Snapshots of Excel/ Python

Problem 1 (d) – *Snapshot Only, see Answers Sheet for Comparisons*

Assignment 2, Problem 1(d) - Linear Programming in Excel | Daniel Carpenter

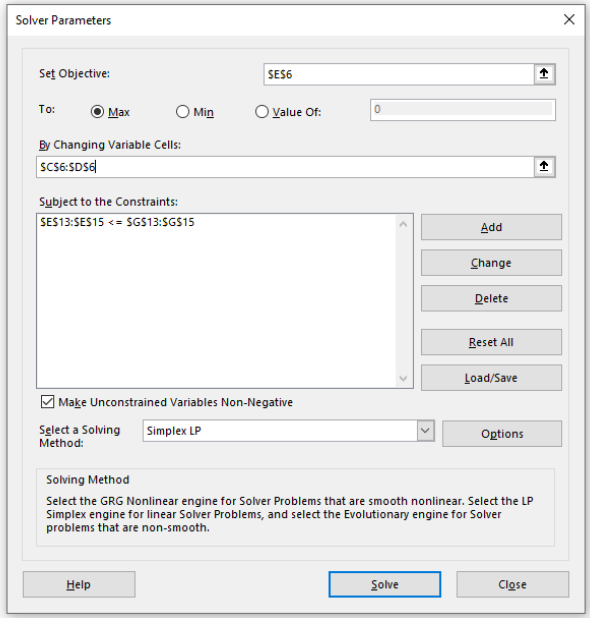
	Hard (units of resources per bag of seeds)	Serrated (units of raw resources per bag of seeds)	***Optimal Profit***
Descision Variables *(profit)*	**5.50573**	**4.27411**	**2,033.23**

Objective Function *(profit per seed bag)*	275.75	120.50	*2,033.23*

Resource	**Hard** *(units of resources per bag of seeds)*	**Serrated** *(units of raw resources per bag of seeds)*	**Total**		**Maximum monthly availability** *(units of Liters, kWh, and cm^3, respectively)*
Water	100.05	60.75	**810.50**	<=	810.50
Electricity	5.50	10.25	**74.09**	<=	655.80
Gas	75.30	24.84	**520.75**	<=	520.75
(non-negative note)	>= 0	>= 0			

Notes to Viewer

Decision Variable Outcome
Calculation
Hard-Coded Input



Problem 1 (e) – *Snapshot Only, see Answers Sheet for Comparisons*

Spyder (Python 3.8)

File Edit Search Source Run Debug Consoles Projects Tools View Help

sers\daniel.carpenter\OneDrive - the Chickasaw Nation\Documents\GitHub\OU-DSA\Systems Optimization\02 - Notes and Assignments\Week 03 - Linear Prog Modeling\ASSN - Linear Programming

...otes and Assignments\Week 03 - Linear Prog Modeling\ASSN - Linear Programming\Python LP Solutions.py

temp.py | Python LP Solutions.py*

```python
"""
from gurobipy import *

# Create New Model to Optimize
problem1Model = Model("Reddy_Mikks_Company")


# DECISION VARIABLES ==========================================================

hardSeed = {}
hardSeed = problem1Model.addVar(vtype = GRB.CONTINUOUS,
                                lb = 0,
                                ub = GRB.INFINITY)

serratedSeed = {}
serratedSeed = problem1Model.addVar(vtype = GRB.CONTINUOUS,
                                    lb = 0,
                                    ub = GRB.INFINITY)


# OBJECTIVE FUNCTION ==========================================================

objectiveFun = (275.75 * hardSeed) + (120.50 * serratedSeed)

# Set objective function to objectiveFun
problem1Model.setObjective(objectiveFun)

# Define whether to minimize or maximze the objective function
problem1Model.modelSense = GRB.MAXIMIZE

problem1Model.update()


# ADD CONSTRAINTS =============================================================

# Water Constraint
problem1Model.addConstr((100.05*hardSeed) + (60.75*serratedSeed) <= 810.50)

# Electricity Constraint
problem1Model.addConstr((5.50*hardSeed)   + (10.25*serratedSeed) <= 655.80)

# Gas Constraint
problem1Model.addConstr((75.30*hardSeed)  + (24.84*serratedSeed) <= 520.75)


# SOLVE MODEL =================================================================

# OptimiobjectiveFune
print("\n============== Problem 1 =================\n")
problem1Model.optimize()

# Print Output
if problem1Model.status == GRB.OPTIMAL:
    print("\n-----------------------------------------")
    print("\nOptimal Profit (USD):\t", problem1Model.objVal)
    print("\nOptimal Seed Amounts (Number of Seed Bags):")
    print("Hard Seed Amount:\t", hardSeed.x)
    print("Serrated Seed Amount:\t", serratedSeed.x)
```

no matches

Name	Type	Size	Value
hardSeed	Var	1	Var object of gurobipy module
objectiveFun	LinExpr	1	LinExpr object of gurobipy module
problem1Model	Model	1	Model object of gurobipy module
problem2Model	Model	1	Model object of gurobipy module
saudiBarrels	Var	1	Var object of gurobipy module
serratedSeed	Var	1	Var object of gurobipy module
venezuelaBarrels	Var	1	Var object of gurobipy module

Help | Variable explorer | Plots | Files

Console 1/A

```
Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.22.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/daniel.carpenter/OneDrive - the Chickasaw Nation/Documents/GitHub/OU-DSA/Systems Optimization/02 - Notes and Assignments/Week 03 -
Linear Prog Modeling/ASSN - Linear Programming/Python LP Solutions.py', wdir='C:/Users/daniel.carpenter/OneDrive - the Chickasaw Nation/Documents/GitHub/OU-DSA/
Systems Optimization/02 - Notes and Assignments/Week 03 - Linear Prog Modeling/ASSN - Linear Programming')
Restricted license - for non-production use only - expires 2022-01-13

============== Problem 1 =================

Gurobi Optimizer version 9.1.2 build v9.1.2rc0 (win64)
Thread count: 10 physical cores, 20 logical processors, using up to 20 threads
Optimize a model with 3 rows, 2 columns and 6 nonzeros
Model fingerprint: 0xfbc788fd
Coefficient statistics:
  Matrix range     [6e+00, 1e+02]
  Objective range  [1e+02, 3e+02]
  Bounds range     [0e+00, 0e+00]
  RHS range        [5e+02, 8e+02]
Presolve removed 1 rows and 0 columns
Presolve time: 0.00s
Presolved: 2 rows, 2 columns, 4 nonzeros

Iteration    Objective       Primal Inf.    Dual Inf.      Time
       0    2.2338368e+03   1.499712e+01   0.000000e+00      0s
       2    2.0332340e+03   0.000000e+00   0.000000e+00      0s

Solved in 2 iterations and 0.00 seconds
Optimal objective  2.033233991e+03

-----------------------------------------
```

IPython console | History

LSP Python: ready | conda (Python 3.8.8) | main [7] | Line 18, Col 1 | ASCII | CRLF | RW | Mem 34%

5:17 PM
9/13/2021

Problem 2 (i) – *Snapshot Only, see Answers Sheet for Comparisons*

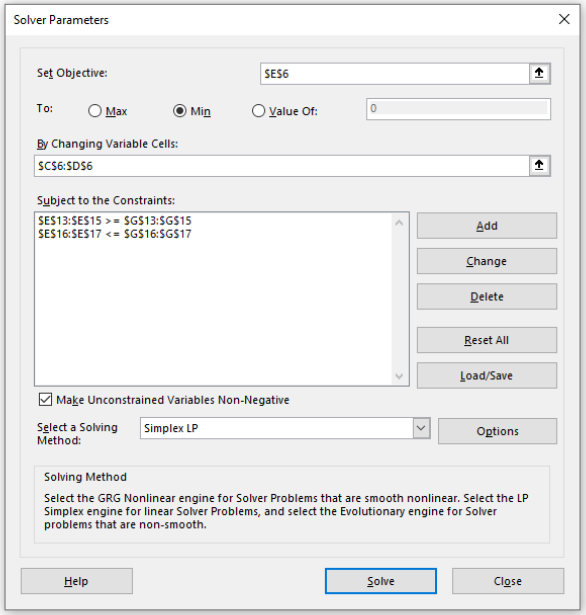
Assignment 2, Problem 2(i) - Linear Programming in Excel | Daniel Carpenter

	Saudi Arabia *(barrels of crude oil)*	Venezuela *(barrels of crude oil)*	*Optimal Cost*
Descision Variables *(cost)*	2,000.0	3,500.0	$ 92,500.00
Objective Function *(total cost per barrel)*	$ 20.00	$ 15.00	*92,500.00*

Material	Saudi Arabia *(barrels of crude oil)*	Venezuela *(barrels of crude oil)*	Total		Constraint of Barrel Production per Day
Gasoline	0.3	0.4	2,000.00	>=	2,000.0
Diesel	0.4	0.2	1,500.00	>=	1,500.0
Lubricants	0.2	0.3	1,450.00	>=	500.0
Maximum Production per Day	1.0	-	2,000.00	<=	9,000.0
Maximum Production per Day	-	1.0	3,500.00	<=	6,000.0
(non-negative note)	>= 0	>= 0			

Notes to Viewer

Decision Variable Outcome
Calculation
Hard-Coded Input



Problem 2 (j) – *Snapshot Only, see Answers Sheet for Comparisons*

Spyder (Python 3.8)

File Edit Search Source Run Debug Consoles Projects Tools View Help

...otes and Assignments\Week 03 - Linear Prog Modeling\ASSN - Linear Programming\Python LP Solutions.py

temp.py | Python LP Solutions.py

```python
problem2Model = Model("Reddy_Mikks_Company")


# DECISION VARIABLES ==========================================================

saudiBarrels = {}
saudiBarrels = problem2Model.addVar(vtype = GRB.CONTINUOUS,
                                    lb = 0,
                                    ub = GRB.INFINITY)

venezuelaBarrels = {}
venezuelaBarrels = problem2Model.addVar(vtype = GRB.CONTINUOUS,
                                        lb = 0,
                                        ub = GRB.INFINITY)


# OBJECTIVE FUNCTION ==========================================================

objectiveFun = (20 * saudiBarrels) + (15 * venezuelaBarrels)

# Set objective function to objectiveFun
problem2Model.setObjective(objectiveFun)

# Define whether to minimize or maximze the objective function
problem2Model.modelSense = GRB.MINIMIZE

problem2Model.update()


# ADD CONSTRAINTS =============================================================

# Gasoline Constraint
problem2Model.addConstr((0.3*saudiBarrels) + (0.4*venezuelaBarrels) >= 2000)

# Diesel Constraint
problem2Model.addConstr((0.4*saudiBarrels) + (0.2*venezuelaBarrels) >= 1500)

# Luibricant Constraint
problem2Model.addConstr((0.2*saudiBarrels) + (0.3*venezuelaBarrels) >= 500)

# Max production per Day Constraint - Saudi, Venezuela (respectively)
problem2Model.addConstr(saudiBarrels     <= 9000)
problem2Model.addConstr(venezuelaBarrels <= 6000)


# SOLVE MODEL =================================================================

# OptimiobjectiveFune
print("\n\n============= Problem 2 =================\n")
problem2Model.optimize()

# Print Output
if problem2Model.status == GRB.OPTIMAL:
    print("\n-----------------------------------------")
    print("\nOptimal Cost (USD):\t", problem2Model.objVal)
    print("\nOptimal Barrels per Day:")
    print("Saudi Arabia Barrels:\t", saudiBarrels.x)
    print("Venezuela Barrels:\t", venezuelaBarrels.x)
```

no matches

Jsers\daniel.carpenter\OneDrive - the Chickasaw Nation\Documents\GitHub\OU-DSA\Systems Optimization\02 - Notes and Assignments\Week 03 - Linear Prog Modeling\ASSN - Linear Programming

Name	Type	Size	Value
hardSeed	Var	1	Var object of gurobipy module
objectiveFun	LinExpr	1	LinExpr object of gurobipy module
problem1Model	Model	1	Model object of gurobipy module
problem2Model	Model	1	Model object of gurobipy module
saudiBarrels	Var	1	Var object of gurobipy module
serratedSeed	Var	1	Var object of gurobipy module
venezuelaBarrels	Var	1	Var object of gurobipy module

Help | Variable explorer | Plots | Files

Console 1/A

```
============= Problem 2 =================

Gurobi Optimizer version 9.1.2 build v9.1.2rc0 (win64)
Thread count: 10 physical cores, 20 logical processors, using up to 20 threads
Optimize a model with 5 rows, 2 columns and 8 nonzeros
Model fingerprint: 0x8ba5bdbd
Coefficient statistics:
  Matrix range     [2e-01, 1e+00]
  Objective range  [2e+01, 2e+01]
  Bounds range     [0e+00, 0e+00]
  RHS range        [5e+02, 9e+03]
Presolve removed 2 rows and 0 columns
Presolve time: 0.01s
Presolved: 3 rows, 2 columns, 6 nonzeros

Iteration    Objective       Primal Inf.    Dual Inf.      Time
       0    8.1555929e+04   3.126878e+02   0.000000e+00      0s
       1    9.2500000e+04   0.000000e+00   0.000000e+00      0s

Solved in 1 iterations and 0.01 seconds
Optimal objective  9.250000000e+04

-----------------------------------------

Optimal Cost (USD):  92500.0

Optimal Barrels per Day:
Saudi Arabia Barrels:    2000.0
Venezuela Barrels:   3500.0

In [2]:
```

IPython console | History

LSP Python: ready | conda (Python 3.8.8) | main [7] | Line 10, Col 13 | ASCII | CRLF | RW | Mem 34%

5:17 PM
9/13/2021