The revised simplex method encodes complementary slackness. It turns out the revised simplex method actually brings with it some information about some things we’ve already discussed.

Look back to step 1 of the revised simplex method: we computed a vector I called $\mathbf{y} = (A_B^{-1})^T \mathbf{c}_B$, by way of solving the linear system $A_B^T \mathbf{y} = \mathbf{c}_B$. What does this vector $\mathbf{y}$ actually mean? Let’s go through an example and think about it; it was maximize $\mathbf{c} \cdot \mathbf{x}$ subject to $A \mathbf{x} \leq \mathbf{b}$ and $\mathbf{x} \geq 0$ for

$$A = \begin{bmatrix} 2 & 1 & 1 & 3 \\ 1 & 3 & 1 & 2 \end{bmatrix} \quad \mathbf{c} = \begin{bmatrix} 6 \\ 8 \\ 5 \\ 9 \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} 5 \\ 3 \end{bmatrix}.$$ 

Last time we started with a dictionary with $x_1, x_4$ basic (which would have taken two pivots from the initial dictionary) and did another pivot to get $x_1, x_2$ basic. Let’s say after that we do another pivot, which gets us to a dictionary with

$$\mathbf{x}_B = \begin{bmatrix} x_1 \\ x_3 \end{bmatrix} \quad \mathbf{x}_N = \begin{bmatrix} x_2 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} \quad \mathbf{b}^* = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

(remember I’m using $\mathbf{b}^*$ here to denote the vector of constants for this new dictionary that has $\mathbf{x}_B$ as its basic variables). This one will turn out to be optimal, so we expect that running our next iteration will tell us we don’t have an entering variable. But what will we actually compute to check this?

As we said above, the first step is to solve $A_B^T \mathbf{y} = \mathbf{c}_B$. So going back we read off

$$A_B = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \quad \mathbf{c}_B = \begin{bmatrix} 6 \\ 5 \end{bmatrix}.$$ 

So our system of equations for $A_B^T \mathbf{y} = \mathbf{c}_B$ is

$$2y_1 + y_2 = 6 \quad y_1 + y_2 = 5$$

which of course we can solve to $y_1 = 1$, $y_2 = 4$. And then we need to compute the row vector $\mathbf{c}_N^T - \mathbf{y}^T A_N$ which has the coefficients of the objective function:

$$\begin{bmatrix} 8 & 9 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 4 \end{bmatrix} = \begin{bmatrix} -5 & -2 & -1 & -4 \end{bmatrix}.$$ 

Sure enough, all of the coefficients are negative so we can conclude we’ve found an optimal dictionary, and thus an optimal solution:

$$\begin{bmatrix} x_1^* \\ x_2^* \\ x_3^* \\ x_4^* \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 1 \\ 0 \end{bmatrix}. $$

(make sure you can see how I got this from knowing $\mathbf{x}_B$, $\mathbf{x}_N$, and $\mathbf{b}^*$ for our optimal dictionary!). Thinking for a moment, asking for the dictionary to be optimal is asking $\mathbf{x}_N^T - \mathbf{y}^T A_N \leq 0$, or equivalently (after rearranging and transposing) that $\mathbf{c}_N \leq A_N^T \mathbf{y}$. So really this is asking that our $y_1, y_2$ above satisfy a bunch of inequalities:

$$y_1 + 3y_2 \geq 8 \quad 3y_1 + 2y_2 \geq 9 \quad y_1 \geq 0 \quad y_2 \geq 0.$$ 

Now I’ve written down two variables $y_1, y_2$ and a bunch of equalities and inequalities they have to satisfy. If you look at them, you’ll realize they’re all from the dual linear program! Specifically if we look at $A^T \mathbf{y} \geq \mathbf{c}$
and $\vec{y} \geq 0$, we get a list of 6 inequalities, and I’ve asked for two of them to be equalities. This should be sounding a lot like complementary slackness! Asking $A_N^\top \vec{y} = \vec{c}_B$ is asking some of the constraints in the dual to be made into equalities (the ones corresponding to basic variables in our dictionary, which take positive values in our case), and then checking that if $A_N^\top \vec{y} \geq \vec{c}_N$ is checking if the solution $\vec{y}$ we got was feasible (which, in our case, it was). So in the process of checking that $\vec{x}^*$ above was optimal, I produced an optimal dual solution $\vec{y}^*$ in basically the same way that I would have done using the theorem of complementary slackness - and this calculation was built into our revised simplex method.

What would happen with a non-optimal dictionary? Well, we’d still solve $A_N^\top \vec{y} = \vec{c}_B$ like we would when trying to apply complementary slackness. But then we’d find $A_N^\top \vec{y} \geq \vec{c}_N$ did not hold, i.e. there was a positive coefficient in the objective function. This would mean $\vec{y}$ is not a feasible solution to the dual, just like we’d expect from complementary slackness (since we constructed it from a feasible solution $\vec{x}^*$ to the primal which was not optimal).

**Economic interpretation of this.** The revised simplex calculations also encode the idea of “marginal costs” / “shadow prices” we talked about before. For our problem, let’s think of $x_1, x_2, x_3, x_4$ being quantities of four different things we could build in a factory, and the two constraints are limitations based on the resources available (let’s say how much labor we have available, and how many machines we have available to use for the production). Choosing $x_1$ and $x_3$ basic means we’ve chosen to devote all of our resources to those two goods, resulting in 2 units of $x_1$ (taking 4 hours of labor time and 2 machines) and 1 unit of $x_3$ (taking 1 hour of labor time and 1 machine). Our objective function tells us that $x_1$ returns a profit of $6$ and $x_2$ a profit of $5$.

By deciding our labor and machines are worth using in these configurations, solving $A_N^\top \vec{y} = \vec{c}_B$ assigns “shadow prices” $y_1$ and $y_2$ to labor and machines: we get $y_1$ is that each hour of labor is worth $1$ of production and each machine is worth $4$. (Plug those back into how much of each resource we’re using to produce each type of good, and the total profit from each, and check that it adds up!)

Now, what does the comparison of $\vec{c}_N$ and $A_N^\top \vec{y}$ mean? Well, each entry of $\vec{c}_N$ is the unit profit for doing a certain thing ($8$ per unit of $x_2$, $9$ per unit of $x_4$, and then $0$ each for leaving unused labor or machines - i.e. for allowing slack in those variables). And each entry of $A_N^\top \vec{y}$ is the unit cost for doing those things, in terms of the shadow prices we’ve set - if we value labor at $1$ per hour and machines at $4$ per use, then producing a unit of $x_2$ costs $13$ because it uses 1 hour of labor and 3 machines. Since the profit is only $8$, this is not a good idea and we would not want to start producing $x_2$ at the expense of what we’re already doing - this corresponds to the coefficient for $x_2$ being -$5$ in our dictionary and thus us not choosing $x_2$ to enter. Similarly the profit per unit of $x_4$ is $9$ but the cost is $11$ because it uses 3 hours of labor and 2 machines, so again it’s not worth it. And $x_5$ and $x_6$ correspond to leaving labor or machines idle, which (of course) returns no profit but costs $1$ and $4$ based on how we’ve valued these things. So all of the changes we can make from our current setup (setting $x_1 = 2$ and $x_3 = 1$) will make less profit, hence we don’t do them.

On the other hand, what if we were in a dictionary where one or more of the values of $\vec{c}_N$ is greater than the corresponding value of $A_N^\top \vec{y}$? This means that the profit per unit of the corresponding variable $x_i$ is more than the cost of producing it according to our current shadow prices (which are based on our current production choices). That means that starting to produce $x_i$ (and decreasing whatever we’re currently producing as appropriate) will make us more profit - hence $x_i$ can be chosen as an entering variable, going to a new dictionary which corresponds to a new solution with $x_i$ being produced.