A COLUMN GENERATION APPROACH FOR EVALUATING DELIVERY EFFICIENCIES OF COLLIMATOR TECHNOLOGIES IN IMRT TREATMENT PLANNING

by

Merve Gören

B.S., Industrial Engineering, Boğaziçi University, 2012

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Industrial Engineering
Boğaziçi University

2014

ACKNOWLEDGEMENTS

I am grateful for having the opportunity to work with Assoc. Prof. Z. Caner Taşkın. I would like to thank him for his patience to my impatient behaviours, for his sensibility and understanding in my down times and for his endless support in every struggle that I face. Throughout this study, he encouraged me to push my limits. This work would not be completed without his wise advices and suggestions.

I also would like to express my gratitude to Prof. İ. Kuban Altınel and Prof. Serpil Sayın for taking part in my thesis jury. I am also grateful to Prof. Ümit Bilge for accepting me as a member of the BUFAIM laboratory.

I would like to thank my dear friend Gökalp for being there to help whenever I need. His support made everything easier for me all the time. I also want to thank Gökçe and Mert for the times we spent together in BUFAIM and Oylum, Nisa, Banu, Turgut, İsmail and the rest of my fellow assistants for all the laughter and beautiful memories.

I am grateful to my friends Serkan, Tekin, Çağrışım, Gökhan and Azize for trusting and encouraging me. I let go all the thoughts, clear my mind and fill up my life energy during the times I spend with them.

I will never be able to find the correct words to express my gratitude to my family. Their trust, support and infinite love are the building blocks behind all my accomplishments. I owe them everything I have in my life.

I also thank TUBITAK for their financial support during my graduate study.

ABSTRACT

A COLUMN GENERATION APPROACH FOR EVALUATING DELIVERY EFFICIENCIES OF COLLIMATOR TECHNOLOGIES IN IMRT TREATMENT PLANNING

Intensity Modulated Radiation Therapy (IMRT) is a form of cancer treatment which delivers radiation beams to the patient from several directions by using a linear accelerator and a collimator. At the leaf sequencing optimization step of the IMRT treatment planning, the intensity matrices are decomposed into a set of deliverable apertures and their associated intensities. Collimator systems used in IMRT can form different geometric shapes of apertures depending on their physical capabilities. Hence, comparing the delivery efficiency of different collimator technologies is important to determine the value added by the different technologies. In this thesis, we compare the efficiency of using regular, rotating and dual multileaf collimator (MLC) systems under different combinations of consecutiveness, interdigitation and rectangular constraints and a virtual freeform collimator. We formulate the problem of minimizing total beam-on time (BOT) as a large-scale linear programming problem. To deal with its dimensionality, we propose a column generation approach. Although there exists a general master problem structure, subproblem depends on the used collimator system technology. Therefore, we model each subproblem individually and apply a different solution method to each of them. We test our approach on a set of clinical problem instances. Our results indicate that the dual MLC under consecutiveness constraint yields very similar beam-on time as a virtual freeform collimator which can form any possible segment shape by opening or closing each bixel independently. Our approach provides a ranking between other collimator technologies in terms of their delivery efficiencies.

ÖZET

DOZ YOĞUNLUK AYARLI RADYOTERAPİ TEDAVİ PLANLAMASINDA KULLANILAN FARKLI IŞIN YÖNLENDİRİCİ TEKNOLOJİLERİNİN SÜTUN TÜRETME TEKNİĞİ KULLANILARAK KIYASLANMASI

Doz Yoğunluk Ayarlı Radyoterapi (DYAR), bir lineer hızlandırıcı ve ışın yönlendirici kullanarak hastaya farklı açılardan radyasyon ışınları veren kanser tedavi yöntemidir. DYAR tedavi planlamasının yaprak dizilim optimizasyonu aşamasında, ışın profil matrisleri uygulanabilir altmatrislere ve onların ilişkili yoğunluklarına ayrıştırılır. DYAR tedavi planlamasında kullanılan ışın yönlendiriciler fiziksel özelliklerine bağlı olarak farklı altmatrisler oluştururlar. Bu sebeple farklı ışın yönlendirici teknolojilerinin tedavi verimlilliklerinin kıyaslanması oldukça önemlidir. Bu tezde ardışıklık, çakışma ve dikdörtgensel kısıtların farklı kombinasyonları altında; normal, döner başlıklı ve çift katmanlı çok yapraklı ışın yönlendiricilerinin ve sanal serbest ışın yönlendiricisinin tedavi verimlilikleri kıyaslanmıştır. Toplam ışın gönderim süresinin enküçüklendiği problem, büyük çaplı lineer programlama modeli olarak oluşturulmuştur. Problemin boyutlarıyla başa çıkmak için sütun türetme tekniğini kullanılmıştır. Genel bir ana problem yapısı olmasına rağmen, altproblem yapısı kullanılan ışın yönlendirici teknolojisine bağlı olarak değiştiğinden her bir altproblem ayrı olarak modellenip çözülmüştür. Önerilen çözüm tekniği klinik problem örnekleri üzerinde test edilmiştr. Sonuçlar toplam ışın gönderim süresi açısından farklı ışın yönlendiricilerinin sıralamasını ortaya çıkarmıştır. Beklendiği gibi en kısa ışın gönderim süresi sanal serbest ışın yönlendiricisi tarafından bulunmuştur. Çift katmanlı çok yapraklı ışın yönlendiricisi de yaklaşık olarak sanal serbest ışın yönlendiricisi kadar iyi sonuçlar vermektedirler. Buna karşılık, sadece dikdörtgensel şekiller oluşturabilen ışın yönlendiricilerin çok uzun ışın gönderim süreleri verdiği gözlemlenmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

A	Intensity matrix
a_{ij}	Required intensity value for bixel (i,j)
B_{ij}	Binary variable denoting whether the bottom leaf of column j is in row i
B'_{ij}	Binary variable denoting whether bixel (i,j) is covered by the bottom leaf
c_1	Initial position of the left leaf in the single pass algorithm
c_2	Initial position of the right leaf in the single pass algorithm
c_1^*	Optimal position of the left leaf in the single pass algorithm
c_2^*	Optimal position of the right leaf in the single pass algorithm
H_{ij}	Binary variable denoting whether bixel (i,j) is exposed horizontally
L	Maximum intensity value for the intensity matrix example
L_{ij}	Binary variable denoting whether the left leaf of row i is in column j
L'_{ij}	Binary variable denoting whether bixel (i,j) is covered by the left leaf
M	A big number
M_1	First shape matrix example
M_2	Second shape matrix example
M_3	Third shape matrix example
M_4	Forth shape matrix example
M_5	Fifth shape matrix example
M_6	Sixth shape matrix example
M_7	Seventh shape matrix example
M_8	Eighth shape matrix example
m	Number of rows of the intensity matrix A
n	Number of columns of the intensity matrix A
r_N	Reduced cost of the MP which defines the objective of the SP

R_{ij}	Binary variable denoting whether the right leaf of row i is in column j
R'_{ij}	Binary variable denoting whether bixel (i,j) is covered by the right leaf
S_{ijk}	Binary variable denoting whether bixel (i,j) is exposed in shape matrix k
S_{ij}	Binary variable denoting whether bixel (i,j) is exposed in the shape matrix
T_{ij}	Binary variable denoting whether the top leaf of column j is in row i
T'_{ij}	Binary variable denoting whether bixel (i,j) is covered by the top leaf
v	Sum of the coefficients in the single pass algorithm
$\bar{v}$	Maximum sum of the coefficients in the single pass algorithm
v^*	Optimal objective value as a result of single pass algorithm
V_{ij}	Binary variable denoting whether bixel (i,j) is exposed vertically
X_k	BOT value for shape matrix k
y	Binary variable denoting whether a column consecutive shape matrix is used in the decomposition done by rotating MLC under consecutiveness constraint
κ	Cardinality of the set of all feasible shape matrices
$\widehat{\kappa}$	Cardinality of the subset of shape matrices used at RMP
λ_{ij}	Dual variable associated with the $(i,j)^{th}$ equality constraint of RMP
τ	Performance ratio
τ_{Free}	BOT performance ratio of the freeform collimator
τ_{Dual}	BOT performance ratio of the dual MLC
τ_{Reg_Cons}	BOT performance ratio of the regular MLC under consecutiveness constraint
τ_{Reg_Int}	BOT performance ratio of the regular MLC under interdigitation constraint

τ_{Rect}	BOT performance ratio of the regular MLC under rectangular constraint
τ_{Rot_Cons}	BOT performance ratio of the rotating MLC under consecutiveness constraint
τ_{Rot_Int}	BOT performance ratio of the rotating MLC under interdigitation constraint
$P_s(\tau)$	Percentage of times that collimator system s can find BOT values that are within factor τ of the minimum BOT

LIST OF ACRONYMS/ABBREVIATIONS

BAO	Beam Angle Optimization
BOT	Beam-on Time
DAO	Direct Aperture Optimization
GPU	Graphics Processing Unit
IMRT	Intensity Modulated Radiation Therapy
LSO	Leaf Sequencing Optimization
MP	Master Problem
MIP	Mixed Integer Programming
MLC	Multileaf Collimator
RMP	Restricted Master Problem
SP(λ)	Subproblem

1. INTRODUCTION

Radiotherapy is a form of cancer treatment that delivers ionizing radiation or high energy x-ray beams with the aim of damaging the DNA of cancerous cells and stopping their growth [1]. Since radiation beam can affect cancerous and normal tissue alike, treatment plan must be done carefully. Intensity Modulated Radiation Therapy (IMRT) is a kind of radiotherapy which delivers radiation beams to the patient from several directions (i.e. beam orientations) by using a linear accelerator and a collimator. While linear accelerator rotates around the patient, collimator shapes the beams by moving their leaves that forms different segments or apertures. This technique provides high degree of control over the dose distribution that is received by a patient [2].

IMRT treatment can be delivered either statically or dynamically. In the static approach, also called "step and shoot" technique, the leaves are stationary while the radiation beam is on. In other words, linear accelerator stops at a predetermined beam angle position, new aperture is formed and only then the radiation source is turned on. In the dynamic approach, new apertures are formed while the radiation is on, so collimator leaves keep moving during the treatment [3]. In this thesis, we focus on the static approach.

IMRT treatment planning is typically composed of three phases. The first phase is called beam angle optimization (or geometry problem). Radiation is delivered through a set of beam angles and the goal of beam angle optimization is to determine those different angles [4–6]. The second phase is called fluence map optimization (or intensity problem). The fluence map or intensity profile exists for each beam angle and it is a nonnegative integer matrix of intensity values. The objective of this phase is to give the required dose of radiation to malignant tissues while healthy ones are spared [7–9]. In this thesis, we investigate the last phase, called the leaf sequencing optimization (LSO) or realization problem. In LSO, the intensity matrices are decomposed into a set of deliverable apertures (i.e. segments or shape matrices) and their associated intensities [10–12].

Dividing the IMRT treatment plan into stages and dealing with them separately may cause loss in treatment quality. Hence, some integrated approaches aim to solve multiple stages of the IMRT problem at once. For instance, direct aperture optimization (DAO) approaches integrate the fluence map optimization and leaf sequencing optimization stages and try solving the problem as a single optimization problem [13–16].

Collimator systems used in IMRT can form different geometric shapes of apertures depending on their physical capabilities. Hence, different kinds of collimator systems lead to changes in the feasible aperture shapes. The regular multileaf collimator (MLC) is a type of collimator which is composed of a set of leaf pairs; left and right leaves with the same sizes. In the IMRT systems with regular MLC, the linear accelerator rotates around the patient, stops at a predetermined angle and the radiation is transmitted through the aperture constructed by the regular MLC. Another collimator type is the rotating MLC. It is a regular MLC with the ability of head rotation by 90°. In this system, the apertures are constructed first with left-right pairs of leaves, then the MLC rotates and apertures are constructed with top-bottom pairs of leaves; or vice versa. Dual MLC is a kind of collimator that has two orthogonal pairs of leaves; one pair operating horizontally and the other pair independently operating vertically. Hence, it is expected to construct more complex apertures [17, 18].

There are some common constraints that appear in the collimator systems and restricting the feasible aperture shapes even more. Those constraints can be summarized as consecutiveness constraint (i.e. consecutive ones property), interdigitation constraint (i.e. interleaf motion (collision) constraint), connectedness constraint, rectangular constraint (i.e. jaws only), minimum leaf separation, maximum leaf spread and tongue and groove constraint. The detailed explanations of the constraints are provided in the upcoming chapters.

We can measure the efficiency of the decomposition in LSO stage by using total beam-on time (BOT), total set-up time or total treatment time as the evaluation criteria. BOT is the total time that radiation source is on and it is found proportionally from the sum of the aperture intensities. Setup time occurs due to the change of

apertures and it is approximately estimated as a function of total number of apertures used in the decomposition. The treatment time is estimated as the summation of beam-on time and setup time. Keeping these criteria at their minimum level is of interest and we investigate the leaf sequencing optimization problem with the objective of minimizing total beam-on time in this thesis.

MLCs are technologically advanced systems which are therefore difficult and expensive to build, operate and maintain [19]. Additional features and more flexibility means additional costs. Hence, the comparison of different collimator technologies is important to determine the value added by the additional features. In this thesis, we compare the efficiency of using regular, rotating and dual MLC systems under different combinations of consecutiveness, interdigitation and rectangular constraints; and a virtual freeform collimator, which is a collimator that can form any possible segment shape [20]. We formulate the problem as a large-scale linear programming problem. To deal with its dimensionality, we apply column generation approach.

The rest of this thesis is organized as follows: In Chapter 2, we give a literature review of the topic. Available literature is investigated and analysed from the view of studied constraints and MLC types. In Chapter 3, we construct a large-scale linear programming formulation of the minimum BOT problem. To deal with its dimensionality, we revise the formulation in Section 3.2 and apply column generation approach. In Section 3.3, subproblem types for different collimator technologies are explained. We present the results of our method on clinical data in Chapter 4. Finally we provide concluding remarks and discuss future research directions in Chapter 5.

2. LITERATURE REVIEW

There are many studies done in the field of IMRT. Since we focus on the static approach and leaf sequencing stage of the IMRT, our literature review focuses on these topics. Additionally, this study aims to compare the effects of the constraints on the treatment performance of different collimator systems. Hence, the literature review is done in two sections; which are constraints and collimator systems studied in the literature.

2.1. Constraints

We find several constraints during the literature survey. In the following sections, we explain those constraints and summarize the related literature to see how they are handled in the studies.

2.1.1. Consecutiveness Constraint

Consecutiveness constraint is an important restriction that applies to many collimator systems. It means that the open area constructed by the pairs of leaves should be contiguous. Since LSO problem is equivalent to the intensity matrix decomposition problem, the consecutive ones property forces each decomposed bixel row of the intensity matrix to have consecutive exposed bixels [1, 13].

To visualize the appearance of a shape matrix under consecutiveness constraint, let us give an example matrix M_1 as follows: $M_1 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$. At the first row of M_1 first two bixels are blocked by the left leaf. At the second row, right leaf blocks the third bixel and at the last row, it blocks the second and third bixel. All open bixels are sequenced consecutively and therefore, matrix M_1 satisfies the consecutive ones restriction. If the last row of M_1 were $\begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$, it would break the rule and cannot be

constructed by the collimators under consecutiveness constraint.

Men *et al.* model DAO as a convex optimization problem and solve it with column generation algorithm in [13]. Consecutiveness constraint is taken into account in the subproblem. Based on the fact that under this constraint the problem is decomposed by each bixel row, it is stated that optimal solution under consecutiveness can be found by first finding the optimal leaf setting for each row, then combining these leaf settings to form the optimal aperture shape. They apply a single pass algorithm to find the optimal leaf setting for each row. According to the algorithm, the minimum difference between the sum of the coefficients for all bixels considered so far and the maximum value of these sums encountered so far gives an optimal solution of the subproblem with consecutiveness constraint. In [14], Men *et al.* also consider DAO and column generation method but with a difference of its implementation on the graphics processing unit (GPU) which allows parallel computing with affordable prices. In that paper, they also deal with the consecutiveness constraint by applying single pass algorithm.

Taşkın *et al.* [11] investigate the problem of decomposing a fluence map into row-convex matrices under consecutiveness constraint such that total treatment time is minimized. They propose a bilevel optimization algorithm to solve the problem and use allowable intensity multisets (i.e. a set of intensities used in apertures to cover intensity map) to construct master problem. Tight valid inequalities are generated to cut off violations of the consecutiveness constraint.

Cambazard *et al.* [10] compute the optimal value of minimum beam-on time LSO problem for a single row as the sum of increments of the intensity values. Based on that fact, they formulate single row LSO problem with minimization of number of apertures (i.e. decomposition cardinality) as a shortest path problem. Nodes are represented as possible partitions of intensities of each bixel in a row. For example intensity two can be partitioned as $\{1, 1\}$ related with 2 apertures or as $\{2\}$ related with one aperture. Each node is connected to the all next layer nodes with an arc. The costs of arcs are showing the number of additional weights (apertures) needed to be added. Shortest

path from source to sink gives an optimal solution as expected.

Wake *et al.* [21] construct a mixed integer programming (MIP) formulation for minimizing total treatment time under consecutiveness constraint. Consecutiveness constraint is represented as a linear constraint set. Solving the created MIP formulation takes long time. They generate a step up algorithm to reduce the CPU time. Step up algorithm takes a lower bound on the number of apertures and runs the MIP formulation with that bound. Until it finds a feasible solution, the number of apertures is increased gradually. In this manner, it finds an optimal solution with less computation time.

Baatar *et al.* [22] summarize two mixed integer linear programming models from the literature obeying consecutive ones property with the objective of minimizing total number of used apertures. The first one, unit radiation model, uses the information of exposed bixels and the position of the left and right leaves for each row. The second model, leaf-implicit model, is based on the sum of increments idea of Cambazard *et al.* in [10]. Then, they create a counter model based on the idea of counting the patterns according to their BOT values. Instead of looking for a weighted shape matrix decomposition of the intensity matrix, they seek an unweighted decomposition of the aggregate patterns for each intensity level. The decomposition cardinality of those patterns gives the sum of BOT of the original fluence map again.

2.1.2. Rectangular Constraint

Rectangles are the simplest aperture shapes that can be formed. IMRT can be delivered only using conventional jaws which are devices that can only form rectangular apertures. This restriction is named as the rectangular constraint. Since MLC systems are very costly, there are some studies about how to use simpler devices which use jaws only and can only form rectangular structures in IMRT treatment. Hence, this constraint is worthwhile for investigation [13].

To visualize the appearance of a shape matrix under rectangular constraint, we

give an example matrix M_2 as follows: $M_2 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$. To satisfy the rectangular constraint, there must be only one open rectangular area in the shape matrix. Hence, matrix M_2 satisfies the rectangular restriction.

Taşkın *et al.* study the LSO problem with the minimization of total number of apertures objective under the rectangular constraint in [19]. The problem is formulated as an MIP problem. The formulation is strengthened with the generated valid inequalities and improved upper and lower bounds are yielded with the partitioning approach. In that paper, there is a discussion about applying Benders decomposition approach to the problem. However, it is found to be insufficient in the end. In [12], the same problem is solved to optimality with the combinatorial Benders decomposition approach. The difference between the two methods is that in Benders decomposition approach cuts are based on linear programming duality; in combinatorial Benders decomposition approach they are based on minimal infeasible subsystems.

Men *et al.* [13] also deal with the rectangular constraint. A rectangular aperture can be represented with a couple of first and last row and a couple of leftmost and rightmost columns of a beam. In this situation, if the range of rows is fixed, single pass algorithm explained in Section 2.1.1 can be applied to the pricing problem associated with rectangular constraint. Finally an optimal solution can be found by enumerating all $O(m^2)$ possible collections of consecutive rows and selecting the best one given that m is the number of rows of the intensity matrix.

2.1.3. Interdigitation Constraint

Interdigitation constraint is about the adjacent rows. It states that the left and right leaf pairs of a row cannot overlap with the right and left pairs of the adjacent row respectively. In other words, opposing leaves of adjacent rows cannot overlap [1, 23].

Since interdigitation constraint is hard to understand from the verbal definition,

we visualize the appearance of a shape matrix under interdigitation constraint and give an example matrix M_3 as follows: $M_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$. The left leaf of the first row is located on the leftmost position, call position 0, and right leaf of the second row is at the rightmost position, call position 3; therefore they do not overpass each other. Additionally, the right leaf of the first row and left leaf of the second row are located at position 1 and they do not overpass each other either. So first two rows obey the interdigitation restriction. It can be seen that second and third rows' leaves do not overpass each other. Hence, matrix M_3 satisfies the interdigitation restriction. If we changed the second row of M_3 as $\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$, the new matrix would look like $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$ and it would break the rule between first and second rows and cannot be constructed by the collimators under interdigitation constraint.

Men *et al.* [13] deal with interdigitation and connectedness constraints by creating a shortest path network. Connectedness constraint means that open area of an aperture must be single. In the paper, connectedness is explained as combination of interdigitation and unitedness of the bixel rows with one or more exposed bixels. To handle the constraints, the nodes are defined by each potential leaf setting in each bixel row. Additionally, source and sink nodes are added to represent the top and bottom of the aperture. Arcs are drawn between the feasible leaf setting combinations and from source, to sink nodes. Arc costs are defined as the sum of all coefficients corresponding to exposed bixels. With this construction, a shortest path gives an optimal solution.

Baatar *et al.* [22] revise the counter model explained in Section 2.1.1 to account for interdigitation constraint. To account for the interdigitation, previous decision variable is divided into two; representing left and right leaf positions and necessary restrictions are created with those new variables.

Kalinowski models the minimum BOT problem as the maximum weight of a path

in a graph in [23]. Interdigitation constraint is represented as two linear constraint sets by using the left and right leaf positions of each row. For every bixel, a node is created and arcs and weights is said to be modifiable to account for only interdigitation.

Boland *et al.* [24] formulate intensity matrix decomposition problem with minimum BOT as a nonlinear MIP formulation. Consecutiveness and interdigitation are taken into account in the formulation. They apply column generation approach and solve the subproblem as a shortest path problem. Graph has m layers coming from the m rows of the shape matrix. Nodes show each potential leaf setting in each bixel row; additionally a start and end nodes are added. Arcs are placed to exclude interdigitation and a return arc is added from end node to start. Arc lengths are generated from the subproblem objective. In this way, every cycle in the graph gives one shape matrix to the solution.

2.1.4. Minimum Leaf Separation Constraint

Minimum leaf separation constraint forces a minimum leaf opening in each bixel row. Under this constraint, not every intensity matrix can be decomposed for every leaf opening [2, 25].

In [25], the minimum leaf separation constraint is represented as a linear set of constraint for every row. It is stated that under this constraint the problem of finding if it is possible to decompose a matrix is polynomial. In addition to that, since adjusting a leaf opening of a row cannot affect or be affected from the other rows, optimal solution under minimum leaf separation constraint can be found by first finding the optimal leaf setting for each row, then combining these leaf settings to form the optimal aperture shape as in the consecutiveness constraint.

2.1.5. Maximum Leaf Spread Constraint

Maximum leaf spread constraint forces a maximum distance between the edge of leftmost left and rightmost right leaf [2]. This constraint causes field splitting problem

to arise which is explained in [26]. Due to the maximum leaf spread constraint of the MLC design, a large fluence map cannot be delivered at once and need for splitting it into several sub-matrices arise. However, such an action may cause increased complexity and more beam-on time usage. To deal with that in [26] a close-to-linear time algorithm for solving the field splitting problem for minimizing the maximum beam-on time of the sub-matrices is generated by modelling the field splitting as a shortest path problem with a special layered structure. In that structure, consecutiveness constraint is also considered. In the end, high quality field splitting results are gained.

2.1.6. Tongue and Groove Constraint

Tongue and groove constraint occurs due to the tongue and groove leaf arrangement of the MLCs. This arrangement is a special design for reducing the radiation leakage and minimizing friction between leaves. As it can be seen from Figure 2.1b (adapted from [2]), it may cause underdosage and leakage. Hence, there are several studies about preventing tongue and groove effect [2, 23, 25, 27].

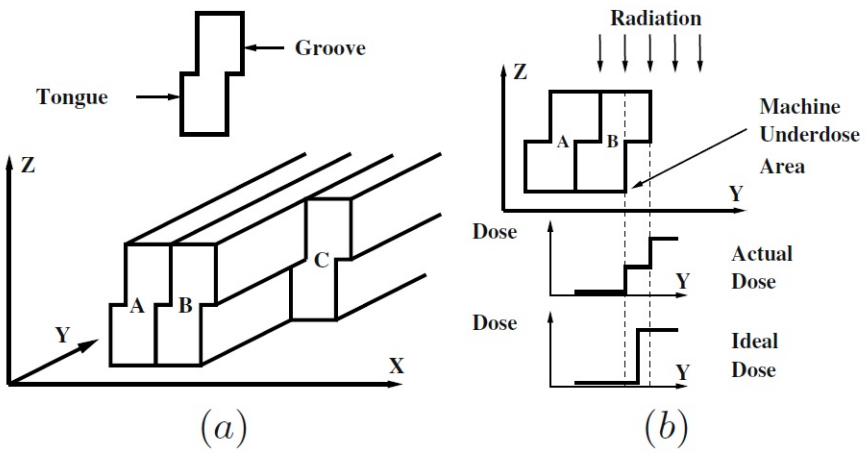

Figure 2.1. (a) The tongue-and-groove design. (b) Underdosage and leakage areas caused by the tongue or groove design.

As explained in Section 2.1.3, Kalinowski models the minimum BOT problem as the maximum weight of a path in a graph in [23]. He checks column based consecutive

bixels to take into account and represent tongue and groove constraint. To obey the tongue and groove constraint; if the bixel with bigger intensity is exposed in a shape matrix, then its column consecutive bixel with smaller intensity must also be exposed. Based on that, he creates tongue and groove error which indicates underdosage and is always nonnegative. Tongue and groove error is minimized if all shape matrices satisfy tongue and groove constraint. As the one for interdigitation, for every bixel, he creates a node and then arcs and weights accordingly to account for tongue and groove.

Salari *et al.* [27] deal with the tongue and groove constraint from a robust optimization point of view. They take the DAO model as used in [13] and add proper upper and lower bounds to the dose deposition coefficients used in that model. Dose deposition coefficient basically shows the dose delivered to a specific voxel by a specific beamlet. With the addition of those bounds, the model becomes robust against the tongue and groove effect. They solve the robust model with column generation algorithm. The subproblem is solved with the same shortest path approach of [24]; hence, the model is extended easily to account for interdigitation and connectedness too.

2.2. Collimator Systems

Different collimator technologies cause changes in the feasible shape matrix set. Therefore, the treatment quality changes depending on the used collimator system. In the following sections, we define different collimator technologies mentioned in the literature and summarize the studies done about them.

2.2.1. Regular MLC

In Section 2.1, the mentioned collimator systems are regular MLCs (i.e. 2-bank MLCs). The regular MLC has a set of leaf pairs; left and right leaves with the same sizes equal to the one bixel size usually. In the regular MLC systems, the linear accelerator rotates around the patient, stops at a predetermined angle and the radiation is transmitted through the aperture constructed by the regular MLC.

LSO problem with minimizing BOT objective is shown to be polynomially solvable in the literature if we use regular MLC without any restrictions (i.e. under consecutiveness constraint only) [28–30]. However, the same problem with the objective of minimizing total setup time or total treatment time is shown to be strongly NP-hard [31]. Therefore, there is a large number of heuristics developed for these problems such as the approaches of Agazaryan *et al.* [32] and Siochi [33]. Exact solution approaches are proposed in Ernst *et al.* [34], Langer *et al.* [35] with minimum number of used apertures objective and in Wake *et al.* [21], Taşkın *et al.* [11], Mason *et al.* [36] with total treatment time objective.

Shape matrix examples M_1, M_2 and M_3 mentioned in Sections 2.1.1, 2.1.2 and 2.1.3 can be constructed by the regular MLC. However, regular MLC cannot construct shape matrix $M_4 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$ because the last row of M_4 cannot be constructed by one left and one right leaf set.

2.2.2. Rotating MLC and Dual MLC

Rotating MLC is a regular MLC with the ability of rotation by 90°. In this system, the apertures are constructed first with left-right pairs of leaves, then the MLC rotates and apertures are constructed with top-bottom pairs of leaves; or vice versa. On the other hand, dual MLC has two orthogonal pairs of leaves. So it can construct more complex apertures.

Example matrices $M_5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$ and $M_6 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ are a row consecutive and a column consecutive shape matrix examples respectively and they can be constructed by using a rotating MLC. To visualize the appearance of the shape matrices constructed by a dual MLC, we give the example matrix $M_7 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$. If we assume that bixel

$(3,1)$ is blocked by a left leaf and bixel $(2,3)$ is blocked by a right leaf, a top leaf is necessary to block the bixel $(1,2)$. Hence, M_7 is neither row nor column consecutive and it cannot be constructed by a regular or rotating MLC.

Blin *et al.* [18] prove the hardness of the decomposition problem of an intensity matrix using rotating collimator and dual MLC. It is shown that they are NP-hard when minimizing total number of apertures and total BOT. To show the hardness of them, they define a reduction from the NP-Complete 3-Hitting Set problem. Then they show that there is an algorithm based on linear programming and rounding techniques which produces an approximate solution for minimizing the total beam-on time.

Webb [17] compares the regular MLC, rotating MLC and dual MLC by using Monte-Carlo simulation. He ignores all constraints except for the consecutiveness. The results show that dual MLC is advantageous over the others because it creates the segments faster, unwanted irradiation may be eliminated in concave figures and less number of apertures and beam-on time is generated with it. Additionally, rotating MLC is turned out to be advantageous over the regular MLC.

2.2.3. Freeform Collimator

To investigate the capabilities of MLCs, Anderson *et al.* compare ten conceptual collimators by using Monte-Carlo simulation in [20]. They used collimators using jaws, leaves, bars or single-bixel attenuators whose shapes can be seen in Figure 2.2 (adapted from [20]). They also create a virtual freeform collimator to provide a basis in the comparisons. Freeform collimator can create any possible segment shape by opening or closing each bixel independently. In this thesis, like this study a freeform collimator is created for comparison purposes.

All shape matrix examples mentioned so far can be constructed by a freeform collimator. However, shape matrix $M_8 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ cannot be constructed by a collimator

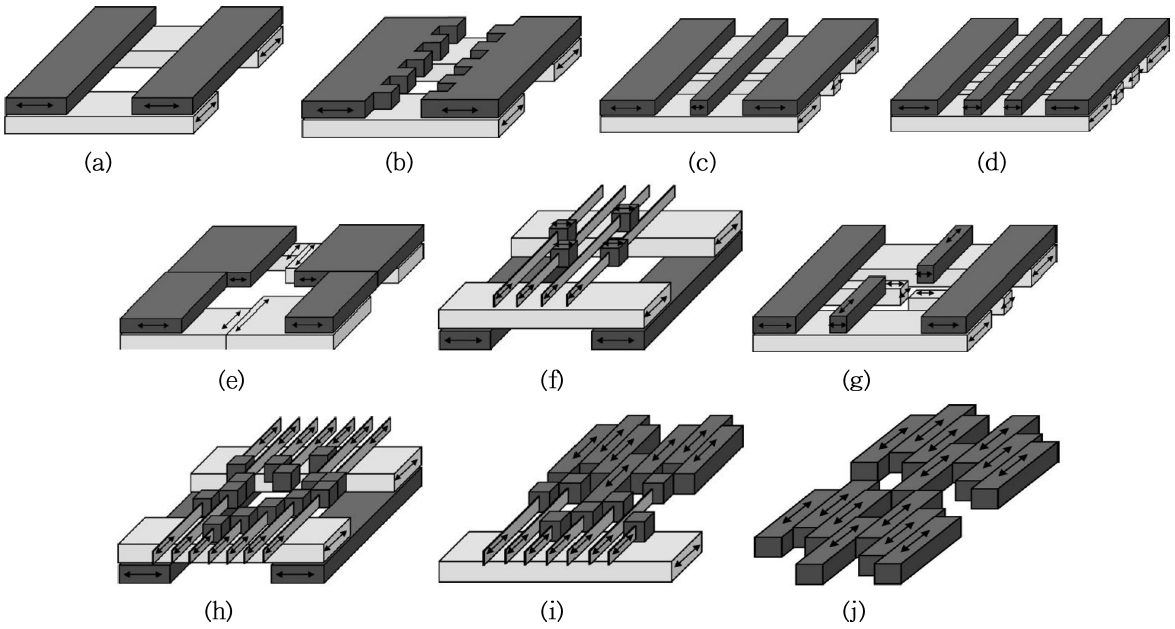

Figure 2.2. (a) Jaws alone. (b) Toothed jaws. (c) Jaws and two bars. (d) Jaws and four bars. (e) Split jaws. (f) Jaws and four floating SBAs. (g) Jaws and four floating leaves. (h) Variable aperture collimator. (i) Leaves and SBAs. (j) Multi-leaf collimator.

other than freeform. Even with a dual MLC it is not possible to close the bixel $(2, 2)$.

3. PROBLEM FORMULATION

In this chapter, a general LSO problem formulation with minimum BOT objective is proposed. To deal with its dimensionality, we apply a column generation approach. We construct a general restricted master problem and subproblem formulation first. Then the individual subproblem formulations of different collimator technologies are investigated. We propose a solution method for each individual subproblem.

3.1. General LSO Problem with Minimum BOT Objective

We propose a general LSO problem formulation with minimum BOT objective. We adapted this formulation from Boland *et al.* [24] with a different notation. This formulation covers all collimator types and constraints. The summary for the definitions of indices, parameters and decision variables used in the formulation is as follows:

Indices are:

- k is the shape matrix index, $k = 1, \ldots, \kappa$
- i is the row index, $i = 1, \ldots, m$
- j is the column index, $j = 1, \ldots, n$

Parameters are:

- κ is the cardinality of the set of all feasible shape matrices
- A is the (mxn) intensity matrix
- a_{ij} is the required intensity value for bixel (i,j), $\forall i = 1, \ldots, m, j = 1, \ldots, n$
- S_{ijk} is $\begin{cases} 1 & \text{if bixel } (i,j) \text{ is exposed in shape matrix } k, \quad \forall i = 1, \ldots, m, j = 1, \ldots, n \\ 0 & \text{otherwise,} \qquad\qquad\qquad\qquad\quad \forall i = 1, \ldots, m, j = 1, \ldots, n \end{cases}$

Decision Variable is:

- X_k is the BOT value for shape matrix k, $\forall k = 1, \ldots, \kappa$

The adapted formulation is as follows:

$$\textbf{(MP:)} \quad \min \sum_{k=1}^{\kappa} X_k \tag{3.1}$$

$$\text{s.t.} \sum_{k=1}^{\kappa} S_{ijk} X_k = a_{ij} \qquad \forall i = 1, \ldots, m, \quad j = 1, \ldots, n \tag{3.2}$$

$$X_k \geq 0 \qquad \forall k = 1, \ldots, \kappa \tag{3.3}$$

In the MP formulation, the objective function (3.1) minimizes the total beam-on time of the shape matrices. The constraint (3.2) assures that the required amount of dosage is delivered to each bixel. Finally, the constraint (3.3) enforces non-negativity restriction of the beam-on time values.

Note that MP is a linear programming problem with just one set of constraints and κ many variables. The number of feasible shapes, κ, varies according to the physical capabilities of the used collimator system. It is a polynomial function of (m,n) in the rectangle case while it is exponential for other cases. To deal with this problem and create a unified method for all technologies, we apply column generation approach.

3.2. Column Generation Approach

The MP formulation is not easy to solve because it typically has vast number of feasible shape matrices. Additionally, many of these apertures will not be used in the solution. In other words, their intensity values will be zero in an optimal solution. Thus, we generate a subset of this set, and use that subset to solve a restricted version of the MP. Then we solve a subproblem to identify new useful shape matrices to add the subset that will improve the current solution or conclude that the current solution is optimal. There are some studies using column generation approach in IMRT context. As explained in Section 2.1.3, Boland *et al.* [24] propose column generation approach to solve MP formulation for the case of using a regular MLC under interdigitation constraint. Furthermore, as explained in Section 2.1.1, Men *et al.* use column generation

algorithm to solve the direct aperture optimization problem in [13] and [14] . We refer the reader to [37] for details about column generation approach.

Let us first define a restricted master problem (RMP) that has only a subset of columns, i.e. shape matrices. If the MP formulation is run over not the entire shape matrix set (i.e. κ many shape matrices) but a subset of shape matrices ($\widehat{\kappa} < \kappa$ many shape matrices), RMP is generated. The restricted master problem can be formulated as:

$$\textbf{(RMP:)} \qquad \min \sum_{k=1}^{\widehat{\kappa}} X_k \tag{3.4}$$

$$\text{s.t.} \sum_{k=1}^{\widehat{\kappa}} S_{ijk} X_k = a_{ij} \qquad \forall i = 1, \ldots, m, \quad j = 1, \ldots, n \tag{3.5}$$

$$X_k \geq 0 \qquad \forall k = 1, \ldots, \widehat{\kappa} \tag{3.6}$$

A feasible subset of shape matrices is needed to start the column generation approach. In principle such a feasible subset can be identified by running a heuristic to obtain a feasible set of shapes. However, this approach would require availability of a heuristic for each collimator type. Therefore, we take a different approach and generate the initial subset as mxn many unit shape matrices (S_{ij} matrices with a single 1 in bixel (i, j) and 0 elsewhere, $\forall i = 1, \ldots, m, j = 1, \ldots, n$). Note that this subset always gives a feasible solution for the collimator types we investigate in this paper and the BOT value of that solution is the sum of the intensities ($\sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij}$).

Let us denote the dual variables associated with the constraints (3.5) by λ_{ij}. Reduced costs of all variables can be found using the optimal values of dual variables. In this way, the columns that price out favourably can be included in the RMP. Hence, the subproblem is defined to identify the shape matrix with the least reduced cost (i.e. the most promising shape matrix). Given an optimal solution of RMP, subproblem

(SP(λ)) can be formulated as follows:

$$\textbf{(SP}(\lambda)\textbf{:)} \qquad \min\ 1 - \sum_{i=1}^{m}\sum_{j=1}^{n} \lambda_{ij} S_{ij} \tag{3.7}$$

$$\text{s.t. S is a feasible shape matrix} \tag{3.8}$$

$$S_{ij} \in \{0,1\} \qquad \forall i = 1,\ldots,m \quad j = 1,\ldots,n \tag{3.9}$$

The SP objective function (3.7) aims to find a shape matrix with the minimum reduced cost. The reduced cost is calculated by the multiplication of the dual variables λ_{ij} associated with the constraints (3.5) and the shape matrix S_{ij}. Note that RMP's structure does not depend on the selected collimator systems and their restrictions of concern. However, each collimator type leads to a different SP(λ) feasible region. Hence, constraint (3.8) means ensuring the feasibility of the shape matrix for the conditions of the investigate SP(λ) case. If optimal objective value of SP(λ) is nonnegative, then the current solution to the RMP optimally solves the original formulation MP as well. If SP(λ)'s optimal objective value is negative, we add the new column to the RMP and solve it again.

3.3. Subproblem Types for Different Collimator Technologies

SP(λ) depends on the used collimator system properties. The aim in the construction and solution of SP(λ) is to find a shape matrix that has the least reduced cost and obeys the used system restrictions. Therefore, in the following sections we explain the SP(λ) of the cases of our concern and discuss their solution procedures.

3.3.1. Regular MLC with Consecutiveness Constraint

To visualize the shape matrices of a feasible decomposition done by regular MLC with consecutiveness constraint, we give an example fluence map below which is decomposed using six regular consecutive apertures, whose intensities are all chosen to

be 1, yielding a total BOT of 6.

$$\begin{bmatrix} 4 & 0 & 2 \\ 3 & 5 & 0 \\ 1 & 2 & 3 \end{bmatrix} = 1 \times \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$
$$+ 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + 1 \times \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

It is known that the LSO problem of regular MLC obeying the consecutiveness constraint with the objective of minimizing the total BOT can be solved in polynomial time [28]. There are various algorithms for solving that problem. For example, the minimum BOT value can be found as $\sum_{i=1}^{m} \sum_{j=0}^{n-1} \max\left(a_{i,(j+1)} - a_{ij}, 0\right)$ as in [10]. Since the aim of this thesis is to construct a common basis for comparison, we investigate the use of column generation method for this case. If we adapt the consecutiveness constraint defined in [21], the subproblem formulation for each row i obeying consecutive ones property can be written as follows:

$$\min \sum_{j=1}^{n} -\lambda_{ij} S_{ij} \tag{3.10}$$

$$\text{s.t. } S_{ij_1} - S_{ij_2} + S_{ij_3} \leq 1 \qquad \forall j_1 = 1, \ldots, n-2, \quad \forall j_3 = j_1 + 2, \ldots, n, \quad \forall j_2 = j_1 + 1, \ldots, j_3 - 1 \tag{3.11}$$

$$S_{ij} \in \{0, 1\} \qquad \forall j = 1, \ldots, n \tag{3.12}$$

Constraint (3.11) represents the consecutiveness constraint. It eliminates a (1 0 1) row construction while keeping the other possibilities. In other words with this constraint, it is impossible to have a closed bixel in the middle of a row, if there are open bixels in both left and right.

Instead of solving that integer programming formulation, we find the optimal leaf setting for each row by a single pass algorithm similar to [13] and [38]. According to the algorithm to find the leaf positions of a single row i, we sum $-\lambda_{ij}$ terms cumulatively over index j. Since our aim is to minimize the cumulative $-\lambda_{ij}$ sum in the SP(λ) objective (3.7), we save the maximum value of the cumulative sum encountered so far. To get rid of those big $-\lambda_{ij}$ terms, we pull the left leaf to the j^{th} index where we obtain the maximum cumulative sum value. In this way, we close the bixels with big $-\lambda_{ij}$ values. Then we compare the maximum with the cumulative sum value. The position that we obtain the minimum difference between these two values gives the optimal right leaf position. The algorithm gives us the optimal leaf positions for a row. We apply this single pass algorithm to all rows, then combine the best leaf settings for each row and obtain an optimal aperture shape. The pseudo code of the revised

Set the sum of the coefficients $v := 0$*, maximum sum of the coefficients* $\bar{v} := 0$*,*
optimal value $v^* := 0$
Set initial position of the left leaf $c_1 := 0$ *and optimal position of the left leaf* $c_1^* := 0$
Set initial position of the left right $c_2 := 0$ *and optimal position of the right leaf*
$c_2^* := 0$
Set $j := 1$
repeat
    Set $v := v + (-\lambda_{ij})$
    if *If* $v \geq \bar{v}$
        $\bar{v} := v$
        $c_1 := c_2$
    if *If* $v - \bar{v} \leq v^*$
        $v^* := v - \bar{v}$
        $c_1^* := c_1$
        $c_2^* := c_2 + 1$
    $c_2 := c_2 + 1$
    $j := j + 1$
until $c_2 \geq n$

Figure 3.1. Pseudo code of the revised single pass algorithm.

single pass algorithm for row $i, \forall i = 1, \ldots, m$ is given in Figure 3.1. At the end of revised single pass algorithm, c_1^* and c_2^* give the optimal aperture for row i and v^* gives the corresponding optimal value. The complexity of this algorithm for a single row is O(n). Since to form an aperture shape we need to run it over all rows, the complexity becomes O(mn).

3.3.2. Regular MLC with Interdigitation Constraint

To visualize the shape matrices of a feasible decomposition done by regular MLC with interdigitation constraint, we give the same fluence map defined in Section 3.3.1 below and decompose it by using five apertures under regular interdigitation constraint, whose intensities are chosen to be 1, 2, 1, 1 and 1 respectively, yielding a total BOT of 6.

$$\begin{bmatrix} 4 & 0 & 2 \\ 3 & 5 & 0 \\ 1 & 2 & 3 \end{bmatrix} = 1 \times \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + 2 \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
$$+ 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} + 1 \times \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

As mentioned in Section 2.1.3, interdigitation constraint is about the adjacent rows; therefore the problem is not separable with respect to rows as the previous case. Therefore, adapting the formulations in [24], we add two decision variables to formulate the problem. Decision variables of the formulation are defined as follows:

$$L_{ij} : \begin{cases} 1 \text{ if left leaf of row } i \text{ is in column } j, & \forall i = 1, \ldots, m, \quad j = 0, \ldots, n \\ 0 \text{ otherwise}, & \forall i = 1, \ldots, m, \quad j = 0, \ldots, n \end{cases}$$

$$R_{ij} : \begin{cases} 1 \text{ if right leaf of row } i \text{ is in column } j, & \forall i = 1, \ldots, m, \quad j = 1, \ldots, n+1 \\ 0 \text{ otherwise}, & \forall i = 1, \ldots, m, \quad j = 1, \ldots, n+1 \end{cases}$$

According to the definition of Boland *et al.* [24], the left and right leaves are positioned as shown in Figure 3.2. Left leaf can be at positions shown with L, and right leaf at R.

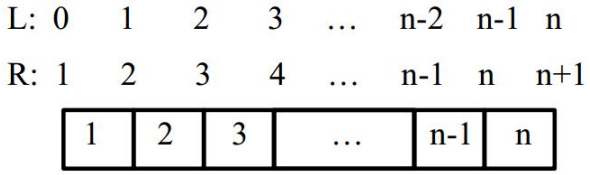

Figure 3.2. Possible leaf positions.

With those variables, the subproblem under interdigitation constraint can be formulated as:

$$\min \sum_{i=1}^{m}\sum_{j=1}^{n} -\lambda_{ij} S_{ij} \tag{3.13}$$

$$\text{s.t. } S_{ij} = \sum_{l=0}^{j-1} L_{il} - \sum_{r=1}^{j} R_{ir} \qquad \forall i = 1,\ldots,m \quad j = 1,\ldots,n \tag{3.14}$$

$$\sum_{j=0}^{n} L_{ij} = 1 \qquad \forall i = 1,\ldots,m \tag{3.15}$$

$$\sum_{j=1}^{n+1} R_{ij} = 1 \qquad \forall i = 1,\ldots,m \tag{3.16}$$

$$\sum_{l=0}^{j-1} L_{il} \geq \sum_{r=1}^{j} R_{i-1,r} \qquad \forall i = 2,\ldots,m \quad j = 1,\ldots,n \tag{3.17}$$

$$\sum_{l=0}^{j-1} L_{il} \geq \sum_{r=1}^{j} R_{i+1,r} \qquad \forall i = 1,\ldots,m-1 \quad j = 1,\ldots,n \tag{3.18}$$

$$S_{ij} \in \{0,1\} \qquad \forall i = 1,\ldots,m \quad j = 1,\ldots,n \tag{3.19}$$

$$L_{ij} \in \{0,1\} \qquad \forall i = 1,\ldots,m \quad j = 0,\ldots,n \tag{3.20}$$

$$R_{ij} \in \{0,1\} \qquad \forall i = 1,\ldots,m \quad j = 1,\ldots,n+1 \tag{3.21}$$

Constraint (3.14) is for binding the three decision variables to each other. It ensures that if bixel (i,j) is closed by one of the leaves, then it is closed in the aperture and if it is not closed in both of them, it is exposed in the aperture. Constraints (3.15) and (3.16) are required due to the fact that there must be one left and right leaf on

each row. The constraints (3.17) and (3.18) are the interdigitation constraints. Finally, constraints (3.41), (3.20) and (3.21) state the binary decision variables.

To solve that SP(λ), as it is done in [13] and [24], we define a shortest path network. The nodes are defined by each potential leaf setting in each bixel row and represented by (i, c_1, c_2) given that i is the row index ($i = 1, \ldots, m$), c_1 and c_2 are the potential left and right leaf positions ($c_1 = 0, \ldots, n$, $c_2 = 1, \ldots, n+1$). In addition to those, we add a source and a sink node to represent the top and bottom of the aperture. We draw arcs between the feasible leaf setting combinations. To satisfy interdigitation constraint, we define an arc between nodes (i, c_1, c_2) and ($i+1$, c_1', c_2') if and only if $c_1' \leq c_2 - 1$ and $c_2' \geq c_1 + 1$. We set arc costs as the sum of all objective function coefficients corresponding to the exposed bixels ($-\sum_{j=c_1+1}^{c_2-1} \lambda_{ij}$). With this construction, a shortest path from source to sink node gives a shape matrix with the least reduced cost.

In each layer of the network, there are $\left[\frac{(n+1)(n+2)}{2}\right]$ many nodes. The total number of nodes is $\left[\frac{m(n+1)(n+2)}{2} + 2\right]$. The total number of arcs is $O(m^2n^4)$. Then, complexity of the algorithm is $O(m^2n^4)$.

In Figure 3.3, a complete shape matrix graph of a 2x2 matrix is given. Arcs are drawn between the feasible leaf setting combinations of the interdigitation constraint. In Figure 3.4, two shape matrix representations on the complete graph of Figure 3.3 are given. So here $m = 2$, $n = 2$, $i = 1, 2$, $c_1 = 0, 1, 2$ and $c_2 = 1, 2, 3$. The first path shown with straight arrows represents the following shape matrix: $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$; the second path shown with dotted arrows represents the following shape matrix: $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$.

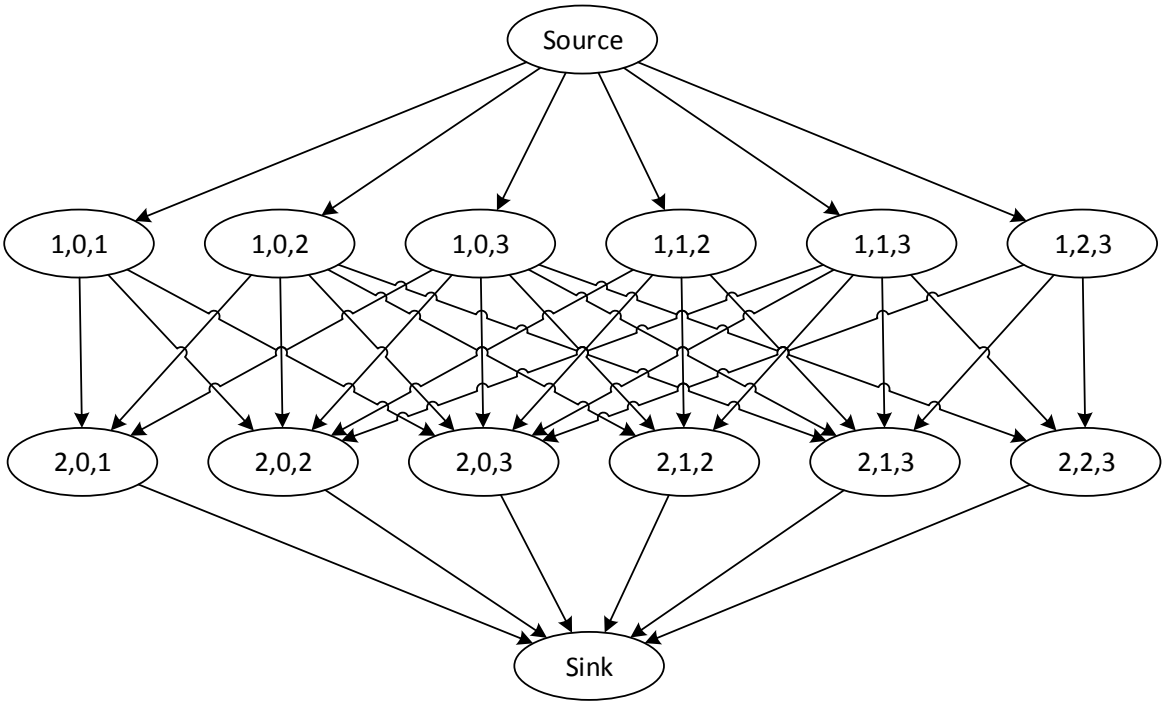

Figure 3.3. The complete shape matrix graph of a 2x2 matrix.

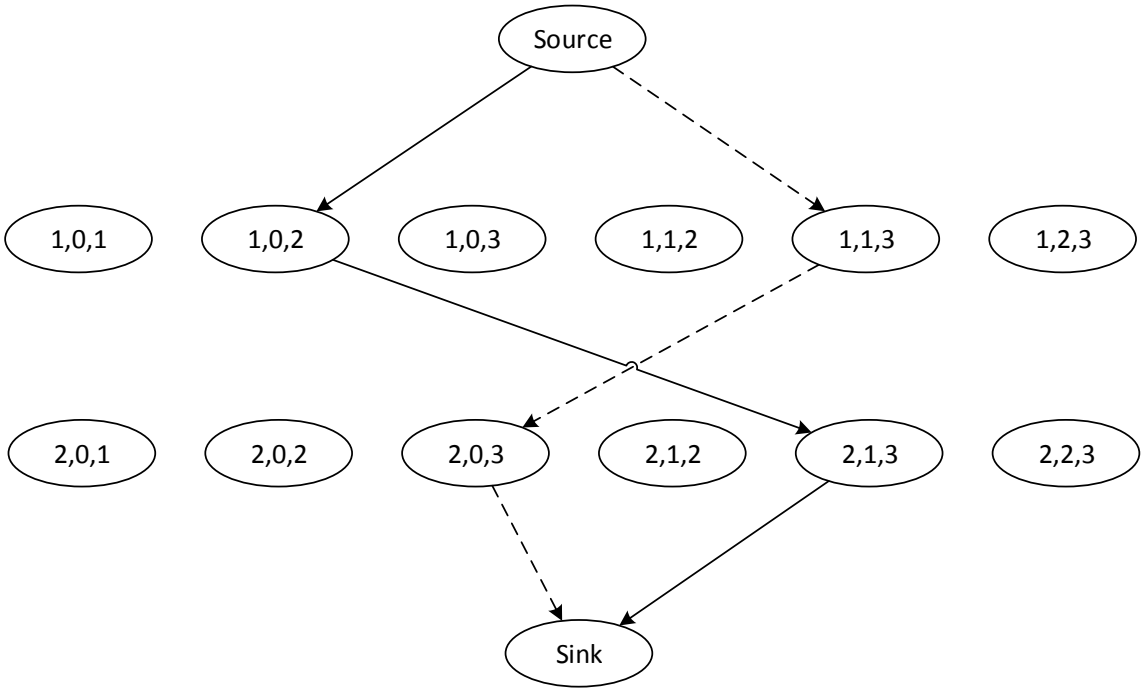

Figure 3.4. Two shape matrix representations on the shape matrix graph.

3.3.3. Regular MLC with Rectangular Constraint

To visualize the shape matrices of a feasible decomposition done by regular MLC with rectangular constraint, we give the same fluence map defined in Section 3.3.1 below and decompose it by using seven rectangular apertures, whose intensities are chosen to be 4, 2, 2, 2, 1, 1 and 2 respectively, yielding a total BOT of 14.

$$\begin{bmatrix} 4 & 0 & 2 \\ 3 & 5 & 0 \\ 1 & 2 & 3 \end{bmatrix} = 4 \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + 2 \times \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + 2 \times \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$
$$+ 2 \times \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} + 1 \times \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} + 1 \times \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix} + 2 \times \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The subproblem under rectangular constraint can be formulated by using modified versions of the consecutiveness constraint (3.11). We formulate SP(λ) of regular MLC under rectangular constraint as follows:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} -\lambda_{ij} S_{ij} \tag{3.22}$$

$$\text{s.t. } S_{ij_1} - S_{ij_2} + S_{ij_3} \leq 1 \qquad \forall i = 1, \ldots, m$$
$$\forall j_1 = 1, \ldots, n-2$$
$$\forall j_3 = j_1 + 2, \ldots, n$$
$$\forall j_2 = j_1 + 1, \ldots, j_3 - 1 \tag{3.23}$$

$$S_{i_1 j} - S_{i_2 j} + S_{i_3 j} \leq 1 \qquad \forall j = 1, \ldots, n$$
$$\forall i_1 = 1, \ldots, m-2$$
$$\forall i_3 = i_1 + 2, \ldots, m$$
$$\forall i_2 = i_1 + 1, \ldots, i_3 - 1 \tag{3.24}$$

$$S_{(i-\sigma)j} - S_{ij} + S_{i(j-\theta)} \leq 1 \qquad \forall i = 2, \ldots, m \quad \forall \sigma = 1, \ldots, i-1 \quad \forall j = 2, \ldots, n \quad \forall \theta = 1, \ldots, j-1 \tag{3.25}$$

$$S_{(i+\sigma)j} - S_{ij} + S_{i(j-\theta)} \leq 1 \qquad \forall i = 1, \ldots, m-1 \quad \forall \sigma = 1, \ldots, m-i \quad \forall j = 2, \ldots, n \quad \forall \theta = 1, \ldots, j-1 \tag{3.26}$$

$$S_{(i-\sigma)j} - S_{ij} + S_{i(j+\theta)} \leq 1 \qquad \forall i = 2, \ldots, m \quad \forall \sigma = 1, \ldots, i-1 \quad \forall j = 1, \ldots, n-1 \quad \forall \theta = 1, \ldots, n-j \tag{3.27}$$

$$S_{(i+\sigma)j} - S_{ij} + S_{i(j+\theta)} \leq 1 \qquad \forall i = 1, \ldots, m-1 \quad \forall \sigma = 1, \ldots, m-i \quad \forall j = 1, \ldots, n-1 \quad \forall \theta = 1, \ldots, n-j \tag{3.28}$$

$$S_{ij} \in \{0, 1\} \qquad \forall i = 1, \ldots, m \quad \forall j = 1, \ldots, n \tag{3.29}$$

Constraints (3.23) and (3.24) ensure the row and column consecutiveness respectively. Constraint (3.25) checks that if bixels $((i-\sigma), j)$ and $(i, (j-\theta))$ are open, then bixel (i, j) must also be open. In this manner all different directions to form a rectangle are checked via constraints (3.25), (3.26), (3.27) and (3.28) and it is assured that there is one open rectangle in the shape matrix.

To form a rectangle, two row selections (r_1, r_2) and two column selections (c_1,

c_2) need to be made among possible m and n options. Suppose that which rows are exposed in the rectangle aperture is known in advance, then the SP(λ) under rectangular constraint becomes equivalent to the SP(λ) under consecutiveness as in Section 3.3.1. There are O(m^2) possible row selections since only two row selection among m is enough to construct a rectangle as explained in [13]. Hence, a rectangular aperture can be formed by applying a slightly modified single pass algorithm for all O(m^2) possible row selections and selecting the minimum one. Modification in the algorithm is done at step 1; its new version is setting $v = v + \sum_{i=r_1}^{r_2} -\lambda_{ij}$. This version of single pass algorithm has complexity of O($m^2 n$). However, this can be adjusted by checking $\min\{m, n\}$. Since a rectangle is consecutive in terms of rows and columns, if $n < m$, then the single pass algorithm for all O(n^2) possible column selections is applied. With this point of view, the complexity of the applied solution methodology is modified as $\mathrm{O}\big(\min\{m,n\}^2 \max\{m,n\}\big)$.

Note that rotating a rectangular aperture yields another rectangular aperture. Hence, we do not investigate rotating MLC with rectangular constraint.

3.3.4. Rotating MLC with Consecutiveness Constraint

To visualize the shape matrices of a feasible decomposition done by rotating MLC with consecutiveness constraint, we give the same fluence map defined in Section 3.3.1 below and decompose it by using four row or column consecutive apertures, whose intensities are chosen to be 1, 1, 2 and 1 respectively, yielding a total BOT of 5.

$$\begin{bmatrix} 4 & 0 & 2 \\ 3 & 5 & 0 \\ 1 & 2 & 3 \end{bmatrix} = 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} + 2 \times \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + 1 \times \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

In rotating MLC, either a row- or column-consecutive shape matrix is constructed during the treatment. Hence, either row- or column-consecutiveness constraint must be active in the feasible region of the SP(λ). To handle that situation, we generate a binary variable y such that $y = 1$ if a column consecutive shape matrix is used and

$y = 0$ otherwise. The subproblem is formulated as:

$$\min \sum_{i=1}^{m}\sum_{j=1}^{n} -\lambda_{ij} S_{ij} \tag{3.30}$$

$$\text{s.t. } S_{ij_1} - S_{ij_2} + S_{ij_3} - y \leq 1 \qquad \forall i = 1, \ldots, m$$
$$\forall j_1 = 1, \ldots, n-2,$$
$$\forall j_3 = j_1 + 2, \ldots, n,$$
$$\forall j_2 = j_1 + 1, \ldots, j_3 - 1 \tag{3.31}$$

$$S_{i_1 j} - S_{i_2 j} + S_{i_3 j} - (1 - y) \leq 1 \qquad \forall j = 1, \ldots, n$$
$$\forall i_1 = 1, \ldots, m-2,$$
$$\forall i_3 = i_1 + 2, \ldots, m,$$
$$\forall i_2 = i_1 + 1, \ldots, i_3 - 1 \tag{3.32}$$

$$S_{ij} \in \{0, 1\} \qquad \forall i = 1, \ldots, m$$
$$\forall j = 1, \ldots, n \tag{3.33}$$

$$y \in \{0, 1\} \tag{3.34}$$

Constraint (3.31) (resp. constraint (3.32)) is active if a row-(resp. column) consecutive shape matrix is constructed during the treatment. Since either a row- or column-consecutive shape matrix is constructed during the treatment with rotating MLC, either constraint (3.31) or (3.32) must be active at each iteration. If $y = 1$, constraint (3.32) is active, but constraint (3.31) is not active since $S_{ij_1} = 1$, $S_{ij_2} = 0$ and $S_{ij_3} = 1$ construction is possible. Likewise, if $y = 0$, constraint (3.31) is active; but since the maximum value that $S_{i_1 j} - S_{i_2 j} + S_{i_3 j} - 1$ can take is 1, constraint (3.32) is not active.

To solve that problem, we apply our single pass algorithm twice; once to find a row-consecutive shape matrix as in Section 3.3.1 and once to find a column-consecutive shape matrix. We then compare the two shape matrices and select the one with the smaller objective value as the optimal solution of SP(λ).

To find a column consecutive shape matrix, the single pass algorithm is modified

again. This time, single pass algorithm for column j is constructed basically and at step 0, i is set to 0 and at step 3, i is incremented with c_2. Therefore, finally, solving the subproblem with consecutiveness in rotating MLC is equivalent to finding the minimum of regular single pass algorithm for all rows and modified single pass algorithm for all columns. Hence, the complexity of the applied solution procedure becomes O($2mn$).

3.3.5. Rotating MLC with Interdigitation Constraint

To visualize the shape matrices of a feasible decomposition done by rotating MLC with consecutiveness constraint, we give the same fluence map defined in Section 3.3.1 below and decompose it by using four apertures under row or column interdigitation restriction, whose intensities are chosen to be 1, 2, 1 and 1 respectively, yielding a total BOT of 5.

$$\begin{bmatrix} 4 & 0 & 2 \\ 3 & 5 & 0 \\ 1 & 2 & 3 \end{bmatrix} = 1 \times \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} + 2 \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Likewise in Section 3.3.4, we apply interdigitation either row based or column based in rotating MLC. Hence, the integer subproblem formulation of the problem is done in 2 ways. First we apply the row based interdigitation formulation as in Section 3.3.2. Then the column based interdigitation formulation is applied and the minimum objective value among these two will determine an optimal solution to the subproblem of rotating MLC with interdigitation constraint.

In the formulation of the column based subproblem under interdigitation, two new variables are created as follows:

$$T_{ij} : \begin{cases} 1 \text{ if top leaf of column } j \text{ is in row } i, & \forall i = 0, \ldots, m, \quad \forall j = 1, \ldots, n \\ 0 \text{ otherwise,} & \forall i = 0, \ldots, m, \quad \forall j = 1, \ldots, n \end{cases}$$

$$B_{ij}: \begin{cases} 1 \text{ if bottom leaf of column } j \text{ is in row } i, & \forall i = 1, \ldots, m+1, \quad \forall j = 1, \ldots, n \\ 0 \text{ otherwise}, & \forall i = 1, \ldots, m+1, \quad \forall j = 1, \ldots, n \end{cases}$$

We formulate the column based subproblem under interdigitation as:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} -\lambda_{ij} S_{ij} \tag{3.35}$$

$$\text{s.t. } S_{ij} = \sum_{t=0}^{i-1} T_{tj} - \sum_{b=1}^{i} B_{bj} \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.36}$$

$$\sum_{i=0}^{m} T_{ij} = 1 \qquad \forall j = 1, \ldots, n \tag{3.37}$$

$$\sum_{i=1}^{m+1} B_{ij} = 1 \qquad \forall j = 1, \ldots, n \tag{3.38}$$

$$\sum_{t=0}^{i-1} T_{tj} \geq \sum_{b=1}^{i} B_{b,j-1} \qquad \forall i = 1, \ldots, m \qquad \forall j = 2, \ldots, n \tag{3.39}$$

$$\sum_{t=0}^{i-1} T_{tj} \geq \sum_{b=1}^{i} B_{b,j+1} \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n-1 \tag{3.40}$$

$$S_{ij} \in \{0, 1\} \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.41}$$

$$T_{ij} \in \{0, 1\} \qquad \forall i = 0, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.42}$$

$$B_{ij} \in \{0, 1\} \qquad \forall i = 1, \ldots, m+1 \qquad \forall j = 1, \ldots, n \tag{3.43}$$

In this formulation, the meanings of the constraints are the same with the constraints (3.14 – 3.21) defined in Section 3.3.1. To solve the column based subproblem, as it is done in Section 3.3.2, a shortest path network is defined. The nodes are defined by each potential leaf setting in each bixel column and represented by (j, r_1, r_2) given that j is the column index ($j = 1, \ldots, n$), r_1 and r_2 are the potential top and bottom leaf positions ($r_1 = 0, \ldots, m$, $r_2 = 1, \ldots, m+1$). In addition to those, source and sink nodes are added to represent the leftmost and rightmost of the aperture. Arcs are drawn between the feasible leaf setting combinations. To satisfy interdigitation constraint, an arc is defined between nodes (j, r_1, r_2) and ($j+1$, r_1', r_2') if and only

if $r_1' \leq r_2 - 1$ and $r_2' \geq r_1 + 1$. Arc costs are defined as the sum of all objective function coefficients corresponding to the exposed bixels ($-\sum_{i=r_1+1}^{r_2-1} \lambda_{ij}$). With this construction, a shortest path from source to sink node gives the shape matrix under column interdigitation constraint with the least reduced cost.

We solve the shortest path problem twice; once over columns and once over rows. We generate two shape matrices in this manner and the one with the minimum objective value among these two determines an optimal solution to the SP(λ). The complexity of this algorithm is $O(m^2n^4 + n^2m^4)$, which is $O(\max\{m^2n^4, n^2m^4\})$. m^2n^4 term comes from the row based interdigitation part of the algorithm as in Section 3.3.2 and n^2m^4 is adapted for the column based interdigitation part.

3.3.6. Dual MLC with Consecutiveness Constraint

To visualize the shape matrices of a feasible decomposition done by dual MLC, we give the same fluence map defined in Section 3.3.1 below and decompose it by using five apertures, whose intensities are all chosen to 1, yielding a total BOT of 5.

$$\begin{bmatrix} 4 & 0 & 2 \\ 3 & 5 & 0 \\ 1 & 2 & 3 \end{bmatrix} = 1 \times \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + 1 \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

In dual MLC, there are two separate orthogonal layers. We call the left-right leaf setting as the horizontal layer and top-bottom leaf setting as the vertical layer. Note that LSO problem of the dual MLC is NP-hard when minimizing BOT [18]. Hence, we formulate the SP(λ) corresponding to dual MLC as an integer programming problem.

Since an aperture is formed by combination of the two layers, they must be represented in the formulation and the shape matrix must be constructed from their intersection. Hence, the formulation is constructed by the combinations of formulations in Sections 3.3.2 and 3.3.5 as follows:

$$\min \sum_{i=1}^{m}\sum_{j=1}^{n} -\lambda_{ij} S_{ij} \tag{3.44}$$

$$\text{s.t. } S_{ij} \geq \sum_{l=0}^{j-1} L_{il} - \sum_{r=1}^{j} R_{ir} + \sum_{t=0}^{i-1} T_{tj} - \sum_{b=1}^{i} B_{bj} - 1 \qquad \forall i = 1, \ldots, m$$

$$\forall j = 1, \ldots, n \tag{3.45}$$

$$S_{ij} \leq \sum_{l=0}^{j-1} L_{il} - \sum_{r=1}^{j} R_{ir} \qquad \forall i = 1, \ldots, m$$

$$\forall j = 1, \ldots, n \tag{3.46}$$

$$S_{ij} \leq \sum_{t=0}^{i-1} T_{tj} - \sum_{b=1}^{i} B_{bj} \qquad \forall i = 1, \ldots, m$$

$$\forall j = 1, \ldots, n \tag{3.47}$$

$$\sum_{j=0}^{n} L_{ij} = 1 \qquad \forall i = 1, \ldots, m \tag{3.48}$$

$$\sum_{j=1}^{n+1} R_{ij} = 1 \qquad \forall i = 1, \ldots, m \tag{3.49}$$

$$\sum_{i=0}^{m} T_{ij} = 1 \qquad \forall j = 1, \ldots, n \tag{3.50}$$

$$\sum_{i=1}^{m+1} B_{ij} = 1 \qquad \forall j = 1, \ldots, n \tag{3.51}$$

$$S_{ij} \in \{0, 1\} \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.52}$$

$$L_{ij} \in \{0, 1\} \qquad \forall i = 1, \ldots, m \qquad \forall j = 0, \ldots, n \tag{3.53}$$

$$R_{ij} \in \{0, 1\} \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n+1 \tag{3.54}$$

$$T_{ij} \in \{0, 1\} \qquad \forall i = 0, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.55}$$

$$B_{ij} \in \{0, 1\} \qquad \forall i = 1, \ldots, m+1 \qquad \forall j = 1, \ldots, n \tag{3.56}$$

The constraint (3.45) is in the formulation to make sure that the cell (i, j) constructed by horizontal and vertical layers is exposed if and only if it is open in both of the horizontal and vertical layers. The constraints (3.46) and (3.47) assure that if the cell (i, j) is blocked in one of the layers, then it must be blocked in the shape matrix. Constraints (3.48), (3.49), (3.50) and (3.51) ensure that there is only

one left and right leaf in each row of horizontal layer, and there is only one top and bottom leaf in each column of vertical layer. Constraints (3.52 – 3.56) represent the non-negativity restrictions of the variables. This construction of formulation already takes into account the consecutiveness restriction applied to the layers due to the fact that the leaves are represented with different variables and each row and each column can only have one leaf from each options of left, right, top and bottom.

Since the constraints (3.45 – 3.51) include summations, these constraints become denser as i and j get bigger. Since sparse constraints are preferred from a computations point of view, we reformulate the problem. Redefined decision variables are equal to 1 if bixel (i, j) is covered by the corresponding leaf and 0 otherwise, $\forall i = 1, \ldots, m, j = 1, \ldots, n$. L'_{ij} represents left leaf, R'_{ij} right leaf, T'_{ij} top leaf and B'_{ij} bottom leaf. We also add two more decision variables to represent whether bixel (i, j) is exposed vertically, V_{ij}, or horizontally, H_{ij}, $\forall i = 1, \ldots, m, j = 1, \ldots, n$. We reformulate the problem as:

$$\min \sum_{i=1}^{m}\sum_{j=1}^{n} -\lambda_{ij} S_{ij} \tag{3.57}$$

$$\text{s.t. } S_{ij} \geq V_{ij} + H_{ij} - 1 \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.58}$$

$$S_{ij} \leq V_{ij} \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.59}$$

$$S_{ij} \leq H_{ij} \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.60}$$

$$L'_{ij} + R'_{ij} + H_{ij} = 1 \qquad \forall i = 1 \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.61}$$

$$L'_{ij} \leq L'_{i(j-1)} \qquad \forall i = 1, \ldots, m \qquad \forall j = 2, \ldots, n \tag{3.62}$$

$$R'_{ij} \leq R'_{i(j+1)} \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n-1 \tag{3.63}$$

$$T'_{ij} + B'_{ij} + V_{ij} = 1 \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.64}$$

$$T'_{ij} \leq T'_{(i-1)j} \qquad \forall i = 2, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.65}$$

$$B'_{ij} \leq B'_{(i+1)j} \qquad \forall i = 1, \ldots, m-1 \qquad \forall j = 1, \ldots, n \tag{3.66}$$

$$L'_{ij}, R'_{ij}, T'_{ij}, B'_{ij} \in \{0, 1\} \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.67}$$

$$S'_{ij}, V'_{ij}, H'_{ij} \geq 0 \qquad \forall i = 1, \ldots, m \qquad \forall j = 1, \ldots, n \tag{3.68}$$

The constraint (3.58) ensures that the bixel (i,j) is exposed if it is open in both of the horizontal and vertical layers. The constraints (3.59) and (3.60) ensure that if the bixel (i,j) is blocked in one of the layers, then it must be blocked in the shape matrix. The constraints (3.61) and (3.64) state that bixel (i,j) can either be exposed horizontally (resp. vertically) or covered by left (resp. top) or right (resp. bottom) leaf. The constraints (3.62) and (3.63) enforce that if bixel (i,j) is covered by the left leaf (resp. right leaf), then the left of that bixel (resp. right of that bixel) must also be covered by the same leaf. Constraints (3.65) and (3.66) apply the corresponding restrictions on top and bottom leaves. Finally constraints (3.67) and (3.68) define variable types. S'_{ij}, V'_{ij} and H'_{ij} variables do not need to be defined as binary, since they will take binary values if $L'_{ij}, R'_{ij}, T'_{ij}$ and B'_{ij} variables are binary.

3.3.7. Freeform Collimator

To visualize the shape matrices of a feasible decomposition done by freeform collimator, we give the same fluence map defined in Section 3.3.1 and decompose it. We get the same decomposition solution given in Section 3.3.6 which is by using five apertures whose intensities are all chosen as 1, yielding a total BOT of 5.

Freeform collimator can form any possible aperture shape. Therefore, the subproblem of the freeform collimator has no constraints and can be seen as follows:

$$\min \ 1 - \sum_{i=1}^{m} \sum_{j=1}^{n} \lambda_{ij} S_{ij} \tag{3.69}$$

$$\text{s.t. } S_{ij} \in \{0,1\} \qquad \forall i = 1,\dots,m \qquad \forall j = 1,\dots,n \tag{3.70}$$

As solution approach, we employ a simple logic. If $-\lambda_{ij} < 0$, then $S_{ij} = 1$; meaning that bixel (i,j) is open. Else, $S_{ij} = 0$; meaning that bixel (i,j) is closed. Since any segment shape is possible, this simple algorithm gives an optimal solution to the SP(λ) of freeform collimator.

4. RESULTS

We have implemented our column generation algorithm for all cases, using CPLEX 12.5, running on a Windows 7 PC with a 3.60 GHz CPU and 32 GB RAM. Our test problem instances are gathered from the treatment plans of eleven head-and-neck cancer patients. Each patient is treated using five beam angles. First 25 of this data set is used in various articles such as [19], [12], [36], [11] and [34]. The maximum intensity value for all 55 instances is $L = 20$.

In the following tables, the column "$Name$" shows the instance name. "m" shows the number of rows and "n" the number of columns of the given intensity matrix. "Reg_Cons" represents the results obtained by using regular MLC under consecutiveness constraint (Section 3.3.1), "Reg_Int" represents regular MLC under interdigitation constraint (Section 3.3.2), "$Rect$" collimators that can only form rectangular shapes (Section 3.3.3), "Rot_Cons" rotating MLC under consecutiveness constraint (Section 3.3.4), "Rot_Int" rotating MLC under interdigitation constraint (Section 3.3.5), "$Free$" virtual freeform collimator (Section 3.3.7) and "$Dual$" dual MLC under consecutiveness constraint (Section 3.3.6) respectively.

Table 4.1 provides the CPU time of the algorithm for each collimator system. Our preliminary results indicate that the average CPU time of the dual MLC is 167.60 seconds. On the other hand, its maximum CPU time, 3778.75 seconds, is very high. Therefore, the CPU time of "$Dual$" needs improvement. To determine the bottleneck of the algorithm, we measured the time spent in the MP and time spent in the SP(λ). As it can be seen from the left half of the Table 4.2, algorithm spends most of its time in solving the subproblem. Hence, we make an arrangement in the stopping condition to improve the CPU time of the subproblem. In SP(λ), finding one shape matrix satisfying $r_N = 1 - \sum_{i=1}^{m} \sum_{j=1}^{n} \lambda_{ij} S_{ij} < 0$ is enough to improve RMP. Therefore, instead of solving SP(λ) to optimality, we stop when we find an integer solution and check whether it is the optimal solution or its objective value is less than -1. If the answer is yes, we stop our search in SP(λ) and put that shape matrix to the RMP shape

matrix subset. Furthermore, if the SP(λ) optimal solution which is sent to RMP is found greater than -1, this means that there are no shape matrices left in the feasible SP(λ) region to improve the RMP, we are at the optimum and column generation algorithm stops. After those adjustments, the shortened CPU time of the algorithm is reported at the second half of the Table 4.2. In terms of CPU time, our algorithm performs well after modifications. The longest running time of the code is less than 15 minutes which is reasonable since the dual MLC case is proven to be NP-hard in [18]. Additionally, average running time of the dual MLC is decreased to 60.08 seconds. As a result, we use the new stopping condition for the dual MLC case in the rest of the analyses.

Table 4.3 displays the comparison of the two dual formulations: Constraints (3.44 – 3.56) (i.e. *"First Formulation"*) and constraints (3.57 – 3.67) (i.e. *"Second Formulation"*). Both formulations solve the problem to optimality; therefore their BOT values are the same and not repeated on the table. However, as Table 4.3 indicates, there are differences slightly in terms of total CPU time (i.e. *"CPU Time (sec)"*) and the number of used apertures (i.e. *"Number of Used Apertures"*) and there are considerable differences in terms of CPU time spent in solving MP (i.e. *"MP CPU Time (sec)"*) and SP(λ) (i.e. *"SP CPU Time (sec)"*) and the number of generated apertures in the treatment (i.e. *"Number of Generated Apertures"*). The first formulation performs slightly better than the second one. Therefore, the first formulation is used and reported in the rest of the analyses.

The BOT results for different collimator types can be found at Table 4.4. The test instance characteristics are summarized in the first three columns of the table. According to the results, freeform collimator always yields the minimum BOT value as expected. Therefore, for each case we find the ratio of BOT value to freeform BOT value and call those proportions "τ_{Free}," "τ_{Dual}," "τ_{Reg_Cons}," "τ_{Reg_Int}," "τ_{Rect}," "τ_{Rot_Cons}" and "τ_{Rot_Int}," respectively. The last rows of the Table 4.4, named as "$min\ \tau$," "$avg\ \tau$" and "$max\ \tau$," shows the minimum, average and maximum ratios for the collimator

Table 4.1. CPU times of different collimator technologies.

Name	**Dual**	**Free**	**Reg_Cons**	**Reg_Int**	**Rect**	**Rot_Cons**	**Rot_Int**
case1_beam1	24.20	0.19	0.79	0.85	0.26	0.93	1.74
case1_beam2	7.27	0.16	0.40	0.52	0.19	0.35	0.99
case1_beam3	25.52	0.21	1.01	1.38	0.34	0.80	2.39
case1_beam4	19.64	0.23	1.09	1.82	0.36	0.94	2.34
case1_beam5	8.97	0.16	0.49	0.75	0.22	0.61	1.54
case2_beam1	335.23	0.59	13.47	20.20	1.31	17.27	34.04
case2_beam2	143.39	0.40	6.80	9.49	0.84	6.38	22.89
case2_beam3	114.70	0.40	6.74	13.04	0.75	6.26	17.75
case2_beam4	196.45	0.44	6.41	11.77	0.90	15.11	33.49
case2_beam5	60.83	0.35	4.55	6.96	0.72	2.83	16.43
case3_beam1	600.89	0.66	14.25	35.00	1.82	26.69	112.05
case3_beam2	26.43	0.29	1.72	3.61	0.50	1.56	8.54
case3_beam3	72.86	0.46	6.22	10.98	0.88	5.88	26.95
case3_beam4	61.20	0.41	4.06	7.76	0.83	6.34	19.63
case3_beam5	33.61	0.26	1.32	4.08	0.50	2.33	12.70
case4_beam1	2273.91	0.71	12.54	24.43	1.69	28.26	85.81
case4_beam2	29.90	0.35	1.44	3.38	0.56	3.04	11.86
case4_beam3	192.13	0.58	4.81	8.16	1.05	6.16	15.66
case4_beam4	168.88	0.53	4.89	10.42	1.12	11.57	37.12
case4_beam5	24.29	0.29	1.73	3.49	0.43	1.68	10.07
case5_beam1	87.98	0.24	1.74	2.59	0.35	1.81	5.71
case5_beam2	25.73	0.19	0.99	0.95	0.26	0.64	2.02
case5_beam3	22.58	0.22	0.85	1.57	0.30	1.03	3.17
case5_beam4	38.16	0.23	1.61	2.28	0.35	2.04	4.37
case5_beam5	10.90	0.18	0.55	0.51	0.26	0.60	1.26
case6_beam1	21.08	0.20	1.07	2.07	0.29	1.20	3.74
case6_beam2	4.47	0.14	0.20	0.32	0.15	0.22	0.42
case6_beam3	8.73	0.18	0.45	1.53	0.25	0.59	2.05
case6_beam4	11.13	0.17	0.43	1.17	0.23	0.97	2.37
case6_beam5	3.62	0.14	0.20	0.32	0.16	0.22	0.52
case7_beam1	133.59	0.22	1.32	2.19	0.35	1.48	5.40
case7_beam2	52.90	0.24	53.95	2.01	0.36	1.15	4.74
case7_beam3	44.30	0.26	2.37	3.21	0.41	2.34	5.67
case7_beam4	31.09	0.26	1.84	3.40	0.52	1.57	7.55
case7_beam5	45.54	0.29	2.60	2.37	0.62	1.55	8.87
case8_beam1	56.19	0.23	1.12	2.68	0.38	1.34	4.42
case8_beam2	18.32	0.22	1.41	1.78	0.33	0.78	2.18
case8_beam3	22.35	0.24	0.95	1.81	0.39	0.79	2.82
case8_beam4	20.28	0.22	1.23	1.65	0.35	1.16	3.07
case8_beam5	30.32	0.20	1.40	1.59	0.30	0.97	2.16
case9_beam1	23.02	0.19	0.68	1.05	0.26	0.91	3.27
case9_beam2	10.66	0.17	0.42	0.59	0.22	0.36	1.10
case9_beam3	13.10	0.17	0.65	0.98	0.22	0.50	1.07
case9_beam4	14.14	0.19	1.03	1.35	0.29	0.69	2.96
case9_beam5	11.04	0.18	0.83	0.92	0.25	0.76	2.25
case10_beam1	3778.75	0.74	9.30	16.77	1.28	30.61	62.04
case10_beam2	20.68	0.24	1.14	2.23	0.34	1.20	6.56
case10_beam3	59.31	0.46	3.12	5.10	0.76	2.97	10.22
case10_beam4	53.24	0.43	4.35	11.48	0.86	3.11	21.62
case10_beam5	37.29	0.38	2.16	7.17	0.60	3.84	26.04
case11_beam1	40.64	0.21	0.81	2.96	0.35	1.23	5.32
case11_beam2	9.43	0.18	0.60	2.17	0.37	0.66	2.93
case11_beam3	14.79	0.18	0.52	1.63	0.25	0.48	2.09
case11_beam4	19.57	0.20	1.00	3.36	0.35	0.81	4.45
case11_beam5	2.88	0.16	0.25	0.82	0.21	0.25	0.97
Minimum	2.88	0.14	0.20	0.32	0.15	0.22	0.42
Average	167.60	0.29	3.60	4.96	0.52	3.92	12.72
Maximum	3778.75	0.74	53.95	35.00	1.82	30.61	112.05

Table 4.2. Performance comparison of old and new stopping condition for dual MLC.

	Dual_Old			Dual_New		
Name	**CPU Time (sec)**	**MP CPU Time (sec)**	**SP CPU Time (sec)**	**CPU Time (sec)**	**MP CPU Time (sec)**	**SP CPU Time (sec)**
case1_beam1	24.20	0.70	23.09	23.57	0.83	22.58
case1_beam2	7.27	0.19	6.89	8.65	0.30	8.22
case1_beam3	25.52	0.72	24.36	20.35	0.82	19.37
case1_beam4	19.64	0.99	18.17	22.13	1.12	20.86
case1_beam5	8.97	0.39	8.34	10.54	0.48	9.90
case2_beam1	335.23	20.09	312.67	155.08	18.96	135.81
case2_beam2	143.39	9.66	132.10	103.58	9.22	94.10
case2_beam3	114.70	6.05	107.11	77.81	5.78	71.76
case2_beam4	196.45	6.70	188.68	89.15	5.77	83.14
case2_beam5	60.83	4.10	55.54	55.03	4.13	50.67
case3_beam1	600.89	19.95	578.08	187.35	22.31	164.69
case3_beam2	26.43	1.42	24.34	26.49	1.45	24.84
case3_beam3	72.86	4.13	67.54	61.83	4.94	56.65
case3_beam4	61.20	3.47	56.73	53.08	3.95	48.93
case3_beam5	33.61	1.67	31.16	32.68	1.89	30.59
case4_beam1	2273.91	34.65	2229.64	396.16	34.48	361.10
case4_beam2	29.90	1.61	27.53	30.82	2.03	28.60
case4_beam3	192.13	4.54	185.72	83.43	5.46	77.70
case4_beam4	168.88	4.74	162.57	82.03	4.81	76.97
case4_beam5	24.29	0.92	22.91	26.99	1.24	25.59
case5_beam1	87.98	2.72	84.40	66.57	2.97	63.38
case5_beam2	25.73	0.87	24.37	26.06	1.05	24.84
case5_beam3	22.58	1.11	20.97	21.59	1.10	20.32
case5_beam4	38.16	2.15	35.26	36.57	2.30	34.06
case5_beam5	10.90	0.43	10.18	12.60	0.51	11.93
case6_beam1	21.08	0.95	19.70	19.25	1.01	18.06
case6_beam2	4.47	0.12	4.21	4.83	0.18	4.53
case6_beam3	8.73	0.46	8.00	10.12	0.56	9.40
case6_beam4	11.13	0.40	10.49	8.73	0.49	8.11
case6_beam5	3.62	0.10	3.39	4.10	0.16	3.83
case7_beam1	133.59	2.73	129.57	92.29	3.10	88.97
case7_beam2	52.90	1.92	50.39	48.98	2.36	46.41
case7_beam3	44.30	2.50	41.08	39.66	2.77	36.69
case7_beam4	31.09	2.14	28.31	32.97	2.40	30.39
case7_beam5	45.54	2.22	42.61	38.76	2.33	36.23
case8_beam1	56.19	1.50	54.01	42.47	1.58	40.69
case8_beam2	18.32	0.96	16.86	17.21	0.99	16.06
case8_beam3	22.35	1.31	20.48	20.53	1.33	19.03
case8_beam4	20.28	0.94	18.84	23.07	1.22	21.68
case8_beam5	30.32	1.05	28.75	28.21	1.23	26.80
case9_beam1	23.02	0.76	21.90	23.27	0.96	22.14
case9_beam2	10.66	0.32	10.12	11.08	0.42	10.51
case9_beam3	13.10	0.53	12.24	13.25	0.62	12.47
case9_beam4	14.14	0.68	13.11	17.06	0.88	16.01
case9_beam5	11.04	0.51	10.27	13.12	0.67	12.28
case10_beam1	3778.75	42.31	3729.19	856.28	139.03	716.41
case10_beam2	20.68	0.92	19.26	22.96	1.08	21.71
case10_beam3	59.31	2.53	54.33	50.60	2.57	47.74
case10_beam4	53.24	3.05	48.98	45.48	3.54	41.70
case10_beam5	37.29	1.92	34.39	35.32	1.99	33.13
case11_beam1	40.64	1.39	38.96	31.00	1.54	29.30
case11_beam2	9.43	0.48	8.76	9.52	0.54	8.83
case11_beam3	14.79	0.47	14.13	14.53	0.65	13.73
case11_beam4	19.57	0.75	18.60	16.14	0.79	15.21
case11_beam5	2.88	0.11	2.63	3.73	0.18	3.44
Minimum	2.88	0.10	2.63	3.73	0.16	3.44
Average	167.60	3.82	162.76	60.08	5.73	54.15
Maximum	3778.75	42.31	3729.19	856.28	139.03	716.41

Table 4.3. Comparison of two dual MLC SP(λ) formulations.

	First Formulation					Second Formulation				
Name	**CPU Time (sec)**	**MP CPU Time (sec)**	**SP CPU Time (sec)**	**Number of Generated Apertures**	**Number of Used Apertures**	**CPU Time (sec)**	**MP CPU Time (sec)**	**SP CPU Time (sec)**	**Number of Generated Apertures**	**Number of Used Apertures**
case1_beam1	23.57	0.83	22.58	617	94	14.75	0.76	13.82	643	94
case1_beam2	8.65	0.30	8.22	382	74	5.29	0.20	4.97	383	74
case1_beam3	20.35	0.82	19.37	612	101	15.46	0.71	14.59	622	101
case1_beam4	22.13	1.12	20.86	645	113	16.61	0.94	15.49	649	113
case1_beam5	10.54	0.48	9.90	485	83	6.37	0.35	5.89	460	83
case2_beam1	155.08	18.96	135.81	1427	213	200.18	37.23	162.40	2532	213
case2_beam2	103.58	9.22	94.10	1211	173	76.59	10.07	66.18	1388	173
case2_beam3	77.81	5.78	71.76	1089	163	56.61	5.99	50.35	1174	163
case2_beam4	89.15	5.77	83.14	977	188	77.92	8.69	68.93	1238	188
case2_beam5	55.03	4.13	50.67	954	154	37.30	4.17	32.90	999	154
case3_beam1	187.35	22.31	164.69	1508	219	448.75	196.90	250.94	4233	219
case3_beam2	26.49	1.45	24.84	717	127	17.78	1.46	16.13	748	127
case3_beam3	61.83	4.94	56.65	949	172	47.31	5.95	41.08	1061	172
case3_beam4	53.08	3.95	48.93	874	160	36.77	4.32	32.19	963	160
case3_beam5	32.68	1.89	30.59	804	120	21.85	1.74	19.90	828	120
case4_beam1	396.16	34.48	361.10	2454	206	703.40	308.44	393.44	5260	206
case4_beam2	30.82	2.03	28.60	751	140	22.43	1.87	20.38	773	140
case4_beam3	83.43	5.46	77.70	1142	168	57.56	6.49	50.75	1284	168
case4_beam4	82.03	4.81	76.97	1007	174	69.43	8.04	61.06	1385	174
case4_beam5	26.99	1.24	25.59	632	131	14.49	0.98	13.34	622	131
case5_beam1	66.57	2.97	63.38	916	123	41.00	3.22	37.55	1169	123
case5_beam2	26.06	1.05	24.84	707	96	16.12	0.88	15.06	720	96
case5_beam3	21.59	1.10	20.32	661	107	15.96	1.00	14.80	667	107
case5_beam4	36.57	2.30	34.06	872	120	24.04	2.04	21.81	867	120
case5_beam5	12.60	0.51	11.93	515	81	5.99	0.32	5.54	476	81
case6_beam1	19.25	1.01	18.06	622	105	12.49	0.89	11.44	657	105
case6_beam2	4.83	0.18	4.53	310	53	2.29	0.11	2.07	284	53
case6_beam3	10.12	0.56	9.40	479	97	7.34	0.45	6.76	463	97
case6_beam4	8.73	0.49	8.11	457	92	6.72	0.38	6.20	466	92
case6_beam5	4.10	0.16	3.83	266	59	1.93	0.10	1.72	253	60
case7_beam1	92.29	3.10	88.97	1124	115	62.51	3.55	58.66	1463	115
case7_beam2	48.98	2.36	46.41	850	122	22.96	1.75	21.02	797	122
case7_beam3	39.66	2.77	36.69	861	133	29.85	2.61	27.01	925	133
case7_beam4	32.97	2.40	30.39	799	134	24.20	2.13	21.89	798	134
case7_beam5	38.76	2.33	36.23	806	133	26.72	2.04	24.48	822	133
case8_beam1	42.47	1.58	40.69	732	119	28.00	2.13	25.65	973	119
case8_beam2	17.21	0.99	16.06	621	103	12.60	0.80	11.65	622	103
case8_beam3	20.53	1.33	19.03	666	111	14.96	1.29	13.51	688	111
case8_beam4	23.07	1.22	21.68	682	106	13.51	0.89	12.46	671	106
case8_beam5	28.21	1.23	26.80	709	101	18.50	1.01	17.31	705	101
case9_beam1	23.27	0.96	22.14	609	98	12.06	0.66	11.25	582	98
case9_beam2	11.08	0.42	10.51	458	80	7.66	0.33	7.19	460	80
case9_beam3	13.25	0.62	12.47	559	84	8.62	0.51	7.97	557	84
case9_beam4	17.06	0.88	16.01	599	99	11.87	0.70	11.01	596	99
case9_beam5	13.12	0.67	12.28	536	91	9.39	0.53	8.71	544	91
case10_beam1	856.28	139.03	716.41	3783	217	835.41	235.32	598.64	5373	217
case10_beam2	22.96	1.08	21.71	671	112	15.08	0.90	14.01	665	111
case10_beam3	50.60	2.57	47.74	1083	125	33.23	2.30	30.67	1075	125
case10_beam4	45.48	3.54	41.70	965	155	30.28	3.19	26.86	938	155
case10_beam5	35.32	1.99	33.13	812	139	25.57	1.90	23.45	839	139
case11_beam1	31.00	1.54	29.30	560	126	23.75	2.79	20.77	879	126
case11_beam2	9.52	0.54	8.83	412	106	4.64	0.43	4.09	393	106
case11_beam3	14.53	0.65	13.73	454	104	6.60	0.50	5.96	450	104
case11_beam4	16.14	0.79	15.21	445	122	8.82	0.91	7.77	510	122
case11_beam5	3.73	0.18	3.44	263	77	1.60	0.09	1.40	234	78
Minimum	3.73	0.16	3.44	263	53	1.60	0.09	1.40	234	53
Averages	60.08	5.73	54.15	820.02	123.96	61.26	16.07	44.93	1015.02	123.98
Maximum	856.28	139.03	716.41	3783	219	835.41	308.44	598.64	5373	219

Table 4.4. Beam-on time values and performance ratios of different collimator types.

Name	m	n	Free	τ_{Free}	Dual	τ_{Dual}	Reg_Cons	τ_{Reg_Cons}	Reg_Int	τ_{Reg_Int}	Rect	τ_{Rect}	Rot_Cons	τ_{Rot_Cons}	Rot_Int	τ_{Rot_Int}
case1beam1	15	14	20	1	20.00	1.00	40	2.00	51	2.55	176.00	8.80	25.00	1.25	34.00	1.70
case1beam2	11	15	20	1	20.00	1.00	34	1.70	37	1.85	121.00	6.05	20.00	1.00	21.50	1.08
case1beam3	15	15	20	1	20.00	1.00	31	1.55	36	1.80	147.00	7.35	20.00	1.00	28.00	1.40
case1beam4	15	15	20	1	20.00	1.00	33	1.65	34	1.70	136.00	6.80	24.00	1.20	27.00	1.35
case1beam5	11	15	20	1	20.00	1.00	34	1.70	40	2.00	115.00	5.75	20.00	1.00	25.00	1.25
case2beam1	18	20	20	1	20.00	1.00	34	1.70	40	2.00	194.00	9.70	20.50	1.03	29.00	1.45
case2beam2	17	19	20	1	20.33	1.02	41	2.05	48	2.40	207.00	10.35	20.33	1.02	27.65	1.38
case2beam3	18	18	20	1	20.00	1.00	49	2.45	52	2.60	237.00	11.85	23.00	1.15	32.00	1.60
case2beam4	18	18	20	1	20.00	1.00	51	2.55	52	2.60	258.00	12.90	20.00	1.00	27.88	1.39
case2beam5	17	18	20	1	20.00	1.00	39	1.95	40	2.00	207.00	10.35	21.00	1.05	26.50	1.33
case3beam1	22	17	20	1	20.33	1.02	41	2.05	43	2.15	266.00	13.30	26.50	1.33	27.60	1.38
case3beam2	15	19	20	1	20.00	1.00	46	2.30	49	2.45	151.00	7.55	20.00	1.00	21.43	1.07
case3beam3	20	17	20	1	20.00	1.00	49	2.45	55	2.75	278.00	13.90	27.67	1.38	31.17	1.56
case3beam4	19	17	20	1	20.00	1.00	43	2.15	57	2.85	287.00	14.35	23.00	1.15	28.40	1.42
case3beam5	15	19	20	1	20.00	1.00	34	1.70	51	2.55	182.00	9.10	20.00	1.00	27.78	1.39
case4beam1	19	22	20	1	20.67	1.03	40	2.00	62	3.10	275.00	13.75	23.00	1.15	42.50	2.13
case4beam2	13	24	20	1	20.00	1.00	69	3.45	73	3.65	232.00	11.60	21.50	1.08	27.00	1.35
case4beam3	18	23	20	1	20.00	1.00	39	1.95	40	2.00	189.00	9.45	24.00	1.20	26.00	1.30
case4beam4	17	23	20	1	20.00	1.00	42	2.10	51	2.55	235.00	11.75	20.43	1.02	28.00	1.40
case4beam5	12	24	20	1	20.00	1.00	73	3.65	77	3.85	260.00	13.00	21.50	1.08	27.06	1.35
case5beam1	15	16	20	1	20.00	1.00	26	1.30	41	2.05	158.00	7.90	20.00	1.00	27.50	1.38
case5beam2	13	17	20	1	20.00	1.00	41	2.05	65	3.25	156.00	7.80	20.00	1.00	25.57	1.28
case5beam3	14	16	20	1	20.00	1.00	34	1.70	38	1.90	180.00	9.00	20.00	1.00	25.33	1.27
case5beam4	14	16	20	1	20.00	1.00	40	2.00	42	2.10	145.00	7.25	20.50	1.03	23.33	1.17
case5beam5	12	17	20	1	20.00	1.00	44	2.20	51	2.55	147.00	7.35	20.00	1.00	23.00	1.15
case6beam1	13	15	20	1	20.00	1.00	36	1.80	40	2.00	129.00	6.45	26.00	1.30	27.50	1.38
case6beam2	8	15	20	1	20.00	1.00	33	1.65	45	2.25	94.00	4.70	20.00	1.00	20.00	1.00
case6beam3	10	17	20	1	20.00	1.00	45	2.25	49	2.45	160.00	8.00	23.50	1.18	26.50	1.33
case6beam4	10	16	20	1	20.00	1.00	35	1.75	53	2.65	150.00	7.50	21.33	1.07	28.00	1.40
case6beam5	8	15	20	1	20.00	1.00	35	1.75	40	2.00	88.00	4.40	20.00	1.00	23.00	1.15
case7beam1	16	15	20	1	20.00	1.00	39	1.95	66	3.30	256.00	12.80	25.50	1.28	47.00	2.35
case7beam2	14	17	20	1	20.00	1.00	38	1.90	43	2.15	130.00	6.50	20.00	1.00	22.50	1.13
case7beam3	15	16	20	1	20.00	1.00	48	2.40	49	2.45	161.00	8.05	23.00	1.15	26.00	1.30
case7beam4	15	16	20	1	20.00	1.00	31	1.55	32	1.60	139.33	6.97	20.00	1.00	21.25	1.06
case7beam5	17	17	20	1	20.00	1.00	41	2.05	42	2.10	149.00	7.45	20.00	1.00	23.33	1.17
case8beam1	17	14	20	1	20.00	1.00	30	1.50	34	1.70	173.00	8.65	21.50	1.08	32.00	1.60
case8beam2	15	15	20	1	20.00	1.00	38	1.90	52	2.60	111.00	5.55	20.00	1.00	21.50	1.08
case8beam3	17	14	20	1	20.00	1.00	35	1.75	36	1.80	173.67	8.68	23.50	1.18	26.00	1.30
case8beam4	16	14	20	1	20.00	1.00	33	1.65	39	1.95	144.00	7.20	20.00	1.00	23.50	1.18
case8beam5	15	15	20	1	20.00	1.00	38	1.90	39	1.95	119.00	5.95	21.00	1.05	28.00	1.40
case9beam1	14	13	20	1	20.00	1.00	33	1.65	44	2.20	154.00	7.70	24.50	1.23	26.25	1.31
case9beam2	13	14	20	1	20.00	1.00	35	1.75	51	2.55	135.00	6.75	20.00	1.00	26.00	1.30
case9beam3	14	13	20	1	20.00	1.00	34	1.70	35	1.75	127.00	6.35	20.00	1.00	25.00	1.25
case9beam4	14	14	20	1	20.00	1.00	38	1.90	39	1.95	139.00	6.95	20.00	1.00	23.07	1.15
case9beam5	14	14	20	1	20.00	1.00	42	2.10	47	2.35	134.00	6.70	20.00	1.00	26.00	1.30
case10beam1	17	25	20	1	20.57	1.03	46	2.30	78	3.90	348.00	17.40	25.75	1.29	34.75	1.74
case10beam2	10	26	20	1	20.00	1.00	66	3.30	78	3.90	179.00	8.95	20.67	1.03	29.00	1.45
case10beam3	18	24	20	1	20.00	1.00	45	2.25	47	2.35	184.00	9.20	20.00	1.00	22.00	1.10
case10beam4	15	24	20	1	20.00	1.00	44	2.20	57	2.85	211.00	10.55	20.00	1.00	21.25	1.06
case10beam5	13	27	20	1	20.00	1.00	48	2.40	54	2.70	196.00	9.80	20.00	1.00	23.25	1.16
case11beam1	12	14	20	1	20.00	1.00	39	1.95	39	1.95	168.00	8.40	25.00	1.25	26.50	1.33
case11beam2	12	13	20	1	20.00	1.00	27	1.35	27	1.35	103.50	5.18	20.00	1.00	20.00	1.00
case11beam3	11	13	20	1	20.00	1.00	29	1.45	29	1.45	93.00	4.65	20.00	1.00	20.00	1.00
case11beam4	12	13	20	1	20.00	1.00	27	1.35	27	1.35	106.00	5.30	20.00	1.00	20.00	1.00
case11beam5	9	14	20	1	20.00	1.00	29	1.45	29	1.45	82.00	4.10	20.00	1.00	20.00	1.00
$min\ \ \tau$				1		1.00		1.30		1.35		4.10		1.00		1.00
$avg\ \ \tau$				1		1.00		1.99		2.33		8.61		1.08		1.32
$max\ \ \tau$				1		1.03		3.65		3.90		17.40		1.38		2.35

technologies to the freeform collimator.

Table 4.4 indicates that the virtual freeform collimator yields the smallest BOT results among all collimators and the minimum possible BOT value is $L = 20$ as expected. Furthermore, since rectangles are very simple shapes, it is also not surprizing that the collimators that can only form rectangular shapes have the largest BOT values. On the average, the closest value to minimum BOT found by rectangular collimators is 8.61 times the minimum value found by freeform collimator. However, it is interesting to see that the dual MLC matches the freeform BOT values for 51 out of 55 instances.

It is not easy to evaluate the differences in the efficiency of using different collimator technologies from the Table 4.4. Therefore, to visualize the results, we compare them by using a performance profile chart [39]. We use BOT value of the treatment as performance metric. Figure 4.1 shows $P_s(\tau)$, which is the percentage of times that collimator system s can find BOT values that are within factor τ of the minimum BOT. For example, if we choose $\tau = 2$ from Table 4.4, for "Rot_Int" $P_s(\tau) = 0.96$ and if we choose $\tau = 3$, for "Reg_Int" $P_s(\tau) = 0.87$. This indicates that "Rot_Int" can find BOT values in between 20 and 40 ($= 2 \times 20$) in 96% of the cases and "Reg_Int" can find BOT values in between 20 and 60 ($= 3 \times 20$) in 87% of the cases. $P_s(1)$ shows the percentage of times that collimator system s can find the overall minimum BOT values. From $\tau = 1$ line of the Figure 4.1, we see that all of the minimum BOT values are found by the freeform collimator, which is expected. Dual MLC performs very similar to freeform collimator. Then rotating MLC under consecutiveness constraint follows them in terms of BOT efficiency. It matches the minimum for 28 instances. Rotating MLC under interdigitation constraint matches the minimum for only 5 instances and the rest of the collimator systems could not find the overall minimum BOT value for any of the instances. The worst performance on the BOT values is obtained by the regular MLC under rectangular constraint. As it can be seen from Table 4.4, its minimum deviation from the minimum BOT value is 4.10, which is larger than the maximum deviation of other collimators.

To investigate the mean and median behaviour of the results, we create a box

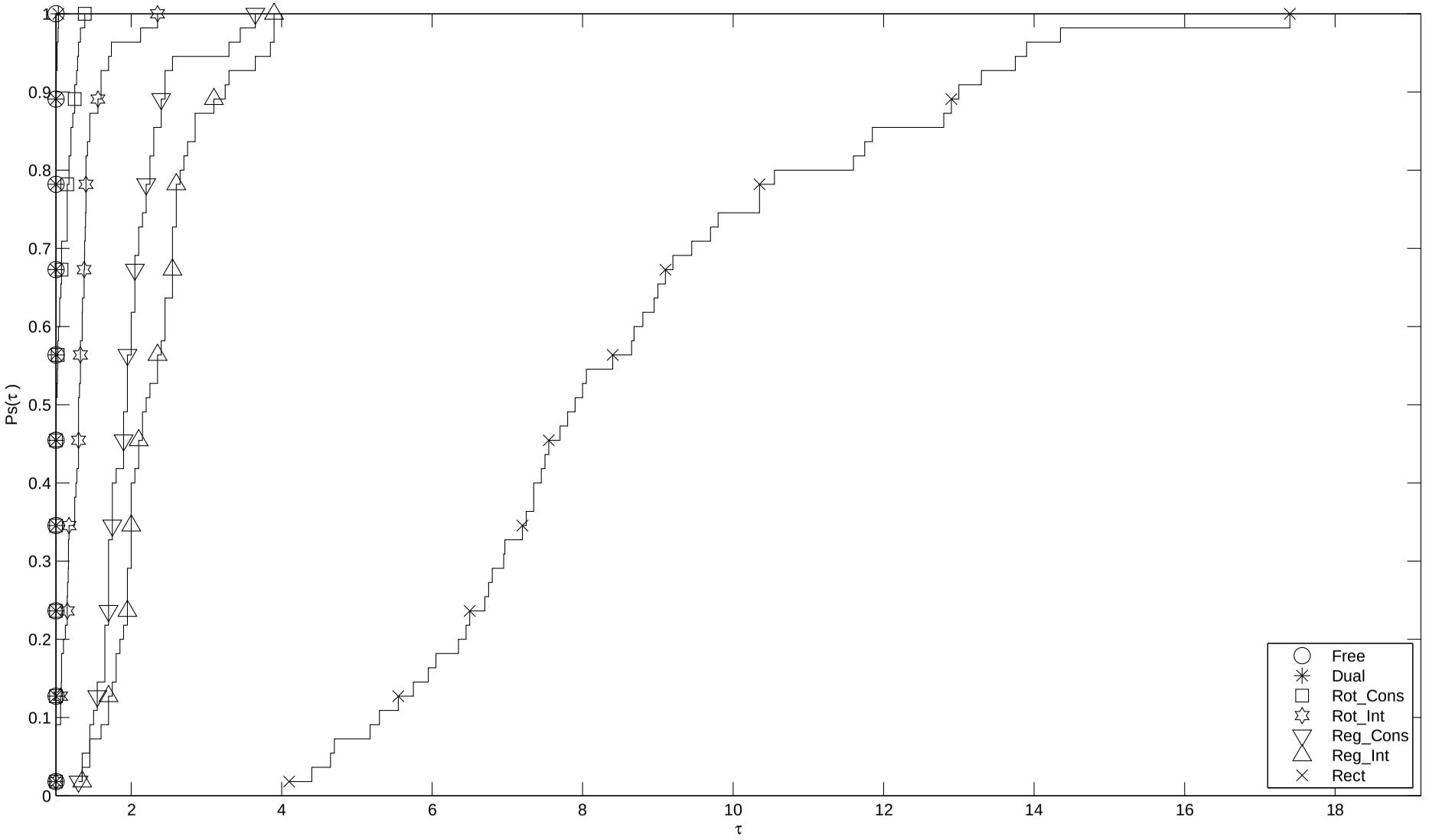

Figure 4.1. The performance profile chart for BOT values used as performance metric.

plot in Figure 4.2. In the boxplot, upper and lower edges of the boxes represent the 25% and 75% quartiles of the data. The line extending vertically from the boxes indicates variability outside the quartiles. The top and the bottom of the lines show the maximum and minimum data value excluding outliers. Outliers are plotted with "x" symbol. If the distance of the data point from the box exceeds the difference between the 25% and 75% quartiles of the data, the point is labelled as outlier. The circles in the boxes show the mean values and the lines show the median values of the instance results.

The dual MLC and freeform collimator have the minimum median and almost the same mean values. Rotating MLC under consecutiveness gives similar results as them, but it has several outliers. The big difference between the results of regular MLC under rectangular constraint and the others is visible in figure 4.2. The minimum value found by the regular MLC under rectangular constraint is bigger than the maximum value found by the others. Additionally, mean value of regular MLC under rectangular constraint is higher than the median value due to the presence of a big outlier (case10beam1, whose BOT value is 348).

LSO problem is evaluated with two criteria in the literature: BOT and the number of apertures. Tables 4.7, 4.8 and 4.9 provides the generated and used aperture numbers in the solutions of the test instances. It should be noted that we do not consider obtaining less aperture number criteria in our formulations or algorithms. We need to find a comparison basis to understand the position of our solutions compared to the optimum number of aperture results. Therefore, we search the papers which use the same data instances that we use. First 25 of the instances are used in Taşkın *et al.* (2010) [11] and Taşkın *et al.* (2012) [19]. Table 4.5 and 4.6 gives the number of apertures comparison of the regular MLC under rectangular and consecutiveness constraints with the results of Taşkın *et al.* (2012) [19] and Taşkın *et al.* (2010) [11] respectively. The number of used apertures results of Taşkın et al. (2012) [19] and Taşkın et al. (2010) [11] are obtained from a problem with lexicographic objective. They minimize the number of used apertures such that the BOT value is less than the minimum BOT. The column *"GAP"* shows the difference between the two results and

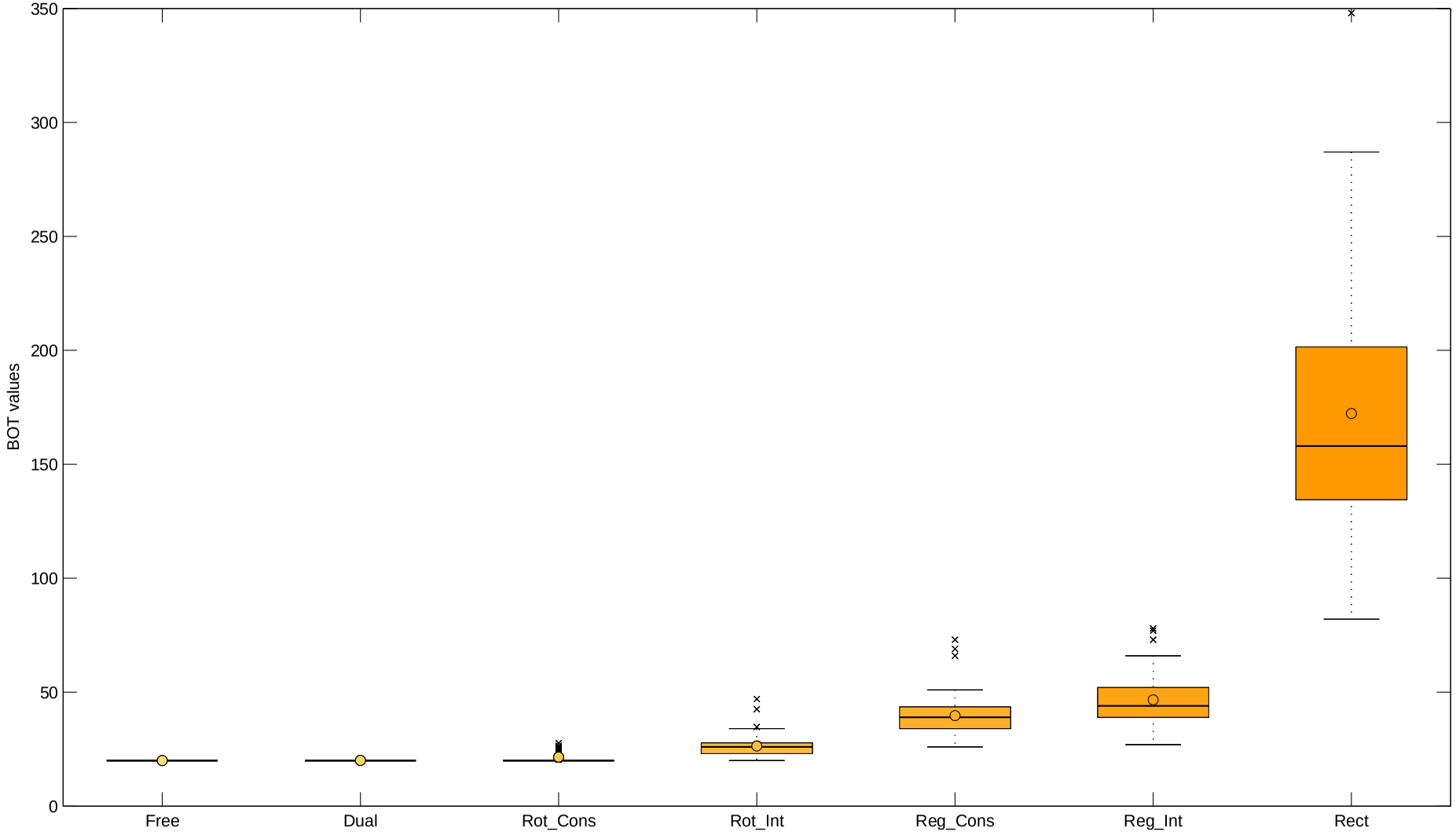

Figure 4.2. The Box Plot for BOT results of different collimator technologies.

is calculated with the formulation of *(Our result - Their result)/(Their result)*. At the end of the tables, minimum, average and maximum differences are shown. According to those results, even if the case *"Rect"* has the worst results in terms of BOT values, it is closer to the best upper bound value of the used aperture number. The gap of the case *"Reg-Cons"* is bigger; on the average it is 9.71. There is no data available for the comparison of other cases. However, we can conclude that since we did not explicitly consider the aperture numbers in our approach, those numbers are possibly away from the optimum.

To visualize the difference between the best upper bounds and our results of used aperture numbers, we draw performance profile charts of those data for *"Reg_Cons"* case in Figure 4.4 and for *"Rect"* case in Figure 4.3. It can be seen from the figures that the jaws only collimators produce optimum BOT results by using close to best upper bound number of apertures. However, regular MLC under consecutiveness constraint can obtain optimum BOT values by using at least 6 times more number of apertures than the best upper bound.

Even if the number of apertures results are not very close to the optimum values, we compared the number of used apertures results of different collimator technologies in Figure 4.5 by using a performance profile chart. Figure 4.5 indicates that minimum number of apertures is used by freeform collimator as expected. Jaws only collimators also use less number of apertures compared to others; but the minimum number of apertures they use is nearly 3 times bigger than the number of apertures used by freeform collimator. The rest of the collimator technologies act similarly and at the worst case, they use apertures in the quantity of nearly 13 times bigger than the minimum used aperture number.

Table 4.5. Number of apertures comparison of the regular MLC under rectangular constraint with the results of Taşkın *et al.* (2012) [19].

	Rect		
Name	**Number of Used Apertures Results of our Column Generation Approach**	**Best UB of Number of Used Apertures Results of Taşkın et al. (2012)**	**GAP**
case1_beam1	72	66	0.09
case1_beam2	58	50	0.16
case1_beam3	70	62	0.13
case1_beam4	73	62	0.18
case1_beam5	54	53	0.02
case2_beam1	131	107	0.22
case2_beam2	121	96	0.26
case2_beam3	116	96	0.21
case2_beam4	140	111	0.26
case2_beam5	120	92	0.30
case3_beam1	140	116	0.21
case3_beam2	80	70	0.14
case3_beam3	138	111	0.24
case3_beam4	129	103	0.25
case3_beam5	83	76	0.09
case4_beam1	120	108	0.11
case4_beam2	100	95	0.05
case4_beam3	115	92	0.25
case4_beam4	117	101	0.16
case4_beam5	105	86	0.22
case5_beam1	79	71	0.11
case5_beam2	72	63	0.14
case5_beam3	79	68	0.16
case5_beam4	75	66	0.14
case5_beam5	63	57	0.11
min GAP			0.02
avg GAP			0.17
max GAP			0.30

Table 4.6. Number of apertures comparison of the regular MLC under consecutiveness constraint with the results of Taşkın *et al.* (2010) [11].

	Reg_Cons		
Name	**Number of Used Apertures Results of our Column Generation Approach**	**Best UB of Number of Used Apertures Results of Taşkın et al. (2010)**	**GAP**
case1_beam1	94	11	7.55
case1_beam2	73	10	6.30
case1_beam3	101	11	8.18
case1_beam4	112	11	9.18
case1_beam5	81	10	7.10
case2_beam1	213	14	14.21
case2_beam2	172	13	12.23
case2_beam3	160	13	11.31
case2_beam4	188	14	12.43
case2_beam5	154	14	10.00
case3_beam1	217	13	15.69
case3_beam2	126	14	8.00
case3_beam3	171	13	12.15
case3_beam4	161	13	11.38
case3_beam5	120	13	8.23
case4_beam1	205	16	11.81
case4_beam2	139	16	7.69
case4_beam3	166	15	10.07
case4_beam4	174	15	10.60
case4_beam5	129	17	6.59
case5_beam1	123	10	11.30
case5_beam2	96	12	7.00
case5_beam3	107	10	9.70
case5_beam4	119	12	8.92
case5_beam5	80	13	5.15
min GAP			5.15
avg GAP			9.71
max GAP			15.69

Table 4.7. Number of apertures comparison of the first 20 data instances.

	Dual		**Free**		**Reg_Cons**		**Reg_Int**		**Rect**		**Rot_Cons**		**Rot_Int**	
Name	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**
case1_beam1	617	94	303	17	588	94	467	94	294	72	613	94	560	91
case1_beam2	382	74	238	13	424	73	369	72	238	58	371	74	431	73
case1_beam3	612	101	325	17	638	101	533	95	346	70	541	101	582	100
case1_beam4	645	113	337	18	618	112	546	111	352	73	570	113	515	111
case1_beam5	485	83	247	15	477	81	393	80	258	54	485	83	483	83
case2_beam1	1427	213	572	16	1388	213	1179	212	638	131	1401	213	1279	213
case2_beam2	1211	173	495	18	1199	172	1072	170	531	121	1026	173	1351	173
case2_beam3	1089	163	486	20	1232	160	1176	161	500	116	1071	161	1180	161
case2_beam4	977	188	511	19	1049	188	1001	188	535	140	1396	188	1511	172
case2_beam5	954	154	459	19	1091	154	873	148	496	120	804	154	1136	152
case3_beam1	1508	219	592	19	1359	217	1556	218	730	140	1660	218	2161	214
case3_beam2	717	127	411	18	754	126	691	122	432	80	685	127	868	120
case3_beam3	949	172	511	20	1046	171	1014	172	540	138	978	172	1382	169
case3_beam4	874	160	483	20	921	161	866	158	523	129	1029	159	1196	160
case3_beam5	804	120	404	20	692	120	737	114	434	83	800	120	1191	119
case4_beam1	2454	206	623	18	1433	205	1491	201	703	120	1802	206	2228	202
case4_beam2	751	140	452	20	659	139	667	134	472	100	796	140	961	141
case4_beam3	1142	168	581	16	1046	166	946	160	606	115	1066	168	1069	168
case4_beam4	1007	174	564	19	1032	174	971	172	619	117	1352	174	1560	168
case4_beam5	632	131	419	19	755	129	676	128	427	105	651	132	952	130

Table 4.8. Number of apertures comparison of the second 20 data instances.

	Dual		Free		Reg_Cons		Reg_Int		Rect		Rot_Cons		Rot_Int	
Name	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**
case5_beam1	916	123	362	12	761	123	690	120	356	79	735	123	935	113
case5_beam2	707	96	316	17	675	96	507	96	308	72	514	96	603	93
case5_beam3	661	107	330	18	576	107	499	106	331	79	582	107	620	104
case5_beam4	872	120	343	14	757	119	613	119	357	75	770	120	740	115
case5_beam5	515	81	284	16	538	80	384	80	294	63	503	81	493	79
case6_beam1	622	105	299	18	611	104	573	101	313	71	636	105	649	100
case6_beam2	310	53	172	14	271	53	256	51	175	42	276	53	270	52
case6_beam3	479	97	266	20	401	97	448	93	286	76	448	97	453	97
case6_beam4	457	92	251	20	408	91	396	92	271	65	574	92	504	92
case6_beam5	266	59	179	14	254	59	250	54	176	45	261	60	284	60
case7_beam1	1124	115	355	18	715	115	702	114	357	89	714	116	947	115
case7_beam2	850	122	359	15	788	122	662	121	371	75	619	122	748	122
case7_beam3	861	133	372	17	839	133	625	132	388	84	788	132	701	130
case7_beam4	799	134	373	18	717	134	626	131	431	90	639	134	795	132
case7_beam5	806	133	421	19	864	133	656	133	465	87	652	133	882	133
case8_beam1	732	119	356	17	632	119	672	118	372	85	645	119	729	118
case8_beam2	621	103	327	18	744	101	557	98	339	57	532	103	502	100
case8_beam3	666	111	352	20	559	114	515	115	373	95	510	115	549	111
case8_beam4	682	106	329	20	679	104	532	106	352	69	625	105	610	100
case8_beam5	709	101	325	16	822	100	631	96	323	69	606	101	576	101

Table 4.9. Number of apertures comparison of the last 15 data instances.

	Dual		Free		Reg_Cons		Reg_Int		Rect		Rot_Cons		Rot_Int	
Name	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**	**Number of Generated Apertures**	**Number of Used Apertures**
case9_beam1	609	98	279	19	512	98	437	97	285	68	566	98	684	92
case9_beam2	458	80	261	17	437	80	415	79	258	55	387	80	458	77
case9_beam3	559	84	265	16	540	83	470	81	256	63	449	84	423	83
case9_beam4	599	99	294	20	669	99	503	99	312	70	491	99	632	95
case9_beam5	536	91	286	14	607	90	480	87	282	68	545	91	633	84
case10_beam1	3783	217	641	19	1195	215	1260	208	662	159	1823	216	2000	217
case10_beam2	671	112	371	20	685	110	553	110	386	84	649	112	836	110
case10_beam3	1083	125	557	20	1062	126	895	123	559	89	943	125	1051	121
case10_beam4	965	155	515	20	1024	153	884	155	562	118	826	155	1087	153
case10_beam5	812	139	489	20	833	136	767	135	508	109	920	139	1379	135
case11_beam1	560	126	295	19	448	128	504	128	346	92	517	128	572	127
case11_beam2	412	106	262	20	417	103	440	103	354	82	415	106	419	106
case11_beam3	454	104	246	16	394	103	413	103	275	69	373	104	394	104
case11_beam4	445	122	277	17	504	120	532	120	344	69	443	122	509	122
case11_beam5	263	77	203	20	271	78	288	78	242	68	255	78	279	78
Minimum	263	53	172	12	254	53	250	51	175	42	255	53	270	52
Average	820.02	123.96	375.00	17.80	738.36	123.31	670.16	121.67	398.96	87.49	733.24	124.02	846.22	121.65
Maximum	3783	219	641	20	1433	217	1556	218	730	159	1823	218	2228	217

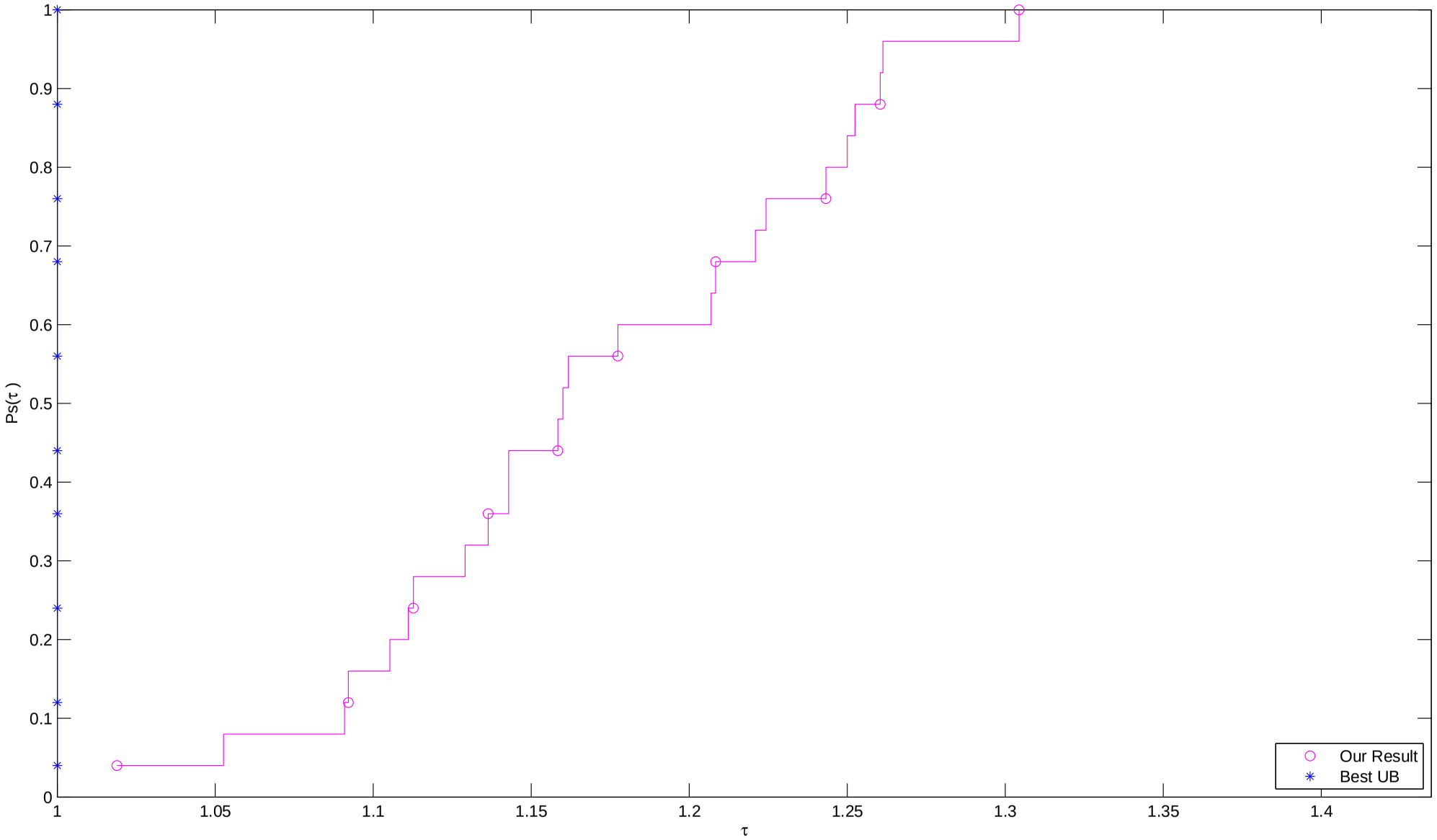

Figure 4.3. The performance profile chart for number of apertures comparison of the regular MLC under rectangular constraint with the results of Taşkın *et al.* (2012) [19].

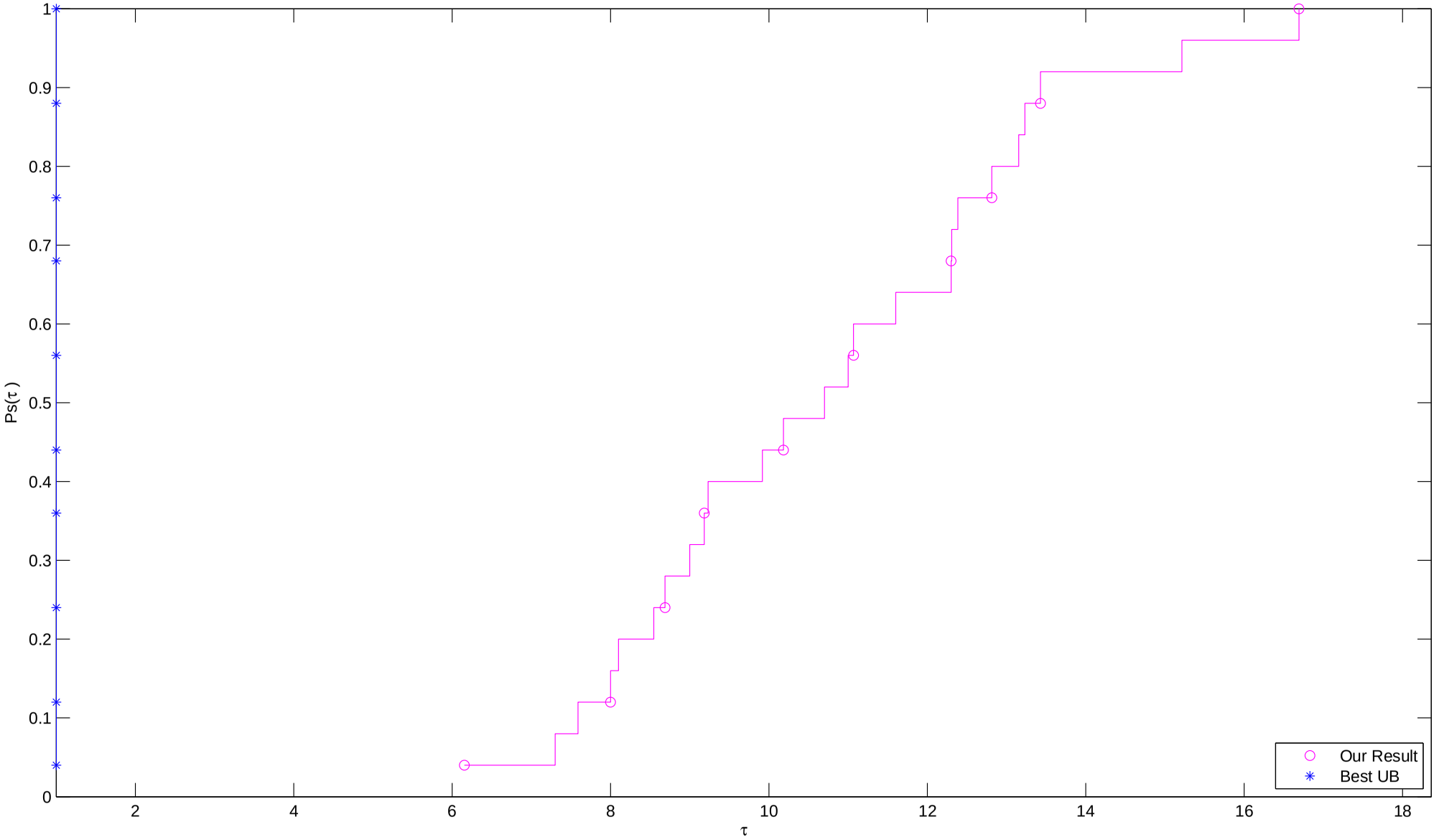

Figure 4.4. The performance profile chart for number of apertures comparison of the regular MLC under consecutiveness constraint with the results of Taşkın *et al.* (2010) [11].

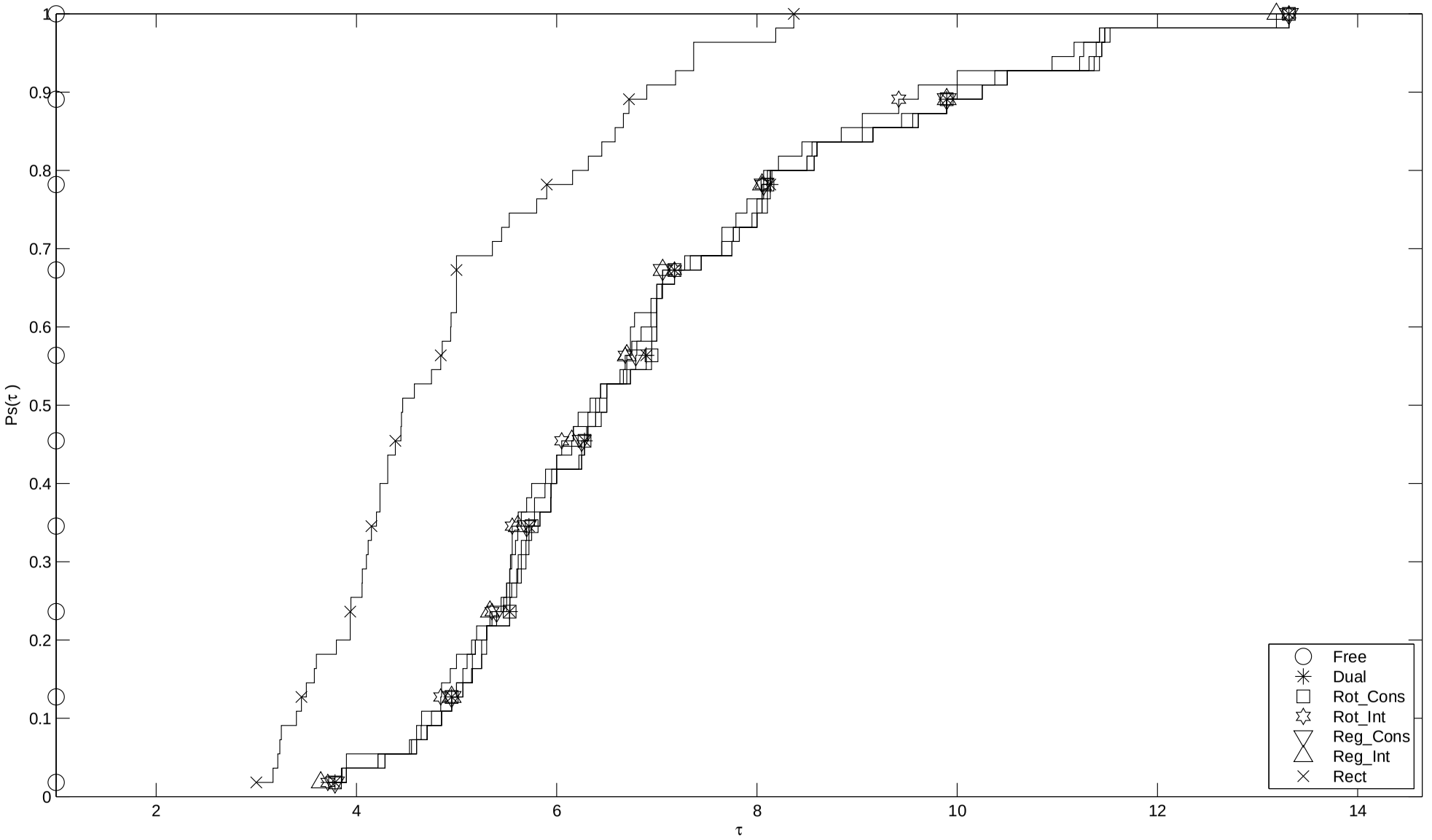

Figure 4.5. The performance profile chart for number of used aperture values used as performance metric.

5. CONCLUSIONS AND FUTURE RESEARCH

In this thesis, we investigated the leaf sequencing optimization problem with minimum beam-on time objective. We compared the efficiency of using different collimator technologies under different restrictions. We focused on the testing of regular, rotating and dual multileaf collimator systems under different combinations of consecutiveness, interdigitation and rectangular constraints and a virtual freeform collimator. Our aim was to generate a comparison basis for them and formulate a general leaf sequencing optimization problem with minimum beam on time objective. We applied column generation approach to deal with the dimensionality of the formulation. In column generation procedure, the subproblem structure changed depending on the used machinery. Hence, we generated seven different subproblems to solve each collimator case.

Our tests on clinical data show that the dual MLC under consecutiveness constraint acts almost as well as a virtual freeform collimator in terms of beam-on time values. This result is due to the fact that dual MLC is the most flexible collimator system after the freeform collimator. The efficiency ranking goes as follows; rotating MLC under consecutiveness, rotating MLC under interdigitation, regular MLC under consecutiveness, regular MLC under interdigitation and finally regular MLC under rectangular constraint. Collimators that can only form rectangular shapes yield very high results in terms of beam-on time. According to our analysis, it can be concluded that as the complexity of the collimator system increases and the restrictions on the system decreases, lower beam-on time results are obtained as expected and the dual MLC is close to the ultimate limit.

Even if we did not take into account the number of apertures evaluation criteria in our approach, we evaluated our solutions from the used aperture numbers point of view. The results indicate that the minimum number of apertures number is used by the freeform collimator. The jaws only collimator also uses close to minimum number of apertures. The rest of the collimator technologies perform alike and worse than these

two. It should be noted that since we do not consider obtaining less aperture number criteria in our formulations or algorithms, these are not general technology-dependent results. They are the results of a side analysis of our results.

As a future work direction, the analysis can be extended by including the other collimator types and constraints. For example, collimator technologies under different combinations of minimum leaf separation, maximum leaf spread and tongue and groove constraints can be evaluated in terms of the delivery efficiencies and compared with the available results. Furthermore, the proposed column generation approach can be revised to account for the number of used apertures. This revision makes the MP an integer programming problem. Therefore, branch and price algorithm must be applied to solve the problem. In that case, we expect longer CPU times. Therefore, dealing with the CPU time will be another concern.

REFERENCES

1. Ehrgott, M., Ç. Güler, H. W. Hamacher and L. Shao, "Mathematical Optimization in Intensity Modulated Radiation Therapy", *4OR*, Vol. 6, No. 3, pp. 199–262, 2008.

2. Chen, D. Z. and C. Wang, "Algorithms for Geometric Problems in Intensity-Modulated Radiation Therapy", *RIMS Kokyuroku*, Vol. 1641, pp. 45–67, 2009.

3. Jing, J., R. Cao, X. Pei, Y. Wu, G. Li and H. Lin, "Optimization of Multileaf Collimator Leaf Sequences based on Multiple Algorithms", *Bioinformatics and Biomedical Engineering (iCBBE), 2010 4th International Conference on*, pp. 1–4, IEEE, 2010.

4. Craft, D., "Local Beam Angle Optimization with Linear Programming and Gradient Search", *Physics in Medicine and Biology*, Vol. 52, No. 7, p. N127, 2007.

5. Lim, G. J., J. Choi and R. Mohan, "Iterative Solution Methods for Beam Angle and Fluence Map Optimization in Intensity Modulated Radiation Therapy Planning", *OR Spectrum*, Vol. 30, No. 2, pp. 289–309, 2008.

6. Wang, X., X. Zhang, L. Dong, H. Liu, Q. Wu and R. Mohan, "Development of Methods for Beam Angle Optimization for IMRT using an Accelerated Exhaustive Search Strategy", *International Journal of Radiation Oncology, Biology and Physics*, Vol. 60, No. 4, pp. 1325–1337, 2004.

7. Zhang, Y. and M. Merritt, "Fluence Map Optimization in IMRT Cancer Treatment Planning and a Geometric Approach", *Multiscale optimization methods and applications*, pp. 205–227, 2006.

8. Aleman, D. M., D. Glaser, H. E. Romeijn and J. F. Dempsey, "Interior Point Algorithms: Guaranteed Optimality for Fluence Map Optimization in IMRT", *Physics in Medicine and Biology*, Vol. 55, No. 18, p. 5467, 2010.

9. Tuncel, A. T., F. Preciado, R. L. Rardin, M. Langer and J.-P. P. Richard, "Strong Valid Inequalities for Fluence Map Optimization Problem under Dose-Volume Restrictions", *Annals of Operations Research*, Vol. 196, No. 1, pp. 819–840, 2012.

10. Cambazard, H., E. O'Mahony and B. O'Sullivan, "A Shortest Path-Based Approach to the Multileaf Collimator Sequencing Problem", *Discrete Applied Mathematics*, Vol. 160, No. 1, pp. 81–99, 2012.

11. Taşkın, Z. C., J. C. Smith, H. E. Romeijn and J. F. Dempsey, "Optimal Multileaf Collimator Leaf Sequencing in IMRT Treatment Planning", *Operations Research*, Vol. 58, No. 3, pp. 674–690, 2010.

12. Taşkın, Z. C. and M. Cevik, "Combinatorial Benders Cuts for Decomposing IMRT Fluence Maps using Rectangular Apertures", *Computers & Operations Research*, Vol. 40, No. 9, pp. 2178–2186, 2013.

13. Men, C., H. E. Romeijn, Z. C. Taşkın and J. F. Dempsey, "An Exact Approach to Direct Aperture Optimization in IMRT Treatment Planning", *Physics in Medicine and Biology*, Vol. 52, No. 24, p. 7333, 2007.

14. Men, C., X. Jia and S. B. Jiang, "GPU-Based Ultra-Fast Direct Aperture Optimization for Online Adaptive Radiation Therapy", *Physics in Medicine and Biology*, Vol. 55, No. 15, p. 4309, 2010.

15. Salari, E. and J. Unkelbach, "A Column-Generation-Based Method for Multi-Criteria Direct Aperture Optimization", *Physics in Medicine and Biology*, Vol. 58, No. 3, p. 621, 2013.

16. Broderick, M., M. Leech and M. Coffey, "Direct Aperture Optimization as a Means of Reducing the Complexity of Intensity Modulated Radiation Therapy Plans", *Radiation Oncology*, Vol. 4, No. 8, pp. 1–7, 2009.

17. Webb, S., "A 4-Bank Multileaf Collimator Provides a Decomposition Advantage

for Delivering Intensity-Modulated Beams by Step-and-Shoot", *Physica Medica*, Vol. 28, No. 1, pp. 1–6, 2012.

18. Blin, G., P. Morel, R. Rizzi and S. Vialette, "Towards Unlocking the Full Potential of Multileaf Collimators", *SOFSEM 2014: Theory and Practice of Computer Science*, p. 138, 2014.

19. Taşkın, Z. C., J. C. Smith and H. E. Romeijn, "Mixed-Integer Programming Techniques for Decomposing IMRT Fluence Maps using Rectangular Apertures", *Annals of Operations Research*, Vol. 196, No. 1, pp. 799–818, 2012.

20. Anderson, J., R. Symonds-Tayler and S. Webb, "Investigating the Fundamentals of IMRT Decomposition using Ten Simple Collimator Models", *Physics in Medicine and Biology*, Vol. 51, No. 9, p. 2225, 2006.

21. Wake, G. M., N. Boland and L. S. Jennings, "Mixed Integer Programming Approaches to Exact Minimization of Total Treatment Time in Cancer Radiotherapy using Multileaf Collimators", *Computers & Operations Research*, Vol. 36, No. 3, pp. 795–810, 2009.

22. Baatar, D., N. Boland, S. Brand and P. J. Stuckey, "CP and IP Approaches to Cancer Radiotherapy Delivery Optimization", *Constraints*, Vol. 16, No. 2, pp. 173–194, 2011.

23. Kalinowski, T., "Reducing the Tongue-and-Groove Underdosage in MLC Shape Matrix Decomposition", *Algorithmic Operations Research*, Vol. 3, No. 2, 2008.

24. Boland, N., H. W. Hamacher and F. Lenzen, "Minimizing Beam-On Time in Cancer Radiation Treatment using Multileaf Collimators", *Networks*, Vol. 43, No. 4, pp. 226–240, 2004.

25. Engelbeen, C. and S. Fiorini, *The Segmentation Problem in Radiation Therapy*, Ph.D. Thesis, Université Libre de Bruxelles, 2010.

26. Wu, X., X. Dou, J. E. Bayouth and J. M. Buatti, "An Almost Linear Time Algorithm for Field Splitting in Radiation Therapy", *Computational Geometry: Theory and Applications*, Vol. 46, No. 6, pp. 673–687, 2013.

27. Salari, E., C. Men and H. E. Romeijn, "Accounting for the Tongue-and-Groove Effect using a Robust Direct Aperture Optimization Approach", *Medical Physics-New York-Institute of Physics*, Vol. 38, No. 3, p. 1266, 2011.

28. Engel, K., "A New Algorithm for Optimal Multileaf Collimator Field Segmentation", *Discrete Applied Mathematics*, Vol. 152, No. 1, pp. 35–51, 2005.

29. Ahuja, R. K. and H. W. Hamacher, "A Network Flow Algorithm to Minimize Beam-On Time for Unconstrained Multileaf Collimator Problems in Cancer Radiation Therapy", *Networks*, Vol. 45, No. 1, pp. 36–41, 2005.

30. Kamath, S., S. Sahni, J. Li, J. Palta and S. Ranka, "Leaf Sequencing Algorithms for Segmented Multileaf Collimation", *Physics in Medicine and Biology*, Vol. 48, No. 3, p. 307, 2003.

31. Baatar, D., H. W. Hamacher, M. Ehrgott and G. J. Woeginger, "Decomposition of Integer Matrices and Multileaf Collimator Sequencing", *Discrete Applied Mathematics*, Vol. 152, No. 1, pp. 6–34, 2005.

32. Agazaryan, N. and T. D. Solberg, "Segmental and Dynamic Intensity-Modulated Radiotherapy Delivery Techniques for Micro-Multileaf Collimator", *Medical Physics*, Vol. 30, No. 7, pp. 1758–1767, 2003.

33. Siochi, R. A. C., "Variable Depth Recursion Algorithm for Leaf Sequencing", *Medical Physics*, Vol. 34, No. 2, pp. 664–672, 2007.

34. Ernst, A. T., V. H. Mak and L. R. Mason, "An Exact Method for the Minimum Cardinality Problem in the Treatment Planning of Intensity-Modulated Radiotherapy", *INFORMS Journal on Computing*, Vol. 21, No. 4, pp. 562–574, 2009.

35. Langer, M., V. Thai and L. Papiez, "Improved Leaf Sequencing Reduces Segments or Monitor Units Needed to Deliver IMRT using Multileaf Collimators", *Medical Physics*, Vol. 28, No. 12, pp. 2450–2458, 2001.

36. Mason, L. R., V. H. Mak-Hau and A. T. Ernst, "An Exact Method for Minimizing the Total Treatment Time in Intensity-Modulated Radiotherapy", *Journal of the Operational Research Society*, Vol. 63, No. 10, pp. 1447–1456, 2012.

37. Wolsey, L., *Integer Programming*, Wiley Series in Discrete Mathematics and Optimization, Wiley New York, 1998.

38. Romeijn, H. E., R. K. Ahuja, J. F. Dempsey and A. Kumar, "A Column Generation Approach to Radiation Therapy Treatment Planning using Aperture Modulation", *SIAM Journal on Optimization*, Vol. 15, No. 3, pp. 838–862, 2005.

39. Dolan, E. D. and J. J. Moré, "Benchmarking Optimization Software with Performance Profiles", *Mathematical Programming*, Vol. 91, No. 2, pp. 201–213, 2002.