Parallel Independence in Attributed Graph Rewriting

Thierry Boy de la Tour

Univ. Grenoble Alpes, CNRS, Grenoble INP, LIG, 38000 Grenoble, France

`thierry.boy-de-la-tour@imag.fr`

In order to define graph transformations by the simultaneous application of concurrent rules, we have adopted in previous work a structure of attributed graphs stable by unions. We analyze the consequences on parallel independence, a property that characterizes the possibility to resort to sequential rewriting. This property turns out to depend not only on the left-hand side of rules, as in algebraic approaches to graph rewriting, but also on their right-hand side. It is then shown that, of three possible definitions of parallel rewriting, only one is convenient in the light of parallel independence.

1 Introduction

The notion of parallel independence from [18, 12] has been studied mostly in the algebraic approaches to graph rewriting, see [7]. It basically consists in a condition on concurrent transformations of an object that not only guarantees but characterizes the possibility to apply the transformations sequentially in any order such that all such sequences of transformations yield the same result.

When two transformations are involved, with rules r_1 and r_2, this takes the form of the diamond property and is known as the *Local Church-Rosser Problem* [7]; it consists in finding a condition (called parallel independence) on direct transformations $H_1 \xleftarrow{r_1} G \xrightarrow{r_2} H_2$ that is equivalent to the existence of direct transformations $H_1 \xrightarrow{r_2} H \xleftarrow{r_1} H_2$ with the same redexes, hence to the existence of two equivalent sequences of transformations $G \xrightarrow{r_1} H_1 \xrightarrow{r_2} H$ and $G \xrightarrow{r_2} H_2 \xrightarrow{r_1} H$. It is obvious that non overlapping redexes always entail parallel independence, the difficulty of the problem is that the reverse does not hold and that, depending on the rules, some amount of overlap may be allowed. The notion of parallel independence is also instrumental in defining Critical Pairs (as pairs of transformations that are not parallel independent) that are central in proving confluence of sets of production rules [14, 8].

This notion should therefore also be considered in algorithmic approaches to graph rewriting. Indeed, the informal description of parallel independence given above makes perfect sense out of the algebraic approach; it is purely operational. Consider for instance Python's multiple assignment $a, b := b, a$, an elegant expression that swaps the values of a and b. We naturally understand this as a parallel expression $a := b \parallel b := a$. If a and b have the same value then the two assignments can be evaluated in sequence in any order, yielding the same result independently of the chosen order; they are parallel independent. If however they have distinct values, the two sequential evaluations yield different results (and none corresponds to the intended meaning); the two assignments are parallel dependent. Parallel dependence also typically occurs in cellular automata when rules are applied to neighbor cells, because of the overlap. Hence sequential applications of rules in an undetermined order would result in non deterministic automata.

These examples show that there is a legitimate way of computing by applying *simultaneously* concurrent transformations that may not be parallel independent, even though the result may not be reachable by sequential transformations. Swapping the values of a and b cannot be performed by applying $a := b$ or $b := a$ sequentially. This calls for a notion of *parallel transformation* for defining the result of such simultaneous applications of rules.

One such transformation has been defined in [5], in an algorithmic approach that is adopted here. It is based on directed graphs where vertices and arrows are equipped with *sets* of attributes and enables a definition of a *union* of such graphs, given in Section 2. This is a fundamental difference with terms or termgraphs and leads to a natural definition of parallel transformation in Section 3.

The consequences of these definitions on parallel independence are analyzed in Sections 4 and 5. The results of these sections are also from [5], we give them here in a slightly simpler setting and without proofs, focusing on comparisons with the algebraic approach to graph rewriting where parallel independence has been originally formulated.

In Section 6 we analyze the notion of parallel rewriting. We first define a notion of *regularity* that ensures the absence of conflicts between concurrent rules. We then show that this notion is too restricted to encompass parallel independence, and generalize it to the *effective deletion property* (also from [5]).

Section 7 is devoted to comparisons with the algebraic notion of *parallel coherence* from [4]. It is shown that its translation to the present framework, though more general than regularity, is still too restricted to encompass parallel independence. It is also shown to be the right algebraic translation of the effective deletion property. Concluding remarks and related works are presented in Section 8.

2 Attributed Graphs

We assume a many-sorted signature Σ and a set $\mathcal{V}$ of *variables*, disjoint from Σ, such that every variable has a Σ-sort. For any finite $X \subseteq \mathcal{V}$, $\mathcal{T}(\Sigma, X)$ denotes the algebra of Σ-terms over X. For any Σ-algebra $\mathscr{A}$, let $\lfloor \mathscr{A} \rfloor$ be the disjoint union of the carrier sets of the Σ-sorts in $\mathscr{A}$.

An *attributed graph* (or *graph* for short) G is a tuple $(\dot{G}, \vec{G}, \acute{G}, \grave{G}, \mathscr{A}_G, \mathring{G})$ where $\dot{G}, \vec{G}$ are sets whose elements are respectively called *vertices* and *arrows*, $\acute{G}, \grave{G}$ are the *source* and *target* functions from $\vec{G}$ to $\dot{G}$, $\mathscr{A}_G$ is a Σ-algebra and $\mathring{G}$ is an *attribution of G*, i.e., a function from $\dot{G} \cup \vec{G}$ to $\mathscr{P}(\lfloor \mathscr{A}_G \rfloor)$. The elements of $\lfloor \mathscr{A}_G \rfloor$ are called *attributes*, and we assume that $\dot{G}$, $\vec{G}$ and $\lfloor \mathscr{A}_G \rfloor$ are pairwise disjoint. G is *unlabeled* if $\mathring{G}(x) = \varnothing$ for all $x \in \dot{G} \cup \vec{G}$, it is *finite* if the sets $\dot{G}$, $\vec{G}$ and $\mathring{G}(x)$ are finite. The *carrier* of G is the set $\lfloor G \rfloor \stackrel{\text{def}}{=} \dot{G} \cup \vec{G} \cup \lfloor \mathscr{A}_G \rfloor$.

A graph H is a *subgraph* of G, written $H \lhd G$, if the *underlying graph* $(\dot{H}, \vec{H}, \acute{H}, \grave{H})$ of H is a subgraph of G's underlying graph (in the usual sense), $\mathscr{A}_H = \mathscr{A}_G$ and $\mathring{H}(x) \subseteq \mathring{G}(x)$ for all $x \in \dot{H} \cup \vec{H}$.

Graphs are better specified as pictures. Vertices and arrows will be named and their attributes will be listed after each name, separated from it by | (which is omitted if the attribute is $\varnothing$). Since graphs may not be connected, they will be surrounded by a rectangle with rounded corners, as in:

$$H = \boxed{x \xrightarrow{\ f\ } y \mid 1} \lhd \boxed{x \mid 1 \overset{\ f\ }{\underset{g \mid 0}{\rightrightarrows}} y \mid 0, 1 \quad z} = G$$

where H is the graph such that $\dot{H} = \{x, y\}$, $\vec{H} = \{f\}$, $\acute{H}(f) = x$, $\grave{H}(f) = y$, $\mathring{H}(x) = \mathring{H}(f) = \varnothing$, $\mathring{H}(y) = \{1\}$ and similarly for G (the Σ-algebra $\mathscr{A}_H = \mathscr{A}_G$ must contain at least 0 and 1).

A *morphism* α from graph H to graph G, written $\alpha : H \to G$, is a function from $\lfloor H \rfloor$ to $\lfloor G \rfloor$ such that the restriction of α to $\dot{H} \cup \vec{H}$ is a morphism from H's to G's underlying graphs (that is, $\acute{G} \circ \alpha = \alpha \circ \acute{H}$ and $\grave{G} \circ \alpha = \alpha \circ \grave{H}$, this restriction of α is called the *underlying graph morphism of α*), the restriction of α to $\lfloor \mathscr{A}_H \rfloor$ is a Σ-homomorphism from $\mathscr{A}_H$ to $\mathscr{A}_G$, denoted $\mathring{\alpha}$, and $\mathring{\alpha} \circ \mathring{H}(x) \subseteq \mathring{G} \circ \alpha(x)$ for all $x \in \dot{H} \cup \vec{H}$. Note that $H \lhd G$ iff $\lfloor H \rfloor \subseteq \lfloor G \rfloor$ and the canonical injection from $\lfloor H \rfloor$ to $\lfloor G \rfloor$ is a morphism from H to

G. For all $F \lhd H$, the *image* $\alpha(F)$ is the smallest subgraph of *G* w.r.t. the order $\lhd$ such that $\alpha|_{\lfloor F \rfloor}$ is a morphism from *F* to $\alpha(F)$.

An *isomorphism* is a morphism that has an inverse morphism. We write $H \simeq G$ if there is an isomorphism from *H* to *G*. A morphism $\mu : H \to G$ is a *matching* if the underlying graph morphism of μ is injective. For any $F \lhd H$ it is then easy to see that

$$\mu(F) = (\mu(\dot{F}),\ \mu(\vec{F}),\ \mu \circ \acute{F} \circ \mu^{-1},\ \mu \circ \grave{F} \circ \mu^{-1},\ \mathscr{A}_G,\ \mathring{\mu} \circ \mathring{F} \circ \mu^{-1}\ ).$$

Given two attributions *l* and l' of *G* let $l \setminus l'$ (resp. $l \cap l'$, $l \cup l'$) be the attribution of *G* that maps any *x* to $l(x) \setminus l'(x)$ (resp. $l(x) \cap l'(x)$, $l(x) \cup l'(x)$). If *l* is an attribution of a subgraph $H \lhd G$, it is implicitly extended to the attribution of *G* that is identical to *l* on $\dot{H} \cup \vec{H}$ and maps any other entry to $\varnothing$.

Unions of graphs can only be formed between *joinable* graphs, i.e., graphs that have a common part. We start with a simpler notion of joinable functions.

Definition 2.1 (joinable functions). Two functions $f : D \to C$ and $g : D' \to C'$ are *joinable* if $f(x) = g(x)$ for all $x \in D \cap D'$. Then, the *meet* of *f* and *g* is the function $f \curlywedge g : D \cap D' \to C \cap C'$ that is the restriction of *f* (or *g*) to $D \cap D'$. The *join* $f \curlyvee g$ is the unique function from $D \cup D'$ to $C \cup C'$ such that $f = (f \curlyvee g)|_D$ and $g = (f \curlyvee g)|_{D'}$.

For any set *I* and any *I*-indexed family $(f_i : D_i \to C_i)_{i \in I}$ of pairwise joinable functions, let $\curlyvee_{i \in I} f_i$ be the only function from $\bigcup_{i \in I} D_i$ to $\bigcup_{i \in I} C_i$ such that $f_i = (\curlyvee_{i \in I} f_i)|_{D_i}$ for all $i \in I$.

We see that any two restrictions $f|_A$ and $f|_B$ of the same function *f* are joinable, and then $f|_A \curlywedge f|_B = f|_{A \cap B}$ and $f|_A \curlyvee f|_B = f|_{A \cup B}$. Conversely, if *f* and *g* are joinable then each is a restriction of $f \curlyvee g$.

Definition 2.2 (joinable graphs). Two graphs *H* and *G* are *joinable* if $\mathscr{A}_H = \mathscr{A}_G$, $\dot{H} \cap \vec{G} = \vec{H} \cap \dot{G} = \varnothing$, and the functions $\acute{H}$ and $\acute{G}$ (and similarly $\grave{H}$ and $\grave{G}$) are joinable. We can then define the graphs

$$H \sqcap G \ \overset{\text{def}}{=} \ (\ \dot{H} \cap \dot{G},\ \vec{H} \cap \vec{G},\ \acute{H} \curlywedge \acute{G},\ \grave{H} \curlywedge \grave{G},\ \mathscr{A}_H,\ \mathring{H} \cap \mathring{G}\ ),$$

$$H \sqcup G \ \overset{\text{def}}{=} \ (\ \dot{H} \cup \dot{G},\ \vec{H} \cup \vec{G},\ \acute{H} \curlyvee \acute{G},\ \grave{H} \curlyvee \grave{G},\ \mathscr{A}_H,\ \mathring{H} \cup \mathring{G}\ ).$$

Similarly, if $(G_i)_{i \in I}$ is an *I*-indexed family of graphs that are pairwise joinable, and $\mathscr{A}$ is an algebra such that $\mathscr{A} = \mathscr{A}_{G_i}$ for all $i \in I$, then let

$$\bigsqcup_{i \in I} G_i \ \overset{\text{def}}{=} \ (\ \bigcup_{i \in I} \dot{G}_i,\ \bigcup_{i \in I} \vec{G}_i,\ \curlyvee_{i \in I} \acute{G}_i,\ \curlyvee_{i \in I} \grave{G}_i,\ \mathscr{A},\ \bigcup_{i \in I} \mathring{G}_i\ ).$$

It is easy to see that these structures are graphs: the sets of vertices and arrows are disjoint and the adjacency functions have the correct domains and codomains. If $I = \varnothing$ the chosen algebra $\mathscr{A}$ is generally obvious from the context. Note that if *H* and *G* are joinable then $H \sqcap G \lhd H \lhd H \sqcup G$. Similarly, if the G_i's are pairwise joinable then $G_j \lhd \bigsqcup_{i \in I} G_i$ for all $j \in I$. We see that any two subgraphs of *G* are joinable, and that $H \lhd G$ iff $H \sqcap G = H$ iff $H \sqcup G = G$. These operations are commutative and, on triples of pairwise joinable graphs, they are associative and distributive over each other. For any two graphs *H*, *G* there exists $G' \simeq G$ such that *H* and G' are joinable (one possibility is to take $\dot{G}' \cap \vec{H} = \varnothing$ and $\vec{G}' \cap (\dot{H} \cup \vec{H}) = \varnothing$).

For any sets *V*, *A* and attribution *l*, we say that *G is disjoint from V, A, l* if $\dot{G} \cap V = \varnothing$, $\vec{G} \cap A = \varnothing$ and $\mathring{G}(x) \cap l(x) = \varnothing$ for all $x \in \dot{G} \cup \vec{G}$. We write $G \setminus [V, A, l]$ for the largest subgraph of *G* (w.r.t. $\lhd$) that is disjoint from *V*, *A*, *l*. This provides a natural way of removing objects from an attributed graph. It is easy to see that this subgraph always exists (it is the union of all subgraphs of *G* disjoint from *V*, *A*, *l*), hence rewriting steps will not be restricted by a *gluing condition* as in the Double-Pushout approach (see [11]).

3   Applying Rules in Parallel

Definition 3.1 (rules, matchings)**.** For any finite $X \subseteq \mathscr{V}$, a (Σ, X)-*graph* is a finite graph G such that $\mathscr{A}_G = \mathscr{T}(\Sigma, X)$. Let $\mathrm{Var}(G) \overset{\text{def}}{=} \bigcup_{x \in \dot{G} \cup \vec{G}} \big(\bigcup_{t \in \mathring{G}(x)} \mathrm{Var}(t) \big)$, where $\mathrm{Var}(t)$ is the set of variables occurring in t.

A *rule* r is a triple (L, K, R) of (Σ, X)-graphs such that L and R are joinable, $L \sqcap R \lhd K \lhd L$ and $\mathrm{Var}(L) = X$ (see Remark 3.2 below). The rule r is *unlabeled* if L, K and R are unlabeled.

A *matching of r in* a graph G is a matching μ from L to G that is *consistent*, i.e., such that $\mathring{\mu}(\mathring{L}(x) \setminus \mathring{K}(x)) \cap \mathring{\mu}(\mathring{K}(x)) = \varnothing$ (or equivalently $\mathring{\mu}(\mathring{L}(x) \setminus \mathring{K}(x)) = \mathring{\mu}(\mathring{L}(x)) \setminus \mathring{\mu}(\mathring{K}(x))$) for all $x \in \dot{K} \cup \vec{K}$. We denote $\mathscr{M}(r, G)$ the set of all matchings of r in G (they all have domain $\lfloor L \rfloor$).

We consider finite sets $\mathscr{R}$ of rules such that for all $r, r' \in \mathscr{R}$, if $(L, K, R) = r \neq r' = (L', K', R')$ then $\lfloor L \rfloor \neq \lfloor L' \rfloor$, so that $\mathscr{M}(r, G) \cap \mathscr{M}(r', G) = \varnothing$ for any graph G; we then write $\mathscr{M}(\mathscr{R}, G)$ for $\biguplus_{r \in \mathscr{R}} \mathscr{M}(r, G)$. For any $\mu \in \mathscr{M}(\mathscr{R}, G)$ there is a unique rule $\mathrm{r}_\mu \in \mathscr{R}$ such that $\mu \in \mathscr{M}(\mathrm{r}_\mu, G)$, and its components are denoted $\mathrm{r}_\mu = (\mathrm{L}_\mu, \mathrm{K}_\mu, \mathrm{R}_\mu)$.

Remark 3.2. If X were allowed to contain a variable v not occurring in L, then v would freely match any element of $\mathscr{A}_G$ and the set $\mathscr{M}(r, G)$ would contain as many matchings with essentially the same effect. Also note that $\mathrm{Var}(R) \subseteq \mathrm{Var}(L)$, R and K are joinable and $R \sqcap K = L \sqcap R$. The fact that K is not required to be a subgraph of R allows the possible deletion by other rules of data matched by K but not by R.

A rewrite step may involve the creation of new vertices in a graph, corresponding to the vertices of a rule that have no match in the input graph, i.e., those in $\dot{R} \setminus \dot{L}$ (or similarly may create new arrows). These vertices should really be new, not only different from the vertices of the original graph but also different from the vertices created by other transformations (corresponding to other matchings in the graph). We simply reuse the vertices x from $\dot{R} \setminus \dot{L}$ by *indexing* them with any relevant matching μ, each time yielding a new vertex (x, μ) which is obviously different from any new vertex (x, v) for any other matching $v \neq \mu$, and also from any vertex of G. This is similar to a construction of colimits in the category of sets.

Definition 3.3 (graph $G_\mu^\uparrow$ and matching $\mu\uparrow$)**.** For any rule $r = (L, K, R)$, graph G and $\mu \in \mathscr{M}(r, G)$ we define a graph $G_\mu^\uparrow$ together with a matching $\mu\uparrow$ of R in $G_\mu^\uparrow$. We first define the sets

$$\dot{G}_\mu^\uparrow \overset{\text{def}}{=} \mu(\dot{R} \cap \dot{K}) \cup ((\dot{R} \setminus \dot{K}) \times \{\mu\}) \quad \text{and} \quad \vec{G}_\mu^\uparrow \overset{\text{def}}{=} \mu(\vec{R} \cap \vec{K}) \cup ((\vec{R} \setminus \vec{K}) \times \{\mu\}).$$

Next we define $\mu\uparrow$ by: $\mathring{\mu}\uparrow \overset{\text{def}}{=} \mathring{\mu}$ and for all $x \in \dot{R} \cup \vec{R}$, if $x \in \dot{K} \cup \vec{K}$ then $\mu\uparrow(x) \overset{\text{def}}{=} \mu(x)$ else $\mu\uparrow(x) \overset{\text{def}}{=} (x, \mu)$. Since the restriction of $\mu\uparrow$ to $\dot{R} \cup \vec{R}$ is bijective, then $\mu\uparrow$ is a matching from R to the graph

$$G_\mu^\uparrow \overset{\text{def}}{=} \big(\dot{G}_\mu^\uparrow, \vec{G}_\mu^\uparrow, \mu\uparrow \circ \acute{R} \circ \mu\uparrow^{-1}, \mu\uparrow \circ \grave{R} \circ \mu\uparrow^{-1}, \mathscr{A}_G, \mathring{\mu}\uparrow \circ \mathring{R} \circ \mu\uparrow^{-1} \big).$$

By construction $\mu\uparrow(R) = G_\mu^\uparrow$, the matchings μ and $\mu\uparrow$ are joinable and $\mu \curlywedge \mu\uparrow$ is a matching from $R \sqcap K$ to $\mu(R \sqcap K)$. It is easy to see that the graph G and the graphs $G_\mu^\uparrow$ are pairwise joinable.

For any set $M \subseteq \mathscr{M}(\mathscr{R}, G)$ of matchings in a graph G we define below how to transform G by applying simultaneously the rules associated with matches in M. This simply consists in first removing simultaneously all the vertices, arrows and attributes that are matched by L_μ but not by K_μ for any $\mu \in M$, and then in adding simultaneously all the images of the right-hand sides R_μ.

Definition 3.4 (graph $G\|_M$)**.** For any graph G and set $M \subseteq \mathscr{M}(\mathscr{R}, G)$ let

$$G\|_M \overset{\text{def}}{=} G \setminus [\mathrm{V}_M, \mathrm{A}_M, \ell_M] \sqcup \bigsqcup_{\mu \in M} G_\mu^\uparrow \text{ where}$$

$$V_M \stackrel{\text{def}}{=} \bigcup_{\mu \in M} \mu(\dot{L}_\mu \setminus \dot{K}_\mu), \quad A_M \stackrel{\text{def}}{=} \bigcup_{\mu \in M} \mu(\vec{L}_\mu \setminus \vec{K}_\mu) \text{ and } \ell_M \stackrel{\text{def}}{=} \bigcup_{\mu \in M} \mathring{\mu} \circ (\mathring{L}_\mu \setminus \mathring{K}_\mu) \circ \mu^{-1}.$$

If M is a singleton $\{\mu\}$ we write $G\|_\mu$ for $G\|_M$, V_μ for V_M, etc.

Example 3.5. We represent the simultaneous assignment $a, b := b, a$ by two rules that correspond to the simple assignments $a := b$ and $b := a$. For this we use a signature Σ with two constants a and b of sort `identifier`, and a set of variables $\mathcal{V}$ with two variables u and v of sort `integer`. The environment is represented by two nodes x and y, each attributed by an identifier and its value. More precisely, let $\mathcal{A}$ be the Σ-algebra where the sort `integer` is interpreted as $\mathcal{A}_{\text{integer}} = \mathbb{Z}$, the sort `identifier` as $\mathcal{A}_{\text{identifier}} = \{a, b\}$ and each constant as itself. We consider the environment where $a = 1$ and $b = -1$; this is represented by the graph

$$G = \boxed{x \mid a, 1 \qquad y \mid b, -1}$$

We consider the rules

$$r_1 = (\boxed{x_1 \mid a, u \qquad y_1 \mid b, v}, \boxed{x_1 \mid a \qquad y_1 \mid b, v}, \boxed{x_1 \mid a, v})$$

$$r_2 = (\boxed{x_2 \mid a, u \qquad y_2 \mid b, v}, \boxed{x_2 \mid a, u \qquad y_2 \mid b}, \boxed{y_2 \mid b, u})$$

that correspond to $a := b$ and $b := a$ respectively. We see that r_1 removes the content u associated to a and replaces it by the content v associated to b. There is exactly one matching μ_i of rule r_i in G for $i = 1, 2$, given by

	x_1	y_1	a	b	u	v
μ_1	x	y	a	b	1	-1

	x_2	y_2	a	b	u	v
μ_2	x	y	a	b	1	-1

Let $M = \{\mu_1, \mu_2\}$. Since no vertex or arrow is removed we have $V_M = A_M = \varnothing$. We also have

$$\begin{aligned}
\ell_M(x) &= \left(\mathring{\mu}_1 \circ (\mathring{L}_{\mu_1} \setminus \mathring{K}_{\mu_1}) \circ \mu_1^{-1}(x)\right) \cup \left(\mathring{\mu}_2 \circ (\mathring{L}_{\mu_2} \setminus \mathring{K}_{\mu_2}) \circ \mu_2^{-1}(x)\right) \\
&= \mathring{\mu}_1\left(\mathring{L}_{\mu_1}(x_1) \setminus \mathring{K}_{\mu_1}(x_1)\right) \cup \mathring{\mu}_2\left(\mathring{L}_{\mu_2}(x_2) \setminus \mathring{K}_{\mu_2}(x_2)\right) \\
&= \mathring{\mu}_1(\{u\}) \cup \mathring{\mu}_2(\varnothing) \\
&= \{1\}
\end{aligned}$$

and similarly $\ell_M(y) = \{-1\}$, so that $G \setminus [V_M, A_M, \ell_M] = \boxed{x \mid a \qquad y \mid b}$. Finally, we see that

$$G^{\uparrow}_{\mu_1} = \mu_1 {\uparrow} (R_{\mu_1}) = \mu_1 {\uparrow} (\boxed{x_1 \mid a, v}) = \boxed{x \mid a, -1}$$

$$G^{\uparrow}_{\mu_2} = \mu_2 {\uparrow} (R_{\mu_2}) = \mu_2 {\uparrow} (\boxed{y_2 \mid b, u}) = \boxed{y \mid b, 1}$$

and hence

$$G\|_M = \boxed{x \mid a \qquad y \mid b} \sqcup \boxed{x \mid a, -1} \sqcup \boxed{y \mid b, 1} = \boxed{x \mid a, -1 \qquad y \mid b, 1}$$

that represents the environment where $a = -1$ and $b = 1$, i.e., where the initial values of a and b have been swapped. Note that the same transformation can obviously be performed by the single rule

$$(\boxed{x_1 \mid a, u \qquad y_1 \mid b, v}, \boxed{x_1 \mid a \qquad y_1 \mid b}, \boxed{x_1 \mid a, v \qquad y_1 \mid b, u})$$

More importantly, this rule can be computed from r_1 and r_2 (see [4]).

In Definition 3.4 $G\|_M$ is guaranteed to be a graph since the $\sqcup$ operation is only applied on joinable graphs. Every morphism $\mu\uparrow$ is a matching from the right-hand side R_μ to the result $G\|_M$ of the transformation. The case where M is a singleton defines the classical semantics of one sequential rewrite step.

Definition 3.6 (sequential rewriting). For any finite set of rules $\mathscr{R}$, we define the relation $\longrightarrow_{\mathscr{R}}$ of *sequential rewriting* by stating that, for all graphs G and H,

$$G \longrightarrow_{\mathscr{R}} H \text{ iff there exists some } \mu \in \mathscr{M}(\mathscr{R},G) \text{ such that } H \simeq G\|_\mu.$$

4 Sequential Independence

In the Double-Pushout approach to graph rewriting (see [11]), production rules are spans $L \leftarrow K \rightarrow R$, with two morphisms from an *interface K* to the left- and right-hand sides L, R. These objects and morphisms are taken in a category, possibly of some sort of graphs. Direct derivations are diagrams

$$
\begin{array}{ccccc}
L & \longleftarrow & K & \longrightarrow & R \\
\downarrow\mu_1 & & \downarrow & & \downarrow \\
G & \longleftarrow & D & \longrightarrow & H
\end{array}
$$

where the two squares are *pushouts*, i.e., a form of union. Since objects, say D and R, can always be understood modulo isomorphisms, their union cannot be defined without specifying what they have in common; this is the rôle of K and of the morphisms from K to D and R. If for instance K is empty then the pushout H is the disjoint union (or direct sum, or co-product) of D and R. Hence the right square adds something to D, and inversely the left square removes something from G. Hence H is obtained from G by removing an image of L and writing an image of R, with the possibility that L and R share a common part given by K. This very general approach has a drawback: depending on G and μ_1 the object D may not exist, and if it does it may not be unique.

In this approach sequential independence is a property of two consecutive direct transformations, formulated as the existence of two commuting morphisms j_1 and j_2 as shown below.

$$
\begin{array}{ccccccccccccc}
L_1 & \longleftarrow & K_1 & \longrightarrow & R_1 & & L_2 & \longleftarrow & K_2 & \longrightarrow & R_2 \\
\downarrow\mu_1 & & \downarrow & & & j_2 \quad j_1 & & & \downarrow & & \downarrow \\
G & \longleftarrow & D_1 & \longrightarrow & H_1 & \longleftarrow & & D_2 & \longrightarrow & H_2
\end{array}
$$

It is then proven by the Local Church-Rosser Theorem that the two production rules can be applied in reverse order to G and yield the same result H_2 (we may call this the *swapping property*). Of course, the matchings $\mu_1 : L_1 \rightarrow G$ and $\mu_2 : L_2 \rightarrow H_1$ are then replaced by other matchings $\mu_1' : L_1 \rightarrow H_1'$ and $\mu_2' : L_2 \rightarrow G$ that are related to μ_1 and μ_2. A drawback of this definition is that it does not account for longer sequences of direct transformations. Indeed, if three consecutive steps are given by (μ_1, μ_2, μ_3), it is possible to swap μ_1 with μ_2 if they are sequential independent, and similarly for μ_2 and μ_3, but this does not imply that μ_1 and μ_3 can be swapped under these hypotheses (because the matchings, and hence the direct transformations, are modified by the swapping operations). We would need to express sequential independence between μ_1 and μ_3, but the definition does not apply since they are not consecutive steps.

More elaborate notions of equivalence between sequences of direct transformations are thus required (see the notion of *shift equivalence* in [7, chapter 3.5]).

Because of the specificities of our framework (no pushouts, horizontal morphisms are only canonical injections, and there may be no such morphism from K to R) we need a different definition of sequential independence. It is natural to think of the swapping property itself as the definition of sequential independence, since it describes the operational meaning of parallel independence, but we are faced with another problem. We are dealing with possibly infinite sets of matchings of rules in a graph, and we cannot form a notion of infinite sequences of rewrite steps (because each step may both remove and add data). Yet we do not wish to restrict the notion to finite sets, not simply for the sake of generality but also because it is closely related to parallel independence, a notion that can naturally be defined on infinite sets (see Section 5).

We may however use Definition 3.4 to handle infinite sets of matchings, by using the graph $G\|_M$ to stand for the result of an (independent) sequence of transformations. We may thus express sequential independence as a generalized swapping property, where the swap is performed between one transformation and all the others (taken in parallel). Yet this definition would not imply that all subsets of a sequential independent set are sequential independent, hence it needs to be stated in a more general way, by swapping any transformation with any others (and not only with all the others).

Definition 4.1. [sequential independence] For any graph G and set $M \subseteq \mathcal{M}(\mathcal{R}, G)$, we say that M is *sequential independent* if for all $N \subseteq M$ and all $\mu \in M \setminus N$,

- $\mu(L_\mu) \lhd G\|_N$, hence there is a canonical injection j from $\mu(L_\mu)$ to $G\|_N$,
- there exists an isomorphism α such that $\alpha(G\|_{N\cup\{\mu\}}) = (G\|_N)\|_{j\circ\mu}$ and α is the identity on G.

The isomorphism α in Definition 4.1 is necessary to account for the difference between the isomorphic graphs $\mu\uparrow(R_\mu)$ and $(j\circ\mu)\uparrow(R_\mu)$, i.e., to transform vertices or arrows of the form (x, μ) into $(x, j\circ\mu)$ (but there is no need to be that specific in the definition).

It is then easy to see (by induction on the cardinality of M) that

Proposition 4.2. *For any graph G and finite set $M \subseteq \mathcal{M}(\mathcal{R}, G)$, if M is sequential independent then*

$$G \longrightarrow^\star_{\mathcal{R}} G\|_M.$$

Of course there is usually more than one sequence of rewriting steps from G to $G\|_M$, since under the hypothesis they can be swapped; but without it there is generally none (as illustrated in Example 3.5). And the fact that there is one such sequence does not imply sequential independence, i.e., the converse of Proposition 4.2 is obviously not true.

5   Parallel Independence

In the Double-Pushout approach, parallel independence is a property of two direct transformations of the same object G, formulated as the existence of two commuting morphisms j_1 and j_2 as shown below.

The Local Church-Rosser Theorem mentioned above actually shows that μ and ν are parallel independent iff they correspond to a sequential independent pair (μ, ν') (where $\nu' : L_2 \to H_1$ is related to ν). It is the symmetry between μ and ν that entails the swapping property. This is remarkable since parallel independence does not refer to the *results* of the transformations involved, while the result of the sequences of transformations is central in the swapping property (as in Definition 4.1).

This definition of parallel independence can easily be lifted to sets M of matchings (or direct transformations) by considering all possible pairs $\mu, \nu \in M$, with a slight caveat. In this definition the two direct transformations may be identical, thus stating a property of a single transformation that is not shared by all direct transformations. But Definition 3.4 does not allow to apply any member μ of M more than once (because applying μ any number of times in parallel would jeopardize determinism of $\Longmapsto_{\mathscr{R}}$, see Definition 6.6 below). For this reason we will only consider pairs of distinct matchings (so that singletons M shall be considered as parallel independent, see below).

Our goal is therefore to formulate parallel independence in the present framework, in order to obtain an equivalence similar to the Local Church-Rosser Theorem. Considering that the pushout complement D_1 is replaced by the graph $G \setminus [V_\mu, A_\mu, \ell_\mu]$, the commuting property of j_2 amounts to $\nu(L_2) \lhd G \setminus [V_\mu, A_\mu, \ell_\mu]$, that can be more elegantly expressed as $\nu(L_2) \sqcap \mu(L_1) \lhd \mu(K_1)$, or $\nu(L_\nu) \sqcap \mu(L_\mu) \lhd \mu(K_\mu)$ using our notations. This simply means that any graph item that is matched by two concurrent rules cannot be removed. The commuting property of j_1 is obtained by swapping μ and ν.

However, our treatment of attributes makes it possible to recover in the right-hand side an attribute that has been deleted in the left-hand side (this is of course not possible for vertices or arrows). This possibility should therefore be accounted for in the notion of parallel independence, i.e., an attribute that is matched twice may be deleted provided it is recovered. This can be expressed as

$$\nu(L_\nu) \sqcap \mu(L_\mu) \lhd \mu(K_\mu) \sqcup \mu{\uparrow}(R_\mu) \text{ for all } \mu, \nu \in M \text{ such that } \mu \neq \nu.$$

However, this is not a sufficient condition for sequential independence.

Example 5.1. We consider the following graph and rules:

$$G = \boxed{x \mid 0} \text{ where } \lfloor \mathscr{A}_G \rfloor = \{0\}$$

$$r_1 = (\boxed{x_1 \mid 0}, \boxed{x_1}, \boxed{x_1})$$

$$r_2 = (\boxed{x_2}, \boxed{x_2}, \boxed{x_2 \mid 0})$$

There is a unique matching μ of r_1 (resp. ν of r_2) in G, given by $\mu(x_1) = \nu(x_2) = x$ and $\mathring{\mu}(0) = \mathring{\nu}(0) = 0$. We see that $M = \{\mu, \nu\}$ is not sequential independent. Indeed, let $N = \{\nu\}$, then $G = \mu(L_\mu) \lhd G\|_N = G$ (hence j is the identity morphism of G), but $\big(G\|_N\big)\|_\mu = G\|_\mu = \boxed{x}$ is not isomorphic to $G\|_M = G$.

Yet we see that

$$\nu(L_\nu) \sqcap \mu(L_\mu) = \boxed{x} \lhd \boxed{x} = \mu(K_\mu) \sqcup \mu{\uparrow}(R_\mu)$$

$$\mu(L_\mu) \sqcap \nu(L_\nu) = \boxed{x} \lhd \boxed{x \mid 0} = \nu(K_\nu) \sqcup \nu{\uparrow}(R_\nu),$$

which proves that this condition is true for all pairs of distinct elements of M; hence it is not sufficient to ensure sequential independence.

The problem in Example 5.1 is that the attribute 0 of x is considered as being matched only once (by L_μ), while it is actually also matched by R_ν. This leads to the following definition.

Definition 5.2 (parallel independence). For any graph G and set $M \subseteq \mathscr{M}(\mathscr{R}, G)$, we say that M is *parallel independent* if

$$(\nu(L_\nu) \sqcup \nu{\uparrow}(R_\nu)) \sqcap \mu(L_\mu) \lhd \mu(K_\mu) \sqcup \mu{\uparrow}(R_\mu) \quad \text{for all } \mu, \nu \in M \text{ such that } \mu \neq \nu.$$

This definition may seem strange, but it is easy to see that on unlabeled graphs it amounts to $\nu(L_\nu) \sqcap \mu(L_\mu) \lhd \mu(K_\mu)$ for all $\mu \neq \nu$, i.e., to the standard algebraic notion of parallel independence (translated to the present framework).

It turns out that Definition 5.2 provides the expected characterization of sequential independence.

Theorem 5.3. *For any graph G and set $M \subseteq \mathscr{M}(\mathscr{R}, G)$, M is parallel independent iff M is sequential independent.*

The (rather long) proof of Theorem 5.3 can be found in [5].

We therefore see that Definition 5.2 arises as a characterization of sequential independence that does not refer to the results of the transformations, and indeed that does not rely on the definition of $G\|_M$ (Definition 3.4), though of course it does rely on the definitions of unions of graphs, of rules and of the matchings $\mu{\uparrow}$ (Definitions 2.2, 3.1 and 3.3). Note also that Definition 5.2 depends explicitly on the right-hand sides of rules, in contrast with the general algebraic definition of parallel independence given above, or with the Essential Condition of parallel independence in [6].

6 Parallel Rewriting

We have not yet defined a relation of parallel rewriting as we did for sequential rewriting (Definition 3.6). The reason is that two matchings may conflict as one retains (in $R \sqcap K$) what another removes.

Example 6.1. We consider the following unlabeled rule r and graph G.

$$r = (\boxed{x \overset{f}{\underset{f'}{\rightleftarrows}} x'}, \boxed{x}, \boxed{x}) \qquad G = \boxed{y \overset{g}{\underset{h}{\rightleftarrows}} z}$$

There are two matchings μ_1, μ_2 of r in G, given by

	x	x'	f	f'
μ_1	y	z	g	h

	x	x'	f	f'
μ_2	z	y	h	g

According to rule r with matching μ_1, the node $\mu_1(x') = z$ and the arrows $\mu_1(f) = g$ and $\mu_1(f') = h$ have to be removed, and the node $\mu_1(x) = y$ should occur in the result of the transformation. But with matching μ_2, the node $\mu_2(x') = y$ should be removed and the node $\mu_2(x) = z$ should be preserved. There is a conflict between μ_1 and μ_2 on the nodes of G (but not on its arrows).

Let $M = \{\mu_1, \mu_2\}$, then $V_M = \mu_1(\{x'\}) \cup \mu_2(\{x'\}) = \{y, z\} = \dot{G}$ hence $G \setminus [V_M, A_M, \ell_M]$ is empty and

$$G\|_M = \mu_1(\boxed{x}) \sqcup \mu_2(\boxed{x}) = \boxed{y} \sqcup \boxed{z} = \boxed{y \qquad z}$$

The transformation offered by Definition 3.4 performs deletions before unions, which means that these conflicts are resolved by giving priority to retainers over removers. But if the deletion actions of a rule are not executed in a parallel transformation, how can we claim that this rule has been executed (or applied) in parallel with others? Thus, in order to define parallel rewriting with a clear semantics we need to rule out such conflicts.

A natural restriction is therefore to make sure that the items that should be removed, i.e., those contained in V_M, A_M or ℓ_M, have indeed been removed from the result.

Definition 6.2 (regularity). For any graph G and set $M \subseteq \mathcal{M}(\mathcal{R},G)$, we say that M is *regular* if $G\|_M$ is disjoint from V_M, A_M, ℓ_M.

As for sequential independence, this property of M can be characterized as a property of pairs of elements of M.

Lemma 6.3. *For any graph G and set $M \subseteq \mathcal{M}(\mathcal{R},G)$,*

$$M \text{ is regular } \text{ iff } v{\uparrow}(R_v) \sqcap \mu(L_\mu) \lhd \mu(K_\mu) \text{ for all } \mu, v \in M.$$

Proof. Let $H = \bigsqcup_{v \in M} G_v^{\uparrow}$, then $G\|_M = G \setminus [V_M, A_M, \ell_M] \sqcup H$ is disjoint from V_M, A_M, ℓ_M iff H is. We have

$$\dot{H} \cap V_M = \left(\bigcup_{v \in M} \dot{G}_v^{\uparrow} \right) \cap \left(\bigcup_{\mu \in M} \mu(\dot{L}_\mu \setminus \dot{K}_\mu) \right) = \bigcup_{\mu, v \in M} v{\uparrow}(\dot{R}_v) \cap \mu(\dot{L}_\mu) \setminus \mu(\dot{K}_\mu)$$

hence $\dot{H} \cap V_M = \varnothing$ iff $v{\uparrow}(\dot{R}_v) \cap \mu(\dot{L}_\mu) \setminus \mu(\dot{K}_\mu) = \varnothing$ for all $\mu, v \in M$, but this is equivalent to $v{\uparrow}(\dot{R}_v) \cap \mu(\dot{L}_\mu) \subseteq \mu(\dot{K}_\mu)$. Similarly we see that $\vec{H} \cap A_M = \varnothing$ iff $v{\uparrow}(\vec{R}_v) \cap \mu(\vec{L}_\mu) \subseteq \mu(\vec{K}_\mu)$ for all $\mu, v \in M$.

For every vertex or arrow x of $G\|_M$ we have

$$\mathring{H}(x) \cap \ell_M(x) = \bigcup_{v \in M} \mathring{v} \circ \mathring{R}_v \circ v^{-1}(x) \cap \ell_M(x)$$

$$= \bigcup_{\mu, v \in M} \mathring{v} \circ \mathring{R}_v \circ v^{-1}(x) \cap \mathring{\mu} \circ (\mathring{L}_\mu \setminus \mathring{K}_\mu) \circ \mu^{-1}(x)$$

$$= \bigcup_{\mu v \in M} \mathring{v} \circ \mathring{R}_v \circ v^{-1}(x) \cap \mathring{\mu} \circ \mathring{L}_\mu \circ \mu^{-1}(x) \setminus \mathring{\mu} \circ \mathring{K}_\mu \circ \mu^{-1}(x)$$

by using the fact that μ is consistent. We therefore see that $\mathring{H}(x) \cap \ell_M(x) = \varnothing$ holds iff $\mathring{v} \circ \mathring{R}_v \circ v^{-1}(x) \cap \mathring{\mu} \circ \mathring{L}_\mu \circ \mu^{-1}(x) \subseteq \mathring{\mu} \circ \mathring{K}_\mu \circ \mu^{-1}(x)$ holds for all $\mu, v \in M$. By definition M is regular iff $\dot{H} \cap V_M = \vec{H} \cap A_M = \varnothing$ and $\mathring{H} \cap \ell_M$ is empty everywhere, hence M is regular iff $v{\uparrow}(R_v) \sqcap \mu(L_\mu) \lhd \mu(K_\mu)$ for all $\mu, v \in M$. $\qquad \square$

Corollary 6.4. *M is regular iff all its subsets are regular.*

These nice properties, and the fact that regularity ensures the absence of conflicts, are however not sufficient in the light of parallel independence. Indeed, we now show that a parallel independent set may not be regular.

Example 6.5. Let us consider rules $r_1 = (L_1, K_1, R_1)$ and $r_2 = (L_2, K_2, R_2)$ where the graphs L_1, K_1 and R_1 have only one vertex x_1, the graphs L_2, K_2 and R_2 have only one vertex x_2, and the attributes are as pictured below (u, v are variables and f is a unary function symbol). Let $\mathscr{A}_G$ be the algebra with carrier set $\{0\}$ where f is interpreted as the constant function 0, and let G be the graph that has a unique vertex x with attribute 0.
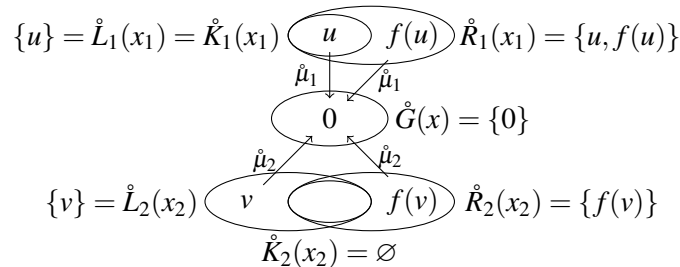
There are exactly two matchings of $\{r_1, r_2\}$ in G: μ_1 and μ_2 defined by $\mu_1(x_1) = \mu_2(x_2) = x$ and $\mathring{\mu}_1(u) = \mathring{\mu}_2(v) = 0$. Let $M = \{\mu_1, \mu_2\}$, we see by Lemma 6.3 that M is not regular since

$$\mu_1{\uparrow}(R_1) \sqcap \mu_2(L_2) = \mu_1(\boxed{x_1 \mid u, f(u)}) \sqcup \mu_2(\boxed{x_2 \mid v}) = \boxed{x \mid 0} = G$$

is not a subgraph of $\mu_2(K_2) = \mu_2(\boxed{x_2}) = \boxed{x}$ (or equivalently because $G\|_M = G$ is not disjoint from ℓ_M).

However, we see that M is sequential independent since the matchings can be applied sequentially in any order, yielding in both cases the graph G. Equivalently, M is parallel independent since

$$(\mu_2(L_2) \sqcup \mu_2{\uparrow}(R_2)) \sqcap \mu_1(L_1) = (G \sqcup G) \sqcap G = G \lhd G \sqcup G = \mu_1(K_1) \sqcup \mu_1{\uparrow}(R_1)$$
$$(\mu_1(L_1) \sqcup \mu_1{\uparrow}(R_1)) \sqcap \mu_2(L_2) = (G \sqcup G) \sqcap G = G \lhd \boxed{x} \sqcup G = \mu_2(K_2) \sqcup \mu_2{\uparrow}(R_2).$$

Note that conversely a set may be regular and not parallel independent, as is the case of the set M in Example 3.5.

We obviously need a more comprehensive notion of parallel rewriting, one that applies *at least* on all parallel independent sets of matchings. We see in Example 6.5 that the two rules do clash on the attribute 0 of x, but the clash is settled by their right-hand sides. This leads to the following definition from [5].

Definition 6.6 (effective deletion property, parallel rewriting)**.** For any graph G, a set $M \subseteq \mathcal{M}(\mathcal{R}, G)$ is said to satisfy the *effective deletion property* if $G\|_M$ is disjoint from $V_M, A_M, \ell_M \setminus \ell_M^{\uparrow}$, where

$$\ell_M^{\uparrow} \stackrel{\text{def}}{=} \bigcup_{\mu \in M} \mathring{\mu} \circ (\mathring{R}_\mu \setminus \mathring{K}_\mu) \circ \mu^{-1}.$$

For any finite set of rules $\mathcal{R}$, we define the relation $\Longrightarrow_{\mathcal{R}}$ of *parallel rewriting* by stating that, for all graphs G and H, $G \Longrightarrow_{\mathcal{R}} H$ iff there exists a set $M \subseteq \mathcal{M}(\mathcal{R}, G)$ that has the effective deletion property and such that $H \simeq G\|_M$. We write $G \Longmapsto_{\mathcal{R}} H$ if $M = \mathcal{M}(\mathcal{R}, G)$.

The effective deletion property is obviously more general than regularity. The example below shows that it is strictly more general than regularity.

Example 6.7. We consider again Example 5.1 where $M = \{\mu, \nu\}$ is not sequential independent, hence by Theorem 5.3 M is not parallel independent. We have $V_M = A_M = \emptyset$ and $\ell_M(x) = \{0\}$. Since $G\|_M = \boxed{x \mid 0}$ is not disjoint from V_M, A_M, ℓ_M then M is not regular. But

$$\ell_M^{\uparrow}(x) = \mathring{\mu} \circ (\mathring{R}_1 \setminus \mathring{K}_1) \circ \mu^{-1}(x) \cup \mathring{\nu} \circ (\mathring{R}_2 \setminus \mathring{K}_2) \circ \nu^{-1}(x) = \{0\},$$

hence $\ell_M(x) \setminus \ell_M^{\uparrow}(x) = \varnothing$ and therefore M has the effective deletion property.

It has been shown in [5] that $\Longmapsto_{\mathcal{R}}$ is deterministic up to isomorphism, that is, if $G \Longmapsto_{\mathcal{R}} H$, $G' \Longmapsto_{\mathcal{R}} H'$ and $G \simeq G'$ then $H \simeq H'$. In particular, it is possible to represent any cellular automaton by a suitable rule r and a class of graphs that correspond to configurations of the automaton (every vertex corresponds to a cell), such that $\Longmapsto_r$ (restricted to such graphs) is the transition function of the automaton. Furthermore, it is proved in [5] (as a lemma to Theorem 5.3) that

Theorem 6.8. *For any graph G and set $M \subseteq \mathcal{M}(\mathcal{R}, G)$ if M is parallel independent then M has the effective deletion property.*

Hence effective deletion supports a definition of parallel rewriting that is general enough to handle parallel independence. Besides, Example 6.7 also shows that the effective deletion property is strictly more general than parallel independence.

We also see that

Corollary 6.9. *If $M \subseteq \mathcal{M}(\mathcal{R}, G)$ is finite and parallel independent then $G \longrightarrow_{\mathcal{R}}^{\star} G\|_M$ and $G \Longrightarrow_{\mathcal{R}} G\|_M$.*

Proof. By Theorem 6.8 we have $G \Longrightarrow_{\mathcal{R}} G\|_M$. By Theorem 5.3 M is sequential independent, hence by Proposition 4.2 we have $G \longrightarrow_{\mathcal{R}}^{\star} G\|_M$. □

Hence in this case parallel and sequential rewriting meet, and parallel rewriting can be said to yield a *correct* result w.r.t. sequential rewriting.

7 Parallel Coherence

One drawback of the effective deletion property is that it cannot be characterized as a property of pairs of elements of M, as the following example shows.

Example 7.1. We consider the following graph and rule

$$G = \boxed{x \mid 0} \text{ where } \lfloor \mathscr{A}_G \rfloor = \{0,1\}$$

$$r_3 = (\boxed{x_3 \mid 0}, \boxed{x_3 \mid 0}, \boxed{x_3 \mid 0, 1})$$

and also the rules r_1, r_2 of Example 5.1. For $i = 1, 2, 3$ let μ_i be the unique matching of r_i in G such that $\mathring{\mu}_i$ is the identity function of $\{0,1\}$. Let $M = \{\mu_1, \mu_2, \mu_3\}$ and $N = \{\mu_1, \mu_3\}$. We obviously have $V_M = V_N = A_M = A_N = \varnothing$. We see that $\ell_{\mu_1}(x) = \{0\}$ and $\ell_{\mu_2}(x) = \ell_{\mu_3}(x) = \varnothing$, so that $\ell_N(x) = \ell_M(x) = \{0\}$,

$$G\|_N = \boxed{x} \sqcup \boxed{x} \sqcup \boxed{x \mid 0, 1} = \boxed{x \mid 0, 1} \text{ and } G\|_M = \boxed{x} \sqcup \boxed{x} \sqcup \boxed{x \mid 0} \sqcup \boxed{x \mid 0, 1} = \boxed{x \mid 0, 1}$$

We then see that $\ell_{\mu_1}^{\uparrow}(x) = \varnothing$, $\ell_{\mu_2}^{\uparrow}(x) = \{0\}$ and $\ell_{\mu_3}^{\uparrow}(x) = \{1\}$, so that $\ell_N^{\uparrow}(x) = \{1\}$ and $\ell_M^{\uparrow}(x) = \{0,1\}$. Hence $\ell_M(x) \setminus \ell_M^{\uparrow}(x) = \varnothing$ and $\ell_N(x) \setminus \ell_N^{\uparrow}(x) = \{0\}$, and M but not N has the effective deletion property.

The reader may find strange that the conflict between r_1 and r_3 could be settled by some other rule, here r_2. This means that we need the whole of M to decide wether all conflicts are settled. For this reason the effective deletion property may appear as too general.

Another possibility for defining parallel rewriting is to translate to the present framework the notion of parallel coherence that has been devised in order to define algebraic parallel graph transformation (see [4]). In that paper we used production rules of the form $L \leftarrow K \leftarrow I \rightarrow R$ that do not require a morphism from K to R. Direct derivations are commuting diagrams

$$L \longleftarrow K \longleftarrow I \longrightarrow R$$
$$G \longleftarrow D \longrightarrow H$$

where the squares are pushouts. Note that a standard Double-Pushout can be obtained with $K = I$. *Parallel coherence*, as a property of two direct transformations of the same object G, is defined as the existence of two commuting morphisms j_1 and j_2 as shown below.

$$R_1 \longleftarrow I_1 \dashrightarrow K_1 \longrightarrow L_1 \qquad L_2 \longleftarrow K_2 \longleftarrow I_2 \longrightarrow R_2$$
$$H_1 \longleftarrow D_1 \dashrightarrow G \longleftarrow D_2 \longrightarrow H_2$$

This notion clearly generalizes algebraic parallel independence and is therefore a good candidate. In the present framework the object I_2 is replaced by the graph $R_2 \sqcap K_2$, hence the commuting property of j_2 amounts to $v(R_2 \sqcap K_2) \lhd G \setminus [V_\mu, A_\mu, \ell_\mu]$, that can be expressed as $\mu(L_\mu) \sqcap v(R_v \sqcap K_v) \lhd \mu(K_\mu)$. This simply means that any graph item that is matched by some $R \sqcap K$ cannot be removed by any rule.

Definition 7.2 (parallel coherence). For any graph G and set $M \subseteq \mathcal{M}(\mathcal{R}, G)$, we say that M is *parallel coherent* if

$$v(R_v \sqcap K_v) \sqcap \mu(L_\mu) \lhd \mu(K_\mu) \text{ for all } \mu, v \in M.$$

We easily show that this notion is more general than regularity.

Lemma 7.3. *For any graph G and set $M \subseteq \mathcal{M}(\mathcal{R}, G)$, if M is regular then M is parallel coherent.*

Proof. By Lemma 6.3 we have $v\!\uparrow\!(R_v) \sqcap \mu(L_\mu) \lhd \mu(K_\mu)$ for all $\mu, v \in M$. Since $R_v \sqcap K_v \lhd R_v$ then $v(R_v \sqcap K_v) = v\!\uparrow\!(R_v \sqcap K_v) \lhd v\!\uparrow\!(R_v)$, hence $v(R_v \sqcap K_v) \sqcap \mu(L_\mu) \lhd v\!\uparrow\!(R_v) \sqcap \mu(L_\mu) \lhd \mu(K_\mu)$, hence M is parallel coherent. $\qquad\square$

It is easy to see that the converse does not hold (use for instance Example 5.1). We now show that parallel coherence is a restriction of the (possibly too general) effective deletion property.

Theorem 7.4. *For any graph G and set $M \subseteq \mathcal{M}(\mathcal{R}, G)$, if M is parallel coherent then M has the effective deletion property.*

Proof. Let $H = G\|_M$ then as in the proof of Lemma 6.3 we have

$$\dot{H} \cap V_M = \bigcup_{\mu, v \in M} v\!\uparrow\!(\dot{R}_v) \cap \mu(\dot{L}_\mu) \setminus \mu(\dot{K}_\mu).$$

But $\mu(\dot{L}_\mu) \subseteq \dot{G}$ and by Definition 3.3 we have

$$
\begin{aligned}
\dot{G} \cap v\!\uparrow\!(\dot{R}_v) &= \dot{G} \cap \dot{G}_v^{\uparrow} \\
&= \dot{G} \cap \left(v(\dot{R}_v \cap \dot{K}_v) \cup ((\dot{R}_v \setminus \dot{K}_v) \times \{v\}) \right) \\
&= v(\dot{R}_v \cap \dot{K}_v),
\end{aligned}
$$

hence

$$\dot{H} \cap V_M = \bigcup_{\mu, v \in M} v(\dot{R}_v \cap \dot{K}_v) \cap \mu(\dot{L}_\mu) \setminus \mu(\dot{K}_\mu) = \varnothing$$

since by parallel coherence $v(\dot{R}_v \cap \dot{K}_v) \cap \mu(\dot{L}_\mu) \subseteq \mu(\dot{K}_\mu)$ for all $\mu, v \in M$. Similarly $\vec{H} \cap A_M = \varnothing$.

For all $x \in \dot{H} \cup \vec{H}$, if $x \notin \dot{G} \cup \vec{G}$ then $\ell_M(x) = \varnothing$ and obviously $\mathring{H}(x) \cap \ell_M(x) \setminus \ell_M^{\uparrow}(x) = \varnothing$. Otherwise $x \in \dot{G} \cup \vec{G}$ hence $v\!\uparrow^{-1}(x) = v^{-1}(x)$ so that

$$\mathring{H}(x) = \left(\mathring{G}(x) \setminus \ell_M(x) \right) \cup \bigcup_{v \in M} \mathring{v} \circ \mathring{R}_v \circ v^{-1}(x).$$

Using the identity $A = (A \setminus B) \cup (A \cap B)$ for all sets A and B we have

$$
\begin{aligned}
\mathring{v} \circ \mathring{R}_v \circ v^{-1}(x) = &\left(\mathring{v} \circ \mathring{R}_v \circ v^{-1}(x) \setminus \mathring{v} \circ (\mathring{R}_v \cap \mathring{K}_v) \circ v^{-1}(x) \right) \\
&\cup \left(\mathring{v} \circ \mathring{R}_v \circ v^{-1}(x) \cap \mathring{v} \circ (\mathring{R}_v \cap \mathring{K}_v) \circ v^{-1}(x) \right)
\end{aligned}
$$

for all $v \in M$. By parallel coherence we have $\mathring{v} \circ (\mathring{R}_v \cap \mathring{K}_v) \circ v^{-1}(x) \cap \mathring{\mu} \circ \mathring{L}_\mu \circ \mu^{-1}(x) \subseteq \mathring{\mu} \circ \mathring{K}_\mu \circ \mu^{-1}(x)$ for all $\mu, v \in M$, and since μ is consistent we get

$$\mathring{v} \circ (\mathring{R}_v \cap \mathring{K}_v) \circ v^{-1}(x) \cap \ell_M(x) = \bigcup_{\mu \in M} \mathring{v} \circ (\mathring{R}_v \cap \mathring{K}_v) \circ v^{-1}(x) \cap \mathring{\mu} \circ (\mathring{L}_\mu \setminus \mathring{K}_\mu) \circ \mu^{-1}(x)$$

$$= \bigcup_{\mu \in M} \mathring{v} \circ (\mathring{R}_v \cap \mathring{K}_v) \circ v^{-1}(x) \cap \mathring{\mu} \circ \mathring{L}_\mu \circ \mu^{-1}(x) \setminus \mathring{\mu} \circ \mathring{K}_\mu \circ \mu^{-1}(x)$$

$$= \varnothing,$$

hence

$$\mathring{H}(x) \cap \ell_M(x) = \bigcup_{v \in M} \mathring{v} \circ \mathring{R}_v \circ v^{-1}(x) \cap \ell_M(x)$$

$$= \bigcup_{v \in M} \left(\mathring{v} \circ \mathring{R}_v \circ v^{-1}(x) \setminus \mathring{v} \circ (\mathring{R}_v \cap \mathring{K}_v) \circ v^{-1}(x) \right) \cap \ell_M(x).$$

Finally, by using the obvious fact that $f(A) \setminus f(A \cap B) \subseteq f(A \setminus B)$ for any function f, we get

$$\mathring{H}(x) \cap \ell_M(x) \subseteq \bigcup_{v \in M} \mathring{v} \circ (\mathring{R}_v \setminus \mathring{K}_v) \circ v^{-1}(x) \cap \ell_M(x) \subseteq \ell_M^\uparrow(x)$$

hence $\mathring{H}(x) \cap \ell_M(x) \setminus \ell_M^\uparrow(x) = \varnothing$. This proves that H is disjoint from V_M, A_M, $\ell_M \setminus \ell_M^\uparrow$ and therefore that M has the effective deletion property. □

Yet parallel coherence is not sufficient in the light of parallel independence, as we now show.

Proposition 7.5. *Parallel coherence does not generalize parallel independence.*

Proof. In Example 6.5 is exhibited a set $M = \{\mu_1, \mu_2\}$ that is shown to be parallel independent. But we see that

$$\mu_1(R_1 \sqcap K_1) \sqcap \mu_2(L_2) = \mu_1(\boxed{x_1 \mid u}) \sqcap \mu_2(\boxed{x_2 \mid v}) = \boxed{x \mid 0} = G$$

is not a subgraph of $\mu_2(K_2) = \boxed{x}$, hence M is not parallel coherent. □

Parallel coherence is therefore too restricted to support a definition of parallel rewriting in the present framework. The problem here as above is that deleted attributes can be recovered by the right-hand side of rules, and that this possibility is not accounted for in the algebraic definitions, since these do not distinguish between graph items and attributes.

To summarize, we have established that the following implications hold, and no other:

regularity $\Rightarrow$ parallel coherence $\Rightarrow$ effective deletion property $\Leftarrow$ parallel independence.

We see this as an endorsement of parallel rewriting based on the effective deletion property (Definition 6.6), even if it is the only property that cannot be characterized simply on pairs of matchings. This suggests that the effective deletion property would be worth transposing to an algebraic framework. But there is no straightforward way of doing this, as can now be shown.

Corollary 7.6. *For any set of unlabeled rules $\mathscr{R}$, any unlabeled graph G and any subset $M \subseteq \mathscr{M}(\mathscr{R}, G)$,*

M is regular iff M is parallel coherent iff M has the effective deletion property.

Proof. Assume that M has the effective deletion property, then $G\|_M$ is disjoint from $V_M, A_M, \ell_M \setminus \ell_M^\uparrow$ hence so is $\bigsqcup_{\nu \in M} \nu{\uparrow}(R_\nu)$. For all $\mu \in M$ we have $V_\mu \subseteq V_M$ and $A_\mu \subseteq A_M$, hence $\bigsqcup_{\nu \in M} \nu{\uparrow}(R_\nu)$ is disjoint from V_μ, A_μ, $\varnothing$ and therefore so is $\nu{\uparrow}(R_\nu)$ for every $\nu \in M$. Thus $\nu{\uparrow}(\dot{R}_\nu) \cap \mu(\dot{L}_\mu) \setminus \mu(\dot{K}_\mu) = \varnothing$ and $\nu{\uparrow}(\vec{R}_\nu) \cap \mu(\vec{L}_\mu) \setminus \mu(\vec{K}_\mu) = \varnothing$, which is equivalent to $\nu{\uparrow}(\dot{R}_\nu) \cap \mu(\dot{L}_\mu) \subseteq \mu(\dot{K}_\mu)$ and $\nu{\uparrow}(\vec{R}_\nu) \cap \mu(\vec{L}_\mu) \subseteq \mu(\vec{K}_\mu)$. Since these graphs are unlabeled, this entails that $\nu{\uparrow}(R_\nu) \sqcap \mu(L_\mu) \lhd \mu(K_\mu)$ for all $\mu, \nu \in M$, hence that M is regular by Lemma 6.3. The equivalences follow by Lemma 7.3 and Theorem 7.4. $\qquad\square$

Hence an algebraic approach to parallel graph transformation that would apply to the category of (unlabeled) graphs could not distinguish these notions. In this sense parallel coherence is already the right algebraic translation of the effective deletion property (and of regularity), even if it is too weak to account for the special treatment of attributes in the present non algebraic framework.

8 Related Work and Conclusion

Many notions of attributed graphs exist in the literature. For instance, in [17, 10] graph items can hold at most one attribute. This means that concurrent rules could possibly conflict because of their right-hand sides, if two rules required to attribute distinct values to the same graph item. Our choice of attaching sets of attributes to vertices and arrows means that new attributes are freely included in those sets, and thus avoids conflicting right-hand sides. Indeed, we see from Definition 6.6 that a conflict must involve an element of V_M, A_M or ℓ_M. Hence the right-hand sides of rules never *create* conflicts, though they may *settle* the conflicts created in the left-hand sides and are therefore relevant to parallel independence.

Other notions of attributed graphs that allow unbounded attributes are possible, for instance the E-graphs from [11]. But the fact that in E-graphs a single value can be referenced several times as attribute of a vertex or arrow means that the number of matchings of rules may uselessly inflate.

Another approach to parallelism is to accept overlapping, non independent matchings and ask the user to decide what to do in particular situations [13]. The present approach shows that the user can be spared this work not just on parallel independent matchings, but on the larger class of sets that satisfy the effective deletion property (or parallel coherence in an algebraic framework).

It is also possible to restrict by design all overlaps to vertices, as is the case in Hyperedge Replacement Systems [9], and still be able to specify powerful parallel transformations [15], though in a non deterministic way. Note that these are asynchronous models of parallelism, where determinism amounts to confluence. This property has been widely studied in term rewriting; it becomes more subtle when acyclic term graphs are considered [16], and more elusive when cycles are allowed [2, 1]. Our model of parallelism is a synchronous one where deterministic transformations can be designed without reference to confluence [3], as in cellular automata.

The use of parallel transformations to define sequential independence in an algebraic approach to graph rewriting (as in Definition 4.1) could be worth investigating.

References

[1] Zena M. Ariola & Stefan Blom (1997): *Cyclic Lambda Calculi*. In Martín Abadi & Takayasu Ito, editors: *Theoretical Aspects of Computer Software - TACS '97*, *LNCS* 1281, Springer, pp. 77–106, doi:10.1007/BFb0014548.

[2] Zena M. Ariola & Jan Willem Klop (1996): *Equational Term Graph Rewriting*. *Fundamenta Informaticae* 26(3/4), pp. 207–240, doi:10.3233/FI-1996-263401.

[3] T. Boy de la Tour & R. Echahed (2020): *Combining Parallel Graph Rewriting and Quotient Graphs*. In: *13th International Workshop, WRLA 2020, LNCS* 12328, Springer, pp. 1–18, doi:10.1007/978-3-030-63595-4_1.

[4] T. Boy de la Tour & R. Echahed (2020): *Parallel Coherent Graph Transformations*. In: *Proceedings of WADT 2020, the 25th International Workshop on Algebraic Development Techniques*, LNCS, Springer, to appear, see also CoRR (abs/1904.08850).

[5] T. Boy de la Tour & R. Echahed (2020): *Parallel Rewriting of Attributed Graphs*. *Theoretical Computer Science* 848, pp. 106–132, doi:10.1016/j.tcs.2020.09.025.

[6] A. Corradini, D. Duval, M. Löwe, L. Ribeiro, R. Machado, A. Costa, G. Azzi, J. S. Bezerra & L. M. Rodrigues (2018): *On the Essence of Parallel Independence for the Double-Pushout and Sesqui-Pushout Approaches*. In R. Heckel & G. Taentzer, editors: *Graph Transformation, Specifications, and Nets - In Memory of Hartmut Ehrig, LNCS* 10800, Springer, pp. 1–18, doi:10.1007/978-3-319-75396-6_1.

[7] Andrea Corradini, Ugo Montanari, Francesca Rossi, Hartmut Ehrig, Reiko Heckel & Michael Löwe (1997): *Algebraic Approaches to Graph Transformation - Part I: Basic Concepts and Double Pushout Approach*. In Grzegorz Rozenberg, editor: *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations*, World Scientific, pp. 163–246, doi:10.1142/9789812384720_0003.

[8] A. Costa, J. Bezerra, G. Azzi, L. Rodrigues, T. R. Becker, R. G. Herdt & R. Machado (2016): *Verigraph: A System for Specification and Analysis of Graph Grammars*. In L. Ribeiro & T. Lecomte, editors: *Formal Methods: Foundations and Applications SBMF 2016, LNCS* 10090, Springer, pp. 78–94, doi:10.1007/978-3-319-49815-7_5.

[9] F. Drewes, H.-J. Kreowski & A. Habel (1997): *Hyperedge Replacement Graph Grammars*. In G. Rozenberg, editor: *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations*, World Scientific, pp. 95–162, doi:10.1142/9789812384720_0002.

[10] Dominique Duval, Rachid Echahed, Frédéric Prost & Leila Ribeiro (2014): *Transformation of Attributed Structures with Cloning*. In Stefania Gnesi & Arend Rensink, editors: *Fundamental Approaches to Software Engineering - FASE 2014, LNCS* 8411, Springer, pp. 310–324, doi:10.1007/978-3-642-54804-8_22.

[11] Hartmut Ehrig, Karsten Ehrig, Ulrike Prange & Gabriele Taentzer (2006): *Fundamentals of Algebraic Graph Transformation*. Monographs in Theoretical Computer Science. An EATCS Series, Springer, doi:10.1007/3-540-31188-2.

[12] Hartmut Ehrig & Hans-Jörg Kreowski (1976): *Parallelism of Manipulations in Multidimensional Information Structures*. In: *Mathematical Foundations of Computer Science, LNCS* 45, Springer, pp. 284–293, doi:10.1007/3-540-07854-1_188.

[13] Ole Kniemeyer, Günter Barczik, Reinhard Hemmerling & Winfried Kurth (2007): *Relational Growth Grammars - A Parallel Graph Transformation Approach with Applications in Biology and Architecture*. In: *Third International Symposium AGTIVE, Revised Selected and Invited Papers*, pp. 152–167, doi:10.1007/978-3-540-89020-1_12.

[14] Leen Lambers, Hartmut Ehrig & Fernando Orejas (2008): *Efficient Conflict Detection in Graph Transformation Systems by Essential Critical Pairs*. *Electron. Notes Theor. Comput. Sci* 211, pp. 17–26, doi:10.1016/j.entcs.2008.04.026.

[15] Ivan Lanese & Ugo Montanari (2005): *Synchronization Algebras with Mobility for Graph Transformations*. *Electr. Notes Theor. Comput. Sci* 138(1), pp. 43–60, doi:10.1016/j.entcs.2005.05.004.

[16] D. Plump: *Term Graph Rewriting*. In: *Handbook of Graph Grammars and Computing by Graph Transformation, Volume 2: Applications, Languages and Tools*, doi:10.1142/9789812815149_0001.

[17] Detlef Plump & Sandra Steinert (2004): *Towards Graph Programs for Graph Algorithms*. In: *Second International Conference, ICGT 2004, LNCS* 3256, pp. 128–143, doi:10.1007/978-3-540-30203-2_11.

[18] Barry K. Rosen (1975): *A Church-Rosser Theorem for Graph Grammars*. *Sigact News* 7(3), pp. 26–31, doi:10.1145/1008343.1008344.