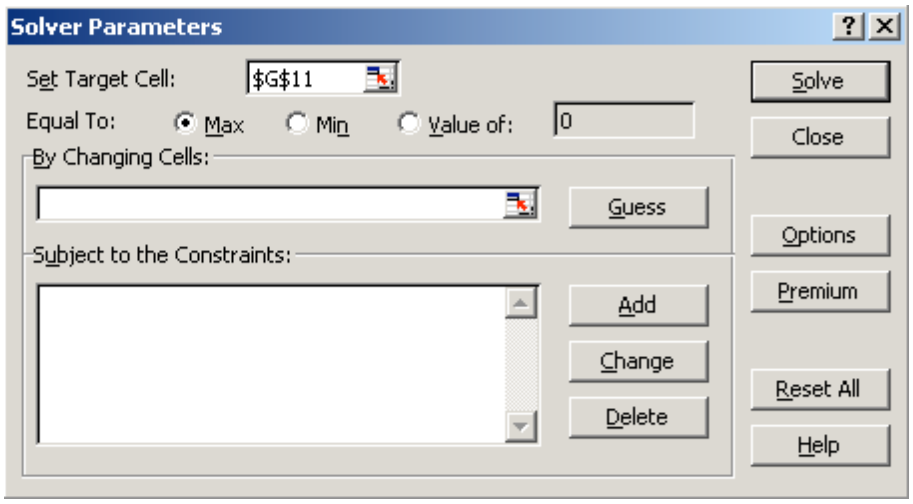
Defining the Target Cell

Choose the “Solver” from the Tools menu. If it is not available, you may need to go back to the install CD and be sure that the “Solver” add-in is installed (it is *not* installed in the “typical install” option for Excel).

To select the cell you wish to optimize, select the “Set Target Cell” window within the Solver dialogue box, and then either

- click on the cell you wish to optimize, or
- type the address of the cell you wish to optimize.
- Choose either “Max” or “Min” depending on whether the objective is to maximize or minimize the target cell.

	B	C	D	E	F	G
3		Tables	Chairs			
4	Profit	$20.00	$15.00			
5						
6		Bill of Materials		Total Used		Available
7	Large Bricks	2	1	3	<=	6
8	Small Bricks	2	2	4	<=	8
9						
10		Tables	Chairs			Total Profit
11	Production Quantity	1	1			$35.00


Solver Parameters

Set Target Cell: $G$11

Equal To: [✓ Max] Min Value of: 0

By Changing Cells:

Subject to the Constraints:

Note:

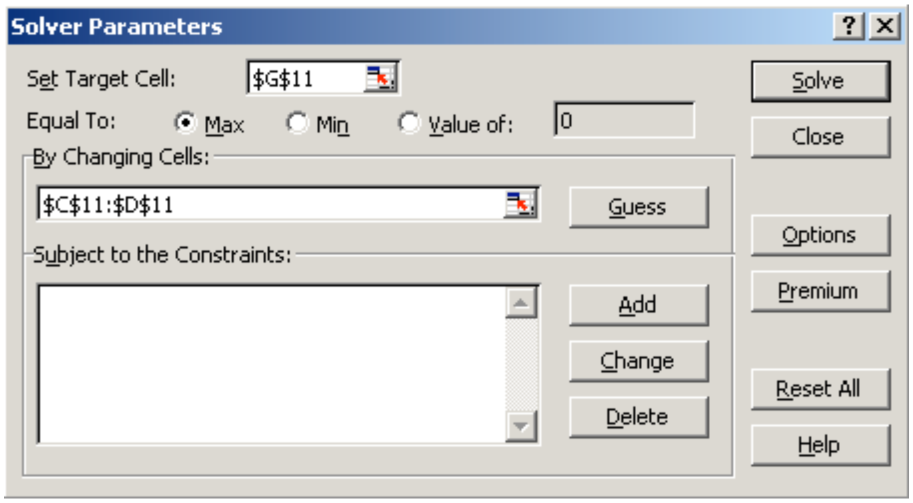
- The target cell must be a single cell (there can only be one objective)
- The target cell should contain an equation that defines the objective and depends on the decision variables

Identifying the Changing Cells

You next tell Excel which cells are decision variables—i.e., which cells Excel is allowed to change when trying to optimize. Move the cursor to the “By Changing Cells” window, and either

- drag the cursor across all cells you wish to treat as decision variables, or
- type the addresses of every cell you wish to treat as a decision variable, separating them by commas.

	B	C	D	E	F	G
3		Tables	Chairs			
4	Profit	$20.00	$15.00			
5						
6		Bill of Materials		Total Used		Available
7	Large Bricks	2	1	3	<=	6
8	Small Bricks	2	2	4	<=	8
9						
10		Tables	Chairs			Total Profit
11	Production Quantity	1	1			$35.00



Set Target Cell: $G$11

Equal To: [✓ Max] Min Value of: 0

By Changing Cells: $C$11:$D$11

Subject to the Constraints:

If you wish to use the “dragging” method, but the decision variables do not all lie in a connected rectangle in the spreadsheet you can “drag” them in one group at a time:

- drag the cursor across one group of decision variables,
- put a comma after that group in the “By Changing Cells” window,
- drag the cursor across the next group of decision variables,
- etc....

Adding Constraints

To begin entering constraints, click on the "Add" button to the right of the constraints window. A new dialogue box will appear. The cursor will be in the "Cell Reference" window within this dialogue box.

- click on the cell that contains the quantity you want to constrain, or
- type the cell address that contains the quantity you want to constrain.

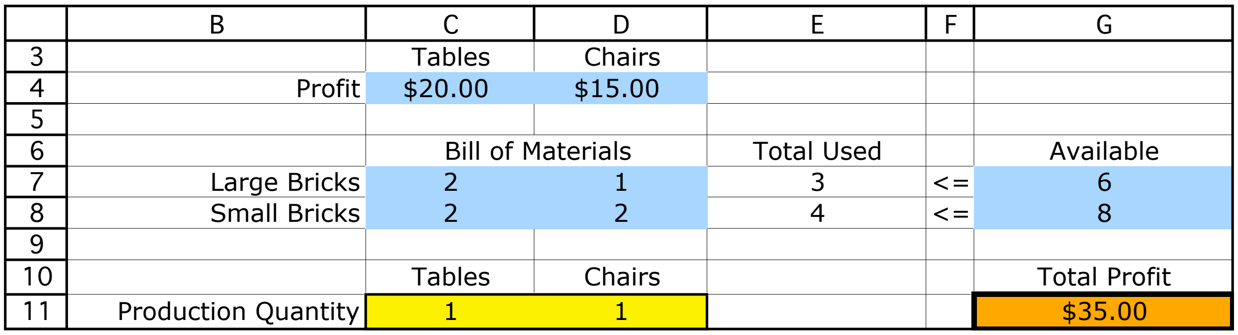
The default inequality that first appears for a constraint is "<=". To change this,

- click on the arrow beside the "<=" sign.
- Select the inequality (or equality) you wish from the list provided.

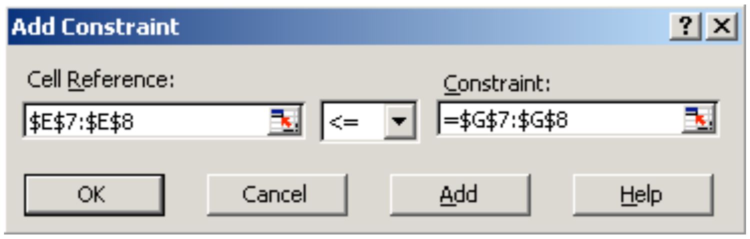
Notice that you may also force a decision variable to be an integer or binary (i.e., either 0 or 1) using this window. We will use this feature later in the course.

After setting the inequality, move the cursor to the "Constraint" window.

- click on the cell you want to use as the constraining value for that constraint, or
- type the number or the cell reference you want to use as the constraining value for that constraint, or
- type a number that you want to use as the constraining value.



	B	C	D	E	F	G
3		Tables	Chairs			
4	Profit	$20.00	$15.00			
5						
6		Bill of Materials		Total Used		Available
7	Large Bricks	2	1	3	<=	6
8	Small Bricks	2	2	4	<=	8
9						
10		Tables	Chairs			Total Profit
11	Production Quantity	1	1			$35.00



Add Constraint

Cell Reference: $E$7:$E$8

<=

Constraint: =$G$7:$G$8

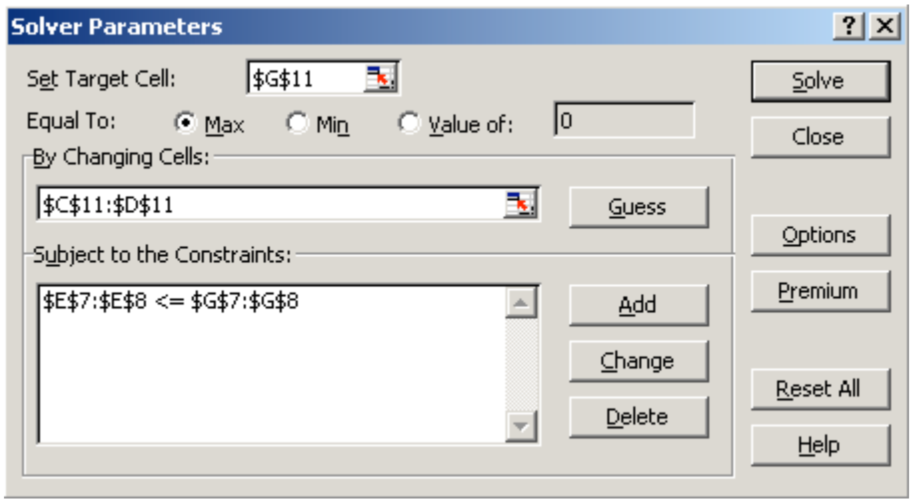
You may define a set of like constraints (e.g., all <= constraints, or all >= constraints) in one step if they are in adjacent rows (as was done here). Simply select the range of cells for the set of constraints in both the "Cell Reference" and "Constraint" window.

After you are satisfied with the constraint(s),

- click the "Add" button if you want to add another constraint, or
- click the "OK" button if you want to go back to the original dialogue box.

Some Important Options

The Solver dialogue box now contains the optimization model, including the target cell (objective function), changing cells (decision variables), and constraints.



Solver Parameters

Set Target Cell: $G$11

Equal To: [✓ Max] Min Value of: 0

By Changing Cells: $C$11:$D$11

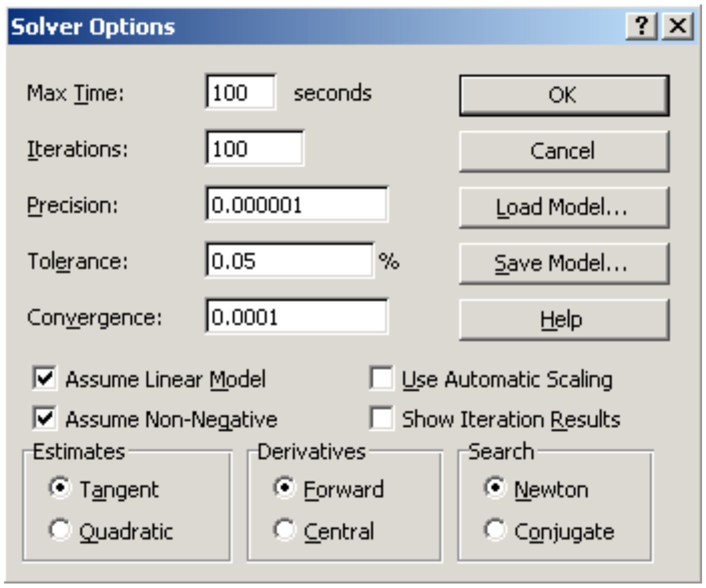
Subject to the Constraints:

$E$7:$E$8 <= $G$7:$G$8

Once you are satisfied with the optimization model you have input, there is one more very important step. Click on the "Options" button in the Solver dialogue box, and click in both the "Assume Linear Model" and the "Assume Non-Negative" box.

The "Assume Linear Model" option tells the Excel Solver that it is a *linear* program that is being solved. This speeds the solution process, makes it more accurate, and enables the more informative sensitivity report.

The "Assume Non-Negative" box adds non-negativity constraints to *all* of the decision variables.



Solver Options

Max Time: 100 seconds

Iterations: 100

Precision: 0.000001

Tolerance: 0.05 %

Convergence: 0.0001

[✓ Assume Linear Model] Use Automatic Scaling

[✓ Assume Non-Negative] Show Iteration Results

Estimates: [✓ Tangent] Quadratic

Derivatives: [✓ Forward] Central

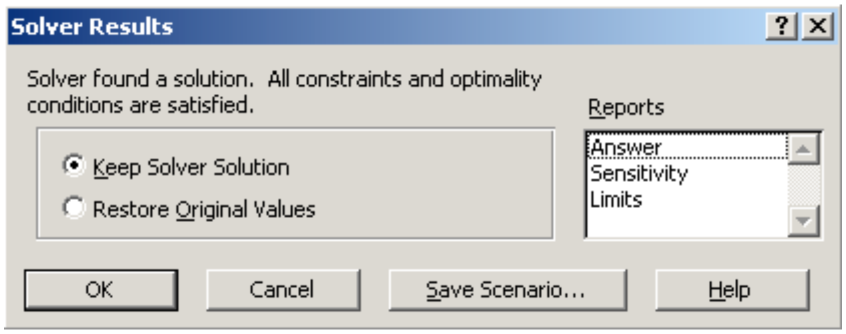
Search: [✓ Newton] Conjugate

The Solution

After setting up the model, and selecting the appropriate options, it is time to click "Solve". When it is done, you will receive one of four messages:

- "Solver found a solution. All constraints and optimality conditions are satisfied". This means that Solver has found the optimal solution.
- "Cell values did not converge". This means that the objective function can be improved to infinity. You may have forgotten a constraint (perhaps the non-negativity constraints) or made a mistake in a formula.
- "Solver could not find a feasible solution". This means that Solver could not find a feasible solution to the constraints you entered. You may have made a mistake in typing the constraints or in entering a formula in your spreadsheet. There is also a small chance that Solver has made an error. (This bug shows up occasionally—try immediately solving again and it will sometimes find a solution the second time. If it does not, chance are the model is not feasible.)
- "Conditions for Assume Linear Model not satisfied". You may have included a formula in your model that is nonlinear. Be sure that your objective function and constraints only include SUM or SUMPRODUCT functions, and don't multiply or divide changing cells together. There is also a small chance that Solver has made an error. (This bug shows up occasionally—try immediately solving again and it will sometimes now solve. If it does not, chance are the model is truly not linear.)

If Solver finds an optimal solution, you have some options. First, you must choose whether you want Solver to keep the optimal values in the spreadsheet (you usually want this one) or go back to the original numbers you typed in. Click the appropriate box to make you selection. You also get to choose what kind of reports you want. For our class, you will often want to select "Sensitivity Report". Once you have made your selections, click on "OK". To view the sensitivity report, click on the "Sensitivity Report" tab in the lower-left-hand corner of the window.



Solver found a solution. All constraints and optimality conditions are satisfied.

● Keep Solver Solution
○ Restore Original Values

Reports
Answer
Sensitivity
Limits

	B	C	D	E	F	G
3		Tables	Chairs			
4	Profit	$20.00	$15.00			
5						
6		Bill of Materials		Total Used		Available
7	Large Bricks	2	1	6	<=	6
8	Small Bricks	2	2	8	<=	8
9						
10		Tables	Chairs			Total Profit
11	Production Quantity	2	2			$70.00

Four Assumptions of Linear Programming

Linearity

Divisibility

Certainty

Nonnegativity

Why Use Linear Programming?

Linear programs are easy (efficient) to solve

The best (optimal) solution is guaranteed to be found (if it exists)

Useful *sensitivity analysis* information is generated

Many problems are *essentially* linear

Graphical Solution of Linear Programs

Maximize $Z = (\$15)C + (\$20)T$

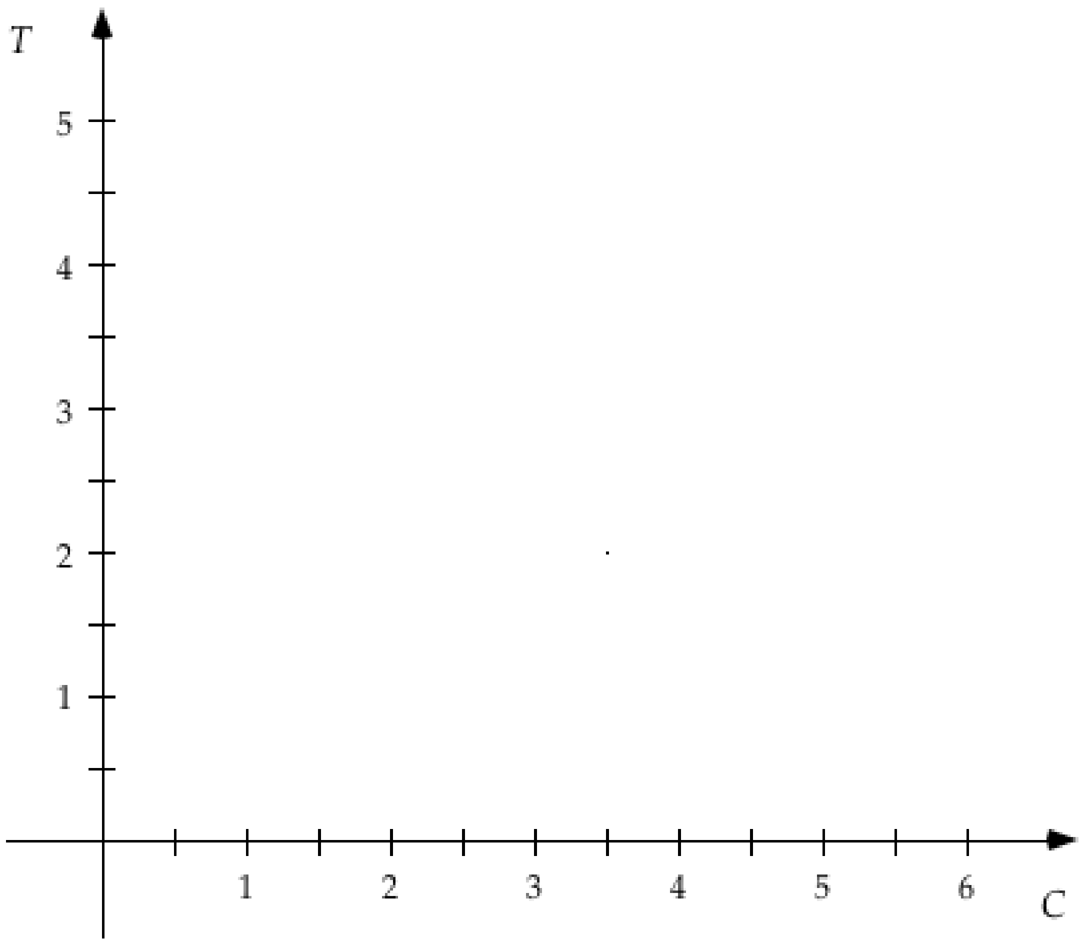
subject to

Large Bricks: $C + 2T \leq 6$

Small Bricks: $2C + 2T \leq 8$

and

$C \geq 0, \;\; T \geq 0.$



Properties of Linear Programming Solutions

1. An optimal solution must lie on the boundary of the feasible region.

2. There are exactly four possible outcomes of linear programming:

    a. A unique optimal solution is found.

    b. An infinite number of optimal solutions exist.

    c. No feasible solutions exist.

    d. The objective function is unbounded (there is no *optimal* solution).

3. If an LP has one optimal solution, it *must* be at a corner point.

4. If an LP has many optimal solutions, at least two of these optimal solutions are at corner points.

Example with Multiple Optimal Solutions

Maximize $Z = (\$15)C + (\$30)T$

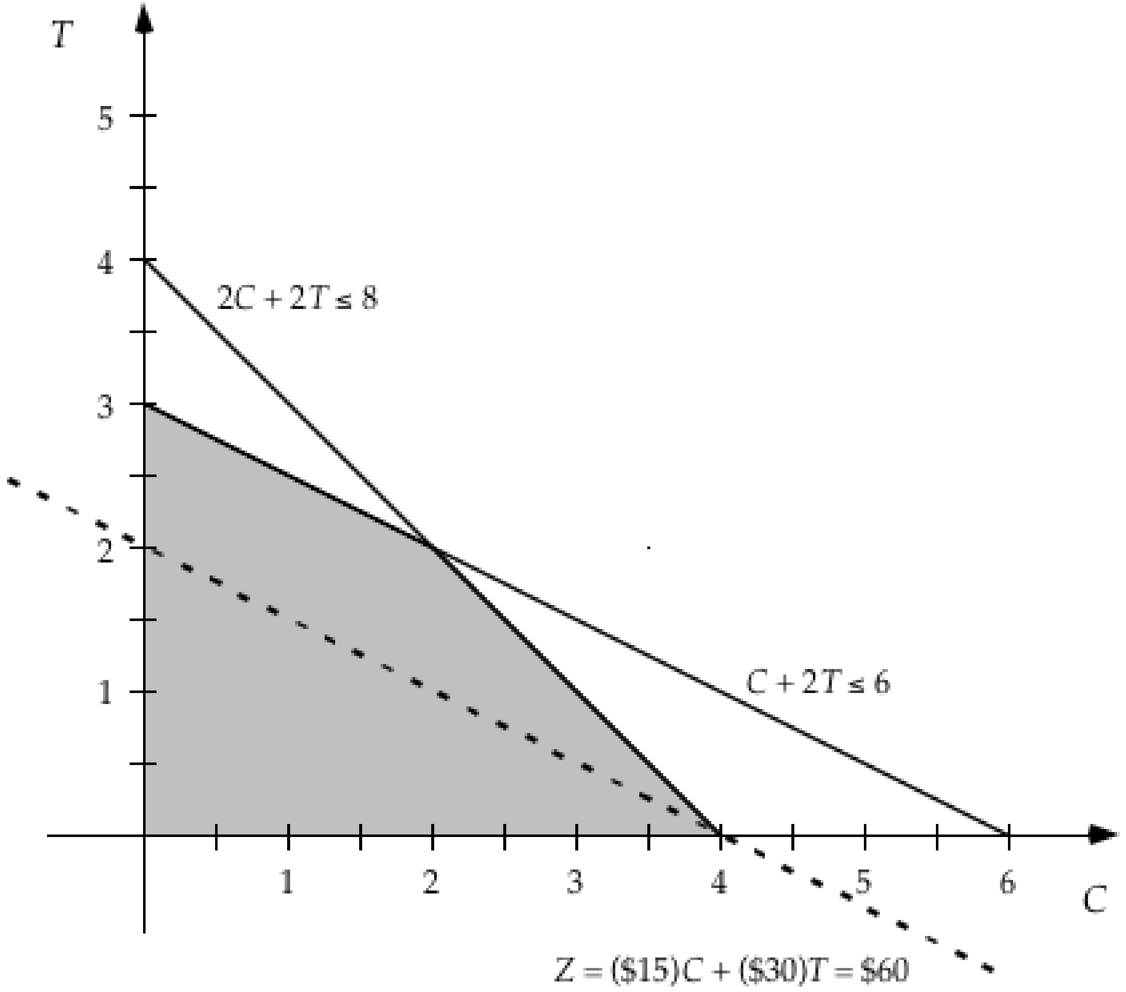
subject to

Large Bricks: $C + 2T \leq 6$

Small Bricks: $2C + 2T \leq 8$

and

$C \geq 0, \;\; T \geq 0.$



Example with No Feasible Solution

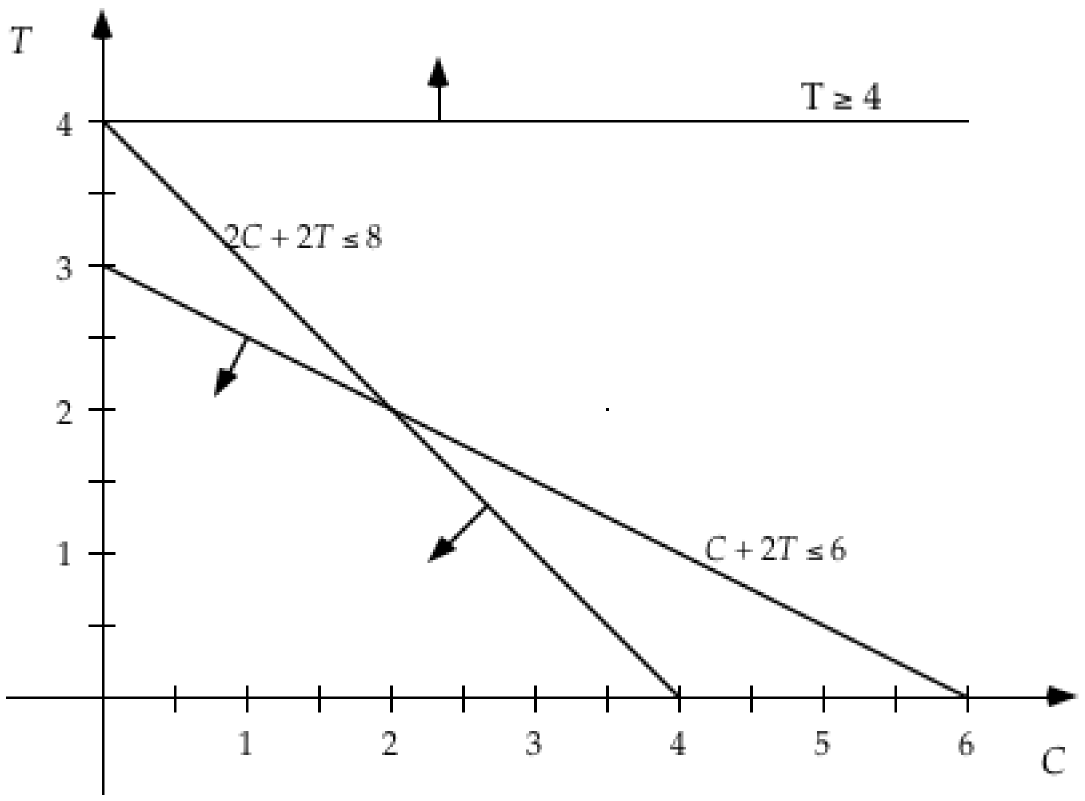
Maximize $Z = (\$15)C + (\$30)T$

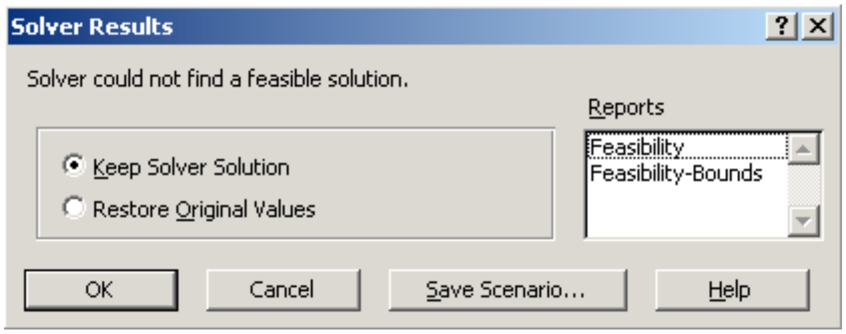
subject to

Large Bricks:	$C + 2T \leq 6$
Small Bricks:	$2C + 2T \leq 8$
Contract:	$T \geq 4$

and

$C \geq 0, \;\; T \geq 0.$




Solver Results

Solver could not find a feasible solution.

Keep Solver Solution [✓]
Restore Original Values

Reports: Feasibility, Feasibility-Bounds

Example with Unbounded Solution

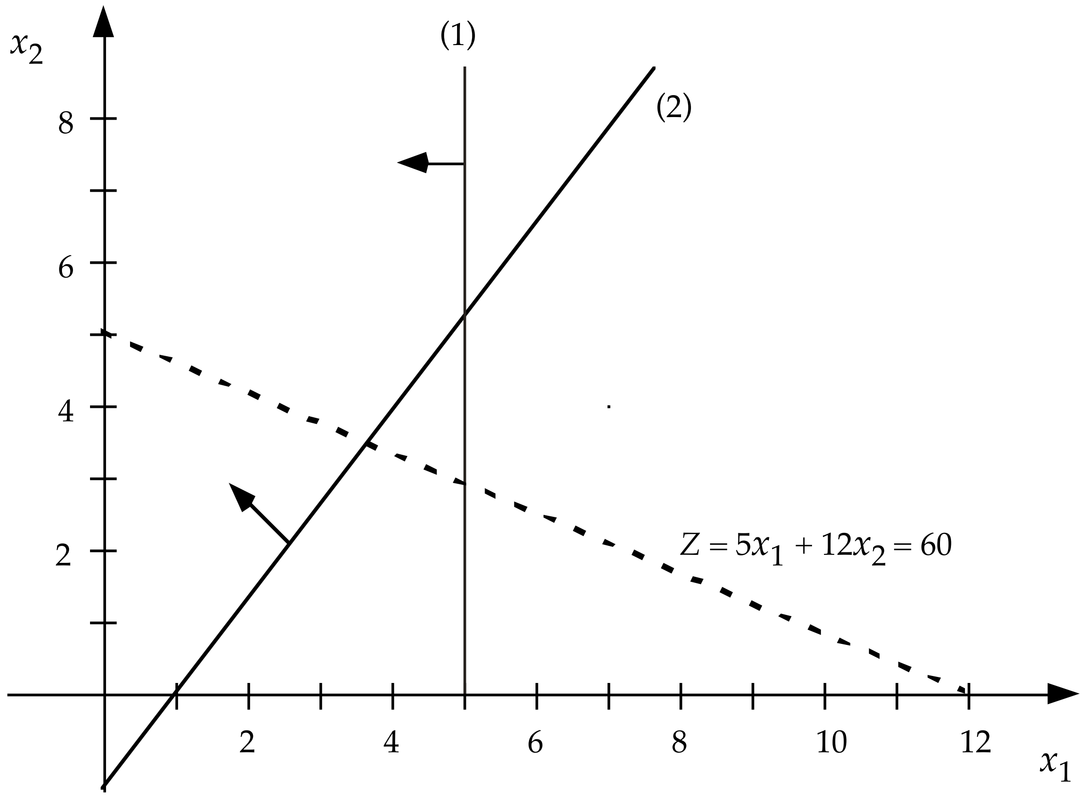
$$\text{Maximize } Z = 5x_1 + 12x_2$$

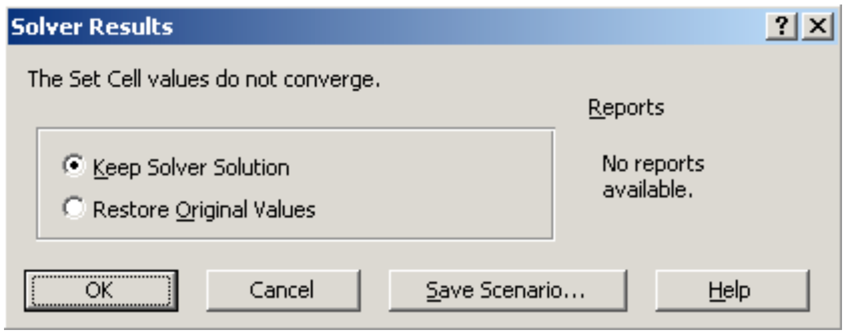
subject to

$$\begin{aligned} &(1) \quad x_1 &\le 5 \\ &(2) \quad 2x_1 - x_2 &\le 2 \end{aligned}$$

and

$$x_1 \ge 0, \ \ x_2 \ge 0.$$



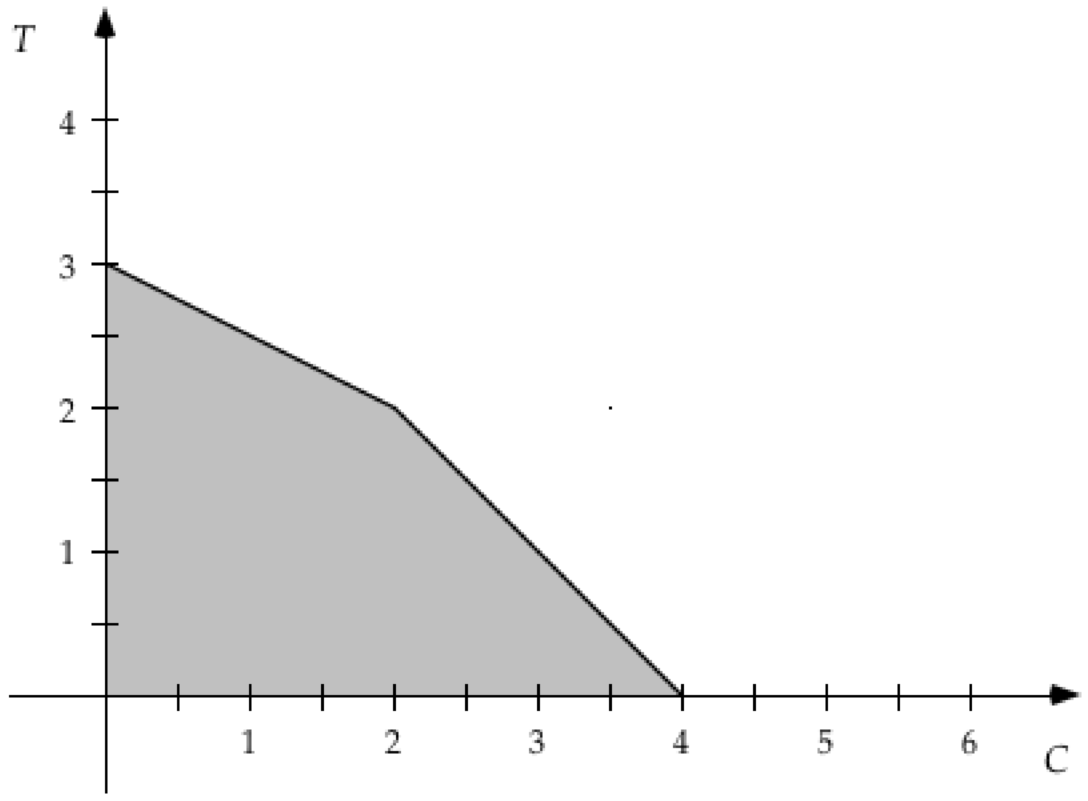


The Set Cell values do not converge.

[✓ Keep Solver Solution]
Restore Original Values

No reports available.

The Simplex Method



The simplex method algorithm:

- Start at a feasible corner point (often the origin).
- Check if adjacent corner points improve the objective function:
- If so, move to adjacent corner and repeat step 2.
- If not, current corner point is optimal. Stop.