On the Computational Complexity of Partitioning Weighted Points into a Grid of Quadrilaterals

Alexander Idelberger* Maciej Liśkiewicz*

Abstract

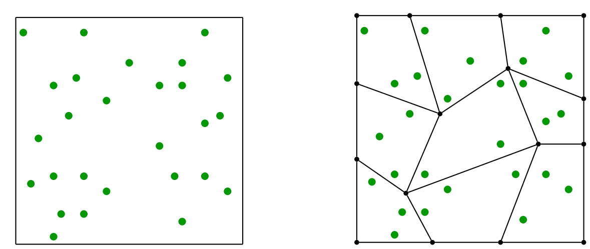
In the paper the computational complexity of the following partitioning problem is studied: Given a rectangle R in the plane, a set Q of positive-weighted points in R, and two positive integers n_1, n_2, find a partitioning of R into quadrilaterals whose dual graph is an $n_1 \times n_2$ grid such that each quadrilateral contains points of equal total weight. If such a partitioning does not exist, find a solution that minimizes some objective function. This problem is motivated by applications in image processing including, among others, image enhancement and similarity retrieval, and it is closely related to the table cartogram problem introduced recently by Evans et al. [ESA 2013]. While there exist fast algorithms that find optimal partitions in 1-dimension, the 2-dimensional case seems to be much harder to solve. Pichon et al. [ICIP 2003] proposed a heuristic yielding admissible solutions, but the computational complexity of the problem has so far remained open. In this paper we prove that a decision version of the problem is NP-hard.

1 Introduction

We study the following geometric problem to which we refer as *q-grid partitioning*: For a given rectangle $R = [0, a] \times [0, b]$ in the plane, a finite set of positive-weighted points $Q = \{q_1, \ldots, q_m\}$ in R, and two positive integers n_1 and n_2, find a partitioning of the rectangle R into $n_1 \times n_2$ quadrilateral faces whose dual graph is an $n_1 \times n_2$ grid such that each face contains points of equal total weight. Particularly, if to each point in Q we assign a unit weight, every quadrilateral face should contain the same number of points. If such a partitioning does not exist, the task is to find a solution, among possibly many, that minimizes some objective function. Figure 1 shows an example instance of the problem and an optimal solution.

In a more general setting of the problem, which we call *q-grid partitioning with a reference table*, we are given, besides weighted points Q in $[0, a] \times [0, b]$, a 2-dimensional $n_1 \times n_2$ table of non-negative desired reference weights. Then the task is to partition the rectangle into $n_1 \times n_2$ quadrilateral faces each containing points of total weight equal to the corresponding desired value. If such a partitioning does not exist, a solution minimizing some objective function is required. To express the initial partitioning problem in this setting, one needs to define each reference value equal to $\frac{1}{n_1 \cdot n_2} \sum_{q \in Q} \omega(q)$, where ω denotes the weight function.

The above problem is a natural generalization of the well-studied 1-dimensional case: given a set of m positive-weighted points Q in an interval $[0, a] \subset \mathbb{R}$, an integer n, and a reference vector of size n, find a partitioning of the interval into n subintervals $[0, x_1), [x_1, x_2), \ldots, [x_{n-1}, a]$ which minimizes some objective function.



(a) Input: a set of points in square $[0,1] \times [0,1]$ and integers $n_1 = n_2 = 3$.

(b) Output: an optimal partitioning into 3×3 quadrilateral faces.

Figure 1: A points set with a unit weight assigned to any point. The task is to partition the square (a) into 3×3 quadrilateral faces each containing the same number of points. Figure (b) shows an optimal solution.

1.1 Motivation

Our study is motivated by applications in image processing, including image enhancement – one of the central problems in image processing (see e.g. [7]). Basic, well-known image enhancement techniques are histogram equalization and histogram specification.

A histogram of an image in a d dimensional color space can be represented by a weighted point set $Q \subset [0, a]^d$, with $\omega(q)$ describing the number of pixels of a color q. E.g. in the RGB color space each color is represented by a vector with three components Red, Green, and Blue and it is a point in the unit cube $[0, 1]^3$.

The grey-scale histogram equalization problem is formulated as the 1-dimensional partitioning problem with the sample variance $\frac{1}{n} \sum_i (s_i - \mu)^2$ or the average absolute deviation $\frac{1}{n} \sum_i |s_i - \mu|$ as objective functions. Here, s_i

*Institute for Theoretical Computer Science, Universität zu Lübeck, alex@pirx.de, liskiewi@tcs.uni-luebeck.de

denotes the total weight of points in the i-th interval and $\mu = \frac{1}{n}\sum_{q\in Q}\omega(q)$. The grey-scale histogram specification can be expressed as 1-dimensional partitioning with reference vector $r_1, r_2, \ldots, r_n$ that represents a desired histogram.

Histogram equalization for color images becomes a much more difficult challenge because of the multidimensional nature of color. The difficulty of the problem arises due to the correlation between the color components as well as the complexity of the human perception. The research in image processing led to the development of two main classes of algorithms: the first one operating in the RGB space and the second one operating in nonlinear color spaces (for more details see e.g. [1, 5, 9, 14, 16, 18]). The complexity questions studied in this paper concern algorithmic problems representing colors in linear spaces.

The most straightforward extension of grey-scale histogram equalization to color images is to apply it for each color band separately, obtaining an orthogonal grid. However, since this approach ignores the correlation between the color components, it is not suitable for color enhancement and related tasks. Using an orthogonal grid and taking the correlation into account results in the modeling (discussed in more detail below) analyzed by Grigni and Manne [8]. The more appropriate extension, proposed by Pichon et al. [16], initially partitions the color space of the image histogram Q into cells of a scaled regular mesh. Then the mesh is iteratively deformed minimizing the absolute deviation. Mapping the cells of the deformed mesh to the corresponding cells of a regular mesh yields the color transformation. Thus, the method generates an almost uniform color histogram making an efficient use of the color space. The main challenge in this approach is to find an optimal deformed mesh.

1.2 Known Results

Chang and Wong [3] and independently Chow and Kou [4] were the first to provide efficient algorithms for the 1-dimensional partitioning. The proposed algorithms solve the problem minimizing $\sum_i |s_i - r_i|$ in time $\mathcal{O}(m \cdot n)$. Next, Chang and Wong [2] generalized their tree-search technique for arbitrary p-norms, in particularly for $\sum_i (s_i - r_i)^2$ with a $\mathcal{O}(m^2 \cdot n)$ time bounded algorithm. Kundu [11], independently to [2], provided a shortest paths algorithm for the 1-dimensional partitioning minimizing $\sum_i (s_i - \mu)^2$, yielding the same time complexity $\mathcal{O}(m^2 \cdot n)$.

The computational complexity of the partitioning of $Q \subset [0, a] \times [0, b]$ into a grid of quadrilaterals has so far remained open. In [16] Pichon et al. give a heuristic for this problem but the proposed algorithm does not provide optimal solutions. Moreover the authors do not analyze the approximation factor of the algorithm. In the conclusions of [11], Kundu claims that the problem can be formulated as shortest path problem and solved efficiently. Later [12], he observes that the problem is much more difficult than the 1-dimensional case and that the suggested approach does not work.

1.3 Our Contribution

In this paper we prove that a decision version of the partitioning problem is NP-hard: given a set of weighted points $Q = \{q_1, \ldots, q_m\}$ in a rectangle $R = [0, a] \times [0, b]$ and two positive integers n_1, n_2, find an $n_1 \times n_2$ q-grid partitioning of R minimizing the deviation in the maximum-norm. Moreover, we show that the problem is NP-hard also for some other important objective functions. We show that the problem remains NP-hard even if Q contains points of integer coordinates, i.e. if $Q \subset [0, a] \times [0, b] \cap \mathbb{N}^2$. To prove these results we show a polynomial time reduction from the planar version of the 1-In-3-SAT problem, which is known to be NP-hard [13].

We leave as an open question if the partitioning problems are in NP. We conjecture an affirmative answer. The main difficulty in proving this is to show that there must exist an optimal q-grid partitioning the coordinates of which have bounded precision, i.e. such that their representations have polynomial size with respect to the total size of the input.

1.4 Related Results

Recently, Evans et al. introduced in [6] the concept of *table cartogram* – a new model of 2-dimensional cartogram. The input of the table cartogram is an $n_1 \times n_2$ table of non-negative weights and the output is a rectangle R with area equal to the sum of the input weights partitioned into $n_1 \times n_2$ quadrilateral faces each with area equal to the corresponding weight. Evans et al. proved that for any instance of the problem there exists a feasible solution even if the definition of the area of a region is generalized to the weight of a region defined as an integral over some positive density function $\omega : R \to \mathbb{R}^+$. Moreover, the construction requires only polynomial time under some computability assumptions on ω. Our q-grid partitioning with a reference table problem can be formulated as the table cartogram with the *discrete* density function defined for any region A of R as $\text{Area}(A) = \sum_{q\in A}\omega(q)$. Our results show that the table cartogram becomes intractable for such discrete density functions.

In the literature, many similar partitioning problems have been studied. E.g. in [8] Grigni and Manne proved that the following problem is NP-complete: given weighted points $Q = [0, a] \times [0, b] \cap \mathbb{N}^2$, find partitions of intervals: $x_0 = 0 \le x_1 \le \ldots \le x_{n_1} = a$ and $y_0 = 0 \le y_1 \le \ldots \le y_{n_2} = b$ such that the maximum over all $s_{i,j}$ is less than a given constant c. Here $s_{i,j}$ de-

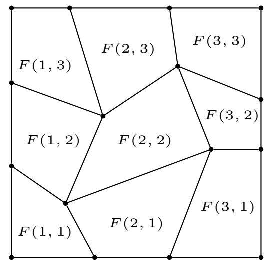


Figure 2: A 3×3 q-grid partitioning of a square area.

notes the sum of weights of points in the (i, j)-th face of the orthogonal grid $\{x_0, x_1, \ldots, x_{n_1}\} \times \{y_0, y_1, \ldots, y_{n_2}\}$. Note that this problem has purely combinatorial character while the problem studied in our paper is of geometrical nature.

2 A Formal Definition of the Partitioning Problem

A *geometric graph* $G = (V, E)$ consists of vertices V considered as points in the plane, and edges as distinct, straight-line segments with endpoints in V. Two geometric graphs $G = (V, E)$ and $G' = (V', E')$ are *topologically isomorphic* if there is a homeomorphism $h : \mathbb{R}^2 \to \mathbb{R}^2$ such that $V' = h(V)$ and $E' = \{h(e) \mid e \in E\}$. A *regular mesh* of the size $n_1 \times n_2$, denoted as M_{n_1,n_2}, is a geometric graph with nodes $V_{n_1,n_2} = \{0, \ldots, n_1\} \times \{0, \ldots, n_2\}$ and edges vw iff $\|v - w\|_1 = 1$.

Let $R = [0, a] \times [0, b]$ be a given axis-parallel rectangle. Graph $G = (V, E)$ is called an $n_1 \times n_2$ *q-grid partitioning* of R if G is topologically isomorphic with a regular mesh M_{n_1,n_2} via a homeomorphism h such that h maps the sides of the rectangle $[0, n_1] \times [0, n_2]$ to the sides of R in the following way: $[0, n_1] \times 0 \mapsto [0, a] \times 0$; $[0, n_1] \times n_2 \mapsto [0, a] \times b$; $0 \times [0, n_2] \mapsto 0 \times [0, b]$; $n_1 \times [0, n_2] \mapsto a \times [0, b]$. For any $(i, j) \in V_{n_1,n_2}$ we will denote the point $h((i, j))$ in V as $v(i, j)$. Moreover, the interior of the quadrilateral (called the face or cell) $\big(v(i, j), v(i-1, j), v(i-1, j-1), v(i, j-1)\big)$ will be denoted as $F(i, j)$. For an example of a q-grid partitioning and its faces see Fig. 2.

Note that because a regular mesh M_{n_1,n_2} is a planar graph embedded, also any topologically isomorphic graph to M_{n_1,n_2} is a planar embedding.

An instance of the q-grid partitioning problem consists of numbers $a, b \in \mathbb{R}^+$ representing a rectangle $R = [0, a] \times [0, b]$, a finite set of weighted points inside R, and positive integers n_1, n_2 describing the size of a grid. The weighted points are specified as a tuple (Q, ω), where $Q \subset R$ and ω is a positive function $\omega : Q \to \mathbb{R}^+$. We call the tuple $P = (Q, \omega)$ a *points instance*. An admissible solution of the problem is an $n_1 \times n_2$ graph G which is a q-grid partitioning of R and the aim is to minimize some specific objective function.

In this paper we consider the following three types of objective functions for the partitioning[1].

Definition 1 *Assume* $R = [0, a] \times [0, b]$, $P = (Q, \omega)$, *with* $Q \subset R$, *and let* $G = (V, E)$ *be an* $n_1 \times n_2$ *q-grid partitioning of* R. *Define weight* $s_{i,j}$ *of face* $F(i, j)$, *as* $s_{i,j} = \sum_{q \in Q \cap F(i,j)} \omega(q)$ *for any* i, j, *with* $0 < i \le n_1$, $0 < j \le n_2$ *and let* $\mu = \sum_{q \in Q} \omega(q)/(n_1 \cdot n_2)$. *Then, we define the following objective functions:*

1. *The maximum over all weights of faces:* $d_{\max}(P, G) = \max_{i,j} s_{i,j}$.
2. *The deviation from the mean in the p-norm:* $d_p(P, G) = \sum_{i,j} |s_{i,j} - \mu|^p$.
3. *The deviation from the mean in the maximum-norm:* $d_\infty(P, G) = \max_{i,j} |s_{i,j} - \mu|$.

Next, we generalize (2) and (3) to objective functions for the q-grid partitioning with a given reference table $\{r_{i,j}\}$ *as* $\sum_{i,j} |s_{i,j} - r_{i,j}|^p$, *resp.* $\max_{i,j} |s_{i,j} - r_{i,j}|$.

Definition 2 (Q-grid Partitioning (QGP)) *Assume* d *is an objective function. Then the problem, denoted as* d-QGP, *is defined as follows: For given real numbers* $a, b > 0$ *representing* $R = [0, a] \times [0, b]$, *points instance* $P = (Q, \omega)$, *with* $Q \subset R$, *integers* n_1, n_2, *and a real number* $c \ge 0$ *decide if there exists an* $n_1 \times n_2$ *q-grid partitioning* G *of* R *with* $d(P, G) \le c$.

3 The Complexity of the Q-grid Partitioning

In this section we give a proof of the NP-hardness for the d_∞ function. Subsequently, we show how to extend this to other norms and special variants of the problem. For further details we refer to [10].

Theorem 1 *The* d_∞-QGP *problem is* NP-*hard.*

Proof. We show the result by a reduction of the Planar-1-In-3-SAT problem, which is known to be NP-complete [13]. Recall that yes-instances of the problem are planar 3-CNF formulae, which are one-in-three satisfiable, i.e., each clause has exactly one true literal. A CNF formula φ is *planar* if the graph $G_\varphi = (V, E)$ defined as

$$V = \{x_i \mid \text{variable } x_i \text{ in } \varphi\} \cup \{c_j \mid \text{clause } c_j \text{ in } \varphi\},$$
$$E = \{\{x_i, c_j\} \mid x_i \text{ is a variable in the clause } c_j\}$$

can be embedded in the plane. To illustrate the reduction, we consider the formula

$$\varphi = \underbrace{(x_1 \vee x_2 \vee \neg x_3)}_{c_1} \wedge \underbrace{(\neg x_1 \vee x_2 \vee x_4)}_{c_2} \qquad (1)$$

with two clauses c_1, c_2. A planar embedding of the corresponding graph G_φ is shown in Fig. 3.

[1]Without loss of generality we only consider such q-grid partitionings G that for all $q \in Q$ the point q does not belong to an edge of G.

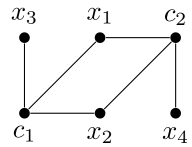


Figure 3: Planar embedding of G_φ with nodes x_1, x_2, x_3, x_4, c_1, c_2 for the formula φ defined in eq. (1).

For a given planar 3-CNF formula φ to construct an instance of d_∞-QGP, we start with a specification of a, b, n_1, and n_2, and then in the course of the proof, we gradually build the points instance $P = (Q, \omega)$ and define the threshold c. The proof is divided into two parts. The first part deals with restricting the flexibility of $n_1 \times n_2$ q-grids G which partition $[0, a] \times [0, b]$. We obtain that whenever the objective value $d_\infty(P, G)$ is less than or equal to a threshold t, every vertex $v(i, j)$ of G must lie within a certain region. The second part consists of a description of gadgets to embed φ into $P = (Q, \omega)$. The construction guarantees that the objective value $d_\infty(P, G)$ will be less than or equal to $c < t$ if and only if the formula is one-in-three satisfiable.

3.1 Enforcing the Regions for Grid Vertices

Firstly, we define $n_1 = n_2 = n$ and choose n such that the gadgets encoding an instance φ of the PLANAR-1-IN-3-SAT problem can be embedded into a grid of size $n \times n$. By [17] we know that, to get the appropriate size, the value can be chosen such that $n \in \mathcal{O}(|G_\varphi|) = \mathcal{O}(|\varphi|)$. For our example φ, it is sufficient to choose $n = 8$. Next, we define $a = b = 8 \cdot n$ and denote, for short, $R = [0, 8 \cdot n] \times [0, 8 \cdot n]$.

In this part of the proof we choose some particular points in R of integer coordinates. To get the appropriate weights, we use a scaling factor

$$\alpha = 16 \cdot n^2$$

depending on n and an auxiliary function

$$g : \{1, \ldots, n\}^2 \times \{1, 2, 3, 4\} \to \mathbb{R}^+$$

defined as follows:

$$g(i, j, l) = \begin{cases} \alpha \cdot 2^{4(i-1)+4n(j-1)} & \text{if } l = 1 \\ \alpha \cdot 2^{4(i-1)+4n(j-1)+1} & \text{if } l = 2 \\ \alpha \cdot 2^{4(i-1)+4n(j-1)+2} & \text{if } l = 3 \\ \alpha \cdot 2^{4n^2} - g(i, j, 1) \\ \quad - g(i, j, 2) - g(i, j, 3) & \text{if } l = 4 \end{cases} \tag{2}$$

Note that $g(i, j, 4) = \alpha \cdot 2^{4n^2} - \alpha \cdot 7 \cdot 2^{4(i-1)+4n(j-1)}$, for any i and j.

Observation 2 *Mapping g is an injective function.*

Proof. We use the bijection $(i, j) \mapsto (i-1) + n(j-1)$ between $\{1, \ldots, n\} \times \{1, \ldots, n\}$ and $\{0, \ldots, n^2 - 1\}$. Then g can be represented equivalently as

$$g'(k, l) = \begin{cases} \alpha \cdot 2^{l-1} \cdot 2^{4k} & \text{if } 1 \le l \le 3 \\ \alpha \cdot 2^{4n^2} - \alpha \cdot 7 \cdot 2^{4k} & \text{if } l = 4 \end{cases}$$

for all $k \in \{0, \ldots, n^2 - 1\}$ and $l \in \{1, 2, 3, 4\}$. For any k we have: $g'(k, 1) < g'(k, 2) < g'(k, 3) < g'(k + 1, 1)$ and $g'(k + 1, 4) < g'(k, 4)$. Moreover, it is true that $g'(n^2 - 1, 3) < g'(n^2 - 1, 4)$. This completes the proof. □

Thus, the range of g:

$$\text{range}(g) = \{g(i, j, l) \mid (i, j, l) \in \text{dom}(g)\}$$

contains $4n^2$ elements of the total sum $n^2 \cdot \alpha \cdot 2^{4n^2}$. Below we show that there exists a unique partitioning of range(g) such that each subset of the partitioning contains numbers of total sum

$$\gamma = \alpha \cdot 2^{4n^2}.$$

Lemma 3 *Let X be a subset of range(g). Then X satisfies $\sum_{x \in X} x = \gamma$ iff there exist $i, j \in \{1, \ldots, n\}$ such that $X = \{g(i, j, l) \mid l \in \{1, 2, 3, 4\}\}$.*

Proof. For any $i, j \in \{1, \ldots, n\}$, the sum $\sum_{l=1}^{4} g(i, j, l)$ obviously evaluates to γ. Thus, if X contains $g(i, j, l)$ with $l = 1, 2, 3, 4$ and a certain pair i, j, the sum of the elements equals γ.

To prove the opposite direction, let X be any set with $\sum_{x \in X} x = \gamma$. According to the inequalities:

$$\sum_{i,j} \sum_{l \in \{1,2,3\}} g(i, j, l) < \gamma < 2 \cdot g(n, n, 4)$$

the set must contain exactly one element with $l = 4$, namely $g(i_0, j_0, 4)$ for some i_0, j_0.

Suppose X contains any element $g(i, j, l)$ with $i + nj > i_0 + nj_0$ and arbitrary l, it follows

$$g(i_0, j_0, 4) + g(i, j, l) > \gamma$$

and thus there is no such element in X.

On the other hand, the sum of all elements with $i + nj < i_0 + nj_0$ and $l = 1, 2, 3$ is less than any element $g(i_0, j_0, l)$ with $l = 1, 2, 3$. Thus X must contain $g(i_0, j_0, l)$ for $l = 1, 2, 3, 4$. □

This property will be utilized to define the weights of the points instance. We construct Q starting with points $(8i - 1, 8j - 1), (8i - 7, 8j - 1), (8i - 1, 8j - 7)$, and $(8i - 7, 8j - 7)$, for all $i, j \in \{1, \ldots, n\}$ and define the corresponding weights as $g(i, j, 1), g(i, j, 2), g(i, j, 3)$, and $g(i, j, 4)$. Till the end of this part of the proof, let P denote the above points instance (for an example see Fig. 4).

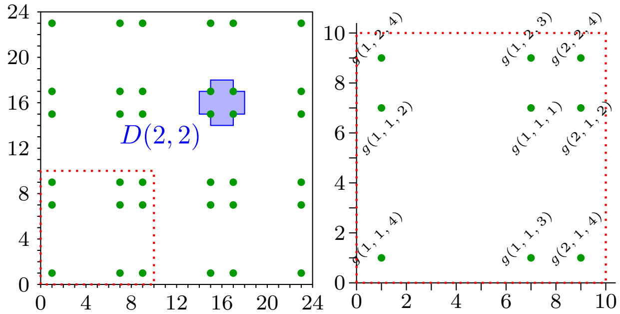


Figure 4: The points instance P defined in the first part of the reduction, for $n = 3$. The forced region $D(2, 2)$, in which the vertex $v(2, 2)$ shall be placed, is highlighted in blue. On the right, the red dotted part is shown in detail.

From Lemma 3 and by observing that all values of g are multiples of α, it follows that, if a face $F(i', j')$ of an $n \times n$ q-grid G which partitions R does not contain points of values exactly $g(i, j, \{1, 2, 3, 4\})$, its value $s_{i',j'}$ will differ from γ by at least α. Throughout the proof, further points will be added to Q, but it is ensured that the total sum of their weights will be at most $\alpha/4$.

So far, the sum of all weights is $n^2 \cdot \gamma$. Regardless of further added points, for the mean μ will satisfy the inequalities:

$$\gamma \;\leq\; \mu \;\leq\; \gamma + \alpha/(4n^2) \;\leq\; \gamma + \alpha/4. \tag{3}$$

It follows that a face of G which does not contain points of weights exactly $g(i, j, \{1, 2, 3, 4\})$ results in $d_\infty(P, G) > \alpha/2$. Hence, t will be defined as

$$t = \alpha/2.$$

We define for all i, j, with $1 \leq i, j \leq n$ square $S_{i,j}$ as:

$$\begin{aligned} S_{i,j} = \big(&(8i-1, 8j-1), (8i-7, 8j-1), \\ &(8i-7, 8j-7), (8i-1, 8j-7)\big). \end{aligned} \tag{4}$$

Lemma 4 *Let G be an $n \times n$ q-grid partitioning of R such that $d_\infty(P, G) \leq t$. Then, no edge of G intersects any square $S_{i,j}$. Particularly, no vertex of G lies inside any square $S_{i,j}$.*

Proof. Firstly, we observe that all vertices of any of the squares must lie in one face of G, as they each represent points with weights $g(i, j, \{1, 2, 3, 4\})$. Thereby we get that there is no edge in G which splits the vertices of a square into two faces.

Suppose that there is an edge of G which intersects one of the squares. Let $S_{i,j}$ be such a square and let F denote a face which contains all vertices of $S_{i,j}$. Face F must be of a concave shape since the convex hull of the contained points is the square itself.

Let us denote by u_1 the vertex of F which lies inside the convex hull of the remaining vertices u_2, u_3, u_4 of F and let u_2 and u_3 be adjacent to u_1. Since F includes all vertices of $S_{i,j}$, the vertices u_2, u_3, and u_4, that form a triangle, must lie in the exterior of $S_{i,j}$ and sides $\overline{u_1u_2}$ and $\overline{u_1u_3}$ must intersect the square $S_{i,j}$. Two cases can occur: each of $\overline{u_1u_2}$ and $\overline{u_1u_3}$ intersects either once or twice sides of $S_{i,j}$. In the first case $\overline{u_1u_2}$ and $\overline{u_1u_3}$ must cross the same side of $S_{i,j}$. In the second case, the last crossings (going from u_1 to u_k) must belong also to the same side of $S_{i,j}$. Let $\overline{AB}$ be the respective side of $S_{i,j}$. Figure 5 illustrates the first case.

Now, depending on whether $\overline{AB}$ is the upper, lower, left, or right side, four cases should be considered. Since they are symmetric to each other, we analyze only one of them.

So, assume $\overline{AB}$ is the left side of $S_{i,j}$. Moreover, if multiple squares $S_{i,j}$ exist which are similarly intersected by two sides of their respective faces, then assume additionally that we have chosen a square $S_{i,j}$ of maximal index i. We show that, if $i \leq n-1$, then both sides $\overline{u_2u_4}$ and $\overline{u_3u_4}$ must cross the left side of $S_{i+1,j}$ and if $i = n$, then $\overline{u_2u_4}$ and $\overline{u_3u_4}$ must cross the right boundary of R. In the first case we get a contradiction to our assumption that index i is maximal: indeed since G is planar any face containing the vertices of $S_{i+1,j}$ must cross the left side of $S_{i+1,j}$. Case $i = n$ contradicts the construction of G: the graph should be completely contained in R.

Let (x_2, y_2) and (x_3, y_3) be coordinates of u_2 and u_3, respectively. From our analysis above it follows that $x_2, x_3 \leq 8i - 7$ and that all vertices of $S_{i,j}$ lay above the side $\overline{u_2u_4}$ resp. below $\overline{u_3u_4}$. Moreover, recall that F does not contain any vertex of the remaining squares $S_{i',j'}$. To evaluate the minimum value for the x-coordinate of u_4, we first consider the straight line passing through the points $(8i-7, 8j-9)$ and $(8i-1, 8j-7)$. We obtain the equation for the line

$$y = x/3 - 8i/3 + 8j - 20/3.$$

For $x = 8i + 1$, i.e. for the x-coordinate of the left side of $S_{i+1,j}$ if $i \leq n-1$ or $x = 8n + 1 = a + 1$ if $i = n$, we get $y = 8j - 19/3 < 8j - 4$. Similarly, for the straight line passing through $(8i-7, 8j+1)$ and $(8i-1, 8j-1)$ we get for $x = 8i+1$ that $y > 8j - 4$. This implies that the x-coordinate of u_4 must be bigger that $8i + 1$. As the y-coordinate of u_4 must satisfy $8j - 7 \leq y \leq 8j - 1$, this completes the proof. □

Next, we define regions inside R, which we denote as $D(i, j)$ and call *forced regions*. The definition is as follows: For all $i, j \in \{0, \dots, n\}$ we consider a rectangular polygon with the centroid in $(8i, 8j)$ such that it is obtained from a square of side length 4 by cutting off in each corner a unit square. Then $D(i, j)$ is defined as the intersection of the interior of the polygon for i, j and R. For an example $D(i, j)$, see Fig. 4.

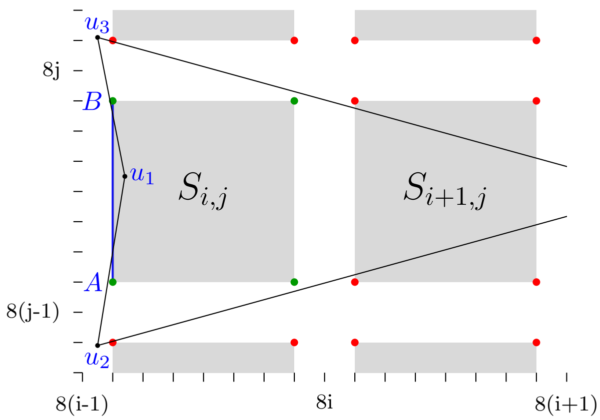


Figure 5: Possible shape of a face which on the one hand touches the interior of $S_{i,j}$ but on the other hand also includes all green highlighted vertices of $S_{i,j}$ and excludes all other red points of the points instance.

Lemma 5 *Let G be an $n \times n$ q-grid partitioning of R such that $d_\infty(P, G) \leq t$. Then: (1) each vertex $v(i, j) \in V(G)$ lies in the region $D(i, j)$ and (2) each point $q \in Q$ with $\omega(q) = g(i, j, l)$ is included in the face $F(i, j)$ of G.*

Proof. To prove property (1) suppose that there exists a vertex of $V(G)$, denoted by u_1, which is not contained by any region $D(\hat{\imath}, \hat{\jmath})$. Due to Lemma 4, u_1 cannot be situated inside any square. Thus, it must be located in $R \setminus (\bigcup_{\hat{\imath},\hat{\jmath}} D(\hat{\imath}, \hat{\jmath}) \cup \bigcup_{i,j} S_{i,j})$.

Assume u_1 is vertex of a face F which encloses a square $S_{i,j}$, for some i, j. Let us first consider that u_1 is placed in a strip adjacent to $S_{i,j}$. Depending on the location: to the left, right, below or above of $S_{i,j}$, there are four cases to be considered. As they are symmetric, we consider only the case that u_1 is placed to the left of $S_{i,j}$. If (x, y) denote coordinates of u_1, this leads to constraints: $8i-9 < x < 8i-7$ and $8j-6 < y < 8j-2$.

The next and previous vertices of u_1 in face F are again denoted by u_2 and u_3, while u_2 shall be the vertex below $S_{i,j}$ and u_3 the one above. To enclose $S_{i,j}$, u_2 and u_3 must be as far as possible to the right. We can determine the best position of u_2 along a line through the bottom-leftmost position of u_1, i. e. $(8i-9, 8j-2)$, and the bottom left corner of $S_{i,j}$, $(8i-7, 8j-1)$. The legal position furthest to the right is the intersection of the line and the square $S_{i,j-1}$, i. e. $u_2 = (8i-3, 8j-9)$. Analogously, the best possible placement of u_3 would be $(8i-3, 8j+1)$. Even though u_2 and u_3 cannot both have these coordinates (we use different positions of u_1), there is no legal position of the forth vertex u_4 of F. Passing a first line through u_2 and the bottom right corner of $S_{i,j}$ as well as the second line through u_3 and the top right corner, the intersection of both lines, representing the leftmost position of u_4, is $(8i+2, 8j-4)$ and thereby in the interior of $S_{i+1,j}$. An example of such a situation with $u_1 = (8i-9, 8j-4)$ is shown in Fig. 6. This contradiction to Lemma 4 establishes that any vertex of G is contained in one region $D(\hat{\imath}, \hat{\jmath})$.

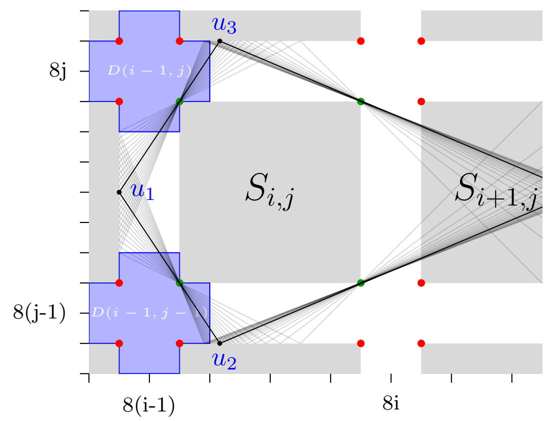


Figure 6: Positioning of a vertex u_1 which is neither placed inside any square $S_{i',j'}$ nor inside any region $D(\hat{\imath}, \hat{\jmath})$. The resulting face violates Lemma 4.

If vertex u_1 of F is placed in a strip which is not adjacent to $S_{i,j}$ then by essentially much easier case analysis, we can prove that there is no legal position of the vertex u_4 of F. There are four cases to consider: u_1 is located in a vertical non-adjacent strip of the x-coordinates $\leq 8i-7$ or $\geq 8i-1$ or it is located in a horizontal non-adjacent strip of the y-coordinates $\leq 8j-7$ or $\geq 8j-1$. For each case one can easily show that if both sides $\overline{u_1u_2}$ and $\overline{u_1u_3}$ do not intersect any square $S_{i',j'}$ between u_1 and $S_{i,j}$, then the sides $\overline{u_2u_4}$ and $\overline{u_3u_4}$ must intersect any square $S_{i'',j''}$. This contradicts Lemma 4.

To complete the proof of property (1), we still need to show that $v(i, j)$ is in the particular region $D(i, j)$. We show this together with property (2). We prove first that two adjacent vertices u_1 and u_2 cannot be placed in the same region $D(\hat{\imath}, \hat{\jmath})$. Let F be the face of vertices u_1, u_2, u_3, and u_4, which encloses some $S_{i,j}$. Firstly, we can rule out the case that $S_{i,j}$ is not adjacent to $D(\hat{\imath}, \hat{\jmath})$ as there is obviously no way to construct F in such case. Among different analog cases, we analyze the situation where $i = \hat{\imath}+1$ and $j = \hat{\jmath}+1$. Placing u_1 and u_2 in optimal way results in $u_1 = (8i-6, 8j-9)$ and $u_2 = (8i-9, 8j-6)$. To determine the best position of u_3 and u_4, we pass a straight line through the corners $(8i-1, 8j-7)$ and $(8i-7, 8j-1)$ of $S_{i,j}$. The intersection of these lines and the next respective square describes the optimal position of u_3 and u_4, i. e. $u_3 = (8i-6.2, 8j+1)$ and $u_4 = (8i+1, 8j-6.2)$. Analyzing the edge between u_3 and u_4 reveals that the top right corner of $S_{i,j}$ cannot be included into F, and thus u_1 and u_2 cannot be in the same region.

Now we can establish the position of all vertices on the boundary of R. The vertices at the corner points of R have a fixed position in $(0, 0), (0, 8n)(8n, 0)$, or $(8n, 8n)$ respectively and thereby they belong to $D(0, 0), D(0, n), D(n, 0)$, and $D(n, n)$ respectively. For each edge of R, there are $n-1$ vertices to be placed on the boundary between the first and the last vertex of their boundary. Furthermore, there are exactly $n-1$ regions along the boundary in which they can be placed. Due to the planarity of the mesh, all vertices of the boundary must be placed according to $v(i,j) \in D(i,j)$.

The proof for the remaining vertices and the faces is done by induction. A face $F(i,j)$ must enclose a square $S_{i',j'}$ and the positions of $v(i-1,j-1), v(i-1,j)$, and $v(i,j-1)$ are restricted to $D(i-1,j-1), D(i-1,j)$, and $D(i,j-1)$, respectively. According to the induction hypothesis, we know either $i' > i$ or $i' = i \wedge j' \geq j$. If $F(i,j)$ would not enclose $S_{i,j}$, the vertex $v(i,j)$ would be placed close to $v(i-1,j-1)$ to exclude the weight at $(8i-7, 8j-7)$. Hence, no other square can be enclosed in $F(i,j)$. By including $S_{i,j}$, the position of $v(i,j)$ must be in $D(i,j)$. □

By combining Lemma 4 and 5 we get the crucial property of point instance P: if $d_\infty(P,G) \leq t$ then any face $F(i,j)$ of G has to enclose the square $S_{i,j}$.

To see that it is actually possible to provide an $n \times n$ q-grid G with $d_\infty(P,G) \leq t$, the vertex $v(i,j)$ can be placed at $(8i, 8j)$. Beyond that, any positioning of $v(i,j)$ such that the Manhattan distance to $(8i, 8j)$ is less than 1, currently results in a q-grid G with $d_\infty(P,G) = 0$.

3.2 Embedding of the Boolean Formula

As the shape of the q-grid partitioning is sufficiently limited, we can now describe the embedding of the formula φ into the points instance. To this aim, to the set Q of weighted points constructed in the first part of the proof, we add further points and define their weights in an appropriate way.

We will ensure that the final points instance $P = (Q, \omega)$ has mean $\mu = \gamma + 4$. Thus, the additional points will make up $4n^2$. To keep the shape of q-grid partitionings G as before, we still require that $d_\infty(P,G) \leq t = \alpha/2$. Thus, the foundation of the choice of $\alpha = 16n^2$ is to ensure that new points do not invalidate the inequality. Finally, the threshold to determine whether φ is satisfiable or not will be $c = 1$. According to the objective function used, all faces must contain points of weight $\mu \pm 1$ to conform to c.

In the following, we describe the new weighted point set as a two dimensional array of fields. A *field* i,j corresponds to a square $S_{i,j}$. Thereby we get an array of $n \times n$ fields. By Lemma 4 a field i,j is always completely included in the face $F(i,j)$.

Some of the fields, called *active*, will be directly used to embed the formula φ. The remaining fields will play a passive role in the reduction and we will call them *inactive*. To encode an inactive field, it gets an additional weight equals $4-\epsilon$, where $0 < \epsilon < 1$, which will be achieved by adding a weighted point in the interior of

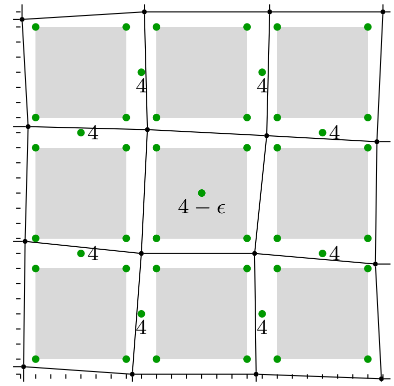


Figure 7: Clip of a possible gadget of a variable x_i, containing a cycle of eight fields and an inactive field in the center. The q-grid shows a negative configuration of the gadget, i. e. $\beta(x_i) = 0$.

the square. Thus, such inactive fields will not differ from μ by more than 1. The adjustment ϵ will provide the desired value of μ and thereby a certain among of inactive fields must be present. One can determine ϵ, or equivalently the necessary number of inactive fields, through

$$\epsilon = \frac{2 \cdot \text{number of clauses}}{\text{number of inactive fields}}.$$

Any field which is a component of one of the gadgets described below is active. Typically, for active fields no additional weighted points will be placed in the interior of their respective squares. However, in some cases points of weight 1 will be added.

Two adjacent (active) fields can be *connected*. Therefore, an additional weighted point is placed right in the middle between the two adjacent fields, such that this point can be included in either of the two respective faces. We will call these points *connection points*.

Connected fields can form a *path*. A path is a sequence of connected fields where the fields have no additional weight but all connection points are of weight 4. Paths must not intersect or overlap each other.

A variable x_i of φ will be embedded as such a path by forming a closed cycle. Considering a q-grid which gives a partition of objective value at most 1, the cycle has exactly two possible configurations. Either the connection points are all included into the next face of the cycle in clockwise order or they belong to the next face in counter-clockwise order. We interpret a clockwise orientation as assignment $\beta(x_i) = 1$ while a counter-clockwise orientation as $\beta(x_i) = 0$. An example of a variable and its configuration is shown in Fig. 7.

Next, we describe the gadget of a literal. A literal is implemented as a path branching of the cycle representing the respective variable. A branch consists of two consecutive fields on a path. Their connection point has only weight 2, but the second field has an additional weight 1. Furthermore, the first field is adjacent to the branching path. They are also connected by a connection point of weight 2 and the first field of the path has an additional weight 1. A positive literal branches off in clockwise direction while a negative literal branches off in counter-clockwise direction. The path of a literal leads to the gadget of the clause it is part of. Considering a q-grid G of objective value $\leq c$, the branched path has two possible configurations: the first, we call *positive*, means that the connection point always belongs to the next field on the path. The negative configuration indicates that the connection point always belongs to the previous field on the path. The configuration of a path is determined by the configuration of the cycle it is connected to. A clockwise branch is positive if and only if the cycle is clockwise orientated. The opposite is true for counter-clockwise branches.

The gadget of a clause is a field where exactly three paths representing three literals end. All three are connected to the field by a connection point of weight 4. Considering a q-grid which has objective value $\leq c$, the clause has exactly three possible configurations, each with exactly one incoming path positive. These configurations exactly represent the three valid assignments of one-in-three satisfiability. Thus, there exists a q-grid G of objective value $\leq c$ if and only if φ is one-in-three satisfiable. Figure 8 illustrates the example formula φ. Fields are displayed as squares with their respective additional weights in the inside. Connections between fields are symbolized by edges between the squares. The weights of the connection points are placed on the edge.

Due to the planarity of φ, there is always such a planar embedding of φ in the described way. The time complexity of the reduction is dominated by the size of the resulting points instance. The representation of the points instance and its weights is polynomial in n and thereby polynomial in $|\varphi|$. As the reduction has a polynomial time complexity, the NP-hardness follows.

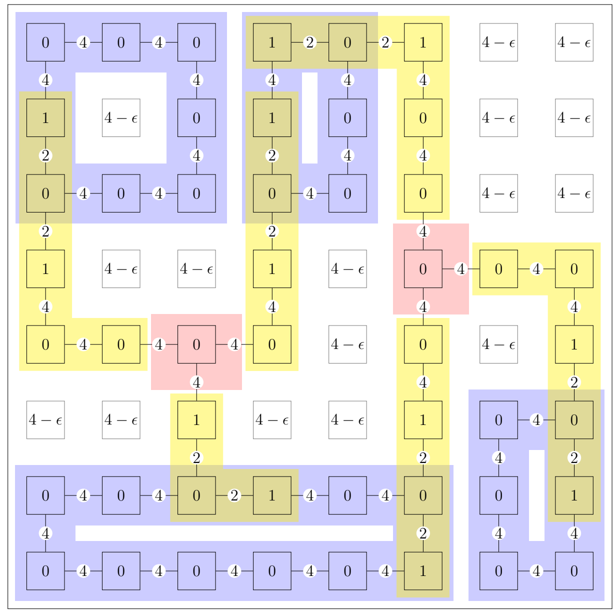
This completes the proof of Theorem 1. □

3.3 Other Objective Functions

The proofs of NP-hardness for other objective functions can be obtained similarly. Primarily, the values of α, t, and c in the proof must be chosen appropriately.

Corollary 6 *d-QGP is NP-hard for any d_p function and for $d_{\max}$.*

From the proof it follows that the problems are NP-hard even if (1) the weights of the points are integers, (2) the points of the weighted point set have integer coordinates, or (3) the q-grid is restricted to convex faces.



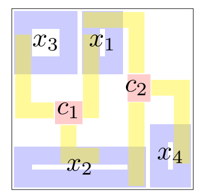


Figure 8: Possible reduction of φ defined in eq. (1), for $n = 8$. Gadgets are additionally highlighted as follows: blue is the cycle of a variable, yellow is the literal including the branching, and red is a clause gadget. The mapping of the gadgets is shown to the right.

4 Conclusions and Open Problems

For ease of notation, the definition of the q-grid partitioning problem was limited to two dimensions. Nevertheless, the partitioning can be extended to higher dimensions in which case the problem remains NP-hard.

We have proven that the decision versions of the problems are hard for NP but we leave as an open question if they are in NP. The main difficulty here is to show that for any instance there exists an optimal q-grid partitioning G of vertices with coordinates of polynomial-size. A natural approach to solve this issue is to express d-QGP as a quadratic program in a form for which the quadratic programming problem is known to belong to NP (like e.g. [15, 19]). Unfortunately, a straightforward formulation leads to programs having quadratic constraints for which we do not know if the program has a basic optimum of polynomial size.

References

[1] N. Bassiou and C. Kotropoulos. Color image histogram equalization by absolute discounting back-off. *Comp. Vision and Image Understanding*, 107(1-2):108–122, 2007.

[2] S.-K. Chang and Y. Wong. Ln norm optimal histogram matching and application to similarity retrieval. *Com-*

puter Graphics and Image Processing, 13(4):361 – 371, 1980.

[3] S.-K. Chang and Y.-W. Wong. Optimal histogram matching by monotone gray level transformation. *Communications of the ACM*, 21(10):835–840, Oct. 1978.

[4] W.-M. Chow and L. T. Kou. *Matching two digital pictures.* IBM Res. Rep. RC6870, IBM T.J. Watson Res., 1977.

[5] J. Duan and G. Qiu. Novel histogram processing for colour image enhancement. In *Multi-Agent Security and Survivability, 2004 IEEE First Symposium on*, pages 55–58. IEEE, 2004.

[6] W. S. Evans, S. Felsner, M. Kaufmann, S. G. Kobourov, D. Mondal, R. I. Nishat, and K. Verbeek. Table cartograms. In *ESA*, pages 421–432, 2013.

[7] R. C. Gonzalez and R. E. Woods. *Digital image processing.* Addison-Wesley, Upper Saddle River, New Jersey, third edition edition, 2008.

[8] M. Grigni and F. Manne. On the complexity of the generalized block distribution. In *PAISP*, volume 1117 of *LNCS*, pages 319–326. Springer, 1996.

[9] J.-H. Han, S. Yang, and B.-U. Lee. A novel 3-d color histogram equalization method with uniform 1-d gray scale histogram. *Image Processing, IEEE Transactions on*, 20(2):506–512, Feb 2011.

[10] A. Idelberger. Multidimensional histogram equalization: Modelling and computational analysis (in German). Bachelor's thesis, Universität zu Lübeck, Germany, 2013.

[11] S. Kundu. A solution to histogram-equalization and other related problems by shortest path methods. *Pattern Recognition*, 31(3):231 – 234, 1998.

[12] S. Kundu. Private email communication. 28-th March 2012, March 2012.

[13] P. Laroche. Satisfiabilité de 1-parmi-3 planaire est np-complet. *Comptes rendus de l'Académie des sciences. Série 1, Mathématique*, 316(4):389–392, 1993.

[14] P. A. Mlsna and J. J. Rodriguez. A multivariate contrast enhancement technique for multispectral images. *Geoscience and Remote Sensing, IEEE Transactions on*, 33(1):212–216, 1995.

[15] K. G. Murty and S. N. Kabadi. Some np-complete problems in quadratic and nonlinear programming. *Mathematical programming*, 39(2):117–129, 1987.

[16] E. Pichon, M. Niethammer, and G. Sapiro. Color histogram equalization through mesh deformation. In *Int. Conf. on Image Proc.*, pages II – 117–20 vol.3, sept 2003.

[17] Y. Shiloach. *Arrangements of Planar Graphs on the Planar Lattices.* PhD thesis, Weizmann Institute of Science, Rehovot, Israel, 1976.

[18] P. Trahanias and A. Venetsanopoulos. Color image enhancement through 3-d histogram equalization. In *Pattern Recognition, 1992. Vol. III. Conference C: Image, Speech and Signal Analysis, Proceedings., 11th IAPR International Conference on*, pages 545–548. IEEE, 1992.

[19] S. A. Vavasis. Quadratic programming is in np. *Inf. Process. Lett.*, 36(2):73–77, 1990.