



HAL
open science

A Superposition-Based Calculus for Diagrammatic Reasoning

Nicolas Peltier, Mnacho Echenim, Rachid Echahed, Mehdi Mhalla

► **To cite this version:**

Nicolas Peltier, Mnacho Echenim, Rachid Echahed, Mehdi Mhalla. A Superposition-Based Calculus for Diagrammatic Reasoning. PPDP 2021: 23rd International Symposium on Principles and Practice of Declarative Programming, 2021, Tallinn, Estonia. <10.1145/3479394.3479405>. <hal-03372948>

HAL Id: hal-03372948

<https://hal.science/hal-03372948v1>

Submitted on 11 Oct 2021

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.



HAL Authorization

A Superposition-Based Calculus for Diagrammatic Reasoning (Long Version)

Rachid Echahed
Mnacho Echenim
Mehdi Mhalla
Nicolas Peltier

Univ. Grenoble Alpes, LIG, CNRS/GINP
F-38000, Grenoble, France

ABSTRACT

We introduce a class of rooted graphs which are expressive enough to encode various kinds of classical or quantum circuits. We then follow a set-theoretic approach to define rewrite systems over the considered graphs. Afterwards, we tackle the problem of equational reasoning with the graphs under study and we propose a new Superposition calculus to check the unsatisfiability of formulas consisting of equations or disequations over these graphs. We establish the soundness and refutational completeness of the calculus.

CCS CONCEPTS

• **Theory of computation** → **Automated reasoning; Equational logic and rewriting.**

KEYWORDS

Equational Reasoning, Graph, Superposition Calculus

ACM Reference Format:

Rachid Echahed, Mnacho Echenim, Mehdi Mhalla, and Nicolas Peltier. 2021. A Superposition-Based Calculus for Diagrammatic Reasoning (Long Version). In *23rd International Symposium on Principles and Practice of Declarative Programming (PPDP 2021), September 6–8, 2021, Tallinn, Estonia*. ACM, New York, NY, USA, 20 pages. <https://doi.org/10.1145/3479394.3479405>

1 INTRODUCTION

Rewrite systems play an important rôle in defining operational semantics of declarative programming languages such as functional programming [3, 26, 32] or functional-logic programming [16, 24]. In general, the basic objects of such declarative languages are trees, e.g., lambda terms or first-order terms. In this paper, we consider rewrite systems operating on a class of rooted graphs. Such systems are tailored to model applications based on circuits building and simulation. The road to computing by means of graph rewrite systems started in the seventies [19] using the well-known algebraic double pushout approach. Since then, several approaches of graph rewriting have been proposed. One can distinguish two main

streams of research : the algebraic approaches using constructs borrowed from category theory [14, 15, 18, 29] on one hand and the set-theoretic or algorithmic approaches [4, 17, 20] based on actual algorithms/operations involved in graph transformations on the other hand. All the proposed approaches define dedicated solutions to the delicate action of replacing a given subgraph by another in the rewriting process. Subgraph replacement is more difficult to achieve in general if compared to subterm (subtree) replacement. This difficulty is mainly due to the way one handles the possible edges connecting nodes of the subgraph to be replaced with the rest of the considered graph. In this work, we propose a set-theoretic approach to define a class of rewrite systems over rooted graphs. We introduce the basic operations over graphs needed to work out the machinery used to define rewriting steps. Due to the shape of the graphs we consider, our definition of graph rewriting is rather close to term rewriting and thus avoids the difficulties behind subgraph replacement actions.

Rule-based programming offers the possibility to write programs in a declarative way by stating rules or inferences behind the intended algorithm without caring for the control in a first step. Another benefit of rule-based programming is the underlying mathematical foundation which allows one to make some formal reasoning on programs. For example, in the case of term rewrite systems, equational reasoning has been developed intensively and now is routine in most theorem provers. Unfortunately, reasoning on graph structures is still in its infancy despite the progress made in using Hoare-like calculi [9, 22, 34] or model-checkers [35, 39]. In this paper, we also tackle the problem of reasoning with graphs and propose an extension to graphs of the well-known Superposition calculus.

The Superposition calculus [1] is the most successful automated proof procedure for equational reasoning. The calculus may be viewed as an extension of the Knuth-Bendix algorithm [28] to a wider class of equational formulæ, involving disjunctions (e.g., conditional rules) and quantifiers. It also generalizes the narrowing procedure [25] by handling equations that cannot be oriented and may be viewed as an extension of the Resolution calculus [31, 37], which is the most efficient proof procedure for predicate logic, to sets of clauses containing the equality predicate. The calculus is presented as a set of inference rules, deducing new assertions from axioms or previously generated assertions, together with a generic redundancy criterion that permits to prune the search space and discard many inferences. This calculus is *sound*, in the sense that every derived formula is a logical consequence of the premises, and

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

PPDP 2021, September 6–8, 2021, Tallinn, Estonia

© 2021 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-8689-0/21/09...\$15.00

<https://doi.org/10.1145/3479394.3479405>

refutationally complete: if a set of axioms E is contradictory, then a contradiction (i.e., an empty clause, equivalent to *false*) can always be generated by the calculus. It is also generic, and can be uniformly applied to any set of axioms that can be expressed in first-order logic. Most efficient theorem provers, such as Vampire [36], E [38] or Spass [42], are based on the Superposition calculus. Numerous extensions of the Superposition calculus have been proposed to handle specific theories or extensions of first-order logic, for instance higher-order logic [6], inductive and co-inductive datatypes [7], cancellative monoids [41], or general-purpose theory-reasoning [5]. In the present paper, we extend the calculus to perform equational reasoning on *graphs* instead of terms and we establish the soundness and refutational completeness of the obtained proof procedure.

Graphs are ubiquitous in programming and verification. They are often used for example to model pointer-based data-structures, and equations between graphs may be viewed as specifications of functions operating on such data-structures. In this context, proving general properties of these functions involves performing equational reasoning on graphs. In quantum computing, graphs have become particularly useful for the formal verification of quantum algorithms and protocols, and several such languages have been developed over the past years for such an analysis, including the ZW calculus [23], the ZH calculus [2], the ZX calculus [13], the SZX [10] calculus, or more recently the PBS-calculus [11]. The reason these languages are so useful is that standard approaches to verifying the correctness of quantum algorithms and protocols involve computations on complex matrices, and such approaches are non-intuitive and error-prone. On the other hand, graph-based languages permit to represent the same algorithms and protocols in a more intuitive way, by abstracting the numerical values and matrices, and replacing computations on these matrices by rewrite rules. Using graph-based languages to represent quantum algorithms thus permits to reduce many verification tasks to testing the equivalence of two graphs modulo a set of equations, which represent properties of the considered computations.

Extending the Superposition calculus to reason on graphs is thus a very desirable feature, with promising applications in programming. A first, rather straightforward, approach would be to encode graphs into first-order terms, stating all the necessary properties as axioms. However, such an approach would be very inefficient and the produced proofs would be extremely long and hardly readable, since they will be cluttered by the many reasoning steps necessary to handle trivial transformations related to associativity, commutativity, isomorphism etc. A second, more promising, approach is to tune the Superposition calculus so that it applies directly on graphs, which would replace terms as basic objects, with built-in algorithms to cope with basic graph properties. This approach is appealing but it raises three difficult issues. First it has been known for a long time that well-established properties of term rewriting fail when graphs are considered. In particular, orthogonal ground rewrite systems operating on graphs are not confluent in general [27]: for instance the system $\{f(x) \rightarrow x, g(x) \rightarrow x\}$ is not confluent if cyclic graphs are considered. In fact, it is known that confluence of terminating set of rules operating on graphs is undecidable [33]. Unfortunately, the confluence of ground rewrite systems plays a key rôle for establishing the completeness of the Superposition calculus,

because, informally, term rewrite systems are used to construct models of saturated clause sets (i.e. of the clause sets on which no non-redundant inference apply and containing no contradiction), to prove that such sets are indeed satisfiable. A second issue, related to the first one, is that equality between graphs is not preserved by context embedding, for instance two isomorphic graphs may become non-isomorphic when additional vertices and edges are added. Finally, the most difficult problem is that the Superposition calculus relies on the existence of reduction orders, i.e., of well-founded orders which are total on ground terms (i.e., on terms containing no variables) and closed under context embedding and under replacement. Such orders are used to orient ground equations into rewrite rules in order to construct suitable representations of models for saturated sets containing no contradiction. However, such reduction orders are very difficult (if not impossible) to define for graphs, because the order must be closed under isomorphism.

In the present paper we overcome these issues by considering a specific class of graphs, called DRL-graphs, which possess properties similar to those used in term rewriting. We provide a Superposition-based calculus to check the satisfiability of sets of equations and disequations between the considered class of graphs and we prove that the calculus is sound and refutationally complete. To tackle the problem mentioned above related to the existence of reduction orders, the completeness proof relies on non-well-founded orders and non-terminating rules, which comes at the cost of a more restrictive redundancy criterion. It should be noted that using non-terminating rules makes the confluence proof more difficult, since establishing local confluence is not sufficient.

Graphs may contain variables, denoting terms labeling the nodes and the calculus may be used to synthesize terms satisfying some graph properties, by propagating unifiers. In Section 8, we shall see that this class is general enough to denote circuits occurring in quantum programming. We wish to emphasize that the considered class is strictly more expressive than that of terms, for instance the satisfiability problem is undecidable even if the considered formula contains no variable (see Theorem 45).

Related Work

The very recent research report [21] is very close to our work. It pursues similar aims, namely to construct automatically equational derivations on diagrammatic structures, more precisely on ZX diagrams [12]. The considered approach, which is practically very successful, is different from ours: it relies on the use of the standard Superposition calculus [1] together with an encoding of ZX diagrams and rules by terms and equations between them (including an embedding of higher-order terms to cope with schematic rules). In our work, we bypass this encoding by defining a Superposition calculus that applies directly on graphs. The advantage is that we do not have to model graph properties by first-order axioms (such as the associativity and commutativity of graph constructors, see, e.g., [21, Equations 303-304, p. 80]), instead these properties are taken into account in the definition of the inference rules. We do not use schematic rules to cope with variadic gates, instead instances of these rules with fixed arities may be generated on demand during the proof search.

Numerous works in graph rewriting use an algebraic framework based on categories theory, using mostly the double pushout approach introduced in the seminal paper [19] (which has since been generalized in many directions). Here, we prefer to use an algorithmic, set theoretical framework. This low-level approach, although less general, is more suited to deal with the intricacies in the definition of the inference rules and in the completeness proof. The considered structures are also closer to those considered usually in superposition-based theorem proving, which is more convenient for extending the calculus to graphs.

In [8], it is proven that confluence is decidable for terminating rewrite systems on graphs with an interface, which embed those considered in the present paper (as mentioned above, confluence is undecidable in general in graph rewriting). In our results, the rewrite rules we consider are not necessarily terminating (and equations are not even orientable in general). In Section 7, confluence is used as a tool to construct models of saturated sets of clauses. It is easy to check that a terminating set of rules with no non joinable critical pair is always saturated (in the sense of Definition 53 and w.r.t. some adequate order), thus the decidability of confluence could be re-established for our particular class of graphs using Lemma 56 in the present paper.

We wish to emphasize that the notion of “completeness” considered in the present paper refers to the completeness of the inference rules, i.e., to their ability to detect that graphs are reducible to each other. It should not be confused with work asserting the completeness of the diagrammatic axiomatization w.r.t. a particular semantics and application domain (such as [40]).

The rest of the paper is organized as follows. Some basic definitions are recalled in the next section. The class of rooted graphs is introduced in Section 3 together with some basic operations on these graphs as well as the considered class of graph rewrite systems. Sections 4 and 5 define the graph formulas and their semantics. In Section 6, the new Superposition calculus is presented and its completeness is established in Section 7. Section 8 provides examples of the considered graphs and concluding remarks are made in Section 9.

2 BASIC DEFINITIONS AND NOTATIONS

For any partial function f , we denote by $dom(f)$ its domain, i.e., the set of elements x such that $f(x)$ is defined and by $img(f)$ the set $\{f(x) \mid x \in dom(f)\}$. For any function f and for any $D \subseteq dom(f)$, we denote by $f|_D$ the restriction of f to D . The integer interval $\{a, a + 1, \dots, b\}$ is denoted by $\llbracket a, b \rrbracket$ (it is empty if $a > b$). The function of domain $\{t_1, \dots, t_n\}$ mapping t_i to s_i for all $i \in \llbracket 1, n \rrbracket$ is denoted by $\{t_i \mapsto s_i \mid i \in \llbracket 1, n \rrbracket\}$.

Any partial function f operating on some set S may be extended into a function operating on tuples or sets of elements in S , using the relations: $f(\langle t_1, \dots, t_n \rangle) \stackrel{\text{def}}{=} \langle f(t_1), \dots, f(t_n) \rangle$ and $f(\{t_1, \dots, t_n\}) \stackrel{\text{def}}{=} \{f(t_1), \dots, f(t_n)\}$. If $t_i \notin dom(f)$ for some $i = 1, \dots, n$ then $f(\langle t_1, \dots, t_n \rangle)$ and $f(\{t_1, \dots, t_n\})$ are undefined. These relations may be applied recursively, e.g., if t_1, \dots, t_n are themselves sets or tuples of elements.

The notation $g \circ f$ denotes as usual the composition of g and f , with $dom(g \circ f) = \{x \in dom(f) \mid f(x) \in dom(g)\}$ and $(g \circ$

$f)(x) = g(f(x))$ for every $x \in dom(g \circ f)$. According to the previous convention, if functions are viewed as sets of pairs, then for every function f and for every injective function g such that $img(f) \cup dom(f) \subseteq dom(g)$, $g(f)$ is a function with $dom(g(f)) = g(dom(f))$, $img(g(f)) = g(img(f))$, and $(g(f))(g(x)) = g(f(x))$, for any $x \in dom(f)$. Note that, by definition, $g(f) = g \circ f \circ g^{-1}$.

With a slight abuse of notations, we sometimes use set notations on tuples when the order and duplication of components are irrelevant, i.e., we write $u \in \langle t_1, \dots, t_n \rangle$ to state that $u \in \{t_1, \dots, t_n\}$, $\langle t_1, \dots, t_n \rangle \cup E$ to denote the set $\{t_1, \dots, t_n\} \cup E$ or $\langle t_1, \dots, t_n \rangle = \langle t'_1, \dots, t'_m \rangle \cup \langle t''_1, \dots, t''_k \rangle$ for $\{t_1, \dots, t_n\} = \{t'_1, \dots, t'_m, t''_1, \dots, t''_k\}$.

DEFINITION 1. Let Σ be a set of function symbols and let \mathcal{V} be a countable set of variables. Each symbol $f \in \Sigma$ is associated with a unique arity $ar(f)$. The set of terms \mathcal{T} is the least set such that $\mathcal{V} \subseteq \mathcal{T}$ and $t_1, \dots, t_n \in \mathcal{T} \Rightarrow f(t_1, \dots, t_n) \in \mathcal{T}$, for every $f \in \Sigma$ with $n = ar(f)$. We denote by $\mathcal{V}(t)$ the set of variables occurring in t . A term t is ground if $\mathcal{V}(t) = \emptyset$.

A substitution σ is a total mapping from \mathcal{V} to \mathcal{T} . As usual, for any term t , $\sigma(t)$ denotes the term obtained from t by replacing every variable x by $\sigma(x)$. A substitution σ is ground if $\sigma(x)$ is ground for every $x \in dom(\sigma)$. A unifier of a set of pairs E is a substitution σ such that $\sigma(t) = \sigma(s)$ holds for all pairs $(t, s) \in E$. E is unifiable if it admits a unifier. A substitution σ is more general than a substitution θ iff there exists a substitution σ' such that $\theta = \sigma' \circ \sigma$. It is well known that every unifiable set of pairs admits a most general unifier (mgu).

3 ROOTED GRAPHS

3.1 Definitions

We define the class of graphs on which the adaptation of the Superposition calculus will be applied. Intuitively, this class consists of labeled graphs with a distinguished sequence of nodes which can be viewed as an interface that constrains the operations of replacing a subgraph by another graph that can be performed. More precisely, the replacement is possible only if all the edges between the original graph and the replaced subgraph have an incident node inside the interface.

Let \mathcal{N} be a fixed countable set of nodes, disjoint from the set of terms, and let S be a set of sorts. We consider a function $sort$ mapping every node $\alpha \in \mathcal{N}$ to a sort in S and a pre-order \preceq on \mathcal{N} . We write $\alpha \triangleq \alpha'$ if $\alpha \preceq \alpha'$ and $\alpha' \preceq \alpha$, and $\alpha \triangleleft \alpha'$ for $\alpha \preceq \alpha'$ and $\alpha' \not\triangleq \alpha$.

DEFINITION 2. A DRL-graph (for Directed Rooted Labeled graph) G is a tuple $\langle \mathcal{N}_G, \mathcal{R}_G, \mathcal{E}_G, l_G \rangle$ where:

- \mathcal{N}_G is a finite subset of \mathcal{N} .
- \mathcal{R}_G is a finite sequence of nodes in \mathcal{N}_G with no repetition, called the roots of G . We denote by \mathcal{N}_G^* the set of nodes $\mathcal{N}_G \setminus \mathcal{R}_G$.
- \mathcal{E}_G is a finite set of ordered pairs (written $(\alpha \rightarrow \beta)$) of nodes in \mathcal{N}_G , called edges. We write $(\alpha \rightleftarrows \beta)$ to denote any edge $(\alpha \rightarrow \beta)$ or $(\beta \rightarrow \alpha)$.
- l_G is a function mapping each node $\alpha \in \mathcal{N}_G^*$ to a term (called the label of α).

Two DRL-graphs G and H are disjoint if $N_G \cap N_H = \emptyset$. We denote by $\mathcal{V}(G) \stackrel{\text{def}}{=} \bigcup_{\alpha \in N_G^*} \mathcal{V}(l_G(\alpha))$ the set of variables occurring in the labels of nodes in G . A DRL-graph G is ground if $\mathcal{V}(G) = \emptyset$.

Note that the function l_G is not defined over the nodes in \mathcal{R}_G .

DEFINITION 3. For any substitution σ and DRL-graph G , $\sigma(G)$ denotes the DRL-graph $\langle N_G, \mathcal{R}_G, \mathcal{E}_G, l' \rangle$, where $l'(\alpha) = \sigma(l_G(\alpha))$, for all $\alpha \in N_G^*$.

DEFINITION 4. A DRL-graph H is a DRL-subgraph of G (written $H \leq G$) if the following conditions hold:

- (1) $N_H \subseteq N_G$;
- (2) If $(\alpha \rightarrow \beta) \in \mathcal{E}_H$ then $(\alpha \rightarrow \beta) \in \mathcal{E}_G$.
- (3) If $(\alpha \rightarrow \beta) \in \mathcal{E}_G$ and $\alpha, \beta \in N_H$ then $(\alpha \rightarrow \beta) \in \mathcal{E}_H$.
- (4) If $(\alpha \rightleftharpoons \beta) \in \mathcal{E}_G$, $\beta \notin N_H$ and $\alpha \in N_H$, then α occurs in \mathcal{R}_H .
- (5) If α occurs in \mathcal{R}_G and $\alpha \in N_H$, then α occurs in \mathcal{R}_H .
- (6) l_H is the restriction of l_G to the nodes in N_H^* .

PROPOSITION 5. The relation \leq is transitive and reflexive.

PROOF. It is immediate to check that \leq is reflexive. Assume that $H \leq I \leq G$. Then:

- (1) $N_I \subseteq N_H \subseteq N_G$ hence $N_I \subseteq N_G$.
- (2) We have $(\alpha \rightarrow \beta) \in \mathcal{E}_I \Rightarrow (\alpha \rightarrow \beta) \in \mathcal{E}_H \Rightarrow (\alpha \rightarrow \beta) \in \mathcal{E}_G$.
- (3) If $(\alpha \rightarrow \beta) \in \mathcal{E}_G$ and $\alpha, \beta \in N_I^*$, then by Condition 5 of Definition 4 $\alpha, \beta \in N_H^*$, thus $(\alpha \rightarrow \beta) \in \mathcal{E}_H$ and $(\alpha \rightarrow \beta) \in \mathcal{E}_I$.
- (4) Assume that $(\alpha \rightleftharpoons \beta) \in \mathcal{E}_G$, $\alpha \in N_I$ and $\beta \notin N_I$. If $\beta \in N_H$, then $(\alpha \rightleftharpoons \beta) \in \mathcal{E}_H$ and we deduce that α must occur in \mathcal{R}_I because $I \leq H$. If $\beta \notin N_H$ then α occurs in \mathcal{R}_H since $H \leq G$. Thus $\alpha \in \mathcal{R}_I$, since $I \leq H$.
- (5) $l_I = l_H|_{N_I^*} = (l_G|_{N_H^*})|_{N_I^*} = l_G|_{N_I^*}$ (because $N_I^* \subseteq N_H^*$).

□

PROPOSITION 6. If $H \leq G$ then $\mathcal{R}_G \subseteq N_G \setminus N_H^*$.

PROOF. Assume for a contradiction that $\alpha \in \mathcal{R}_G \cap N_H^*$. Then by definition, α is a node in N_H that does not occur in \mathcal{R}_H , but this is impossible by Condition 5 of Definition 4. □

In what follows, we define an equivalence relation on graphs to formalize the fact that the properties we prove do not depend on the actual nodes that occur in the graph.

DEFINITION 7. An N -mapping μ is a partial injective mapping from nodes to nodes, such that $\text{sort}(\alpha) = \text{sort}(\mu(\alpha))$ and $\alpha \triangleq \mu(\alpha)$. For technical convenience we assume that $\mu(t) = t$ holds for every term t , i.e., every N -mapping is extended to the identity on terms.

PROPOSITION 8. If G is a DRL-graph, then for every N -mapping μ with $\text{dom}(\mu) \supseteq N_G$, $\mu(G)$ is a DRL-graph. Moreover, $N_{\mu(G)} = \mu(N_G)$, $\mathcal{E}_{\mu(G)} = \mu(\mathcal{E}_G)$, $\mathcal{R}_{\mu(G)} = \mu(\mathcal{R}_G)$ and $l_{\mu(G)} = \mu \circ l_G = \mu \circ l_G \circ \mu^{-1} = l_G \circ \mu^{-1}$.

PROOF. The result follows immediately from the definition of $\mu(G)$ (see Section 2) and from the fact that μ is injective. □

Similarly, because N -mappings are injective, we have the following result:

PROPOSITION 9. If $H \leq G$ and $N_G \subseteq \text{dom}(\mu)$ for an N -mapping μ , then $\mu(H) \leq \mu(G)$.

DEFINITION 10. We write $G \sim G'$ if there exists μ such that $\text{dom}(\mu) \supseteq N_{G'}$ and $G = \mu(G')$.

PROPOSITION 11. The relation \sim is an equivalence relation.

PROOF.

- **Reflexivity.** The identity on N_G is an N -mapping, thus $G \sim G$.
- **Symmetry.** If $G \sim G'$, then $G = \mu(G')$, where μ is an N -mapping such that $\text{dom}(\mu) \supseteq N_{G'}$. Since μ is injective, μ^{-1} is well-defined, $\mu^{-1}(G) = (\mu^{-1} \circ \mu)(G') = G'$ and $\text{dom}(\mu^{-1}) = \text{img}(\mu) \supseteq \mu(N_{G'}) = N_G$. Hence $G' \sim G$.
- **Transitivity.** If $G \sim G' \sim G''$ then there exist N -mappings μ and μ' such that $G = \mu(G')$, $G' = \mu'(G'')$, $\text{dom}(\mu) \supseteq N_{G'}$ and $\text{dom}(\mu') \supseteq N_{G''}$. Then since the composition of two injective functions is also injective, $\mu \circ \mu'$ is an N -mapping, and $G = (\mu \circ \mu')(G'')$ with $\text{dom}(\mu \circ \mu') \supseteq N_{G''}$. Hence $G \sim G''$. □

We now define a replacement operation on the considered class of graphs. Intuitively, this replacement operation applied to a graph permits to replace one of its subgraphs by another graph, provided a so-called *substitutability* condition is satisfied. This condition will guarantee that the replacement operation on graphs enjoys properties similar to those of standard term rewriting.

DEFINITION 12. Let G, H and H' be DRL-graphs such that $H \leq G$, $\mathcal{R}_H = \langle \alpha_1, \dots, \alpha_n \rangle$ and $\mathcal{R}_{H'} = \langle \alpha'_1, \dots, \alpha'_m \rangle$. The DRL-graphs H and H' are root-compatible if $n = m$ and for every $i \in \llbracket 1, n \rrbracket$, we have $\text{sort}(\alpha_i) = \text{sort}(\alpha'_i)$ and $\alpha_i \triangleq \alpha'_i$.

DEFINITION 13. A DRL-graph H' is substitutable for H in G if H and H' are root-compatible and $N_G \cap N_{H'} \subseteq N_H$.

An immediate consequence of this definition is the following property:

PROPOSITION 14. If H' is substitutable for H in G then $(N_G \setminus N_H) \cap N_{H'} = \emptyset$.

DEFINITION 15. Let G, H and H' be DRL-graphs such that DRL-graph H' is substitutable for H in G , $H \leq G$, $\mathcal{R}_H = \langle \alpha_1, \dots, \alpha_n \rangle$ and $\mathcal{R}_{H'} = \langle \alpha'_1, \dots, \alpha'_n \rangle$. The mapping from H to H' in G , is the function $\mu_G^{H \rightarrow H'} \stackrel{\text{def}}{=} \{\alpha_i \mapsto \alpha'_i \mid 1 \leq i \leq n\} \cup \{\alpha \mapsto \alpha \mid \alpha \in N_G \setminus N_H\}$.

Let $\mu \stackrel{\text{def}}{=} \mu_G^{H \rightarrow H'}$. We denote by $G[H'/H]$ the DRL-graph G' defined as follows:

- (1) $N_{G'} \stackrel{\text{def}}{=} (N_G \setminus N_H) \cup N_{H'}$.
- (2) $\mathcal{R}_{G'} \stackrel{\text{def}}{=} \mu(\mathcal{R}_G)$.
- (3) $\mathcal{E}_{G'} \stackrel{\text{def}}{=} \mu(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{H'}$.
- (4) For every $\alpha \in N_{G'}$:

$$l_{G'}(\alpha) \stackrel{\text{def}}{=} \begin{cases} l_G(\alpha) & \text{if } \alpha \in N_{G'}^* \setminus N_{H'} \\ l_{H'}(\alpha) & \text{if } \alpha \in N_{H'}^* \\ l_G(\mu^{-1}(\alpha)) & \text{if } \mu^{-1}(\alpha) \in N_G^* \cap \mathcal{R}_H \\ \text{undefined} & \text{otherwise} \end{cases}$$

EXAMPLE 16. Consider the following DRL-graphs (see also Figure 1):



Figure 1: Graphs G , H (dashed lines) and H' in Example 16 (the labels are omitted for readability)

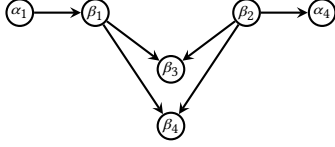


Figure 2: The graph $G[H'/H]$ in Example 16

- G is defined as follows:

$$\begin{aligned} \mathcal{N}_G &\stackrel{\text{def}}{=} \{\alpha_1, \alpha_2, \alpha_3, \alpha_4\} \\ \mathcal{R}_G &\stackrel{\text{def}}{=} \langle \rangle \\ \mathcal{E}_G &\stackrel{\text{def}}{=} \{(\alpha_1 \rightarrow \alpha_2), (\alpha_2 \rightarrow \alpha_3), (\alpha_3 \rightarrow \alpha_4)\} \\ l_G &\stackrel{\text{def}}{=} \{(\alpha_1, a), (\alpha_2, f(x)), (\alpha_3, f(b)), (\alpha_4, b)\} \end{aligned}$$

- H is defined as follows:

$$\begin{aligned} \mathcal{N}_H &\stackrel{\text{def}}{=} \{\alpha_2, \alpha_3\} \\ \mathcal{R}_H &\stackrel{\text{def}}{=} \langle \alpha_2, \alpha_3 \rangle \\ \mathcal{E}_H &\stackrel{\text{def}}{=} \{(\alpha_2 \rightarrow \alpha_3)\} \\ l_H &\stackrel{\text{def}}{=} \emptyset \end{aligned}$$

- H' is defined as follows:

$$\begin{aligned} \mathcal{N}_{H'} &\stackrel{\text{def}}{=} \{\beta_1, \beta_2, \beta_3, \beta_4\} \\ \mathcal{R}_{H'} &\stackrel{\text{def}}{=} \langle \beta_1, \beta_2 \rangle \\ \mathcal{E}_{H'} &\stackrel{\text{def}}{=} \{(\beta_1 \rightarrow \beta_3), (\beta_1 \rightarrow \beta_4), (\beta_2 \rightarrow \beta_3), (\beta_2 \rightarrow \beta_4)\} \\ l_{H'} &\stackrel{\text{def}}{=} \{(\beta_3, c), (\beta_4, d)\} \end{aligned}$$

We assume that all nodes are of the same sort and that all nodes are distinct. Then H is a subgraph of G , and H' is substitutable for H in G . The DRL-graph $G[H'/H]$ is represented in Figure 2, where $l_{G[H'/H]} = \{(\alpha_1, a), (\beta_1, f(x)), (\beta_2, f(b)), (\beta_3, c), (\beta_4, d), (\alpha_4, b)\}$.

PROPOSITION 17. *If H' is substitutable for H in G , then by letting $\mu \stackrel{\text{def}}{=} \mu_G^{H \rightarrow H'}$ and $G' \stackrel{\text{def}}{=} G[H'/H]$, we have $\mu(\mathcal{R}_G) = \mathcal{R}_{G'}$ and $\mu(\mathcal{R}_H) = \mathcal{R}_{H'}$.*

PROPOSITION 18. *If H' is substitutable for H in G , then by letting $G' \stackrel{\text{def}}{=} G[H'/H]$, we have:*

- $\mathcal{N}_{G'} \setminus \mathcal{N}_{H'} = \mathcal{N}_G \setminus \mathcal{N}_H$;
- $\mathcal{N}_{G'}^* \setminus \mathcal{N}_{H'} = \mathcal{N}_G^* \setminus \mathcal{N}_H$.

PROOF. We have $\mathcal{N}_{G'} \setminus \mathcal{N}_{H'} = [(\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_{H'}] \setminus \mathcal{N}_{H'} = (\mathcal{N}_G \setminus \mathcal{N}_H) \setminus \mathcal{N}_{H'} = \mathcal{N}_G \setminus \mathcal{N}_H$ by Proposition 14. Consider a node $\alpha \in \mathcal{N}_G$; we show that $\alpha \in \mathcal{R}_G \setminus \mathcal{N}_H$ if and only if $\alpha \in \mathcal{R}_{G'} \setminus \mathcal{N}_{H'}$;

together with the previous result this will prove the second item of the proposition. First assume that $\alpha \in \mathcal{R}_G \setminus \mathcal{N}_H$ and let μ denote the mapping from H to H' in G . Then by definition, we have $\mu(\alpha) = \alpha \in \mathcal{R}_{G'}$ and by Proposition 14, $\alpha \notin \mathcal{N}_{H'}$; hence the result. Conversely, if $\alpha \in \mathcal{R}_{G'} \setminus \mathcal{N}_{H'}$ then $\alpha \in \mathcal{N}_{G'} \setminus \mathcal{N}_{H'} = \mathcal{N}_G \setminus \mathcal{N}_H$ by the first item of the proposition, and by definition, $\alpha = \mu(\alpha')$ for some $\alpha' \in \mathcal{R}_G$. Since $\alpha \notin \mathcal{N}_H$, necessarily $\alpha' = \alpha$, hence the result. \square

Observe that, with the notations of Definition 15, if $\alpha \in \mathcal{N}_{G'}^* \setminus \mathcal{N}_{H'} = \mathcal{N}_G^* \setminus \mathcal{N}_H$, then necessarily $\mu(\alpha) = \alpha$, so that $\mu^{-1}(\alpha) = \alpha$ and $l_{G'}(\alpha) = l_G(\mu^{-1}(\alpha))$.

LEMMA 19. *Let G be a DRL-graph, let H be a DRL-subgraph of G and let H' be a DRL-graph substitutable for H in G . Then, $G[H'/H]$ is a DRL-graph and $H' \leq G[H'/H]$.*

PROOF. Observe that the function μ in Definition 15 is injective, since $\mathcal{R}_{H'}$ is repetition-free and no node in $\mathcal{N}_G \setminus \mathcal{N}_H$ may occur in $\mathcal{R}_{H'}$, because $\mathcal{N}_G \cap \mathcal{N}_{H'} \subseteq \mathcal{N}_H$ by hypothesis. We show that the domain of μ contains all nodes occurring either in \mathcal{R}_G or in an edge in $\mathcal{E}_G \setminus \mathcal{E}_H$. By Condition 5 of Definition 4, the nodes in \mathcal{R}_G occur in either \mathcal{R}_H or $\mathcal{N}_G \setminus \mathcal{N}_H$.

The function $l_{G'}$ is well-defined: if $\alpha \in \mathcal{N}_{H'}^*$ and $\alpha = \mu(\alpha')$ for $\alpha' \in \mathcal{N}_G^* \setminus \mathcal{N}_H$, then we have $\alpha' \notin \mathcal{N}_{H'}$ (because $\mathcal{N}_G \cap \mathcal{N}_{H'} \subseteq \mathcal{N}_G$), hence $\alpha' \neq \alpha$, and in this case, by definition of μ , α occurs in $\mathcal{R}_{H'}$, hence may not occur in $\mathcal{N}_{H'}^*$, a contradiction. Thus it is straightforward to check that $G[H'/H]$ is a DRL-graph.

We prove that $H' \leq G'$, with $G' \stackrel{\text{def}}{=} G[H'/H]$:

- (1) $\mathcal{N}_{H'} \subseteq \mathcal{N}_{G'} = (\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_{H'}$.
- (2) If $(\alpha \rightarrow \beta) \in \mathcal{E}_{H'}$ then $(\alpha \rightrightarrows \beta) \in \mathcal{E}_{G'} = \mu(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{H'}$.
- (3) If $\alpha, \beta \in \mathcal{N}_{H'}$ and $(\alpha \rightarrow \beta) \in \mathcal{E}_{G'}$, then by definition of $\mathcal{E}_{G'}$, either $(\alpha \rightarrow \beta) \in \mathcal{E}_{H'}$ and the proof is completed, or $(\alpha \rightarrow \beta) \in \mu(\mathcal{E}_G \setminus \mathcal{E}_H)$. In the latter case, we have $\alpha = \mu(\alpha')$, $\beta = \mu(\beta')$ with $(\alpha' \rightarrow \beta') \in \mathcal{E}_G \setminus \mathcal{E}_H$. We show that $\alpha' \in \mathcal{N}_H$. If $\alpha = \alpha'$, then we have $\alpha \in \mathcal{N}_{H'} \cap \mathcal{N}_G$, thus $\alpha = \alpha' \in \mathcal{N}_H$ (since H' is substitutable for H in G). Otherwise, $\alpha \neq \alpha'$, hence by definition of μ , α' occurs in \mathcal{R}_H , thus $\alpha' \in \mathcal{N}_H$. Similarly, $\beta' \in \mathcal{N}_H$, and since $H \leq G$ we deduce that $(\alpha' \rightarrow \beta') \in \mathcal{E}_H$, contradicting the fact that $(\alpha' \rightarrow \beta') \in \mathcal{E}_G \setminus \mathcal{E}_H$.
- (4) Assume that $(\alpha \rightrightarrows \beta) \in \mathcal{E}_{G'}$, $\alpha \in \mathcal{N}_{H'}$ and $\beta \notin \mathcal{N}_{H'}$. By definition of $\mathcal{E}_{G'}$ this entails that $(\alpha \rightrightarrows \beta) \in \mu(\mathcal{E}_G \setminus \mathcal{E}_H)$ and $\alpha = \mu(\alpha')$, $\beta = \mu(\beta')$, for some $\alpha', \beta' \in \mathcal{N}_G$. If $\alpha = \alpha'$ then $\alpha \in \mathcal{N}_{H'} \cup \mathcal{N}_G$, and since H' is substitutable for H in G , we have $\alpha \in \mathcal{N}_H$, thus, by definition of μ , α occurs in \mathcal{R}_H (since $\alpha \in \text{dom}(\mu)$). Otherwise, $\alpha \neq \alpha'$, and α occurs in \mathcal{R}_H by definition of μ .
- (5) Let $\alpha \in \mathcal{N}_{H'}$ be a node occurring in $\mathcal{R}_{G'}$. We have $\alpha = \mu(\alpha')$ for some node occurring in \mathcal{R}_G . By definition of μ , either α occurs in $\mathcal{R}_{H'}$ and the proof is completed, or $\alpha = \alpha'$ with $\alpha \notin \mathcal{N}_H$, which contradicts the hypothesis that H' is substitutable for H in G .
- (6) By definition of $l_{G'}$, we have $l_{H'} = l_{G'}|_{\mathcal{N}_{H'}^*}$.

\square

3.2 Properties of the Replacement Operation

We establish some basic properties of the replacement operation, all of which are similar to their counterparts for the replacement operation on first-order terms.

PROPOSITION 20. *Let G be a DRL-graph and let $H \leq G$. Then H is substitutable for H in G and $G[H/H] = G$.*

PROOF. By definition, $\mathcal{N}_G \cap \mathcal{N}_H \subseteq \mathcal{N}_H$ thus H is substitutable for H in G . The conditions of Definition 15 (with $H = H'$) entail that μ is the identity on $\mathcal{N}_G \setminus \mathcal{N}_H^*$ and that $\mathcal{N}_{G'} = (\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_H = \mathcal{N}_G$, $\mathcal{R}_{G'} = \mathcal{R}_G$, $\mathcal{E}_{G'} = (\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_H = \mathcal{E}_G$ and $l_{G'} = l_G$ (since $l_H = l_G|_{\mathcal{N}_H^*}$). \square

PROPOSITION 21. *Let G, H and H' be DRL-graphs, with $H \leq G$ and H' is substitutable for H in G . Let μ be an \mathcal{N} -mapping with domain $\mathcal{N}_G \cup \mathcal{N}_{H'}$. Then $\mu(H')$ is substitutable for $\mu(H)$ in $\mu(G)$, and:*

$$\mu(G[H'/H]) = \mu(G)[\mu(H')/\mu(H)]$$

PROOF. We have $\mathcal{N}_{\mu(G)} \cap \mathcal{N}_{\mu(H')} = \mu(\mathcal{N}_G) \cap \mu(\mathcal{N}_{H'}) = \mu(\mathcal{N}_G \cap \mathcal{N}_{H'})$ by injectivity of μ , thus $\mathcal{N}_{\mu(G)} \cap \mathcal{N}_{\mu(H')} \subseteq \mu(\mathcal{N}_H) = \mathcal{N}_{\mu(H)}$ (since $\mathcal{N}_G \cap \mathcal{N}_{H'} \subseteq \mathcal{N}_H$, as H' is substitutable for H in G). Consequently, $\mu(H')$ is substitutable for $\mu(H)$ in $\mu(G)$.

Let $G' \stackrel{\text{def}}{=} G[H'/H]$ and $G'' \stackrel{\text{def}}{=} \mu(G)[\mu(H')/\mu(H)]$. We show that $\mu(G') = G''$. Let $\mathcal{R}_H = \langle \alpha_1, \dots, \alpha_n \rangle$, $\mathcal{R}_{H'} = \langle \alpha'_1, \dots, \alpha'_n \rangle$, and:

$$\mu' = \{\alpha_i \mapsto \alpha'_i \mid 1 \leq i \leq n\} \cup \{\alpha \mapsto \alpha \mid \alpha \in \mathcal{N}_G \setminus \mathcal{N}_H\}$$

$$\mu'' = \{\mu(\alpha_i) \mapsto \mu(\alpha'_i) \mid 1 \leq i \leq n\} \cup \{\alpha \mapsto \alpha \mid \alpha \in \mathcal{N}_{\mu(G)} \setminus \mathcal{N}_{\mu(H)}\}$$

Observe that $\mu'' = \mu \circ \mu' \circ \mu^{-1}$. We have:

•

$$\begin{aligned} \mathcal{N}_{\mu(G')} &= \mu(\mathcal{N}_{G'}) = \mu((\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_{H'}) \\ &= \mu(\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mu(\mathcal{N}_{H'}) \\ &= (\mu(\mathcal{N}_G) \setminus \mu(\mathcal{N}_H)) \cup \mu(\mathcal{N}_{H'}) \quad \text{by injectivity of } \mu \\ &= \mathcal{N}_{G''} \end{aligned}$$

•

$$\begin{aligned} \mathcal{E}_{\mu(G')} &= \mu(\mathcal{E}_{G'}) = \mu(\mu'(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{H'}) \\ &= \mu(\mu'(\mathcal{E}_G \setminus \mathcal{E}_H)) \cup \mu(\mathcal{E}_{H'}) = \mu(\mu'(\mathcal{E}_G \setminus \mathcal{E}_H)) \cup \mathcal{E}_{\mu(H')} \\ &= (\mu(\mu'(\mathcal{E}_G)) \setminus \mu(\mu'(\mathcal{E}_H))) \cup \mathcal{E}_{\mu(H')} \quad \text{by injectivity of } \mu, \mu' \\ &= (\mathcal{E}_{\mu(\mu'(G))}) \setminus \mathcal{E}_{\mu(\mu'(H))} \cup \mathcal{E}_{\mu(H')} \end{aligned}$$

Further,

$$\begin{aligned} \mathcal{E}_{G''} &= \mu''(\mathcal{E}_{\mu(G)} \setminus \mathcal{E}_{\mu(H)}) \cup \mathcal{E}_{\mu(H')} \\ &= (\mathcal{E}_{\mu''(\mu(G))} \setminus \mathcal{E}_{\mu''(\mu(H))}) \cup \mathcal{E}_{\mu(H')} \\ &= (\mathcal{E}_{\mu(\mu'(G))} \setminus \mathcal{E}_{\mu(\mu'(H))}) \cup \mathcal{E}_{\mu(H')} \end{aligned}$$

Hence $\mathcal{E}_{\mu(G')} = \mathcal{E}_{G''}$.

- $\mathcal{R}_{\mu(G')} = \mu(\mathcal{R}_{G'}) = \mu(\mu'(\mathcal{R}_G)) = \mu''(\mu(\mathcal{R}_G)) = \mu''(\mathcal{R}_{\mu(G)}) = \mathcal{R}_{G''}$.
- Assume that $\alpha = \mu''(\alpha')$, with $\alpha' \in \mathcal{N}_{\mu(G)}^* \setminus \mathcal{N}_{\mu(H)}^*$. Then $\alpha' = \mu(\alpha'')$, where $\alpha'' \in \mathcal{N}_G^* \setminus \mathcal{N}_H^*$, and $\mu(\mu'(\alpha'')) = \mu''(\alpha') = \alpha$. Furthermore, $l_{G'}(\alpha) = l_{\mu(G)}(\alpha') = l_G(\alpha'') = l_{G'}(\mu'(\alpha'')) = l_{\mu(G')}(\alpha)$, because $\mu^{-1}(\alpha) = \mu'(\alpha'')$. If $\alpha \in \mathcal{N}_{\mu(H')}^*$, then $\alpha = \mu(\alpha')$, with $\alpha' \in \mathcal{N}_H^*$. Moreover, we have $l_{G'}(\alpha) = l_{\mu(H')}(\alpha) = l_{H'}(\alpha') = l_{G'}(\alpha) = l_{\mu(G')}(\alpha)$. Finally, if $\alpha = \mu(\alpha')$ is not of one of the two forms above, then it is easy to check that $\alpha' \notin \mathcal{N}_{H'}^*$, and that there is no α'' such

that $\alpha' = \mu'(\alpha'')$, with $\alpha'' \in \mathcal{N}_G^* \setminus \mathcal{N}_H$. Thus $l_{G'}(\alpha)$ and $l_{G'}(\alpha')$ are both undefined. \square

A similar property holds for substitutions:

PROPOSITION 22. *Let G, H and H' be DRL-graphs, where $H \leq G$ and H' is substitutable for H in G . Let σ be a substitution with domain $\mathcal{V}(G) \cup \mathcal{V}(H')$. Then $\sigma(H')$ is substitutable for $\sigma(H)$ in $\sigma(G)$, and:*

$$\sigma(G[H'/H]) = \sigma(G)[\sigma(H')/\sigma(H)]$$

PROOF. Since a substitution only affects the labels of the DRL-graphs, the first property is immediate, and we only have to prove that, for every node $\alpha \in (\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_{H'}$, $l_{\sigma(G[H'/H])}(\alpha) = l_{\sigma(G)[\sigma(H')/\sigma(H)]}(\alpha)$. By definition, we have $l_{\sigma(G[H'/H])}(\alpha) = \sigma(l_{G[H'/H]}(\alpha))$. Let $\mu \stackrel{\text{def}}{=} \mu_G^{H \rightarrow H'}$, we distinguish several cases.

If $\alpha \in \mathcal{N}_{H'}^*$, then $l_{\sigma(G[H'/H])}(\alpha) = l_{H'}(\alpha)$. Moreover, we also have if $\alpha \in \mathcal{N}_{\sigma(H')}^*$, thus $l_{\sigma(G)[\sigma(H')/\sigma(H)]}(\alpha) = l_{\sigma(H')}(\alpha) = \sigma(l_{H'}(\alpha))$ and the proof is completed.

If $\alpha \in \mathcal{R}_{H'}$, then $l_{\sigma(G[H'/H])}(\alpha) = l_G(\mu^{-1}(\alpha))$. Further, we have $\alpha \in \mathcal{R}_{\sigma(H')}$, thus $l_{\sigma(G)[\sigma(H')/\sigma(H)]}(\alpha) = l_{\sigma(G)}(\mu^{-1}(\alpha)) = \sigma(l_{H'}(\mu^{-1}(\alpha)))$ and the proof is completed.

Finally, if $\alpha \notin \mathcal{N}_{H'}$, then by definition $l_{\sigma(G[H'/H])}(\alpha) = l_G(\alpha)$, and $\alpha \notin \mathcal{N}_{\sigma(H')}$, thus $l_{\sigma(G)[\sigma(H')/\sigma(H)]}(\alpha) = l_{\sigma(G)}(\alpha) = \sigma(l_{H'}(\alpha))$. \square

The next lemma shows that renaming the nodes of some sub-graph does not affect the graphs obtained by replacing this sub-graph.

LEMMA 23. *Let G, H and H' be DRL-graphs, with $H \leq G$ and H' is substitutable for H in G . Let μ be an \mathcal{N} -mapping of domain \mathcal{N}_G such that $\mu(\alpha) = \alpha$ if $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$. Then:*

$$\mu(G)[H'/\mu(H)] = G[H'/H]$$

PROOF. Let $G' \stackrel{\text{def}}{=} G[H'/H]$, $G'' \stackrel{\text{def}}{=} \mu(G)[H'/\mu(H)]$, $\mathcal{R}_H \stackrel{\text{def}}{=} \langle \alpha_1, \dots, \alpha_n \rangle$, $\mathcal{R}_{H'} \stackrel{\text{def}}{=} \langle \alpha'_1, \dots, \alpha'_n \rangle$ and consider the following \mathcal{N} -mappings:

$$\mu_1 \stackrel{\text{def}}{=} \{\alpha_i \mapsto \alpha'_i \mid 1 \leq i \leq n\} \cup \{\alpha \mapsto \alpha \mid \alpha \in \mathcal{N}_G \setminus \mathcal{N}_H\}$$

and

$$\mu_2 \stackrel{\text{def}}{=} \{\mu(\alpha_i) \mapsto \alpha'_i \mid 1 \leq i \leq n\} \cup \{\alpha \mapsto \alpha \mid \alpha \in \mathcal{N}_{\mu(G)} \setminus \mathcal{N}_{\mu(H)}\}.$$

By injectivity of μ , we have $\mathcal{N}_{\mu(G)} \setminus \mathcal{N}_{\mu(H)} = \mu(\mathcal{N}_G) \setminus \mu(\mathcal{N}_H) = \mu(\mathcal{N}_G \setminus \mathcal{N}_H) = \mathcal{N}_G \setminus \mathcal{N}_H$, since $\mu(\alpha) = \alpha$ if $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$. Thus $\mu_2 \circ \mu = \mu_1$. We show that $G' = G''$:

- By Definition 15 (1), $\mathcal{N}_{G'} = (\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_{H'}$ and $\mathcal{N}_{G''} = (\mathcal{N}_{\mu(G)} \setminus \mathcal{N}_{\mu(H)}) \cup \mathcal{N}_{H'} = (\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_{H'} = \mathcal{N}_{G'}$.
- By Definition 15 (2), $\mathcal{R}_{G'} = \mu_1(\mathcal{R}_G)$ and $\mathcal{R}_{G''} = \mu_2(\mathcal{R}_{\mu(G)}) = \mu_2(\mu(\mathcal{R}_G)) = \mu_1(\mathcal{R}_G) = \mathcal{R}_{G'}$.
- By Definition 15 (3), $\mathcal{E}_{G'} = \mu_1(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{H'}$ and $\mathcal{E}_{G''} = \mu_2(\mathcal{E}_{\mu(G)} \setminus \mathcal{E}_{\mu(H)}) \cup \mathcal{E}_{H'} = \mu_2(\mu(\mathcal{E}_G) \setminus \mu(\mathcal{E}_H)) \cup \mathcal{E}_{H'}$. Since μ is injective, we deduce that $\mathcal{E}_{G''} = \mu_2(\mu(\mathcal{E}_G \setminus \mathcal{E}_H)) \cup \mathcal{E}_{H'} = \mu_1(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{H'} = \mathcal{E}_{G'}$.
- Let $\alpha \in \mathcal{N}_G^* \setminus \mathcal{N}_H^*$, with $\alpha = \mu_1(\alpha')$, hence $\alpha = \mu_2(\mu(\alpha'))$. By Definition 15 (4), we have $l_{G'}(\alpha) = l_G(\alpha')$ and $l_{G''}(\alpha) = l_{\mu(G)}(\mu(\alpha')) = l_G(\alpha') = l_{G'}(\alpha)$. If $\alpha \in \mathcal{N}_{H'}^*$, then, again by Definition 15 (4), we have $l_{G'}(\alpha) = l_{G''}(\alpha) = l_{H'}(\alpha)$.

□

This entails that the replacement of a subgraph by an isomorphic subgraph preserves isomorphism:

COROLLARY 24. *Let G, H, H', H'' be DRL-graphs, where $H \leq G$, H'_1 and H'_2 are substitutable for H in G and $H'_1 \sim H'_2$. Then:*

$$G[H'_1/H] \sim G[H'_2/H]$$

PROOF. Since $H'_1 \sim H'_2$, there exists an \mathcal{N} -mapping μ of domain $\mathcal{N}_{H'_2}$ such that $H'_1 = \mu(H'_2)$. Consider the extension μ' of μ to \mathcal{N}_G , such that: $\mu'(\alpha) = \alpha$ if $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$ and all the nodes in $\mathcal{N}_H \setminus \mathcal{N}_{H'_2}$ are mapped to pairwise distinct nodes not occurring in G or H'_1 . Observe that μ is well-defined, since $\mathcal{N}_G \cap \mathcal{N}_{H'_2} \subseteq \mathcal{N}_H$, as H'_2 is substitutable for H in G . We show that μ' is injective. Let α, α' such that $\mu'(\alpha) = \mu'(\alpha')$. If $\alpha, \alpha' \in \mathcal{N}_{H'_2}$ then $\mu'(\alpha) = \mu(\alpha)$ and $\mu'(\alpha') = \mu(\alpha')$ hence $\alpha = \alpha'$ as μ is injective. If $\alpha, \alpha' \in \mathcal{N}_G \setminus \mathcal{N}_H$ then $\mu'(\alpha) = \alpha$ and $\mu'(\alpha') = \alpha'$ hence $\alpha = \alpha'$. If $\alpha, \alpha' \in \mathcal{N}_H \setminus \mathcal{N}_{H'_2}$ then we have $\alpha = \alpha'$ by definition of μ' . If $\alpha \in \mathcal{N}_H \setminus \mathcal{N}_{H'_2}$ and $\alpha' \notin \mathcal{N}_H \setminus \mathcal{N}_{H'_2}$, then by definition of μ' , $\mu'(\alpha') \in \mathcal{N}_G \cup \mu'(\mathcal{N}_{H'_2}) = \mathcal{N}_G \cup \mathcal{N}_{H'_1}$ and $\mu'(\alpha) \notin \mathcal{N}_G \cup \mathcal{N}_{H'_1}$, which contradicts the fact that $\mu'(\alpha) = \mu'(\alpha')$. The only remaining case is (by symmetry): $\alpha \in \mathcal{N}_{H'_2}$ and $\alpha' \in \mathcal{N}_G \setminus \mathcal{N}_H$. Then $\mu'(\alpha) = \mu(\alpha) \in \mathcal{N}_{H'_1}$ and $\mu'(\alpha') = \alpha' \in \mathcal{N}_G$, hence, since H'_1 is substitutable for H in G , we must have $\alpha' \in \mathcal{N}_H$, contradicting the fact that $\alpha' \in \mathcal{N}_G \setminus \mathcal{N}_H$.

We get, by Proposition 21, since μ' coincides with μ on $\mathcal{N}_{H'_2}$:

$$\mu'(G[H'_2/H]) = \mu'(G)[\mu'(H'_2)/\mu'(H)] = \mu'(G)[H'_1/\mu'(H)].$$

By definition of μ' , $\mu'(\alpha) = \alpha$ holds for any node $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$, hence, by Lemma 23, we deduce:

$$\mu'(G[H'_2/H]) = G[H'_1/H].$$

Therefore, $G[H'_1/H] \sim G[H'_2/H]$. □

The next lemma states a form of transitivity of the replacement operation: replacing H by I and then I by I' is equivalent to replacing H by I' .

LEMMA 25. *Let G, H, I, I' , where $H \leq G$, I is substitutable for H in G and I' is substitutable for I in $G[I/H]$. Then I' is substitutable for H in G and $G[I/H][I'/I] = G[I'/H]$.*

PROOF. Let $G' \stackrel{\text{def}}{=} G[I/H]$. We show that I' is substitutable for H in G , i.e., that $\mathcal{N}_G \cap \mathcal{N}_{I'} \subseteq \mathcal{N}_H$. By definition $\mathcal{N}_{G'} = (\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_I$, we show that $(\mathcal{N}_G \setminus \mathcal{N}_H) \cap \mathcal{N}_{I'} = \emptyset$. We have

$$(\mathcal{N}_G \setminus \mathcal{N}_H) \cap \mathcal{N}_{I'} \subseteq [(\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_I] \cap \mathcal{N}_{I'} = \mathcal{N}_{G'} \cap \mathcal{N}_{I'} \subseteq \mathcal{N}_I,$$

because I' is substitutable for I in G' by hypothesis. We deduce that $(\mathcal{N}_G \setminus \mathcal{N}_H) \cap \mathcal{N}_{I'} \subseteq (\mathcal{N}_G \setminus \mathcal{N}_H) \cap \mathcal{N}_I$. But $(\mathcal{N}_G \setminus \mathcal{N}_H) \cap \mathcal{N}_I = \emptyset$ by Proposition 14, hence the result.

We show that $G'[I'/I] = G[I'/H]$. We define $\mu \stackrel{\text{def}}{=} \mu_G^{H \rightarrow I}$, $\mu' \stackrel{\text{def}}{=} \mu_G^{I \rightarrow I'}$ and $\mu'' \stackrel{\text{def}}{=} \mu_G^{H \rightarrow I'}$. Note that $\mu'' = \mu' \circ \mu$. Indeed, by definition we have $\text{dom}(\mu) = \text{dom}(\mu')$. Furthermore, for every $\alpha \in \text{dom}(\mu)$, if $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$ then $\mu(\alpha) = \mu'(\alpha)$, and $\alpha \notin \mathcal{N}_I$ (since I is substitutable for H in G), hence $\mu''(\alpha) = \alpha$. If $\alpha \in \text{dom}(\mu)$ and $\alpha \notin \mathcal{N}_G \setminus \mathcal{N}_H$, then $\alpha = \alpha_i$, for some $i = 1, \dots, n$, with $\mathcal{R}_H = \langle \alpha_1, \dots, \alpha_n \rangle$. By definition, $\mu(\alpha_i) = \alpha'_i$, with $\mathcal{R}_I = \langle \alpha'_1, \dots, \alpha'_n \rangle$ and $\mu'(\alpha'_i) = \alpha''_i$, $\mu''(\alpha_i) = \alpha''_i$ with $\mathcal{R}_{I'} = \langle \alpha''_1, \dots, \alpha''_n \rangle$.

(1) We have

$$\begin{aligned} \mathcal{N}_{G'}[I'/I] &= (\mathcal{N}_{G'} \setminus \mathcal{N}_I) \cup \mathcal{N}_{I'} \\ &= [((\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_I) \setminus \mathcal{N}_I] \cup \mathcal{N}_{I'} \\ &= [(\mathcal{N}_G \setminus \mathcal{N}_H) \setminus \mathcal{N}_I] \cup \mathcal{N}_{I'} \\ &= (\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_{I'} \\ &= \mathcal{N}_G[I'/H], \end{aligned}$$

where the second to last equality is obtained using the fact that, by Proposition 14, $(\mathcal{N}_G \setminus \mathcal{N}_H) \cap \mathcal{N}_I = \emptyset$.

(2) We have $\mathcal{R}_{G'}[I'/I] = \mu'(\mathcal{R}_{G'}) = \mu'(\mu(\mathcal{R}_G)) = \mu''(\mathcal{R}_G)$.

(3) We show that $\mu(\mathcal{E}_G \setminus \mathcal{E}_H) \cap \mathcal{E}_I = \emptyset$. Suppose for a contradiction that $(\alpha' \rightleftharpoons \beta') \in \mu(\mathcal{E}_G \setminus \mathcal{E}_H) \cap \mathcal{E}_I$. Then $\alpha', \beta' \in \mathcal{N}_I$, and there exist nodes α, β such that $\alpha' = \mu(\alpha)$ and $\beta' = \mu(\beta)$, with $(\alpha \rightleftharpoons \beta) \in \mathcal{E}_G \setminus \mathcal{E}_H$. Since $(\alpha \rightleftharpoons \beta) \notin \mathcal{E}_H$, necessarily, $\{\alpha, \beta\} \not\subseteq \mathcal{N}_H$, by Definition 4 (3). Assume w.l.o.g. that $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$. Then by definition $\alpha' = \alpha \in \mathcal{N}_I$. But this is impossible because $(\mathcal{N}_G \setminus \mathcal{N}_H) \cap \mathcal{N}_I = \emptyset$ by Proposition 14. We deduce that

$$\begin{aligned} \mathcal{E}_{G'}[I'/I] &= \mu'(\mathcal{E}_{G'} \setminus \mathcal{E}_I) \cup \mathcal{E}_{I'} \\ &= \mu'([\mu(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_I] \setminus \mathcal{E}_I) \cup \mathcal{E}_{I'} \\ &= \mu'(\mu(\mathcal{E}_G \setminus \mathcal{E}_H) \setminus \mathcal{E}_I) \cup \mathcal{E}_{I'} \\ &= \mu''(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{I'} \\ &= \mathcal{E}_G[I'/H], \end{aligned}$$

where the second to last equality is obtained using the fact that $\mu(\mathcal{E}_G \setminus \mathcal{E}_H) \cap \mathcal{E}_I = \emptyset$ and that $\mu'' = \mu' \circ \mu$.

(4) Let $\alpha \in \mathcal{N}_{G'}[I'/I]$.

- If $\alpha \in \mathcal{N}_{G'}^*[I'/I] \setminus \mathcal{N}_{I'}$, then by Proposition 18 we have $\alpha \in \mathcal{N}_{G'}^*[I'/I] \setminus \mathcal{N}_{I'}$ if and only if $\alpha \in \mathcal{N}_{G'}^* \setminus \mathcal{N}_I$ if and only if $\alpha \in \mathcal{N}_G^* \setminus \mathcal{N}_H$ if and only if $\alpha \in \mathcal{N}_G^*[I'/H] \setminus \mathcal{N}_{I'}$. Thus $l_{G'}[I'/I](\alpha) = l_G(\alpha) = l_{G'}[I'/H](\alpha)$.
- If $\alpha \in \mathcal{N}_{I'}$, then $l_{G'}[I'/I](\alpha) = l_{I'}(\alpha) = l_{G'}[I'/H](\alpha)$.
- If $\beta \stackrel{\text{def}}{=} \mu^{-1}(\alpha) \in \mathcal{N}_{G'}^* \cap \mathcal{R}_I$ then by definition of μ , we must have $\mu^{-1}(\beta) \in \mathcal{R}_H$. We cannot have $\beta \in \mathcal{R}_G$ because otherwise we would have $\beta \in \mathcal{R}_{G'}$; hence $\mu^{-1}(\beta) \in \mathcal{N}_G^* \cap \mathcal{R}_H$. Since $\mu'' = \mu' \circ \mu$, we deduce that $\mu''^{-1}(\alpha) = \mu^{-1}(\beta)$ and $l_{G'}[I'/I](\alpha) = l_{G'}(\beta) = l_G(\mu^{-1}(\beta)) = l_{G'}[I'/H](\alpha)$. By Proposition 18 we have $\alpha \in \mathcal{N}_{G'}^*[I'/I] \setminus \mathcal{N}_{I'}$ if and only if $\alpha \in \mathcal{N}_{G'}^* \setminus \mathcal{N}_I$ if and only if $\alpha \in \mathcal{N}_G^* \setminus \mathcal{N}_H$. This proves that $l_{G'}[I'/I](\alpha) = l_{G'}[I'/H](\alpha)$.

□

In particular, replacing H by H' and then H' by H is equivalent to doing nothing:

COROLLARY 26. *Assume H' is substitutable for H in G and let $G' \stackrel{\text{def}}{=} G[H'/H]$. Then H is substitutable for H' in G' and $G'[H/H'] = G$.*

PROOF. We have $\mathcal{N}_{G'} \cap \mathcal{N}_H = [(\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_{H'}] \cap \mathcal{N}_H = \mathcal{N}_{H'} \cap \mathcal{N}_H \subseteq \mathcal{N}_{H'}$, which show that H is substitutable for H' in G' . By Lemma 25 and Proposition 20, we deduce that $G[H'/H][H/H'] = G[H/H] = G$. □

The next lemma states that the replacement operation does not affect the subgraph that are disjoint from the replaced DRL-graph.

LEMMA 27. Consider the graphs G, H, H', I , and assume that $H \leq G$, $I \leq G$ and H' is substitutable for H in G . If $\mathcal{N}_H \cap \mathcal{N}_I = \emptyset$ then $I \leq G[H'/H]$.

PROOF. Let $G' \stackrel{\text{def}}{=} G[H'/H]$ and $\mu \stackrel{\text{def}}{=} \mu_G^{H \rightarrow H'}$. Note that since H' is substitutable for H in G , we have

$$\mathcal{N}_{H'} \cap \mathcal{N}_I \subseteq \mathcal{N}_H \cap \mathcal{N}_I \subseteq \mathcal{N}_H \cap \mathcal{N}_I = \emptyset,$$

and that since $\mathcal{N}_H \cap \mathcal{N}_I = \emptyset$, if $\alpha \in \mathcal{N}_I$ then $\mu(\alpha) = \alpha$. We verify that $I \leq G'$:

- (1) Since $I \leq G$, by definition $\mathcal{N}_I \subseteq \mathcal{N}_G$. By hypothesis $\mathcal{N}_H \cap \mathcal{N}_I = \emptyset$, thus $\mathcal{N}_I \subseteq \mathcal{N}_G \setminus \mathcal{N}_H \subseteq \mathcal{N}_{G'}$.
- (2) If $(\alpha \rightarrow \beta) \in \mathcal{E}_I$ then $(\alpha \rightarrow \beta) \in \mathcal{E}_G$, because $I \leq G$. Since $\mathcal{N}_H \cap \mathcal{N}_I = \emptyset$ by hypothesis, $(\alpha \rightarrow \beta) \notin \mathcal{E}_H$. Thus $(\alpha \rightarrow \beta) \in (\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{H'}$, and since $\mu(\alpha) = \alpha$ and $\mu(\beta) = \beta$, we deduce that $(\alpha \rightarrow \beta) \in \mu(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{H'} = \mathcal{E}_{G'}$.
- (3) Let $\alpha, \beta \in \mathcal{N}_I$, and assume that $(\alpha \rightarrow \beta) \in \mathcal{E}_{G'} = \mu(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{H'}$. Since $\mathcal{N}_H \cap \mathcal{N}_I = \emptyset$, we cannot have $(\alpha \rightarrow \beta) \in \mathcal{E}_{H'}$. Since $\mu(\alpha) = \alpha$ and $\mu(\beta) = \beta$, necessarily $(\alpha \rightarrow \beta) \in \mathcal{E}_G \setminus \mathcal{E}_H \subseteq \mathcal{E}_G$, and since $I \leq G$, we deduce that $(\alpha \rightarrow \beta) \in \mathcal{E}_I$.
- (4) Assume that $(\alpha \rightleftharpoons \beta) \in \mathcal{E}_{G'} = \mu(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{H'}$, where $\alpha \in \mathcal{N}_I$ and $\beta \notin \mathcal{N}_I$. Then $\mu(\alpha) = \alpha$ and since $\mathcal{N}_{\text{agragraph}B'} \cap \mathcal{N}_I = \emptyset$, we cannot have $(\alpha \rightleftharpoons \beta) \in \mathcal{E}_{H'}$. Let $\beta' \stackrel{\text{def}}{=} \mu^{-1}(\beta)$, note that we cannot have $\beta \in \mathcal{N}_I$ because otherwise we would have $\mu(\beta') = \beta' = \beta \in \mathcal{N}_I$. Thus, $(\alpha \rightleftharpoons \beta') \in \mathcal{E}_G \setminus \mathcal{E}_H \subseteq \mathcal{E}_G$, and since $I \leq G$, we deduce that $\alpha \in \mathcal{R}_I$.
- (5) Assume $\alpha \in \mathcal{R}_{G'} \cap \mathcal{N}_I = \mu(\mathcal{R}_G) \cap \mathcal{N}_I$. Then since $\mu(\alpha) = \alpha$, we have $\alpha \in \mathcal{R}_G \cap \mathcal{N}_I$, hence $\alpha \in \mathcal{R}_I$ because $I \leq G$.
- (6) Consider $\alpha \in \mathcal{N}_I^* \subseteq \mathcal{N}_{G'}^*$. Then by hypothesis, $\alpha \in \mathcal{N}_G^* \setminus \mathcal{N}_H$ and by Proposition 18 we have $\alpha \in \mathcal{N}_{G'}^* \setminus \mathcal{N}_{H'}$. By definition, $l_{G'}(\alpha) = l_G(\alpha) = l_I(\alpha)$, because $I \leq G$.

□

Further, the result of the replacement of two disjoint subgraphs does not depend on the order in which the replacement operations are performed:

LEMMA 28. Let G, H_i, H'_i be DRL-graphs for $i = 1, 2$, where $H_i \leq G$ and H'_i is substitutable for H_i in G . If $\mathcal{N}_{H'_1} \cap \mathcal{N}_{H'_2} = \mathcal{N}_{H_1} \cap \mathcal{N}_{H_2} = \emptyset$ then, for $i, j \in \{1, 2\}$ with $i \neq j$, H'_i is substitutable for H_i in $G[H'_j/H_j]$ and

$$(G[H'_1/H_1])[H'_2/H_2] = (G[H'_2/H_2])[H'_1/H_1].$$

PROOF. By Lemma 27, $H_1 \leq G[H'_2/H_2]$ and $H_2 \leq G[H'_1/H_1]$. We show that H'_1 is substitutable for H_1 in $G[H'_2/H_2]$; the proof that H'_2 is substitutable for H_2 in $G[H'_1/H_1]$ is symmetric. Let $\alpha \in \mathcal{N}_{H'_1} \cap \mathcal{N}_{G[H'_2/H_2]} = \mathcal{N}_{H'_1} \cap \left[(\mathcal{N}_G \setminus \mathcal{N}_{H_2}) \cup \mathcal{N}_{H'_2} \right]$. By hypothesis $\mathcal{N}_{H'_1} \cap \mathcal{N}_{H'_2} = \emptyset$, hence $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_{H_2} \subseteq \mathcal{N}_G$, and since H'_1 is substitutable for H_1 in G , we deduce that $\alpha \in \mathcal{N}_{H_1}$, hence the result.

Let $G'_i \stackrel{\text{def}}{=} G[H'_i/H_i]$ and $G''_i \stackrel{\text{def}}{=} G'_{3-i}[H'_i/H_i]$. We show that $G''_1 = G''_2$. First note that $\mathcal{N}_{H_1} \cap \mathcal{N}_{H'_2} = \mathcal{N}_{H_2} \cap \mathcal{N}_{H'_1} = \emptyset$. Indeed, since $H_1 \leq G$ and H'_2 is substitutable for H_2 , we have $\mathcal{N}_{H_1} \cap \mathcal{N}_{H'_2} \subseteq \mathcal{N}_{H_1} \cap \mathcal{N}_G \cap \mathcal{N}_{H'_2} \subseteq \mathcal{N}_{H_1} \cap \mathcal{N}_{H_2} = \emptyset$. For $i = 1, 2$, we define $\mu_i \stackrel{\text{def}}{=} \mu_G^{H_i \rightarrow H'_i}$ and $\mu'_i \stackrel{\text{def}}{=} \mu_{G'_{3-i}}^{H_i \rightarrow H'_i}$. Note that by definition, for $i = 1, 2$, if $\alpha \in \mathcal{R}_{H_i}$ then $\mu_i(\alpha) = \mu'_i(\alpha) \in \mathcal{R}_{H'_i}$, so that $\mu_{3-i}(\alpha) = \alpha$ and

$\mu'_{3-i}(\mu_i(\alpha)) = \mu_i(\alpha)$. Similarly, if $\alpha \in \mathcal{N}_G \setminus (\mathcal{N}_{H_1} \cup \mathcal{N}_{H_2})$ then for $i = 1, 2$, $\mu_i(\alpha) = \mu'_i(\alpha) = \alpha$.

- We have $\mathcal{N}_{G'_1} = (\mathcal{N}_{G_2} \setminus \mathcal{N}_{H_1}) \cup \mathcal{N}_{H'_1} = ((\mathcal{N}_G \setminus \mathcal{N}_{H_2}) \cup \mathcal{N}_{H'_2}) \setminus \mathcal{N}_{H_1} \cup \mathcal{N}_{H'_1}$. Since $\mathcal{N}_{H_1} \cap \mathcal{N}_{H'_2} = \mathcal{N}_{H_2} \cap \mathcal{N}_{H'_1} = \emptyset$, we deduce that

$$\begin{aligned} \mathcal{N}_{G'_1} &= ((\mathcal{N}_G \cup \mathcal{N}_{H'_1} \cup \mathcal{N}_{H'_2}) \setminus (\mathcal{N}_{H_1} \cup \mathcal{N}_{H_2})) \\ &= ((\mathcal{N}_G \setminus \mathcal{N}_{H_1}) \cup \mathcal{N}_{H'_1} \setminus \mathcal{N}_{H_2}) \cup \mathcal{N}_{H'_2} \\ &= (\mathcal{N}_{G_1} \setminus \mathcal{N}_{H_2}) \cup \mathcal{N}_{H'_2} = \mathcal{N}_{G'_2}. \end{aligned}$$

- By definition we have $\mathcal{R}_{G'_1} = \mu'_1(\mathcal{R}_{G_2}) = \mu'_1(\mu_2(\mathcal{R}_G))$ and $\mathcal{R}_{G'_2} = \mu'_2(\mathcal{R}_{G_1}) = \mu'_2(\mu_1(\mathcal{R}_G))$. Consider $\alpha \in \mathcal{N}_G$. If $\alpha \in \mathcal{R}_{H_i}$ for $i = 1, 2$, then we have

$$\mu'_1(\mu_2(\alpha)) = \mu'_1(\alpha) = \mu_1(\alpha) = \mu'_2(\mu_1(\alpha)).$$

Otherwise $\alpha \in \mathcal{R}_G \setminus (\mathcal{N}_{H_1} \cup \mathcal{N}_{H_2})$, so that

$$\mu'_1(\mu_2(\alpha)) = \mu'_1(\alpha) = \alpha = \mu_1(\alpha) = \mu'_2(\mu_1(\alpha)).$$

We conclude that $\mathcal{R}_{G'_1} = \mathcal{R}_{G'_2}$.

- For $i = 1, 2$, we have $\mu_{3-i}(\mathcal{E}_{H_i}) = \mathcal{E}_{H_i}$ and $\mu'_{3-i}(\mathcal{E}_{H'_i}) = \mathcal{E}_{H'_i}$, hence

$$\begin{aligned} \mathcal{E}_{G'_1} &= \mu'_2(\mathcal{E}_{G_2} \setminus \mathcal{E}_{H_2}) \cup \mathcal{E}_{H'_2} \\ &= \mu'_2((\mu_1(\mathcal{E}_G \setminus \mathcal{E}_{H_1}) \cup \mathcal{E}_{H'_1}) \setminus \mathcal{E}_{H_2}) \cup \mathcal{E}_{H'_2} \\ &= \mu'_2(\mu_1(\mathcal{E}_G \setminus \mathcal{E}_{H_1}) \setminus \mathcal{E}_{H_2}) \cup \mathcal{E}_{H'_1} \cup \mathcal{E}_{H'_2} \\ &= \mu'_2(\mu_1(\mathcal{E}_G \setminus (\mathcal{E}_{H_1} \cup \mathcal{E}_{H_2}))) \cup \mathcal{E}_{H'_1} \cup \mathcal{E}_{H'_2} \\ &= \mu'_1(\mu_2(\mathcal{E}_G \setminus (\mathcal{E}_{H_1} \cup \mathcal{E}_{H_2}))) \cup \mathcal{E}_{H'_1} \cup \mathcal{E}_{H'_2} \\ &= \mathcal{E}_{G'_2} \end{aligned}$$

The second to last line stems from the relation $\forall \alpha \in \mathcal{N}_G$, $\mu'_1(\mu_2(\alpha)) = \mu'_2(\mu_1(\alpha))$ established in the previous item.

- Since $\mathcal{N}_{G'_1} = \mathcal{N}_{G'_2}$ and $\mathcal{R}_{G'_1} = \mathcal{R}_{G'_2}$, we have $\mathcal{N}_{G'_1}^* = \mathcal{N}_{G'_2}^*$.

We show that for all $\alpha \in \mathcal{N}_{G'_1}^*$, $l_{G'_1}(\alpha) = l_{G'_2}(\alpha)$.

– By Proposition 18 we have $\mathcal{N}_{G'_2}^* \setminus (\mathcal{N}_{H'_1} \cup \mathcal{N}_{H'_2}) = \mathcal{N}_{G'_1}^* \setminus (\mathcal{N}_{H'_1} \cup \mathcal{N}_{H'_2}) = \mathcal{N}_{G'_2}^* \setminus (\mathcal{N}_{H_1} \cup \mathcal{N}_{H_2}) = \mathcal{N}_G^* \setminus (\mathcal{N}_{H_1} \cup \mathcal{N}_{H_2})$. Thus, if $\alpha \in \mathcal{N}_{G'_2}^* \setminus (\mathcal{N}_{H'_1} \cup \mathcal{N}_{H'_2})$ then $l_{G'_1}(\alpha) = l_G(\alpha) = l_{G'_2}(\alpha)$.

– If $\alpha \in \mathcal{N}_{H'_2}^*$ then $l_{G'_1}(\alpha) = l_{G'_2}(\alpha) = l_{H'_2}(\alpha) = l_{G'_2}(\alpha)$. The case where $\alpha \in \mathcal{N}_{H'_1}^*$ is proved in a similar way.

– If $\alpha \in \mathcal{N}_{G'_1}^* \cap \mathcal{R}_{H'_1}$ then $\alpha \in \mathcal{N}_{G'_2}^* \setminus \mathcal{N}_{H'_2}$ and $\alpha = \mu_1(\beta) = \mu'_1(\beta)$ for some element $\beta \in \mathcal{R}_{H_1}$. Since $\beta \in \mathcal{N}_{G'_2}^* \setminus \mathcal{N}_{H'_2} = \mathcal{N}_G^* \setminus \mathcal{N}_{H_2}$ (Proposition 18), we have $l_{G'_1}(\alpha) = l_{G'_2}(\beta) = l_G(\beta) = l_{G'_1}(\alpha) = l_{G'_2}(\alpha)$. The case where $\alpha \in \mathcal{N}_{G'_2}^* \cap \mathcal{R}_{H'_2}$ is proved in a similar way.

□

Finally, replacements can be embedded:

LEMMA 29. Let G, H and I be DRL-graphs such that $I \leq H \leq G$, and let I' be a DRL-graph substitutable for I in G . Then I' is substitutable for I in H , $H[I'/I]$ is substitutable for H in G , and:

$$G[I'/I] = G[H[I'/I]/H]$$

PROOF. We have $\mathcal{N}_H \cap \mathcal{N}_{I'} \subseteq \mathcal{N}_G \cap \mathcal{N}_{I'} \subseteq \mathcal{N}_I$ (since I' is substitutable for I in G). Therefore, I' is substitutable for I in H . Similarly, $\mathcal{N}_G \cap \mathcal{N}_H[I'/I] \subseteq \mathcal{N}_G \cap (\mathcal{N}_H \cup \mathcal{N}_{I'}) = \mathcal{N}_H \cup (\mathcal{N}_G \cap \mathcal{N}_{I'}) \subseteq \mathcal{N}_H \cup \mathcal{N}_I = \mathcal{N}_H$, hence $H[I'/I]$ is substitutable for H in G .

Let $G' \stackrel{\text{def}}{=} G[I'/I]$, $H' \stackrel{\text{def}}{=} H[I'/I]$ and $G'' \stackrel{\text{def}}{=} G[H'/H]$; we show that $G' = G''$.

Let $\mu_1 \stackrel{\text{def}}{=} \mu_G^{I \rightarrow I'}$, $\mu_2 \stackrel{\text{def}}{=} \mu_H^{I \rightarrow I'}$ and $\mu_3 \stackrel{\text{def}}{=} \mu_G^{H \rightarrow H'}$. Note that by definition, μ_2 is the restriction of μ_1 to the nodes in $(\mathcal{N}_H \setminus \mathcal{N}_I) \cup \mathcal{R}_I = \mathcal{N}_H \setminus \mathcal{N}_I^*$. We show that for all $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H^* = (\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{R}_H$, we have $\mu_3(\alpha) = \mu_1(\alpha)$. If $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$ then since $\mathcal{N}_I \subseteq \mathcal{N}_H$, we have $\mu_3(\alpha) = \mu_1(\alpha) = \alpha$. Otherwise, $\alpha \in \mathcal{R}_H$ and by Proposition 17, $\mu_3(\alpha) = \mu_2(\alpha)$. Since $\mathcal{R}_H \subseteq \mathcal{N}_H \setminus \mathcal{N}_I^*$ by Proposition 6, we deduce that $\mu_3(\alpha) = \mu_2(\alpha) = \mu_1(\alpha)$.

- By Definition 15 (1), $\mathcal{N}_{G'} = (\mathcal{N}_G \setminus \mathcal{N}_H) \cup \mathcal{N}_{H'} = (\mathcal{N}_G \setminus \mathcal{N}_H) \cup (\mathcal{N}_H \setminus \mathcal{N}_I) \cup \mathcal{N}_{I'} = (\mathcal{N}_G \setminus \mathcal{N}_I) \cup \mathcal{N}_{I'} = \mathcal{N}_{G''}$.
- By Definition 15 (2), $\mathcal{R}_{G'} = \mu_3(\mathcal{R}_G)$ and $\mathcal{R}_{G''} = \mu_1(\mathcal{R}_G)$. Since $\mathcal{R}_G \subseteq \mathcal{N}_G \setminus \mathcal{N}_H^*$ by Proposition 6, we have $\mu_3(\mathcal{R}_G) = \mu_1(\mathcal{R}_G)$ so that $\mathcal{R}_{G'} = \mathcal{R}_{G''}$.
- Consider an edge $(\alpha \rightrightarrows \beta)$ in $\mathcal{E}_H \setminus \mathcal{E}_I$. Then since $I \leq H$, we cannot have $\{\alpha, \beta\} \subseteq \mathcal{N}_I$ and if $\alpha \in \mathcal{N}_I$ then necessarily $\alpha \in \mathcal{R}_I$. This shows that $\{\alpha, \beta\} \subseteq \mathcal{N}_H \setminus \mathcal{N}_I^*$ and therefore, $\mu_2(\mathcal{E}_H \setminus \mathcal{E}_I) = \mu_1(\mathcal{E}_H \setminus \mathcal{E}_I)$. The fact that $\mu_3(\mathcal{E}_G \setminus \mathcal{E}_H) = \mu_1(\mathcal{E}_G \setminus \mathcal{E}_H)$ is proved in a similar way. We have

$$\begin{aligned} \mathcal{E}_{G'} &= \mu_3(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mathcal{E}_{H'} \\ &= \mu_3(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mu_2(\mathcal{E}_H \setminus \mathcal{E}_I) \cup \mathcal{E}_{I'} \\ &= \mu_1(\mathcal{E}_G \setminus \mathcal{E}_H) \cup \mu_1(\mathcal{E}_H \setminus \mathcal{E}_I) \cup \mathcal{E}_{I'} \\ &= \mu_1((\mathcal{E}_G \setminus \mathcal{E}_H) \cup (\mathcal{E}_H \setminus \mathcal{E}_I)) \cup \mathcal{E}_{I'} \quad (\mu_1 \text{ is injective}) \\ &= \mu_1(\mathcal{E}_G \setminus \mathcal{E}_I) \cup \mathcal{E}_{I'} \\ &= \mathcal{E}_{G''} \end{aligned}$$

- Consider a node $\alpha \in \mathcal{N}_{G'}^*$. Since $\mathcal{N}_{G'} = \mathcal{N}_{G''}$ and $\mathcal{R}_{G'} = \mathcal{R}_{G''}$, we also have $\alpha \in \mathcal{N}_{G''}^*$. First assume that $\alpha \in \mathcal{N}_{G'}^* \setminus \mathcal{N}_{I'}$, so that $l_{G'}(\alpha) = l_G(\alpha)$.
 - If $\alpha \in \mathcal{N}_{G''}^* \setminus \mathcal{N}_{H'}$ then by definition $l_{G''}(\alpha) = l_G(\alpha)$.
 - If $\alpha \in \mathcal{N}_{H'}^*$ then since $\alpha \notin \mathcal{N}_{I'}$ by hypothesis and $H \leq G$, we have $l_{G''}(\alpha) = l_{H'}(\alpha) = l_H(\alpha) = l_G(\alpha)$.
 - If $\mu_3^{-1}(\alpha) \in \mathcal{N}_G^* \cap \mathcal{R}_H$ then $l_{G''}(\alpha) = l_G(\mu_3^{-1}(\alpha))$. But since $\alpha \notin \mathcal{N}_I$, we must have $\mu_3^{-1}(\alpha) = \mu_2^{-1}(\alpha) = \alpha$, hence the result.

Assume that $\alpha \in \mathcal{N}_{I'}^*$, so that $l_{G'}(\alpha) = l_{I'}(\alpha)$.

- We cannot have $\alpha \in \mathcal{N}_{G''}^* \setminus \mathcal{N}_{H'}$ because $\mathcal{N}_{I'}^* \subseteq \mathcal{N}_{H'}$.
- If $\alpha \in \mathcal{N}_{H'}^*$ then $l_{G''}(\alpha) = l_{H'}(\alpha) = l_{I'}(\alpha)$ because $\alpha \in \mathcal{N}_{I'}^*$ by hypothesis.
- We cannot have $\mu_3^{-1}(\alpha) \in \mathcal{N}_G^* \cap \mathcal{R}_H$ because otherwise we would have $\alpha \in \mu_3(\mathcal{R}_H) = \mathcal{R}_{H'}$ and since $I' \leq H'$, necessarily $\alpha \in \mathcal{R}_{I'}$.

Now assume that $\alpha \in \mathcal{N}_{G'}^* \cap \mathcal{R}_{I'}$, i.e., $\mu_1^{-1}(\alpha) \in \mathcal{N}_G^* \cap \mathcal{R}_I$, so that $l_{G'}(\alpha) = l_G(\mu_1^{-1}(\alpha))$.

- We cannot have $\alpha \in \mathcal{N}_{G''}^* \setminus \mathcal{N}_{H'}$ because $\mathcal{R}_{I'} \subseteq \mathcal{N}_{H'}$.
- If $\alpha \in \mathcal{N}_{H'}^*$ then we have $l_{G''}(\alpha) = l_{H'}(\alpha) = l_H(\mu_2^{-1}(\alpha)) = l_H(\mu_1^{-1}(\alpha)) = l_G(\mu_1^{-1}(\alpha))$.
- If $\mu_3^{-1}(\alpha) \in \mathcal{N}_G^* \cap \mathcal{R}_H \subseteq \mathcal{N}_G^* \setminus \mathcal{N}_H^*$ then we have $l_{G''}(\alpha) = l_G(\mu_3^{-1}(\alpha)) = l_G(\mu_1^{-1}(\alpha))$ because μ_1 and μ_3 coincide on $\mathcal{N}_G^* \setminus \mathcal{N}_H^*$.

□

3.3 Graph Merging

We introduce the notion of a *merge* of two graphs. Intuitively, the merge of graphs G_1 and G_2 will be any graph G containing all nodes and edges occurring in either G_1 or G_2 , possibly along with some additional edges.

DEFINITION 30. Two DRL-graphs G_1 and G_2 are label-compatible iff the set of pairs $\{(l_{G_1}(\beta), l_{G_2}(\beta)) \mid \beta \in \mathcal{N}_{G_1} \cap \mathcal{N}_{G_2}\}$ admits an mgu.

Given two label-compatible DRL-graphs G_1, G_2 with mgu σ and a set of edges E of the form $(\alpha \rightrightarrows \beta)$ such that $\alpha \in \mathcal{R}_{G_1} \setminus \mathcal{N}_{G_2}$ and $\beta \in \mathcal{R}_{G_2} \setminus \mathcal{N}_{G_1}$, the E -merge of G_1 and G_2 is the graph G defined as follows:

- $\mathcal{N}_G \stackrel{\text{def}}{=} \mathcal{N}_{G_1} \cup \mathcal{N}_{G_2}$;
- $\mathcal{R}_G \stackrel{\text{def}}{=} (\mathcal{R}_{G_1} \cup \mathcal{R}_{G_2}) \setminus (\mathcal{N}_{G_1}^* \cup \mathcal{N}_{G_2}^*)$ (the order of the nodes in \mathcal{R}_G is chosen arbitrarily);
- $\mathcal{E}_G \stackrel{\text{def}}{=} \mathcal{E}_{G_1} \cup \mathcal{E}_{G_2} \cup E$;
- for every node $\alpha \in \mathcal{N}_{G_i}$ with $i = 1, 2$, $l_G(\alpha) \stackrel{\text{def}}{=} \sigma(l_{G_i}(\alpha))$.

A merge of two label-compatible DRL-graphs G_1, G_2 is a DRL-graph G that is an E -merge of G_1 and G_2 , for some set E of edges of the form $(\alpha \rightrightarrows \beta)$ such that $\alpha \in \mathcal{R}_{G_1} \setminus \mathcal{N}_{G_2}$ and $\beta \in \mathcal{R}_{G_2} \setminus \mathcal{N}_{G_1}$.

It is easy to check that merges can be constructed using usual unification algorithms.

PROPOSITION 31. Assume $H, H' \leq G$ and consider the set of edges $E \stackrel{\text{def}}{=} \{(\alpha \rightrightarrows \beta) \mid \alpha \in \mathcal{R}_H \setminus \mathcal{N}_{H'}, \beta \in \mathcal{R}_{H'} \setminus \mathcal{N}_H, (\alpha \rightrightarrows \beta) \in \mathcal{E}_G\}$. Then H and H' are label-compatible and if G' is the E -merge of H and H' , then $G' \leq G$.

PROOF. It is straightforward to verify that H and H' are label-compatible, with the empty mgu. We show that $G' \leq G$, using the fact that, by hypothesis, $H, H' \leq G$.

- (1) By Definition 30, we have $\mathcal{N}_{G'} = \mathcal{N}_H \cup \mathcal{N}_{H'} \subseteq \mathcal{N}_G$.
- (2) All the edges in E are also in \mathcal{E}_G , it is thus straightforward to verify that $\mathcal{E}_{G'} \subseteq \mathcal{E}_G$.
- (3) Assume that $(\alpha \rightarrow \beta) \in \mathcal{E}_{G'}$, where $\alpha, \beta \in \mathcal{N}_{G'}$. If $\alpha, \beta \in \mathcal{N}_H$ or $\alpha, \beta \in \mathcal{N}_{H'}$ then $(\alpha \rightarrow \beta) \in \mathcal{E}_G$ since $H, H' \leq G$. Suppose that $\alpha \in \mathcal{N}_H \setminus \mathcal{N}_{H'}$ and $\beta \in \mathcal{N}_{H'} \setminus \mathcal{N}_H$. Since $H, H' \leq G$, by Definition 4 (4), necessarily $\alpha \in \mathcal{R}_H$ and $\beta \in \mathcal{R}_{H'}$, hence $(\alpha \rightarrow \beta) \in E \subseteq \mathcal{E}_{G'}$. The case where $\alpha \in \mathcal{N}_H \setminus \mathcal{N}_{H'}$ and $\beta \in \mathcal{N}_{H'} \setminus \mathcal{N}_H$ is proved in a similar way.
- (4) Consider $(\alpha \rightrightarrows \beta) \in \mathcal{E}_G$, where $\alpha \in \mathcal{N}_{G'}$ and $\beta \notin \mathcal{N}_{G'}$. Assume $\alpha \in \mathcal{N}_H$, the case where $\alpha \in \mathcal{N}_{H'}$ is proved in a similar way. Since $\beta \notin \mathcal{N}_{G'}$ we have $\beta \notin \mathcal{N}_H$, hence $\alpha \in \mathcal{R}_H$ by Definition 4 (4). It cannot be the case that $\alpha \in \mathcal{N}_{H'}^*$, because otherwise we would have $\alpha \in \mathcal{N}_{H'}$ and $\beta \notin \mathcal{N}_{H'}$, so that $\alpha \in \mathcal{R}_{H'}$, a contradiction. We conclude that $\alpha \in \mathcal{R}_G$.
- (5) Consider a node $\alpha \in \mathcal{R}_G \cap \mathcal{N}_{G'}$ and suppose w.l.o.g. that $\alpha \in \mathcal{N}_H$. Since $H \leq G$, by Definition 4 (5) necessarily $\alpha \in \mathcal{R}_H$. If $\alpha \in \mathcal{N}_{H'}$ then $\alpha \in \mathcal{R}_{H'}$, we deduce that $\alpha \in \mathcal{R}_H \setminus (\mathcal{N}_H^* \cup \mathcal{N}_{H'}^*) \subseteq \mathcal{R}_{G'}$.
- (6) By definition (since the considered substitution σ is empty) $l_{G'}$ is the restriction of l_G to $\mathcal{N}_{G'}$.

□

3.4 Restricting the Class of Graphs: C -Relations

Depending on the intended applications, it is sometimes necessary to further restrict the considered class of graphs. For instance, if graphs are intended to model standard circuits obtained by composing gates associated with inputs and outputs, then it might be necessary to dismiss cyclic graphs, which have no obvious semantics in this particular context. We shall thus consider an arbitrary but fixed class of DRL-graphs C , which is assumed to be closed under isomorphism. The class C should be provided by a user, depending on the considered application (an example is provided in Section 8 to model possibly cyclic circuits).

To perform equational reasoning, we need to ensure that the replacement of a subgraph within a DRL-graph in C results in a DRL-graph that is still in C . It is clear that this property will not hold in general, even if the considered subgraph is also in C . For instance, the replacement of a subgraph in a non-cyclic DRL-graph may create new paths in the DRL-graph, which may create a cycle. To overcome this issue, we shall consider a restriction of the subgraph relation, denoting the particular subgraphs on which replacement is allowed. This relation is called a C -relation and will be denoted by \leq_C . In the following we only state the abstract properties that must be fulfilled by the relation \leq_C , its actual definition will depend on the considered application and on the class C . In most cases, \leq_C will coincide with \leq on graphs in C (this is the case for the class defined in Definition 62). However, we prefer, for the sake of generality, not to make this assumption.

DEFINITION 32. Let C be a set of DRL-graphs, such that $G \in C \wedge G \sim G' \Rightarrow G' \in C$. A C -relation \leq_C is a reflexive and transitive binary relation included in \leq and satisfying the following properties:

- (1) If $H \leq_C G$ then $G \in C$ and $H \in C$.
- (2) If $H \leq_C G$, $H' \in C$ and H' is substitutable for H in G then $H' \leq_C G[H'/H]$ (thus $G[H'/H] \in C$).
- (3) If $H \leq_C G$, then $\mu(H) \leq_C \mu(G)$, for every N -mapping μ .
- (4) If $H \leq_C G$, $I \leq_C G$, H and I are disjoint, $I' \in C$ and I' is substitutable for I in G , then $H \leq_C G[I'/I]$.
- (5) If $G_i \leq_C G$, for $i = 1, 2$, then there exists a merge G' of G_1 and G_2 , such that $G' \leq_C G$.
- (6) If $G \leq_C H$ then for every substitution σ , $\sigma(G) \leq_C \sigma(H)$.

In the following we consider a fixed set of DRL-graphs C , closed under isomorphism, and a fixed C -relation \leq_C . Note that in particular, if C is closed under merging, then the intersection of \leq with C^2 is always a C -relation.

3.5 Graph Rewrite Systems

In what follows, we define the notion of a *graph rewrite system* that is based on the replacement operation. Note that the rewrite relation is parameterized by \leq_C , as the replacement of a subgraph H within G is allowed only if $H \leq_C G$.

DEFINITION 33. A DRL-graph rewrite rule is a pair written $G \rightarrow H$, where G and H are DRL-graphs such that \mathcal{R}_G and \mathcal{R}_H are root-compatible. A DRL-graph rewrite system is a set of DRL-graph rewrite rules.

DEFINITION 34. If R is a DRL-graph rewrite system then we write $G \rightarrow_R G'$ iff there exists a rule $H \rightarrow H' \in R$, a substitution σ and an N -mapping μ of domain $\mathcal{N}_H \cup \mathcal{N}_{H'}$ such that the following conditions hold:

- $\sigma(\mu(H)) \leq_C G$,
- $\sigma(\mu(H'))$ is substitutable for $\sigma(\mu(H))$ in G , and
- $G' \sim G[\sigma(\mu(H'))/\sigma(\mu(H))]$.

We denote by \rightarrow_R^+ the transitive closure of \rightarrow_R . We write $G \rightarrow_R^k G'$ if there exists a sequence of DRL-graphs H_i (for $i = 1, \dots, k$) with $H_1 \sim G$, $H_k \sim G'$ and for every $i \in \llbracket 1, k-1 \rrbracket$, $H_i \rightarrow_R H_{i+1}$. By a slight abuse of notation, we also denote by \rightarrow_R^* the least relation containing \rightarrow_R^+ and \sim .

Note that \rightarrow_R^* is the reflexive and transitive closure of \rightarrow_R , if the latter is viewed as a relation on equivalence classes of DRL-graphs w.r.t. \sim .

The next proposition and lemma state that \rightarrow_R is compatible with isomorphism and graph embedding.

PROPOSITION 35. If $G \rightarrow_R G'$ and $G'' \sim G$, then $G'' \rightarrow_R G'$.

PROOF. By definition of the relation \rightarrow_R , there exists a rule $H \rightarrow H' \in R$, a substitution σ and an N -mapping μ of domain $\mathcal{N}_H \cup \mathcal{N}_{H'}$ such that $\sigma(\mu(H)) \leq_C G$, $\sigma(\mu(H'))$ is substitutable for $\sigma(\mu(H))$ in G , and $G' \sim G[\sigma(\mu(H'))/\sigma(\mu(H))]$. Since $G'' \sim G$, there exists an N -mapping μ' such that $G'' = \mu'(G)$. By Definition 32 (3), we have $\mu'(\sigma(\mu(H))) \leq_C \mu'(G)$. By Proposition 21, $\mu'(\sigma(\mu(H)))$ is substitutable for $\mu'(\sigma(\mu(H)))$ in $\mu'(G)$, and

$$\begin{aligned} \mu'(G[\sigma(\mu(H'))/\sigma(\mu(H))]) &= \mu'(G)[\mu'(\sigma(\mu(H')))/\mu'(\sigma(\mu(H)))] \\ &= G''[\mu'(\sigma(\mu(H')))/\mu'(\sigma(\mu(H)))] \end{aligned}$$

Thus $G' \sim G''[\mu'(\sigma(\mu(H')))/\mu'(\sigma(\mu(H)))]$. Since $\mu'(\sigma(\mu(H))) = \sigma(\mu' \circ \mu(H))$ and $\mu'(\sigma(\mu(H'))) = \sigma(\mu' \circ \mu(H'))$, by Definition 34 we deduce that $G'' \rightarrow_R G'$. □

LEMMA 36. Let R be a DRL-graph rewrite system. Let G be a DRL-graph and I be a DRL-subgraph of G , with $I \leq_C G$. If $I \rightarrow_R^k I'$ for some $k \geq 0$ and I' is substitutable for I in G , then $G \rightarrow_R^k G[I'/I]$.

PROOF. The proof is by induction on k . If $k = 0$, then the proof is an immediate consequence of Proposition 20. Assume that $k > 0$. By Definition 33, there exist a rule $H \rightarrow H' \in R$, a substitution σ and an N -mapping μ of domain $\mathcal{N}_H \cup \mathcal{N}_{H'}$ such that $\sigma(\mu(H)) \leq_C I$, $\sigma(\mu(H'))$ is substitutable for $\sigma(\mu(H))$ in I , and $I'' \sim I[\sigma(\mu(H'))/\sigma(\mu(H))]$ with $I'' \rightarrow_R^{k-1} I'$. We assume w.l.o.g. that:

- The N -mapping μ is such that $\mathcal{N}_{\mu(H')} \cap \mathcal{N}_G \subseteq \mathcal{N}_I$, so that the graph $I[\sigma(\mu(H'))/\sigma(\mu(H))]$ is substitutable for I in G (such an N -mapping is guaranteed to exist because $\mathcal{N}_H \subseteq \mathcal{N}_I$);
- $\mathcal{N}_{I''} \cap \mathcal{N}_G = \emptyset$, so that I'' is substitutable for I in G .

By Corollary 24 we have $G[I''/I] \sim G[I[\sigma(\mu(H'))/\sigma(\mu(H))]/I]$, and by Lemma 29,

$$G[\sigma(\mu(H'))/\sigma(\mu(H))] = G[I[\sigma(\mu(H'))/\sigma(\mu(H))]/I].$$

We deduce that $G[I''/I] \sim G[\sigma(\mu(H'))/\sigma(\mu(H))]$.

By Definition 32, \leq_C is transitive and since we have $I \leq_C G$ and $\sigma(\mu(H)) \leq_C I$, we deduce that $\sigma(\mu(H)) \leq_C G$, which proves that $G \rightarrow_R G[I''/I]$. By Definition 32 (2) we have $I'' \leq_C G$ and

by the induction hypothesis, $G[I''/I] \rightarrow_R^{k-1} G[I'/I]$. Therefore, $G \rightarrow_R^k G[I'/I]$. \square

4 GRAPH FORMULAS

In this section we introduce *DRLG-literals*, which are either equations or disequations between graphs and *DRLG-formulas*, that are sets of DRLG-literals.

DEFINITION 37. A DRLG-equation is an unordered pair written $G \approx H$, where G, H are root-compatible DRL-graphs. A DRLG-disequation is the negation of a DRL-graph equation, written $G \not\approx H$. A DRLG-literal can be either one of a DRLG-equation, a DRLG-disequation or \perp (standing for false). The set of variables occurring in a DRLG-literal \mathcal{L} is defined as follows: $\mathcal{V}(G \approx H) = \mathcal{V}(G \not\approx H) = \mathcal{V}(G) \cup \mathcal{V}(H)$ and $\mathcal{V}(\perp) = \emptyset$. All substitutions and \mathcal{N} -mappings μ are extended to DRLG-literals as follows: $\mu(G \bowtie H) = \mu(G) \bowtie \mu(H)$, for $\bowtie \in \{\approx, \not\approx\}$, and $\mu(\perp) = \perp$.

If $\mathcal{L}, \mathcal{L}'$ are DRLG-literals, we write $\mathcal{L} \sim \mathcal{L}'$ iff either $\mathcal{L} = \mathcal{L}'$ or $\mathcal{L} = (G \bowtie H)$, $\mathcal{L}' = (G' \bowtie H')$, $G \sim G'$ and $H \sim H'$.

A DRLG-formula is a set of DRLG-literals.

REMARK 38. Note that disjunctions are not allowed in our syntax. Considering disjunctions of DRLG-literals would actually be straightforward (with the classical semantics), but it would not bring much generality in our context, because nodes could not be shared between the considered DRLG-literals (only variables could be shared). Stronger forms of disjunction could be considered but it is not clear how their semantics would be defined. This is why formulas are defined as conjunctions of literals. Depending on the application, disjunctions can of course be encoded at the object level by specific nodes within the graphs, with appropriate axioms. The graph languages considered in quantum computing (such as the ZX calculus) do not allow for disjunctions.

DEFINITION 39. For any DRLG-formula S , we denote by $I_g(S)$ the set of ground instances of a DRLG-literal in S , i.e., the set of DRLG-literals $\sigma(\mathcal{L})$ where $\mathcal{L} \sim \mathcal{L}'$ for some $\mathcal{L}' \in S$ and σ is a ground substitution of domain $\mathcal{V}(\mathcal{L})$.

We write $\mathcal{L} \rightarrow_R \mathcal{L}'$ if $\mathcal{L} = G \bowtie H$, $\mathcal{L}' = G' \bowtie H$ and $G \rightarrow_R G'$.

5 SEMANTICS

Depending on the considered application, graphs may be associated with various semantics. For instance, in quantum programming, ZX-diagrams are associated with complex matrices. In classical circuits, graphs may be associated with operators with multiple inputs and outputs. In the present paper, for the sake of genericity, we make no assumption on the way DRL-graphs are interpreted, and we only assume that the chosen semantics is compatible with the replacement of DRL-subgraphs. This leads to the notion of a DRLG-congruence, which lifts the usual notion of congruence to DRL-graphs.

DEFINITION 40. A binary relation \bowtie on DRL-graphs is closed under isomorphisms if $G \bowtie G' \wedge H \sim G \wedge H' \sim G' \Rightarrow H \bowtie H'$.

DEFINITION 41. A binary relation \bowtie on DRL-graphs is closed under embeddings if for all DRL-graphs G, H, H' such that $H \leq_C G$, H' is substitutable for H in G and $H \bowtie H'$, we have $G \bowtie G[H'/H]$.

DEFINITION 42. A DRLG-congruence \leftrightarrow is an equivalence relation between ground DRL-graphs that is closed under isomorphisms and embeddings.

DEFINITION 43. A DRLG-congruence \leftrightarrow validates ϕ if:

- ϕ is a ground DRLG-equation $G \approx H$ and $G \leftrightarrow H$;
- ϕ is a ground DRLG-disequation $G \not\approx H$ and $G \not\leftrightarrow H$;
- ϕ is a formula and \leftrightarrow validates all DRLG-literals in $I_g(\phi)$.

If a DRLG-congruence validates ϕ then it is a model of ϕ , and we say that ϕ is satisfiable. We write $\phi \models \psi$ (and say that ψ is a logical consequence of ϕ) if every model of ϕ is a model of ψ , and $\phi \equiv \psi$ if $\phi \models \psi$ and $\psi \models \phi$.

PROPOSITION 44. For any set of DRLG-equations E , there exists a DRLG-congruence \leftrightarrow_E that is the minimal (w.r.t. \subseteq) model of E .

PROOF. It is easy to check that if \mathcal{M} is a set of models of E , then the relation defined as the intersection of all the relations in \mathcal{M} is also a model of E . Further, E has at least one model, containing all pairs of root-compatible ground DRL-graphs. Hence \leftrightarrow_E can be defined as the intersection of all the models of E . \square

We now prove that the satisfiability problem is undecidable for ground DRLG-formulas. The result emphasizes the difference with the usual ground equational logic (on terms) which is well-known to be decidable

THEOREM 45. The satisfiability problem is undecidable for ground DRLG-formulas.

PROOF. The proof goes by a reduction from the halting problem for Turing machines (TM). Let M be a deterministic TM of the form $(Q, \Gamma, b, \Sigma, q_0, F, \delta)$, where Q is the set of states, q_0 is the initial state, $F \subseteq Q$ is the set of final states, b is the blank symbol, Γ is the alphabet, $\Sigma \subseteq \Gamma$ is the set of input symbols and $\delta : (Q \setminus F) \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$ the transition function. We assume, w.l.o.g., that $Q \cap \Gamma = \emptyset$. The configurations are tuples $c = (q, w, w')$, where q is a state and $w, w' \in \Gamma^+$ denotes the part of the tape before and after the head, respectively (we assume that w, w' are nonempty for technical convenience). Any such configuration, with $w = w_1 \dots w_n$, $w' = w'_1 \dots w'_m$ and $w_i, w'_j \in \Gamma$, may be encoded into a graph $G(c)$ defined as follows (where s, h, e are pairwise distinct symbols not occurring in $\Gamma \cup Q$):

$$\begin{aligned} \mathcal{N}_{G(c)} &= \{\alpha_0, \alpha_1, \alpha_2, \alpha_3, \beta_1, \dots, \beta_n, \beta'_1, \dots, \beta'_m\} \\ \mathcal{R}_{G(c)} &= \langle \rangle \\ \mathcal{E}_{G(c)} &= \{(\beta_i \rightarrow \beta_{i+1}) \mid i = 1, \dots, n-1\} \\ &\quad \cup \{(\beta'_j \rightarrow \beta'_{j+1}) \mid j = 1, \dots, m-1\} \\ &\quad \cup \{(\alpha_1 \rightarrow \beta_1), (\beta_n \rightarrow \alpha_2), (\alpha_2 \rightarrow \beta'_1), (\beta'_m \rightarrow \alpha_3)\} \\ l_{G(c)}(\beta_i) &= w_i \quad (\text{for } i = 1, \dots, n) \\ l_{G(c)}(\beta'_j) &= w'_j \quad (\text{for } j = 1, \dots, m) \\ l_{G(c)}(\alpha_0) &= q \quad l_{G(c)}(\alpha_1) = s \quad l_{G(c)}(\alpha_2) = h \quad l_{G(c)}(\alpha_3) = e \end{aligned}$$

The transition function is encoded by the following DRLG-equations:

$$\begin{array}{ccc} \textcircled{q_1} \rightarrow \textcircled{h} \rightarrow \textcircled{i} \rightarrow \textcircled{c} \rightarrow & \approx & \textcircled{q_2} \rightarrow \textcircled{j} \rightarrow \textcircled{h} \rightarrow \textcircled{k} \rightarrow \\ \textcircled{q_1} \rightarrow \textcircled{h} \rightarrow \textcircled{i} \rightarrow \textcircled{e} & \approx & \textcircled{q_2} \rightarrow \textcircled{j} \rightarrow \textcircled{h} \rightarrow \textcircled{b} \rightarrow \textcircled{e} \end{array}$$

for all q_1, i, q_2, j, k such that $\delta(q_1, i) = (q_2, j, R)$ and $k \in \Gamma$. The second rule encodes the fact that the tape is infinite: if the head

is at the end of the word, a new blank symbol must be created to ensure that there are always symbols after the head (i.e., w' must be nonempty in any configuration (q, w, w')). Similar rules are defined for the left movement (for all q_1, i, q_2, j, k such that $\delta(q_1, i) = (q_2, j, L)$ and $k, l \in \Gamma$):

$$\begin{array}{ccc} \textcircled{q_1} \rightarrow \textcircled{l} \rightarrow \textcircled{k} \rightarrow \textcircled{h} \rightarrow \textcircled{i} \rightarrow & \approx & \textcircled{q_2} \rightarrow \textcircled{l} \rightarrow \textcircled{h} \rightarrow \textcircled{k} \rightarrow \textcircled{j} \rightarrow \\ \textcircled{q_1} \textcircled{s} \rightarrow \textcircled{b} \rightarrow \textcircled{h} \rightarrow \textcircled{i} \rightarrow & \approx & \textcircled{q_2} \textcircled{s} \rightarrow \textcircled{b} \rightarrow \textcircled{h} \rightarrow \textcircled{d} \rightarrow \textcircled{j} \rightarrow \end{array}$$

We also define rules to delete useless blank symbols at the end or at the beginning of the tape:

$$\begin{array}{ccc} \textcircled{s} \rightarrow \textcircled{b} \rightarrow \textcircled{b} \rightarrow & \approx & \textcircled{s} \rightarrow \textcircled{b} \rightarrow \\ \rightarrow \textcircled{b} \rightarrow \textcircled{b} \rightarrow \textcircled{e} & \approx & \rightarrow \textcircled{b} \rightarrow \textcircled{e} \end{array}$$

Let E be the set of such DRLG-equations. It is easy to check that, for all configurations c, c' , $G(c) \leftrightarrow_E G(c')$ if either c is reachable from c' or c' is reachable from c . We assume, w.l.o.g., that the considered TM only ends in a configuration (q_f, b, b) (i.e., the TM clears the tape and ends in a single fixed final state q_f). Then, the TM terminates on the empty word iff E , enriched by the following DRLG-disequation, is unsatisfiable:

$$\textcircled{q_0} \textcircled{s} \rightarrow \textcircled{b} \rightarrow \textcircled{h} \rightarrow \textcircled{b} \rightarrow \textcircled{e} \quad \neq \quad \textcircled{q_f} \textcircled{s} \rightarrow \textcircled{b} \rightarrow \textcircled{h} \rightarrow \textcircled{b} \rightarrow \textcircled{e}$$

□

6 A PROOF PROCEDURE

6.1 Inference Rules

We define a set of inference rules that can be viewed as an adaptation of the Superposition calculus to graph formulas. Similarly to the Superposition calculus, this set of inference rules is designed to saturate a set of DRLG-literals in order to derive \perp when the set is unsatisfiable.

DEFINITION 46. A binary relation \bowtie on DRL-graphs is closed under substitutions if for all DRL-graphs G, H such that $G \bowtie H$, and for every substitution σ , we have $\sigma(G) \bowtie \sigma(H)$.

DEFINITION 47. A graph reduction order \geq is a preorder on DRL-graphs that satisfies the following properties:

- The associated order $>$, defined as $G > H \iff (G \geq H \wedge H \not\geq G)$, is well-founded.
- \geq and $>$ are closed under embeddings, substitutions and isomorphisms.
- For all ground DRL-graphs G, H , we have either $G \geq H$ or $H \geq G$.

We write $G \simeq H$ for $G \geq H \wedge G \leq H$.

EXAMPLE 48. For instance the relation $G \geq H \iff |\mathcal{N}_G| \geq |\mathcal{N}_H|$ is a graph reduction order.

Let \leq be a graph reduction order. We consider the following 3 rules, which apply modulo a renaming of nodes and variables.

Positive Unit Superposition (S^+).

$$\frac{G \approx G' \quad H \approx H'}{I[\sigma(G')/\sigma(G)] \approx I[\sigma(H')/\sigma(H)]}$$

If $\mathcal{V}(G \approx G') \cap \mathcal{V}(H \approx H') = \emptyset$, $\mathcal{N}_G \cap \mathcal{N}_H \neq \emptyset$, I is a merge of G and H with $\text{mgu } \sigma$, $\sigma(G) \leq_C \sigma(I)$, $\sigma(G') \leq_C \sigma(I)$, G' and H' are

respectively substitutable for G and H in I , $I[\sigma(G')/\sigma(G)] \not\approx I$ and $I[\sigma(H')/\sigma(H)] \not\approx I$.

Negative Unit Superposition (S^-).

$$\frac{G \not\approx G' \quad H \approx H'}{\sigma(G)[\sigma(H')/\sigma(H)] \not\approx \sigma(G')}$$

If $\mathcal{V}(G \not\approx G') \cap \mathcal{V}(H \approx H') = \emptyset$, σ is a most general substitution such that $\sigma(H) \leq_C \sigma(G)$, H' is substitutable for H in G , $\sigma(G') \not\approx \sigma(G)$ and $\sigma(H') \not\approx \sigma(H)$.

Reflection (R). Two DRL-graphs G and H are *label-unifiable* iff there exists a substitution σ such that $\sigma(G) = \sigma(H)$.

$$\frac{G \not\approx H}{\perp} \quad \text{If } G \text{ and } H \text{ are label-unifiable.}$$

For every DRLG-formula S , we denote by $\text{Sup}(S)$ the set of the DRLG-literals deducible from S by one of the above rules (in one step). Let $\text{Sup}^*(S) \stackrel{\text{def}}{=} \bigcup_{i=0}^{\infty} \text{Sup}^i(S)$, with:

$$\text{Sup}^i(S) \stackrel{\text{def}}{=} \begin{cases} S & \text{if } i = 0 \\ S \cup \text{Sup}(S) & \text{if } i = 1 \\ \text{Sup}^1(\text{Sup}^{i-1}(S)) & \text{otherwise} \end{cases}$$

We show that all the inference rules are sound, in the sense that every deduced DRLG-literal is a logical consequence of the premises.

THEOREM 49 (SOUNDNESS). *The rules S^+ , S^- and R are sound, i.e., for every DRLG-formula S , $S \models \text{Sup}(S)$. Thus if $\perp \in \text{Sup}^*(S)$ then S is unsatisfiable.*

PROOF. The proof goes by an easy inspection of each rule, using the fact that DRLG-congruences are closed under embeddings and closed under substitutions. We only detail the proof for the rule S^+ , the other cases are similar. Let \leftrightarrow be a DRLG-congruence satisfying $G \approx G'$ and $H \approx H'$. Let $\mathcal{L} = I[\sigma(G')/\sigma(G)] \approx I[\sigma(H')/\sigma(H)]$ be a DRLG-literal deduced from $G \approx G'$ and $H \approx H'$ and S^+ and let θ be a ground substitution of domain $\mathcal{V}(\mathcal{L})$. We have to prove that \leftrightarrow is a model of $\mathcal{L}\theta$, i.e., that $\theta(I[\sigma(G')/\sigma(G)]) \leftrightarrow \theta(I[\sigma(H')/\sigma(H)])$. Consider any ground substitution θ' of domain $\mathcal{V}(G) \cup \mathcal{V}(G') \cup \mathcal{V}(H) \cup \mathcal{V}(H')$ and coinciding with $\sigma \circ \theta$ on all variables in $\mathcal{V}(\mathcal{L})$. Since \leftrightarrow is a model of $G \approx G'$ and $H \approx H'$, we have, by Definition 43: $\theta'(G) \leftrightarrow \theta'(G')$ and $\theta'(H) \leftrightarrow \theta'(H')$. Since I is a merge of G and H , $G \leq_C I$ and $H \leq_C I$. By Definition 32 (6), we deduce that $\theta'(G) \leq_C \theta'(I)$ and $\theta'(H) \leq_C \theta'(I)$. By Proposition 22, we have $\theta'(\sigma(I)[\sigma(G')/\sigma(G)]) = \theta'(I)[\theta'(G')/\theta'(G)]$ and $\theta'(\sigma(I)[\sigma(H')/\sigma(H)]) = \theta'(I)[\theta'(H')/\theta'(H)]$. Since \leftrightarrow is closed under embeddings and since $\theta'(G) \leftrightarrow \theta'(G')$, we deduce that $\theta'(I)[\theta'(G')/\theta'(G)] \leftrightarrow \theta'(I)$. Similarly, $\theta'(I)[\theta'(H')/\theta'(H)] \leftrightarrow \theta'(I)$. Therefore $\theta'(\sigma(I)[\sigma(G')/\sigma(G)]) \leftrightarrow \theta'(\sigma(I)[\sigma(H')/\sigma(H)])$, which entails the result. □

6.2 Redundancy

We now define a notion of redundancy, which is useful to dismiss useless DRLG-literals. It is similar to the one of [1] but more restrictive (in the sense that fewer formulas may be deleted). As explained before, this is due to the fact that total reduction orders are difficult to define for DRL-graphs; in fact, the existence of such orders is an open problem.

DEFINITION 50. For every DRLG-formula S we denote by $R_S^>$ (resp. R_S) the set of rules $G \rightarrow H$ where $G \approx H \in I_g(S)$ and $G > H$ (resp. $G \geq H$).

PROPOSITION 51. If $G \xrightarrow{R_S^*} H$ then $G \geq H$, and if $G \xrightarrow{R_S^+} H$ then $G > H$.

DEFINITION 52. A ground DRLG-literal \mathcal{L} is redundant w.r.t. a DRLG-formula S if one of the following conditions hold.

- (1) $\perp \in S$
- (2) \mathcal{L} is of the form $G \approx H$ where $G \sim H$.
- (3) There exist a DRLG-literal $\mathcal{L}' \in S$ and a substitution σ such that $\sigma(\mathcal{L}') \sim \mathcal{L}$.
- (4) There exists a DRLG-literal \mathcal{L}' such that $\mathcal{L} \xrightarrow{R_S^>} \mathcal{L}'$ and \mathcal{L}' is redundant w.r.t. S .

DEFINITION 53. A DRLG-formula S is saturated if all DRLG-literals in $I_g(\text{Sup}(S))$ are redundant w.r.t. S .

7 COMPLETENESS

In this section we prove the completeness of our calculus, in other words, that if a saturated DRLG-formula is unsatisfiable, then it necessarily contains \perp . Throughout this section, we assume that S denotes a fixed DRLG-formula.

Our first goal is to show that if S is saturated and contains no contradiction then the relation \rightarrow_{R_S} is confluent (Lemma 57). This result will allow us to use this relation as a basis for defining a DRLG-congruence satisfying S (Definition 58). Note that showing local confluence is not sufficient since \rightarrow_{R_S} is not necessarily well-founded (this is an essential difference with the usual completeness proofs for standard Superposition calculi, for which total reduction orders exist). We consider a restriction of $\xrightarrow{R_S^*}$ obtained by bounding the number of steps for which the rewritten graph is greater or equal to some given DRL-graph G .

DEFINITION 54. Let G be a DRL-graph. We write $H \xrightarrow{n}_{|G} H'$ iff there exists a sequence I_i ($i = 1, \dots, m$) with $I_1 = H$, $I_m = H'$, $I_i \xrightarrow{R_S} I_{i+1}$ for all $i = 1, \dots, m-1$ and there are at most n indices i in $\llbracket 1, m-1 \rrbracket$ such that $I_{i+1} \not\leq G$.

PROPOSITION 55. We have the following properties:

- (1) If $H \xrightarrow{R_S^*} I$ then $H \xrightarrow{n}_{|G} I$, for some $n \in \mathbb{N}$.
- (2) If $H \xrightarrow{n}_{|G} I$ then $H \xrightarrow{m}_{|G} I$ for all $m \geq n$.
- (3) If $H < G$ and $H \xrightarrow{R_S^*} I$ then $H \xrightarrow{0}_{|G} I$.
- (4) If $H \leq G$ and $H \xrightarrow{R_S^*} I$ then $H \xrightarrow{0}_{|G} I$.
- (5) If $H \xrightarrow{n}_{|G} H'$ and $H' \xrightarrow{m}_{|G} I$ then $H \xrightarrow{n+m}_{|G} I$.

PROOF. The results follow immediately from the definition of $\xrightarrow{n}_{|G}$ and Proposition 51. \square

The following lemma states a form of local confluence:

LEMMA 56. Assume that S is saturated and $\perp \notin S$. Let G, G_1, G_2 be ground DRL-graphs such that $G \xrightarrow{R_S} G_i$ (for $i = 1, 2$). There exists a DRL-graph G' and numbers $\eta_i \in \{0, 1\}$ such that $G_i \xrightarrow{\eta_i}_{|G} G'$, for all $i = 1, 2$. Furthermore, if $G > G_i$ for some $i = 1, 2$ then $\eta_1 = \eta_2 = 0$.

PROOF. By definition of \rightarrow_{R_S} there exist rules $H_i \rightarrow H'_i$ in R_S , substitutions σ_i and \mathcal{N} -mappings μ_i such that, for $i = 1, 2$, $\sigma_i(\mu_i(H_i)) \leq_C G$, $\sigma_i(\mu_i(H'_i))$ is substitutable for $\sigma_i(\mu_i(H_i))$ in G , and $G_i \sim G[\sigma(\mu_i(H'_i))/\sigma(\mu_i(H_i))]$.

Let $i \in \{1, 2\}$. Since R_S is ground we have $\sigma_i = id$, and since $I_g(S)$ (hence R_S) is closed under isomorphisms, we may assume that $\mu_i = id$, so that $G_i = G[H'_i/H_i]$. Furthermore, by definition of R_S , we have $H_i \geq H'_i$ and since \geq is closed under embeddings, $G \geq G_i$.

We distinguish two cases.

- Assume that $\mathcal{N}_{H_1} \cap \mathcal{N}_{H_2} = \emptyset$. By Lemmas 27 and 28, for every $i, j \in \{1, 2\}$ with $i \neq j$, we have $H_i \leq G_j$, H'_i is substitutable for H_i in G_j and:

$$G_2[H'_1/H_1] = G_1[H'_2/H_2]. \quad (1)$$

By Definition 32 (4), we have $H_i \leq_C G_j$, thus $G_i \rightarrow_R G_i[H'_j/H_j]$, so that $G_i \xrightarrow{\eta_i}_{|G} G_i[H'_j/H_j]$, where:

$$\eta_i = \begin{cases} 1 & \text{if } G_i[H'_j/H_j] \approx G \\ 0 & \text{otherwise} \end{cases}$$

Note that by Equation 1 we necessarily have $\eta_1 = \eta_2$. Since $H_j \geq H'_j$ and \geq is closed under embeddings, we have $G_i \geq G_i[H'_j/H_j]$. If $G > G_i$, for some $i = 1, 2$, then $G > G_i[H'_j/H_j]$, thus by definition $\eta_i = 0$. By Equation 1, we obtain the stated result.

- Assume that $\mathcal{N}_{H_1} \cap \mathcal{N}_{H_2} \neq \emptyset$. By definition of R_S , there exist DRLG-literals $I_i \approx I'_i$ in S , groundsubstitutions θ_i and \mathcal{N} -mappings ν_i (for $i = 1, 2$) such that $H_i = \theta_i(\nu_i(I_i))$ and $H'_i = \theta_i(\nu_i(I'_i))$. We may assume that $I_1 \approx I'_1$ and $I_2 \approx I'_2$ are variable-disjoint, so that θ_1 and θ_2 have disjoint domains. By Definition 32 (5), $\nu_1(I_1)$ and $\nu_2(I_2)$ admit a merge I , with mgu θ , such that $(\theta_1 \cup \theta_2) = \theta' \circ \theta$, for some substitution θ' , and $\theta'(I) \leq_C G$. Since $I_g(S)$ is closed by node renaming, we may also assume, w.l.o.g., that $\nu_i(I'_i) \cap \mathcal{N}_G \subseteq \mathcal{N}_{\nu_i(I_i)}$, so that $\nu_i(I'_i)$ is substitutable for $\nu_i(I_i)$ in I and also that $I[\nu_i(I'_i)/\nu_i(I_i)]$ is substitutable for I in G . Let $i \in \{1, 2\}$ and assume that $I[\theta(\nu_i(I'_i))/\theta(\nu_i(I_i))] > I$. Then, since the order $>$ is closed under substitutions, we have $\theta'(I[\theta(\nu_i(I'_i))/\theta(\nu_i(I_i))]) > \theta'(I)$, and therefore $\theta'(I)[\theta_i(\nu_i(I'_i))/\theta_i(\nu_i(I_i))] > \theta'(I)$. We get

$$G[\theta'(I)[\theta_i(\nu_i(I'_i))/\theta_i(\nu_i(I_i))]/\theta'(I)] > G$$

because \geq is closed under embeddings, thus, by Lemma 29,

$$G[\theta_i(\nu_i(I'_i))/\theta_i(\nu_i(I_i))] > G.$$

Therefore, $G[H'_i/H_i] > G$, which contradicts the fact that $H_i \geq H'_i$.

Consequently, we have $I[\theta(\nu_i(I'_i))/\theta(\nu_i(I_i))] \not> I$ (for $i = 1, 2$). This entails that the rule S^+ is applicable on $I_1 \approx I'_1$ and $I_2 \approx I'_2$ (up to the node renamings ν_1, ν_2), yielding the DRLG-literal:

$$I[\theta(\nu_1(I'_1))/\theta(\nu_1(I_1))] \approx I[\theta(\nu_2(I'_2))/\theta(\nu_2(I_2))].$$

Since S is saturated and $\perp \notin S$, there exist DRL-graphs J_i (for $i = 1, 2$) such that either $J_1 = J_2$ or $J_1 \approx J_2 \in I_g(S)$, and $\theta'(I[\theta(\nu_i(I'_i))/\theta(\nu_i(I_i))]) \xrightarrow{R_S^*} J_i$. We assume that $J_1 \approx J_2 \in I_g(S)$, the proof if $J_1 = J_2$ is similar and simpler. Since

\geq is total on ground DRL-graphs, we have either $J_1 \geq J_2$ or $J_2 \geq J_1$. Assume by symmetry that $J_1 \geq J_2$, so that R_S contains a rule $J_1 \rightarrow J_2$. Using Lemmas 25 and 36, we get $G[\theta'(I[\theta(v_i(I_i'))/\theta(v_i(I_i))])/\theta'(I)] \xrightarrow{*}_{R_S} G[J_i/\theta'(I)]$, i.e., (by Lemma 29) $G_i \xrightarrow{*}_{R_S} G[J_i/\theta'(I)]$. We deduce that $G_2 \xrightarrow{0}_{|G} G[J_2/\theta'(I)]$, and (due to the rule $J_1 \rightarrow J_2$), $G_1 \xrightarrow{\eta}_{|G} G[J_2/\theta'(I)]$, where

$$\eta = \begin{cases} 0 & \text{if } G[J_2/\theta'(I)] < G \\ 1 & \text{otherwise} \end{cases}$$

Note that if $G_i < G$, for some $i = 1, 2$ then necessarily $G[J_2/\theta'(I)] < G$ (as $G_i \leq G$), hence $\eta = 0$. Thus we get the result, with $G' = G[J_2/\theta'(I)]$, $\eta_1 = \eta$ and $\eta_2 = 0$. \square

We write $G_1 \downarrow_{R_S} G_2$ if there exists a DRL-graph H such that $G_i \xrightarrow{*}_{R_S} H$, for all $i = 1, 2$.

LEMMA 57. *If S is saturated and $\perp \notin S$, then the relation \rightarrow_{R_S} is confluent.*

PROOF. Assume that there exist DRL-graphs G, G_1, G_2 such that $G \xrightarrow{*}_{R_S} G_i$ for $i = 1, 2$ and $G_1 \not\downarrow_{R_S} G_2$. W.l.o.g., we assume that G is a minimal (w.r.t. $<$) DRL-graph such that DRL-graphs G_1, G_2 satisfying the above properties exist. Thus, for every DRL-graph G' such that $G' < G$, if $G' \xrightarrow{*}_{R_S} G'_i$ for $i = 1, 2$ then $G'_1 \downarrow_{R_S} G'_2$. By definition of the relation $\xrightarrow{n}_{|G}$, there exist natural numbers n_i (for $i = 1, 2$) such that $G \xrightarrow{n_i}_{|G} G_i$. We prove, by induction on the set (n_1, n_2) , that for every DRL-graph $H \simeq G$, if $H \xrightarrow{n_i}_{|G} G_i$ for $i = 1, 2$ then there exists a DRL-graph G' such that for $i = 1, 2$, we have $G_i \xrightarrow{n_{3-i}}_{|G} G'$. Note that this immediately entails the required result, by taking $H = G$. First assume that $H \in \{G_1, G_2\}$, say, $H = G_1$. Then by taking $G' \stackrel{\text{def}}{=} G_2$ we have $G_1 \xrightarrow{n_2}_{|G} G_2$, and by Proposition 55 (2), $G_2 \xrightarrow{n_1}_{|G} G_2$, hence the result. We now assume that $H \rightarrow_{R_S} H_i \xrightarrow{*}_{R_S} G_i$. Note that we have $H_i \xrightarrow{m_i}_{|G} G_i$ where by Proposition 55, either $H_i \simeq H \simeq G$, $n_i > 0$ because $H_i \not\leq G$ and $m_i = n_i - 1$; or $H_i < G$ and $n_i = m_i = 0$. By Lemma 56, there exists I such that: $H_i \xrightarrow{\eta_i}_{|G} I$, with $\eta_1, \eta_2 \in \{0, 1\}$ and if $H_i < G$ for some $i = 1, 2$ then $\eta_1 = \eta_2 = 0$.

Let $i \in \{1, 2\}$, we have $H_i \xrightarrow{\eta_i}_{|G} I$ and $H_i \xrightarrow{m_i}_{|G} G_i$.

- If $H_i < G$, then by minimality of G there exists a DRL-graph I_i such that $I \xrightarrow{*}_{R_S} I_i$ and $G_i \xrightarrow{*}_{R_S} I_i$. By Proposition 51 we have $I, G_i \leq H_i < G$, hence $I \xrightarrow{0}_{|G} I_i$ and $G_i \xrightarrow{0}_{|G} I_i$ by Proposition 55 (3), and $I \xrightarrow{m_i}_{|G} I_i$ and $G_i \xrightarrow{\eta_i}_{|G} I_i$ by Proposition 55 (2).
- If $H_i \simeq G$, then $m_i = n_i - 1 < n_i$. Moreover, we have either $n_{3-i} \geq 1 \geq \eta_i$ or $n_{3-i} = 0$, and in the latter case $H_{3-i} < G$, so that $\eta_i = 0$ by Lemma 56. In both cases, the pair (m_i, η_i) is strictly smaller than the pair (n_1, n_2) . Consequently, by the induction hypothesis, there exists a DRL-graph I_i such that $I \xrightarrow{m_i}_{|G} I_i$ and $G_i \xrightarrow{\eta_i}_{|G} I_i$.

Thus in both cases we get that for $i = 1, 2$, $I \xrightarrow{m_i}_{|G} I_i$ and $G_i \xrightarrow{\eta_i}_{|G} I_i$, for some graphs I_i . We again distinguish two cases.

- If $I < G$, then by minimality of G there exists a DRL-graph G' such that for $i = 1, 2$, $I_i \xrightarrow{*}_{R_S} G'$, hence $I_i \xrightarrow{0}_{|G} G'$ by Proposition 55 (3). Since $G_i \xrightarrow{\eta_i}_{|G} I_i$, we deduce that $G_i \xrightarrow{\eta_i}_{|G} G'$ by Proposition 55 (5). If $\eta_1 = \eta_2 = 0$ then this entails that $G_i \xrightarrow{n_{3-i}}_{|G} G'$ and the proof is completed. Otherwise, by definition of η_i , we have $H_1 \simeq H_2 \simeq G$, hence $n_1, n_2 > 0$ and $n_1, n_2 \geq \eta_1, \eta_2$, thus we also have $G_i \xrightarrow{n_{3-i}}_{|G} G'$.
- If $I \simeq G$, then necessarily $H_i \simeq G$ for all $i = 1, 2$, thus $m_i = n_i - 1$ and, by applying again the induction hypothesis, there exists G' such that $I_i \xrightarrow{m_{3-i}}_{|G} G'$, hence by Proposition 55 (5) $G_i \xrightarrow{m_{3-i} + \eta_i}_{|G} G'$. Since $\eta_i \leq 1$, we deduce that $G_i \xrightarrow{n_{3-i}}_{|G} G'$, and the proof is completed. \square

DEFINITION 58. *We denote by \leftrightarrow_S the relation defined as follows: $G \leftrightarrow_S H$ iff there exists a DRL-graph I such that $G \xrightarrow{*}_{R_S} I$ and $H \xrightarrow{*}_{R_S} I$*

LEMMA 59. *If S is saturated and does not contain \perp then \leftrightarrow_S is a DRLG-congruence.*

PROOF. It is clear that \leftrightarrow_S is reflexive and symmetric. Since $\sim \xrightarrow{0}_{R_S}$ by definition, it is also closed under isomorphisms.

We now show that \leftrightarrow_S is transitive. If $G_1 \leftrightarrow_S G_2 \leftrightarrow_S G_3$ then there exist H_1 and H_2 such that $G_1 \xrightarrow{*}_{R_S} H_1$, $G_2 \xrightarrow{*}_{R_S} H_1$, $G_2 \xrightarrow{*}_{R_S} H_2$, and $G_3 \xrightarrow{*}_{R_S} H_2$. Since $\xrightarrow{*}_{R_S}$ is confluent by Lemma 57, we deduce that there exists a DRL-graph I such that $H_1 \xrightarrow{*}_{R_S} I$, $H_2 \xrightarrow{*}_{R_S} I$. By transitivity of $\xrightarrow{*}_{R_S}$ we get that $G_1 \xrightarrow{*}_{R_S} I$ and $G_3 \xrightarrow{*}_{R_S} I$, hence $G_1 \leftrightarrow_S G_3$.

There remains to prove that \leftrightarrow_S is closed under embeddings. Consider G, H, H_1, H_2 where $H \leq_C G$, $H_1 \leftrightarrow_S H_2$ and H_1, H_2 are substitutable for H in G . By definition of \leftrightarrow_S there exists a DRL-graph I such that $H_i \xrightarrow{*}_{R_S} I$, for $i = 1, 2$. W.l.o.g. we assume that I is disjoint from G and H_i , hence substitutable for H_i in $G[H_i/H]$. Let $i = 1, 2$, by Lemma 36 we have $G[H_i/H] \xrightarrow{*}_{R_S} G[H_i/H][I/H_i]$ and by Lemma 25 we deduce that $G[H_i/H] \xrightarrow{*}_{R_S} G[I/H]$. Therefore, $G[H_1/H] \leftrightarrow_S G[H_2/H]$. \square

LEMMA 60. *If S is saturated and does not contain \perp then \leftrightarrow_S is a model of S .*

PROOF. Let $G \approx G'$ be a DRLG-equation in S and let θ be a ground substitution of the variables in G, G' . By definition, we have $\theta(G) \approx \theta(G') \in I_g(S)$. Since \geq is total on ground DRL-graphs, either $\theta(G) \geq \theta(G')$ or $\theta(G') \geq \theta(G)$. We assume by symmetry that $\theta(G) \geq \theta(G')$, so that R_S contains a rule $\theta(G) \rightarrow \theta(G')$. Then by definition of \leftrightarrow_S , we have $\theta(G) \leftrightarrow_S \theta(G')$, hence \leftrightarrow_S is a model of $G \approx G'$.

Now consider a ground DRLG-disequation $G \not\approx G'$ that is redundant w.r.t. S ; we prove that $G \not\leftrightarrow_S G'$. This is sufficient to obtain the result, since all the ground instances of a DRLG-disequation occurring in S are redundant w.r.t. S . Assume for a contradiction that $G \leftrightarrow_S G'$, so that there exists H such that $G \xrightarrow{n}_{R_S} H$ and $G' \xrightarrow{n'}_{R_S} H$, for some $n, n' \in \mathbb{N}$. The proof is by induction on the pair $(\{G, G'\}, \{n, n'\})$, ordered by the lexicographic and multiset

extension of the ordering $<$ and of the usual order on natural numbers.

- If there exists I such that $G \rightarrow_{R_S} I$ and $I \neq G'$ is redundant w.r.t. S then by Proposition 51, $I < G$, thus by the induction hypothesis $I \not\leftrightarrow_S G'$. By definition of \leftrightarrow_S we have $G \leftrightarrow_S I$, hence $G \not\leftrightarrow_S G'$.
- The proof is similar if there exists I such that $G' \rightarrow_{R_S} I$ and $G \neq I$ is redundant w.r.t. S .
- Otherwise, $G \neq G'$ must be an instance of a DRLG-literal in S , i.e., there exist a DRLG-disequation $H \neq H'$ occurring in S (up to a renaming of nodes) and a substitution σ such that $G = \sigma(H)$ and $G' = \sigma(H')$. We distinguish two cases.
 - If $n = n' = 0$ then $\sigma(H) \sim \sigma(H')$, thus H and H' are label-unifiable (up to a renaming of nodes). Consequently, the rule R applies. Since S is saturated, this entails that $\perp \in S$, which contradicts the hypotheses of the lemma.
 - Otherwise, we have either $n > 0$ or $n' > 0$. Assume that $G > G'$, $n = 0$ and $n' > 0$. Then we have $G \sim H$, thus $G' \rightarrow_{R_S} G$ and $G' \geq G$ by Proposition 51, a contradiction. The case where $G' > G$, with $n > 0$ and $n' = 0$ is symmetric. Now assume that $n > 0$ and that $G \geq G'$ (the proof where $n' > 0$ and $G' \geq G$ is symmetric). Then there exists a DRLG-equation $J \approx J'$ in S (up to a renaming of nodes) and a substitution θ such that $\theta(J) \leq_C G$, $G \rightarrow_{R_S} G[\theta(J')/\theta(J)] \rightarrow_{R_S}^{n-1} I$ and $\theta(J) \geq \theta(J')$. We assume, w.l.o.g., that $J \approx J'$ and $H \neq H'$ share no variable, so that σ and θ have disjoint domains. Let γ be a most general substitution such that $\gamma(J) \leq_C \gamma(H)$. By definition, there exists γ' such that $(\sigma \cup \theta) = \gamma' \circ \gamma$. Since $\theta(J) \geq \theta(J')$, and \geq is closed under substitutions, we have $\gamma(J') \not\leq \gamma(G')$ (as otherwise we would have $\gamma'(\gamma(J')) > \gamma'(\gamma(G'))$, i.e., $\theta(J') > \theta(J)$). Similarly, since $G \geq G'$ and \geq is closed under substitutions, we deduce that $\gamma(H') \not\leq \gamma(H)$. This entails that the rule S^- is applicable on $H \neq H'$ and $J \approx J'$, yielding: $\mathcal{L} : (\gamma(H)[\gamma(J')/\gamma(J)] \neq \gamma(H'))$. Then $G[\theta(J')/\theta(J)] \approx G'$ is an instance of \mathcal{L} , and since S is saturated, it must be redundant w.r.t. S . By the induction hypothesis, we deduce that $G[\theta(J')/\theta(J)] \leftrightarrow_S G'$, thus $G \leftrightarrow_S G'$.

□

THEOREM 61 (COMPLETENESS). *Any saturated DRLG-formula not containing \perp is satisfiable.*

PROOF. This follows immediately from Lemma 60. □

8 AN APPLICATION TO CIRCUITS

8.1 A Class of DRL-Graphs

In this section we consider a class of graphs, denoted by **CIRCUITS**, which intuitively consists of graphs that represent gates. Informally, a gate with n entries and m outputs is represented in this class by a graph with $2(n + m) + 1$ nodes:

- one node that contains the information about the number of entries and outputs;
- n nodes that represent the entries and m nodes that represent the outputs;

- $n + m$ root nodes that are used for rewriting operations.

We also define the notion of a *subcircuit* and show that it is a C -relation, thus guaranteeing the completeness of the proof procedure defined in Section 6.

DEFINITION 62. *We denote by **CIRCUITS** the class of graphs G satisfying the following properties:*

- (1) *The set of labels is $\mathbb{N} \setminus \{0\}$.*
- (2) *There are sorts gate_n^m , into and from denoting gates with n entries and m outputs, input ports, and output ports respectively.*
- (3) *For each node α of sort gate_n^m there exist exactly n nodes $\alpha_1, \dots, \alpha_n$ of sort into labeled with $1, \dots, n$ respectively and m nodes $\alpha'_1, \dots, \alpha'_m$ of sort from labeled with $1, \dots, m$ respectively, and edges $(\alpha_i \rightarrow \alpha)$ and $(\alpha \rightarrow \alpha'_j)$ for every $i \in \llbracket 1, n \rrbracket$ and $j \in \llbracket 1, m \rrbracket$.*
- (4) *All nodes of sort into have exactly one incoming and at most one outgoing edge; all nodes of sort from have exactly one outgoing and at most one incoming edge.*
- (5) *Every node of sort into which is labeled by $i \in \mathbb{N} \setminus \{0\}$ admits one outgoing edge, the target of which is a node of sort gate_n^m , where $i \leq n$.*
- (6) *Every node of sort from which is labeled by $i \in \mathbb{N} \setminus \{0\}$ admits one incoming edge, the source of which is a node of sort gate_n^m , where $i \leq m$.*
- (7) *All the other edges are of the form $(\alpha \rightarrow \beta)$ where α is of sort from and β is of sort into.*
- (8) *\mathcal{R}_G is of the form $R_f \cdot R_i$, where every node in the sequence R_f is of sort from and admits no incoming edge, and every node in the sequence R_i is of sort into and admits no outgoing edge. The sequence R_f is denoted by $\mathcal{R}_G^{\text{in}}$, and R_i by $\mathcal{R}_G^{\text{out}}$.*

Intuitively, $\mathcal{R}_G^{\text{in}}$ denotes the **inputs** of G and $\mathcal{R}_G^{\text{out}}$ its **outputs**.

EXAMPLE 63. *The empty DRL-graph $\emptyset \stackrel{\text{def}}{=} (\emptyset, \langle \rangle, \emptyset, \emptyset)$, is in **CIRCUITS**, with $\mathcal{R}_\emptyset^{\text{in}} = \mathcal{R}_\emptyset^{\text{out}} = \langle \rangle$.*

DEFINITION 64. *Given a node α in $\mathcal{R}_G^{\text{in}}$, we denote by $\text{trg}_G(\alpha)$ (or simply $\text{trg}(\alpha)$) when there is no ambiguity) the node β such that $(\alpha \rightarrow \beta) \in \mathcal{E}_G$. Given a node α in $\mathcal{R}_G^{\text{out}}$, we denote by $\text{src}_G(\alpha)$ (or simply $\text{src}(\alpha)$) when there is no ambiguity) the node β such that $(\beta \rightarrow \alpha) \in \mathcal{E}_G$.*

Since every node α in $\mathcal{R}_G^{\text{in}}$ is of sort from, the node β such that $(\alpha \rightarrow \beta) \in \mathcal{E}_G$ exists and is unique; similarly, if α in $\mathcal{R}_G^{\text{out}}$ then the node β such that $(\beta \rightarrow \alpha) \in \mathcal{E}_G$ exists and is unique.

DEFINITION 65. *H is a subcircuit of a DRL-graph $G \in \mathbf{CIRCUITS}$, denoted by $H \leq_C G$, if and only if $H \leq G$ and $H \in \mathbf{CIRCUITS}$.*

LEMMA 66. *Assume that $H \leq_C G$, let H' be a DRL-graph in **CIRCUITS** that is substitutable for H in G and let $G' \stackrel{\text{def}}{=} G[H'/H]$. Then $G' \in \mathbf{CIRCUITS}$.*

PROOF. Let $\mu \stackrel{\text{def}}{=} \mu_G^{H \rightarrow H'}$ (see Definition 15), we prove that $G' \in \mathbf{CIRCUITS}$. Items (1) and (2) of the Definition 62 are immediate to verify.

Item 3 Let $\alpha \in \mathcal{N}_{G'}$ be a node of sort gate_n^m . Note that α cannot be in $\mathcal{R}_G \cup \mathcal{R}_{H'}$. If $\alpha \in \mathcal{N}_{H'}$, then because $H' \in \mathbf{CIRCUITS}$,

it is guaranteed that there exist exactly n nodes $\alpha_1, \dots, \alpha_n$ of sort into labeled with $1, \dots, n$ respectively and m nodes $\alpha'_1, \dots, \alpha'_m$ of sort from labeled with $1, \dots, m$ respectively, and edges $(\alpha_i \rightarrow \alpha)$ and $(\alpha \rightarrow \alpha'_j)$ in $\mathcal{E}_{H'}$ for all $i \in \llbracket 1, n \rrbracket$ and $j \in \llbracket 1, m \rrbracket$. Since $\mathcal{E}_{H'} \subseteq \mathcal{E}_{G'}$, we have the result.

Otherwise, $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$. Let $i \in \llbracket 1, n \rrbracket$, we prove that there exists a node α'_i of sort into labeled with i such that $(\alpha'_i \rightarrow \alpha) \in \mathcal{E}_{G'}$. The proof for nodes of sort from is similar. Since $G \in \text{CIRCUITS}$, there exists a node $\alpha_i \in \mathcal{N}_G$ of sort into labeled with i such that $(\alpha_i \rightarrow \alpha) \in \mathcal{E}_G$. Necessarily $(\alpha_i \rightarrow \alpha) \in \mathcal{E}_G \setminus \mathcal{E}_H$, and we have the result by taking $\alpha'_i \stackrel{\text{def}}{=} \mu(\alpha_i)$.

Item 4 Consider a node $\alpha \in \mathcal{N}_{G'}$ of sort into, and first assume that $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$, so that $\mu(\alpha) = \alpha$ and $\alpha \notin \mathcal{N}_{H'}$ by Proposition 14. By definition there exists a unique edge $(\alpha' \rightarrow \alpha) \in \mathcal{E}_G \setminus \mathcal{E}_H$, and since there can be no edge of the form $(\alpha'' \rightarrow \alpha)$ in $\mathcal{E}_{H'}$, α admits $\mu((\alpha' \rightarrow \alpha)) = (\mu(\alpha') \rightarrow \alpha)$ as a unique incoming edge. If $\alpha \in \mathcal{N}_{H'}$ then by definition $\mu(\alpha) = \alpha$ and by Proposition 14, $\alpha \notin \mathcal{N}_G \setminus \mathcal{N}_H$ because $\alpha \notin \mathcal{N}_H$. Since $H \in \text{CIRCUITS}$, the proof in this case is straightforward.

The proof that α admits at most one outgoing edge is carried out in a similar way, and so are the proofs when α is of sort from.

Item 5 Consider a node $\alpha \in \mathcal{N}_{G'}$ of sort into and labeled by i . A case analysis depending on whether $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$ or $\alpha \in \mathcal{N}_{H'}$ as in the previous point shows that the target of α is of sort gate_n^m for some $n \geq i$.

Item 6 The proof is the same as in the previous case.

Item 7 Because μ preserves sorts, it is straightforward to prove that all other edges are of the form $(\alpha \rightarrow \beta)$ where α is of sort from and β is of sort into.

Item 8 By definition we have $\mathcal{R}_{G'} = \mu(\mathcal{R}_G)$, and since G and H' are both in CIRCUITS , $\mathcal{R}_{G'}$ is of the required form. \square

LEMMA 67. Assume that $H \leq_C G$, $H' \leq_C G$ and that H and H' are label-compatible. Then the E -merge of H and H' with

$$E \stackrel{\text{def}}{=} \{(\alpha \rightleftharpoons \beta) \mid \alpha \in \mathcal{R}_H \setminus \mathcal{N}_{H'}, \beta \in \mathcal{R}_{H'} \setminus \mathcal{N}_H, (\alpha \rightleftharpoons \beta) \in \mathcal{E}_G\}$$

is in CIRCUITS .

PROOF. Let G' denote the E -merge of H and H' ; by Proposition 31, $G' \leq G$. It is straightforward to verify that $G' \in \text{CIRCUITS}$: the existence of nodes and edges satisfying the requirements of the definition of CIRCUITS are deduced from the fact that H and H' are both in CIRCUITS , and the uniqueness properties are consequences of the fact that H and H' are both subgraphs of G which is in CIRCUITS . \square

LEMMA 68. The relation \leq_C is a C -relation, with $C = \text{CIRCUITS}$.

PROOF. First note that since $H \in \text{CIRCUITS}$, if $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$ is of sort into then there can be no edge $(\alpha_i \rightarrow \alpha) \in \mathcal{E}_H$. Indeed, necessarily, $\alpha_i \in \mathcal{R}_H^{\text{in}}$ and by definition, α_i cannot admit any outgoing edge. Similarly, if $\alpha \in \mathcal{N}_G \setminus \mathcal{N}_H$ is of sort from then there can be no edge $(\alpha \rightarrow \alpha_i) \in \mathcal{E}_H$.

- (1) By definition, if $H \leq_C G$ then $G, H \in \text{CIRCUITS}$.
- (2) By Lemma 19 we have $H' \leq G[H'/H]$ and by Lemma 66 $G[H'/H] \in \text{CIRCUITS}$.
- (3) It is clear that if $G \in \text{CIRCUITS}$ then $\mu(G) \in \text{CIRCUITS}$ for all N -mappings μ . By Proposition 9, we deduce that if $H \leq_C G$ then $\mu(H) \leq_C \mu(G)$.
- (4) By Lemma 27 we have $H \leq G[I'/I]$ and by Lemma 66, $G[I'/I] \in \text{CIRCUITS}$.
- (5) If $H \leq_C G$ and $H' \leq_C G$ then the merge G' of H and H' considered in Lemma 67 is such that $G' \leq_C G$. \square

8.2 Parallel and Sequential Composition

It is natural to construct a circuit by composing simpler components. For instance the diagrams of the ZX calculus [12], formally defined as morphisms in a dagger compact category, are built from a set of generators via composition and monoidal product (modulo equalities induced by the considered structure). We thus introduce two standard composition operations: parallel and sequential composition. The empty graph is the neutral element for parallel composition, and we introduce the class of so-called *identity graphs*, which can be viewed as neutral elements for sequential composition.

DEFINITION 69. Consider two graphs G_1, G_2 in CIRCUITS , such that $\mathcal{N}_{G_1} \cap \mathcal{N}_{G_2} = \emptyset$. We define the parallel composition of G_1 and G_2 , denoted by $G_1 \otimes G_2$, as the graph G_3 constructed as follows.

- (1) $\mathcal{N}_{G_3} \stackrel{\text{def}}{=} \mathcal{N}_{G_1} \cup \mathcal{N}_{G_2}$;
- (2) $\mathcal{R}_{G_3} \stackrel{\text{def}}{=} \mathcal{R}_{G_1}^{\text{in}} \cdot \mathcal{R}_{G_2}^{\text{in}} \cdot \mathcal{R}_{G_1}^{\text{out}} \cdot \mathcal{R}_{G_2}^{\text{out}}$;
- (3) $\mathcal{E}_{G_3} \stackrel{\text{def}}{=} \mathcal{E}_{G_1} \cup \mathcal{E}_{G_2}$;
- (4) $l_{G_3} \stackrel{\text{def}}{=} l_{G_1} \cup l_{G_2}$.

It is straightforward to check that $G_2 \otimes G_1$ is a DRL-graph when G_1 and G_2 are disjoint.

PROPOSITION 70. Parallel composition is an associative operation on mutually disjoint DRL-graphs, and the empty DRL-graph \emptyset is its neutral element.

PROOF. Immediate. \square

DEFINITION 71. Consider two graphs G_1, G_2 in CIRCUITS , such that $\mathcal{N}_{G_1} \cap \mathcal{N}_{G_2} = \emptyset$, where $\mathcal{R}_{G_1}^{\text{out}} = \langle \alpha_1, \dots, \alpha_n \rangle$ and $\mathcal{R}_{G_2}^{\text{in}} = \langle \beta_1, \dots, \beta_n \rangle$.

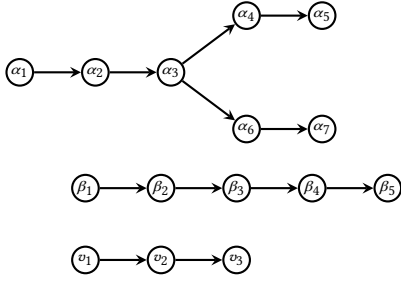
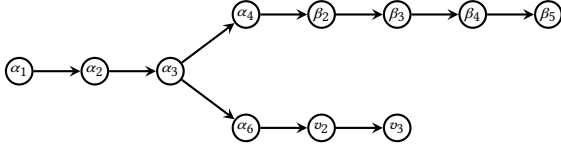
We let $\Upsilon_{G_1}^{G_2}$ denote the mapping on $\mathcal{N}_{G_1} \cup \mathcal{N}_{G_2}$ defined as follows:

- For all $i = 1, \dots, n$, $\Upsilon_{G_1}^{G_2}(\alpha_i) \stackrel{\text{def}}{=} \text{trg}_{G_2}(\beta_i)$ and $\Upsilon_{G_1}^{G_2}(\beta_i) \stackrel{\text{def}}{=} \text{src}_{G_1}(\alpha_i)$;
- For all other nodes $\alpha \in \mathcal{N}_{G_1} \cup \mathcal{N}_{G_2}$, $\Upsilon_{G_1}^{G_2}(\alpha) \stackrel{\text{def}}{=} \alpha$.

We define the sequential composition of G_1 and G_2 , denoted by $G_2 \circ G_1$, as the graph G_3 constructed as follows.

- (1) $\mathcal{N}_{G_3} \stackrel{\text{def}}{=} \Upsilon_{G_1}^{G_2}(\mathcal{N}_{G_1}) \cup \Upsilon_{G_1}^{G_2}(\mathcal{N}_{G_2})$;
- (2) $\mathcal{R}_{G_3} \stackrel{\text{def}}{=} \mathcal{R}_{G_1}^{\text{in}} \cdot \mathcal{R}_{G_2}^{\text{out}}$;
- (3) $\mathcal{E}_{G_3} \stackrel{\text{def}}{=} \Upsilon_{G_1}^{G_2}(\mathcal{E}_{G_1}) \cup \Upsilon_{G_1}^{G_2}(\mathcal{E}_{G_2})$;
- (4) $l_{G_3} \stackrel{\text{def}}{=} l_{G_1} \cup l_{G_2}$.

It is straightforward to check that $G_1 \circ G_2$ is a DRL-graph when G_1 and G_2 satisfy the conditions of Definition 71.

Figure 3: Graphs G_3 , G_1 and G_2 in Example 72Figure 4: Sequential composition of $G_2 \otimes G_1$ and G_3 (Example 72)

EXAMPLE 72. Consider the graph G_3 defined as follows:

$$\begin{aligned} \mathcal{N}_{G_3} &\stackrel{\text{def}}{=} \{\alpha_1, \dots, \alpha_7\} \\ \mathcal{R}_{G_3} &\stackrel{\text{def}}{=} \langle \alpha_1, \alpha_5, \alpha_7 \rangle \\ \mathcal{E}_{G_3} &\stackrel{\text{def}}{=} \{(\alpha_1 \rightarrow \alpha_2), (\alpha_2 \rightarrow \alpha_3), (\alpha_3 \rightarrow \alpha_4), (\alpha_3 \rightarrow \alpha_6)\} \\ &\quad \cup \{(\alpha_4 \rightarrow \alpha_5), (\alpha_6 \rightarrow \alpha_7)\} \\ l_{G_3} &\stackrel{\text{def}}{=} \{(\alpha_2, 1), (\alpha_3, f), (\alpha_4, 1), (\alpha_6, 2)\} \end{aligned}$$

This graph is an element of CIRCUITS , α_3 is of sort gate_1^2 , nodes α_2 , α_5 and α_7 are of sort into and nodes α_1 , α_4 and α_6 are of sort from .

We also consider the graphs G_2 and G_1 , defined as follows:

$$\begin{aligned} \mathcal{N}_{G_2} &\stackrel{\text{def}}{=} \{\beta_1, \dots, \beta_5\} \\ \mathcal{R}_{G_2} &\stackrel{\text{def}}{=} \langle \beta_1, \beta_5 \rangle \\ \mathcal{E}_{G_2} &\stackrel{\text{def}}{=} \{(\beta_1 \rightarrow \beta_2), (\beta_2 \rightarrow \beta_3), (\beta_3 \rightarrow \beta_4), (\beta_4 \rightarrow \beta_5)\} \\ l_{G_2} &\stackrel{\text{def}}{=} \{(\beta_2, 1), (\beta_3, g), (\beta_4, 1)\} \\ \mathcal{N}_{G_1} &\stackrel{\text{def}}{=} \{v_1, v_2, v_3\} \\ \mathcal{R}_{G_1} &\stackrel{\text{def}}{=} \langle v_1 \rangle \\ \mathcal{E}_{G_1} &\stackrel{\text{def}}{=} \{(v_1 \rightarrow v_2), (v_2 \rightarrow v_3)\} \\ l_{G_1} &\stackrel{\text{def}}{=} \{(v_2, 1), (v_3, h)\} \end{aligned}$$

Node β_3 is of sort gate_1^1 , nodes β_1 and β_4 are of sort from ; and nodes β_2 and β_5 is of sort into . Node v_3 is of sort gate_1^0 , node v_1 is of sort from and node v_2 is of sort into . All graphs are depicted in Figure 3, and the graph $(G_2 \otimes G_1) \circ G_3$ is depicted in Figure 4.

It is straightforward to check that by construction we have the following result:

PROPOSITION 73. Assume $G_2 \circ G_1$ is well-defined. We have the following properties:

- $\mathcal{R}_{G_2 \circ G_1}^{\text{in}} = \mathcal{R}_{G_1}^{\text{in}}$ and $\mathcal{R}_{G_2 \circ G_1}^{\text{out}} = \mathcal{R}_{G_2}^{\text{out}}$.

- If $\alpha \in \mathcal{R}_{G_2 \circ G_1}^{\text{in}}$ then $\text{trg}_{G_2 \circ G_1}(\alpha) = \Upsilon_{G_1}^{G_2}(\text{trg}_{G_1}(\alpha))$.
- If $\alpha \in \mathcal{R}_{G_2 \circ G_1}^{\text{out}}$ then $\text{src}_{G_2 \circ G_1}(\alpha) = \Upsilon_{G_1}^{G_2}(\text{src}_{G_2}(\alpha))$.

PROPOSITION 74. If the graphs G_1 , G_2 and G_3 in CIRCUITS are mutually disjoint and such that $G_3 \circ G_2$ and $G_2 \circ G_1$ are well-defined, then the sequential compositions $(G_3 \circ G_2) \circ G_1$ and $G_3 \circ (G_2 \circ G_1)$ are well-defined and equal.

PROOF. We assume

$$\begin{aligned} \mathcal{R}_{G_1}^{\text{out}} &= \langle \alpha_1, \dots, \alpha_n \rangle & \mathcal{R}_{G_2}^{\text{in}} &= \langle \beta_1, \dots, \beta_n \rangle \\ \mathcal{R}_{G_2}^{\text{out}} &= \langle \gamma_1, \dots, \gamma_m \rangle & \mathcal{R}_{G_3}^{\text{in}} &= \langle \delta_1, \dots, \delta_m \rangle \end{aligned}$$

Let $H \stackrel{\text{def}}{=} G_2 \circ G_1$ and $I \stackrel{\text{def}}{=} G_3 \circ G_2$. We begin by proving the following statements.

- (1) If $\alpha \in \mathcal{N}_{G_1}$ then $\Upsilon_H^{G_3} \circ \Upsilon_{G_1}^{G_2}(\alpha) = \Upsilon_{G_1}^I(\alpha)$. First assume that $\alpha \in \mathcal{R}_{G_1}^{\text{out}}$, i.e., that $\alpha = \alpha_i$ for some $i = 1, \dots, n$. Then by construction we have $\Upsilon_H^{G_3} \circ \Upsilon_{G_1}^{G_2}(\alpha) = \Upsilon_H^{G_3}(\text{trg}_{G_2}(\beta_i))$, and by Proposition 73, $\Upsilon_H^{G_3}(\text{trg}_{G_2}(\beta_i)) = \Upsilon_{G_2}^{G_3}(\text{trg}_{G_1}(\alpha))$. Still by Proposition 73 we have $\mathcal{R}_H^{\text{out}} = \mathcal{R}_{G_2}^{\text{out}}$, therefore, $\Upsilon_H^{G_3}(\text{trg}_{G_2}(\beta_i)) = \Upsilon_{G_2}^{G_3}(\text{trg}_{G_2}(\beta_i))$. Otherwise, since the considered graphs are pairwise disjoint and $\mathcal{R}_H^{\text{out}} = \mathcal{R}_{G_2}^{\text{out}}$, we have

$$\Upsilon_H^{G_3} \circ \Upsilon_{G_1}^{G_2}(\alpha) = \Upsilon_H^{G_3}(\alpha) = \alpha = \Upsilon_{G_1}^I(\alpha),$$

hence the result.

- (2) If $\alpha \in \mathcal{N}_{G_2}$ then $\Upsilon_H^{G_3} \circ \Upsilon_{G_1}^{G_2}(\alpha) = \Upsilon_{G_1}^I \circ \Upsilon_{G_2}^{G_3}(\alpha)$. First assume that $\alpha \in \mathcal{R}_{G_2}^{\text{in}}$, i.e., that $\alpha = \beta_i$ for some $i = 1, \dots, n$. Then by Proposition 73 $\mathcal{R}_{G_2}^{\text{in}} = \mathcal{R}_I^{\text{in}}$ and

$$\Upsilon_H^{G_3} \circ \Upsilon_{G_1}^{G_2}(\alpha) = \Upsilon_H^{G_3}(\text{src}_{G_1}(\alpha_i)) = \text{src}_{G_1}(\alpha_i) = \Upsilon_{G_1}^I(\alpha) = \Upsilon_{G_1}^I \circ \Upsilon_{G_2}^{G_3}(\alpha).$$

Now assume that $\alpha \in \mathcal{R}_{G_2}^{\text{out}}$, i.e., that $\alpha = \gamma_i$ for some $i = 1, \dots, m$. Then by Proposition 73 $\mathcal{R}_{G_2}^{\text{out}} = \mathcal{R}_H^{\text{out}}$ and

$$\Upsilon_H^{G_3} \circ \Upsilon_{G_1}^{G_2}(\alpha) = \Upsilon_H^{G_3}(\alpha) = \text{trg}_{G_3}(\delta_i) = \Upsilon_{G_1}^I(\text{trg}_{G_3}(\delta_i)) = \Upsilon_{G_1}^I \circ \Upsilon_{G_2}^{G_3}(\alpha).$$

Otherwise we have

$$\Upsilon_H^{G_3} \circ \Upsilon_{G_1}^{G_2}(\alpha) = \Upsilon_H^{G_3}(\alpha) = \alpha = \Upsilon_{G_1}^I(\alpha) = \Upsilon_{G_1}^I \circ \Upsilon_{G_2}^{G_3}(\alpha),$$

hence the result.

- (3) If $\alpha \in \mathcal{N}_{G_3}$ then $\Upsilon_H^{G_3}(\alpha) = \Upsilon_{G_1}^I \circ \Upsilon_{G_2}^{G_3}(\alpha)$. First assume that $\alpha \in \mathcal{R}_{G_3}^{\text{in}}$, i.e., that $\alpha = \delta_i$ for some $i = 1, \dots, m$. Then by construction we have $\Upsilon_{G_1}^I \circ \Upsilon_{G_2}^{G_3}(\alpha) = \Upsilon_{G_1}^I(\text{src}_{G_2}(\gamma_i))$, and by Proposition 73, $\Upsilon_H^{G_3}(\alpha) = \text{src}_H(\gamma_i) = \Upsilon_{G_1}^{G_2}(\text{src}_{G_2}(\gamma_i))$. Still by Proposition 73 we have $\mathcal{R}_H^{\text{in}} = \mathcal{R}_{G_2}^{\text{in}}$, therefore, $\Upsilon_H^{G_3}(\text{trg}_{G_2}(\beta_i)) = \Upsilon_{G_2}^{G_3}(\text{trg}_{G_2}(\beta_i))$. Otherwise, since the considered graphs are pairwise disjoint and $\mathcal{R}_H^{\text{in}} = \mathcal{R}_{G_2}^{\text{in}}$, we have

$$\Upsilon_H^{G_3}(\alpha) = \alpha = \Upsilon_{G_2}^{G_3}(\alpha) = \Upsilon_{G_1}^I \circ \Upsilon_{G_2}^{G_3}(\alpha),$$

hence the result.

Using the statements above, we deduce that:

$$\begin{aligned}
\mathcal{E}_{I \circ G_1} &= \Upsilon_{G_1}^I(\mathcal{E}_{G_1}) \cup \Upsilon_{G_1}^I(\mathcal{E}_I) \\
&= \Upsilon_{G_1}^I(\mathcal{E}_{G_1}) \cup \Upsilon_{G_1}^I \circ \Upsilon_{G_2}^{G_3}(\mathcal{E}_{G_2}) \cup \Upsilon_{G_1}^I \circ \Upsilon_{G_2}^{G_3}(\mathcal{E}_{G_3}) \\
&= \Upsilon_H^{G_3} \circ \Upsilon_{G_1}^{G_2}(\mathcal{E}_{G_1}) \cup \Upsilon_H^{G_3} \circ \Upsilon_{G_1}^{G_2}(\mathcal{E}_{G_2}) \cup \Upsilon_H^{G_3}(\mathcal{E}_{G_3}) \\
&= \Upsilon_H^{G_3}(\mathcal{E}_H) \cup \Upsilon_H^{G_3}(\mathcal{E}_{G_3}) \\
&= \mathcal{E}_{G_3 \circ H}
\end{aligned}$$

We prove in a similar way that $\mathcal{N}_{I \circ G_1} = \mathcal{N}_{G_3 \circ H}$. Since $\mathcal{R}_{I \circ G_1} = \mathcal{R}_{G_3 \circ H}$ and $l_{I \circ G_1} = l_{G_3 \circ H}$ by construction, we have the result. \square

We have the following commutation rule that shows that parallel and sequential composition can be switched:

LEMMA 75. Consider the pairwise disjoint graphs G_1, G_2, G_3, G_4 in CIRCUITS, and assume $|\mathcal{R}_{G_1}^{\text{in}}| = |\mathcal{R}_{G_2}^{\text{out}}|$ and $|\mathcal{R}_{G_3}^{\text{in}}| = |\mathcal{R}_{G_4}^{\text{out}}|$. Then we have:

$$(G_2 \otimes G_4) \circ (G_1 \otimes G_3) = (G_2 \circ G_1) \otimes (G_4 \circ G_3).$$

PROOF. We define the following graphs:

$$\begin{aligned}
H_1 &\stackrel{\text{def}}{=} G_2 \circ G_1 & H_2 &\stackrel{\text{def}}{=} G_4 \circ G_3 & H_3 &\stackrel{\text{def}}{=} H_1 \otimes H_2 \\
I_1 &\stackrel{\text{def}}{=} G_1 \otimes G_3 & I_2 &\stackrel{\text{def}}{=} G_2 \otimes G_4 & I_3 &\stackrel{\text{def}}{=} I_2 \circ I_1
\end{aligned}$$

and prove that $H_3 = I_3$.

First note that by construction, we have $\mathcal{R}_{I_1}^{\text{in}} = \mathcal{R}_{G_1}^{\text{in}} \cdot \mathcal{R}_{G_3}^{\text{in}}$ and $\mathcal{R}_{I_2}^{\text{out}} = \mathcal{R}_{G_2}^{\text{out}} \cdot \mathcal{R}_{G_4}^{\text{out}}$. Since $|\mathcal{R}_{G_1}^{\text{in}}| = |\mathcal{R}_{G_2}^{\text{out}}|$ and $|\mathcal{R}_{G_3}^{\text{in}}| = |\mathcal{R}_{G_4}^{\text{out}}|$ by hypothesis, I_3 is well-defined. Furthermore, because all graphs are pairwise disjoint, it is straightforward to verify that $\Upsilon_{I_1}^{I_2} = \Upsilon_{G_1}^{G_2} \cup \Upsilon_{G_3}^{G_4}$.

(1)

$$\begin{aligned}
\mathcal{N}_{H_3} &= \mathcal{N}_{H_1} \cup \mathcal{N}_{H_2} \\
&= \Upsilon_{G_1}^{G_2}(\mathcal{N}_{G_1}) \cup \Upsilon_{G_1}^{G_2}(\mathcal{N}_{G_2}) \cup \Upsilon_{G_3}^{G_4}(\mathcal{N}_{G_3}) \cup \Upsilon_{G_3}^{G_4}(\mathcal{N}_{G_4}) \\
&= \Upsilon_{I_1}^{I_2}(\mathcal{N}_{G_1}) \cup \Upsilon_{I_1}^{I_2}(\mathcal{N}_{G_2}) \cup \Upsilon_{I_1}^{I_2}(\mathcal{N}_{G_3}) \cup \Upsilon_{I_1}^{I_2}(\mathcal{N}_{G_4}) \\
&= \Upsilon_{I_1}^{I_2}(\mathcal{N}_{G_1} \cup \mathcal{N}_{G_3}) \cup \Upsilon_{I_1}^{I_2}(\mathcal{N}_{G_2} \cup \mathcal{N}_{G_4}) \\
&= \Upsilon_{I_1}^{I_2}(\mathcal{N}_{I_1}) \cup \Upsilon_{I_1}^{I_2}(\mathcal{N}_{I_2}) \\
&= \mathcal{N}_{I_3}.
\end{aligned}$$

(2)

$$\begin{aligned}
\mathcal{R}_{H_3} &= \mathcal{R}_{H_1}^{\text{in}} \cdot \mathcal{R}_{H_2}^{\text{in}} \cdot \mathcal{R}_{H_1}^{\text{out}} \cdot \mathcal{R}_{H_2}^{\text{out}} \\
&= \mathcal{R}_{G_1}^{\text{in}} \cdot \mathcal{R}_{G_3}^{\text{in}} \cdot \mathcal{R}_{G_2}^{\text{out}} \cdot \mathcal{R}_{G_4}^{\text{out}} \\
&= \mathcal{R}_{I_1}^{\text{in}} \cdot \mathcal{R}_{I_2}^{\text{out}} \\
&= \mathcal{R}_{I_3}.
\end{aligned}$$

(3)

$$\begin{aligned}
\mathcal{E}_{H_3} &= \mathcal{E}_{H_1} \cup \mathcal{E}_{H_2} \\
&= \Upsilon_{G_1}^{G_2}(\mathcal{E}_{G_1}) \cup \Upsilon_{G_1}^{G_2}(\mathcal{E}_{G_2}) \cup \Upsilon_{G_3}^{G_4}(\mathcal{E}_{G_3}) \cup \Upsilon_{G_3}^{G_4}(\mathcal{E}_{G_4}) \\
&= \Upsilon_{I_1}^{I_2}(\mathcal{E}_{G_1}) \cup \Upsilon_{I_1}^{I_2}(\mathcal{E}_{G_2}) \cup \Upsilon_{I_1}^{I_2}(\mathcal{E}_{G_3}) \cup \Upsilon_{I_1}^{I_2}(\mathcal{E}_{G_4}) \\
&= \Upsilon_{I_1}^{I_2}(\mathcal{E}_{G_1} \cup \mathcal{E}_{G_3}) \cup \Upsilon_{I_1}^{I_2}(\mathcal{E}_{G_2} \cup \mathcal{E}_{G_4}) \\
&= \Upsilon_{I_1}^{I_2}(\mathcal{E}_{I_1}) \cup \Upsilon_{I_1}^{I_2}(\mathcal{E}_{I_2}) \\
&= \mathcal{E}_{I_3}.
\end{aligned}$$

(4)

$$\begin{aligned}
l_{H_3} &= l_{H_1} \cup l_{H_2} \\
&= l_{G_1} \cup l_{G_2} \cup l_{G_3} \cup l_{G_4} \\
&= l_{I_1} \cup l_{I_2} \\
&= l_{I_3}.
\end{aligned}$$

\square

We now define the class of *identity graphs*. Intuitively, these graphs can be viewed as parallel compositions of wires. Their sequential composition with a graph leaves the latter unchanged, up to a renaming of nodes.

DEFINITION 76. A graph G is an id-graph if it is of the following form:

- $\mathcal{N}_G = \{\alpha_1, \alpha_2\}$, $\mathcal{R}_G = \langle \alpha_1, \alpha_2 \rangle$, $\mathcal{E}_G = \{(\alpha_1 \rightarrow \alpha_2)\}$, $l_G = \emptyset$,
- α_1 is of sort from and α_2 is of sort into.

The empty graph is an id_0 -graph and for $k > 0$, a graph G is an id_k -graph if it is of the form $H \otimes H'$, where H is an id-graph and H' is an id_{k-1} -graph.

It is easy to check that G is an id_1 -graph iff it is an id-graph and that every id_k -graph is in CIRCUITS. In what follows, we will denote by I_k any id_k -graph.

PROPOSITION 77. Consider a graph G in CIRCUITS, and let $n \stackrel{\text{def}}{=} \mathcal{R}_G^{\text{out}}$ and $m \stackrel{\text{def}}{=} \mathcal{R}_G^{\text{in}}$. Then $G \sim (G \circ I_n)$ and $G \sim (I_m \circ G)$.

PROOF. Let $H \stackrel{\text{def}}{=} G \circ I_n$, we prove that $G \sim H$, the proof for the other statement is similar. Let $\mathcal{R}_G^{\text{in}} = \langle \alpha_1, \dots, \alpha_n \rangle$, $\mathcal{R}_H^{\text{in}} = \langle \beta_1, \dots, \beta_n \rangle$ and $\mathcal{R}_H^{\text{out}} = \langle \beta'_1, \dots, \beta'_n \rangle$. Let $\Upsilon \stackrel{\text{def}}{=} \Upsilon_{I_n}^G$. Note that by construction, for all $i = 1, \dots, n$, we have $\Upsilon(\beta'_i) = \text{trg}(\alpha_i)$ and $\Upsilon(\alpha_i) = \text{src}(\beta'_i) = \beta_i$.

Consider an \mathcal{N} -mapping μ such that for all $i = 1, \dots, n$, $\mu(\beta_i) = \alpha_i$ and for all $\alpha \in \mathcal{N}_G^*$, $\mu(\alpha) = \alpha$. We show that $\mu(H) = G$.

•

$$\begin{aligned}
\mathcal{N}_{\mu(H)} &= \mu(\Upsilon(\mathcal{N}_G) \cup \Upsilon(\mathcal{N}_{I_n})) \\
&= \mu((\mathcal{N}_G \setminus \{\alpha_i \mid i = 1, \dots, n\}) \cup \{\beta_i \mid i = 1, \dots, n\}) \\
&= (\mathcal{N}_G \setminus \{\alpha_i \mid i = 1, \dots, n\}) \cup \{\alpha_i \mid i = 1, \dots, n\} \\
&= \mathcal{N}_G.
\end{aligned}$$

•

$$\mathcal{R}_{\mu(H)} = \mu(\mathcal{R}_G^{\text{in}} \cdot \mathcal{R}_{I_n}^{\text{out}}) \quad (2)$$

$$= \mu(\mathcal{R}_G^{\text{in}}) \cdot \mu(\mathcal{R}_G^{\text{out}}) \quad (3)$$

$$= \mathcal{R}_G^{\text{in}} \cdot \mathcal{R}_G^{\text{out}} \quad (4)$$

$$= \mathcal{R}_G. \quad (5)$$

- Let $E \stackrel{\text{def}}{=} \{(\alpha_i \rightarrow \text{trg}(\alpha_i)) \mid i = 1, \dots, n\}$, so that no node α_i occurs in $\mathcal{E}_G \setminus E$. We have:

$$\begin{aligned}
\mathcal{E}_{\mu(H)} &= \mu(\Upsilon(\mathcal{E}_G) \cup \Upsilon(\mathcal{E}_{I_n})) \\
&= \mu(\Upsilon(\mathcal{E}_G \setminus E) \cup \Upsilon(E) \cup \mu(\Upsilon(\mathcal{E}_{I_n}))) \\
&= \mu(\mathcal{E}_G \setminus E) \cup \mu(\{(\beta_i \rightarrow \text{trg}(\alpha_i)) \mid i = 1, \dots, n\}) \\
&= (\mathcal{E}_G \setminus E) \cup E \\
&= \mathcal{E}_G.
\end{aligned}$$

- $l_H = \mu(l_G \cup l_{I_n}) = \mu(l_G) = l_G$.

□

Another class of graphs that are frequently used in the construction of circuits are *swap-graphs*, which intuitively permit to change the order of wires.

DEFINITION 78. A graph G is a swap-graph if it is of the following form:

- $N_G = \mathcal{R}_G = \langle \alpha_1, \alpha_2, \alpha_3, \alpha_4 \rangle, \mathcal{E}_G = \{(\alpha_1 \rightarrow \alpha_4), (\alpha_2 \rightarrow \alpha_3)\}$,
- $l_G = \emptyset$ and the nodes α_1 and α_2 are of sort `from` and nodes α_3 and α_4 are of sort `into`.

We define the subclass of *elementary circuits*, which are compositions of so-called *elementary gates*, identity graphs and swap-graphs.

DEFINITION 79. A graph G is an elementary gate if it is in *CIRCUITS* and the following conditions hold:

- N_G contains a single node of sort `gatenm`, for some n and m .
- If $(\alpha_i \rightarrow \beta_i)$ and $(\alpha_j \rightarrow \beta_j)$ are edges such that β_i occurs before β_j in $\mathcal{R}_G^{\text{in}}$, then $l_G(\alpha_i) < l_G(\alpha_j)$.
- If $(\alpha_i \rightarrow \beta_i)$ and $(\alpha_j \rightarrow \beta_j)$ are edges such that α_i occurs before α_j in $\mathcal{R}_G^{\text{out}}$, then $l_G(\beta_i) < l_G(\beta_j)$.

A graph G is an elementary circuit if it is obtained by sequential and parallel compositions from elementary gates, id-graphs and swap-graphs.

EXAMPLE 80. Graphs G_1, G_2 and G_3 in Example 72 are all elementary gates, hence $(G_2 \otimes G_1) \circ G_3$ is an elementary circuit.

The class of elementary circuits is general enough to encode several classes of graphs that are used in diagrammatic reasoning, including the orthogonal diagrams of [30] or the ZX calculus [12], thus making it to use the proof procedure of Section 6 to automatically verify formulas involving such graphs. In particular, the rules of the ZX calculus may be viewed as a set of DRLG-equations E_{ZX} on *CIRCUITS*, and checking whether two ZX diagrams G and H (denoted by DRL-graphs in *CIRCUITS*) are equivalent boils down to checking whether the DRLG-formula $E_{ZX} \cup \{G \neq H\}$ is unsatisfiable. Note that some of the nodes occurring in the ZX diagrams are labeled by real numbers in the interval $(-2\pi, 2\pi)$ (a *phase*), which cannot be denoted in our formalism. However, one may consider the fragment called $ZX_{\pi/4}$ which corresponds to the so called *Clifford+T quantum mechanics* and is *approximately universal*, in the sense that any quantum evolution (i.e., any unitary map) can be approximated by a subset of gates that can be represented with a ZX diagram where phases are multiples of $\frac{\pi}{4}$. Thus only a finite number of possible constants need to be considered [40].

9 DISCUSSION

We have defined an extension of the Superposition calculus to a class of graphs that is general enough to encode the graph-based languages that are used to perform diagrammatic reasoning on quantum algorithms and protocols. This calculus is complete, meaning that if a graph formula is unsatisfiable, then it is guaranteed that the calculus will generate \perp . We now intend to adapt an implementation of the Saturation calculus to design a tool on which

we will be able to evaluate the efficiency of the calculus and investigate how it can be improved for specific subclasses of graphs – for example, by defining suitable orderings for such subclasses in order to reduce the search space as much as possible.

There are several lines of future work that would be interesting to explore. We plan to investigate how variables representing graphs can be introduced into the calculus. During the construction of a refutation using a calculus that can handle such variables, these variables would be instantiated by graphs with specific properties that guarantee correctness, and would for example permit to automatically synthesize circuits. Another promising topic is to extend graph literals with constraints. This would permit to represent algorithms in a more concise or natural way, while still being able to automatically verify these algorithms.

REFERENCES

- [1] L. Bachmair and H. Ganzinger. Rewrite-based equational theorem proving with selection and simplification. *Journal of Logic and Computation*, 3(4):217–247, 1994.
- [2] M. Backens and A. Kissinger. ZH: A complete graphical calculus for quantum computations involving classical non-linearity. In *15th Annual Conference of Quantum Physics and Logic (QPL)*, 2018. arXiv:1805.02175.
- [3] H. P. Barendregt. Functional programming and lambda calculus. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics*, pages 321–363. Elsevier and MIT Press, 1990.
- [4] H. P. Barendregt, M. C. J. D. van Eekelen, J. R. W. Glauert, R. Kennaway, M. J. Plasmeijer, and M. R. Sleep. Term graph rewriting. In *PARLE, Parallel Architectures and Languages Europe, Volume II: Parallel Languages*, volume 259 of LNCS, pages 141–158. Springer, 1987.
- [5] P. Baumgartner and U. Waldmann. Hierarchic superposition with weak abstraction. In M. P. Bonacina, editor, *Automated Deduction - CADE-24 - 24th International Conference on Automated Deduction, Lake Placid, NY, USA, June 9-14, 2013. Proceedings*, volume 7898 of *Lecture Notes in Computer Science*, pages 39–57. Springer, 2013.
- [6] A. Bentkamp, J. Blanchette, S. Cruanes, and U. Waldmann. Superposition for lambda-free higher-order logic. *Log. Methods Comput. Sci.*, 17(2), 2021.
- [7] J. C. Blanchette, N. Peltier, and S. Robillard. Superposition with datatypes and codatatypes. In D. Galmiche, S. Schulz, and R. Sebastiani, editors, *Automated Reasoning - 9th International Joint Conference, IJCAR 2018, Held as Part of the Federated Logic Conference, FloC 2018, Oxford, UK, July 14-17, 2018. Proceedings*, volume 10900 of *Lecture Notes in Computer Science*, pages 370–387. Springer, 2018.
- [8] F. Bonchi, F. Gadducci, A. Kissinger, P. Sobocinski, and F. Zanasi. Confluence of graph rewriting with interfaces. In *26th European Symposium on Programming (21/04/17 - 28/04/17)*, February 2017.
- [9] J. H. Brenas, R. Echahed, and M. Strecker. Proving correctness of logically decorated graph rewriting systems. In *1st International Conference on Formal Structures for Computation and Deduction, FSCD 2016, June 22-26, 2016, Porto, Portugal*, volume 52 of *LIPICs*, pages 14:1–14:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.
- [10] T. Carette, D. Horsman, and S. Perdrix. SZX-calculus: Scalable graphical quantum reasoning. *arXiv preprint arXiv:1905.00041*, 2019.
- [11] A. Clément and S. Perdrix. PBS-calculus: A graphical language for quantum-controlled computations. *arXiv preprint arXiv:2002.09387*, 2020.
- [12] B. Coecke and R. Duncan. Interacting quantum observables: categorical algebra and diagrammatics. *New Journal of Physics*, 13(4):043016, apr 2011.
- [13] B. Coecke and R. Duncan. Tutorial: Graphical calculus for quantum circuits. In *International Workshop on Reversible Computation*, pages 1–13. Springer, 2012.
- [14] A. Corradini, D. Duval, R. Echahed, F. Prost, and L. Ribeiro. The PBPO graph transformation approach. *J. Log. Algebr. Meth. Program.*, 103:213–231, 2019.
- [15] A. Corradini, T. Heindel, F. Hermann, and B. König. Sesqui-pushout rewriting. In *Graph Transformations, ICGT 2006*, volume 4178 of LNCS, pages 30–45. Springer, 2006.
- [16] F. Durán, S. Eker, S. Escobar, N. Marti-Oliet, J. Meseguer, R. Rubio, and C. L. Talcott. Programming and symbolic computation in Maude. *J. Log. Algebraic Methods Program.*, 110, 2020.
- [17] R. Echahed. Inductively sequential term-graph rewrite systems. In *Graph Transformations, ICGT 2008*, volume 5214 of LNCS, pages 84–98. Springer, 2008.
- [18] H. Ehrig, K. Ehrig, U. Prange, and G. Taentzer. *Fundamentals of Algebraic Graph Transformation*. Monographs in Theoretical Computer Science. An EATCS Series. Springer, 2006.
- [19] H. Ehrig, M. Pfender, and H. J. Schneider. Graph-grammars: An algebraic approach. In *14th Annual Symposium on Switching and Automata Theory, Iowa City*,

- Iowa, USA, October 15–17, 1973, pages 167–180, 1973.
- [20] J. Engelfriet and G. Rozenberg. Node replacement graph grammars. In *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations*, pages 1–94. World Scientific, 1997.
- [21] J. Gorard, M. Namuduri, and X. D. Arsiwalla. Zx-calculus and extended wolfram model systems II: fast diagrammatic reasoning with an application to quantum circuit simplification. *CoRR*, abs/2103.15820, 2021.
- [22] A. Habel, K.-H. Pennemann, and A. Rensink. Weakest preconditions for high-level programs. In *Graph Transformations (ICGT), Natal, Brazil*, volume 4178 of LNCS, pages 445–460, Berlin, September 2006. Springer Verlag.
- [23] A. Hadzihasanovic. The algebra of entanglement and the geometry of composition. *PhD, University of Oxford*, arXiv:1709.08086, 2017.
- [24] M. Hanus. Functional logic programming: From theory to Curry. In A. Voronkov and C. Weidenbach, editors, *Programming Logics - Essays in Memory of Harald Ganzinger*, volume 7797 of *Lecture Notes in Computer Science*, pages 123–168. Springer, 2013.
- [25] J. Hullot. Canonical forms and unification. In W. Bibel and R. A. Kowalski, editors, *5th Conference on Automated Deduction, Les Arcs, France, July 8–11, 1980, Proceedings*, volume 87 of *Lecture Notes in Computer Science*, pages 318–334. Springer, 1980.
- [26] G. Hutton. *Programming in Haskell*. Cambridge University Press, 2 edition, 2016.
- [27] R. Kennaway, J. W. Klop, M. R. Sleep, and F. de Vries. On the adequacy of graph rewriting for simulating term rewriting. *ACM Trans. Program. Lang. Syst.*, 16(3):493–523, 1994.
- [28] D. Knuth and P. Bendix. Simple word problems in universal algebra. In J. Leech, editor, *Computational Problems in Abstract Algebra*, pages 263–297. Pergamon Press, 1970.
- [29] Y. Lafont. Diagram rewriting and operads, 2009. Lecture given at the Thematic school : Operads, CIRM, Luminy (Marseille).
- [30] Y. Lafont and P. Rannou. Diagram rewriting for orthogonal matrices: A study of critical peaks. In A. "Voronkov, editor, *Rewriting Techniques and Applications*, pages 232–245. Springer Berlin Heidelberg, 2008.
- [31] A. Leitsch. *The resolution calculus*. Springer. Texts in Theoretical Computer Science, 1997.
- [32] R. Milner, M. Tofte, and R. Harper. *Definition of standard ML*. MIT Press, 1990.
- [33] D. Plump. Confluence of graph transformation revisited. In A. Middeldorp, V. van Oostrom, F. van Raamsdonk, and R. C. de Vrijer, editors, *Processes, Terms and Cycles: Steps on the Road to Infinity, Essays Dedicated to Jan Willem Klop, on the Occasion of His 60th Birthday*, volume 3838 of *Lecture Notes in Computer Science*, pages 280–308. Springer, 2005.
- [34] C. M. Poskitt and D. Plump. A Hoare calculus for graph programs. In *5th International Conference on Graph Transformations (ICGT2010)*, volume 6372 of LNCS, pages 139–154. Springer, 2010.
- [35] A. Rensink. The GROOVE simulator: A tool for state space generation. In *Second International Workshop on Applications of Graph Transformations with Industrial Relevance, AGTIVE 2003*, volume 3062 of LNCS, pages 479–485. Springer, 2003.
- [36] A. Riazanov and A. Voronkov. Vampire 1.1 (system description). In *Proceedings of the International Joint Conference on Automated Reasoning (IJCAR'01)*, pages 376–380. Springer LNCS 2083, 2001.
- [37] J. A. Robinson. A machine-oriented logic based on the resolution principle. *J. Assoc. Comput. Mach.*, 12:23–41, 1965.
- [38] S. Schulz. The E Equational Theorem Prover. <http://www4.informatik.tu-muenchen.de/schulz/WORK/eprover.html>.
- [39] D. Varró. Automated formal verification of visual modeling languages by model checking. *Journal of Software and Systems Modeling*, 3(2):85–113, May 2004.
- [40] R. Vilmart, S. Perdrix, and E. Jeandel. Completeness of the zx-calculus. *Logical Methods in Computer Science*, 16, 2020.
- [41] U. Waldmann. Cancellative abelian monoids and related structures in refutational theorem proving (part I). *J. Symb. Comput.*, 33(6):777–829, 2002.
- [42] C. Weidenbach, B. Afshordel, U. Brahm, C. Cohrs, T. Engel, E. Keen, C. Theobalt, and D. Topic. System description: SPASS version 1.0.0. In *Proceedings of the 16th Conference on Automated Deduction (CADE-16)*, pages 378–382. Springer LNCS 1632, 2001.