Fundamenta Informaticae 72 (2006) 1–29
IOS Press

Adhesive High-Level Replacement Systems: A New Categorical Framework for Graph Transformation

Hartmut Ehrig, Julia Padberg and Ulrike Prange*
Technical University of Berlin, Germany
ehrig@cs.tu-berlin.de;padberg@cs.tu-berlin.de;uprange@cs.tu-berlin.de

Annegret Habel
Carl v. Ossietzky University of Oldenburg
annegret.habel@informatik.uni-oldenburg.de

Abstract. Adhesive high-level replacement (HLR) systems are introduced as a new categorical framework for graph transformation in the double pushout (DPO) approach, which combines the well-known concept of HLR systems with the new concept of adhesive categories introduced by Lack and Sobociński.

In this paper we show that most of the HLR properties, which had been introduced to generalize some basic results from the category of graphs to high-level structures, are valid already in adhesive HLR categories. This leads to a smooth categorical theory of HLR systems which can be applied to a large variety of graphs and other visual models. As a main new result in a categorical framework we show the Critical Pair Lemma for the local confluence of transformations. Moreover we present a new version of embeddings and extensions for transformations in our framework of adhesive HLR systems.

Keywords: high-level replacement systems, adhesive categories, adhesive HLR categories, graph transformation, local confluence of transformations, Critical Pair Lemma

1. Introduction

Traditionally the theory of graph transformations is based on the notion of labeled graphs. In order to meet the requirements of different application areas the theory has been extended to several varieties of

*Address for correspondence: Technical University of Berlin, Sekr. FR6-1, Franklinstr. 28/29, 10587 Berlin, Germany

graphs including hypergraphs, term graphs and bipartite graphs in the sense of Petri nets. A first attempt to unify the different approaches is the concept of high-level replacement (HLR) systems. In fact, HLR systems have been introduced in [10] to generalize the double pushout approach from graphs [5] to various kinds of high-level structures, including also algebraic specifications and Petri nets. In order to generalize basic results, like the Local Church-Rosser, the Parallelism and the Concurrency Theorem, several different conditions have been introduced in [10], called HLR conditions. The theory of HLR systems has been applied to a large number of example categories, where all the HLR conditions have been verified explicitly. These conditions are a collection of all the properties which are used in the categorical proofs of the basic results. Up to now it has not been analyzed how far these HLR properties are independent from each other or are consequences of a more general principle.

This problem has been solved recently by Lack and Sobociński in [18] by introducing the notion of adhesive categories. They have shown that the concept of "van Kampen squares", short VK squares, motivated by topology [2], can be considered as such a general principle. Roughly spoken a VK square is a pushout square which is stable under pullbacks, and that vice versa pullback squares are stable under combined pushouts and pullbacks. The key idea of adhesive categories is the requirement that pushouts along monomorphisms are VK squares. This property is valid not only in the categories **Sets** and **Graphs**, but also in several varieties of graphs, which have been used in the theory of graph grammars and graph transformation [3] up to now. On the other hand Lack and Sobociński were able to show in [18] that most of the HLR conditions required in [10] can be shown for adhesive categories. Together with the results in [10] this implies that the basic results for the theory of graph transformation mentioned above are valid in adhesive categories, where only the Parallelism Theorem requires in addition the existence of binary coproducts.

Unfortunately the concept of adhesive categories incorporates an important restriction, which rules out several interesting application categories. The HLR framework in [10] is based on a distinguished class M of morphisms, which is restricted to the class of all monomorphisms in adhesive categories. This restriction rules out the category (**SPEC**, M) of all algebraic specifications with class M of all strict injective specification morphisms (see [10]) and several other integrated specification techniques like algebraic high-level nets [24, 9] and different kinds of attributed graphs [22, 17], which are important in the area of graph transformation and HLR systems.

In this paper we combine the advantages of HLR and of adhesive categories by introducing the new concept of "adhesive HLR categories". Roughly spoken an adhesive HLR category is an adhesive category with a suitable subclass M of monomorphisms, which is closed under pushouts and pullbacks. As main results we show that adhesive HLR categories are closed under product, slice, coslice and functor category constructions and that most of the important HLR properties of [10] are valid. These results are generalizations of corresponding results in [18], where we remove the restrictions, that M is the class of all monomorphisms and that adhesive categories in [18] are required to have all pullbacks instead of pullbacks along M-morphisms only.

In sections 2 - 4 of this paper we review and recover the basic results for HLR systems in [10] and adhesive grammars in [18] in the framework of adhesive HLR categories and systems. Moreover, we present in section 5 a new version of the results for the embedding and extension of transformations [5, 25, 14]. This is the basis to show in section 6 another main result of this paper: We present a categorical version of the Critical Pair Lemma for the local confluence of transformations, discussed for hypergraphs in [26] and for attributed graphs in [17], in our new framework of adhesive HLR systems. Constructions and results in section 4 - 6 are motivated by examples based on graphs in the sense of the traditional approach of graph transformation in [5]. An application of the categorical theory to typed attributed graphs is presented in [27, 16, 8].

A short version of this article has been presented at ICGT'04 in Rome, September 2004 [11]. This extended version includes not only proof ideas, but full proofs for most of the main results and motivating examples for the new results. An even more detailed version is presented in the first part of the diploma thesis [27].

Acknowledgement

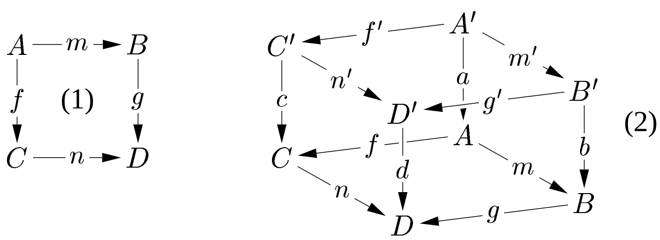
We are grateful to Paolo Baldan, Pawel Sobociński and several members of the TMR network SEGRAVIS and the IFIP WG 1.3 for interesting discussions concerning the new concept of adhesive and adhesive HLR categories and systems. Moreover we would like to thank the referees for their comments leading to several improvements.

2. Van Kampen Squares and Adhesive Categories

In this section we review adhesive categories as introduced by Lack and Sobociński in [18], motivated by the work of Sassone and Sobociński on deriving bisimulation congruences in [28].

The basic notion of adhesive categories is that of a so called van Kampen square. The intuitive idea of a van Kampen square is that of a pushout which is stable under pullbacks and that vice versa pushout preservation implies pullback stability. The name van Kampen derives from the relationship between these squares and the Van Kampen Theorem in topology [2].

Definition 2.1. (van Kampen square)
A pushout (1) is a van Kampen (VK) square, if for any commutative cube (2) with (1) in the bottom and the back faces being pullbacks holds: the top is a pushout ⇔ the front faces are pullbacks.



In the definition of adhesive categories only those VK squares are considered, where m is a monomorphism. In this case the square is called a pushout along a monomorphism. The first interesting property of VK squares in [18] shows that in this case also n is a monomorphism and the square is also a pullback.

Definition 2.2. (adhesive category)
A category $\boldsymbol{C}$ is an adhesive category, if

1. $\boldsymbol{C}$ has pushouts along monomorphisms, i.e. pushouts, where at least one of the given morphisms is a monomorphism,

2. $\boldsymbol{C}$ has pullbacks,

3. pushouts along monomorphisms are VK squares.

The most basic example of an adhesive category is the category **Sets** of sets. Moreover it is shown in [18] that adhesive categories are closed under product, slice, coslice and functor category construction. This implies immediately that also the category **Graphs** of graphs $G = (E \overset{s,t}{\rightrightarrows} V)$, and also several variants like typed graphs, labelled graphs and hypergraphs are adhesive categories. This is a first indication that adhesive categories are suitable for graph transformation. Counterexamples for adhesive categories are **Pos** (partially ordered sets), **Top** (topological spaces), **Gpd** (groupoids) and **Cat** (categories), where pushouts along monomorphisms fail to be VK squares (see [18]).

The main reason why adhesive categories are important for the theory of graph transformation and its generalization to high-level replacement systems (see [10]) is the fact that most of the HLR conditions required in [10] are shown to be valid already in adhesive categories (see [18]). This implies that basic results like the Local Church-Rosser Theorem and the Concurrency Theorem (see [10]) are valid already in the framework of adhesive categories, while the Parallelism Theorem needs in addition the existence of binary coproducts.

The main advantage of adhesive categories compared with HLR categories in [10] is the fact that the requirements for adhesive categories are much more smooth than the variety of different HLR conditions in [10], which have been stated as needed in the categorical proofs of the corresponding results mentioned above.

On the other hand HLR categories in [10] are based on a class M of morphisms, which is restricted to the class of all monomorphisms in adhesive categories. This rules out several interesting examples. In order to avoid this problem we combine the two concepts leading to the notion of adhesive HLR categories. This includes the concept of quasiadhesive categories as introduced in [19, 29] for the special class M of regular monomorphisms. For this reason we study immediately adhesive HLR categories in the next section.

3. Adhesive HLR Categories

As motivated in the previous section we will combine the concepts of adhesive categories [18] and HLR categories [10] leading to the new concept of adhesive HLR categories. Most of the results presented in this section are generalizations of results for adhesive categories in [18], but we present new interesting examples which are not instantiations of adhesive categories.

The main difference of adhesive HLR categories compared with adhesive categories is the fact that we consider a distinguished subclass M of monomorphisms instead of the class of all monomorphisms. Moreover we require only the existence of pullbacks along M-morphisms and not for general morphisms.

Definition 3.1. (adhesive HLR category)
A category $\boldsymbol{C}$ with a distinguished class M of morphisms is called adhesive HLR category, if

1. M is a class of monomorphisms closed under isomorphisms and closed under composition ($f : A \to B \in M$, $g : B \to C \in M \Rightarrow g \circ f \in M$) and decomposition ($g \circ f \in M$, $g \in M \Rightarrow f \in M$),

2. $\boldsymbol{C}$ has pushouts and pullbacks along M-morphisms and M-morphisms are closed under pushouts and pullbacks, i.e. given the square (1) in Definition 2.1, then (1) being a pushout and $m \in M$ implies $n \in M$, and (1) being a pullback and $n \in M$ implies $m \in M$,

3. pushouts in $\boldsymbol{C}$ along M-morphisms are VK squares.

Remark 3.1. As pointed out by one of the referees the decomposition property of M is a consequence of M being closed under pullbacks. In fact the following diagram (1) with $g \circ f \in M$ and $g \in M$ is a pullback, because g is a monomorphism. Hence M closed under pullbacks implies $f \in M$.

$$\begin{array}{ccc} A & \xrightarrow{f} & B \\ {\scriptstyle id_A}\downarrow & (1) & \downarrow{\scriptstyle g} \\ A & \xrightarrow{g \circ f} & D \end{array}$$

For similar reasons the closure of M under isomorphisms is a consequence of the closure under pullbacks.

Example 3.1. 1. All examples of adhesive categories are adhesive HLR categories for the class M of all monomorphisms. As shown in [18] this includes the category (**Sets**, M) of sets, (**Graphs**, M) of graphs and several variants of graphs like typed, labelled and hypergraphs discussed above. An explicit verification of the properties of adhesive (HLR) categories for **Sets** is given in [27].

2. Quasiadhesive categories as introduced in [19] are adhesive HLR categories with the class M of all regular monomorphisms (where a monomorphism is called regular if it is the equalizer of two morphisms). As shown in [4], term graphs are a quasiadhesive category, so they are also an adhesive HLR category.

3. The category (**Spec**, M_1) of algebraic specifications with the class M_1 of all monomorphisms is not adhesive, because pushouts along monomorphisms are not necessarily pullbacks. But (**Spec**, M_2) with the class M_2 of all strict injective specification morphisms is an HLR2 category in the sense of [10], a quasiadhesive category (see [19]) and hence also an adhesive HLR category.

4. An important new example is the category ($\mathbf{AGraphs_{ATG}}$, M) of typed attributed graphs with a type graph ATG and the class M of all injective morphisms with isomorphisms on the data part. In [16] we explicitly show that this is an adhesive HLR category.

Remark 3.2. Most of the results in this paper can also be formulated under slightly weaker assumptions, where the existence of pullbacks is required only if both given morphisms are in M and pushouts along M-morphisms are required to be M-VK squares only, i.e. the VK square property is only required for the case $f \in M$ or $b, c, d \in M$. This weaker version is called "weak adhesive HLR category". The category **PT-Nets** of place transition nets considered in [10] is a weak adhesive HLR category which is not an adhesive HLR category as proposed in [11] (see [15]).

The first important result shows that adhesive HLR categories are closed under product, slice, coslice and functor category construction, which has been shown already for adhesive categories in [19, 18]. This allows to construct new examples from given ones.

Theorem 3.1. (construction of adhesive HLR categories)
Adhesive HLR categories can be constructed as follows:

- If ($\boldsymbol{C}$, M_1) and ($\boldsymbol{D}$, M_2) are adhesive HLR categories, then ($\boldsymbol{C} \times \boldsymbol{D}$, $M_1 \times M_2$) is an adhesive HLR category.
- If ($\boldsymbol{C}$, M) is an adhesive HLR category, then so are the slice category ($\boldsymbol{C}\backslash C$, $M \cap \boldsymbol{C}\backslash C$) and the coslice category ($C\backslash\boldsymbol{C}$, $M \cap C\backslash\boldsymbol{C}$) for any object C in $\boldsymbol{C}$.
- If ($\boldsymbol{C}$, M) is an adhesive HLR category, then every functor category ([$\boldsymbol{X}$, $\boldsymbol{C}$], M-functor transformations) is an adhesive HLR category.

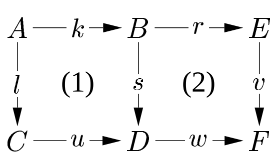
Remark 3.3. An M-functor transformation is a natural transformation $t : F \to G$ of functors where all morphisms $t(X) : F(X) \to G(X)$ are in M. For the categorical constructions see e.g. [23].

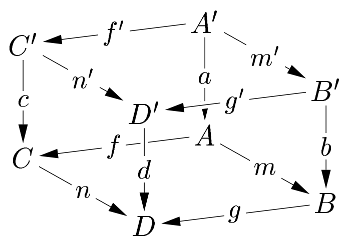
Proof Idea:
In the case of product and functor categories the properties of adhesive HLR categories can be shown componentwise. For slice and coslice categories some standard constructions for pushouts and pullbacks can be used to show the properties. □

The second important result shows that most of the HLR conditions stated in [10, 9] are already valid in adhesive HLR categories.

Theorem 3.2. (HLR properties of adhesive HLR categories)
Given an adhesive HLR category ($\boldsymbol{C}$, M), the following HLR conditions are satisfied.

1. Pushouts along M-morphisms are pullbacks.
2. Pushout-pullback decomposition: Given the following diagram with (1) + (2) being a pushout, (2) being a pullback, $w \in M$ and ($l \in M$ or $k \in M$). Then (1) and (2) are pushouts and also pullbacks.
3. Cube pushout-pullback property: Given the following commutative cube (3), where all morphisms in the top and in the bottom are in M, the top is a pullback and the front faces are pushouts. Then we have: the bottom is a pullback iff the back faces of the cube are pushouts.



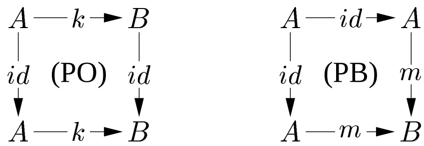


(3)

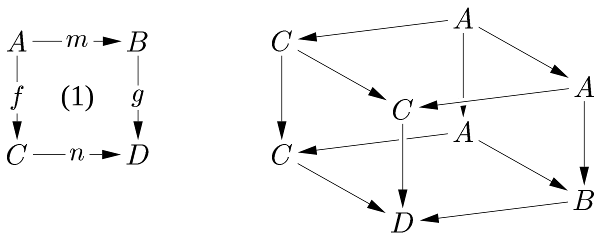
4. Uniqueness of pushout complements for M-morphisms: Given $k : A \to B \in M$ and $s : B \to D$ then there is up to isomorphism at most one C with $l : A \to C$ and $u : C \to D$ such that the diagram (1) is a pushout.

Proof:
In the proofs we use the fact, that

1. for arbitrary morphisms $k : A \to B$ the diagram (PO) below is a pushout and also a pullback,
2. the diagram (PB) below is a pullback $\Leftrightarrow m : A \to B$ is a monomorphism.



Item 1: Consider the pushout (1) with $m \in M$ and the following cube (2) over the given morphisms.



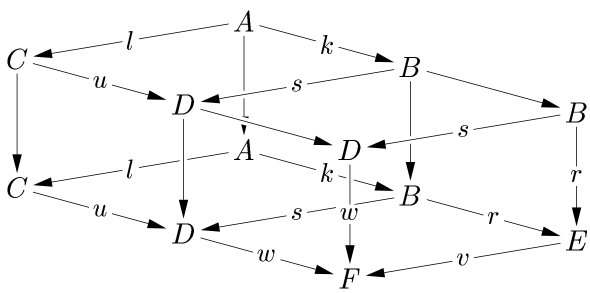
(2)

Since the bottom is the pushout (1) along the M-morphism m it is a VK square. Then we have

- the back left is a pullback,
- the back right is a pullback (since $m \in M$),
- the top is a pushout.

With the VK square property we conclude that the front faces, and therefore (1), are pullbacks.

Item 2: With $w \in M$ we have with Definition 3.1 that also $r \in M$. If $k \in M$ then also $r \circ k \in M$ since M-morphisms are closed under composition. Consider the following cube over the given morphisms.



Since the bottom is the pushout (1)+(2) along the M-morphism l or along the M-morphism $r \circ k$ it is a VK square. Then we have

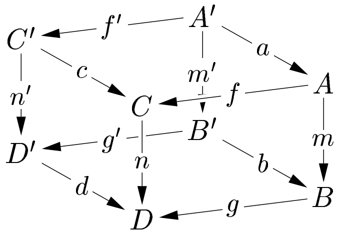
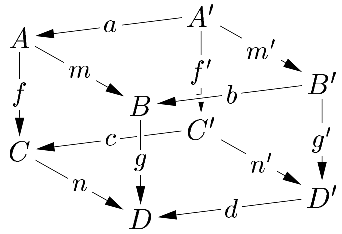
- the back left is a pullback,

- the back right as a composition of pullbacks (since $r \in M$) is a pullback,
- the front left as a composition of pullbacks (since $w \in M$) is a pullback,
- the front right is a pullback per assumption.

With the VK square property we conclude that the top, corresponding to square (1), is a pushout, and the pushout decomposition gives us that also (2) is a pushout.

Item 3: Since the front faces are pushouts along M-morphisms they are also pullbacks. Moreover we have by assumption $f, g, m, n, f', g', m', n' \in M$.

"⇒" Let the bottom be a pullback. Consider the turned cube (2)

 (2)  (3)

Then the following properties apply:

- the bottom is a pushout along the M-morphism g', therefore a VK square,
- the front right is a pullback,
- the front left is a pushout along the M-morphism n', therefore a pullback,
- the back left is a pullback,
- the back right is a pullback (by composition and decomposition of pullbacks).

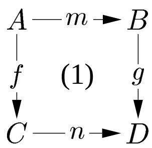
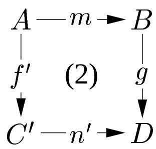
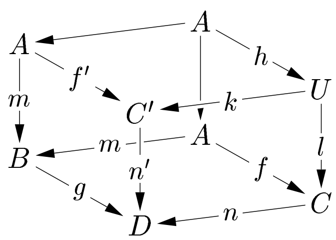
With the VK square property it follows that the top is a pushout, that means the back left in the original cube (1) is a pushout.

By turning the cube once more we get the same result for back right within the original cube (1). Hence the back faces are pushouts.

"⇐" Let the back faces be pushouts within the original cube (1). By turning the cube again we get cube (3) with the bottom, top, back left and front right being pushouts along M-morphisms and the back right being a pullback.

Since the bottom is a VK square and the top is a pushout, the front faces must be pullbacks, that means, the bottom of the original cube is a pullback.

Item 4: Suppose that (1) and (2) are pushouts with $m \in M$. It follows $n, n' \in M$.

   (3)

Consider the cube (3) where $C' \overset{k}{\leftarrow} U \overset{l}{\rightarrow} C$ is the pullback over $C' \overset{n'}{\rightarrow} D \overset{n}{\leftarrow} C$. h is the resulting morphism for this pullback in comparison with the object A and the morphisms f and f', and it holds that $k \circ h = f'$ and $l \circ h = f$.

Pullbacks are closed under composition and decomposition, and since the left faces and the front right are pullbacks and $l \circ h = f$, the back right is a pullback.

Then the following properties apply:

- the bottom is a pushout along the M-morphism m and therefore a VK square,
- the back left is a pullback (with $m \in M$),
- the front left is a pushout along the M-morpism m and therefore a pullback,
- the front right is a pullback per construction,
- the back right is a pullback.

Hence it follows from the VK-square property that the top is a pushout. Since id_A is an isomorphism and pushouts preserve isomorphisms also k is an isomorphism. For similar reasons we can conclude by exchanging the roles of C and C' that l is an isomorphism.

That means C and C' are isomorphic. □

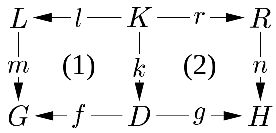
Remark 3.4. The HLR conditions stated above together with the existence of binary coproducts compatible with M correspond roughly to the HLR conditions in [9] resp. HLR2 and two of the HLR2* conditions in [10].

4. Adhesive HLR Systems

In this section we use the concept of adhesive HLR categories introduced in definition 3.1 to present the basic notions and results of adhesive HLR systems in analogy to HLR systems in [10]. The Local Church-Rosser Theorem and the Parallelism Theorem are shown to be valid in [10] for HLR1 categories, and the Concurrency Theorem for HLR2 categories, where the existence of binary coproducts is only needed for the Parallelism Theorem. Using the properties of adhesive HLR categories in section 3 we can immediatly conclude that the Local Church-Rosser Theorem and the Concurrency Theorem are valid in adhesive HLR categories and the Parallelism Theorem in adhesive HLR categories with binary coproducts.

Definition 4.1. (adhesive HLR system)
An adhesive HLR system $AS = (\boldsymbol{C}, M, P)$ consists of an adhesive HLR category ($\boldsymbol{C}$, M) and a set of productions P, where

1. a production $p = L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R$ consists of objects L, K and R called left-hand side, gluing object and right-hand side respectively, and morphisms $l : K \to L, r : K \to R$ with $l, r \in M$,
2. a direct transformation $G \overset{p,m}{\Longrightarrow} H$ via a production p and a morphism $m : L \to G$, called match, is given by the following diagram, called DPO-diagram, where (1) and (2) are pushouts, and n is called comatch,



3. a transformation is a sequence $G_0 \Rightarrow G_1 \Rightarrow ... \Rightarrow G_n$ of direct transformations, written $G_0 \overset{*}{\Rightarrow} G_n$ for $n \geq 1$. In the case $n = 0$ we have an identical transformation or $G_0 \overset{\sim}{=} G'_0$.

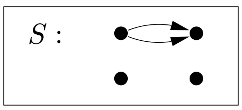
4. If we have in addition a start object S we have an adhesive HLR grammar $AG = (AS, S)$ and the language $L(AG)$ consists of all objects G in $\boldsymbol{C}$ derivable from the start object S by a transformation, i.e. $L(AG) = \{G \mid S \overset{*}{\Rightarrow} G\}$.

Remark 4.1. 1. An adhesive HLR system is on the one hand an HLR system in the sense of [10], where in [10] we have in addition a distinguished class T of terminal objects, and on the other hand an adhesive grammar in the sense of [18], provided that the class M is the class of all monomorphisms.

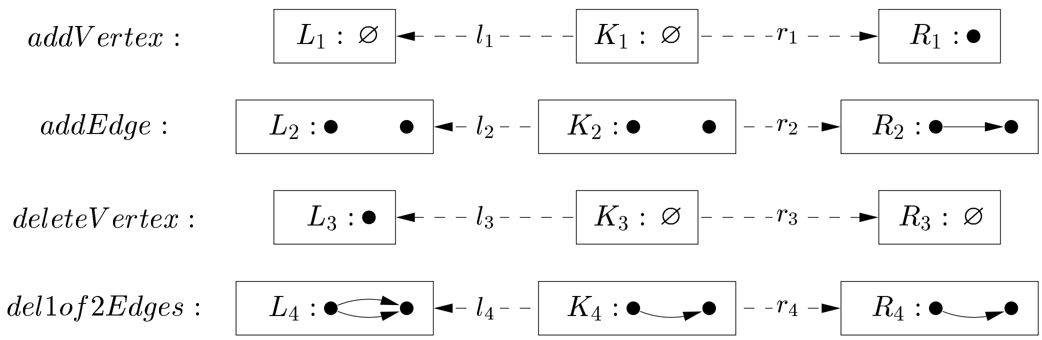
2. A direct transformation $G \overset{p,m}{\Longrightarrow} H$ is uniquely determined up to isomorphism by the production p and the match m, because due to Theorem 3.2 item 4 pushout complements along M-morphisms in adhesive HLR categories are unique up to isomorphism. For this reason we allow $G_0 \overset{\sim}{=} G'_0$ for $G_0 \overset{*}{\Rightarrow} G'_0$ in the case $n = 0$.

3. All the examples for HLR1 and HLR2 systems considered in [10] and all systems over adhesive HLR categories considered in Ex. 3.1 are adhesive HLR systems, which includes especially the classical graph transformation approach in [5].

Example 4.1. (adhesive HLR grammar $\boldsymbol{ExAHS}$, transformations)
In the following we introduce as an example the adhesive HLR grammar $ExAHS$ based on graphs and the class M of injective graph morphisms (see example 3.1). We define $ExAHS$ = (**Graphs**, M, P, S). S is the following start graph

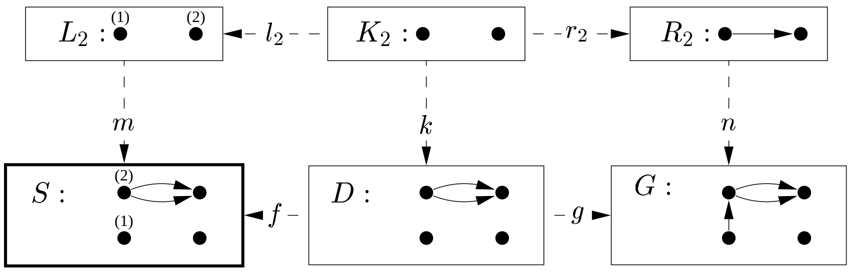


and P = $\{addVertex$, $addEdge$, $deleteVertex$, $del1of2Edges\}$ is defined by the following productions, where all morphisms are inclusions (and therefore are in M).

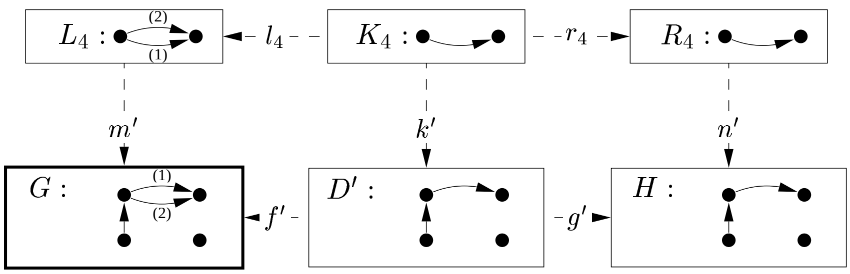


The production $addVertex$ adds a single node to a graph, and with $addEdge$ a directed edge between two vertices can be inserted. On the other hand $deleteVertex$ deletes a single vertex, and $del1of2Edges$ deletes one of two parallel edges.

There are different options to transform our start graph S by applying one of the given productions. For example, we can use the production $addEdge$ to insert an additional edge as shown in the following diagram. The node labels indicate the mapping m between L_2 and S.



In the resulting graph G we delete one of the two edges between the upper nodes by applying the production $del1of2Edges$.



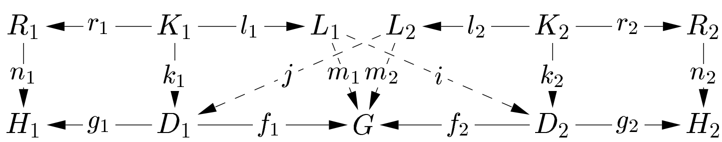
Together the direct transformations $S \overset{addEdge,m}{\Longrightarrow} G$ and $G \overset{del1of2Edges,m'}{\Longrightarrow} H$ form the transformation $S \overset{*}{\Rightarrow} H$.

Basic Results

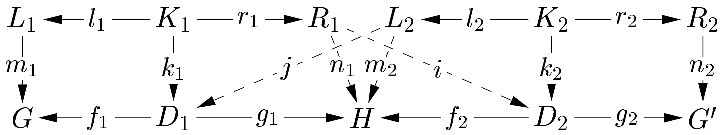
The following basic results are shown for HLR2 categories in [10] and they are rephrased for adhesive categories in [18]. By exploiting Theorem 3.2 we show that they are also valid for adhesive HLR systems.

The Local Church-Rosser Theorem is based on the notions of parallel and sequential independent direct transformations in the sense of [10]. Intuitively spoken, parallel independence means that the matches of two direct transformation only overlap in gluing items. Sequential independence means, that no necessary part for the second transformation is created by the first one, and also no part created by the first transformation is deleted by the second one.

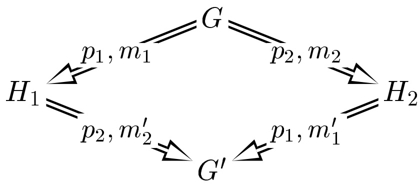
Definition 4.2. (parallel independence)
Two direct transformations $G \overset{p_1,m_1}{\Longrightarrow} H_1$ and $G \overset{p_2,m_2}{\Longrightarrow} H_2$ are parallel independent if there exist morphisms $i : L_1 \to D_2$ and $j : L_2 \to D_1$ such that $f_2 \circ i = m_1$ and $f_1 \circ j = m_2$.



Definition 4.3. (sequential independence)
Two direct transformations $G \overset{p_1,m_1}{\Longrightarrow} H \overset{p_2,m_2}{\Longrightarrow} G'$ are sequentially independent if there exist morphisms $i : R_1 \to D_2$ and $j : L_2 \to D_1$ such that $f_2 \circ i = n_1$ and $g_1 \circ j = m_2$.



Theorem 4.1. (Local Church-Rosser Theorem)
Let $AS = (\boldsymbol{C}, M, S, P)$ be an adhesive HLR system. Given two parallel independent direct transformations $G \overset{p_1,m_1}{\Longrightarrow} H_1$ and $G \overset{p_2,m_2}{\Longrightarrow} H_2$ then there is an object G' and transformations $H_1 \overset{p_2,m_2'}{\Longrightarrow} G'$ and $H_2 \overset{p_1,m_1'}{\Longrightarrow} G'$ such that both $G \overset{p_1,m_1}{\Longrightarrow} H_1 \overset{p_2,m_2'}{\Longrightarrow} G'$ and $G \overset{p_2,m_2}{\Longrightarrow} H_2 \overset{p_1,m_1'}{\Longrightarrow} G'$ are sequentially independent.



Vice versa given two sequentially independent direct transformations $G \overset{p_1,m_1}{\Longrightarrow} H_1 \overset{p_2,m_2'}{\Longrightarrow} G'$ then there is an object H_2 and transformations $G \overset{p_2,m_2}{\Longrightarrow} H_2 \overset{p_1,m_1'}{\Longrightarrow} G'$ such that $G \overset{p_1,m_1}{\Longrightarrow} H_1$ and $G \overset{p_2,m_2}{\Longrightarrow} H_2$ are parallel independent.

Proof:
According to Theorem 3.2 ($\boldsymbol{C}$, M) satisfies all the HLR1 properties (except binary coproducts), which are used in [10] to prove the Local Church-Rosser Theorems I + II. □

In order to show the Parallelism Theorem in [10] the adhesive HLR category is required to satisfy all HLR1 properties including binary coproducts in order to build parallel productions $p_1 + p_2 = (L_1 + L_2 \overset{l_1+l_2}{\longleftarrow} K_1 + K_2 \overset{r_1+r_2}{\longrightarrow} R_1 + R_2)$ from given productions $p_i = (L_i \overset{l_i}{\leftarrow} K_i \overset{r_i}{\rightarrow} R_i)$ ($i = 1, 2$).

Theorem 4.2. (Parallelism Theorem)
Let $AS = (\boldsymbol{C}, M, S, P)$ be an adhesive HLR system, where ($\boldsymbol{C}$, M) has binary coproducts.

1. Synthesis: Given a sequentially independent transformation $G \Rightarrow H \Rightarrow G'$ via productions (p_1, p_2), then there is a "synthesis construction" leading to a parallel transformation $G \Rightarrow G'$ via $p_1 + p_2$.

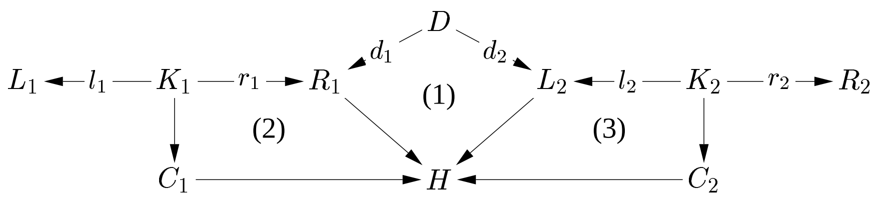
2. Analysis: Given a parallel transformation $G \Rightarrow G'$ via $p_1 + p_2$ then there is an "analysis construction" leading to two sequentially independent transformations $G \Rightarrow H_1 \Rightarrow G'$ via (p_1, p_2) and $G \Rightarrow H_2 \Rightarrow G'$ via (p_2, p_1).

3. Bijective correspondence: The constructions "synthesis" and "analysis" are inverse to each other up to isomorphism.

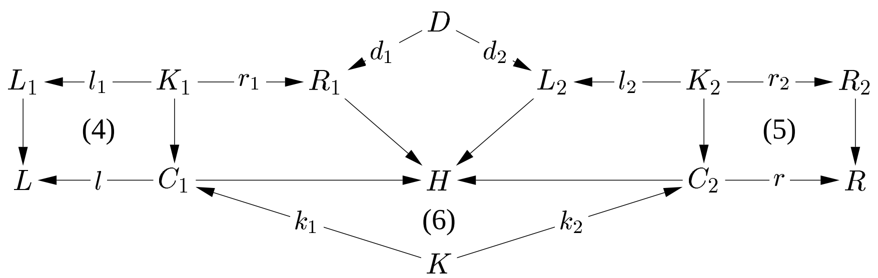
Proof:
$\boldsymbol{C}$ has binary coprocucts, and with Theorem 3.2 ($\boldsymbol{C}$, M) satisfies all the HLR1 properties which are used in [10] to prove the Parallelism Theorem. □

Finally let us rephrase the Concurrency Theorem for adhesive HLR systems, which is based on the notions of a dependency relation D for productions p_1, p_2, the corresponding D-concurrent production $p_1 *_D p_2$ and D-related transformations in the sense of [10].

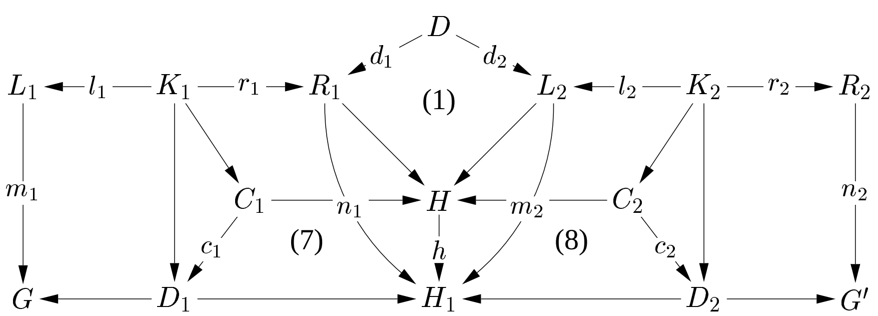
Definition 4.4. (dependency relation)
Given two productions p_1 and p_2 with $p_i = (L_i \stackrel{l_i}{\leftarrow} K_i \stackrel{r_i}{\rightarrow} R_i)$ $(i = 1, 2)$. An object D with morphisms $d_1 : D \to R_1$ and $d_2 : D \to L_2$ is a dependency relation for p_1 and p_2 if the pushout (1) with the pushout object H and the pushout complements (2) and (3) over $K_1 \stackrel{r_1}{\rightarrow} R_1 \to H$ and $K_2 \stackrel{l_2}{\rightarrow} L_2 \to H$ exist.



Definition 4.5. (D-concurrent production and D-related transformation)
Given a dependency relation $R_1 \stackrel{d_1}{\leftarrow} D \stackrel{d_2}{\rightarrow} L_2$ for the productions p_1 and p_2, the D-concurrent production $p_1 *_D p_2$ is defined by $p_1 *_D p_2 = (L \stackrel{l \circ k_1}{\longleftarrow} K \stackrel{r \circ k_2}{\longrightarrow} R)$ as shown in the following diagram, where (4) and (5) are pushouts and (6) is a pullback.



A transformation sequence $G \stackrel{p_1,m_1}{\Longrightarrow} H_1 \stackrel{p_2,m_2}{\Longrightarrow} G'$ is called D-related if $n_1 \circ d_1 = m_2 \circ d_2$ and there are morphisms $c_1 : C_1 \to D_1$ and $c_2 : C_2 \to D_2$ such that (7) and (8) are pushouts, whereas h is induced by pushout (1) and morphisms n_1 and m_2.



Theorem 4.3. (Concurrency Theorem)
Let $AS = (\boldsymbol{C}, M, S, P)$ be an adhesive HLR system, D a dependency relation for (p_1, p_2) and $p_1 *_D p_2$ the corresponding D-concurrent production.

1. Synthesis: Given a D-related transformation $G \Rightarrow H \Rightarrow G'$ via (p_1, p_2), then there is a "synthesis construction" leading to a direct transformation $G \Rightarrow G'$ via $p_1 *_D p_2$.

2. Analysis: Given a direct transformation $G \Rightarrow G'$ via $p_1 *_D p_2$ then there is an "analysis construction" leading to a D-related transformation $G \Rightarrow H \Rightarrow G'$ via (p_1, p_2).

3. Bijective correspondence: The constructions "synthesis" and "analysis" are inverse to each other up to isomorphism.

Proof:
According to Theorem 3.2 ($\boldsymbol{C}$, M) satisfies all the HLR2 properties (except binary coproducts), which are used in [10] to prove the Concurrency Theorem. □

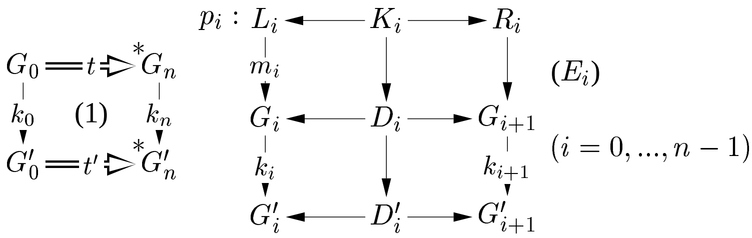
Remark 4.2. The construction of a canonical dependency relation D for (p_1, p_2) from a transformation sequence $G \Rightarrow H \Rightarrow G'$ via (p_1, p_2) is based on HLR2* conditions in [10], where especially the twisted-triple-pushout condition needs a slightly stronger version of adhesive HLR categories. Due to an investigation of Pawel Sobociński we need in addition that arbitrary pushouts are stable under pullbacks. This corresponds to one direction of the VK squares in Definition 2.1, which is still true for **Sets**, **Graphs** and other categories. But note that the opposite direction of Definition 2.1 for arbitrary pushouts is not valid in **Sets**, **Graphs** etc. (see [18]). Moreover there is an alternative way for the construction using the pair factorization property (see Definition 6.2).

5. Embedding and Extension of Adhesive HLR Transformations

In this section we present a categorical version of the Embedding Theorem for graph transformation (see [5]) using the concept of initial pushouts first introduced in [25]. The embedding theorem is not only important for the theory of graph transformation, but also for the component framework for system modelling introduced in [13]. In [14] it is shown how to verify the extension properties used in the generic component concept of [13] in the framework of HLR systems. The Embedding Theorem and the Extension Theorem presented for adhesive HLR systems in this section combine the results for both areas and will also be used in the next section to show the Local Confluence Theorem. The key notion is the concept of initial pushouts, which formalizes the construction of boundary and context in [5]. The important new property going beyond [25] is the fact that initial pushouts are closed under double pushouts under certain conditions. As in section 4 also in this section we assume that we have an adhesive HLR system.

We start with the definition of an extension diagram in the sense of [13, 14] which means that a transformation t is extended to a transformation t' via an extension morphism.

Definition 5.1. (extension diagram)
An extension diagram is a diagram (1), where $k_0 : G_0 \to G'_0$ is a morphism, called extension morphism, and $t : G_0 \overset{*}{\Rightarrow} G_n$ and $t' : G'_0 \overset{*}{\Rightarrow} G'_n$ are transformations via the same productions $(p_0, ..., p_{n-1})$ and matches $(m_0, ..., m_{n-1})$ resp. $(k_0 \circ m_0, ..., k_{n-1} \circ m_{n-1})$ defined by the following DPOs (E_i).

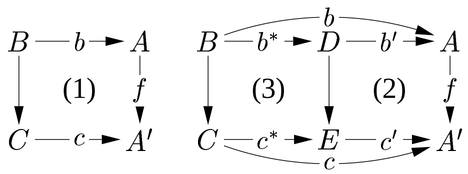


Remark 5.1. 1. The extension diagram (1) is completely determined (up to isomorphism) by $t : G_0 \overset{*}{\Rightarrow} G_n$ and $k_0 : G_0 \to G'_0$ (using the uniqueness of pushouts and pushout complements).

2. Extension diagrams are closed under horizontal and vertical composition (using corresponding composition properties of pushouts).

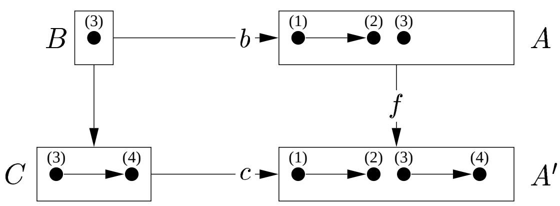
The main problem is now to determine under which condition a transformation $t : G_0 \overset{*}{\Rightarrow} G_n$ and an extension morphism $k_0 : G_0 \to G'_0$ lead to an extension diagram. The key notion is that of an initial pushout, which will be required for the extension morphism k_0 in the consistency condition below and formalizes the construction of boundary and context in [5].

Definition 5.2. (initial pushout, boundary and context)
Given $f : A \to A'$, a morphism $b : B \to A$ with $b \in M$ is called the boundary over f if there is a pushout complement such that (1) is an initial pushout over f. Initiality of (1) over f means, that for every pushout (2) with $b' \in M$ there exist unique morphisms $b^* : B \to D$ and $c^* : C \to E$ with b^*, $c^* \in M$ such that $b' \circ b^* = b$, $c' \circ c^* = c$ and (3) is pushout. Then B is called the boundary object and C the context w.r.t. $f : A \to A'$.



Remark 5.2. In the classical case of graph transformations [5] the boundary B of an injective graph morphism $f : A \to A'$ consists of all nodes in $x \in A$ such that $f(x)$ is adjacent to an edge in $A' \backslash f(A)$. These nodes are necessary to glue A to the context graph $C = A' \backslash f(A) \cup f(b(B))$ in order to obtain A' as gluing of A and C via B in the initial pushout (1).

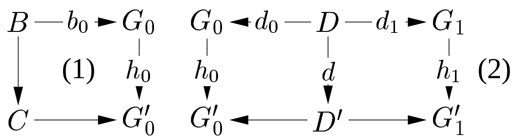
Example 5.1. (initial pushouts in Graphs)
Consider the following morphism $f : A \to A'$ induced by the node labels. Node (3) is the only one adjacent to an edge in $A' \backslash f(A)$ and therefore has to be in the boundary object B. The context object C contains nodes (3) and (4) and the edge between them. All morphisms are inclusions.

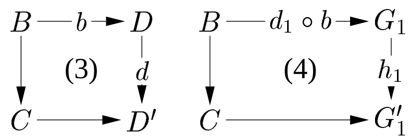


In **Graphs**, the initial pushouts over arbitrary morphisms exist. If the given graph morphism $f : A \to A'$ is not injective, we have to add to the boundary object B all nodes and edges $x, y \in A$ with $f(x) = f(y)$ and these nodes, that are the source or target of two edges that are equally mapped by f.

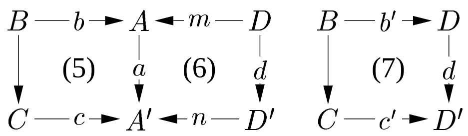
As pointed out in the introduction of this section the closure of initial pushouts under double pushouts is an important technical lemma which uses an additional class M' of morphisms which is closed under pushouts along M-morphisms. M' is closed under pushouts along M-morphisms means, that for a pushout $C \overset{n}{\to} D \overset{g}{\leftarrow} B$ over $C \overset{f}{\leftarrow} A \overset{m}{\to} B$ with $m, n \in M$ and $f \in M'$ it holds that also $g \in M'$. This is analogously defined for pullbacks.

Lemma 5.1. (closure property of initial pushouts)
Let M' be a class of morphisms closed under pushouts and pullbacks along M-morphisms. Moreover we assume to have initial pushouts over M'-morphisms. Then initial pushouts over M'-morphisms are closed under double pushouts. That means given an initial pushout (1) over $h_0 \in M'$ and a double pushout diagram (2) with d_0, $d_1 \in M$, then (3) and (4) are initial pushouts over $d \in M'$ respectively $h_1 \in M'$ for the unique $b : B \to D$ with $d_0 \circ b = b_0$ obtained by the initiality of (1).



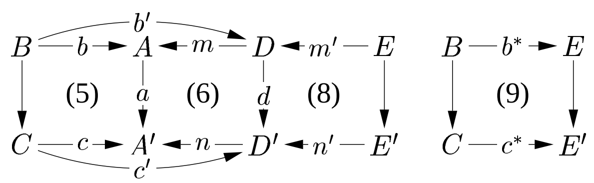


Proof:
Part I: Initial pushouts are closed under pushouts (in the opposite direction).
Given an initial pushout (5) over a and a pushout (6) with $m \in M$, then there is an initial pushout (7) over d with $m \circ b' = b$ and $n \circ c' = c$.



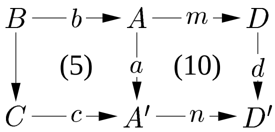
Since (5) is an initial pushout b' and c' uniquely exist such that (7) is a pushout. It remains to show the initiality.

For any pushout (8) we have that the composition (8) + (6) is a pushout, and with (5) being an initial pushout there are morphisms $b^* : B \to E$, $c^* : C \to E' \in M$ with $m \circ m' \circ b^* = b = m \circ b'$, $n \circ n' \circ c^* = c = n \circ c'$ and (9) is a pushout.

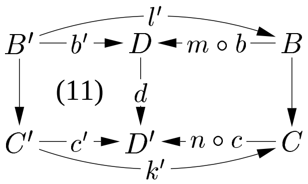


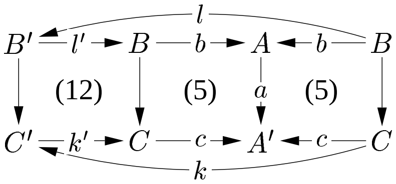
Since m and n are monomorphisms, it holds $b' = m' \circ b^*$ and $c' = n' \circ c$, therefore (7) is an initial pushout.

Part II: Initial pushouts are closed under pushouts (in the same direction).
Given an initial pushout (5) over $a \in M'$ and a pushout (10) with $m \in M$, then the composition (5) + (10) is an initial pushout.



Since M'-morphisms are closed under pushouts also $d \in M'$. Then the initial pushout (11) over d exists. Comparing (5) + (10) with (11) we get unique morphisms $l' : B' \to B$, $k' : C' \to C \in M$ with $m \circ b \circ l' = b'$, $n \circ c \circ k' = c'$ and (12) is a pushout.





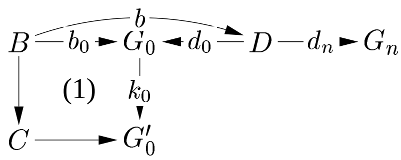
Then also (12) + (5) is a pushout and from the initial pushout (5) we get unique morphisms $l : B \to B'$, $k : C \to C' \in M$ with $b \circ l' \circ l = b$ and $c \circ k' \circ k = c$. With b and c being monomorphisms we get $l' \circ l = id_B$ and $k' \circ k = id_C$, and since l', k' are monomorphisms they are also isomorphisms. That means that (5) + (10) and (11) are isomorphic and (5) + (10) is an initial pushout over d.

Part III: Initial pushouts are closed under double pushouts.
(3) being an initial pushout follows directly from Part I.

(1) is a pushout along the M-morphism b and therefore a pullback, and since M' is closed under pullbacks also $B \to C \in M'$. Then we have $d \in M'$ and $h_1 \in M'$ with M' closed under pushouts, and by applying part II also (4) is an initial pushout. □

The following consistency condition for a transformation $t : G_0 \overset{*}{\Rightarrow} G_n$ and an extension morphism $k_0 : G_0 \to G_0'$ means intuitively that the boundary B of k_0 is preserved by t. In order to formulate this property we use the notion of a derived span $der(t) = G_0 \leftarrow D \to G_n$ of the transformation t, which connects the first and the last object.

Definition 5.3. (derived span and consistency)
The derived span of a direct transformation $G \overset{p,n}{\Longrightarrow} H$ as shown in Definition 4.1 is the span $G \leftarrow D \to H$. The derived span $der(t) = (G_0 \overset{d_0}{\leftarrow} D \overset{d_n}{\to} G_n)$ of a transformation $t : G_0 \overset{*}{\Rightarrow} G_n$ is the composition via pullbacks of the spans of the corresponding direct transformations.
A morphism $k_0 : G_0 \to G_0'$ is called consistent w.r.t. a transformation $t : G_0 \overset{*}{\Rightarrow} G_n$ with the derived span $der(t) = (G_0 \overset{d_0}{\leftarrow} D \overset{d_n}{\to} G_n)$ if there exist an initial pushout (1) over k_0 and a morphism $b \in M$ with $d_0 \circ b = b_0$.

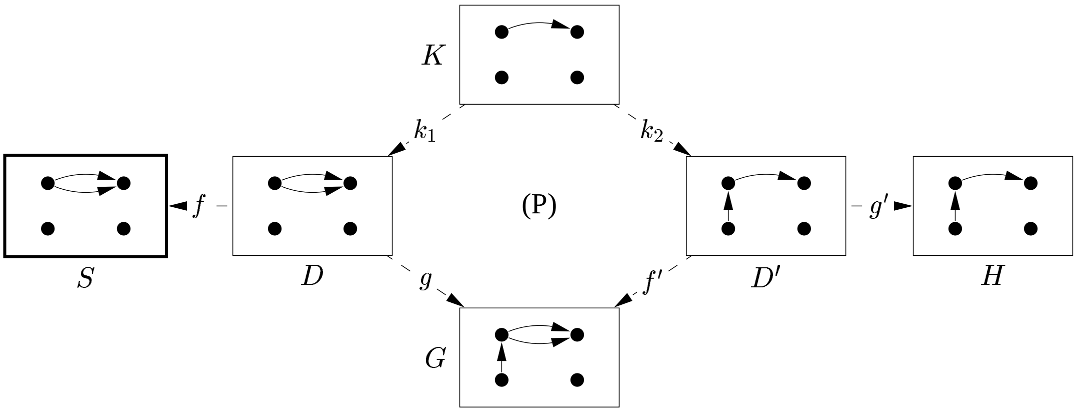


Remark 5.3. 1. The morphisms of the span $G \leftarrow D \rightarrow H$ are in M because M is closed under pushouts. This implies that the compositions of these spans exist and are M-morphisms, because pullbacks along M-morphisms exist and M-morphisms are closed under pullbacks in adhesive HLR categories.

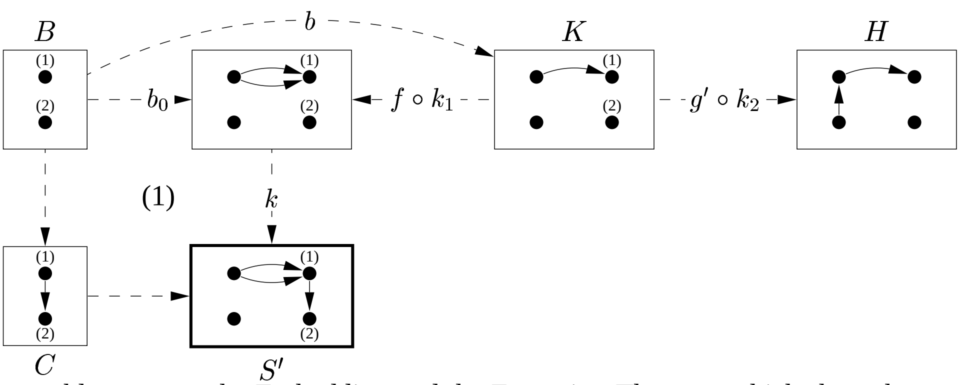
2. The consistency condition in [5], called JOIN condition, requires a suitable family $b_i : B \rightarrow D_i$ of morphisms from the boundary B to the context graphs D_i of the direct transformations. In fact, our consistency condition is equivalent to the existence of a corresponding family $(b_i)_{i=0,\ldots,n-1}$.

3. For the definition of consistency and for Theorem 5.1 below – but not for Theorem 5.2 – it would be sufficient to require the existence of a pushout over k_0 instead of an initial one. Moreover we need only the conditions 1 and 2 of Definition 3.1.

Example 5.2. (derived span and consistency)
Consider the direct transformation sequence $t : S \Rightarrow G \Rightarrow H$ from example 4.1. The following diagram shows the construction of the derived span $der(t) = (S \overset{f \circ k_1}{\longleftarrow} K \overset{g' \circ k_2}{\longrightarrow} H)$ with the pullback (P).



Then the extension morphism $k : S \rightarrow S'$ as shown in the following diagram is consistent with respect to the transformation $t : S \overset{*}{\Rightarrow} H$. We can construct the inital pushout (1) over k - similar to the construction in example 5.1 - and for the depicted injective morphism b it holds that $f \circ k_1 \circ b = b_0$.

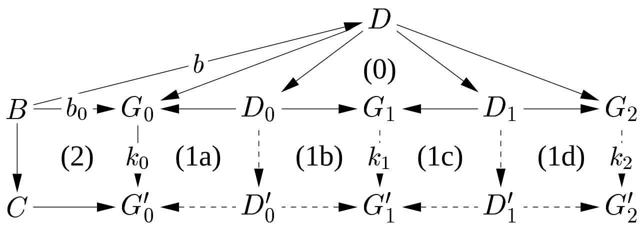


Now we are able to prove the Embedding and the Extension Theorem which show that consistency is sufficient and also necessary for the construction of extension diagrams. Moreover, we obtain a direct construction of the extension $k_n : G_n \rightarrow G'_n$ in the extension diagram.

Theorem 5.1. (Embedding Theorem)
Given a transformation $t : G_0 \overset{*}{\Rightarrow} G_n$ and a morphism $k_0 : G_0 \to G'_0$ which is consistent w.r.t. t, then there is an extension diagram for t and k_0 (see (1) in Definition 5.1).

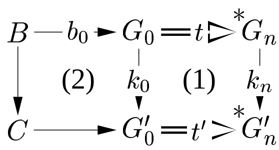
Proof:
This theorem is proven by induction over the number of direct transformations. We show only the case $n = 2$, which can be adapted to arbitrary n.

[n = 2] We construct the pullback (0) leading to the derived span $G_0 \leftarrow D_0 \leftarrow D \to D_1 \to G_2$ of the transformation $t : G_0 \Rightarrow^* G_2$. Given k_0 consistent w.r.t. t, we have the initial pushout (2) over k_0 and $b : B \to D$.



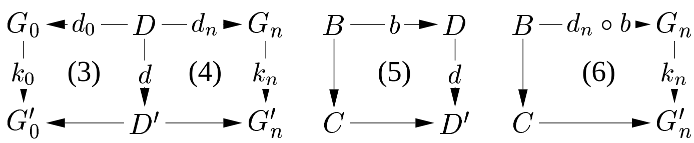
This leads to the M-morphisms $B \to D_0$ and $B \to D_1$ such that first D'_0 can be constructed as pushout object of $B \to D_0$ and $B \to C$ leading by decomposition to the pushout (1a) and by construction to the pushout (1b). Then D'_1 can be constructed as the pushout object of $B \to D_1$ and $B \to C$ leading by decomposition to the pushout (1c) and by construction to the pushout (1d). The given transformation $t : G_0 \overset{*}{\Rightarrow} G_2$ together with the pushouts (1a) - (1d) constitutes the required extension diagram. □

Theorem 5.2. (Extension Theorem)
Given a transformation $t : G_0 \overset{*}{\Rightarrow} G_n$ with the derived span $der(t) = (G_0 \overset{d_0}{\leftarrow} D \overset{d_n}{\to} G_n)$ and an extension diagram (1)



with the initial pushout (2) over $k_0 \in M'$ for some class M' closed under pushouts and pullbacks along M-morphisms and with initial pushouts over M'-morphisms, then we have

1. k_0 consistent w.r.t. $t : G_0 \overset{*}{\Rightarrow} G_n$ with morphism $b : B \to D$,
2. a direct transformation $G'_0 \Rightarrow G'_n$ via $der(t)$ and match k_0 given by pushouts (3) and (4) with d, $k_n \in M'$,
3. initial pushouts (5) and (6) over d resp. k_n.



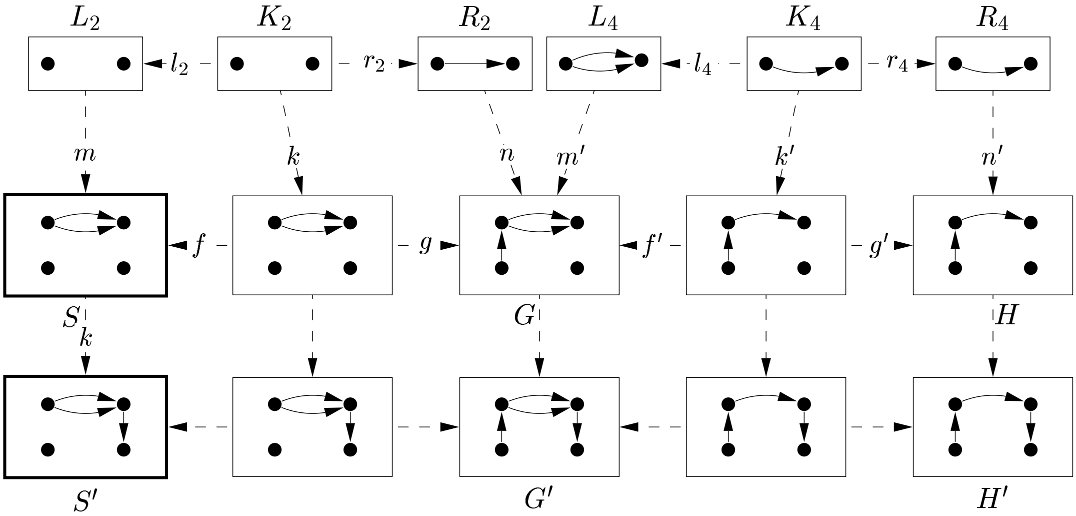
Proof:
This theorem is proven by induction over the number of direct transformations. We show only the case $n = 2$, which can be adapted to arbitrary n.

[n = 2] Given t and k_0 with the initial pushout (2) and the extension diagram given by pushouts (1a) - (1d) in proof of Theorem 5.1, where D is the pullback object in (0). Initiality of (2) and the pushout (1a) lead to $b_0^* : B \to D_0$ and by Lemma 5.1 to an initial pushout over k_1. This new initiality and the pushout (1c) lead to $b_1 : B \to D_1$. The morphisms b_0^* and b_1 lead to an induced $b : B \to D$ - using the pullback properties of (0) - which allows to show the consistency of k_0 w.r.t. t. This consistency immediately implies the pushout complement D' in (3) and the pushout (4) of d and d_n. Finally, the double pushout (3), (4) implies by Lemma 5.1 the initial pushouts (5) and (6) from (2). □

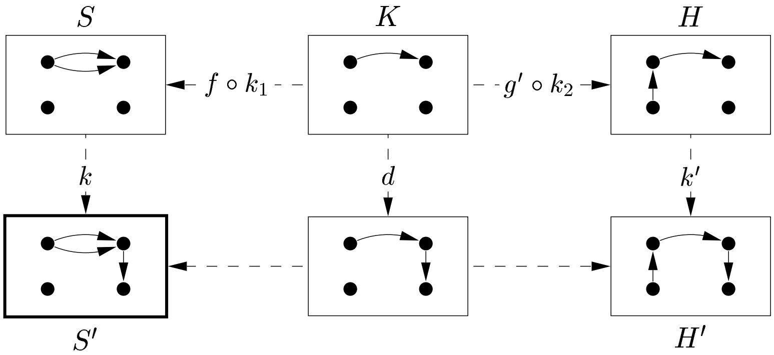
Remark 5.4. The extension theorem shows

1. The consistency of k_0 w.r.t. t is necessary for the existence of the extension diagram.
2. The extension diagram (1) can be represented by a direct transformation with the match k_0 and the comatch k_n (see Definition 4.1).
3. The extension $k_n : G_n \to G'_n$ can be constructed by a pushout (6) of G_n and the context C along the boundary B with $d_n \circ b : B \to G_n$.

Example 5.3. (embedding and extension theorem)
Consider the transformation sequence $t : S \Rightarrow G \Rightarrow H$ from example 4.1 and the extension morphism $k : S \to S'$ from example 5.2, that is consistent with respect to t. Then we can conclude with the Embedding Theorem that there is an extension diagram over k and t as shown in the following diagram.

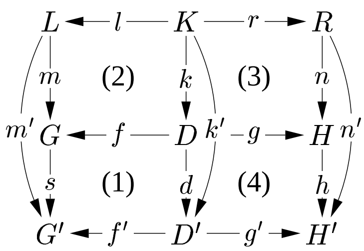


In the other direction, for the extension diagram above with the derived span $der(t) = (S \overset{f \circ k_1}{\longleftarrow} K \overset{g' \circ k_2}{\longrightarrow} H)$ we have shown in example 5.2 that the initial pushout over k exists. Applying theorem 5.2 with class $M' = M$ of injective graph morphisms we can conclude that k is consistent with respect to t, that there are initial pushouts over d and k' and there is a transformation $S \Rightarrow H'$ via $der(t)$ and k with d and k' being injective.



The inverse construction to the embedding of a transformation into a larger context - as shown in the Embedding Theorem 5.1 - is the restriction of a transformation to a smaller context, which is shown now for the special case of direct transformations.

Theorem 5.3. (Restriction Theorem)
Given a direct transformation $G' \overset{p,m'}{\Longrightarrow} H'$, a morphism $s : G \to G' \in M$ and a match $m : L \to G$ such that $s \circ m = m'$, then there is a direct transformation $G \overset{p,m}{\Rightarrow} H$ leading to the following extension diagram.



Proof:
First we construct the pullback (1) over s and f' and obtain the induced morphism k by (1) in comparison with $m \circ l$ and k'. With the pushout-pullback decomposition both (1) and (2) are pushouts using $l, s \in M$. Now we construct the pushout (3) over k and r, obtain the induced morphism h and by pushout decomposition also (4) is a pushout. □

Remark 5.5. In fact, it is sufficient to require $s \in M'$ in Theorem 5.3 for a suitable morphism class M', where the M-M' pushout-pullback decomposition property holds: Theorem 3.2 item 2 holds also with $l \in M$ and $w \in M'$ in the diagram (1)+(2) of Theorem 3.2.

For an extension of the Restriction Theorem to transformations of length $n \geq 2$ see [5] in the graph case.

6. Critical Pairs and Local Confluence of Adhesive HLR systems

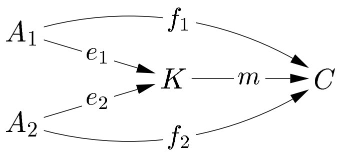
Critical pairs and local confluence have been studied for hypergraph transformation in [26] and for typed attributed graph transformation in [17]. In this section we present a categorical version in adhesive HLR categories. As additional requirements we only need an E'-M' pair factorization for cospans of morphisms - in analogy to the well-known epi-mono-factorization of morphisms - and initial pushouts over M'-morphisms. These assumptions are stated where necessary. Otherwise we only assume to have an adhesive HLR system.

It is well-known that local confluence and termination imply confluence. But we only analyze local confluence in this paper and no termination nor general confluence.

Definition 6.1. (confluence, local confluence)
A pair of transformations $H_1 \overset{*}{\Leftarrow} G \overset{*}{\Rightarrow} H_2$ is confluent if there are transformations $H_1 \overset{*}{\Rightarrow} X$ and $H_2 \overset{*}{\Rightarrow} X$. An adhesive HLR system is locally confluent, if this property holds for each pair of direct transformations, it is confluent, if it holds for all pairs of transformations.

In order to define and construct critical pairs we introduce the notion of an E'-M' pair factorization.

Definition 6.2. (E'-M' pair factorization)
An adhesive HLR category has an E'-M' pair factorization, if M' is a class of morphisms closed under pushouts and pullbacks along M-morphisms and E' is a class of morphism pairs with the same codomain, where we have for each pair of morphisms $f_1 : A_1 \to C$, $f_2 : A_2 \to C$ that there is an object K and morphisms $e_1 : A_1 \to K$, $e_2 : A_2 \to K$, $m : K \to C$ with $(e_1, e_2) \in E'$, $m \in M'$ such that $m \circ e_1 = f_1$ and $m \circ e_2 = f_2$.



Remark 6.1. It is sufficient to require this property for matches $f_i = m_i : L_i \to G$ (i = 1, 2). The closure properties of M' are needed in Theorem 6.1 and Theorem 6.2.

The intuitive idea of morphism pairs $(e_1, e_2) \in E'$ in most example categories is that the pair is jointly surjective resp. jointly epimorphic. This can be achieved in categories $\boldsymbol{C}$ with binary coproducts and an E_0-M_0 factorization of morphisms, where E_0 is a class of epimorphisms and M_0 is a class of monomorphisms. Given $A_1 \overset{f_1}{\to} C \overset{f_2}{\leftarrow} A_2$ we simply take an E_0-M_0 factorization $f = m \circ e$ of the induced morphism $f : A_1 + A_2 \to C$ and define $e_1 = e \circ i_1$ and $e_2 = e \circ i_2$, where i_1, i_2 are the coproduct injections. In this case we have that $M' = M_0$ is a class of monomorphisms. If the category has no binary coproducts, or the construction above is not always adequate - as in the case of typed attributed graph transformation - we may have another alternative to obtain an E'-M' pair factorization. In [16] an explicit E'-M' pair factorization for typed attributed graph transformation is provided, where M'-morphisms are not necessarily injective on the data type part and hence M' is not a class of monomorphisms.

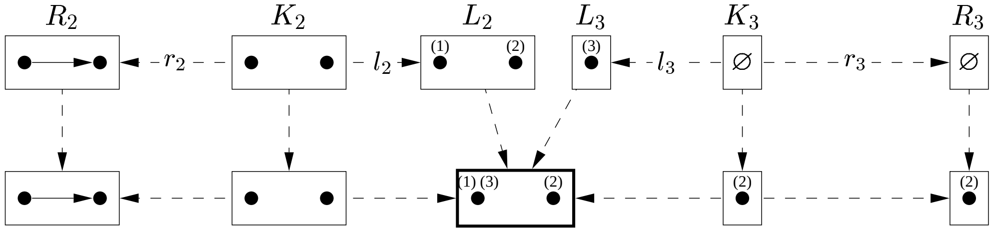
The main idea to prove local confluence is to show local confluence explicitely only for critical pairs based on the notion of parallel independence (see [10]).

Definition 6.3. (critical pair)
Given an E'-M' pair factorization, a critical pair is a pair of parallel dependent direct transformations $P_1 \overset{p_1, o_1}{\Longleftarrow} K \overset{p_2, o_2}{\Longrightarrow} P_2$ such that $(o_1, o_2) \in E'$ for the corresponding matches o_1 and o_2.

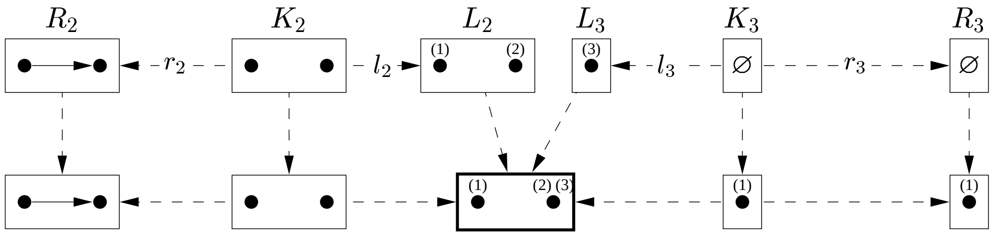
Note that this definition may include the case $p_1 = p_2$ and $o_1 = o_2$, which implies $P_1 \cong P_2$ and hence immediately local confluence.

Example 6.1. (critical pairs in $ExAHS$)
Consider the adhesive HLR system $ExAHS$ introduced in example 4.1. We use an E'-M' pair factorization, where in E' are pairs of jointly epimorphic morphisms and M' is the class of all monomorphisms. Then we have the following five critical pairs (up to isomorphism).

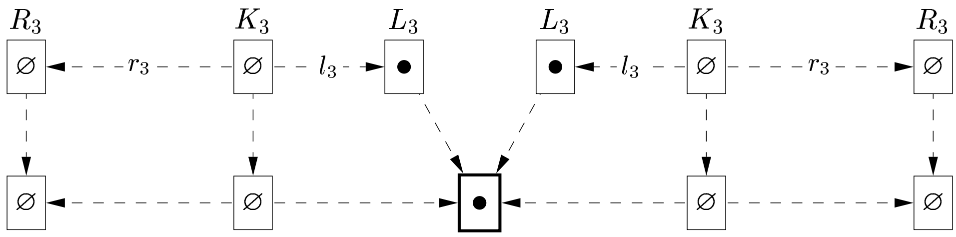
The first one consists of the productions $addEdge$ and $deleteVertex$, where $deleteVertex$ deletes the souce node of the edge inserted by $addEdge$. Therefore these transformations are non-parallel independent. The choice of the matches and their codomain object makes sure they are jointly surjective.



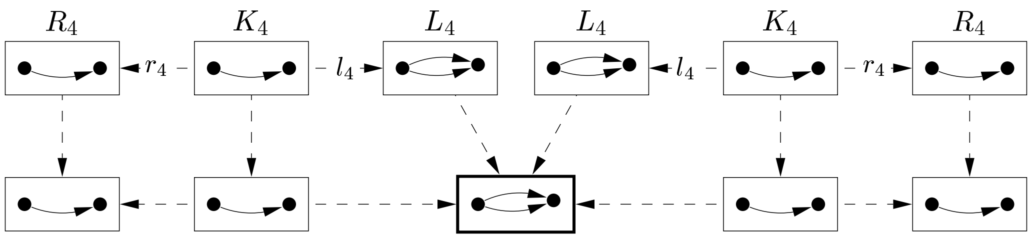
The second critical pair has the same productions $addEdge$ and $deleteVertex$, but $deleteVertex$ deletes the target node of the edge inserted by $addEdge$.



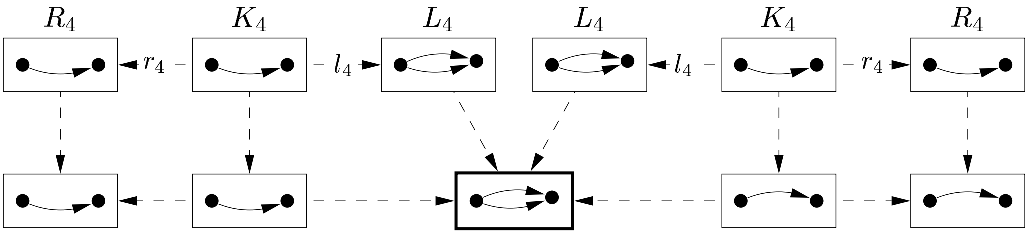
The third critical pair deals with the production $deleteVertex$ twice: the same vertex is deleted by both transformations.



The fourth critical pair contains the production $del1of2edges$ twice, where the same edge is deleted by both transformations.

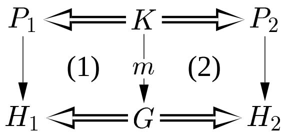


The last critical pair consists also of the production $del1of2edges$ twice, where, however, now different edges are deleted by the transformations. The first one deletes the upper and the second one the lower edge. This violates again parallel independence, because in order to apply $del1of2edges$ both edges are necessary.



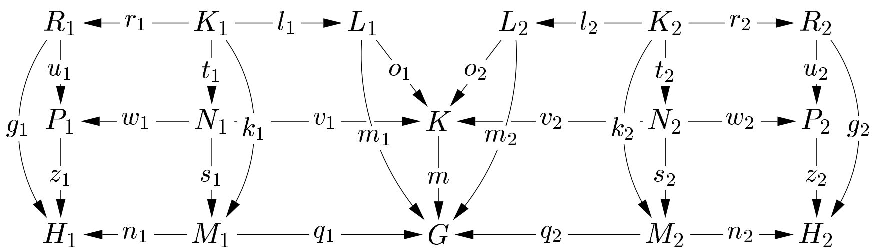
The first step towards local confluence is to show the completeness of critical pairs.

Theorem 6.1. (completeness of critical pairs)
Consider an adhesive HLR system with an E'-M' pair factorization and $M' \subseteq M$. For each pair of non-parallel independent direct transformations $H_1 \overset{p_1,m_1}{\Longleftarrow} G \overset{p_2,m_2}{\Longrightarrow} H_2$ there is a critical pair $P_1 \overset{p_1,o_1}{\Longleftarrow} K \overset{p_2,o_2}{\Longrightarrow} P_2$ with extension diagrams (1) and (2) and $m \in M'$.



Remark 6.2. If $M' \nsubseteq M$ we have to require in addition that the pushout-pullback decomposition property holds (see Remark 5.5).

Proof:
With the E'-M' pair factorization for m_1 and m_2 we get an object K and morphisms $m : K \to G \in M'$, $o_1 : L_1 \to K$ and $o_2 : L_2 \to K$ with $(o_1, o_2) \in E'$ such that $m_1 = m \circ o_1$ and $m_2 = m \circ o_2$. Theorem 5.3 gives us the following extension diagrams.



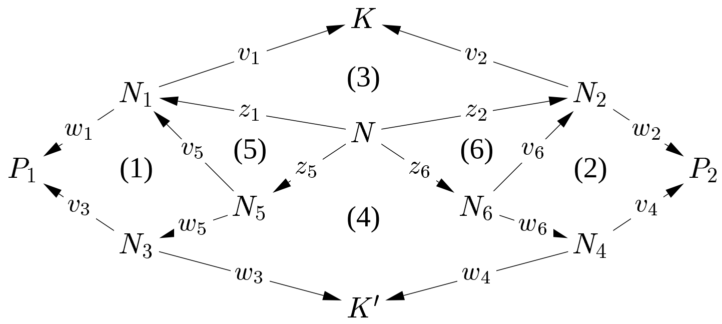
$P_1 \overset{p_1,o_1}{\Longleftarrow} K \overset{p_2,o_2}{\Longrightarrow} P_2$ are non-parallel independent. Otherwise there are morphisms $i : L_1 \to N_2$ and $j : L_2 \to N_1$ with $v_2 \circ i = o_1$ and $v_1 \circ j = o_2$. Then $q_2 \circ s_2 \circ i = m \circ v_2 \circ i = m \circ o_1 = m_1$ and $q_1 \circ s_1 \circ j = m \circ v_1 \circ j = m \circ o_2 = m_2$, that means $H_1 \overset{p_1,m_1}{\Longleftarrow} G \overset{p_2,m_2}{\Longrightarrow} H_2$ are parallel independent, contradiction. □

From [26] in the case of hypergraph transformation it is known already that the confluence of critical pairs is not sufficient to show the local confluence in general. In fact, we need a slightly stronger property, called strict confluence.

Definition 6.4. (strict confluence of critical pairs)
A critical pair $P_1 \overset{p_1,o_1}{\Longleftarrow} K \overset{p_2,o_2}{\Longrightarrow} P_2$ is called strictly confluent, if we have

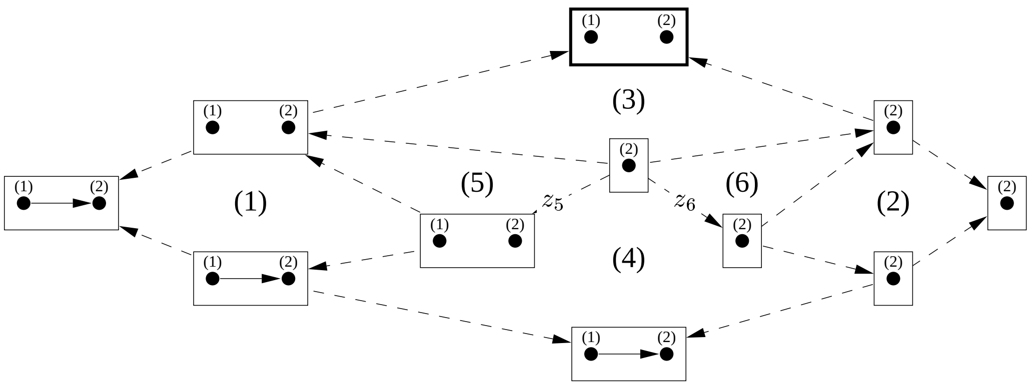
1. confluence: the critical pair is confluent, i.e. there are transformations $P_1 \overset{*}{\Rightarrow} K'$, $P_2 \overset{*}{\Rightarrow} K'$ with derived spans $der(P_i \overset{*}{\Rightarrow} K') = P_i \overset{v_{i+2}}{\leftarrow} N_{i+2} \overset{w_{i+2}}{\rightarrow} K'$ for $i = 1, 2$.

2. strictness: Let $der(K \overset{p_i,o_i}{\Longrightarrow} P_i) = K \overset{v_i}{\leftarrow} N_i \overset{w_i}{\rightarrow} P_i$ ($i = 1, 2$) and N_5, N_6 and N be the pullback objects of the pullbacks (1), (2) and (3), respectively. Then there are morphisms z_5 and z_6 such that (4), (5) and (6) commute.



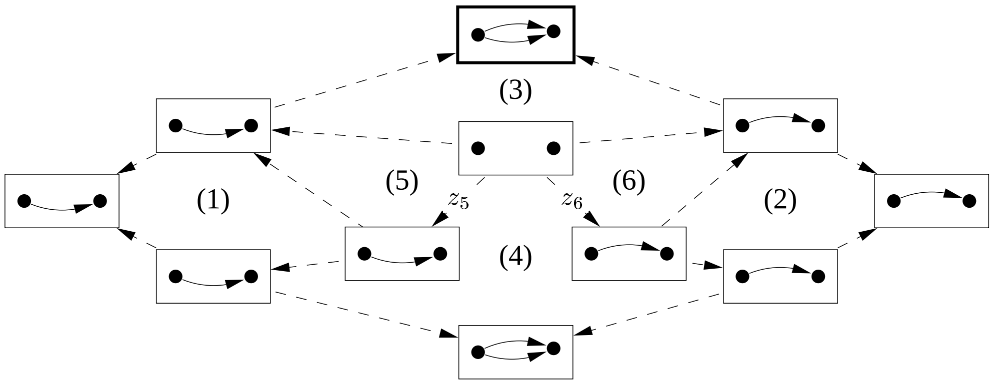
Remark 6.3. The strictness condition is a combination of corresponding conditions stated in [26] and [17]. More precisely, commutativity of (4) is required in [26] and that of (5) and (6) in [17]. In [26] however, commutativity of (5) and (6) seems to be a consequence of the inclusion properties. The intuitive idea of strictness is that the common part N, which is preserved by each transformation of the critical pair, is also preserved by the transformations $P_1 \overset{*}{\Rightarrow} K'$ and $P_2 \overset{*}{\Rightarrow} K'$ and mapped by the same morphism $N \rightarrow K'$.

Example 6.2. (strict confluence in $ExAHS$)
In our adhesive HLR system $ExAHS$, all critical pairs defined in example 6.1 are strictly confluent. The confluence of the first and the second critical pair is established by applying no further transformation to the first graph and applying $addVertex$ and $addEdge$ to the second one. This is shown in the following diagram for the first critical pair, and analogously works for the second one. The strictness condition holds for the shown morphisms z_5 and z_6.



The third critical pair is confluent, since both transformations result in the empty graph. In the strictness diagram all graphs except for K are empty, and therefore the strictness condition is fulfilled. Similarly for the fourth critical pair, both transformations result in the same graph with two nodes and one edge between them. This is the graph for all objects in the strictness diagram except for K, which has two edges between the two nodes.

For the last critical pair, we can reverse the deletion of the edges by applying the production $addEdge$ for both graphs. The following diagram shows that the strictness condition holds, since all morphisms are inclusions.

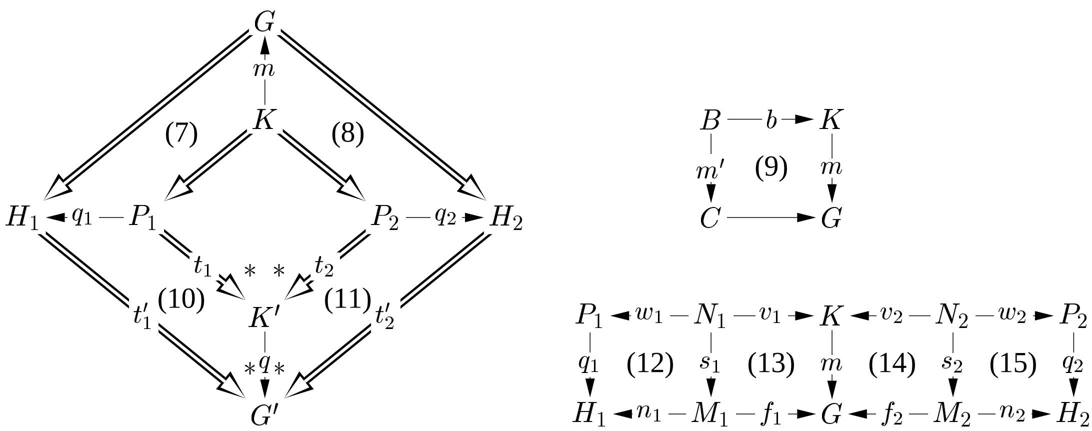


Finally our last main result states that strict confluence of all critical pairs implies local confluence. This result is also known as the Critical Pair Lemma (see [1, 26, 17]).

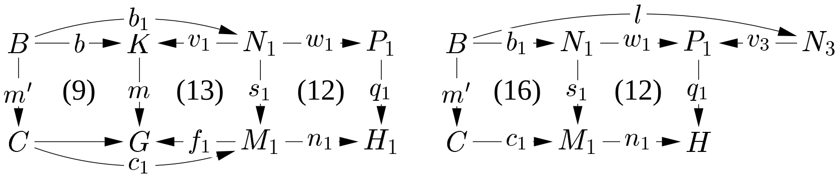
Theorem 6.2. (Local Confluence Theorem - Critical Pair Lemma)
An adhesive HLR system with an E'-M' pair factorization, $M' \subseteq M$ and initial pushouts over M'-morphisms is locally confluent, if all its critical pairs are strictly confluent.

Remark 6.4. See Remark 6.2 for the case $M' \not\subseteq M$. In the proof we need that M is closed under decomposition (see Definition 3.1 It. 1) in order to show $b_3 \in M$.

Proof:
Given a pair of direct transformations $H_1 \overset{p_1,m_1}{\Longleftarrow} G \overset{p_2,m_2}{\Longrightarrow} H_2$ we have to show the existence of the transformations $t_1' : H_1 \overset{*}{\Rightarrow} G'$ and $t_2' : H_2 \overset{*}{\Rightarrow} G'$. If the given pair is parallel independent this follows from the Local Church-Rosser Theorem. If the given pair is not parallel independent, Theorem 6.1 implies the existence of a critical pair $P_1 \overset{p_1,o_1}{\Longleftarrow} K \overset{p_2,o_2}{\Longrightarrow} P_2$ with extension diagrams (7) and (8) and $m \in M'$. By assumption this critical pair is strictly confluent leading to transformations $t_1 : P_1 \overset{*}{\Rightarrow} K'$, $t_2 : P_2 \overset{*}{\Rightarrow} K'$ and the diagram in Definition 6.4.



Now let (9) be an initial pushout over $m \in M'$ and consider the pairs of pushouts (12), (13) and (14), (15) corresponding to the extension diagrams (7) and (8) respectively.



Initiality of (9) applied to the pushout (13) leads to unique $b_1, c_1 \in M$ such that $v_1 \circ b_1 = b$, $m_1 \circ c_1 = C \to G$ and (16) is a pushout. By Lemma 5.1 (16) is an initial pushout over s_1. Dually we obtain $b_2, c_2 \in M$ with $v_2 \circ b_2 = b$. Using the pullback property of (3) in Definition 6.4 we obtain a unique $b_3 : B \to N$ with $z_1 \circ b_3 = b_1$ and $z_2 \circ b_3 = b_2$. Moreover $b_1, z_1 \in M$ implies $b_3 \in M$ by the decomposition property of M. In order to show the consistency of q_1 w.r.t. t_1 we have to construct $b'_3 \in M$ such that $v_3 \circ b'_3 = w_1 \circ b_1$ where (16)+(12) is the initial pushout over q_1 by Lemma 5.1. In fact, this holds for $b'_3 = w_5 \circ z_5 \circ b_3 \in M$ using (5) in Definition 6.4.

Dually q_2 is consistent w.r.t. t_2 using $b'_4 = w_6 \circ z_6 \circ b_3 \in M$ and (6) in Definition 6.4. By the Embedding Theorem we obtain extension diagrams (10) and (11), where the morphism $q : K' \to G'$ is the same in both cases. This equality can be shown using part 3 of the Extension Theorem, where q is determined by an initial pushout of $m' : B \to C$ and $w_3 \circ b'_3 : B \to K'$ in the first and $w_4 \circ b'_4 : B \to K'$ in the second case and we have $w_3 \circ b'_3 = w_4 \circ b'_4$ using the commutativity of (4) in Definition 6.4. □

Example 6.3. (local confluence of $ExAHS$)
In $ExAHS$, we have $M' = M$ and initial pushouts over injective graph morphisms (see example 5.1). Therefore all preconditions for theorem 6.2 are fulfilled. Since all critical pairs in $ExAHS$ are strictly confluent, as shown in example 6.2, $ExAHS$ is locally confluent.

7. Conclusion

In this paper based on our ICGT'04 paper [11] we have introduced adhesive HLR categories and systems combining the framework of adhesive categories in [18] and of HLR systems in [10]. We claim that this new framework is most important for different theories of graphs and graphical structures in computer science, which are mainly based on pushout constructions. As shown in this paper this includes first of all the double pushout approach in the theory of graph transformation and HLR systems [5, 10, 3], where important new results have been presented in this framework which are already applied to typed attributed graph transformation in [16]. Constraints and application conditions for DPO-transformations of adhesive HLR systems are considered already in [7]. On the other hand pushouts and different kinds of graphs have also been used to derive well-behaved labeled transition systems, where bisimulation is a congruence, by Leifer and Milner in [21], by Sassone and Sobociński in [28] and by König and Ehrig in [12]. In fact there is a close relationship between these approaches, which is discussed in [6] concerning bigraphs in [21] and DPO-approach with borrowed context in [12]. For a more detailed discussion of all three approaches and the role of adhesive categories in this context we refer to [29]. In any case we agree with [18] that the role of adhesive categories - and even more adhesive HLR categories - for all kinds of applications is most likely to become comparable to the role of cartesian closed categories for simply typed lambda calculi as pointed out by Lambek and Scott in [20].

References

[1] Baader, F., Nipkow, T.: *Term Rewriting and All That,* Cambridge University Press, 1998.

[2] Brown, R., Janelidze, G.: Van Kampen Theorems for Categories of Covering Morphisms in Lextensive Categories, *Journal of Pure and Applied Algebra*, **119**, 1997, 255–263.

[3] Corradini, A., Ehrig, H., Heckel, R., Löwe, M., Montanari, U., Rossi, F.: Algebraic Approaches to Graph Transformation I : Basic Concepts and Double Pushout Approach, in: *Handbook of Graph Grammars and Computing by Graph Transformation, Volume 1: Foundations* (G. Rozenberg, Ed.), chapter 3, World Scientific, 1997.

[4] Corradini, A., Gadducci, F.: On Term Graphs as an Adhesive Category, *Workshop Termgraph 2004*, ENTCS, Elsevier, 2004.

[5] Ehrig, H.: Introduction to the Algebraic Theory of Graph Grammars (A Survey), in: *Graph Grammars and their Application to Computer Science and Biology*, vol. 73 of *LNCS*, Springer, 1979, 1–69.

[6] Ehrig, H.: Bigraphs meet Double Pushouts, *EATCS Bulletin No. 78*, 2002, 72–85.

[7] Ehrig, H., Ehrig, K., Habel, A., Pennemann, K.: Theory of Constraints and Application Conditions: From Graphs to High-Level Structures, *Fundamenta Informaticae*, 2005, this issue.

[8] Ehrig, H., Ehrig, K., Prange, U., Taentzer, G.: Fundamental Theory for Typed Attributed Graphs and Graph Transformation, *Fundamenta Informaticae*, 2005, this issue.

[9] Ehrig, H., Gajewsky, M., Parisi-Presicce, F.: High-Level Replacement Systems with Applications to Algebraic Specifications and Petri Nets, in: *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 3: Concurrency, Parallelism, and Distribution* (G. Rozenberg, U. Montanari, H. Ehrig, H.-J. Kreowski, Eds.), chapter 6, World Scientific, 1999, 341–400.

[10] Ehrig, H., Habel, A., Kreowski, H.-J., Parisi-Presicce, F.: Parallelism and Concurrency in High-Level Replacement Systems, *Math. Struct. in Comp. Science*, **1**, 1991, 361–404.

[11] Ehrig, H., Habel, A., Padberg, J., Prange, U.: Adhesive High-Level Replacement Categories and Systems, in: *Proceedings of ICGT 2004* (H. Ehrig, G. Engels, F. Parisi-Presicce, G. Rozenberg, Eds.), vol. 3256 of *LNCS*, Springer, 2004, 144–160.

[12] Ehrig, H., König, B.: Deriving Bisimulation Congruences in the DPO Approach to Graph Rewriting, in: *Proc. FOSSACS 2004*, vol. 2987 of *LNCS*, Springer, 2004, 151 – 166.

[13] Ehrig, H., Orejas, F., Braatz, B., Klein, M., Piirainen, M.: A Generic Component Concept for System Modeling, in: *Proc. FASE 2002*, vol. 2306 of *LNCS*, Springer, 2002, 33–48.

[14] Ehrig, H., Orejas, F., Braatz, B., Klein, M., Piirainen, M.: A Component Framework for System Modeling Based on High-Level Replacement Systems, *Software and Systems Modeling 3(2)*, 2004, 114–134.

[15] Ehrig, H., Prange, U.: Weak Adhesive High-Level Replacement Categories and Systems: A Unifying Framework for Graph and Petri Net Transformations, in: *Algebra, Meaning and Computation*, LNCS, Springer, 2006, To appear.

[16] Ehrig, H., Prange, U., Taentzer, G.: Fundamental Theory for Typed Attributed Graph Transformation, in: *Proceedings of ICGT 2004* (H. Ehrig, G. Engels, F. Parisi-Presicce, G. Rozenberg, Eds.), vol. 3256 of *LNCS*, Springer, 2004, 161–177.

[17] Heckel, R., Küster, J., Taentzer, G.: Confluence of Typed Attributed Graph Transformation, in: *Proc. ICGT 2002*, vol. 2505 of *LNCS*, Springer, 2002, 161–176.

[18] Lack, S., Sobociński, P.: Adhesive Categories, in: *Proc. of FOSSACS '04*, vol. 2987 of *LNCS*, Springer, 2004, 273–288.

[19] Lack, S., Sobociński, P.: Adhesive and Quasiadhesive Categories, *Theoretical Informatics and Applications*, **39**(3), 2005, 511–546.

[20] Lambek, J., Scott, P.: *Introduction to Higher Order Categorical Logic*, Cambridge University Press, 1986.

[21] Leifer, J., Milner, R.: Deriving Bisimulation Congruences for Reactive Systems, in: *Proc. CONCUR 2000*, vol. 1877 of *LNCS*, Springer, 2000, 243–258.

[22] Löwe, M., Korff, M., Wagner, A.: An Algebraic Framework for the Transformation of Attributed Graphs, in: *Term Graph Rewriting: Theory and Practice*, John Wiley and Sons Ltd., 1993, 185–199.

[23] MacLane, S.: *Categories for the Working Mathematician*, Springer, 1972.

[24] Padberg, J., Ehrig, H., Ribeiro, L.: Algebraic High-Level Net Transformation Systems, in: *Mathematical Structures in Computer Science Vol 2*, 1995, 217–256.

[25] Padberg, J., Taentzer, G.: *Embedding of Derivations in High-Level Replacement Systems*, Technical Report 1993/9, TU Berlin, 1993.

[26] Plump, D.: Hypergraph Rewriting: Critical Pairs and Undecidability of Confluence, in: *Term Graph Rewriting: Theory and Practice* (M. Sleep, M. Plasmeijer, M. van Eekelen, Eds.), chapter 15, John Wiley & Sons Ltd, 1993, 201–213.

[27] Prange, U.: *Confluence of Adhesive HLR Systems with Applications to Typed Attributed Graph Transformation Systems. Diploma Thesis*, Technical Report 2004/22, TU Berlin, 2004.

[28] Sassone, V., Sobociński, P.: Deriving Bisimulation Congruences: 2-Categories vs Precategories, in: *Proc. FOSSACS 2003*, vol. 2620 of *LNCS*, Springer, 2003, 409–424.

[29] Sobociński, P.: *Deriving Process Congruences from Reaction Rules*, Ph.D. Thesis, BRICS Dissertation Series, 2004.