Electronic Communications of the EASST
Volume 73 (2016)

Graph Computation Models
Selected Revised Papers from GCM 2015

On Single-Pushout Rewriting of Partial Algebras

Michael Löwe and Marius Tempelmeier

21 pages

Guest Editors: Detlef Plump
Managing Editors: Tiziana Margaria, Julia Padberg, Gabriele Taentzer
ECEASST Home Page: http://www.easst.org/eceasst/ ISSN 1863-2122

On Single-Pushout Rewriting of Partial Algebras

Michael Löwe[1] **and Marius Tempelmeier**[2]

[1] michael.loewe@fhdw.de
Fachhochschule für die Wirtschaft Hannover, Germany

[2] marius@tempelmeier.de
Sennheiser electronic GmbH & Co. KG

Abstract: We introduce Single-Pushout Rewriting for *arbitrary* partial algebras. Thus, we give up the usual restriction to graph structures, which are algebraic categories with unary operators only. By this generalisation, we obtain an integrated and straightforward treatment of graphical structures (objects) and attributes (data). We lose co-completeness of the underlying category. Therefore, a rule is no longer applicable at any match. We characterise the new application condition and make constructive use of it in some practical examples.

Keywords: Graph Transformation, Single Pushout Rewriting, Partial Algebras

1 Introduction

The current frameworks for the (algebraic) transformation of typed graphs are not completely satisfactory from the software engineering perspective. For example, it is hardly possible to specify and handle associations with "at-most-one"-multiplicity, since most frameworks are based on some (adhesive) categories of graphs which allow multiple edges between the same pair of vertices.[1]

Another example is the handling of attributes. The standard approaches to the transformation of attributed graphs, compare for example [4, 14], explicitly distinguish two parts, i. e. the *structure part* (objects and links) which can be changed by transformations and the base-type and -operation part (data) which is immutable. Typically, objects can be attributed with data via some special edges the source of which is in the structure part and the target of which is data. This set-up either leads to set-valued or immutable attribute structures. Both is not satisfactory from the software engineering point of view.[2]

Another problem in current frameworks for attributed graphs is the infiniteness of rules stipulated by the infiniteness of the term algebra which is typically used in the rules. Even if the algebra for the objects which are subject to transformation is finite (for example integers modulo some maximum), the term algebra tends to contain infinitely many terms.

All these problems are more or less caused by the usage of total algebras. In this paper, we use partial algebras instead as the underlying category for single-pushout rewriting. In partial algebras, operation definitions can be changed without deleting and adding an object (edge). Thus,

[1] Typically, some negative application conditions [8] are employed to handle these requirements making the framework more complicated.

[2] In object-oriented programming languages, for example, attributes have the standard multiplicity `0..1`.

we get a straightforward model for "at-most-one" associations. We also give up the distinction between structure and data, i. e. we allow arbitrary signatures which are able to integrate both parts. We lose co-completeness of the base category and import some application conditions into single-pushout rewriting. But we gain a seamless integration of structure and data. Finally, partial term algebras in the rules help to keep rules finite.

The paper is a short version of [17] and an extended version of [15]. Section 2 introduces our concept of partial algebra. We show explicitly the similarities between partial algebras and hypergraphs. Section 3 provides sufficient and necessary conditions for the existence of pushouts in categories of partial algebras and partial morphisms. It contains our main results. Section 4 defines the new single-pushout approach and compares it to the sesqui-pushout approach [3]. Section 5 demonstrates by some examples that the application conditions in the introduced rewriting approach are useful in many situations. Finally, Section 6 discusses related work and provides some conclusions.

2 Graphs and Partial Algebras

In this section, we introduce the basic notions and constructions for partial algebras. We use a rather unusual approach in order to emphasise the close connection of categories of partial algebras to categories of hypergraphs. We employ a kind of objectification for partial mappings. A partial map $f : A \to B$ is not just a subset of $A \times B$ satisfying the uniqueness condition $(*)$ $(a,b_1),(a,b_2) \in f$ implies $b_1 = b_2$. Instead, a partial map $f : A \to B$ is a triple $(f, d_f : f \to A, c_f : f \to B)$ consisting of a set f of *map entries* together with two *total* maps $d_f : f \to A$ and $c_f : f \to B$ which provide the argument and the return value for every entry respectively. The uniqueness condition $(*)$ above translates to $\forall e_1, e_2 \in f : d_f(e_1) = d_f(e_2) \Longrightarrow e_1 = e_2$ in this set-up.

A *signature* $\Sigma = (S,O)$ consists of a set of sorts S and a domain- and co-domain-indexed family of operations $O = (O_{w,v})_{w,v \in S^*}$.[3] A *graph* G wrt. a signature consists of a carrier set G_s (of vertices) for every sort $s \in S$ and a set of hyperedges $\left(f^G, d^G_f : f^G \to G^w, c^G_f : f^G \to G^v\right)$ for every operation $f \in O_{w,v}$ and $w,v \in S^*$ where the *total* mappings d^G_f and c^G_f provide the "arguments" and "return values".[4] A *graph morphism* $h : G \to H$ between two graphs G and H wrt. the same signature $\Sigma = (S,O)$ consists of a family of vertex mappings $h = (h_s : G_s \to H_s)_{s \in S}$ and a family of edge mappings $h^O = \left(h^O_f : f^G \to f^H\right)_{f \in O}$ such that, for all operations $f \in O_{w,v}$ and for all edges $e \in f^G$, we have:[5]

$$d^H_f\left(h^O_f(e)\right) = h^w\left(d^G_f(e)\right) \text{ and } c^H_f\left(h^O_f(e)\right) = h^v\left(c^G_f(e)\right).$$

[3] Note that we generalise the usual notion of signature which allows single sorts as co-domain specification for operations only. Operations in $O_{w,\varepsilon}$ will be interpreted as predicates, operations in $O_{w,v}$ with $|v| > 1$ will be interpreted as operations which deliver several results simultaneously.

[4] For $w \in S^*$ and a family $(G_s)_{s \in S}$ of sets, G^w is recursively defined by (i) $G^\varepsilon = \{*\}$, (ii) $G^w = G_s$ if $w = s \in S$ and (iii) $G^w = G^v \times G^u$ if $w = vu$.

[5] Given a sort-indexed family of mappings $(f_s : G_s \to H_s)_{s \in S}$, $f^w : G^w \to H^w$ is recursively defined for every $w \in S^*$ by (i) $f^\varepsilon = \{(*,*)\}$, (ii) $f^w = f_s$ if $w = s \in S$, and (iii) $f^w = f^v \times f^u$, if $w = vu$.

The category of all graphs and graph morphisms wrt. a signature Σ is denoted by $\mathscr{G}_\Sigma$ in the following.[6] $\mathscr{G}_\Sigma$ is complete and co-complete. All limits and co-limits can be constructed component-wise on the underlying sets. The pullback for a co-span $B \xrightarrow{m} A \xleftarrow{n} C$ is given by the *partial product* $B \times_{(m,n)} C$ with the two *projection morphisms* $\pi_B^{B\times C} : B \times_{(m,n)} C \to B$ and $\pi_C^{B\times C} : B \times_{(m,n)} C \to C$:

$$\begin{aligned}
\forall s \in S : \left(B \times_{(m,n)} C\right)_s &= \{(x,y) :: m_s(x) = n_s(y)\} \\
\forall f \in O_{w,v} : f^{\left(B\times_{(m,n)}C\right)} &= \left\{(x,y) :: m_f^O(x) = n_f^O(y)\right\} \\
\forall f \in O_{w,v} : d_f^{\left(B\times_{(m,n)}C\right)} &::= (x,y) \mapsto d_f^B(x) \,||^w\, d_f^C(y) \\
\forall f \in O_{w,v} : c_f^{\left(B\times_{(m,n)}C\right)} &::= (x,y) \mapsto c_f^B(x) \,||^v\, c_f^C(y),
\end{aligned}$$

where the operator $||^w : B^w \times C^w \to (B\times C)^w$ transforms pairs of w-tuples into w-tuples of pairs: $\left(\left(x_1,\ldots,x_{|w|}\right),\left(y_1,\ldots,y_{|w|}\right)\right) \mapsto \left((x_1,y_1),\ldots\left(x_{|w|},y_{|w|}\right)\right)$. A graph $G \in \mathscr{G}_{\Sigma=(S,O)}$ is a *partial algebra*, if it satisfies for all $f \in O$:

$$\forall e_1,e_2 \in f^G : d_f^G(e_1) = d_f^G(e_2) \implies e_1 = e_2. \tag{1}$$

The full sub-category of $\mathscr{G}_\Sigma$ consisting of all partial algebras[7] is denoted by $\mathscr{A}_\Sigma$ in the following. In a partial algebra A, operations $f \in O_{\varepsilon,v}$ with $|v| > 0$ are interpreted as (partial) *constants*, i. e. $f^A : A^\varepsilon \to A^v$ is a partial map from the standard one-element set $A^\varepsilon = \{*\}$ into A^v. Symmetrically, operations $p \in O_{w,\varepsilon}$ with $|w| > 0$ are interpreted as *predicates*, since $p^A : A^w \to \{*\}$ is a partial map into the standard one-element set, i. e. it determines a subset on A^w only, namely the part of A^w where it is defined. Finally for operations $f \in O_{\varepsilon,\varepsilon}$, there are only two possible interpretations in A, namely $f^A = \emptyset$ (false) or $f^A = \{(*,*)\}$ (true). Thus f^A is just a boolean flag.

Due to formulae (1) being a set of Horn-axioms, $\mathscr{A}_\Sigma$ is an epireflective sub-category of $\mathscr{G}_\Sigma$, i. e. there is a reflection that maps a graph $G \in \mathscr{G}_\Sigma$ to a pair $\left(G^{\mathscr{A}} \in \mathscr{A}_\Sigma, \eta_G : G \to G^{\mathscr{A}}\right)$ such that any graph morphism $h : G \to A$ with $A \in \mathscr{A}_\Sigma$ has a unique extension $h^* : G^{\mathscr{A}} \to A$ with $h^* \circ \eta_G = h$. Since epireflective subcategories are closed wrt. products and sub-objects defined by regular monomorphisms (equalisers), the limits in $\mathscr{A}_\Sigma$ coincide with the limits constructed in $\mathscr{G}_\Sigma$. $\mathscr{A}_\Sigma$ has also all co-limits, since epireflections preserve co-limits. In general, however, the co-limits in $\mathscr{A}_\Sigma$ do not coincide with the co-limits constructed in $\mathscr{G}_\Sigma$. The reflection provides the necessary correction. If, for example, $(b : A \to B, c : A \to C)$ is a span in $\mathscr{A}_\Sigma$ and $(c^* : B \to D, b^* : C \to D)$ is its pushout constructed in $\mathscr{G}_\Sigma$, $\left(\eta_D \circ c^* : B \to D^{\mathscr{A}}, \eta_D \circ b^* : C \to D^{\mathscr{A}}\right)$ is the pushout in $\mathscr{A}_\Sigma$.

Besides being complete and co-complete, the most important property of $\mathscr{A}_\Sigma$ for the rest of the paper is the existence of right adjoints to all inverse image functors. If we fix an algebra $A \in \mathscr{A}_\Sigma$, $\mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}A$ denotes the *category of all (weak) sub-algebras* of A. The objects in $\mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}A$

[6] The identities in $\mathscr{G}_\Sigma$ are given by families of identity mappings, and composition of morphisms is provided by component-wise composition of the underlying mappings.

[7] Note that the interpretation of an operation $f \in O_{w,v}$ in a partial algebra A is a partial map: Due to condition (1), the assignment $\left(f^A, d_f^A : f^A \to A^w, c_f^A : f^A \to A^v\right) \mapsto \left\{\left(d_f^A(e), c_f^A(e)\right) :: e \in f^A\right\}$ provides a partial map from A^w to A^v. And, for a partial map $f : A^w \to A^v$, there is the inverse assignment $f \mapsto (f, d_f^A ::= (d,c) \mapsto d, c_f^A ::= (d,c) \mapsto c)$ up to renaming of the elements in f^A.

are all monomorphisms $m : M \rightarrowtail A$ and a morphism in $\mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}A$ from $m : M \rightarrowtail A$ to $n : N \rightarrowtail A$ is a (mono)morphism $h : M \rightarrowtail N$ in $\mathscr{A}_\Sigma$ such that $n \circ h = m$.[8] For an $\mathscr{A}_\Sigma$-morphism $g : A \to B$, the *inverse image functor* $g^* : \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}B \to \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}A$ maps an object $m : M \to B \in \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}B$ to $\pi_A^{A\times M} : A \times_{(g,m)} M \to A$ and a morphism $h : (m : M \to B) \to (n : N \to B)$ to the uniquely determined morphism $g^*(h) : A \times_{(g,m)} M \to A \times_{(g,n)} N$ such that $\pi_A^{A\times N} \circ g^*(h) = \pi_A^{A\times M}$ and $\pi_N^{A\times N} \circ g^*(h) = h \circ \pi_M^{A\times M}$.

Proposition 1 *In a category $\mathscr{A}_\Sigma$ of partial algebras, every inverse image functor $g^* : \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}} B \to \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}A$ has a right adjoint called $g_* : \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}A \to \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}B$.*

Proof. Given a sub-algebra $m : M \rightarrowtail A$, we construct the sub-algebra $g_*(M) \subseteq B$ and the inclusion morphism $g_*(m) : g_*(M) \hookrightarrow B$ as follows:

$$\begin{aligned}
\forall s \in S : g_*(M)_s &= \left\{ b \in B_s :: \forall a \in g_s^{-1}(b)\, \exists x \in M : m_s(x) = a \right\} \text{ and} \\
\forall f \in O_{w,v} : f^{g_*(M)} &= \left\{ e \in f^B :: \forall e_a \in g_f^{O\,-1}(e)\, \exists e_x \in M : m_f^O(e_x) = e_a \right\},
\end{aligned}$$

such that $d_f^{g_*(M)} = d^B_{f\,|f^{g_*(M)}}$ and $c_f^{g_*(M)} = c^B_{f\,|f^{g_*(M)}}$ for every operation.

The co-unit $\varepsilon_m : g^*(g_*(m : M \rightarrowtail A)) \to (m : M \rightarrowtail A)$ can be defined on every element $(a,b) \in A \times_{(g,g_*(m))} g_*(M)$ by $\varepsilon_m(a,b) = c$ such that $m(c) = a$. Note that ε_m is completely defined, since, by definition of $g_*(m)$, a must have a pre-image wrt. m for every pair $(a,b) \in A \times_{(g,g_*(m))} g_*(M)$. It is uniquely defined, since m is monic. By definition of ε_m, $m \circ \varepsilon_m = g^*(g_*(m)) = \pi_A^{A\times_{(g,g_*(m))}g_*(M)}$ which means that ε_m is a morphism in $\mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}A$.

Now, let an object $x : X \rightarrowtail B \in \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}B$ and a morphism $k : g^*(x) \to m \in \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}A$, i. e. $m \circ k = \pi_A^{A\times_{(g,x)}X}$ be given. We construct $k^* : x \to g_*(m)$ by $e \mapsto x(e)$ for every $e \in X$. The mappings of k^* are completely defined: (i) if $x(e) \notin g(A)$, $x(e) \in g_*(M)$ because $g^{-1}(x(e)) = \emptyset$, and, otherwise, the existence of k with $m \circ k = \pi_A^{A\times_{(g,x)}X}$ enforces that every g-pre-image of $x(e)$ has a pre-image under m. By definition, $g_*(m) \circ k^* = x$. By definition of the inverse image functor, $g^*(k^*) : \left(A \times_{(g,x)} X\right) \to \left(A \times_{(g,g_*(m))} g_*(M)\right)$ maps (a,e) to $(a,k^*(e))$. Thus, $\varepsilon\left(g_*(k^*)(a,e)\right) = \varepsilon(a,k^*(e)) = c$ with $m(c) = a$ and $k(a,e) = c'$ with $m(c') = \pi_A^{A\times_{(g,x)}X}(a,e) = a$. Since m is monic, $c = c'$. The morphism k^* is uniquely determined, since $g_*(M) \subseteq B$ and $g_*(m)$ is monic. $\square$

Corollary 1 *If g is monic, the co-units of g_* are isomorphisms.*

For every algebra A, the category $\mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}A$ of all (weak) sub-algebras of A has two important sub-categories, namely $\mathscr{A}_\Sigma{\downarrow}^{\mathrm{F}}A$ and $\mathscr{A}_\Sigma{\downarrow}^{\mathrm{C}}A$, i. e. the category of all full respectively closed sub-algebras of A. A morphism $m : M \to A$ is *full*, if for every $e \in f^A$, $f \in O_{w,v}$, and $w, v \in S^*$ with $d_f^A(e) = m^w(x)$ and $c_f^A(e) = m^v(y)$ for some $(x,y) \in M^w \times M^v$ there is $e' \in f^M$ with $e = m(e')$; it is *closed*, if for every $e \in f^A$, $f \in O_{w,v}$, and $w, v \in S^*$ with $d_f^A(e) = m^w(x)$ for $x \in M^w$ there is $e' \in f^M$ with $e = m(e')$.[9] By definition, $\mathscr{A}_\Sigma{\downarrow}^{\mathrm{C}}A \subseteq \mathscr{A}_\Sigma{\downarrow}^{\mathrm{F}}A \subseteq \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}}A$ for every

[8] Note that every sub-graph of A is a sub-algebra of A, since all monomorphisms in $\mathscr{G}_\Sigma$ are regular and epireflective sub-categories are closed wrt. regular sub-objects.

[9] Note that, in the case of total algebras, every inclusion morphism is closed!

$A \in \mathscr{A}_\Sigma$. The three different sorts of sub-algebras induce three factorisation systems in every category of partial algebras, namely factorisations in (i) closed epimorphisms[10] and arbitrary monomorphisms, (ii) morphisms with surjective vertex mappings and full monomorphisms, and (iii) arbitrary epimorphisms and closed monomorphisms.

Since pullbacks preserve full and closed monomorphisms[11], we obtain restrictions $g^*_{\mathrm{F}} : \mathscr{A}_\Sigma{\downarrow}^{\mathrm{F}} B \to \mathscr{A}_\Sigma{\downarrow}^{\mathrm{F}} A$ and $g^*_{\mathrm{C}} : \mathscr{A}_\Sigma{\downarrow}^{\mathrm{C}} B \to \mathscr{A}_\Sigma{\downarrow}^{\mathrm{C}} A$ of the inverse image functor $g^* : \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}} B \to \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}} A$ for every morphism $g : A \to B$.

Proposition 2 *In a category $\mathscr{A}_\Sigma$ of partial algebras, every inverse image functor $g^*_{\mathrm{F}} : \mathscr{A}_\Sigma{\downarrow}^{\mathrm{F}} B \to \mathscr{A}_\Sigma{\downarrow}^{\mathrm{F}} A$ has a right adjoint called $g^{\mathrm{F}}_* : \mathscr{A}_\Sigma{\downarrow}^{\mathrm{F}} A \to \mathscr{A}_\Sigma{\downarrow}^{\mathrm{F}} B$. If g is monic, the co-units of g^{F}_* are isomorphisms.*

Proof. We show that $g_* : \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}} A \to \mathscr{A}_\Sigma{\downarrow}^{\mathrm{M}} B$ maps full monomorphisms to full monomorphisms. Let $m : M \rightarrowtail A$ be full, $e \in f^B$ with $f \in O_{w,v}$, and $w, v \in S^*$ such that (i) $d^B_f(e) = g_*(m)^w(x)$ and (ii) $c^B_f(e) = g_*(m)^v(y)$. If $g(e') = e$, properties (i) and (ii) together with the construction of g_* in the proof of Proposition 1 imply $d^A_f(e') = m^w(x')$ and $c^A_f(e') = m^v(y')$ for some $(x', y') \in M^w \times M^v$. Since m is full, $e' = m^O_f(e'')$. Thus, every e' with $g(e') = e$ has a pre-image under m^O_f which implies that e has a pre-image under $g_*(m)^O_f$. □

The inverse image functors between closed sub-algebras do not possess right adjoints in general as the following examples illustrate:

Example 1 *(Missing right adjoint) Consider the signature $\Sigma^2 = (S_2, O^2)$ with*

$$\begin{aligned}
S_2 &= \{s\} \text{ and } O^2_{w,v} = \begin{cases} \{f\} & w = ss, v = s \\ \emptyset & \text{otherwise} \end{cases}, \text{ the three algebras} \\
A &::= A_s = \{2\},\ f^A = \emptyset, \\
B &::= B_s = \{0,1,2\},\ f^B = \left(\{f^B\},\ d^B_f(f^B) = (0,1),\ c^B_f(f^B) = 2\right), \\
M &::= M_s = \emptyset,\ f^M = \emptyset,
\end{aligned}$$

the morphism $g : A \to B ::= 2 \mapsto 2$, and the closed monomorphism $m : M \rightarrowtail A ::= \emptyset$. There are seven closed sub-algebras of B, namely B itself and six algebras $\subseteq^x : B^x \to B$, $x \in \{1,2,3,4,5,6\}$ with completely undefined f and carriers $B^1_s = \emptyset$, $B^2_s = \{0\}$, $B^3_s = \{1\}$, $B^4_s = \{2\}$, $B^5_s = \{0,2\}$, and $B^6_s = \{1,2\}$ for sort s.[12] *None of these algebras can be the right adjoint object for m: B itself and cases $x \in \{4,5,6\}$ are excluded since there is no object in M where the object $(2,2)$ in the partial product $A \times_{(g,\subseteq^x)} B^x$ can be mapped to. The cases $x \in \{1,2,3\}$ provide an empty partial product of $\subseteq^x$ with g. Thus, there is the empty morphism from $\pi_A : A \times_{(g,\subseteq^x)} B^x \to A$ to m which can play the role of the co-unit. But there is no morphism from $\subseteq^2$ to $\subseteq^1$ or $\subseteq^3$ and there is no morphisms from $\subseteq^3$ to $\subseteq^2$. Thus, having $O_{w,v} \neq \emptyset$ for $|w| \geq 2$ and $|v| \geq 1$ in the signature prevents the existence of right adjoints in some cases.*

[10] A morphism $h : A \to B$ is closed, if its image in B is a closed sub-algebra of B.
[11] Pullbacks preserve all monomorphisms in a class $\mathscr{N}$, if the underlying category has an epi-mono-factorisation system $(\mathscr{E}, \mathscr{N})$.
[12] Note that the carrier $\{0,1\}$ for s is not closed!

Also constants, i. e. operations in $O_{\varepsilon,v}$ with $|v| \geq 1$, are harmful. A simple example provides the signature $\Sigma^0 = \left(S_0, O^0\right)$ with

$$S_0 \;=\; \{s\} \text{ and } O^0_{w,v} = \begin{cases} \{f\} & w = \varepsilon, v = s \\ \emptyset & \text{otherwise} \end{cases} \text{, the three algebras}$$

$$\begin{aligned} A &::= A_s = \{f, 0\},\, f^A = \left(\{f^A\},\, d^A_f(f^A) = *,\, c^A_f(f^A) = f\right) \\ B &::= B_s = \{f\},\, f^B = \left(\{f^B\},\, d^B_f(f^B) = *,\, c^B_f(f^B) = f\right) \\ M &::= M_s = \{f,\},\, f^M = \left(\{f^M\},\, d^M_f(f^M) = *,\, c^M_f(f^M) = f\right) \end{aligned}$$

the morphism $g : A \to B ::= f \mapsto f, 0 \mapsto f$, and the closed monomorphism $m : M \rightarrowtail A ::= f \mapsto f$. There is only one closed sub-algebra of B, namely B itself. But id_B cannot be the right adjoint object for m, since there is no map $\varepsilon : \left(\pi_A : A \times_{(g,\mathrm{id}_B)} B\right) \to m$. The problematic object in $A \times_{(g,\mathrm{id}_B)} B$ is $(0, f)$ the A-projection of which provides 0 which has no pre-image under m. □

Definition 1 (Graph Structure) A *graph structure* $\Gamma = (S, (O_{w,v})_{w,v \in S^*})$ is a signature with unary operations only, i. e. $O_{w,v} = \emptyset$ if $v \neq \varepsilon$ and $|w| \neq 1$.[13]

Proposition 3 *For a graph structure Γ, the inclusion functor $\subseteq : \mathscr{A}_\Gamma{\downarrow}^{\mathrm{C}}A \to \mathscr{A}_\Gamma{\downarrow}^{\mathrm{F}}A$ has a right adjoint.*

Proof. (Sketch) Given a graph structure $\Gamma = (S, O)$ and a full sub-algebra $(m : M \rightarrowtail A) \in \mathscr{A}_\Gamma$, construct the following family of carriers:

$$\left(M'_s = \{x \in M_s \;::\; f^A(x) \in {M'}^{\,v} \text{ for all } v \in S^* \text{ and } f \in O_{s,v}\}\right)_{s \in S}.$$

Let M' be the largest full sub-algebra of A contained in $(M'_s)_{s \in S}$. Since $M' \subseteq M$, we can use this inclusion as the required co-unit. It is easy to see, that M' is a closed sub-algebra of A. And, by construction, any other closed sub-algebra of A the carriers of which are contained in M is a sub-algebra of M'. □

Corollary 2 *Given a graph structure Γ, every inverse image functor $g^*_{\mathrm{C}} : \mathscr{A}_\Gamma{\downarrow}^{\mathrm{C}}B \to \mathscr{A}_\Gamma{\downarrow}^{\mathrm{C}}A$ has a right adjoint called $g^{\mathrm{C}}_* : \mathscr{A}_\Gamma{\downarrow}^{\mathrm{C}}A \to \mathscr{A}_\Gamma{\downarrow}^{\mathrm{C}}B$. If g is monic, the co-units of g^{C}_* are isomorphisms.*

3 Partial Morphisms on Partial Algebras

In order to obtain frameworks for single-pushout rewriting, we proceed from the category of partial algebras with *total* morphisms to the categories of partial algebras and *partial* morphisms. In this section, we investigate the conditions under which pushouts can be constructed in these categories.

[13] Note that this notion of graph structure is a straightforward extension of the one in [12] to signatures with predicate symbols and operations with products as co-domain!

3.1 Pushouts in Arbitrary Categories of Partial Morphisms

Let $\mathscr{C}$ be a category and $\mathscr{S}$ a sub-class of its monomorphisms such that

(C1) $\mathscr{C}$ has a factorisation system $(\mathscr{E},\mathscr{S})$ for a class of (epi)morphisms $\mathscr{E}$,

(C2) $\mathscr{S}$ contains all isomorphisms and is closed wrt. composition and prefix[14],

(C3) $\mathscr{C}$ has pullbacks along $\mathscr{S}$-morphisms which preserve $\mathscr{S}$-morphisms[15].

A *concrete $\mathscr{S}$-partial morphism* is a span of $\mathscr{C}$-morphisms $(p : K \rightarrowtail P, q : K \to Q)$ such that $p \in \mathscr{S}$. Two concrete $\mathscr{S}$-partial morphisms (p_1,q_1) and (p_2,q_2) are equivalent and denote the same *abstract $\mathscr{S}$-partial morphism* if there is an isomorphism i such that $p_1 \circ i = p_2$ and $q_1 \circ i = q_2$; in this case we write $(p_1,q_1) \equiv (p_2,q_2)$ and $[(p,q)]_\equiv$ for the class of spans that are equivalent to (p,q). The *category of $\mathscr{S}$-partial morphisms $\mathscr{C}^\mathscr{S}$* over $\mathscr{C}$ has the same objects as $\mathscr{C}$ and abstract $\mathscr{S}$-partial morphisms as arrows. The identity $\mathrm{id}_A^{\mathscr{C}^\mathscr{S}}$ for an object A is defined by $\mathrm{id}_A^{\mathscr{C}^\mathscr{S}} = [(\mathrm{id}_A,\mathrm{id}_A)]_\equiv$ and composition of $\mathscr{S}$-partial morphisms $[(p : K \rightarrowtail P, q : K \to Q)]_\equiv$ and $[(r : J \rightarrowtail Q, s : J \to R)]_\equiv$ is given by

$$[(r,s)]_\equiv \circ_{\mathscr{C}^\mathscr{S}} [(p,q)]_\equiv = \left[(p \circ_\mathscr{C} r' : M \rightarrowtail P, s \circ_\mathscr{C} q' : M \to R)\right]_\equiv$$

where $(M, r' : M \rightarrowtail K, q' : M \to J)$ is an arbitrarily chosen pullback of q and r.

Note that there is the faithful embedding functor $\iota : \mathscr{C} \to \mathscr{C}^\mathscr{S}$ defined by identity on objects and $(f : A \to B) \mapsto [\mathrm{id}_A : A \rightarrowtail A, f : A \to B]_\equiv$ on morphisms. We call $[d : A' \rightarrowtail A, f : A' \to B]_\equiv$ a *total* morphism, if d is an isomorphism, and, by a slight abuse of notation, write $[d,f] \in \mathscr{C}$. From now on, we mean the abstract $\mathscr{S}$-partial morphism $[f,g]_\equiv$ if we write $(f : B \rightarrowtail A, g : B \to C)$.

The single-pushout approach defines direct derivations by a single pushout in a category of partial morphisms. There is a general result for the existence of pushouts in a category $\mathscr{C}^\mathscr{S}$ of partial morphisms based on the notions *final $\mathscr{S}$-triple* and *$\mathscr{S}$-hereditary pushout* in the underlying category $\mathscr{C}$ of total morphisms.

Definition 2 (Final $\mathscr{S}$-triple) A *$\mathscr{S}$-triple* for a pair $((l,r),(p,q))$ of morphisms in $\mathscr{C}^\mathscr{S}$ with common domain is given by a collection $\left(\overline{p},p^*,\overline{r},\overline{l},l^*,\overline{q}\right)$ of $\mathscr{C}$-morphisms such that p^*, $\overline{p}$, l^*, $\overline{l}$ are in $\mathscr{S}$ and *(i)* $(\overline{r},\overline{p})$ is pullback of (r,p^*), *(ii)* $(\overline{q},\overline{l})$ is pullback of (q,l^*), and *(iii)* $l \circ \overline{p} = p \circ \overline{l}$. A $\mathscr{S}$-triple $\left(\overline{p},p^*,\overline{r},\overline{l},l^*,\overline{q}\right)$ for $((l,r),(p,q))$ is *final*, if, for any other $\mathscr{S}$-triple $\left(p',p'^*,r',l',l'^*,q'\right)$, there is a unique collection (u_1,u_2,u_3) of ($\mathscr{S}$-)morphisms such that *(iv)* $\overline{p} \circ u_1 = p'$, *(v)* $\overline{l} \circ u_1 = l'$, *(vi)* $p^* \circ u_2 = p'^*$, *(vii)* $u_2 \circ r' = \overline{r} \circ u_1$, *(viii)* $l^* \circ u_3 = l'^*$, and *(ix)* $u_3 \circ q' = \overline{q} \circ u_1$, compare left part of Fig. 1.

In [13], a sufficient condition for the existence of final $\mathscr{S}$-triples is shown.

Proposition 4 *A category $\mathscr{C}^\mathscr{S}$ of partial morphisms has all final $\mathscr{S}$-triples, if* (i) *the inverse image functor $g^*_\mathscr{S} : \mathscr{A}_\Sigma{\downarrow}^\mathscr{S} B \to \mathscr{A}_\Sigma{\downarrow}^\mathscr{S} A$ has a right adjoint for each morphism $g : A \to B$ and* (ii) *every $\mathscr{S}$-chain $(m_i : A_{i+1} \rightarrowtail A_i)_{i \in \mathbb{N}_0}$ has a limit $(l_i : A^* \rightarrowtail A_i)_{i \in \mathbb{N}_0}$.*[16]

[14] Closure of $\mathscr{S}$ wrt. composition and prefix means that, given $g \in \mathscr{S}$, $g \circ f \in \mathscr{S} \Leftrightarrow f \in \mathscr{S}$.

[15] I. e. $g' \in \mathscr{S}$, if (f',g') is pullback in $f \circ g' = g \circ f'$ and $g \in \mathscr{S}$.

[16] That the chain morphisms are in $\mathscr{S}$ is implied by the existence of $(\mathscr{E},\mathscr{S})$-factorisations.

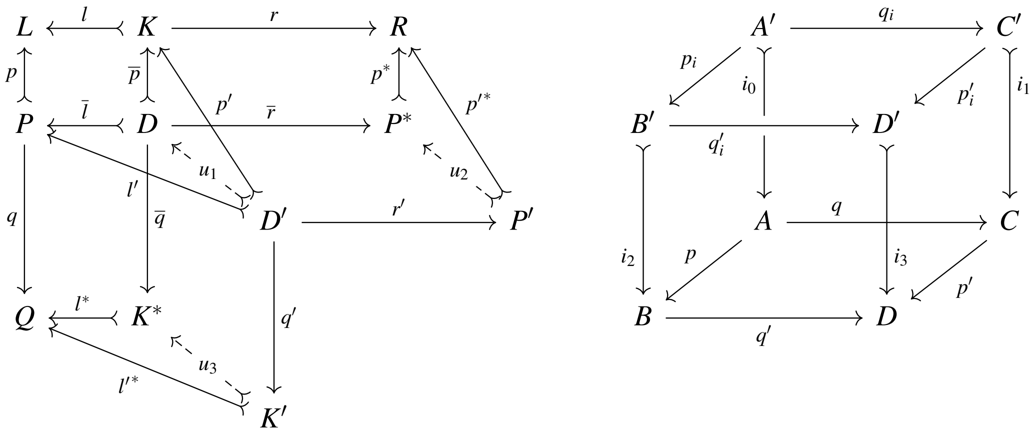

Figure 1: Final Triple and Hereditary Pushout

Definition 3 ($\mathscr{S}$-Hereditary pushout) A pushout (q',p') of (p,q) in $\mathscr{C}$ is *$\mathscr{S}$-hereditary* if for each commutative cube as in the right part of Figure 1, which has pullback squares (p_i,i_0) and (q_i,i_0) of (i_2,p) and (i_1,q) resp. as back faces such that i_1 and i_2 are in $\mathscr{S}$, in the top square, (q_i',p_i') is pushout of (p_i,q_i), if and only if, in the front faces, (p_i',i_1) and (q_i',i_2) are pullbacks of (i_3,p') and (i_3,q') resp. and i_3 is in $\mathscr{S}$.[17]

Note that (q_i',p_i') is hereditary pushout of (p_i,q_i) in Figure 1, if (q',p') is hereditary pushout of (p,q).

Proposition 5 (Pushout in $\mathscr{C}^{\mathscr{S}}$) *A given span of partial morphisms $(l:K\rightarrowtail L, r:K\to R)$ and $(p:P\rightarrowtail L, q:P\to Q)$ has a pushout $((l^*,r^*),(p^*,q^*))$ in $\mathscr{C}^{\mathscr{S}}$, if and only if there is* (i) *a final $\mathscr{S}$*-triple *$\overline{l}:D\to P$, $\overline{p}:D\to K$, $\overline{r}:D\to P^*$, $\overline{q}:D\to K^*$, $p^*:P^*\to R$, $l^*:K^*\to Q$ for $((l,r),(p,q))$ and* (ii) *a $\mathscr{S}$*-hereditary pushout *$(r^*:K^*\to H, q^*:P^*\to H)$ for $(\overline{q},\overline{r})$ in $\mathscr{C}$, compare sub-diagrams (1) – (3) and (4) resp. in Figure 2.*

$$\begin{tikzcd}
L & K \arrow[l, tail, "l"'] \arrow[r, "r"] & R \\
P \arrow[u, tail, "p"] \arrow[ur, phantom, "(1)"] & D \arrow[l, tail, "\overline{l}"'] \arrow[u, tail, "\overline{p}"'] \arrow[r, "\overline{r}"] \arrow[ur, phantom, "(2)"] & P^* \arrow[u, tail, "p^*"'] \\
Q & K^* \arrow[l, tail, "l^*"] \arrow[r, "r^*"'] \arrow[ul, phantom, "(3)"] & H \arrow[ul, phantom, "(4)"] \\
\arrow[from=2-1, to=3-1, "q"'] \arrow[from=2-2, to=3-2, "\overline{q}"] \arrow[from=2-3, to=3-3, "q^*"]
\end{tikzcd}$$

Figure 2: Pushout in $\mathscr{C}^{\mathscr{S}}$

The proof can be found in [16]. A version of the proof that does not presuppose a choice of pullbacks that is compatible with pullback composition and decomposition is contained in [17].

[17] For details on hereditary pushouts see [10, 11]

3.2 Pushouts of Partial Morphisms on Partial Algebras

With the results of Section 2, we can construct categories with three different types of partial morphisms over partial algebras, namely $\mathscr{A}_\Sigma^{\mathrm{M}}$, $\mathscr{A}_\Sigma^{\mathrm{F}}$, and $\mathscr{A}_\Gamma^{\mathrm{C}}$ where M is the class of all, F the class of all full, C the class of all closed monomorphisms, Σ is an arbitrary signature, and Γ is a graph structure. The general results about pushouts of partial morphisms carry over to these categories as follows:

Proposition 6 (Final triple) *In $\mathscr{A}_\Sigma^{\mathrm{M}}$, $\mathscr{A}_\Sigma^{\mathrm{F}}$, and $\mathscr{A}_\Gamma^{\mathrm{C}}$, every pair $((l,r),(p,q))$ of partial morphisms with common domain has a final triple.*

Proof. We use the condition of Proposition 4. The existence of right adjoints to inverse image functors is given by Propositions 1 and 2 as well as Corollary 2. Limits of $\mathscr{S}$-chains exist, since $\mathscr{A}_\Sigma$ has all limits. □

Corollary 3 (Pushout) *Partial morphisms $(l : K \rightarrowtail L, r : K \to R)$ and $(p : P \rightarrowtail L, q : P \to Q)$ have a pushout in $\mathscr{A}_\Sigma^{\mathrm{M}}$, $\mathscr{A}_\Sigma^{\mathrm{F}}$, and $\mathscr{A}_\Gamma^{\mathrm{C}}$, if and only if the pushout of $(\overline{q},\overline{r})$ is* M-, F-, *or* C-*hereditary respectively, where $\overline{l} : D \to P$, $\overline{p} : D \to K$, $\overline{r} : D \to P^*$, $\overline{q} : D \to K^*$, $p^* : P^* \to R$, $l^* : K^* \to Q$ is a final* M-, F-, *or* C-*triple of $((l,r),(p,q))$, see Figure 2.*

Proof. Direct consequence of Proposition 5 and Proposition 6. □

Thus, the existence of pushouts in $\mathscr{A}_\Sigma^{\mathrm{M}}$, $\mathscr{A}_\Sigma^{\mathrm{F}}$, or $\mathscr{A}_\Gamma^{\mathrm{C}}$ only depends on the involved $\mathscr{A}_\Sigma$-pushout being M-, F-, or C-hereditary, resp. We start with the analysis of M-hereditary pushouts in $\mathscr{A}_\Sigma$.

Proposition 7 (Sufficient condition for M-hereditariness) *If a pushout in $\mathscr{A}_\Sigma$ is also pushout in $\mathscr{G}_\Sigma$, then it is* M-*hereditary in $\mathscr{A}_\Sigma$.*

Proof. Let a commutative cube as in the right part of Fig. 1 in $\mathscr{A}_\Sigma$ be given such that the back faces are pullbacks. Then this is also a situation in $\mathscr{G}_\Sigma$ and the back faces are also pullbacks in $\mathscr{G}_\Sigma$, since $\mathscr{A}_\Sigma$ is an epireflection of $\mathscr{G}_\Sigma$.

Let the front faces be pullbacks in $\mathscr{A}_\Sigma$ and i_3 be a monomorphism. Then the front faces are also pullbacks in $\mathscr{G}_\Sigma$. Since all pushouts in $\mathscr{G}_\Sigma$ are hereditary, D' together with p'_i and q'_i is pushout in $\mathscr{G}_\Sigma$. Since (i) $\mathscr{A}_\Sigma$ is closed wrt. sub-algebras, (ii) D is in $\mathscr{A}_\Sigma$, and (iii) i_3 is monic, D' is also in $\mathscr{A}_\Sigma$ and its reflector $\eta_{D'}$ is an isomorphism.[18] Thus, D' together with p'_i and q'_i is pushout in $\mathscr{A}_\Sigma$.

Let (D',q'_i,p'_i) be pushout of (p_i,q_i) in $\mathscr{A}_\Sigma$. Construct (D'',q''_i,p''_i) as pushout of (p_i,q_i) in $\mathscr{G}_\Sigma$. We obtain the epic reflector $\eta_{D''} : D'' \twoheadrightarrow D'$ with $p'_i = \eta_{D''} \circ p''_i$ and $q'_i = \eta_{D''} \circ q''_i$. Since D'' is pushout, we also get $i'_3 : D'' \rightarrowtail D$ with $i'_3 \circ p''_i = p' \circ i_1$ and $i'_3 \circ q''_i = q' \circ i_2$. Since $i_3 \circ \eta_{D''} \circ p''_i = i_3 \circ p'_i = p' \circ i_1 = i'_3 \circ p''_i$ and $i_3 \circ \eta_{D''} \circ q''_i = i_3 \circ q'_i = q' \circ i_2 = i'_3 \circ q''_i$, we can conclude $i_3 \circ \eta_{D''} = i'_3$. Since all pushouts in $\mathscr{G}_\Sigma$ are hereditary, i'_3 is monic implying that $\eta_{D''}$ is monic as well. Thus, $\eta_{D''}$ is an isomorphism and D' is also the pushout in $\mathscr{G}_\Sigma$. This immediately provides monic i_3 and pullbacks in the front faces of the cube in the right part of Fig. 1. □

But not all pushouts in $\mathscr{A}_\Sigma$ are M-hereditary. Here is a typical example:

[18] Note that morphisms in $\mathscr{G}_\Sigma$ that are both epic and monic are isomorphisms.

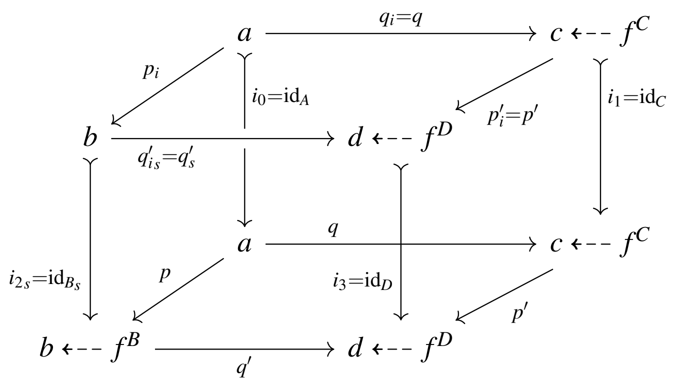

Figure 3: Simple Non-Hereditary Pushout in $\mathscr{A}_\Sigma$

Example 2 *Consider the signature* $\Sigma^c = (S_c, O^c)$ *with*

$$S_c \quad = \quad \{s\} \text{ and } O^c_{w,v} = \begin{cases} \{f\} & w = \varepsilon, v = s \\ \emptyset & \text{otherwise,} \end{cases}$$

the three algebras

$$\begin{aligned} A \quad &::= \quad A_s = \{a\},\, f^A = \emptyset, \\ B \quad &::= \quad B_s = \{b\},\, f^B = \left(\{f^B\},\, d^B_f(f^B) = *,\, c^B_f(f^B) = b\right), \\ C \quad &::= \quad C_s = \{c\},\, f^C = \left(\{f^C\},\, d^C_f(f^C) = *,\, c^C_f(f^C) = c\right), \end{aligned}$$

and the two morphisms $p : A \to B ::= a \mapsto b$ *and* $q : A \to C ::= a \mapsto c$*. The pushout of* (p,q) *in* $\mathscr{A}^{\mathrm{P}}_{\Sigma^c}$ *consists of the algebra*

$$D ::= D_s = \{d\},\, f^D = \left(\{f^D\},\, d^D_f(f^D) = *,\, c^D_f(f^D) = d\right)$$

and the two morphisms

$$\begin{aligned} p' : C \to D \quad &::= \quad c \mapsto d, f^C \mapsto f^D \\ q' : B \to D \quad &::= \quad b \mapsto d, f^B \mapsto f^D. \end{aligned}$$

This pushout is depicted in the bottom of Fig. 3 and is not M*-hereditary. We construct the following cube of morphisms, compare Fig. 3:* $A' = A$, $i_0 = \mathrm{id}_A$, B' *is defined by* $B'_s = B_s$ *and* $f^{B'} = \emptyset$, i_2 *maps* b *in* B'_s *to* b *in* B_s, $C' = C$, $i_1 = \mathrm{id}_C$, $q_i = q$, *and* p_i *maps* a *to* b*. Note that* (i_0, q_i) *is pullback of* (q, i_1) *and* (i_0, p_i) *is pullback of* (p, i_2)*. Constructing* $(D' = D, p'_i = p', q'_i ::= b \mapsto d)$ *as the pushout of* (p', q')*, we obtain* $i_3 = \mathrm{id}_D$*. But* (i_2, q'_i) *is not pullback of* (q', i_3)*:* $B \times_{(q', i_3)} D'$ *contains a defined constant for* f*, since* $i_3(f^D) = q'(f^B)$*, and* B' *does not.* □

Note that the $\mathscr{A}_\Sigma$-pushout of the morphisms p and q in Example 2 does not coincide with the pushout of p and q constructed in the larger category $\mathscr{G}_\Sigma$. The pushout in $\mathscr{G}_\Sigma$ is the graph

$$G \quad ::= \quad G_s = \{g\}, f^G = \left(\{f^G_C, f^G_B\}, d^G_f(f^G_C) = d^G_f(f^G_B) = *, c^G_f(f^G_C) = c^G_f(f^G_B) = g\right)$$

together with the morphisms

$$\begin{array}{rcl} p'' : C \to G & ::= & c \mapsto g, f^C \mapsto f^G_C \\ q'' : B \to G & ::= & b \mapsto g, f^B \mapsto f^G_B. \end{array}$$

The partial algebra D is the epireflection of the graph G and the reflector $\eta_G : G \to D$ maps as follows: $g \mapsto d$, $f^G_C \mapsto f^D$, and $f^G_B \mapsto f^D$. The identification $\eta_G(f^G_C) = \eta_G(f^G_B) = f^D$ of the reflector provided the possibility to construct the commutative cube in Example 2 that disproves M-hereditariness of the pushout of (p,q). The following proposition shows that this construction of a counterexample is always possible if the pushouts in $\mathscr{A}_\Sigma$ and $\mathscr{G}_\Sigma$ are different.

As a prerequisite, we need the following construction.

Definition 4 (Vertex- and hyperedge-generated sub-span) Let a signature $\Sigma = (S,O)$ and an $\mathscr{A}_\Sigma$-span $B \xleftarrow{p} A \xrightarrow{q} C$ be given. For $s \in S$, let $\equiv^{p,q}_s \subseteq (B_s \uplus C_s) \times (B_s \uplus C_s)$ denote the equivalence generated by $\{(p_s(z), q_s(z)) :: z \in A_s\}$ and, for $f \in O$, let $\equiv^{p,q}_f \subseteq \left(f^B \uplus f^C\right) \times \left(f^B \uplus f^C\right)$ denote the equivalence generated by $\{(p^O_f(z), q^O_f(z)) :: z \in f^A\}$.

For a vertex $e \in B_s$, the sub-algebras $B^e \subseteq B$, $C^e \subseteq C$, and $A^e \subseteq A$ are defined by $B^e_s = B_s \cap [e]_{\equiv^{p,q}_s}$, $C^e_s = C_s \cap [e]_{\equiv^{p,q}_s}$, $A^e_s = \{a \in A_s :: p_s(a) \in [e]_{\equiv^{p,q}_s}\}$, $B^e_{s'} = C^e_{s'} = A^e_{s'} = \emptyset$ for all $s' \neq s$, and $f^B = f^C = f^A = \emptyset$ for all $f \in O$.

For a hyperedge $e \in f^B$ with $d^B_f(e) = (d_1, \ldots, d_n)$ and $c^B_f(e) = (c_1, \ldots, c_m)$, the sub-algebras $B^e \subseteq B$, $C^e \subseteq C$, and $A^e \subseteq A$ are defined by $B^e_s = \bigcup_{x \in e^\leftrightarrow} B^x_s$, $C^e_s = \bigcup_{x \in e^\leftrightarrow} C^x_s$, $A^e_s = \bigcup_{x \in e^\leftrightarrow} A^x_s$, $f^{B^e} = f^B \cap [e]_{\equiv^{p,q}_f}$, $f^{C^e} = f^C \cap [e]_{\equiv^{p,q}_f}$, $f^{A^e} = \{n \in f^A :: p^O_f(n) \in [e]_{\equiv^{p,q}_f}\}$, and $f'^{B^e} = f'^{C^e} = f'^{A^e} = \emptyset$ for all $f' \neq f \in O$ where $e^\leftrightarrow = \{d_1, \ldots, d_n, c_1, \ldots, c_m\}$.

Proposition 8 (Necessary condition for M-hereditariness) *If a pushout in $\mathscr{A}_\Sigma$ is* M*-hereditary, it is also pushout in $\mathscr{G}_\Sigma$.*

Proof. Let $(p : A \to B, q : A \to C)$ be a span of morphisms in $\mathscr{A}_\Sigma$, let $(E, q'' : B \to E, p'' : C \to E)$ be its pushout in $\mathscr{G}_\Sigma$, and let $(q' : B \to D, p' : C \to D)$ be its pushout in $\mathscr{A}_\Sigma$. Since $\mathscr{A}_\Sigma$ is epireflective sub-category of $\mathscr{G}_\Sigma$, we know that $D = E^{\mathscr{A}}$, $q' = \eta_E \circ q''$ and $p' = \eta_E \circ p''$ where $\eta_E : E \to E^{\mathscr{A}}$ is the reflector for the graph E. Suppose D and E are not isomorphic, then there are $x, y \in E$ such that $x \neq y$ and $\eta_E(x) = \eta_E(y) = z$. Since p'' and q'' are jointly epic in $\mathscr{G}_\Sigma$, both x and y have pre-images under p'' and/or q''. Let $x'', y'' \in B \uplus C$ be these pre-images and suppose, without loss of generality, $x'' \in B$.

We construct a cube as in Fig. 1(right part) using the construction of Definition 4 for x'': Let $B' = B^{x''}$, $C' = C^{x''}$, and $A' = A^{x''}$, $i_0 : A' \rightarrowtail A$, $i_1 : C' \rightarrowtail C$, and $i_2 : B' \rightarrowtail B$ be the sub-algebra inclusions, and let $p' = p_{|A'}$ and $q' = q_{|A'}$. By construction, (p_i, i_0) and (q_i, i_0) are pullbacks. Since $x \neq y$, we have $[x'']_{\equiv^{p,q}} \neq [y'']_{\equiv^{p,q}}$ and can conclude that y'' has neither a pre-image under i_1 nor under i_2. Construct the $\mathscr{A}_\Sigma$-pushout $(p'_i : C' \to D', q'_i : B' \to D')$ of (q', p') which provides the morphisms $i'_3 : D' \to E$ and $i_3 = \eta_E \circ i'_3 : D' \to D$ making the whole cube commute. By construction, $z = \eta_E(x) = \eta_E \circ q''(x'') = \eta_E \circ q'' \circ i_2(x'') = \eta_E \circ i'_3 \circ q'_i(x'') = i_3 \circ q'_i(x'')$. Thus, z has a pre-image under i_3 and $z = q'(y'')$ or $z = p'(y'')$. Since y'' has neither a pre-image under i_1 nor under i_2, one of the front faces of the constructed cube fails to be a pullback. $\square$

Theorem 1 *A pushout in $\mathscr{A}_\Sigma$ is* M*-hereditary, iff it is also pushout in $\mathscr{G}_\Sigma$.*

Now, we investigate F-hereditary pushouts in $\mathscr{A}_\Sigma$. In Example 2, the morphism i_2 is not full, compare Figure 3. If monomorphisms are restricted to full ones, the situation in Figure 3 cannot occur. This observation leads to the following characterisation of F-hereditary pushouts in $\mathscr{A}_\Sigma$.

Theorem 2 *A pushout (q', p') of (p,q) in $\mathscr{A}_\Sigma$ is* F*-hereditary, iff (q'_s, p'_s) is pushout of (p_s, q_s) in the category of sets and mappings for all sorts $s \in \Sigma$.*

Proof. "$\Leftarrow$": Let a commutative cube as in the right part of Fig. 1 in $\mathscr{A}_\Sigma$ be given such that the back faces are pullbacks. Let the top be pushout in $\mathscr{A}_\Sigma$. Then, for every sort s, the pushout $\left(E_s, {q''_i}_s : B'_s \to E_s, {p''_i}_s : C'_s \to E_s\right)$ of $\left({p_i}_s, {q_i}_s\right)$ provides epic $u_s : E_s \to D'_s$. Since all pushouts of mappings are hereditary, we obtain monic ${i'_3}_s : E_s \to D_s$ with ${i_3}_s \circ u_s = {i'_3}_s$. Thus, u_s is monic and epic which implies, for all sorts s, $(D'_s, {p'_i}_s, {q'_i}_s)$ is pushout, ${i_3}_s$ is monic, and $\left({p'_i}_s, {i_1}_s\right)$ as well as $\left({q'_i}_s, {i_2}_s\right)$ are pullbacks. That i_3 is full, follows from p' and q' being jointly epic and i_1 and i_2 being full. Given $d \in f^D$ with $d = i_3(d')$ and $d = p'(c)$, there is $c' \in f^{C'}$ with $i_1(c') = c$ and $p'_i(c) = d'$, since i_1 is full. Therefore (p'_i, i_1) is pullback. An analog argument provides that (q'_i, i_2) is pullback. Vice versa, if i_3 is full monomorphism and (p'_i, i_1) as well as (q'_i, i_2) are pullbacks, they are pullbacks for each sort mapping. This implies that $\left({p'_i}_s, {q'_i}_s\right)$ is pushout for every sort s. It remains to show that p'_i and q'_i are jointly epic on hyperedges. But this is a direct consequence of p' and q' being jointly epic and i_1 and i_2 being full.

"$\Rightarrow$": We repeat the argument in the proof of Proposition 8: Let there be a sort s, such that the pushout $(E_s, q''_s : B_s \to E_s, p''_s : C_s \to E_s)$ is different from (D_s, q'_s, p'_s). Then there are $x \neq y$ such that $u_s(x) = z = u_s(y)$ for the unique morphism $u_s : E_s \to D_s$ with $u_s \circ q''_s = q'_s$ and $u_s \circ p''_s = p'_s$. Without loss of generality, assume y has a pre-image y'' under p'' and/or q'' and $x = q''_s(x'')$. Consider again the sub-span of (p,q) induced by x'': Let A', B', and C' be the *full* sub-algebras of A, B, and C induced by $A^{x''}$, $B^{x''}$, and $C^{x''}$ respectively, let $p_i = p_{|A'}$ and $q_i = q_{|A'}$, and let $i_0 : A' \rightarrowtail A$, $i_1 : C' \rightarrowtail C$, and $i_2 : B' \rightarrowtail B$ be the inclusions. We obtain pullbacks (i_0, p') and (i_0, q'). Constructing the pushout (q'_i, p'_i) of (p_i, q_i), z has a pre-image under i_3. We also know $z = q'(y'')$ or $z = p'(y'')$. By construction, y'' has neither a pre-image under i_1 nor under i_2. Thus, either (i_2, q'_i) or (i_1, p'_i) is no pullback. □

The analysis of C-hereditary pushouts in $\mathscr{A}_\Gamma$ for a graph structure Γ is more complicated than the analysis of M- and F-hereditary pushouts as the following example demonstrates.

Example 3 (C-hereditary pushout) *Consider the span $(p : A \to B, q : A \to C)$ in $\mathscr{A}_\Gamma$ where the underlying graph structure $\Gamma^{\mathrm{C}} = \left(S_{\mathrm{C}}, O^{\mathrm{C}}\right)$ is defined by*

$$S_{\mathrm{C}} \quad = \quad \{s\} \text{ and } O^{\mathrm{C}}_{w,v} = \begin{cases} \{f\} & w = s, v = s \\ \emptyset & \text{otherwise,} \end{cases}$$

the three algebras A, B, and C are given by

$$\begin{aligned} A \quad &::= \quad A_s = \{a_{1;1}, a_{1;3}, a_{3;1}\},\ f^A = \emptyset, \\ B \quad &::= \quad B_s = \{b_1, b_2, b_3\},\ f^B ::= b_1 \mapsto b_2,\ b_2 \mapsto b_3, \\ C \quad &::= \quad C_s = \{c_1, c_2, c_3\},\ f^C ::= c_1 \mapsto c_2,\ c_2 \mapsto c_3, \end{aligned}$$

and the two morphisms p and q map as follows:

$$\begin{aligned} p_s : A_s \to B_s &::= a_{1;1} \mapsto b_1, a_{1;3} \mapsto b_1, a_{3;1} \mapsto b_3 \\ q_s : A_s \to C_s &::= a_{1;1} \mapsto c_1, a_{1;3} \mapsto c_3, a_{3;1} \mapsto c_1. \end{aligned}$$

The pushout of (p,q) in $\mathscr{A}_\Gamma$ consists of the algebra

$$D ::= D_s = \{d_{2;2}, d_{1,3;1,3}\},\ f^D ::= d_{2;2} \mapsto d_{1,3;1,3}, d_{1,3;1,3} \mapsto d_{2;2}$$

and the two morphisms

$$\begin{aligned} p' : C \to D &::= c_1 \mapsto d_{1,3;1,3}, c_2 \mapsto d_{2;2}, c_3 \mapsto d_{1,3;1,3} \\ q' : B \to D &::= b_1 \mapsto d_{1,3;1,3}, b_2 \mapsto d_{2;2}, b_3 \mapsto d_{1,3;1,3}. \end{aligned}$$

This pushout in $\mathscr{A}_\Gamma$ does not induce a pushout in the category of sets and maps for the sort s, since $p'_s(c_2) = d_{2;2} = q'_s(b_2)$ without $b_2 \equiv^{p,q} c_2$. Nevertheless, it is C*-hereditary, since the only possible sub-spans of (p,q) in a commutative cube as in Figure 1(right part) with pullbacks in the back faces and closed monomorphisms i_1 and i_2 are $(\emptyset,\emptyset)$ and (p,q) itself. Having, for example c_2 in C' requires also $c_3 \in C'$. This implies $b_1, b_2, b_3 \in B'$ due to $p(a_{1;3}) = b_1$ and $q(a_{1;3}) = c_3$. Now, $p(a_{3;1}) = b_3$ and $q(a_{3;1}) = c_1$ requires $c_1 \in C'$. Thus, having one of the elements of B and C in the sub-span requires having all of the elements in the sub-span. This is due to the cyclic structure established by f^B, f^C, and the mapping of $a_{1;3}$ and $a_{3;1}$ by p and q.* □

Definition 5 (Hierarchical span) Given a graph structure $\Gamma = (S,O)$ and a span $(p : A \to B, q : A \to C)$ in $\mathscr{A}_\Gamma$, its *reachability relation* R on the disjoint union $\biguplus_{s\in S} B_s \uplus \biguplus_{s\in S} C_s$ of the carriers of B and C is defined by $x\mathrm{R}y$ if *(i)* $x \equiv_s^{p,q} y$ for some sort s, *(ii)* $(z_1,\dots,z_n) = f^{B\uplus C}(x)$ for some $f \in O_{s,v}$, $n \geq 1$, and $y = z_i$ for $1 \leq i \leq n$, or *(iii)* $x\mathrm{R}z$ and $z\mathrm{R}y$.[19] A span (p,q) is *hierarchical*, if $x\mathrm{R}y$ and $y\mathrm{R}x$ implies $x \equiv^{p,q} y$. $\mathrm{R}(x)$ denotes the elements reachable from x, i. e. $\mathrm{R}(x) = \{z \in \biguplus_{s\in S} B_s \uplus \biguplus_{s\in S} C_s :: x\mathrm{R}z\}$

Theorem 3 *A pushout $(q' : B \to D, p' : C \to D)$ of a hierarchical span (p,q) in $\mathscr{A}_\Gamma$ is* C*-hereditary, iff (q'_s, p'_s) is pushout of (p_s, q_s) for all sorts $s \in \Sigma$.*

Proof. (Sketch) For the "⇐"-part, we can repeat the arguments in the "⇐"-part of the proof for Theorem 2 substituting fullness by closedness. The "⇒"-part also follows the outline given by the "⇒"-part of the proof for Theorem 2. Having $x \neq y \in E_s$ and $u_s(x) = u_s(y)$ where $(E_s, p''_s : C_s \to E_s, q''_s : B_s \to E_s)$ is the pushout of (q_s, p_s) and $u_s : E_s \to D_s$ the mediating map with $u_s \circ p''_s = p'_s$ and $u_s \circ q''_s = q'_s$, we know for the pre-images x'' of x and y'' of y under p'' and/or q'' that either x'' is not reachable from y'' or y'' is not reachable from x''. Without loss of generality, suppose the second case and construct B' as the full sub-algebra of B induced by $B \cap \mathrm{R}(x'')$, C' as the full sub-algebra of C induced by $C \cap \mathrm{R}(x'')$, and A' as the full sub-algebra of A induced by $\{a \in A :: p(a) \in \mathrm{R}(x'')\}$. Let $i_0 : A' \to A$, $i_1 : C' \to C$, and $i_2 : B' \to B$ be the corresponding closed inclusions and $p_i = p_{|A'}$ and $q_i = q_{|A'}$. Since we know that y'' neither has a pre-image under i_1 nor under i_2, either (i_1, p'_i) or (i_2, q'_i) is not a pullback. □

[19] $f^{B\uplus C} : B_s \uplus C_s \to B^v \uplus C^v$ is the uniquely determined co-product of f^B and f^C.

$$\begin{tikzcd}
L \arrow[d, "m"'] \arrow[dr, phantom, "(1)"] & K \arrow[l, tail, "l"'] \arrow[r, tail, "r"] \arrow[d, "m\langle l\rangle"] \arrow[dr, phantom, "(2)"] & R \arrow[d, "m\langle t\rangle"] \\
G & K^* \arrow[l, tail, "l\langle m\rangle"] \arrow[r, tail, "r\langle m\rangle"'] & t@m
\end{tikzcd}$$

Figure 4: Single- versus Sesqui-Pushout Transformation

Definition 6 (Hierarchical graph structure) A graph structure $\Gamma = (S, O)$ is hierarchical, if the reflexive/transitive closure of the relation $\preceq$ on S defined by $s \preceq s'$ if there is $f \in O_{s,vs'u}$ with $v, u \in S^*$ is a partial order.

Corollary 4 *For a hierarchical graph structure Γ, a pushout (q', p') of a span (p, q) in $\mathscr{A}_\Gamma$ is* C*-hereditary, iff (q'_s, p'_s) is pushout of (p_s, q_s) for all sorts $s \in \Sigma$.*

4 Rewriting of Partial Algebras: Some Theory

In this section, we introduce single-pushout rewriting in the categories of partial algebras. For a rich theory, we restrict transformations to those that produce total co-matches. All definitions and propositions in the following presuppose an arbitrary underlying category $\mathscr{A}_\Sigma^{\mathrm{M}}$, $\mathscr{A}_\Sigma^{\mathrm{F}}$, and $\mathscr{A}_\Gamma^{\mathrm{C}}$ for a signature Σ or hierarchical graph structure Γ.[20]

Definition 7 (Rule and transformation) A *transformation rule* t is a *partial* morphism $t = (l : K \rightarrowtail L, r : K \to R)$. There is a *direct transformation* of a *host graph* G to the *result graph* $t@m$ with a rule $t : L \to R$ if there are *total* morphisms $m : L \to G$ and $m\langle t\rangle : R \to t@m$ as well as a *partial* morphism $t\langle m\rangle = (l\langle m\rangle : K^* \to G, r\langle m\rangle : K^* \to t@m) : G \to t@m$ such that $(t\langle m\rangle, m\langle t\rangle)$ is pushout of (t, m). In a direct transformation, m is called *match*, $m\langle t\rangle$ *co-match*, and $t\langle m\rangle$ *trace* of the transformation.[21]

Since we restricted transformations to total co-matches, we obtain a close connection of our transformations to Sesqui-Pushout Rewritings in the sense of [3], which are composed of final pullback complements and pushouts.

Theorem 4 (Single- and sesqui-pushout transformation) *There is a transformation with rule $t = (l : K \rightarrowtail L, r : K \rightarrowtail R)$, match $m : L \to G$, (total) co-match $m\langle t\rangle : R \to t@m$ and trace $t\langle m\rangle = (l\langle m\rangle : K^* \rightarrowtail G, r\langle m\rangle : K^* \rightarrowtail t@m)$, if and only if there is a total morphism $m\langle l\rangle : K \to K^*$ such that* (i) $(l, m\langle l\rangle)$ is pullback of $(m, l\langle m\rangle)$, (ii) $l\langle m\rangle = m_*(l)$, and (iii) $(r\langle m\rangle, m\langle t\rangle)$ is hereditary pushout of $(r, m\langle l\rangle)$, compare (1) and (2) in Fig. 4.

Proof. Direct consequence of the construction of final triples in [17] and the fact that the match and the co-match are total. □

[20] Note that the theory in this section is also valid in $\mathscr{A}_\Sigma^{\mathrm{C}}$ for arbitrary signatures, although existence of a direct transformation with a rule at a given match is unpredictable in this general case, compare Section 3.

[21] The graph $t@m$ is uniquely determined up to isomorphism by the transformation!

$$
\begin{tikzcd}
L_1 \arrow[d, "e''_1"] \arrow[dd, bend right, "m_1"'] & K_1 \arrow[l, tail, "l_1"'] \arrow[r, "r_1"] \arrow[d, "e'_1"'] \arrow[dd, bend left, "m_1\langle l_1\rangle"] & R_1 \arrow[dr, "e_1"] \arrow[ddr, "m_1\langle t_1\rangle"'] & & L_2 \arrow[dl, "e_2"'] \arrow[ddl, "m_2"] & K_2 \arrow[l, tail, "l_2"'] \arrow[r, "r_2"] \arrow[d, "e'_2"] \arrow[dd, bend right, "m_2\langle l_2\rangle"'] & R_2 \arrow[d, "e''_2"'] \arrow[dd, bend left, "m_2\langle t_2\rangle"] \\
L'_1 \arrow[d, tail, "s'_1"] & K'_1 \arrow[l, tail, "l'_1"'] \arrow[rr, "r'_1"] \arrow[d, tail, "s_1"'] & & M \arrow[d, tail, "s"] & & K'_2 \arrow[ll, tail, "l'_2"'] \arrow[r, "r'_2"] \arrow[d, tail, "s_2"] & R'_2 \arrow[d, tail, "s'_2"'] \\
G & D_1 \arrow[l, tail, "l_1\langle m_1\rangle"] \arrow[rr, "r_1\langle m_1\rangle"'] & & t_1@m_1 & & D_2 \arrow[ll, tail, "l_2\langle m_2\rangle"] \arrow[r, "r_2\langle m_2\rangle"'] & t_2@m_2
\end{tikzcd}
$$

Figure 5: Hereditary Pushout and Final Pullback Complement

Thus, our set-up of single-pushout rewriting *is* Sesqui-Pushout Rewriting at left-linear rules with the additional requirement that the involved pushout on the transformation's right-hand side is hereditary. This close connection between Single-Pushout and Sesqui-Pushout Rewriting leads to a rich theory for Single-Pushout Rewriting with total co-matches in general and in the special case of partial algebra transformation. In this paper, we present the concurrency theorem. It is a good example to demonstrate how far the theory for Single-Pushout Rewriting of total algebras carries over to partial algebras and where the differences are. We additionally presuppose:

(C4) $\mathscr{C}$ has all finite co-limits and

(C5) For every morphism $g \in \mathscr{S}$, all co-units of g_* are isomorphisms.

Definition 8 (Concurrent rule) Two transformations $t_1\langle m_1\rangle : G \to t_1@m_1$ and $t_2\langle m_2\rangle : t_1@m_1 \to t_2@m_2$ with rules $t_1 : L_1 \to R_1$ and $t_2 : L_2 \to R_2$ constitute a *concurrent rule* $t_2\langle m_2\rangle \circ t_1\langle m_1\rangle : G \to t_2@m_2$, if the co-match of the first and the match of the second are "jointly epic", i. e. the co-product morphisms $\{m_1\langle t_1\rangle, m_2\} : R_1 + L_2 \to t_1@m_1$ making the diagram commute is in $\mathscr{E}$.[22]

Theorem 5 *Given two transformations* $t_1\langle m_1\rangle : G \to t_1@m_1$ *and* $t_2\langle m_2\rangle : t_1@m_1 \to t_2@m_2$, *there is a direct transformation with a concurrent rule* $t_2\langle n_2\rangle \circ t_1\langle n_1\rangle$ *at a match n, such that* $(t_2\langle n_2\rangle \circ t_1\langle n_1\rangle)\langle n\rangle = t_2\langle m_2\rangle \circ t_1\langle m_1\rangle$.

Proof. Consider Figure 5. The first transformation uses rule $t_1 = (l_1 : K_1 \to L_1, r_1 : K_1 \to R_1)$ at match m_1 and produces trace $t_1\langle m_1\rangle = (l_1\langle m_1\rangle, r_1\langle m_1\rangle)$ and co-match $m_1\langle t_1\rangle$ and the second uses rule $t_2 = (l_2 : K_2 \to L_2, r_2 : K_2 \to R_2)$ at match m_2 with trace $t_2\langle m_2\rangle = (l_2\langle m_2\rangle, r_2\langle m_2\rangle)$ and co-match $m_2\langle t_2\rangle$. Let $(R_1 + L_2, i_1 : R_1 \to R_1 + L_2, i_2 : L_2 \to R_1 + L_2)$ be the co-product of R_1 and L_2 and let $\{m_1\langle t_1\rangle, m_2\} : R_1 + L_2 \to t_1@m_1$ be the unique morphism with $\{m_1\langle t_1\rangle, m_2\} \circ i_1 = m_1\langle t_1\rangle$ and $\{m_1\langle t_1\rangle, m_2\} \circ i_2 = m_2$. Construct $(e : R_1 + L_2 \to M, s : M \rightarrowtail t_1@m_1)$ as the $(\mathscr{E}, \mathscr{S})$-factorisation of $\{m_1\langle t_1\rangle, m_2\}$ and set $e_1 = e \circ i_1$ and $e_2 = e \circ i_2$. Now, e_1 and e_2 are "jointly epic".

Construct (s_1, r'_1) as the pullback of $(r_1\langle m_1\rangle, s)$ and let e'_1 be the morphism making the diagram commute. Since pullbacks preserve morphisms in $\mathscr{S}$, $s_1 \in \mathscr{S}$. Now, (r'_1, e_1) is hereditary pushout of (e'_1, r_1), since (i) $(r_1\langle m_1\rangle, s \circ e_1)$ is hereditary pushout of $(r_1, m_1\langle l_1\rangle)$, (ii) (e_1, id_{R_1}) is

[22] Compare (C1) on page 7!

pullback of $(m_1\langle t_1\rangle, s)$, (iii) $(e'_1, \mathrm{id}_{K_1})$ is pullback of $(m_1\langle l_1\rangle, s_1)$, and (iv) (r_1, id_{K_1}) is pullback of (id_{R_1}, r_1). Properties (ii) and (iii) are implied by s and s_1 being monic. Construct (s'_1, l'_1) such that (l'_1, s_1) is pullback of $(l_1\langle m_1\rangle, s'_1)$ and $s'_1 = l_1\langle m_1\rangle_*(s_1)$ and e''_1 as the morphism making the diagram commute. Since also $(e'_1, \mathrm{id}_{K_1})$ is pullback of $(m_1\langle l_1\rangle, s_1)$, $(e''_1, \mathrm{id}_{L_1})$ is pullback of (m_1, s'_1), (l_1, id_{K_1}) is pullback of (id_{L_1}, l_1), and final pullback complements are stable under pullbacks, $l'_1 = e''_{1*}(l_1)$. Thus, (l'_1, r'_1) and e_1 constitute a pushout of (l_1, r_1) and e''_1 in $\mathscr{C}^{\mathscr{S}}$. This implies that $(t_1\langle m_1\rangle, s)$ is pushout of (l'_1, r'_1) and s'_1.

Construct (s_2, l'_2) as the pullback of $(l_2\langle m_2\rangle, s)$ and let e'_2 be the morphism making the diagram commute such that $s_2 \in \mathscr{S}$. Since also $(e'_2, \mathrm{id}_{K_2})$ is pullback of $(m_2\langle l_2\rangle, s_2)$, (e_2, id_{L_2}) is pullback of (m_2, s), and (l_2, id_{K_2}) is pullback of (id_{L_2}, l_2), $l'_2 = e_{2*}(l_2)$. Finally, let (r'_2, e''_2) be the pushout of (r_2, e'_2) and s'_2 the morphism making the diagram commute. Since the pushout $(r_2\langle m_2\rangle, m_2\langle t_2\rangle)$ is hereditary, $(e'_2, \mathrm{id}_{K_2})$ is pullback of (l'_2, s_2), and (id_{K_2}, r_2) is pullback of (id_{R_2}, r_2), we conclude that (e''_2, r'_2) is hereditary pushout of (e'_2, r_2). Thus, (l'_2, r'_2) and e''_2 constitute a pushout of (l_2, r_2) and e_2 in $\mathscr{C}^{\mathscr{S}}$, and $(t_2\langle m_2\rangle, s'_2)$ is pushout of (l'_2, r'_2) and s.

Since pushouts compose, we found the concurrent rule $t_2\langle e_2\rangle \circ t_1\langle e''_1\rangle$ and the match s'_1 such that $(t_2\langle e_2\rangle \circ t_1\langle e''_1\rangle)\langle s'_1\rangle = t_2\langle m_2\rangle \circ t_1\langle m_1\rangle$. □

The opposite of Theorem 5, i. e. that every transformation with a concurrent rule at a match in $\mathscr{S}$ can be decomposed into transformations with the component rules, is true in suitable categories of total algebra where *all* pushouts are hereditary. This is no longer the case in categories of partial algebras as the following example demonstrates.

Example 4 (Hidden "element") *Consider again the signature Σ^c of Example 2, the category $\mathscr{A}^{\mathrm{M}}_{\Sigma^c}$, the partial algebras*

$$\begin{aligned} A &::= A_s = \{a\},\, f^A = \emptyset \text{ and} \\ B &::= B_s = \{b\},\, f^B = \left(\{f^B\},\, d^B_f(f^B) = *,\, c^B_f(f^B) = b\right), \end{aligned}$$

and the total morphism $p : A \to B ::= p_s(a) = b$. Instantiate the diagram in Fig. 5 by $L_1 = L'_1 = K_1 = K'_1 = K_2 = K'_2 = R_2 = R'_2 := A$, $R_1 = L_2 = M := B$, $l_1 = l'_1 = e''_1 = e'_1 = r_2 = r'_2 = e'_2 = e''_2 := \mathrm{id}_A$, $e_1 = e_2 = \mathrm{id}_B$, and $r_1 = r'_1 = l_2 = l'_2 := p$. These definitions result in $t_2 \circ t_1 : L_1 \to R_2 = (\mathrm{id}_A, \mathrm{id}_A)$. Now let $G = B$ and $s'_1 = p$. Then $t_2 \circ t_1$ is applicable at s'_1 and produces the trace $t_2 \circ t_1\langle s'_1\rangle = (\mathrm{id}_B, \mathrm{id}_B)$ and the co-match $s'_1\langle t_2 \circ t_1\rangle = p$. But t_1 is not applicable at $s'_1 \circ e''_1 = p$, since the transformation in $\mathscr{G}^{\mathrm{M}}_{\Sigma}$ results in a pushout as in Figure 3 which is not hereditary in $\mathscr{A}^{\mathrm{M}}_{\Sigma}$.

Thus, the fact that not all pushouts in categories of partial algebras are hereditary hinders transfer of some theoretical results from the total case to partial algebras. This disadvantage in theory pays off in practical applications as the examples in the next section will show.

5 Rewriting of Partial Algebras: Some Applications

Given a signature Σ, rewriting in $\mathscr{A}^{\mathrm{M}}_{\Sigma}$, $\mathscr{A}^{\mathrm{F}}_{\Sigma}$, and $\mathscr{A}^{\mathrm{C}}_{\Gamma}$ differs in the possibilities for a rule's left-hand side to delete hyperedges[23] and the concrete application conditions stipulated by the fact

[23] In $\mathscr{A}^{\mathrm{M}}_{\Sigma}$, arbitrary hyperedges (operation and/or predicate definitions) can be deleted without deleting any vertex, in $\mathscr{A}^{\mathrm{F}}_{\Sigma}$, deletion of a hyperedge requires simultaneous deletion of at least one vertex adjacent to the hyperedge (in the

that a transformation requires a hereditary pushout on its right-hand side. In this section, we consider examples in $\mathscr{A}^{\mathrm{M}}_{\Sigma}$ only, since the application conditions can easily be checked.

Proposition 9 *For rule $t = (l, r)$ and match candidate m in $\mathscr{A}^{\mathrm{M}}_{\Sigma}$, there is a morphism $m\langle l\rangle$ such that $(l, m\langle l\rangle)$ is pullback of $(m_*(l), m)$, if and only if m is* conflict-free*, i. e. $m(x) = m(y)$ and t defined for x implies that t is defined for y.*[24]

Proof. Using the construction of the right adjoint in Proposition 1, it is easy to check that the co-unit of m_* for l is an isomorphism, iff this condition holds. □

This result leads to the following procedure for rule application at a given match candidate:

Algorithm 1 Application of rule $(l : K \rightarrowtail L, r : K \to R)$ at match candidate $m : L \to G$ is performed in five steps:

1. Check the condition of Proposition 9! Proceed, if valid, abort otherwise!
2. Construct $(m_*(l) : K^* \to G)$ and $l\langle m\rangle : K \to K^*$, such that $(l\langle m\rangle, l)$ is pullback of $(m, m_*(l))$!
3. Construct the pushout $(r\langle m\rangle : K^* \to H, m\langle t\rangle : R \to H)$ of r and $l\langle m\rangle$ in $\mathscr{G}_{\Sigma}$!
4. Check uniqueness condition (1) on page 3 for H!
5. Finish transformation if the check in 4 is positive, rollback otherwise!

```
sorts O, Int
opns  i:O --> Int  (······►)
      +:Int,Int --> Int
```

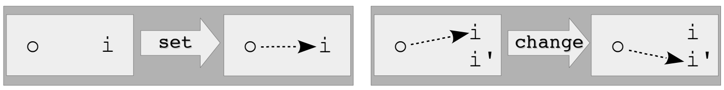

Figure 6: Setting and Changing an Attribute

The application condition in Algorithm 1(4), can be usefully exploited in many practical applications as a condition that prevents rule application. Our first example is a simple integer attribute `i` that can be *set* or *changed* for objects of type `O`. Figure 6 shows the underlying signature[25] and the two rules. Note that, due to the check in Algorithm 1(4), the `set`-rule can only be applied in a situation where the `i`-attribute of `o` has not been set yet. If there is an old value, the `change`-rule must be applied.

```
sorts O
opns  ≤:O,O  (······►)
```

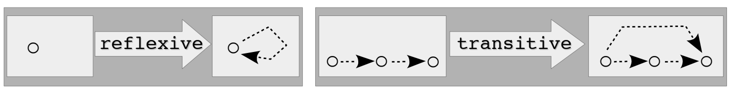

Figure 7: Reflexive/Transitive Closure

domain or codomain), and, in $\mathscr{A}^{\mathrm{C}}_{\Gamma}$, deletion of a hyperedge is only possible, if the domain vertex of this hyperedge is deleted as well.

[24] This condition is called *conflict-freeness* in [12] and [3] as well.

[25] In the signature, we declare the visualisations for the operations in brackets.

The next example handles the reflexive and transitive closure of a relation on the set `O`. We just apply the two rules `reflexive` and `transitive` as long as there are matches. Note that the algorithm terminates, since the rule `reflexive` cannot add loops to objects that possess a loop already. This is again due to the application condition in Algorithm 1(4). If all "abbreviations" are added, also the rule `transitive` is not applicable any more.

The next example in Figure 8 shows a typical copying process, here for a structure that is built up by the partial dyadic operation `t`. The unary predicate `r` marks the root of the structure to be copied. The `start` and the three `copy` rules perform the actual copy process. And the operation `c` keeps track of already built copies. Again, the application condition of single pushout rewriting for partial algebras guarantees that *exactly one* copy is made. Note that this copy mechanism works for hierarchical and even cyclic structures.

```
sorts N               (   )
opns  t:N,N --> N     (   )
      c:N   --> N     (   )
      r:N             (   )
```

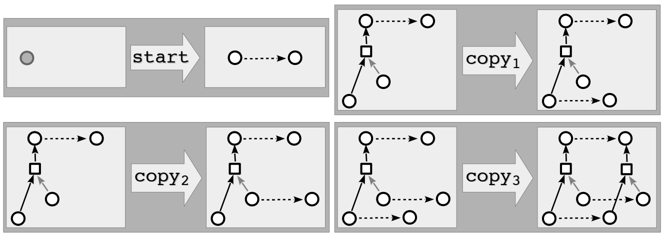

Figure 8: Copying

A situation as in the preceding example often occurs in the software engineering area of model transformation where objects in a source model must be mapped to objects in a target model. The transformation process starts with a completely undefined mapping and stops if the mapping is sufficiently defined. Uniqueness of the mapping has to be guaranteed during the whole transformation process. This is a perfect application scenario for our new rewriting approach. As an example consider the mapping of (small) object-oriented models to relational database

```
OO = sorts Class( ),Attribute( ),Association( )
     opns  owner: Attribute --> Class      ( )
           source: Association --> Class   ( )
           target: Association --> Class   ( )

DB = sorts Table( ),Column( ),PrimaryKey( ),ForeignKey( )
     opns  in: Column --> Table                      ( )
           at: PrimaryKey --> Column                 ( )
           at: ForeignKey --> Column                 ( )
           references: ForeignKey --> PrimaryKey     ( )

MT = OO + DB +
     opns  to: Class --> Table          ( )
           to: Attribute --> Column     ( )
           to: Association --> Table    ( )
```

Figure 9: Object-oriented Model, Relational Schema, and Inter-Model Mapping

schemata. The underlying signature is depicted in Figure 9. Here, an object-oriented model

(OO-part in Fig. 9) consists of classes, (untyped) attributes, and binary associations. A relational schema (DB-part in Fig. 9) comprises tables and columns. Columns can be marked as primary and/or foreign keys. The transformation signature is completed by the inter-model mapping in the MT-part in Figure 9. It specifies that classes shall be mapped to tables, attributes to columns, and associations to (junction) tables. The three rules that implement this transformation process are depicted in Figure 10. The condition in Algorithm 1(4) guarantees that the process stops if everything is mapped.

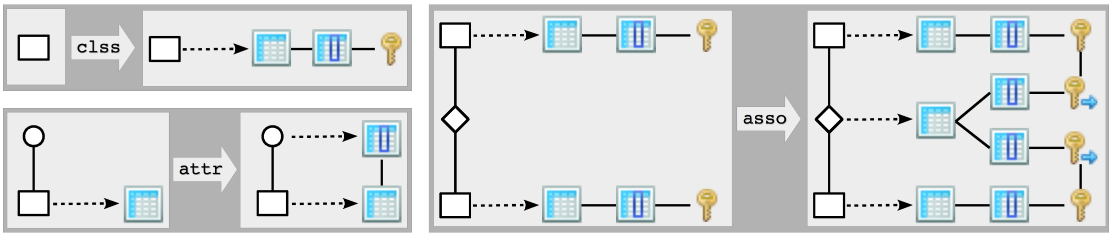

Figure 10: Transformation of Classes, Attributes, and Associations

The mapping of classes, attributes, and associations is one-to-one (up to some implementation details). It becomes more intricate, if inheritance comes into play. Figure 11 shows the extended object-oriented model and two variants $\mathtt{MT}_{\mathtt{I1}}$ and $\mathtt{MT}_{\mathtt{I2}}$ for the extension of the mapping.

$\mathtt{OO}_{\mathtt{I}}$ = OO + <u>opns</u> super: Class --> Class (——▷)
$\mathtt{MT}_{\mathtt{I1}}$= MT + $\mathtt{OO}_{\mathtt{I}}$ + <u>opns</u> to: Class, Class --> Table (······►)
$\mathtt{MT}_{\mathtt{I2}}$= MT + $\mathtt{OO}_{\mathtt{I}}$ + <u>opns</u> to: Class, Class --> ForeignKey (······►)

Figure 11: Object-Orientation with Inheritance

The first variant can be used to realise the pattern called *Single Table Inheritance* in [6], which puts a complete inheritance hierarchy into one table. The corresponding transformation rule single is depicted in Figure 12(left part).[26] The second variant is appropriate for the pattern called *Class Table Inheritance* in [6], which implements inheritance by a foreign key constraint on the primary key of the table for the sub-class referencing the primary key of the table for the super-class. The transformation rule join for this pattern can be found in Figure 12(right part).

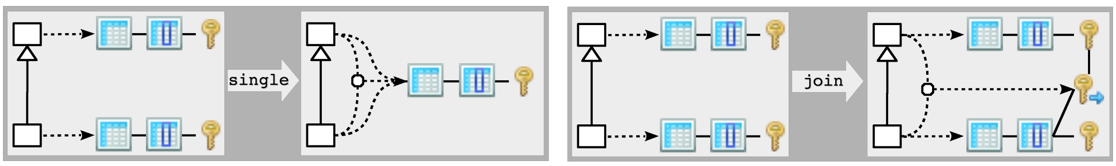

Figure 12: Handling Inheritance

[26] Note that the right-hand side of the rule is not injective!

All examples in this section demonstrate that the built-in application condition in Single-Pushout Rewriting for Partial Algebras is useful to control termination in transformations.

6 Related Work and Conclusions

We have introduced single-pushout rewriting of arbitrary partial algebras. As usual, transformations are defined by a single pushout of partial morphisms. Thus, general composition and decomposition properties of pushouts can be exploited for a rich theory. The new approach is built on a category of partial morphisms that *does not have all* pushouts. We provided a good characterisation of the situations which admit pushouts by hereditariness of underlying pushouts of total morphisms, compare Theorems 1, 2, and 3. Informally, pushouts can be built if the applied rule does not try to define operations where they are defined already. This application condition can easily be checked in every concrete situation. By some examples, we showed the practical relevance of the application condition for system design and the termination of derivation sequences. Within our approach, we do not have to distinguish between graph structures (objects and links) and data structures (base-types and -operations). We can easily model associations and attributes with at-most-one-multiplicity.

There are only a few articles in the literature that address rewriting of partial algebras, for example [2] and [1] for the double- and single-pushout approach resp. But both papers stay in the framework of signatures with *unary* operation symbols only and aim at an underlying category of partial morphisms that is co-complete. Aspects of partial algebras occur in all papers that are concerned with relabelling of nodes and edges, for example [9], or that invent mechanisms for exchanging the attribute value without deleting and adding an object, for example [7]. Most of these approaches avoid “real” partial algebras by completing them to total ones by some undefined-values. Thus, our approach is new, seems promising wrt. theoretical results, and shows some application potentials.

References

[1] Peter Burmeister, Miquel Monserrat, Francesc Rosselló, and Gabriel Valiente. Algebraic transformation of unary partial algebras II: single-pushout approach. *Theor. Comput. Sci.*, 216(1-2):311–362, 1999.

[2] Peter Burmeister, Francesc Rosselló, Joan Torrens, and Gabriel Valiente. Algebraic transformation of unary partial algebras I: double-pushout approach. *Theor. Comput. Sci.*, 184(1-2):145–193, 1997.

[3] Andrea Corradini, Tobias Heindel, Frank Hermann, and Barbara König. Sesqui-pushout rewriting. In Andrea Corradini, Hartmut Ehrig, Ugo Montanari, Leila Ribeiro, and Grzegorz Rozenberg, editors, *ICGT*, volume 4178 of *Lecture Notes in Computer Science*, pages 30–45. Springer, 2006.

[4] Hartmut Ehrig, Karsten Ehrig, Ulrike Prange, and Gabriele Taentzer. *Fundamentals of Algebraic Graph Transformation*. Springer, 2006.

[5] Hartmut Ehrig, Gregor Engels, Hans-Jörg Kreowski, and Grzegorz Rozenberg, editors. *Graph Transformations - 6th International Conference, ICGT 2012, Bremen, Germany, September 24-29, 2012. Proceedings*, volume 7562 of *Lecture Notes in Computer Science*. Springer, 2012.

[6] Martin Fowler. *Patterns of Enterprise Application Architecture*. Addison-Wesley, 2003.

[7] Ulrike Golas. A general attribution concept for models in M-adhesive transformation systems. In Ehrig et al. [5], pages 187–202.

[8] Annegret Habel, Reiko Heckel, and Gabriele Taentzer. Graph grammars with negative application conditions. *Fundam. Inform.*, 26(3/4):287–313, 1996.

[9] Annegret Habel and Detlef Plump. M,n-adhesive transformation systems. In Ehrig et al. [5], pages 218–233.

[10] Tobias Heindel. Hereditary pushouts reconsidered. In Hartmut Ehrig, Arend Rensink, Grzegorz Rozenberg, and Andy Schürr, editors, *ICGT*, volume 6372 of *Lecture Notes in Computer Science*, pages 250–265. Springer, 2010.

[11] Richard Kennaway. Graph rewriting in some categories of partial morphisms. In Hartmut Ehrig, Hans-Jörg Kreowski, and Grzegorz Rozenberg, editors, *Graph-Grammars and Their Application to Computer Science*, volume 532 of *Lecture Notes in Computer Science*, pages 490–504. Springer, 1990.

[12] Michael Löwe. Algebraic approach to single-pushout graph transformation. *Theor. Comput. Sci.*, 109(1&2):181–224, 1993.

[13] Michael Löwe. A unifying framework for algebraic graph transformation. Technical Report 2012/03, FHDW-Hannover, 2012.

[14] Michael Löwe, Martin Korff, and Annika Wagner. An algebraic framework for the transformation of attributed graphs. In M. R. Sleep et al, editor, *Term Graph Rewriting: Theory and Practice*, pages 185 – 199. Wiley, 1993.

[15] Michael Löwe and Marius Tempelmeier. Single-pushout rewriting of partial algebras. In Detlef Plump, editor, *Proceedings of the 6th International Workshop on Graph Computation Models, L'Aquila, Italy, July 20, 2015*, volume 1403 of *CEUR Workshop Proceedings*, pages 82–96. CEUR-WS.org, 2015.

[16] Miquel Monserrat, Francesc Rossello, Joan Torrens, and Gabriel Valiente. Single pushout rewriting in categories of spans I: The general setting. Technical Report LSI-97-23-R, Department de Llenguatges i Sistemes Informtics, Universitat Politcnica de Catalunya, 1997.

[17] Marius Tempelmeier and Michael Löwe. Single-Pushout Transformation partieller Algebren. Technical Report 2015/1 (in German), FHDW-Hannover, 2015.