IDM Final Review

Jason Harrington

May 5 2024

Part 1: Written

1. (a) True or False: Heuristics are only implemented when solving an IP or LP is impossible.

   (b) True or False: One advantage of pyomo over PuLP is that pyomo has support for non-linear programs.

   (c) True or False: Prim's algorithm yields an exact optimal solution.

   (d) True or False: Any question that is determined to be in NP-Hard will always take longer to solve than a question that is determined to be in P.

   (e) Level n of a binary heap has how many nodes? (except for the bottom level)

   (a) 2^n
   (b) $2n - 2$
   (c) n^2
   (d) $2n + 2$
   (e) none of the above

   (f) A problem that is NP-Hard can not be solved in polynomial time, but can be verified in polynomial time.

   (g) If a problem that is NP-complete was proven to be able to be solved in polynomial time, it would show that all problems in NP are actually in P.

   (h) Match the network optimization algorithm with the problem type. Match the property of matter from the left column with the appropriate measurement device or technique.

Kernighan-Lin	Minimal Spanning Tree
Dijkstra's	Shortest Path
Prim's	Graph Bisection

   (i) Short Answer: Explain when you would want to solve a problem using Constraint Programming instead of other methods covered in this course.

   If you are more concerned about finding a feasible solution rather than finding an optimal one. Or if "optimality" is difficult to define.

2. You have been assigned to arrange the songs on the cassette version of Madonna's latest album. A cassette tape has two sides (1 and 2). The songs on each side of the cassette must total between 14 and 16 minutes in length. The length and type of each song are given in the following table.

Maddona's Latest Album		
Song Number	Song Type	Song Length (minutes)
1	Ballad	4
2	Hit	5
3	Ballad	3
4	Hit	2
5	Ballad	4
6	Hit	3
7	Hit	5
8	Ballad and Hit	4

Let $X_i = \begin{cases} 1 & \text{if song } i \text{ is selected for side 1 } \forall i \in 1, ..., 8 \\ 0 & \text{if song } i \text{ is instead selected for side 2} \end{cases}$

The assignment of songs to the tape must satisfy the following conditions.

1. Each side must have exactly two ballads.
2. Side 1 must have at least three hit songs.
3. Either song 5 or song 4 must be on side 1.
4. If songs 2 and 4 are on side 1, then song 5 must be on side 2.

(a) Formulate the logical constraint for condition 1.

$X_1 + X_3 + X_5 + X_8 = 2$

(b) Formulate the logical constraint for condition 2.

$X_2 + X_4 + X_6 + X_7 + X_8 \geq 3$

(c) Formulate the logical constraint for condition 3.

$X_4 + X_5 \geq 1$

(d) Formulate the logical constraint for condition 4.

$X_2 + X_4 \leq 2 - X_5$

3. Consider the following integer program and the optimal tableau for its relaxation.

$$\begin{array}{ll} \max & 2x + 5y \\ \text{s.t.} & x + y \leq 5 \\ & 2x + 3y \leq 11 \\ & x, y \in \mathbb{Z} \end{array}$$

z	x	y	s_1	s_2	RHS
1	$\frac{4}{3}$	0	0	$\frac{5}{3}$	$\frac{55}{3}$
0	$\frac{1}{3}$	0	1	$-\frac{1}{3}$	$\frac{4}{3}$
0	$\frac{2}{3}$	1	0	$\frac{1}{3}$	$\frac{11}{3}$

Perform one iteration of cutting plane simplex.

$\frac{1}{3}x + S_1 - S_2 + \frac{2}{3}S_2 = \frac{4}{3}$

$-1 + S_1 - S_2 = \frac{-1}{3}x - \frac{2}{3}S_2 + \frac{1}{3} \leq \frac{1}{3}$

$\frac{-1}{3}x - \frac{2}{3}S_2 + \frac{1}{3} + S_3 = 0$

new constraint: $\frac{-1}{3}x - \frac{2}{3}S_2 + S_3 = \frac{-1}{3}$

Z	X	Y	S_1	S_2	S_3	RHS
1	4/3	0	0	5/3	0	55/3
0	1/3	0	1	-1/3	0	4/3
0	2/3	1	0	1/3	0	11/3
0	-1/3	0	0	-2/3	1	-1/3

min ratio test:

$\left|\frac{5/3}{-2/3}\right| = \frac{5}{2} \Rightarrow$ Pivot on S_2

$\left|\frac{4/3}{-1/3}\right| = 4$

$R_4 = R_4 \cdot \frac{-3}{2}$

Z	X	Y	S_1	S_2	S_3	RHS
1	4/3	0	0	5/3	0	55/3
0	1/3	0	1	-1/3	0	4/3
0	2/3	1	0	1/3	0	11/3
0	1/2	0	0	1	-3/2	1/2

$R_1 = R_1 - 5/3\, R_4$

$R_2 = R_2 + 1/3\, R_4$

$R_3 = R_3 - 1/3\, R_4$

Z	X	Y	S_1	S_2	S_3	RHS
1	1/2	0	0	0	5/2	35/2
0	1/2	0	1	0	-1/2	3/2
0	1/2	1	0	0	1/2	7/3
0	1/2	0	0	1	-3/2	1/2

4. The Jonah Lee Narcotics Corporation produces the drug Rozac from four chemicals. Today they must produce 1,000 lb of the drug. The three active ingredients in Rozac are A, B, and C. By weight, at least 8% of Rozac must consist of A, at least 4% of B, and at least 2% of C. The cost per pound of each chemical and the amount of each active ingredient in 1 lb of each chemical are given in the following Table. It is necessary that at least 100 lb of chemical 2 be used. Formulate an LP whose solution would determine the cheapest way of producing today's batch of Rozac.

Chemical	Cost (per pound)	*A*	*B*	*C*
1	8	0.03	0.02	0.01
2	10	0.06	0.04	0.01
3	11	0.10	0.03	0.04
4	14	0.12	0.09	0.04

(a) Formulate this problem as an explicit linear program (i.e. a linear program that includes the coefficients in the model).

DVs: X_1 = amount of chemical 1 used (pounds)
X_2 = amount of chemical 2 used (pounds)
X_3 = amount of chemical 3 used (pounds)
X_4 = amount of chemical 4 used (pounds)

Obj: $\min\ z = 8x_1 + 10x_2 + 11x_3 + 14x_4$

s.t. $X_1 + X_2 + X_3 + X_4 = 1000$ (production quantity)
$.03X_1 + .06X_2 + .1X_3 + .12X_4 \geq 80$ (active ingredient 1)
$.02X_1 + .04X_2 + .03X_3 + .08X_4 \geq 40$ (active ingredient 2)
$.01X_1 + .01X_2 + .04X_3 + .04X_4 \geq 20$ (active ingredient 3)
$X_2 \geq 100$ (chem 2 min)
$X_1, X_2, X_3, X_4 \geq 0$

(b) Give an algebraic formulation of the linear program in your answer to part (a). Clearly define any sets, parameters, or matrices used in your formulation. The formulation should not contain any coefficients from the problem.

Sets: C = set of all chemicals

I = set of all active ingredients

Parameters: p_i = cost of producing 1 pound of chemical i $\forall i \in C$

a_{ij} = pounds of active ingredient j in chemical i $\forall i \in C$ $\forall j \in I$

T = # of pounds to be produced that day

b_j = % of active ingredient j needed in total compound $\forall j \in I$

m_i = min quantity of chemical i to be produced $\forall i \in C$

Variables: x_i pound of chemical i used in compound $\forall i \in C$

Obj: $\min z = \sum_{i=1}^{|C|} p_i x_i$

s.t.:

$$\sum_{i=1}^{|C|} x_i = T$$

$$\sum_{i=1}^{|C|} a_{ij} \cdot x_i \geq T \cdot b_j \quad \forall j \in I$$

$$x_i \geq m_i \quad \forall i \in C$$

$$x_i \geq 0 \quad \forall i \in C$$

(c) Formulate the constraints from your answer to part (b) the corresponding lines in **PuLP/Python** Assume any PuLP/Python variables referenced in that line have already been defined and are named as you named them in part (b). Also, assume the PuLP model object variable is named model.

```
model += lp.lpSum(Price[c] * X[c] for c in C)

model += lp.lpSum(X[c] for c in C) == T

for item in I:
    model += lp.lpSum(a[c][item] * X[c] for c in C) == T · b[item]

for chem in C:
    model += X[chem] >= m[chem]
```

(d) Circle and explain in the space below 3 mistakes with the following pyomo implementation for the constraints for this LP.

```
from pyomo.environ import *

model = AbstractModel()

model.chemicals = Set()
model.active_ingredients = Set()
model.costs = Param(model.chemicals)
model.min_active_ingrendient = Param(model.chemicals)
model.ing_val = Param(model.active_ingredients, model.chemicals)
model.amount = Var(model.chemicals, within=NonNegativeReals)

def costRule(model):                                   #1
    return sum(costs[n] * amount[n] for n in chemicals)

model.cost = Objective(rule=costRule)

def activeIngredientRule(model):                       #2
    s = sum(model.ing_val[a, c] * model.amount[c] for c in model.chemicals)
    value = s/sum(model.amount[c] for c in model.chemicals)
    return (model.min_active_ingredient[a], value, 100)

model.nutrientConstraint = activeIngredientRule        #3
```

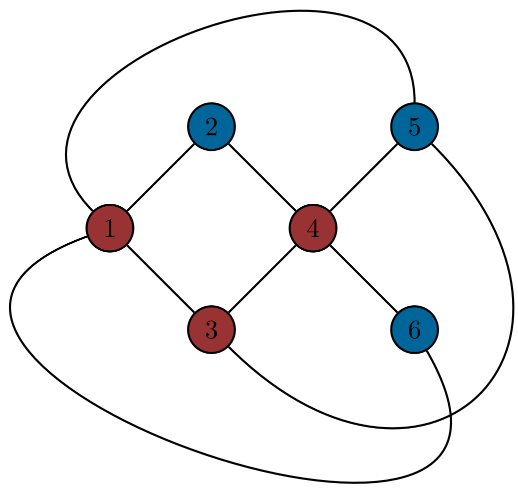
#1: need model.costs[n]

#2: need a as arg

#3. need Constraint(rule = activeIngredientRule)

5. (a) For the following graph, calculate the E_js, I_js, D_js and g_{ij}s for the first iteration of the Kernighan-Lin algorithm.

$$A = \{2, 5, 6\}$$
$$B = \{1, 3, 4\}$$



$g_{12} = 4 - 2(1) = 2$
$g_{15} = 5 - 2(1) = 3$
$g_{16} = 4 - 2(1) = 2$

$g_{32} = 1 - 2(0) = 1$
$g_{35} = 2 - 2(1) = 0$
$g_{36} = 1 - 2(0) = 1$

$g_{42} = 4 - 2(1) = 2$
$g_{45} = 5 - 2(1) = 3$
$g_{46} = 4 - 2(1) = 2$

j	E_j	I_j	D_j
1	3	1	2
2	2	0	2
3	1	2	-1
4	3	1	2
5	3	0	3
6	2	0	2

(b) What g_{ij}(s) has the highest value?

g_{15}, g_{45}

(c) What is the next step in the Kernighan-Lin algorithm after determining the largest g_{ij}?

Update D_j's that are adj to nodes i and j for largest g_{ij}

6. (a) Given the following Constraint Programming formulation, fill in the blanks for the incomplete implementation using Google OR-Tools below. Assume any OR-Tools/Python variables referenced in that line have already been defined and have the same names as in the formulation snippet. Also, assume the OR-Tools model object variable is named model

x_1, x_2, x_3 are integer variables with domains: $\{9, 10, 11, 12, 13, 14, 15, 16, 17\}$

Constraints:
x_1 is even
$x_2 - x_3$ is a multiple of 3
$x_3 + 2$ is a multiple of 4
$16 \leq 2x_1 - x_2 + 3x_1 \leq 100$

Code:

```
model.AddModuloEquality(0, x1 , 2)

model.AddModuloEquality(0, X1-X2 , 3)

model.AddModuloEquality( 2 , x3 , 4)

model.Add(2x1 − x2 + 3x3 <= 100 )

model.Add(2x1 − x2 + 3x3 >= 16 )
```

(b) Write the **OR-Tools** line/lines of code that you would write to add an objective function to the model that minimizes the sum of all the variables:

model.Minimize(sum[X_1, X_2, X_3])