



OR 7230 NETWORK ANALYSIS AND ADVANCED OPTIMIZATION

Aggressive Tumor Targeting with Advanced Optimization

By

Himanshu Saxena

23rd April 2019

Contents

Glossary	3
Task 1	4
1. Initial Model	4
2. Relaxed Model	4
3. Non-Critical Area Penalty Model.....	5
4. Modifications	6
5. Modeling the Modifications	6
Task 2	8
1. Initial Model	8
2. Relaxed Model	8
Model 1	8
Model 2	8
Model 3	9
Model 4	9
Model 5	9
Model 6	10
3. Non-Critical Area Penalty Model.....	10
4. Modifications	11
Appendix A: AMPL Code	13
1. Initial Model	13
2. Relaxed Model	15
Model 1	15
Model 2	17
Model 3	18
Model 4	20
Model 5	21
Model 6	23
3. Non-Critical Area Penalty Model.....	24
4. Modifications	27
Model 1	27
Model 2	29
Model 3	32
Model 4	34

Glossary

The following is a comprehensive listing of all the mathematical symbols that will be used in the formulation of the different models in this project:

Sets

B : Total number of beams

R : Rows in the problem matrix

C : Columns in the problem matrix

Parameters

$BV_{i,x,y}$; $i \in B, x \in R, y \in C$: Beam intensity value i from beam at point x, y

t_{min} : Minimum radiation value needed on tumorous area

c_{max} : Maximum radiation value allowed on critical area

nc_{max} : Maximum radiation value allowed on non-critical area

Variables

X_i ; $i \in B$: Intensity of the beam

A_{tumor} : Tumor Area, defined as a set of (x, y) values

$A_{critical}$: Critical Area, defined as a set of (x, y) values

$A_{non-critical}$: Non-critical Area, defined as a set of (x, y) values

$CA_{x,y}$: Slack variable for critical area dosage constraint

$TA_{x,y}$: Slack variable for tumor area dosage constraint

$NA_{x,y}$: Slack variable for non-critical area dosage constraint

Task 1

1. Initial Model

For this project, it was decided that the doctor's primary concern is the application of as much radiation to the tumor as possible, while still keeping the damage to the critical areas to a minimum. Since a patient undergoing radiotherapy is most likely in the advanced stages of persistent cancerous growth, it is necessary to pursue an aggressive strategy to target and destroy as much of the tumorous region as possible. Keeping this strategy in mind, the linear programming (LP) problem below was formulated:

$$\begin{aligned}
 & \text{Maximize } \sum_{i \in B} \sum_{x,y \in A_{tumor}} X_i * BV_{i,x,y} \\
 & \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \geq t_{min} ; \forall (x,y) \in A_{tumor} \\
 & \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \leq c_{max} ; \forall (x,y) \in A_{critical} \\
 & \text{subject to } X_i \geq 0
 \end{aligned}$$

It seeks to maximize the amount of radiation delivered to the tumorous region in a patient's body, while still respecting the upper and lower bounds on radiation delivered to critical and tumor regions respectively.

2. Relaxed Model

While the previously formulated LP is suitable for radiotherapy optimization, it is a rigid formulation that depends on the critical area and tumor radiation limits; some degree of relaxation must be allowed for cases where the oncologist cannot concretely provide the best limits. As such, it is necessary to introduce slack variables for both the minimum tumor dosage and maximum critical area dosage constraints. In the process of modeling this scenario, six different objective functions (five with the same constraints) were considered:

$$\begin{aligned}
 & \text{Minimize } \sum_{x,y \in A_{critical}} CA_{x,y} \text{ \{Minimize variation in dosage for critical area\} (1)} \\
 & \text{Minimize } \sum_{x,y \in A_{tumor}} TA_{x,y} \text{ \{Minimize variation in dosage for tumorous area\} (2)} \\
 & \text{Minimize } \sum_{x,y \in A_{critical}} CA_{x,y} + \sum_{x,y \in A_{tumor}} TA_{x,y} \text{ \{Minimize variation for both areas\} (4)} \\
 & \text{Minimize } \sum_{i \in B} \sum_{x,y \in A_{critical}} X_i * BV_{i,x,y} + \sum_{x,y \in A_{critical}} CA_{x,y} + \sum_{x,y \in A_{tumor}} TA_{x,y} \\
 & \quad \text{\{Minimize total dosage for critical area and variations in dosage for both areas\} (5)} \\
 & \text{Minimize } \sum_{i \in B} \sum_{x,y \in A_{critical}} X_i * BV_{i,x,y} + \sum_{x,y \in A_{critical}} CA_{x,y} + 2 * \sum_{x,y \in A_{tumor}} TA_{x,y} \\
 & \quad \text{\{Minimize total dosage for critical area and variation in dosage for critical area and 2*tumor area\} (6)}
 \end{aligned}$$

$$\begin{aligned}
 & \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \geq t_{min} - TA_{x,y}; \forall (x,y) \in A_{tumor} \\
 & \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \leq c_{max} + CA_{x,y}; \forall (x,y) \in A_{critical} \\
 & \text{subject to } X_i, TA_{x,y}, CA_{x,y} \geq 0
 \end{aligned}$$

The other model, #3, differs slightly from the others in terms of its constraints, and is shown below:

$$\begin{aligned}
& \text{Minimize } \sum_{x,y \in A_{tumor}} TA_{x,y} \text{ \{Minimize variation in dosage for tumor area\}} \textbf{(3)} \\
& \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \geq t_{min} - TA_{x,y}; \forall (x,y) \in A_{tumor} \\
& \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} = c_{max} + CA_{x,y}; \forall (x,y) \in A_{critical} \\
& \text{subject to } X_i, TA_{x,y}, CA_{x,y} \geq 0
\end{aligned}$$

For this model, the change in the critical area constraint is meant to strictly fix the critical area dosage so that the critical area constraint can have increased variation. As stated in the beginning of the report, the goal is to aggressively combat the tumorous region. As such, the tumor dosage slack variable is given more weightage than the slack for critical area dosage, i.e., it is more important to increase tumor dosage, which means that less tumor slack is needed.

All of these models will be considered and programmed into AMPL. Based on contextual requirements from the surgeon, any of the models can be picked and implemented. The cost and both slack values can be considered in order to evaluate the performance of the model.

3. Non-Critical Area Penalty Model

It is important to consider the possibility of less-than-optimal resolution on the images of the cancerous growth. Therefore, there is the possibility that radiation could be inadvertently applied to regions bordering the critical area. As such, in order to prevent accidentally applying any radiation to these non-critical border areas, a slight penalty must be applied that seeks to reduce the attractiveness of feasible solutions that radiate non-critical border zones. The following model (which incorporates model #6 from the previous section) attempts to address this issue:

$$\begin{aligned}
& \text{Minimize } \sum_{i \in B} \sum_{x,y \in A_{critical}} X_i * BV_{i,x,y} + \sum_{x,y \in A_{critical}} CA_{x,y} + 2 * \sum_{x,y \in A_{tumor}} TA_{x,y} \\
& \quad + 100 * \sum_{x,y \in A_{non-critical}} NA_{x,y} \\
& \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \geq t_{min} - TA_{x,y}; \forall (x,y) \in A_{tumor} \\
& \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \leq c_{max} + CA_{x,y}; \forall (x,y) \in A_{critical} \\
& \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \leq nc_{max} + NA_{x,y}; \forall (x,y) \in A_{non-critical} \\
& \text{subject to } X_i, TA_{x,y}, CA_{x,y}, NA_{x,y} \geq 0
\end{aligned}$$

This formulation introduces a new constraint, that of the non-critical area dosage, whereby the radiation applied has to be lower than a predefined non-critical area upper bound. Furthermore, the objective function also has the non-critical slack taken into account, with a heavy weightage. As such, the set of feasible solutions will prioritize those that have the lowest amount of radiation overflow onto neighboring non-critical regions. **NOTE:** In the AMPL code, the convention nc_{min} was used instead of nc_{max} .

4. Modifications

In order to increase the scope of this study, different variations on the classic radiotherapy example can be considered. The four variations are:

- Minimize total dosage for critical area with hard limits for both the tumor and critical area constraint slack variables. This will ensure that the system has some flexibility, such as upper and lower constraints defined by the oncologist as the situation demands. However, this limits the amount of flexibility enjoyed by the system and forces it to operate between some arbitrarily defined slack limits.
- Minimize total dosage for critical area with an additional constraint added as an upper bound on radiation dosage for non-critical areas. Since radiation is a highly damaging process for healthy tissue, this will ensure that the patient is treated with sufficient caution.
- Minimize the total dosage on the critical area and varying the diameter of the beam. Increasing the beam's diameter will ensure that radiation will be delivered to a wider column/row of cells. This has the obvious benefit of being able to target a greater amount of cancerous tissue per beamlet, but has the drawback of making the system more inflexible. This variation might be considered for patients with a highly advanced form of tumor that has metastasized and takes up a significant amount of space. As such, it would be more efficient to target this tumor with a wide-diameter beam, since it will be more efficient. This can also directly translate into manufacturing cost savings, since fewer beams (and fewer supporting components, machine complexity, etc.) will be required.
- Minimize the total dosage on the critical area and reducing the number of beams. The savings that a reduced number of beams will generate is an important consideration, given that cancer treatment is a drawn-out and expensive process. Any optimization in regards to cost per beamlet will be a significant improvement.

5. Modeling the Modifications

It was decided that all four variations could be modeled as the following:

$$\begin{aligned}
 & \text{Minimize } \sum_{i \in B} \sum_{x,y \in A_{critical}} X_i * BV_{i,x,y} + \sum_{x,y \in A_{critical}} CA_{x,y} + 2 * \sum_{x,y \in A_{tumor}} TA_{x,y} \\
 & \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \geq t_{min} - TA_{x,y}; \forall (x,y) \in A_{tumor} \\
 & \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \leq c_{max} + CA_{x,y}; \forall (x,y) \in A_{critical} \\
 & \text{subject to } X_i \geq 0, TA_{x,y} \geq -2, -2 \leq CA_{x,y} \leq 2
 \end{aligned}$$

This model takes into account the hard limits set on the critical area. For example, taking the general bounds of minimum dosage of 10 on tumorous regions and maximum dosage of 2 units

on critical areas, means that the tumor radiation dosage can go from 8 to any amount, and the critical area dosage can vary between 0 and 4.

This model takes into account the non-critical area.

$$\begin{aligned}
& \text{Minimize } \sum_{i \in B} \sum_{x,y \in A_{critical}} X_i * BV_{i,x,y} + \sum_{x,y \in A_{critical}} CA_{x,y} + 2 * \sum_{x,y \in A_{tumor}} TA_{x,y} \\
& \quad + \sum_{x,y \in A_{non-critical}} NA_{x,y} \\
& \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \geq t_{min} - TA_{x,y}; \forall (x,y) \in A_{tumor} \\
& \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \leq c_{max} + CA_{x,y}; \forall (x,y) \in A_{critical} \\
& \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \leq nc_{max} + NA_{x,y}; \forall (x,y) \in A_{non-critical} \\
& \text{subject to } X_i, TA_{x,y}, CA_{x,y}, NA_{x,y} \geq 0
\end{aligned}$$

Finally, the last two models: variation in the diameter of the beam and reducing the number of beams, do not need a change in the model. Rather, model #6 from section 2 will be applied to these scenarios with a different beam_raw.txt and model.dat files respectively. As such, the core model itself does not change, as shown:

$$\begin{aligned}
& \text{Minimize } \sum_{i \in B} \sum_{x,y \in A_{critical}} X_i * BV_{i,x,y} + \sum_{x,y \in A_{critical}} CA_{x,y} + 2 * \sum_{x,y \in A_{tumor}} TA_{x,y} \\
& \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \geq t_{min} - TA_{x,y}; \forall (x,y) \in A_{tumor} \\
& \text{subject to } \sum_{i \in B} X_i * BV_{i,x,y} \leq c_{max} + CA_{x,y}; \forall (x,y) \in A_{critical} \\
& \text{subject to } X_i, TA_{x,y}, CA_{x,y} \geq 0
\end{aligned}$$

Task 2

In this section, we implement the mathematical formulation in AMPL codes. The results of each implementation are discussed in detail.

1. Initial Model

In this section, our objective function is to maximize the radiation intensity on the tumor area, while implementing the procedure in such a way that a minimum intensity of 10 is applied at the tumor area and a maximum intensity of 2 is applied on the critical area.

Based on output, tumor limit is voided. According to constraint, the radiation intensity on tumor has a lower limit of 10 but the output shows a value of 6.05. Thus, the constraint is not satisfied, and the result is unbounded solution.

Result: Unbounded solution

2. Relaxed Model

In order to help the oncologist set the lower limit for tumor region and upper limit for critical area, we formulated 6 models that allowed for slight variation on the limits along with making sure that model is feasible. The values of objective function, slack variable for critical area (CA), slack variable for tumor area (TA), along with total dosage on critical and tumor area are obtained as follows:

Model 1

In this model our objective is to minimize variation in dosage for critical area.

Results:

Objective value = 0
CA=0
TA=5640
Total dosage on critical area = 0
Total dosage on tumor area = 0

Dosage to tumor and critical region is zero as we are reducing variation to critical area.

Model 2

In this model our objective is to minimize variation in dosage for tumorous area.

Results:

Objective value = 0
CA=2245.38
TA=0
Total dosage on critical area = 2488.018
Total dosage on tumor area = 12939.28

Dosage to tumor and critical region increases as we are reducing variation to tumor area.

Model 3

The objective is same as previous model, to minimize variation in dosage for tumor area. However, we changed our constraints to decreasing the maximum dosage constraint for critical area by critical area dosage slack and, decreasing the lower limit of the tumor area dosage by tumor area slack.

Results:

Objective value = 0
CA=2304.02
TA=0
Total dosage on critical area = 2576.017
Total dosage on tumor area = 12557.25

Dosage to tumor and critical region decreases as we reduce variation to tumor area and change constraint sign of critical area dosage constraint.

Model 4

In this model the objective is to minimize variation for both, critical and tumor areas.

Results:

Objective value = 576.7926
CA=401.967
TA=174.826
Total dosage on critical area = 627.886
Total dosage on tumor area = 11314.24

Model 5

In this model the objective is to minimize total dosage for critical area and variations in dosage for both areas

Results:

Objective value = 954.1397
CA=182.61
TA=527.85
Total dosage on critical area = 243.6802
Total dosage on tumor area = 10404.86

Dosage to tumor and critical region are 243.6802 and 10404.86 respectively as we are reducing variation to critical area, tumor area and dosage on critical area simultaneously.

Model 6

The objective is to minimize total dosage for critical area and variation in dosage for critical area and 2*tumor area. We changed our constraints to decreasing the maximum dosage constraint for critical area by critical area dosage slack and, decreasing the lower limit of the tumor area dosage by tumor area slack.

Results:

Objective value = 1345.63
CA= 443.472
TA= 171.331
Total dosage on critical area = 559.49
Total dosage on tumor area = 11818.17

Dosage to tumor and critical region are 559.4993 and 11818.18 respectively as we are reducing variation to critical area, 2*tumor area and dosage on critical area simultaneously.

Final Conclusion:

We are using model 6 because we are getting satisfactory results in terms of lowering critical area dosage and slacks on tumor area and critical area.

3. Non-Critical Area Penalty Model

In this model, we introduced a non-critical slack with a heavy weight of 100 in order to penalize radiation delivery to regions close to critical areas. Also, we added a new constraint for dosage on non-critical area with an upper bound dosage.

NOTE: In the AMPL code, the convention nc_{min} was used instead of nc_{max} .

Results:

Current dosage to Non-Critical Area beside Critical Area = 33.93
After Model Implementation-
Objective Value= 1544.07
CA= 417.17
TA= 263.35
NA= 0
Total dosage on critical area = 600.19
Total dosage on tumor area = 10801.18
Total Dosage on Non-Critical Area besides Critical Area= 13.88

Dosage to tumor and critical region are 600.19 and 10801.19 respectively as we are reducing variation to critical area, 2*tumor area, 100*non-critical and dosage on critical area simultaneously

4. Modifications

In this part we implemented 4 modifications from Task 1 Part 5 in AMPL

Results:

Modification 1: Limit slack/surplus variables.

Objective Value= 1344.3
CA= -240.11
TA= 776.279
Total dosage on critical area = 31.8
Total dosage on tumor area = 10655.6

Dosage to critical area reduces significantly as we are reducing variation to critical area, 2*tumor area and dosage on critical area simultaneously with fixed limits on slack variables

Modification 2: Apply constraint on overall dosage for non-critical area.

Current dosage to Overall Non-Critical Area = 43079.4

After Model Implementation-

Objective Value= 6870.4
CA= 264.3
TA= 2506.9
NA= 1156.6
Total dosage on critical area = 435.5
Total dosage on tumor area = 3332.9
Total Dosage on Non-Critical Area beside Critical Area= 7343.6

Dosage to non-critical area reduces significantly as we are reducing variation to critical area, 2*tumor area, non-critical area and dosage on critical area simultaneously with introducing extra constraint on non-critical area dosage.

Modification 3: Increasing the area covered by beam

Changed value-

Objective Value= 0
CA= 0
TA= 0
Total dosage on critical area = 0
Total dosage on tumor area = 173

Initial value-

Objective Value= 0
CA= 0
TA= 0
Total dosage on critical area = 0
Total dosage on tumor area = 218.5

Changing area covered by beam decreases total dosage on tumor area while keeping dosage on critical area 0.00

Modification 4: Reducing number of beams

Changed value-

Objective Value= 1348.7
CA= 440.9
TA= 176.4
Total dosage on critical area = 554.8
Total dosage on tumor area = 11799.8

Initial Value-

Objective Value= 1345.63
CA= 443.47
TA= 171.33
Total dosage on critical area = 559.50
Total dosage on tumor area = 11818.18

Changing number of beam from 126 to 120 slightly decreases total dosage on tumor and critical area.

Appendix A: AMPL Code

1. Initial Model

We are using the following files:

- * Model: Cure_Cancer_T1.mod
- * Data: Data_Import_T1.dat
- * Run: Cure_Cancer_T1.run
- * Output: Output_T1.out
- * Result: beam_raw.txt, critical_raw.txt, tumour_raw.txt

Cure_Cancer_T1.mod

```
#Number of beams
param num_beams >= 1, integer;

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Intensity per beam
var X {i in BEAMS} >= 0;
```

```

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

#Maximize total dosage in tumor area
maximize total_tumor_dosage: sum {i in BEAMS} sum {(x,y) in tumor_area} X[i] *
beam_values[i,x,y];

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min;

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= critical_max;

```

Data_Import_T1.dat

```

# Number of rows for input data file
param num_rows := 60;

# Number of columns for input data file
param num_cols := 80;

#Number of beams decided for the problem
param num_beams := 126;

#Minimum dosage at each tumor_values[x,y]
param tumor_min := 10;

#Maximum dosage at each critical_values[x,y]
param critical_max := 2;

#Reading the raw data for beams
read {i in BEAMS, x in ROWS, y in COLUMNS} beam_values[i,x,y] < beam_raw.txt;

#Reading the raw data for tumor matrix
read {x in ROWS, y in COLUMNS} tumor_values[x,y] < tumor_raw.txt;

#Reading the raw data for critical matrix
read {x in ROWS, y in COLUMNS} critical_values[x,y] < critical_raw.txt;

```

Cure_Cancer_T1.run

```

reset;
model Cure_Cancer_T1.mod;
data Data_Import_T1.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';
solve;

```

```
print {i in BEAMS}:X [i]>Output_T1.out;
```

2. Relaxed Model

We are using the following files:

- * Model: Cure_Cancer_T2_Model_1.mod, Cure_Cancer_T2_Model_2.mod, Cure_Cancer_T2_Model_3.mod, Cure_Cancer_T2_Model_4.mod, Cure_Cancer_T2_Model_5.mod, Cure_Cancer_T2_Model_6.mod
- * Data: Data_Import_T2.dat
- * Run: Cure_Cancer_T2_Model_1.run, Cure_Cancer_T2_Model_2.run, Cure_Cancer_T2_Model_3.run, Cure_Cancer_T2_Model_4.run, Cure_Cancer_T2_Model_5.run, Cure_Cancer_T2_Model_6.run
- * Output: Output_T2.out, Output_T2_Model_1.out, Output_T2_Model_1_Critical.out, Output_T2_Model_1_Tumor.out, Output_T2_Model_2.out, Output_T2_Model_2_Critical.out, Output_T2_Model_2_Tumor.out, Output_T2_Model_3.out, Output_T2_Model_3_Critical.out, Output_T2_Model_3_Tumor.out, Output_T2_Model_4.out, Output_T2_Model_4_Critical.out, Output_T2_Model_4_Tumor.out, Output_T2_Model_5.out, Output_T2_Model_5_Critical.out, Output_T2_Model_5_Tumor.out, Output_T2_Model_6.out, Output_T2_Model_6_Critical.out, Output_T2_Model_6_Tumor.out
- * Result: beam_raw.txt, critical_raw.txt, tumor_raw.txt

Model 1

Cure_Cancer_T2_Model_1.mod

```
#Number of beams
param num_beams >= 1, integer;

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;
```

```

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Intensity per beam
var X {i in BEAMS} >= 0;

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=0;

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=0;

#Minimize variation in dosage for critical area
minimize total_variation_dosage: sum{(x,y) in critical_area} CA[x,y] ;

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= critical_max + CA[x,y];

```

Cure_Cancer_T2_Model_1.run

```

reset;
model Cure_Cancer_T2_Model_1.mod;
data Data_Import_T2.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';
solve;
print {i in BEAMS}:X [i]>Output_T2_Model_1.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_1_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_1_Critical.out;
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];

```


Model 2

Cure_Cancer_T2_Model_2.mod

```
#Number of beams
param num_beams >= 1, integer;

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Intensity per beam
var X {i in BEAMS} >= 0;

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=0;

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=0;

#Minimize variation in dosage for tumor area
```

```

minimize total_variation_dosage: sum{(x,y) in tumor_area} TA[x,y];

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= critical_max + CA[x,y];

```

Cure_Cancer_T2_Model_2.run

```

reset;
model Cure_Cancer_T2_Model_2.mod;
data Data_Import_T2.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';
solve;
print {i in BEAMS}:X [i]>Output_T2_Model_2.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_2_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_2_Critical.out;
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];

```

Model 3

Cure_Cancer_T2_Model_3.mod

```

#Number of beams
param num_beams >= 1, integer;

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

```

```

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Intensity per beam
var X {i in BEAMS} >= 0;

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=0;

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=0;

#Minimize variation in dosage for tumor area
minimize total_variation_dosage: sum{(x,y) in tumor_area} TA[x,y];

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] == critical_max + CA[x,y];

```

Cure_Cancer_T2_Model_3.run

```

reset;
model Cure_Cancer_T2_Model_3.mod;
data Data_Import_T2.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';
solve;
print {i in BEAMS}:X [i]>Output_T2_Model_3.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_3_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_3_Critical.out;

```

```
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];
```

Model 4

Cure_Cancer_T2_Model_4.mod

```
#Number of beams
param num_beams >= 1, integer;

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Intensity per beam
var X {i in BEAMS} >= 0;

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=0;
```

```

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=0;

#Minimize variation in dosage for tumor and critical area
minimize total_variation_dosage: sum{(x,y) in tumor_area} TA[x,y] + sum{(x,y) in
critical_area} CA[x,y] ;

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= critical_max + CA[x,y];

```

Cure_Cancer_T2_Model_4.run

```

reset;
model Cure_Cancer_T2_Model_4.mod;
data Data_Import_T2.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';
solve;
print {i in BEAMS}:X [i]>Output_T2_Model_4.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_4_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_4_Critical.out;
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];

```

Model 5

Cure_Cancer_T2_Model_5.mod

```

#Number of beams
param num_beams >= 1, integer;

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns

```

```

set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Intensity per beam
var X {i in BEAMS} >= 0;

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=0;

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=0;

#Minimize total dosage for critical area along-with variation in dosage for tumor and
critical area
minimize total_variation_dosage: sum {i in BEAMS} sum {(x,y) in critical_area} X[i] *
beam_values[i,x,y] + sum{(x,y) in critical_area} CA[x,y]+ sum{(x,y) in tumor_area}
TA[x,y];

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= critical_max + CA[x,y];

```

Cure_Cancer_T2_Model_5.run

```

reset;
model Cure_Cancer_T2_Model_5.mod;
data Data_Import_T2.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';

```

```

solve;
print {i in BEAMS}:X [i]>Output_T2_Model_5.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_5_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_5_Critical.out;
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];

```

Model 6

Cure_Cancer_T2_Model_6.mod

```

#Number of beams
param num_beams >= 1, integer;

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Intensity per beam
var X {i in BEAMS} >= 0;

# Defining tumor area

```

```

set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=0;

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=0;

#Minimize total dosage for critical area along-with variation in dosage for 2*tumor
and critical area
minimize total_variation_dosage: sum {i in BEAMS} sum {(x,y) in critical_area} X[i] *
beam_values[i,x,y] + sum{(x,y) in critical_area} CA[x,y]+2*sum{(x,y) in tumor_area}
TA[x,y];

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] == critical_max + CA[x,y];

```

Cure_Cancer_T2_Model_6.run

```

reset;
model Cure_Cancer_T2_Model_6.mod;
data Data_Import_T2.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';
solve;
print {i in BEAMS}:X [i]>Output_T2_Model_6.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_6_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T2_Model_6_Critical.out;
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];

```

3. Non-Critical Area Penalty Model

We are using the following files:

- * Model: Cure_Cancer_T3_Model_1.mod
- * Data: Data_Import_T3.dat,
- * Run: Cure_Cancer_T3_Model_1.run
- * Output: Output_T3_Model_1.out, Output_T3_Model_1_Critical.out,
Output_T3_Model_1_Non_Critical.out, Output_T3_Model_1_Tumor.out
- * Result: beam_raw.txt, critical_raw.txt, non_critical_besides_critical.txt, tumor_raw.txt

Cure_Cancer_T3_Model_1.mod

```
#Number of beams
param num_beams >= 1, integer;

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param non_critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Lower limit for radiation therapy tumor region
param non_critical_min;

#Intensity per beam
var X {i in BEAMS} >= 0;

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

# Defining non critical area
set non_critical_area := {x in ROWS, y in COLUMNS: non_critical_values[x,y] > 0};
```

```

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=0;

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=0;

#Slack variable for critical dosage constraint
var NA {(x,y) in non_critical_area} >=0;

#Minimize total dosage for critical area along-with variation in dosage for 2*tumor
and critical area
minimize total_variation_dosage: sum {i in BEAMS} sum {(x,y) in critical_area} X[i] *
beam_values[i,x,y]+ 100*sum{(x,y) in non_critical_area} NA[x,y] + sum{(x,y) in
critical_area} CA[x,y]+2*sum{(x,y) in tumor_area} TA[x,y];

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= critical_max + CA[x,y];

#Constraint on dosage at each non_critical location [x,y]
subject to non_critical_limit {(x,y) in non_critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= non_critical_min + NA[x,y];

```

Data_Import_T3.dat

```

# Number of rows for input data file
param num_rows := 60;

# Number of columns for input data file
param num_cols := 80;

#Number of beams decided for the problem
param num_beams := 126;

#Maximum dosage at each critical_values[x,y]
param critical_max := 2;

#Minimum dosage at each critical_values[x,y]
param tumor_min := 10;

#Minimum dosage at each non_critical_values[x,y]
param non_critical_min := 2;

#Reading the raw data for beams
read {i in BEAMS, x in ROWS, y in COLUMNS} beam_values[i,x,y] < beam_raw.txt;

#Reading the raw data for tumor matrix
read {x in ROWS, y in COLUMNS} tumor_values[x,y] < tumor_raw.txt;

```

```
#Reading the raw data for critical matrix
read {x in ROWS, y in COLUMNS} critical_values[x,y] < critical_raw.txt;

#Reading the raw data for non-critical matrix
read {x in ROWS, y in COLUMNS} non_critical_values[x,y] <
non_critical_besides_critical.txt;
```

Cure_Cancer_T3_Model_1.run

```
reset;
model Cure_Cancer_T3_Model_1.mod;
data Data_Import_T3.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';
solve;
print {i in BEAMS}:X [i]>Output_T3_Model_1.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T3_Model_1_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T3_Model_1_Critical.out;
print {(x,y) in non_critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T3_Model_1_Non_Critical.out;
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];
display sum{(x,y) in non_critical_area} NA[x,y];
```

4. Modifications

Model 1

Hard limits on critical area slack

We are using the following files:

- * Model: Cure_Cancer_T5_Model_1.mod
- * Data: Cure_Cancer_T5_Model_1.dat
- * Run: Cure_Cancer_T5_Model_1.run
- * Output: Output_T5_Model_1.out, Output_T5_Model_1_Critical.out, Output_T5_Model_1_Tumor.out
- * Result: beam_raw.txt, critical_raw.txt, tumor_raw.txt

Cure_Cancer_T5_Model_1.mod

```
#Number of beams
param num_beams >= 1, integer;
```

```

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Intensity per beam
var X {i in BEAMS} >= 0;

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=-2;

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=-2,<=2;

#Minimize total dosage for critical area along-with variation in dosage for 2*tumor
and critical area
minimize total_variation_dosage: sum {i in BEAMS} sum {(x,y) in critical_area} X[i] *
beam_values[i,x,y]+ sum{(x,y) in critical_area} CA[x,y]+2*sum{(x,y) in tumor_area}
TA[x,y];

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

```

```

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= critical_max + CA[x,y];

```

Cure_Cancer_T5_Model_1.dat

```

# Number of rows for input data file
param num_rows := 60;

# Number of columns for input data file
param num_cols := 80;

#Number of beams decided for the problem
param num_beams := 126;

#Maximum dosage at each critical_values[x,y]
param critical_max := 2;

#Minimum dosage at each critical_values[x,y]
param tumor_min := 10;

#Reading the raw data for beams
read {i in BEAMS, x in ROWS, y in COLUMNS} beam_values[i,x,y] < beam_raw.txt;

#Reading the raw data for tumor matrix
read {x in ROWS, y in COLUMNS} tumor_values[x,y] < tumor_raw.txt;

#Reading the raw data for critical matrix
read {x in ROWS, y in COLUMNS} critical_values[x,y] < critical_raw.txt;

```

Cure_Cancer_T5_Model_1.run

```

reset;
model Cure_Cancer_T5_Model_1.mod;
data Cure_Cancer_T5_Model_1.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';
solve;
print {i in BEAMS}:X [i]>Output_T5_Model_1.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T5_Model_1_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T5_Model_1_Critical.out;
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];

```

Model 2

Including non-critical area

We are using the following files:

- * Model: Cure_Cancer_T5_Model_2.mod
- * Data: Cure_Cancer_T5_Model_2.dat
- * Run: Cure_Cancer_T5_Model_2.run
- * Output: Output_T5_Model_2.out, Output_T5_Model_2_Critical.out, Output_T5_Model_2_Non_Critical.out, Output_T5_Model_2_Tumor.out
- * Result: beam_raw.txt, critical_raw.txt, non_critical_area.txt, tumor_raw.txt

Cure_Cancer_T5_Model_2.mod

```
#Number of beams
param num_beams >= 1, integer;

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param non_critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Lower limit for radiation therapy tumor region
param non_critical_min;

#Intensity per beam
var X {i in BEAMS} >= 0;
```

```

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

# Defining non critical area
set non_critical_area := {x in ROWS, y in COLUMNS: non_critical_values[x,y] > 0};

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=0;

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=0;

#Slack variable for critical dosage constraint
var NA {(x,y) in non_critical_area} >=0;

#Minimize total dosage for critical area along-with variation in dosage for 2*tumor
and critical area
minimize total_variation_dosage: sum {i in BEAMS} sum {(x,y) in critical_area} X[i] *
beam_values[i,x,y]+ sum{(x,y) in non_critical_area} NA[x,y] + sum{(x,y) in
critical_area} CA[x,y]+2*sum{(x,y) in tumor_area} TA[x,y];

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= critical_max + CA[x,y];

#Constraint on dosage at each non_critical location [x,y]
subject to non_critical_limit {(x,y) in non_critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= non_critical_min + NA[x,y];

Cure_Cancer_T5_Model_2.dat

# Number of rows for input data file
param num_rows := 60;

# Number of columns for input data file
param num_cols := 80;

#Number of beams decided for the problem
param num_beams := 126;

#Maximum dosage at each critical_values[x,y]
param critical_max := 2;

#Minimum dosage at each critical_values[x,y]
param tumor_min := 10;

```

```

#Minimum dosage at each non_critical_values[x,y]
param non_critical_min := 2;

#Reading the raw data for beams
read {i in BEAMS, x in ROWS, y in COLUMNS} beam_values[i,x,y] < beam_raw.txt;

#Reading the raw data for tumor matrix
read {x in ROWS, y in COLUMNS} tumor_values[x,y] < tumor_raw.txt;

#Reading the raw data for critical matrix
read {x in ROWS, y in COLUMNS} critical_values[x,y] < critical_raw.txt;

#Reading the raw data for non-critical matrix
read {x in ROWS, y in COLUMNS} non_critical_values[x,y] < non_critical_area.txt;

```

Cure_Cancer_T5_Model_2.run

```

reset;
model Cure_Cancer_T5_Model_2.mod;
data Cure_Cancer_T5_Model_2.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';
solve;
print {i in BEAMS}:X [i]>Output_T5_Model_2.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T5_Model_2_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T5_Model_2_Critical.out;
print {(x,y) in non_critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T5_Model_2_Non_Critical.out;
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];
display sum{(x,y) in non_critical_area} NA[x,y];

```

Model 3

Modify diameter of beam

We are using the following files:

- * Model: Cure_Cancer_T5_Model_3.mod
- * Data: Cure_Cancer_T5_Model_3.dat
- * Run: Cure_Cancer_T5_Model_3.run
- * Output: Output_T5_Model_3.out, Output_T5_Model_3_Critical.out,
Output_T5_Model_3_Tumor.out
- * Result: beam_raw.txt, critical_raw.txt, tumor_raw.txt, beam_raw_modified.txt, specs.txt

Cure_Cancer_T5_Model_3.mod

```

#Number of beams
param num_beams >= 1, integer;

```



```

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Intensity per beam
var X {i in BEAMS} >= 0;

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=0;

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=0;

#Minimize total dosage for critical area along-with variation in dosage for 2*tumor
and critical area
minimize total_variation_dosage: sum {i in BEAMS} sum {(x,y) in critical_area} X[i] *
beam_values[i,x,y]+ sum{(x,y) in critical_area} CA[x,y]+2*sum{(x,y) in tumor_area}
TA[x,y];

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

```

```
#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= critical_max + CA[x,y];
```

Cure_Cancer_T5_Model_3.dat

```
# Number of rows for input data file
param num_rows := 8;

# Number of columns for input data file
param num_cols := 8;

#Number of beams decided for the problem
param num_beams := 5;

#Maximum dosage at each critical_values[x,y]
param critical_max := 2;

#Minimum dosage at each critical_values[x,y]
param tumor_min := 10;

#Reading the raw data for beams
read {i in BEAMS, x in ROWS, y in COLUMNS} beam_values[i,x,y] <
beam_raw_modified.txt;

#Reading the raw data for tumor matrix
read {x in ROWS, y in COLUMNS} tumor_values[x,y] < tumor_raw.txt;

#Reading the raw data for critical matrix
read {x in ROWS, y in COLUMNS} critical_values[x,y] < critical_raw.txt;
```

Cure_Cancer_T5_Model_3.run

```
reset;
model Cure_Cancer_T5_Model_3.mod;
data Cure_Cancer_T5_Model_3.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';
solve;
print {i in BEAMS}:X [i]>Output_T5_Model_3.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T5_Model_3_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T5_Model_3_Critical.out;
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];
```

Model 4

Reducing number of beams

We are using the following files:

- * Model: Cure_Cancer_T5_Model_4.mod,
- * Data: Cure_Cancer_T5_Model_4.dat,
- * Run: Cure_Cancer_T5_Model_4.run,
- * Output: Output_T5_Model_4.out, Output_T5_Model_4_Critical.out, Output_T5_Model_4_Tumor.out
- * Result: beam_raw.txt, critical_raw.txt, non_critical_area.txt, tumor_raw.txt

Cure_Cancer_T5_Model_4.mod

```
#Number of beams
param num_beams >= 1, integer;

#Number of rows
param num_rows >= 1, integer;

#Number of columns
param num_cols >= 1, integer;

#Number of beams in the data file to be read
set BEAMS := 1 .. num_beams;

#Number of rows
set ROWS := 1 .. num_rows;

#Number of columns
set COLUMNS := 1 .. num_cols;

#Values of eacy beam from the data file
param beam_values {BEAMS, ROWS, COLUMNS} >= 0;

#Matrix values for tumor region
param tumor_values {ROWS, COLUMNS} >= 0;

#Matrix values for critical region
param critical_values {ROWS, COLUMNS} >= 0;

#Upper limit for radiation therapy critical region
param critical_max;

#Lower limit for radiation therapy tumor region
param tumor_min;

#Intensity per beam
var X {i in BEAMS} >= 0;

# Defining tumor area
set tumor_area := {x in ROWS, y in COLUMNS: tumor_values[x,y] > 0};

# Defining critical area
set critical_area := {x in ROWS, y in COLUMNS: critical_values[x,y] > 0};
```

```

#Slack variable for tumor dosage constraint
var TA {(x,y) in tumor_area} >=0;

#Slack variable for critical dosage constraint
var CA {(x,y) in critical_area} >=0;

#Minimize total dosage for critical area along-with variation in dosage for 2*tumor
and critical area
minimize total_variation_dosage: sum {i in BEAMS} sum {(x,y) in critical_area} X[i] *
beam_values[i,x,y]+ sum{(x,y) in critical_area} CA[x,y]+2*sum{(x,y) in tumor_area}
TA[x,y];

#Constraint on dosage at each tumor location [x,y]
subject to tumor_limit {(x,y) in tumor_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] >= tumor_min - TA[x,y];

#Constraint on dosage at each critical location [x,y]
subject to critical_limit {(x,y) in critical_area} : sum {i in BEAMS} X[i] *
beam_values[i,x,y] <= critical_max + CA[x,y];

```

Cure_Cancer_T5_Model_4.dat

```

# Number of rows for input data file
param num_rows := 60;

# Number of columns for input data file
param num_cols := 80;

#Number of beams decided for the problem
param num_beams := 120;

#Maximum dosage at each critical_values[x,y]
param critical_max := 2;

#Minimum dosage at each critical_values[x,y]
param tumor_min := 10;

#Reading the raw data for beams
read {i in BEAMS, x in ROWS, y in COLUMNS} beam_values[i,x,y] < beam_raw.txt;

#Reading the raw data for tumor matrix
read {x in ROWS, y in COLUMNS} tumor_values[x,y] < tumor_raw.txt;

#Reading the raw data for critical matrix
read {x in ROWS, y in COLUMNS} critical_values[x,y] < critical_raw.txt;

```

Cure_Cancer_T5_Model_4.run

```

reset;
model Cure_Cancer_T5_Model_4.mod;
data Cure_Cancer_T5_Model_4.dat;
option solver 'D:/Northeastern University/Network Analysis and Advanced
Optimization/AMPL/ampl_mswin64/xpress.exe';

```

```

solve;
print {i in BEAMS}:X [i]>Output_T5_Model_4.out;
print {(x,y) in tumor_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T5_Model_4_Tumor.out;
print {(x,y) in critical_area} : sum {i in BEAMS} X[i] * beam_values[i,x,y] >
Output_T5_Model_4_Critical.out;
display sum{(x,y) in tumor_area} TA[x,y];
display sum{(x,y) in critical_area} CA[x,y];

```