



HAL
open science

Undirected Graphs of Entanglement Two

Walid Belkhir, Luigi Santocanale

► **To cite this version:**

| Walid Belkhir, Luigi Santocanale. Undirected Graphs of Entanglement Two. 2007. hal-00144355v1

HAL Id: hal-00144355

<https://hal.science/hal-00144355v1>

Preprint submitted on 3 May 2007 (v1), last revised 30 Aug 2007 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Undirected Graphs of Entanglement Two

Walid Belkhir and Luigi Santocanale

Laboratoire d'Informatique Fondamentale de Marseille
Université de Provence

Abstract. Entanglement is a complexity measure of directed graphs that originates in fixed point theory. This measure has shown its use in designing efficient algorithms to verify logical properties of transition systems. We are interested in the problem of deciding whether a graph has entanglement at most k . As this measure is defined by means of games, game theoretic ideas naturally lead to design polynomial algorithms that, for fixed k , decide the problem. Known characterizations of directed graphs of entanglement at most 1 lead, for $k = 1$, to design even faster algorithms. In this paper we present an explicit characterization of *undirected* graphs of entanglement at most 2. With such a characterization at hand, we devise a linear time algorithm to decide whether an undirected graph has this property.

1 Introduction

Entanglement is a complexity measure of finite directed graphs introduced in [1, 2] as a tool to analyze the descriptive complexity of the Propositional Modal μ -calculus. Roughly speaking, its purpose is to quantify to what extent cycles are intertwined in a directed graph. Its game theoretic definition – by means of robbers and cops – makes it reasonable to consider entanglement a generalization of the tree-width of undirected graphs [3] to another kind of graphs, a role shared with other complexity measures appeared in the literature [4–7].

A peculiar aspect of entanglement, and also our motivation for studying it among the other measures, is its direct filiation from fixed point theory. Its first occurrence takes place within the investigation of the variable hierarchy [8, 9] of the Propositional Modal μ -Calculus [10]. The latter, hereby noted \mathbb{L}_μ , is nowadays a well known and appreciated logic, capable to express many computational properties of transition systems while allowing their verification in some feasible way. As a μ -calculus [11] \mathbb{L}_μ increases the expressive power of Hennessy-Milner logic, i.e. multimodal logic \mathbb{K} , by adding to it least and greatest fixed point operators that bind monadic variables. Showing that there are μ -formulas ϕ_n that are semantically equivalent to no formula with less than n bound variables is the variable hierarchy problem for a μ -calculus. It has been observed that the variable hierarchy is orthogonal to the alternation depth hierarchy – which crucially takes into account both the least and the greatest fixed point operators. As a matter of fact, the variable hierarchy can be defined and investigated as soon as just one fixed point operator is considered. Possibly, iteration theories [12] are a more appropriate framework within which to study this hierarchy.

The relationship between entanglement and the number of bound variables in a μ -term might be too technical to be elucidated here. Let us say, however, that entanglement roughly is a syntactic analogous of the variable hierarchy, the latter being defined only w.r.t. a given semantics. To argue in this direction, the relevant fact is Proposition 14 of [1], stating that the entanglement of a directed graph is the minimal feedback of its finite unravellings.

A second important topic in fixed point theory is the model checking problem for \mathbb{L}_μ . The main achievement of [1] states that parity games whose underlying graphs have bounded entanglement can be solved in polynomial time. This is a relevant result for the matter of verification, since model checking \mathbb{L}_μ is reducible in linear time to the problem of deciding the winner of a parity game. This result appears to be related to the variable hierarchy of \mathbb{L}_μ , even if we believe that the relationship still need to be clarified. Nonetheless, Berwanger’s result calls for the problem of *deciding whether a graph has entanglement at most k* , a problem which we address in this paper. When settled, we can try to exploit the main result of [1], for example by designing algorithms to model check \mathbb{L}_μ that may perform well in practice. An obvious procedure to solve parity games, which depends on having at disposal a fast algorithm to decide whether the entanglement of a parity game is small, is as follows. If the entanglement of a parity game is small, then Berwanger’s algorithm is used to solve it and, otherwise, a standard exponential algorithm [13, 14] is used. At present the complexity of the problem we address is unclear. We shall argue that, for fixed k , deciding whether a graph has entanglement at most k is a problem in the class P. The algorithms solving these problems can be combined to show that computing the entanglement of a graph is in the class EXPTIME. We have no reasons to believe that the problem is in NP.

In this paper we show that deciding whether an undirected graph G belongs to \mathcal{U}_2 , the class of undirected graphs of entanglement at most 2, can be solved in time $O(|V_G|)$. The algorithm that we shall present crucially depends on two characterizations of the class \mathcal{U}_2 . One of them proceeds by excluded subgraphs: an undirected graph belongs to \mathcal{U}_2 if and only if it does not contain (i) a simple cycle of length strictly greater than 4, (ii) a length 3 simple cycle whose vertices have all degree 3, (iii) a length 4 simple cycle with two adjacent vertices of degree 3. A second characterization constructs the class \mathcal{U}_2 from a class of atomic graphs, called the *molecules*, and an operation, the *legal collapse*, that glues together two graphs along a prescribed pair of vertices. The two characterizations may be appreciated on their own, independently of the algorithm they give rise. Entanglement is an intrinsically dynamic concept, due to its game theoretic definition. As such it is not an easy object of study, while the two characterizations prepare it for future investigations with standard mathematical tools. They also suggest that entanglement is a quite robust notion, henceforth worth being studied independently of its fix-point theoretic background. The algebraic characterization recalls the well known fact that graphs of fixed arbitrary tree-width may be constructed by means of an algebra of pushouts and relabelings

[15]. The algebra of legal collapses suggests that, for entanglement, it might be possible to develop an analogous generic algebraic framework.

Clearly, a work that still need to be carried out is to look for some useful characterization of *directed* graphs of entanglement at most k . At present, characterizations are known only for $k \leq 1$ [1, Proposition 3]. We believe that the results presented here suggest useful directions to achieve this goal. In particular, a suggestive path is to generalize the algebra of molecules and legal collapses to an undirected setting. This path might be a feasible one considering that many scientists have recently developed ideas and methods to lift some algebraic framework from an undirected to a directed setting. W.r.t. the algebra of entanglement, a source of ideas might be the recent development of directed homotopy theory from concurrency [16].

2 Entanglement Games

The entanglement of a finite digraph G , denoted $\mathcal{E}(G)$, was defined in [1] by means of some games $\mathcal{E}(G, k)$, $k = 0, \dots, |V_G|$. The game $\mathcal{E}(G, k)$ is played on the graph G by Thief against Cops, a team of k cops. The rules are as follows. Initially all the cops are placed outside the graph, Thief selects and occupies an initial vertex of G . After Thief's move, Cops may do nothing, may place a cop from outside the graph onto the vertex currently occupied by Thief, may move a cop already on the graph to the current vertex. In turn Thief must choose an edge outgoing from the current vertex whose target is not already occupied by some cop and move there. If no such edge exists, then Thief is caught and Cops win. Thief wins if he is never caught. The entanglement of G is the least $k \in \mathbb{N}$ such that k cops have a strategy to catch the thief on G . It will be useful to formalize these notions.

Definition 1. *The entanglement game $\mathcal{E}(G, k)$ of a digraph G is defined by:*

- *Its positions are of the form (v, C, P) , where $v \in V_G$, $C \subseteq V_G$ and $|C| \leq k$, $P \in \{\text{Cops}, \text{Thief}\}$.*
- *Initially Thief chooses $v_0 \in V$ and moves to $(v_0, \emptyset, \text{Cops})$.*
- *Cops can move from (v, C, Cops) to (v, C', Thief) where C' can be*
 1. C : *Cops skip,*
 2. $C \cup \{v\}$: *Cops add a new Cop on the current position,*
 3. $(C \setminus \{x\}) \cup \{v\}$: *Cops move a placed Cop to the current position.*
- *Thief can move from (v, C, Thief) to (v', C, Cops) if $(v, v') \in E_G$ and $v' \notin C$.*

Every finite play is a win for Cops, and every infinite play is a win for Thief. We let

$$\mathcal{E}(G) = \min\{k \mid \text{Cops have a winning strategy in } \mathcal{E}(G, k)\}.$$

It is not difficult to argue that there exist polynomial algorithms that, for fixed $k \geq 0$ decide on input G whether $\mathcal{E}(G) \leq k$. Such an algorithm constructs the game $\mathcal{E}(G, k)$ whose size is polynomial in $|V_G|$ and $|E_G|$, since k is fixed. Since

the game $\mathcal{E}(G, k)$ is clopen, i.e. it is a parity game of depth 1, it is well known [14] that such game can be solved in linear time w.r.t. the size of the graph underlying $\mathcal{E}(G, k)$.

In [1] the authors proved that $\mathcal{E}(G) = 0$ if and only if G is acyclic, and that $\mathcal{E}(G) \leq 1$ if and only if each strongly connected component of G has a vertex whose removal makes the component acyclic. Using these results it was argued that deciding whether a graph has entanglement at most 1 is a problem in NLOGSPACE.

While wondering for a characterization of graphs of entanglement at most 2, we observed that such a question has a clear answer for *undirected* graphs. To deal with this kind of graphs, we recall that an undirected edge $\{u, v\}$ is just a pair $(u, v), (v, u)$ of directed edges. We can use the results of [1] to give characterizations of undirected graphs of entanglement at most 1. To this goal, for $n \geq 0$ define the n -star of center x_0 , noted $\zeta_{x_0}^n$, to be the undirected graph (V, E) where $V = \{x_0, a_1, \dots, a_n\}$ and $E = \{\{x_0, a_1\}, \dots, \{x_0, a_n\}\}$. More generally, say that a graph is a star if it is isomorphic to some $\zeta_{x_0}^n$. Then we can easily deduce:

Proposition 2. *If G is an undirected graph, then $\mathcal{E}(G) = 0$ if and only if $E_G = \emptyset$, and $\mathcal{E}(G) \leq 1$ if and only if G is a disjoint union of stars.*

To end this section we state a Lemma that later will be used often. We remark that its scope does not restrict to undirected graphs.

Lemma 3. *If H is a subgraph of G then $\mathcal{E}(H) \leq \mathcal{E}(G)$.*

As a matter of fact, Thief can choose an initial vertex from H and then he can restrict his moves to edges of H . In this way he can simulate a winning strategy from $\mathcal{E}(H, k)$ to a winning strategy in $\mathcal{E}(G, k)$.

3 Molecules, Collapses, and the Class ζ_2

In this section we introduce a class of graphs and prove that the graphs in this class have entanglement at most 2. It will be the goal of the next sections to prove that these are all the graphs of entanglement at most 2.

Definition 4. *A molecule $\theta_{a,b}^{\varepsilon,n}$, where $\varepsilon \in \{0, 1\}$ and $n \geq 0$, is the undirected graph (V, E) with $V = \{a, b, c_1, \dots, c_n\}$ and*

$$E = \begin{cases} \{\{a, c_1\}, \dots, \{a, c_n\}, \{b, c_1\}, \dots, \{b, c_n\}\}, & \varepsilon = 0, \\ \{\{a, b\}, \{a, c_1\}, \dots, \{a, c_n\}, \{b, c_1\}, \dots, \{b, c_n\}\}, & \varepsilon = 1. \end{cases}$$

The glue points of a molecule $\theta_{a,b}^{\varepsilon,n}$ are a, b . Its dead points are c_1, \dots, c_n .

It is not difficult to prove that molecules have entanglement at most 2.

Definition 5. *Let G_1 and G_2 be two undirected graphs with $V_{G_1} \cap V_{G_2} = \emptyset$, let $a_1 \in V_{G_1}$ and $a_2 \in V_{G_2}$. The collapse of G_1 and G_2 on vertices a_1 and a_2 ,*

denoted $G_1 \bigoplus_{a_1, a_2}^z G_2$, is the graph G defined as follows:

$$\begin{aligned} V_G &= (V_{G_1} \setminus \{a_1\}) \cup (V_{G_2} \setminus \{a_2\}) \cup \{z\}, \text{ where } z \notin V_{G_1} \cup V_{G_2}, \\ E_G &= \{ \{x_1, y_1\} \in E_{G_1} \mid a_1 \notin \{x_1, y_1\} \} \cup \{ \{x_2, y_2\} \in E_{G_2} \mid a_2 \notin \{x_2, y_2\} \} \\ &\quad \cup \{ \{x, z\} \mid \{x, a_1\} \in E_{G_1} \text{ or } \{x, a_2\} \in E_{G_2} \}. \end{aligned}$$

We remark that \bigoplus is a coproduct in the category of pointed undirected graphs and, for this reason, this operation is commutative and associative up to isomorphism. The graph η , whose set of vertices is a singleton, is a neutral element. As we have observed, a molecule is an undirected graph coming with a distinguished set of vertices, its glue points. Let us call a pair (G, Gl) with $Gl \subseteq V_G$ a *glue graph*. For glue graphs we can define what it means that a collapse is legal.

Definition 6. *If G_1, G_2 are glue graphs, then we say that $G_1 \bigoplus_{a,b}^z G_2$ is a legal collapse if $a \in Gl_{G_1}$ and $b \in Gl_{G_2}$. We shall then use the notation $G_1 \overline{\bigoplus}_{a,b}^z G_2$ and define*

$$Gl_{G_1 \overline{\bigoplus}_{a,b}^z G_2} = (Gl_{G_1} \setminus \{a\}) \cup (Gl_{G_2} \setminus \{b\}) \cup \{z\},$$

so that $G_1 \overline{\bigoplus}_{a,b}^z G_2$ is a glue graph.

Observe that the graph η can be made into a unit for the legal collapse by letting $Gl_\eta = V_\eta$. Even if the operation \bigoplus is well defined only after the choice of the two glue points that are going to be collapsed, it should be clear what it means that a family of glue graphs is closed under legal collapses.

Definition 7. *We let ζ_2 be the least class of glue graphs containing the molecules, the unit η , and closed under legal collapses and graph isomorphisms.*

We need to make precise some notation and terminology. Firstly we shall abuse of notation and write

$$G = H \overline{\bigoplus}_v K$$

to mean that there exist subgraphs H, K of G such that $v \in Gl_G \cap V_H \cap V_K$ and G is isomorphic to the legal collapse $H \overline{\bigoplus}_{v,v}^z K$. Notice that if H and K are distinct from η , then v is an articulation point of G . Second, we shall say that a graph G belongs to ζ_2 to mean that there exists a subset $Gl \subseteq V_G$ such that the glue graph (G, Gl) belongs to ζ_2 . We can now state the main result of this section.

Proposition 8. *If G belongs to the class ζ_2 , then $\mathcal{E}(G) \leq 2$.*

Proof. Observe that, given a molecule $\theta_{a,b}^{\varepsilon,n}$ occurring in an algebraic expression for G , we can rearrange the summands of the algebraic expression to write

$$G = L \overline{\bigoplus}_a \theta_{a,b}^{\varepsilon,n} \overline{\bigoplus}_b R \tag{1}$$

where $L, R \in \zeta_2$. A Cops winning strategy in the game $\mathcal{E}(G, 2)$ is summarized as follows. If Thief occupies some vertex of the molecule $\theta_{a,b}^{\varepsilon,n}$, Cops will place its two cops on a and b , in some order. By doing that, Cops will force Thief to move (i) on the left component L , in which case Cops can reuse the cop on b on L , (ii) on the molecule $\theta_{a,b}^{\varepsilon,n}$, in which case Thief will be caught in a dead point of the molecule, (iii) on the right component R , in which case Cops can reuse the cop on a on R .

Cops can recursively use the same strategy in $\mathcal{E}(L, 2)$ and $\mathcal{E}(R, 2)$. The recursion terminates as soon as in the expression (1) for G we have $L = R = \eta$. \square

The reader will have noticed similarities between the strategy proposed here and the strategy needed in [1] to argue that undirected trees have entanglement at most 2. As a matter of fact, graphs in ζ_2 have an underlying tree structure. For a glue graph G , define the derived graph ∂G as follows: its vertices are the glue points of G , and $\{a, b\} \in E_{\partial G}$ if either $\{a, b\} \in E_G$ or there exists $x \in V_G \setminus Gl_G$ such that $\{a, x\}, \{x, b\} \in E_G$. The following Proposition is not difficult to prove.

Proposition 9. *A glue graph G is in ζ_2 if and only if ∂G is a forest, and each $x \in V_G \setminus Gl_G$ has exactly two neighbors, which moreover are glue points.*

4 Combinatorial Properties

The goal of this section is to setup the tools for the characterization Theorem 16. We deduce some combinatorial properties of undirected graphs of entanglement at most 2. To this goal, let us say that a simple cycle is long if its length is strictly greater than 4, and say otherwise that it is short. Also, let us call a simple cycle of length 3 (resp. 4) a triangle (resp. square).

Proposition 10. *An undirected graph G such that $\mathcal{E}(G) \leq 2$ satisfies the following conditions:*

- a simple Cycle of G is Short, (CS)
- a triangle of G has at least one vertex of degree 2, (No-3C)
- a square of G cannot have two adjacent vertices
of degree strictly greater than 2. (No-AC)

Condition (No-3C) excludes the graph scheme on the left of figure 1, made up of a triangle and 3 distinct Collapses. Notice that the vertices x, y, z in the figure might not be distinct. Condition (No-AC) excludes the graph scheme on the right of figure 1, made up of a square and two Adjacent Collapses. Even for this scheme x, y might not be distinct. We shall see with Theorem 16 that these properties completely characterize the class of undirected graphs of entanglement at most 2. Proposition 10 is an immediate consequence of Lemma 3 and of the following Lemmas 11, 12, 13.

For $n \geq 0$ let P_n be the path with n vertices and $n - 1$ edges: $V_{P_n} = \{0, \dots, n - 1\}$ and $\{i, j\} \in E_{P_n}$ iff $|i - j| = 1$. For $n \geq 3$, let C_n be the cycle with n vertices and edges: $V_{C_n} = \{0, \dots, n - 1\}$ and $\{i, j\} \in E_{C_n}$ iff $|i - j| \equiv 1 \pmod n$.

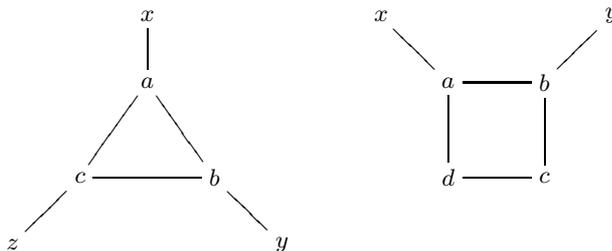


Fig. 1. The graphs $3C$ and AC

Lemma 11. *If $n \geq 5$ then $\mathcal{E}(C_n) \geq 3$.*

Proof. To describe a winning strategy for Thief in the game $\mathcal{E}(C_n, 2)$ consider that the removal of one or two vertices from C_n transforms such graph into a disjoint union $P_i + P_j$ with $i + j \geq n - 2 \geq 3$: notice in particular that $i \geq 2$ or $j \geq 2$. In a position of the form $(v, C, Thief)$ with $v \in C$, Thief moves to a component P_i with $i \geq 2$. From a position of the form $(v, C, Thief)$ with $v \notin C$, v in some component P_i , and $i \geq 2$, Thief moves to some other vertex in the same component. This strategy can be iterated infinitely often, showing that Thief will never be caught. \square

Lemma 12. *Let $3C$ be the graph on the left of figure 1. We have $\mathcal{E}(3C) \geq 3$.*

Proof. A winning strategy for Thief in the game $\mathcal{E}(3C, 2)$ is as follows. By moving on a, b, c , Thief can force Cops to put two cops there, say for example on a and b . Thief can then escape to c and iterate moves on the edge $\{c, z\}$ to force Cops to move one cop on one end of this edge. From a position of the form $(c, C, Thief)$ with $c \in C$, Thief moves to a free vertex among a, b . From a position of the form $(z, C, Thief)$ with $c \notin C$ Thief moves to c and forces again Cops to occupy two vertices among a, b, c . Up to a renaming of vertices, such a strategy can be iterated infinitely often, showing that Thief will never be caught.

Observe that the proof does not depend on x, y, z being distinct. \square

Lemma 13. *Let AC be the graph on the right of figure 1. We have $\mathcal{E}(AC) \geq 3$.*

Proof. By moving on a, b, c, d , Thief can force Cops to put two cops either on a, c or on b, d : let us say a, c . Thief can then escape to b and iterate moves on the edge $\{b, y\}$ to force Cops to move one cop on one end of this edge. From a position of the form $(b, C, Thief)$ with $b \in C$, Thief moves to a free vertex among a, c . From a position of the form $(y, C, Thief)$ with $b \notin C$ Thief moves to b and forces again Cops to occupy either a, c or b, d . Up to a renaming of vertices, such a strategy can be iterated infinitely often, showing that Thief will never be caught. Again, we observe that the strategy does not depend on x, y being distinct. \square

We end this section by pointing out that $\mathcal{E}(C_n) = \mathcal{E}(3C) = \mathcal{E}(AC) = 3$ ($n \geq 5$).

5 Characterization of Entanglement at Most 2

In this section we accomplish the characterization of the class of undirected graphs of entanglement at most 2: we prove that this class coincides with ζ_2 .

The following Lemma is the key observation by which the induction works in the proof of Proposition 15. It is worth, before stating it, to recall the difference between \bigoplus , the collapse of two ordinary undirected graphs, and $\overline{\bigoplus}$, the legal collapse of two glue graphs.

Lemma 14. *Let G be an undirected graph satisfying (No-3C) and (No-AC). If $G = \theta_{v,b}^{\varepsilon,n} \bigoplus_b H$ and $H \in \zeta_2$, then there is a subset $Gl' \subseteq V_G$ such that (H, Gl') is a glue graph in ζ_2 , $b \in Gl'$, and moreover G is the result of the legal collapse $G = \theta_{v,b}^{\varepsilon,n} \overline{\bigoplus}_b (H, Gl')$. Consequently, $G \in \zeta_2$, with v a glue point of G .*

The proof of the Lemma doesn't present difficulties.

Proposition 15. *If G is an undirected graph satisfying (CS), (No-3C), and (No-AC), then $G \in \zeta_2$.*

Proof. The proof is by induction on $|V_G|$. Clearly the Proposition holds if $|V_G| = 1$, in which case $G = \eta \in \zeta_2$. Let us suppose the Proposition holds for all graphs H such that $|V_H| < |V_G|$.

If all the vertices in G have degree less than or equal to 2, then G is a disjoint union of paths and cycles of length at most 4. Clearly such a graph belongs to ζ_2 . Otherwise, let v_0 be a vertex such that $\deg_G(v_0) \geq 3$ and consider the connected components G_ℓ , $\ell = 1, \dots, h$, of the graph $G \setminus \{v_0\}$. Let $G_\ell^{v_0}$ be the subgraph of G induced by $V_{G_\ell} \cup \{v_0\}$. We shall show that this graph is of the form

$$G_\ell^{v_0} = \theta_{v_0,v_1}^{\varepsilon,m} \bigoplus_{v_1} H, \quad (2)$$

for some $\varepsilon \in \{0, 1\}$, $m \geq 0$, and a graph $H \in \zeta_2$.

Clearly, if G_ℓ is already a connected component of G , then $G_\ell \in \zeta_2$ by the inductive hypothesis. We can pick any $v_1 \in V_{G_\ell}$ and argue that formula (2) holds with $m = \varepsilon = 0$, $H = G_\ell$.

Otherwise, let $\mathcal{N}_\ell = \{a_1, \dots, a_n\}$, $n \geq 1$, be the set of vertices of $G_\ell^{v_0}$ at distance 1 from v_0 . We claim that either the subgraph of G_ℓ induced by \mathcal{N}_ℓ , noted \mathcal{N}_{G_ℓ} , is a star or there exists a unique $v_1 \in G_\ell$ at distance 1 from \mathcal{N}_ℓ , and moreover the subgraph of G_ℓ induced by $\mathcal{N}_\ell \cup \{v_1\}$ is a star. In both cases, a vertex of such a star which is not the center has degree 2 in G .

(i) If $E_{\mathcal{N}_{G_\ell}} \neq \emptyset$, then \mathcal{N}_{G_ℓ} is a star. Let us suppose that $\{a_1, a_2\} \in E_{G_\ell}$. Since G_ℓ is connected, if $a_k \in \mathcal{N}_\ell \setminus \{a_1, a_2\}$ then there exists a path from a_k to both a_1 and a_2 . Condition (CS) implies that either $\{a_1, a_k\} \in E_{G_\ell}$, or $\{a_k, a_2\} \in E_{G_\ell}$. If $x_0 \in V_{G_\ell} \setminus \{a_2\}$ then there cannot be a simple path $a_k \dots x_0 \dots a_1$ otherwise $v_0 a_k \dots x_0 \dots a_1 a_2 v_0$ is a long cycle. Therefore, a simple path from a_k to a_1 is of the form $a_k a_1$ or $a_k a_2 a_1$. By condition (No-3C) it is not the case that $\{a_k, a_1\}, \{a_k, a_2\} \in E_{G_\ell}$, otherwise $\{v_0, a_1, a_2, a_k\}$ is a clique of cardinality 4. Finally, if $\{a_k, a_1\} \in E_{G_\ell}$ and $a_l \in \mathcal{N}_\ell \setminus \{a_1, a_2, a_k\}$, then $\{a_l, a_1\} \in E_{G_\ell}$ as well, by condition (CS), otherwise $v_0 a_k a_1 a_2 a_l v_0$ is a long cycle. Therefore, if $|\mathcal{N}_\ell| > 2$,

then \mathcal{N}_{G_ℓ} is a star with a prescribed center, which we can assume to be a_1 . Since $\deg_G(v_0) \geq 3$, by condition (No-3C) only a_1 among vertices in \mathcal{N}_ℓ may have degree greater than 2. Otherwise $|\mathcal{N}_\ell| = 2$ and again at most one among a_i , $i = 1, 2$, has $\deg_G(a_i) > 2$. Again, we can assume that $\deg_G(a_2) = 2$. We deduce that the subgraph of $G_\ell^{v_0}$ induced by $\{v_0\} \cup \mathcal{N}_\ell$ is of the form $\theta_{v_0, a_1}^{1, n-1}$.

(ii) If $E_{\mathcal{N}_{G_\ell}} = \emptyset$, then we distinguish two cases. If $|\mathcal{N}_\ell| = 1$, then the subgraph of $G_\ell^{v_0}$ induced by $\{v_0\} \cup \mathcal{N}_\ell$ is $\theta_{v_0, a_1}^{1, 0}$. Otherwise, if $|\mathcal{N}_\ell| \geq 2$, between any two distinct vertices in \mathcal{N}_ℓ there must exist a path in G_ℓ , since G_ℓ is connected. By condition (CS), if $a_i \dots x_{i,j} \dots a_j$ is a simple path from a_i to a_j with $x_{i,j} \in V_{G_\ell} \setminus \mathcal{N}_\ell$, then $\{a_i, x_{i,j}\}, \{a_j, x_{i,j}\} \in E_{G_\ell}$. Also (CS) implies that, for fixed i , $x_{i,k} = x_{i,j}$ if $k \neq j$, otherwise $v_0 a_k x_{i,k} a_i x_{i,j} a_j v_0$ is a long cycle. We can also assume that $x_{i,j} = x_{j,i}$, and therefore $x_{i,j} = x_{i,k} = x_{l,k}$ whenever $i \neq j$ and $l \neq k$. Thus we can write $x_{i,j} = v_1$ for a unique v_1 at distance 2 from v_0 . Since $|\mathcal{N}_\ell| \geq 2$ and $\deg_G(v_0) \geq 3$, condition (No-AC) implies that $\deg_G(a_i) = 2$ for $i = 1, \dots, n$. We have shown that in this case the subgraph of $G_\ell^{v_0}$ induced by $\mathcal{N}_\ell \cup \{v_0, v_1\}$ is a molecule $\theta_{v_0, v_1}^{0, n}$, with $n \geq 2$.

Until now we have shown that (2) holds with H a graph of entanglement at most 2. Since for such a graph $|V_H| < |V_G|$, the induction hypothesis implies $H \in \zeta_2$. Lemma 14 in turn implies that $G_\ell^{v_0} \in \zeta_2$, with v_0 a glue point of $G_\ell^{v_0}$. Finally we can use

$$G = G_1^{v_0} \overline{\oplus}_{v_0} G_2^{v_0} \overline{\oplus}_{v_0} \dots \overline{\oplus}_{v_0} G_h^{v_0},$$

to deduce that $G \in \zeta_2$. □

We can now state our main achievement.

Theorem 16. *For a finite undirected graph G , the following are equivalent:*

1. G has entanglement at most 2,
2. G satisfies conditions (CS), (No-3C), (No-AC),
3. G belongs to the class ζ_2 .

As a matter of fact, we have shown in the previous section that 1 implies 2, in this section that 2 implies 3, and in section 3 that 3 implies 1.

6 A Linear Time Algorithm

In this section we present a linear time algorithm that decides whether a connected undirected graph G has entanglement at most 2. As we shall see, the algorithm essentially depends on the fact that we can test whether a graph G contains a long cycle in time $O(|V_G|)$. It is then immediate to argue that there exists a linear time algorithm to test whether an undirected graph has entanglement at most 2. To this goal, observe first that the tree width of a graph satisfying condition (CS) is at most 3, by [17]. Considering that conditions (No-3C) and (No-AC) are expressible in first order logic, it is then possible to use Courcelle's

Theorem [15, §6.3] to argue that (No-3C) and (No-AC) can be verified in linear time on graphs with no long cycles.

An algorithm LONG-CYCLE-IN(G) that detects whether G has some long cycle – where long means here strictly longer than a fixed k – is obtained by modifying the standard algorithm to detect cycles by Deep-First-Search as follows. An array δ keeps track of the distances of nodes v from the root in the DFS tree, the value $\delta(v)$ being computed as soon as the vertex v is discovered. As soon as a back edge (d, a) (i.e. d is being visited and a is already visited) is discovered, the value $\delta(d) - \delta(a) + 1$ is computed. If this number is greater than k then a long cycle has been detected, the algorithm halts and accepts. Otherwise it continues as usual in a DFS and rejects if all the edges have been explored. Since a graph G with no long cycles has at most $(k + 1)|V| - 1$ undirected edges, the algorithm LONG-CYCLE-IN(G) runs in time $O(|V_G|)$.

The algorithm we present in the following is suggested by Theorem 16, stating that $\mathcal{E}(G) \leq 2$ if and only if $G \in \zeta_2$. The algorithm takes as inputs connected graphs but it is straightforward to modify it to consider graphs that are not necessarily connected.

```

1  ENTANGLEMENT-TWO( $G$ )
2  // Input an undirected connected graph  $G$ , accept if  $G \in \zeta_2$ 
3  if LONG-CYCLE-IN( $G$ ) then reject
4  foreach  $v \in V_G$ 
5    do color( $v$ )  $\leftarrow$  Not_Visited
6         $\mathcal{I}$ -type( $v$ )  $\leftarrow$  Not_Affected
7        deg $_G$ ( $v$ )  $\leftarrow$  | $vE$ |
8   $v_0 \leftarrow$  a vertex s.t. deg $_G$ ( $v_0$ )  $\neq$  2
9  if  $v_0 =$  Not_Found then accept
10 VISIT( $v_0$ )
11 accept

1  VISIT( $v$ )
2  color( $v$ )  $\leftarrow$  Visited
3   $N \leftarrow$  NEXT-GLUE-POINTS( $v$ )
4  foreach  $x \in N$ 
5    do VISIT( $x$ )
6  return

```

The algorithm is similar to a DFS that explores and constructs on the fly the graph ∂G mentioned within Proposition 9. A main difference is that the algorithm immediately tests whether G has long cycles. If the test fails, then G is sparse, the size of E_G being linear in V_G , and neighbors of a glue point in the graph ∂G can be computed efficiently. The algorithm begins by choosing a vertex that necessarily has to be a glue point of G provided $G \in \zeta_2$. If such a vertex is not found, then the graph G is a cycle of length at most 4, and the algorithm accepts. The computation of the neighbors in the graph ∂G is performed by the function NEXT-GLUE-POINTS which also verifies that the subgraph of G , rooted at v , connected to v , and not yet explored, is a collapse on v of graphs of the form $\theta_{v,w}^{\varepsilon,n} \bigoplus_w H$. The proof of Lemma 14 shows that if $\varepsilon + n > 0$, then a

graph of the form $\theta_{v,w}^{\varepsilon,n} \bigoplus_w H$ belongs to ζ_2 if and only if H belongs to ζ_2 with w a glue point of H . Thus the Lemma ensures that the algorithm is sound and correct.

The implementation of the function NEXT-GLUE-POINTS relies on the following considerations. Let G be a graph satisfying (CS), $v \in V_G$, and denote by xE^v the set $\{y \in xE \mid y \neq v\}$. If $x \in vE$ and $\deg_G(x) = 2$, let $s(x)$ be the unique $y \in xE^v$ and $S_2 = \{s(x) \mid x \in vE \text{ and } \deg_G(x) = 2\}$. Let $S_{n_2} = \{x \mid x \in vE \text{ and } \deg_G(x) \neq 2\} \setminus S_2$. Let \mathcal{I} be the family of subsets $\{S_2, S_{n_2}\} \cup \{xE^v \mid x \in S_{n_2}\}$, and observe that $S_2 \cap S_{n_2} = \emptyset$ by definition.

Lemma 17. *A graph G satisfying (CS) is of the form*

$$(\theta_{v,x_1}^{\varepsilon_1,n_1} \bigoplus_{x_1} H_1) \bigoplus_v \dots \bigoplus_v (\theta_{v,x_h}^{\varepsilon_h,n_h} \bigoplus_{x_h} H_h)$$

if and only if $A \cap B = \emptyset$, for each pair $A, B \in \mathcal{I}$, $A \neq B$.

```

1  NEXT-GLUE-POINTS( $v$ )
2  glue_points  $\leftarrow$   $\emptyset$ 
3  foreach  $x \in vE$ 
4    do  $\mathcal{I\_type}(x) \leftarrow$  Not_Affected
5  foreach  $x \in vE$ 
6    do if color( $x$ ) = Not_Visited and  $\deg_G(x) = 2$ 
7      then color( $x$ )  $\leftarrow$  Visited
8      foreach  $y \in xE$ 
9        do if  $y \neq v$  and  $\mathcal{I\_type}(y) =$  Not_Affected
10       then  $\mathcal{I\_type}(y) \leftarrow S_2$ 
11       glue_points  $\leftarrow$  glue_points  $\cup$   $\{y\}$ 
12 foreach  $x \in vE$ 
13   do if color( $x$ ) = Not_Visited and  $\deg_G(x) \neq 2$ 
14     and  $\mathcal{I\_type}(x) \neq S_2$ 
15     then if  $\mathcal{I\_type}(x) \neq$  Not_Affected then reject
16      $\mathcal{I\_type}(x) \leftarrow S_{n_2}$ 
17     foreach  $y \in xE$ 
18       do if color( $x$ ) = Not_Visited
19         then if  $\mathcal{I\_type}(y) \neq$  Not_Affected then reject
20          $\mathcal{I\_type}(y) \leftarrow N_{S_{n_2}}$ 
21     glue_points  $\leftarrow$  glue_points  $\cup$   $\{x\}$ 
22 return glue_points

```

We leave the reader to verify that the algorithm visits the adjacency list of each vertex a bounded numbers of times. In view of the fact that if a graph G satisfies (CS) then $|E_G| \leq 5|V_G|$, we can therefore state:

Fact. *Algorithm ENTANGLEMENT-TWO(G) runs in time $O(|V_G|)$.*

Finally, we add a remark on the tree width of undirected graphs in the class ζ_2 . The reader may have noticed that graphs in ζ_2 have tree width at most 2. However, in view of [1, Proposition 8], not every graph of tree width 2 has entanglement 2: in particular the tree width of the graph 3C is 2, its entanglement is 3.

References

1. Berwanger, D., Grädel, E.: Entanglement—a measure for the complexity of directed graphs with applications to logic and games. In: Logic for programming, artificial intelligence, and reasoning. Volume 3452 of Lecture Notes in Comput. Sci. Springer, Berlin (2005) 209–223
2. Berwanger, D.: Games and Logical Expressiveness. PhD thesis, RWTH Aachen (2005)
3. Seymour, P.D., Thomas, R.: Graph searching and a min-max theorem for tree-width. *J. Combin. Theory Ser. B* **58**(1) (1993) 22–33
4. Gottlob, G., Leone, N., Scarcello, F.: Hypertree decompositions: A survey. In: MFCS '01: Proceedings of the 26th International Symposium on Mathematical Foundations of Computer Science, London, UK, Springer-Verlag (2001) 37–57
5. Johnson, T., Robertson, N., Seymour, P.D., Thomas, R.: Directed tree-width. *J. Combin. Theory Ser. B* **82**(1) (2001) 138–154
6. Safari, M.A.: d-width: a more natural measure for directed tree width. In: Mathematical foundations of computer science 2005. Volume 3618 of Lecture Notes in Comput. Sci. Springer, Berlin (2005) 745–756
7. Berwanger, D., Dawar, A., Hunter, P., Kreutzer, S.: Dag-width and parity games. In Durand, B., Thomas, W., eds.: STACS. Volume 3884 of Lecture Notes in Computer Science., Springer (2006) 524–536
8. Berwanger, D., Grädel, E., Lenzi, G.: On the variable hierarchy of the modal mu-calculus. In Bradfield, J., ed.: Computer Science Logic, CSL 2002. Volume 2471 of LNCS., Springer-Verlag (2002) 352–366
9. Berwanger, D., Lenzi, G.: The variable hierarchy of the μ -calculus is strict. In: STACS 2005, Proceedings of the 22nd Symposium on Theoretical Aspects of Computer Science. Volume 3404 of LNCS., Springer-Verlag (2005) 97–109
10. Kozen, D.: Results on the propositional μ -calculus. *Theoret. Comput. Sci.* **27**(3) (1983) 333–354
11. Arnold, A., Niwiński, D.: Rudiments of μ -calculus. Volume 146 of Studies in Logic and the Foundations of Mathematics. North-Holland Publishing Co., Amsterdam (2001)
12. Bloom, S.L., Ésik, Z.: Iteration theories. Springer-Verlag, Berlin (1993)
13. Zielonka, W.: Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theoret. Comput. Sci.* **200**(1-2) (1998) 135–183
14. Jurdzinski, M.: Small progress measures for solving parity games. In Reichel, H., Tison, S., eds.: STACS. Volume 1770 of Lecture Notes in Computer Science., Springer (2000) 290–301
15. Courcelle, B.: Graph rewriting: an algebraic and logic approach. In: Handbook of theoretical computer science, Vol. B. Elsevier, Amsterdam (1990) 193–242
16. Goubault, E., Raufen, M.: Dihomotopy as a tool in state space analysis. In Rajsbaum, S., ed.: LATIN. Volume 2286 of Lecture Notes in Computer Science., Springer (2002) 16–37
17. Birmele, E.: Tree-width and circumference of graphs. *J. Graph Theory* **43**(1) (2003) 24–25
18. Cormen, T.H., Leiserson, C.E., Rivest, R.L.: Introduction to Algorithms. first edn. The MIT Electrical Engineering and Computer Science Series. The MIT Press (1990)