On Rewriting XPath Queries Using Views

Foto Afrati*
National Technical University of Athens,Greece
afrati@softlab.ntua.gr

Rada Chirkova
North Carolina State University, USA
chirkova@csc.ncsu.edu

Manolis Gergatsoulis
Ionian University, Greece
manolis@ionio.gr

Benny Kimelfeld†‡
IBM Almaden Research Center
kimelfeld@us.ibm.com

Vassia Pavlaki*
National Technical University of Athens, Greece
vpavlaki@softlab.ntua.gr

Yehoshua Sagiv†
The Hebrew University of Jerusalem, Israel
sagiv@cs.huji.ac.il

ABSTRACT

The problem of rewriting a query using a materialized view is studied for a well known fragment of XPath that includes the following three constructs: wildcards, descendant edges and branches. In earlier work, determining the existence of a rewriting was shown to be coNP-hard, but no tight complexity bound was given. While it was argued that Σ_3^p is an upper bound, the proof was based on results that have recently been refuted. Consequently, the exact complexity (and even decidability) of this basic problem has been unknown, and there have been no practical rewriting algorithms if the query and the view use all the three constructs mentioned above.

It is shown that under fairly general conditions, there are only two candidates for rewriting and hence, the problem can be practically solved by two containment tests. In particular, under these conditions, determining the existence of a rewriting is coNP-complete. The proofs utilize various novel techniques for reasoning about XPath patterns. For the general case, the exact complexity remains unknown, but it is shown that the problem is decidable.

1. INTRODUCTION

Rewriting queries using views is one of the fundamental problems in databases with practical applications in information integration, data warehousing, Web-site design and query optimization. For relational databases, there is an extensive literature that deals with large fragments of SQL [1, 2, 6, 12, 16] and investigates various issues, including the complexity of the problem and efficient techniques for finding rewritings. However, for XML databases and XPath queries, there is only preliminary work. A widely studied practical fragment of XPath is $\mathsf{XP}^{\{//,[],*\}}$ consisting of tree patterns with child and descendant edges, branches and wildcards. This fragment has been recognized as an important fragment of XPath [8, 10, 14, 17]. The rewriting problem for this fragment was studied only in [17] where it was shown to be coNP-hard, but no tight complexity bound was given. They also argued that Σ_3^p is an upper bound, but their proof was based on results of [8] that have recently been refuted [10]. Consequently, the exact complexity (and even decidability) of this basic problem has been unknown. In this work, we study several sub-fragments of $\mathsf{XP}^{\{//,[],*\}}$ with the aim of determining the exact complexity of the problem and developing practical techniques that apply to XPath queries and views that are commonly used.

In the case of $\mathsf{XP}^{\{//,[],*\}}$, the rewriting problem is quite challenging. The difficulty arises from the combination of descendant edges, branches and wildcards which adds a limited form of disjunction. Even the containment problem is significantly more complex (i.e., coNP-complete [14]) for queries of $\mathsf{XP}^{\{//,[],*\}}$, compared to the three sub-fragments that are obtained by not allowing either wildcards, descendant edges or branches. For these three sub-fragments, containment is in PTIME [14] because it is characterized by the existence of a homomorphism, which is not true in the case of $\mathsf{XP}^{\{//,[],*\}}$. In fact, [17] showed that the rewriting problem for those three sub-fragments is in PTIME precisely because one only has to look for a homomorphism to determine containment.

It is rather difficult to show that the rewriting problem is in coNP when the existence of a homomorphism is not a necessary condition for containment. Yet, we are able to do that by using the following approach. We define the notion of *natural rewriting candidates,* which can be constructed in linear time, and check (by employing a containment test) whether one of them is indeed a rewriting. We prove several sufficient conditions that guarantee the *completeness* of our approach, namely, if a rewriting cannot be found among the natural candidates, then there is none at all. Moreover, we also prove that for the (large and practically important) sub-fragments of $\mathsf{XP}^{\{//,[],*\}}$ defined by those sufficient conditions, the rewriting problem is coNP-complete. In fact, the only "inefficient" step of our algorithm is the (generally coNP complete) test for equivalence of our candidate view-based

*The project is co-funded by the European Social Fund (75%) and National Resources (25%)—Operational Program for Educational and Vocational Training II (EPEAEK II) and particularly the Program PYTHAGORAS.

†This research was supported by The Israel Science Foundation (Grant 893/05).

‡Work was done while the author was at Hebrew University.

EDBT 2009, March 24–26, 2009, Saint Petersburg, Russia.

rewriting to the input query. These results are presented in Section 4.

The second type of our results is aimed at simplifying the given instance of the rewriting problem by transforming it into a new one that could be solved by means of the above sufficient conditions (or other methods, e.g., those of [17]). These techniques are presented in Section 5. We actually show how to get new sufficient conditions (for completeness) by combining these techniques with the results of Section 4.

The importance of our results is twofold. First, we significantly enlarge the sub-fragments of $\mathsf{XP}^{\{//,[\,],*\}}$ for which the rewriting problem can be solved in practice (our algorithms involve only a few containment tests, which might take exponential time but only in the size of the query and the view definition). Second, we develop new proof techniques for analyzing and reasoning about queries of the fragment $\mathsf{XP}^{\{//,[\,],*\}}$.

The lack of theoretical foundations on rewriting XPath queries using views is evident in related works (like [3, 5, 13, 18]) that use incomplete algorithms (e.g., XPath matching) for answering XPath queries using cached views. The problem of finding maximally contained (instead of equivalent) rewritings, either in the absence or presence of a schema, is studied in [11] for the fragment $\mathsf{XP}^{\{//,[\,]\}}$ (i.e., without wildcards). Query answering using views has been studied extensively for the class of regular path queries [4, 9] and in semistructured databases [15]. In [7], the problem of query reformulation for XML publishing is stated and solved in a general setting that allows both XML and relational storage for the data. In [10] the notions of redundancy and minimization are explored for the same fragment of XPath we study in this work. However, unlike the case of conjunctive queries, results on rewriting XPath queries are not easily derived from what is known about minimization of those queries.

2. FORMAL SETTING

2.1 XML Trees and Patterns

A *rooted* tree t is a directed graph with a designated node, denoted by $root(t)$, such that every other node of t is reachable from $root(t)$ through a unique directed path. In a *labeled* tree, every node n has a *label* which is denoted by label(n). We use $\mathcal{N}(t)$ and $\mathcal{E}(t)$ to denote the set of nodes and edges respectively, of a tree t.

Consider a tree t and an edge $(n_1, n_2) \in \mathcal{E}(T)$. Node n_1 is the *parent* of n_2, while n_2 is a *child* of n_1. A node n_1 is an *ancestor* of n_2 (and n_2 is a *descendant* of n_1) if t has a directed path from n_1 to n_2. The node n_1 is a *proper ancestor* of n_2 (and n_2 is a *proper descendant* of n_1) if, in addition, $n_1 \neq n_2$. Given a node n of t, we use t^n_Δ to denote the subtree of t that is rooted at n. The subtree of t that comprises the node n, one child m of n (including the edge connecting n to m) and the subtree t^m_Δ is called a *branch* of n in t. Observe that the number of branches of a node n is the number of children of n.

We consider two types of rooted, labeled trees that represent XML documents and queries, respectively. A document is called an *XML tree* (or *tree* for short) and its labels come from an infinite set Σ. We use $\mathbf{T}_\Sigma$ to denote the set of all the trees with labels from Σ. XPath queries are called *patterns* and they are different from XML trees in three aspects. First, the labels of a pattern come from the set $\Sigma \cup \{*\}$, where $*$ is the "wildcard" symbol ($* \notin \Sigma$). Second, a pattern P has two types of edges: $\mathcal{E}_/(P)$ is the set of *child edges* and $\mathcal{E}_{//}(P)$ is the set of *descendant edges*. Third, a pattern P has an *output* node that is denoted by $out(P)$. We define the special *empty pattern* and denote it by Υ.

As an example, Figure 1 depicts four patterns. Nodes are denoted as circles with labels inside them. Child edges and descendant edges are depicted by single and double lines, respectively. Note that the direction of edges is not explicitly shown, but is assumed to be from top to bottom. Output nodes are denoted by thicker circles.

Patterns represent the fragment $\mathsf{XP}^{\{//,[\,],*\}}$ of XPath that was investigated in [8, 10, 14, 17] and is described by the grammar

$$q \Longrightarrow\ q/q \ \mid\ q//q \ \mid\ q[q] \ \mid\ l \ \mid\ *$$

where l is a label in Σ. Next, we consider the result of applying a pattern to a tree.

Definition 2.1 (Embeddings / Weak Embeddings). *An* embedding *from a (nonempty) pattern P to a tree t is a mapping $e : \mathcal{N}(P) \to \mathcal{N}(t)$ with the following properties.*

- Root preserving. $e(root(P)) = root(t)$.
- Label preserving. *For all nodes $n \in \mathcal{N}(P)$, either* label(n) $= *$ *or* label(n) $=$ label($e(n)$).
- Child preserving. *For all edges $(n_1, n_2) \in \mathcal{E}_/(P)$, node $e(n_2)$ of t is a child of node $e(n_1)$.*
- Descendant preserving. *For all edges $(n_1, n_2) \in \mathcal{E}_{//}(P)$, node $e(n_2)$ is a proper descendant of node $e(n_1)$.*

If e is not root preserving, but satisfies the other three properties, then it is called a weak embedding.

Given an embedding $e : \mathcal{N}(P) \to \mathcal{N}(t)$, we usually denote by o the image of the output node, i.e., $o = e(out(P))$. The embedding e *produces* the tree t^o_Δ, that is, the subtree of t that is rooted at o. We denote by $P(t)$ the *result* of applying the pattern P to the tree t. It is naturally defined as the set of subtrees produced by all embeddings from P to t. Similarly, $P^w(t)$ is the set of all subtrees t^o_Δ, such that there is a weak embedding e of P in t with $o = e(out(P))$. The result of applying the empty pattern Υ to any tree (under either the regular or weak semantics) is the empty set. The pattern P can also be applied to a set of trees $\mathcal{T}$ and the result, denoted by $P(\mathcal{T})$ (resp., $P^w(\mathcal{T})$) is $\cup_{t\in\mathcal{T}} P(t)$ (resp., $\cup_{t\in\mathcal{T}} P^w(t)$).

If there is an embedding from a pattern P to a tree t, then t is a *model* of P. It is often useful to consider *canonical models* [14] rather than general ones. Next, we define this type of models. We denote by $\perp$ a special label of Σ. Throughout the paper, we assume that the patterns at hand do not include $\perp$ as a node label. A canonical model for a pattern P is any tree t that is obtained from P by the following two steps. *(1)* Each occurrence of the label $*$ is replaced with $\perp$, *(2)* Each descendant edge is replaced by a path of one or more edges, where all the internal nodes are labeled with $\perp$. We use $Mod(P)$ and $CMod(P)$ to denote the set of all models and all canonical models of P, respectively.

2.2 Containment and Equivalence

Containment and equivalence are defined as usual.

DEFINITION 2.2 (CONTAINMENT/EQUIVALENCE). *A pattern P_1 is* contained *in a pattern P_2, denoted by $P_1 \sqsubseteq P_2$, if $P_1(t) \subseteq P_2(t)$ for all trees $t \in \mathbf{T}_\Sigma$. The patterns P_1 and P_2 are* equivalent*, denoted by $P_1 \equiv P_2$, if $P_1 \sqsubseteq P_2$ and $P_2 \sqsubseteq P_1$, that is, $P_1(t) = P_2(t)$ for all trees $t \in \mathbf{T}_\Sigma$.*

Recall that an embedding is root preserving. Relaxing this condition leads to the following definition.

DEFINITION 2.3 (WEAK CONTAINMENT/EQUIVALENCE). *A pattern P_1 is* weakly contained *in a pattern P_2, denoted by $P_1 \sqsubseteq^w P_2$, if $P_1^w(t) \subseteq P_2^w(t)$ for all $t \in \mathbf{T}_\Sigma$. The patterns P_1 and P_2 are* weakly equivalent*, denoted by $P_1 \equiv^w P_2$, if $P_1 \sqsubseteq^w P_2$ and $P_2 \sqsubseteq^w P_1$, that is, $P_1^w(t) = P_2^w(t)$ for all $t \in \mathbf{T}_\Sigma$.*

Containment of P_1 in P_2 means that if a subtree t^o_Δ of t is produced by some embedding of P_1 in t, then t^o_Δ is also produced by an embedding of P_2 in t. In contrast, weak containment allows t^o_Δ to be produced by a weak embedding of P_2 in t. Thus, containment implies weak containment, but the converse is not necessarily true. Moreover, if P_1 and P_2 are equivalent, then they are also weakly equivalent. However, the opposite direction does not always hold.

In [14], it is shown that, in order to test containment (and equivalence) of patterns, it is enough to consider the canonical models. Formally, for all patterns P_1 and P_2 it holds that $P_1 \sqsubseteq P_2$ if and only if $CMod(P_1) \subseteq Mod(P_2)$. A similar test can be used for weak containment [10].

2.3 Pattern Composition

The *greatest lower bound* of two labels l_1 and l_2, denoted by $glb(l_1, l_2)$, is defined as follows. If $l \in \Sigma \cup \{*\}$, then $glb(l, l) = glb(l, *) = glb(*, l) = l$. If $l_1, l_2 \in \Sigma$ and $l_1 \neq l_2$, then $glb(l_1, l_2) = \diamond$ (where $\diamond \notin \Sigma$).

The *composition* of a pattern R with a pattern V, denoted by $R \circ V$, is obtained as follows. Let l^r_R be the label of the root of R and let l^o_V be the label of the output node of V. If $glb(l^r_R, l^o_V) = \diamond$, then $R \circ V = \Upsilon$ (the empty pattern). Otherwise, $R \circ V$ is obtained by merging the output node of V with the root of R and assigning the label $glb(l^r_R, l^o_V)$ to the merged node. Note that the children of the merged node are all those of $out(V)$ and $root(R)$. The pattern $R \circ V$ has the same root as V and the same output node as R. As a special case, if $root(R) = out(R)$, then the merged node is the output node of $R \circ V$.

As an example, Figure 1 shows three patterns: R, V and their composition $R \circ V$. Note that the merged node of $R \circ V$ is marked as m and its label is $*$, since both the output node of V and the root of R are labeled with $*$. Had one of these two nodes been labeled with $l \in \Sigma$ and the other with either $*$ or l, then l would have been the label of m.

In [17], it is shown that applying $R \circ V$ to a tree is the same as first applying V and then applying R.

PROPOSITION 2.4. [17] *$R \circ V(t) = R(V(t))$ holds for all trees $t \in \mathbf{T}_\Sigma$.*

Based on Proposition 2.4, the problem of rewriting a query using a view is defined in the next section.

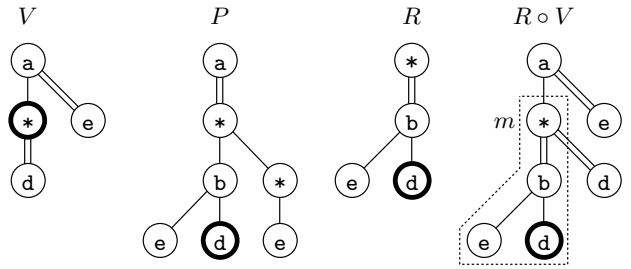

Figure 1: A rewriting example

2.4 Rewriting Queries using Views

A *materialized view* is the result of precomputing a pattern V; namely, V has already been applied to a tree t and the result $V(t)$ is available. When a new pattern P is posed as a query over t, we may want to use the materialized view instead of applying P directly to t. Therefore, we need to find a pattern R, such that applying R to $V(t)$ produces the same result as applying P to t, that is, $R(V(t)) = P(t)$. Furthermore, this equality should hold for all $t \in \mathbf{T}_\Sigma$.

By Proposition 2.4, the problem can be reformulated as follows. We say that R is an *equivalent rewriting* (or just *rewriting*) of P using V if $R \circ V \equiv P$. As an example, consider the patterns V, P and R of Figure 1. It can be shown that the composition $R \circ V$ is equivalent to P. Thus, R is a rewriting of P using V. The *rewriting-existence problem* is that of determining, for a pattern P and a view V, whether there is an equivalent rewriting R of P using V.

3. PRELIMINARY TOOLS AND RESULTS

In this section, we present some basic concepts and results that are later used.

3.1 Selection Paths and Sub-Patterns

The *selection path* of a nonempty pattern P is the path from the root to the output node. The nodes on the selection path are called *selection nodes*, while the edges on the selection path are called *selection edges*. The *depth* of a selection node v is the distance (i.e., number of edges) from the root to v. For example, the depth of the root is 0. We usually denote the depth of the output node by d and we say that d is also the depth of P. For $0 \leq k \leq d$, the *k-node* is the selection node at depth k. We extend the notion of depth as follows. For all nodes v of P, the depth of v is that of its deepest ancestor on the selection path.

Consider a pattern P of a depth d, and let k be an integer such that $0 \leq k \leq d$. The *k-sub-pattern* of P, denoted by $P^{\geq k}$, consists of all nodes v of P, such that the depth of v is greater than or equal to k. In other words, $P^{\geq k}$ is the subtree of P that is rooted at the k-node of P. The output node of $P^{\geq k}$ is that of P. The *k-upper-pattern* of P, denoted by $P^{\leq k}$, comprises all nodes of P at a depth of no more than k. That is, $P^{\leq k}$ is obtained from P by pruning the subtree rooted at the $(k{+}1)$-node. The output node of $P^{\leq k}$ is the k-node of P. Note that $P^{\geq 0}$ and $P^{\leq d}$ are the same as P. We similarly define $P^{>k}$ $(0 \leq k < d)$ and $P^{<k}$ $(0 < k \leq d)$.

The next proposition shows some basic properties of k-sub-patterns of (weakly) equivalent patterns. In the proof of Proposition 3.1, and in some subsequent proofs, we use a

transformation[1] τ that constructs a tree $\tau(P)$ from a pattern P by replacing each occurrence of $*$ with $\perp$ (recall our assumption that $\perp$ does not appear in any of the patterns at hand). Note that each node of P has exactly one corresponding node in $\tau(P)$, and similarly for the edges.

PROPOSITION 3.1. *Let P_1 and P_2 be two weakly equivalent patterns with depths d_1 and d_2, respectively. For all k, where $0 \leq k \leq d_1$, the following hold.*

1. *$d_1 = d_2$.*
2. *The k-sub-patterns of P_1 and P_2 are weakly equivalent, i.e., $P_1^{\geq k} \equiv^w P_2^{\geq k}$.*
3. *The k-nodes of P_1 and P_2 have the same label.*

PROOF. **Part 1**. Let $t = \tau(P_1)$ and let o be the node of t that corresponds to the output node of P_1. Clearly, there is an embedding e_1 of P_1 in t that produces the subtree t^o_Δ. Since $P_1 \sqsubseteq^w P_2$, there is a weak embedding e_2 of P_2 in t that produces t^o_Δ. By the construction of t, the depth of o in t is exactly d_1. From the fact that t^o_Δ is produced by a weak embedding e_2 of P_2 in t, we conclude that $d_1 \geq d_2$. By symmetry, it follows that $d_1 = d_2$.

Part 2. We prove that $P_1^{\geq k} \sqsubseteq^w P_2^{\geq k}$ (the other direction is symmetric). Suppose that e_1^k is an embedding of $P_1^{\geq k}$ in a tree t, such that e_1^k produces t^o_Δ. We have to prove the existence of a weak embedding e_2^k of $P_2^{\geq k}$ in t, such that e_2^k produces t^o_Δ. Let $t_1 = \tau(P_1^{<k})$ and let m^{k-1} be the node of t_1 that corresponds to the $(k-1)$-node of $P_1^{<k}$. We combine t_1 and t into a tree t' by adding an edge from m^{k-1} to the root of t. Then, e_1^k can be naturally extended to an embedding e_1 of P_1 in t', such that e_1 produces t^o_Δ. Since $P_1 \equiv^w P_2$, there is a weak embedding e_2 of P_2 in t' that produces t^o_Δ. The embedding e_2 maps the k-node n_k of P_2 to a node $e_2(n_k)$ in t', such that the depth of $e_2(n_k)$ is at least k. It follows that $e_2(n_k)$ is a descendant of $root(t)$, because the distance from $root(t')$ to $root(t)$ is k. So, the restriction of e_2 to the nodes of $P_2^{\geq k}$ is the required embedding e_2^k.

Part 3. From Part 1, we know that P_1 and P_2 have the same depth d. Consider an embedding e_1 that maps each node of P_1 to its corresponding node in $t = \tau(P_1)$. Clearly, the depth of $o = e_1(out(P_1))$ is d. Let n_k be the k-node of P_1 ($0 \leq k \leq d$) and let $m_k = e_1(n_k)$. Note that m_k is on the path from $root(t)$ to o. Since $P_1 \equiv^w P_2$, there is a weak embedding e_2 of P_2 in t that produces t^o_Δ. Since the depth of P_2 is d, the embedding e_2 maps $root(P_2)$ to $root(t)$, and it maps the k-node of P_2 to node m_k of t. Therefore, if the k-node of P_2 has a label $l \neq *$, then the k-node of P_1 must have the same label (recall that in the construction of t, occurrences of $*$ were replaced with $\perp$). A symmetric argument shows that the same holds in the opposite direction. □

We *combine* a pattern P_1 with a pattern P_2 by choosing a k-node of P_1 and introducing a descendant edge from that node to the root of P_2. The combined pattern, denoted by $P_1 \overset{k}{\Longrightarrow} P_2$, has the same root as P_1 while its output node is that of P_2. For example, if in a pattern P a descendant edge enters the k-node, then $P^{<k} \overset{k-1}{\Longrightarrow} P^{\geq k}$ is the same as the pattern P.

[1]Essentially, τ generates the minimal canonical model.

The following proposition shows that if a descendant edge enters the k-node of a pattern P, then the k-sub-pattern $P^{\geq k}$ can be replaced with any weakly equivalent pattern Q while preserving equivalence to the original pattern P.

PROPOSITION 3.2. *Let P be a pattern of depth d. Let $1 \leq k \leq d$ and suppose that the k-sub-pattern $P^{\geq k}$ is weakly equivalent to a pattern Q. If a descendant edge enters the k-node of P, then $P \equiv (P^{<k} \overset{k-1}{\Longrightarrow} Q)$.*

PROOF. We show that $P \sqsubseteq (P^{<k} \overset{k-1}{\Longrightarrow} Q)$; the other direction is proved similarly. Let e be an embedding of P in a tree t that produces the subtree t^o_Δ. Let e_1 be the restriction of e to $P^{<k}$. Suppose that e maps the k-node of P to node m of t. Since Q is weakly equivalent to $P^{\geq k}$, there is a weak embedding e_2 of Q in t^m_Δ that produces t^o_Δ. Observe that m is a proper descendant of $e_1(n_{k-1})$, where n_{k-1} is the $(k-1)$-node of P; hence, so is $e_2(root(Q))$. It follows that the mapping $\hat{e}$ that maps the nodes of $P^{<k}$ as e_1 and the nodes of Q as e_2 is an embedding of $P^{<k} \overset{k-1}{\Longrightarrow} Q$ in t that produces the same subtree of t as e_2, namely, t^o_Δ. This completes the proof. □

Propositions 3.1(2) and 3.2 imply that if the patterns P_1 and P_2 are equivalent and a descendant edge enters the k-node of P_1, then the k-sub-pattern $P_1^{\geq k}$ can be replaced with the k-sub-pattern $P_2^{\geq k}$ while preserving equivalence.

COROLLARY 3.3. *Suppose that $P_1 \equiv P_2$ and both patterns are of depth d. If a descendant edge enters the k-node of P_1 ($1 \leq k \leq d$), then $(P_1^{<k} \overset{k-1}{\Longrightarrow} P_2^{\geq k}) \equiv P_1$.*

3.2 Preliminary Results on Rewriting

In [17], it was shown that the rewriting-existence problem is coNP-hard. They also argued that this problem is in Σ^p_3, but their proof was based on the results of [8], which have recently been refuted in [10]. The next proposition shows that this problem is decidable.

PROPOSITION 3.4. *The rewriting-existence problem is decidable.*

PROOF. (*Sketch*) Consider a pattern P and a view V, and suppose that R is a rewriting of P using V. Let k be the depth of V. The *height* of a pattern is the maximal number of edges on any path from the root to a leaf.

Part 2 of Proposition 3.1 shows that $(R \circ V)^{\geq k}$ is weakly equivalent to $P^{\geq k}$. It is easy to show that weakly equivalent patterns have the same height and the same set of labels. Consequently, the height of R is at most that of $P^{\geq k}$ and its set of labels is contained in that of $P^{\geq k}$. Furthermore, without loss of generality (abbr. w.l.o.g.), we can assume that R is *non-redundant* [10]. Let $\mathcal{R}$ be a maximal set of patterns R' with the above three properties of R, such that $\mathcal{R}$ does not include isomorphic patterns (where the meaning of *isomorphism* is the obvious one, e.g., as defined in [10]). It is easy to show (e.g., by induction on the height of $P^{\geq k}$) that $\mathcal{R}$ is finite and, moreover, can be constructed by a Turing machine. So, to determine whether there is a rewriting of P using V, it is enough to test for all $R' \in \mathcal{R}$, whether $R' \circ V$ is equivalent to P (which is a coNP-complete problem [14]). □

The above proof implies an algorithm for finding a rewriting, and it can be shown that the running time is at most double exponential. The next proposition discusses a special type of rewriting, namely, when the output node of the view V is its root.

PROPOSITION 3.5. *Let P and R be patterns. Consider a view V, such that $root(V) = out(V)$. If $R \circ V \equiv P$, then $R \circ V \equiv P \circ V$.*

PROOF. Observe that the root of $P \circ V$ is also the root of both P and V. Moreover, the selection path of $P \circ V$ is the same as that of P. Consequently, if there is an embedding e from $P \circ V$ to a tree t, then the restriction of e to the nodes of P is an embedding from P to t that produces the same output as e. It thus follows that $P \circ V \sqsubseteq P$.

For the other direction, we need to show that $P \sqsubseteq P \circ V$. Suppose that e_1 is an embedding of P in a tree t. The equivalence $R \circ V \equiv P$ implies that there is an embedding e_2 of $R \circ V$ in t, such that $e_1(out(P)) = e_2(out(R \circ V))$. Let e be the embedding from $P \circ V$ to t that maps every node of P as e_1 and every node of V as e_2. Since P and $P \circ V$ have the same output node, $e_1(out(P)) = e(out(P \circ V))$, i.e., e generates the same output as e_1. We also need to show that e is a well-defined embedding of $P \circ V$ in t. Since $P \circ V$ is obtained by merging the roots of P and V, we should prove that $e_1(root(P)) = root(t) = e_2(root(V))$. The first equality holds, because e_1 is an embedding from P to t. The second follows from the fact that $root(V)$ is the root of $R \circ V$ and e_2 is an embedding of $R \circ V$ in t. Thus, the existence of e implies that $P \sqsubseteq P \circ V$. □

The above proposition remains correct even if equivalence is replaced with weak equivalence. Before showing that, we need to prove the following proposition.

PROPOSITION 3.6. *Let P_1 and P_2 be weakly equivalent patterns. Suppose that e' is an embedding of P_1 in a tree t. Then there are weak embeddings e_1 and e_2 of P_1 and P_2, respectively, in t such that*

- $e_1(root(P_1)) = e_2(root(P_2))$, *and*
- $e_1(out(P_1)) = e'(out(P_1)) = e_2(out(P_2))$.

PROOF. Consider the set of all the embeddings of P_1 in t that produce t^o_Δ, where $o = e'(out(P_1))$. Let e_1 be an embedding from this set, such that the depth of the image of $root(P_1)$ is maximal. We similarly choose e_2 from the set of all embeddings of P_2 in t that produce t^o_Δ. Suppose, by way of contradiction, that $e_1(root(P_1)) \neq e_2(root(P_2))$. Note that both images are on the path from $root(t)$ to o. W.l.o.g., suppose that $e_1(root(P_1))$ is deeper than $e_2(root(P_2))$. Since $P_1 \equiv^w P_2$, there exists a weak embedding of P_2 in the subtree of t that is rooted at $e_1(root(P_1))$, such that the output is t^o_Δ. This contradicts the choice of e_2. □

Now we can prove Proposition 3.7 that corresponds to Proposition 3.5 and considers the case of weak equivalence.

PROPOSITION 3.7. *Let P and R be patterns. Consider a view V, such that $root(V) = out(V)$. If $R \circ V \equiv^w P$, then $R \circ V \equiv^w P \circ V$.*

PROOF. The first part of the proof of Proposition 3.5 does not assume that $R \circ V \equiv P$. Thus, $P \circ V \sqsubseteq P$ always holds provided that $root(V) = out(V)$, and so does $P \circ V \sqsubseteq^w P$.

Next, we show that $P \sqsubseteq^w P \circ V$. Suppose that $\hat{e}_1$ is an embedding of P in a tree t. The weak equivalence $R \circ V \equiv^w P$ and Proposition 3.6 imply that there are weak embeddings e_1 and e_2 of P and $R \circ V$, respectively, in t such that the following equalities hold.

$$e_2(root(R \circ V)) = e_1(root(P)) \quad (1)$$

$$e_2(out(R \circ V)) = e_1(out(P)) = \hat{e}_1(out(P)) \quad (2)$$

Let e be the weak embedding of $P \circ V$ in t that maps every node of P as e_1 and every node of V as e_2. Clearly, $e(out(P \circ V)) = e_1(out(P))$, since P and $P \circ V$ have the same output node. So, Equation (2) implies that $e(out(P \circ V)) = \hat{e}_1(out(P))$, i.e., e produces the same output as $\hat{e}_1$. The fact that e is a well-defined weak embedding of $P \circ V$ in t follows immediately from Equation (1). □

4. NATURAL REWRITING CANDIDATES

Consider a pattern P and a view V with depths d and k, respectively. By Proposition 3.1, if R is a rewriting of P using V, then $R' \equiv^w P^{\geq k}$, where k is the depth of V and $R' = (R \circ V)^{\geq k}$. Intuitively, it may seem that $P^{\geq k}$ is the only possible candidate for a rewriting. This intuition, however, is misleadingly narrow. As an example, consider again the patterns P, V and R of Figure 1. Although R is a rewriting, $P^{\geq 1}$ is not. Nevertheless, in this case, we can obtain a rewriting from $P^{\geq 1}$ by *relaxing* the edges that emanate from its root, namely, replacing all of them with descendant edges. This example leads us to the definition of *natural candidates.*

Let Q be a pattern. We use $Q_{\mathrm{r}//}$ to denote the pattern that is obtained by relaxing the edges that emanate from the root of Q. Observe that $Q \sqsubseteq Q_{\mathrm{r}//}$. Now, consider a pattern P and a view V with depths d and k, respectively. The pattern R' is a *natural rewriting candidate* (or just *natural candidate*) w.r.t. P and V if R' is either $P^{\geq k}$ or $P^{\geq k}_{\mathrm{r}//}$. As an example, the middle part of Figure 2 depicts the natural candidates w.r.t. the patterns P and V of Figure 1. When P and V are clear from the context, we may simply say that R' is a *natural candidate*.

Our approach to the rewriting problem is, usually, to test whether one of the natural candidates is a solution. This can be done by checking equivalence, which is a coNP-complete problem [14]. In the remainder of this paper, we give conditions that guarantee the *completeness* of this approach, namely, if we do not find a rewriting, then one does not exist. First, we define some terminology.

The pattern R' is a *potential rewriting* w.r.t. P and V when the following condition holds: If there is some rewriting, then R' is also a rewriting; in other words, if R' is not a rewriting, then one does not exist. Again, when P and V are clear from the context, we just say that R' is a potential rewriting. Our results provide conditions that guarantee the existence of a potential rewriting among the two natural candidates. One may ponder whether it could be that some rewriting exists even when none of the natural candidates is one. This problem is still open.

Let P be a pattern and V be a view. In the sequel, d and k denote the depths of P and V respectively. Proposition 3.1 implies that if $k > d$, then there is no rewriting of P using V.

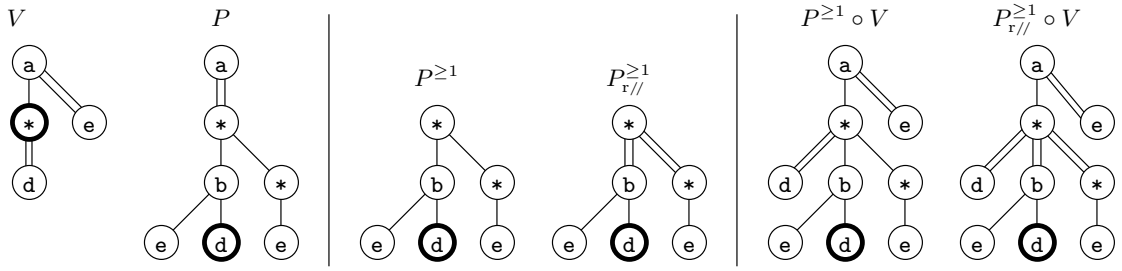

Figure 2: Patterns P and V, the natural candidates and their compositions with V

If $k = d$, then it is rather straightforward to show that every rewriting R satisfies $P^{\geq k} \equiv (R \circ V)^{\geq k}$, which implies that $P^{\geq k}$ is a rewriting. So, if $k = d$ then a natural candidate is a potential rewriting (moreover, the rewriting-existence problem is coNP-complete under the assumption of $d = k$). Therefore, in the sequel we assume that $k < d$.

4.1 Guaranteeing Completeness

In this section, we prove that some common properties of patterns guarantee that at least one natural candidate is a potential rewriting. Recall that d and k are the depths of the query pattern P and the view pattern V, respectively.

4.1.1 Properties of the Query

First, we consider properties of the query pattern P that guarantee the existence of a potential rewriting among the two natural candidates. For that, we need the notion of *stability* [10]. We say that a pattern Q is *stable* if the following holds. For all patterns Q', if $Q' \equiv^w Q$, then $Q' \equiv Q$; that is, weak equivalence to Q is the same as ordinary equivalence to Q. The next proposition follows from the results of [10].

PROPOSITION 4.1. *A pattern Q is stable in each of the following cases.*

- *The label of $root(Q)$ is not $*$.*
- *The depth of Q is* 0.
- *The depth of Q is at least* 1 *and Q contains a label of Σ that does not appear in $Q^{\geq 1}$.*

Note that the third condition above means that one of the branches that emanate from the root has a label of Σ that does not appear in $Q^{\geq 1}$. The next proposition is rather straightforward.

PROPOSITION 4.2. *Let R be a rewriting of P using V. If $(R \circ V)^{\geq k} \equiv P^{\geq k}$, then $P^{\geq k}$ is a rewriting of P using V.*

Part 2 of Proposition 3.1 and Proposition 4.2 imply the following sufficient condition for one of the natural candidates to be a rewriting provided that there is one.

THEOREM 4.3. *If $P^{\geq k}$ is stable, then it is a potential rewriting.*

As a special case, Theorem 4.3 and Proposition 4.1 show that if $*$ is not the label of the k-node of P, then a rewriting exists if and only if $P^{\geq k}$ is one. Observe that if the label of the k-node of P is $*$ and that of $out(V)$ is not, then a rewriting does not exist (by Part 3 of Proposition 3.1).

The following theorem considers the case where no descendant edge appears on the path from the root of P to the k-node.

THEOREM 4.4. *If the selection path of $P^{\leq k}$ has only child edges, then $P^{\geq k}$ is a potential rewriting.*

In the remainder of this section, we prove Theorem 4.4. We assume that R is an equivalent rewriting of P using V, and we will show that $P^{\geq k}$ is also such a rewriting. The following proposition is rather straightforward and its proof is omitted.

PROPOSITION 4.5. *Let Q and Q' be equivalent patterns. Suppose that the first i edges in the selection paths of both Q and Q' are child edges. Then the i-sub-patterns of Q and Q' are equivalent.*

If the selection path of V consists of only child edges, then by Proposition 4.5, $P^{\geq k} \equiv (R \circ V)^{\geq k}$. Furthermore, from Proposition 4.2, it follows that $P^{\geq k}$ is a rewriting of P using V. So, in the remainder of this proof, we assume that the selection path of V contains at least one descendant edge.

Consider a pattern Q and let n be a node that is not in Q. Let n/Q and $n//Q$ be the patterns obtained by connecting n to $root(Q)$ with a child and descendant edge, respectively. Note that n is the root of both n/Q and $n//Q$. The next lemma can be proved by a straightforward adaptation of the proof of Lemma 4.7 in [10].

LEMMA 4.6. *Let Q and Q' be patterns, and let n be a node of neither Q nor Q'. If $n//Q \equiv n/Q'$, then $n//Q \equiv n//Q_{\mathrm{r}//}$ and $n//Q_{\mathrm{r}//} \equiv n/Q_{\mathrm{r}//}$.*

Consider V and the minimal i, such that a descendant edge connects the i-node to the $(i+1)$-node. Let n_i^v and n_i^p denote the i-nodes of V and P, respectively. By the choice of i, the selection path of V has only child edges above n_i^v. This implies that $P^{\geq i} \equiv (R \circ V)^{\geq i}$. By this equivalence and the fact that a child edge and a descendant edge connect n_i^v and n_i^p, respectively, to the next node on their selection paths, it follows that $n_i^p/P^{\geq i+1} \equiv n_i^v//(R \circ V)^{\geq i+1}$. Lemma 4.6 implies that $n_i^v//(R \circ V)^{\geq i+1} \equiv n_i^v/Q'$, where Q' is obtained from $(R \circ V)^{\geq i+1}$ by replacing the outgoing edges of the root with descendant edges. Therefore, in $R \circ V$, the branch of n_i^v that includes the $(i+1)$-node can be replaced with n_i^v/Q' while preserving equivalence. After this replacement, a descendant edge connects the $(i+1)$-node and the $(i+2)$-node. So, we can continue this replacement repeatedly until we finish at the k-node. Let Q be the result. Then the following hold for Q. *(1)* $Q \equiv R \circ V$; *(2)* The first k selection edges of Q are child edges; *(3)* For $i < j < k$, all the outgoing edges of the j-node of Q are descendant edges, except the one that leads to the $(j+1)$-node; and *(4)* All the outgoing edges of the k-node of Q are descendant edges.

Observe that all of the selection nodes of P at depths $i+1,\ldots,k$ are necessarily wildcard nodes. Otherwise, one can easily construct a model of $R \circ V$ that is not one of P. By Part 3 of Proposition 3.1, we conclude that this is also the case for Q. Consequently, one can get an equivalent pattern by transforming the incoming edge of the k-node of Q into a descendant edge (since all the outgoing edges of the k-node are descendants). Furthermore, by using the same argument, this can also be done with the incoming edge of the $(k-1)$-node and so on, until the $(i+1)$-node. So, let Q_w be the equivalent pattern that is obtained by this process. That is, Q_w is identical to Q, except that the edges between the i-node to the k-node are all descendant edges. Observe that Q_w can be formulated as the composition $R_{\mathrm{r}//} \circ V_w$, where V_w is obtained from V by transforming some child edges to descendant ones (hence $V \sqsubseteq V_w$). To conclude the proof, we show that the following proposition holds. Recall that $R_{\mathrm{r}//} \circ V_w$ is equivalent to Q and, hence, it is equivalent to $R \circ V$ and P.

PROPOSITION 4.7. $P^{\geq k} \circ V \equiv R_{\mathrm{r}//} \circ V_w$.

PROOF. Observe that the k-sub-pattern of $R_{\mathrm{r}//} \circ V_w$ is the k-sub-pattern of Q. Since the first k selection edges of Q are child edges, we conclude from Proposition 4.5 that $P^{\geq k} \equiv (R_{\mathrm{r}//} \circ V_w)^{\geq k}$. To prove that $P^{\geq k} \circ V \sqsubseteq R_{\mathrm{r}//} \circ V_w$, recall that $V \sqsubseteq V_w$. So, from $P^{\geq k} \equiv (R_{\mathrm{r}//} \circ V_w)^{\geq k}$ it follows that $P^{\geq k} \circ V \sqsubseteq R_{\mathrm{r}//} \circ V_w$, as claimed.

To prove the other direction, $R_{\mathrm{r}//} \circ V_w \sqsubseteq P^{\geq k} \circ V$, recall that $R_{\mathrm{r}//} \circ V_w \equiv R \circ V$. Note that $(R \circ V)^{\geq k} \sqsubseteq (R_{\mathrm{r}//} \circ V_w)^{\geq k}$ (since the latter is obtained from the former by transforming the child edges emanating from the root to descendant ones) and, as shown above, $(R_{\mathrm{r}//} \circ V_w)^{\geq k} \equiv P^{\geq k}$. So, $(R \circ V)^{\geq k} \sqsubseteq P^{\geq k}$. It follows that $R \circ V \sqsubseteq P^{\geq k} \circ V$ and, consequently, $R_{\mathrm{r}//} \circ V_w \sqsubseteq P^{\geq k} \circ V$, as claimed. □

The following corollary of Theorems 4.3 and 4.4 shows that the rewriting-existence problem is coNP-complete in the cases considered in this section. Observe that membership in coNP follows from the theorems, while coNP-hardness is obtained by rather straightforward reductions from the problem of testing containment of patterns [14].

COROLLARY 4.8. *Under each of the following assumptions, the rewriting-existence problem is coNP-complete.*

1. *$P^{\geq k}$ satisfies one or more of the properties of Q in Proposition 4.1.*
2. *The selection path of $P^{\leq k}$ has only child edges.*

4.1.2 Properties of the View

We now consider properties of the view pattern V. The following theorem shows that one of the natural candidates is a potential rewriting provided that a descendant edge enters the output node of V.

THEOREM 4.9. *If a descendant edge enters the output node of V, then $P^{\geq k}$ is a potential rewriting.*

PROOF. Suppose that there is a rewriting R of P using V. We show that $P^{\geq k}$ is such a rewriting. Since $R \circ V \equiv P$ and a descendant edge enters the output node of V, Corollary 3.3 implies that the k-sub-pattern of $R \circ V$ can be replaced with the k-sub-pattern of P, i.e., the following holds.

$$(R \circ V) \equiv \left((R \circ V)^{<k} \stackrel{k-1}{\Longrightarrow} P^{\geq k}\right) \tag{3}$$

By Proposition 3.1, the two equivalent patterns of (3) have k-sub-patterns that are weakly equivalent. Thus,

$$(R \circ V)^{\geq k} \equiv^{w} P^{\geq k}.$$

By definition, $(R \circ V)^{\geq k}$ and $R \circ (V^{\geq k})$ are the same. Thus,

$$R \circ (V^{\geq k}) \equiv^{w} P^{\geq k}$$

and since $root(V^{\geq k}) = out(V^{\geq k})$, Proposition 3.7 implies that

$$R \circ (V^{\geq k}) \equiv^{w} P^{\geq k} \circ (V^{\geq k}). \tag{4}$$

As noted above, the left-hand side of (4) is the same as $(R \circ V)^{\geq k}$ and, similarly, the right-hand side is identical to $(P^{\geq k} \circ V)^{\geq k}$. Hence, we get the following:

$$(R \circ V)^{\geq k} \equiv^{w} (P^{\geq k} \circ V)^{\geq k}. \tag{5}$$

We use (5) and Proposition 3.2 to replace, in $R \circ V$, the k-sub-pattern $(R \circ V)^{\geq k}$ with $(P^{\geq k} \circ V)^{\geq k}$. The result is $P^{\geq k} \circ V$ and, so, $P \equiv R \circ V \equiv P^{\geq k} \circ V$, as required. □

The following theorem considers the case where the selection path of V does not contain descendant edges.

THEOREM 4.10. *If the selection path of V has only child edges, then at least one of the natural candidates is a potential rewriting.*

As an example, consider again the patterns of Figure 2. Observe that V has only one selection edge, which is a child edge. As mentioned earlier, $P^{\geq 1}$ is not a rewriting of P using V. However, we prove later that, in this case, the natural candidate $P^{\geq 1}_{\mathrm{r}//}$ is a potential rewriting and, indeed, the reader can verify that it is actually a rewriting of P using V.

In the remainder of this section, we prove Theorem 4.10. By Theorem 4.4, if the first k selection edges of P are child edges, then $P^{\geq k}$ is a potential rewriting. So, to prove Theorem 4.10, it suffices to consider the case where the selection path of V comprises only child edges and at least one of the first k selection edges of P is a descendant edge.

We assume that R is a rewriting of P using V and that (n_i, n_{i+1}) is a descendant edge among the first k edges of P. The following holds.

LEMMA 4.11. *If p is a directed path of $R \circ V$ that starts at $out(V)$ and consists of only child edges, then p has only wildcard labels and it does not contain $out(R)$.*

PROOF. If p contains $out(R)$, then the selection path of $R \circ V$ consists of only child edges while that of P does not. Hence, it is easy to come up with a tree t, such that $P(t)$ contains a subtree that cannot be produced by any embedding of $R \circ V$ in t.

Suppose, by way of contradiction, that p contains a node n labeled with $l \neq *$. Then, every embedding of $R \circ V$ in some tree maps n to a node v labeled with l, such that the distance from $e(out(V))$ to v is at most $|p|$ (i.e., the number of edges of p). Now, consider the canonical model t of P that is obtained by replacing (n_i, n_{i+1}) with a long path (e.g., of a length twice the height of $R \circ V$), such that all the interior nodes on that path have a new label. Let o be the node of t that corresponds to $out(P)$. An embedding of $R \circ V$ that maps $out(R)$ to o must map $out(V)$ and n to two of the new nodes. Thus, we obtain a contradiction. □

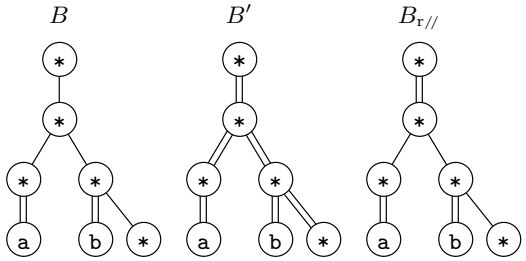

Figure 3:

By using Lemma 4.11, the following is shown.

LEMMA 4.12. *The following hold.*

1. $R_{\mathrm{r}//} \equiv R$.
2. $V^{\geq k}_{\mathrm{r}//} \equiv V^{\geq k}$.
3. $P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//} \equiv P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}$.
4. $R_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//} \equiv R \circ V^{\geq k}$.

PROOF. Let B be a branch of R. We prove that $B \equiv B_{\mathrm{r}//}$. We assume that the outgoing edge of the root of B is a child edge (otherwise the claim is trivial). Let p be a maximal path of B that starts at $root(B)$ and visits only child edges. Suppose that p ends at node n. Lemma 4.11 implies that the label of n is $*$ and $n \neq out(R)$. Besides, as p is maximal, n is either a leaf or all the outgoing edges of n are descendant. In either case, we can replace the incoming edge of n with a descendant edge. Continuing this process, we will end up replacing the outgoing edge of $root(B)$ with a descendant one. Let B' be B after that last replacement. For illustration, Figure 3 contains examples of B, B' and $B_{\mathrm{r}//}$. Clearly, $B \sqsubseteq B_{\mathrm{r}//} \sqsubseteq B' \equiv B$. Thus $B \equiv B_{\mathrm{r}//}$, as claimed. A similar argument shows that every branch of $V^{\geq k}$ remains equivalent if its uppermost edge is replaced with a descendant edge. This proves Parts 1 and 2. From these two parts, we easily get Parts 3 and 4. □

We also need the following lemma.

LEMMA 4.13. $R_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//} \equiv P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$.

PROOF. We first show that $R_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//} \sqsubseteq P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$. Let e be an embedding of $R_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$ in a tree t, such that $e(out(R)) = o$. It is enough to show an embedding $\tilde{e}$ of $P^{\geq k}_{\mathrm{r}//}$ in t, such that $\tilde{e}(out(P)) = o$. From Part 4 of Lemma 4.12, we conclude that there exists an embedding e' of $R \circ V^{\geq k}$ in t, such that $e'(out(R)) = o$. Since $P^{\geq k} \equiv^{w} R \circ V^{\geq k}$, it follows that there is a weak embedding e'' of $P^{\geq k}$ in t, such that $e''(out(P)) = o$. Thus, we obtain $\tilde{e}$ from e'' by simply mapping $root(P^{\geq k})$ to $root(t)$. $\tilde{e}$ is a legal embedding since all the outgoing edges of $root(P^{\geq k}_{\mathrm{r}//})$ are of descendant type.

We now prove that $P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//} \sqsubseteq R_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$. Let t be a canonical model of $P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$ and o be the node of t that corresponds to the output node of P. We need to find an embedding e of $R_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$ in t, such that $e(out(R)) = o$. Let t_s be obtained from t by shortening each of the paths that correspond to the outgoing edges of the root of $P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$ to one edge. Thus, we obtain a canonical model of $P^{\geq k} \circ V^{\geq k}$. In particular, there is an embedding e' of $P^{\geq k}$ in t_s, such that $e'(out(P)) = o$. It follows that there is a weak embedding e'' of $R \circ V^{\geq k}$ in t_s, such that $e''(out(R)) = o$. Hence, there is an embedding e of $R_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$ in t_s, such that $e(out(R)) = o$. Obviously, e is an embedding of $R_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$ in t (since all the outgoing edges of the root are descendant), as required. □

Finally, the following lemma completes the proof of Theorem 4.10.

LEMMA 4.14. $P^{\geq k}_{\mathrm{r}//}$ *is a potential rewriting.*

PROOF. We need to show that $R \circ V \equiv P^{\geq k}_{\mathrm{r}//} \circ V$. For that, we prove that $R \circ V^{\geq k} \equiv P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}$. Part 4 of Lemma 4.12 shows that $R \circ V^{\geq k} \equiv R_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$. Then, from Lemma 4.13, we get that $R_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//} \equiv P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//}$. Finally, from Part 3 of Lemma 4.12 we have $P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}_{\mathrm{r}//} \equiv P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}$. We conclude that $R \circ V^{\geq k} \equiv P^{\geq k}_{\mathrm{r}//} \circ V^{\geq k}$, as claimed. □

We conclude this section with the following corollary of Theorems 4.9 and 4.10.

COROLLARY 4.15. *The rewriting-existence problem is coNP-complete under each of the following assumptions.*

1. *A descendant edge enters the output node of V.*
2. *The selection path of V does not have descendant edges.*

4.1.3 Correlation Between the Query and the View

We have shown that there is a potential rewriting among the natural candidates in each of the following cases. First, the selection path of either $P^{\leq k}$ or V has only child edges. Second, a descendant edge enters the output node of V. If neither case holds, then we can still get a sufficient condition for completeness by considering the last descendant edge on the selection path of P, namely, the one that is closest to the output node.

Consider two edges e_1 and e_2 that appear on the selection paths of P and V, respectively. We say that e_1 and e_2 are *corresponding* selection edges if they appear at the same depth, namely, for some $1 \leq i \leq k$, both connect the $(i-1)$-node to the i-node.

The following theorem shows that $P^{\geq k}$ is a potential rewriting if the last descendant edge on the selection path of P corresponds to a descendant edge of V. This result is the basis of an extension that is described in the next section. An important element in the proof is showing that if the rewriting R has a descendant edge e on the selection path, then the following holds. Consider the part of the selection path of $R \circ V$ between the edge of V that corresponds to the last descendant edge of P and the edge e. All the branches of $R \circ V$ that emanate from this part of the selection path are redundant.

THEOREM 4.16. *Let P be a pattern and let V be a view, such that the last descendant edge on the selection path of P corresponds to a descendant edge on the selection path of V. Then, the following hold.*

- $P^{\geq k}$ *is a potential rewriting.*

- *The rewriting-existence problem is coNP-complete.*

PROOF. Suppose that R is a rewriting of P using V. Let j be the depth of the node of P into which the last descendant edge of the selection path of P enters. Observe that if $j = k$, then this case has been previously solved (a descendant edge enters the output node of V). So, we assume that $j < k$. The node at depth j of P is denoted by n_j^p and that at depth j of V is denoted by n_j^v.

We first prove that $R \circ V \sqsubseteq P^{\geq k} \circ V$. Consider a tree t and let e be an embedding of $R \circ V$ in t, such that $e(out(R)) = o$ (i.e., e produces t^o_Δ). Furthermore, assume that e is an embedding that maps n_j^v to the deepest node of t, among all the embeddings that produce t^o_Δ. Let $v_j = e(n_j^v)$ and $v_r = e(root(R))$. We denote by t_j and t_r the subtrees of t that are rooted at v_j and v_r, respectively. To show that an embedding of $P^{\geq k} \circ V$ in t produces t^o_Δ, it is enough to prove that an embedding of $P^{\geq k}$ in t_r produces ${t_r}^o_\Delta$. Since $P^{\geq j} \equiv^w (R \circ V)^{\geq j}$, there is a weak embedding e' of $P^{\geq j}$ in t_j, such that $e'(out(P)) = o$. However, e' must be an embedding of $P^{\geq j}$ in t_j, namely, e' maps $root(P^{\geq j})$ to $root(t_j)$, or else e does not satisfy the condition that it maps n_j^v as deeply as possible. Since $P^{\geq j}$ has only child edges on its selection path, we conclude that the depth of o in t_j is $d-j$. Therefore, e maps each edge of the selection path of $(R \circ V)^{\geq j}$ to a single edge of t_j. So, the path from v_j to v_r has $k-j$ edges. It follows that e' induces an embedding of $P^{\geq k}$ in t_r that produces ${t_r}^o_\Delta$, as required.

We now prove that $P^{\geq k} \circ V \sqsubseteq R \circ V$. If all of the selection edges of R are child edges, then the claim is obvious, as $(P^{\geq k} \circ V)^{\geq k} \equiv (R \circ V)^{\geq k}$. So assume that R has a descendant edge connecting u_{q-1} to u_q, where u_i is the i-node of $R \circ V$. Consider a canonical model t of $P^{\geq k} \circ V$ and let o be the node of t that corresponds to $out(P)$. Let t_j and t_q be the subtrees of t that are rooted at the nodes that correspond to the j-node and q-node of $P^{\geq k} \circ V$, respectively. To complete the proof, we next show that some weak embedding of $(R \circ V)^{\geq j}$ in t_j produces t^o_Δ.

Let t' be the subtree that is constructed by placing above t_q a canonical model $\hat{t}$ of $(R \circ V)^{\leq q-1}$. Moreover, the node of $\hat{t}$ corresponding to the $(q-1)$-node of $R \circ V$ is connected to the root of t_q by a long (e.g., of length $2d$) path of nodes that have a new label. Since there is a weak embedding of $(R \circ V)^{\geq q}$ in t_q, it follows that $t^o_\Delta \in (R \circ V)(t')$. Now consider the subtree t'' of t' that is induced by the $q - j$ suffix of the long path and t_q. Since $P \equiv R \circ V$, there is an embedding of $P^{\geq j}$ in t'' and, consequently, a weak embedding of $(R \circ V)^{\geq j}$ in t'' (both embeddings produce t^o_Δ). But t'' can be obtained from a subtree of t_j by removing branches and changing labels to the new one. It follows that there is a weak embedding e of $(R \circ V)^{\geq j}$ in t_j such that e produces t^o_Δ, as claimed. □

As an example, consider the patterns V, P_1 and P_2 of Figure 4. The condition of Theorem 4.16 is satisfied by V and P_1 since the last descendant edge on the selection path of P_1 is the second, and the second selection edge of V is also descendant. Observe that this condition is not satisfied in the case of V and P_3 since the first selection edge of V is a child edge. Also note that the last descendant edge in the selection path of P_2 is the fifth, so there is no corresponding edge of V. In the following section, we extend Theorem 4.16 to accommodate both P_2 and P_3.

5. REWRITING TECHNIQUES

In this section, we describe several techniques that can be used for extending results on rewriting, e.g., either those of [17] or the ones given in the previous section. These techniques are based on the following approach. Given a pattern P and a view V, we create a new pattern P' and a new view V'. We show that if a rewriting of P using V exists, then a rewriting of P' using V' can be transformed into a rewriting of P using V and vice versa. This is useful, because P' and V' are more likely to fall in the resolved cases. In addition, if P' and V' satisfy one of the conditions of the previous section and none of the natural candidates w.r.t. P' and V' is a rewriting, then there is no rewriting of P using V.

We actually show how to bring about new, easily-described syntactic conditions, which guarantee that at least one natural candidates is a potential rewriting. This is done by combining the techniques of this section with results of the previous section. As usual, the pattern P and the view V have depths d and k, respectively, and $d > k$.

5.1 Utilizing Stability

The first technique is a reduction of the original P to a stable sub-pattern of P. More precisely, the following proposition shows that it is enough to solve the problem for a stable sub-pattern of P and the corresponding sub-pattern of V (provided that both exist).

PROPOSITION 5.1. *Suppose that there is a rewriting of P using V, and that $P^{\geq i}$ is stable for some $0 \leq i \leq k$. Then a pattern R' is a rewriting of P using V if and only if it is a rewriting of $P^{\geq i}$ using $V^{\geq i}$.*

PROOF. Suppose that R' is a rewriting of P using V. By Part 2 of Proposition 3.1, the patterns $(R' \circ V)^{\geq i}$ and $P^{\geq i}$ are weakly equivalent. Hence, they are equivalent, because the latter is stable. Since $(R' \circ V)^{\geq i} \equiv R' \circ (V^{\geq i})$ holds, the claim follows.

Now, suppose that $R' \circ V^{\geq i} \equiv P^{\geq i}$. Let R be a rewriting of P using V, that is, $P \equiv R \circ V$. Part 2 of Proposition 3.1 and the stability of $P^{\geq i}$ imply that $(R \circ V)^{\geq i} \equiv P^{\geq i}$. Consequently, $(R \circ V)^{\geq i} \equiv R' \circ V^{\geq i}$. This means that in $R \circ V$, we can replace the sub-pattern $(R \circ V)^{\geq i}$ with $R' \circ V^{\geq i}$ while preserving equivalence. In this way we get $R' \circ V$. Therefore, R' is a rewriting of P using V, as claimed. □

By combining Proposition 5.1, Theorem 4.4, Theorem 4.10 and Proposition 4.1, we get the following corollary.

COROLLARY 5.2. *Let $0 \leq i \leq k$. If the i-node on the selection path of P (resp. V) is not labeled with $*$ and only child edges connect it to the k-node of P (resp. V), then at least one of the natural candidates is a potential rewriting. Furthermore, in this case the rewriting-existence problem is coNP-complete.*

Next, we use Proposition 5.1 in proving that some natural candidate is a potential rewriting if the pattern P is in the normal form $\text{GNF}_{/*}$, which is a generalization of the normal form $\text{NF}_{/*}$ introduced in [10] (in particular, every pattern in $\text{NF}_{/*}$ is also in $\text{GNF}_{/*}$, but not necessarily vice versa). In the following definition of $\text{GNF}_{/*}$, note that a pattern is *linear* if it forms a path; that is, each node has at most one child.

DEFINITION 5.3 (GENERALIZED NORMAL FORM-GNF). *Consider a pattern Q of depth d. We say that Q is in $GNF_{/*}$ if for all $1 \leq i \leq d$, at least one of the following holds.*

1. *A child edge enters the i-node of Q.*
2. *$Q^{\geq i}$ is stable.*
3. *$Q^{\geq i}$ is linear.*

THEOREM 5.4. *If P is in $GNF_{/*}$, then at least one of the natural candidates is a potential rewriting.*

PROOF. Consider the maximal $1 \leq i \leq k$, such that $P^{\geq i}$ is stable; if there is no such i, then let $i = 0$. If $i = k$, then the claim follows immediately from Theorem 4.3. So, we assume that $i < k$. If the selection path of P has only child edges between the i-node and the k-node, then Proposition 5.1 and Theorem 4.4 imply the claim. It remains to deal with the case that for some $i < j \leq k$, a descendant edge enters the j-node of P. We consider the smallest j that has this property. By Proposition 4.1, the maximality of i implies that $*$ is the only label that appears on the path from the j-node to the k-node. By the definition of $\text{GNF}_{/*}$ and the properties of i and j, we get that $P^{\geq j}$ is linear. Consequently, applying the following transformation to P produces an equivalent pattern P'. We replace all the descendant edges between the $(j-1)$-node and the k-node with child edges, and relax the outgoing edge of the k-node (namely, it becomes a descendent edge). Note that the k-node has a single outgoing edge, because $P^{\geq j}$ is linear. This transformation preserves also the equivalence of $P^{\geq i}$ and $P'^{\geq i}$. Hence, $P'^{\geq i}$ is stable. In addition, the selection path of $P'^{\geq i}$ has only child edges. Thus, the claim is proven by Proposition 5.1, Theorem 4.4, the equivalence of P and P', and the fact that the natural candidate $P'^{\geq k}$ of P' is the same as the natural candidate $P^{\geq k}_{\text{r}//}$ of P. □

5.2 Ignoring All-But-Last Descendant Edges

Thus far we have dealt with descendant edges on the selection path of V if one of those either enters the output node of V or corresponds to the last descendent edge on the selection path of P. In this section, we show how to ignore the part of V (and the corresponding part of P) above the last descendant edge on the selection path of V. First, we give a few definitions.

Consider a pattern Q. The *depth* of a selection edge (m, n) of Q is the same as that of n. Now, let l be a label. We construct the pattern $l//Q$ by creating a new root r that is labeled with l, and connecting r to the root of Q with a descendant edge. The following proposition is quite straightforward.

PROPOSITION 5.5. *Let P_1 and P_2 be two patterns such that $P_1 \equiv^w P_2$. Then $l//P_1 \equiv l//P_2$ for all $l \in \Sigma \cup \{*\}$.*

Using Proposition 5.5, the following is shown.

PROPOSITION 5.6. *Let i be the maximal depth of a descendant edge on the selection path of V. Then:*

1. *If R is a rewriting of P using V, then R is a rewriting of $*//P^{\geq i}$ using $*//V^{\geq i}$.*
2. *If R' is a rewriting of $*//P^{\geq i}$ using $*//V^{\geq i}$, then R' is a potential rewriting w.r.t. P and V (i.e., it is a rewriting if there is one).*

PROOF. (Proof of 1.) From $R \circ V \equiv P$ and Proposition 3.1(2) we get $(R \circ V)^{\geq i} \equiv^w P^{\geq i}$. Since $(R \circ V)^{\geq i}$ is the same as $R \circ V^{\geq i}$, we get $R \circ V^{\geq i} \equiv^w P^{\geq i}$. Now from Proposition 5.5 we know that $*//(R \circ V^{\geq i}) \equiv *//P^{\geq i}$. Note that the left part of this equivalence is the same as $R \circ (*//V^{\geq i})$. Therefore, we get that R is a rewriting of $*//P^{\geq i}$ using $*//V^{\geq i}$.

(Proof of 2.) Since R' is a rewriting of $*//P^{\geq i}$ using $*//V^{\geq i}$, then $R' \circ (*//V^{\geq i}) \equiv *//P^{\geq i}$. Note that $R' \circ (*//V^{\geq i})$ can be written as $*//(R' \circ V^{\geq i})$. Therefore, we conclude that $*//(R' \circ V^{\geq i}) \equiv *//P^{\geq i}$. Applying Proposition 3.1(2) to this equivalence we get

$$(R' \circ V^{\geq i}) \equiv^w P^{\geq i}. \tag{6}$$

Since the edge that enters the i-th node of V is a descendant edge, V can be written as $V^{<i} \overset{i-1}{\Longrightarrow} V^{\geq i}$. Combining this with $R \circ V \equiv P$, we get $R \circ (V^{<i} \overset{i-1}{\Longrightarrow} V^{\geq i}) \equiv P$. It is, however, easy to see that the left part of the equivalence is identical to $V^{<i} \overset{i-1}{\Longrightarrow} (R \circ V^{\geq i})$; hence

$$V^{<i} \overset{i-1}{\Longrightarrow} (R \circ V^{\geq i}) \equiv P. \tag{7}$$

Now, by applying Proposition 3.1(2) to (7) we get $(V^{<i} \overset{i-1}{\Longrightarrow} (R \circ V^{\geq i}))^{\geq i} \equiv^w P^{\geq i}$ and because the left part of this weak equivalence is identical to $R \circ V^{\geq i}$, we get

$$R \circ V^{\geq i} \equiv^w P^{\geq i}. \tag{8}$$

Because of (8) and by using Proposition 3.2, we can replace $R \circ V^{\geq i}$ by $P^{\geq i}$ in the left side of (7) and obtain the equivalent pattern $V^{<i} \overset{i-1}{\Longrightarrow} P^{\geq i}$. Therefore, (7) becomes:

$$V^{<i} \overset{i-1}{\Longrightarrow} P^{\geq i} \equiv P. \tag{9}$$

Because of (6), by applying Proposition 3.2 on (9) we can replace $P^{\geq i}$ by $R' \circ V^{\geq i}$ obtaining the equivalent pattern

$$V^{<i} \overset{i-1}{\Longrightarrow} (R' \circ V^{\geq i}) \equiv P. \tag{10}$$

Since $V^{<i} \overset{i-1}{\Longrightarrow} (R' \circ V^{\geq i})$ can be rewritten as $R' \circ (V^{<i} \overset{i-1}{\Longrightarrow} V^{\geq i})$ or equivalently as $R' \circ V$, we derive $R' \circ V \equiv P$. □

Proposition 5.6 and Theorem 4.16 immediately imply the following extension of the latter.

COROLLARY 5.7. *If the deepest descendant edge on the selection path of V is at least as deep as the deepest descendant edge on the selection path of P, then $P^{\geq k}$ is a potential rewriting. Furthermore, in this case the rewriting-existence problem is coNP-complete.*

As an example, V and P_3 (but not P_2) of Figure 4 satisfy the conditions of Corollary 5.7. Consequently, $P_3^{\geq 3}$ is a potential rewriting.

5.3 Pattern Extension and Output Lifting

Consider a pattern Q and let l be a label. The *l-extension* of Q, denoted by Q^{+l}, is obtained by adding new nodes that are connected by child edges as follows. We add a child labeled with l to $out(Q)$, and a child labeled with $*$ to each leaf of Q; if $out(Q)$ is a leaf, it only gets the child labeled with l. For example, see the patterns V, P_2, V^{+*} and $P_2^{+\mu}$ of Figure 4. Now, suppose that the depth of Q is h. For $0 \leq j \leq h$, the pattern $Q^{j\rightarrow}$ is the same as Q, except that

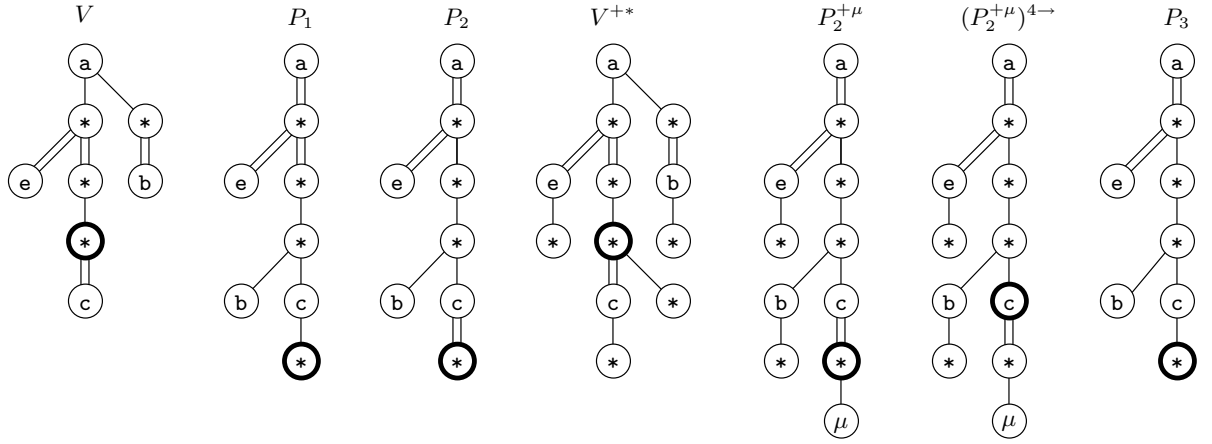

Figure 4: Correlation, label extension and output lifting

the output node is the j-node (instead of the h-node). For example, $Q^{h\rightarrow}$ is Q itself, and in $Q^{(h-1)\rightarrow}$ the output node is the parent of $out(Q)$. As another example, see the pattern $(P_2^{+\mu})^{4\rightarrow}$ of Figure 4. In the remainder of this section, we assume that μ is a label that appears in none of the patterns at hand; in particular, in neither P nor V. The following proposition is rather straightforward.

PROPOSITION 5.8. *Let P_1 and P_2 be two patterns. Then, $P_1 \equiv P_2$ if and only if $P_1^{+\mu} \equiv P_2^{+\mu}$.*

We now consider the following transformation that is applicable if for some $k \leq j \leq d$, the j-node of P has a non-$*$ label. If so, we first extend P and V with the labels μ and $*$, respectively, and then define the j-node as the new output node. Thus, we actually generate a new pattern $P' = (P^{+\mu})^{j\rightarrow}$ and a new view $V' = V^{+*}$. The next theorem shows that this transformation preserves existence (and nonexistence) of rewritings. Moreover, it shows that a rewriting R of P using V can be easily obtained from the one found for P' and V'.

THEOREM 5.9. *Let P, V and R be patterns. Suppose that for some $k \leq j \leq d$, the label of the j-node of P is not $*$. Then, R is a rewriting of P using V if and only if $(R^{+\mu})^{(j-k)\rightarrow}$ is a rewriting of $(P^{+\mu})^{j\rightarrow}$ using V^{+*}.*

PROOF. Let $R' = (R^{+\mu})^{(j-k)\rightarrow}$, $P' = (P^{+\mu})^{j\rightarrow}$ and $V' = V^{+*}$. We start with the "only if" direction. We assume that R is a rewriting of P using V and we need to prove that $R' \circ V' \equiv P'$. We first prove that $R' \circ V' \sqsubseteq P'$. Consider a canonical model t' of $R' \circ V'$. Let t be obtained from t' by pruning the leaves. Clearly, t is a canonical model of $R \circ V$. We denote by v_i the node of t that corresponds to the i-node of $R \circ V$; in particular, v_d corresponds to $out(R)$. The equivalence $R \circ V \equiv P$ implies that there is an embedding e of P in t that maps $out(P)$ to v_d. By Part 3 of Proposition 3.1, $R \circ V$ and P have the same number of nodes with non-$*$ labels on their selection paths. Therefore, the embedding e must map the j-node of P to v_j (recall that the j-node has a non-$*$ label). Thus, we can extend e to an embedding e' of P' in t', such that e' maps the j-node to v_j, as required. The proof for the other direction, $P' \sqsubseteq R' \circ V'$, is similar.

We now prove the "if" direction. For that, we assume that R' is a rewriting of P' using V' and we need to show that $R \circ V \equiv P$. We first prove that $R \circ V \sqsubseteq P$. Consider a canonical model t of $R \circ V$, and let o_r and o_v be the nodes of t that correspond to $out(R)$ and $out(V)$, respectively. Let t' be obtained from t by adding a child with the label $\perp$ to o_v and to each leaf (other than o_r), and a child with the label μ to o_r. Then t' is a canonical model of $R' \circ V'$. Consequently, there is an embedding e' of P' in t'. The embedding e' must map $out(P)$ to o_r, because o_r is the only node having a child labeled with μ. The embedding e' induces a mapping e of P in t, such that e produces $t_\Delta^{o_r}$. The proof of $P \sqsubseteq R \circ V$ is similar. □

Theorem 5.9 shows that if a label of Σ appears on the selection path of P between depth k and depth d, then the following can be done. In order to find a rewriting of P using V (or deciding that none exists), it is sufficient to look for a rewriting R' of $(P^{+\mu})^{j\rightarrow}$ using V^{+*}, such that R' has the form $(R^{+\mu})^{(j-k)\rightarrow}$ for some pattern R. If such R' is found, then the pattern R is a rewriting of P using V. The next proposition shows that R is a natural candidate if and only if $(R^{+\mu})^{(j-k)\rightarrow}$ is so. The proof is rather straightforward and therefore omitted.

PROPOSITION 5.10. *Let P, V and R be patterns and suppose that for some $k \leq j \leq d$, the j-node of P has a non-$*$ label. Then, R is a natural candidate w.r.t. P and V if and only if $(R^{+\mu})^{(j-k)\rightarrow}$ is a natural candidate w.r.t. $(P^{+\mu})^{j\rightarrow}$ and V^{+*}.*

From Theorem 5.9 and Proposition 5.10, we conclude the following corollary.

COROLLARY 5.11. *Let P and V be patterns and suppose that for some $k \leq j \leq d$, the j-node of P has a non-$*$ label. Then the following hold.*

- *There is a rewriting of P using V if and only if there is a rewriting of $(P^{+\mu})^{j\rightarrow}$ using V^{+*}.*
- *$(P^{+\mu})^{j\rightarrow}$ and V^{+*} have a rewriting among the natural candidates if and only if so do P and V.*

From Corollary 5.11, we conclude that the technique of this section is useful not just for finding a rewriting R, but also to prove that the natural candidates w.r.t. P and V

contain a potential rewriting. In particular, if the results of the previous sections are applicable to $(P^{+\mu})^{j\rightarrow}$ and V^{+*}, then we can use them for P and V. As an example, we can generalize Corollary 5.7 as follows. For the purpose of deciding whether the condition of the corollary holds, we can ignore every descendant edge $e = (m, n)$ on the selection path of P below the k-node, provided that a label other than $*$ appears (at least once) between the k-node and m. Consider, for instance, the patterns V and P_2 of Figure 4. By ignoring the descendant edge of P_2 below the label c, we get that $P_2^{\leq 3}$ is a potential rewriting.

6. CONCLUSION

In this work, we have studied the rewriting problem in a widely used fragment of XPath. The problem was known to be coNP-hard, but there was no upper bound. We have shown that for large sub-fragments, the problem is coNP-complete. These are practical results because the input size is typically very small. Moreover, our results cover most of the queries and views that are used in real-world scenarios. To be convinced of this point, one should realize that it is not easy to contrive meaningful queries and views that can "beat" all our methods.

To prove our results, we have developed new techniques for reasoning about patterns of $\mathsf{XP}^{\{//,[],*\}}$. We believe that these techniques will be useful for investigating other problems pertaining to $\mathsf{XP}^{\{//,[],*\}}$. These techniques are not based on query minimization and furthermore they do not get an inspiration from techniques in [10]. In particular, it is not known whether a non-redundant XPath query in $\mathsf{XP}^{\{//,[],*\}}$ is also minimal. The work in [10] shows that for two normal forms, this property holds (namely, a non-redundant query is also minimal). But even when this property holds, it only yields a Σ_2^p upper bound for the rewriting problem, while in this work we give coNP-complete results. Moreover, the generalized normal form presented in Section 5.1 covers a much larger class of queries than the corresponding normal forms presented in [10] because it is based only on properties of the selection path (rather than the whole query); hence, the generalized normal form covers many queries for which it is not known whether minimization is the same as non-redundancy.

Quite a few problems remain open. First, finding the exact complexity of the general case or, at least, a better upper bound than our plain decidability result. Second, is there an example where none of the possible rewritings is a natural candidate? Third, is it possible to extend our results to the problem of maximally contained rewritings? Fourth, given a set of queries that are frequently asked, what is an optimal set of views that should be maintained so that the queries could be evaluated as quickly as possible? Naturally, this problem is inherently related to caching on the World-Wide Web. Fifth, formulating and solving the problem of rewriting a query using multiple views.

Acknowledgments

We thank the anonymous referees for valuable comments.

7. REFERENCES

[1] F. N. Afrati, C. Li, and P. Mitra. Rewriting queries using views in the presence of arithmetic comparisons. *Theor. Comput. Sci.*, 368(1-2):88–123, 2006.

[2] F. N. Afrati, C. Li, and J. D. Ullman. Using views to generate efficient evaluation plans for queries. *J. Comput. Syst. Sci.*, 73(5):703–724, 2007.

[3] A. Balmin, F. Özcan, K. S. Beyer, R. Cochrane, and H. Pirahesh. A framework for using materialized XPath views in XML query processing. In *VLDB*, pages 60–71. Morgan Kaufmann, 2004.

[4] D. Calvanese, G. D. Giacomo, M. Lenzerini, and M. Y. Vardi. Answering regular path queries using views. In *ICDE*, pages 389–398, 2000.

[5] L. Chen and E. A. Rundensteiner. XCache: XQuery-based caching system. In *WebDB*, pages 31–36, 2002.

[6] S. Cohen, W. Nutt, and A. Serebrenik. Rewriting aggregate queries using views. In *PODS*, pages 155–166. ACM, 1999.

[7] A. Deutsch and V. Tannen. Reformulation of XML queries and constraints. In *ICDT*, pages 225–241. Springer, 2003.

[8] S. Flesca, F. Furfaro, and E. Masciari. On the minimization of Xpath queries. In *VLDB*, pages 153–164, 2003.

[9] G. Grahne and A. Thomo. Query containment and rewriting using views for regular path queries under constraints. In *PODS*, pages 111–122. ACM, 2003.

[10] B. Kimelfeld and Y. Sagiv. Revisiting redundancy and minimization in an XPath fragment. In *EDBT*, pages 61–72. ACM, 2008.

[11] L. V. S. Lakshmanan, H. Wang, and Z. Zhao. Answering tree pattern queries using views. In *VLDB*, pages 571–582. ACM, 2006.

[12] A. Y. Levy, A. O. Mendelzon, Y. Sagiv, and D. Srivastava. Answering queries using views. In *PODS*, pages 95–104. ACM, 1995.

[13] B. Mandhani and D. Suciu. Query caching and view selection for XML databases. In *VLDB*, pages 469–480. ACM, 2005.

[14] G. Miklau and D. Suciu. Containment and equivalence for a fragment of XPath. *J. ACM*, 51(1):2–45, 2004.

[15] Y. Papakonstantinou and V. Vassalos. Query rewriting for semistructured data. In *SIGMOD Conference*, pages 455–466. ACM, 1999.

[16] J. D. Ullman. Information integration using logical views. *Theor. Comput. Sci.*, 239(2):189–210, 2000.

[17] W. Xu and Z. M. Özsoyoglu. Rewriting XPath queries using materialized views. In *VLDB*, pages 121–132, 2005.

[18] L. H. Yang, M. L. Lee, and W. Hsu. Efficient mining of XML query patterns for caching. In *VLDB*, pages 69–80, 2003.