Convex Aggregation Problems in Z^2

Yan Gérard

To cite this version:

Yan Gérard. Convex Aggregation Problems in Z^2. 21st IAPR International Conference, DGCI 2019,, Mar 2019, Marne la Vallée, France. 10.1007/978-3-030-14085-4
$_34.hal-02091208$

Convex Aggregation Problems in $\mathbb{Z}^2$

Yan Gérard[1]

Université Clermont Auvergne - LIMOS, Clermont-Ferrand, France

Abstract. We introduce a family of combinatorial problems of digital geometry that we call *convex aggregation* problems. Two variants are considered. In *Unary* convex aggregation problems, a first lattice set $A \subseteq \mathbb{Z}^d$ called *support* and a family of lattice sets $B^i \subseteq \mathbb{Z}^d$ called *pads* are given. The question to determine whether there exists a non-empty subset of pads (the set of their indices is denoted I) whose union $A \cup_{i \in I} B^i$ with the support is convex. In the *binary* convex aggregation problem, the input contains the support set $A \subseteq \mathbb{Z}^2$ and pairs of pads B^i and $\overline{B}^i$. The question is to aggregate to the support either a pad B^i or its correspondent $\overline{B}^i$ so that the union $A \cup_{i \in I} B^i \cup_{i \notin I} \overline{B}^i$ is convex.
We provide a first classification of the classes of complexities of these two problems in dimension 2 under different assumptions: if the support is 8-connected and the pads included in its enclosing rectangle, if the pads are all disjoint, if they intersect and at least according to the chosen kind of convexity. In the polynomial cases, the algorithms are based on a reduction to Horn-SAT while in the NP-complete cases, we reduce 3-SAT to an instance of convex aggregation.

Keywords: Digital Geometry, Discrete Tomography, Convexity, Complexity, Horn-SAT

1 Introduction

The first convex aggregation problem has been considered in 1996 in Discrete Tomography where it has been used for the reconstruction of $\mathcal{HV}$-convex polyominoes i.e $\mathcal{HV}$-convex 4-connected lattice sets [1]. More generally, convex aggregation problems are the bottleneck of the reconstruction of convex lattice sets in different problems of Discrete Tomography [2–4]. Although they might be of major interest for the field, this class of combinatorial problems have never been neither formulated independently nor investigated in a rather exhaustive manner. The first parameter of this family of problems is the chosen kind of convexity. In the lattice $\mathbb{Z}^2$, the two main ones are denoted here $\mathcal{HV}$ and $\mathcal{C}$-convexity. A lattice set $S \subseteq \mathbb{Z}^2$ is $\mathcal{HV}$-convex if its intersection with any horizontal or vertical line is a set of consecutive points and $S \subset \mathbb{Z}^d$ is $\mathcal{C}$-convex if it is equal to its intersection with its real convex hull $S = \mathrm{conv}_{\mathbb{R}^d}(S) \cap \mathbb{Z}^d$ (Fig. 1). The $\mathcal{HV}$-convexity is a directional convexity reduced to the horizontal and vertical directions. The $\mathcal{C}$-convexity is the classical convexity of geometry of numbers. The $\mathcal{C}$-convex lattice sets are the lattice polytopes.

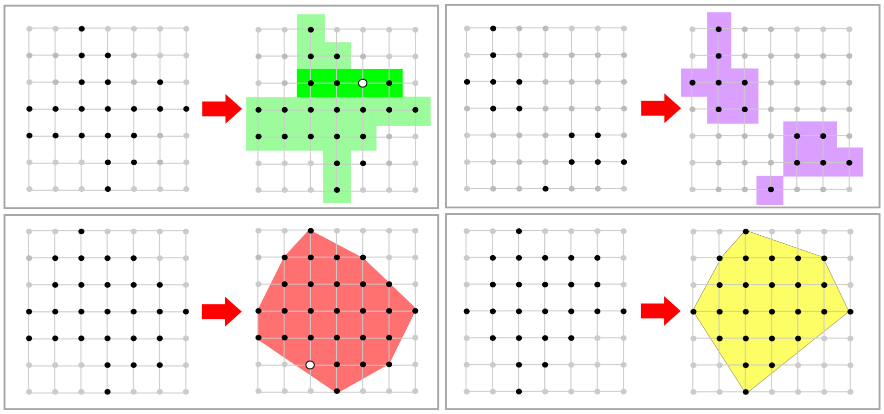

Fig. 1. ***$\mathcal{HV}$* and *$\mathcal{C}$*-convexities**. Top left, the set is not $\mathcal{HV}$-convex while the top-right, the lattice set is $\mathcal{HV}$-convex. Below, the left lattice set is not $\mathcal{C}$-convex while the right set is $\mathcal{C}$-convex. Notice that neither $\mathcal{HV}$-convexity, nor $\mathcal{C}$-convexity imply digital connectivity.

In Discrete Tomography, the problem of convex aggregation appears as follows: a partial convex solution $A \subseteq \mathbb{Z}^2$ and a list of conjugate subsets $B^i, \overline{B}^i$ of $\mathbb{Z}^2$ with an index i from 1 to n have been computed. In the sequel, the subset A is called the *support* while the subsets $B^i \subseteq \mathbb{Z}^2$ and $\overline{B}^i \subseteq \mathbb{Z}^2$ are called *pads*. The problem is to aggregate either the pad B^i or its conjugate $\overline{B}^i$ to the support so that the union of the support with all chosen pad is convex. In other words, the problem is to find a list of indices I so that the union $A \cup_{i \in I} B^i \cup_{i \notin I} \overline{B}^i$ is convex. These problems of convex aggregation are called *binary*. We choose this term to highlight the difference with another class of problems that we call *unary*. In unary convex aggregation problems, the input is a support set $A \subseteq \mathbb{Z}^2$ and a family of pads $B^i \subseteq \mathbb{Z}^2$. The problem is to find a non-empty set of pads whose union $A \cup_{i \in I} B^i$ with the support is convex.

The purpose of the paper is to determine the classes of complexities of these two problems by considering either the $\mathcal{HV}$ or the $\mathcal{C}$-convexity in the general case and under several assumptions on the pads and the support.

In the next section (Sect. 2), we start by stating formally the problems. Sect. 3 provides a very brief state of the art and the results. Some complexities are shown to be polynomial and others NP-complete. The polynomial-time algorithms are given in Sect. 4 while Sect. 5 provides sketches of proofs of NP-completeness.

2 Problem statement

For reducing the number of notations, we consider the chosen convexity as a parameter denoted convexity. We introduce first the problem of unary convex aggregation $1\mathtt{CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n})$ and secondly its binary variant $2\mathtt{CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n}, (\overline{B}^i)_{1\leq i\leq n})$.

Problem 1. $1\mathtt{CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n})$
Input: - a finite lattice set $A \subseteq \mathbb{Z}^d$ that we call *support*,
- a family of finite lattice sets $B^i \subseteq \mathbb{Z}^d$ that we call *pads*, with an index i going from 1 to n.
Output: Does there exist a non-empty set of indices $I \subseteq [1 .. n]$ such that the union $A \cup_{i\in I} B^i$ of the pads B^i with $i \in I$ and A is convex (Fig. 2)?

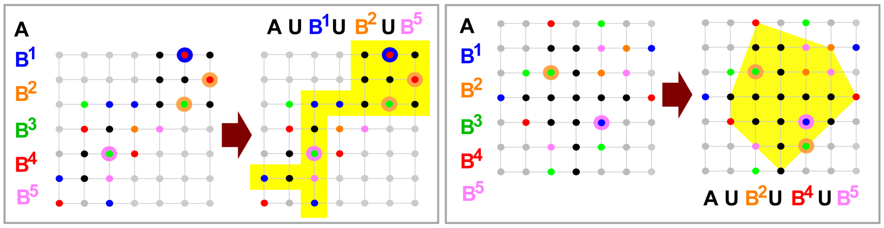

Fig. 2. Two instances of $1\mathtt{CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n})$ **and their solutions**. The input is a set A (black points) with pads B^i (each pad is a set of colored points) while a solution is a non-empty set of indices I such that the union $A \cup_{i\in I} B^i$ is convex. On the left, we consider the $\mathcal{HV}$-convexity and the $\mathcal{C}$-convexity on the right.

Notice that the support A from an instance can be removed by adding it to all the pads: the instance $1\mathtt{CA}_{\text{convexity}}(\emptyset, (A \cup B^i)_{1\leq i\leq n})$ is equivalent with $1\mathtt{CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n})$.

We introduce now the second class of problems of convex aggregation occuring in Discrete Tomography and we call them *binary*.

Problem 2. $2\mathtt{CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n}, (\overline{B}^i)_{1\leq i\leq n})$
Input: - a finite lattice set $A \subseteq \mathbb{Z}^d$ that we call *support*,
- a family of pairs of finite lattice sets $B^i \subseteq \mathbb{Z}^2$ and $\overline{B}^i \subseteq \mathbb{Z}^2$ that we call *pads*, with an index i going from 1 to n.
Output: Does there exist a set of indices $I \subseteq [1 .. n]$ such that the union $A \cup_{i\in I} B^i \cup_{i\notin I} \overline{B}^i$ is convex (Fig. 3)?

Note that with a non-convex support A, the unary problem $1\mathtt{CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n})$ is a particular case of $2\mathtt{CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n}), (\overline{B}^i)_{1\leq i\leq n})$ with empty conjugate pads $\overline{B}^i$. It follows that the complexity of the binary problems is necessarily at least the complexity of the unary cases.

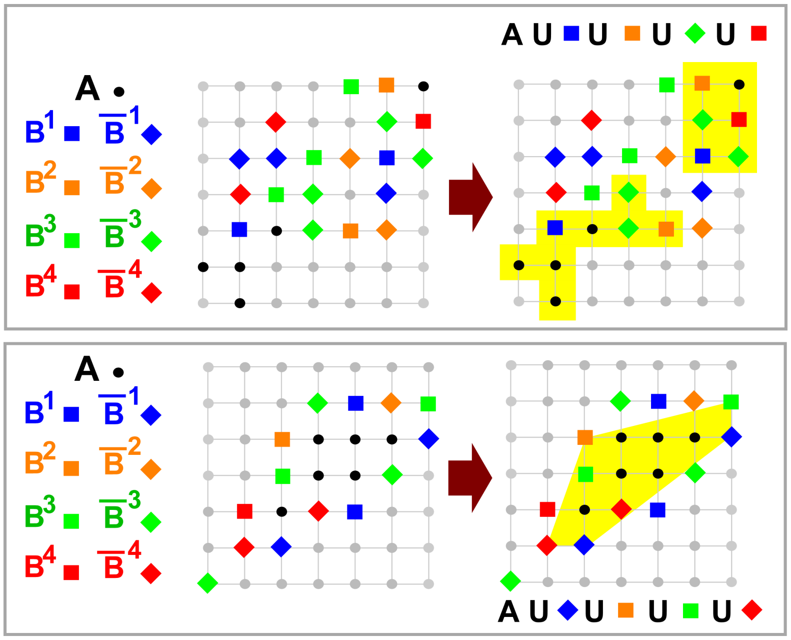

Fig. 3. Two instances of $2\mathtt{CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n}, (\overline{B}^i)_{1\leq i\leq n})$ **and their solutions**. The input is a set A (black points) and for each index i, a pair of conjugate pads B^i and $\overline{B}^i$. The points of the pads B^i are represented by squares while the points of their conjugate $\overline{B}^i$ are represented by diamonds, with a different color for each index. The problem of binary convex aggregation is to aggregate to the support A either the squares or the diamonds of each color, so that the union $A \cup_{i\in I} B^i \cup_{i\notin I} \overline{B}^i$ becomes convex. The instances are drawn on the left (above with $\mathcal{HV}$-convexity, below with $\mathcal{C}$-convexity) and a solution is given on the right.

3 State of the Art and Results

The state of the art is mainly reduced to the following result known in Discrete Tomography. It requires to define the *$\mathcal{HV}$-grid* of a finite lattice set $A \subseteq \mathbb{Z}^2$ as the finite grid $[\min(x(A)) \mathinner{..} \max(x(A))] \times [\min(y(A)) \mathinner{..} \max(y(A))] \subseteq \mathbb{Z}^2$ where x and y are the two coordinates of $\mathbb{Z}^2$. We recall also that a lattice set A of $\mathbb{Z}^2$ is 8-*connected* if any pair of points $(a, a') \in A^2$ is connected by a discrete path $(a_i)_{1\leq i\leq n}$ of points of A verifying $a_1 = a$, $a_n = a'$ and $\forall i \in [1 \mathinner{..} n-1], d_2(a_i, a_{i+1}) \leq 1$ where d_2 denotes the Euclidean distance.

Theorem 1. *If A is 8-connected, and all the pads are disjoint and included in the $\mathcal{HV}$-grid of A, $2\mathtt{CA}_{\mathcal{HV}}(A, (B^i)_{1\leq i\leq n}, (\overline{B}^i)_{1\leq i\leq n})$ can be solved in polynomial time [1].*
If A is 8-connected, and all the pads are included in the $\mathcal{HV}$-grid of A, $1\mathtt{CA}_{\mathcal{HV}}(A, (B^i)_{1\leq i\leq n})$ can be solved in polynomial time.

The original proof is stated in the binary case with disjoint pads [1]. It is based on a simple idea: encode the choice to add either the pad B^i or $\overline{B}^i$ by a Boolean variable that we denote $\mathfrak{b}_i$ (with the convention that $\mathfrak{b}_i = 1$ if B^i is added to A and then $i \in I$ while $\mathfrak{b}_i = 0$ if $i \notin I$). Then, with disjoint pads, the $\mathcal{HV}$-convexity of the union of the pads is encoded by a conjunction of 2-clauses. It reduces the problem of convex aggregation to an instance of 2-SAT and allows us to solve it in polynomial time [5].

The result does not hold without the assumption of disjoint pads. The approach of [1] remains valid for unary $\mathcal{HV}$-convex aggregation with disjoint and non-disjoint pads under the assumption that the support A is 8-connected and the pads included in the $\mathcal{HV}$-grid of A. The $\mathcal{HV}$-convexity of a solution is encoded by a Horn-SAT instance and can be solved in polynomial time [6].

On the other hand, different authors have noticed that the same problem obtained by replacing the $\mathcal{HV}$-convexity by the $\mathcal{C}$-convexity is more difficult [3, 7]. The constraint of $\mathcal{C}$-convexity can be formulated with 3-clauses. 3-SAT being NP-complete, this approach does not provide a solution in polynomial time. Does it mean that the problem of convex aggregation is itself NP-complete? It is one of the questions that we investigate in what follows.

We determine in the paper a first classification of the complexities of the problems of convex aggregation.

Theorem 2. *The classes of complexities of the problems of convex aggregation under several assumptions are given in Tab. 1.*

Problem	Unary				Binary			
	8-connected support with pads in the $\mathcal{HV}$-grid of A				8-connected support with pads in the $\mathcal{HV}$-grid of A			
	disjoint pads		disjoint pads		disjoint pads		disjoint pads	
$\mathcal{HV}$-convexity	P (Th. 1)	P (Th. 1)	P	NP	P (Th. 1)	NP	NP	NP
$\mathcal{C}$-convexity	P	NP	P	NP	NP	NP	NP	NP

Table 1. Complexities of unary and binary convex aggregation problems with either $\mathcal{HV}$ or $\mathcal{C}$-convexity, with or without two different assumptions. We can assume that the support is 8-connected and the pads included in the $\mathcal{HV}$-grid of A and secondly, we can assume that the pads are disjoint. In each cell, *P* means polynomial-time and *NP* is written for NP-complete.

In the case of binary convex aggregation, the only polynomial case among the configurations that we investigate is the one of Theorem 1. All the others are NP-complete. In particular, the condition that the pads are disjoint or the 8-connectivity of the support is not sufficient to provide a polynomial time algorithm.

We do not provide here polynomial time algorithms for binary $\mathcal{C}$-convex aggregation but only results of NP-completeness. With specific properties of the pads, such polynomial-time algorithms might be obtained [7] but these assumptions used in Discrete Tomography are too specific to fit into the scope of this general classification. The reader should just keep in mind that some assumptions on the pads might lead to polynomial time algorithms. We conjecture in particular that the problem $\mathtt{2CA}_{\text{convexity}}(A,(B^i)_{1\leq i\leq n},(\overline{B}^i)_{1\leq i\leq n})$ is polynomial with convex disjoint pads.

4 Polynomial time algorithms

Theorem 2 states that with disjoint pads, $\mathtt{1CA}_{\text{convexity}}$ $(A,(B^i)_{1\leq i\leq n})$ can be solved in polynomial in the two cases of convexity (convexity $=\mathcal{HV}$ is the case 1 and convexity $=\mathcal{C}$ in the case 2). The purpose of this section is to describe the two algorithms.

According to the same strategy as the proof of Theorem 1, we encode the choice to add or not the pad B^i to A by a Boolean variable. By convention, $\mathfrak{b}_i$ is equal to 1 if i is in the set I (in other words, if the pad B^i is added to the union) and $\mathfrak{b}_i=0$ if it is excluded. The main idea of the two algorithms is to express the convexity of the union $A\cup_{I\in I}B^i$ by a conjunction of clauses.

In the following, a point of the lattice $\mathbb{Z}^2$ is said *covered* if it is in the union $A\cup_{1\leq i\leq n}B^i$ of the pads and the support and *uncovered* otherwise. In terms of data structure, it is practical to build two tables. The first one contains the abscissas of the covered points and for each abscissa, the ordered list of their ordinates with the index of the pad containing the point. The second table uses first the ordinates and secondly, the abscissas. Denoting the number of pads n and the total number of points $N=|A|+\sum_{i=1}^{n}|B^i|$, the two tables can be computed in $O(N\log(N))$ time, with a storage in $O(N)$.

4.1 Case 1 - with $\mathcal{HV}$-convexity and disjoint pads

We consider the problem $\mathtt{1CA}_{\mathcal{HV}}(A,(B^i)_{1\leq i\leq n})$ with disjoint pads. We provide an algorithm in $O(N^3+n^4)$. Its strategy is to build a conjunction of clauses characterizing the $\mathcal{HV}$-convexity of the union $A\cup_{I\in I}B^i$:

1. For all pairs of points $(a,b)\in\cup_{1\leq i\leq n}A\times B^i$ with a and b on the same row or on the same column, we consider the integer points in the segment of end points a and b. If there is an uncovered point, the pad B^i can be directly excluded since its union with the support A cannot be convex (there is a hole in between). Otherwise, we build a clause $\mathfrak{b}_i \implies \mathfrak{b}_j$ for all the pads

b^j of the points on the segment. It expresses the constraint that the points in-between have to be included in the union.

2. For all pairs of points $(b, b') \in \cup_{1\leq i\leq j\leq n} B^i \times B^j$ with b and b' on the same row or on the same column, we consider the integer points in-between. If there is an uncovered point, we build the clause $\mathfrak{b}_i \implies \overline{\mathfrak{b}}_j$. It expresses that the two pads cannot be chosen simultaneously. Otherwise, for all the pads B^k having points between b and b', we add the clause $\mathfrak{b}_i \wedge \mathfrak{b}_j \implies \mathfrak{b}_k$ (notice that we use here the assumption of disjoint pads. Otherwise the clause could be $\mathfrak{b}_i \wedge \mathfrak{b}_j \implies \mathfrak{b}_k \vee \mathfrak{b}_{k'}$ if the pads B^k and $B^{k'}$ have for instance a common point between b and b').
3. We remove the redundant clauses. The number of clauses becomes $O(n^3)$.

The clauses built with this algorithm guarantee the $\mathcal{HV}$-convexity of the corresponding union $A \cup_{1\leq i\leq n} B^i$. Conversely, an $\mathcal{HV}$-convex union $A \cup_{i\in I} B^i$ provides a solution of the instance of satisfiability. In the case where the support A is $\mathcal{HV}$-convex, a null solution can satisfy the clauses but the empty set of indices $I = \emptyset$ is not a valid solution of $\mathtt{1CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n})$. This case apart (we consider it afterwards), finding a non-null solution of the conjunction of clauses is equivalent to the problem of convex aggregation $\mathtt{1CA}_{\text{convexity}}(A, (B^i)_{1\leq i\leq n})$. The clauses are either of the form $\mathfrak{b}_i \implies \mathfrak{b}_j$, $\mathfrak{b}_i \implies \overline{\mathfrak{b}}_j$ or $\mathfrak{b}_i \wedge \mathfrak{b}_j \implies \mathfrak{b}_k$. They can respectively be rewritten as $\overline{\mathfrak{b}}_i \vee \mathfrak{b}_j$, $\overline{\mathfrak{b}}_i \vee \overline{\mathfrak{b}}_i$ and $\overline{\mathfrak{b}}_i \vee \overline{\mathfrak{b}}_j \vee \mathfrak{b}_k$. They have at most one positive literal. It makes them Horn clauses and Horn-SAT can be solved in linear time [6]. The condition of non-nullity can be written as $\vee_{1\leq i\leq n}\mathfrak{b}_i$ and it is not a Horn clause. A naive way to search for a non-null solution of the Horn-SAT instance is to fix successively $\mathfrak{b}_i = 1$ and to solve the new SAT instance, now in linear time. If $\mathfrak{b}_1 = 1$ does not provide a solution, try $\mathfrak{b}_2 = 1$ and so on until $\mathfrak{b}_n = 1$ if no solution has been found previously.

Complexity

The number of pairs of points considered in the algorithm is bounded by N^2. Then it generates at most a cubic number of clauses $O(N^3)$ in a cubic time $O(N^3)$. Their number is reduced to $O(n^3)$ by removing the redundant clauses in $O(N^3)$ time. The resolution of the n instances of the main Horn-SAT instance obtained by fixing $\mathfrak{b}_i = 1$ requires at most n times $O(n^3)$, namely $O(n^4)$. It provides an algorithm with an overall complexity in $O(N^3 + n^4)$.

4.2 Case 2 - $\mathcal{C}$-convexity with disjoint pads

We consider now the problem $\mathtt{1CA}_{\mathcal{C}}(A, (B^i)_{1\leq i\leq n})$ with disjoint pads and provide an algorithm in time $O(N^4 + n^5)$. Its strategy is similar to the algorithm given in the previous subsection. In order to reduce the number of cases, we introduce a supplementary Boolean variable $\mathfrak{b}_0$ associated with the support A (its value will be fixed to 1 after the generation of all clauses). We build an instance of satisfiability expressing the $\mathcal{C}$-convexity of $A \cup_{i\in I} B^i$ with the following process:

For all triplets of points $(a, b, c) \in (A \cup_{1\leq i\leq n} B^i)^3$, we consider the triangle $T_{a,b,c}$ with vertices a, b and c. The indices of the pads of the points a, b and c

are denoted i, j and k. If one of the points of $T_{a,b,c}$ is uncovered, then the three variables cannot be true in the same time. It is expressed by the conjunction $\overline{\mathfrak{b}}_i \vee \overline{\mathfrak{b}}_j \vee \overline{\mathfrak{b}}_k$. If all the points of the triangle are covered, we add the clause $\mathfrak{b}_i \wedge \mathfrak{b}_j \wedge \mathfrak{b}_k \implies \mathfrak{b}_q$ for all the indices q of the pads of the points in the triangle $T_{a,b,c}$. This clause can be rewritten as $\overline{\mathfrak{b}}_i \vee \overline{\mathfrak{b}}_j \vee \overline{\mathfrak{b}}_k \vee \mathfrak{b}_q$. They are both Horn clauses. Once all triangles have been considered, we fix $b_0 = 1$ and remove the redundant clauses. Their final number becomes bounded by $O(n^4)$.

According to Carathéodory's theorem in dimension 2, the convex hull of a finite set S is the union of the triangles with vertices in S [8]. It follows that S is convex iff no triangle with vertices in S has a point outside S. It explains that the 3-clauses and 4-clauses obtained from all the triangles with vertices in $A \cup_{1 \leq i \leq n} B^i$ guarantee the $\mathcal{C}$-convexity of the union $A \cup_{\in I} B^i$. The $\mathcal{C}$-convexity is expressed by the clauses. As in the previous case, the constraint to find a non-empty union of pads $\cup_{i \in I} B^i$ has to taken into account. If the support A is convex, we consider n sub-instances of the Horn-SAT instance with $\mathfrak{b}_i = 1$ for all indices from 1 to n. By construction, this sequence of problems is equivalent with $\mathtt{1CA}_{\mathcal{C}}(A, (B^i)_{1 \leq i \leq n})$ and it can be solved in polynomial time.

Complexity

The number of triplets of points considered by the algorithm is bounded by N^3. We have at most a quartic number of clauses $O(N^4)$ generated in a quartic time $O(N^4)$. Their number is reduced to $O(n^4)$ by removing the redundant clauses in $O(N^4)$ time. The resolution of the n instances of the main Horn-SAT instance with $\mathfrak{b}_i = 1$ requires at most n times $O(n^4)$, namely $O(n^5)$. It provides an algorithm with an overall complexity in $O(N^4 + n^5)$.

5 Proofs of NP-completeness

We prove here that the convex aggregation problems are NP-complete in the five following cases:

1. $\mathtt{1CA}_{\mathcal{HV}}(A, (B^i)_{1 \leq i \leq n})$ where convexity $= \mathcal{HV}$.
2. $\mathtt{1CA}_{\mathcal{C}}(A, (B^i)_{1 \leq i \leq n})$ where convexity $= \mathcal{C}$.
3. $\mathtt{2CA}_{\mathcal{HV}}(A, (B^i)_{1 \leq i \leq n}, (\overline{B}^i)_{1 \leq i \leq n})$ with a 8-connected support and pads included in the $\mathcal{HV}$-grid of A.
4. $\mathtt{2CA}_{\mathcal{HV}}(A, (B^i)_{1 \leq i \leq n}, (\overline{B}^i)_{1 \leq i \leq n})$ with disjoint pads.
5. $\mathtt{2CA}_{\mathcal{C}}(A, (B^i)_{1 \leq i \leq n}, (\overline{B}^i)_{1 \leq i \leq n})$ with disjoint pads.

For any of these five problems, given a set of indices I, we can test in polynomial time whether I is a solution of the instance. It follows that all these problems are in NP. Secondly, we have to provide a polynomial time reduction of an instance of a known NP-complete problem to each one of these problems. The five proofs follow the same strategy: reduce an instance of 3-SAT.

We assume that we have an instance of 3-SAT with n Boolean variables $(\mathfrak{b}_i)_{1 \leq i \leq n} \in \{0,1\}^n$ and N 3-clauses. The goal is to build an instance of convex aggregation which admits a solution if and only if the 3-SAT instance is feasible. The strategy is to use pads B^i to represent literals. In the case of unary convex aggregation, we use the pad B^i of index i to represent the literal $\mathfrak{b}_i$ and the pad B^{i+n} of index $i+n$ to represent the literal $\overline{\mathfrak{b}}_i$. Therefore, we use $2n$ pads for representing n variables. In the case of binary convex aggregation, the pad B^i represents the literal $\mathfrak{b}_i$ while the conjugate pad $\overline{B}^i$ represents its negation $\overline{\mathfrak{b}}_i$. According to this strategy, the purpose of each proof is to provide a geometrical structure where the constraint of convex aggregation encodes, first, the relations of negation between two literals and secondly, the relations given by the 3-clauses.

5.1 Case 3 - $\mathcal{HV}$-convex binary aggregation with non-disjoint pads

We show how to reduce a 3-SAT instance with n Boolean variables and c clauses to an equivalent instance $2\mathtt{CA}_{\mathcal{HV}}(A,(B^i)_{1\leq i\leq n},(\overline{B}^i)_{1\leq i\leq n})$ of polynomial size. The structure is made by a support of size $(2c+1)\times(c+1)$. Its upper points can be used to encode the clauses with Horizontal convexity (Fig. 4). The equivalence between the instance of 3-SAT and the corresponding instance of $2\mathtt{CA}_{\mathcal{HV}}(A,(B^i)_{1\leq i\leq n},(\overline{B}^i)_{1\leq i\leq n})$ is a consequence of the rewriting of the clauses as implications before their encoding with pads. Then, the NP-completeness of the problem is obvious.

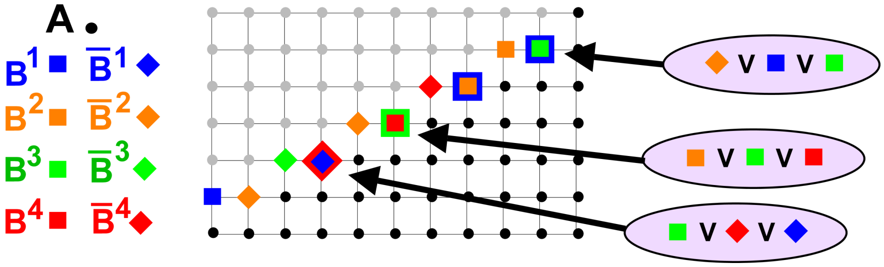

Fig. 4. Reduction of 3-SAT instance to $2\mathtt{CA}_{\mathcal{HV}}(A,(B^i)_{1\leq i\leq n})$. We encode the 3-SAT instance $(\overline{\mathfrak{b}}_1\vee\overline{\mathfrak{b}}_2)\wedge(\mathfrak{b}_3\vee\overline{\mathfrak{b}}_1\vee\overline{\mathfrak{b}}_4)\wedge(e\mathfrak{b}_2\vee\mathfrak{b}_3\vee\mathfrak{b}_4)\wedge(\mathfrak{b}_4\vee\mathfrak{b}_1\vee\mathfrak{b}_2)\wedge(\overline{\mathfrak{b}}_2\vee\mathfrak{b}_1\vee\mathfrak{b}_3)$. The clauses can be rewritten as $\mathfrak{b}_1 \implies \overline{\mathfrak{b}}_2$, $\overline{\mathfrak{b}}_3 \implies \overline{\mathfrak{b}}_1\vee\overline{\mathfrak{b}}_4$, $\overline{\mathfrak{b}}_2 \implies \mathfrak{b}_3\vee\mathfrak{b}_4$, $\overline{\mathfrak{b}}_4 \implies \mathfrak{b}_1\vee\mathfrak{b}_2$, $\mathfrak{b}_2 \implies \mathfrak{b}_1\vee\mathfrak{b}_3$ before being encoded on each row.

5.2 Cases 1 and 4 - $\mathcal{HV}$-convex aggregation

We provide now a sketch of reduction of any 3-SAT instance with n Boolean variables and c clauses to an equivalent instance $1\mathtt{CA}_{\mathcal{HV}}(A,\,(B^i)_{1\leq i\leq n})$ of polynomial size. The strategy is illustrated in Fig. 5. Two rectangular regions are used to encode a relation of negation between B^i and B^{n+i}: the pad B^i is joined with A $(i\in I)$ if and only if the pad B^{i+n} is not joined with A $(i+n\notin I)$. Therefore the pad B^i represents the literal $\mathfrak{b}_i$ while B_{n+i} represents its negation $\overline{\mathfrak{b}}_i$. Then each clause is encoded on an upper row. The size of the geometrical structure is $(3+3n+c)\times(2n+c)$. The equivalence between the instance of 3-SAT and the corresponding instance of $1\mathtt{CA}_{\mathcal{HV}}(A,(B^i)_{1\leq i\leq n})$ provides the wanted result.

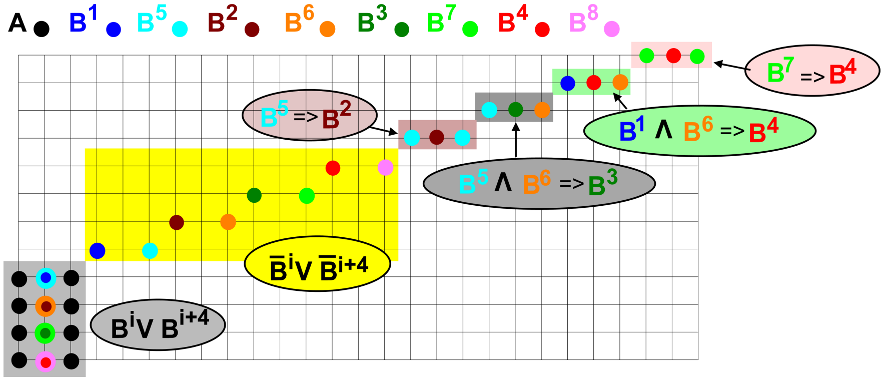
A
B^1
B^5
B^2
B^6
B^3
B^7
B^4
B^8
$B^7 \Rightarrow B^4$
$B^5 \Rightarrow B^2$
$B^1 \wedge B^6 \Rightarrow B^4$
$B^5 \wedge B^6 \Rightarrow B^3$
$\overline{B}^i \vee \overline{B}^{i+4}$
$B^i \vee B^{i+4}$

Fig. 5. Reduction of 3-SAT instance to $1\mathtt{CA}_{\mathcal{HV}}(A,(B^i)_{1\leq i\leq n})$. We encode the 3-SAT instance $(\mathfrak{b}_1 \vee \mathfrak{b}_2)\wedge(\mathfrak{b}_1 \vee \mathfrak{b}_2 \vee \mathfrak{b}_3)\wedge(\overline{\mathfrak{b}}_1 \vee \mathfrak{b}_2 \vee \mathfrak{b}_4)\wedge(\overline{\mathfrak{b}}_3 \vee \mathfrak{b}_4)$. The clauses can be rewritten as $\overline{\mathfrak{b}}_1 \implies \mathfrak{b}_2$, $\overline{\mathfrak{b}}_1 \wedge \overline{\mathfrak{b}}_2 \implies \mathfrak{b}_3$, $\mathfrak{b}_1 \wedge \overline{\mathfrak{b}}_2 \implies \mathfrak{b}_4$ and $\overline{\mathfrak{b}}_3 \implies \mathfrak{b}_4$. The $\mathcal{HV}$-convexity in the grey and yellow zones guarantee that we have to aggregate either B^i or B^{i+n} but not both. Then, as the pad B^i encodes the literal $\mathfrak{b}_i$, the pad B^{i+n} encodes its negation $\overline{\mathfrak{b}}_i$. In the upper part, each clause is easily encoded by a row.

The NP-completeness of the case 4 is obtained with the same strategy and a more simple structure. The pad B^i represents the literal $\mathfrak{b}_i$ while its conjugate $\overline{B}^i$ represents directly its negation $\overline{\mathfrak{b}_i}$. It allows us to encode any 3-SAT instance with a row for each clause (as in the upper part of Fig. 5). It makes the NP-completeness of $2\mathtt{CA}_{\mathcal{HV}}(A,(B^i)_{1\leq i\leq n},(\overline{B}^i)_{1\leq i\leq n})$ with disjoint pads straightforward.

5.3 Cases 2 and 5 - $\mathcal{C}$-convex aggregation.

We build a new geometrical structure with two goals: encode a negation relation between the pads B^i and B^{i+n}, and encode the c clauses of the 3-SAT instance. The structure has a support made by a polygon of consecutive edges of coordinates $(4k,4)$ with k going from 0 to $2n+c$ as drawn in Fig. 6. On each border, the polygon has 3 integer points which can be added or removed without any interference with the other points of the same kind. Each triplet of points is called a *nest*. The strategy of the algorithm is to use each nest to encode either the negation between the literals, or the clauses. We need 2 nests for each Boolean variable to encode the negation relation between the pads B^i and B^{i+n} and 1 nest per 3-clause.

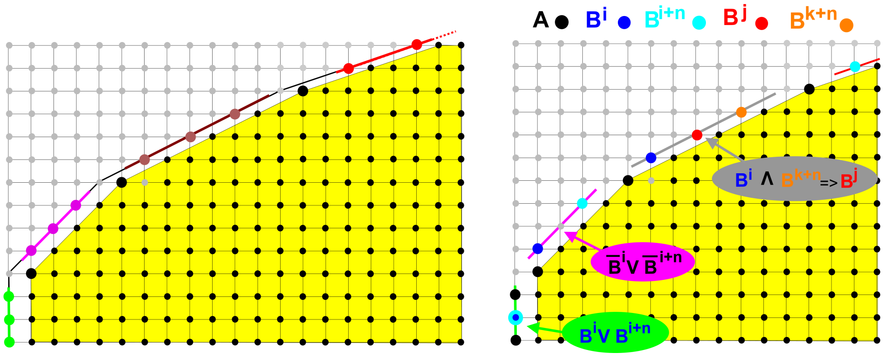
A ●
B^i ●
B^{i+n} ●
B^j ●
B^{k+n} ●
$B^i \wedge B^{k+n} \Rightarrow B^j$
$\overline{B}^i \vee \overline{B}^{i+n}$
$B^i \vee B^{i+n}$

Fig. 6. Reduction of 3-SAT instance to $2\mathtt{CA}_{\mathcal{C}}(A,(B^i)_{1\leq i\leq n})$. On the left, we provide the construction of the nests i.e the colored triplets of points on the left of the polygon of edges of coordinates $(4k,4)$. On the right, the two lower nests are used to encode the negation relation between the pads B^i and B^{i+n} so that B^{k+n} represents the literal $\overline{\mathfrak{b}}_i$. The third nest is used to encode $\mathfrak{b}_i \wedge \overline{\mathfrak{b}}_k \implies \mathfrak{b}_j$, namely $\overline{\mathfrak{b}}_i \vee \mathfrak{b}_k \vee \mathfrak{b}_j$.

This construction allows to prove that $1\mathtt{CA}_{\mathcal{C}}(A,(B^i)_{1\leq i\leq n})$ is NP-complete. It proves of course that $2\mathtt{CA}_{\mathcal{C}}(A,(B^i)_{1\leq i\leq n},(\overline{B}^i)_{1\leq i\leq n})$ is NP-complete since it is true with empty pads $\overline{B}^i$. For proving the case 5, the NP-completeness of $2\mathtt{CA}_{\mathcal{C}}(A,(B^i)_{1\leq i\leq n},(\overline{B}^i)_{1\leq i\leq n})$ with disjoint pads, we have to notice that the intersections between the pads B^i and B^{i+n} used for encoding the negation of $\mathfrak{b}_i$ are now useless. The pads $\overline{B}^i$ encode directly the negative literals $\overline{\mathfrak{b}}_i$. Then, only one nest per clause is sufficient to encode geometrically the 3-SAT instance (Fig. 7). As the construction has a polynomial size and can be done in polynomial time, it proves the NP-completeness of $2\mathtt{CA}_{\mathcal{C}}(A,(B^i)_{1\leq i\leq n},(\overline{B}^i)_{1\leq i\leq n})$ with disjoint pads. We can at last notice that the support A is 8-connected and that there is no difficulty to have pads included in the $\mathcal{HV}$-grid of A. It follows that $2\mathtt{CA}_{\mathcal{C}}(A,(B^i)_{1\leq i\leq n},(\overline{B}^i)_{1\leq i\leq n})$ remains NP-complete in this subcase.

6 Perspectives

Binary convex aggregation is the bottleneck of several problems of Discrete Tomography. The problem of binary convex aggregation $2\mathtt{CA}_{\mathcal{HV}}(A,(B^i)_{1\leq i\leq n},(\overline{B}^i)_{1\leq i\leq n})$ with an 8-connected support and disjoint pads is the last step of a possible strategy for reconstructing $\mathcal{C}$-convex lattice sets from their horizontal and vertical X-rays [7,3]. Unless $P=NP$, its NP-completeness shown in previous section lets no hope to tackle this last step with a general polynomial time algorithm. It shows that structural properties of the pads are necessary in this framework to achieve this strategy.

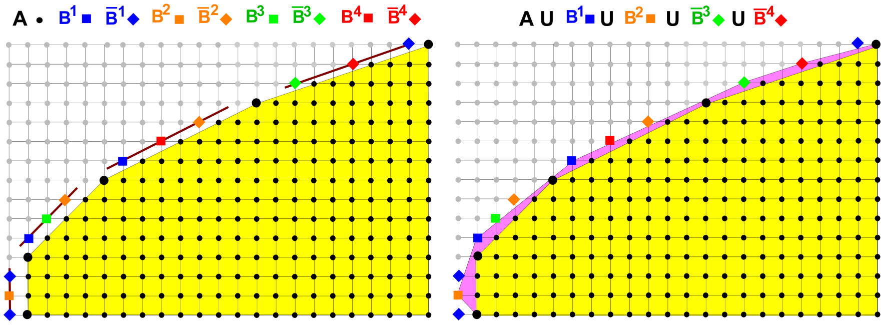

Fig. 7. Reduction of 3**-SAT instance to** $2\mathtt{CA}_{\mathcal{C}}(A, (B^i)_{1\leq i\leq n}, (\overline{B}^i)_{1\leq i\leq n})$ **with disjoint pads**. We encode the 3-SAT instance $(\mathfrak{b}_1 \vee \mathfrak{b}_2) \wedge (\overline{\mathfrak{b}}_1 \vee \mathfrak{b}_2 \vee \mathfrak{b}_3) \wedge (\overline{\mathfrak{b}}_1 \vee \mathfrak{b}_2 \vee \mathfrak{b}_4) \wedge (\mathfrak{b}_3 \vee \mathfrak{b}_1 \vee \overline{\mathfrak{b}}_4)$. By rewriting the clauses $\overline{\mathfrak{b}}_1 \implies \mathfrak{b}_2$, $\mathfrak{b}_1 \wedge \overline{\mathfrak{b}}_2 \implies \mathfrak{b}_3$, $\mathfrak{b}_1 \wedge \overline{\mathfrak{b}}_2 \implies \mathfrak{b}_4$ and $\overline{\mathfrak{b}}_3 \wedge \overline{\mathfrak{b}}_1 \implies \overline{\mathfrak{b}}_4$, each one is encoded in a nest. A solution of the convex aggregation problem $(A \cup B^1 \cup B^2 \cup \overline{B}^3 \cup \overline{B}^4)$ provides a solution of the 3-SAT instance ($\mathfrak{b}_1 = 1$, $\mathfrak{b}_2 = 1$, $\mathfrak{b}_3 = 0$ and $\mathfrak{b}_4 = 0$).

Acknowledgement This work has been sponsored by the French government research program "Investissements d'Avenir" through the IDEX-ISITE initiative 16-IDEX-0001 (CAP 20-25).

References

1. Elena Barcucci, Alberto Del Lungo, Maurice Nivat, and Renzo Pinzani. Reconstructing convex polyominoes from horizontal and vertical projections. Theoretical Computer Science, 155(2):321–347, 1996.
2. Sara Brunetti and Alain Daurat. Reconstruction of convex lattice sets from tomographic projections in quartic time. Theoretical Computer Science, 406(1):55 – 62, 2008. Discrete Tomography and Digital Geometry: In memory of Attila Kuba.
3. Paolo Dulio, Andrea Frosini, Simone Rinaldi, Lama Tarsissi, and Laurent Vuillon. First steps in the algorithmic reconstruction of digital convex sets. In Combinatorics on Words - 11th International Conference, WORDS 2017, Montréal, QC, Canada, September 11-15, 2017, Proceedings, pages 164–176, 2017.
4. Yan Gerard. Regular switching components. Theoretical Computer Science, Special Issue in the memory of Maurice Nivat(To appear), 2019.
5. Bengt Aspvall, Michael F. Plass, and Robert Endre Tarjan. A linear-time algorithm for testing the truth of certain quantified boolean formulas. Information Processing Letters, 8(3):121 – 123, 1979.
6. William F. Dowling and Jean H. Gallier. Linear-time algorithms for testing the satisfiability of propositional horn formulae. The Journal of Logic Programming, 1(3):267 – 284, 1984.
7. Yan Gerard. Polynomial Time Reconstruction of Regular Convex Lattice Sets from their Horizontal and Vertical X-Rays. preprint hal-01854636, August 2018.
8. C. Carathéodory. Über den variabilitätsbereich der koeffizienten von potenzreihen, die gegebene werte nicht annehmen. Mathematische Annalen, 64(1):95–115, 1907.