Transformation Rules with Nested Application Conditions: Critical Pairs, Initial Conflicts & Minimality

Leen Lambers

Hasso Plattner Institute, University of Potsdam, Germany

Fernando Orejas

Universitat Politècnica de Catalunya, Barcelona, Spain

Abstract

Recently, initial conflicts were introduced in the framework of $\mathcal{M}$-adhesive categories as an important optimization of critical pairs. In particular, they represent a proper subset such that each conflict is represented in a minimal context by a unique initial one. The theory of critical pairs has been extended in the framework of $\mathcal{M}$-adhesive categories to rules with nested application conditions (ACs), restricting the applicability of a rule and generalizing the well-known negative application conditions. A notion of initial conflicts for rules with ACs does not exist yet.

In this paper, on the one hand, we extend the theory of initial conflicts in the framework of $\mathcal{M}$-adhesive categories to transformation rules with ACs. They represent a proper subset again of critical pairs for rules with ACs, and represent each conflict in a minimal context uniquely. They are moreover symbolic because we can show that in general no finite and complete set of conflicts for rules with ACs exists. On the other hand, we show that critical pairs are minimally $\mathcal{M}$-complete, whereas initial conflicts are minimally complete. Finally, we introduce important special cases of rules with ACs for which we can obtain finite, minimally ($\mathcal{M}$-)complete sets of conflicts.

Keywords: Graph Transformation, Critical Pairs, Initial Conflicts, Application Conditions

1. Introduction

Detecting and analyzing conflicts is an important issue in software analysis and design, which has been addressed successfully using powerful techniques from graph transformation (see, e.g., [1, 2, 3, 4]), most of them based on critical pair analysis. The *power of critical pairs* is a consequence of the fact that: a) they are complete, in the sense that they represent all conflicts; b) there is a finite number of them; and c) they can be computed statically. The main problem is that their computation has exponential complexity in the size of the preconditions of the rules. For this reason, some significantly smaller subsets of critical pairs that are still complete have been defined [5, 6, 7], clearing the way for a more efficient computation. In particular, recently, in [6], a new approach for conflict detection was introduced based on a different intuition. Instead of considering conflicts in a minimal context, as for critical pairs, we used the notion of initiality to characterize a complete set of minimal conflicts, showing that *initial conflicts* form a proper subset of critical pairs. In particular, we have that every conflict is represented by a unique initial conflict, as opposed to the fact that each conflict may be represented by many critical pairs.

Most of the work on critical pairs only applies to *plain* graph transformation systems, i.e. transformation systems with unconditional rules. Nevertheless, in practice, we often need to limit the application of rules, defining some kind of *application conditions* (ACs). In this sense, in [8, 3] we defined critical pairs for rules with negative application conditions (NACs), and in [9, 10] for the general case of ACs, where conditions are as expressive as arbitrary first-order formulas on graphs. However, to our knowledge, no work has addressed up to now the problem of finding significantly smaller subsets of critical pairs for this kind of rules.

In this paper we present new results along two different lines. In the first line of work, using the fact that critical pairs are $\mathcal{M}$-initial conflicts ([6]) and through the related notions of completeness and $\mathcal{M}$-completeness, we study the minimality of ($\mathcal{M}$-)complete sets of conflicts. In particular, we show how we can obtain a minimal ($\mathcal{M}$-)complete set of conflicts out of an $\mathcal{M}$-complete set. And, in the second line of work, we generalize the theory of initial conflicts to rules with ACs in the framework of $\mathcal{M}$-adhesive transformation systems. In particular, the main contributions of this paper (as summarized in Table 1 in Section 7) are:

- The definition of the *notion of initial conflict for rules with ACs*, based on a notion of *symbolic transformation pair*, showing that the set of initial conflicts is a *proper subset* of the set of critical pairs and that it is complete[1]. Moreover, as in the plain case, every conflict is an instance of a *unique* initial conflict.

- A *characterization of minimally ($\mathcal{M}$-)complete sets* of transformation pairs w.r.t. parallel dependence, both for plain rules and for rules with ACs, in the sense that, no such set (up to isomorphism) with smaller cardinality exists. In particular, using the notion of ($\mathcal{M}$-)initiality, we show that $\mathcal{M}$-initial conflicts (i.e. critical pairs) are minimally $\mathcal{M}$-complete and initial conflicts are minimally complete.

- A *reduction construction* that allows us to obtain a minimally complete or $\mathcal{M}$-complete set of conflicts $\mathcal{S}$ out of any $\mathcal{M}$-complete set $\mathcal{S}'$ by removing all conflicts that are considered *($\mathcal{M}$-)redundant*. In particular, we present a counter-example that shows that critical pairs for rules with NACs [8, 3] are not minimally $\mathcal{M}$-complete. Using the reduction construction we can however build a minimally complete or minimally $\mathcal{M}$-complete subset from the set

[1] Provided that the considered category has initial conflicts for the plain case.

of critical pairs.

- The identification of a class of so-called *regular* initial conflicts that demonstrate a certain kind of regularity in their application conditions. This allows us to unfold them into a *minimally ($\mathcal{M}$-)complete (and in the case of graphs also finite) set of conflicts.* In particular, we show that, in the case of rules with NACs, initial conflicts are regular, implying that our initial conflicts represent a *conservative extension* of the critical pair theory for rules with NACs.

The paper is organized as follows. We describe *related work* in Section 2 and, in Section 3, we present some *preliminary material*, where we also include some new results. More precisely, in subsection 3.1 and subsection 3.2 we briefly reintroduce the framework of $\mathcal{M}$-adhesive categories and of rules with ACs; in subsection 3.3 we reintroduce critical pairs for rules with ACs following [9, 10]; in subsection 3.4 we reintroduce initial conflicts for plain rules, and in subsection 3.5 we introduce ($\mathcal{M}$-)initial parallel independent transformation pairs and demonstrate their minimal ($\mathcal{M}$-)completeness. This result is used in Section 4, where we present *initial conflicts for rules with ACs*, and show their completeness. Then, in Section 5 we present our results about minimally complete sets of transformation pairs w.r.t. parallel dependence, including the reduction construction to build such sets. Afterwards, in Section 6 we show our results on *unfolding initial conflicts* and on obtaining minimally ($\mathcal{M}-$)complete sets of conflicts for rules with NACs. Finally, we conclude in Section 7 discussing some future work.

This paper is an extended version of [11], presented at ICGT 2020. Apart from including full proofs for all our results, this extended version includes important new results about minimal ($\mathcal{M}$-)completeness. In particular, they have allowed us to show how we can obtain sets of minimally ($\mathcal{M}$-)complete conflicts for graph transformation rules with NACs.

2. Related Work

Most work on checking *confluence* for rule-based rewriting systems is based on the seminal paper from Knuth and Bendix [12], who reduced the problem of checking local confluence to checking the joinability of a finite set of *critical pairs* obtained from superposing or overlapping the left hand sides of pairs of rewriting rules. This technique has been extensively studied and applied in the area of term rewriting systems (see, for instance, [13]), and it was introduced in the area of *graph transformation* by Plump [14, 15, 16] in the context of term-graph and hypergraph rewriting. Moreover, he also proved that (local) confluence of graph transformation systems is undecidable, even for terminating systems, as opposed to what happens in the area of term rewriting systems. However, recently, in [17] it is shown that confluence of terminating DPO transformation of graphs with interfaces is decidable. The authors explain that the reason is that interfaces play the same role as variables in term rewriting systems, where confluence is undecidable for terminating ground (i.e., without variables) systems, but decidable for non-ground ones.

The notion of critical pairs in the area of graph transformation, as introduced by Plump [14, 15, 16], has the characteristic that their computation is exponential in the size of the preconditions of the rules. For this reason, different *proper subsets of critical pairs* with a considerably reduced size were studied that are still complete [5, 6, 7], clearing the way for a more efficient computation. The notion of *essential critical pair* [5] for graph transformation systems already allowed for a significant reduction, and, the notion of *initial conflict* [6], introduced for the more general $\mathcal{M}$-adhesive systems, allowed for an even larger reduction. A problem with initial conflicts is that not all $\mathcal{M}$-adhesive categories necessarily have them. In this sense, in [6] it is shown that, in particular, typed graphs have initial conflicts and, a bit later, [7] extended that result proving that arbitrary $\mathcal{M}$-adhesive categories satisfying some given conditions also have initial conflicts. Moreover, they provided a simple way of constructing the initial pair of transformations for a given conflict.

A recent line of work concentrates on the development of *multi-granular conflict detection techniques* [18, 4, 19]. In particular, an extensive literature survey shows [4] that conflict detection is used at different levels of granularity depending on its application field. The overview shows that conflict detection can be used for the analysis and design phase of software systems (e.g. for finding inconsistencies in requirement specifications), for model-driven engineering (e.g. supporting model version management), for testing (e.g. generation of interesting test cases), or for optimizing rule-based computations (e.g. avoiding backtracking). These multi-granular techniques are presented for rules without application conditions (ACs). Our work builds further foundations for providing multi-granular techniques also in the case of rules with ACs in the future.

The use of (negative) *application conditions* and of graph constraints, to limit the application of graph transformation rules, was introduced in [20, 21, 22]. Based on this notion of graph constraints, in [23], Rensink presented a logic for expressing graph properties, closely related to the logic of nested conditions of Habel and Penneman [24], shown to have the same expressive power as first-order logic on graphs, and being (refutationally) complete as demonstrated in Lambers and Orejas [25]. Checking confluence for graph transformation systems with application conditions (ACs) has been studied in [8, 3] for the case of negative application conditions (NACs), and in [9, 10] for the more general case of ACs. Moreover, Bruggink et al. generalized the Local Confluence Theorem to conditional reactive systems [26], a general abstract framework for rewriting, in which reactive systems à la Leifer and Milner are enriched with ACs. In the case of rules with ACs, it is an open issue to also come up with proper subsets of critical pairs of considerably reduced size (analogous to the previously mentioned works for rules without ACs).

3. Preliminaries

We start with a very brief introduction of $\mathcal{M}$-adhesive categories. We then revisit *rules with nested application conditions (ACs)* (cf. subsection 3.2) as well as the main parts of *critical pair theory* for this type of rules [9, 10] (cf. subsection 3.3). Thereafter, we reintroduce the notion of *initial conflicts* [6] for *plain* rules, i.e. rules without nested application conditions (cf. subsection 3.4). We also introduce the notion of *($\mathcal{M}$-)initial parallel independent transformation pairs* as a counterpart (cf. subsection 3.5) to ($\mathcal{M}$-)initial conflicts. They play a particular role when defining initial conflicts for rules with ACs in subsection 4.3 and for showing that critical pairs for rules with ACs actually coincide with $\mathcal{M}$-initial conflicts. We assume that the reader is acquainted with the basic theory of DPO graph transformation and, in particular, the standard definitions of typed graphs and typed graph morphisms (see, e.g., [27]) and its associated category, $\mathbf{Graphs}_{TG}$.

3.1. Graphs & High-Level Structures

The results presented in this paper do not only apply to a specific class of graph transformation systems, like standard (typed) graph transformation systems, but to systems over any $\mathcal{M}$-adhesive category [28]. The idea behind the consideration of $\mathcal{M}$-adhesive categories is to avoid similar investigations for different instantiations like e.g. different kinds of graphs, Petri nets, hypergraphs, and algebraic specifications. An $\mathcal{M}$-adhesive category is a category $\mathcal{C}$ with a distinguished morphism class $\mathcal{M}$ of monomorphisms satisfying certain properties. The most important one is the (weak) van Kampen (VK) property stating a certain kind of compatibility of pushouts and pullbacks along $\mathcal{M}$-morphisms.

Definition 1 ($\mathcal{M}$-adhesive category)**.** *An* $\mathcal{M}$-adhesive category $(\mathcal{C}, \mathcal{M})$ *consists of a category* $\mathcal{C}$ *and a class* $\mathcal{M}$ *of monomorphisms in* $\mathcal{C}$ *such that the following properties hold:*

1. *$\mathcal{M}$ is closed under isomorphisms ($f \in \mathcal{M}$, g isomorphism (or vice versa) implies $g \circ f \in \mathcal{M}$), composition ($f, g \in \mathcal{M}$ implies $g \circ f \in \mathcal{M}$), and decomposition ($g \circ f \in \mathcal{M}$, $g \in \mathcal{M}$ implies $f \in \mathcal{M}$).*
2. *$\mathcal{C}$ has pushouts and pullbacks along $\mathcal{M}$-morphisms, i.e. pushouts and pullbacks, where at least one of the given morphisms is in $\mathcal{M}$, and $\mathcal{M}$-morphisms are closed under pushouts and pullbacks, i.e. given a pushout (1) as in the figure below, $m \in \mathcal{M}$ implies $n \in \mathcal{M}$ and, given a pullback (1), $n \in \mathcal{M}$ implies $m \in \mathcal{M}$.*
3. *Pushouts in $\mathcal{C}$ along $\mathcal{M}$-morphisms are vertical weak van Kampen (VK) squares, short $\mathcal{M}$-VK squares, i.e. for any commutative cube in $\mathcal{C}$ where we have a pushout with $m \in \mathcal{M}$ in the bottom, $b, c, d \in \mathcal{M}$ and the back faces are pullbacks, it holds: the top is pushout iff the*

front faces are pullbacks.

$$
\begin{tikzcd}
A \arrow[r] \arrow[d, hook, "m"'] \arrow[dr, phantom, "(1)"] & C \arrow[d, hook, "n"] \\
B \arrow[r] & D
\end{tikzcd}
\qquad
\begin{tikzcd}[row sep=small, column sep=small]
& A' \arrow[rr] \arrow[dl] \arrow[dd] & & C' \arrow[dl] \arrow[dd, hook, "c"] \\
B' \arrow[rr, crossing over] \arrow[dd, hook, "b"'] & & D' & \\
& A \arrow[rr, dashed, "f"] \arrow[dl, hook, "m"'] & & C \arrow[dl, hook] \\
B \arrow[rr] & & D \arrow[from=uu, crossing over, hook, "d"'] &
\end{tikzcd}
$$

Moreover, the results in this paper require an $\mathcal{M}$-adhesive category where additional properties hold. In particular, we require that our categories have binary coproducts (for the results concerned with $\mathcal{M}$-initiality), initial pushouts (for the Local Confluence Theorem), describing the existence of a special "smallest" pushout over a morphism, and $\mathcal{E}'$-$\mathcal{M}$ pair factorizations (for the results concerned with shifting application conditions as well as initiality), extending the classical epi-mono factorization to a pair of morphisms with the same codomain:

Definition 2 (Initial pushouts, $\mathcal{E}'$-$\mathcal{M}$ pair factorizations)**.** *Let $\langle \mathcal{C}, \mathcal{M} \rangle$ be an $\mathcal{M}$-adhesive category.*

1. *$\langle \mathcal{C}, \mathcal{M} \rangle$ has* initial pushouts over $\mathcal{M}$-morphisms*, i.e., for every morphism $f\colon A \hookrightarrow A'$ with $f \in \mathcal{M}$, there exists an initial pushout over f. A morphism $b\colon B \to A \in \mathcal{M}$ is a* boundary *over f if there is a pushout complement of f and b such that (1) in the diagram below is an initial pushout over f.* Initiality *of (1) over f means that, for every pushout (3) with $b' \in \mathcal{M}$, there exist unique morphisms $b^*, c^* \in \mathcal{M}$ such that $b' \circ b^* = b$, $c' \circ c^* = c$ and (2) is a pushout. B is called the* boundary *object and C the* context *with respect to f.*
2. *$\langle \mathcal{C}, \mathcal{M} \rangle$ has a* unique $\mathcal{E}'$-$\mathcal{M}$ pair factorization *for a given class of morphism pairs $\mathcal{E}'$ with the same codomain, i.e., for each pair of morphisms $f_1\colon A_1 \to C$ and $f_2\colon A_2 \to C$, there exist a unique (up to isomorphism) object K and unique (up to isomorphism) morphisms $e_1\colon A_1 \to K$, $e_2\colon A_2 \to K$, and $m\colon K \hookrightarrow C$ with $(e_1, e_2) \in \mathcal{E}'$ and $m \in \mathcal{M}$ such that $m \circ e_1 = f_1$ and $m \circ e_2 = f_2$. Notice that this means that if $(f_1\colon A_1 \to C, f_2\colon A_2 \to C) \in \mathcal{E}'$, then (f_1, f_2, id_C) is its pair factorization.*

$$
\begin{tikzcd}
B \arrow[r, hook, "b"] \arrow[d] \arrow[dr, phantom, "(1)"] & A \arrow[d, "f"] \\
C \arrow[r, hook, "c"'] & A'
\end{tikzcd}
\qquad
\begin{tikzcd}
B \arrow[r, hook, "b^*"] \arrow[d] \arrow[rr, bend left, hook, "b"] \arrow[dr, phantom, "(2)"] & D \arrow[r, hook, "b'"] \arrow[d] \arrow[dr, phantom, "(3)"] & A \arrow[d, "f"] \\
C \arrow[r, hook, "c^*"'] \arrow[rr, bend right, hook, "c"'] & E \arrow[r, hook, "c'"'] & A'
\end{tikzcd}
\qquad
\begin{tikzcd}
A_1 \arrow[dr, "e_1"'] \arrow[drr, bend left, "f_1"] & & \\
& K \arrow[r, hook, "m"] & C \\
A_2 \arrow[ur, "e_2"] \arrow[urr, bend right, "f_2"'] & &
\end{tikzcd}
$$

Assumption 1. *We assume that $\langle \mathcal{C}, \mathcal{M} \rangle$ is an $\mathcal{M}$-adhesive category with a unique $\mathcal{E}'$-$\mathcal{M}$ pair factorization and binary coproducts. For the Local Confluence Theorem for initial conflicts of rules with ACs* [2]*, we in addition need initial pushouts (cf. subsection 4.4).*

[2] Although it is a straightforward generalization of the one for critical pairs, we do not explicitly state it in this paper, since we concentrate on the study of critical pairs, initial conflicts and minimal completeness here.

Remark 1 ($\langle \text{Graphs}_{TG}, \mathcal{M} \rangle$, $\langle \text{PTNets}, \mathcal{M} \rangle$, $\langle \text{Spec}, \mathcal{M}_{strict} \rangle$ are $\mathcal{M}$-adhesive and satisfy additional properties [27, 28])**.** *In particular, the category $\langle \text{Graphs}_{TG}, \mathcal{M} \rangle$ with the class $\mathcal{M}$ of all injective typed graph morphisms is an $\mathcal{M}$-adhesive category. It has a unique $\mathcal{E}'$-$\mathcal{M}$ pair factorization where $\mathcal{E}'$ is the class of jointly surjective typed graph morphism pairs (i.e., the morphism pairs (e_1, e_2) such that for each $x \in K$ there is a pre-image $a_1 \in A_1$ with $e_1(a_1) = x$ or $a_2 \in A_2$ with $e_2(a_2) = x$). Binary coproduct objects correspond to disjoint unions of graphs. All other examples are also $\mathcal{M}$-adhesive categories and satisfy the additional properties for suitable choices of $\mathcal{M}$ and $\mathcal{E}'$.*

3.2. Rules with Application Conditions and Parallel Independence

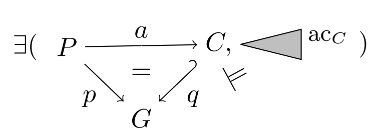
We reintroduce nested application conditions [24] (in short, application conditions, or just ACs) following [10]. They generalize the corresponding notions in [22, 2, 29], where a negative (positive) application condition, short NAC (PAC), over a graph P, denoted $\neg\exists a$ ($\exists a$) is defined in terms of a morphism $a : P \to C$. Informally, a morphism $m : P \to G$ satisfies $\neg\exists a$ ($\exists a$) if there does not exist a morphism $q : C \to G$ extending a to m (if there exists q extending a to m). Then, an AC (also called *nested AC*) is either the special condition true or a pair of the form $\exists(a, \text{ac}_C)$ or $\neg\exists(a, \text{ac}_C)$, where the first case corresponds to a PAC and the second case to a NAC, and in both cases ac_C is an additional AC on C. Intuitively, a morphism $m : P \to G$ satisfies $\exists(a, \text{ac}_C)$ if m satisfies a and the corresponding extension q satisfies ac_C. Moreover, ACs (and also NACs and PACs) may be combined with the usual logical connectors.

Definition 3 (application condition and satisfaction)**.** *An* application condition ac_P *over an object P is inductively defined as follows:*

- *For every morphism $a \colon P \to C$ and every application condition ac_C over C, $\exists(a, \text{ac}_C)$ is an application condition over P.*
- *For application conditions c, c_i over P with $i \in I$ (for finite index sets I), $\neg c$ and $\wedge_{i \in I} c_i$ are application conditions over P.*

We define inductively when a morphism satisfies *an application condition:*

- *A morphism $p \colon P \to G$ satisfies an application condition $\exists(a, \text{ac}_C)$, denoted $p \models \exists(a, \text{ac}_C)$, if there exists an $\mathcal{M}$-morphism q such that $q \circ a = p$ and $q \models \text{ac}_C$.*
- *A morphism $p \colon P \to G$ satisfies $\neg c$ if p does not satisfy c and satisfies $\wedge_{i \in I} c_i$ if it satisfies each c_i ($i \in I$).*



Note that the empty conjunction (equivalent to true), satisfied by each morphism, serves as base case in the inductive definition. Moreover, $\exists a$ (resp. $\forall(a, \mathrm{ac}_C)$) abbreviates $\exists(a, \mathrm{true})$ (resp. $\neg\exists(a, \neg\mathrm{ac}_C)$).

ACs are used to restrict the application of rules to a given object. The idea is to equip the precondition (or left hand side) of rules with an application condition[3]. Then we can only apply a given rule to an object G if the corresponding match morphism satisfies the AC of the rule. However, for technical reasons[4], we also introduce the application of rules *disregarding* the associated ACs.

Definition 4 (rules and transformations)**.** *A* rule $\rho = \langle p, \mathrm{ac}_L \rangle$ *consists of a* plain *rule* $p = \langle L \hookleftarrow I \hookrightarrow R \rangle$ *with* $I \hookrightarrow L$ *and* $I \hookrightarrow R$ *morphisms in* $\mathcal{M}$ *and an application condition* ac_L *over* L*.*

$$\begin{tikzcd} \mathrm{ac}_L \arrow[r, phantom, "\triangleright"] & L \arrow[d, "m"'] \arrow[dr, phantom, "(1)"] & I \arrow[l, hook'] \arrow[r, hook] \arrow[d] \arrow[dr, phantom, "(2)"] & R \arrow[d, "m^*"] \\ & G & D \arrow[l, hook'] \arrow[r, hook] & H \end{tikzcd}$$

A direct transformation $t : G \Rightarrow_{\rho,m,m^*} H$ *consists of two pushouts (1) and (2), called DPO, with match* m *and comatch* m^* *such that* $m \models \mathrm{ac}_L$*.* $G \hookleftarrow D \hookrightarrow H$ *is called the* derived span *of* t*. An* AC-disregarding direct transformation $G \Rightarrow_{\rho,m,m^*} H$ *consists of DPO (1) and (2), where* m *does not necessarily need to satisfy* ac_L*. Given a set of rules* $\mathcal{R}$ *for* $\langle \mathcal{C}, \mathcal{M} \rangle$*, the triple* $\langle \mathcal{C}, \mathcal{M}, \mathcal{R} \rangle$ *is an* $\mathcal{M}$-adhesive system*.*

Remark 2. *In the rest of the paper we assume that each rule (resp. transformation or $\mathcal{M}$-adhesive system) comes with ACs. Otherwise, we state that we have a* plain *rule (resp. transformation or $\mathcal{M}$-adhesive system). This plain case can also be seen as a special case of a rule (resp. transformation or $\mathcal{M}$-adhesive system) with ACs in the sense that the ACs are (equivalent to) true.*

ACs can be shifted over morphisms and rules (from right to left and vice versa) as shown in the following lemma (for constructions see [30] [5] and [24, 30], respectively). We only describe the right to left case in Lemma 2, since the left to right case is symmetrical.

Lemma 1 (shift ACs over morphisms [30])**.** *For each morphism* $b\colon P \to P'$ *and application condition* ac_P*, there is a construction* Shift *translating morphisms and application conditions to application conditions (as inductively defined below) such that for each morphism* $n\colon P' \to H$ *it*

[3]We could have also allowed to equip the right-hand side of rules with an additional AC, but this case can be reduced to rules with left ACs only as shown in Lemma 2.

[4]For example, symbolic transformation pairs as introduced later, or also critical pairs for rules with ACs (see Definition 8) consist of transformations that do not need to satisfy the associated ACs.

[5]Since this construction entails the enumeration of jointly epimorphic morphism pairs, its computation has exponential complexity in the size of the precondition of the rule and the size of the AC.

holds that $n \circ b \models \mathrm{ac}_P \Leftrightarrow n \models \mathrm{Shift}(b, \mathrm{ac}_P)$.

$$\begin{tikzcd} P \arrow[r, "b"] \arrow[d, "a"'] \arrow[dr, phantom, "(1)"] & P' \arrow[d, "a'"] \\ C \arrow[r, hook, "b'"'] \arrow[d, phantom, "\triangle"] & C' \\ \mathrm{ac}_C & \end{tikzcd}$$

$\mathrm{Shift}(b, \exists(a, \mathrm{ac}_C)) = \bigvee_{(a',b') \in \mathcal{F}} \exists(a', \mathrm{Shift}(b', \mathrm{ac}_C))$

if $\mathcal{F} = \{(a', b') \in \mathcal{E}' \mid b' \in \mathcal{M}$ *and (1) commutes*$\} \neq \emptyset$

$\mathrm{Shift}(b, \exists(a, \mathrm{ac}_C)) = \mathrm{false}$ *if* $\mathcal{F} = \emptyset$.

For Boolean formulas over ACs, Shift *is extended in the usual way.*

Lemma 2 (shift ACs over rules [24, 30])**.** *For each application condition* ac_R *on* R *of a rule* ρ*, there is a construction* L *translating rules and application conditions to application conditions (as inductively defined below) such that for every* $G \Rightarrow_{\rho,m,m^*} H$ *it holds that* $m \models \mathrm{L}(\rho, \mathrm{ac}_R) \Leftrightarrow m^* \models \mathrm{ac}_R$.

$$\begin{tikzcd} L \arrow[d, "b"'] \arrow[dr, phantom, "(2)"] & K \arrow[l, hook', "l"'] \arrow[r, hook, "r"] \arrow[d] \arrow[dr, phantom, "(1)"] & R \arrow[d, "a"] \\ Y \arrow[d, phantom, "\triangle"] & Z \arrow[l, hook', "l^*"] \arrow[r, hook, "r^*"'] & X \arrow[d, phantom, "\triangle"] \\ \mathrm{L}(\rho^*, \mathrm{ac}_X) & & \mathrm{ac}_X \end{tikzcd}$$

$\mathrm{L}(\rho, \exists(a, \mathrm{ac}_X)) = \exists(b, \mathrm{L}(\rho^*, \mathrm{ac}_X))$ *if* $\langle r, a \rangle$ *has a pushout complement (1) and* $\rho^* = \langle Y \hookleftarrow Z \hookrightarrow X \rangle$ *is the derived rule by constructing the pushout (2).*

$\mathrm{L}(\rho, \exists(a, \mathrm{ac}_X)) = \mathrm{false}$*, otherwise.*

For Boolean formulas over ACs, L *is extended in the usual way.*

For *parallel independence*, when working with rules with ACs, we need not only that each rule does not delete any element which is part of the match of the other rule, but also that the resulting transformation defined by each rule application still satisfies the ACs of the other rule application.

Definition 5 (transformation pairs and parallel independence)**.** *A* transformation pair $H_1 \Leftarrow_{\rho_1,o_1} G \Rightarrow_{\rho_2,o_2} H_2$ *is* parallel independent *if there exist morphisms* $d_{12}\colon L_1 \to D_2$ *and* $d_{21}\colon L_2 \to D_1$ *such that* $k_2 \circ d_{12} = o_1$*,* $c_2 \circ d_{12} \models \mathrm{ac}_{L_1}$*,* $k_1 \circ d_{21} = o_2$*, and* $c_1 \circ d_{21} \models \mathrm{ac}_{L_2}$.

$$\begin{tikzcd} & & \mathrm{ac}_{L_1} & \mathrm{ac}_{L_2} & & \\ R_1 \arrow[d] & I_1 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & L_1 \arrow[u, phantom, "\triangleleft"] \arrow[dr, "o_1"'] \arrow[drr, dotted, "d_{12}"] & L_2 \arrow[u, phantom, "\triangleright"] \arrow[d, "o_2"] \arrow[dll, dotted, "d_{21}"'] & I_2 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & R_2 \arrow[d] \\ H_1 & D_1 \arrow[l, hook', "c_1"] \arrow[rr, hook, "k_1"'] & & G & D_2 \arrow[l, hook', "k_2"] \arrow[r, hook, "c_2"'] & H_2 \end{tikzcd}$$

We say that a transformation pair is *in conflict*, *conflicting* or also *parallel dependent* if it is not parallel independent. We distinguish different conflict types, generalizing straightforwardly the conflict characterization introduced for rules with NACs [8]. The transformation pair $H_1 \Leftarrow_{\rho_1,o_1} G \Rightarrow_{\rho_2,o_2} H_2$ is a *use-delete* (resp. *delete-use*) conflict if in Definition 5 the commuting morphism d_{12} (resp. d_{21}) does not exist, i.e. the second (resp. first) rule deletes something used by the first (resp. second) one. Moreover, it is an *AC-produce* (resp. *produce-AC*) conflict if in Definition 5 the commuting morphism d_{12} (resp. d_{21}) exists, but an extended match is produced by the second (resp. first) rule that does not satisfy the rule AC of the first (resp. second) rule. If a transformation pair is an *AC-produce* or *produce-AC* conflict, then we also say that it is an *AC conflict* or *AC conflicting*.

Remark 3 (use-delete XOR AC-produce)**.** *A use-delete (resp. delete-use) conflict cannot occur simultaneously to an AC-produce (resp. produce-AC) conflict. This is because the AC of the first (resp. second) rule can only be violated if there exists an extended match for the first (resp. second) rule. However, a use-delete (resp. delete-use) conflict may occur simultaneously to a produce-AC (resp. AC-produce) conflict, since in this case the extended match for the first (resp. second) rule does not exist, whereas the extended match for the second (resp. first) rule exists and violates the AC, i.e. both conflict types occur on opposite sides of the diagram in Definition 5.*

For grasping the notion of completeness of transformation pairs w.r.t. a property like parallel (in-)dependence, it is first important to understand how two transformations via the same rules can be related via extension diagrams. In particular, an *extension diagram* consists of two transformation sequences $t\colon G_0 \Rightarrow^* G_n$ and $t'\colon G'_0 \Rightarrow^* G'_n$ via the same rules and *extension morphism* $k_0\colon G_0 \to G'_0$ that maps G_0 to G'_0 as shown in the following diagram on the left. For each rule application and direct transformation, we have two double pushout diagrams as shown on the right, where the rule ρ_{i+1} is applied to both G_i and G'_i. We also say that t is *extended to* t', or that t' *extends* t via the extension morphism k_0 and the corresponding extension diagram. Moreover, given a *transformation pair* $tp : H_1 \Leftarrow_{\rho_1,m_1} G \Rightarrow_{\rho_2,m_2} H_2$ and $tp' : H'_1 \Leftarrow_{\rho_1,m'_1} G' \Rightarrow_{\rho_2,m'_2} H'_2$ we say that tp extends to tp' (or tp' extends tp) if they are related via extension diagrams and some common extension morphism $f : G \to G'$.

$$
\begin{tikzcd}
G_0 \arrow[r, Rightarrow, "*"] \arrow[d, "k_0"'] \arrow[dr, phantom, "(1)"] & G_n \arrow[d, "k_n"] \\
G'_0 \arrow[r, Rightarrow, "*"] & G'_n
\end{tikzcd}
\qquad
\begin{tikzcd}
L_{i+1} \arrow[d] & I_{i+1} \arrow[l] \arrow[r] \arrow[d] & R_{i+1} \arrow[d] \\
G_i \arrow[d] & D_i \arrow[l] \arrow[r] \arrow[d] & G_{i+1} \arrow[d] \\
G'_i & D'_i \arrow[l] \arrow[r] & G'_{i+1}
\end{tikzcd}
$$

We introduce two different notions of completeness, distinguishing $\mathcal{M}$-completeness from regular completeness, depending on the membership of the extension morphism in $\mathcal{M}$.

Definition 6 (($\mathcal{M}$-)completeness of transformation pairs)**.** *A set of* transformation pairs $\mathcal{S}$ *for a pair of rules* $\langle\rho_1, \rho_2\rangle$ *is* complete *(resp.* $\mathcal{M}$-complete*) w.r.t. parallel (in-)dependence if each parallel (in-)dependent direct transformation pair* $H_1 \Leftarrow_{\rho_1,m_1} G \Rightarrow_{\rho_2,m_2} H_2$ *extends some pair* $P_1 \Leftarrow_{\rho_1,o_1} K \Rightarrow_{\rho_2,o_2} P_2$ *from* $\mathcal{S}$ *via some extension morphism* $m : K \to G$ *(resp.* $m \in \mathcal{M}$*).*

$$
\begin{tikzcd}
P_1 \arrow[d] & K \arrow[l, Leftarrow, "{\rho_1, o_1}"'] \arrow[r, Rightarrow, "{\rho_2, o_2}"] \arrow[d, "m"] & P_2 \arrow[d] \\
H_1 & G \arrow[l, Leftarrow, "{\rho_1, m_1}"] \arrow[r, Rightarrow, "{\rho_2, m_2}"'] & H_2
\end{tikzcd}
$$

Figure 1: ($\mathcal{M}$-)completeness of transformation pairs

It is known that critical pairs (resp. initial conflicts) for *plain rules* are $\mathcal{M}$-complete (resp. complete) w.r.t. parallel dependence [27, 6]. In subsection 3.3, we reintroduce the fact that critical pairs for rules with ACs are $\mathcal{M}$-complete w.r.t. parallel dependence, but as symbolic transformation pairs. We learn in Section 4 that initial conflicts for rules with ACs are also complete in this symbolic way.

3.3. Critical Pairs

Critical pairs for plain rules are just parallel dependent transformation pairs, where morphisms o_1 and o_2 are in $\mathcal{E}'$ (i.e., roughly, K is an overlapping of L_1 and L_2). In the category of **Graphs** they lead to finite and $\mathcal{M}$-complete sets of finite conflicts [27] (assuming that the rule graphs are also finite).

When rules include ACs, we cannot use the same notion of critical pair since, as we show in Theorem 4, in general, for any two rules with ACs, there is no complete set of transformation pairs that is finite. To avoid this problem, our critical pairs for rules with ACs are symbolic and include special ACs, as in [9, 10], where they are proved to be $\mathcal{M}$-*complete*. They are moreover finite in the category of **Graphs** (assuming again that the rules are finite).

In particular, critical pairs are based on the notion of *symbolic transformation pairs*, which are pairs of *AC-disregarding transformations* on some object K with two special ACs on K. These two ACs, ac_K (*extension AC*) and ac^*_K (*conflict-inducing AC*), are used to characterize which extensions of this pair, via some morphism $m : K \to G$, give rise to a transformation pair that is parallel dependent. If $m \models \mathrm{ac}_K$, then $m \circ o_1 : L_1 \to G$ and $m \circ o_2 : L_2 \to G$ are two morphisms, satisfying the associated ACs of ρ_1 and ρ_2, respectively. Moreover, if $m \models \mathrm{ac}^*_K$, then the two transformations $H_1 \Leftarrow_{\rho_1, m\circ o_1} G \Rightarrow_{\rho_2, m\circ o_2} H_2$ are parallel dependent. Symbolic transformation pairs allow us to present critical pairs as well as initial conflicts (cf. subsection 3.4) in a compact and unified way, since they both are instances of symbolic transformation pairs. Finally, note that each symbolic transformation pair $stp_K : \langle tp_K, ac_K, ac^*_K\rangle$ is by definition uniquely determined (up to isomorphism and equivalence of the extension AC and conflict-inducing AC) by its underlying AC-disregarding transformation pair.

Definition 7 (symbolic transformation pair)**.** *Given rules* $\rho_1 = \langle p_1, \mathrm{ac}_{L_1}\rangle$ *and* $\rho_2 = \langle p_2, \mathrm{ac}_{L_2}\rangle$*, a* symbolic transformation pair $stp_K : \langle tp_K, ac_K, ac^*_K\rangle$ *for* $\langle \rho_1, \rho_2\rangle$ *consists of a pair* $tp_K : P_1 \Leftarrow_{\rho_1, o_1} K \Rightarrow_{\rho_2, o_2} P_2$ *of AC-disregarding transformations together with ACs* ac_K *and* ac^*_K *on* K *given by:*

$\mathrm{ac}_K = \mathrm{Shift}(o_1, \mathrm{ac}_{L_1}) \wedge \mathrm{Shift}(o_2, \mathrm{ac}_{L_2})$, *called* extension AC, *and*

$\mathrm{ac}^*_K = \neg(\mathrm{ac}^*_{K,d_{12}} \wedge \mathrm{ac}^*_{K,d_{21}})$, *called* conflict-inducing AC

with $\mathrm{ac}^*_{K,d_{12}}$ *and* $\mathrm{ac}^*_{K,d_{21}}$ *given as follows:*

$$
\begin{array}{ll}
\textit{if } (\exists\, d_{12} \textit{ with } k_2{\circ}d_{12}{=}o_1) & \textit{then } \mathrm{ac}^*_{K,d_{12}} = \mathrm{L}(p_2^*, \mathrm{Shift}(c_2{\circ}d_{12}, \mathrm{ac}_{L_1})) \\
 & \textit{else } \mathrm{ac}^*_{K,d_{12}} = \mathrm{false} \\
\textit{if } (\exists\, d_{21} \textit{ with } k_1{\circ}d_{21}{=}o_2) & \textit{then } \mathrm{ac}^*_{K,d_{21}} = \mathrm{L}(p_1^*, \mathrm{Shift}(c_1{\circ}d_{21}, \mathrm{ac}_{L_2})) \\
 & \textit{else } \mathrm{ac}^*_{K,d_{21}} = \mathrm{false}
\end{array}
$$

where $p_1^* = \langle K \xleftarrow{k_1} D_1 \xhookrightarrow{c_1} P_1 \rangle$ *and* $p_2^* = \langle K \xleftarrow{k_2} D_2 \xhookrightarrow{c_2} P_2 \rangle$ *are defined by the corresponding double pushouts.*

$$
\begin{tikzcd}
p_1 : & R_1 \arrow[d] & I_1 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & L_1 \arrow[r, phantom, "\triangleleft\, \mathrm{ac}_{L_1}"] \arrow[dr, "o_1"'] \arrow[drrr, dotted, "d_{12}"] & & L_2 \arrow[l, phantom, "\mathrm{ac}_{L_2}\, \triangleright"] \arrow[dl, "o_2"] \arrow[dlll, dotted, "d_{21}"'] & I_2 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & R_2 \arrow[d] & : p_2 \\
p_1^* : & P_1 & D_1 \arrow[l, hook', "c_1"] \arrow[rr, hook, "k_1"'] & & K & & D_2 \arrow[ll, hook', "k_2"] \arrow[r, hook, "c_2"'] & P_2 & : p_2^*
\end{tikzcd}
$$

A *critical pair*[6] is now a symbolic transformation pair in a minimal context such that it can be extended to at least one pair of transformations being parallel dependent (or conflict).

Definition 8 (critical pair)**.** *Given rules* $\rho_1 = \langle p_1, \mathrm{ac}_{L_1} \rangle$ *and* $\rho_2 = \langle p_2, \mathrm{ac}_{L_2} \rangle$*, a* critical pair *for* $\langle \rho_1, \rho_2 \rangle$ *is a symbolic transformation pair* $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$*, where the match pair* (o_1, o_2) *of* tp_K *is in* $\mathcal{E}'$*, and there exists a morphism* $m\colon K \to G \in \mathcal{M}$ *such that* $m \models \mathrm{ac}_K \wedge \mathrm{ac}^*_K$ *and* $m_i = m \circ o_i$*, for* $i = 1, 2$*, satisfy the gluing conditions, i.e.* m_i *has a pushout complement w.r.t.* p_i*.*

Note that critical pairs for rules with ACs represent a conservative extension of critical pairs for plain rules in the following sense. Each critical pair tp_K for the plain rules $\langle p_1, p_2 \rangle$ corresponds uniquely to a critical pair $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ for $\langle \rho_1, \rho_2 \rangle$ with $\rho_1 = \langle p_1, \mathrm{ac}_{L_1} \rangle$ and $\rho_2 = \langle p_2, \mathrm{ac}_{L_2} \rangle$ such that ac_{L_1} and ac_{L_2} are true. This is because ac_K and ac^*_K are true, since either $\mathrm{ac}^*_{K,d_{12}}$ or $\mathrm{ac}^*_{K,d_{21}}$ needs to be false with tp_K a use-delete/delete-use conflict.

Definition 9 (($\mathcal{M}$-)completeness of symbolic transformation pairs)**.** *A set of symbolic transformation pairs* $\mathcal{S}$ *for a pair of rules* $\langle \rho_1, \rho_2 \rangle$ *is* complete *(resp.* $\mathcal{M}$-complete*) w.r.t. parallel dependence if each parallel dependent direct transformation* $H_1 \Leftarrow_{\rho_1, m_1} G \Rightarrow_{\rho_2, m_2} H_2$ *extends some symbolic transformation pair* $stp_K : \langle tp_K : P_1 \Leftarrow_{\rho_1, o_1} K \Rightarrow_{\rho_2, o_2} P_2, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ *from* $\mathcal{S}$ *as depicted in Figure 1 with extension morphism* $m : K \to G$ *(resp.* $m : K \to G \in \mathcal{M}$*) and* $m \models \mathrm{ac}_K \wedge \mathrm{ac}^*_K$*.*

[6] A symbolic transformation pair with matches belonging to $\mathcal{E}'$ is called a weak critical pair in [9, 10]

Theorem 1 ($\mathcal{M}$-completeness of critical pairs [9, 10])**.** *The set of* critical pairs *for a pair of rules $\langle \rho_1, \rho_2 \rangle$ is* $\mathcal{M}$-complete *w.r.t. parallel dependence. Moreover, each critical pair $P_1 \Leftarrow_{\rho_1,o_1} K \Rightarrow_{\rho_2,o_2} P_2$ for $\langle \rho_1, \rho_2 \rangle$ extends to a parallel dependent pair $H_1 \Leftarrow_{\rho_1,m_1} G \Rightarrow_{\rho_2,m_2} H_2$ via extension morphism $m\colon K \to G \in \mathcal{M}$ such that $m \models \mathrm{ac}_K \wedge \mathrm{ac}^*_K$.*

Note that based on $\mathcal{M}$-completeness it is possible to formulate also a Local Confluence Theorem for critical pairs of rules with ACs for $\mathcal{M}$-adhesive categories with $\mathcal{M}$-initial pushouts [9, 10].

3.4. Initial Conflicts for Plain Rules

Initial conflicts for plain rules follow an alternative approach to the original idea of critical pairs. Instead of considering all conflicting transformations in a minimal context (materialized by a pair of jointly epimorphic matches), initial conflicts use the notion of *initiality of transformation pairs* to obtain a more declarative view on the minimal context of critical pairs. Each initial conflict is a critical pair but not the other way round. Moreover, all initial conflicts for plain rules are complete w.r.t. parallel dependence and they still satisfy the Local Confluence Theorem for plain rules. Consequently, initial conflicts for plain rules represent an important, proper subset of critical pairs for performing static conflict detection as well as local confluence analysis. One contribution of this paper is to demonstrate how to achieve a similar situation for rules with ACs.

Definition 10 (($\mathcal{M}$-)initial transformation pair)**.** *Given a pair of plain direct transformations $tp : H_1 \Leftarrow_{p_1,m_1} G \Rightarrow_{p_2,m_2} H_2$, then $tp^I : H^I_1 \Leftarrow_{p_1,m^I_1} G^I \Rightarrow_{p_2,m^I_2} H^I_2$ is an* initial transformation pair *(resp. $\mathcal{M}-$initial transformation pair) for tp if it can be extended to tp via extension diagrams (1) and (2) and extension morphism (resp. $\mathcal{M}$-morphism) f^I as in Figure 2 such that for each transformation pair $tp' : H'_1 \Leftarrow_{p_1,m'_1} G' \Rightarrow_{p_2,m'_2} H'_2$ that can be extended to tp via extension diagrams (3) and (4) and extension morphism (resp. $\mathcal{M}$-morphism) f as in Figure 2 it holds that tp^I can be extended to tp' via unique extension diagrams (5) and (6) and unique vertical morphism (resp. $\mathcal{M}$-morphism) f'^I s.t. $f \circ f'^I = f^I$.*

$$
\begin{tikzcd}
H^I_1 \arrow[d, "g^I_1"'] \arrow[r, phantom, ""] & G^I \arrow[l, Rightarrow, "{p_1,m^I_1}"'] \arrow[r, Rightarrow, "{p_2,m^I_2}"] \arrow[d, "f^I"] & H^I_2 \arrow[d, "g^I_2"] \\
H_1 \arrow[ur, phantom, "(1)"] & G \arrow[l, Rightarrow, "{p_1,m_1}"'] \arrow[r, Rightarrow, "{p_2,m_1}"] \arrow[ur, phantom, "(2)"] & H_2
\end{tikzcd}
\qquad
\begin{tikzcd}
H^I_1 \arrow[d, "g'^I_1"'] & G^I \arrow[l, Rightarrow, "{p_1,m^I_1}"'] \arrow[r, Rightarrow, "{p_2,m^I_2}"] \arrow[d, "f'^I"] & H^I_2 \arrow[d, "g'^I_2"] \\
H'_1 \arrow[d, "g_1"'] \arrow[ur, phantom, "(5)"] & G' \arrow[l, Rightarrow, "{p_1,m'_1}"'] \arrow[r, Rightarrow, "{p_2,m'_2}"] \arrow[d, "f"] \arrow[ur, phantom, "(6)"] & H'_2 \arrow[d, "g_2"] \\
H_1 \arrow[ur, phantom, "(3)"] & G \arrow[l, Rightarrow, "{p_1,m_1}"'] \arrow[r, Rightarrow, "{p_2,m_2}"] \arrow[ur, phantom, "(4)"] & H_2
\end{tikzcd}
$$

Figure 2: Initial transformation pair $H^I_1 \Leftarrow_{p_1,m^I_1} G^I \Rightarrow_{p_2,m^I_2} H^I_2$ for $H_1 \Leftarrow_{p_1,m_1} G \Rightarrow_{p_2,m_2} H_2$

As shown in [6] an ($\mathcal{M}$-)initial transformation pair is *unique* up to isomorphism w.r.t. a given transformation pair for plain rules.

The notion of initial conflicts is based on the requirement of the *existence of initial transformation pairs* for parallel dependent or conflicting plain transformation pairs. Note that for the category of typed graphs, it is shown in [6] that this requirement holds. Moreover, [7] extended that result proving that arbitrary $\mathcal{M}$-adhesive categories fulfilling some extra conditions also satisfy it. In the case of $\mathcal{M}$-initiality, no additional requirement is needed, since $\mathcal{E}'$-$\mathcal{M}$ pair factorization is enough to ensure the existence of $\mathcal{M}$-initial transformation pairs for each parallel dependent transformation pair [6].

Definition 11 (existence of ($\mathcal{M}$-)initial transformation pair for conflict [6])**.** *A plain $\mathcal{M}$-adhesive system has* ($\mathcal{M}$-)initial transformation pairs for conflicts *if, for each transformation pair tp in conflict, the ($\mathcal{M}$-)initial transformation pair tp^I exists.*

Now initial conflicts for plain rules represent the set of all possible "smallest" conflicts. It is shown in [6] that, for a plain $\mathcal{M}$-adhesive system, critical pairs are $\mathcal{M}$-initial conflicts and each initial conflict is a special critical pair.

Definition 12 (($\mathcal{M}$-)initial conflict for plain rules[6])**.** *Given a plain $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts, a pair of direct transformations in conflict $tp : H_1 \Leftarrow_{p_1,m_1} G \Rightarrow_{p_2,m_2} H_2$ is an* initial conflict *(resp. $\mathcal{M}$-initial conflict) if it is isomorphic to the initial transformation pair (resp. $\mathcal{M}$-initial transformation pair) tp^I for tp.*

Initial conflicts for plain rules are complete as transformation pairs w.r.t. parallel dependence [6], whereas critical pairs, i.e., $\mathcal{M}$-initial conflicts, for plain rules are $\mathcal{M}$-complete [27].

Theorem 2 (completeness of initial conflicts [6])**.** *Consider a plain $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts. The set of* initial conflicts *for a pair of plain rules $\langle p_1, p_2\rangle$ is* complete *w.r.t. parallel dependence.*

The Local Confluence Theorem (requiring initial POs) can be formulated for initial conflicts of plain rules [6] similarly to the one for classical critical pairs (for plain rules) [27].

3.5. ($\mathcal{M}$-)Initial Parallel Independent Transformation Pairs for Plain Rules

In this section, we show the existence of ($\mathcal{M}$-)initial transformation pairs for *parallel independent transformation pairs*, allowing us to define an *($\mathcal{M}$-)complete* set also w.r.t. parallel independence.

We start by showing the existence of $\mathcal{M}$-initial transformation pairs for parallel independent transformation pairs:

Lemma 3 (existence of $\mathcal{M}$-initial transformation pair for parallel independent transformation pair)**.** *Given transformation rules $\langle p_1, p_2\rangle$, for every parallel independent transformation pair tp :*

$H_1 \Leftarrow_{p_1,o_1} G \Rightarrow_{p_2,o_2} H_2$, the transformation pair $tp^I : H_1^I \Leftarrow_{p_1,o_1'} G^I \Rightarrow_{p_2,o_2'} H_2^I$, where (o_1', o_2') is the pair factorization of (o_1, o_2), is $\mathcal{M}$-initial with respect to tp.

Proof. We define tp^I using $\mathcal{E}'$-$\mathcal{M}$ pair factorization. In particular, we know that there are unique morphisms $o_1' : L_1 \to G^I, o_2' : L_2 \to G^I, m : G^I \to G$, such that $(o_1', o_2') \in \mathcal{E}'$, $m \in \mathcal{M}$, $o_1 = m \circ o_1'$ and $o_2 = m \circ o_2'$. We know because of the Restriction Theorem [27] and since $m \in \mathcal{M}$ that o_1', o_2' satisfy the gluing conditions such that $tp^I : H_1^I \Leftarrow_{p_1,o_1'} G^I \Rightarrow_{p_2,o_2'} H_2^I$ exists. This means that we can extend tp^I to tp via m.

$$\begin{tikzcd}
p_1 : R_1 \arrow[d] & I_1 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & L_1 \arrow[dr, "o_1'"] \arrow[ddr, bend right, "o_1"'] & & L_2 \arrow[dl, "o_2'"'] \arrow[ddl, bend left, "o_2"] & I_2 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & R_2 : p_2 \arrow[d] \\
H_1^I \arrow[d] & D_1^I \arrow[l, hook', "c_1"] \arrow[rr, hook, "k_1"] \arrow[d] & & G^I \arrow[d, "m"] & & D_2^I \arrow[ll, hook', "k_2"'] \arrow[r, hook, "c_2"] \arrow[d] & H_2^I \arrow[d] \\
H_1 & D_1 \arrow[l, hook'] \arrow[rr, hook] & & G & & D_2 \arrow[ll, hook'] \arrow[r, hook] & H_2
\end{tikzcd}$$

Now, let us assume that $tp_0 : H_{01} \Leftarrow_{p_1,o_{01}} G_0 \Rightarrow_{p_2,o_{01}} H_{02}$ can be extended to tp via an $\mathcal{M}$-morphism m'. This means that $o_1 = m' \circ o_{01}$ and $o_2 = m' \circ o_{02}$. By $\mathcal{E}'$-$\mathcal{M}$ pair factorization, there are unique morphisms $o_{01}' : L_1 \to G_0', o_{02}' : L_2 \to G_0', m'' : G_0' \to G_0$, such that $(o_{01}', o_{02}') \in \mathcal{E}'$, $m'' \in \mathcal{M}$, $o_{01} = m'' \circ o_{01}'$ and $o_{02} = m'' \circ o_{02}'$. But this means that $o_1 = m \circ m'' \circ o_{01}'$ and $o_2 = m \circ m'' \circ o_{02}'$. By uniqueness of $\mathcal{E}'$-$\mathcal{M}$ pair factorization, this means that G^I and G_0' are isomorphic, so we have that tp^I can be extended to tp_0 via $m'' \circ i$, where i is the isomorphism from G^I to G_0'.

$$\begin{tikzcd}
L_1 \arrow[dr, "o_{01}'"] \arrow[ddr, bend right, "o_{01}"'] \arrow[dddr, bend right=60, "o_1"'] & & L_2 \arrow[dl, "o_{02}'"'] \arrow[ddl, bend left, "o_{02}"] \arrow[dddl, bend left=60, "o_2"] \\
& G_0' \arrow[d, "m''"'] & \\
& G_0 \arrow[d, "m'"'] & \\
& G &
\end{tikzcd}$$

$\square$

Based on the existence of $\mathcal{M}$-initial transformation pairs w.r.t. parallel independence, we can now define an $\mathcal{M}$-initial parallel independent transformation pair.

Definition 13 ($\mathcal{M}$-initial parallel independent transformation pair)**.** *A pair of parallel independent plain transformations* $tp : H_1 \Leftarrow_{p_1,m_1} G \Rightarrow_{p_2,m_2} H_2$ *is an* $\mathcal{M}$-initial parallel independent transformation pair *if it is isomorphic to its $\mathcal{M}$-initial transformation pair.*

In the following proposition, we characterize the set of $\mathcal{M}$-initial parallel independent transformation pairs:

Proposition 1 ($\mathcal{M}$-initial parallel independent transformation pairs)**.** *Given transformation rules $\langle p_1, p_2 \rangle$, the parallel independent pair of transformations $tp = P_1 \Leftarrow_{p_1,o_1} K \Rightarrow_{p_2,o_2} P_2$ depicted in the diagram below is an $\mathcal{M}$-initial transformation pair with respect to parallel independence if and only if $\langle o_1, o_2 \rangle \in \mathcal{E}'$*

$$
\begin{tikzcd}
R_1 \arrow[d] & I_1 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & L_1 \arrow[dr, "o_1"'] \arrow[drrr, dotted, "d_{12}"] & & L_2 \arrow[dl, "o_2"] \arrow[dlll, dotted, "d_{21}"'] & I_2 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & R_2 \arrow[d] \\
P_1 & D_1 \arrow[l, hook', "c_1"] \arrow[rr, hook, "k_1"'] & & K & & D_2 \arrow[ll, hook', "k_2"] \arrow[r, hook, "c_2"'] & P_2
\end{tikzcd}
$$

Proof. If $\langle o_1, o_2 \rangle \notin \mathcal{E}'$, according to Lemma 3, the transformation pair $tp^I : H_1^I \Leftarrow_{p_1,o_1'} G^I \Rightarrow_{p_2,o_2'} H_2^I$, where (o_1', o_2') is the pair factorization of (o_1, o_2), is $\mathcal{M}$-initial with respect to tp. But this means that tp is not $\mathcal{M}$-initial.

Conversely, if $\langle o_1, o_2 \rangle \in \mathcal{E}'$ then $\langle o_1, o_2 \rangle$, together with id_K is its pair factorization, which means, by Lemma 3, that tp is $\mathcal{M}$-initial. □

Corollary 1 ($\mathcal{M}$-completeness of $\mathcal{M}$-initial parallel independent transformation pairs)**.** *The set of $\mathcal{M}$-initial parallel independent transformation pairs is* $\mathcal{M}$-complete *w.r.t. parallel independence.*

Proof. This follows directly from the property of $\mathcal{E}'$-$\mathcal{M}$ pair factorization, the Restriction Theorem, and Proposition 1. □

The proof of the existence of initial transformation pairs for parallel independent transformation pairs requires the existence of binary coproducts. In this proof we will use the following lemma:

Lemma 4 (extensions of coproduct transformation pair)**.** *Given rules $p_1 : L_1 \leftarrow I_1 \rightarrow R_1$ and $p_2 : L_2 \leftarrow I_2 \rightarrow R_2$ and transformation pairs $tp : H_1 \Leftarrow_{p_1,m_1} G \Rightarrow_{p_2,m_2} H_2$ and $tp_{L_1+L_2} : R_1 + L_2 \Leftarrow_{p_1,i_1} L_1 + L_2 \Rightarrow_{p_2,i_2} L_1 + R_2$, where tp is parallel independent, we have that the coproduct mediating morphism $m : L_1 + L_2 \rightarrow G$ defines the extension diagram:*

$$\begin{tikzcd}
R_1 + L_2 \arrow[d] & L_1 + L_2 \arrow[l, Rightarrow] \arrow[r, Rightarrow] \arrow[d, "m"] & L_1 + R_2 \arrow[d] \\
H_1 & G \arrow[l, Rightarrow] \arrow[r, Rightarrow] & H_2
\end{tikzcd}$$

Proof. Let tp be:

$$\begin{tikzcd}
R_1 \arrow[d] & I_1 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & L_1 \arrow[dr, "m_1"'] \arrow[drrr, dotted, "d_{12}"] & & L_2 \arrow[dl, "m_2"] \arrow[dllll, dotted, "d_{21}"'] & I_2 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & R_2 \arrow[d] \\
H_1 & D_1 \arrow[l, hook', "c_1"] \arrow[rr, hook, "k_1"'] & & G & & D_2 \arrow[ll, hook', "k_2"] \arrow[r, hook, "c_2"'] & H_2
\end{tikzcd}$$

Let us prove that if $m : L_1 + L_2 \to G$ is the mediating morphism for the coproduct, satisfying that $m \circ i_1 = m_1$ and $m \circ i_2 = m_2$, where $i_1 : L_1 \to L_1 + L_2$ and $i_2 : L_2 \to L_1 + L_2$ are the coproduct morphisms, then m defines the extension diagram:

$$\begin{tikzcd}
R_1 + L_2 \arrow[d] \arrow[r, phantom, "(1)"] & L_1 + L_2 \arrow[l, Rightarrow] \arrow[r, Rightarrow] \arrow[d, "m"] \arrow[r, phantom, "(2)"] & L_1 + R_2 \arrow[d] \\
H_1 & G \arrow[l, Rightarrow] \arrow[r, Rightarrow] & H_2
\end{tikzcd}$$

In particular, we have to prove that (1) and (2) define extension diagrams. Let us argue w.r.t. extension diagram (1) (we can argue analogously for (2)).

We have to prove that (3) and (4) are pushouts:

$$\begin{tikzcd}
R_1 + L_2 \arrow[d] \arrow[r, phantom, "(3)"] & I_1 + L_2 \arrow[l, hook'] \arrow[r, hook] \arrow[d] \arrow[r, phantom, "(4)"] & L_1 + L_2 \arrow[d, "m"] \\
H_1 & D_1 \arrow[l, hook'] \arrow[r, hook] & G
\end{tikzcd}$$

We have that squares (5), (6) and (7) below are pushouts, therefore (4) is also a pushout, according to the Butterfly Lemma (see [27]). Similarly, since squares (8), (9) and (10) below are pushouts, for the same reason, (3) is also a pushout.

$$\begin{tikzcd}
 & I_1 \arrow[r, hook] \arrow[d] \arrow[r, phantom, "(5)"] & L_1 \arrow[d, "m_1"] \\
L_2 \arrow[r, "d_{21}"] \arrow[d] \arrow[r, phantom, "(6)"] & D_1 \arrow[r, hook, "k_1"] \arrow[d] \arrow[r, phantom, "(7)"] & G \arrow[d] \\
L_2 \arrow[r, "d_{21}"'] & D_1 \arrow[r, hook, "k_1"'] & G
\end{tikzcd}
\qquad
\begin{tikzcd}
 & I_1 \arrow[r, hook] \arrow[d] \arrow[r, phantom, "(8)"] & R_1 \arrow[d] \\
L_2 \arrow[r, "d_{21}"] \arrow[d] \arrow[r, phantom, "(9)"] & D_1 \arrow[r, hook, "c_1"] \arrow[d] \arrow[r, phantom, "(10)"] & H_1 \arrow[d] \\
L_2 \arrow[r, "d_{21}"'] & D_1 \arrow[r, hook, "c_1"'] & H_1
\end{tikzcd}$$

□

Lemma 5 (existence of initial transformation pair for parallel independent transformation pair)**.** *Given a pair of parallel independent plain direct transformations* $tp : H_1 \Leftarrow_{p_1,m_1} G \Rightarrow_{p_2,m_2} H_2$,

$$
\begin{tikzcd}
p_1 : R_1 \arrow[d] & I_1 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & L_1 \triangleleft^{\mathrm{ac}_{L_1}} \arrow[dr, "i_1"'] \arrow[rrrd, dotted, "d_{12}"] & & {}^{\mathrm{ac}_{L_2}}\triangleright L_2 \arrow[dl, "i_2"] \arrow[llld, dotted, "d_{21}"'] & I_2 \arrow[l, hook'] \arrow[r, hook] \arrow[d] & R_2 : p_2 \arrow[d] \\
p_1^* : R_1 + L_2 \arrow[d] & I_1 + L_2 \arrow[l, hook', "c_1"] \arrow[rr, hook, "k_1"'] \arrow[d] & & L_1 + L_2 \arrow[d, "m"'] & & L_1 + I_2 \arrow[ll, hook', "k_2"] \arrow[r, hook, "c_2"'] \arrow[d] & L_1 + R_2 : p_2^* \arrow[d] \\
H_1 & D_1 \arrow[l, hook'] \arrow[rr, hook] & & G & & D_2 \arrow[ll, hook'] \arrow[r, hook] & H_2
\end{tikzcd}
$$

Figure 3: initial parallel independent transformation pair $tp_{L_1+L_2}$ for parallel independent AC-disregarding transformation pair tp_G

then $tp_{L_1+L_2} : R_1{+}L_2 \Leftarrow_{p_1,i_1} L_1{+}L_2 \Rightarrow_{p_2,i_2} L_1{+}R_2$*, where* $i_1 : L_1 \to L_1{+}L_2$ *and* $i_2 : L_2 \to L_1{+}L_2$ *are the coproduct morphisms, is initial for* tp*.*

Proof. By Lemma 4, we know that (1)+(2) is an extension diagram, where $m : L_1 + L_2 \to G$ is the mediating morphism for the coproduct $L_1 + L_2$.

$$
\begin{tikzcd}
R_1 + L_2 \arrow[d] \arrow[dr, phantom, "(1)"] & L_1 + L_2 \arrow[l, Rightarrow] \arrow[r, Rightarrow] \arrow[d, "m"] \arrow[dr, phantom, "(2)"] & L_1 + R_2 \arrow[d] \\
H_1 & G \arrow[l, Rightarrow] \arrow[r, Rightarrow] & H_2
\end{tikzcd}
$$

Let us now assume that $tp' : H_1' \Leftarrow_{p_1,m_1'} G' \Rightarrow_{p_2,m_2'} H_2'$ can be extended to tp via $f' : G' \to G$, defining extension diagrams (5)+(6).

$$
\begin{tikzcd}
R_1 + L_2 \arrow[d] \arrow[dr, phantom, "(3)"] & L_1 + L_2 \arrow[l, Rightarrow] \arrow[r, Rightarrow] \arrow[d, "m'"] \arrow[dr, phantom, "(4)"] & L_1 + R_2 \arrow[d] \\
H_1' \arrow[d] \arrow[dr, phantom, "(5)"] & G' \arrow[l, Rightarrow] \arrow[r, Rightarrow] \arrow[d, "f'"] \arrow[dr, phantom, "(6)"] & H_2' \arrow[d] \\
H_1 & G \arrow[l, Rightarrow] \arrow[r, Rightarrow] & H_2
\end{tikzcd}
$$

We know that there is a unique morphism $m' : L_1 + L_2 \to G'$, such that $m \circ i_1 = m_1'$ and $m \circ i_2 = m_2'$, defining by Lemma 4 the extension diagrams (3)+(4). Hence, we only have to prove that $f' \circ m' = m$, but we know that $m : L_1 + L_2 \to G$ is the unique morphism that defines the outer extension diagrams (3)+(4)+(5)+(6), thus $f' \circ m' = m$. $\square$

Because of uniqueness of initial transformation pairs up to isomorphism, it thus follows that for each pair of *plain rules* $\langle p_1, p_2 \rangle$ there is a *unique initial parallel independent transformation pair* $tp_{L_1+L_2} : R_1{+}L_2 \Leftarrow_{p_1,i_1} L_1{+}L_2 \Rightarrow_{p_2,i_2} L_1{+}R_2$. Note that this is different from the situation for conflicts for plain rules, where initial transformation pairs may differ from conflict to conflict.

Consequently, in general for a pair of rules we can have different initial conflicts, but there exists always a unique initial parallel independent transformation pair.

Definition 14 (initial parallel independent transformation pair)**.** *A pair of parallel independent plain transformations* $tp : H_1 \Leftarrow_{p_1,m_1} G \Rightarrow_{p_2,m_2} H_2$ *is an* initial parallel independent transformation pair *if it is isomorphic to the transformation pair* $tp_{L_1+L_2} : R_1 + L_2 \Leftarrow_{p_1,i_1} L_1 + L_2 \Rightarrow_{p_2,i_2} L_1 + R_2$.

The one-element set consisting of the initial parallel independent transformation pair for a given pair of rules is *complete w.r.t. parallel independence.*

Theorem 3 (completeness of initial parallel independent transformation pairs)**.** *The set consisting of the initial parallel independent transformation pair* $tp_{L_1+L_2} : R_1 + L_2 \Leftarrow_{p_1,i_1} L_1 + L_2 \Rightarrow_{p_2,i_2} L_1 + R_2$ *for a pair of plain rules* $\langle p_1, p_2 \rangle$ *is* complete *w.r.t. parallel independence.*

Proof. This follows directly from Lemma 5 and Definition 14. □

4. Initial Conflicts for Rules with ACs

We start with showing why it is not possible to straightforwardly generalize the idea of initial conflicts from plain rules to rules with ACs. On the one hand, *conflict inheritance* does not hold any more such that not each transformation pair that can be extended to a conflicting one is conflicting again, which was the basis for being able to show completeness of initial conflicts for plain rules. Actually, the reverse of inheritance, what we call conflict co-inheritance, does not hold either, i.e., not each transformation pair that extends a conflicting one is conflicting again (cf. subsection 4.1). Moreover, it is *impossible* in general to find a *finite and complete set of finite conflicts* for rules with ACs (cf. subsection 4.2) as illustrated for the category of **Graphs**. Finiteness is a basic prerequisite however to be able to practically compute a complete (i.e. representative) set of conflicts statically. This motivates again the need for having *symbolic transformation pairs* as introduced in Definition 7, allowing us to define *initial conflicts* (cf. subsection 4.3) as a set of specific symbolic transformation pairs, being complete w.r.t. parallel dependence indeed (as shown in subsection 4.4). This set as well as its elements are also finite, for example, in the case of graphs (and provided that the rules are finite).

4.1. Conflict Inheritance

Conflicts are in general not inherited (as opposed to the case of plain rules [6]) such that not each (initial) transformation pair that can be extended to a conflicting one will be conflicting again. This may happen in particular for AC conflicts. Use-delete (resp. delete-use) conflicts for rules with ACs are still inherited.

Lemma 6 (Use-delete (delete-use) conflict inheritance)**.** *Given two transformation pairs tp and tp' such that tp' extends tp, if tp is in use-delete (resp. delete-use) conflict so is tp'.*

Proof. The proof is completely analogous to the one for the conflict inheritance lemma for plain rules in [6] and use-delete (resp. delete-use) conflicts. $\square$

Example 1 (Neither inheritance nor co-inheritance for AC conflicts)**.** *Consider rules $p_1 : \bigcirc \leftarrow \bigcirc \rightarrow \bigcirc\!\!\rightarrow\!\!\bigcirc$ (with AC true), producing an outgoing edge with a node, and $p_2 : \bigcirc \leftarrow \bigcirc \rightarrow \bigcirc\ \bigcirc$ with NAC $\neg\exists n : \bigcirc \rightarrow \bigcirc\ \bigcirc\ \bigcirc$, producing a node only if two other nodes do not exist already. Consider graph $G = \bigcirc\ \bigcirc$, holding two nodes. Applying both rules to G (with the matches sharing one node in G) we obtain a produce-AC conflict since the first rule creates a third node, forbidden by the second rule. Now both rules can be applied similarly to the shared node in the subgraph $G' = \bigcirc$ of G obtaining parallel independent transformations, illustrating that AC-conflicts are not inherited.*

Assume that p_2 would have the more complex AC $(\neg\exists n : \bigcirc \rightarrow \bigcirc\ \bigcirc\ \bigcirc) \vee (\exists p : \bigcirc \rightarrow \bigcirc\ \bigcirc\ \bigcirc\ \bigcirc)$, then the transformation pair arising from their application to G sharing one node with their matches is still produce-AC conflicting. Now the application of both rules to the extended graph $G'' = \bigcirc\ \bigcirc\ \bigcirc\ \bigcirc$ (sharing with the extended matches the same node as in G) would satisfy the AC and would be moreover parallel independent, illustrating that AC-conflicts are not co-inherited.

4.2. Complete Set of Conflicts

We show that in $\mathcal{M}$-adhesive categories it is in general impossible to find a finite and complete set of finite conflicts for rules with ACs as illustrated for the category **Graphs** (under the assumption[7] that graph transformation rules are finite).

Theorem 4. *Given finite rules $\rho_1 = \langle p_1, \mathrm{ac}_{L_1}\rangle$ and $\rho_2 = \langle p_2, \mathrm{ac}_{L_2}\rangle$ for the $\mathcal{M}$-adhesive category* **Graphs***, in general, there is no finite set of finite transformation pairs $\mathcal{S}$ for ρ_1 and ρ_2 that is complete w.r.t. parallel dependence.*

Proof. ACs over the empty graph $\emptyset$ express so-called graph properties. Graph properties formulated this way have the same expressive power as first-order logic (FOL) on graphs[8] as shown in [24]. This means that we can express any graph property equivalently using a first-order formula. For the same reason, we can state any graph property for graphs without isolated nodes using a first-order formula (i.e., any graph property that, in particular, implies that the given graph has no isolated nodes).

Now consider the following two rules $\rho_1 = \langle \emptyset \leftarrow \emptyset \rightarrow \emptyset, c\rangle$ and $\rho_2 = \langle \emptyset \leftarrow \emptyset \rightarrow 1_N, true\rangle$, with 1_N the graph consisting of an isolated node and c some property (expressible using ACs over the

[7] Without this assumption even in the case of plain rules the set of critical pairs would already be infinite.

[8] FOL on graphs is standard first-order logic with two additional built-in predicates: $Node(n)$ -to state that n is a node and $Edge(e, n, n')$ to state that e is an edge from n to n'.

empty graph) about graphs without isolated nodes. That is, the first rule can be applied to a graph G, if $G \models c$, and it leaves G unchanged; and the second rule, which is always applicable, adds an isolated node to G. Thus the set of transformation pairs associated to these rules is $\mathcal{S}_0 = \{G \Longleftarrow_{\rho_1} G \Longrightarrow_{\rho_2} G \oplus 1_N \mid G \models c\}$. Note that all the transformation pairs in $\mathcal{S}_0$ are AC conflicts, since $G \oplus 1_N$ does not satisfy c having an isolated node, which means that the set of conflicts of ρ_1 and ρ_2 is precisely $\mathcal{S}_0$. In particular, for any graph G either $G \models c$[9] and both rules can be applied to G in a unique way (since there is a unique match $h : \emptyset \to G$), or $G \not\models c$ such that ρ_1 cannot be applied. This means that, for any G, there is at most one transformation pair $G \Longleftarrow_{\rho_1} G \Longrightarrow_{\rho_2} G \oplus 1_N$ starting from G. Consequently, if G and G' satisfy c, any morphism $h : G \to G'$ defines an extension diagram between their associated transformations. For these reasons, even if it is an abuse of notation, given sets of transformation pairs $\mathcal{S}_0$ ($\mathcal{S}$), we write $G \in \mathcal{S}_0$ (resp. $\mathcal{S}$) meaning $G \Longleftarrow_{\rho_1} G \Longrightarrow_{\rho_2} G \oplus 1_N \in \mathcal{S}_0$ (resp. $\mathcal{S}$).

Now let us assume that a finite set $\mathcal{S}$ of conflicts for rules ρ_1 and ρ_2 exists that is complete w.r.t. parallel dependence. This means that $G \in \mathcal{S}_0$ if and only if there is a $G' \in \mathcal{S}$ and a morphism $h : G' \to G$. We know, by the property of epi-mono factorization, that any morphism $h : G' \to G$ can be decomposed into $h = m \circ e$ with m mono and e epi. Moreover, since G' is assumed to be finite, there is a finite number of epimorphisms whose source is G'. Let $Epi_{G'}$ be the set $\{G'' \mid \text{there is an epimorphism } e : G' \to G''\}$, then we would have that $G \in \mathcal{S}_0$ if and only if there is a $G' \in \mathcal{S}$, a $G'' \in Epi_{G'}$ and a monomorphism $m : G'' \to G$. Note that, by definition of satisfaction (cf. Definition 3), the property that there is a monomorphism $m : G'' \to G$ is equivalent to $G \models \exists(\emptyset \to G'', \mathit{true})$. Therefore $G \in \mathcal{S}_0$ if and only if there is a $G' \in \mathcal{S}$, and a $G'' \in Epi_{G'}$ such that $G \models \exists(\emptyset \to G'', \mathit{true})$. But this means that $G \in \mathcal{S}_0$ if and only if there is a $G' \in \mathcal{S}$ such that $G \models \big(\bigvee_{G'' \in Epi_{G'}} \exists(\emptyset \to G'', \mathit{true})\big)$, or equivalently $G \models c'$, where c' is the condition

$$c' = \big(\bigvee_{\substack{G'' \in Epi_{G'} \\ G' \in \mathcal{S}}} \exists(\emptyset \to G'', \mathit{true})\big).$$

This means however that c and c' are logically equivalent, but this is a contradiction, since it is not possible to represent any arbitrarily complex first-order formula, for instance a universally quantified formula, in terms of a finite disjunction of existential atoms. Therefore, our assumption was wrong and $\mathcal{S}$ is in general infinite. $\square$

[9]A graph property is an application condition over the empty graph $\emptyset$ (or, in the general case, the initial object in the category of graphical structures considered), thus composed of literals of the form $c = \exists(\emptyset \to G', c')$. In particular, we say that $G \models c$ if $i_G \models c$ with i_G the unique morphism from $\emptyset$ to G.

4.3. Initial Conflicts and Critical Pairs for Rules with ACs

We generalize the notion of *initial conflicts* for plain rules to rules with ACs. In particular, we introduce them as special symbolic transformation pairs (cf. Def. 7). They are *conflict-inducing* meaning that there needs to exist an unfolding of the symbolic transformation pair into a conflicting transformation pair. Moreover, their AC-disregarding transformation pair needs to be an initial conflict or initial parallel independent transformation pair. We also show formally the *relationship between initial conflicts and critical pairs* as reintroduced in subsection 3.3. In particular, we demonstrate that initial conflicts represent a proper subset of critical pairs again. Moreover, analogous to the case of plain rules, we are able to show that critical pairs coincide with $\mathcal{M}$-initial conflicts for rules with ACs, demonstrating that in this sense initial conflicts represent a conservative extension.

Definition 15 (unfolding of symbolic transformation pair)**.** *Given a symbolic transformation pair* $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K\rangle$ *for a rule pair* $\langle \rho_1, \rho_2\rangle$*, then its unfolding* $\mathcal{U}(stp_K)$ *consists of all transformation pairs* $H_1 \Leftarrow_{\rho_1,m_1} G \Rightarrow_{\rho_2,m_2} H_2$ *that extend the AC-disregarding transformation pair* tp_K *via some extension morphism* $m : K \to G$*.*

Remark 4 (non-empty unfolding)**.** *Note that the unfolding of a symbolic transformation pair is not empty if there exists an extension morphism* $m : K \to G$ *satisfying the gluing conditions as well as* ac_K *for the derived spans (as can be followed directly from the Embedding Theorem [9, 10] for rules with ACs, since* m *would be boundary as well as AC-consistent).*

Definition 16 (conflict-inducing symbolic transformation pair)**.** *Given rules* $\rho_1 = \langle p_1, \mathrm{ac}_{L_1}\rangle$ *and* $\rho_2 = \langle p_2, \mathrm{ac}_{L_2}\rangle$*, a symbolic transformation pair* $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K\rangle$ *for* $\langle\rho_1, \rho_2\rangle$ *is* conflict-inducing *if there exists a pair of conflicting transformations in its unfolding* $\mathcal{U}(stp_K)$*.*

Remark 5 (conflict-inducing & unfolding)**.** *The unfolding of a conflict-inducing symbolic transformation pair may contain parallel independent transformations. Consider rules* $\rho_1 = \langle p_1, true\rangle$ *and* $\rho_2 = \langle p_2, \neg\exists n\rangle$ *from Example 1 and symbolic transformation pair* $stp' : \langle tp_{G'}, \mathrm{ac}_{G'}, \mathrm{ac}^*_{G'}\rangle$*, with* $tp_{G'}$ *the AC-disregarding transformation pair arising from applying rules* p_1 *and* p_2 *to* $G' = \bigcirc$*,* $\mathrm{ac}_{G'} = \neg\exists n' : \bigcirc \to \bigcirc\ \bigcirc\ \bigcirc$*, and* $\mathrm{ac}^*_{G'} = (\exists p' : \bigcirc \to \bigcirc\ \bigcirc) \vee (\exists p'' : \bigcirc \to \bigcirc\ \bigcirc\ \bigcirc)$*. Then* stp' *is a conflict-inducing symbolic transformation pair, since its unfolding includes the parallel dependent transformation pair* tp_G *arising from applying the rules* ρ_1 *and* ρ_2 *to* $G = \bigcirc\ \bigcirc$*. The extension morphism* $m : G' \to G$ *fulfills* $\mathrm{ac}_{G'}$ *and* $\mathrm{ac}^*_{G'}$ *indeed. However, the transformation pair* $tp_{G'}$ *satisfies all ACs, belongs to the unfolding* $\mathcal{U}(stp)$ *accordingly, but is parallel independent (as described in Example 1 and derivable from the fact that* $\mathrm{ac}^*_{G'}$ *is not fulfilled for the extension morphism* $id_{G'}$*).*

An *initial conflict* (resp. *$\mathcal{M}$-initial conflict*) is a conflict-inducing symbolic transformation pair with its AC-disregarding transformation pair being initial (resp. $\mathcal{M}$-initial). Note that we say that an AC-disregarding transformation pair is ($\mathcal{M}$-)initial if it is ($\mathcal{M}$-)initial as plain transformation pair (cf. Figure 3). Remember that each symbolic transformation pair is uniquely determined by its underlying AC-disregarding transformation pair. This means that the set of ($\mathcal{M}$-)initial conflicts basically consists of a filtered set of plain ($\mathcal{M}$-)initial conflicts (those that are conflict-inducing as symbolic transformation pair) together with the set of ($\mathcal{M}$-)initial parallel independent transformation pairs (in case they are conflict-inducing as symbolic transformation pairs). Recall that the set of initial (resp. $\mathcal{M}$-initial) parallel independent transformation pairs for plain rules consists of the singleton $\{tp_{L_1+L_2}\}$ (resp. the set of parallel independent transformation pairs with matches in $\mathcal{E}'$).

Definition 17 (($\mathcal{M}$-)initial conflict)**.** *Consider an $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts along plain rules. An* initial conflict *(resp.* $\mathcal{M}$-initial conflict*) for rules $\rho_1 = \langle p_1, \mathrm{ac}_{L_1}\rangle$ and $\rho_2 = \langle p_2, \mathrm{ac}_{L_2}\rangle$ is a conflict-inducing symbolic transformation pair $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K\rangle$ with the AC-disregarding transformation pair tp_K being initial (resp. $\mathcal{M}$-initial), i.e. either tp_K is an ($\mathcal{M}$-)initial conflict for rules p_1 and p_2 (in this case stp_K is called a* use-delete/delete-use ($\mathcal{M}$-)initial conflict*) or it is an ($\mathcal{M}$-)initial parallel independent transformation pair (in this case stp_K is called an* AC ($\mathcal{M}$-)initial conflict*).*

Analogous to the case of plain rules, $\mathcal{M}$-initial conflicts coincide with critical pairs:

Proposition 2 ($\mathcal{M}$-initial conflicts are critical pairs)**.** *A symbolic tranformation pair stp is a critical pair if and only if stp is an $\mathcal{M}$-initial conflict.*

Proof. Direct consequence of Proposition 1 and of the definitions of critical pairs (i.e., $\mathcal{M}$-initial conflicts) for plain rules and of critical pairs for rules with ACs (Definition 8). $\square$

Note that as explained in Remark 5 the unfolding of a conflict-inducing symbolic transformation pair (and in particular of an AC initial conflict) may entail apart from (at least one) conflicting transformation pair(s) also parallel independent transformation pairs. All conflicts in the unfolding of an AC initial conflict are AC conflicts, and never use-delete/delete-use conflicts (because otherwise we would get a contradiction using Lemma 6).

Example 2 (initial conflict)**.** *Consider again the rules from Example 1. Applying both rules to $L_1 + L_2 = \bigcirc\ \bigcirc$ (with disjoint matches) we obtain the AC initial conflict $stp_K = stp_{L_1+L_2} = \langle tp_{L_1+L_2}, \mathrm{ac}_{L_1+L_2}, \mathrm{ac}^*_{L_1+L_2}\rangle$. Thereby $\mathrm{ac}_{L_1+L_2}$ is equivalent to $\neg\exists(\bigcirc_1\ \bigcirc_2 \to \bigcirc_1\ \bigcirc_2\ \bigcirc) \wedge \neg\exists(\bigcirc_1\ \bigcirc_2 \to \bigcirc_{1,2}\ \bigcirc\ \bigcirc)$, expressing that when during extension both nodes are merged, no two additional nodes, otherwise not one additional node should be given. Moreover, $\mathrm{ac}^*_{L_1+L_2}$ is equivalent to $\exists(\bigcirc_1\ \bigcirc_2 \to \bigcirc_{1,2}\ \bigcirc) \vee \exists(\bigcirc_1\ \bigcirc_2 \to \bigcirc_1\ \bigcirc_2)$, expressing that either both nodes are not merged during extension, otherwise one additional node should be present for a conflict to arise. Both transformation pairs* *(the conflicting one from $G = \bigcirc\ \bigcirc$ as well as the parallel independent one from its subgraph $G' = \bigcirc$, sharing the merged node in their matches) described in Example 1 belong to its unfolding.*

Each initial conflict is in particular also a critical pair.

Theorem 5 (initial conflict is critical pair)**.** *Consider an $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts along plain rules. Each initial conflict $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ is a critical pair.*

Proof. Given some initial conflict $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$, for rules $\rho_1 = (p_1, ac_1), \rho_2 = (p_2, ac_2)$ we have two cases:

- If tp_K is an initial conflict for the AC-disregarding rules p_1, p_2, then since every initial conflict for plain rules is a critical pair [6] and, as discussed above, each critical pair tp_K for the plain rules $\langle p_1, p_2 \rangle$ corresponds uniquely to the critical pair stp_K we have that stp_K is a critical pair for ρ_1, ρ_2.
- If $tp_K = tp_{L_1+L_2}$ then, by Proposition 1, we also have that stp_K is a critical pair for ρ_1, ρ_2, since we can easily see that the coproduct injections $(i_1 : L_1 \to L_1 + L_2, i_2 : L_2 \to L_1 + L_2) \in \mathcal{E}'$. In particular, we know that there should exist a unique pair factorization $(i'_1 : L_1 \to C, i'_2 : L_2 \to C) \in \mathcal{E}'$ and $m : C \to L_1 + L_2$, such that $i_1 = m \circ i'_1$ and $i_2 = m \circ i'_2$. But, by the universal property of coproducts, there is a unique morphism $m' : L_1 + L_2 \to C$, such that $i'_1 = m' \circ i_1$ and $i'_2 = m' \circ i_2$. Then, $m \circ m'$ satisfies that $m \circ m' \circ i_1 = m \circ i'_1 = i_1$ and $m \circ m' \circ i_2 = m \circ i'_2 = i_2$, which implies $m \circ m' = id_{L_1+L_2}$ by the universal property of coproducts. Similarly, $m' \circ m \circ i'_1 = m' \circ i_1 = i'_1$ and $m' \circ m \circ i'_2 = m' \circ i_2 = i'_2$, which implies $m' \circ m = id_C$, by the uniqueness of pair factorization. This means that $L_1 + L_2$, i_1 and i_2 are isomorphic to C, i'_1 and i'_2, implying $(i_1, i_2) \in \mathcal{E}'$.

□

The reverse direction of Theorem 5 does not hold, i.e. in the case of rules with ACs initial conflicts represent also a *proper subset* of the set of critical pairs. This proper subset relation holds already in the case of plain rules. Each critical pair for plain rules tp_K corresponds uniquely to a critical pair $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ with ac_K true and ac^*_K. Thus in this special case we would have as many critical pairs that are no initial conflicts as for the case with plain rules. More generally, critical pairs $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ where tp_K represents a use-delete/delete-use conflict (but is not initial yet) are represented by the initial conflict $stp_I : \langle tp_I, \mathrm{ac}_I, \mathrm{ac}^*_I \rangle$ with tp_I the unique initial conflict for tp_K as plain transformation pair. Moreover, critical pairs $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ where tp_K is parallel independent as plain transformation pair are represented by one initial conflict $stp_{L_1+L_2} : \langle tp_{L_1+L_2}, \mathrm{ac}_{L_1+L_2}, \mathrm{ac}^*_{L_1+L_2} \rangle$ with $tp_{L_1+L_2}$ the initial parallel independent transformation pair.

Example 3 (initial conflicts: proper subset of critical pairs). *Consider again the rules from Example 1 and their application to $G' = \bigcirc$. The symbolic transformation pair $stp_{G'} : \langle tp_{G'}, \mathrm{ac}_{G'}, \mathrm{ac}^*_{G'} \rangle$ is a critical pair, but not an initial conflict. In particular, this critical pair is represented by the unique AC initial conflict $stp_{L_1+L_2} : \langle tp_{L_1+L_2}, \mathrm{ac}_{L_1+L_2}, \mathrm{ac}^*_{L_1+L_2} \rangle$ (which is also a critical pair).*

4.4. Completeness

We show that initial conflicts are complete (not $\mathcal{M}$-complete as in the case of critical pairs, cf. Theorem 1) w.r.t. parallel dependence as symbolic transformation pairs.

Theorem 6 (completeness of initial conflicts). *Consider an $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts along plain rules. The set of* initial conflicts *for a pair of rules $\langle \rho_1, \rho_2 \rangle$ is* complete *w.r.t. parallel dependence.*

Proof. Given a parallel dependent pair of transformations $tp_G : H_1 \Leftarrow_{\rho_1,m_1} G \Rightarrow_{\rho_2,m_2} H_2$ we need to show that some initial conflict via rules ρ_1 and ρ_2 exists that can be extended to tp_G via some extension morphism $m : K \to G$ with $m \models \mathrm{ac}_K \wedge \mathrm{ac}^*_K$.

Assume that tp_G is a use-delete/delete-use conflict. Then tp_G is also a use-delete/delete-use conflict as AC-disregarding transformation pair. This means that an initial conflict tp_K for the plain rules p_1 and p_2 exists according to Theorem 2 that can be extended via some extension morphism $m : K \to G$ into tp_G as AC-disregarding transformation pair. The symbolic transformation pair $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ is obviously conflict-inducing. We moreover show that $m \models \mathrm{ac}_K \wedge \mathrm{ac}^*_K$. It follows that $m \models \mathrm{ac}_K$ because of Lemma 1 and the fact that the matches of tp_G satisfy ac_{L_1} and ac_{L_2}. Moreover $m \models \mathrm{ac}^*_K$ since tp_K is an initial conflict (i.e. delete-use) for the plain rules p_1 and p_2 such that ac^*_K is true.

Assume that tp_G is not a use-delete/delete-use conflict, but it is an AC conflict. Since tp_G is not a use-delete/delete-use conflict we know that it is parallel independent as AC-disregarding transformation pair. This means that the initial parallel independent transformation pair $tp_{L_1+L_2} : R_1 + L_2 \Leftarrow_{p_1,i_1} L_1 + L_2 \Rightarrow_{p_2,i_2} L_1 + R_2$ for the plain rules p_1 and p_2 can be extended via morphism $m : L_1 + L_2 \to G$ to tp_G as AC-disregarding transformation pair (as illustrated in Figure 3). The symbolic transformation pair $stp_{L_1+L_2} : \langle tp_{L_1+L_2}, \mathrm{ac}_{L_1+L_2}, \mathrm{ac}^*_{L_1+L_2} \rangle$ is obviously conflict-inducing. We moreover show that $m \models \mathrm{ac}_{L_1+L_2} \wedge \mathrm{ac}^*_{L_1+L_2}$. It follows that $m \models \mathrm{ac}_{L_1+L_2}$ because of Lemma 1 and the fact that the matches of tp_G satisfy ac_{L_1} and ac_{L_2}. Moreover $m \models \mathrm{ac}^*_{L_1+L_2}$ with $\mathrm{ac}^*_{L_1+L_2,d_{12}} = \mathrm{L}(p^*_2, \mathrm{Shift}(c_2 \circ d_{12}, \mathrm{ac}_{L_1}))$ and $\mathrm{ac}^*_{L_1+L_2,d_{21}} = \mathrm{L}(p^*_1, \mathrm{Shift}(c_1 \circ d_{21}, \mathrm{ac}_{L_2}))$ because of Lemma 1, Lemma 2, the fact that diagonal morphisms for plain parallel independence are unique w.r.t. making the corresponding triangles commute, and the fact that tp_G is AC conflicting (i.e. either ac_{L_1} or ac_{L_2} are not satisfied by the extended matches into H_2 and H_1, respectively). □

Remark 6 (uniqueness of initial conflicts). *It holds again that for each conflict a* unique *(up-to-isomorphism) initial conflict exists representing it, since this property is inherited from the one for* *plain rules [6] and the fact that the initial parallel independent pair of transformations is unique w.r.t. a given rule pair.*

The Local Confluence Theorem [9, 10] for rules with ACs[10] still holds in case we substitute the set of critical pairs by initial conflicts, and moreover requiring initial pushouts. The proof runs completely analogously. The only difference is that for this proof, we need initial pushouts over general morphisms whereas in the proof in [9, 10] initial pushouts over $\mathcal{M}$-morphisms are sufficient.

5. Minimal Completeness

In previous sections we have seen that critical pairs (resp. initial conflicts), both in the plain case and in the case of rules with ACs, are $\mathcal{M}$-complete (resp. complete) with respect to parallel dependence. Similarly we have seen for the plain case that initial (resp. $\mathcal{M}$-initial) parallel independent transformation pairs are complete with respect to parallel independence. In this section, we will see, firstly, that these sets are minimal, i.e., that we cannot find smaller sets that are ($\mathcal{M}$-)complete (cf. subsection 5.1). Moreover, we will also show that minimally complete (resp. $\mathcal{M}$-complete) sets of conflicts coincide precisely with the sets of initial (resp. $\mathcal{M}$-initial) conflicts for the case of plain rules and with the sets of concrete initial (resp. $\mathcal{M}$-initial) conflicts for the case of rules with ACs, provided that ($\mathcal{M}$-)initial transformation pairs exist (cf. subsection 5.2). Then, in the last subsection, we will see how, out of an ($\mathcal{M}$-)complete set of conflicts, we can extract a subset that is minimally ($\mathcal{M}$-)complete.

5.1. ($\mathcal{M}$-)Initial Conflicts are Minimally ($\mathcal{M}$-)Complete

We start with defining what we understand by minimally complete sets of (symbolic) transformation pairs.

Definition 18 (minimal ($\mathcal{M}$-)completeness)**.** *A set of* (symbolic) transformation pairs $\mathcal{S}$ *for a pair of rules is* minimally complete *(resp.* minimally $\mathcal{M}$-complete*) with respect to* parallel dependence *if $\mathcal{S}$ is complete (resp. $\mathcal{M}$-complete) w.r.t. parallel dependence and there exists no other set with smaller cardinality, that is also complete (resp. $\mathcal{M}$-complete) w.r.t. parallel dependence.*

A set of transformation pairs $\mathcal{S}$ *for a pair of rules is* minimally complete *(resp.* minimally $\mathcal{M}$-complete*) with respect to* parallel independence *if $\mathcal{S}$ is complete (resp. $\mathcal{M}$-complete) w.r.t. parallel independence and there exists no other set with smaller cardinality, that is also complete (resp. $\mathcal{M}$-complete) w.r.t. parallel independence.*

[10] On top of strict confluence as in the case of plain rules, also so-called AC-compatibility is required.

In the particular case of non-symbolic transformation pairs, we also say that we have a *minimally complete set of conflicts (or parallel independent transformation pairs)* if we have a set of transformation pairs that is minimally complete w.r.t. parallel dependence (or parallel independence, resp.).

We start with studying minimal completeness w.r.t. parallel independence for plain rules. In Proposition 1 we characterized the set of $\mathcal{M}$-initial parallel independent transformation pairs w.r.t. parallel independence and in Corollary 1 we showed their $\mathcal{M}$-completeness. In Theorem 3, we proved that the set of initial parallel independent transformation pairs consisting just of the transformation pair $tp_{L_1+L_2}$, is also complete w.r.t. parallel independence. We can now show that these sets are minimal.

Proposition 3 (minimal ($\mathcal{M}$-)completeness of ($\mathcal{M}$-initial) parallel independent transformation pairs for plain rules w.r.t parallel independence)**.** *The sets of initial (resp. $\mathcal{M}$-initial) parallel independent transformation pairs for a given pair of plain rules $\langle p_1, p_2 \rangle$ are* minimally complete *(resp. minimally $\mathcal{M}$-complete) w.r.t. parallel independence.*

Proof. In the case of the set of initial parallel independent transformation pairs, the proof is trivial, since the set consists just of one element, up to isomorphism.

In the case of the set $\mathcal{S}$ of $\mathcal{M}$-initial parallel independent transformation pairs, the proof is similar to the previous propositions. Assume that there exists a smaller set $\mathcal{S}'$ being complete (resp. $\mathcal{M}$-complete) w.r.t. parallel dependence. As before, we prove that the cardinality of $\mathcal{S}'$ cannot be smaller than the cardinality of $\mathcal{S}$. If it were, there would exist two parallel independent transformation pairs tp_1, tp_2 for $\langle \rho_1, \rho_2 \rangle$ and transformation pairs $tp'_1, tp'_2 \in \mathcal{S}$, $tp \in \mathcal{S}'$, such that tp'_1 and tp'_2 are not isomorphic, tp'_1 is $\mathcal{M}$-initial for tp_1, tp'_2 is $\mathcal{M}$-initial for tp_2, and tp can be extended to tp_1 and tp_2 via some $\mathcal{M}$-morphisms m_1, m_2. But this would imply, by $\mathcal{M}$-initiality of tp'_1 and tp'_2 that both tp'_1 and tp'_2 can be extended to tp. Consequently, tp'_1 can be extended to tp_2 and tp'_2 can be extended to tp_1. But, by $\mathcal{M}$-initiality of tp'_1 and tp'_2, they would be isomorphic, contradicting the hypothesis. □

We can see that critical pairs and initial conflicts for plain rules are minimally ($\mathcal{M}$-)complete sets of conflicts:

Proposition 4 (minimal ($\mathcal{M}$)-completeness of ($\mathcal{M}$)-initial conflicts for plain rules)**.** *Consider an $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts via plain rules. The set of initial conflicts $\mathcal{S}$ (resp. $\mathcal{M}$-initial conflicts, i.e. critical pairs) up-to-isomorphism, for rules $\langle p_1, p_2 \rangle$ is* minimally complete *(resp. minimally $\mathcal{M}$-complete) w.r.t. parallel dependence.*

Proof. Assume that there exists a smaller set $\mathcal{S}'$ being complete (resp. $\mathcal{M}$-complete) w.r.t. parallel dependence. We can see that the cardinality of $\mathcal{S}'$ cannot be smaller than the cardinality of $\mathcal{S}$. If it were, there would exist two conflicts tp_1, tp_2 for $\langle p_1, p_2 \rangle$ and transformation pairs $tp'_1, tp'_2 \in \mathcal{S}$, $tp \in \mathcal{S}'$, such that tp'_1 and tp'_2 are not isomorphic, tp'_1 is initial (resp. $\mathcal{M}$-initial) for tp_1, tp'_2 is initial (resp. $\mathcal{M}$-initial) for tp_2, and tp can be extended to tp_1 and tp_2 via some morphisms (resp. $\mathcal{M}$-morphisms) m_1, m_2. But this would imply, by initiality (resp. $\mathcal{M}$-initiality) of tp'_1 and tp'_2 that both tp'_1 and tp'_2 can be extended to tp. Consequently, tp'_1 can be extended to tp_2 and tp'_2 can be extended to tp_1. But, by initiality (resp. $\mathcal{M}$-initiality) of tp'_1 and tp'_2, they would be isomorphic, contradicting the hypothesis. □

Critical pairs and initial conflicts for rules with ACs are also minimally $\mathcal{M}$-complete:

Proposition 5 (minimal ($\mathcal{M}$)-completeness of ($\mathcal{M}$)-initial conflicts)**.** *Consider an $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts via plain rules. The set of initial conflicts $\mathcal{S}$ (resp. $\mathcal{M}$-initial conflicts, i.e. critical pairs) up-to-isomorphism, for rules $\langle \rho_1, \rho_2 \rangle$ is* minimally complete *(resp. minimally $\mathcal{M}$-complete) w.r.t. parallel dependence.*

Proof. The proof is very similar to the previous one. Assume that there exists a smaller set $\mathcal{S}'$ being complete (resp. $\mathcal{M}$-complete) w.r.t. parallel dependence. Let $\mathcal{S}^{pl}$ ($\mathcal{S}'^{pl}$) be the equally sized sets of plain transformation pairs via the rules $\langle p_1, p_2 \rangle$ derived from $\mathcal{S}$ and $\mathcal{S}'$, respectively, by extracting merely the corresponding plain transformation pairs.

We can see that the cardinality of $\mathcal{S}'^{pl}$ cannot be smaller than the cardinality of $\mathcal{S}^{pl}$. If it were, there would exist two transformation pairs tp_1, tp_2 for $\langle \rho_1, \rho_2 \rangle$ and transformation pairs $tp'_1, tp'_2 \in \mathcal{S}^{pl}$, $tp \in \mathcal{S}'^{pl}$, such that tp'_1 and tp'_2 are not isomorphic, tp'_1 is initial (resp. $\mathcal{M}$-initial) for tp_1, tp'_2 is initial (resp. $\mathcal{M}$-initial) for tp_2, and tp can be extended to tp_1 and tp_2 via some morphisms (resp. $\mathcal{M}$-morphisms) m_1, m_2. But this would imply, by initiality (resp. $\mathcal{M}$-initiality) of tp'_1 and tp'_2 that both tp'_1 and tp'_2 can be extended to tp. Consequently, tp'_1 can be extended to tp_2 and tp'_2 can be extended to tp_1. But, by initiality (resp. $\mathcal{M}$-initiality) of tp'_1 and tp'_2, they would be isomorphic, contradicting the hypothesis. □

5.2. Minimally ($\mathcal{M}$-)Complete Sets of Conflicts

So far, we have seen that the set of critical pairs (i.e., set of $\mathcal{M}$-initial conflicts) is minimally $\mathcal{M}$-complete, whereas the set of initial conflicts is minimally complete w.r.t. parallel dependence. Then we may ask if the converse is also true when starting from a minimally ($\mathcal{M}$-)complete set of concrete, i.e. non-symbolic, transformation pairs. To this extent we generalize the notion of ($\mathcal{M}$-)initial conflicts to the case of rules with ACs, where the transformation pairs satisfy the ACs and remain concrete, i.e. without reverting to symbolic transformation pairs in order to keep the set of initial conflicts finite (cf. Section 4). The definition of ($\mathcal{M}$-)initial transformation pair basically remains identical, except from the fact that we have rules with ACs and the matches of the transformation pairs satisfy the ACs. Then we can indeed show that a minimally complete (resp. $\mathcal{M}$-complete) set of conflicts equals the sets of *concrete* initial conflicts (resp. $\mathcal{M}$-initial conflicts), provided that ($\mathcal{M}$-)initial transformation pairs for conflicts exist for the considered $\mathcal{M}$-adhesive system.

Definition 19 (($\mathcal{M}$-)initial transformation pair, concrete ($\mathcal{M}$-)initial conflict)**.** *Given a pair of direct transformations $tp : H_1 \Leftarrow_{\rho_1,m_1} G \Rightarrow_{\rho_2,m_2} H_2$, then $tp^I : H_1^I \Leftarrow_{\rho_1,m_1^I} G^I \Rightarrow_{\rho_2,m_2^I} H_2^I$ is an* initial transformation pair *(resp. $\mathcal{M}$-initial transformation pair) for tp if it can be extended to tp via extension diagrams (1) and (2) and extension morphism f^I (resp. $\mathcal{M}$-morphism) as in Figure 2 such that for each transformation pair $tp' : H_1' \Leftarrow_{\rho_1,m_1'} G' \Rightarrow_{\rho_2,m_2'} H_2'$ that can be extended to tp via extension diagrams (3) and (4) and extension morphism (resp. $\mathcal{M}$-morphism) f as in Figure 2 it holds that tp^I can be extended to tp' via unique extension diagrams (5) and (6) and unique vertical morphism (resp. $\mathcal{M}$-morphism) f'^I s.t. $f \circ f'^I = f^I$.*

The fact that, in the case of plain rules, ($\mathcal{M}$-)initial transformation pairs for conflicts exist for $\mathcal{M}$-adhesive systems in particular $\mathcal{M}$-adhesive categories does not mean that the same will happen in the case of rules with ACs. Actually, the following example shows that this is not the case for typed graphs:

Example 4 (no initial transformation pairs)**.** *Assume that we are working with typed graphs including nodes of three types, 1, 2, and 3. Consider rules $\rho_1 = (p_1, \mathrm{ac}_1)$ and $\rho_2 = (p_2, \mathrm{ac}_2)$, where:*

- $p_1 : [\circ_1 \to \circ_1] \leftarrow [\circ_1 \to \circ_1] \to [\circ_1 \to \circ_1 \to \circ_1]$
- $\mathrm{ac}_1 = \mathit{true}$
- $p_2 : [\circ_1 \to \circ_1] \leftarrow [\circ_1\ \circ_1] \to [\circ_1\ \circ_1]$
- $\mathrm{ac}_2 = \Big(\exists(n : [\circ_1 \to \circ_1] \to [\circ_1 \to \circ_1\ \circ_2], \mathit{true}) \vee \exists(n : [\circ_1 \to \circ_1] \to [\circ_1 \to \circ_1\ \circ_3], \mathit{true})\Big)$.

That is, given two nodes of type 1 and an edge between them, p_1 produces an additional outgoing edge to an additional node of type 1; given two nodes of type 1 and an edge between them, p_2 deletes the edge only if the given graph includes a node of type 2 or a node of type 3.

Now consider the conflict $tp_{23} : [\circ_1 \to \circ_1 \to \circ_1\ \ \circ_2\ \ \circ_3] \Leftarrow [\circ_1 \to \circ_1\ \ \circ_2\ \ \circ_3] \Rightarrow [\circ_1\ \circ_1\ \ \circ_2\ \ \circ_3]$, there exist $\mathcal{M}$-morphisms $m_2 : [\circ_1 \to \circ_1\ \ \circ_2] \to [\circ_1 \to \circ_1\ \ \circ_2\ \ \circ_3]$ and $m_3 : [\circ_1 \to \circ_1\ \ \circ_3] \to [\circ_1 \to \circ_1\ \ \circ_2\ \ \circ_3]$. Disregarding the ACs, tp_{23} would have $tp : [\circ_1 \to \circ_1 \to \circ_1] \Leftarrow [\circ_1 \to \circ_1] \Rightarrow [\circ_1\ \circ_1]$ as unique initial transformation pair, witnessing the delete-use conflict that, after applying p_2, we cannot apply p_1. However, tp cannot be an initial transformation pair for tp_{23}, since the match for applying ρ_2 does not satisfy ac_2. Instead, we have two different transformation pairs witnessing the conflict that can be extended to tp_{23}, consisting of $tp_2 : [\circ_1 \to \circ_1 \to \circ_1\ \ \circ_2] \Leftarrow [\circ_1 \to \circ_1\ \ \circ_2] \Rightarrow [\circ_1\ \circ_1\ \ \circ_2]$ and $tp_3 : [\circ_1 \to \circ_1 \to \circ_1\ \ \circ_3] \Leftarrow [\circ_1 \to \circ_1\ \ \circ_3] \Rightarrow [\circ_1\ \circ_1\ \ \circ_3]$ that can be extended to tp_{23}. Now if an *initial transformation pair for tp_{23} exists, it must be either tp_2, tp_3 or tp_{23} itself, since no other transformation pair can be extended to tp_{23}. It cannot be tp_2 (tp_3), since then it should be possible to extend it to tp_3 (tp_2). Moreover, it cannot be tp_{23} itself, since tp_{23} cannot be extended to tp_2 (or tp_3). As a consequence, this example will not have concrete initial conflicts. We can reason analogously for the case of $\mathcal{M}$-initiality using this example.*

Nevertheless, currently we do not know if in the case of rules with some specific kind of ACs, like in the case of NACs, ($\mathcal{M}$-)initial transformation pairs exist.

Definition 20 (existence of ($\mathcal{M}$-)initial transformation pair for conflict)**.** *An $\mathcal{M}$-adhesive system with ACs has* ($\mathcal{M}$-)initial transformation pairs for conflicts *if, for each concrete transformation pair tp in conflict, the ($\mathcal{M}$-)initial transformation pair tp^I exists.*

Similar to the plain case, it can be derived that ($\mathcal{M}$-)initial transformation pairs are unique. Assuming that they exist, we can then define concrete ($\mathcal{M}$-)initial conflicts.

Definition 21 (concrete ($\mathcal{M}$-)initial conflict)**.** *Given an $\mathcal{M}$-adhesive system for which initial (resp. $\mathcal{M}$-initial) transformation pairs for conflicts exists. A parallel dependent transformation pair $tp : H_1 \Leftarrow_{\rho_1,m_1} G \Rightarrow_{\rho_2,m_2} H_2$ is a* concrete initial conflict *(resp. concrete $\mathcal{M}$-initial conflict) if it is equal (up to isomorphism) to its initial transformation pair (resp. $\mathcal{M}$-)initial transformation pair.*

Now, we can show that if a given $\mathcal{M}$-adhesive system has ($\mathcal{M}$-)initial transformation pairs for conflicts, then any minimally ($\mathcal{M}$-)complete set of conflicts must be equal, up to isomorphism, to the set of concrete ($\mathcal{M}$-)initial conflicts. This means that, in the case of plain rules (where $\mathcal{M}$-initial transformation pairs exist), the only minimally ($\mathcal{M}$-)complete sets of conflicts are the sets of ($\mathcal{M}$-)initial conflicts. But this is not necessarily true in the case of rules with ACs.

Theorem 7 (minimally ($\mathcal{M}$-)complete sets of conflicts are concrete ($\mathcal{M}$-)initial conflicts)**.** *Given an $\mathcal{M}$-adhesive system with ($\mathcal{M}$-)initial transformation pairs for conflicts, if $\mathcal{S}$ is a set of conflicts that is minimally ($\mathcal{M}$-)complete w.r.t. parallel dependence for rules $\langle\rho_1, \rho_2\rangle$ then $\mathcal{S}$ consists of all concrete ($\mathcal{M}$-)initial conflicts.*

Proof. Let tp' be a concrete initial conflict, since $\mathcal{S}$ is assumed to be complete there is a $tp \in \mathcal{S}$, such that there is a morphism (resp. $\mathcal{M}$-morphism) h, such that h extends tp to tp'. But we also have that, if tp'' is the initial transformation pair of tp', then there is a morphism (resp. $\mathcal{M}$-morphism) g, such that g extends tp'' to tp. This means that $h \circ g$ extends tp'' into tp', implying that tp, tp' and tp'' are isomorphic, i.e., tp is initial. As a consequence, we may conclude that all concrete initial conflicts are in $\mathcal{S}$, but if $\mathcal{S}$ is minimal then no other transformation pair should be in $\mathcal{S}$. $\square$

5.3. Constructing Minimally ($\mathcal{M}$-)Complete Sets of Conflicts from $\mathcal{M}$-Complete Sets

Given an $\mathcal{M}$-complete set of conflicts $\mathcal{S}$, we are going to see that we can reduce it to a minimally ($\mathcal{M}$-)complete set of conflicts, $\mathcal{S}'$, by removing all its *($\mathcal{M}$-)redundant* transformation pairs. We will use this reduction in the following section to show that for special cases of rules with ACs, we can construct minimally ($\mathcal{M}$-)complete sets of conflicts. In particular, we will see that this works for special initial conflicts, for which an unfolding into an $\mathcal{M}$-complete set of conflicts exists indeed. Note also that because of Theorem 7 we can use the reduction in particular also to reduce any given $\mathcal{M}$-complete set of conflicts into the set of ($\mathcal{M}$-)initial conflicts (provided that they exist for the given transformation system).

Definition 22 (($\mathcal{M}$-)redundant transformation pair, ($\mathcal{M}$-)maximal reduction)**.** *Given a set of transformation pairs $\mathcal{S}$ for a pair of rules $\langle\rho_1, \rho_2\rangle$, we say that tp is* redundant *(resp. $\mathcal{M}$-redundant) with respect to $\mathcal{S}$ if there is a transformation pair $tp' \in \mathcal{S}$ that can be extended to tp via an extension morphism h (resp. an extension ($\mathcal{M}$-)morphism).*

We say that $\mathcal{S}'$ is a maximal reduction *of $\mathcal{S}$ (resp. an $\mathcal{M}$-maximal reduction), if $\mathcal{S}' \subseteq \mathcal{S}$, every transformation pair in $\mathcal{S} \setminus \mathcal{S}'$ is redundant (resp. $\mathcal{M}$-redundant) with respect to $\mathcal{S}'$, and no $tp \in \mathcal{S}'$ is redundant (resp. $\mathcal{M}$-redundant) with respect to $\mathcal{S}' \setminus \{tp\}$.*

Theorem 8. *Given a set $\mathcal{S}$ of transformation pairs, if $\mathcal{S}$ is $\mathcal{M}$-complete with respect to parallel dependence for rules $\langle\rho_1, \rho_2\rangle$, then:*

1. *If $\mathcal{S}'$ is an $\mathcal{M}$-maximal reduction of $\mathcal{S}$, then $\mathcal{S}'$ is minimally $\mathcal{M}$-complete with respect to parallel dependence for $\langle\rho_1, \rho_2\rangle$.*
2. *Similarly, if $\mathcal{S}'$ is a maximal reduction of $\mathcal{S}$, then $\mathcal{S}'$ is minimally complete with respect to parallel dependence for $\langle\rho_1, \rho_2\rangle$.*

Proof. We prove the second statement. The proof for the first statement is completely analogous.

Let tp be a conflict for rules $\langle\rho_1, \rho_2\rangle$. Because of $\mathcal{M}$-completeness of $\mathcal{S}$ there must exist some $tp_0 \in \mathcal{S}$ such that tp_0 can be extended to tp via some $\mathcal{M}$-morphism m. In case that $tp_0 \notin \mathcal{S}'$, there must be some $tp_0' \in \mathcal{S}'$ such that tp_0' can be extended to tp_0 via some morphism m'. Therefore, tp_0' can be extended to tp via $m' \circ m$. Otherwise, if $tp_0 \in \mathcal{S}'$, we trivially have that tp_0 can be extended to tp via m.

For minimality, let us assume that $\mathcal{S}''$ is another complete set of transformation pairs with respect to parallel dependence for rules $\langle\rho_1, \rho_2\rangle$. Moreover, let us assume, without loss of generality, that $\mathcal{S}''$ does not include any proper subset that is also complete. Let us see that for every $tp'' \in \mathcal{S}''$, there must exist a $tp' \in \mathcal{S}'$, such that tp'' can be extended to tp' via some morphism g. Let us suppose that there is no $tp' \in \mathcal{S}'$, such that tp'' can be extended to tp'. Since $\mathcal{S}'$ is complete, there must exist a $tp_1' \in \mathcal{S}'$ such that tp_1' can be extended to tp'' via some morphism h and, since

$\mathcal{S}''$ is complete, there must exist a $tp''_1 \in \mathcal{S}''$, with $tp''_1 \neq tp''$, such that tp''_1 can be extended to tp'_1 via some morphism h', implying that tp''_1 can be extended to tp'_1 via $h \circ h'$, contradicting the hypothesis that $\mathcal{S}''$ does not include any proper subset that is also complete. But, if for every every $tp'' \in \mathcal{S}''$, there is a $tp' \in \mathcal{S}'$, such that tp'' can be extended to tp' via some morphism g, this means that the cardinality of $\mathcal{S}'$ is smaller or equal than the cardinality of $\mathcal{S}''$.

□

Note that for obtaining a minimally complete set, we even obtain a slightly more general result as the one in the above theorem in the sense that we can also start with a complete set (instead of an $\mathcal{M}$-complete one). This generalization also holds for the following corollary in the sense that we can obtain under the given conditions the set of concrete initial conflicts starting from a minimally complete (instead of an $\mathcal{M}$-complete) set.

Corollary 2. *Given an $\mathcal{M}$-adhesive system with ($\mathcal{M}$-)initial transformation pairs for conflicts, and given a set $\mathcal{S}$ of transformation pairs, if $\mathcal{S}$ is $\mathcal{M}$-complete with respect to parallel dependence for rules $\langle \rho_1, \rho_2 \rangle$, then:*

1. *If $\mathcal{S}'$ is an $\mathcal{M}$-maximal reduction of $\mathcal{S}$, then $\mathcal{S}'$ is the set of concrete $\mathcal{M}$-initial conflicts for $\langle \rho_1, \rho_2 \rangle$.*
2. *Similarly, if $\mathcal{S}'$ is a maximal reduction of $\mathcal{S}$, then $\mathcal{S}'$ is the set of concrete initial conflicts for $\langle \rho_1, \rho_2 \rangle$.*

Proof. Direct consequence of Theorem 7 and Theorem 8. □

6. Unfoldings of Initial Conflicts

Initial conflicts are a very compact (but symbolic) way of representing the set of all parallel dependent transformation pairs for rules with ACs. However, from a user point of view they may not provide much intuition about where are the problems that give rise to these conflicts, especially in the case of AC-conflicts. Nevertheless, even if we have seen in Theorem 4 that, in general, given two finite graph transformation rules with ACs, there is no finite set of transformation pairs which is complete or $\mathcal{M}$-complete, there are special cases where two rules with ACs have complete sets of transformation pairs which are finite. For example, this is true in the prominent case of rules with negative application conditions (NACs), as shown in [8, 3]. Hence, in these cases being able to compute minimally complete or $\mathcal{M}$-complete sets of conflicts would provide a compact and intuitive solution to the problem of characterizing the existing conflicts between two rules with ACs.

In this section, we show a *sufficient condition* for being able to unfold initial conflicts into an *$\mathcal{M}$-complete set of conflicts* that is *finite* if the set of initial conflicts is finite (cf. subsection 6.1).

We demonstrate moreover that this sufficient condition is fulfilled for the special case of having merely *NACs* as rule application conditions (cf. subsection 6.2). Finally, we show that in this case we can obtain finite sets of conflicts that are minimally complete for rules with NACs, which in general are subsets of the critical pairs for rules with NACs, as introduced in [8].

6.1. Finite and $\mathcal{M}$-Complete Unfolding

We introduce so-called *regular initial conflicts* leading to $\mathcal{M}$-complete sets of conflicts, by unfolding them in some particular way (cf. *disjunctive unfolding* in Definition 23). The idea is that the extension and conflict-inducing AC (ac_K and ac^*_K, respectively) of such a regular initial conflict $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ have a specific form that is amenable to finding $\mathcal{M}$-complete unfoldings. We expect the condition $\mathrm{ac}_K \wedge \mathrm{ac}^*_K$ to consist of a *disjunction of positive literals* (conditions of the form $\exists(a_i : K \to C_i, c_i)$) with a so-called *negative remainder* (i.e. a condition $c_i = \wedge_{j \in J} \neg\exists(b_j : C_i \to C_j, d_j)$). Intuitively, this means that there is a finite number of possibilities to unfold the symbolic conflict into a concrete conflict by adding some specific positive context (expressed by the morphism a_i). The negative remainder c_i ensures that by adding this positive context to the context K of the symbolic transformation pair within the initial conflict, we indeed find a conflict when not extending further at all. Moreover, it expresses under which condition the corresponding concrete representative conflict leads to further conflicts by extension. Finally, the sets of $\mathcal{M}$-complete conflicts built using the disjunctive unfolding can be shown to be *finite* if the set of initial conflicts it is derived from is finite.

Definition 23 (regular initial conflict, disjunctive unfolding)**.** *Consider an $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts along plain rules. Given an initial conflict stp_K : $\langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ for rules $\langle \rho_1, \rho_2 \rangle$, then we say that it is* regular *if $\mathrm{ac}_K \wedge \mathrm{ac}^*_K$ is equivalent to a condition $c = \vee_{i \in I} \exists(a_i : K \to C_i, c_i)$ with $c_i = \wedge_{j \in J} \neg\exists(b_j : C_i \to C_j, d_j)$ a condition on C_i, b_j non-isomorphic and I some non-empty index set. Given a regular initial conflict stp_K : $\langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$, then $\mathcal{U}^{\mathcal{D}}_c(stp_K) = \cup_{i \in I}\{tp_{C_i} : D_{1,i} \Leftarrow_{\rho_1, a_i \circ o_1} C_i \Rightarrow_{\rho_2, a_i \circ o_2} D_{2,i}\}$ is the* disjunctive unfolding *of stp_K associated to c.*

In the following, for simplicity, we will just write $\mathcal{U}^{\mathcal{D}}(stp_K)$ if c can be left implicit.

Remark 7 (disjunctive unfolding)**.** *The disjunctive unfolding of a regular conflict is non-empty, but might consist of less elements than literals in the disjunction $\vee_{i \in I} \exists(a_i : K \to C_i, c_i)$. It might be the case that some of the morphisms a_i do not satisfy the gluing condition of the derived spans. If this is the case, then also every extension morphism starting from there will not satisfy the gluing condition such that we can safely ignore these cases from the disjunctive unfolding.*

Theorem 9 (finite and $\mathcal{M}$-complete unfolding)**.** *Consider an $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts along plain rules. Given a rule pair $\langle \rho_1, \rho_2 \rangle$ with set $\mathcal{S}$ of initial* *conflicts such that each initial conflict stp in S is regular, then* $\cup_{stp \in \mathcal{S}} \mathcal{U}^{\mathcal{D}}(stp)$ *is* $\mathcal{M}$*-complete w.r.t. parallel dependence. Moreover,* $\cup_{stp \in \mathcal{S}} \mathcal{U}^{\mathcal{D}}(stp)$ *is finite if* $\mathcal{S}$ *is finite.*

Proof. Because each disjunctive unfolding of a regular initial conflict consists of a finite number of elements (see finite index set Definition 3), the set $\cup_{stp \in \mathcal{S}} \mathcal{U}^{\mathcal{D}}(stp)$ is finite as soon as the set $\mathcal{S}$ of all initial conflicts is finite.

We now show that the set $\cup_{stp \in \mathcal{S}} \mathcal{U}^{\mathcal{D}}(stp)$ (consisting of transformation pairs) is also $\mathcal{M}$-complete w.r.t. parallel dependence. From Theorem 6 we know that the set of initial conflicts S (consisting of symbolic transformation pairs) is complete w.r.t. parallel dependence. This means that there exists some $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ with AC-disregarding transformation pair $tp_K : P_1 \Leftarrow_{\rho_1, o_1} K \Rightarrow_{\rho_2, o_2} P_2$ from S that can be extended to tp_G via some extension morphism $m : K \to G$ with $m \models \mathrm{ac}_K \wedge \mathrm{ac}^*_K$.

Consequently, since $m \models \mathrm{ac}_K \wedge \mathrm{ac}^*_K$ we know that because of having only regular initial conflicts $m \models \vee_{i \in I} \exists(a_i : K \to C_i, c_i)$. This means that $m \models \exists(a_i : K \to C_i, c_i)$ for some $i \in I$ meaning that there exists some $q_i : C_i \to G \in \mathcal{M}$ such that $q_i \models c_i$ and $q_i \circ a_i = m$. Because of the Restriction Theorem for plain rules [27] and the fact that q_i is in $\mathcal{M}$ we know that there exists a pair of plain transformations via matches $a_i \circ o_1$ and $a_i \circ o_2$ that can be extended to tp_G via extension morphism q_i. Now we have to show that the matches $a_i \circ o_1$ and $a_i \circ o_2$ of this transformation pair tp_{C_i} indeed satisfy the conditions ac_{L_1} and ac_{L_2}, respectively. Moreover, we argue that the transformation pair tp_{C_i} is conflicting. To this extent, consider the identity morphism id_{C_i} satisfying trivially c_i. Consequently, $a_i \models \exists(a_i : K \to C_i, c_i)$, and because of regularity it follows that $a_i \models \mathrm{ac}_K \wedge \mathrm{ac}^*_K$. By the Embedding Theorem [10, 30] it then follows that we indeed obtain a pair of transformations with $a_i \circ o_1$ and $a_i \circ o_2$ satisfying the rule ACs ac_{L_1} and ac_{L_2}, since $a_i \models \mathrm{ac}_K$ making it AC-consistent for both AC-disregarding transformations in tp_K indeed. Moreover, because of Lemma 6.2 (characterization of parallel dependency with ACs) in [10, 30] tp_{C_i} is also parallel dependent, since $a_i \models \mathrm{ac}^*_K$.

Since pushouts and pushout complements are unique up to isomorphism this pair of transformations tp_{C_i} (built for the matches $a_i \circ o_1$ and $a_i \circ o_2$) is indeed equivalent to some transformation pair from $\mathcal{U}^{\mathcal{D}}(stp_K)$. As a consequence we have indeed found an extension diagram extending $tp_{C_i} : D_{1,i} \Leftarrow_{\rho_1, a_i \circ o_1} C_i \Rightarrow_{\rho_2, a_i \circ o_2} D_{2,i}$ in $\mathcal{U}^{\mathcal{D}}(stp_K)$ to tp_G via q_i. $\square$

It is possible to automatically check if some initial conflict is regular by using dedicated automated reasoning [25] as well as symbolic model generation for ACs [31] as follows. The reasoning mechanism [25] is shown to be refutationally complete ensuring that if the condition $\mathrm{ac}_K \wedge \mathrm{ac}^*_K$ of some initial conflict is unsatisfiable, this will be detected eventually. Moreover, the related symbolic model generation mechanism [31] is able to automatically transform each condition $\mathrm{ac}_K \wedge \mathrm{ac}^*_K$ into some disjunction $\vee_{i \in I} \exists(a_i : K \to C_i, c_i)$ with c_i a negative remainder if such an equivalence holds.

We can reduce the (finite) $\mathcal{M}$-complete disjunctive unfolding to a (finite) minimally ($\mathcal{M}$-)complete set of transformation pairs.

Corollary 3 (minimally ($\mathcal{M}$-)complete unfolding)**.** *Under the same conditions of Theorem 9, if $\mathcal{S}'$ is a maximal reduction (resp. $\mathcal{M}$-maximal reduction) of $\cup_{stp \in \mathcal{S}} \mathcal{U}^{\mathcal{D}}(stp)$, then $\mathcal{S}'$ is minimally complete (resp. $\mathcal{M}$-complete) with respect to parallel dependence for the same rules.*

Proof. Direct consequence of Theorem 8, and Theorem 9 $\square$

6.2. Unfolding for Rules with NACs

We show that in the case of having rules with NACs[11], initial conflicts are regular. This means that in this special case there exists a complete set of conflicts that is e.g. in the case of graphs (and assuming finite rules) also finite. This conforms to the findings in [8, 3], where an $\mathcal{M}$-complete set of critical pairs – as specific set of conflicts – for graph transformation rules with NACs was introduced [8] (and generalized to $\mathcal{M}$-adhesive transformation systems [3]).

Theorem 10 (regular initial conflicts for rules with NACs)**.** *Consider an $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts along plain rules. Every initial conflict stp_K : $\langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ for a pair of rules $\langle \rho_1, \rho_2 \rangle$ with $\mathrm{ac}_{L_i} = \wedge_{j \in J_i} \neg \exists n_j : L_i \to N_j$ for $i = 1, 2$ and J_i some finite index set, is* regular*. In particular, $\mathrm{ac}_K \wedge \mathrm{ac}^*_K$ is equivalent to a condition $\vee_{i \in I} \exists (a_i : K \to C_i, c_i)$ with $c_i = \wedge_{q \in Q} \neg \exists n_q$ a condition on C_i and I some non-empty index set.*

Proof. This follows directly from Definition 7 and the constructions [30] related to Lemma 2 and Lemma 1. In particular, ac_K (arising from shifting each rule NAC over the match morphisms into K) consists of a conjunction of NACs again, and ac^*_K becomes true or consists of a (non-empty) disjunction of PACs. We obtain by shifting (using Lemma 1) each NAC over each PAC morphism ($\exists id_K$ in the case ac^*_K becomes true) a condition that is equivalent to a disjunction of literals of the form $\exists (a_i : K \to C_i, \wedge_{q \in Q} \neg \exists n_q)$. $\square$

The negative remainder c_i of each literal in $\vee_{i \in I} \exists (a_i : K \to C_i, c_i)$ of a regular initial conflict for rules with NACs thus consists of a set of NACs. Intuitively this means that we obtain for each initial conflict an $\mathcal{M}$-complete set of conflicts by adding the context described by a_i. As long as no NAC from c_i is violated we can extend such a conflict to further ones.

Corollary 4 ($\mathcal{M}$-complete unfolding: rules with NACs)**.** *Consider an $\mathcal{M}$-adhesive system, with initial transformation pairs for conflicts along plain rules. Given a rule pair $\langle \rho_1, \rho_2 \rangle$ with $\mathrm{ac}_{L_i} = \wedge_{j \in J_i} \neg \exists n_j : L_i \to N_j$ for $i = 1, 2$, then $\cup_{stp \in \mathcal{S}} \mathcal{U}^{\mathcal{D}}(stp)$ is $\mathcal{M}$-complete w.r.t. parallel dependence.*

[11] A rule with NACs consists of a plain rule with a conjunction of NACs as application condition, which is the most common way of using NACs since their introduction in [22].

Proof. This follows directly from Theorem 9 and Theorem 10. □

We show moreover that the initial conflict definition is a *conservative extension* of the critical pair definition for rules with NACs as given in [8, 3]. In particular, we show that each conflict in the disjunctive unfolding of an initial conflict as chosen in the proof of Theorem 10 is in particular a critical pair for rules with NACs. Note that a critical pair for rules with NACs is a conflicting pair of transformations such that (1) its plain transformations have jointly epimorphic matches and are use-delete/delete-use conflicting, or (2) the transformations are AC conflicting (and possibly also use-delete/delete-use conflicting) in such a way that one of the rules produces elements responsible for violating one of the NACs not violated yet before rule application without considering additional context not stemming already from one of the rules or the violated NAC (i.e. technically the morphism violating the NAC and the corresponding co-match need to be jointly surjective).

Let us recall the definition[12] of critical pairs for rules with NACs [8], before showing that initial conflicts for rules with ACs as defined in this paper represent a conservative extension in the sense of Theorem 11.

Definition 24 (critical pair)**.** *A* critical pair *is a pair of direct transformations $K \overset{p_1,m_1}{\Rightarrow} P_1$ with NAC_{p_1} and $K \overset{p_2,m_2}{\Rightarrow} P_2$ with NAC_{p_2} such that:*

1. (a) $\neg\exists h_{12} : L_1 \to D_2 : d_2 \circ h_{12} = m_1$ *and* (m_1, m_2) *in* $\mathcal{E}'$
   (use-delete conflict)
   or
   (b) *there exists* $h_{12} : L_1 \to D_2$ *s.t.* $d_2 \circ h_{12} = m_1$*, but for one of the NACs* $n_1 : L_1 \to N_1$ *of* p_1 *there exists a morphism* $q_{12} : N_1 \to P_2 \in \mathcal{M}$ *s.t.* $q_{12} \circ n_1 = e_2 \circ h_{12}$*, and thus,* $e_2 \circ h_{12} \not\models NAC_{n_1}$*, and* (q_{12}, m'_2) *in* $\mathcal{E}'$ *(forbid-produce conflict)*

or

2. (a) $\neg\exists h_{21} : L_2 \to D_1 : d_1 \circ h_{21} = m_2$ *and* (m_1, m_2) *in* $\mathcal{E}'$
   (delete-use conflict)
   or
   (b) *there exists* $h_{21} : L_2 \to D_1$ *s.t.* $d_1 \circ h_{21} = m_2$*, but for one of the NACs* $n_2 : L_2 \to N_2$ *of* p_2 *there exists a morphism* $q_{21} : N_2 \to P_1 \in \mathcal{M}$ *s.t.* $q_{21} \circ n_2 = e_1 \circ h_{21}$*, and thus,* $e_1 \circ h_{21} \not\models NAC_{n_2}$*, and* (q_{21}, m'_1) *in* $\mathcal{E}'$ *(produce-forbid conflict)*

[12] We assume that the class $\mathcal{Q} = \mathcal{M}$, since we for simplicity do not distinguish between morphisms used to satisfy (or violate) a graph condition ($\mathcal{Q}$-morphisms) and $\mathcal{M}$-morphisms (as analogously assumed in the previous seminal work w.r.t. rules with ACs [10, 30].

$$
\begin{tikzcd}
 & & N_1 \arrow[rr, dotted, no head] \arrow[ddll, dotted, bend right, "q_{21}"'] & & N_2 \arrow[ddrr, dotted, bend left, "q_{12}"] & & \\
R_1 \arrow[d, "m_1'"'] & K_1 \arrow[l, "r_1"'] \arrow[r, "l_1"] \arrow[d] & L_1 \arrow[u, "n_1"] \arrow[dr, "m_1"] \arrow[drrr, dotted, "h_{12}"] & & L_2 \arrow[u, "n_2"] \arrow[dl, "m_2"'] \arrow[dlll, dotted, "h_{21}"'] & K_2 \arrow[l, "l_2"'] \arrow[r, "r_2"] \arrow[d] & R_2 \arrow[d, "m_2'"] \\
P_1 & D_1 \arrow[l, "e_1"] \arrow[rr, "d_1"'] & & K & & D_2 \arrow[ll, "d_2"] \arrow[r, "e_2"'] & P_2
\end{tikzcd}
$$

Theorem 11 (conservative unfolding)**.** *Consider an $\mathcal{M}$-adhesive system with initial transformation pairs for conflicts along plain rules. Given some initial conflict $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ for a pair of rules $\langle \rho_1, \rho_2 \rangle$ with $\mathrm{ac}_{L_i} = \wedge_{j \in J_i} \neg \exists n_j : L_i \to N_j$ for $i = 1, 2$ and J_i some finite index set, then each conflict as chosen in the proof of Theorem 10 in $\mathcal{U}^{\mathcal{D}}(stp)$ is in particular a critical pair for $\langle \rho_1, \rho_2 \rangle$ as given in [8, 3].*

Proof. Recall that an initial conflict $stp_K : \langle tp_K, \mathrm{ac}_K, \mathrm{ac}^*_K \rangle$ consists in particular of an initial conflict for plain rules or of an initial parallel independent pair of transformations for plain rules. Having rules with NACs only, we can unfold such an initial conflict into a set of conflicting transformations tp_{C_i} with each conflict stemming from one literal in the finite disjunction $\vee_{i \in I} \exists (a_i : K \to C_i, c_i)$ with c_i a condition of the form $\wedge_{q \in Q} \neg \exists n_q$. When extending the initial parallel independent pair via some $a_i : L_1 + L_2 \to C_i$, the corresponding transformation pair remains plain parallel independent such that we in particular obtain a critical pair satisfying (1.b) or (2.b) according to Definition 24. Moreover we know that (q_{12}, m_2') and (q_{21}, m_1') belong to $\mathcal{E}'$ by construction and we know that tp_{C_i} is AC-conflicting indeed. In case that we extend an initial conflict for the plain rules to a real conflict for the rules with NACs, we obtain a critical pair either satisfying (1.a) or (2.a) according to Definition 24 in case no additional context is added by the positive application condition a_i stemming from the disjunction $\vee_{i \in I} \exists (a_i : K \to C_i, c_i)$ in the unfolding, or satisfying (1.b) or (2.b) as in the previous case. $\square$

Example 5 (conservative unfolding)**.** *Consider again the rules from Example 1 (having only NACs as ACs) and their application to the graph $G = \bigcirc\ \bigcirc$. The corresponding transformation pair tp_G is a critical pair for rules with NACs as given in [8, 3]. This is because it is in particular a conflicting pair of transformations, and the morphism violating the NAC (since finding the three nodes) and therefore causing the conflict after applying the first rule to $G = \bigcirc\ \bigcirc$ obtaining some graph $H_1 = \bigcirc\!\!\rightarrow\!\!\bigcirc\ \bigcirc$ is jointly surjective together with the corresponding co-match. As argued already in Example 2 this critical pair for rules with NACs belongs to the unfolding (and in particular to the disjunctive unfolding) of the unique AC initial conflict $stp_{L_1+L_2} : \langle tp_{L_1+L_2}, \mathrm{ac}_{L_1+L_2}, \mathrm{ac}^*_{L_1+L_2} \rangle$.*

Critical pairs for rules with NACs as introduced in [8, 3] are not minimally $\mathcal{M}$-complete as the following example illustrates. We can however reduce it to a finite, minimally ($\mathcal{M}$-)complete subset.

Example 6 (Critical pairs for rules with NACs are not minimally $\mathcal{M}$-complete)**.** *Consider the rules* $p_1 : \bigcirc \leftarrow \bigcirc \rightarrow \bigcirc\!\!\rightarrow\!\!\bigcirc$ *with NAC* $\neg\exists n : \bigcirc \rightarrow \bigcirc\ \bigcirc\ \bigcirc$ *and* $p_2 : \bigcirc \leftarrow \bigcirc \rightarrow \bigcirc\ \bigcirc$ *with NAC* $\neg\exists n : \bigcirc \rightarrow \bigcirc\!\!\rightarrow\!\!\bigcirc$*. Then we have a critical pair* tp_G *starting from* $G = \bigcirc\ \bigcirc$ *by applying both rules to* G *with the matches sharing one node in* G*. This critical pair is a produce-forbid/forbid-produce conflict. Now consider the graph* $G' = \bigcirc$ *and the critical pair* $tp_{G'}$ *starting from* G' *by applying both rules to* G' *with the matches sharing the only node in* G'*. This critical pair is a produce-forbid conflict, but not a forbid-produce conflict. Now* $tp_{G'}$ *can be extended injectively to* tp_G *illustrating that the set of critical pairs for rules with NACs is not minimally* $\mathcal{M}$*-complete and can still be reduced.*

Corollary 5 (concrete ($\mathcal{M}$-)initial conflicts for rules with NACs)**.** *Under the same conditions of Corollary 4, if* $\mathcal{S}$ *is a maximal reduction (resp.* $\mathcal{M}$*-maximal reduction) of the set of critical pairs for* $\langle \rho_1, \rho_2 \rangle$*, as given in [8, 3], then* $\mathcal{S}$ *is minimally complete (resp. minimally* $\mathcal{M}$*-complete) with respect to parallel dependence for* $\langle \rho_1, \rho_2 \rangle$*.*

Proof. Direct consequence of Theorem 7, Theorem 8, and Theorem 11 □

7. Conclusion and Outlook

In this paper we have *generalized the theory of initial conflicts* (from plain rules, i.e. rules without application conditions) to *rules with application conditions* (ACs) in the framework of $\mathcal{M}$-adhesive transformation systems as summarized in Table 1.

We build on the notion of symbolic transformation pairs, since it turns out that it is not possible to find a complete set of concrete conflicting transformation pairs in the case of rules with ACs. We have shown that initial conflicts are complete w.r.t. parallel dependence as symbolic transformation pairs. Moreover, initial conflicts represent (analogous to the case of plain rules) proper subsets of critical pairs in the sense that for each critical pair (or also for each conflict), there exists a unique initial conflict representing it. We have shown that initial conflicts (resp. critical pairs) are *minimally complete* (resp. minimally ($\mathcal{M}$-)complete), in the case of plain rules and rules with ACs. In addition, we have shown how to extract a minimally ($\mathcal{M}$-)complete set of transformation pairs from an $\mathcal{M}$-complete one. We concluded the paper by showing sufficient conditions for finding unfoldings of initial conflicts that lead to (finite and) minimally ($\mathcal{M}$-)complete sets of conflicts (in particular for the case of rules with NACs). Thereby we have shown that initial conflicts for rules with ACs represent a conservative extension of the critical pair theory for rules with NACs.

As future work we want to study in more detail the case of *rules with NACs*. We have seen that critical pairs, introduced and proved to be $\mathcal{M}$-complete in [8, 3] are not minimal, though our techniques show how to extract a minimally ($\mathcal{M}$-)complete subset. However the straightforward

	plain rules	**rules with NACs**	**rules with ACs**
critical pairs (CPs)	set of conflicts, $\mathcal{M}$-complete [14, 15, 16, 27]	set of conflicts, $\mathcal{M}$-complete [8, 3]	symbolic $\mathcal{M}$-complete [9, 10]
minimally $\mathcal{M}$-complete	yes (Prop. 4)	no (Example 5)	yes (Prop. 5)
existence of min. $\mathcal{M}$-complete finite set of conflicts?	yes, CPs	yes, proper subset of CPs (Cor. 5)	not guaranteed (Thm. 4)
initial conflicts	proper subset of CPs, complete [6, 7]	symbolic (Def. 17), regular (Thm. 10), conservative extension of CPs (Thm. 11)	symbolic (Def. 17), proper subset of CPs (Thm. 5) conservative extension of CPs (Prop. 4.3)
minimally complete	yes (Prop. 4)	yes (Prop. 5)	yes (Prop. 5)
existence of min. complete finite set of conflicts?	yes, initial conflicts	yes, proper subset of CPs (Cor. 5)	not guaranteed (Thm. 4)

Table 1: Critical pairs versus initial conflicts

implementation of this reduction would probably not be very efficient. Hence, we would like to find efficient procedures for their computation. This will be based on the study of the open problem if ($\mathcal{M}$-)initial transformation pairs exist for the case of NACs.

We also aim at finding *further interesting classes* allowing finite and (minimally) complete unfoldings into *sets of conflicts*. This will serve as a guideline to be able to *develop and implement efficient conflict detection* techniques for rules with (specific) ACs, which has been an open challenge until today.

We are moreover planning to develop *(semi-)automated detection of unfoldings* of initial conflicts into concrete conflicts for rules with arbitrary ACs using dedicated automated reasoning and model finding for graph conditions [32, 25, 31]. It would be interesting to investigate in which *use cases* initial conflicts (or critical pairs) are useful already as symbolic transformation pairs, and in which use cases we rather need to consider unfoldings indeed. This is in line with the research on multi-granular conflict detection [18, 4, 19] investigating different levels of granularity that can be interesting from the point of view of applying conflict detection to different use cases.

Finally, we plan to investigate conflict detection in the light of initial conflict theory for *attributed graph transformation* [27, 33, 34], and in particular the case of rules with so-called attribute conditions more specifically. It would also be interesting to further investigate initial conflicts for transformation rules (with ACs) not following the DPO approach. For example, one may consider the single-pushout (SPO) approach introduced in [35], which is a generalization of the DPO framework where only one morphism defines the rule, which may be partial to allow deletion. In [22], SPO rules with negative application conditions are considered and the Local Confluence and Parallelism Theorems are shown. As far as we know, a theory on SPO rules with nested application conditions is missing. Moreover, the implications of initial conflict theory for the case of *graphs with inheritance* [36] or *rule amalgamation* [37, 38] need to be further investigated.

Acknowledgement. We thank Jens Kosiol for pointing out that the set of initial conflicts for plain rules is not only complete, but also minimally complete. We moreover thank the anonymous reviewers for their valuable comments helping us to improve our paper.

[1] J. H. Hausmann, R. Heckel, G. Taentzer, Detection of conflicting functional requirements in a use case-driven approach: a static analysis technique based on graph transformation, in: W. Tracz, M. Young, J. Magee (Eds.), Proceedings of the 24th International Conference on Software Engineering, ICSE 2002, ACM, 2002, pp. 105–115. `doi:10.1145/581339.581355`.

[2] M. Koch, L. V. Mancini, F. Parisi-Presicce, Graph-based specification of access control policies, J. Comput. Syst. Sci. 71 (1) (2005) 1–33. `doi:10.1016/j.jcss.2004.11.002`.

[3] L. Lambers, Certifying rule-based models using graph transformation, Ph.D. thesis, Berlin

Institute of Technology (2009).
URL http://opus.kobv.de/tuberlin/volltexte/2010/2522/

[4] L. Lambers, D. Strüber, G. Taentzer, K. Born, J. Huebert, Multi-granular conflict and dependency analysis in software engineering based on graph transformation, in: M. Chaudron, I. Crnkovic, M. Chechik, M. Harman (Eds.), Proceedings of the 40th International Conference on Software Engineering, ICSE 2018, ACM, 2018, pp. 716–727. doi:10.1145/3180155.3180258.

[5] L. Lambers, H. Ehrig, F. Orejas, Efficient conflict detection in graph transformation systems by essential critical pairs, Electr. Notes Theor. Comput. Sci. 211 (2008) 17–26.

[6] L. Lambers, K. Born, F. Orejas, D. Strüber, G. Taentzer, Initial conflicts and dependencies: Critical pairs revisited, in: R. Heckel, G. Taentzer (Eds.), Graph Transformation, Specifications, and Nets - In Memory of Hartmut Ehrig, Vol. 10800 of LNCS, Springer, 2018, pp. 105–123. doi:10.1007/978-3-319-75396-6_6.

[7] G. G. Azzi, A. Corradini, L. Ribeiro, On the essence and initiality of conflicts in m-adhesive transformation systems, J. Log. Algebr. Meth. Program. 109.

[8] L. Lambers, H. Ehrig, F. Orejas, Conflict detection for graph transformation with negative application conditions, in: A. Corradini, H. Ehrig, U. Montanari, L. Ribeiro, G. Rozenberg (Eds.), Graph Transformations, Third International Conference, ICGT 2006, Vol. 4178 of LNCS, Springer, 2006, pp. 61–76. doi:10.1007/11841883_6.

[9] H. Ehrig, A. Habel, L. Lambers, F. Orejas, U. Golas, Local confluence for rules with nested application conditions, in: H. Ehrig, A. Rensink, G. Rozenberg, A. Schürr (Eds.), Graph Transformations - 5th International Conference, ICGT 2010, Vol. 6372 of LNCS, Springer, 2010, pp. 330–345. doi:10.1007/978-3-642-15928-2_22.

[10] H. Ehrig, U. Golas, A. Habel, L. Lambers, F. Orejas, $\mathcal{M}$-adhesive transformation systems with nested application conditions. part 2: Embedding, critical pairs and local confluence, Fundam. Inform. 118 (1-2) (2012) 35–63. doi:10.3233/FI-2012-705.

[11] L. Lambers, F. Orejas, Initial conflicts for transformation rules with nested application conditions, in: F. Gadducci, T. Kehrer (Eds.), Graph Transformation - 13th International Conference, ICGT 2020, Held as Part of STAF 2020, Vol. 12150 of Lecture Notes in Computer Science, Springer, 2020, pp. 109–127. doi:10.1007/978-3-030-51372-6_7.

[12] D. Knuth, P. Bendix, Simple word problems in universal algebras, in: J. Leech (Ed.), Computational Problems in Abstract Algebra, Pergamon Press, 1970, pp. 263–297.

[13] E. Ohlebusch, Advanced Topics in Term Rewriting, Springer, 2002.

[14] D. Plump, Hypergraph rewriting: Critical pairs and undecidability of confluence, in: R. Sleep, R. Plasmeijer, M. van Eekelen (Eds.), Term Graph Rewriting: Theory and Practice, John Wiley, 1993, pp. 201–213.

[15] D. Plump, Critical pairs in term graph rewriting, in: Proceedings of the 19th International Symposium on Mathematical Foundations of Computer Science (MFCS '94), Kosice, Slovakia, 1994, pp. 556–566.

[16] D. Plump, Confluence of graph transformation revisited, in: Processes, Terms and Cycles: Steps on the Road to Infinity, Essays Dedicated to Jan Willem Klop, on the Occasion of His 60th Birthday, 2005, pp. 280–308.

[17] F. Bonchi, F. Gadducci, A. Kissinger, P. Sobociński, F. Zanasi, Confluence of graph rewriting with interfaces, in: H. Yang (Ed.), Programming Languages and Systems - 26th European Symposium on Programming, ESOP 2017, Held as Part of ETAPS 2017, Vol. 10201 of LNCS, Springer, 2017, pp. 141–169. `doi:10.1007/978-3-662-54434-1\_6`.
URL `https://doi.org/10.1007/978-3-662-54434-1_6`

[18] K. Born, L. Lambers, D. Strüber, G. Taentzer, Granularity of conflicts and dependencies in graph transformation systems, in: J. de Lara, D. Plump (Eds.), Graph Transformation - 10th International Conference, ICGT 2017, Held as Part of STAF 2017, Vol. 10373 of LNCS, Springer, 2017, pp. 125–141. `doi:10.1007/978-3-319-61470-0\_8`.

[19] L. Lambers, K. Born, J. Kosiol, D. Strüber, G. Taentzer, Granularity of conflicts and dependencies in graph transformation systems: A two-dimensional approach, J. Log. Algebraic Methods Program. 103 (2019) 105–129. `doi:10.1016/j.jlamp.2018.11.004`.

[20] H. Ehrig, A. Habel, Graph grammars with application conditions, in: G. Rozenberg, A. Salomaa (Eds.), The Book of L, Springer, Berlin, 1986, pp. 87–100.

[21] R. Heckel, A. Wagner, Ensuring consistency of conditional graph rewriting - a constructive approach, Electr. Notes Theor. Comput. Sci. 2 (1995) 118–126.

[22] A. Habel, R. Heckel, G. Taentzer, Graph grammars with negative application conditions, Fundam. Inform. 26 (3/4) (1996) 287–313. `doi:10.3233/FI-1996-263404`.

[23] A. Rensink, Representing first-order logic using graphs, in: Graph Transformations, Second International Conference, ICGT 2004, Vol. 3256 of LNCS, Springer, 2004, pp. 319–335.

[24] A. Habel, K. Pennemann, Correctness of high-level transformation systems relative to nested conditions, Mathematical Structures in Computer Science 19 (2) (2009) 245–296. doi:10.1017/S0960129508007202.

[25] L. Lambers, F. Orejas, Tableau-based reasoning for graph properties, in: Graph Transformation - 7th International Conference, ICGT 2014, Vol. 8571 of LNCS, Springer, 2014, pp. 17–32.

[26] H. J. S. Bruggink, R. Cauderlier, M. Hülsbusch, B. König, Conditional reactive systems, in: S. Chakraborty, A. Kumar (Eds.), IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2011, December 12-14, 2011, Mumbai, India, Vol. 13 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2011, pp. 191–203. doi:10.4230/LIPIcs.FSTTCS.2011.191.
URL https://doi.org/10.4230/LIPIcs.FSTTCS.2011.191

[27] H. Ehrig, K. Ehrig, U. Prange, G. Taentzer, Fundamentals of Algebraic Graph Transformation, Monographs in Theoretical Computer Science. An EATCS Series, Springer, 2006. doi:10.1007/3-540-31188-2.

[28] H. Ehrig, U. Golas, F. Hermann, Categorical frameworks for graph transformation and HLR systems based on the DPO approach, Bulletin of the EATCS 102 (2010) 111–121.
URL http://eatcs.org/beatcs/index.php/beatcs/article/view/158

[29] H. Ehrig, K. Ehrig, A. Habel, K. Pennemann, Theory of constraints and application conditions: From graphs to high-level structures, Fundam. Inform. 74 (1) (2006) 135–166.

[30] H. Ehrig, U. Golas, A. Habel, L. Lambers, F. Orejas, $\mathcal{M}$-adhesive transformation systems with nested application conditions. part 1: parallelism, concurrency and amalgamation, Mathematical Structures in Computer Science 24 (4). doi:10.1017/S0960129512000357.

[31] S. Schneider, L. Lambers, F. Orejas, Symbolic model generation for graph properties, in: M. Huisman, J. Rubin (Eds.), Fundamental Approaches to Software Engineering - 20th International Conference, FASE 2017, Held as Part of ETAPS 2017, Vol. 10202 of LNCS, Springer, 2017, pp. 226–243. doi:10.1007/978-3-662-54494-5_13.

[32] K. Pennemann, Development of correct graph transformation systems, Ph.D. thesis, University of Oldenburg, Germany (2009).
URL http://oops.uni-oldenburg.de/volltexte/2009/948/

[33] I. Hristakiev, D. Plump, Attributed graph transformation via rule schemata: Church-rosser theorem, in: P. Milazzo, D. Varró, M. Wimmer (Eds.), Software Technologies: Applications and Foundations - STAF 2016 Collocated Workshops: DataMod, GCM, HOFM,

MELO, SEMS, VeryComp, Vienna, Austria, July 4-8, 2016, Revised Selected Papers, Vol. 9946 of Lecture Notes in Computer Science, Springer, 2016, pp. 145–160. `doi:10.1007/978-3-319-50230-4\_11`.

[34] G. Kulcsár, F. Deckwerth, M. Lochau, G. Varró, A. Schürr, Improved conflict detection for graph transformation with attributes, in: A. Rensink, E. Zambon (Eds.), Proceedings Graphs as Models, GaM@ETAPS 2015, London, UK, 11-12 April 2015, Vol. 181 of EPTCS, 2015, pp. 97–112. `doi:10.4204/EPTCS.181.7`.

[35] M. Löwe, Algebraic approach to single-pushout graph transformation, Theor. Comput. Sci. 109 (1&2) (1993) 181–224. `doi:10.1016/0304-3975(93)90068-5`.

[36] U. Golas, L. Lambers, H. Ehrig, F. Orejas, Attributed graph transformation with inheritance: Efficient conflict detection and local confluence analysis using abstract critical pairs, Theor. Comput. Sci. 424 (2012) 46–68. `doi:10.1016/j.tcs.2012.01.032`.

[37] G. Taentzer, U. Golas, Towards local confluence analysis for amalgamated graph transformation, in: F. Parisi-Presicce, B. Westfechtel (Eds.), Graph Transformation - 8th International Conference, ICGT 2015, Held as Part of STAF 2015, L'Aquila, Italy, July 21-23, 2015. Proceedings, Vol. 9151 of Lecture Notes in Computer Science, Springer, 2015, pp. 69–86. `doi:10.1007/978-3-319-21145-9\_5`.

[38] K. Born, G. Taentzer, An algorithm for the critical pair analysis of amalgamated graph transformations, in: R. Echahed, M. Minas (Eds.), Graph Transformation - 9th International Conference, ICGT 2016, in Memory of Hartmut Ehrig, Held as Part of STAF 2016, Vienna, Austria, July 5-6, 2016, Proceedings, Vol. 9761 of Lecture Notes in Computer Science, Springer, 2016, pp. 118–134. `doi:10.1007/978-3-319-40530-8\_8`.