Theory of Computing Systems manuscript No.
(will be inserted by the editor)

The Complexity of Grid Coloring

Daniel Apon · **William Gasarch** · **Kevin Lawler**

Received: date / Accepted: date

Abstract A c-coloring of the grid $G_{N,M} = [N] \times [M]$ is a mapping of $G_{N,M}$ into $[c]$ such that no four corners forming a rectangle have the same color. In 2009 a challenge was proposed to find a 4-coloring of $G_{17,17}$. Though a coloring was produced, finding it proved to be difficult. This raises the question of whether there is some complexity lower bound. Consider the following problem: given a partial c-coloring of the $G_{N,M}$ grid, can it be extended to a full c-coloring? We show that this problem is NP-complete. We also give a Fixed Parameter Tractable algorithm for this problem with parameter c.

Keywords Coloring · Complexity · Grid

Mathematics Subject Classification (2020) 11Y16 · 05D10

1 Introduction

Definition 1

1. If $x \in \mathbb{N}$ then $[x]$ denotes the set $\{1, \ldots, x\}$. $G_{N,M}$ is the set $[N] \times [M]$.

Definition 2 Let $n, m, c \geq 1$.

1. A *rectangle of* $G_{N,M}$ is a subset of the form

$$\{(a,b), (a+d_1,b), (a+d_1,b+d_2), (a,b+d_2)\} \subseteq G_{N,M}$$

for some $a, b, d_1, d_2 \in \mathbb{N}$ with $d_1, d_2 \geq 1$ Note that we are only looking at the four corners of the rectangle—nothing else.

Daniel Apon
University of Maryland at College Park, MD, 20742 E-mail: dapon.crypto@gmail.com

William Gasarch
University of Maryland at College Park, MD, 20742 E-mail: gasarch@umd.edu

Kevin Lawler
Permanent, Berkeley CA, 94710 E-mail: kevin@permenentco.com

2. Let $\chi : G_{N,M} \to [c]$. A *monochromatic rectangle* is a rectangle where all 4 elements of it are colored the same.
3. Let $\chi : G_{N,M} \to [c]$. If there are no rectangles with all four corners the same color then we call χ a *c-coloring*. If the c is understood we may just say *a coloring*. We sometimes use the term *proper c-coloring* rather than *c-coloring* to stress the fact that it has no monochromatic rectangles.
4. A grid $G_{N,M}$ is *c-colorable* if there is a c-coloring of it.

Fenner et al. [2] explored the following problem:

Which grids are c-colorable for a given fixed c?

2 History and Our Results

We state some of the results of Fenner et al. [2].

1. For all $c \geq 2$, $G_{c+1,c\binom{c+1}{2}+1}$ is not c-colorable.
2. For all c there exists a finite number of grids, denoted OBS_c, such that $G_{N,M}$ is c-colorable iff it doesn't contain any element of OBS_c. OBS stands for *the obstruction set*.
3. $\mathrm{OBS}_2 = \{G_{3,7}, G_{5,5}, G_{7,3}\}$. This was obtained without the aid of a computer program.
4. $\mathrm{OBS}_3 = \{G_{19,4}, G_{16,5}, G_{13,7}, G_{11,10}, G_{10,11}, G_{7,13}, G_{5,16}, G_{4,19}\}$. A computer aided search was used to find a 3-coloring of $G_{10,10}$.
5. $$\mathrm{OBS}_4 = \{G_{41,5}, G_{31,6}, G_{29,7}, G_{25,9}, G_{23,10}, G_{22,11}, G_{21,13}, G_{19,17}\} \bigcup$$ $$\{G_{17,19}, G_{13,21}, G_{11,22}, G_{10,23}, G_{9,25}, G_{7,29}, G_{6,31}, G_{5,41}\}$$
   The authors were stuck for a long time trying to find 4-colorings of $G_{17,17}$, $G_{17,18}$, $G_{18,18}$, $G_{12,21}$, and $G_{10,22}$ (we omit the symmetric cases which follow automatically, i.e., if there is a 4-coloring of $G_{22,10}$ then there is one for and $G_{10,22}$). They believed these were all 4-colorable. William Gasarch put a bounty of $17^2 = 289$ dollars for a 4-coloring of $G_{17,17}$ and posted this challenge to ComplexityBlog [3]. Bernd Steinbach and Christian Posthoff found 4-colorings of $G_{17,17}$, $G_{18,18}$, and $G_{12,21}$ and received the reward. Brad Larsen found a 4-coloring of $G_{22,10}$. Brad Larsen posted the 4-coloring saying he used a SAT-solver but he did not elaborate. Steinbach and Posthoff published their results and their methods. In brief, they used a very deep analysis that allowed for a strong reduction of the problem, and then used the Universal SAT-Solver clasp. See their articles [9–14] and a book edited by Steinbach [8] that has several chapters explaining how they found a 4-coloring of $G_{12,21}$ in detail. These results completed the search for OBS_4.
6. Finding OBS_5 seems to be beyond current technology.

The difficulty of 4-coloring $G_{17,17}$ and pinning down OBS_5 raise the following question: is the problem of grid coloring hard? In Section 3 we define the *Grid Coloring Extension Problem*. In Section 4 we show this problem is NP-complete. Does this

R								
R								
B		R						
R				B				
R					B			
R								
R	R	R	R	R	R	R	R	R

Fig. 1 Example of a partial coloring of $G_{9,7}$

really indicate that 4-coloring is hard? In Section 5 we discuss the issue. In Section 6 we show that the grid coloring extension problem is Fixed Parameter Tractable with parameter c. Does this really give a way to find extensions quickly? In Section 6.2 we discuss this issue. In Section 7 we present open problems.

3 Definition of the Grid Coloring Extension Problem

Definition 3 Let $N, M, c \in \mathbb{N}$.

1. Given a grid $G_{N,M}$, a *cell* is an element $(i, j) \in G_{N,M}$.
2. A *partial c-coloring* χ *of* $G_{N,M}$ is a mapping of a subset of $G_{N,M}$ to $[c]$ with no monochromatic rectangle on the points where it is defined. See Figure 1 for an example.
3. If χ is a partial c-coloring of $G_{N,M}$ then χ' is *an extension of* χ if χ' is a partial c-coloring of $G_{N,M}$ which
   (a) is defined on every cell that χ is defined,
   (b) agrees with χ on those cells,
   (c) may be defined on more cells, and
   (d) has no monochromatic rectangles.
4. A *total mapping* χ *of* $G_{N,M}$ *to* $[c]$ is a mapping of $G_{N,M}$ to $[c]$. This would normally just be called a mapping, but we use the term total to distinguish it from a partial mapping.

Definition 4 GCE is the following problem:

- *Input* $N, M, c \geq 1$ and χ a partial c-coloring of $G_{N,M}$. The numbers N, M, c are in unary.
- *Output* YES if there is an extension of χ to a total c-coloring of $G_{N,M}$, NO otherwise.

GCE stands for *Grid Coloring Extension*.

We show that GCE is NP-complete. This result may explain why the original 17×17 challenge was so difficult. Then again—it may not. We discuss this further in Section 5.

4 GCE is NP-complete

Before showing that GCE is NP-complete we briefly discuss its status within NP. We first state and prove an easy upper bound.

Theorem 1 GCE $\in$ NTIME($O(N^2M^2)$) *with certificate of size* $O(NM\log c)$.

Proof Here is a nondeterministic algorithm

1. *Input* (N,M,c,χ).
2. Guess an extension χ' of the c-coloring χ to a total mapping of $G_{N,M}$ to $[c]$. Note that χ', the certificate, is of size $O(NM\log c)$.
3. For all
$$\{(a,b),(a+d_1,b),(a+d_1,b+d_2),(a,b+d_2)\} \subseteq G_{N,M}$$
do the following
   (a) Check if
$$\chi'(a,b) = \chi'(a+d_1,b) = \chi'(a+d_1,b+d_2) = \chi'(a,b+d_2).$$
   (b) If yes then this branch stops and *outputs* NO.
   (c) If no then (a) if this is the last rectangle to check then stop and *output* YES, (b) if not then proceed to the next rectangle.

Each execution of the loop body takes $O(1)$ time. To get a time bound we need an upper bound on how often the loop is executed. This is upper bounded by the number of rectangles.

The number of ways to pick a is N. The number of ways to pick b is M. The number of ways to pick d_1 is $\leq N$. The number of ways to pick d_2 is $\leq M$. Hence the number of rectangles is $O(N^2M^2)$. Hence the runtime of any one branch is $O(N^2M^2)$. Hence the algorithm is in NTIME($O(N^2M^2)$).

□

We obtain a better bound. Kreveld and De Berg [6] proved, in our notation, the following lemma.

Lemma 1 *There is an algorithm that will, given a set of cells* $P \subseteq G_{N,M}$*, determine if* P *contains a rectangle, in time* $O((NM)^{3/2})$.

From this result we show the following:

Theorem 2 GCE $\in$ NTIME($O(c(MN)^{3/2})$).

Proof Given (N,M,c,χ) the witness is a proposed extension χ' of χ to a c-coloring of $G_{N,M}$. The following algorithm tries to verifies that χ' is a coloring.

1. *Input* (N,M,c,χ,χ'). We assume the colors are $\{1,\ldots,c\}$.
2. Verify that χ' is an extension of χ. This takes $O(NM)$ steps.
3. For all $1 \leq i \leq c$
   (a) Let P be the set of cells colored i. It takes $O(MN)$ time to identify P.

(b) Use the algorithm from Lemma 1 to determine if P contains a rectangle. This takes time $O((NM)^{3/2})$. If the algorithm says P contains a rectangle then *output* NO and stop. Otherwise proceed to the next i.

4. (If the algorithm got here then, for all $1 \le i \le c$, there is no i-colored rectangle.) *Output* YES and stop.

The time spent in the For-Loop dominates everything else. That time is clearly $O(c(NM)^{3/2})$.

□

We make one observation about GCE and SAT before our proof. It is an easy exercise to express the question $(N,M,c,\chi) \in$ GCE as a SAT formula. (This was the starting point for the work of Steinback and Posthoff with χ being the empty function.) This shows that GCE reduces to SAT but not that SAT reduces to GCE. Hence this reduction does not help us obtain a lower bound on the complexity of GCE.

We now show GCE is NP-complete.

Theorem 3 GCE *is* NP-*complete.*

Proof By Theorem 2, GCE $\in$ NP.

We give a reduction of 3SAT to GCE. The input will be a 3CNF formula

$$\phi(x_1,\dots,x_n) = C_1 \wedge \cdots \wedge C_m$$

with n free variables and m clauses. The output will be (N,M,c,χ) where

- $N,M,c \in \mathbb{N}$,
- χ is a partial c-coloring of $G_{N,M}$, and
- $\phi \in$ 3SAT iff $(N,M,c,\chi) \in$ GCE.

We can assume that ϕ never has a clause that contains either (1) the same literal twice, or (2) a variable and its negation. Condition (1) will be needed in Part III of the construction. Condition (2) will be needed in the proof of Claim 4.

The reduction we show you *does not quite work!*; however, it has most of the ideas needed. There is a problem with it that will be revealed when we try to prove Claim 4. During that proof we will see what goes wrong and modify the construction so that Claim 4 is true.

Visualize the full grid as a core subgrid with additional entries to the left and below. These additional entries are there to enforce that some colors in the core grid occur only once.

$(2,4)$							
$(2,4)$							
$(2,4)$							
T		$(2,4)$					
$(2,4)$							
$(2,4)$							
$(2,4)$							
$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$

Fig. 2 Cell $(2,4)$ is colored $(2,4)$. No other cell can be colored $(2,4)$ in a proper coloring.

Conventions

1. Throughout this proof *extension* means *an extension that uses the colors T,F on some of the uncolored cells and does not have a monochromatic rectangle*. It may or may not extend to the entire grid.
2. In our figures we will have literals labeling some of the rows and clauses labeling some of the columns. These are not part of the construction. The literals and clauses are visual aids. We may refer to *row* x_7 or *column* C_3.
3. In our figures we will have double lines to separate things. These lines are not part of the construction. These are visual aids.
4. The colors will be T, F, and some of the $(i,j) \in G_{N,M}$. Many of the cells that are in the core grid will be colored (i,j) where that is their position in the core grid. In the figures we will denote the color by D for distinct. Part I of the construction will make sure that no other cell in the core grid can have that color.

The reduction is in four parts. We will mainly construct a core grid which will be $2n+m$ by $2n+2m+1$ (when we later modify the construction the core grid will be bigger, though still linear in n,m).

In all figures the left bottom cell of the core grid is indexed $(1,1)$.

Part I: Forcing a color to appear only once in the core grid.

For (i,j) in the core grid we will often set $\chi(i,j)$ to (i,j) and then never reuse (i,j) in the core grid. By doing this, we make having a monochromatic rectangle rare and have control over when that happens.

We show how to color the cells that are not in the core grid to achieve this. Part I will be the final step in the reduction since we need to know the size of the grid before we can apply it; however, we show Part I first.

Say we want the cell $(2,4)$ in the core grid to be colored $(2,4)$ and we do not want this color appearing anywhere else in the core grid. We can do the following: add a column of $(2,4)$'s to the left end (with one exception) and a row of $(2,4)$'s at the bottom. See Figure 2.

It is easy to see that in any extension of he coloring of the grid in Figure 2 the only cells that can have the color $(2,4)$ are those shown to already have that color. It is also easy to see that the color T we have will not help to create any monochromatic rectangles since there are no other T's in its column. The T we are using *is* the same

$(5,3)$	$(2,4)$								
$(5,3)$	$(2,4)$								
$(5,3)$	$(2,4)$								
$(5,3)$	T		$(2,4)$						
T	$(2,4)$					$(5,3)$			
$(5,3)$	$(2,4)$								
$(5,3)$	$(2,4)$								
$(5,3)$	$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$	$(2,4)$
$(5,3)$	$(5,3)$	$(5,3)$	$(5,3)$	$(5,3)$	$(5,3)$	$(5,3)$	$(5,3)$	$(5,3)$	$(5,3)$

Fig. 3 $(2,4)$ and $(5,3)$ within a sub-grid

T that will later mean *true*. We could have used F. We do not want to use new colors since we would have no control over where else they could be used.

What if some other cell needs to have a unique color? Let's say we also want to color cell $(5,3)$ in the core grid with $(5,3)$ and do not want to color anything else in the core grid $(5,3)$. Then we use the grid in Figure 3.

It is easy to see that in any extension of the coloring of the grid in Figure 3 the only cells that can have the color $(2,4)$ or $(5,3)$ are those shown to already have those colors.

For the rest of the construction we will only show the core grid. If we denote a color as D (short for *Distinct*) in the cell (i,j) then this means that

1. cell (i,j) is color (i,j), and
2. we have used the above gadget to make sure that (i,j) does not occur as a color in any other cell of the core grid.

Note that when we have D in the $(2,4)$ cell and in the $(5,3)$ cell, they denote different colors.

Part II: Forcing $(x,\bar{x})$ to be colored (T,F) or (F,T).

The first column of the core grid will have $2n$ blanks and then m D's. We will use the m D's later. Figure 4 illustrates what we do in the $n=4$ case.

We will arrange things so that the color of the blanks in Figure 4 will all be either T or F. We refer to the color of the cell next to x_i as *the color of* x_i. Same for $\bar{x}_i$.

It is easy to see that in any extension of the coloring of Figure 4:

- If x_i is colored T then $\bar{x}_i$ is colored F.
- If x_i is colored F then $\bar{x}_i$ is colored T.

We leave it to the reader to generalize Figure 4 to n variables.

We will call the left most column, which is mostly blank, *the literal column*.

This part is what will need to be adjusted. It will turn out that we need several copies of each literal. During the proof of Claim 4 we will see why this is true and how to achieve it.

Part III: Forcing the coloring to satisfy a single clause

	D	D	D	D	D	D	D	D	D
	D	D	D	D	D	D	D	D	D
	D	D	D	D	D	D	D	D	D
	D	D	D	D	D	D	D	D	D
$\bar{x}_4$		D	D	D	D	D	D	T	F
x_4		D	D	D	D	D	D	T	F
$\bar{x}_3$		D	D	D	D	T	F	D	D
x_3		D	D	D	D	T	F	D	D
$\bar{x}_2$		D	D	T	F	D	D	D	D
x_2		D	D	T	F	D	D	D	D
$\bar{x}_1$		T	F	D	D	D	D	D	D
x_1		T	F	D	D	D	D	D	D

Fig. 4 Literal Gadget with four variables

										C_1	C_1	C_2	C_2	C_3	C_3	C_4	C_4
	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	T	T
	D	D	D	D	D	D	D	D	D	D	D	D	D	T	T	D	D
	D	D	D	D	D	D	D	D	D	D	D	T	T	D	D	D	D
	D	D	D	D	D	D	D	D	D	T	T	D	D	D	D	D	D
$\bar{x}_4$		D	D	D	D	D	D	T	F	X	X	X	X	X	X	X	X
x_4		D	D	D	D	D	D	T	F	X	X	X	X	X	X	X	X
$\bar{x}_3$		D	D	D	D	T	F	D	D	X	X	X	X	X	X	X	X
x_3		D	D	D	D	T	F	D	D	X	X	X	X	X	X	X	X
$\bar{x}_2$		D	D	T	F	D	D	D	D	X	X	X	X	X	X	X	X
x_2		D	D	T	F	D	D	D	D	X	X	X	X	X	X	X	X
$\bar{x}_1$		T	F	D	D	D	D	D	D	X	X	X	X	X	X	X	X
x_1		T	F	D	D	D	D	D	D	X	X	X	X	X	X	X	X

Fig. 5 Clause setup

For each clause $C = L_1 \vee L_2 \vee L_3$ we will use two columns. These columns will be called *clause columns*.

Before saying what we put into the columns, Figure 5 is the initial setup in the case of $n = 4$ and $m = 4$. We leave it to the reader to generalize to n, m. The X's in Figure 5 will be replaced by T's, F's, or blanks in the next step.

Let $C = L_1 \vee L_2 \vee L_3$. Figure 6 illustrates how we color, or leave blank, the cells in the C-column.

Note that since we never have the same literal appearing twice in a clause, the construction of the Clause Gadget can be carried out.

We redraw Figure 6 as Figure 7 for ease of use. We refer to the partial coloring in Figure 7 as χ.

		$\cdots$	C	C	$\cdots$
	D	$\cdots$	T	T	$\cdots$
	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\cdots$
L_3		$\cdots$	D	F	$\cdots$
	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\cdots$
L_2		$\cdots$			$\cdots$
	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\cdots$
L_1		$\cdots$	F	D	$\cdots$
	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\cdots$

Fig. 6 The clause gadget

		C	C
	D	T	T
L_3		D	F
L_2			
L_1		F	D

Fig. 7 The clause gadget—easier to work with

		C	C
	D	T	T
L_3	F	D	F
L_2	F		
L_1	F	F	D

Fig. 8 L_1, L_2, L_3 all set to F

Claim 1: Let χ denote the partial coloring shown in Figure 7. If χ' is an extension of χ then χ' cannot have the L_1, L_2, L_3 cells all colored F.

Proof of Claim 1:

Assume, by way of contradiction, that L_1, L_2, L_3 are all colored F. Then we have the partial coloring in Figure 8

The reader can verify that if the two blank cells of Figure 8 are colored TT, TF, FT, or FF, there will be a monochromatic rectangle.

End of Proof of Claim 1

Claim 2: Let χ' be an extension of the coloring in Figure 7 that colors L_1, L_2, L_3 but not the other two blank cells. Assume that χ' colors L_1, L_2, L_3 anything except F, F, F. Then χ' can be extended to color the two blank cells.

Proof of Claim 2

There are seven cases based on (L_1, L_2, L_3) being labeled FFT, FTF, FTT, TFF, TFT, TTF, TTT. For each one we give a coloring of the remaining two blank cells so that no monochromatic rectangle is formed.

Case 1

		C	C
	D	T	T
L_3	F	D	F
L_2	F	F	T
L_1	T	F	D

Case 2

		C	C
	D	T	T
L_3	F	D	F
L_2	T	T	F
L_1	F	F	D

Case 3

		C	C
	D	T	T
L_3	F	D	F
L_2	T	T	F
L_1	T	F	D

Case 4

		C	C
	D	T	T
L_3	T	D	F
L_2	F	T	F
L_1	F	F	D

Case 5

		C	C
	D	T	T
L_3	T	D	F
L_2	F	T	F
L_1	T	F	D

Case 6

		C	C
	D	T	T
L_3	T	D	F
L_2	T	T	F
L_1	F	F	D

Case 7

		C	C
	D	T	T
L_3	T	D	F
L_2	T	T	F
L_1	T	F	D

End of Proof of Claim 2

Part IV: Putting it all together

Recall that $\phi(x_1,\ldots,x_n) = C_1 \wedge \cdots \wedge C_m$ is a 3CNF formula. We first define the core grid and later define the entire grid and N,M,c. The core grid will have $2n+m$ rows and $2m+2n+1$ columns (when we later modify the construction the core grid will be bigger though still linear in n,m) The $2n$ left-most columns are partially colored, and labeled with literals, as described in Part II. The m top-most rows are colored, and labeled with clauses, as described in Part III. The rest of the core grid is colored as described in Part III.

The core grid is now complete. For every (i,j) that is colored (i,j), we perform the method in Part I to make sure that (i,j) is the only cell with color (i,j). Let the number of such (i,j) be E. The number of colors c is $E+2$. This will force everything else to be colored T or F. Note that $E = \Theta(NM)$.

In Figure 9 we present the core grid for the instance of GCE obtained if the original formula is

$$(x_1 \vee x_2 \vee \bar{x}_3) \wedge (x_1 \vee x_2 \vee x_4) \vee (\bar{x}_2 \vee x_3 \vee x_4).$$

Claim 3: Let $\phi(x_1,\ldots,x_n)$ be a 3CNF formula. Let (N,M,c,χ) be the result of the reduction described above. If $(N,M,c,\chi) \in$ GCE then $\phi \in$ 3SAT.

Proof of Claim 3

Assume that $(N,M,c,\chi) \in$ GCE. According to the construction in Part II the first column gives a valid truth assignment for $x_1,\ldots,x_n$ (and hence also for $\bar{x}_1,\ldots,\bar{x}_n$). By Claim 1, for every clause $C = L_1 \vee L_2 \vee L_3$ this truth assignment cannot assign L_1, L_2 and L_3 all to F. Hence this is a satisfying assignment, so $\phi \in$ 3SAT.

End of Proof of Claim 3

We will now try to show that if $\phi \in$ 3SAT then $(N,M,c,\chi) \in$ GCE. **We will fail!** This will motivate us to modify our construction.

Claim 4 (which is false): Let $\phi(x_1,\ldots,x_n)$ be a 3CNF formula. Let (N,M,c,χ) be the result of the reduction described above. If $\phi \in$ 3SAT then $(N,M,c,\chi) \in$ GCE.

Proof of Claim 4 (which will fail)

										C_1	C_1	C_2	C_2	C_3	C_3
	D	D	D	D	D	D	D	D	D	D	D	D	D	T	T
	D	D	D	D	D	D	D	D	D	D	D	T	T	D	D
	D	D	D	D	D	D	D	D	D	T	T	D	D	D	D
$\overline{x}_4$		D	D	D	D	D	D	T	F	D	D	D	D	D	D
x_4		D	D	D	D	D	D	T	F	D	D	D	F	D	F
$\overline{x}_3$		D	D	D	D	T	F	D	D	D	F	D	D	D	D
x_3		D	D	D	D	T	F	D	D	D	D	D	D		
$\overline{x}_2$		D	D	T	F	D	D	D	D	D	D	D	D	F	D
x_2		D	D	T	F	D	D	D	D					D	D
$\overline{x}_1$		T	F	D	D	D	D	D	D	D	D	D	D	D	D
x_1		T	F	D	D	D	D	D	D	F	D	F	D	D	D

Fig. 9 Example with $(x_1 \vee x_2 \vee \overline{x}_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge (\overline{x}_2 \vee x_3 \vee x_4)$

	$\cdots$	D	D	$\cdots$	C	$\cdots$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	D	$\vdots$
$\overline{x}$	$\cdots$	T	F	$\cdots$	F	$\cdots$
x	$\cdots$	T	F	$\cdots$	F	$\cdots$

Fig. 10 Case 2 of Claim 4

Assume $\phi \in$ 3SAT. Let $(b_1, \ldots, b_n)$ be a satisfying truth assignment where, for $1 \le i \le n$, $b_i \in \{T, F\}$. We use this to obtain a coloring of $G_{N,M}$ that is an extension of χ.

Color the literal column in the obvious way: the entry labeled with literal L is labeled the truth assignment of L. We now show how we try to color the blank cells in the clause columns.

Let $C = L_1 \vee L_2 \vee L_3$ be a clause. The part of the grid associated to it is in Figure 6.

The literal column we have already colored. Since the assignment was satisfying, at least one of L_1, L_2, L_3 was set to T. We use Claim 2 to extend the coloring to the blank cells. This forms a grid coloring. We try to prove this coloring is proper.

Assume, by way of contradiction, that there is a monochromatic rectangle which we call R.

Case 1 There is a clause C such that R uses the two T's associated to C. The only way these T's can be involved in a monochromatic rectangle is if the two blank cells associated to C are colored T. By the 7 cases in Claim 2 this cannot occur.

Case 2 There is a variable x such that R uses the two T's or two F's associated to x. Figure 10 shows what this looks like (we only include the relevant parts). We assume x is the first variable in C (the other cases are either similar or cannot occur).

No clause-column has two T's in it, so R must be colored F. The only way there can be two F's in the literal-column is if they are associated to a literal and its negation, as in Figure 10. However, the only way that configuration can happen is if x

										C_1	C_1	C_2	C_2	C_3	C_3
	D	D	D	D	D	D	D	D	D	D	D	D	D	T	T
	D	D	D	D	D	D	D	D	D	D	D	T	T	D	D
	D	D	D	D	D	D	D	D	D	T	T	D	D	D	D
$\overline{x}_4$	***T***	D	D	D	D	D	D	T	F	D	D	D	D	D	D
x_4	***F***	D	D	D	D	D	D	T	F	D	D	D	F	D	F
$\overline{x}_3$	***F***	D	D	D	D	T	F	D	D	D	F	D	D	D	D
x_3	***T***	D	D	D	D	T	F	D	D	D	D	D	D	***T***	***F***
$\overline{x}_2$	***T***	D	D	T	F	D	D	D	D	D	D	D	D	F	D
x_2	***F***	D	D	T	F	D	D	D	D	***F****	***T***	***F****	***T***	D	D
$\overline{x}_1$	***F***	T	F	D	D	D	D	D	D	D	D	D	D	D	D
x_1	***T***	T	F	D	D	D	D	D	D	$F*$	D	$F*$	D	D	D

Fig. 11 Example with $(x_1 \vee x_2 \vee \overline{x}_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge (\overline{x}_2 \vee x_3 \vee x_4)$

and $\overline{x}$ are in the same clause. This cannot happen since ϕ has no clauses with both a variable and its negation in it.
Case 3 R uses the literal column and one of the clause columns. By Claim 2, R is not monochromatic.
Case 4 The only case left is if R uses two clause columns. *This can occur!* This is where the construction fails! We give an example. Recall that Figure 9 is the instance of GCE from the formula

$$(x_1 \vee x_2 \vee \overline{x}_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge (\overline{x}_2 \vee x_3 \vee x_4).$$

Lets say we take the satisfying truth assignment

$$x_1 = T, x_2 = F, x_3 = T, x_4 = F.$$

If we put these in the literal column and use the proof of Claim 2 to color the blank cells in the clause columns, the result is the coloring of the entire grid seen in Figure 11. The boldfaced colors are the ones caused by the truth assignment. The asterisks show a monochromatic rectangle. Hence the construction produces a non-proper coloring and is incorrect.
End of the Proof of Claim 4 (that failed)

The way to avoid Case 4 is if we had *several* copies of each literal so that if two clauses use the same literal, they will use different copies of it. How many? The number of copies of literal L has to be at least the number of clauses that L appears in. It will be convenient to have the number of copies of L and of $\overline{L}$ be the same. Hence if x appears in m_1 clauses, and $\overline{x}$ appears in m_2 clauses, then we'll add $\max\{m_1, m_2\}$ rows for each of these literals.

Rather than give the general construction, we do an example with the case that gave us trouble before:

$$(x_1 \vee x_2 \vee \overline{x}_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge (\overline{x}_2 \vee x_3 \vee x_4).$$

																		C_1	C_1	C_2	C_2	C_3	C_3
	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	T	T
	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	T	T	D	D
	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	T	T	D	D	D	D
$\overline{x}_4$		D	D	D	D	D	D	D	D	D	D	D	D	D	D	T	F	D	D	D	D	D	D
x_4		D	D	D	D	D	D	D	D	D	D	D	D	D	D	T	F	D	D	D	F	D	F
$\overline{x}_3$		D	D	D	D	D	D	D	D	D	D	D	D	T	F	D	D	D	F	D	F	D	D
x_3		D	D	D	D	D	D	D	D	D	D	D	D	T	F	D	D	D	D	D	D		
$\overline{x}_2$		D	D	D	D	D	D	D	D	D	D	T	F	D	D	D	D	D	D	D	D	D	D
x_2		D	D	D	D	D	D	D	D	T	F	T	F	D	D	D	D	D	D			D	D
$\overline{x}_2$		D	D	D	D	D	D	T	F	T	F	D	D	D	D	D	D	D	D	D	D	F	D
x_2		D	D	D	D	D	D	T	F	D	D	D	D	D	D	D	D			D	D	D	D
$\overline{x}_1$		D	D	D	D	T	F	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
x_1		D	D	T	F	T	F	D	D	D	D	D	D	D	D	D	D	D	D	F	D	D	D
$\overline{x}_1$		T	F	T	F	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
x_1		T	F	D	D	D	D	D	D	D	D	D	D	D	D	D	D	F	D	D	D	D	D

Fig. 12 Example with $(x_1 \vee x_2 \vee \overline{x}_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge (\overline{x}_2 \vee x_3 \vee x_4)$

in Figure 12. We prove that the literals x_1 and $\overline{x}_1$ behave as they should. From the example and the proof about $x_1, \overline{x}_1$, the reader will be able to work out the general construction.

Claim 5 Let χ' be a proper extension of the coloring in Figure 12 that colors all entries of the literal column. Then the following three statements are true.

1. The x_1 cells all have the same color: T or F.
2. The $\overline{x}_1$ cells all having the same color: T or F.
3. The color of x_1 and $\overline{x}_1$ are different.

Proof of Claim 5

We coordinate the grid by having the bottom left cell be $(1,1)$ (which is blank), the cell to the right has coordinate $(2,1)$ (which has T), and the cell above it is $(1,2)$ (which is blank).

Assume $\chi'(1,1) = T$ (the case where $\chi'(1,1) = F$ is similar).
Since $\chi'(1,1) = T$, $\chi'(2,1) = T$, and $\chi'(2,2) = T$, we have $\chi'(1,2) = F$.
Since $\chi'(1,2) = F$, $\chi'(5,2) = F$, and $\chi'(5,3) = F$, we have $\chi'(1,3) = T$.
Since $\chi'(1,3) = T$, $\chi'(6,3) = T$, and $\chi'(6,4) = T$, we have $\chi'(1,4) = F$.

End of Proof of Claim 5

From this the reader can work out the general construction.

□

Recap and the Actual Values of N, M, c

Our goal was to, given a 3CNF formula

$$\phi(x_1, \ldots, x_n) = C_1 \wedge \cdots \wedge C_m,$$

$\overline{x}_1$	D	D	D	D	D	D	T	F	T	F
x_1	D	D	D	D	D	D	T	F	D	D
$\overline{x}_1$	D	D	D	D	T	F	T	F	D	D
x_1	D	D	T	F	T	F	D	D	D	D
$\overline{x}_1$	T	F	T	F	D	D	D	D	D	D
x_1	T	F	D	D	D	D	D	D	D	D

Fig. 13 Three Occurrence of x_1

with n free variables and m clauses, output an instance of GCE such that

$$\phi \in \text{3SAT iff } (N, M, c, \chi) \in \text{GCE}.$$

We have described the partial coloring χ. For the sake of completeness we now specify N, M, c. We first find the dimensions of the core grid.

Definition 5 o_i is the maximum of the number of occurrence of x_i and $\overline{x}_i$.

Lemma 2 $\sum_{i=1}^{n} o_i \leq 3m$.

Proof Since o_i is the number of times one of $\{x_i, \overline{x}_i\}$ occurs the sum is bounded by the number of occurrence of variables. Since the formula is in 3CNF form, the number of occurrence of variables is $3m$.

□

We use Lemma 2 to get upper bounds on several quantities including N, M, c.

Number of rows in the core grid The literal column will have o_i rows labeled x_i and o_i rows labeled $\overline{x}_i$. Hence the literal column will have $\sum_{i=1}^{n} 2o_i$ blank cells. Every clause C induces a row. (The row has all D's except for two T's under the columns labeled C; however, we do no need that to count the number of rows.) Hence there are m additional rows. Therefore the number of rows in the core grid is

$$N' = m + \sum_{i=1}^{n} 2o_i = m + 2\sum_{i=1}^{n} o_i \leq m + 3m = 3m.$$

Number of columns in the core grid Each variable x_i induces a rectangle of height $2o_i$ and width $4o_i - 2$. See Figure 9 for an example with $o_1 = 1$, Figure 12 for an example with $o_1 = 2$, and Figure 13 for an example with $o_1 = 3$. Each clause adds 2 columns. Therefore the number of columns in the core grid is

$$M' = 2m + \sum_{i=1}^{n} (4o_i - 2) = 2m + 4\sum_{i=1}^{n} o_i \leq 2m + 4 \times 3m = 14m \leq 14m.$$

Note that N', M' are linear in n, m as promised earlier. However, the non-core part of the grid will add an $O(m^2)$ term to the size of N, M.

The Number of Blank Cells, T-Cells, F-Cells, and Colors

The first column has $\sum_{i=1}^{n} 2o_i$ blank cells. Each column labeled with a clause has 1 blank cell. Hence the number of blank cells is

$$B = m + \sum_{i=1}^{n} 2o_i = m + 2\sum_{i=1}^{n} o_i.$$

Each column that is not labeled with a clause has one T and one F. Each column labeled with a clause has one T and one F. Hence the number of cells labeled with a T or an F is

$$2M'$$

Every cell that is neither blank, T, or F has a distinct color. Hence the number of new colors that are not T or F is

$$E = N'M' - B - 2M' \leq N'M' \leq 42m^2.$$

and the total number of colors is

$$c = E + 2 \leq 42m^2 + 2.$$

The real values of N, M

We now deal with the non-core part of the grid. For every color that is not T or F we add one row and one column to the grid (see Part I of the construction). Hence

$$M = M' + E \leq 14m + 42m^2 = O(m^2).$$

$$N = N' + E \leq 3m + 42m^2 = O(m^2).$$

Note that M, N are polynomial in the length of ϕ.

5 What the NP-Completeness Result Does and Does Not Tell Us

The motivation for this paper was

Why was finding if $G_{17,17}$ is 4-colorable so hard?

Towards this goal we showed, in Theorem 3, that GCE is NP-complete. But does this really capture the problem we want to study? We give several reasons why not. These will point to further investigations.

1) The reduction in Theorem 3 takes a 3CNF formula

$$\phi(x_1, \ldots, x_n) = C_1 \wedge \cdots \wedge C_m$$

and produces an instance (N, M, c, χ) of GCE such that

$$\phi \in \text{3SAT iff } (N, M, c, \phi) \in \text{GCE}.$$

In this instance $c = \Theta(NM)$. Hence our reduction only shows that GCE is hard if c is rather large. So what happens if c is small? See next point.

2) What happens if c is small? In Section 6 we show that GCE is Fixed Parameter Tractable. In particular, the problem is in time $O(N^2M^2) + 2^{O(c^4+\log c)}$. This leads to the following open problem: find a framework to show that some problems in FPT are hard.

3) The 17×17 challenge can be rephrased as proving that $(17, 17, 4, \chi) \in$ GCE where χ is the empty partial coloring. This is a special case of GCE since none of the cell are pre-colored. It is possible that the case where χ is the empty coloring is easy. While we doubt this is true, we have not eliminated the possibility. How to deal with this issue? We define the problem that is probably the one we really want to find the complexity of.

Definition 6 GC is the following problem:

- *Input* $M, N, c \in \mathbb{N}$. The numbers N, M, c are in unary. So formally the input is $(1^M, 1^N, 1^c)$ where 1^x means $1 \cdots 1$ (x times).
- *Output* YES if there is a total c-coloring of $G_{N,M}$, NO otherwise.

GC stands for *Grid Coloring*.

Clearly GC $\in$ NP. Is this problem NP-hard? Alas no (assuming P $\neq$ NP).

Definition 7 A set $X \subseteq \{0,1\}^*$ is *sparse* if there exists a polynomial p such that

$$(\forall n)[|X \cap \{0,1\}^n| \leq p(n)]$$

Note that GC $\subseteq 1^* \times 1^* \times 1^*$ and hence is a sparse set.
We state a theorem that indicates sparse sets are not NP-hard.

Theorem 4

1. *(Mahaney [7], see also [4] for an alternative proof). If there exists a sparse set that is* NP*-hard by an m-reduction then* P = NP.
2. *(Karp-Lipton Theorem [5]) If there exists a sparse set that is* NP*-hard by a Turing-reductions then* $\Sigma_2^p = \Pi_2^p$.

Hence GC is likely to not be NP-complete under either m-reductions or Turing-Reductions.

If in the GC problem we express N, M in binary, then we cannot show that GC is in NP since the obvious witness, the coloring, is exponential in the length of the input. The formulation in binary does not get at the heart of the problem, since we believe it is hard because the number of possible colorings is large, not because N, M are large.

6 Fixed Parameter Tractability

Consider the problem where the number of colors is fixed at some c. We will see that this problem is Fixed Parameter Tractable by presenting two FPT algorithms for it.

The algorithms we present are not only FPT; they achieve this by means of a *polynomial kernel*. We discuss this in a subsection after the algorithms.

How well does the algorithm do in practice? We discuss this in a second subsection after the algorithm. In particular we discuss how much time and space the algorithm takes on one of our motivating problems: determining if there is a 4-coloring of $G_{17,17}$. The punchline will be that the algorithm takes too much time, and too much space, to be practical. Even so, we present the algorithm in the hope that some clever reader can come up with a way around these limitations, perhaps in practice if not in theory.

Definition 8 Let $c \in \mathbb{N}$. GCE_c is the following problem:

- *Input* $N, M \geq 1$ and χ a partial c-coloring of $G_{N,M}$. The numbers N, M are in unary.
- *Output* YES if there is an extension of χ to a total c-coloring of $G_{N,M}$, NO otherwise. (The algorithm can easily be modified to also *output* the extension as well as the YES.)

Clearly $\mathrm{GCE}_c \in \mathrm{DTIME}(c^{O(NM)})$. We will show that GCE_c is in time $O(N^2M^2) + 2^{O(c^6+\log c)}$ and then improve the algorithm to show that GCE_c is in time $O(N^2M^2) + 2^{O(c^4+\log c)}$.

Lemma 3 *For all* $u \geq 0$, $\sum_{s=0}^{u} \binom{u}{s} 2^s = 3^u$.

Proof

$$3^u = (1+2)^u = \sum_{s=0}^{u} \binom{u}{s} 2^s.$$

The last equality is by the binomial theorem.

□

Lemma 4 *Assume that $G_{N,M}$ is partially c-colored by χ. Let S be a set of cells that are not colored by χ. Let $|S| = s$. Let χ^* be a (not necessarily proper) extension of χ that colors all of the cells of S. We can determine whether χ^* is a proper c-coloring in time $O(NMs)$.*

Proof Here is the algorithm.

For each $(x,y) \in S$ do the following.

For each $1 \leq x' \leq N$ and $1 \leq y' \leq M$ determine if

$$\chi^*(x,y) = \chi^*(x',y) = \chi^*(x,y') = \chi^*(x',y').$$

If the equality ever holds then *output* NO and stop.

(If you get here then the equality never happened.) *Output* YES and stop.

The first for-loop goes s iterations. The second for-loop goes NM iterations. The body of the for-loop is $O(1)$ time. Hence the run time is $O(NMs)$.

□

Lemma 5 *Let $N, M, c \in \mathbb{N}$. Let χ be a partial c-coloring of $G_{N,M}$. Let U be the uncolored grid cells. Let $|U| = u$. There is an $O(cuNM3^u)$ time algorithm that will determine if χ can be extended to a full c-coloring.*

Proof For $S \subseteq U$ and $0 \le i \le c$ let

$$f(S,i) = \begin{cases} \text{YES} & \text{if } \chi \text{ can be extended to color } S \text{ using only colors } \{1,\ldots,i\}; \\ \text{NO} & \text{if not.} \end{cases}$$

We assume throughout that the coloring χ has already been applied.

We are interested in $f(U,c)$; however, we use a dynamic program to compute $f(S,i)$ for all $S \subseteq U$ and $0 \le i \le c$. Note the base cases:

1. $f(\emptyset, i) = \text{YES}$.
2. If $S \neq \emptyset$ then $f(S,0) = \text{NO}$.

Claim 1 Let $S \subseteq U$ and $1 \le i \le c$. Assume that, for all S' such that $|S'| < |S|$, for all $0 \le i \le c$, $f(S',i)$ is known. Also assume that $f(S,i-1)$ is known. Let $|S| = s$. Then $f(S,i)$ can be determined in time $O(NMs2^s)$.

Proof of Claim 1

If $f(S,i-1) = \text{YES}$ then clearly $f(S,i) = \text{YES}$. If not then here is our plan: We want to find (or show there is no such) nonempty $T \subseteq S$ such that the following holds:

- $f(S-T,i-1) = \text{YES}$. Hence there is a proper extension of χ which uses colors $\{1,\ldots,i-1\}$ on $S-T$. Let χ^* be that coloring. Note that, for all nonempty $T \subseteq S$ the value $f(S-T,i-1)$ is known since $|S-T| < |S|$.
- The extension of χ obtained by coloring all cells in T with i is a proper coloring.

If we find such a T then clearly $f(S,i) = \text{YES}$ by using χ^* to color $S-T$ with $\{1,\ldots,i-1\}$ and then coloring all cells in T with i. The algorithm below tries to find such a T. It will be clear that if the algorithm says YES then there is such a T and hence $f(S,i) = \text{YES}$. We will need to prove that if the algorithm says NO then $f(S,i) = \text{NO}$.

1. If $f(S,i-1) = \text{YES}$ then *output* YES and stop.
2. For all nonempty $T \subseteq S$ do the following (Note that there are $2^s - 1$ nonempty sets T.)
   (a) Let χ' be the extension of χ that colors all cells in T with i.
   (b) If χ' is not a proper coloring then go to the next T. Note that, by Lemma 4, this takes $O(NM|T|) = O(NMs)$ time.
   (c) If $f(S-T,i-1) = \text{NO}$ then go to the next T. Note that we know the value of $f(S-T,i-1)$ because $|S-T| < |S|$. This step takes $O(1)$ time.
   (d) If the algorithm got to this step then the following have happened:
      i. χ' is proper.
      ii. $f(S-T,i-1) = \text{YES}$. Hence there is a proper extension of χ which uses colors $\{1,\ldots,i-1\}$ on $S-T$. Let χ^* be that coloring.

   The extensions χ' and χ^* can easily be combined to properly extend χ to S with colors $\{1,\ldots,i\}$. Hence $f(S,i) = \text{YES}$ and we have the coloring itself.

3. If the algorithm got to this step then no T worked. We will show that in this case $f(S,i) = \text{NO}$.

The algorithm just specified has 2^s iterations that take $O(NMs)$ each. Hence the algorithm runs in time $O(NMs2^s)$.

Clearly if the above algorithm outputs YES then $f(S,i) = \text{YES}$. We need to show if the output is NO then $f(S,i) = \text{NO}$.

Claim 2: If the above algorithm outputs NO then $f(S,i) = \text{NO}$.

Proof of Claim 2: If the above algorithm outputs NO then, for all nonempty $T \subseteq S$ at least one of the following cases holds:

Case 1: The extension of χ to T formed by coloring cells of T with i is not proper.

Case 2: $f(S-T, i-1) = \text{NO}$.

Assume, by way of contradiction, that $f(S,i) = \text{YES}$. Let COL be a proper extension of χ to S. Let T be the subset of S that is colored i.

Since COL is a proper extension of χ, the extension of χ to T formed by coloring cells in T with i is proper. So Case 1 does not apply to T.

Since COL is a proper extension of χ there is a proper extension of χ to $S-T$ that only uses $\{1,\ldots,i-1\}$. So Case 2 does not apply to T.

Neither case applies, which is a contradiction.

End of Proof of Claim 2

End of Proof of Claim 1

We use Claim 1 in the following dynamic program.

1. *Input*(M,N,c,χ) such that χ is a partial c-coloring of $G_{N,M}$. Let U be the set of cells that are not colored by χ. Let $|U| = u$.
2. Set up a 2 dimensional table indexed by $S \subseteq U$ and $0 \le i \le c$.
3. Set $f(\emptyset, i) = \text{YES}$.
4. If $S \neq \emptyset$ then set $f(S,0) = \text{NO}$.
5. For $S \subseteq U$ (go in order of size)
   For $i = 1$ to c determine $f(S,i)$ using Claim 1 which takes time $O(NMs2^s)$.

Note that the amount of time taken in the inner loop, $O(NMs2^s)$ is independent of c. That is why c will only appear linearly in the run time.

The number of subsets of U that have s cells is $\binom{u}{s}$. Hence the total time spent in the loop is O-of the following:

$$\sum_{i=1}^{c}\sum_{s=0}^{u}\binom{u}{s} sNM2^s \le cuNM\sum_{s=0}^{u}\binom{u}{s}2^s$$

By Lemma 3, $\sum_{s=0}^{u}\binom{u}{s}2^s = 3^u$, so we obtain $O(cuNM3^u)$.

□

The following two theorems are easy; however, we include the proofs for completeness.

Lemma 6 *Assume $c+1 \le N$ and $c\binom{c+1}{2} < M$. Then $G_{N,M}$ is not c-colorable. Hence, for any χ, $(N,M,\chi) \notin \text{GCE}_c$.*

Proof Assume, by way of contradiction, that there is a c-coloring of $G_{N,M}$. Since every column has at least $c+1$ cells, each column must have two cells that have the same color. Map every column to some $(\{i,j\},a)$ such that the ith and the jth entry in that column are both colored a. Since the number of $(\{i,j\},a)$ is

$$\binom{N}{2} \times c \leq \binom{c+1}{2} \times c < M,$$

two columns must map to the same $(\{i,j\},a)$. This will create a monochromatic rectangle, which is a contradiction.

□

Lemma 7 *Assume $N \leq c$ and $M \in \mathbb{N}$. If χ is a partial c-coloring of $G_{N,M}$ then* $(N,M,\chi) \in \mathrm{GCE}_c$.

Proof The partial c-coloring χ can be extended to a full c-coloring as follows: for each column use a different color for each blank cell, making sure that all of the new colors in that column are different from each other and from the already existing colors given by χ.

□

Theorem 5 GCE_c *can be computed in time* $O(N^2M^2) + 2^{O(c^6+\log c)}$.

Proof

1. *Input* (N,M,χ).
2. If $N \leq c$ or $M \leq c$ then test if χ is a partial c-coloring of $G_{N,M}$. If so then *output* YES. If not then *output* NO. (This works by Lemma 7.) This takes time $O(N^2M^2)$. Henceforth we assume $c+1 \leq N,M$.
3. If $c\binom{c+1}{2} < M$ or $c\binom{c+1}{2} < N$ then *output* NO and stop. (This works by Lemma 6.)
4. The only case left is $c+1 \leq N,M \leq c\binom{c+1}{2}$. We will apply Lemma 5. Note that the number of uncolored cells, u, is

$$\leq NM \leq (c\binom{c+1}{2})^2 \leq (c \times \frac{(c+1)^2}{2})^2 = O(c^6).$$

Hence the run time of this step is

$$O(cuNM3^u) = O(cc^6c^63^{c^6}) = 2^{O(c^6+\log c)}.$$

Step 2 takes $O(N^2M^2)$, and Step 4 takes time $2^{O(c^6+\log c)}$. Hence the entire algorithm takes time $O(N^2M^2) + 2^{O(c^6+\log c)}$.

□

Can we do better? Yes, but it will require a result from a paper by Fenner et al. [2, Corollary 2.12].

Lemma 8 *Let* $1 \leq c' \leq c-1$.

1. *If $N \geq c+c'$ and $M > \frac{c}{c'}\binom{c+c'}{2}$ then $G_{N,M}$ is not c-colorable.*

2. *If $N \geq 2c$ and $M > 2\binom{2c}{2}$ then $G_{N,M}$ is not c-colorable. (This follows from a weak version of the $c' = c-1$ case of Part I.)*

Theorem 6 GCE_c *can be computed in time $O(N^2M^2) + 2^{O(c^4+\log c)}$.*

Proof

1. *Input* (N,M,χ). Let $u = NM$ which is a bound on the number of cells that are not colored.
2. If $N \leq c$ or $M \leq c$ then test if χ is a partial c-coloring of $G_{N,M}$. If so then *output* YES. If not then *output* NO. (This works by Lemma 7.) This takes time $O(N^2M^2)$.
3. Let $c' = N - c$ and $c'' = M - c$.
4. If $c' \leq c-1$ then do the following. Note that $N = c + c'$ and $M = c + c''$.
   (a) If $M > \frac{c}{c'}\binom{c+c'}{2}$, then *output* NO and stop. (This works by Lemma 8.)
   (b) If $M \leq \frac{c}{c'}\binom{c+c'}{2}$ then do the following. By Lemma 5 we can determine if χ can be extended to a total c-coloring in time $O(cuNM3^u)$. Since $c \leq N$ and $u \leq NM$ we have
   $$O(cuNM3^u) = O(NNMNM3^{NM}) \leq O(N^3M^2 3^{NM}) \leq 2^{O(NM+\log(NM))}.$$
   Note that $NM \leq (c+c')\frac{c}{c'}\binom{c+c'}{2}$. On the interval $1 \leq c' \leq c-1$ the function $(c+c')\frac{c}{c'}\binom{c+c'}{2}$ achieves its maximum when $c' = 1$, where it is
   $(c+1)c\binom{c+1}{2} \leq O(c^4)$. Hence $O(NM + \log(NM)) \leq O(c^4 + \log c)$. Therefore the runtime is bounded by $2^{O(c^4+\log c)}$.
   (c) If $N > \frac{c}{c''}\binom{c+c''}{2}$, or $N \leq \frac{c}{c''}\binom{c+c''}{2}$, then proceed similar to the last two steps.
5. (This is not code. This is commentary.) We have taken care of the cases where
   $N \leq c$
   $N = c+1$ (this is the $c' = 1$ case)
   $N = c+2$ (this is the $c' = 2$ case)
   $\vdots$
   $N = c+c-1$ (This is the $c' = c-1$ case).
   Hence we have taken care of all cases where $N \leq 2c-1$. Similarly, we have taken care of all cases where $M \leq 2c-1$. Henceforth we assume $2c \leq N,M$.
6. If $M > 2\binom{2c}{2}$ or $N > 2\binom{2c}{2}$ then *output* NO and stop. (This works by Lemma 8.)
7. The only case left is $2c \leq N,M \leq 2\binom{2c}{2} = O(c^2)$. By Lemma 5 we can determine if χ can be extended in time $O(cuNM3^u)$. Since $u = NM = O(c^4)$ we have time
   $$O(c(NM)^2 3^u) = O(cc^8 3^{c^4}) = 2^{O(c^4+logc)}.$$
   Step 2 and Step 4 together take time $O(N^2M^2) + 2^{O(c^4+\log c)}$.

□

6.1 Polynomial Kernels

Definition 9 Let A be a set in FPT, with parameter c. A has a *polynomial kernel* if there is a polynomial time (in the length of the input) algorithm that takes input P and produces a new problem P' such that

1. $P \in A$ iff $P' \in A$.
2. The size of P' is bounded by a function of c.

Our algorithms for GCE_c took the input and either solved the problem easily or concluded that the problem had size bounded by a polynomial in c. Hence our algorithms showed that GCE_c has a polynomial kernel.

6.2 Time and Space in the Real World

We have shown that GCE_c can be computed in time $O(N^2M^2) + 2^{O(c^4+\log c)}$. If the partially-filled grid has u empty spaces then the space is $O(c \times 2^u)$. Hence if the algorithm is run on the empty grid, so $u = NM$, the space is $O(c2^{NM})$.

We now look at what happens if $N = M = 17$, $c = 4$, and we start with the empty grid.

Time Even for small c the additive term $2^{O(c^4+\log c)}$ is the real time-sink. We generously assume the O-of term has constant 1 to get that the time is $2^{4^4+\log 4} = 2^{258} \sim 10^{77}$. We generously assume that every step takes one nano-second. Note that one nanosecond is 10^{-9} seconds. Hence the time is $10^{77} \times 10^{-9} = 10^{68}$ seconds. This is over 2^{200} years.

Space We generously assume the O-of term has constant 1 to get that the space is $4 \times 2^{17\times 17} = 4 \times 2^{289}$. This is roughly 10^{87} which is larger than Eddington's estimate of the number of protons in the universe (10^{80}).

A cleverer algorithm that reduces the time or space is desirable. By Theorem 3 the time cannot be made polynomial unless P=NP.

7 Open Problems

We reiterate briefly the open problems stated in Section 5 and add some new ones.

1. The problem we really want to study is the grid coloring extension problems with the empty grid. As noted in Section 5 this problem is a sparse set, and such sets cannot be NP-complete (unless P=NP). What is needed is a framework for proving that some sparse sets are likely not in P.
2. In Theorem 6 we showed that GCE_c can be computed in time $O(N^2M^2)+2^{O(c^4+\log c)}$. Can this be improved? The last term cannot be a polynomial in c (unless P=NP); however, it is plausible that a smaller exponential will suffice. Is there a proof that, under assumptions, the exponent cannot be lowered? What is needed is a framework to prove some FPT problems are hard.
3. Even without a theoretical improvement to our FPT algorithms, are there heuristics one can use to speed them up in practice?
4. We have studied grid colorings that avoid monochromatic *rectangles*. One can study avoiding monochromatic *squares*. The following is known:
   (a) By a corollary to the Gallai-Witt theorem (itself a generalization of van der Waerden's theorem): for all c there exists $N = N(c)$ such that, for all c-coloring of $G_{N,N}$, there is a monochromatic square (all corners the same

color). The proof gives an enormous upper bound even for $N(2)$; however, in reality $N(2)$ may be smaller, as we will see in the next few points.

(b) Bacher and Eliahou [1] showed the following:
   i. There is a 2-coloring of $G_{13,\infty}$ that has no monochromatic squares.
   ii. There is a 2-coloring of $G_{14,14}$ that has no monochromatic squares.
   iii. For any 2-coloring of $G_{14,15}$, there is a monochromatic square.
   iv. Hence the obstruction set is $\{G_{14,15}, G_{15,14}\}$.

With this in mind, we pose the following open question: is the following set NP-complete:

$$\{(N,M,c,\chi) : \chi \text{ is extendable to a } c\text{-coloring of } G_{N,M} \text{ with no monochromatic squares}\}.$$

5. One can also look at other shapes to avoid have monochromatic.

Acknowledgements We thank Amy Apon, Doug Chen, Jacob Gilbert, Matt Kovacs-Deak, Stasys Junka, Jon Katz, Clyde Kruskal, Nathan Hayes, Erika Melder, Erik Metz, and Rishab Pallepati for proofreading and discussion.

We thank Wing Ning Li for pointing out that the case of N,M binary, while it seems to not be in NP, is actually unknown.

We thank Jacob Gilbert, David Harris, and Daniel Marx for pointing out many improvements in the fixed parameter algorithm which we subsequently used.

We thank Tucker Bane, Richard Chang, Peter Fontana, David Harris, Jared Marx-Kuo, Jessica Shi, and Marius Zimand, for listening to Bill present these results and hence clarifying them.

We thank the referee for *many* helpful comments including a complete reworking of the proof of Theorem 3.

References

1. Bacher, R., Eliahou, S.: Extremal binary matrices without constant 2-squares. Journal of Combinatorics **1**(1), 77–100 (2010). `http://dx.doi.org/10.4310/JOC.2010.v1.n1.a6`
2. Fenner, S., Gasarch, W., Glover, C., Purewal, S.: Rectangle free colorings of grids (2012). `http://arxiv.org/abs/1005.3750`
3. Gasarch, W.: The 17×17 challenge. Worth \$289.00. This is not a joke (2009). November 30, 2009 entry on ComplexityBlog (Google Fortnow Blog)
4. Homer, S., Longpré, L.: On reductions of NP sets to sparse sets. J. Comput. Syst. Sci. **48**(2), 324–336 (1994). `https://doi.org/10.1016/S0022-0000(05)80006-6`
5. Karp, R., Lipton, R.: Some connections between nonuniform and uniform complexity classes. In: Proceedings of the Twelfth Annual ACM Symposium on the Theory of Computing, Los Angeles CA, pp. 302–309 (1980)
6. van Kreveld, M.J., de Berg, M.: Finding squares and rectangles in sets of points. BIT **31**(2), 202–219 (1991). `https://doi.org/10.1007/BF01931281`
7. Mahaney, S.: Sparse complete sets for NP: Solution to a conjecture of Berman and Hartmanis. Journal of Computer and System Sciences **25**, 130–143 (1982)
8. Steinbach, B. (ed.): Recent progress in the boolean domain. Cambridge Scholars Publishing, Newcastle upon Tyne,UK (2014)
9. Steinbach, B., Posthoff, C.: Extremely complex 4-colored rectangle-free grids: Solution of open multiple-valued problems. In: D.M. Miller, V.C. Gaudet (eds.) 42nd IEEE International Symposium on Multiple-Valued Logic, ISMVL 2012, Victoria, BC, Canada, May 14-16, 2012, pp. 37–44. IEEE Computer Society (2012). Https://doi.org/10.1109/ISMVL.2012.12

10. Steinbach, B., Posthoff, C.: The solution of ultra large grid problems. In: 21st International Workshop on Post-Binary USLI Systems (2012)
11. Steinbach, B., Posthoff, C.: Utilization of permutation classes for solving extremely complex 4-colorable rectangle-free grids. In: Proceedings of the IEEE 2012 international conference on systems and informatics (2012)
12. Steinbach, B., Posthoff, C.: Rectangle-free colorings of extremely complex grids using 4 colors. J. Multiple Valued Log. Soft Comput. **21**(5-6), 609–625 (2013). http://www.oldcitypublishing.com/journals/mvlsc-home/mvlsc-issue-contents/mvlsc-volume-21-number-5-6-2013/mvlsc-21-5-6-p-609-625/
13. Steinbach, B., Posthoff, C.: Solution of the last open four-colored rectangle-free grid: An extremely complex multiple-valued problem. In: 43rd IEEE International Symposium on Multiple-Valued Logic, ISMVL 2013, Toyama, Japan, May 22-24, 2013, pp. 302–309. IEEE Computer Society (2013). https://doi.org/10.1109/ISMVL.2013.51
14. Steinbach, B., Posthoff, C.: The last unsolved four-colored rectangle-free grid: The solution of extremely complex multiple-valued problems. J. Multiple Valued Log. Soft Comput. **25**(4-5), 461–490 (2015). http://www.oldcitypublishing.com/journals/mvlsc-home/mvlsc-issue-contents/mvlsc-volume-25-number-4-5-2015/mvlsc-25-4-5-p-461-490/

Conflict of interest

Not Applicable.