Lince Rumainum & Jennifer Vance
DSA 5113
Advanced Analytics & Metaheuristics
Group 8 - HW 3

Problem 1 – Boomer Sooner Air Services

Assumptions:

- Using the fuel burn from Table 1: Flight Details instead of the calculating the fuel burn rate of 310 gallons per hour.
- Ramp fee at home airport (KCID) is $0.

The **objective** is to minimize cost fuel plan

$$\sum_{a}^{A} x_a fuelCost_a + Z_a rampFee_a$$

Where A = {KCID1, KACK, KMMU, KBNA, KTUL, KCID2} , list of airports

Decision Variables:

X_a : the amount of fuel being purchased at the airport (in gallons)
Y_a : the amount of fuel inventory at the airport (in pounds)
Z_a : ramp fee being charged or not (binary value – 0 indicates no charged and 1 indicates fee charged)

Set:

AIRPORT = KCID1, KACK, KMMU, KBNA, KTUL, KCID2

Parameters:

fuelConversion – convert fuel in gallon to its weight in pounds = 6.7
$fuelBurn_a$ – amount of fuel burn to get to that airport, $a \in A$
$minLandingFuel_a$ – minimum fuel needed at landing, $a \in A$
$minPurchasedToWaiveFee_a$ – minimum fuel (in gallon) needed to be purchased to waive ramp fee, $a \in A$
$rampFee_a$ – ramp fee when fuel quantity is less than minimum requirement, $a \in A$
$passenger_a$ – the number of passengers being picked up at that airport, $a \in A$
avePassWeight – the average weight of the passenger with their luggage = 200 lbs
BOW – Basic Operating Weight = 22,800 lbs
maxRampWeight – maximum weight at takeoff = 36,400 lbs
maxLandingWeight – maximum weight at landing = 31,800 lbs
fuelTankCap – maximum fuel capacity = 14,000 lbs

Constraints:

(inventory at KCID1) $Y_{KCID1} = 0$

(minimum fuel purchase at KCID1) $X_{KCID1} \geq 7000 \,/\, fuelConversion$

(fuel flow (in pounds) from KCID1 to KACK)

$$Y_{KCID1} + X_{KCID1} \cdot fuelConversion \geq fuelBurn_{KACK} + X_{KACK}$$

(fuel flow (in pounds) from KACK to KMMU)

$$Y_{KACK} + X_{KACK} \cdot fuelConversion \geq fuelBurn_{KMMU} + X_{KMMU}$$

(fuel flow (in pounds) from KMMU to KBNA)

$$Y_{KMMU} + X_{KMMU} \cdot fuelConversion \geq fuelBurn_{KBNA} + X_{KBNA}$$

(fuel flow (in pounds) from KBNA to KTUL)

$$Y_{\mathrm{KBNA}} + X_{\mathrm{KBNA}} \cdot fuelConversion \geq fuelBurn_{\mathrm{KTUL}} + X_{\mathrm{KTUL}}$$

(fuel flow (in pounds) from KTUL to KCID2)

$$Y_{\mathrm{KTUL}} + X_{\mathrm{KTUL}} \cdot fuelConversion \geq fuelBurn_{KCID2} + X_{KCID2}$$

(minimum fuel inventory at each airport) $Y_{\mathrm{a}} \geq minLandingFuel_{\mathrm{a}}$, a $\in$ A

(maximum Ramp Weight) , a $\in$ A

$$passengers_{\mathrm{a}} * avePassWeight + BOW + Y_{\mathrm{a}} + X_{\mathrm{a}} \cdot fuelConversion \leq maxRampWeight$$

(maximum Landing Weight) , a $\in$ A

$$passengers_{\mathrm{a}} * avePassWeight + BOW + Y_{\mathrm{a}} \leq maxLandingWeight$$

(maximum tank capacity) , a $\in$ A

$$Y_{\mathrm{a}} + X_{\mathrm{a}} \cdot fuelConversion \leq fuelTankCap$$

(waived ramp Fee) , a $\in$ A

$$minPurchasedToWaiveFee_{\mathrm{a}} - X_{\mathrm{a}} \leq minPurchasedToWaiveFee_{\mathrm{a}} \cdot \mathrm{Z}_{\mathrm{a}}$$

(charge Ramp Fee) , a $\in$ A

$$X_{\mathrm{a}} \geq minPurchasedToWaiveFee_{\mathrm{a}} \cdot (1 - \mathrm{Z}_{\mathrm{a}})$$

Part a

Therefore, with the constraints listed, the minimum cost for the fuel plan is **$23,949.18**. Details of fuel inventory and purchases are shown below:

```
ampl: model Group8-HW3-p1.mod;
CPLEX 12.9.0.0: optimal integer solution; objective 23949.1791
9 MIP simplex iterations
0 branch-and-bound nodes


At KCID1 airport:
Purchasing fuel 13200.00 pounds (1970.15 gallon) cost: $7880.60

At KACK airport:
Fuel inventory: 8100.00 pounds (1208.96 gallon)
Purchasing fuel 1300.00 pounds (194.03 gallon) cost: $1614.33
Fuel purchased < 600 gallons. Ramp fee charged: $800.00

At KMMU airport:
Fuel inventory: 7200.00 pounds (1074.63 gallon)
Fuel purchased < 500 gallons. Ramp fee charged: $750.00

At KBNA airport:
Fuel inventory: 2500.00 pounds (373.13 gallon)
Purchasing fuel 8550.00 pounds (1276.12 gallon) cost: $8269.25

At KTUL airport:
Fuel inventory: 7250.00 pounds (1082.09 gallon)
Purchasing fuel 3350.00 pounds (500.00 gallon) cost: $4635.00

At KCID2 airport:
Fuel inventory: 7000.00 pounds (1044.78 gallon)


The minimum of fuel plan cost: $23949.18
```

In the case of no tankering, the cost goes up to **$29,857.91**. No tankering was achieved by changing the fuel flows and minimum fuel inventory constraints inequality to equality so any fuel being purchased is the fuel needed for minimum inventory and input fuel flow will always to output fuel flow. Details of fuel inventory and purchases are shown below:

```
At KCID1 airport:
Purchasing fuel 7600.00 pounds (1134.33 gallon) cost: $4537.31

At KACK airport:
Fuel inventory: 2500.00 pounds (373.13 gallon)
Purchasing fuel 2200.00 pounds (328.36 gallon) cost: $2731.94
Fuel purchased < 600 gallons. Ramp fee charged: $800.00

At KMMU airport:
Fuel inventory: 2500.00 pounds (373.13 gallon)
Purchasing fuel 4700.00 pounds (701.49 gallon) cost: $6306.42

At KBNA airport:
Fuel inventory: 2500.00 pounds (373.13 gallon)
Purchasing fuel 3800.00 pounds (567.16 gallon) cost: $3675.22
Fuel purchased < 650 gallons. Ramp fee charged: $600.00

At KTUL airport:
Fuel inventory: 2500.00 pounds (373.13 gallon)
Purchasing fuel 8100.00 pounds (1208.96 gallon) cost: $11207.01

At KCID2 airport:
Fuel inventory: 7000.00 pounds (1044.78 gallon)


The minimum of fuel plan cost: $29857.91
```

Part b

i. When at least 200 gallons needed if fuel is purchased, a new decision variable to indicate whether or not fuel is introduced.
subject to minFuelBuyAtKCID1:fuelPurchaseQty['KCID1']*fuelBuyIndicator['KCID1'] >= 200;
subject to minFuelBuyAtKACK:fuelPurchaseQty['KACK']*fuelBuyIndicator['KACK'] >= 200;
subject to minFuelBuyAtKMMU:fuelPurchaseQty['KMMU']*fuelBuyIndicator['KMMU'] >= 200;
subject to minFuelBuyAtKBNA:fuelPurchaseQty['KBNA']*fuelBuyIndicator['KBNA'] >= 200;
subject to minFuelBuyAtKTUL:fuelPurchaseQty['KTUL']*fuelBuyIndicator['KTUL'] >= 200;

Decision Variables:
F_a : fuel is being purchased or not (binary value – 0 indicates no purchase and 1 indicates purchased)

Constraints:
(minimum buy at KCID1) $X_{\text{KCID1}} \cdot F_{\text{KCID1}} \geq 200$
(minimum buy at KACK) $X_{\text{KACK}} \cdot F_{\text{KACK}} \geq 200$
(minimum buy at KMMU) $X_{\text{KMMU}} \cdot F_{\text{KMMU}} \geq 200$
(minimum buy at KBNA) $X_{\text{KBNA}} \cdot F_{\text{KBNA}} \geq 200$
(minimum buy at KTUL) $X_{\text{KTUL}} \cdot F_{\text{KTUL}} \geq 200$

ii. The change in solution and cost with the new constraints can be seen below. The minimum cost for the fuel plan is **$24,462.16**.

```
At KCID1 airport:
Purchasing fuel 13200.00 pounds (1970.15 gallon) cost: $7880.60

At KACK airport:
Fuel inventory: 8100.00 pounds (1208.96 gallon)
Purchasing fuel 1340.00 pounds (200.00 gallon) cost: $1664.00
Fuel purchased < 600 gallons. Ramp fee charged: $800.00

At KMMU airport:
Fuel inventory: 7240.00 pounds (1080.60 gallon)
Purchasing fuel 1340.00 pounds (200.00 gallon) cost: $1798.00
Fuel purchased < 500 gallons. Ramp fee charged: $750.00

At KBNA airport:
Fuel inventory: 3880.00 pounds (579.10 gallon)
Purchasing fuel 7170.00 pounds (1070.15 gallon) cost: $6934.57

At KTUL airport:
Fuel inventory: 7250.00 pounds (1082.09 gallon)
Purchasing fuel 3350.00 pounds (500.00 gallon) cost: $4635.00

At KCID2 airport:
Fuel inventory: 7000.00 pounds (1044.78 gallon)


The minimum of fuel plan cost: $24462.16
```

Code files:

Group8-HW3-p1.dat, Group8-HW3-p1a.mod, Group8-HW3-p1b.mod.

Problem 2 – We Got Gas!

```
reset;
option solver cplex;
###params
#L of gas to store
param A = 75000;
param B = 50000;
param C = 25000;
param D = 80000;
param E = 20000;

#individual costs
param T1_A = 1;
param T1_B = 2;
param T1_C = 3;
param T1_D = 1;
param T1_E = 1;
param T2_A = 2;
param T2_B = 3;
param T2_C = 4;
param T2_D = 1;
param T2_E = 1;
param T3_A = 2;
param T3_B = 3;
param T3_C = 1;
param T3_D = 2;
```

```
param T3_E = 1;
param T4_A = 1;
param T4_B = 3;
param T4_C = 2;
param T4_D = 2;
param T4_E = 1;
param T5_A = 4;
param T5_B = 1;
param T5_C = 1;
param T5_D = 3;
param T5_E = 1;
param T6_A = 4;
param T6_B = 4;
param T6_C = 4;
param T6_D = 4;
param T6_E = 1;
param T7_A = 5;
param T7_B = 5;
param T7_C = 5;
param T7_D = 5;
param T7_E = 5;
param T8_A = 3;
param T8_B = 2;
param T8_C = 1;
param T8_D = 2;
param T8_E = 5;

###vars
#tank capacity
var T1A;
var T1B;
var T1C >= 0;
var T1D >= 0;
var T1E >= 0;
var T2A >= 0;
var T2B >= 0;
var T2C >= 0;
var T2D >= 0;
var T2E >= 0;
var T3A >= 0;
var T3B >= 0;
var T3C >= 0;
var T3D >= 0;
var T3E >= 0;
var T4A >= 0;
var T4B >= 0;
var T4C >= 0;
var T4D >= 0;
```

```
var T4E >= 0;
var T5A >= 0;
var T5B >= 0;
var T5C >= 0;
var T5D >= 0;
var T5E >= 0;
var T6A >= 0;
var T6B >= 0;
var T6C >= 0;
var T6D >= 0;
var T6E >= 0;
var T7A >= 0;
var T7B >= 0;
var T7C >= 0;
var T7D >= 0;
var T7E >= 0;
var T8A >= 0;
var T8B >= 0;
var T8C >= 0;
var T8D >= 0;
var T8E >= 0;
#cost for each tank
var T1 = T1A*T1_A + T1B*T1_B + T1C*T1_C + T1D*T1_D + T1E*T1_E;
var T2 = T2A*T2_A + T2B*T2_B + T2C*T2_C + T2D*T2_D + T2E*T2_E;
var T3 = T3A*T3_A + T3B*T3_B + T3C*T3_C + T3D*T3_D + T3E*T3_E;
var T4 = T4A*T4_A + T4B*T4_B + T4C*T4_C + T4D*T4_D + T4E*T4_E;
var T5 = T5A*T5_A + T5B*T5_B + T5C*T5_C + T5D*T5_D + T5E*T5_E;
var T6 = T6A*T6_A + T6B*T6_B + T6C*T6_C + T6D*T6_D + T6E*T6_E;
var T7 = T7A*T7_A + T7B*T7_B + T7C*T7_C + T7D*T7_D + T7E*T7_E;
var T8 = T8A*T8_A + T8B*T8_B + T8C*T8_C + T8D*T8_D + T8E*T8_E;
###objective
minimize cost: T1 + T2 + T3 + T4 + T5 + T6 + T7 + T8;

###constraints
#storage space per tank
subject to T_1: T1 <= 25000;
subject to T_2: T2 <= 25000;
subject to T_3: T3 <= 30000;
subject to T_4: T4 <= 60000;
subject to T_5: T5 <= 80000;
subject to T_6: T6 <= 85000;
subject to T_7: T7 <= 100000;
subject to T_8: T8 <= 50000;
#amount of gas to be stored per gas type
subject to TA: T1A + T2A + T3A + T4A + T5A + T6A + T7A + T8A = 75000;
subject to TB: T1B + T2B + T3B + T4B + T5B + T6B + T7B + T8B = 50000;
subject to TC: T1C + T2C + T3C + T4C + T5C + T6C + T7C + T8C = 25000;
subject to TD: T1D + T2D + T3D + T4D + T5D + T6D + T7D + T8D = 80000;
```

subject to TE: T1E + T2E + T3E + T4E + T5E + T6E + T7E + T8E = 20000;
solve;

```
ampl: model 'C:\Users\clog_\Desktop\amplS2021\gas.mod';
CPLEX 20.1.0.0: sensitivity
CPLEX 20.1.0.0: optimal solution; objective 220000
1 dual simplex iterations (1 in phase I)
```

The minimal cost to store the gas is 220000. I'm assuming that is $220000. The breakdown of the cost of each tank is below.

```
ampl: display T1, T2, T3, T4, T5, T6, T7, T8;
T1 = 5000
T2 = 0
T3 = 25000
T4 = 60000
T5 = 80000
T6 = 0
T7 = 0
T8 = 50000
```

This breakdown makes sense given what we can see from glancing at the data. For example, tank 7 (T7) is 0 because it is expensive to use.

Code files:
Group8-HW3-p2.mod.

Problem 3 – ED On-call

Assumptions:

- Procedure will not be happening at the same time so one doctor can cover any procedure.
- Doctor that can perform a specific procedure is given one million procedure cost so it will never be chosen.

The **objective** is to total procedure costs:

$$\sum_{p}^{P}\sum_{d}^{D} x_{p,d} \cdot procedureCost_{p,d}$$

Where $d \in DOCTOR \; and \; p \in PROCEDURE$
Set:
DOCTOR = {Doc1, Doc2, Doc3, Doc4, Doc5, Doc6}
PROCEDURE = {P1, P2, P3, P4, P5, P6, P7, P8}
Decision Variables:
$X_{p,d}$: whether or not doctor is chosen for specific procedure (binary value – 0 indicates not chosen and 1 indicates chosen)
Parameters:
procedureCost(PROCEDURE, DOCTOR) – cost of procedure for specific doctor
Constraints:
(a doctor is chosen for every procedure)

$$\sum_{d}^{D} x_{p,d} \cdot procedureCost_{p,d} \geq 1, p \in PROCEDURE$$

Part b

Below is the data file for ED On-Call:

```
#Group8 - Rumainum & Vance
#HW 3
#AMPL model file for Problem 3-3
#ED On-Call

#indicate it's a data file---------------------------------------
data;

#parameters and sets---------------------------------------------
# set of doctors
set DOCTOR := Doc1 Doc2 Doc3 Doc4 Doc5 Doc6;
# set of procedures
set PROCEDURE := P1 P2 P3 P4 P5 P6 P7 P8;

# cost of procedure depending on which doctor
# Default cost of $1 million is used for some procedures
#   because that doctor can NOT perform that procedure.
#   Since the goal is to minimize cost, the doctor with
#   default price will not be chosen to do the procedure.
param procedureCost:
    Doc1  Doc2  Doc3  Doc4  Doc5  Doc6:=
P1  3100  4800     .     .  2950     .
P2     .  4800     .  3750  2950     .
P3     .  4800  2500  3750     .     .
P4  3100     .  2500  3750     .  1400
P5  3100     .     .  3750  2950     .
P6     .  4800     .     .  2950  1400
P7     .  4800  2500     .     .  1400
P8  3100     .  2500  3750     .     .;
```

```
#Group8 - Rumainum & Vance
#HW 3
#AMPL model file for Problem 3-3
#ED On-Call

#reset ampl--------------------------------------------------------------------
reset;

#options-----------------------------------------------------------------------
option solver cplex;

#parameters and sets-----------------------------------------------------------
set DOCTOR;    #set of doctors
set PROCEDURE; #set of procedures

# parameter variable:
# cost of procedure depending on the doctor.
#   Since the goal is to minimize cost, the doctor with
#   default price will not be chosen to do the procedure.
param procedureCost{PROCEDURE, DOCTOR}  default 1000000;

#decision variables-------------------------------------------------------------
#whether or not the doctor is choosen for the procedure
var x{PROCEDURE, DOCTOR} binary;

#objective: minimize total cost----------------------------------------
minimize totalProcedureCost:  sum {p in PROCEDURE, d in DOCTOR} (x[p,d] * procedureCost[p,d]);

#constraints:-------------------------------------------------------------------
#each procedure is covered by a doctor
subject to CoveredProcedure{p in PROCEDURE}: sum{d in DOCTOR} (x[p,d] * procedureCost[p,d]) >= 1;

#data-------------------------------------
data Group8-HW3-p3.dat

#commands--------------------------------
solve; # solve the minimize totalProcedureCost

#spaces
printf "\n\n";
```

Below is the results for ED On- Call:

```
ampl: model Group8-HW3-p3.mod;
CPLEX 12.9.0.0: optimal integer solution; objective 18050
0 MIP simplex iterations
0 branch-and-bound nodes


For the procedure P1, we will need Doc5. The cost: $2950.00
For the procedure P2, we will need Doc5. The cost: $2950.00
For the procedure P3, we will need Doc3. The cost: $2500.00
For the procedure P4, we will need Doc6. The cost: $1400.00
For the procedure P5, we will need Doc5. The cost: $2950.00
For the procedure P6, we will need Doc6. The cost: $1400.00
For the procedure P7, we will need Doc6. The cost: $1400.00
For the procedure P8, we will need Doc3. The cost: $2500.00

The minimum of total procedure cost: $18050.00
```

The minimum total cost is $18,050.00 by selecting:

- Doc 3 to do Procedure 3 and Procedure 8
- Doc 5 to do Procedure 1, Procedure 2, and Procedure 5
- Doc 6 to do Procedure 4, Procedure 6, and Procedure 7

Code files:

Group8-HW3-p3.dat, Group8-HW3-p3.mod.

Problem 4 – Galaxy Industries Revisited

Assumptions:

- Procedure will not be happening at the same time so one doctor can cover any procedure.
- Doctor that can perform a specific procedure is given one million procedure cost so it will never be chosen.

The **objective** is to maximize total profit:

$$(8 * SR + 5 * Z) - (3.75 * dSR1 + 3.25 * dSR2 + 2.80 * dSR3 + 1.90 * dSR4 + 1.95 * dZ1 + 2.15 * dZ2 + 2.95 * dZ3)$$

Decision Variables:

var SR >=0, <=3500, integer; #number of Space Rays
var Z >=0, <=7000, integer; #number of Zappers

var dSR1 >=0; # piecewise component 1 of var SR
var dSR2 >=0; # piecewise component 2 of var SR
var dSR3 >=0; # piecewise component 3 of var SR
var dSR4 >=0; # piecewise component 4 of var SR

var ySR1 binary; #to model piecewise cost for var SR
var ySR2 binary; #to model piecewise cost for var SR
var ySR3 binary; #to model piecewise cost for var SR

var dZ1 >=0; # piecewise component 1 of var Z
var dZ2 >=0; # piecewise component 2 of var Z
var dZ3 >=0; # piecewise component 3 of var Z

Variables for Disjunctive constraints (Either/Or)

indicator if the total units is under or over 3,000 units
var zIndicator binary;
constant for the Disjunctive constraints
param M := 4000;
Parameters:
procedureCost(PROCEDURE, DOCTOR) – cost of procedure for specific doctor
Constraints:
SPACE RAYS
connect SR with all its delta components
totalSR: SR = dSR1 + dSR2 + dSR3 + dSR4

We have to use all of d1 before you use d2,... dSR4 with SR upper bound is 3,500 units
0 to 500 units
piece1a: 500*ySR1 <= dSR1
piece1b: dSR1 <= 500

for 501 to 1,000 units
subject to piece2a: 500*ySR2 <= dSR2
subject to piece2b: dSR2 <= 500*ySR1

for 1,001 units to 2,000 units
subject to piece3a: 1000*ySR3 <= dSR3
subject to piece3b: dSR3 <= 1000*ySR2

the rest of the units, which is 1,500 units
subject to piece4: dSR4 <= 1500*ySR3

ZAPPERS
connect Z with all its delta components with Z upper bound is 7,000 units
totalZ: Z = dZ1 + dZ2 + dZ3

Since the price for Zappers are increasing for every delta components, the bounds are:
z1Bounds: dZ1 <= 1500
z2Bounds: dZ2 <= 500
z3Bounds: dZ3 <= 5000

up to 7,000 pounds of special plastic per week
2*SR + Z <= 7000
up to 250 hours (15,000 minutes) of production time per week
3*SR + 4*Z <= 15000

Using disjunctive constraint, if total units is more than 3,000 units, at least 45% are Zappers
(less than 3,000 units purchased) SR + Z <= 3000 + M*zIndicator
(greater than 3,000 units purchased): 0.45 *(SR + Z) <= Z + M*(1-zIndicator)

```
#Group8 - Rumainum & Vance
#HW 3
#AMPL model file for Problem 3-4
#Galaxy Industries Revisited

#reset ampl-------------------------------------------------------------------
reset;

#options----------------------------------------------------------------------
option solver cplex;

#decision variables-----------------------------------------------------------
#SR - Space Ray
#Z  - Zapper
#d  - delta
#1..n - component number
#i.e.: dSR1 - delta of piecewise component 1 of var SR

#Since there is a limit of 7,000 lbs of special plastic per week
#Space Ray's upper bound is 3,500 since it needs 2lbs of plastic
#Zapper's    upper bound is 7,000 since it needs 1lbs of plastic
var SR >=0, <=3500, integer; #number of Space Rays
var  Z >=0, <=7000, integer; #number of Zappers

var dSR1 >=0;  # piecewise component 1 of var SR
var dSR2 >=0;  # piecewise component 2 of var SR
var dSR3 >=0;  # piecewise component 3 of var SR
var dSR4 >=0;  # piecewise component 4 of var SR

var ySR1 binary; #to model piecewise cost for var SR
var ySR2 binary; #to model piecewise cost for var SR
var ySR3 binary; #to model piecewise cost for var SR

var dZ1 >=0;  # piecewise component 1 of var Z
var dZ2 >=0;  # piecewise component 2 of var Z
var dZ3 >=0;  # piecewise component 3 of var Z

#Variables for Disjunctive constraints (Either/Or)
#indicator if the total units is under or over 3,000 units
var zIndicator binary;
#constant for the Disjunctive constraints
param M := 4000;

#objective: maximize the company's profit per week-----------------------------------------
# revenue minus cost
maximize totalProfit: (8*SR + 5*Z) - (3.75*dSR1 + 3.25*dSR2 + 2.80*dSR3 + 1.90*dSR4 + 1.95*dZ1 + 2.15*dZ2 + 2.95*dZ3);

#constraints:-----------------------------------------------------------------
#---Space Rays
#connect SR with all its delta components;
subject to totalSR: SR = dSR1 + dSR2 + dSR3 + dSR4;

#ensure that the piece wise costs are used correctly,
#i.e., you have to use all of d1 before you use d2,...

# SR upper bound is 3,500 units
# 0 to 500 units
subject to piece1a: 500*ySR1 <= dSR1;
subject to piece1b: dSR1 <= 500;
# next 500 units
subject to piece2a: 500*ySR2 <= dSR2;
subject to piece2b: dSR2 <= 500*ySR1;
# next 1,000 units
subject to piece3a: 1000*ySR3 <= dSR3;
subject to piece3b: dSR3 <= 1000*ySR2;
# the rest of the units, which is 1,500 units
subject to piece4: dSR4 <= 1500*ySR3;

#--ZAPPERS
# connect Z with all its delta components
subject to totalZ: Z = dZ1 + dZ2 + dZ3;

# the price for Zappers are increasing for every delta components
# Z upper bound is 7,000 units
subject to z1Bounds: dZ1 <= 1500;
subject to z2Bounds: dZ2 <=  500;
subject to z3Bounds: dZ3 <= 5000;

# up to 7,000 pounds of special plastic per week
subject to plastic: 2*SR + Z <=  7000;
# up to 250 hours (15,000 minutes) of production time per week
subject to labor: 3*SR + 4*Z <= 15000;

# Using disjunctive constraint
# if total units is more than 3,000 units, at least 45% are Zappers
subject to lessThan3000: SR + Z <= 3000 + M*zIndicator;
subject to greaterThan3000: 0.45 *(SR + Z) <= Z + M*(1-zIndicator);

#commands---------------------------------
solve; # solve the maximize profit per week
```

```
The total of Space Ray guns made: 2390 units
The total of Space Ray guns made for the cost of $3.75:  500 units
The total of Space Ray guns made for the cost of $3.25:  500 units
The total of Space Ray guns made for the cost of $2.80: 1000 units
The total of Space Ray guns made for the cost of $1.90:  390 units
The total cost of of the Space Ray guns: $7041.00
The total revenue of of the Space Ray guns: $19120.00

The total of Zappers guns made: 1957 units
The total of Zappers guns made for the cost of $1.95: 1500 units
The total of Zappers guns made for the cost of $2.15:  457 units
The total of Zappers guns made for the cost of $2.95:    0 units
The total cost of of the Zappers guns: $3907.55
The total revenue of of the Zappers guns: $9785.00

The total weight of special plastic used: 6737 pounds out of 7000 pounds
The total labor time used: 14998 out of 15000 minutes
The total units made: 4347 units
The total percentage of Zapper guns made: 45.02%
The maximum total profit: $17956.45
```

By recreating the Galaxy Industry Inc, the total profit is **$17,956.45.**

Code files:
Group8-HW3-p4.mod.

Problem 5 – Valid Inequalities