Local Problems, Planar Local Problems and Linear Time

Régis Barbanchon and Etienne Grandjean

GREYC, Université de Caen, 14032 Caen Cedex, France
{regis.barbanchon,etienne.grandjean}@info.unicaen.fr

Abstract. This paper aims at being a step in the precise classification of the many NP-complete problems which belong to NLIN (nondeterministic linear time complexity on random-access machines), but are seemingly not NLIN-complete. We define the complexity class LIN-LOCAL – the class of problems linearly reducible to problems defined by Boolean local constraints – as well as its planar restriction LIN-PLAN-LOCAL. We show that both "local" classes are rather computationally robust and that SAT and PLAN-SAT are complete in classes LIN-LOCAL and LIN-PLAN-LOCAL, respectively. We prove that some unexpected problems that involve some seemingly global constraints are complete for those classes. E.g., VERTEX-COVER and many similar problems involving cardinality constraints are LIN-LOCAL-complete. Our most striking result is that PLAN-HAMILTON – the planar version of the Hamiltonian problem – is LIN-PLAN-LOCAL and even is LIN-PLAN-LOCAL-complete. Further, since our linear-time reductions also turn out to be parsimonious, they yield new DP-completeness results for UNIQUE-PLAN-HAMILTON and UNIQUE-PLAN-VERTEX-COVER.

1 Introduction and Discussion

Since the publication of the famous Cook-Levin's theorem, two fundamental and complementary questions arise about time complexity: 1) What are the connections between deterministic time and nondeterministic time? 2) What is the precise complexity of usual NP-complete problems? It seems that any progress in proving complexity lower bounds for concrete NP-complete problems is conditioned by progress in both questions.

An interesting result concerning Question 1 is the separation result DTIME(n) $\neq$ NTIME(n) by Paul et al. [22] for linear time on Turing machines (TMs). However, its significance is weakened by the lack of any similar result known for other general-purpose computation models such as Random Access Machines (RAMs) and by the widespread feeling that linear time complexity on deterministic TMs is too restrictive.

Concerning Question 2, the second author defined and investigated (in a series of papers [12, 13, 15]), the classes DLIN and NLIN of problems which are (deterministically, resp. nondeterministically) decided in linear time on a certain type of RAM. It was argued [13, 15] that DLIN is robust and captures the notion of linear time as used in algorithmic design. At least as importantly, the class NLIN contains most of the natural NP-complete problems, including the 21 problems of [17], as asserted in [12, 13], and a few of them, e.g., RISA (Reduction of Incompletely Specified Automata [9, 11]), are also NLIN-complete under DTIME(n)-reductions. This implies:

$$\text{DLIN} \neq \text{NLIN iff RISA} \notin \text{DLIN, and}$$
$$\text{RISA} \notin \text{DTIME}(n), \text{since DTIME}(n) \subsetneq \text{NTIME}(n) \subseteq \text{NLIN}.$$

In contrast, as argued in [13], it is unlikely that SAT is NLIN-complete because it can be solved on a RAM by the following algorithm (so called NSUBLIN algorithm) that performs $O(n)$ deterministic steps *and only* $O(n/\log n)$ *nondeterministic steps*:

Input: A propositional formula F of m variables $p_0, \cdots, p_{m-1}$.
(N) Guess an assignment $I \in \{0,1\}^m$ for $p_0, \cdots, p_{m-1}$.
(D) Check that $I \models F$.

Note that $n = \mathit{length}(F) \geq \sum_{i<m} \mathit{length}(p_i) = \Omega(m \log m)$ yields complexity $m = O(n/\log n)$ for Phase (N), and that Phase (D) is performed in deterministic linear time. More generally, many classical NP-complete problems, including HAMILTONIAN-CYCLE, VERTEX-COVER, 3COL, etc., have similar NSUBLIN algorithms. Further, the planar versions of those problems, PLAN-SAT, PLAN-VERTEX-COVER, etc., seem still easier NP-complete problems since a divide-and-conquer strategy based on a planar separator theorem [20] can be applied to solve them in deterministic *sub-exponential* time $2^{O(n^{1/2})}$.

In an effort to investigate the conjecture DLIN $\neq$ NLIN (seemingly weaker than P $\neq$ NP), the present paper aims at being a step in the precise classification of the many problems which lie "somewhere below" NLIN. First, it is striking to observe that a number of problems are linearly equivalent to SAT (e.g., 3COL, 3DM, KERNEL), i.e., are linearly reducible to SAT and conversely, as observed by several authors [4, 6, 13, 24]. Second, logic plays a fundamental role: As Fagin proved that Existential Second-Order logic (ESO) on finite first-order structures exactly characterizes NP [7], Grandjean et al. [14] proved that, on *unary functional* first-order structures (i.e., finite structures over a signature that consists of relation and function symbols of arity ≤ 1), NLIN is exactly characterized by the logic ESO(1), that is the set of sentences of the form: $\exists \bar{f}\ \forall x\ \varphi$, where $\bar{f}$ is a list of relations and function symbols of arity ≤ 1 and φ is a quantifier-free formula. Since we conjecture that SAT and the NSUBLIN problems are not NLIN-complete, we look for a sub-logic of ESO(1) that can express them. A natural candidate is the set of sentences of the form:

$$\exists \bar{U}\ \forall x\ \varphi\,, \tag{1}$$

where $\bar{U}$ is a list of *unary relation* symbols (i.e., set symbols) and φ is a quantifier-free formula. Lautemann and Weinzinger [18] investigated such a logic they denoted Monadic-NLIN and proved that it expresses a number of natural NP-complete problems including SAT, 3COL and KERNEL, on some kind of ordered functional structures. Formulas (1) are special Monadic-ESO formulas, and unfortunately, it is well-known that this logic can only express "local" properties. In contrast, some easily computable (DLIN) properties such as graph connectivity cannot be defined in Monadic-ESO even in the presence of a built-in linear order [5, 8, 23]. Lautemann and Weinzinger [18] proved similar non-expressibility results for their logically defined class Monadic-NLIN. So we feel that the set of problems definable by Formulas (1) cannot be regarded as a complexity class. We think that any robust sequential time complexity class has to be closed under DLIN-reductions, because, on the one hand the sorting problem belongs to DLIN as proved in [13], and on the other hand we are convinced that DLIN is the minimal robust class for sequential time. This justifies the following definition of the complexity class LIN-LOCAL: A *local problem* is a set of unary functional first-order structures satisfying a sentence of the form (1) that we call a *local sentence*. A decision problem is LIN-LOCAL if it is DLIN-reducible to some local problem. The main feature of the class LIN-LOCAL is its *minimal* way to use nondeterminism by restricting it to happen in parallel at the end of any linear algorithm with an amount of $O(1)$ bits used per element.

A discussion of the notion of locality is required at this point. Any condition $(S \models \exists \bar{U}\ \forall x\ \varphi)$ is checked locally by consulting, for each element a of the structure S, the "colors" of a and of its "neighbors", i.e., the truth values of the monadic predicates of $a, f_0(a), \cdots, f_{k-1}(a)$, where $f_0, \cdots, f_{k-1}$ are the unary functions of S.[1]

Note that the "locality" results from the interaction of the local sentence with the underlying digraph $G(S) = (V, E)$ associated to S, that is defined by $V = Domain(S)$ and $E = \{(x, y) \in V^2,\ \exists f_i\ f_i(x) = y\}$. This graph is out-degree-bounded. It is natural to try to strengthen locality by adding one or both of the following semantical conditions over $G(S)$:

(B) Degree-Boundedness: $G(S)$ is *degree-bounded*. This can be obtained by requiring that each f_i be *bijective*.
(P) Planarity: $G(S)$ is *planar*.[2]

Regarding condition (P), we investigate a new complexity class, denoted LIN-PLAN-LOCAL, which is the class of decision problems DLIN-reducible to some planar local problem, i.e., a local problem over structures S whose underlying digraphs $G(S)$ are planar.

Let us now describe the main contributions of this paper. First, we justify the robustness of our complexity classes LIN-LOCAL and LIN-PLAN-LOCAL. Both are not modified if condition (B) is required together with several syntactical restrictions on φ, such as the use of at most two functions and at most one ESO monadic predicate. That strengthens the significance of the following series of

[1] Note that looking only at the immediate neighbors of a is possible because, as far as LIN-LOCAL problems are concerned, we can always assume w.l.g. that no functional composition occurs in φ.

[2] We will confuse a planar graph with one of its possible embeddings. This is justified by the fact that such a planar embedding is DLIN-computable [21].

inclusions of "linear classes" all conjectured to be strict:

$$\text{DLIN} \subseteq \text{LIN-PLAN-LOCAL} \subseteq \text{LIN-LOCAL} \subseteq \text{NLIN}\,.$$

One possible argument is that it would be a breakthrough if any of the following known inclusions in Turing Machine deterministic time classes could be improved:

$$\begin{aligned} \text{LIN-PLAN-LOCAL} &\subseteq \text{DTIME}(2^{O(n^{1/2})})\,, \\ \text{LIN-LOCAL} &\subseteq \text{DTIME}(2^{O(n/\log n)})\,, \\ \text{NLIN} &\subseteq \text{DTIME}(2^{O(n)})\,. \end{aligned}$$

Our second series of contributions are the proofs that many (planar) NP-complete problems are LIN-LOCAL-complete (resp. LIN-PLAN-LOCAL-complete). It is easily proved that SAT (resp. PLAN-SAT) is LIN-LOCAL-complete (resp. LIN-PLAN-LOCAL-complete). In other words, LIN-LOCAL (resp. LIN-PLAN-LOCAL) is exactly the set of problems DLIN-reducible to SAT (resp. PLAN-SAT). As a consequence, the numerous problems linearly equivalent to SAT (3COL, 3DM, KERNEL, etc., see [4] for a survey) are also LIN-LOCAL-complete, and we can prove that many of their planar restrictions are similarly LIN-PLAN-LOCAL-complete. The most surprising contributions of this paper are about some usual problems mixing local conditions with seemingly *global* (i.e., non-local) conditions:

- *cardinality conditions in the non-planar case*: problems such as VERTEX-COVER; All the cardinality problems are LIN-LOCAL and most of the usual NP-complete cardinality problems (e.g., VERTEX-COVER, DOMINATING-SET, MAX-SAT) are also LIN-LOCAL-complete.
- *connectivity conditions in the planar case*: the typical example is HAMILTON.[3] All the many variants of the planar HAMILTON problem are LIN-PLAN-LOCAL-complete. In particular, they are LIN-LOCAL, and hence all of them have a $O(2^{O(n^{1/2})})$ deterministic algorithm based on [20].

By lack of space, some of these results are presented in technical reports [1, 2, 3]. The "locality" of PLAN-HAMILTON contrasts with the conjecture that the general HAMILTON problem *is not* LIN-LOCAL. In that direction, Lautemann and Weinzinger [18] proved that HAMILTON does not belong to their class Monadic-NLIN, which means this problem *is not local* even in the presence of some kind of linear order.

Finally, we observe that all our reductions are not only DLIN-computable but also *parsimonious*. More precisely they state a bijective DLIN-computable correspondence between the solutions of the involved problems. As a side effect, this yields some new results about the status of some planar problems in

[3] The generic term HAMILTON refers to any of the many variants of the HAMILTONIAN-GRAPH problem: The input graph may be directed or not, and degree-bounded or not. We may test the existence of either a Hamiltonian cycle or a Hamiltonian path. In the latter case, the ends of the path may be fixed or free.

DP.[4] More importantly, the fact that our linear reductions can be made parsimonious strengthens our feeling that the LIN-LOCAL-complete problems (SAT, VERTEX-COVER, KERNEL, etc.) are very closely related to each other,

The paper is organized as follows: In Sect. 2, we define the classes LIN-LOCAL and LIN-PLAN-LOCAL and show their robustness. The LIN-LOCALITY (resp. LIN-PLAN-LOCALITY) of SAT (resp. PLAN-SAT) is also proved in this section. Section 3 is devoted to the LIN-LOCALITY of cardinality problems and Sect. 4 shows the LIN-PLAN-LOCALITY of PLAN-HAMILTON.

2 LIN-LOCAL Problems and SAT

In this section, we define precisely local structures, local sentences and our complexity classes LIN-LOCAL and LIN-PLAN-LOCAL. Moreover, we prove that those classes are rather robust under several changes of their definitions and that SAT and PLAN-SAT are respectively complete for them.

Definition 1 (Unary Structures). *A unary structure $S = (\mathcal{U}, \sigma)$ is a first-order structure over a finite universe $\mathcal{U}$ and a signature $\sigma = (\mathcal{F}, \mathcal{L})$, where $\mathcal{L}$ is a list of unary relations $L_0, \cdots, L_{p-1}$ (the* labelling *predicates), and $\mathcal{F}$ is a list of unary functions $f_0, \cdots, f_{k-1}$ (the* neighborhood *functions).*[5]

Example 1. The set of undirected graphs (without isolated vertex) can be represented, e.g., by a set $\mathcal{S}_G$ of unary structures $(\mathcal{U}, \sigma_G)$ with $\sigma_G = (\mathcal{F}_G, \mathcal{L}_G)$, $\mathcal{F}_G = (next, edge)$ and $\mathcal{L}_G = \emptyset$, as in Fig. 1. A graph $G(V, E)$ corresponds to a universe $\mathcal{U}$ of $2|E|$ elements, where each vertex $x \in V$ of degree d is represented by d elements $x_1, \cdots, x_d \in \mathcal{U}$ linked in a circular list via the function *next*, and where each edge (x, y) is represented by a circular list of length two linking two elements x_i and y_j via the function *edge*.

Definition 2 (Underlying Graph). *The underlying digraph of a unary structure $S = (\mathcal{U}, \sigma)$ with $\sigma = (\mathcal{F}, \mathcal{L})$ is defined by $G(S) = (V, E)$, where $V = \mathcal{U}$ and $E = \{(x, y) \in V^2,\ \exists f_i \in \mathcal{F}\ f_i(x) = y\}$. We say that S is planar if G is planar, i.e., has a planar embedding.*

Definition 3 (Local Problem, Description). *A* local problem *Π over a set $\mathcal{S}$ of unary σ-structures is the subset of $\mathcal{S}$ defined by $S \in \Pi$ iff $S \models \exists \mathcal{C}\ \forall x\ \varphi$ where $\mathcal{C}$ is a list of unary relation symbols $C_0, \cdots, C_{q-1}$ (the* coloring *predicates), and φ is a quantifier-free one-variable $(\sigma, \mathcal{C})$-formula. The tuple $(\mathcal{S}, \sigma, \mathcal{C}, \varphi)$ is called the description of the local problem and is identified to problem Π.*

Definition 4 (Planar Local Problem). *A* planar local problem *$\Pi = (\mathcal{S}, \sigma, \mathcal{C}, \varphi)$ is a local problem over a set $\mathcal{S}$ of planar structures.*

[4] A problem is in DP if it is defined by the conjunction of two conditions, one in NP, and the other one in co-NP.

[5] As usual, we confuse each relation or function symbol and its interpretation. Also, for convenience, we shall often view monadic predicates as functions to $\{0, 1\}$.

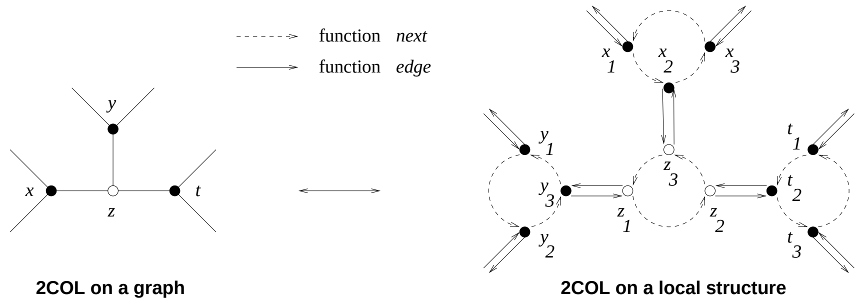

Fig. 1. 2COL on graphs and its translation on unary structures

Example 2. The problem $\Pi_{2col} = (\mathcal{S}_G, \sigma_G, \mathcal{C}_{2col}, \varphi_{2col})$, where $\mathcal{C}_{2col} = (Black)$ and φ_{2col} is $[Black(x) \Longrightarrow Black(next(x))] \wedge [Black(x) \Longrightarrow \neg Black(edge(x))]$, is the set of σ_G-structures associated to the graph problem 2COL. The problem $\Pi_{3col} = (\mathcal{S}_G, \sigma_G, \mathcal{C}_{3col}, \varphi_{3col})$ representing the graph problem 3COL can be similarly defined with $\mathcal{C}_{3col} = (Red, Green, Blue)$.

Definition 5 (Bijective Description, Minimal Description). *Let Π be a local problem of description $(\mathcal{S}, \mathcal{C}, \sigma, \varphi)$ with $\sigma = (\mathcal{F}, \mathcal{L})$. Π is a* bijective description *if $\mathcal{S}$ only uses bijective functions. Π is a* minimal description *if the equality is not used and no functional composition occurs in φ (i.e., φ is syntactically restricted to express conditions over the predicates f on x and its immediate neighborhood) and φ uses a minimal number of symbols: More precisely, at most one coloring predicate C_0, at most one labelling predicate L_0, and at most two neighborhood functions f_0 and f_1.*

Example 3. The local problem $(\mathcal{S}_G, \sigma_G, \mathcal{C}_{2col}, \varphi_{2col})$ is a minimal bijective description. The local problem $(\mathcal{S}_G, \sigma_G, \mathcal{C}_{3col}, \varphi_{3col})$ is a bijective but not minimal description, since it uses three coloring predicate symbols.

As previously argued, local problems cannot represent any consistent time complexity class if they are not closed under DLIN reductions. This justifies the following definition:

Definition 6 (LIN-LOCAL Class). *A decision problem Π is* LIN-LOCAL *if it is DLIN-reducible to a local problem Π'. Similarly, Π is* LIN-PLAN-LOCAL *if it is DLIN-reducible to a planar local problem Π'. For convenience, one says that any description of Π' is also a description of Π.*

It is easy to prove that (PLAN-)SAT is LIN-(PLAN)-LOCAL even with a bijective (but non-minimal) description [3]. It is trickier to prove the stronger theorem:

Theorem 1.
- *SAT is LIN-LOCAL and has a minimal bijective description.*
- *PLAN-SAT is LIN-PLAN-LOCAL and has a minimal bijective description.*

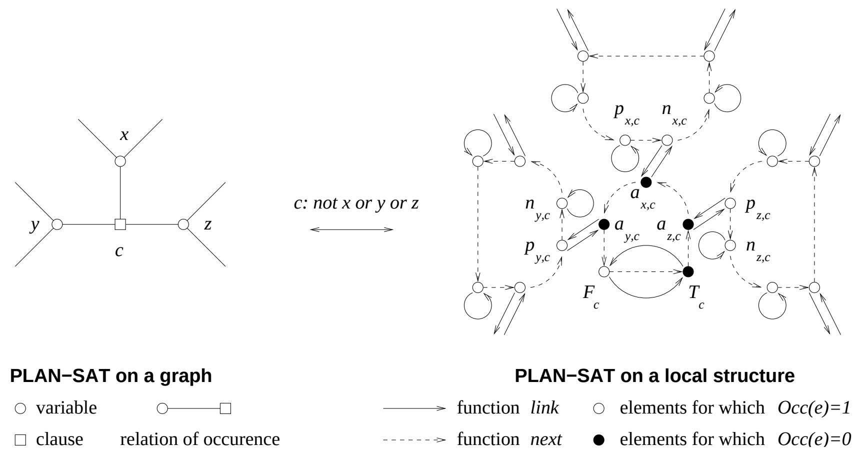

Fig. 2. Minimal bijective description for (PLAN-)SAT

Proof. We represent any (PLAN-)SAT instance by a (planar) $(\mathcal{F}, \mathcal{L})$-structure S where $\mathcal{F} = (next, link)$, $\mathcal{L} = (Occ)$, and $next$, $link$ are bijections.

- $\mathcal{U} = Domain(S)$ contains: for each clause c, the two elements T_c and F_c (the true and false constants), and for each occurrence of a variable v in a clause c , an element $a_{v,c}$ (the accumulators of the truth values in c) and two elements $p_{v,c}$ and $n_{v,c}$ (meant to represent v and $\neg v$ in c).
- The predicate Occ is mainly the label for occurrences: It maps all the $p_{v,c}$ and $n_{v,c}$ to 1 (true), and maps all the $a_{v,c}$ to 0 (false). A first trick is that it also maps all the F_c to 1 and all the T_c to 0.
- For each variable v, the function $next$ binds all the $p_{v,c}$ and $n_{v,c}$ in an alternating directed cycle. For each clause c, it also binds T_c, all the accumulators $a_{v,c}$ and F_c in this order in a directed cycle.
- The function $link$ mainly binds occurrences to accumulators: If a variable v occurs positively in a clause c then we define $link(p_{v,c}) = a_{v,c}$, $link(a_{v,c}) = p_{v,c}$, and the self-loop $link(n_{v,c}) = n_{v,c}$ (the symmetric case happens if v occurs negatively in c). The second trick is that for each clause c, we define the 2-cycle $link(T_c) = F_c$, $link(F_c) = T_c$.

The construction is clearly DLIN-computable and can be made planarity-preserving as shown in Fig. 2. The local formula uses *only one* color $True$ which holds the truth values of all the $p_{v,c}$ and $n_{v,c}$. For all the T_c (resp. F_c) it will be shown to be 1 (resp. 0), and for any accumulator $a_{v,c}$ it will hold the accumulated truth-values of the occurrences linked to all its successors via function $next$ up to F_c. The local sentence is $\exists\, True\ \forall x$:

$$\begin{aligned} &[Occ(x) \Longrightarrow (True(next(x)) \oplus True(x))] \\ \wedge\ &[\neg Occ(x) \Longrightarrow (True(x) \iff (True(next(x)) \vee True(link(x))))]\,. \end{aligned}$$

The first constraint coerces all the $n_{v,c}$ and $p_{v,c}$ of a variable v to have opposite values. Also, since $Occ(F_c) = 1$ and $next(F_c) = T_c$ for any clause c, it forces that F_c and T_c have opposite values. The second constraint implies that, for each clause c, the value of the predicate $True$ is non-increasing along the arrows *next* from T_c to F_c (including T_c because $Occ(T_c) = 0$). Since $True(F_c) \neq True(T_c)$ because of the first constraint, this implies that $True(F_c) = 0$ and $True(T_c) = 1$. This also means that $True(T_c)$, which accumulates the truth-value of the final occurrence and the truth-value of F_c, indeed holds a copy of the truth-bit of the final accumulator. It follows that there is at least one $a_{v,c}$ such that $True(a_{v,c}) = 1$, i.e., such that the truth value of v (represented by $True(n_{v,c})$ and $True(p_{v,c})$) satisfies c. □

Theorem 2. – *SAT is LIN-LOCAL-complete.*
– *PLAN-SAT is LIN-PLAN-LOCAL-complete.*

Theorems 1 and 2 imply that:

Corollary 1. *Each LIN-LOCAL or PLAN-LIN-LOCAL problem has a minimal bijective description.*

The LIN-LOCAL-hardness of SAT is obtained by a straightforward unfolding of the universal quantifier $\forall x$ over the universe of the unary structure and is left to the reader. The proof of the LIN-PLAN-LOCAL-hardness of PLAN-SAT is technically more involved because the planarity of structures must be preserved, despite of the possible compositions occurring in φ. It needs the following lemma, whose proof is presented in [3].

Lemma 1. *Any local sentence $\exists \mathcal{C}\ \forall x\ \varphi$ is logically equivalent to another local sentence $\exists \mathcal{C}'\ \forall x\ \varphi'$ in CNF which is composition-free, i.e., such that no functional composition occurs in φ'.*

Proof (of Theorem 2, sketch). Assume now that φ verifies Lemma 1, the reduction to PLAN-SAT of a planar local problem $\Pi = (\mathcal{S}, \sigma, \mathcal{C}, \varphi)$ over a structure $S = (\mathcal{U}, \sigma)$ consists in building a planarity-preserving SAT-gadget of size $O(d(x))$ to simulate φ around each element $a \in \mathcal{U}$ of degree $d(a) = d^-(a) + d^+(a)$. All the gadgets are then connected following the embedding of $G(S)$. In [3] we present a uniform way to build such a gadget using Lichtenstein's planar crossover-box for PLAN-SAT [19, 16]. □

3 LIN-LOCAL Problems and Cardinality Problems

In this section, we show that augmenting the local constraints by constraints over the cardinalities of the unary relations does not change the class LIN-LOCAL in the general case. This does not seem to hold in the plane.

Definition 7 (Cardinality Problem). *Define $\#C$ to be the cardinality of a unary relation C. A cardinality constraint is a constraint of the form $(\#C_i \perp K)$ or of the form $(\#C_i \perp \#C_j)$ where C_i and C_j are unary relations symbols, K is a constant, and $\perp$ is a comparison relation among $=$, $\leq$. A cardinality problem is* *a problem characterized by both local constraints and cardinality constraints, i.e., by some sentence of the extended form* $\exists \mathcal{C}\ (\forall x\ \varphi_1) \wedge \varphi_2$ *where* φ_1 *is a quantifier-free formula over* x*,* σ *and* $\mathcal{C}$*, and* φ_2 *is some Boolean combination of cardinality constraints.*

Example 4. A large number of natural NP-complete problems such as VERTEX-COVER, DOMINATING-SET, MAX-SAT, etc. [9] can be viewed as cardinality problems. E.g., the vertex-cover of a graph with less than K vertices can be converted into a cardinality problem $\Pi_{vc} = (\mathcal{S}_G, \sigma_{vc}, \mathcal{C}_{vc}, \varphi_{vc})$, where $\mathcal{C}_{vc} = (Cover, Count)$, and σ_{vc} is σ_G augmented by one monadic predicate *Repr* which identifies exactly one element per cycle of the function *next* (recall from Example 1 that such a cycle represents one vertex of the original graph). Clearly, Π_{vc} is defined by $\exists Cover, Count\ (\#Count \leq K) \wedge \forall x$:

$$\begin{aligned} &[Cover(x) \vee Cover(edge(x))] \\ \wedge &[Cover(x) \iff Cover(next(x))] \\ \wedge &[Count(x) \iff (Cover(x) \wedge Repr(x))] \,. \end{aligned}$$

We give a uniform argument that shows that each cardinality constraint is linearly SAT-expressible. The construction essentially uses a linear-sized SAT-adder that computes the correct cardinalities. As a consequence:

Theorem 3. *All the cardinality problems are LIN-LOCAL.*

Proof (sketch). Let C be a monadic predicate over a universe $\mathcal{U} = \{e_0, \cdots, e_{n-1}\}$. The main problem consists in building a SAT-gadget of size $O(n)$ which outputs a list of $\ell = O(\log n)$ Boolean variables holding the cardinality $\#C$ in binary.

W.l.g., assume that n is an exact power of 2, $n = 2^\ell$. Our adder uses a divide-and-conquer strategy on $\ell+1$ levels (numbered from 0 to ℓ): Level 0 consists of a list of 2^ℓ 1-bits numbers $(X_0^0, \cdots, X_0^{2^\ell-1})$, namely the bits $C(\mathcal{U})$ themselves. For any $1 < k \leq \ell$, level k consists of $2^{\ell-k}$ numbers $(X_k^0, \cdots, X_k^{2^{\ell-k}-1})$, such that $X_k^j = X_{k-1}^{2j} + X_{k-1}^{2j+1}$. Since the sum of two numbers of b bits fits in $b+1$ bits, each number at level k has $k+1$ bits. This way, the list of level ℓ consists of a single number X_ℓ^0 of $\ell+1$ bits holding $\#C$. Encoding all the binary additions with a carry-propagation scheme takes size and time $O(s(n))$ where $s(n)$ is the total number of bits over all levels, and $s(n) = \sum_{k=0}^{\ell}(k+1)2^{\ell-k} = O(n)$ as required. Finally, it is straightforward to build gadgets of size $O(\ell)$ to implement the arithmetic circuits for any comparison $\perp$ between any output cardinalities or constants. □

In [16], Hunt et al. show that #PLAN-VERTEX-COVER is #P-complete via a planarity-preserving and *weakly* parsimonious reduction from 1-EX-MONO-3SAT to VERTEX-COVER.[6] This problem is parsimoniously DLIN-equivalent to SAT, even in the plane. Since Hunt et al.'s reduction in [16] is also DLIN, this shows together with Theorem 3:

---

[6] Given a set of monotone 3-clauses (i.e., lists of 3 variables), 1-EX-MONO-3SAT is the problem of the existence of an assignment that satisfies *exactly* one variable in each 3-clause.

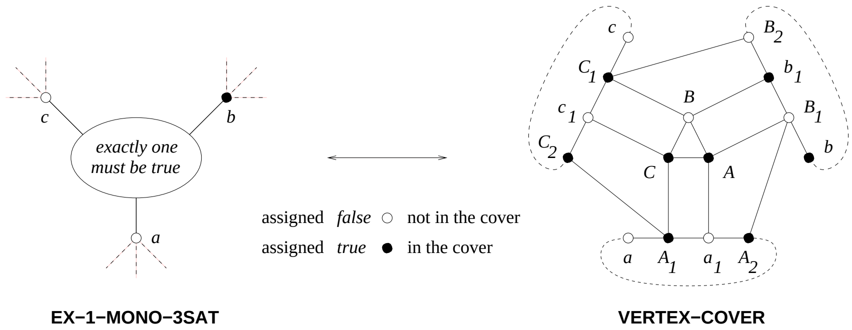

Fig. 3. The reduction from 1-EX-MONO-3SAT to VERTEX-COVER

Theorem 4. – *VERTEX-COVER is LIN-LOCAL-complete.*
– *PLAN-VERTEX-COVER is LIN-PLAN-LOCAL-hard.*

As noted above, Hunt et al.'s reduction is only *weakly* parsimonious. We improve it to make it parsimonious. This implies:

Theorem 5. *UNIQUE-PLAN-VERTEX-COVER is DP-complete under randomized polynomial reductions.*

Proof (sketch). Since it is known that UNIQUE-PLAN-1-EX-MONO-3SAT is DP-complete under randomized polynomial reductions, we only have to give a parsimonious polynomial reduction from PLAN-1-EX-MONO-3SAT to PLAN-VERTEX-COVER. Further, our reduction will be in DLIN. Let I be an input of PLAN-1-EX-MONO-3SAT with m 3-clauses (and hence $3m$ occurrences of variables). Our output-graph G for PLAN-VERTEX-COVER has $15m$ vertices and we ask for a cover K of cardinality $\leq 8m$. Each variable x in I of degree d (i.e., occurring d times) has an associated even cycle e_x in G of length $4d$ (i.e., 4 vertices by occurrence), and each 3-clause r in I has an associated triangle t_r in G. Occurrence-vertices are connected to 3-clause-vertices according to Fig. 3.

The truth-values of the variables a, b, c in I are witnessed by the membership to K of the corresponding vertices a, b, c in G. Simple arguments of cardinality developed in [3] imply the one-to-one correspondence between the configurations in I and G depicted in Fig. 3. □

4 LIN-PLAN-LOCAL Problems and PLAN-HAMILTON

In this section, we show that the many variants of the HAMILTON problem become LIN-PLAN-LOCAL when restricted to planar instances.

Theorem 6. *PLAN-HAMILTON is LIN-PLAN-LOCAL-complete.*

In [1], it was proved that all the cited variants of PLAN-HAMILTON are equivalent under parsimonious DLIN reductions. Thus, to show the LIN-LOCALITY of all these variants of PLAN-HAMILTON, we only have to find a DLIN-reduction from, say, the planar undirected Hamiltonian cycle to PLAN-SAT. The

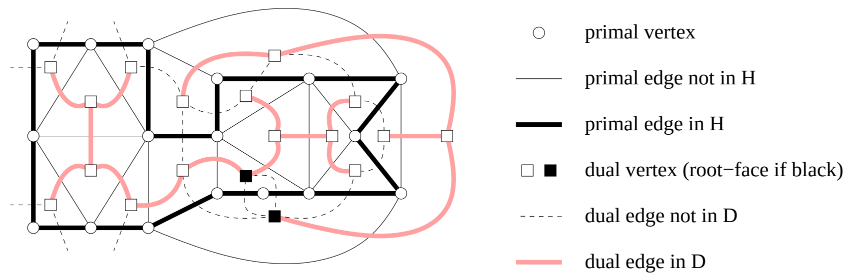

Fig. 4. A planar Hamiltonian cycle

converse DLIN reduction gives us the LIN-PLAN-LOCAL-hardness of PLAN-HAMILTON and is presented in [1] for space reasons. Since the latter reduction turns out to be parsimonious, it shows the DP-completeness of UNIQUE-PLAN-HAMILTON and answers a question stated as open in [16].

Corollary 2. *UNIQUE-PLAN-HAMILTON is DP-complete under randomized polynomial reductions.*

The rest of this paper is devoted to the proof of the LIN-PLAN-LOCALITY of PLAN-HAMILTON. Note that the problem of the (planar) Hamiltonian partition – i.e., the partition of the vertices of a graph into simple disjoint cycles – is easily DLIN-reducible to (PLAN-)SAT. However, in the general case, SAT does not seem to be able to detect whether there is only one cycle in such a partition. We show that it is indeed possible in the plane, using the following fact:

Fact 1 (Jordan Curve Theorem). *Any collection of k disjoint simple closed curves lying in a plane or a sphere split the surface into exactly $k+1$ maximal connected regions.*

Let $G(V,E)$ be a connected planar graph embedded in the plane, and let $G'(V',E')$ be its dual graph. Let H be a Hamiltonian partition in G. H is viewed as a set of edges. For any set of edges S, define $comp(S)$ to be the number of maximal connected components in S. Denote by H' the set of edges in G' that are dual to H. Define $D = E' \setminus H'$ (see Figs. 4 and 5). The following claim is an immediate consequence of Fact 1:

Claim 1. $comp(D) = comp(H) + 1$.

From now on, an arbitrary outer-face f_{out} is chosen for G. For any cycle C, denote $ext(C)$ (resp. $int(C)$) the exterior (resp. interior) region of C relative to f_{out}. We say that a cycle C_1 of H is a *max-cycle* if $C_2 \notin ext(C_1)$ for any cycle C_2 of H. Similarly, a cycle C_1 of H is a *min-cycle* if $C_2 \notin int(C_1)$ for any cycle C_2 of H. It is not difficult to see (though lengthy to prove) that:

Claim 2. *A connected component of D is acyclic (i.e.,is a tree) iff it lies:*

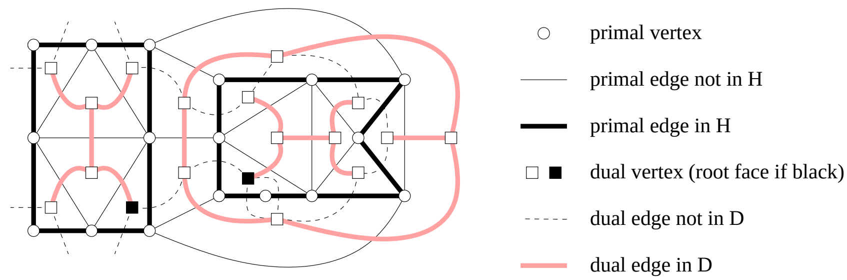

Fig. 5. A planar Hamiltonian partition into 2 disjoint cycles

– *either in the interior of a min-cycle,*
– *or in the exterior of a max-cycle, provided that H has no other max-cycle.*

The proof of Claim 2 is omitted for space reasons and is presented in [2]. (see Figs. 4 and 5 for an intuition). This gives us a first characterization of planar Hamiltonicity:

Claim 3. *$comp(H) = 1$ iff D is a forest of exactly two trees.*[7]

Proof. $(\Longrightarrow)$: If $comp(H) = 1$, then let C_1 be the unique cycle in H: by Claim 1, D has 2 components M_1 and M_2, lying in $int(C_1)$ and $ext(C_1)$ respectively. Since C_1 is both a min-cycle and the unique max-cycle in H, Lemma 2 applies twice, and M_1 and M_2 are both trees.

$(\Longleftarrow)$: In particular $comp(D) = 2$, and by Claim 1, $comp(H) = 1$. □

The idea of the reduction is to locally coerce H to be a Hamiltonian partition of disjoint cycles in the primal graph G while locally constraining D to be a forest of exactly two trees in the dual graph G'. While the former task is easy, the only way one can think of for the latter is to view the trees directed from the leaves to their roots and locally constrain each face but two (the two root faces) to select one adjacent face for its father in the same component of D. However, this leaves the possibility of generating non-tree parasitic components in D (called unicycles) that are trees whose roots are connected in a single circuit. Figure 5 shows such a unicycle. If unicycles occur in D, then H cannot be a Hamiltonian cycle, but these components are left undetected by the system of constraints described above because unicycles do not have any root. Hopefully, the following claim relaxes Claim 3 in a way that will allow us to forbid these unicycles.

Claim 4. *$comp(H) = 1$ iff two trees with adjacent roots exist in D.*

[7] This result is already known in connection with the 4-colors theorem, and the two trees of D are furthermore induced by their sets of vertices. Since this implies that G' is 4-colorable, this motivated a famous conjecture stating that any 3-connected cubic planar graph is Hamiltonian. The conjecture was proved false [25, 10].

Proof. ($\Longrightarrow$) : Since $comp(H) = 1$, Claim 3 applies and D is a forest of exactly two trees M_1 and M_2 lying in $int(C_1)$ and $ext(C_1)$ resp., where C_1 is the unique cycle of H. We only have to exhibit two adjacent roots r_1 and r_2: Choose an arbitrary edge e of C_1, and let $e' = (f, g)$ be its dual edge, such that f lies in $int(C_1)$ and g lies in $ext(C_1)$. $f \in M_1$ and $g \in M_2$, so we can choose $r_1 = f$ and $r_2 = g$.

($\Longleftarrow$) : Let M_1 and M_2 be two trees of D of adjacent roots r_1 and r_2. By Lemma 2, each one must lie either in the interior of a min-cycle or in the exterior of the unique max-cycle. Suppose $comp(H) > 1$, then there are two cases:

- $M_1 \in int(C_i)$ and $M_2 \in int(C_j)$ where C_i and C_j are disjoint min-cycles in H. Since G is planar and C_i and C_j are disjoint, any path from r_1 to r_2 in G' must contain an intermediate vertex lying in $ext(C_i) \cap ext(C_j)$. Hence, r_1 and r_2 are not adjacent, a contradiction.
- $M_1 \in int(C_i)$ and $M_2 \in ext(C_j)$ where C_i is a min-cycle and C_j is the unique max-cycle in H. Since the max-cycle is unique, $C_i \in int(C_j)$, and since $comp(H) > 1$, we conclude that $C_i \neq C_j$. Since G is a planar connected graph and C_i and C_j are disjoint, any path from r_1 to r_2 in G' contains a third vertex lying in $ext(C_i) \cap int(C_j)$. Hence, r_1 and r_2 are not adjacent, a contradiction. □

In [2], we show that we *do not need to guess* r_1 and r_2 because they can be chosen deterministically in linear time. Assume now r_1 and r_2 are fixed. We give the DLIN reduction from PLAN-HAMILTON to PLAN-SAT completing the proof of the LIN-PLAN-LOCALITY of PLAN-HAMILTON. For the sake of readability, we assume that the special clauses $1/N(\ell_1, \cdots, \ell_d)$ and $2/N(\ell_1, \cdots, \ell_d)$ – which are satisfied iff exactly one (resp. two) literal among the ℓ_i ($1 \leq i \leq d$) are assigned true – are available (these special clauses are easy to implement parsimoniously in the plane using standard clauses, as shown in [2]). Here is the SAT-system satisfied iff G is Hamiltonian: (see also Fig. 4):

- *Set of variables:* Each edge $e \in E \cup E'$ has an associated Boolean variable $thick_e$, asserting that "$e \in H \cup D$". Each face $f \in V'$ of degree d has d associated Boolean variables $father_f^{e'}$, one for each edge $e' = (f, g) \in E'$, asserting that "$e' \in D$ and g is the father of f in D".
- *H is a Hamiltonian partition of G:* For each vertex $v \in V$ of degree d, with incident edges $e_1, \cdots, e_d$, generate the constraint $2/N(thick_{e_1}, \cdots, thick_{e_d})$.
- *D equals $E' - H'$:* For each edge $e \in E$ and its dual edge e', generate the clauses $(thick_e \vee thick_{e'})$ and $(\neg thick_e \vee \neg thick_{e'})$.
- *Each face distinct from r_1 and r_2 has exactly one father:* For each face $f \in V'$ of degree d, $f \notin \{r_1, r_2\}$, with incident edges $e'_1, \cdots, e'_d$, generate the constraint $1/N(father_f^{e'_1}, \cdots, father_f^{e'_d})$.
- *Both adjacent roots r_1 and r_2 have no father:* For each edge $e' \in E'$ incident to a root $r \in \{r_1, r_2\}$, generate the unit clause $(\neg father_r^{e'})$.
- *D is consistently directed:* For each edge $e' = (f, g) \in E'$, generate the clauses $(father_f^{e'} \Longrightarrow thick_{e'})$, $(father_g^{e'} \Longrightarrow thick_{e'})$, $(thick_{e'} \Longrightarrow father_f^{e'} \vee father_g^{e'})$, and $(\neg father_g^{e'} \vee \neg father_f^{e'})$.

The system is built in time $O(|G|+|G'|) = O(|G|)$, including the computation of the dual graph G', and its correctness is an immediate consequence of Claims 3 and 4. In [2], we show how to embed our SAT-system in the plane for each face of G.

5 Conclusion and Further Research

In relation to our class LIN-LOCAL, Lautemann and Weinzinger [18] previously defined Monadic-NLIN as the class of decision problems whose inputs are unary functional structures S and which are defined by some local sentence $\exists \mathcal{C}\ \forall x\ \varphi$ on any expanded structure $(S, Succ)$ where $Succ$ is a list of "compatible" successor functions. [18] proved that the class Monadic-NLIN is logically robust – since it is closed under some logical quantifier-free reduction (which is DLIN-computable) – and meaningful since it contains a number of complete problems via that logical reduction, including SAT, KERNEL, etc. However, we think that this class cannot be viewed as a complexity class because such a class should be closed under some computational device, which is is seemingly not the case for Monadic-NLIN. The main interest of our classes LIN-LOCAL and LIN-PLAN-LOCAL is their great wealth of complete problems – under DLIN reductions – some of which are surprising. Our most significant and most technical result states that the problem PLAN-HAMILTON is LIN-PLAN-LOCAL, which means that it is "essentially local". We conclude this paper by suggesting further research related to our work:

1. Give other logical or algebraic or computational definitions of the classes LIN-LOCAL and LIN-PLAN-LOCAL.
2. Prove that HAMILTON is not LIN-LOCAL; That would be a breakthrough since it implies both LIN-LOCAL $\subsetneq$ NLIN and HAMILTON $\notin$ DLIN which yields DLIN $\neq$ NLIN.
3. Give an intrinsic characterization of the class of problems DLIN-reducible to HAMILTON (it includes interesting NP-complete problems about trees, connectivity, etc.).

References

[1] R. Barbanchon. Planar Hamiltonian problems and linear parsimonious reductions. Tech. report, Les Cahiers du GREYC 1, 2001. (postscript available at http://www.info.unicaen.fr/algo/publications).

[2] R. Barbanchon. The problems Sat and Hamilton are equivalent under linear parsimonious reductions in the plane. Tech. report, Les Cahiers du GREYC 4, 2001. (postscript available at http://www.info.unicaen.fr/algo/publications).

[3] R. Barbanchon and E. Grandjean. Local problems and linear time. Tech. report, Les Cahiers du GREYC 8, 2001. (postscript available at http://www.info.unicaen.fr/algo/publications).

[4] Nadia Creignou. The class of problems that are linearly equivalent to Satisfiability or a uniform method for proving NP-completeness. *Theoretical Computer Science*, 145(1–2):111–145, 1995.

[5] M. de Rougemont. Second order and inductive definability on finite structures. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 33:47–63, 1987.

[6] A. K. Dewdney. Linear time transformations between combinatorial problems. *Internat. J. Computer Math.*, 11:91–110, 1982.

[7] R. Fagin. Generalized first-order spectra and polynomial-time recognizable sets. *Complexity and Computation*, 7:43–73, 1974.

[8] R. Fagin. Monadic generalized spectra. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 21:89–96, 1975.

[9] M. R. Garey and D. S. Johnson. *Computers and Intractability*. W.H. Freeman and Co., 1979.

[10] M. R. Garey, D. S. Johnson, and R. E. Tarjan. The planar hamiltonian circuit problem is NP-complete. *SIAM Journal on Computing*, 5(4):704–714, 1976.

[11] E. Grandjean. A nontrivial lower bound for an NP problem on automata. *SIAM Journal on Computing*, 19:438–451, 1990.

[12] E. Grandjean. Linear time algorithms and NP-complete problems. *SIAM Journal on Computing*, 23(3):573–597, 1994.

[13] E. Grandjean. Sorting, linear time and the satisfiability problem. *Annals of Mathematics and Artificial Intelligence*, 16:183–236, 1996.

[14] E. Grandjean and F. Olive. Monadic logical definability of nondeterministic linear time. *Computational Complexity*, 7(1):54–97, 1998.

[15] E. Grandjean and T. Schwentick. Machine-independent characterizations and complete problems for deterministic linear time. Tech. report, Les Cahiers du GREYC 10, 1999. To appear in SIAM Journal on Computing.

[16] H. B. Hunt III, M. V. Marathe, V. Radhakrishnan, and R. E. Stearns. The complexity of planar counting problems. *SIAM Journal on Computing*, 27(4):1142–1167, August 1998.

[17] R. M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

[18] C. L. Lautemann and B. Weinzinger. Monadic-NLIN and quantifier-free reductions. In *CSL, 8th annual conference of the EACSL, Lect. Notes Comput. Sci.*, volume 1683 of *LNCS*, pages 322–337, 1999.

[19] D. Lichtenstein. Planar formulae and their uses. *SIAM Journal on Computing*, 11(2):329–343, 1982.

[20] R. J. Lipton and R. E. Tarjan. Applications of a planar separator theorem. *SIAM Journal on Computing*, 9(3):615–627, 1980.

[21] K. Mehlhorn and P. Mutzel. On the embedding phase of the Hopcroft and Tarjan planarity testing algorithm. *Algorithmica*, 16(2):233–242, 1996.

[22] W. J. Paul, N. Pippenger, E. Szeméredi, and W. T. Trotter. On determinism versus non-determinism and related problems. In *24th Annual Symposium on Foundations of Computer Science*, pages 429–438. IEEE Computer Society Press, 1982.

[23] T. Schwentick. On winning Ehrenfeucht games and Monadic NP. *Annals of Pure and Applied Logic*, 79(1):61–92, 1996.

[24] R. E. Stearns and H. B. Hunt III. Power indices and easier hard problems. *Mathematical Systems Theory*, 23(4):209–225, 1990.

[25] W.T. Tutte. On Hamilton circuits. *J. London Math. Soc.*, 21:98–101, 1946.