15.053 February 27, 2007

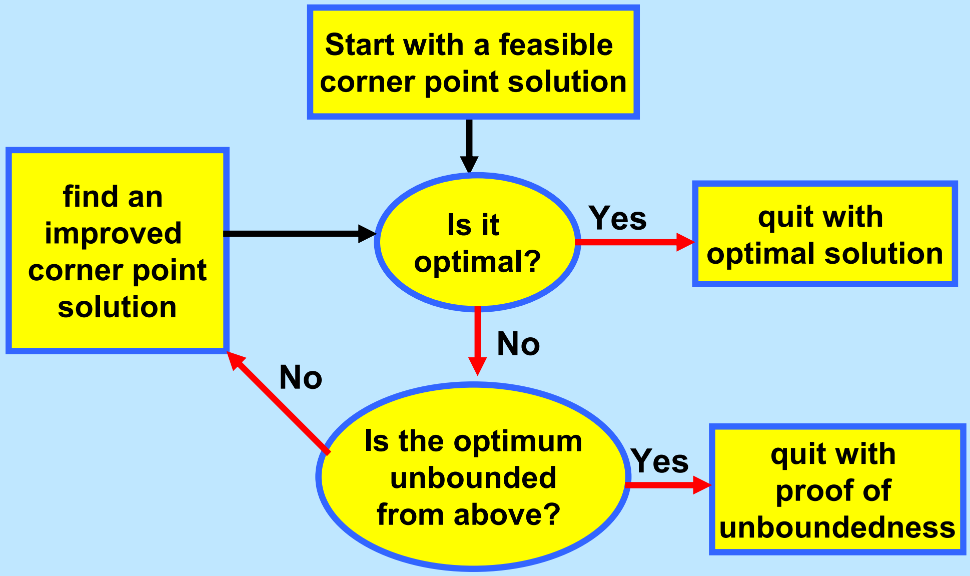
- Simplex Method Continued

Today's Lecture

- Review of the simplex algorithm.
- Formalizing the approach
- Alternative Optimal Solutions
- Obtaining an initial bfs
- Is the simplex algorithm finite?

Quote of the day:
Everyone designs who devises courses of action aimed at changing existing situations into preferred ones.
-- Herbert Simon

The simplex algorithm
(for max problems)



LP Canonical Form

z	x_1	x_2	x_3	x_4	x_5		
1	1	0	0	2	0	=	3
0	2	0	1	3	0	=	9
0	1	1	0	-1	0	=	1
0	0	0	0	2	1	=	4

The basic variables are z, x_2, x_3 and x_5.

The non-basic variables are x_1 and x_4.

The basic feasible solution (bfs) is

$z = 3, x_1 = 0, x_2 = 1, x_3 = 9, x_4 = 0$, and $x_5 = 4$.

The Simplex Pivot Rule and Optimality Conditions

z	x_1	x_2	x_3	x_4	x_5		
1	1	0	0	2	0	=	3
0	2	0	1	3	0	=	9
0	1	1	0	-1	0	=	1
0	0	0	0	2	1	=	4

optimal

Pivot in a variable whose cost-row coefficient is negative.

If all z-row coefficients are non-negative then the current basic feasible solution is optimal.

Optimality Conditions

z	x_1	x_2	x_3	x_4	x_5		RHS
1	$-\bar{c}_1$	0	0	$-\bar{c}_4$	0	=	3
0	2	0	1	3	0	=	9
0	1	1	0	-1	0	=	1
0	0	0	0	2	1	=	4

The bar indicates that it is possibly the coefficient after some pivots

***Opt. conditions*: The bfs is optimal if $-\bar{c}_j \geq 0$ for all j.**

***Maximize** $z = \bar{c}_1 + \bar{c}_4$*

Writing the objective function coefficients with a negative looks awkward. But there is a good reason for it. We usually think of this objective function as maximize $z = 1x_1 + 2x_4$. In this case, we let $c_1 = 1$ and $c_4 = 2$. But in the tableau, we have the coefficients $-c_1$ and $-c_4$.

Remember that a bar over the letter indicates that it may be after a pivot or more. c_1 and c_4 without a bar denote the original data.

Determining the exiting variable

z	x_1	x_2	x_3	x_4	x_5		RHS
1	1	0	0	-2	0	=	3
0	2	0	1	3	0	=	9
0	1	1	0	-1	0	=	1
0	0	0	0	2	1	=	4

$x_3 = 9 - 3\Delta$ $\quad \Delta \le 9/3$

$x_2 = 1 + \Delta$

$x_5 = 4 - 2\Delta$ $\quad \Delta \le 4/2$

1. **Choose an entering variable with negative z-row coeff.**
2. **Set the value of the entering variable to Δ, and make Δ, as large as possible while maintaining feasibility.**

By increasing the size of Δ, we are moving along an edge of the feasible region. In this case, we can let Δ be as large as 4/2 before a variable becomes negative. Setting $\Delta = 2$ results in another corner point solution.

The min ratio rule

z	x_1	x_2	x_3	x_4	x_5		RHS
1	1	0	0	-2	0	=	3
0	2	0	1	3	0	=	9
0	1	1	0	-1	0	=	1
0	0	0	0	2	1	=	4

$x_3 = 9 - 3\Delta$ $\quad \Delta \le 9/3$

$x_2 = 1 + \Delta$

$x_5 = 4 - 2\Delta$ $\quad \Delta \le 4/2$

1. **To determine the exiting variable, take the min ratio of the RHS coefficient divided by the coefficient of the entering variable, among all those whose coefficient is strictly positive.**

8

In order to carry out the simplex algorithm, we don't really need the algorithm to work with Δ. We can shortcut the procedure by directly identifying the next component of the tableau on which to pivot.

The value of Δ was obtained by taking the ratio of components of the RHS with the corresponding components of the column for x_4, the entering variable. Δ is the min of these ratios, as restricted to ratios that are non-negative. We will next make use of how Δ was computed as the minimum of ratios.

The min ratio rule

Entering variable is x_s with negative $-\bar{c}_s < 0$

z	x_1	x_2	x_3	x_s	x_5		
1	1	0	0	$-\bar{c}_s$	0	=	z_0
0	2	0	1	$\bar{a}_{1s}$	0	=	$\bar{b}_1$
0	1	1	0	$\bar{a}_{2s}$	0	=	$\bar{b}_2$
0	0	0	0	$\bar{a}_{3s}$	1	=	$\bar{b}_3$

Min Ratio = min{ $\bar{b}_i$ / $\bar{a}_{is}$: $\bar{a}_{is}$ > 0 and i = 1 to m }.

Usually, the "argmin" value is denoted as r.

In this case, r = 3. Pivot on $\bar{a}_{rs}$

9

This slide is the same as the previous slide except that we introduce new notation. The RHS coefficients are denoted as b's. The column coefficients (constraint matrix coefficients) are denoted as a's. The entering variable is always denoted as x_s (I don't know why, but it is convenient to have a common notation).

The bars over the letters indicate that it may not be the original data, but could have been obtained after one or more pivots.

The ratios are restricted to cases in which $a_{is} > 0$.

The argmin means that we are interested in the index r that gives the min ratio. The min ratio itself will give the value of x_s after the pivot. But the index r (the argmin) will tell us that we need to pivot on a_{rs} next. This is more useful information for the algorithm.

Dantzig himself introduced the argmin term into mathematics, and was proud of this achievement. It's really hard to find a concept in math that is fundamental but that does not already have a name.

Minimum Ratio Rule

Pivot out the basic variable in row r, where

$r = \text{argmin}_i \{ \bar{b}_i / \bar{a}_{is} : \bar{a}_{is} > 0\}$, and thus

$\bar{b}_r / \bar{a}_{rs} = \min \{ \bar{b}_i / \bar{a}_{is} : \bar{a}_{is} > 0\}$.

If $\bar{a}_{is} \leq 0$ for all i, then the solution is unbounded.

The pivot

z	x_1	x_2	x_3	x_4	x_5		RHS
1	1	0	0	0	1	=	7
0	2	0	1	0	-3/2	=	3
0	1	1	0	0	1/2	=	3
0	0	0	0	1	1/2	=	2

1. The objective value strictly improves so long as $\bar{b}_r > 0$.
2. The new (basic feasible solution) bfs is feasible.

If we pivot in a variable with negative cost coefficient in the z-row, the next bfs will always have a strictly larger RHS of the z-row so long as the RHS coefficients of the constraints are strictly positive. (Look carefully at a pivot and you will see why.) But if the RHS is zero for some constraint, it is possible that the RHS coefficient of the z-row will not change. In this case, the pivot will lead to a solution with the same objective function.

If you look at this situation more closely, what is actually happening is that the one of the basic variables had value 0, and the new basic variable will also have value 0, and the solution does not change at all. What changes is the tableau and the basic variables, but not the solution. You can verify this by changing the RHS of constraint 3 from a 4 to a 0 prior to the pivot. After the pivot, the new bfs will be the same solution as the bfs prior to the pivot, even though the tableau changes.

Alternative Optima (maximization)

Recall: $z + 4\,x_2 = 8$

Note that z does not depend on x_1 .

z	x_1	x_2	x_3	x_4		
1	0	4	0	0	=	8
0	1	3	1	0	=	6
0	-1	2	0	1	=	2

This basic feasible solution is optimal! There is an alternative optimum because the non-basic variable x_1 has a 0 cost-coefficient.

If x_1 enters the basis, what variable leaves?

12

If a non-basic variable (say x_1) has a coefficient of 0, it means that the objective value will not change if we increase x_1 by a little bit, and adjust the basic variables to maintain feasibility. We recognize when there are multiple optimum solutions by looking for zeros for non-basic variables in the z-row.

Alternative Optima (maximization)

z	x_1	x_2	x_3	x_4		
1	$-\bar{c}_1$	$-\bar{c}_2$	0	0	=	8
0	1	3	1	0	=	6
0	-1	2	0	1	=	2

There may be alternative optima if $-\bar{c}_j \geq 0$ for all j and $\bar{c}_j = 0$ for some j where x_j is non-basic

Use the min ratio rule to determine which variable leaves the basis.

If we increase x_1, we are moving along an edge of the feasible region. To find the next corner point, we would pivot the same as in the usual simplex algorithm.

Alternative Optima (maximization)

z	x_1	x_2	x_3	x_4		RHS
1	0	4	0	0	=	8
0	1	3	1	0	=	6
0	0	5	1	1	=	8

Perform the pivot.

Note: the solution is different, but the objective value is the same.

This is different from the case in which the RHS is 0. We actually find a different corner point solution, but it has the same objective value. Recall in the two dimensional examples how it is possible to have two corner points optimal, and all the points on the line segment joining these two points are also optimal.

Review of notation

A basic feasible solution is optimal if $-\bar{c}_j \geq 0$ for all j.

Assumption: the entering variable is x_s (and so $-\bar{c}_s < 0$)

Pivot out the basic variable in row r, where

$$r = \text{argmin}_i \ \{ \bar{b}_i / \bar{a}_{is} : \bar{a}_{is} > 0\}, \text{ and thus}$$

$$\bar{b}_r / \bar{a}_{rs} = \min \{ \bar{b}_i / \bar{a}_{is} : \bar{a}_{is} > 0\}.$$

If $\bar{a}_{is} \leq 0$ for all i, then the solution is unbounded.

15

LP notation seems to be unnecessary for much of what we do in 15.053. But it is critical if one models a very large LP, say with 1000 constraints. It turns out that one can describe the LP very efficiently by using abstract notation. But if one had to type out the 1000 constraints, it would be an enormous amount of work.

The notation is also very useful for describing the algorithm. Try rigorously describing the min ratio test if you don't have any notation. I'll bet that it would be very difficult for a friend who has not had 15.053 to figure out exactly what you mean.

Simplex Method (Max Form)

Step 0. The problem is in canonical form and $\bar{b} \geq 0$.

Step 1. If $-\bar{c} \geq 0$ then stop. The solution is optimal. If we continue, then there exists some $-\bar{c}_j < 0$.

Step 2. Choose any non-basic variable to pivot in with $-\bar{c}_s < 0$, e.g., $-\bar{c}_s = \min_j \{ -\bar{c}_j \mid -\bar{c}_j < 0 \}$. If $\bar{a}_{is} \leq 0$ for all i, then stop; the LP is unbounded. If we continue, then there exists some $\bar{a}_{is} > 0$.

Step 3. Pivot out the basic variable in row r, where r is chosen by the min ratio rule, that is $r = \text{argmin}_i (\bar{b}_i / \bar{a}_{is} : \bar{a}_{is} > 0)$.

Step 4. Replace the basic variable in row r with variable x_s and re-establish canonical form (i.e., pivot on the coefficient $\bar{a}_{rs}$.)

Step 5. Go to Step 1.

Time for a mental break

Even smart people get it wrong occasionally.

> **Even considering the improvements possible... the gas turbine could hardly be considered a feasible application to airplanes because of the difficulties of complying with the stringent weight requirements.**
> **-- US National Academy Of Science, 1940**

> **People have been talking about a 3,000 mile high-angle rocket shot from one continent to another, carrying an atomic bomb and so directed as to be a precise weapon... I think we can leave that out of our thinking.**
> **-- Dr. Vannevar Bush, 1945**

I love these quotes. And I'm sure that I've said things just as wrong (but hopefully not frequently).

Fooling around with alternating current is a waste of time. Nobody will use it, ever.
-- Thomas Edison

There is not the slightest indication that nuclear energy will be obtainable.
-- Albert Einstein 1932

Rail travel at high speed is not possible because passengers, unable to breathe, would die of asphyxia.
-- Dr. Dionysus Lardner, 1793-1859

Inventions have long since reached their limit, and I see no hope for future improvements.
-- Julius Frontenus, 10 A... D

The Phase 1 method to obtain an initial bfs

We need an initial bfs.

Approach: create a new LP that is related to the original LP and has the following features:

1. **The solution x = 0 is feasible for the new LP**
2. **An optimal solution for the new LP is feasible for the original problem.**

19

We know how to optimize an LP starting with a corner point solution. Amazingly one can, in general, find a corner point solution for an LP by optimizing a related linear program, one for which a corner point solution is readily obtained. It's sort of like lifting yourself by your own bootstraps, except that we cannot lift ourselves up by our own bootstraps. (Or maybe it's not like that.)

Phase 1: Creating a new LP

The Original LP

Maximize $z = 2x_1 + 3x_2 + x_3$

subject to

$$x_1 + x_2 + x_3 = 2$$
$$x_1 + 2x_2 + 3x_3 = 4$$
$$x_1, x_2, x_3 \geq 0$$

Remark: $x_1 = 1$, $x_2 = 0$, $x_3 = 1$ is a feasible corner point, but let's pretend we don't know that.

Step 1. Ignore the objective

subject to

$$x_1 + x_2 + x_3 = 2$$
$$x_1 + 2x_2 + 3x_3 = 4$$
$$x_1, x_2, x_3 \geq 0$$

20

Our goal is to obtain a corner point solution for the original LP. We don't need the objective function in order to identify some corner point, and so we delete it. We'll bring it back later when we need it again.

Phase 1: Creating a new LP

$$x_1 + x_2 + x_3 = 2$$
$$x_1 + 2x_2 + 3x_3 = 4$$
$$x_1, x_2, x_3 \geq 0$$

Step 2. "Relax" the constraints so that a corner point solution is really easy to obtain for the "relaxed" problem. (A relaxed problem is less constrained.)

$$x_1 + x_2 + x_3 \leq 2$$
$$x_1 + 2x_2 + 3x_3 \leq 4$$
$$x_1, x_2, x_3 \geq 0$$

x = 0 is a corner point solution.

21

When we relax the inequality constraints, we obtain a different feasible region, one that is much larger than feasible region of our original LP. Most importantly, the point x = 0 is a corner point solution of this relaxed problem, even though it is not feasible for the original problem.

Phase 1: Creating a new LP

$$\begin{aligned} x_1 + x_2 + x_3 &\le 2 \\ x_1 + 2x_2 + 3x_3 &\le 4 \\ x_1, x_2, x_3 &\ge 0 \end{aligned}$$

Step 3. Add in slacks variables, which we will call *artificial variables* since they are not part of the original problem. Create an objective so that an optimal solution x*,y* for the new LP will be feasible for the original problem (assuming a feasible solution exists).

minimize $w = y_1 + y_2$

$$\begin{aligned} x_1 + x_2 + x_3 + y_1 \quad &= 2 \\ x_1 + 2x_2 + 3x_3 + \quad y_2 &= 4 \\ y_1, y_2 &\ge 0 \end{aligned}$$

22

The corner point x = 0 is useful because it is the starting corner point for a simplex algorithm. But we must choose an objective for this relaxed problem so that the following is true: an optimal corner point solution for the relaxed problem will be a feasible corner point solution for the original problem, assuming that such a solution exists.

The objective is easier to express if we add in slack variables. So, we next add in slack variables, which are called "artificial variables" for the Phase 1 problem. This is to emphasize that they were not part of the original problem. We then create an objective of minimizing the sum of the artificial variables. What we really want is a solution with objective = 0.

$$
\begin{aligned}
x_1 + x_2 + x_3 &= 2 \\
x_1 + 2x_2 + 3x_3 &= 4 \\
x_1, x_2, x_3 &\geq 0
\end{aligned}
$$

Original constraints.

$x_1 = 1, x_2 = 0, x_3 = 1$ is a feasible solution.

minimize $w = y_1 + y_2$

$$
\begin{aligned}
x_1 + x_2 + x_3 + y_1 &= 2 \\
x_1 + 2x_2 + 3x_3 + y_2 &= 4 \\
y_1, y_2 &\geq 0
\end{aligned}
$$

Phase 1 linear program.

$x_1 = 1, x_2 = 0, x_3 = 1, y_1 = 0, y_2 = 0$ is an optimal solution.

Theorem. A feasible solution for the original problem will induce an optimal solution with w = 0 for the Phase 1 LP.

Theorem. An optimal solution for the Phase 1 LP with w = 0, will induce a feasible solution for the original problem.

If you have questions, perhaps Tim the Turkey and Cleaver the Beaver will be able to help out on the next slides.

How do we know that the Phase 1 LP will have a feasible and optimal solution?

The Phase 1 LP always has a feasible corner point by setting x to 0 and letting the "artificial variables" be basic? And we know that the objective function is at least 0. So, there must be a minimum cost solution.

Tim

Cleaver

How do we know that an optimal solution for the Phase 1 problem will give a feasible solution?

If a feasible solution x* exists for the original problem, then the solution x*,y* with y* = 0 is feasible for the Phase 1 LP and has objective 0. This has to be optimal, and the simplex method will find it.

But what if there is no feasible solution for the original problem?

In this case, the Phase 1 LP will have an optimal solution in which $y \neq 0$, and the optimum objective is greater than 0. This will prove that there was no feasible solution to the original problem.

So, to get a feasible corner point solution, all we need to do is to solve the Phase 1 LP?

That's right.

Except that we need to carry out a little more algebra to make it all work. You'll see why in a couple of slides.

Phase 1: Tableau Version

Original Constraints

x_1	x_2	x_3		
1	1	1	=	2
1	2	3	=	4

Phase 1 LP constraints

x_1	x_2	x_3	y_1	y_2		
1	1	1	1	0	=	2
1	2	3	0	1	=	4

x_1	x_2	x_3	y_1	y_2		
1	1	1	1	0	=	2
1	2	3	0	1	=	4

Phase 1 LP constraints

w	x_1	x_2	x_3	y_1	y_2		
1	0	0	0	1	1	=	0
0	1	1	1	1	0	=	2
0	1	2	3	0	1	=	4

Phase 1 LP

$w + y_1 + y_2 = 0.$ $w = -y_1 - y_2$ **maximize w**

29

You will notice that the tableau at the bottom is not in canonical form. The basic variables are y_1 and y_2, but their coefficients are not 0 in the z-row. (Try saying this last phrase fast three times.) This is easily remedied as we shall see on the next slide.

Get into canonical form with basic variables y_1, y_2.

w	x_1	x_2	x_3	y_1	y_2		
1	-2	-3	-4	0	0	=	-6
0	1	1	1	1	0	=	2
0	1	2	3	0	1	=	4

Subtract constraint 1 from the z-row

Subtract constraint 2 from the z-row

We get rid of the positive coefficient for y_1 in the z-row by subtracting constraint 1 from the z-row. We get rid of the positive coefficient for y_2 in the z-row by subtracting two times constraint 2 from the z-row. In general, we can get rid of the non-zero coefficients for y in the z-row by adding or subtracting multiples of the constraints one at a time.

Solve the Phase 1 Problem using the Simplex Algorithm

w	x_1	x_2	x_3	y_1	y_2		
1	0	-1	-2	2	0	=	-2
0	1	1	1	1	0	=	2
0	0	1	2	-1	1	=	2

Let variable x_1 enter.
x_2 or x_3 would have been able to enter as well.

Variable y_1 leaves the basis. Pivot on the "1".

Now we can run the simplex method on the Phase 1 problem.

One more pivot

Optimal !

w	x_1	x_2	x_3	y_1	y_2		
1	0	0	0	1	1	=	0
0	1	.5	0	1.5	-.5	=	1
0	0	.5	1	-.5	.5	=	1

Let variable x_3 enter.

x_2 would have been able to enter as well.

Variable y_2 leaves the basis. Pivot on the "2".

We solved the Phase 1 problem in 2 pivots. It may be tempting to believe that the Phase 1 problem is much easier to solve than the original problem. It turns out that this is not the case. For many linear programs, it takes a huge amount of time to find a feasible corner point using the Phase 1 approach. The number of pivots to solve the Phase 1 problem can be enormously large, possibly as large as 2^n.

Now use the bfs to start off the simplex algorithm for the original problem

x_1	x_2	x_3		
1	.5	0	=	1
0	.5	1	=	1

bfs for the original problem, but without the objective.

z	x_1	x_2	x_3		
1	-2	-3	-1		0
0	1	.5	0	=	1
0	0	.5	1	=	1

Including the objective.

Again, we have a basic feasible solution, the tableau is not in canonical form. Again, it is because the z-row has non-zero coefficients for the basic variables. And again, this is easily fixed by adding multiples of the constraints to the z-row, as we shall see on the next slide.

Get back into canonical form. Solving the original LP is called Phase 2 of the simplex algorithm

w	x_1	x_2	x_3		
1	0	-1.5	0		3
0	1	.5	0	=	1
0	0	.5	1	=	1

Add two times constraint 1 to the z-row.

Add constraint 2 to the z-row.

After performing these iterations, we are in canonical form, and are ready to carry out the simplex algorithm.

Then Optimize

w	x_1	x_2	x_3			Optimal !
1	3	0	0		6	
0	2	1	0	=	2	
0	-1	0	1	=	0	

Variable x_2 enters the basis.

Pivot on the .5

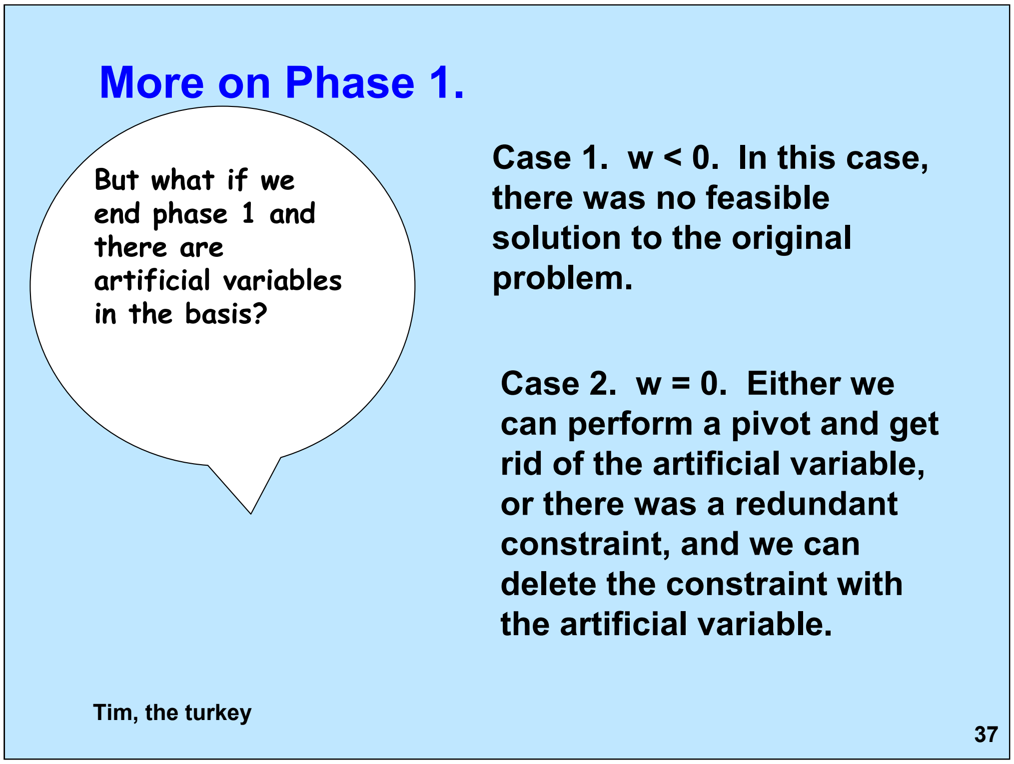
In this case, one pivot was enough. The basic solution is: $x_1 = 0$, $x_2 = 2$, $x_3 = 0$. Notice that the basic variable x_3 has a value of 0. This is referred to as "degeneracy". But we don't have to worry about degeneracy here because the basic feasible solution satisfies the optimality conditions, and so we can stop.

There is an entire tutorial on degeneracy, which is required reading for this subject. Degeneracy adds lots of technical challenges to the simplex algorithm. Put another way, if you understand how the simplex algorithm behaves when there is degeneracy, you really understand the simplex algorithm very well.

Phase 1 method: a summary

1. Write the constraints of the original LP
2. Create a relaxed LP by changing equality constraints to "≤ constraints"
3. Add slack variables (called ***artificial variables*** since they are not part of the original problem), and minimize the sum of the artificial variables.
4. Put the tableau in canonical form
5. Solve the Phase 1 Problem
6. Find the bfs, and then put the tableau in canonical form
7. Solve the original LP starting with this feasible bfs.

Other books treat Phase 1 a little differently, but what was presented here is equivalent to the Phase 1 method described elsewhere.

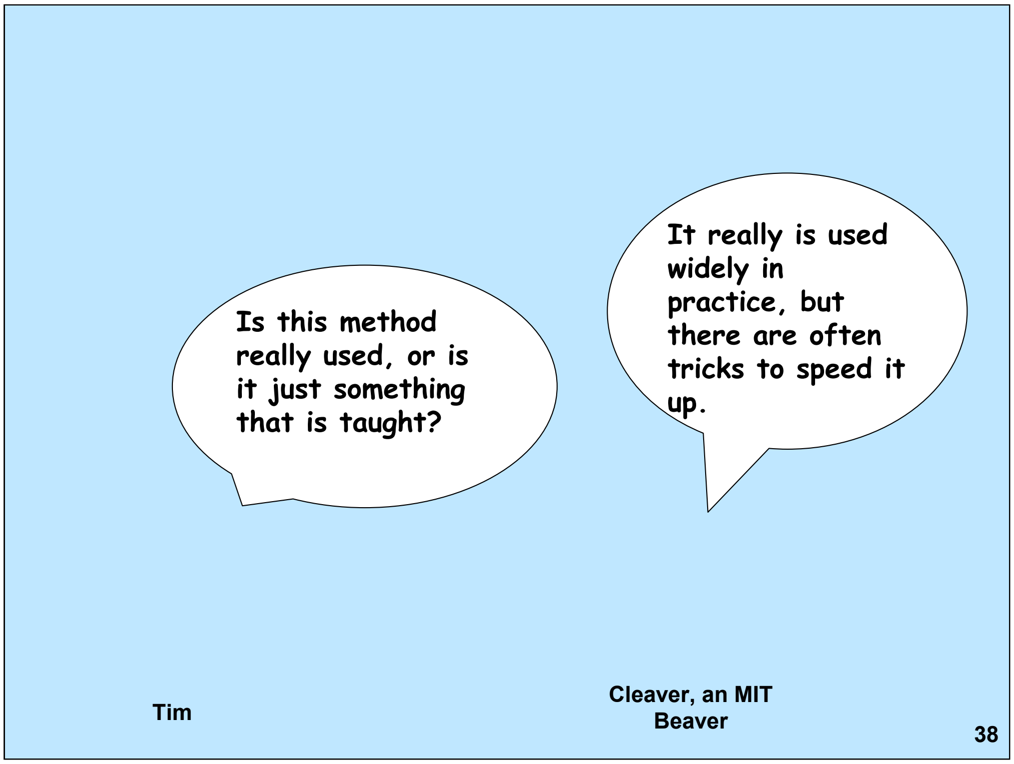


I know, you probably thought that this was the end of the story. It is for practical purposes. But I want to at least give some remaining details to fill in the last steps.

If the optimum objective value is less than 0 (recall we are maximizing the negative of the sum of the artificial variables), then there is no feasible solution to the original problem.

A technical challenge comes one of the basic variables for the optimum solution is y_j for some j and $y_j = 0$. Recall that we wanted a basic feasible solution of the original problem. But this means that we don't want the basis to contain any artificial variables.

The slide explains how to deal with this technical difficulty. But don't worry about it. We won't test you on it, and it is rather technical.

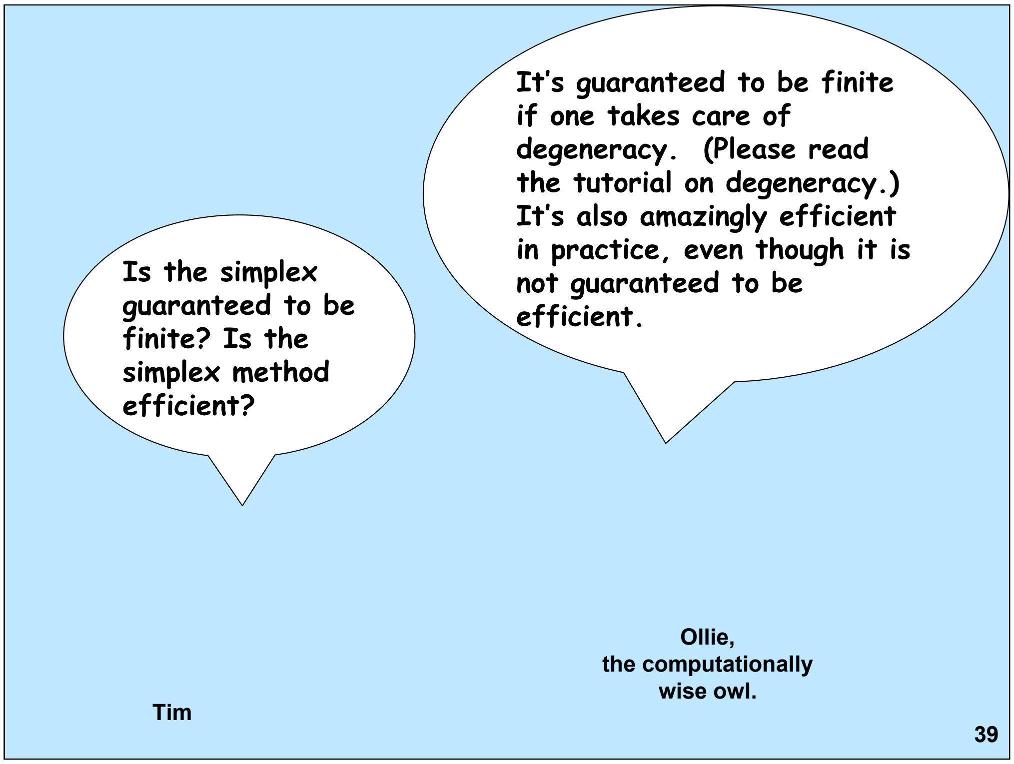


The simplex method requires a starting solution that is a corner point. So, some method is required to find a corner point solution, and it usually is a Phase 1 approach.

However, solving large problems can take a very long time, and it helps to start with a corner point that is very close to the optimum solution. One generic speedup technique is as follows: use some heuristic to try to get a corner point solution that is close to optimum. A heuristic is a (hopefully clever) approach that works pretty well but is not guaranteed to produce the desired solution.

If the heuristic produces a corner point solution, then use it as a starting point for Phase 2 of the simplex algorithm.

Otherwise, the heuristic may produce a basic solution that is not feasible, but only violates a small number of constraints. One can add artificial slack or surplus variables for those constraints, do a little more pivoting, and then have an excellent starting point for Phase 1. We don't teach techniques like this in 15.053 because they require too much expertise about performance of the simplex algorithm. But they can dramatically improve running times in practice.



If the simplex algorithm moves from corner point to corner point, and if it constantly improves the objective function, then it never repeats a corner point. Since there are a finite number of corner points (at most the number of ways of choosing m basic variables out of a possible n variables, where m is the number of equality constraints), the number of pivots is finite, and the algorithm is finite.

And now, it's time for …..

40

As usual, "Who wants a piece of candy?" is not stored on the website.

Overview

- **The simplex method has been a huge success in optimization.**
  - It solves linear programs efficiently
  - We can solve problems with millions of variables
  - It can be a starting point for problems that are not linear

- **The simplex method requires some simple techniques to get started**
  - Transformation into standard form
  - Phase 1 of the simplex algorithm

- **In practice it requires lots of "engineering"**
  - numerical stability
  - choosing pivot rules that are fast in practice
  - carrying out fast linear algebra
  - CPLEX is one of several good LP codes

Summary of today's lecture

- The simplex method starts with an initial bfs and (under non-degeneracy) each pivot improves the objective function.
  - It is very effective in practice.
  - very good rules for choosing the entering variable
  - very good implementations in practice to speed up the linear algebra
- Degeneracy: 0's in RHS can lead to no improvement in the objective function.
- 0's in the "reduced costs" can lead to alternative optima