On Maximum Weight Objects Decomposable into Based Rectilinear Convex Objects

Mahmuda Ahmed, Iffat Chowdhury, Matt Gibson,
Mohammad Shahedul Islam, and Jessica Sherrette

Department of Computer Science
University of Texas at San Antonio
San Antonio, TX USA
{mahmed,ichowdhu,gibson,msislam,jsherett}@cs.utsa.edu

Abstract. Our main concern is the following variant of the image segmentation problem: given a weighted grid graph and a set of vertical and/or horizontal base lines crossing through the grid, compute a maximum-weight object which can be decomposed into based rectilinear convex objects with respect to the base lines. Our polynomial-time algorithm reduces the problem to solving a polynomial number of instances of the maximum flow problem.

1 Introduction

An area of work that has recently attracted extensive attention in the pattern recognition and computer vision communities is *image segmentation*. It is the process of partitioning a digital image into multiple objects for better representation and analysis of an image. From another view point, image segmentation is assigning labels to the pixels of an image such that the pixels with the same label define a particular object which may have certain visual characteristics. In practice image segmentation is used to detect objects and boundaries in the image. An example, in *medical imaging*, image segmentation is used to help locate tumors and other pathologies, measure tissue volumes, computer-guided surgery, diagnosis, treatment planning, study of anatomical structure etc. There are many other applications of image segmentation including fingerprint recognition, traffic control systems and agriculture imaging.

Image Segmentation as an Optimization Problem. Finding a "good" segmentation is often treated as an optimization problem, see for example [2,12,13,4,5,9,7,1]. Using the framework of Asano et al. [2] we are given a weighted grid graph where each grid cell corresponds to a pixel in the original image and weights on the grid cells are related to the likelihood that the particular pixel is in the object we wish to identify (positive weights are assigned to grid cells whose corresponding pixel is likely in the object and negative weights are assigned to grid cells whose corresponding pixel is likely in the background). Then we attempt to find some subset of the grid that optimizes an objective function

F. Dehne, R. Solis-Oba, and J.-R. Sack (Eds.): WADS 2013, LNCS 8037, pp. 1–12, 2013.

subject to some constraints. Let G be an $\sqrt{n} \times \sqrt{n}$ four-neighborhood grid graph. For $1 \leq i \leq \sqrt{n}$ and $1 \leq j \leq \sqrt{n}$ the grid cell p at the (i, j) position in the grid has a real value $w(p)$ called the weight of p. We call i the x-coordinate of p and j the y-coordinate of p and let p_x (resp. p_y) denote the x-coordinate (resp. y-coordinate) of p. A *region* (or *object*) R will be defined as any subset of grid cells, and we define the weight of R to be $w(R) = \sum_{p \in R} w(p)$. We are interested in computing the region R with maximum weight subject to some constraints.

Research has shown that knowledge of the geometric shape of the object that you are looking for can greatly increase an algorithm's effectiveness in practice, see for example [10,19,3,18,11]. Polynomial-time algorithms have been given which identify an optimal solution for the following classes of objects: x-monotone regions, based monotone regions, rectilinear convex regions, and star-shaped regions [4,9,8].

Objects Decomposable into Elementary Shapes. Chun et al. [7] consider the maximum-weight region problem with a twist on the constraints of some previous works. They are interested in finding a maximum-weight region that may not have simple geometric structure, but can be *decomposed* into objects with simple geometric structure. A region R can be decomposed into m objects of a particular structure if and only if there exists a coloring of the grid cells of R using m colors such that each of the objects induced by the grid cells of each of the color classes have the desired structure.

This type of problem is very interesting from both a practical perspective as well as a theoretical perspective. It is interesting in practice because an algorithm for such a problem can identify more complicated objects while still allowing control of the topology of the output object. It is interesting from a theoretical perspective because the decomposition constraints of the problem poses an interesting computational challenge to overcome. If we instead consider finding objects which are the union of m objects with simple geometric structure, the problem often becomes much harder (for example, finding the maximum-weight object that is the union of two star-shaped objects is NP-hard [7]). The decomposition variant of the problems may admit polynomial-time algorithms, but it is not trivial to design such an algorithm for many classes of objects even when $m = 2$.

Chun et al.[7] give an efficient algorithm for computing the maximum-weight region that can be decomposed into two digital star-shaped regions with respect to two given "center" grid cells. Gibson et al. [14] give a maximum-flow based algorithm for the same 2-star problem and recently Gibson et al. [15] extend the result of [14] to identify the maximum-weight object decomposable into c star-shaped objects for any constant c in polynomial time.

Chun et al. [7] consider the problem of computing the maximum-weight object decomposable into *based monotone object* with respect to a set of k given base lines. A base line of the grid graph G is a vertical (x $=$ i) or horizontal (y $=$ j) path of grid cells across the grid for $1 \leq i \leq \sqrt{n}$ and $1 \leq j \leq \sqrt{n}$. For a given horizontal base line $l : y = i$, a *based monotone object* is a union of segments of columns intersecting the base line. See Figure 1 (a) and (b) for an illustration. They do not require a based monotone object for a particular base line to be a connected region. This allows them to use the base lines to partition the grid into $O(k^2)$ subproblems which they solve independently using dynamic programming. Recently Chun et. al [6] gave an algorithm for finding the optimal baseline locations using quadtree decomposition.

Our Contribution. Given a weighted grid graph, we are interested in identifying the maximum-weight object decomposable into *Based Rectilinear Convex* (BRC) objects with respect to c given base lines for a constant c. For a given horizontal base line $l : y = i$, a BRC object with respect to l satisfies the properties of being a based monotone object with the additional constraint that the intersection of the object with any horizontal line is always undivided (a symmetric notion is defined for vertical base lines). See Figure 1 (c) and (d) for an illustration. In contrast to a based monotone object, a BRC object is by definition a connected object. Therefore, as opposed to the based monotone case, the base lines do not decompose the grid into subproblems which can be solved independently.

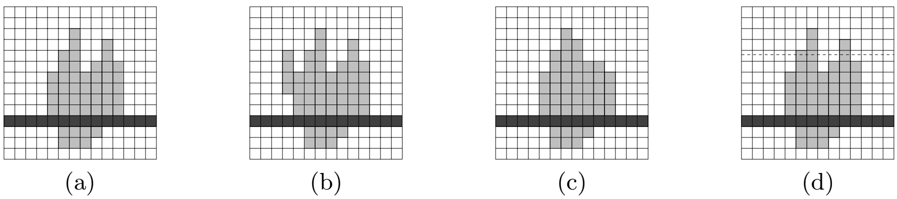


Fig. 1. Part (a) is a based monotone object with respect to the base line. Part (b) is not a based monotone object. Part (c) is a BRC object with respect to the base line. Part (d) is not a BRC object (the intersection of the object with the dotted line is not connected).

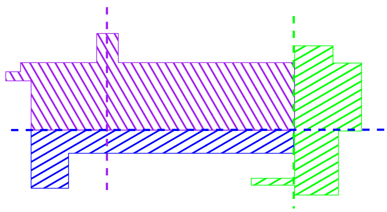


Fig. 2. A 3-BRC object with respect to 3 given base lines

We call an object which can be decomposed into c different BRC objects a *c-BRC object*. See Figure 2 for an illustration. When $c = 1$ the problem is easily solvable, but until now there has been no polynomial-time algorithm given even when $c = 2$. Our main contribution is the following theorem.

Theorem 1. *There exists a polynomial-time algorithm which computes a maximum-weight object decomposable into based rectilinear objects with respect to a set of c given base lines in a weighted grid graph for any constant $c \geq 1$.*

We prove Theorem 1 by giving a polynomial-time algorithm for computing a maximum-weight 2-BRC object for a restricted special case of the 2-BRC problem. We solve this restricted special case by observing some key geometric properties of a BRC object and show that these observations allow us to reduce the problem to computing the maximum-weight closed set in a polynomial number of appropriately defined directed graphs. It is well known that a maximum-weight closed set can be computed in polynomial time [17,16] via a reduction to the maximum flow problem. We then show how to carefully reduce the c-BRC problem to several instances of the restricted 2-BRC problem. This reduction will be done in a way so that the solution to the 2-BRC instances can be merged to obtain an optimal solution for the c-BRC instance.

To guarantee that our algorithm returns an optimal solution, our algorithm iteratively guesses the structure of an optimal solution. For each guess, we compute the maximum-weight c-BRC object which corresponds to this guess. We show that by making a polynomial number of guesses, we can guarantee that we guess the correct structure for an optimal solution; however, the polynomial is too large to be of practical interest. That being said, our result shows how the structure of a solution can be used to reduce the problem to the maximum flow problem. If this structure is given as input by a user or is found via a heuristic, then our work shows that the problem can be reduced to solving a small number of maximum flow instances which would be of practical interest. Also, our technique can easily be modified to compute the complement of a maximum weight c-BRC object (this may be more efficient for some inputs).

Organization of the Paper. In Section 2, we give an algorithm which computes a maximum-weight 2-BRC object for a restricted version of the problem. In Section 3, we extend the result to find a maximum-weight c-BRC object for any constant $c \geq 2$.

2 Algorithm for a Restricted 2-BRC Problem

In this section, we give a polynomial-time algorithm for a restricted version of 2-BRC object using an $\sqrt{n} \times \sqrt{n}$ four-neighborhood grid graph G and two base lines at the boundary of the grid. We show that this problem can be solved by computing the maximum-weight closed set for a linear number of appropriately constructed directed graphs. Given a weighted, directed graph $D = (V, E)$, a *closed set* is a subset of the vertices $C \subseteq V$ such that if $u \in C$ and $(u, v) \in E$ then $v \in C$. Intuitively, if C is a closed set then there is no edge from a vertex in C to a vertex in $V \setminus C$. The weight of a closed set C is simply the sum of the weights of the vertices in C.

Preliminaries. Initially, we assume that the base lines are parallel and without loss of generality the base lines are at $y = 1$ and $y = \sqrt{n}$. At the end of this section, we show how to handle the case where base lines are perpendicular. We view each grid cell as having a x-coordinate and a y-coordinate (the grid cell in the lower left corner has x-coordinate = y-coordinate = 1 and the grid cell in the upper right corner has x-coordinate = y-coordinate = $\sqrt{n}$). If O is a BRC object with respect to the top of the grid, we say that O is a type-N BRC object (its base line is the "northern" base line). Similarly, if O is a BRC object with respect to the bottom of the grid, we say that O is a type-S BRC object.

Let O be a BRC object and without loss of generality assume it is a type-S object. A peak of O is a grid cell $p \in O$ for which no other grid cells $p' \in O$ have y-coordinate greater than the y-coordinate of p. Similarly, for type-N object, a peak is a pixel with minimum y-coordinate over all pixels in the object. We define a peak line of O to be a vertical line through the grid which contains a peak. See Figure 3 (a) for an illustration.

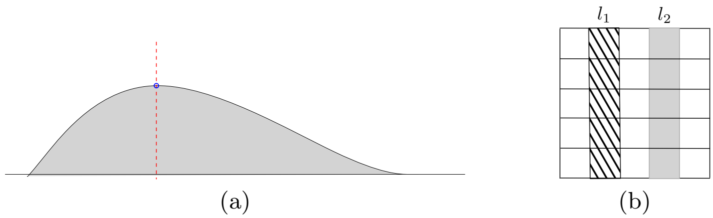


Fig. 3. Peak lines: (a) A peak line where the shaded region is a type-S BRC object. (b) The patterned and shaded portion of the grid are the peak lines l_1 and l_2 respectively.

The following observation is the key idea that allows us to reduce the restricted 2-BRC problem to a maximum-weight closed set problem. The proof has been omitted due to lack of space.

Observation 2. *Let O be a subset of grid cells in the grid, and let l be the vertical line through the grid at $x = \alpha$. Then O is a type-S (resp. type-N) BRC object with respect to peak line l if and only if the following properties hold:*

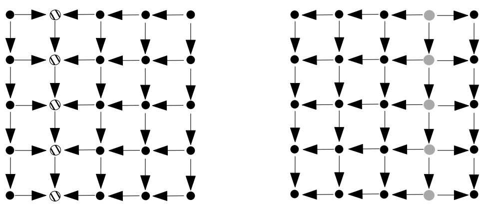
1. *for each $o \in O$ such that $o_x \leq \alpha$, each grid cell p such that $p_x = o_x$ and $p_y \leq o_y$ (resp. $p_y \geq o_y$) is in O and each grid cell q such that $q_y = o_y$ and $o_x \leq q_x \leq \alpha$ we have $q \in O$.*
2. *for each $o' \in O$ such that $o'_x > \alpha$, each grid cell p' such that $p'_x = o'_x$ and $p'_y \leq o'_y$ (resp. $p'_y \geq o'_y$) is in O and each grid cell q' such that $q'_y = o'_y$ and $o'_x \geq q'_x \geq \alpha$ we have $q' \in O$.*

The consequence of Observation 2 is that if we know a peak line for each BRC object, then we can compute them via a single maximum-weight closed set computation in an appropriately defined directed graph (we can guess all possible pairs of peak lines using $\sqrt{n} \times \sqrt{n} = n$ guesses).

Construction of the Directed Graph. We now describe the construction of the directed graph to find the 2-BRC object with respect to two peak lines l_1 and l_2. See Figure 3 (b) which shows the peak line l_1 for type-S object and l_2 for type-N object. For the remainder section, we assume when we mention a 2-BRC object, we refer a 2-BRC object with respect to l_1 and l_2.

We call our graph $D_{\{l_1,l_2\}}$. There are two "sections" of vertices in $D_{\{l_1,l_2\}}$, and each grid cell in G has exactly one vertex in each of these sections. The vertices in a closed set from the first section will determine what grid cells are in the type-S BRC object in G, and the vertices in a closed set from the second section will determine what grid cells are in the type-N BRC object in G. Let V_1 denote the vertices in the section for the type-S BRC object, and let us define V_2 similarly for type-N BRC object. For a grid cell g, let v_g^1 denote its corresponding vertex in V_1 and let v_g^2 denote its corresponding vertex in V_2. For ease of description, we view V_1 and V_2 being embedded in the same layout as the grid cells in G.

We will now define three edge sets E_1, E_2, and E_3. E_1 will consist of edges with both endpoints in V_1, E_2 will consist of edges with both endpoints in V_2, and E_3 will consist of edges with their tail in V_1 and their head in V_2. Let us now define the edge set E_1. See Figure 4(a) for an illustration. In V_1, every vertex has an edge to a vertex directly 'below' it (if it exists). And all the "horizontally adjacent" vertices have an edge between the corresponding vertices directed towards the peak line l_1. These are the all edges in the edge set E_1.



(a) (b)

Fig. 4. (a) The arrangement of edge set of E_1 where patterned line is the peak line l_1 and this vertex set is in V_1. (b) The arrangement of edge set of E_2 where lightly shaded line is the peak line l_2 and this vertex set is in V_2.

We now describe the edge set E_2. See Figure 4(b) for an illustration. Similarly, in V_2, every vertex has an edge to a vertex directly below it (if it exists). But all the horizontally adjacent vertices have an edge between the corresponding vertices directed away from the peak line. These are all the edges in the set E_2.

The edge set E_3 consists of the directed edges (v_g^1, v_g^2) for each grid cell g. This completes the construction of the edge sets E_1, E_2, and E_3.

Our directed graph $D_{\{l_1,l_2\}}$ has vertex set $V := V_1 \cup V_2$ and edge set $E := E_1 \cup E_2 \cup E_3$. We assign weights on the vertices as follows. The weight of each vertex $v_g^1 \in V_1$ is set to be $w(g)$. The weight of each vertex $v_g^2 \in V_2$ is set to be $-w(g)$. This completes the construction of the graph.

Relationship between a Closed Set and a 2-BRC Object. We now describe a function T which will take as input a subset of vertices in $D_{\{l_1,l_2\}}$ and outputs a subset of grid cells in G. Fix any subset $V' \subseteq V$ of $D_{\{l_1,l_2\}}$. For any vertex $v_g^1 \in V' \cap V_1$, the corresponding grid cell g is in $T(V')$. For any vertex $v_g^2 \in V_2 \setminus V'$, the corresponding grid cell g is in $T(V')$. In other words, a grid cell g is in $T(V')$ if v_g^1 is in V' or if v_g^2 *is not* in V'. If v_g^1 is not in V' and v_g^2 is in V', then g is not in $T(V')$. We will prove in Lemma 1 that if V' is a closed set of $D_{\{l_1,l_2\}}$ then $T(V')$ is a 2-BRC object whose weight is the same as the weight of V' (minus a constant).

We now define another function T' which takes as input a 2-BRC object and returns a set of vertices in $D_{\{l_1,l_2\}}$. T' is the inverse of T. Fix R to be any subset of grid cells that can be decomposed into a type-S BRC object and a type-N BRC object. Fix such a decomposition, and color the grid cells in the type-S BRC object red and the cells in the type-N BRC object blue. Let us call the red grid cells R_1 and the blue grid cells R_2. For each red cell $r \in R_1$ we have that $v_r^1 \in T'(R)$ and $v_r^2 \in T'(R)$. For each blue cell $b \in R_2$ we have $v_b^1 \notin T'(R)$ and $v_b^2 \notin T'(R)$. For all uncolored cells g we have $v_g^1 \notin T'(R)$ and $v_g^2 \in T'(R)$. This concludes the definition of the function $T'(R)$, and in Lemma 2 we will prove that $T'(R)$ is a closed set in $D_{\{l_1,l_2\}}$ and has weight equal to R (minus a constant).

Note that we have $T'(T(C)) = C$ for every closed set C and $T(T'(R)) = R$ for every 2-BRC object. Thus proving Lemma 1 and Lemma 2 will complete the proof that the maximum-weight region in G that is decomposable into two BRC objects with respect to the peak lines can be computed by finding a maximum-weight closed set in $D_{\{l_1,l_2\}}$. The proof of Lemma 2 is similar to the proof of Lemma 1 and is omitted due to lack of space.

Lemma 1. *Fix any closed set C of $D_{\{l_1,l_2\}}$. Then $T(C)$ is a 2-BRC object and has weight equal to C (minus a constant).*

Proof. We first show that $T(C)$ is a 2-BRC object. Let C_1 be $C \cap V_1$, and abusing notation let $T(C_1) \subseteq T(C)$ be the grid cells g such that $v_g^1 \in C_1$. We will argue that $T(C_1)$ is a type-S BRC object by showing $T(C_1)$ satisfies properties 1 and 2 of Observation 2. We can show this is true by considering the construction of $D_{\{l_1,l_2\}}$. There is an edge in $D_{\{l_1,l_2\}}$ from v_g^1 to the vertex corresponding to the grid cell towards l_1 on the same horizontal line and the vertex directly below it. Since C is a closed set, it follows that both of these vertices must also be in the closed set. It follows from a simple inductive argument that for any $v_g' \in C_1$, all of the vertices v_c^1 which are between v_g' and the peak line l_1 on the same horizontal line and all of the vertices $v_{c'}^1$ which are below v_g' on the same vertical line will be in C. See Figure 4 (a). By the definition of T, it must be that all such grid cells c and c' are in $T(C)$. We thus have by Observation 2 that $T(C_1)$ is a type-S BRC object.

Now let C_2 be $C \cap V_2$, and abusing notation let $T(C_2) \subseteq T(C)$ be the grid cells g such that $v_g^2 \notin C_2$. We will now show that $T(C_2)$ is a type-N BRC object. Let α_2 denote the x-coordinate of the points on l_2. We remind the reader that by the definition of T, vertices in $V_2 \setminus C_2$ correspond with the grid cells that are in $T(C_2)$. Again, to show that $T(C_2)$ is type-N BRC object, we will show that properties 1 and 2 of Observation 2 hold for $T(C_2)$. Suppose for the sake of contradiction that $g \in T(C_2)$ (without loss of generality assume $g_x \leq \alpha_2$) but there is a grid cell g' such that $g_x = g'_x$ and $g_y < g'_y$ and g' is not in $T(C_2)$. Since g' is not in $T(C_2)$, we have $v^2_{g'} \in C_2$. According to the construction of $D_{\{l_1,l_2\}}$, there must be an edge from $v^2_{g'}$ to the vertex below it. Since C is a closed set, we must have that these vertices are in C_2. An inductive argument follows that all of the vertices corresponding to grid cells below g' on the same vertical line must be in C_2. This of course implies that $g \notin T(C_2)$, a contradiction. We have the similar argument for a grid cell g'' such that $g''_y = g_y$ and $g_x \leq g''_x \leq \alpha_2$. We thus prove the properties 1 and 2 of Observation 2 and hence $T(C_2)$ is a type-N BRC object.

We will now argue that $T(C)$ is a 2-BRC object. We will prove this by showing that $T(C_1)$ and $T(C_2)$ are disjoint. This is easy to see from the definition of the edge set E_3. Let g be some grid cell in $T(C_1)$. By definition, this implies that $v^1_g \in C_1$. The edge (v^1_g, v^2_g) is in E_3, and since C_1 is a closed set it must be that $v^2_g \in C_2$. This implies that for any $g \in T(C_1)$, we have $g \notin T(C_2)$. This completes the proof that $T(C_1)$ and $T(C_2)$ are disjoint and therefore $T(C)$ can be decomposed into two BRC objects.

This concludes the proof that $T(C)$ is a 2-BRC object, and we will now prove that C and $T(C)$ have the same weight (minus a constant). First let w_1 be the sum of the weights of the vertices in C_1, and let w_2 be the sum of the weights of the vertices in C_2. The weight of the closed set is exactly $w_1 + w_2$. The corresponding grid cell for each vertex in C_1 is also in $T(C)$, and moreover has the exact same weight. So the sum of the weights of the grid cells in $T(C_1)$ is w_1. Recall that the vertices in C_2 correspond to the exact set of grid cells that are not in $T(C_2)$, and thus the weight of the grid cells in $T(C_2)$ is $w(V_2) + w_2$ (we remind the reader that the weight of a vertex in C_2 is the negative of the weight of its corresponding grid cell). Therefore, the weight of the grid cells in $T(C)$ is $w_1 + w_2 + w(V_2)$. Since $w_1 + w_2$ is the weight of C, we conclude that the weight of C is equal to the weight of the grid cells in $T(C)$ minus $w(V_2)$. This concludes the proof of the lemma. □

Lemma 2. *Fix any subset R of grid cells in G that is a 2-BRC object. Then $T'(R)$ is a closed set in $D_{\{l_1,l_2\}}$ and has weight equal to R (minus a constant).*

So now we have that if C is a maximum-weight closed set of $D_{\{l_1,l_2\}}$, then $T(C)$ is a maximum-weight 2-BRC object. There are n total pairs of peak lines, so we can check all possible pairs. One of those pairs will correspond with the peak lines for the maximum-weight 2-BRC object, and therefore the maximum-weight 2-BRC object obtained for these peak lines will be the maximum-weight 2-BRC object for the entire problem.

Handling Perpendicular Base Lines. Now we will assume that the base lines can be perpendicular. Since we now have a vertical base line, the “sides” of the grid can be base lines. In this setting, we say an object O is a type-W (resp. type-E)

BRC object if O is a BRC object with respect to the "western" (resp. "eastern") base line.

Without loss of generality, assume we have the southern base line and the western base line and we wish to find the maximum-weight 2-BRC object decomposable into a type-S BRC object and a type-W BRC object. We can compute this object using a similar approach as to what we used in Section 2 by slightly changing the construction of the directed graph. Note that a peak line for a type-W object is a horizontal line (perpendicular to the base line). Suppose we are given a vertical peak line l_1 and a horizontal peak line l_2. We will construct the directed graph $D_{\{l_1,l_2\}}$ slightly differently. The vertex set will again be $V_1 \cup V_2$ where V_1 will be used to identify the type-S object and V_2 will be used to identify the type-W object. The edge sets E_1 and E_3 will be exactly as defined above, but the edge set E_2 will change. The edge set E_2 will consist of horizontal edges directed towards the base line and vertical edges directed away from the peak line. The weights are assigned the same way as before. We can argue similarly as we did in Lemma 1 and Lemma 2 that for a maximum-weight closed set C of $D_{\{l_1,l_2\}}$, we have $T(C)$ is a maximum-weight 2-BRC object with respect to l_1 and l_2, and by considering all $\sqrt{n} \times \sqrt{n}$ possible pairs of base lines we can compute in polynomial time the maximum-weight 2-BRC object with respect to the southern and western base lines. We conclude that for any two base lines, we can compute in polynomial-time the maximum-weight object for the restricted 2-BRC problem.

3 Extension to the c-BRC Problem

We now give a polynomial-time algorithm for the original problem in which we are given a weighted grid graph G and c base lines, and we wish to compute a maximum-weight c-BRC object. Our algorithm iteratively makes guesses about the structure of an optimal solution OPT. Using this structure, we reduce the problem to several instances of the restricted 2-BRC problem. The reduction is handled in two parts. First we decompose the grid into $O(c^2)$ rectangular-subgrid instances of a restricted version of the 4-BRC problem. This restricted version will be similar to the restricted 2-BRC problem considered in Section 2 (base lines are at the boundary of the grid). The key property of this instance is that for each instance I of the restricted 4-BRC problem, we have that $I \cap OPT$ can be decomposed into at most 4 BRC objects with respect to the base lines at the boundary of I. We then use a digital Voronoi diagram to break the restricted 4-BRC problem into at most 5 instances of the restricted 2-BRC problem, which we solve using the algorithm given in Section 2. The reduction is carefully done so that the merging of the solutions will be a feasible c-BRC object. When we correctly guess the structure of OPT, we show that the merged solutions will be an optimal c-BRC object. This approach is similar in flavor to the approach of Gibson et al. [15] for computing the maximum-weight object decomposable into c star-shaped objects for any fixed c. An overview of our reduction is now given. Further details and the algorithm have been omitted due to lack of space.

Reduction to the Restricted 4-BRC Problem. We will now reduce the c-BRC problem into $O(c^2)$ instances of the *restricted 4-BRC problem* in which there are at most four base lines, each of which are at the boundary of the grid. Let $B_1, B_2, \ldots, B_c$ be the c disjoint BRC objects that OPT decomposes into. Consider some B_i, and without loss of generality assume that the base line of B_i is horizontal. Let b_i denote the intersection of B_i with its base line. Let ℓ_i and r_i denote the leftmost and rightmost grid cell of b_i respectively, and consider the vertical paths through ℓ_i and r_i. Note that by the definition of a BRC object, any $g \in B_i$ cannot be "outside" of these vertical paths. Similarly a B_i with a vertical base line has a vertical b_i and must be between two horizontal paths.

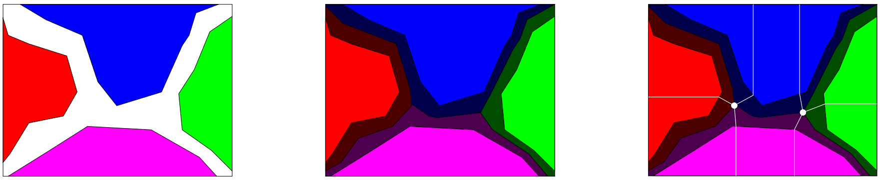
Now consider any grid cell $g \in G$, and consider shooting four axis-parallel rays from g in all four directions until it hits the boundary of G or hits a b_i. We can hit at most four b_is, and g can only be in a B_i such that a ray shot from g hit b_i. To see this, first note that g can only be in a B_i such that the vertical or horizontal line through g hits b_i (otherwise g would be outside of the "vertical paths" described above). Now suppose for the sake of contradiction that an axis-parallel ray from $g \in B_i$ to b_i pierces through a $b_{i'}$. By the definition of BRC object, we have that every grid cell along this ray is in B_i including the grid cell $g' \in b_{i'}$ pierced by the ray. That implies $g' \in B_i$ and $g' \in B_{i'}$, a contradiction.

The subset of grid cells whose axis-parallel rays hit the same set of b_i (from the same "ray direction") induce the desired rectangular instances of the restricted 4-BRC problem. Let I be one such instance, and consider $I \cap OPT$. As we just argued, there will be grid cells from c' different B_i in I for some $1 \leq c' \leq 4$ (at most one from each "direction"). Clearly, $B_i \cap I$ for each of these B_i will be a BRC object with respect to a unique "side" of I. Thus we can view I as an instance of the restricted 4-BRC problem where we have c' base lines, each on a different "side" of I. We note that the optimal 4-BRC object for this problem may not be $OPT \cap I$. We can fix this issue by modifying the weights of the grid cells along the base line.

Reduction to the Restricted 2-BRC Problem. We now suppose we are given an instance I of the restricted 4-BRC problem and we give an overview of how to reduce it to at most 5 instances of the restricted 2-BRC problem. Due to lack of space, the details have been omitted.

First note that if there are at most 2 base lines in I, then the problem is already an instance of the 2-BRC problem. It remains to show how to handle the cases in which we have three or four base lines in a single instance. For the rest of the paper, we will assume that our instance will have all four base lines (it will be clear how to handle the case when we have three base lines).

We will now give a high level overview of the details of the decomposition of the restricted 4-BRC problem into several instances of the 2-BRC problem. Let $OPT(I)$ denote $OPT \cap I$. $OPT(I)$ can be decomposed into type-N, a type-E, a type-S, and a type-W BRC objects, so fix such a decomposition and let N, E, S, and W respectively denote each of these objects. We will consider the *digital Voronoi diagram* for these four sets. That is, we will partition the grid cells of G into four Voronoi regions $V(N), V(E), V(S)$, and $V(W)$ such that any grid cell in a Voronoi region is "closer" to that particular BRC object than it is to any of the other three. See Figure 5 for an illustration. We will show that we can use the vertices of the Voronoi diagram to help us partition the problem into instances of the 2-BRC problem. Intuitively, a vertex of the Voronoi diagram occurs where three or four different Voronoi regions "touch each other". Consider a vertex of the Voronoi diagram, and suppose this is a vertex where exactly three of the Voronoi regions "come together". For this vertex, we will find three paths in the grid. Each path will begin at this vertex and will end at one of the base lines (one path per base line). The paths will be chosen in a way such that they will consist of grid cells which all belong to the same Voronoi region (the Voronoi region associated with the base line at which the path ends). We find these paths for each of the vertices of the Voronoi diagram, and we will show that if we remove the grid cells in these paths then we are left with a constant number of connected components, each of which contains grid cells from at most 2 Voronoi regions. This allows us to use the algorithm of Section 2 to compute the maximum-weight 2-BRC object from these components and merge the solutions together to obtain $OPT(I)$.



(a) (b) (c)

Fig. 5. Decomposing into 2-BRC instances. (a) Suppose this is $OPT(I)$. (b) The Voronoi diagram associated with $OPT(I)$. (c) The vertices and paths used to decompose into 2-BRC instances.

References

1. Anzai, S., Chun, J., Kasai, R., Korman, M., Tokuyama, T.: Effect of corner information in simultaneous placement of k rectangles and tableaux. Discrete Mathematics, Algorithms and Applications 2(4), 527–537 (2010)
2. Asano, T., Chen, D.Z., Katoh, N., Tokuyama, T.: Efficient algorithms for optimization-based image segmentation. Int. J. Comput. Geometry Appl. 11(2), 145–166 (2001)
3. Chan, T.F., Zhu, W.: Level set based shape prior segmentation. In: CVPR (2), pp. 1164–1170 (2005)
4. Chen, D.Z., Chun, J., Katoh, N., Tokuyama, T.: Efficient algorithms for approximating a multi-dimensional voxel terrain by a unimodal terrain. In: Chwa, K.-Y., Munro, J.I. (eds.) COCOON 2004. LNCS, vol. 3106, pp. 238–248. Springer, Heidelberg (2004)

5. Chen, D.Z., Hu, X.S., Luan, S., Wu, X., Yu, C.X.: Optimal terrain construction problems and applications in intensity-modulated radiation therapy. In: Möhring, R.H., Raman, R. (eds.) ESA 2002. LNCS, vol. 2461, pp. 270–283. Springer, Heidelberg (2002)
6. Chun, J., Horiyama, T., Ito, T., Kaothanthong, N., Ono, H., Otachi, Y., Tokuyama, T., Uehara, R., Uno, T.: Base location problems for base-monotone regions. In: Ghosh, S.K., Tokuyama, T. (eds.) WALCOM 2013. LNCS, vol. 7748, pp. 53–64. Springer, Heidelberg (2013)
7. Chun, J., Kasai, R., Korman, M., Tokuyama, T.: Algorithms for computing the maximum weight region decomposable into elementary shapes. In: Dong, Y., Du, D.-Z., Ibarra, O. (eds.) ISAAC 2009. LNCS, vol. 5878, pp. 1166–1174. Springer, Heidelberg (2009)
8. Chun, J., Korman, M., Nöllenburg, M., Tokuyama, T.: Consistent digital rays. Discrete & Computational Geometry 42(3), 359–378 (2009)
9. Chun, J., Sadakane, K., Tokuyama, T.: Efficient algorithms for constructing a pyramid from a terrain. IEICE Transactions 89-D(2), 783–788 (2006)
10. Das, P., Veksler, O., Zavadsky, V., Boykov, Y.: Semiautomatic segmentation with compact shape prior. Image Vision Comput. 27(1-2), 206–219 (2009)
11. Freedman, D., Zhang, T.: Interactive graph cut based segmentation with shape priors. In: CVPR (1), pp. 755–762 (2005)
12. Fukuda, T., Morimoto, Y., Morishita, S., Tokuyama, T.: Data mining using two-dimensional optimized accociation rules: Scheme, algorithms, and visualization. In: Jagadish, H.V., Mumick, I.S. (eds.) SIGMOD Conference, pp. 13–23. ACM Press (1996)
13. Fukuda, T., Morimoto, Y., Morishita, S., Tokuyama, T.: Data mining with optimized two-dimensional association rules. ACM Trans. Database Syst. 26(2), 179–213 (2001)
14. Gibson, M., Han, D., Sonka, M., Wu, X.: Maximum weight digital regions decomposable into digital star-shaped regions. In: Asano, T., Nakano, S.-I., Okamoto, Y., Watanabe, O. (eds.) ISAAC 2011. LNCS, vol. 7074, pp. 724–733. Springer, Heidelberg (2011)
15. Gibson, M., Varadarajan, K., Wu, X.: On a planar segmentation problem (2012)
16. Hochbaum, D.S.: A new - old algorithm for minimum-cut and maximum-flow in closure graphs. Networks 37(4), 171–193 (2001)
17. Picard, J.-C.: Maximal closure of a graph and applications to combinatorial problems. Management Science 22(11), 1268–1272 (1976)
18. Thiruvenkadam, S.R., Chan, T.F., Hong, B.-W.: Segmentation under occlusions using selective shape prior. SIAM J. Imaging Sciences 1(1), 115–142 (2008)
19. Veksler, O.: Star shape prior for graph-cut image segmentation. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 454–467. Springer, Heidelberg (2008)