Module 10: More LP Models - Part II

Reading Material: 10.1 – Introduction

This module illustrates additional common LP applications for situations that decision makers routinely face in practice. It represents another evolutionary step in our LP modeling mastery and could easily be combined with Module 8 and called Intermediate LP Applications.

The scenarios that follow are meant as in-class exercises; the students can struggle with the formulation of the problem, or even try to come up with a good intuitive manual solution to the problem. Then, the reader/student is gently led to a proper LP model by the end of the module. The “a-ha’ moment occurs, and then practice with practice problems!

There are two specific applications illustrated in this version of the module: staffing/rostering type LP applications, and the classic blending model.

Reading Material: 10.2 – Staffing Models

10.2.1 Introduction to Scheduling/Rostering LP Models

LP is often useful in practice when there is a need to develop a high-level planning schedule for a situation that has varying staffing requirements. For instance, we might want to determine the optimal number of people working on specifically defined shifts that satisfy minimum staffing requirements over a given time frame. Obviously, the assumption is that these minimum targets have been established based on some criteria like expected service quality or anticipated demand, among others. Examples might include staffing call centers (call volumes translated into personnel requirements, then shifts are predetermined and a model decides how to best meet those volumes), nurses and other medical applications (how many nurses of a certain skill level are required to work each shift to meet the projected needs of the hospital/facility), sales staffs in a retail operation, staffing checkout lines in supermarkets (how frustrating this scenario is at times!), and a variety of other circumstances in which an overall aggregate plan is required to try to best meet customer demands.

The mantra in these types of problem might be best stated as “Shifts are different from Days/Hours.” Staffing requirements are usually stated for a finite “time slice” that is shorter than the length of shifts. Thus, shifts ‘cross’ (serve, work) multiple time slices and time slices are covered by multiple shifts. The goal in these problems would be to find the optimal number of people on each shift that meets the required staffing levels. Reasonable criteria could include minimizing cost, minimizing the total number of workers required, and other reasonable factors.

Note that we are not assigning specific people to shifts, but determining an optimal aggregate number necessary per shift. Assignment models will be visited later in the book.

10.2.2 Aggregate Scheduling at St. Joseph's Thrift shop – Problem Description

The St. Joseph Thrift shop is open 5 days per week, Monday through Friday. The Thrift Shop workers are paid based on which days of the week they work. On Monday, $71 (per day), Tuesday $58.50, Wednesday $63.25, Thursday $79, and Friday $68.5.

Thrift shop workers work 3 days per week. Possible shifts must include either Monday or Friday, and then two of the other 3 days. So, one valid shift to consider is Monday-Tuesday-Wednesday. Monday-Wednesday-Friday is NOT a valid shift. Also, no more than five people can be assigned to any one shift. This is a store policy.

Based on the usual number of people who shop at the large thrift store, and the amount of incoming donated goods, it is desired to have at least 8 people work on Monday, 11 on Tuesday, 6 on Wednesday, 11 on Thursday, and 12 on Friday.

Determine the possible shifts to which workers can be assigned. Then, find the optimal number of workers for each shift that minimizes total cost while at least meeting the minimum required workers per day.

10.2.3 Model Setup St. Joseph – Decision Variables

The first step in our scheduling problem is to determine possible shifts. Very specific guidelines were provided about the shifts – 3 days per week, either Monday or Friday, and two of the three other weekdays. Each possible shift will represent a set of days covered. All possible combinations will be identified, and they will be the model decision variables.

Some shifts include Monday; some shifts include Friday; NO shifts include both. Regardless of whether a shift covers Monday or Friday, the shift will also have two midweek days. There are three possible combinations of splitting up the midweek days into unique 2-day pairs: Tuesday-Wednesday, Tuesday-Thursday, and Wednesday-Thursday.

Thus, with two possible end of week days and three possible combinations of midweek dyads, there will be a total of six possible shifts. They are fully enumerated below:

Monday-Tuesday-Wednesday

Monday-Tuesday-Thursday

Monday-Wednesday-Thursday

Friday-Tuesday-Wednesday

Friday-Tuesday-Thursday

Friday-Wednesday-Thursday

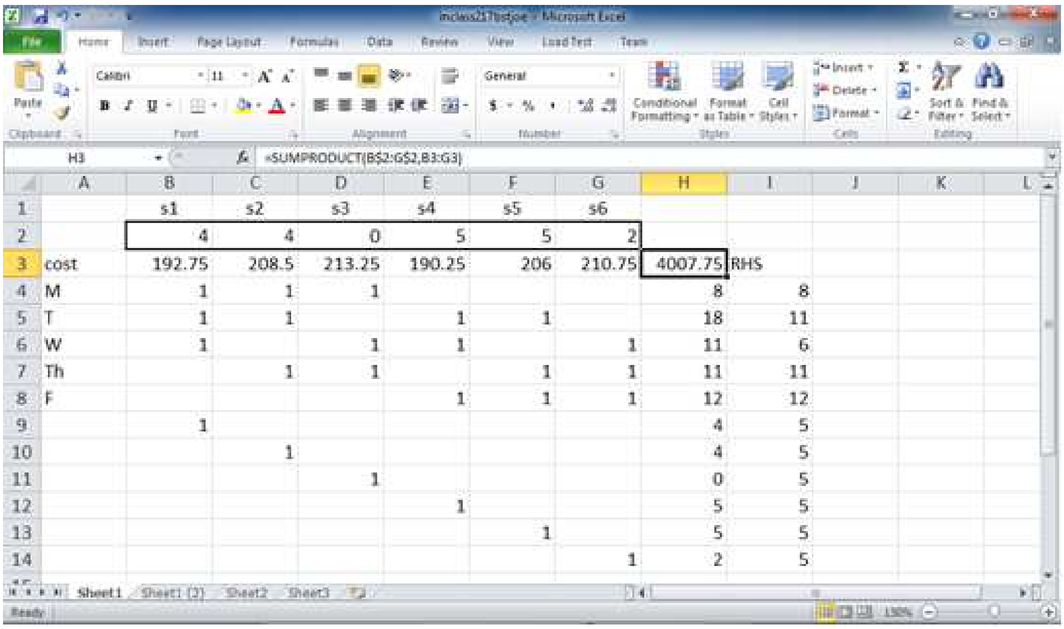
Each of these six shifts corresponds to a decision variable – representing the number of people who should work that shift in the optimal schedule. Call these shifts S1, S2, . . . S6 (S1 is the first combination of Monday-Tuesday-Wednesday, etc.).

10.2.4 Model Development – Objective Function – Minimize Cost

The problem description states that we are to minimize cost in staffing. Other objectives of interest could be possible, including circumstances in which there are multiple objectives. The text will revisit some of these issues later.

We have identified the six possible shifts available for consideration in the development of the optimal staffing plan. Each day has been assessed a unique cost; each shift will be the sum of the individual costs of the 3 days composing that shift. Therefore, the total cost of the first shift, S1, consisting of people who work Monday, Tuesday, and Wednesday, is \$71 + \$58.50 + \$63.25 = \$192.75 per person. Similarly, for shift 2, the cost can be calculated as \$71 + \$58.50 + \$79 = \$208.50.

Row 3 in Figure 10.1 shows the results of the cost calculations. Power users of EXCEL may desire to have EXCEL calculate the shift costs – that is certainly possible but not shown here.


H3 =SUMPRODUCT(B$2:G$2,B3:G3)

	A	B	C	D	E	F	G	H	I
1		s1	s2	s3	s4	s5	s6		
2		4	4	0	5	5	2		
3	cost	192.75	208.5	213.25	190.25	206	210.75	4007.75	RHS
4	M	1	1	1				8	8
5	T	1	1		1	1		18	11
6	W	1		1	1		1	11	6
7	Th		1	1		1	1	11	11
8	F				1	1	1	12	12
9		1						4	5
10			1					4	5
11				1				0	5
12					1			5	5
13						1		5	5
14							1	2	5

Figure 10.1

Figure 10.2.5 Model Development – Constraints

The core constraints for scheduling/rostering problems are the time slice minimum requirements building on the mantra “Shifts are different from days.” One must ensure that shifts are staffed to meet or exceed the daily minimum requirements. Thus, the core scheduling model has a constraint for each DAY.

First, to start the discussion, identify which shifts cover Monday. In other words, identify (in decision variables) the total number of people who are assigned to shifts that work on Mondays. People on the first three shifts, S1, S2, and S3, all work Monday. So, the sum of the number of people assigned to these three shifts must be greater than or equal to minimum staffing requirements for Monday (given as eight people). Algebraically,

S1 + S2 + S3 >= 8.

For Tuesday – There are four shifts that cover (or work) on Tuesday – S1, S2, S4, and S5. The minimum staffing requirement for Tuesday is 11, so algebraically,

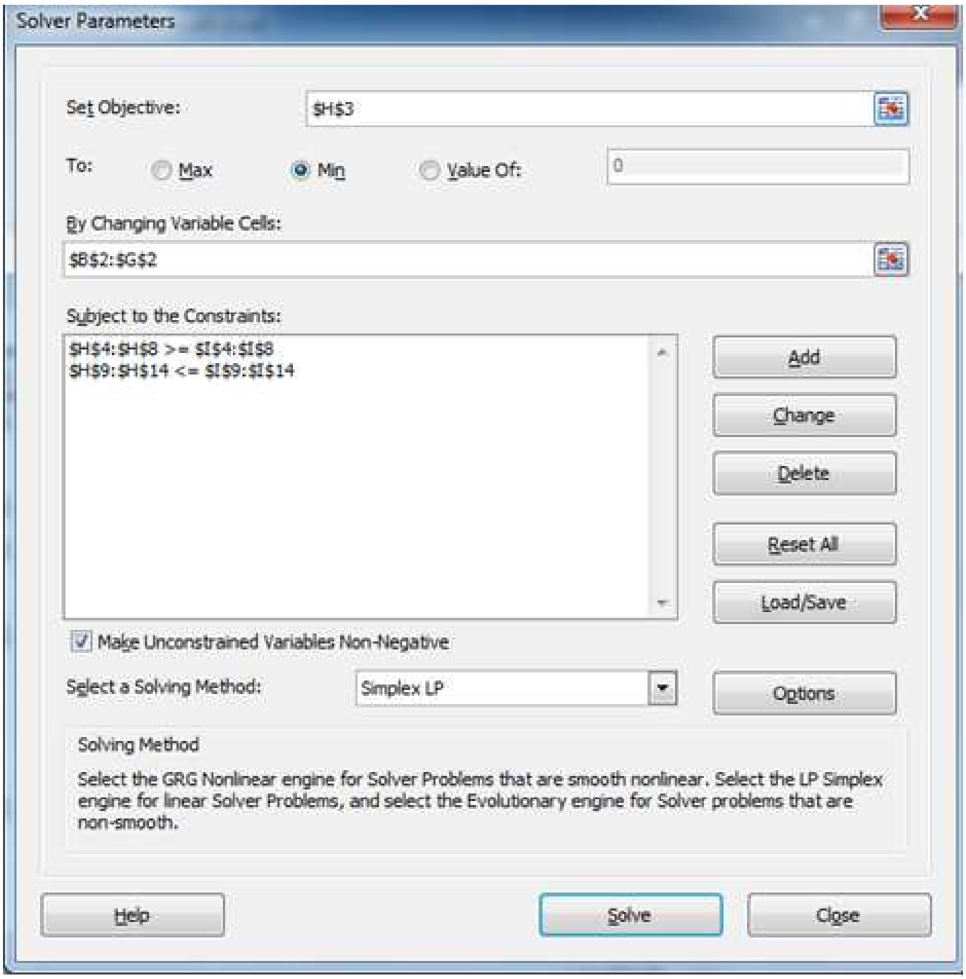
S1 + S2 + S4 + S5 >= 11.

I trust you can see the pattern developing. Each day (time slice) will have a constraint that forces minimum staffing requirements to be met. The constraints are greater than or equal to for a couple of reasons. First, it is normally infeasible to try to force all days (hours, etc.) requirements to be met exactly. Second, it really isn’t always a bad thing to exceed minimum requirements in practice to handle demand increases and similar unpredictable phenomena.

Note that after the algebraic constraints are entered in the spreadsheet (see Rows 4 through 8 in Figure 10.1 again), each column (shift) shows the pattern of the time increments covered by that shift. Thus, these constraints can be quickly entered in EXCEL once we note this algebraic insight.

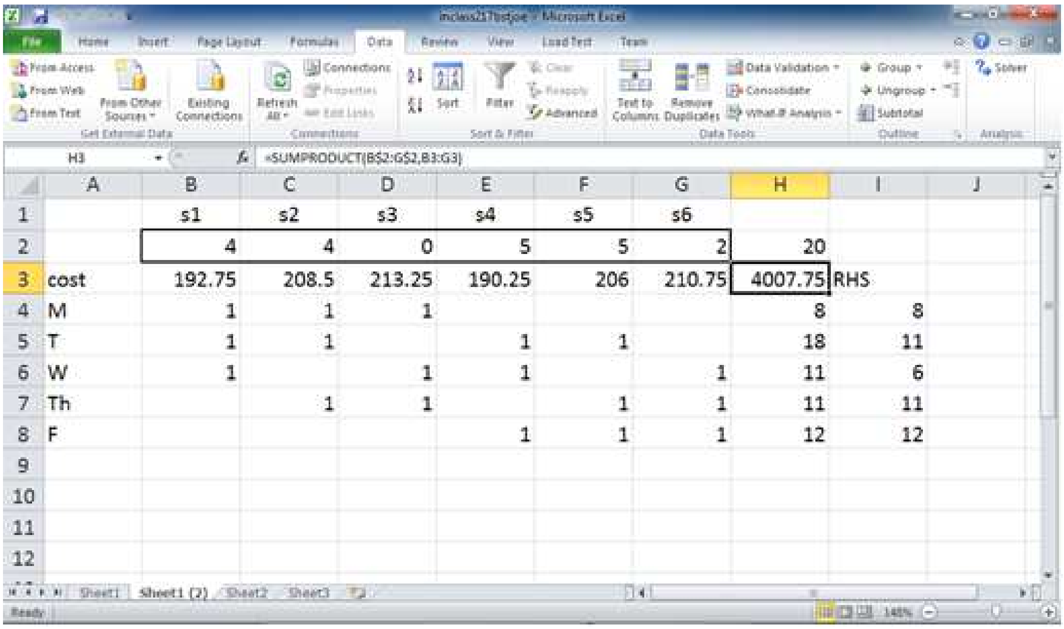
One additional requirement must also be implemented – the restriction that shifts can be no larger than five people. Again, keep in mind the difference between shift and day. This restriction means that the values in Row 3 (the decision variables) must be less than or equal to 5.

As with the Tulsa SodBusters Farm problem seen previously in the text, this constraint can be implemented in two different, yet equivalent ways. Figure 10.1 (with Solver in Figure 10.2) illustrates the six additional constraints (one for each shift) explicitly in the spreadsheet, and Figure 10.3 (along with the Solver implementation in Figure 10.4) shows the behind the scene implementation.



Solver Parameters
Set Objective:
$H$3
To:
Max
Min
Value Of:
0
By Changing Variable Cells:
$B$2:$G$2
Subject to the Constraints:
$H$4:$H$8 >= $I$4:$I$8
$H$9:$H$14 <= $I$9:$I$14
Add
Change
Delete
Reset All
Load/Save
Make Unconstrained Variables Non-Negative
Select a Solving Method:
Simplex LP
Options
Solving Method
Select the GRG Nonlinear engine for Solver Problems that are smooth nonlinear. Select the LP Simplex engine for linear Solver Problems, and select the Evolutionary engine for Solver problems that are non-smooth.
Help
Solve
Close

Figure 10.2



	A	B	C	D	E	F	G	H	I
1		s1	s2	s3	s4	s5	s6		
2		4	4	0	5	5	2	20	
3	cost	192.75	208.5	213.25	190.25	206	210.75	4007.75	RHS
4	M	1	1	1				8	8
5	T	1	1		1	1		18	11
6	W	1		1	1		1	11	6
7	Th		1	1		1	1	11	11
8	F				1	1	1	12	12

Figure 10.3

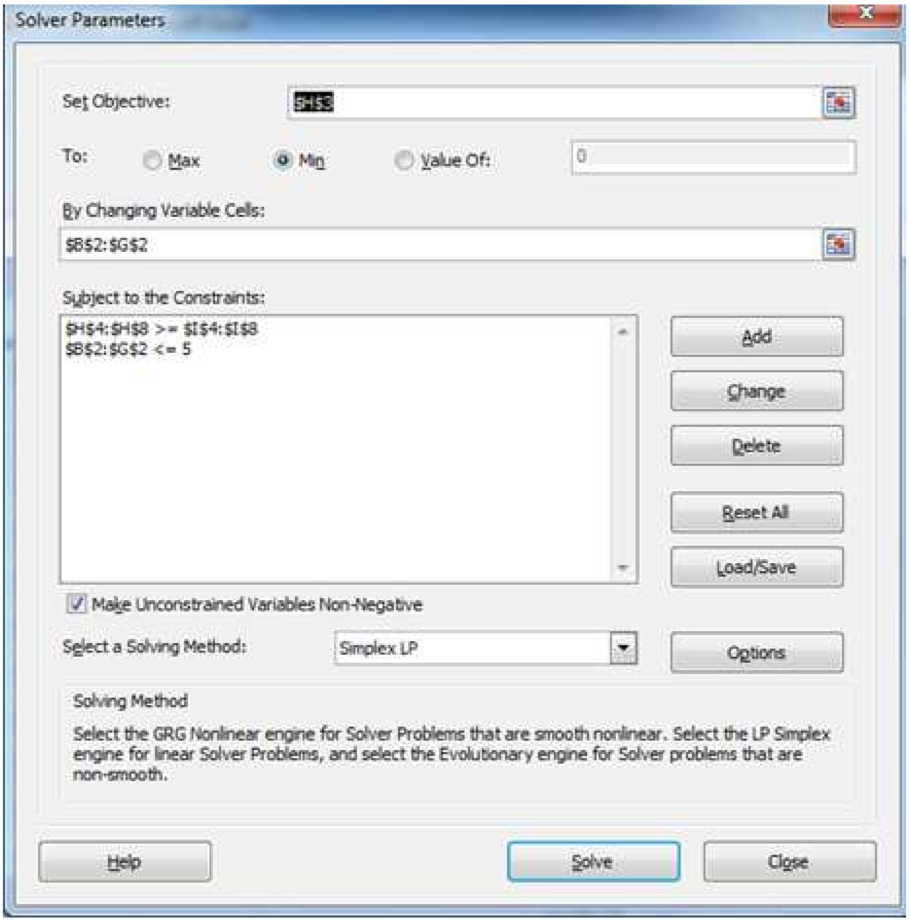


Figure 10.4

In both spreadsheets, Column H is the active formula of the model – the use of SUMPRODUCT (Row $2: Row x), where x is the current row. Column J is the right-hand side (RHS) of the constraints. As shown in Figures 10.2 and 10.4, the constraints are implemented by specifying the proper relationships between Columns H (left-hand side [LHS]) and J (RHS). H3 is the target cell, minimizing cost.

10.2.6 The Model Solution for St. Joseph's

The optimal solution (in both models, because they are equivalent) provides a least cost staffing solution of $4007.75 per week. The optimal schedule would have four people work shift 1, 4 people; shift 2, 5 people; both shift 4 and 5; and 2 people shift 6. There may be alternative solutions that provide the same least cost solution, but that is not investigated here.

Note the binding constraints: Monday, Thursday, and Friday's minimum staffing requirements are binding, as are the constraints limiting the size of shifts 4 and 5 to no more than five people. Thus, if more than five people were allowed per shift, one could reduce cost. Similarly, the personnel requirements for Monday, Thursday, and Friday also constrain the solution and would lead to a cost reduction if one of them could be loosened.

Note the seven extra people beyond the minimum requirements working on Tuesday, and the five extra people working on Wednesday. If these numbers of extra staff are unacceptable, additional constraints could be added that limit the difference between required staff and the actual number of people working under the optimal schedule (Column H less Column J). Module practice problems will explore this potential additional model restriction.

Also note cell H2 in Figure 10.4. It is an extra cell with formula SUM (B2:G2). This calculates the total number of employees used in the optimal solution. It could be a potential alternative objective for the problem. Although the criteria is similar to total costs, it likely would lead to different results. Again, future practice problems will allow you to explore different potential criterion.

This concludes our core scheduling model presentation. Next, another very useful type of model, termed a "blending" model, is examined.

Module 10: RM 10-3

Some of the first applications of LPin industry came during the 1950s when energy companies used LP to determine optimal ways of blending oil feed stocks into petroleum products. Oil feedstocks have different attributes (such as octane, etc.) and thus must be optimally blended together to meet final consumer product requirements. The next example will use something tastier than petroleum – a hypothetical neighborhood setting using a fundamental building block of the food pyramid – chocolate.

10.3.1 Annual Cedar Oaks Neighborhood Garage Sale: M&M Candy Blend

You have been asked to find the optimal way of mixing three different kinds of "pure" M&M's (blue, orange, and black) into four products that you will sell at your annual neighborhood garage sale.

We will be treating the costs of the M&M's as "relevant" costs – and each color has a unique per-unit cost (blue = $2 per unit, orange = $2.95 per unit, black = $2.5 per unit). Consider that you have the opportunity to purchase up to 20 units of blue, 15 units of orange, and 30 units of black M&M's – but no more. Use this as your upper bound of resources. (Using units so as not to get into a lb vs. oz vs. grams discussion.)

Traditionally, the four different M&M mixes (COWBOY, BRUISE, POTPOURRI, BRONCO) have different popularity and are sold at different prices (normalized to units to be comparable to the costs). Sales price for the products (per unit) are $4.00, $2.50, $1.75, and $3.50, respectively. Again, with plenty of demand history known (it's an old neighborhood), we need to mix together at least 25 units of COWBOY, at least 12 units of BRONCO (a lot of Colorado expats or Boise State fans live here?), and at least 5 units of the other two (for the sake of variety).

Each mix has flexibility in its composition, but with some guidelines. All mixes can use all three M&M's, even if there are no specific requirements stated below.

COWBOY – must consist of at least 40% orange, at least 40% black, and no more than 15% blue.

BRUISE – must consist of at least 40% blue and at least 40% black

POTPOURRI – must consist of at least 25% of each of the three inputs

BRONCO – must consist of no more than 60% orange, no more than 60% blue, and no more than 15% black

Find the optimal way of mixing the three raw materials into four products that maximize profit (sales less costs). Do not worry about whole numbers.

10.3.2 Model Setup – Preliminary Statements

An aside: Around Halloween a few years ago, M&M's did make a special bag with orange and black candies together. Given my present school colors (also orange and black), that was quite handy when putting together goody bags for high school programs and so forth. Alas, they quit making those special bags. I normally use yellow as orange in my in-class exercise, and typically have to substitute purple for blue. But the props are edible! (And I can claim my colorblindness makes the differences indistinguishable).

Before we start modeling, let us take a little time to distinguish what is and what is not a "blending" problem. Two past problems are worth revisiting – Brock's Boar-atorium and the Tulsa SodBusters Farm problem.

Brock's Boar-atorium blended together corn, oats, flax, and barley to make one product. We called it a diet problem and modeled it in traditional row/column format. It does have some attributes of blending because we merge together unspecified proportions of four grains to optimally make our ONE product. And that is key – there is just one product. In essence, we had a 1 × 4 problem – 1 product blended with 4 ingredients.

On the other hand, the Tulsa SodBusters Farm problem dealt with deciding how many units of two types of Fescue should be planted on two different farms – a 2 × 2 problem. We illustrated how a matrix approach for specifying our decision variables worked well (as an alternative to row/column format). So, the Tulsa SodBusters Farm problem helped us determine the optimal blend of Fescue to plant at each farm.

Returning to Brock, if our production situation was different and were blending multiple products from the 4 grains, say 3 products, each with different grain requirements (but still flexible requirements), then a blending formulation would be appropriate – it would then be a 3 × 4 problem (3 products made with variable requirements from 4 grains).

So, if we are making x products from y ingredients *and the ingredient proportions are variable*, matrix-style decision variables are a very efficient and effective LP modeling implementation strategy in EXCEL.

The statement *ingredient proportions are variable* is very important. If a product's composition must be an exact value (i.e., no flexibility), there is no need for decision variables (because there are no decisions to be made!).

To illustrate this point, we will temporarily modify the Cedar Oaks problem. Suppose the four products (COWBOY, BRUISE, POTPOURRI, and BRONCO) are defined with the following EXACT specifications:

COWBOY is made up of exactly 50% ORANGE, 40% BLACK, and 10% BLUE.

BRUISE is made up of exactly 45% BLACK, 40% BLUE, and 20% ORANGE.

POTPOURRI is made up of exactly 30% ORANGE, 35% BLACK, and 35% BLUE.

BRONCO is made up of exactly 45% ORANGE, 45% BLUE, and 10% BLACK.

In this situation, an appropriate model would include only four decision variables, one for each product. There is no variation in the composition of the mixes – we have “fixed” ingredient composition. Each ingredient would still be a constraint – but there is no need for a matrix-style approach to modeling. Only when there is flexibility in product composition do we include the blending variables.

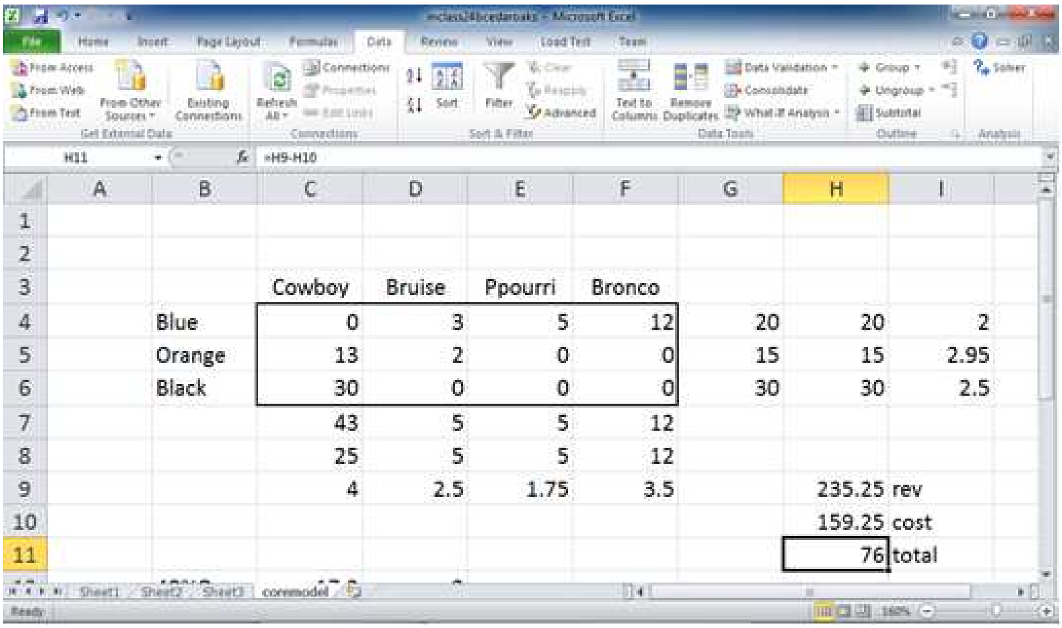
Over many years, I have seen this issue create some indigestion for my students. Thus, I am trying to “head off at the pass” potential trouble spots.

Now, let us return to our original broadcast (and the original Cedar Oaks problem!).

10.3.3 Model Setup – Decision Variables and Core Constraints

Cedar Oaks wishes to blend 4 products from 3 kinds of M&M’s – there is flexibility in the mix. Thus, a blending model is appropriate, as is a matrix style of decision variables. Four products, 3 raw materials – that is a 4 × 3 (or 3 × 4, either view is fine) problem. Thus, a total of 12 decision variables, representing the amount of (BLUE, ORANGE, BLACK) blended into (COWBOY, BRUISE, POTPOURRI, and BRONCO).

Figure 10.5 shows an EXCEL model of the basic core model. THIS IS NOT A COMPLETE REPRESENTATION OF THE PROBLEM STATEMENT ABOVE. I propose attacking our more complicated modules modularly – getting a core chunk working, then adding on additional layers of constraints once we are certain our core model is correct. This approach borrows from good traditional programming practice.



	A	B	C	D	E	F	G	H	I
1									
2									
3			Cowboy	Bruise	Ppourri	Bronco			
4		Blue	0	3	5	12	20	20	2
5		Orange	13	2	0	0	15	15	2.95
6		Black	30	0	0	0	30	30	2.5
7			43	5	5	12			
8			25	5	5	12			
9			4	2.5	1.75	3.5		235.25	rev
10								159.25	cost
11								76	total

Figure 10.5

C4:F6 are the changing cells/decision variables representing the amount of (BLUE, ORANGE, BLACK) blended into (COWBOY, BRUISE, POTPOURRI, and BRONCO).

We have opted for the columns to represent our products, and the rows to represent our raw materials. It is okay to have it reversed.

Row 7 represents column sums (=SUM(C4:C6), etc.). Thus, Row 7 represents the total amount of the products mixed. Column G represents row sums (=SUM(C4:F4), etc.). Thus, Column G represents the total amount of the raw materials blended together in making our products.

Row 8 is the minimum amount of the mixes we must produce (from the problem description). Thus, one family of constraints (our demand constraints) will have Row 7 >= Row 8.

The maximum amounts of the three types of M&M's available are stated in Column H (again, from problem specifications). Another core set of constraints will be the supply constraints, Column G <= Column H.

Note: For the next Module, keep in the back of your minds the terms *demand* and *supply*. This will be helpful.

Row 9 is the sales price per unit for the four products. Column I is the cost per unit for the three raw materials. We will calculate the overall net profit of the model by separately calculating the revenue, the costs, then finding their difference, which becomes the target cell (objective function cell).

Cell H9 calculates the revenue of a solution: =SUMPRODUCT(C7:F7,C9:F9).

Cell H10 calculates the cost of a solution: =SUMPRODUCT(G4:G6,I4:I6).

Cell H11 is equal to H9 – H10 and represents the net profit and is to be maximized.

This core model, missing all the composition constraints, was set up and solved in Figure 10.5 (The Solver dialog box is NOT shown, because the constraints are clearly identified above).

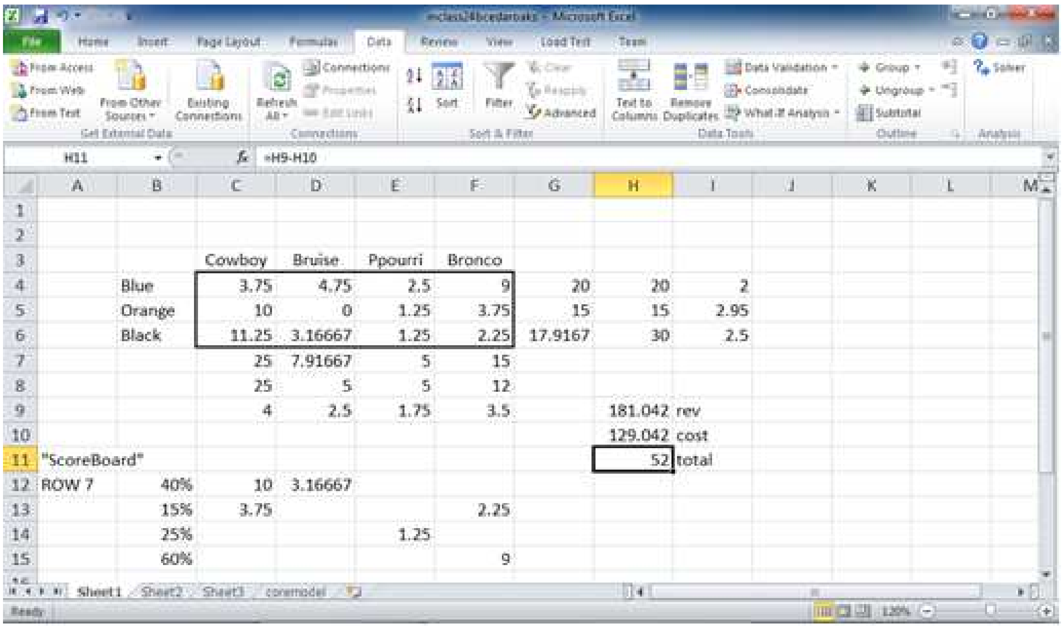
The total profit of this solution is $76, but of course it violates nearly all the composition constraints (e.g., COWBOY does not consist of at least 40% orange M&M's, BRONCO is 100% blue, etc.). But again, this "layer at a time" strategy makes creating good models much easier (IMHO) than trying to throw all model parameters in EXCEL at once. I suggest adopting this approach in practice.

The next section adds the composition constraints and completes the model.

10.3.4 Creating the Model "Scoreboard" – An Approach to Implementation

One could implement the composition constraints in various ways. Remember, there is not a rule that says we must use a matrix approach for this kind of problem! As mentioned before, the old-fashioned row/column form can always work – here, one could have 12 columns for decision variables, and construct the algebraic representation of the constraints in a traditional fashion. There is ABSOLUTELY NOTHING wrong with that approach here. But the matrix approach has an efficiency advantage.

The strategy for implementing the composition constraints will be to create a scoreboard of metrics below the decision variable matrix. The scoreboard is found in the general area of cells B12 through F15 in Figure 10.6.



H11 =H9-H10

	A	B	C	D	E	F	G	H	I
1									
2									
3			Cowboy	Bruise	Ppourri	Bronco			
4		Blue	3.75	4.75	2.5	9	20	20	2
5		Orange	10	0	1.25	3.75	15	15	2.95
6		Black	11.25	3.16667	1.25	2.25	17.9167	30	2.5
7			25	7.91667	5	15			
8			25	5	5	12			
9			4	2.5	1.75	3.5		181.042	rev
10								129.042	cost
11	"ScoreBoard"							52	total
12	ROW 7	40%	10	3.16667					
13		15%	3.75			2.25			
14		25%			1.25				
15		60%				9			

Figure 10.6

We will first examine the COWBOY mix, then move more quickly through the rest of the products.

There are three composition constraints for COWBOY: At least 40% ORANGE, at least 40% BLACK, and no more than 15% BLUE. Each requirement must be modeled independently.

Let's use the following notations for the algebraic work: C-OR (Orange in Cowboy Mix), C-BK (Black M&M's in Cowboy Mix) and C-BL (Blue M&M's in Cowboy Mix).

The total COWBOY mix made = C-BL + C-OR + C-BK (EXCEL cells C4, C5, and C6, added up in C7).

Our first constraint: 40% of the total mix must be ORANGE. Algebraically:

C-OR >= .4 * (C-BL + C-OR + C-BK)

We could rearrange terms and get it into row/column form – but we're not using row/column form. Instead, look at the EXCEL cells that represent the components:

C-OR = C5. (C-BL + C-OR + C-BK) = C7.

The constraint, using EXCEL cell references, simplifies to C5 >= .4*C7.

REMEMBER: Our philosophy to ensure success is to let EXCEL do calculations and the Solver solve. So we explicitly let EXCEL calculate our RHS targets.

In the scoreboard area of the spreadsheet, calculate .4*C7 (this is done in C12). Then add a constraint in the solver:

C5 >= C12.

Simply highlight Cell C5 in the LHS entry in the constraint dialogue box and C12 in the RHS.

One down, two to go (for COWBOY).

Second constraint: 40% of the total mix must be black.

This is very similar to the orange M&M requirement. Algebraically, we want:

C-BK >= .4 * (C-BL + C-OR + C-BK)

C-BK is Cell C6. The RHS is already calculated in the scoreboard (in C12). So, the second constraint can be added (without additional formulae in EXCEL) as:

C6 >= C12.

Third constraint: No more than 15% of the COWBOY mix can be blue

Cell C4 is C-BL.

The RHS of this constraint is calculated in cell C13. The formula in C13 is =.15* C7.

Upon entering this formula, add the final COWBOY constraint:

C4 <= C13.

To summarize, the proportions of interest are explicitly calculated in the scoreboard section of EXCEL. Their exact spreadsheet location doesn't matter – the proper constraints are created through pointing and clicking the appropriate LHS and RHS placeholders.

Column B in the scoreboard area is only a label to delineate the calculations in the scoreboard. There is no correct order, and to prove this point, I left the RHS values in the same disorder that I would have created them when first solving the model.

Now we can create the composition constraints for the other three products.

There are two constraints for BRUISE: BLUE and BLACK must both be (individually) at least 40% of the total mix. (Orange is not constrained for BRUISE.)

D12 = .4* D7 – this represents the target 40% value of BRUISE.

We add two constraints: D4 >= D12 and D6 >=D12. This forces BLUE (and BLACK) to be at least 40% of the overall mix.

POTPOURRI has three composition requirements. Each component must make up at least 25% of the total mix. We can do this fairly quickly. E14 is set to = 0.25*E7 – which is the 25% benchmark figure. Then highlight all three decision variable cells for POTPOURRI (E4:E6) as the LHS of the constraint, and set it >= a single cell in the RHS (E14). We modeled three constraints at once!

Finally, there are also three composition constraints for BRONCO.

Cell F15 = 0.6* F7 and Cell F13 = .15*F7 represent the target RHS's.

We add: F4:F5 <= F15 and F6 <= F13 to complete the model.

I purposely avoided much algebra in setting up the constraints. That is one of the efficiencies of matrix decision variables.

10.3.5 – Final Comments and Solution

From Figure 10.6, you can see the optimal solution provides $52 of profit, a reduction from $76 from the previous core model that ignored the composition requirements. All desired proportions are met. Interestingly, not all raw materials were used in making the optimal M&M's mixes. There were leftover black M&M's, even though all of the more expensive orange were used. The composition constraints, the different profitability and the demand requirements all contribute to the decision of the model that it is best to not use all available M&M's to maximize profit.

Reading Material: 10.4 – Conclusion

Two useful applications of LP were illustrated in this module. Future editions of the text may add other useful generic models here – one such type of LP model would be scenarios in which one is managing production or resources over multiple periods of time.

The goal of this module is to expand our ability at representing reality through our "algebra on steroids." We continue that in the next module, as we will use skills already mastered to formulate models to assist us in the supply chain and distribution areas.