**Department of Industrial Engineering & Operations Research**

**IEOR 162: Linear Programming & Network Flows (Spring 2022)**

**These notes are not the ultimate notes for the course and as such do not contain all material covered in lecture, they only serve as a reference. The notes contains material covered on Mon 2/14/22, Wed 2/16/22 and material that will be covered on Mon 2/21/22.**

# 1 Sensitivity Analysis

## 1.1 Change of Objective Function Coefficients

By how much (in what range) can change coefficients so that same solution remains optimal (But objective value may change)

$$\begin{array}{rrcl} \max & 3x_1 + 2x_2 & & \\ \text{s.t.} & 2x_1 + x_2 & \leq & 100 \\ & x_1 + x_2 & \leq & 80 \\ & x_1 & \leq & 40 \\ & x_1, x_2 & \geq & 0 \end{array}$$

$$\max c_1x_1 + 2x_2 \implies \text{slope} = \frac{-c_1}{2}$$

We need the slope to be between the constraints' slopes, therefore, $c_1$ must satisfy $-2 \leq \frac{-c_1}{2} \leq -1$

$$-2 \leq \frac{-c_1}{2} \implies c_1 \leq 4$$

$$\frac{-c_1}{2} \leq -1 \implies c_1 \geq 2$$

Therefore

$$2 \leq c_1 \leq 4$$

$$\max 3x_1 + c_2x_2 \implies \text{slope} = \frac{-3}{c_2}$$

We need the slope to be between the constraints' slopes, therefore, $c_2$ must satisfy $-2 \leq \frac{-3}{c_2} \leq -1$

$$-2 \leq \frac{-3}{c_2} \implies \begin{cases} -2c_2 \geq -3 & c_2 < 0 \\ 2c_2 \geq 3 & c_2 > 0 \end{cases} \implies \begin{cases} 2c_2 \leq 3 & c_2 < 0 \\ c_2 \geq \frac{3}{2} & c_2 > 0 \end{cases} \implies \begin{cases} c_2 < 0 & c_2 < 0 \\ c_2 \geq \frac{3}{2} & c_2 > 0 \end{cases}$$

$$\frac{-3}{c_2} \leq -1 \implies \begin{cases} 3 \leq c_2 & c_2 < 0 \\ c_2 \leq 3 & c_2 > 0 \end{cases} \implies 0 < c_2 \leq 3$$

Therefore

$$\frac{3}{2} \leq c_2 \leq 3$$

*Note that the statements $3 \geq c_2$ and $c_2 < 0$ are contradictory since both can not be true at the same time which is why we know $c_2$ must be positive.

## 1.2 Standard Form

We can convert a linear program with inequalities to standard form by adding a slack variable and changing to equalities.

$$\begin{array}{lrcl} \max & 3x_1 + 2x_2 & & \\ \text{s.t.} & 2x_1 + x_2 + s_1 & = & 100 \\ & x_1 + x_2 + s_2 & = & 80 \\ & x_1 + s_3 & = & 40 \\ & x_1, x_2, s_1, s_2, s_3 & \geq & 0 \end{array}$$

We then have what is called the 5-array to represent points: $(x_1, x_2, s_1, s_2, s_3)$

If the 5-array is equal to $(20, 60, 0, 0, 20)$ like it is for the optimal solution then the

$$\text{basis} = \{x_1, x_2, s_3\}$$

$$\text{not in the basis} = \{s_1, s_2\}$$

## 1.3 Change in Right Hand Side (RHS)

Let us think about if we changed the RHS of the "finishing" constraint by $\Delta$:

$$2x_1 + x_2 \leq 100 + \Delta$$

What is the range of $\Delta$ so that the same basis remains optimal? We need the same constraints to be binding.

$$\begin{cases} 2x_1 + x_2 = 100 + \Delta \\ x_1 + x_2 = 80 \end{cases} \implies \begin{cases} x_1 = 20 + \Delta \\ x_2 = 60 - \Delta \end{cases}$$

This means that our new 5-array will be $(20 + \Delta, 60 - \Delta, 0, 0, 20 - \Delta)$. We need to make sure that this stays feasible, therefore, all entries must be non-negative.

$$\begin{cases} 20 + \Delta \geq 0 \\ 60 - \Delta \geq 0 \\ 20 - \Delta \geq 0 \end{cases} \implies \begin{cases} \Delta \geq -20 \\ 60 \geq \Delta \\ 20 \geq \Delta \end{cases} \implies -20 \leq \Delta \leq 20$$

*Note (not related to sensitivity) if we let $\Delta$ be equal to $-20$ then we get the 5-array $(0, 80, 0, 0, 0)$, this means that our extreme point is the intersection of three constraints. We can then choose any two of the zeros to be non-basic and then let the third zero be basic.

## 1.4 Shadow Price

We can calculate the shadow price (also referred to as the dual price) by computing the objective function for the solution derived by changing the RHS of the first constraint.

$$3(20 + \Delta) + 2(60 - \Delta) = 180 + 1\Delta$$

This means that the shadow price of our first constraint is 1. This means that if someone were to offer to buy one finishing hour off of you, you would be willing to sell it for a price $\geq 1$, and inversely if someone were to offer to sell you one finishing hour, you would be willing to buy it for a price $\leq 1$. Note this is only valid inside the range of $\Delta$ we found above, otherwise we will have to resolve the problem.

## 1.5 Changes in the RHS of Other Constraints

### 1.5.1 Carpentry Constraint

$$x_1 + x_2 \leq 80 + \Delta$$

$$\begin{cases} 2x_1 + x_2 = 100 \\ x_1 + x_2 = 80 + \Delta \end{cases} \implies \begin{cases} x_1 = 20 - \Delta \\ x_2 = 60 + 2\Delta \end{cases}$$

Then we check for feasibility to get the range where this basic solution (intersection of the first two constraints) is feasible:

$$(20 - \Delta, 60 + 2\Delta, 0, 0, 20 + \Delta) \implies \begin{cases} 20 - \Delta \geq 0 \\ 60 + 2\Delta \geq 0 \\ 20 + \Delta \geq 0 \end{cases} \implies \begin{cases} \Delta \leq 20 \\ \Delta \geq -30 \\ \Delta \geq -20 \end{cases} \implies -20 \leq \Delta \leq 20$$

To evaluate the shadow price we plug this solution into the objective value:

$$3(20 - \Delta) + 2(60 + 2\Delta) = 180 + 1\Delta$$

Our shadow price for the carpentry constraint is, therefore, 1. Note in this case the range of $\Delta$ and the shadow price are the same as for the first constraint, this will not in general be the case.

### 1.5.2 Demand of Soldiers: a non-binding constraint

$$x_1 \leq 40 + \Delta$$

$$\begin{cases} 2x_1 + x_2 = 100 \\ x_1 + x_2 = 80 \end{cases} \implies \Delta \text{ has no effect on the optimal solution}$$

We still have to check when the current basic solution is still feasible.

$$(20, 60, 0, 0, 20 + \Delta) \implies 20 + \Delta \geq 0 \implies \Delta \geq -20$$

Note this means that we can increase the upper limit on the demand for soldiers as much as we want without changing the basic solution, but can only decrease it by 20 before we affect the optimal basis.

Now to look at the shadow price...

$$3(20 + 0\Delta) + 2(60 + 0\Delta) = 180 + 0\Delta$$

This means that the shadow price of the soldier demand constraint is equal to 0. Note, in general if we have a nonbinding constraint the shadow price associated with it will be equal to 0.