A Compact Representation for Least Common Subsumers in the Description Logic $\mathcal{ALE}$

Chan Le Duc [a,*], Nhan Le Thanh [b], Marie-Christine Rousset [c]
[a] *LSIS (UMR-CNRS 6186), Université du Sud Toulon-Var, Bât. R, BP.20132, 83957 La Garde - Cedex, France*
E-mail: {chan.leduc}@univ-tln.fr
[b] *Laboratoire I3S, UNSA-CNRS, 2000 route des lucioles, Les Algorithmes - bât. Euclide B, BP.121, 06903 Sophia Antipolis - Cedex, France*
E-mail: {nhan.le-thanh}@unice.fr
[c] *LSR-IMAG, BP 72, 38402 St Martin d'Heres - Cedex, France*
E-mail: {marie-christine.rousset}@imag.fr

Abstract
This paper introduces a compact representation which helps to avoid the exponential blow-up in space of the Least Common Subsumer (lcs) of two $\mathcal{ALE}$-concept descriptions. Based on the compact representation we define a space of specific graphs which represents all $\mathcal{ALE}$-concept descriptions including the *lcs*. Next, we propose an algorithm exponential in time and polynomial in space for deciding subsumption between concept descriptions represented by graphs in this space. These results provide better understanding of the double exponential blow-up of the approximation of $\mathcal{ALC}$-concept descriptions by $\mathcal{ALE}$-concept descriptions: double exponential size of the approximation in the ordinary representation is unavoidable in the worst case.
Keywords: Description Logics, Least Common Subsumer, Approximation.

1. Introduction

Description logics can be used as a formalism for representing ontologies. The OWL language [10], which is becoming a standard language for ontologies, is founded on description logics. If we ignore role constructors and general concept inclusions, the OWL-Lite and OWL-DL [10] languages are respectively comparable to the $\mathcal{ALE}$ and $\mathcal{ALC}$ description logics. The major difference between $\mathcal{ALE}$ and $\mathcal{ALC}$ is that the disjunction constructor is absent from $\mathcal{ALE}$. As a result, deciding subsumption in $\mathcal{ALE}$ is NP-complete [6] while it is PSPACE-complete for $\mathcal{ALC}$ [7].

In this paper, we revisit two problems which have been recently addressed in description logics: the computation of the least common subsumer (*lcs*) of two concept descriptions and the approximation from a description logic L_1 to a description logic L_2.

As mentioned in recent work [3], computing the lcs is a useful inference task for the bottom-up construction of knowledge bases in description logics. It also can be used for computing similarity between concept description of different ontologies. Finally, it plays a central role for computing approximations. An algorithm for computing the approximation where $L_1 = \mathcal{ALC}$ and $L_2 = \mathcal{ALE}$ is presented in [1]. It returns a concept description whose size may be double exponential in the size of the input. This algorithm is based on an exponential algorithm which computes the *lcs* of concept descriptions in $\mathcal{ALE}$. As shown in [4], the exponential size of *lcs* cannot be avoided if we use the ordinary representation of *normalized* concept descriptions, whose size may be exponential compared to the initial (not normalized) concept descriptions.

Some recent results extend those presented in [3] and [1] to more expressive description logics. For

*Corresponding author: Chan Le Duc, LSIS (UMR-CNRS 6186), Université du Sud Toulon-Var, Bât. R, BP.20132, 83957 La Garde - Cedex, France

AI Communications
ISSN 0921-7126, IOS Press.

instance a double exponential algorithm for computing *lcs* in $\mathcal{ALEN}$ is presented in [11]. It yields a double exponential upper bound for the size of the *lcs* of two $\mathcal{ALEN}$-concept descriptions. Another result described in [12] provides an algorithm for computing lcs in $\mathcal{FLE}+$ ($\mathcal{FLE}$ with transitive roles). Nevertheless, the complexity of this algorithm is not given.

The first contribution of this paper is a compact representation for $\mathcal{ALE}$-concept descriptions which avoids the exponential blow-up of the size of the description trees built from *normalized* concept descriptions as in [3]. This (polynomial) compact representation is a graph, which is directly built from the description trees obtained from the *weakly normalized* concept descriptions. A *weakly normalized* concept description is obtained by applying the normalization rules presented in [3] except for the normalization rule responsible of the exponential blow-up of the size of the normalized concept description (and thus of the resulting description tree). This new representation of a weakly normalized concept description is called its *ϵ-tree* because it replaces the effective application of the expansive normalization rule by adding what we have called *ϵ-edges* to the description trees corresponding to the *weakly normalized* concept descriptions. Then, *normalization graphs* are built from *ϵ-trees* by making explicit bottom-concepts in labels of nodes in *ϵ-trees* (as $\mathcal{ALE}$ allows for the bottom-concept and negated concept names). We also obtain a polynomial compact representation of the *lcs* of two $\mathcal{ALE}$-concept descriptions by defining the *product* of two normalization graphs. Finally, we exploit this compact representation for providing an algorithm for checking subsumption between concept descriptions or *lcs* of concept descriptions in polynomial space and exponential time.

The second contribution of this paper is to show that the lower bound for computing the approximation of $\mathcal{ALC}$-concept descriptions by $\mathcal{ALE}$-concept descriptions is double exponential in the size of the input. This result answers *partially*[1] the question left open in [1] on the existence of an exponential algorithm for the approximation of $\mathcal{ALC}$ by $\mathcal{ALE}$. It also shows the limit of the compact representation that we have introduced: it cannot prevent the exponential blow-up resulting from n-ary *lcs* computation.

[1] This will be discussed in Section 6

The paper is organized as follows.

In Section 2, we provide the formal background on which this paper is based. In particular, we distinguish the weak normal form and the strong normal form of a concept description, depending on the normalization rules that are applied. We also recall the definition of description trees introduced in [3].

In Section 3, we define the ϵ-tree of a weakly normalized concept description, and the normalization graph resulting from the ϵ-tree. Next, we provide the transformation algorithm from a normalization graph into the description tree of the corresponding *normalized* concept description. We also provide a polynomial space (and exponential time) algorithm exploiting the normalization graphs for checking subsumption in $\mathcal{ALE}$.

In Section 4, we define the product of normalization graphs and we show how it can be exploited for computing a polynomial representation of the *lcs* of two $\mathcal{ALE}$-concept descriptions. We also show that the algorithms introduced in Section 3 for normalization graphs can be extended to their products.

In Section 5, we prove that the lower bound of the size of the approximation of an $\mathcal{ALC}$- concept description by an $\mathcal{ALE}$-concept description in the compact graph representation (and a fortiori in the ordinary representation) is double exponential in the size of the input. The reason is that the double exponential blow-up is due to the computation of the n-ary *lcs*, and the compact graph representation that we have introduced in this paper cannot prevent the exponential blow-up resulting from the computation of the n-ary *lcs*.

Finally, Section 6 concludes and provides a brief discussion on the results obtained in this paper.

2. Formal background

In this section we will briefly present important notions of description logics and existing results about the *lcs* computation. Details of this formalism can be found in [9]. Let N_C be a set of primitive concepts and N_R be a set of primitive roles. The logic $\mathcal{FLE}$ uses the following constructors to build concept descriptions : conjunction ($C \sqcap D$), value restriction $\forall r.C$, existential restriction $\exists r.C$ and the top-concept $\top$. The logic $\mathcal{ALE}$ is extended from $\mathcal{FLE}$ by further adding primitive negation

$\neg P$ and the bottom-concept $\bot$. The logic $\mathcal{ALC}$ is extended from $\mathcal{ALE}$ by allowing for disjunction ($C \sqcup D$). Let $\Delta^{\mathcal{I}}$ be a non-empty set of individuals. Let $.^{\mathcal{I}}$ be a function that transforms each primitive concept $P \in N_C$ into $P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and each primitive role $r \in N_R$ into $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The semantics of a concept description are inductively defined owing to the interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, .^{\mathcal{I}})$ as in the table below.

Syntax	Semantics
$\top$	$\Delta^{\mathcal{I}}$
$\bot$	$\emptyset$
$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
$\forall r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \forall y{:}(x,y) \in r^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$
$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
$\exists r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y{:}(x,y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$

- **Subsumption**. Let C, D be concept descriptions. C subsumes D, $C \sqsubseteq D$, iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for all interpretation $\mathcal{I}$.
- **Least Common Subsumer**. Let C_1, C_2 be concept descriptions in a description logic. C is a least common subsumer of C_1, C_2 ($lcs(C_1, C_2)$ for short) iff $C_i \sqsubseteq C$ for all $i \in \{1, 2\}$, and if C' is a concept description such that $C_i \sqsubseteq C'$ for all $i \in \{1, 2\}$, then $C \sqsubseteq C'$.
- **Approximation**. Let C be a concept description in a L_1 and D be a concept description in a L_2 where L_1, L_2 are description logics. D is called upper L_2-approximation of C ($D = approx_{L_2}(C)$ for short) iff $C \sqsubseteq D$ and, if $C \sqsubseteq D'$ and $D' \sqsubseteq D$, then $D' \equiv D$ for all L_2-concept description D'.

The depth of an $\mathcal{ALE}$-concept description C is inductively defined as follows: i) $depth(P) = depth(\neg P) = depth(\top) = depth(\bot) := 0$ ($P \in N_C$); ii) $depth(C \sqcap D) := max(depth(C), depth(D))$; iii) $depth(\exists r.C) = depth(\forall r.C) := depth(C) + 1$.

Definition 1 (*$\mathcal{ALE}$-description tree*) [3] *Given a set N_C of primitive concepts and a set N_R of primitive roles. A description tree is of the form $\mathcal{G} = (V, E, v^0, l)$ where*

- *V is the set of nodes of $\mathcal{G}$;*
- *$E \subseteq V \times (N_R \cup \forall N_R) \times V$ is a finite set of edges labeled with role names r ($\exists$-edges) or with $\forall r$ ($\forall r$-edges); $\forall N_R := \{\forall r \mid r \in N_R\}$;*
- *v^0 is the root of $\mathcal{G}$;*
- *l is a labeling function mapping the nodes in V to finite set $\{P_1, ..., P_k\}$ where each P_i, $1 \leq i \leq k$, is one of the following forms : $P_i \in N_C$, $P_i = \neg P$ for some $P \in N_C$, or $P_i = \bot$. The empty label corresponds to the top-concept $\top$.*

In [3], the authors have proposed a procedure for transforming an $\mathcal{ALE}$-concept description C into the corresponding $\mathcal{ALE}$-description tree $\mathcal{G}(C) = (V, E, v^0, l)$ as follows. Every $\mathcal{ALE}$-concept description C can be written as $C = P_1 \sqcap ... \sqcap P_n \sqcap \exists r_1.C_1 \sqcap ... \sqcap \exists r_m.C_m \sqcap \forall s_1.D_1 \sqcap ... \sqcap \forall s_k.D_k$ where $P_i \in N_C \cup \{\neg P | P \in N_C\} \cup \{\top, \bot\}$. Then,
If $depth(C) = 0$ then $V := \{v^0\}$, $E := \emptyset$ and $l(v^0) := \{P_1, ..., P_n\} \setminus \{\top\}$.
If $depth(C) > 0$ then for $1 \leq i \leq m$, let $\mathcal{G}_i = (V_i, E_i, v_i^0, l_i)$ be the inductively defined $\mathcal{ALE}$-description tree corresponding to C_i, and for $1 \leq j \leq k$, let $\mathcal{G}'_j = (V'_j, E'_j, w_j^0, l'_j)$ be the inductively defined $\mathcal{ALE}$-description tree corresponding to D_j where V_i and V'_j are pairwise disjoint. Then,

- $V := \{v^0\} \cup V_i \cup V'_j$,
- $E := \{(v^0 r_i v_i^0) | 1 \leq i \leq m\} \cup \{(v^0 \forall s_j w_j^0) | 1 \leq j \leq k\} \cup \bigcup_{1 \leq i \leq m} E_i \cup \bigcup_{1 \leq j \leq k} E'_j$,
- $l(v) := \begin{cases} \{P_1, ..., P_n\} \setminus \{\top\}, v = v^0 \\ l_i(v), \quad v \in V_i, 1 \leq i \leq m \\ l'_j(v), \quad v \in V'_j, 1 \leq j \leq k \end{cases}$

Conversely, every $\mathcal{ALE}$-description tree $\mathcal{G}(C) = (V, E, v^0, l)$ can be transformed into an $\mathcal{ALE}$-concept description $C_{\mathcal{G}}$ as follows.
If $depth(\mathcal{G}(C)) = 0$ then $V = \{v^0\}$ and $E = \emptyset$. If $l(v^0) = \emptyset$ then $C_{\mathcal{G}} = \top$, otherwise, we have $l(v^0) = \{P_1, ..., P_n\}$, $n \geq 1$, $P_i \in N_C \cup \{\neg P | P \in N_C\} \cup \{\bot\}$ and define $C_{\mathcal{G}} := P_1 \sqcap ... \sqcap P_n$.
If $depth(\mathcal{G}(C)) > 0$ then $l(v^0) = \{P_1, ..., P_n\}$, $n \geq 0$, $P_i \in N_C \cup \{\neg P | P \in N_C\} \cup \{\bot\}$ and let $\{v_1, ..., v_m\}$ be the set of all successors of v^0 where $(v^0 r_i v_i) \in E$, $1 \leq i \leq m$ for some $r_i \in N_R$, and let $\{w_1, ..., w_k\}$ be the set of all successors of v^0 where $(v^0 \forall s_i w_i) \in E$, $1 \leq i \leq k$ for some $s_i \in N_R$. Furthermore, let $C_1, ..., C_m$ ($D_1, ..., D_k$) the inductively defined $\mathcal{ALE}$-concept descriptions corresponding to the subtrees of $\mathcal{G}$ with roots v_i, $1 \leq i \leq m$ (w_i, $1 \leq i \leq k$). We define $C_{\mathcal{G}} :=$

$P_1 \sqcap ... \sqcap P_n \sqcap \exists r_1.C_1 \sqcap ... \sqcap \exists r_m.C_m \sqcap \forall s_1.D_1 \sqcap ... \sqcap \forall s_k.D_k$.
The definition of the depth for $\mathcal{ALE}$-concept descriptions corresponds to the depth of its description tree. In addition, a node $v \in V$ of a description tree is called $\forall r$-successor (r-successor) if there exists an edge $(w \forall r v) \in E$ ($(w r v) \in E$). In this case, we also say that v is a $\forall r$-successor (r-successor) of w.

Definition 2 (*normalization rules*) [3] *The normal form of an $\mathcal{ALE}$-concept description C is obtained from C by exhaustively applying the following normalization rules:*

1. $\forall r.E \sqcap \forall r.F \rightarrow \forall r.(E \sqcap F)$
2. $\forall r.E \sqcap \exists r.F \rightarrow \forall r.\ E \sqcap \exists r.(E \sqcap F)$
3. $\forall r.\top \rightarrow \top$
4. $E \sqcap \top \rightarrow E$
5. $P \sqcap \neg P \rightarrow \bot$ for each $P \in N_C$
6. $\exists r.\bot \rightarrow \bot$
7. $E \sqcap \bot \rightarrow \bot$

where E, F are two $\mathcal{ALE}$-concept descriptions and $r \in N_R$.

Note that rules 3, 4 as specified in Definition 2 need to be applied once to $\mathcal{ALE}$-concept descriptions. However, the application of rule 2 (rule 1) can lead to the application of rules 1 (rule 2), 5, 6, 7 several times. The normalization of an $\mathcal{ALE}$-concept description C can be carried out in two steps. The first step consists of the application of all rules as specified in Definition 2 except for rule 2. This step yields an $\mathcal{ALE}$-concept description C' where $C' \equiv C$, whose each conjunction contains at most one value restriction (at the same depth as the conjunction). The second step consists of the application of rules 1, 2, 5, 6, 7 to the concept description C'. In the second step, rules 1, 2 need to be exhaustively applied once since the application of rules 5, 6, 7 does not lead to the application of rules 1, 2 again. The concept description obtained from the second step is in the normal form according to Definition 2.
From these remarks, we introduce weak and strong normal forms for $\mathcal{ALE}$-concept descriptions, corresponding to the two normalization steps described above.

Definition 3 *An $\mathcal{ALE}$-concept description C is in weak normal form if C is obtained from an $\mathcal{ALE}$-concept description by exhaustively applying all rules as specified in Definition 2, with the exception of rule 2.. An $\mathcal{ALE}$-concept description C' is in strong normal form if C' is obtained from an $\mathcal{ALE}$-concept description in weak normal form by applying exhaustively rules 1, 2, 5, 6, 7 as specified in Definition 2.*

It is obvious that the application of all rules (as specified in Definition 2) with the exception of rule 2, does not increase the size of concept descriptions. However, as shown in [4], the size of $\mathcal{ALE}$-concept descriptions in strong normal form may increase exponentially. This exponential blow-up in space is caused by the application of rule 2. Example 1, which is taken from [4], demonstrates this effect.

Example 1 *We define the following sequence C_1, C_2, C_3, ... of $\mathcal{ALE}$-concept descriptions.*

$$C_n := \begin{cases} \exists r.P \sqcap \exists r.Q & n = 1 \\ \exists r.P \sqcap \exists r.Q \sqcap \forall r.C_{n-1}, & n \geq 1 \end{cases}$$

For each $1 < k < n$, the application of rule 2. leads to copy C_{k-1} to the two existential restrictions $\exists r.P$ and $\exists r.Q$. This implies that for each $1 < k < n$, the two existential restrictions in the expression under the value restriction of C_{k-1} are inductively copied to each existential restriction $\exists r.P$ and $\exists r.Q$. Therefore, the normal form of C_n has at least 2^n existential restrictions.

Accordingly with the notation introduced in [3], we denote $\mathcal{G}_C$ the description tree obtained from a concept description C in strong normal form, and we denote $\mathcal{G}(C)$ the description tree obtained from a concept description C in weak normal form. We transfer the semantics of concept descriptions to description trees as follows: for an interpretation $(\Delta^{\mathcal{I}}, .^{\mathcal{I}})$, and a concept description C in strong (respectively weak) normal form: $\mathcal{G}_C^{\mathcal{I}} := C^{\mathcal{I}}$ (respectively $\mathcal{G}(C)^{\mathcal{I}} := C^{\mathcal{I}}$). Note that $C \equiv C_{\mathcal{G}_C}$ and $C \equiv C_{\mathcal{G}(C)}$ since the normalization rules preserve equivalence.
In addition, let $\mathcal{G} = (V, E, v^0, l)$ be an $\mathcal{ALE}$-description tree. We denote $\mathcal{G}(v_i) = (V_{\mathcal{G}(v_i)}, E_{\mathcal{G}(v_i)}, v_i, l_{\mathcal{G}(v_i)})$ as the subtree of $\mathcal{G}$ whose root is $v_i \in V$.

Definition 4 [3] (*homomorphism*) *A mapping $\varphi : V_H \rightarrow V_G$ from an $\mathcal{ALE}$-description tree $\mathcal{H} = (V_H, E_H, m^0, l_H)$ to an $\mathcal{ALE}$-description tree $\mathcal{G} = (V_G, E_G, n^0, l_G)$ is called homomorphism iff, the following conditions are satisfied:*

1. $\varphi(m^0) = n^0$;

2. *For all* $n \in V_H$*, we have* $l_H(n) \subseteq l_G(\varphi(n))$ *or* $l_G(\varphi(n))=\{\bot\}$*;*
3. *For all* $(nrm) \in E_H$*, either* $\varphi(n)r\varphi(m) \in E_G$*, or* $\varphi(n) = \varphi(m)$ *and* $l_G(\varphi(n))=\{\bot\}$*; and*
4. *For all* $(n\forall rm) \in E_H$*, either* $\varphi(n)\forall r\varphi(m) \in E_G$*, or* $\varphi(n) = \varphi(m)$ *and* $l_G(\varphi(n))=\{\bot\}$*.*

Additionally, if φ is a bijection and φ^{-1} is also a homomorphism from $\mathcal{G}$ to $\mathcal{H}$, then φ is called isomorphism.
Note that the existence of two homomorphisms: φ from $\mathcal{H}$ to $\mathcal{G}$ and φ' from $\mathcal{G}$ to $\mathcal{H}$, does not imply that there exists an isomorphism from $\mathcal{H}$ to $\mathcal{G}$. In general, it is not necessary that $\varphi' = \varphi^{-1}$.
A polynomial algorithm for checking the existence of a homomorphism between two $\mathcal{ALE}$-description trees has been proposed in [3]. Moreover, the authors have shown that the characterization of subsumption by homomorphisms requires that description trees must be built from $\mathcal{ALE}$-concept descriptions in strong normal form.

Theorem 1 [3] *Let* C*,* D *be* $\mathcal{ALE}$*-concept descriptions, then* $C \sqsubseteq D$ *if and only if there exists a homomorphism from* $\mathcal{G}_D$ *to* $\mathcal{G}_C$*.*

Therefore, if we use directly the algorithm in [3] for checking whether there exists a homomorphism between such two $\mathcal{ALE}$-description trees $\mathcal{G}_C$ and $\mathcal{G}_D$, it will take an exponential space in the worst case since the size of these trees may be exponential in the size of input concept descriptions.
Concerning the *lcs*, according to the work in [3], there always exists a *lcs* of $\mathcal{ALE}$-concept descriptions and it is unique. The computing of the *lcs* for two $\mathcal{ALE}$-concept descriptions C, D requires that C, D are in strong normal form. Next, the normalized concept descriptions have to be transformed into $\mathcal{ALE}$-description trees $\mathcal{G}_C$ and $\mathcal{G}_D$. The $\mathcal{ALE}$-description tree for the *lcs* will be the product tree of trees $\mathcal{G}_C$ and $\mathcal{G}_D$.

3. $\mathcal{E}$-tree and normalization graphs

This section introduces a specific data structure, called *normalization graph*, for representing strong normal $\mathcal{ALE}$-concept descriptions in polynomial space.
We first introduce ϵ-trees, denoted as $\mathcal{G}^\epsilon(C)$, which are built from description trees corresponding to $\mathcal{ALE}$-concept descriptions in weak normal form. This structure allows for substituting the application of rules 1, 2 (as specified in Definition 2) to $\mathcal{ALE}$-concept descriptions by adding ϵ-edges to the corresponding description trees. Normalization graphs, denoted as $\mathcal{G}^\epsilon_C$, are formed from ϵ-trees by adding some elements in order to capture rules 5, 6, 7 (as specified in Definition 2). Next, we provide an algorithm for transforming normalization graphs into $\mathcal{ALE}$-description trees. We will show that description trees obtained from normalization graphs by applying this algorithm are isomorphic to description trees built from $\mathcal{ALE}$-concept descriptions in strong normal form. We end this section by an algorithm for deciding subsumption between concept descriptions represented by normalization graphs.
We need the following notations. We denote N'_C as the union $N_C \cup \{\neg P \mid P \in N_C\} \cup \{\bot\}$. Let $\mathcal{G} = (V_G, E_G, v^0, l_G)$ be an $\mathcal{ALE}$-description tree. We denote $|\mathcal{G}|$ as the depth of $\mathcal{G}$ and v^k as a node at level k of $\mathcal{G}$ where $v^k \in V_G$. Hence, we can write $(v^k e v^{k+1}) \in E_G$ for all $0 \le k \le \mathcal{G}$. For the sake of simplicity, we can assume that $N_R = \{r\}$. All result obtained can be applied to a general set N_R.

Definition 5 (ϵ*-tree*) *Let* C *be an* $\mathcal{ALE}$*-concept description in weak normal form and* $\mathcal{G}(C) = (V, E, v^0, l)$ *be its description tree. The* ϵ*-tree* $\mathcal{G}^\epsilon(C) = (V,\ E \cup E^\epsilon,\ l)$ *is built from* $\mathcal{G}(C)$ *as follows.*

1. *For each* $v \in V$*, an* ϵ*-edge* $(v\epsilon v)$ *is added to* E^ϵ*.*
2. *For each level* k *where* $0 \le k \le |\mathcal{G}(C)| - 1$*, for each* ϵ*-edge* $(v^k_i \epsilon v^k_j)$ *at level* k*,*
   (a) *If there exist two edges* $(v^k_i \forall r v^{k+1}_i)$*,* $(v^k_j \forall r v^{k+1}_j) \in E$ *and* $v^{k+1}_i \neq v^{k+1}_j$*, then the* ϵ*-edge* $(v^{k+1}_i \epsilon v^{k+1}_j)$ *is added to* E^ϵ*.*
   (b) *If there exist two edges* $(v^k_i r v^{k+1}_i)$*,* $(v^k_j \forall r v^{k+1}_j) \in E$ *(or* $(v^k_i \forall r v^{k+1}_i)$*,* $(v^k_j r v^{k+1}_j) \in E$*) and* $v^{k+1}_i \neq v^{k+1}_j$*, then the* ϵ*-edge* $(v^{k+1}_i \epsilon v^{k+1}_j)$ *(or* $(v^{k+1}_j \epsilon v^{k+1}_i)$*) is added to* E^ϵ*.*
3. *Node* v_0 *is called the root of the* ϵ*-tree* $\mathcal{G}^\epsilon(C)$*. For each node* $v \in V$*, its predecessor in* $\mathcal{G}^\epsilon(C)$*, denoted as* $p(v)$*, is its predecessor in* $\mathcal{G}(C)$*. The level of a node* $v \in V$ *in* $\mathcal{G}^\epsilon(C)$ *is defined as being the depth of the node* v *in* $\mathcal{G}(C)$*.*

Note that $\mathcal{G}^\epsilon(C)$ as defined in Definition 5 is an oriented graph. However, it becomes a tree if the ϵ-edges are deleted from that graph. Let $(vev') \in E$ where $e \in N_R \cup \{\forall r | r \in N_R\}$. We say that v' is an e-successor of v, or v is an e-predecessor of v'. If $(v\epsilon v') \in E^\epsilon$, we say that v' is an ϵ-successor of v. We denote $p(p(...p(v)...)$ (n times) as $p^n(n)$, $v \in V$ and $p^0(v) = v$.

Remark 1 *The transformation of an $\mathcal{ALE}$-concept description C into the ϵ-tree as described in Definition 5 takes at most a polynomial time in the size of C. In fact, it holds that the size of the weak normal form of C is bounded by the size of C and the number of added ϵ-edges is bounded by $|V|^2$ where $|V|$ is the number of nodes of the description tree obtained from the weak normal form of C.*

According to the work in [3], if $\mathcal{ALE}$-concept descriptions are represented by $\mathcal{ALE}$-description trees, the normalization by the rules in Definition 2 leads to copy $\forall r$-subtrees to r-successors in $\mathcal{ALE}$-description trees. The aim of ϵ-trees is to avoid the copying of subtrees by memorizing references to subtrees to be copied. These references are represented as ϵ-edges in ϵ-trees. However, a $\forall r$-subtree can be copied to many r-successors, *i.e.*, one node may be connected to many nodes by ϵ-edges. Hence, the predecessor function p, involved in Definition 5, allows one to determine the "right neighbours" of a node thanks to its predecessor. It means that a node at level k may belong to several *k-neighbourhoods*, which is composed of right neighbour nodes. Definition 6 will formalize this idea.

Definition 6 ($neighbourhood$) *Let $\mathcal{G}^\epsilon(C) = (V,E \cup E^\epsilon,l)$ be an ϵ-tree where $v^0 \in V$ is its root. At level 0 of $\mathcal{G}^\epsilon(C)$, there is a unique 0-neighbourhood, denoted $N^0 = \{v^0\}$.*
For each $(k-1)$-neighbourhood n^{k-1}, $n^{k-1} = \{v_1^{k-1}, ..., v_m^{k-1}\} \subseteq V$ $(0 < k \leq |\mathcal{G}^\epsilon(C)|)$ such that $\bot \notin l(v_1^{k-1}) \cup ... \cup l(v_m^{k-1})$, the set $N(n^{k-1})$ of k-neighbourhoods generated from n^{k-1} is defined as follows.

1. *If there exists an edge $(v^{k-1}\forall r v^k) \in E$ such that $v^{k-1} \in n^{k-1}$ then we obtain a k-neighbourhood $n^k \in N(n^{k-1})$,*
   $n^k := \{v^k | (v^{k-1}\forall r v^k) \in E,\ v^{k-1} \in n^{k-1}\}$
2. *For each r-successor v^k of all $v_i^{k-1} \in n^{k-1}$, we obtain a k-neighbourhood $n^k \in N(n^{k-1})$,*
   $n^k{:=}\{v^k\} \cup V^\epsilon_{v^k}$ *where*
   $V^\epsilon_{v^k}{:=}\{v_i^k|\ (v^k\epsilon v_i^k) \in E^\epsilon,\ p(v_i^k) \in n^{k-1}\}$

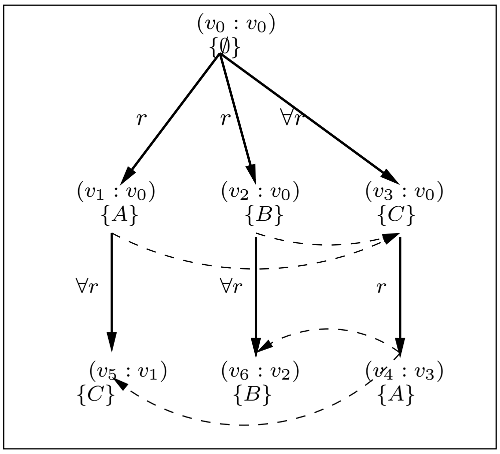

Figure 1. ϵ-tree $\mathcal{G}^\epsilon(D)$

The unique k-neighbourhood $n^k \in N(n^{k-1})$ generated from $\forall r$-successors (as defined by item 1. of Definition 6), is called $\forall r$-neighbourhood of n^{k-1}. The k-neighbourhoods $n^k \in N(n^{k-1})$ generated from r-successors (as defined by item 2. of Definition 6), are called r-neighbourhoods of n^{k-1}. If there is not any confusion, we say k-neighbourhood, $\forall r$-neighbourhood and r-neighbourhood respectively for neighbourhood at level k, neighbourhood generated from $\forall r$-successors and neighbourhood generated from r-successors of nodes in a $(k-1)$-neighbourhood.
In addition, for each k-neighbourhood n^k $(k > 0)$ a $(k-1)$-neighbourhood n^{k-1} is uniquely determined such that $n^k \in N(n^{k-1})$. Hence, for each k-neighbourhood n^k $(k > 0)$ we denote $N^{-1}(n^k)$ as the $(k-1)$-neighbourhood such that $n^k \in N(N^{-1}(n^k))$, and $N^{-n}(n^k)$ as $N^{-1}(...(N^{-1}(n^k))...)$ (n times).
In the following, we denote $\mathsf{label}(n^k)$ as the label of a k-neighbourhood $n^k{=}\{v_1^k,...,v_m^k\}$ where $\mathsf{label}(n^k){:=}l(v_1^k) \cup ... \cup l(v_m^k)$ if $\bot \notin l(v_1^k) \cup ... \cup l(v_m^k)$ and $\mathsf{label}(n^k){:=}\{\bot\}$, otherwise.

Example 2 *Let $D{:=}\exists r.(A \sqcap \forall r.C) \sqcap \exists r.(B \sqcap \forall r.B) \sqcap \exists r.(C \sqcap \forall r.A)$. The ϵ-tree $\mathcal{G}^\epsilon(D)$ is illustrated in Figure* 1.

In this figure, each node is associated with its name, predecessor and label (note that all ϵ-cycles in the figure are hidden for simplifying the presentation). For example, the node v_1 has predecessor

v_0 and label $\{A\}$. Since there is an ϵ-edge $(v_0 \epsilon v_0)$, v_2 is a r-successor of v_0 and v_3 is a $\forall r$-successor of v_0, hence the ϵ-edge $(v_2 \epsilon v_3)$ is added according to the condition 2.(b) of Definition 5. In contrast, since there is not any ϵ-edge between v_1 and v_2, no ϵ-edge connects nodes v_5 and v_6. The value of the predecessor function $p(v_i)$ is the r-predecessor or the $\forall r$-predecessor of v_i.

Intuitively, the neighbourhood notion allows us to determine nodes that have to be grouped when applying rules 1, 2. More precisely, computing the neighbourhoods of an ϵ-tree $\mathcal{G}^\epsilon(C)$ yields the nodes of the description tree corresponding to the concept description C' obtained from C by applying exhaustively rules 1 and 2. The following algorithm performs this transformation, *i.e.*, the algorithm transforms a graph, in which the notions of level and neighbourhood are well defined, into an $\mathcal{ALE}$-description tree. To get started, we consider that the input graph of Algorithm 1 is an ϵ-tree.

Algorithm 1 $\mathbf{B}(\mathcal{G}^\epsilon)$

Require: $\mathcal{G}^\epsilon = (V, E \cup E^\epsilon, l)$
Ensure: Description tree $\mathbf{B}(\mathcal{G}^\epsilon) = (V', E', w^0, l')$

$V' = \emptyset$, $E' = \emptyset$
Function ϕ from the set of subsets of V into V'.
At level 0, for the unique 0-neighbourhood of n^0 of $\mathcal{G}^\epsilon$, we set $l'(w^0) := l(v^0)$ and $\phi(n^0) := w^0$.
for all $(k-1)$-neighbourhood n^{k-1} of $\mathcal{G}^\epsilon$ where $1 \leq k \leq |\mathcal{G}^\epsilon|$ and $w^{k-1} = \phi(n^{k-1})$ **do**
  if there exists at least one node $v^{k-1} \in n^{k-1}$ such that $(v^{k-1} \forall r v^k) \in E$ **then**
    Let $n^k \in N(n^{k-1})$ be the $\forall r$-neighbourhood of n^{k-1}
    A node w^k is created and $\phi(n^k) := w^k$
    $V' := V' \cup w^k$ and $E' := E' \cup \{(w^{k-1} \forall r w^k)\}$
    $l'(w^k) := \mathsf{label}(n^k)$
  end if
  for all r-neighbourhood $n^k \in N(n^{k-1})$ **do**
    A node w^k is created and $\phi(n^k) := w^k$
    $V' := V' \cup w^k$ and $E' := E' \cup \{(w^{k-1} r w^k)\}$
    $l'(w^k) := \mathsf{label}(n^k)$
  end for
end for

Figure 2 illustrates the $\mathcal{ALE}$-description tree $\mathbf{B}(\mathcal{G}^\epsilon(D)) = (V', E', w_0, l)$ obtained from executing Algorithm 1 for the ϵ-tree $\mathcal{G}^\epsilon(D) = (V, E \cup E^\epsilon, l)$ in

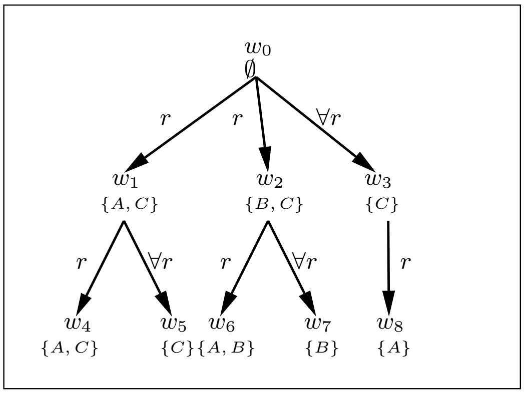

Figure 2. Description tree $\mathbf{B}(\mathcal{G}^\epsilon)$

Figure 1. At level 0, $\mathcal{G}^\epsilon(D)$ has only one 0- neighbourhood (v_0). Thus, $\mathbf{B}(\mathcal{G}^\epsilon(D))$ has root w_0 where $l'(w_0) = l(v_0)$. From the 0-neighbourhood (v_0) of $\mathcal{G}^\epsilon(D)$, we obtain three 1-neighbourhoods: one $\forall r$-neighbourhood (v_3) and two r-neighbourhoods (v_1, v_3) and (v_2, v_3) (since $(v_1 \epsilon v_3) \in E^\epsilon$; $p(v_1), p(v_3) \in \{v_0\}$ and $(v_2 \epsilon v_3) \in E^\epsilon$; $p(v_2), p(v_3) \in \{v_0\}$). Thus, we obtain two nodes $w_1, w_2 \in V'$ which are r-successors of w_0, and a node $w_3 \in V'$ which is $\forall r$-successor of w_0 where
$l'(w_1) = \mathsf{label}\{v_1, v_3\} = \{\{A\} \cup \{C\}\} = \{A, C\}$
$l'(w_2) = \mathsf{label}\{v_2, v_3\} = \{\{B\} \cup \{C\}\} = \{B, C\}$ and
$l'(w_3) = \mathsf{label}\{v_3\} = \{C\}$
From the 1-neighbourhood (v_1, v_3), we obtain two 2-neighbourhoods: a $\forall r$-neighbourhood (v_5) (since $(v_1 \forall r v_5) \in E$) and (v_4, v_5) ($(v_4 \epsilon v_5) \in E^\epsilon$; $p(v_4), p(v_5) \in \{v_1, v_3\}$). Thus, we have two successors: w_5 is the $\forall r$-successor of w_1 where $l'(w_5) = \mathsf{label}\{v_5\} = \{C\}$ and w_4 is a r-successor of w_1 where $l'(w_4) = \mathsf{label}\{v_4, v_5\} = \{A, C\}$. Similarly, from the 1-neighbourhood (v_2, v_3), we obtain a $\forall r$-neighbourhood (v_6) and a r-neighbourhood (v_4, v_6). Thus, we have two successors: w_7 is the $\forall r$-successor of w_2 where $l'(w_7) = \mathsf{label}\{v_6\} = \{B\}$ and w_6 is a r-successor of w_2 where $l'(w_6) = \mathsf{label}\{v_4, v_6\} = \{A, B\}$. Finally, from the 1-neighbourhood (v_3), we obtain the r-neighbourhood (v_4). Thus, we have a successor w_8 that is the r-successor of w_3 where $l'(w_8) = \mathsf{label}\{v_4\} = \{A\}$.

As mentioned in Section 2, rules 5, 6, 7 (in Definition 2) need to be applied after applying rules 1, 2. This means that the normalization (by rules 1, 2, 5, 6, 7) makes explicit the bottom-concept $\perp$ in the labels of nodes in the description tree.

More precisely, rule 5 makes occurring the bottom-concept $\perp$ in labels that contain P and $\neg P$ for some $P \in N_C$. Rules 6, 7 propagate the bottom-concept $\perp$ from nodes whose label contains the bottom-concept $\perp$ to their r-ancestors (a node w is called r-ancestor of a node v if the path from w to v includes only r-edges).

From Algorithm 1, a neighbourhood in ϵ-tree $\mathcal{G}^\epsilon$ corresponds to a node of description tree $\mathbf{B}(\mathcal{G}^\epsilon)$, *i.e.*, Algorithm 1 builds a mapping ϕ (*cf.* Algorithm 1) from the set of neighbourhoods to the nodes of the description tree. In particular, a path from a node v^l to a node v^k: $(v^l r v^{l+1})$,..., $(v^{k-1} r v^k)$ $(l < k)$ in description tree $\mathbf{B}(\mathcal{G}^\epsilon)$ corresponds to the following neighbourhoods n^l, n^{l+1}, ..., n^k in $\mathcal{G}^\epsilon$ such that $n^{l+1} \in N(n^l)$, ..., $n^k \in N(n^{k-1})$, $\phi(n^l)$=v^l,..., $\phi(n^k)$=v^k and n^{m+1} is a r-neighbourhood of n^m for all $l \leq m < k$.

On the other hand, the application of rules 5, 6, 7 implies that the label of a node v^1 in the description tree (that corresponds to a weak normal concept description *after* applying rules 1 and 2) contains the bottom-concept $\perp$ iff either the label of v^l contains explicitly $\perp$ (or P , $\neg P$ for some $P \in N_C$) or there exists a path composed of r-edges from v^k to a node v^l $(k > l)$ such that the label of v^k contains explicitly $\perp$ (or P , $\neg P$ for some $P \in N_C$).

From these remarks, we conclude that a neighbourhood n^l in ϵ-tree $\mathcal{G}^\epsilon$ corresponding to a node containing (explicitly and implicitly) the bottom-concept $\perp$ in the tree $\mathbf{B}(\mathcal{G}^\epsilon)$ iff either the label of n^l contains explicitly $\perp$ (or P , $\neg P$ for some $P \in N_C$) or there exist the following neighbourhoods n^l, n^{l+1}, ..., n^k in $\mathcal{G}^\epsilon$ such that $n^{l+1} \in N(n^l)$, ..., $n^k \in N(n^{k-1})$; $\phi(n^l)$=v^l,..., $\phi(n^k)$=v^k; n^{m+1} is a r-neighbourhood of n^m for all $l \leq m < k$ and the label of n^k contains explicitly $\perp$ (or P , $\neg P$ for some $P \in N_C$). We say that such neighbourhoods n^l in $\mathcal{G}^\epsilon$ contain a clash.

The following definition formalizes the notion of clash.We recall that the predecessor function $p(v^k)$ defined as the r- predecessor or $\forall r$- predecessor of v^k in ϵ-trees, allows us to access to an ancestor v^l of v^k, *i.e.*, $v^l = p^{k-l}(v^k)$.

Definition 7 (clash) *Let* $\mathcal{G}^\epsilon(C)$= $(V$,$E \cup E^\epsilon$,$l)$ *be an* ϵ*-tree and* v^0 *is its root.*

1. *For all* $v_i^k \in V$ *such that* $l(v_i^k) = \{\perp\}$*, we define a 1-clash, denoted* $[v_i^k]$.
2. *For each pair of* $\forall r$*-successors* $v_i^k, v_j^k \in V$ *if there exist* $(v_i^k \epsilon v_j^k) \in E^\epsilon$ *and* $P \in N_C$ *such that* $P, \neg P \in l(v_i^k) \cup l(v_j^k)$*, we define a 2-clash, denoted* $[v_i^k, v_j^k]$.
3. *Let* $v_i^k, v_j^k \in V$ *such that*
   (a) *If* $v_i^k \neq v_j^k$ *then* $(v_i^k \epsilon v_j^k) \in E^\epsilon$ *and there exists* $P \in N_C$ *such that* $P, \neg P \in l(v_i^k) \cup l(v_j^k)$.
   (b) *If* $v_i^k = v_j^k$ *then* $l(v_i^k)$=$\{\perp\}$.

   We define a q-clash, denoted $[v_1^l, ..., v_q^l]$, $\{v_1^l, ..., v_q^l\} \subseteq n^l$ *where* n^l *is a* $\forall r$*-neighbourhood or* $n^l = \{v^0\}$ *if there exist neighbourhoods* $n^{l+1} \in N(n^l)$, ..., $n^{k-1} \in N(n^{k-2})$, $n^k \in N(n^{k-1})$ *and* r*-edges* $(v_1^{m-1} r v_2^m) \in E$ *such that* $v_1^m, v_2^m \in n^m$ *for all* $l < m < k$, $(v_1^{k-1} r v_2^k) \in E$, $v_1^l \in n^l$ *and* v_i^k, v_j^k,$v_2^k \in n^k$.

Remark 2 *A clash* $[v_1^l, ..., v_q^l]$ *can be contained in one or several neighbourhoods. The neighbourhoods determined by item 3. of Definition 7 are reachable by the propagation of the bottom-concept* $\perp$ *(or a pair of* $P, \neg P$ *where* $P \in N_C$*) via* r*-neighbourhoods. In addition,* q *is uniquely determined from nodes* v_i^k, v_j^k *and a neighbourhood* n^l *that satisfy the conditions described in Definition 7.*

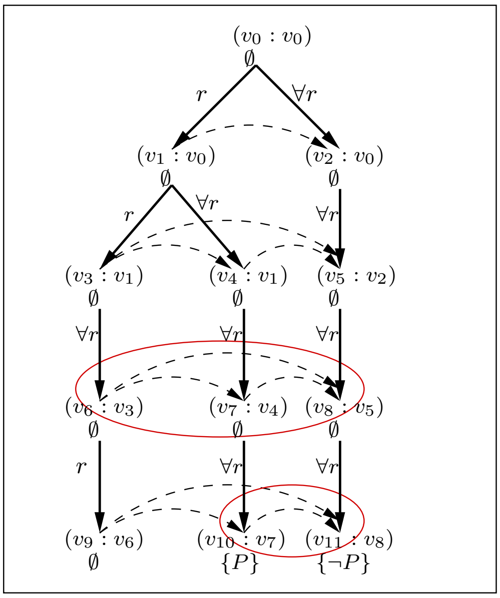

Figure 3. Clashes in ϵ-tree $\mathcal{G}^\epsilon(C)$

Example 3 *Let* $C := \exists r.(\exists r.\forall r.\exists r.\top \sqcap \forall r.\forall r.\forall r.P) \sqcap \forall r.\forall r.\forall r.\forall r.(\neg P)$*. We have two clashes in* ϵ*-tree* $\mathcal{G}^\epsilon(C)$ *as illustrated in Figure 3. According to Definition 7, first, there is a* 2-clash $[v_{10}, v_{11}]$ *since* v_{10} *and* v_{11} *are* $\forall r$*-successors such that there exists an* ϵ*-edge* $(v_{10}\epsilon v_{11})$ *and* $P, \neg P \in l(v_{10}) \cup l(v_{11})$*. Second,* $[v_6, v_7 v_8]$ *is a* 3-clash *since,*

- *there exists a neighbourhood* n^4 *where* $v_9, v_{10}, v_{11} \in n^4$ *and* $P, \neg P \in l(v_{10}) \cup l(v_{11})$ *since there are* ϵ*-edges* $(v_9\epsilon v_{10})$*,* $(v_9\epsilon v_{11})$ *and* $(v_{10}\epsilon v_{11})$*;*
- *there exists a neighbourhood* n^3 *where* $v_6, v_7, v_8 \in n^3$ *since there are* ϵ*-edges* $(v_6\epsilon v_7)$*,* $(v_6\epsilon v_8)$ *and* $(v_7\epsilon v_8)$*;*
- n^4 *is a* r*-neighbourhood generated from* n^3 *since there is a* r*-edge* $(v_6 r v_9)$ *and* $\forall r$*-edges* $(v_7\forall r v_{10})$ *and* $(v_8\forall r v_{11})$*.*

According to Definition 7, each q-clash propagated from a 1-clash $[v^k]$ where $l(v^k) = \{\perp\}$ or from a 2-clash $[v_i^k, v_j^k]$ where $P, \neg P \in l(v_i^k) \cup l(v_j^k)$, is associated with a r-predecessor (the r-predecessor v_1^l in Definition 7). In general, the number of clashes propagated from these 1-clash and 2-clash is bounded by an exponential function with the size of $\mathcal{G}^\epsilon(C)$. In particular, if there exists 1-clash $[v^0]$ where v^0 is the root of an ϵ-tree $\mathcal{G}^\epsilon(C)$, then C is unsatisfiable. A well-known result in [6] shows that the unsatisfiability in $\mathcal{ALE}$ is NP-complete. For this reason, we do not expect that the detection of all clashes in an ϵ-tree is polynomial.

In the reminder of this section, however, we will construct a polynomial structure from an ϵ-tree, namely *normalization graph* (Definition 8), allowing for storing all clashes in the ϵ-tree.

The construction of normalization graphs from ϵ-trees is based heavily on the following lemma.

Lemma 1 *Let* $\mathcal{G}^\epsilon(C) = (V, E \cup E^\epsilon, l)$ *be an* ϵ*-tree. Let* $c^k = [v_1^k, ..., v_q^k]$ $(q > 1)$ *be a* clash *in* $\mathcal{G}^\epsilon(C)$*.*

1. *There exists a node* $v^l \in V$ *of* c^k *such that* $v^l = p^{k-l}(v_i^k)$ *for all* $1 \leq i \leq q$*. Furthermore, for each level* $l \leq j \leq k$ *there exists at most a* r*-successor* v^j *such that* $v^j = p^{k-j}(v_i^k)$ *for some* $v_i^k \in \{v_1^k, ..., v_q^k\}$*.*
2. *There exist exactly* $(q-1)$ *pairs of nodes* $(v^{l_1}, u^{l_1}), ..., (v^{l_{q-1}}, u^{l_{q-1}})$ *such that* $v^{l_1}, ..., v^{l_q} \in V$ *are* r*-successors;* $u^{l_1}, ..., u^{l_{q-1}} \in V$ *are* $\forall r$*-successors;* $p(v^{l_i}) = p(u^{l_i})$ *for all* $i \in \{1, ..., q-1\}$ *and* $v^{l_1} = p^{k-l_1}(v_{i_1}^k)$*, ... ,* $v^{l_{q-1}} = p^{k-l_{q-1}}(v_{i_{q-1}}^k)$ *for some* $(q-1)$ *nodes* $v_{i_1}^k, ..., v_{i_{q-1}}^k \in \{v_1^k, ..., v_q^k\}$*.*

Item 1. of Lemma 1 allows us to determine a r-predecessor v^l and r-successors v^{l+1}, ... , v^h from a clash c^k where $l \leq h < k$. The nodes v^l and v^h are called *head* and *tail* of c^k, respectively. A proof of Lemma 1 can be found in Appendix.

Definition 7 (clash) allows us to detect all clashes in an ϵ-tree. It does not show, however, how these clashes are stored in the ϵ-tree. In Example 3, from the existence of 2-clash $[v_{10}, v_{11}]$ it is required that the label of all neighbourhoods containing nodes v_{10}, v_{11} must contain the bottom-concept $\perp$. Thus, we need a node to store the bottom-concept $\perp$ for this clash. However, it is not possible to use, for example, node v_{11} for this purpose since v_{11} belongs to neighbourhood (v_{11}) which does not include both nodes v_{10}, v_{11}. In addition, from 3-clash $[v_6, v_7, v_8]$ it is required that the label of all neighbourhoods containing nodes v_6, v_7, v_8 must contain the bottom-concept $\perp$. Therefore, we need node(s) to store bottom-concepts $\perp$ for this clash. Similarly, we cannot store the bottom-concept $\perp$ to the label of node v_8.

This problem is solved by extending the neighbourhood notion given in Definition 6 and proposing a structure, called *normalization graph.* This structure, denoted $\mathcal{G}_C^\epsilon$, is extended from ϵ-tree $\mathcal{G}^\epsilon(C)$ by adding some nodes and edges such that it can store bottom-concepts $\perp$ for clashes $[v_1^k, ..., v_q^k]$ in preserving the other neighbourhoods in ϵ-tree $\mathcal{G}^\epsilon(C)$. It is necessary to add new nodes in order to store bottom-concepts $\perp$ for q-clash since, as explained above, nodes in ϵ-trees can be shared by several neighbourhoods and a node belonging to a q-clash may belong to a neighbourhood which does not include this q-clash.

To sum up, the neighbourhood notion for the normalization graph built from an ϵ-tree has to be redefined such that i) it preserves the neighbourhoods in the ϵ-tree if these neighbourhoods do not contain any clash, and ii) it yields a new neighbourhood for each neighbourhood that contains a clash in the ϵ-tree. The neighbourhood notion defined in this way allows not only for guaranteeing a correct transformation (by Algorithm 1) from normalization graphs into description trees but also for extending naturally the product operation of description trees presented in [3] to the product operation of normalization graphs. More precisely,

- The $\forall r$-neighbourhood n^k of a $(k-1)$-neighbourhood n^{k-1} is defined as a set of all $\forall r$-

successors of all nodes belonging to n^{k-1} if there does not exist any node w^k such that $p(w^k) \in n^{k-1}$ and there exists an ϵ-edge $(v^k \epsilon w^k)$ where v^k a $\forall r$-successor of a node belonging to n^{k-1}. Otherwise, n^k includes such a node w^k. Furthermore, for each r-successor v^k of all nodes belonging to n^{k-1}, a r-neighbourhood n^k of n^{k-1} is defined as a set including v^k and all nodes w^k such that there exists an ϵ-edge $(v^k \epsilon w^k)$ and $p(w^k) \in n^{k-1}$.

Now we show how to construct the normalization graph $\mathcal{G}^\epsilon_C$ from a ϵ-tree $\mathcal{G}^\epsilon(C)$ by using the extended neighbourhood notion described above and Lemma 1.

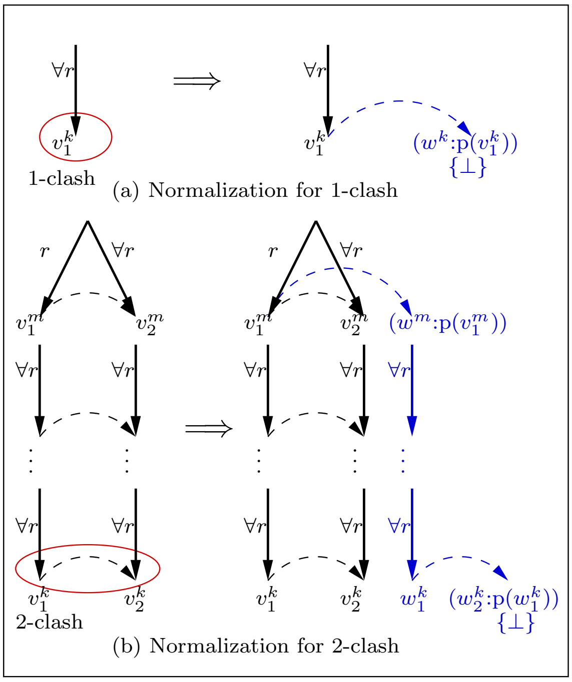

Figure 4. Normalization for clashes

Let c^k be a clash at level k such that there does not exist any clash c'^k where $c'^k \subset c^k$. The aim of the construction is to obtain a node w^k in $\mathcal{G}^\epsilon_C$ for each c^k such that i) $l'(w^k) = \{\bot\}$, ii) there is a $(k-1)$-neighbourhood n^{k-1} in $\mathcal{G}^\epsilon(C)$ such that a k-neighbourhood $n^k \in N(n^{k-1})$ includes c^k iff there is a $(k-1)$-neighbourhood n'^{k-1} in $\mathcal{G}^\epsilon_C$ such that $n^{k-1} \subseteq n'^{k-1}$ and a k-neighbourhood $n'^k \in N(n'^{k-1})$ includes w^k. For example,

1. Let $[v_1^k]$ be a 1-clash in $\mathcal{G}^\epsilon(C)$ (Figure 4). Assume that v_1^k is a $\forall r$-successor (from Definition 7, v_1^k must be a $\forall r$-successor or the root of $\mathcal{G}^\epsilon(C)$). If we add to $\mathcal{G}^\epsilon(C)$ a node w^k such that $l'(w^k)$:=$\{\bot\}$, $p(w^k)$:=$p(v_1^k)$ and an ϵ-edge $(v_1^k \epsilon w^k)$, then w^k belongs to the $\forall r$-neighbourhood of a neighbourhood n'^{k-1} that contains $p(v_1^k)$.
2. Let $[v_1^k, v_2^k]$ be a 2-clash in $\mathcal{G}^\epsilon(C)$ (Figure 4). We have v_1^k, v_2^k are $\forall r$-successors and the r-successor v_1^m is determined by item 1. of Lemma 1. If we add to $\mathcal{G}^\epsilon(C)$ nodes $w^m, ..., w_1^k, w_2^k$ such that $l'(w^m)= ... = l'(w_1^k)$:= $\emptyset$, $l'(w_2^k)$:=$\{\bot\}$; $p(w^m)$:=$p(v_2^m)$, ..., $p(w_1^k)$:=w^{k-1}, $p(w_2^k)$:=$p(w_1^k)$ and ϵ-edges $(v_1^m \epsilon w^m)$, $(w_1^k \epsilon w_2^k)$, then w_2^k belongs to the $\forall r$-neighbourhood of a neighbourhood n'^{k-1} that contains $p(v_1^k)$. In fact, by the construction, there exist a m-neighbourhood n'^m and a $(k-1)$-neighbourhood n'^{k-1} such that $v_1^m, v_2^m, w^m \in n'^m$; $p(v_1^k)$, $p(v_2^k)$, $p(w_1^k)$ $\in n'^{k-1}$ and $n'^m \in N^{-(k-1-m)}(n'^{k-1})$. From the extended neighbourhood notion described above, we obtain $w_2^k \in n'^k$ where n'^k is the $\forall r$-neighbourhood of n'^{k-1}.

Since the number of clashes in an ϵ-tree may be exponential, the size of the graph obtained from the ϵ-tree by the normalization may increase exponentially if a new path of $\forall r$-edges is added for each clash. To avoid the exponential blow-up caused by the normalization, we extend the predecessor function p for nodes in normalization graphs by allowing p to be a set function, *i.e.*, $p(v)$ is a set of nodes. This extension leads to redefine the notion of neighbourhood for normalization graphs since the notion of neighbourhood relies on the the predecessor function p. The extension of predecessor function p will be described in Definition 8 (normalization graph).

Before normalizing an ϵ-tree $\mathcal{G}^\epsilon(C) = (V, E \cup E^\epsilon, l)$ by Definition 8, $\mathcal{G}^\epsilon(C)$ needs to be simplified as follows:

– Let $[v_1^l, ..., v_a^l]$ and $[w_1^k, ..., w_b^k]$ be clashes in $\mathcal{G}^\epsilon(C)$ such that $l < k$ and $p^{k-l}(w_i^k) \in \{v_1^l, ..., v_a^l\}$ for all $w_i^k \in \{w_1^k, ..., w_b^k\}$. From the neighbourhood definition, it holds that for each k-neighbourhood n^k such that $\{w_1^k, ..., w_b^k\} \subseteq n^k$, there exist neighbourhoods n^l, ..., n^{k-1} such that n^l, $n^{l+1} \in N(n^l)$, ... , $n^k \in N(n^{k-1})$ and $\{v_1^l, ..., v_a^l\} \subseteq n^l$. From this claim, if $[v_1^k]$ is a 1-clash in $\mathcal{G}^\epsilon(C)$ then the subtree $\mathcal{G}^\epsilon(C)(v_1^k)$ can be deleted from $\mathcal{G}^\epsilon(C)$. This can be performed by deleting all nodes $v^h \in V$, $h > k$ such that $p^{h-k} = v_1^k$ and all edges such that one of two endpoints belongs to the set of deleted nodes. Additionally, v_1^k is relabeled with $\emptyset$.

Notice that we need only to consider clashes c^k in $\mathcal{G}^\epsilon(C)$ such that there does not exist any clash c'^k where $c'^k \subset c^k$ since if a neighbourhood n^k contains c^k and $c'^k \subseteq c^k$ then n^k contains c'^k. This implies that if c'^k is normalized, *i.e.* any neighbourhood that contains c'^k includes the bottom-concept $\perp$ in its label, then c^k is normalized as well. In addition, according to Lemma 1, each clash c^k has nodes v^l, v^h as its head and tail. If we normalized all subsets of clashes that are grouped according to its head and tail, then every clash in $\mathcal{G}^\epsilon(C)$ will be normalized once.

Definition 8 (*normalization graph*) *Let $\mathcal{G}^\epsilon(C) = (V,E \cup E^\epsilon,l)$ be an ϵ-tree and v^0 is its root. The normalization graph of C is denoted as $\mathcal{G}_C^\epsilon = (V',E' \cup E'^\epsilon,l')$ where $V \subseteq V'$, $E \subseteq E'$, $E^\epsilon \subseteq E'^\epsilon$ and $l'(v){:=}l(v)$ if $v \in V$. Furthermore, we define a predecessor function $\mathcal{P}(v)$, $v \in V'$ where $\mathcal{P}(v)$ is a set of predecessors of v in $\mathcal{G}_C^\epsilon$ and $\mathcal{P}(v){:=}\{p(v)\}$ if $v \in V$. The normalization graph $\mathcal{G}_C^\epsilon$ is obtained from $\mathcal{G}^\epsilon(C)$ as follows. If there exists a* 1-clash $[v^0]$ *then $\mathcal{G}_C^\epsilon{:=}\mathcal{G}(\perp)$. Otherwise,*

1. *For each* 1-clash $c^k{=}[v_1^k]$ *in $\mathcal{G}^\epsilon(C)$, we add a node w^k and an ϵ-edge $(v_1^k \epsilon w^k)$ where $\mathcal{P}(w^k){:=}\{p(v_1^k)\}$ and $l'(w^k){:=}\{\perp\}$.*
2. *For each pair of nodes v^l, v^h $(l < h)$ where v^l is a r-predecessor and v^h is a r-successor, let $c_1^k, ..., c_n^k$ be* clashes *at level k such that v^l, v^h are their head and tail, respectively. For each c_i^k, we denote $V(c_i^k)$ for the set of the r-successors determined by item 1. of Lemma 1 from c_i^k.*
   (a) *For each level $l < m \leq h$, if there exists a r-successor $v'^m \in \bigcup_{i=1}^n V(c_i^k)$ then we add to $\mathcal{G}^\epsilon(C)$ a path of $\forall r$-edges $\{(w^m \forall r w^{m+1}),\dots, (w^{k-1} \forall r w^k)\}$, denoted P_m, where $l'(w^m){=}\ l'(w^{m+1}){=}\ ...{=}l'(w^k)$ $:={\emptyset}$. Furthermore, for each r-successors $v^m \in \bigcup_{i=1}^n V(c_i^k)$ at level m, we add to $\mathcal{G}^\epsilon(C)$ an ϵ-edge $(v^m \epsilon w^m)$.*
   If $m = l + 1$ then we set $\mathcal{P}(w^m){:=}\{v^l\}$. Otherwise, let $v^{h_1}, \dots, v^{h_s} \in \bigcup_{i=1}^n V(c_i^k)$ be nodes at the highest level in $\mathcal{G}^\epsilon(C)$ such that $h^i < m$ and $v^m, v^{h_i} \in c_j^k$ for some $c_j^k \in \{c_1^k, ..., c_n^k\}$. Let $u_1^{m-1}, \dots, u_s^{m-1}$ be nodes, respectively, on the paths $P_{h_1}, \dots, P_{h_s}$ added by this item. We set $\mathcal{P}(w^m) := \{u_1^{m-1}, \dots, u_s^{m-1}\}$, $\mathcal{P}(w^{m+1}){:=}\{w^m\}, \dots,$ $\mathcal{P}(w^k){:=}\ \{w^{k-1}\}$.
   (b) *We add a node u^k and an ϵ-edge $(w^k \epsilon u^k)$ where $\mathcal{P}(w^k){:=}\mathcal{P}(u^k)$, $l'(u^k){:=}\{\perp\}$ and w^k is on the path P_h added by item 2.(a).*

The root of $\mathcal{G}_C^\epsilon$ is the root of $\mathcal{G}^\epsilon(C)$. The level of a node is defined as the number of ordinary edges of a path from the root to that node since the number of ordinary edges of all paths from the root to a node is constant. In addition, in a normalization graph $\mathcal{G}_C^\epsilon{=}(V',E' \cup E'^\epsilon,l')$ we can define $\mathcal{P}^n(v^k) := \bigcup_{v^{k-n+1} \in \mathcal{P}^{n-1}(v^k)} \mathcal{P}(v^{k-n+1})$, $1 < n \leq k$ for each $v^k \in V'$. We denote $p^n(v^k)$ for $\mathcal{P}^n(v^k)$ if $\mathcal{P}^n(v^k)$ is singleton, *i.e.*, $\mathcal{P}^n(v^k){=}\{p^n(v^k)\}$. Note that if v^k is a r-successor or $\forall r$-successor of v^{k-1} where v^{k-1} and v^k in $\mathcal{G}_C^\epsilon$ then $\mathcal{P}(v^k){=}\{v^{k-1}\}$. Therefore, $\mathcal{P}(v^k)$ is not singleton iff v^k is added by Definition 8 and v^k is neither r-successor nor $\forall r$-successor.
Notice that the condition $c_i \not\subseteq c_j$ for all clashes c_i, c_j considered in Definition 8 (normalization graph) guarantees that any neighbourhood n'^k in $\mathcal{G}_C^\epsilon$ containing a clash includes uniquely a node $u^k \in V' \setminus V$ such that $l'(u^k){=}\{\perp\}$.

Remark 3 *It is obvious that the number of clashes propagated from clashes containing the bottom-concept $\perp$ or a pair of $P, \neg P$ where $P \in N_C$, is bounded by an exponential function with the size of $\mathcal{G}^\epsilon(C)$ (for instance, $2^{|V|}$ where $|V|$ is the number of nodes in $\mathcal{G}^\epsilon(C)$). However, the number of nodes and edges ($\forall r$-edges and ϵ-edges) added by Definition 8 (normalization graph) is polynomial in the size of $\mathcal{G}^\epsilon(C) = (V,E \cup E^\epsilon,l)$. In fact, for each pair of r-successor and r-predecessor v^l, v^h and for each level $l \leq m \leq h$, item 2.(a) in Definition 8 adds at most i) $|\mathcal{G}^\epsilon(C)|$ $\forall r$-successors, ii) $|\mathcal{G}^\epsilon(C)|$ $\forall r$-edges, iii) $|V|$ ϵ-edges from r-successors to the $\forall r$-predecessor added at level m. In addition, $|\mathcal{P}(w^m)|$ is bounded by $(m - (l + 1))$ where w^m is a $\forall r$-predecessor but neither $\forall r$-successor nor r-successor. Note that if w^k is a $\forall r$-successor or r-successor then $\mathcal{P}(w^k)$ is singleton. Therefore, for each pair of r-successor and r-predecessor v^l, v^h the number of added nodes is bounded by $|\mathcal{G}^\epsilon(C)| \times (h - l) \leq |\mathcal{G}^\epsilon(C)|^2$. Since the number of* *pairs of nodes v^l, v^h in $\mathcal{G}^\epsilon(C)$ is bounded by $|V|^2$, the number of nodes added by the normalization is bounded by $|\mathcal{G}^\epsilon(C)|^2 \times |V|^2 \leq |V|^4$.*

The normalization of ϵ-trees by Definition 8 adds nodes that are neither r-successor nor $\forall r$-successor. Moreover, the predecessor function p is redefined for normalization graphs such that $p(v)$ (denoted $\mathcal{P}(v)$ in Definition 8) may be a set of nodes. By consequent, the neighbourhood notion needs to be redefined for normalization graphs in order to take into account the new elements that are brought from the normalization.

Definition 9 (*extended-neighbourhood*) *Let $\mathcal{G}^\epsilon_C = (V, E \cup E^\epsilon, l)$ be a normalization graph where $v^0 \in V$ is its root. At level 0 of $\mathcal{G}^\epsilon_C$, there is a unique 0-neighbourhood, denoted $N^0 = \{v^0\}$.*
For each $(k-1)$-neighbourhood n^{k-1}, $n^{k-1} = \{v_1^{k-1}, ..., v_m^{k-1}\} \subseteq V$ $(0 < k \leq |\mathcal{G}^\epsilon_C|)$ such that $\bot \notin l(v_1^{k-1}) \cup ... \cup l(v_m^{k-1})$, the set $N(n^{k-1})$ of k-neighbourhoods generated from n^{k-1} is defined as follows.

1. *If there exists an edge $(v^{k-1} \forall r v^k) \in E$ such that $v^{k-1} \in n^{k-1}$ then we obtain a k-neighbourhood $n^k \in N(n^{k-1})$,*
$$n^k := \begin{cases} V^\epsilon(n^{k-1}), & V^\epsilon(n^{k-1}) \neq \emptyset \\ V^\forall(n^{k-1}), & V^\epsilon(n^{k-1}) = \emptyset \end{cases}$$
where
$V^\forall(n^{k-1}) := \{v^k | (v^{k-1} \forall r v^k) \in E, v^{k-1} \in n^{k-1}\}$
$V^\epsilon(n^{k-1}) := \{v_i^k | (v^k \epsilon v_i^k) \in E^\epsilon,\ v^k \in V^\forall(n^{k-1}),\ v_i^k \notin V^\forall(n^{k-1}),\ \mathcal{P}(v_i^k) \cap n^{k-1} \neq \emptyset\}$
2. *For each r-successor v^k of all $v_i^{k-1} \in n^{k-1}$, we obtain a k-neighbourhood $n^k \in N(n^{k-1})$, $n^k := \{v^k\} \cup V^\epsilon_{v^k}$ where*
$V^\epsilon_{v^k} := \{v_i^k |\ (v^k \epsilon v_i^k) \in E^\epsilon,\ \mathcal{P}(v_i^k) \cap n^{k-1} \neq \emptyset\}$

Remark 4 *From Definition 8 and Definition 9, it holds that if $n^l = \{v_1^l, ..., v_p^l\}$ and $n^k = \{v_1^k, ..., v_q^k\}$ $(l < k)$ are neighbourhoods in a normalization graph $\mathcal{G}^\epsilon_C$ such that $N^{-(k-l)}(n^k) = n^l$ then $\mathcal{P}^{k-l}(v_i^k) \cap n^l \neq \emptyset$ for all $v_i^k \in n^k$. This property holds also for ϵ-trees $\mathcal{G}^\epsilon(C)$ where $\mathcal{P}(v^k)$ is singleton for all nodes v^k in $\mathcal{G}^\epsilon(C)$.*

Figure 5 shows how to normalize an ϵ-tree $\mathcal{G}^\epsilon(C)$ by Definition 8 where
$$C := \forall r.\forall r.\forall r.\exists r.\forall r.P \sqcap \exists r.(\exists r.\forall r.\forall r.\forall r.\neg P \sqcap \forall r.(\exists r.\forall r.\forall r.\neg P \sqcap \forall r.\forall r.\forall r.Q))$$

According to Definition 7 (clash), there are two clashes $[v_{15}, v_{17}]$ and $[v_{16}, v_{17}]$ in $\mathcal{G}^\epsilon(C)$. Lemma 1 allows us to determine that $V([v_{15}, v_{17}]) = \{v_1, v_7, v_{13}\}$, $V([v_{15}, v_{17}]) = \{v_1, v_4, v_{13}\}$ (*cf.* the notation in Definition 8) and v_0, v_{13} are the head and tail of clashes $[v_{15}, v_{17}]$ and $[v_{16}, v_{17}]$. Paths $(w_0, w_1, w_2, w_3, w_4)$, (w_5, w_6, w_7, w_8), (w_9, w_{10}, w_{11}), (w_{12}, w_{13}) and ϵ-edges $(v_1 \epsilon w_0)$, $(v_4 \epsilon w_5)$, $(v_7 \epsilon w_9)$, $(v_{13} \epsilon w_{12})$ are added by item 2.(a) of Definition 8 from the sets $V([v_{15}, v_{17}])$ and $V([v_{15}, v_{17}])$. Next, item 2.(a) determines predecessor function for nodes w_0, w_5, w_9 and w_{12}: $\mathcal{P}(w_0) = \{v_1\}$, $\mathcal{P}(w_5) = \{w_0\}$, $\mathcal{P}(w_9) = \{w_1\}$ and $\mathcal{P}(w_{12}) = \{w_6, w_9\}$. Finally, a node w_{14} where $l'(w_{14}) = \{\bot\}$ and an ϵ-edge $(w_{13} \epsilon w_{14})$ are added by item 2.(b) .
As a result, we obtain that i) $v_{15}, v_{17} \in V^\forall(n^4)$ iff $w_{14} \in V^\epsilon(n^4)$ for some 4-neighbourhood n^4 in the normalization graph $\mathcal{G}^\epsilon_C$ (Figure 5), ii) $v_{16}, v_{17} \in V^\forall(m^4)$ iff $w_{14} \in V^\epsilon(m^4)$ for some 4-neighbourhood m^4 in the normalization graph $\mathcal{G}^\epsilon_C$. To sum up, a normalization graph $\mathcal{G}^\epsilon_C$ preserves all neighbourhoods in its ϵ-tree $\mathcal{G}^\epsilon(C)$ and yields new neighbourhoods, represented by V^ϵ, which correspond to neighbourhoods containing clashes in $\mathcal{G}^\epsilon(C)$. The following lemma assert this important property of normalization graphs.

Lemma 2 *Let $\mathcal{G}^\epsilon(C) = (V, E \cup E^\epsilon, l)$ be an ϵ-tree where $v^0 \in V$ is its root and $\mathcal{G}^\epsilon_C = (V', E' \cup E'^\epsilon, l')$ be its normalization graph. Let n'^k be a k-neighbourhood in $\mathcal{G}^\epsilon_C$. If $k = 0$ then $\mathsf{label}(n'^0) \neq \{\bot\}$ iff n'^0 does not contain any* clash. *For all $k > 0$, the following claims are equivalent:*

1. $\mathsf{label}(n'^k) \neq \{\bot\}$.
2. *There exists a neighbourhood n^k in $\mathcal{G}^\epsilon(C)$ such that $n^k = n'^k \cap V$ and $\mathsf{label}(n^k) = \mathsf{label}(n'^k)$.*
3. *There does not exist any* q-clash *$[v_1^k, .., v_q^k]$ such that $\{v_1^k, .., v_q^k\} \subseteq n^k$, $n^k = V^\forall(n'^{k-1}) \cap V$ and $n'^k \in N(n'^{k-1})$ where n^k is a $\forall r$-neighbourhood in $\mathcal{G}^\epsilon(C)$.*

A proof of Lemma 2 can be found in Appendix.
Algorithm 1 can transform ϵ-trees $\mathcal{G}^\epsilon(C)$ and normalization graphs $\mathcal{G}^\epsilon_C(C)$ into description trees since the neighbourhood, level, predecessor notions are well defined for both graphs. In general, the description tree $\mathbf{B}(\mathcal{G}^\epsilon_C)$ is different from $\mathbf{B}(\mathcal{G}^\epsilon(C))$. However, $\mathbf{B}(\mathcal{G}^\epsilon_C)$ can be obtained from $\mathbf{B}(\mathcal{G}^\epsilon(C))$ by applying the following normalization rules (5g), (6g) and (7g) (g stands for *graph*) which are de-

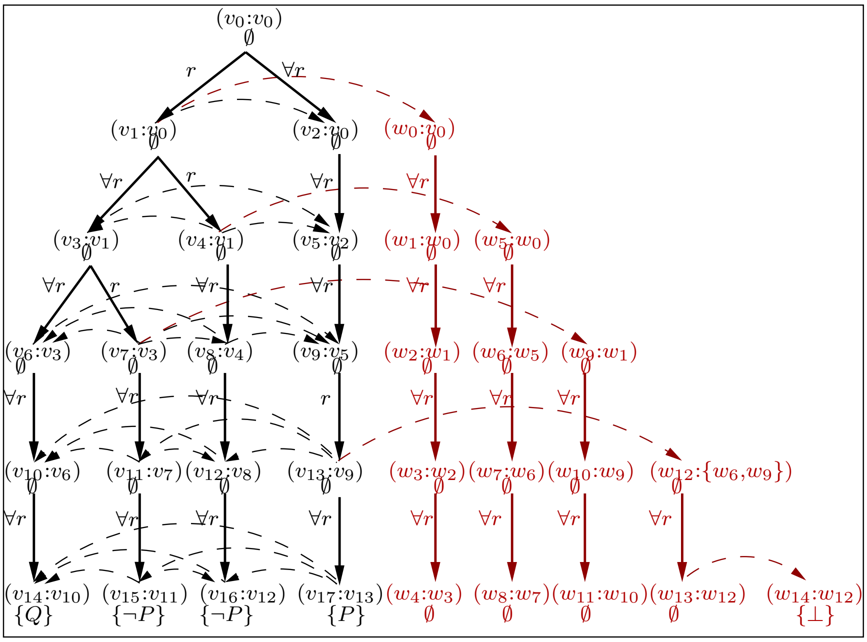

Figure 5. Normalization graph $\mathcal{G}^{\epsilon}_C$

fined for description trees. These rules must correspond to rules 5, 6, 7 (Definition 2) defined for concept descriptions.

Lemma 3 *Let C be an $\mathcal{ALE}$-concept description in the weak normal form. Let $\mathcal{G}^{\epsilon}(C)$ and $\mathcal{G}^{\epsilon}_C$ be the description tree and normalization graph of C, respectively. There exists an isomorphism between $\mathbf{B}(\mathcal{G}^{\epsilon}_C)$ and the description tree $\mathcal{H}$ obtained from $\mathbf{B}(\mathcal{G}^{\epsilon}(C)) = (V_3, E_3, z^0, l_3)$ by exhaustively applying the following rules:*

1. $P, \neg P \in l_3(z)$, $P \in N_C$, $z \in V_3 \rightarrow$ $l_3(z) := \{\bot\}$ *(rule 5g)*
2. $(zrz') \in E_3$, $\mathbf{B}(\mathcal{G}^{\epsilon}(C))(z') = \mathcal{G}(\bot) \rightarrow$ $\mathbf{B}(\mathcal{G}^{\epsilon}(C))(z) := \mathcal{G}(\bot)$ *(rule 6g)*
3. $\bot \in l_3(z)$, $z \in V_3 \rightarrow$ $\mathbf{B}(\mathcal{G}^{\epsilon}(C))(z) := \mathcal{G}(\bot)$ *(rule 7g)*

A proof of Lemma 3 can be found in Appendix.
The following proposition is an important result of this section. It establishes the equivalence between the normalization by the rules in Definition 2 for concept descriptions, and the normalization by Definitions 5 and 8 for description trees.

Proposition 1 *Let C be an $\mathcal{ALE}$-concept description. Let $\mathcal{G}_C$ and $\mathcal{G}^{\epsilon}_C$ be its description tree and normalization graph, respectively. There exists an isomorphism between $\mathbf{B}(\mathcal{G}^{\epsilon}_C)$ and $\mathcal{G}_C$.*

A proof of Proposition 1 can be found in Appendix.

Remark 5 *Proposition 1 and Example 1 yield that the size of $\mathbf{B}(\mathcal{G}^{\epsilon}_C)$ may be exponential in the size of $\mathcal{G}^{\epsilon}_C$. In fact, Example 1 shows that $\mathcal{G}_C$ may be exponential in the size of C and Proposition 1 asserts that there exists an isomorphism between $\mathcal{G}_C$ and $\mathbf{B}(\mathcal{G}^{\epsilon}_C)$.*

We now exploit the results have been obtained in this section to propose a polynomial algorithm in space for deciding subsumption between two $\mathcal{ALE}$-concept descriptions. Such an algorithm is interesting since it enables us to decide subsumption between concepts including *lcs* by manipulating directly corresponding graphs.
Note that the algorithm described in [3] for checking the existence of a homomorphism between two $\mathcal{ALE}$-description trees obtained from *normalized* concept descriptions cannot be used for this aim since it requires that all nodes of description trees is explicitly represented, *i.e.*, it requires an exponential space.
The underlying idea of Algorithm 2 is that checking the existence of a homomorphism from $\mathbf{B}(\mathcal{H}^{\epsilon})$

to $\mathbf{B}(\mathcal{G}^\epsilon)$ can be performed without transforming completely $\mathcal{H}^\epsilon$ and $\mathcal{G}^\epsilon$ into $\mathbf{B}(\mathcal{H}^\epsilon)$ and $\mathbf{B}(\mathcal{G}^\epsilon)$. By fixing on each neighbourhood of $\mathcal{H}^\epsilon$ from the highest level to the root, this process can be carried out by checking the existence of a mapping between *neighbourhood paths* from the root to neighbourhoods at the highest level in $\mathcal{H}^\epsilon$ and $\mathcal{G}^\epsilon$. At each checking step, the algorithm needs memory pieces to store neighbourhood paths in $\mathcal{H}^\epsilon$ and $\mathcal{G}^\epsilon$.

Algorithm 2 check($\mathcal{H}^\epsilon(n^k)$, $\mathcal{G}^\epsilon(m^k)$)

Require: n^k, m^k are k-neighbourhoods, respectively, in normalization graphs $\mathcal{H}^\epsilon$ and $\mathcal{G}^\epsilon$.
Ensure: Answer "**true**" if there exists a homomorphism from $\mathbf{B}(\mathcal{H}^\epsilon)$ to $\mathbf{B}(\mathcal{G}^\epsilon)$. Otherwise, answer "**false**".

```
if label(m^k) = {⊥} then
  return true;
end if
if label(n^k) ⊈ label(m^k) then
  return false;
end if
Let n_1^{k+1}, ..., n_p^{k+1} be (k+1)-neighbourhoods generated from n^k;
Let m_1^{k+1}, ..., m_q^{k+1} be (k+1)-neighbourhoods generated from m^k;
for 1 ≤ i ≤ p do
  found := false;
  for 1 ≤ j ≤ q do
    if n_i^{k+1}, m_j^{k+1} are ∀r-neighbourhoods and
    check(H^ε(n_i^{k+1}), G^ε(m_j^{k+1})) then
      found := true;
    end if
    if n_i^{k+1}, m_j^{k+1} are r-neighbourhoods and
    check(H^ε(n_i^{k+1}), G^ε(m_j^{k+1})) then
      found := true;
    end if
  end for
  if found = false then
    return false;
  end if
end for
return true;
```

Let m^0, n^0 be 0-neighbourhoods of, respectively, normalization graphs $\mathcal{H}^\epsilon$, $\mathcal{G}^\epsilon$. If function **check**($\mathcal{H}^\epsilon(n^0)$, $\mathcal{G}^\epsilon(m^0)$) returns "**true**", there exists a homomorphism from $\mathbf{B}(\mathcal{H}^\epsilon)$ to $\mathbf{B}(\mathcal{G}^\epsilon)$. Otherwise, there does not exist any homomorphism from $\mathbf{B}(\mathcal{H}^\epsilon)$ to $\mathbf{B}(\mathcal{G}^\epsilon)$.

Completeness of the algorithm
Assume that there exists a homomorphism φ from $\mathbf{B}(\mathcal{H}^\epsilon)$ to $\mathbf{B}(\mathcal{G}^\epsilon)$. We have to show that **check**($\mathcal{H}^\epsilon(n^0)$, $\mathcal{G}^\epsilon(m^0)$) returns "**true**".
Let $\{w'_0, ..., w'_n\}$ be a post-order sequence of nodes of $\mathbf{B}(\mathcal{H}^\epsilon)$ (note that w'_n corresponds to the root of $\mathcal{H}^\epsilon$). This sequence corresponds to a sequence of neighbourhoods of $\mathcal{H}^\epsilon$ (Algorithm 1). If there is not any confusion, we can say w'_n for a neighbourhood on $\mathcal{H}^\epsilon$. We will prove the claim by induction on i where $0 \leq i \leq n$.

- Step $i = 0$. Let $v = \varphi(w'_0)$. Since φ is a homomorphism, it is that $l(w'_0) \subseteq l(\varphi(w'_0))$ or $l(\varphi(w'_0)) = \{\perp\}$ ($l(w'_i) = \mathsf{label}(n_i^k)$ where n_i^k is the neighbourhood corresponding to w'_i). Furthermore, since p, q are equal to zero in the algorithm, no iteration is performed. Thus, **check**($\mathcal{H}^\epsilon(w'_0)$, $\mathcal{G}^\epsilon(\varphi(w'_0))$) returns "**true**".
- Induction step $(i-1) \to i$. By induction hypothesis, **check**($\mathcal{H}^\epsilon(w'_j)$, $\mathcal{G}^\epsilon(\varphi(w'_j))$) returns "**true**" for all $0 \leq j < i$. Let $v = \varphi(w'_i)$. Since φ is a homomorphism, it is $l(v) = \{\perp\}$ or $l(w'_i) \subseteq l(v)$. Let $w'_{i_1}, ..., w'_{i_p}$ be the neighbourhoods generated from the neighbourhood w'_i and the edges $(w'_i r_1 w'_{i_1})$,...,$(w'_i r_p w'_{i_p})$. Since $\{w'_0, ..., w'_n\}$ is a post-order sequence, hence $i_1, ... i_p \in \{0, ..., i-1\}$. By induction hypothesis, **check**($\mathcal{H}^\epsilon(w'_{i_l})$, $\mathcal{G}^\epsilon(\varphi(w'_{i_l}))$) returns "**true**" for all $1 \leq l \leq p$. Since φ is a homomorphism and $w'_{i_1}, ..., w'_{i_p}$ are the neighbourhoods generated from the neighbourhood w'_i, hence $\varphi(w'_{i_j})$ have to be the neighbourhoods generated from the neighbourhood v and the edges $(v r_l \varphi(w'_{i_j}))$ for all $1 \leq l \leq p$. This implies that **check**($\mathcal{H}^\epsilon(w'_i)$, $\mathcal{G}^\epsilon(v)$) returns "**true**" since the iteration with index j (second iteration) in the algorithm does not return "**false**" for all $1 \leq j \leq q$.

Soundness of the algorithm
Assume that **check**($\mathcal{H}^\epsilon(n^0)$, $\mathcal{G}^\epsilon(m^0)$) returns "**true**". We have to show that there exists a homomorphism φ from $\mathbf{B}(\mathcal{H}^\epsilon)$ to $\mathbf{B}(\mathcal{G}^\epsilon)$.
Since **check**($\mathcal{H}^\epsilon(n^0)$, $\mathcal{G}^\epsilon(m^0)$) returns "**true**", it is that $l(n^0) \subseteq l(m^0)$ or $l(m^0) = \{\perp\}$ where n^0 and m^0 are neighbourhoods in $\mathcal{H}^\epsilon$ and $\mathcal{G}^\epsilon$. If there is not any confusion, we can say node w_i for a neighbourhood on $\mathcal{H}^\epsilon$ and say node v_i for a neighbourhood on $\mathcal{G}^\epsilon$. We start with $\varphi(w_0) := v_0$. Let $w^1, ..., w^p$ be nodes (neighbourhoods) gener-

ated from the neighbourhood n^0 and $v^1, ..., v^q$ be nodes (neighbourhoods) generated from the neighbourhood m^0. Since **check**($\mathcal{H}^\epsilon(w_0)$, $\mathcal{G}^\epsilon(v_0)$) returns "**true**", according to the algorithm we have for each w^l, $l \in \{1, ..., p\}$, there exists $v^{i_l} \in \{v^1, ..., v^q\}$ such that $l(v^{i_l}) = \{\bot\}$ or $l(w^l) \subseteq l(v^{i_l})$; edges $(w_0 r w^l)$, $(v_0 r v^{i_l})$ (or $(w_0 \forall r w^l)$, $(v_0 \forall r v^{i_l})$) and **check**($\mathcal{H}^\epsilon(w^l)$, $\mathcal{G}^\epsilon(v^{i_l})$) returns "**true**" . By induction hypothesis, for all $1 \leq l \leq p$ there exist homomorphisms φ_l between $\mathbf{B}(\mathcal{H}^\epsilon(w^l))$ and $\mathbf{B}(\mathcal{G}^\epsilon(v^{i_l}))$. For all $1 \leq l \leq p$, we define $\varphi(w^l) := v^{i_l}$ and $\varphi(w) := \varphi_l(w)$ for all w in $\mathbf{B}(\mathcal{H}^\epsilon(w^l))$. This implies that homomorphism φ from $\mathbf{B}(\mathcal{H}^\epsilon)$ to $\mathbf{B}(\mathcal{G}^\epsilon)$ is defined.

Proposition 2 *Let C and D be $\mathcal{ALE}$-concept descriptions, and let $\mathcal{G}^\epsilon_C$ and $\mathcal{G}^\epsilon_D$ be their normalization graphs. Algorithm 2 applied to $\mathcal{G}^\epsilon_C$ and $\mathcal{G}^\epsilon_D$ can decide subsumption between C and D in polynomial space and exponential time.*

A proof of Proposition 2 can be found in Appendix.

4. Product of normalization graphs

This section introduces the product operation of normalization graphs, which is extended from the product operation of description trees (as defined in [3]). In this extension, ϵ-edges including ϵ-cycles will be treated as ordinary edges. In particular, for an ϵ-cycle $(v \epsilon v)$, we say also that v is an ϵ-successor of itself.

Additionally, we need the notion of induced subgraph to treat nodes whose label is equal to $\{\bot\}$. An *induced subgraph* $\mathcal{G}^\epsilon(v^k_1, \ldots, v^k_m)$ of graph $\mathcal{G}^\epsilon$ where $v^k_1, \ldots, v^k_m$ are nodes at level k of $\mathcal{G}^\epsilon$, consists of the set of nodes $v^k_1, \ldots, v^k_m$ and their descendants in $\mathcal{G}^\epsilon$ together with all edges whose endpoints are both in this set of nodes. More precisely, let $\mathcal{G}^\epsilon = (V_G, E_G \cup E^\epsilon_G, l_G)$ and $v^k_1, \ldots, v^k_m \in V_G$. We define an induced subgraph $\mathcal{G}^\epsilon(v^k_1, \ldots, v^k_m) = (V_{G_k}, E_{G_k} \cup E^\epsilon_{G_k}, l_{G_k})$ where

$V_{G_k} := \{v^l \in V_G | \ \mathcal{P}^{l-k}(v^l) \cap \{v^k_1, \ldots, v^k_m\} \neq \emptyset, l \geq k\}$,

$E_{G_k} := \{(v^l e v^{l+1}) \in E_G | \ v^l, v^{l+1} \in V_{G_k}\}$,

$E^\epsilon_{G_k} := \{(v^l \epsilon v'^l) \in E^\epsilon_G | \ v^l, v'^l \in V_{G_k}\}$, and

$l_{G_k}(v) := l_G(v)$ for all $v \in V_{G_k}$.

Note that, from the definition of predecessor function $\mathcal{P}$ for normalization graphs $\mathcal{G}^\epsilon$, a node $v^l \in V_G$ is neither r-successor nor $\forall r$-successor but v^l may belong to an induced subgraph $\mathcal{G}^\epsilon(v^k_1, \ldots, v^k_m)$ if $\mathcal{P}^{l-k}(v^l) \cap \{v^k_1, \ldots, v^k_m\} \neq \emptyset$.

In order to transform a subgraph $\mathcal{G}^\epsilon(v^k_1, \ldots, v^k_m)$ into a description tree, we can apply Algorithm 1 to the graph obtained from $\mathcal{G}^\epsilon(v^k_1, \ldots, v^k_m)$ by replacing the nodes $v^k_1, \ldots, v^k_m$ with a unique node v_0 which is considered as the root of the subgraph. The label of v_0 is set to $\mathsf{label}(v^k_1, ..., v^k_m)$, the outgoing edges of $v^k_1, \ldots, v^k_m$ become those of v_0, and the ϵ-edges between nodes $v^k_1, \ldots, v^k_m$ become the ϵ-cycle of v_0. In particular, if $\{v^k_1, \ldots, v^k_m\}$ is a k-neighbourhood in $\mathcal{G}^\epsilon$ then $\mathcal{G}^\epsilon(v^k_1, ..., v^k_m)$ contains all l-neighbourhoods generated from $\{v^k_1, \ldots, v^k_m\}$ where $l \geq k$. Algorithm 1 yields that the nodes and edges of the tree $\mathbf{B}(\mathcal{G}^\epsilon(v^k_1, \ldots, v^k_m))$ can be obtained from these neighbourhoods.

Definition 10 *Let $\mathcal{G}^\epsilon = (V_G, E_G \cup E^\epsilon_G, l_G)$, $\mathcal{H}^\epsilon = (V_H, E_H \cup E^\epsilon_H, l_H)$ be two normalization graphs where v^0 and w^0 are the roots respectively of $\mathcal{G}^\epsilon$ and $\mathcal{H}^\epsilon$. If $l_G(v^0)$=$\{\bot\}$ ($l_H(w^0)$=$\{\bot\}$) then we define $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$ as a graph obtained from $\mathcal{H}^\epsilon$ ($\mathcal{G}^\epsilon$) by replacing each node w in $\mathcal{H}^\epsilon$ (each node v in $\mathcal{G}^\epsilon$) with (v^0, w) ((v, w^0)). Otherwise, the node (v^0, w^0) labeled with $l_G(v^0) \cap l_H(w^0)$ is the root of $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$. Furthermore, $\mathcal{P}(v^0, w^0)$ is set to $(\mathcal{P}(v^0), \mathcal{P}(w^0))$ and an ϵ-cycle $((v^0, w^0)\epsilon(v^0, w^0))$ is obtained from the ϵ-cycles $(v^0 \epsilon v^0)$, $(w^0 \epsilon w^0)$.*
At each level k such that $0 < k \leq min(|\mathcal{G}^\epsilon|, |\mathcal{H}^\epsilon|)$,

1. *For each r-successor ($\forall r$-successor) v^k_i of v^{k-1}_i in $\mathcal{G}^\epsilon$ and each r-successor ($\forall r$-successor) w^k_j of w^{k-1}_j in $\mathcal{H}^\epsilon$ such that (v^{k-1}_i, v^{k-1}_j) is a node in $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$, we obtain a r-successor ($\forall r$-successor) (v^k_i, w^k_j) of (v^{k-1}_i, v^{k-1}_j) in $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$. Additionally, for each ϵ-successor $v^k_{l'}$ of v^k_l in $\mathcal{G}^\epsilon$ and for each ϵ-successor $w^k_{h'}$ of w^k_h in $\mathcal{H}^\epsilon$ such that (v^k_l, w^k_h) is a node created in $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$, we obtain an ϵ-successor $(v^k_{l'}, w^k_{h'})$ of (v^k_l, w^k_h) in $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$. If $l_G(v^k_i)$ and $l_H(w^k_j)$ $\neq \{\bot\}$ (or $l_G(v^k_{l'})$ and $l_H(w^k_{h'}) \neq \{\bot\}$) then the node (v^k_i, w^k_j) (or $(v^k_{l'}, w^k_{h'})$) is labeled with $l_G(v^k_i) \cap l_H(w^k_j)$ (or $l_G(v^k_{l'}) \cap l_H(w^k_{h'})$). Furthermore, $\mathcal{P}(v^k_i, w^k_j)$ (or $\mathcal{P}(v^k_{l'}, w^k_{h'})$) is set to $(\mathcal{P}(v^k_i), \mathcal{P}(w^k_j))$ (or $(\mathcal{P}(v^k_{l'}), \mathcal{P}(w^k_{h'}))$).*
2. *For all nodes (v^k_0, w^k_1), ...,(v^k_0, w^k_n) (or (v^k_1, w^k_0), ...,(v^k_m, w^k_0)) obtained from item 1. such that $l_G(v^k_0) = \{\bot\}$ (or $l_H(w^k_0) = \{\bot\}$), we obtain a subgraph $(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon)((v^k_0, w^k_1), \ldots, (v^k_0, w^k_n))$ (or $(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon)((v^k_1, w^k_0), \ldots, (v^k_m, w^k_0))$) from the induced subgraph $\mathcal{H}^\epsilon(w^k_1, \ldots, w^k_n)$ (or*

$\mathcal{G}^\epsilon(v_1^k, ..., v_m^k)$) by replacing each its node w (or v) with (v_0^k, w) (or (v, w_0^k)) where (v_0^k, w) (or (v, w_0^k)) is labeled with $l_H(w)$ (or $l_G(v)$). Furthermore, $\mathcal{P}(v_0^k, w)$ (or $\mathcal{P}(v, w_0^k)$) is set to $(\mathcal{P}(v_0^k), \mathcal{P}(w))$ (or $(\mathcal{P}(v), \mathcal{P}(w_0^k))$).

From concept descriptions (taken from [4]) given in Example 4, Figure 6 shows how to compute the product graph of normalization graphs built from these concept descriptions and the description tree obtained from the product graph by applying Algorithm 1.

Example 4 *Let*
$C_3 := \exists r.(\forall r.\forall r.P_3^0) \sqcap \exists r.(\forall r.\forall r.P_3^1) \sqcap$
$\forall r.(\exists r.\forall r.P_2^0 \sqcap \exists r.\forall r.P_2^1 \sqcap \forall r.(\exists r.P_1^0 \sqcap \exists r.P_1^1))$
$D_3 := \exists r.\exists r.\exists r.(P_1^0 \sqcap P_1^1 \sqcap P_2^0 \sqcap P_2^1 \sqcap P_3^0 \sqcap P_3^1)$
where $P_j^i \in N_C$, $r \in N_R$.
Note that ϵ-cycles are useful for computing product graphs. For instance, ϵ-cycle of node u_1 of tree $\mathcal{G}_{D_3}^\epsilon$ and ϵ-edge $(v_1 \epsilon v_3)$ of tree $\mathcal{G}_{C_3}^\epsilon$ yield ϵ-edge $((v_1, u_1)\epsilon(v_3, u_1))$ of tree $\mathcal{G}_{C_3}^\epsilon \times \mathcal{G}_{D_3}^\epsilon$.
To simplify the presentation, the ϵ-cycles are not added to the graphs in the figures.

Remark 6 *The size of the product graph of two normalization graphs $\mathcal{G}^\epsilon = (V_G, E_G \cup E_G^\epsilon, l_G)$, $\mathcal{H}^\epsilon = (V_H, E_H \cup E_H^\epsilon, l_H)$ is bounded by the product of the sizes of these normalization graphs. In fact, it holds that $|V_{G\times H}| \leq |V_G| \times |V_H|$, $|E_{G\times H}| \leq |E_G| \times |E_H|$ and $|E_{G\times H}^\epsilon| \leq |E_G^\epsilon| \times |E_H^\epsilon|$.*

In the sequel, we will show that the level notion for product graphs can be defined from those for normalization graphs and the computation of the product of two normalization graphs preserves important properties of the neighbourhood notion.
These notions guarantee that Algorithm 1 and Algorithm 2 can be applied to product graphs.
Let (v^l, w^l) and (v^k, w^k) $(l < k)$ be two nodes in a product graph $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$. There is a path from (v^l, w^l) to (v^k, w^k) iff there are nodes (v^{l+1}, w^{l+1}),... , (v^{k-1}, w^{k-1}) such that $\mathcal{P}(v^k, w^k) \cap \{(v^{k-1}, w^{k-1})\} \neq \emptyset$,... , $\mathcal{P}(v^{l+1}, w^{l+1}) \cap \{(v^l, w^l)\} \neq \emptyset$ where $\mathcal{P}(v^m, w^m) \cap \{(v^{m-1}, w^{m-1})\} \neq \emptyset$ iff $\mathcal{P}(v^m) \cap \{v^{m-1}\} \neq \emptyset$ and $\mathcal{P}(w^m) \cap \{w^{m-1}\} \neq \emptyset$. Therefore, each path from the root (v^0, w^0) to a node (v^k, w^k) corresponds to two paths: the one is from v^0 to v^k in $\mathcal{G}^\epsilon$ and the other is from w^0 to w^k in $\mathcal{H}^\epsilon$. Moreover, the number of ordinary edges of all paths from v^0 (w^0) to v^k (w^k) is constant since $\mathcal{G}^\epsilon$ and $\mathcal{H}^\epsilon$ are normalization graphs. Thus, the number of ordinary edges of all paths from (v^0, w^0) to (v^k, w^k) on $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$ is constant as well. This allows us to define the level of a node (v^k, w^k) as the number of ordinary edges of all paths from the root to (v^k, w^k). It means that two nodes corresponding to the endpoints of any ϵ-edges are always at the same level.
Therefore, Definition 10 can be extended to n-ary product of graphs as follows:
$\mathcal{G}_{C_1}^\epsilon \times \ldots \times \mathcal{G}_{C_n}^\epsilon := (\mathcal{G}_{C_1}^\epsilon \times \ldots \times \mathcal{G}_{C_{n-1}}^\epsilon) \times \mathcal{G}_{C_n}^\epsilon$

Definition 11 *We denote $\mathcal{T}_\mathcal{L}^\mathcal{E}$ as the set of all normalization graphs and product graphs generated from $\mathcal{L}$-concept descriptions, i.e.,*

$$\mathcal{T}_\mathcal{L}^\mathcal{E} := \bigcup_{n\geq 1}\{\mathcal{G}_{C_1}^\epsilon \times \ldots \times \mathcal{G}_{C_n}^\epsilon \mid C_1, ..., C_n \text{ are } \mathcal{L}\text{-concept descriptions}\}$$

In this paper, we investigate $\mathcal{T}_\mathcal{L}^\mathcal{E}$ where $\mathcal{L} \in \{\mathcal{FLE}, \mathcal{ALE}\}$.
We now clarify how the neighbourhood definition (Definition 9) can be applied to product graphs. Similarly to $\forall r$-neighbourhoods in normalization graphs, the computation of the $\forall r$-neighbourhood at level k of a $(k-1)$-neighbourhood in product graphs takes into account the set of nodes V^ϵ in Definition 9. Differently from $\forall r$-neighbourhoods in normalization graphs where sets $V^\epsilon \neq \emptyset$ include only nodes whose label is equal to $\{\bot\}$, sets V^ϵ corresponding to $\forall r$-neighbourhoods in product graphs can contain nodes which have r-successors or $\forall r$-successors. More precisely,

Lemma 4 *Let $n_G^{k-1} = \{u_1, ..., u_m\}$ and $n_H^{k-1} = \{w_1, ..., w_n\}$ be $(k-1)$-neighbourhoods respectively in $\mathcal{G}^\epsilon$, $\mathcal{H}^\epsilon \in \mathcal{T}_{\mathcal{ALE}}^\mathcal{E}$. Let $n_{G\times H}^{k-1}$ be a $(k-1)$-neighbourhood in $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$. Assume that $\{(u_1, w_1), ..., (u_m, w_n)\} \subseteq n_{G\times H}^{k-1}$ and $l_{G\times H}(u_i, w_j) = \emptyset$, (u_i, w_j) does not have any r-successor and $\forall r$-successor for all $(u_i, w_j) \in n_{G\times H}^{k-1} \setminus \{(u_1, w_1), ..., (u_m, w_n)\}$.*
It holds that there exist r-neighbourhoods ($\forall r$-neighbourhoods) $n_G^k = \{v_1, ..., v_h\}$ and $n_H^k = \{z_1, ..., z_l\}$ such that $n_G^k \in N(n_G^{k-1})$ and $n_H^k \in N(n_H^{k-1})$ iff there exists a r-neighbourhood ($\forall r$-neighbourhood) $n_{G\times H}^k \in N(n_{G\times H}^{k-1})$ such that $\{(v_1, z_1), ..., (v_h, z_l)\} \subseteq n_{G\times H}^k$ and $l_{G\times H}(v_i, z_j) = \emptyset$, (v_i, z_j) does not have any r-successor and $\forall r$-successor for all $(v_i, z_j) \in n_{G\times H}^k \setminus \{(v_1, z_1), ..., (v_h, z_l)\}$.

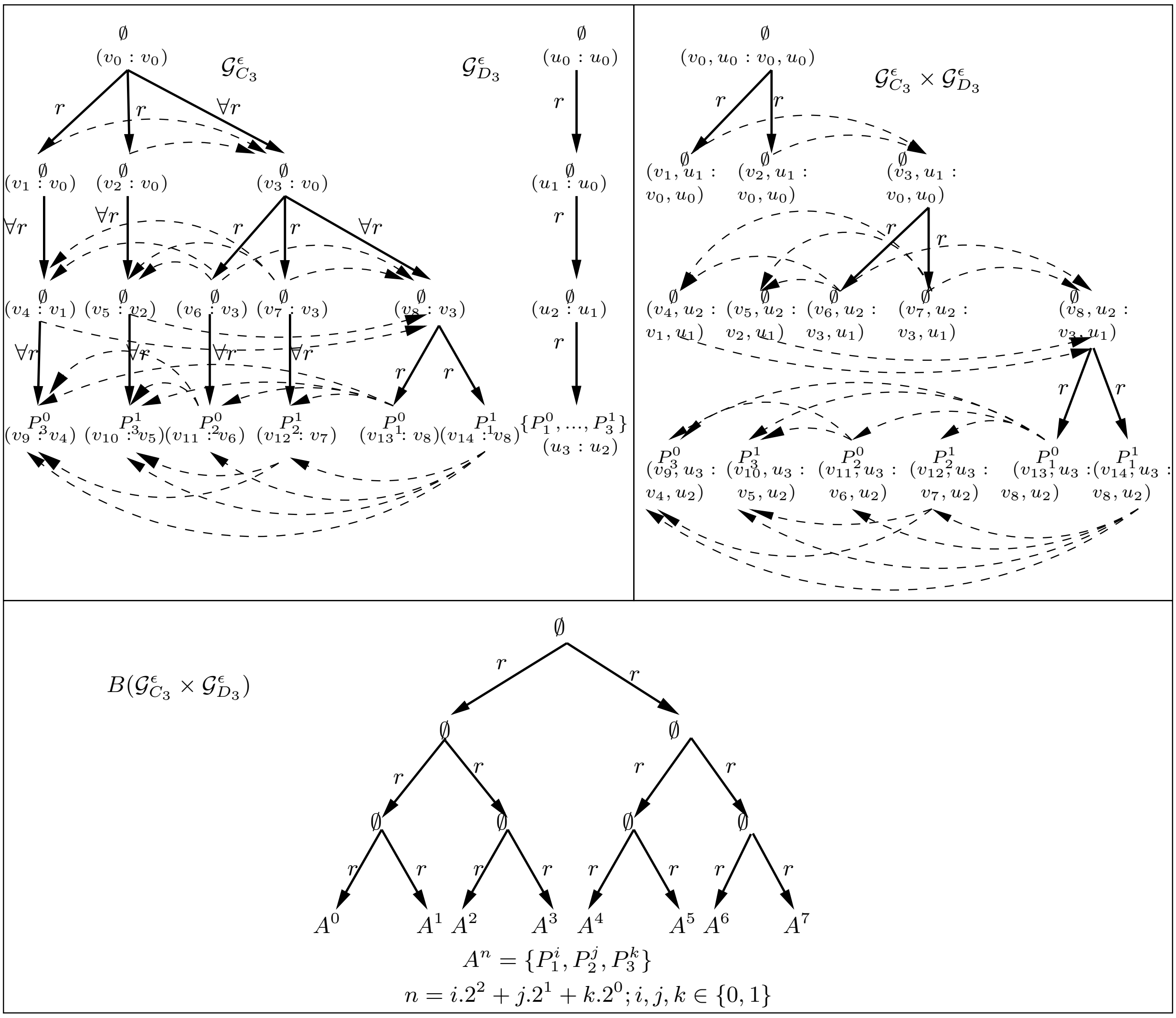

Figure 6. Product of normalization graphs

A proof of Lemma 4 can be found in Appendix. We are now ready to formulate and prove a theorem which establishes the relationship between the product of two graphs in $\mathcal{T}^{\mathcal{E}}$ and the product of two description trees (as defined in [3]) represented by these two graphs.

Theorem 2 *Let $\mathcal{G}^{\epsilon}$, $\mathcal{H}^{\epsilon} \in \mathcal{T}^{\mathcal{E}}_{\mathcal{ALE}}$. There exists an isomorphism between $\mathbf{B}(\mathcal{G}^{\epsilon} \times \mathcal{H}^{\epsilon})$ and $\mathbf{B}(\mathcal{G}^{\epsilon}) \times \mathbf{B}(\mathcal{H}^{\epsilon})$.*

A proof of Theorem 2 can be found in Appendix. This proof builds inductively on the level of graphs an isomorphism between the trees $\mathbf{B}(\mathcal{G}^{\epsilon} \times \mathcal{H}^{\epsilon})$ and $\mathbf{B}(\mathcal{G}^{\epsilon}) \times \mathbf{B}(\mathcal{H}^{\epsilon})$. In fact, assume that for each $(k-1)$-neighbourhood $n^{k-1}_{G\times H}$ in $(\mathcal{G}^{\epsilon} \times \mathcal{H}^{\epsilon})$ we have two corresponding $(k-1)$-neighbourhoods n^{k-1}_G, n^{k-1}_H on $\mathcal{G}^{\epsilon}$ and $\mathcal{H}^{\epsilon}$, respectively. The proof shows that $n^k_G = (u_1, ..., u_m)$, $n^k_H = (w_1, ..., w_n)$ are k-neighbourhoods respectively of n^{k-1}_G and n^{k-1}_H such that $\mathsf{label}(n^k_G) \neq \{\bot\}$, $\mathsf{label}(n^k_H) \neq \{\bot\}$ iff $n^k_{G\times H}$ is a k-neighbourhood of $n^{k-1}_{G\times H}$ such that $\{(u_1, w_1), ..., (u_m, w_n)\} \subseteq n^k_{G\times H}$ and $l_{G\times H}(v_i, z_j) = \emptyset$, (v_i, z_j) does not have any r-successor and $\forall r$-successor for all $(v_i, z_j) \in n^k_{G\times H} \setminus \{(v_1, z_1), ..., (v_h, z_l)\}$. The proof of this claim is based heavily on Lemma 4. Additionally, if $\mathsf{label}(n^k_G) = \{\bot\}$ or $\mathsf{label}(n^k_H) = \{\bot\}$ then, Definition 10 yields that $(\mathcal{G}^{\epsilon} \times \mathcal{H}^{\epsilon})(n^k_{G\times H})$ is equal to

$\mathcal{H}^\epsilon(n_H^k)$ or $\mathcal{G}^\epsilon(n_G^k)$ (up to renaming nodes). Thus, $\mathbf{B}((\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon)(n_{G\times H}^k))$ is equal to $\mathbf{B}(\mathcal{H}^\epsilon(n_H^k))$ or $\mathbf{B}(\mathcal{G}^\epsilon(n_G^k))$. The construction of the isomorphism will be done by proving that $\mathsf{label}(n_G^k) = \{\bot\}$ and $\mathsf{label}(n_H^k) = \{\bot\}$ iff $\mathsf{label}(n_{G\times H}^k) = \{\bot\}$.

Proposition 1 yields that $\mathcal{G}_C$ and $\mathcal{G}_D$ are equal respectively to $\mathbf{B}(\mathcal{G}_C^\epsilon)$ and $\mathbf{B}(\mathcal{G}_D^\epsilon)$ (up to renaming nodes) and Proposition 2 shows that it is sufficient to use normalization graphs $\mathcal{G}_C^\epsilon$, $\mathcal{G}_D^\epsilon$ rather than description trees $\mathcal{G}_C$ and $\mathcal{G}_D$ to decide subsumption between two $\mathcal{ALE}$-description concepts C and D. Moreover, according to an important result in [3], the *lcs* of C and D can be computed as the product $\mathcal{G}_C \times \mathcal{G}_D$. This result and Theorem 2 allow us to represent all *lcs* as product graphs $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$ and decide subsumption between *lcs* by manipulating directly the corresponding product graphs. Thus, we can export the semantics of concept descriptions to graphs of $\mathcal{T}_{\mathcal{ALE}}^{\mathcal{E}}$, *i.e.,* for an interpretation $(\Delta, .^{\mathcal{I}})$, we define $(\mathcal{G}^\epsilon)^{\mathcal{I}} := (C_{B(\mathcal{G}^\epsilon)})^{\mathcal{I}}$. By consequent, we can talk about the subsumption, equivalence, *lcs*, *etc.* for all graphs $\mathcal{G}^\epsilon \in \mathcal{T}_{\mathcal{ALE}}^{\mathcal{E}}$. The following result is a direct consequence of Theorem 2.

Corollary 1 *Let $\mathcal{G}^\epsilon$, $\mathcal{H}^\epsilon \in \mathcal{T}_{\mathcal{ALE}}^{\mathcal{E}}$. The least common subsumer of $\mathcal{G}^\epsilon$, $\mathcal{H}^\epsilon$ can be computed in polynomial space and exponential time .*

Since $\mathcal{G}_C^\epsilon = \mathcal{G}^\epsilon(C)$ if C is a $\mathcal{FLE}$-concept description, the complexity for computing $\mathcal{G}_C^\epsilon$ from C is polynomial in the size of C. Therefore,

Corollary 2 *Let $\mathcal{G}^\epsilon$, $\mathcal{H}^\epsilon \in \mathcal{T}_{\mathcal{FLE}}^{\mathcal{E}}$. The least common subsumer of $\mathcal{G}^\epsilon$, $\mathcal{H}^\epsilon$ can be computed in polynomial time.*

Additionally, according to the definition of *lcs*, we have $C = lcs(C, \bot)$ for every $\mathcal{ALE}$-concept description C. Therefore, Proposition 2 can be generalized as follows.

Proposition 3 *Let $\mathcal{G}^\epsilon$ and $\mathcal{H}^\epsilon$ be two product graphs corresponding to $lcs(C_1, C_2)$ and $lcs(D_1, D_2)$ where C_1, C_2, D_1 and D_2 are $\mathcal{ALE}$-concept descriptions, i.e., $\mathcal{G}^\epsilon = \mathcal{G}_{C_1}^\epsilon \times \mathcal{G}_{C_2}^\epsilon$ and $\mathcal{H}^\epsilon = \mathcal{G}_{D_1}^\epsilon \times \mathcal{G}_{D_2}^\epsilon$. Algorithm 2 applied to $\mathcal{G}^\epsilon$ and $\mathcal{H}^\epsilon$ can decide subsumption between $lcs(C_1, C_2)$ and $lcs(D_1, D_2)$ in polynomial space and exponential time.*

5. On the approximation $\mathcal{ALC}$-$\mathcal{ALE}$

In [1], a double exponential algorithm has been proposed for the approximation $\mathcal{ALC}$-$\mathcal{ALE}$. In this algorithm, the approximation is computed by using the *lcs*. A question left open by the authors concerns the existence of an exponential algorithm for computing the approximation. In the first attempt at finding an answer to this question, we hoped that if there is a method for obtaining a polynomial representation for the *lcs*, such a method may be applied for reducing the exponential blow-up caused by the distribution of disjunctions over conjunctions in the normalization for $\mathcal{ALC}$-concept descriptions. However, though the polynomial representation for the *lcs* presented in Section 4 helps to reduce the size of the approximation, this representation does not allow for reducing the complexity class.
In this section, we formulate and prove a theorem which provides a tight lower bound of the size of the approximation $\mathcal{ALC}$-$\mathcal{ALE}$ in the ordinary representation.

Algorithm 3 $approx_{\mathcal{ALE}}(C)$

Require: C is an $\mathcal{ALC}$-concept description in $\mathcal{ALC}$-normal form $C = C_1 \sqcup \ldots \sqcup C_n$.
Ensure: $approx_{\mathcal{ALE}}(C)$
 if $C \equiv \bot$ **then**
  return $\bot$;
 end if
 if $C \equiv \top$ **then**
  return $\top$;
 else
  return
$$\bigsqcap_{A \in \bigcap_{i=1}^m Prim(C_i)} A \sqcap \bigsqcap_{(C'_1,\ldots,C'_m)\in Ex(C_1)\times\ldots\times Ex(C_m)} \{\exists r.lcs\{approx_{\mathcal{ALE}}(C'_j \sqcap Val(C_j))|1 \le j \le m\}\} \sqcap \forall r.lcs\{approx_{\mathcal{ALE}}(Val(C_j))|1 \le j \le m\}$$
 end if

Theorem 3 *Let C be an $\mathcal{ALC}$-concept description. The size of $approx_{\mathcal{ALE}}(C)$ may be double exponential in the size of C.*

The following proof of Theorem 3 uses some notions and the approximation algorithm described in [1].

Let C is an $\mathcal{ALC}$-concept description where disjunction only occurs within value or existential restrictions. $Prim(C)$ denotes the set of all (negated) concept names occurring on the top-level conjunction of C (the top-level conjunction is not wrapped within a value restriction or existential restriction). $Val(C)$ is the conjunction of all C' occurring in value restrictions of form $\forall r.C'$ on top-level of C. If there is no value restriction on top-level of C then $Val(C) = \top$. $Ex(C)$ is the set of all C' occurring in existential restrictions of form $\exists r.C'$ on top-level of C. The normal form of C is defined as follows. Let C be an $\mathcal{ALC}$-concept description and $C \not\equiv \top$, $C \not\equiv \bot$. C is in $\mathcal{ALC}$-normal form iff C is of the form $C = C_1 \sqcup ... \sqcup C_m$ such that $C_i = \sqcap_{A \in Prim(Ci)} A \sqcap \prod_{C' \in Ex(Ci)} \exists r.C' \sqcap \forall r.Val(C_i)$, $\bot \sqsubset C_i$ where C',$Val(C_i)$ are in $\mathcal{ALC}$-normal form.

If C is in $\mathcal{ALC}$-normal form, the approximation of C can be computed by Algorithm 3 (the approximation algorithm in [1]).

Considering the algorithm, the $\mathcal{ALC}$-normal form of an $\mathcal{ALC}$-concept description C may contain 2^n disjuncts (assume that the initial form of C is the conjunction of n conjuncts and each conjunct is a binary disjunction). Furthermore, the value and existential restrictions in $approx_{\mathcal{ALE}}(C)$ may require to compute the lcs of 2^n terms. If the lcs under the existential restrictions do not subsume each other (absorption), the approximation may contain a double exponential number (2^{2^n}) of existential restrictions which do not subsume each other. This remark is useful for constructing the proof of Theorem 3.

We now characterize some properties that an $\mathcal{ALC}$-concept description C leading to the exponential blow-up should satisfy:

(1) C is the conjunction of n conjuncts and each conjunct is a binary disjunction. Therefore, the $\mathcal{ALC}$-normal form of C has 2^n disjuncts and each disjunct is the conjunction including n conjuncts.

(2) From Algorithm 3, the approximation contains 2^{2^n} existential restrictions $\exists r.lcs\{E_{i1}, ..., E_{ik}\}$ where $k = 2^n$, $i \in \{1, ..., 2^{2^n}\}$. Each E_{ij} should be an existential restriction $\exists r.E_{ij}$ where E_{ij} is the conjunction of concept names belonging to $\{P^u_l, Q_v\}$ for $u, v \in \{1, 2\}$, $l \in \{1, ..., n\}$. Thus, each $lcs\{E_{i1}, ..., E_{ik}\}$ is the conjunction of $\exists r.F^j_i$, and each F^j_i is conjunction of concept names belonging to $\{P^u_r, Q_v\}$ for $u, v \in \{1, 2\}$, $r \in \{1, ..., n\}$.

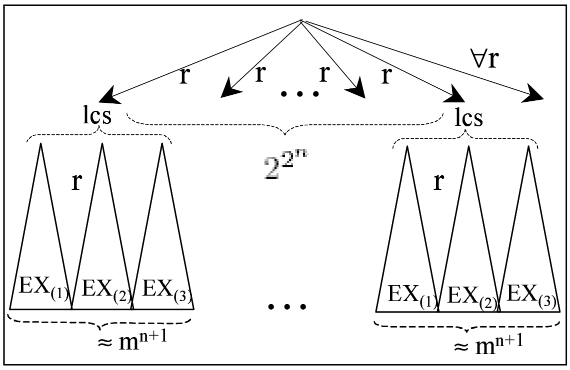

Figure 7. Double exponential $approx_{\mathcal{ALE}}(C)$

Each F^j_i can be considered as a subset of $\{P^u_r, Q_v \mid u, v \in \{1, 2\}, l \in \{1, ..., n\}\}$.

(3) The essential property of $\exists r.lcs\{E_{i1}, ..., E_{ik}\}$ for $k = 2^n$, $i \in \{1, ..., 2^{2^n}\}$ to be guaranteed, is that they do not subsume each other. This means that for each pair of existential restrictions $\exists r.lcs\{E_{i1}, ..., E_{ik}\}$ and $\exists r.lcs\{E_{j1}, ..., E_{jk}\}$, $i, j \in \{1, ..., 2^{2^n}\}$, there exists a conjunct $\exists r.F^r_i$ of $lcs\{E_{i1}, ..., E_{ik}\}$ such that $\exists r.F^r_i \not\sqsubseteq \exists r.F^s_j$ for all conjuncts $\exists r.F^s_j$ of $lcs\{E_{j1}, ..., E_{jk}\}$, and vice versa.

The difficulty in proving the property (3) is due the computing of $lcs\{E_{i1}, ..., E_{ik}\}$. This task may become easier if we partition existential restrictions obtained from $lcs\{E_{i1}, ..., E_{ik}\}$ into groups and identify representative elements of each group. Therefore, we only need to consider representative elements for deciding whether $lcs\{E_{i1}, ..., E_{ik}\}$ is absorbed by $lcs\{E_{j1}, ..., E_{jk}\}$.

The proof of Theorem 3 needs the following lemma.

Lemma 5 *Let* $(A^{i_1}_1, ..., A^{i_n}_n)$ *be a n-dimension vector where* $i_j \in \{1, 2\}$. *Let* I *be a bijection from* $\{1, 2\} \times ... \times \{1, 2\}$ *into* $\{1, .., 2^n\}$ *for numbering all vectors* $\{(A^{i_1}_1, ..., A^{i_n}_n) \mid (i_1, ..., i_n) \in \{1, 2\} \times ... \times \{1, 2\}\}$. *We have that each* $(i_1, ..., i_n) \in \{1, 2\} \times ... \times \{1, 2\}$ *determines uniquely* $k \in \{1, .., 2^n\}$ *such that* $I(\bar{i}_1, ..., \bar{i}_n) = k$ *where* $\bar{i}_h \neq i_h$ *for all* $h \in \{1..n\}$.

The proof of the lemma is trivial since $(\bar{i}_1, ..., \bar{i}_n) \in \{1, 2\} \times ... \times \{1, 2\}$ for each $(i_1, ..., i_n) \in \{1, 2\} \times ... \times \{1, 2\}$ and I is a bijection.

Proof of Theorem 3.

The theorem will be proven if there exists an $\mathcal{ALC}$-concept description C such that the size of

$approx_{\mathcal{ALE}}(C)$ is double exponential in the size of C and f $approx_{\mathcal{ALE}}(C)$ is irreducible. Let
$A^1_k = \exists r.(P^1_k \sqcap \bigsqcap_{i=1..n, i\neq k}(P^1_i \sqcap P^2_i) \sqcap Q_1 \sqcap Q_2)$,
$A^2_k = \exists r.(P^2_k \sqcap \bigsqcap_{i=1..n, i\neq k}(P^1_i \sqcap P^2_i) \sqcap Q_1 \sqcap Q_2)$

for $k \in \{1, ..., n\}$,

$B_1 = \exists r.(Q_1 \sqcap \bigsqcap_{i=1..n}(P^1_i \sqcap P^2_i))$,
$B_2 = \exists r.(Q_2 \sqcap \bigsqcap_{i=1..n}(P^1_i \sqcap P^2_i))$ where $P^i_k, Q_j \in N_C$, $r \in N_R$ for $i, j \in \{1, 2\}$, $k \in \{1, ..., n\}$.

Let C be an $\mathcal{ALC}$-concept description:

$$C := \exists r.B_1 \sqcap \exists r.B_2 \sqcap \bigsqcap_{i=1}^{n}(\forall r.A^1_i \sqcup \forall r.A^2_i)$$

We prove that the number of top-level existential restrictions of $approx_{\mathcal{ALE}}(C)$ is 2^{2^n} and these existential restrictions do not subsume each other.
The $\mathcal{ALC}$-normal form of C is as follows:
$C \equiv C_1 \sqcup ... \sqcup C_m$ where
$C_i \equiv (\exists r.B_1 \sqcap \exists r.B_2 \sqcap \forall r.Val(C_i))$ and
$Val(C_i) = A^{j_1}_1 \sqcap ... \sqcap A^{j_n}_n$, $(j_1, ..., j_n) \in (\{1, 2\} \times ... \times \{1, 2\})$.
According to Algorithm 3, we have:

$$approx_{\mathcal{ALE}}(C) = \bigsqcap_{(B_{i_1}, ..., B_{i_m}) \in (\{B_1, B_2\} \times ... \times \{B_1, B_2\})} \{\exists r.lcs\{(B_{i_j} \sqcap Val(C_j)) | 1 \leq j \leq m\}\} \sqcap \forall r.lcs\{Val(C_j) | 1 \leq j \leq m\} \ (*)$$

Figure 7 shows 2^{2^n} existential restrictions on top-level of the expression (*). The expressions under these existential restrictions are lcs and each one applies to $m = 2^n$ terms. We denote E as the set of of existential restrictions obtained from computing $lcs\{(B_{i_j} \sqcap Val(C_j)) | 1 \leq j \leq m\}$. According to the computing of the n-ary lcs, E may contain m^{n+1} existential restrictions. E is partitioned into three subsets $EX_{(1)}$, $EX_{(2)}$, $EX_{(3)}$ as follows.
Since each tuple $(B_{i_1}, ..., B_{i_m}) \in \{B_1, B_2\} \times ... \times \{B_1, B_2\}$ determines a set E of existential restrictions, we define a set function $\mathcal{E}$ from the domain $\{(B_{i_1}, ..., B_{i_m}) | \ (B_{i_1}, ..., B_{i_m}) \in \{B_1, B_2\} \times ... \times \{B_1, B_2\}\}$ to the set of sets of existential restrictions obtained from the computing of $lcs\{(B_{i_j} \sqcap Val(C_j)) | 1 \leq j \leq m\}$ for all $(i_1, ..., i_m) \in \{1, 2\} \times ... \times \{1, 2\}$.
Each set $\mathcal{E}(X_1, ..., X_m)$ where $(X_1, ..., X_m) \in \{B_1, B_2\} \times ... \times \{B_1, B_2\}$ may contain m^{n+1} existential restrictions but some of them can be subsumed by others. In fact, $lcs\{(X_j \sqcap Val(C_j)) | 1 \leq j \leq m\}$ can be computed as conjunction of $lcs\{E_{i_1}, ..., E_{i_m}\}$ where $E^r_{i_r} \in \{X_r\} \cup Val\{C_r\}$ for all $r \in \{1, ..., m\}$. If $\{E_{i_1}, ..., E_{i_m}\} \subseteq \{E_{l_1}, ..., E_{l_m}\}$ then $lcs\{E_{i_1}, ..., E_{i_m}\} \sqsubseteq lcs\{E_{l_1}, ..., E_{l_m}\}$. Furthermore, we define a selection function:
$\mathcal{S}(X_1, ..., X_m) := \{(E_{i_1}, ..., E_{i_m}) | \ E_{i_r} \in \{X_r\} \cup Val(C_r), r \in \{1, ..., m\}\}$. This implies that
$\mathcal{E}(X_1, ..., X_m) = \{lcs\{E_{i_1}, ..., E_{i_m}\} | \ (E_{i_1}, ..., E_{i_m}) \in \mathcal{S}(X_1, ..., X_m)\}$.
We will identify from all them the representative existential restrictions which form the three following subsets $EX_{(1)}$, $EX_{(2)}$ and $EX_{(3)}$:

1. $EX_{(1)}(X_1, ..., X_m)$ is composed of the existential restrictions (of $\mathcal{E}(X_1, ..., X_m)$) that subsume the following existential restrictions: $lcs(A^1_k, A^2_k) \equiv \exists r.(Q_1 \sqcap Q_2 \sqcap \bigsqcap_{l=1, l\neq k}^{n}(P^1_l \sqcap P^2_l))$ for $k \in \{1..n\}$.
   It is obvious that for each $k \in \{1, ..., n\}$ there exists $(E_{i_1}, ..., E_{i_m}) \in \mathcal{S}(X_1, ..., X_m)$ such that $(E_{i_1}, ..., E_{i_m}) = \{A^1_k, A^2_k\}$ where $E_{i_j} = A^1_k \in Val'(C_j)$ or $E_{i_j} = A^2_k \in Val(C_j)$ for all $j \in \{1..m\}$. Note that $Val'(C_j)$ is denoted for the set of conjuncts in C_j.
2. $EX_{(2)}(X_1, ..., X_m)$ is composed of the existential restrictions that subsume the following existential restrictions:
   $lcs(B_i, B_j) \equiv \exists r.(\bigsqcap_{k=1}^{n}(P^1_k \sqcap P^2_k))$ if there exists $X_p, X_q \in \{X_1, ..., X_m\}$ and $X_p \neq X_q$, or $lcs(B_i, B_i) \equiv \exists r.(Q_i \sqcap \bigsqcap_{k=1}^{n}(P^1_k \sqcap P^2_k))$ for $i \in \{1, 2\}$ if $X_1 = ... = X_m$.
   It is obvious that: $lcs(B_i, B_j) \in \mathcal{E}(X_1, ..., X_m)$ if there exist $X_p, X_q \in \{X_1, ..., X_m\}$, $X_p \neq X_q$. In fact, there exists $(E_{i_1}, ..., E_{i_m}) \in \mathcal{S}(X_1, ..., X_m)$ such that $(E_{i_1}, ..., E_{i_m}) = \{B_i, B_j\}$ where $E_{i_r} = B_i \in \{X_r\} \cup Val(C_r)$ or $E_{i_r} = B_j \in \{X_r\} \cup Val(C_r)$ for $r \in \{1..m\}$. Similarly, $lcs\{B_i, B_i\} \in \mathcal{E}(X_1, ..., X_m)$ if $X_1 = ... = X_m$.
3. $EX_{(3)}(X_1, ..., X_m)$ is composed of the existential restrictions that are subsumed by the following existential restrictions:
   $lcs\{B_{i_k}, A^{l_1}_1, A^{l_2}_2, ..., A^{l_n}_n\}$ where $(l_1, ..., l_n) \in \{1, 2\} \times ... \times \{1, 2\}$, $k = I(\bar{l}_1, ..., \bar{l}_n)$. The function I is defined as follows: each conjunct $\exists r.lcs\{(X_j \sqcap Val(C_j)) | 1 \leq j \leq m\}$ on top-level of $approx_{\mathcal{ALE}}(C)$ where $Val(C_j) = A^{l_1}_1 \sqcap ... \sqcap A^{l_n}_n$, determines $I(l_1, ..., l_n) = j$ where $j = (l_1 - 1).2^{(n-1)} + ... + (l_n - 1).2^0 + 1$ (the binary value of $(l_1, ..., l_n)$ plus 1). It is obvious that I is a bijection. According to Lemma 5, $k = I(\bar{l}_1, ..., \bar{l}_n)$ is uniquely determined from $(l_1, ..., l_n)$ (**).

We have that $lcs\{X_k, A_1^{l_1}, A_2^{l_2}, ..., A_n^{l_n}\} \in \mathcal{E}(X_1, ..., X_m)$ for some $(l_1, ..., l_n) \in \{1,2\} \times ... \times \{1,2\}$, $k = I(\bar{l}_1, ..., \bar{l}_n)$. This is implied from Lemma 5, *i.e.,* for each $(l_1, ..., l_n) \in \{1,2\} \times ... \times \{1,2\}$ there exists $(E_{i_1}, ..., E_{i_m}) \in \mathcal{S}(X_1, ..., X_m)$ such that $(E_{i_1}, ..., E_{i_m}) = \{X_k, A_1^{l_1}, A_2^{l_2}, ..., A_n^{l_n}\}$ where $E_{i_k} = X_k$ for $k{=}I(\bar{l}_1, ..., \bar{l}_n)$ and $E_{i_r} = A_s^{l_s} \in Val(C_r)$ for some $s \in \{1..n\}$, $r \neq k$.

To show $\mathcal{E}(X_1, ..., X_m) = EX_{(1)}(X_1, ..., X_m) \cup EX_{(2)}(X_1, ..., X_m) \cup EX_{(3)}(X_1, ..., X_m)$, we only need to show that if $e \in \mathcal{E}(X_1, ..., X_m)$, $e \notin EX_{(2)}(X_1, ..., X_m)$ and $e \notin EX_{(1)}(X_1, ..., X_m)$, then $e \in EX_{(3)}(X_1, ..., X_m)$.
Indeed, if $e \notin EX_{(2)}(X_1, ..., X_m)$ and $e \notin EX_{(1)}(X_1, ..., X_m)$ then e has to be of the form $lcs\{B_{i_k}, A'_1, A'_2, ..., A'_p\}$ such that $B_{i_k} \in \{B_1, B_2\}$ and $\{A'_1, ..., A'_p\} \subseteq \{A_1^{j_1}, ..., A_n^{j_n}\}$ for some $(j_1, ..., j_n) \in \{1,2\} \times ... \times \{1,2\}$. Note that if there exists $h \in \{1..n\}$ such that $A_h^1, A_h^2 \in \{A'_1, ..., A'_p\}$ then $e \in EX_{(1)}(X_1, ..., X_m)$. Moreover, we have $B_{i_k} = X_k$ where $k{=}I(\bar{j}_1, ..., \bar{j}_n)$ since $A_h^{j_h} \notin Val'(C_k)$ for all $h \in \{1..n\}$.
This means that if for all $(E_{i_1}, ..., E_{i_m}) \in \mathcal{S}(X_1, ..., X_m)$ such that A_h^1 *and* $A_h^2 \notin \{E_{i_1}, ..., E_{i_m}\}$ for all $h \in \{1..n\}$ but B_1 or $B_2 \in \{E_{i_1}, ..., E_{i_m}\}$ and $A_{s_1}^{i_{s_1}}$,..., $A_{s_p}^{i_{s_p}} \in \{E_{i_1}, ..., E_{i_m}\}$ for $s_1, ..., s_p \in \{1..n\}$, then $\{E_{i_1}, ..., E_{i_m}\}{=}\ \{X_k, A_{s_1}^{i_{s_1}}$,..., $A_{s_p}^{i_{s_p}}\}$ such that $\{A_{s_1}^{i_{s_1}}$,..., $A_{s_p}^{i_{s_p}}\} \subseteq \{A_1^{j_1}, ..., A_n^{j_n}\}$ for some $(j_1, ..., j_n) \in \{1,2\} \times ... \times \{1,2\}$ and $E_{i_k}{=}X_k{=}B_1$ or B_2 for $k{=}I(\bar{j}_1, ..., \bar{j}_n)$.
Therefore, $e \sqsubseteq lcs\{B_{i_k}, A_1^{j_1}, A_2^{j_2}, ..., A_n^{j_n}\}$ and thus $e \in e \in EX_{(3)}(X_1, ..., X_m)$.

We now prove that for each couple $(X_1, ..., X_m)$, $(Y_1, ..., Y_m) \in \{B_1, B_2\} \times ... \times \{B_1, B_2\}$, $(X_1, ..., X_m) \neq (Y_1, ..., Y_m)$, the following properties are verified:
$lcs\{(X_u \sqcap Val(C_u)) | 1 \leq u \leq m\} \not\sqsubseteq lcs\{(Y_v \sqcap Val(C_v)) | 1 \leq v \leq m\}$, and
$lcs\{(Y_v \sqcap Val(C_v)) | 1 \leq v \leq m\} \not\sqsubseteq lcs\{(X_u \sqcap Val(C_u)) | 1 \leq u \leq m\}$.
According to the definition of function $\mathcal{E}$, these properties are reformulated as follows:

There exists $e' \in \mathcal{E}(Y_1, ..., Y_m)$ such that $e'' \not\sqsubseteq e'$ for all $e'' \in \mathcal{E}(X_1, ..., X_m)$ and,
there exists $e'' \in \mathcal{E}(X_1, ..., X_m)$ such that $e' \not\sqsubseteq e''$ for all $e' \in \mathcal{E}(Y_1, ..., Y_m)$.

Owing to the partition of $\mathcal{E}$ into the subsets $EX_{(1)}$, $EX_{(2)}$, $EX_{(3)}$, considering only representative existential restrictions of the subsets is enough to prove the properties above.

Let $(X_1, ..., X_m), (Y_1, ..., Y_m) \in \{B_1, B_2\} \times ... \times \{B_1, B_2\}$ and $k_0 \in \{1, .., m\}$ such that $X_{k_0} \neq Y_{k_0}$. Without loss of generality, assume that:
$Y_{k_0} = B_1 = \exists r.(Q_1 \sqcap \bigsqcap_{l=1}^{n}(P_l^1 \sqcap P_l^2))$ and $X_{k_0} = B_2 = \exists r.(Q_2 \sqcap \bigsqcap_{l=1}^{n}(P_l^1 \sqcap P_l^2))$. We pick $e'' = \exists r.(Q_2 \sqcap P_1^{j_1} \sqcap ... \sqcap P_n^{j_n})$ from $EX_{(3)}(X_1, ..., X_m)$ where $\exists r.(Q_2 \sqcap P_1^{j_1} \sqcap ... \sqcap P_n^{j_n}) = lcs\{X_{k_0}, A_1^{j_1}, ..., A_n^{j_n}\}$ and $k_0 = I(\bar{j}_1, ..., \bar{j}_n)$. First, show that $e' \not\sqsubseteq e''$ for all $e' \in E(Y_1, ..., Y_m)$.

- $e' \not\sqsubseteq e''$ for all $e' \in EX_{(1)}(Y_1, ..., Y_m)$ since $\{Q_2, P_1^{j_1}, ..., P_n^{j_n}\} \nsubseteq \{Q_1, Q_2\} \cup \bigcup_{l=1, l \neq h}^{n}\{P_l^1, P_l^2\}$ for $h \in \{1..n\}$.
- $e' \not\sqsubseteq e''$ for all $e' \in EX_{(2)}(Y_1, ..., Y_m)$ since $\{Q_2, P_1^{j_1}, ..., P_n^{j_n}\} \nsubseteq \bigcup_{l=1}^{n}\{P_l^1, P_l^2\}$ and $\{Q_2, P_1^{j_1}, ..., P_n^{j_n}\} \nsubseteq \{Q_1\} \cup \bigcup_{l=1}^{n}\{P_l^1, P_l^2\}$. Note that $(Y_1, ..., Y_m) \neq (B_2, ..., B_2)$ since $Y_{k_0} = B_1$.
- $e' \not\sqsubseteq e''$ for all $e' \in EX_{(3)}(Y_1, ..., Y_m)$, $e' \notin EX_{(1)}(Y_1, ..., Y_m)$ and $e' \notin EX_{(2)}(Y_1, ..., Y_m)$.
  Indeed, e' has to be of the form $lcs\{B_{i_h}, A_{s_1}^{i_{s_1}}$,..., $A_{s_p}^{i_{s_p}}\}$ such that $\{A_{s_1}^{i_{s_1}}$,..., $A_{s_p}^{i_{s_p}}\} \subseteq \{A_1^{l_1}, ..., A_n^{l_n}\}$ for some $(l_1, ..., l_n) \in \{1,2\} \times ... \times \{1,2\}$ and $B_{i_h} = Y_h$ where $h = I(\bar{l}_1, ..., \bar{l}_n)$. This means that there exists $(E_{i_1}, ..., E_{i_m}) \in \mathcal{S}(Y_1, ..., Y_m)$ such that $(E_{i_1}, ..., E_{i_m}) = \{Y_h, A_{s_1}^{i_{s_1}}, ..., A_{s_p}^{i_{s_p}}\}$ where $\{A_{s_1}^{i_{s_1}}$,..., $A_{s_p}^{i_{s_p}}\} \subseteq \{A_1^{l_1}, ..., A_n^{l_n}\}$ for some $(l_1, ..., l_n) \in \{1,2\} \times ... \times \{1,2\}$ and $E_{i_h}{=}Y_h{=}\{Y_h\} \cup Val'(C_h)$ for $h{=}I(\bar{l}_1, ..., \bar{l}_n)$.
  Assume that $h = k_0$. We have $E_{i_h} = E_{i_{k_0}} {=} Y_{k_0} \neq X_{k_0}$ where $Y_{k_0} {=}\ B_1 {=}\ \exists r.(Q_1 \sqcap \bigsqcap_{l=1}^{n}(P_l^1 \sqcap P_l^2))$, $X_{k_0} {=}\ B_2 {=}\ \exists r.(Q_2 \sqcap \bigsqcap_{l=1}^{n}(P_l^1 \sqcap P_l^2))$. Hence, e'' contains Q_2 but e' does not contains Q_2. Thus, $e' \not\sqsubseteq e''$.
  Assume that $h \neq k_0$. If $E_{i_{k_0}}{=}Y_{k_0}{=}B_1$ then, according to the argument above, $e' \not\sqsubseteq e''$. Otherwise, $E_{i_{k_0}}{=}A_r^{\bar{j}_r} \in \{A_1^{\bar{j}_1}, ..., A_n^{\bar{j}_n}\}$ and $A_r^{\bar{j}_r} \in \{A_{s_1}^{i_{s_1}}$,..., $A_{s_p}^{i_{s_p}}\}$ where $k_0{=}I(\bar{j}_1, ..., \bar{j}_n)$. Since $e''{=}lcs\{X_{k_0}, A_1^{j_1}, ..., A_n^{j_n}\}$ hence e'' contains $P_r^{j_r}$ for $j_r \in \{j_1, ..., j_n\}$. On the other hand, since $e' = lcs\{Y_h, A_{s_1}^{i_{s_1}}, ..., A_{s_p}^{i_{s_p}}\}$ and $A_r^{\bar{j}_r} \in \{A_{s_1}^{i_{s_1}}$,..., $A_{s_p}^{i_{s_p}}\}$ hence e' contains $P_r^{\bar{j}_r}$ but not $P_r^{j_r}$. Thus, $e' \not\sqsubseteq e''$.

Similarly, we can show that there exists $e' \in E(Y_1, ..., Y_m)$ such that $e'' \not\sqsubseteq e'$ for all $e'' \in$

$E(X_1, ..., X_m)$. To do it, pick $e' = \exists r.(Q_1 \sqcap P_1^{\bar{j}_1} \sqcap ... \sqcap P_n^{\bar{j}_n})$ from $E_{(3)}(Y_1, ..., Y_m)$ where $\exists r.(Q_1 \sqcap P_1^{j_1} \sqcap ... \sqcap P_n^{j_n}) = lcs\{Y_k, A_1^{j_1}, ..., A_n^{j_n}\}$ and $k = I(\bar{j}_1, ..., \bar{j}_n)$. We proceed in the same way as above. It remains to be proven that there does not exist any $\mathcal{ALE}$-concept description D such that $D \equiv approx_{ALE}(C)$ and the number of existential restrictions in D (as conjuncts on top-level) is smaller than 2^{2^n}. Assume that there exists such a concept description D. Since $C \sqsubseteq D$, the height of the description tree $\mathcal{G}(D)$ is not greater than 2. Furthermore, there exist existential restrictions $\exists r.C_1, \exists r.C_2$ where $C_1, C_2 \in Ex(approx_{ALE}(C))$ and an existential restriction $\exists r.D_1$ where $D_1 \in Ex(D)$ such that $D_1 \equiv C_1$, $D_1 \equiv C_2$. This implies that $C_1 \sqsubseteq C_2$, which contradicts the property of $approx_{ALE}(C)$ shown above. ■

Remark 7 *Algorithm 3 yields immediately* a normalized *$\mathcal{ALE}$-concept description the number of existential restrictions on top-level of which may be double exponential. The fact that the algorithm may generate non-collapsible 2^{2^n} existential restrictions from 2^n disjuncts on top-level of the $\mathcal{ALC}$-form normal of C (as constructed in the proof) cannot be explained by the interaction between value and existential restrictions. By what it means that the compact representation introduced in Section 3 cannot help to reduce the size of the approximation obtained. Hence, the question raised is whether this exponential blow-up is specific to the approximation computation.*

The remainder of this section will show that the exponential blow-up caused by the approximation computation results from the computing of the lcs of n $\mathcal{ALE}$-concept descriptions.
First, we need the following proposition for this purpose.

Proposition 4 *Let $C = C_1 \sqcup ... \sqcup C_n$ be an $\mathcal{ALC}$-concept description where $\bot \sqsubset C_1, ..., C_n$. The approximation of C by $\mathcal{ALE}$-concept description can be computed as follows:*

$$approx_{\mathcal{ALE}}(C) \equiv \mathrm{lcs}\{approx_{\mathcal{ALE}}(C_1), ... , approx_{\mathcal{ALE}}(C_n)\}$$

A proof of Proposition 4 can be found in Appendix.

Algorithm 4 is a direct consequence of Algorithm 3 and Proposition 4.

Algorithm 4 $approx_{\mathcal{ALE}}(C)$

Require: C is an $\mathcal{ALC}$-concept description in $\mathcal{ALC}$-normal form $C = C_1 \sqcup ... \sqcup C_n$.
Ensure: $approx_{\mathcal{ALE}}(C)$
  if $C \equiv \bot$ **then**
    return $\bot$;
  end if
  if $C \equiv \top$ **then**
    return $\top$;
  end if
  if $n = 1$ **then**
    return $\bigsqcap_{A \in prim(C_1)} A \sqcap \bigsqcap_{C' \in ex(C_1)} \exists r.approx_{\mathcal{ALE}}(C' \sqcap val(C_1)) \sqcap \forall r.approx_{\mathcal{ALE}}(val(C_1))$;
  else
    return $\mathrm{lcs}\{approx_{\mathcal{ALE}}(C_1), ..., approx_{\mathcal{ALE}}(C_n)\}$
  end if

Algorithms 3, 4 provide two methods to compute the approximation. This allows us to conclude that a double exponential number of existential restrictions occurring on the top-level of the approximation obtained from Algorithm 3 is due to the computing of the lcs of an exponential number of concept description.
More concretely, Algorithms 3 and 4 establish the following equivalence for $approx_{\mathcal{ALE}}(C)$ (C is constructed in the proof of Theorem 3)

$$\bigsqcap_{(B_{i_1}, ..., B_{i_m}) \in (\{B_1, B_2\} \times ... \times \{B_1, B_2\})} \{\exists r.lcs\{ (B_{i_j} \sqcap Val(C_j)) | 1 \leq j \leq m\}\} \sqcap \forall r.lcs\{Val(C_j) | 1 \leq j \leq m\}$$

$$\equiv$$

$$lcs\{\exists r.B_1 \sqcap \exists r.B_2 \sqcap \forall r.Val(C_1), ... , \exists r.B_1 \sqcap \exists r.B_2 \sqcap \forall r.Val(C_m)\}$$

Note that the left side can be obtained from the right side by computing directly the lcs.
The computing of lcs in the right side requires a normalization. If the rules in Definition 2 are used for the normalization, the size of the normalized concept descriptions increases polynomially. Thus, the exponential blow-up of the computing of lcs in this case is not due to the interaction between value and existential restrictions. This explains why the compact representation pre-

sented in Section 3 does not help to avoid the exponential blow-up. This result is compatible with the result shown in [3], which states that an exponential blow-up may occur for the computing of $lcs\{C_1,...,C_n\}$ where C_i are $\mathcal{EL}$-concept descriptions, *i.e.,* no normalization is necessary.

6. Conclusion and future work

We have presented a specific data structure, called *graph normalization*, for representing $\mathcal{ALE}$-concept descriptions. This data structure can represent the *lcs* of two $\mathcal{ALE}$-concept descriptions in a polynomial space. We have proposed a algorithm polynomial in space and exponential in time for deciding subsumption between $\mathcal{ALE}$-concept descriptions including *lcs*. This result allows us to add to a reasoner a procedure for treating the *lcs* without increasing the complexity of subsumption inference in time and space.
This paper has shown that the size of the approximation $\mathcal{ALC}$-$\mathcal{ALE}$ in the compact representation, and thus, in the ordinary representation may be double exponential. This result together with double exponential complexity of Algorithm 3, as shown in [1], allows us to conclude that lower and upper bounds for the size of the approximation $\mathcal{ALC}$-$\mathcal{ALE}$ in the compact representation, and thus, in the ordinary representation are double exponential. This gives a *partial* answer to the question left open by the authors in [1]. What we can affirm from the results of the present paper is that there does not exist any exponential algorithm for computing the approximation $\mathcal{ALC}$-$\mathcal{ALE}$ in *the ordinary representation*. This affirmation does not mean that there does not exist any exponential algorithm for computing the approximation $\mathcal{ALC}$-$\mathcal{ALE}$. We may obtain a positive answer to this question if there exists a special representation for $\mathcal{ALE}$-concept descriptions, which enables to express the approximation $\mathcal{ALC}$-$\mathcal{ALE}$ in an exponential space.
Our method is based heavily on the characterization of subsumption by homomorphism between description trees presented in [3]. This characterization that is extended to normalization graphs helps to avoid exponential blow-up of the size of the binary *lcs*, but does not allow to avoid double exponential size of the approximations. The computation performed in Section 5 shows that the double exponential blow-up in the approximation algorithm comes from the following two sources: i) distribution of conjunctions over disjunctions (shown in [8]), ii) exponential size of $lcs\{C_1,...,C_n\}$. This explains why normalization graphs which compact only the normalization are not sufficient for avoiding the double exponential blow-up in the approximation algorithm. However, as illustrated in Figure 6, a very compact normalization graph (polynomial) may replace the complete binary tree for representing an $\mathcal{ALE}$-concept description. This could provide an idea for finding a reduction limit of representations for $\mathcal{ALE}$-concept descriptions.
The work in [11] has given a double exponential algorithm for computing the *lcs* in the logic $\mathcal{ALEN}$. This yields a double exponential upper bound for the size of the *lcs* of two $\mathcal{ALEN}$-concept descriptions. Hence, a natural question raised is whether normalization graphs preserve their properties in more expressive Description Logics, for example, $\mathcal{ALEN}$. This question deserves to be studied in future work.

Acknowledgements

The authors wish to thank the anonymous reviewers of an earlier version of this paper for comments and suggestions that have helped us to significantly improve the quality as well as the readability of this paper.

References

[1] S. Brandt, R. Küsters and A.-Y Turhan. Approximation and Difference in Description Logics. In D. Fensel, D. McGuiness, and M.-A Williams, eds. *Proc. of KR-2002,* pages 203-214, San Francisco, CA, 2002. Morgan Kaufmann.

[2] S.Brandt, R. Küsters and A.-Y Turhan. Approximating $\mathcal{ALCN}$-concept description*s. Proc. of DL2002.*

[3] F. Baader, R. Küsters and R. Molitor. Computing Least Common Subsumer in Description Logics with Existential Restrictions. In T. Deen, ed. *Proc. of IJ-CAI-99.* pages 96-101, Stockholm, Sweden, 1999, Morgan Kaufmann.

[4] F. Baader and A.-Y Turhan. On the problem of Computing Small Representations of Least Common Subsumers. In *Proc. of KI2002.* Lecture Notes in Artificial Intelligence. Aachen, Germany, 2002.

[5] F. Baader, R. Molitor and R. Tobies. Tractable and Decidable Fragments of Conceptual Graphs. In W. Cyre and W. Tepfenhart, eds. *Proc. of ICCS99.* Lecture Notes in Computer Science. Springer-Verlag, 1999.

[6] F. M. Donini, B. Hollunder, M. Lenzerini, A. M. Spaccamela, D. Nardi and W. Nutt. The complexity of existential quantification in concept languages. *Artificial Intelligence* (2), 309-327, 1992.

[7] M. Schmidt-Schauß and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence (48), 1-26, 1991.*

[8] B. Nebel. Terminological reasoning is inherently intractable. *Artificial Intelligence* (43), 235-249, 1990.

[9] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. F. Patel-Schneider. The Description Logic Handbook: Theory, Implementation and Applications. *Cambridge Press 2003.*

[10] I. Horrocks, P.F. Patel-Schneider and F. van Harmelen. From $\mathcal{SHIQ}$ and RDF to OWL: The making of a web ontology language. *Journal of Web Semantics.* 1(1):7-26, 2003.

[11] R. Küsters and R. Molitor. Computing Least Common Subsumers in $\mathcal{ALEN}$. In B. Nebel, ed., *Proc. of IJCAI-2001.* pages 219-224, 2001, Morgan Kaufmann.

[12] S. Brandt, R. Küsters and A.Y. Turhan. Extensions of Non-standard inferences to Description Logics with Transitive Roles. In M. Vardi and A. Voronkov, eds, *Proc. of LPAR03*, Lecture Notes in Artificial Intelligence, 2003, Springer-Verlag.

Appendix

Proofs of Propositions and Theorems

Lemma 1 *Let $\mathcal{G}^\epsilon(C) = (V, E \cup E^\epsilon, l)$ be an ϵ-tree. Let $c^k = [v^k_1, ..., v^k_q]$ be a* clash *in $\mathcal{G}^\epsilon(C)$.*

1. *There exists a root $v^l \in V$ of c^k such that v^l=$p^{k-l}(v^k_i)$ for all $1 \le i \le q$. Furthermore, for each level $l \le j \le k$, there exists at most a r-successor v^j such that $v^j = p^{k-j}(v^k_i)$ for some $v^k_i \in \{v^k_1, ..., v^k_q\}$.*
2. *There exist exactly $(q-1)$ pairs of nodes (v^{l_1}, u^{l_1}),..., $(v^{l_{q-1}}, u^{l_{q-1}})$ such that v^{l_1},..., $v^{l_q} \in V$ are r-successors; u^{l_1}, ..., $u^{l_{q-1}} \in V$ are $\forall r$-successors; $p(v^{l_i}) = p(u^{l_i})$ for all $i \in \{1, ..., q-1\}$ and v^{l_1}=$p^{k-l_1}(v^k_{i_1})$, ... , $v^{l_{q-1}}$=$p^{k-l_{q-1}}(v^k_{i_{q-1}})$ for some $(q-1)$ nodes $v^k_{i_1}$, ..., $v^k_{i_{q-1}} \in \{v^k_1, ..., v^k_q\}$.*

Proof: According to Definition 7 (clash), we have $c^k \subseteq n^k$ for some $\forall r$-neighbourhood n^k.

1. Since $\mathcal{G}^\epsilon(C)$ without ϵ-edges is a tree, there exists a node $v^l \in V$ such that v^l=$p^{k-l}(v^k_i)$ for all $1 \le i \le q$. Assume that there exist two r-successors v^m_1, v^m_2 where $l \le m \le k$ such that $v^m_1 = p^{k-m}(v^k_1)$, $v^m_2 = p^{k-m}(v^k_2)$ for some v^k_1, $v^k_2 \in \{v^k_1, ..., v^k_q\}$. Since n^k is a neighbourhood and $v^m_1 = p^{k-m}(v^k_1)$, $v^m_2 = p^{k-m}(v^k_2)$, according to Definition 6 (neighbourhood), there exists a neighbourhood n^m such that v^m_1, $v^m_2 \in n^m$. This contradicts Definition 6.

2. Since $\mathcal{G}^\epsilon(C)$ without ϵ-edge is a tree, for two nodes $v^k_{i_1}$, $v^k_{i_2}$ $\in\{v^k_1, ..., v^k_q\}$, there exists a node v^{c_1} such that v^{c_1}=$p^{k-c_1}(v^k_{i_1})$ =$p^{k-c_1}(v^k_{i_2})$ and $p^{k-h}(v^k_{i_1}) \neq p^{k-h}(v^k_{i_2})$ for all $h \in \{k, ..., c_1+1\}$. Since each node in $\mathcal{G}^\epsilon(C)$ has at most one $\forall r$-successor, there exist a r-successor v^{l_1} and a $\forall r$-successor u^{l_1} such that $p(v^{l_1}) = p(u^{l_1}) = v^{c_1}$ and v^{l_1}, $u^{l_1} \in \{p^{k-l_1}(v^k_{i_1}), p^{k-l_1}(v^k_{i_2})\}$. We show that there exist $(q-1)$ r-successors v^{l_1}, ..., $v^{l_{q-1}}$ and $(q-1)$ $\forall r$-successors u^{l_1}, ..., $u^{l_{q-1}}$ that have the property described above.

By induction, assume that we have built r-successors v^{l_1}, ..., $v^{l_{s-1}}$ and $(s-1)$ $\forall r$-successors u^{l_1}, ..., $u^{l_{s-1}}$ $(s < q)$ from nodes $v^k_{i_1}$,..., $v^k_{i_s}$ $\in\{v^k_1, ..., v^k_q\}$ such that $p(v^{l_1}) = p(u^{l_1})$=v^{c_1}, ..., $p(v^{l_{s-1}}) = p(u^{l_{s-1}})$=$v^{c_{s-1}}$ and $v^{l_1} = p^{k-l_1}(v^k_{i_1})$, ..., $v^{l_{s-1}} = p^{k-l_{s-1}}(v^k_{i_{s-1}})$ (*) where u^{l_i} are $\forall r$-successors. Assume that $l_1 < ... < l_{s-1}$. Let $v^k_{i_{s+1}} \in \{v^k_1, ..., v^k_q\} \setminus \{v^k_{i_1}, ..., v^k_{i_s}\}$.

Assume that $p^{k-c_j}(v^k_{i_{s+1}})$=v^{c_j} for some $c_j \in \{c_1, ..., c_{s-1}\}$. There exists a node v^{c_s} such that $p^{k-c_s}(v^k_{i_{s+1}})$=$p^{k-c_s}(v^k_{i_h})$=v^{c_s} where $v^k_{i_h} \in \{v^k_{i_1}, ..., v^k_{i_s}\}$ and $p^{k-m}(v^k_{i_{s+1}}) \neq p^{k-m}(v^k_{i_1})$,..., $p^{k-m}(v^k_{i_{s+1}}) \neq p^{k-m}(v^k_{i_s})$ for all $m \in \{k, ..., c_s+1\}$. This implies that $c_s > c_j$ since $p^{k-c_j}(v^k_{i_{s+1}})$=v^{c_j}=$p(v^{l_j})$= $p^{k-l_j+1}(v^k_{i_j})$. Let v^{l_s} and u^{l_s} be a r-successor and the $\forall r$-successor, respectively, such that $p(v^{l_s})$= $p(u^{l_s})$= v^{c_s} and v^{l_s}=$p^{k-l_s}(v^k_{i_x})$, u^{l_s}=$p^{k-l_s}(v^k_{i_y})$ where $v^k_{i_x}$=$v^k_{i_{s+1}}$ and $v^k_{i_y} \in \{v^k_{i_1}, ..., v^k_{i_s}\}$, or $v^k_{i_y}$=$v^k_{i_{s+1}}$ and $v^k_{i_x} \in \{v^k_{i_1}, ..., v^k_{i_s}\}$ (**).

If $v^{c_s} = v^{c_j}$ for some $v^{c_j} \in \{v^{c_1}, ..., v^{c_{s-1}}\}$ then $l_s = l_j$ and there exists a neighbourhood n^{l_s} such that v^{l_s}, $v^{l_j} \in n^{l_s}$ (since v^{l_s}, v^{l_j} are ancestors of $v^k_{i_1}, ..., v^k_{i_s}$). From 1. of the lemma, we obtain that $v^{l_s} = v^{l_j}$, and thus $v^k_{i_x} \in \{v^k_{i_1}, ..., v^k_{i_s}\}$. Since each node of $\mathcal{G}^\epsilon(C)$ has at most one $\forall r$-successor, we have $v^k_{i_y} \in \{v^k_{i_1}, ..., v^k_{i_s}\}$ as well. This contradicts (**). Therefore, we have $v^{c_s} \neq v^{c_j}$ for all $v^{c_j} \in \{v^{c_1}, ..., v^{c_{s-1}}\}$, and thus $v^{l_s} \neq v^{l_j}$ for all $v^{l_j} \in \{v^{l_1}, ..., v^{l_{s-1}}\}$. Furthermore, i) if $v^k_{i_x}$=$v^k_{i_{s+1}}$ where v^{l_s}=$p^{k-l_s}(v^k_{i_x})$ then we have $v^{l_1} = p^{k-l_1}(v^k_{i_1})$, ...,

$v^{l_{s-1}} = p^{k-l_{s-1}}(v^k_{i_{s-1}})$, $v^{l_s} = p^{k-l_s}(v^k_{i_{s+1}})$. ii) If $v^k_{i_x} = v^k_{i_s}$ where $v^{l_s}{=}p^{k-l_s}(v^k_{i_x})$ then we have $v^{l_1} = p^{k-l_1}(v^k_{i_1})$, ..., $v^{l_{s-1}} = p^{k-l_{s-1}}(v^k_{i_{s-1}})$, $v^{l_s} = p^{k-l_s}(v^k_{i_s})$. iii) Otherwise, *i.e.*, $v^k_{i_x} \in \{v^k_{i_1}, ..., v^k_{i_s}\}$ and $v^k_{i_x} \neq v^k_{i_s}$ $v^{l_s}{=}p^{k-l_s}(v^k_{i_x})$ then it is that $v^{l_x} = p^{k-l_x}(v^k_{i_x})$ (*). Thus, now $v^{l_x} = p^{k-l_x}(v^k_{i_{s+1}})$ and $v^{l_s} = p^{k-l_s}(v^k_{i_x})$.
Assume that $p^{k-c_j}(v^k_{i_{s+1}}){\neq}v^{c_j}$ for all $c_j \in \{c_1, ..., c_{s-1}\}$. From $l_1 < ... < l_{s-1}$ we have $c_1 < ... < c_{s-1}$. There exists a node v^{c_s} such that $c_s < c_1$ and $p^{k-c_s}(v^k_{i_{s+1}}){=}p^{c-c_s}(v^{c_1}){=}v^{c_s}$ where $p^{k-c_1+m}(v^k_{i_{s+1}}) \neq p^m(v^{c_1})$ for all $m \in \{1, ..., c_s{+}1\}$. Let v^{l_s}, u^{l_s} be a r-successor and a $\forall r$-successor such that $p(v^{l_s}){=}p(v^{l_s}){=}v^{c_s}$. It is obvious that $v^{l_s} \notin \{v^{l_1}, ..., v^{l_{s-1}}\}$. Furthermore, i) if $v^k_{i_h}{=}v^k_{i_{s+1}}$ where $v^{l_s}{=}p^{k-l_s}(v^k_{i_h})$ we have $v^{l_1} = p^{k-l_1}(v^k_{i_1})$, ..., $v^{l_{s-1}} = p^{k-l_{s-1}}(v^k_{i_{s-1}})$, $v^{l_s} = p^{k-l_s}(v^k_{i_{s+1}})$ ii) we have to have that $v^{l_s}{=}p^{k-l_s}(v^k_{i_s})$ and thus $v^{l_1} = p^{k-l_1}(v^k_{i_1})$, ..., $v^{l_{s-1}} = p^{k-l_{s-1}}(v^k_{i_{s-1}})$, $v^{l_s} = p^{k-l_s}(v^k_{i_s})$.
We have shown that there exist $(q-1)$ pairs (v^{l_1}, u^{l_1}), ..., $(v^{l_{q-1}}, u^{l_{q-1}})$ where v^{l_i} are r-successors, u^{l_i} are $\forall r$-successors, $p(v^{l_i}) = p(u^{l_i})$ for all $i \in \{1, ..., q-1\}$ and $v^{l_1}{=}p^{k-l_1}(v^k_{i_1})$, ... , $v^{l_{q-1}}{=} p^{k-l_{q-1}}(v^k_{i_{q-1}})$ for some $(q-1)$ nodes $v^k_{i_1}$, ..., $v^k_{i_{q-1}} \in \{v^k_1, ..., v^k_q\}$. ■

Lemma 2 *Let $\mathcal{G}^\epsilon(C) = (V, E \cup E^\epsilon, l)$ be an ϵ-tree where $v^0 \in V$ is its root and $\mathcal{G}^\epsilon_C{=}(V', E'{\cup}E'^\epsilon, l')$ be its normalization graph. Let n'^k be a r-neighbourhood ($\forall r$-neighbourhood) in $\mathcal{G}^\epsilon_C$. If $k = 0$ then* label$(n'^0) \neq \{\bot\}$ *iff n'^0 does not contain any* clash. *If $k > 0$ then the following claims are equivalent* :

1. label$(n'^k) \neq \{\bot\}$.
2. *There exists a neighbourhood n^k in $\mathcal{G}^\epsilon(C)$ such that $n^k = n'^k \cap V$ and* label(n^k) = label(n'^k).
3. *There does not exist any* q-clash $[v^k_1, .., v^k_q]$ *such that $\{v^k_1, .., v^k_q\} \subseteq n^k$, $n^k = V^\forall(n'^{k-1}) \cap V$ and $n'^k \in N(n'^{k-1})$ where n^k is a $\forall r$-neighbourhood in $\mathcal{G}^\epsilon(C)$.*

Proof: We show 1. ⇔ 2.
If $k = 0$ then, according to Definition 8 (normalization graph), $n'^0 = n^0 = \{v^0\}$. We have label$(n'^0) \neq \{\bot\}$ iff $l'(v^0) \neq \{\bot\}$. This implies that $l(v^0) \neq \{\bot\}$ and $l(v^0)$ is not modified by the normalization (normalization graph) and thus label(n^0) = label(n'^0).

Assume $k > 0$.
I) "1. ⇒ 2.". Assume that label$(n'^{k+1}) \neq \{\bot\}$. This means that label$(n'^k) \neq \{\bot\}$ where $n'^{k+1} \in N(n'^k)$. The induction hypothesis asserts that there exists a neighbourhood n^k in $\mathcal{G}^\epsilon(C)$ such that $n^k = n'^k \cap V$ and $n'^{k+1} \in N(n'^k)$.
If n'^{k+1} does not contain any r-successor then, by Definition 9 (extended-neighbourhood), it is that $n'^{k+1} = V^\forall(n'^k)$ since label$(n'^{k+1}) \neq \{\bot\}$, and thus $V^\epsilon(n'^k) = \emptyset$. Therefore, there exists a neighbourhood $n^{k+1} = V^\forall(n'^k) \cap V = n'^{k+1} \cap V$ where $n^{k+1} \in N(n^k)$ and $n'^{k+1} \in N(n'^k)$. This means that n'^{k+1} is formed from n^{k+1} and some $\forall r$-successors v'^{k+1} that are added by the normalization (Definition 8). From this and the construction of normalization graphs, it is that $l'(v'^{k+1}){=}\emptyset$ for all $v'^{k+1} \in n'^{k+1} \setminus n^{k+1}$. This implies that label(n^{k+1}) = label(n'^{k+1}).
If n'^{k+1} contains a r-successor v^{k+1} then there exists a neighbourhood $m^{k+1} := n^{k+1} \cup \{v^{k+1}\}$ such that $m^{k+1} = n'^{k+1} \cap V$ and $n^{k+1} = V^\forall(n'^k) \cap V$ where $n'^{k+1} \in N(n'^k)$. We have $l(v^{k+1})$ does not contain any bottom-concept $\bot$ since $\mathcal{G}^\epsilon(C)$ is built from C that is in weak normal form. This means that n'^{k+1} is formed from m^{k+1} and some $\forall r$-successors v'^{k+1} that are added by the normalization (Definition 8) where $l'(v'^{k+1}) \neq \{\bot\}$. This implies that label(m^{k+1}) = label(n'^{k+1}).
II) "2. ⇒ 1.". Assume that there exists a r-neighbourhood ($\forall r$-neighbourhood) n^{k+1} in $\mathcal{G}^\epsilon(C)$ such that $n^{k+1} = n'^{k+1} \cap V$ and label(n^{k+1}) = label(n'^{k+1}). By absurdity, assume that label(n'^{k+1}) = $\{\bot\}$. The definition of function label yields that $l(v'^{k+1}) = \{\bot\}$ for some $v'^{k+1} \in n'^{k+1}$. Since $\mathcal{G}^\epsilon(C)$ is built from C in weak normal form, it is that v'^{k+1} is not a r-successor. Moreover, by the normalization (Definition 8) and the simplification of ϵ-trees for 1-clashes, we have $v'^{k+1} \notin n^{k+1}$ and v'^{k+1} is not a $\forall r$-successor. Thus, according to Definition 9 (extended-neighbourhood), $v'^{k+1} \in V^\epsilon(n'^k) = n'^{k+1}$. By consequent, $n^{k+1} \nsubseteq n'^{k+1}$ which is a contradiction.

We show 1. ⇔ 3. by absurdity.
If $k = 0$ then, according to Definition 8 (normalization graph), $n'^0 = n^0 = \{v^0\}$. We have label(n'^0) = $\{\bot\}$ iff $l'(v^0) = \{\bot\}$ or $l(v^0)$ is modified by the normalization (simplification for 1-clash). This implies that label$(n'^0) = \{\bot\}$ iff there exists a 1-clash $[v^0]$. Assume $|\mathcal{G}^\epsilon(C)| > k > 0$.
I) "3. ⇒ 1.". Assume label$(n'^{k+1}) = \{\bot\}$. The definition of function label yields that $l'(u^{k+1}) =$

$\{\bot\}$ for some $u^{k+1} \in n'^{k+1}$. Moreover, we have $\mathsf{label}(n'^k) \neq \{\bot\}$ where $n'^{k+1} \in N(n'^k)$. According to items 1. and 2. of this lemma, there exists a neighbourhood $n^k{=}n'^k \cap V$.

Assume that n'^{k+1} is a r-neighbourhood of n'^k. Let v_i^{k+1} be the r-successor such that $v_i^{k+1} \in n'^{k+1}$. Let n^{k+1} be the r-neighbourhood of n^k in $\mathcal{G}^\epsilon(C)$ such that $v_i^{k+1} \in n^{k+1}$. If there exists $x^{k+1} \in n^{k+1}$ such that $l(x^{k+1}) = \{\bot\}$ then there is a 1-clash $[x^{k+1}]$. From the simplification for 1-clash $[x^{k+1}]$ we have $l'(x^{k+1}) = \emptyset$. Furthermore, if $u^{k+1} \notin n^{k+1}$ then $u^{k+1} \notin n'^{k+1}$ since, according to the construction of the normalization graph, there does not exist any ϵ-edge from v_i^{k+1} to u^{k+1} such that $l'(u^{k+1}) = \{\bot\}$. Thus, $\mathsf{label}(n'^{k+1}) \neq \{\bot\}$, which is a contradiction.

Assume that n'^{k+1} is the $\forall r$-neighbourhood of n'^k. Since u^{k+1} is not a $\forall r$-successor and u^{k+1} is added by the normalization (normalization graph), we have $n'^{k+1} = V^\epsilon(n'^k)$. Let n^{k+1} be the $\forall r$-neighbourhood of n^k in $\mathcal{G}^\epsilon(C)$. We consider the following cases:

i) Assume that u^{k+1} comes from 1-clash $[v^{k+1}]$, *i.e.*, $(v^k \forall r v^{k+1}) \in E$, $(v^{k+1} \epsilon u^{k+1}) \in E'^\epsilon$ and $\mathcal{P}(u^{k+1}) = \{v^k\}$. This implies that $v^{k+1} \in n^{k+1}$. Since $n^k = n'^k \cap V$ hence $n^{k+1} = V^\forall(n'^k) \cap V$.

ii) By the construction of the normalization graph (Definition 8), there exists a node $w_{l_1}^{k+1} \in V'$ such that $(w_{l_1}^{k+1} \epsilon u^{k+1}) \in E'^\epsilon$, $\mathcal{P}(w_{l_1}^{k+1}) = \mathcal{P}(u^{k+1})$ and $w_{l_1}^{k+1} \in V^\forall(n'^k)$. Moreover, there exist $w_{l_1}^{l_1} \in V'$ and a r-successor $v^{l_1} \in V$ where $l_1 < k+1$ such that $(v^{l_1} \epsilon w_{l_1}^{l_1}) \in E'^\epsilon$, $\mathcal{P}^{k-l_1+1}(u^{k+1}) = \mathcal{P}^{k-l_1+1}(w_{l_1}^{k+1}) = \{w_{l_1}^{l_1}\}$ and $\mathcal{P}^{k-l_1+1}(v_{l_1}^{k+1}){=}\{v^{l_1}\}$ for some $v_{l_1}^{k+1} \in n^{k+1}$. Moreover, there is a neighbourhood n'^{l_1} such that $N^{-(k+1-l_1)}(n'^{k+1}){=}n'^{l_1}$ and $v^{l_1}, w_{l_1}^{l_1} \in n'^{l_1}$. There exists a clash c_0^{k+1} such that $v^{l_1} \in V(c_0^{k+1})$ and v^{l_1} is the tail of c_0^{k+1}. According to Definition 9 (extended-neighbourhood), there exists $w_{l_2}^{l_1-1} \in \mathcal{P}(w_{l_1}^{l_1})$ such that $w_{l_2}^{l_1-1} \in n'^{l_1-1}$ and $N^{-1}(n'^{l_1}) = n'^{l_1-1}$. If $w_{l_2}^{l_1-1}$ is not a $\forall r$-successor (nor r-successor) there exists uniquely a r-successor v^{l_2} where $l_2 = l_1 - 1$ is the highest level such that $v^{l_2} \in V(c_0^{k+1})$, $(v^{l_2} \epsilon w_{l_2}^{l_1-1}) \in E'^\epsilon$ and $v^{l_2} \in n'^{l_1-1}$. If $w_{l_2}^{l_1-1}$ is a $\forall r$-successor then there exists uniquely a r-successor v^{l_2} where $l_2 < l_1 - 1$ is the highest level such that $v^{l_2} \in V(c_0^{k+1})$, $(v^{l_2} \epsilon p^{l_1-1-l_2}(w_{l_2}^{l_1-1})) \in E'^\epsilon$, $v^{l_2} \in n'^{l_2}$, $N^{-(l_1-l_2)}(n'^{l_1}) = n'^{l_2}$ and $\mathcal{P}^{k-l_2+1}(v_{l_2}^{k+1}){=}\{v^{l_2}\}$ for some $v_{l_2}^{k+1} \in n^{k+1}$ (note that $\mathcal{P}^{l_1-1+1}(w_{l_2}^{l_1-1})$,... , $\mathcal{P}^{l_1-1+l_2}(w_{l_2}^{l_1-1})$ are singleton).

This process is terminated at $w_{l_p}^{l_p}$ where $p(w_{l_p}^{l_p}) = p(v^{l_p})$, $v^{l_p} \in V$ and we obtain that $V(c_0^{k+1}) = \{v^{l_1}, ..., v^{l_p}\}$ and $\mathcal{P}^{k-l_i+1}(v_{l_i}^{k+1}){=}\{v^{l_i}\}$ where $v_{l_i}^{k+1} \in n^{k+1}$ for all $v^{l_i} \in V(c_0^{k+1})$. According to item 2. of Lemma 1, we can pick nodes $v^{j_1}, ..., v^{j_q}$ from $\{v^{l_1}, ..., v^{l_p}\}$ such that for each v^{j_i}, $j_i \in \{j_1, ..., j_q\}$, there exists a $\forall r$-successor u^{j_i} that satisfies $p(u^{j_i}) = p(v^{j_i})$ and $p^{k-j_i+1}(v_{j_i}^{k+1}){=}v^{j_i}$ for some $v_{j_i}^{k+1} \in n^{k+1}$. By the construction of the normalization graph, we have that $c_0^{k+1} = [v_{j_1}^{k+1}, ..., v_{j_q}^{k+1}, v_{j_{q+1}}^{k+1}]$ where $p^{k-l_p+2}(v_{j_{q+1}}^{k+1}) = p(v^{l_p})$.

To sum up, we have $c_0^{k+1} \subseteq V^\forall(n'^k)$. Since $n^k{=}n'^k \cap V$ hence $n^{k+1}{=}V^\forall(n'^k) \cap V$ where n^{k+1} is the $\forall r$-neighbourhood of n^k.

II) "1. $\Rightarrow$ 3.". Assume that n^{k+1} is the $\forall r$-neighbourhood of n^k in $\mathcal{G}^\epsilon(C)$ and there is a (q+1)-clash $[v_{j_1}^{k+1}, ..., v_{j_{q+1}}^{k+1}]$ such that $\{v_{j_1}^{k+1}, ..., v_{j_{q+1}}^{k+1}\} \subseteq n^{k+1}$ and $n^{k+1} \subseteq V^\forall(n'^k)$ where $n'^{k+1} \in N(n'^k)$. We show that there exists a node u^{k+1} and such that $l'(u^{k+1}) = \{\bot\}$ and $u^{k+1} \in V^\epsilon(n'^k)$.

In fact, if $q = 0$ then there exist $(w_1^k \forall r w_1^{k+1}) \in E'$, $(w_1^{k+1} \epsilon u^{k+1}) \in E'^\epsilon$, $w_1^{k+1} \in V^\forall(n'^k)$ and $u^{k+1} \notin V^\forall(n'^k)$. Thus, $n'^{k+1} = V^\epsilon(n'^k) = \{u^{k+1}\}$. Therefore, $\mathsf{label}(n'^{k+1}) = \{\bot\}$.

Assume that $q > 0$. Let $v^{l_1}, ..., v^{l_p}$ be r-successors built from clash $[v_{j_1}^{k+1}, ..., v_{j_{q+1}}^{k+1}]$ by item 1. of Lemma 1. We have $v^{l_1}{=}p^{k-l_1+1}(v_{l_1}^{k+1})$, ...,$v^{l_q} = p^{k-l_q+1}(v_{l_p}^{k+1})$ for some $v_{l_1}^{k+1}$, ... , $v_{l_p}^{k+1} \in \{v_{j_1}^{k+1}, ..., v_{j_{q+1}}^{k+1}\}$. We have $\{v_{j_1}^{k+1}, ..., v_{j_{q+1}}^{k+1}\}{=}\{v_{l_1}^{k+1}$, ... , $v_{l_p}^{k+1}\}$. Assume that $l_1 > l_2 > ... > l_p$.

Since $\{v_{l_1}^{k+1}, ..., v_{l_p}^{k+1}\} \subseteq V^\forall(n'^k)$ there exists a neighbourhood n'^{l_p} such that $p^{k-l_p+1}(v_{l_1}^{k+1})$,... , $p^{k-l_p+1}(v_{l_p}^{k+1}) \in n'^{l_p}$ and $n'^{l_p}{=}N^{-(k-l_p)}(n'^k)$ (if $v \in V$ then $\mathcal{P}(v)$ is singleton). Note that $v^{l_p} = p^{k-l_p+1}(v_{l_p}^{k+1})$. Since $p(v^{l_p}) = p(w_{l_p}^{l_p})$, $w_{l_p}^{l_p} \in V' \setminus V$ and $(v^{l_p} \epsilon w_{l_p}^{l_p}) \in E'^\epsilon$, we have $w_{l_p}^{l_p} \in n'^{l_p}$.

For each $l_p \leq m \leq k$, from item 1. of Lemma 1, there exists at most a r-successor v^m such that $p^{k-m+1}(v_{l_j}^{k+1}) = v^m$ for some $v_{l_j}^{k+1} \in V^\forall(n'^k)$. This means that there exist neighbourhoods $n'^{l_p+1}{\in}N(n'^{l_p})$, $n'^{l_p+2}{\in}N(n'^{l_p+1})$, ... , $n'^k{\in}N(n'^{k-1})$ such that $p^{k-m+1}(v_{l_1}^{k+1})$, ..., $p^{k-m+1}(v_{l_p}^{k+1}) \in n'^m$. Since $p(v^{l_p})$ and v^{l_1} are the head and tail of clash $[v_{j_1}^{k+1}, ..., v_{j_{q+1}}^{k+1}]$, according

to Definition 8, the clash $[v_{j_1}^{k+1}, ..., v_{j_{q+1}}^{k+1}]$ is treated and coded in the normalization graph $\mathcal{G}_C^{\epsilon}$.
For each r-successor v^m such that $m \in \{l_p, ..., l_1\}$, by the construction of the normalization graph, there exists an ϵ-edge $(v^m \epsilon w^m) \in E'^{\epsilon}$ where $m = l_i$, $w_{l_{i+1}}^{m-1} \in \mathcal{P}(w^m)$ and $\mathcal{P}^{m-l_{i+1}-1}(w_{l_{i+1}}^{m-1}) = \{w_{l_{i+1}}^{l_{i+1}}\}$ (there is a path of $\forall r$-edges from $w_{l_{i+1}}^{l_{i+1}}$ to $w_{l_{i+1}}^{m-1}$). Thus, by Definition 9 (extended-neighbourhood), $v^m, w^m \in n'^m$ if $w_{l_{i+1}}^{m-1} \in n'^{m-1}$. Note that for all $l_{i+1} < m < l_i$ where $l_{i+1}, l_i \in \{l_p, ..., l_1\}$ we have $v^m, w^m \in n'^m$ if $w_{l_{i+1}}^{m-1} \in n'^{m-1}$ since $w^m \in V^{\forall}(n'^{m-1})$.
In consequence, we obtain that $w_{l_1}^k \in n'^k$ where $p^{k-l_1}(w_{l_1}^k) = w_{l_1}^{l_1}$. Thus, $w_{l_1}^{k+1} \in V^{\forall}(n'^k)$ where $(w_{l_1}^k \forall r w_{l_1}^{k+1}) \in E'$. By the construction of the normalization graph, there is an ϵ-edge $(w_{l_1}^{k+1} \epsilon u^{k+1}) \in E'^{\epsilon}$ such that $w_{l_1}^{k+1} \in V^{\forall}(n'^k)$, $u^{k+1} \notin V^{\forall}(n'^k)$ and $l'(u^{k+1})=\{\bot\}$. By the definition of neighbourhood, we obtain $n'^{k+1} = V^{\epsilon}(n'^k) = \{u^{k+1}\}$. Therefore, label$(n'^{k+1}) = \{\bot\}$. ■

Lemma 3 *Let C be an $\mathcal{ALE}$-concept description in the weak normal form. Let $\mathcal{G}^{\epsilon}(C)$ and $\mathcal{G}_C^{\epsilon}$ be the ϵ-tree and normalization graph of C, respectively. There exists an isomorphism between $\mathbf{B}(\mathcal{G}_C^{\epsilon})$ and the description tree $\mathcal{H}$ which is obtained from $\mathbf{B}(\mathcal{G}^{\epsilon}(C))=(V_3, E_3, z^0, l_3)$ by applying exhaustively the following rules (p is the predecessor function of $\mathbf{B}(\mathcal{G}^{\epsilon}(C))$):*

1. $P, \neg P \in l_3(z)$, $P \in N_C$, $z \in V_3 \rightarrow$ $l_3(z) := \{\bot\}$ *(rule 5g)*
2. $(zrz') \in E_3$, $\mathbf{B}(\mathcal{G}^{\epsilon}(C))(z') = \mathcal{G}(\bot) \rightarrow$ $\mathbf{B}(\mathcal{G}^{\epsilon}(C))(z) := \mathcal{G}(\bot)$ *(rule 6g)*
3. $\bot \in l_3(z)$, $z \in V_3 \rightarrow$ $\mathbf{B}(\mathcal{G}^{\epsilon}(C))(z) := \mathcal{G}(\bot)$ *(rule 7g)*

Proof: Let $\mathcal{G}(C) = (V, E, v^0, l)$, its ϵ-tree $\mathcal{G}^{\epsilon}(C) = (V, E \cup E^{\epsilon}, l)$ and normalization graph $\mathcal{G}_C^{\epsilon} = (V', E' \cup E'^{\epsilon}, l')$. Let $\mathbf{B}(\mathcal{G}_C^{\epsilon}) = (V_1, E_1, u^0, l_1)$, $\mathcal{H}=(V_2, E_2, w^0, l_2)$ and $\mathbf{B}(\mathcal{G}^{\epsilon}(C)) = (V_3, E_3, z^0, l_3)$. We will show this lemma by using Lemma 2. First, we prove the following claim:
Lemma For each node $z^k \in V_3$ it holds that $z^k \in V_2$ and $l_2(z^k) = \{\bot\}$ iff

1. there does not exist any path composed of r-edges from z^h to z^l $(l < h)$ in $\mathbf{B}(\mathcal{G}^{\epsilon}(C))$ such that $\bot \in l_3(z^h)$ or $P, \neg P \in l_3(z^h)$, and $p^n(v^k) = z^l$ for some $n > 0$; and
2. $P, \neg P \in l_3(z^k)$ for some $P \in N_C$, or there exists a path composed of r-edges from z^h to z^k $(k < h)$ in $\mathbf{B}(\mathcal{G}^{\epsilon}(C))$ such that $\bot \in l_3(z^h)$ or $P, \neg P \in l_3(z^h)$.

Condition 1. guarantees that $v^k \in V_2$. In fact, if there exist such a path then, from rules (5g), (6g), (7g), we have $l_2(z^h) = \{\bot\}$. Since $p^n(v^k) = z^l$ hence z^k is deleted by rule 6g) or 7g). Conversely, for all nodes $z^l \in V_3$ such that $p^n(v^k) = z^l$ for some $n > 0$, if there does not exist any path composed of r-edges from z^h to z^l $(l < h)$ in $\mathbf{B}(\mathcal{G}^{\epsilon}(C))$ such that $\bot \in l_3(z^h)$ or $P, \neg P \in l_3(z^h)$ then z^k is not deleted from V_3 by rules 6g) or 7g).
In addition, if $P, \neg P \in l_3(z^k)$ for some $P \in N_C$ then $l_2(z^k) = \{\bot\}$. Assume that there exists a path composed of r-edges from z^l to z^k $(k < l)$ in $\mathbf{B}(\mathcal{G}^{\epsilon}(C))$ such that $\bot \in l_3(z^h)$ or $P, \neg P \in l_3(z^h)$, and $p^n(v^k) = z^l$ for some $n > 0$. From rules (5g), (6g), (7g), we have $l_2(z^k) = \{\bot\}$. Conversely, assume that $P, \neg P \notin l_3(z^k)$ for all $P \in N_C$ and $l_2(z^k) = \{\bot\}$. This implies that $l_3(z^k)$ is modified by rules (6g), (7g). Thus, there exists a path composed of r-edges from z^h to z^k $(k < h)$ in $\mathbf{B}(\mathcal{G}^{\epsilon}(C))$ such that $\bot \in l_3(z^h)$ or $P, \neg P \in l_3(z^h)$.
To construct a bijection between V_2 and V_1 , we can construct a bijection ϕ between the sets of neighbourhoods in $\mathcal{G}_C^{\epsilon}$ and $\mathcal{G}^{\epsilon}(C)$. Note that neighbourhoods n^k such that label$(n^k) = \{\bot\}$ correspond to leaves of trees, *i.e.*, $N(n^k) = \emptyset$. We set $\phi(n^0) = m^0$ where n^0, m^0 are 0-neighbourhoods, respectively, in $\mathcal{G}_C^{\epsilon}$ and $\mathcal{G}^{\epsilon}(C)$. It is obvious that label$(n^0) = \{\bot\}$ iff label$(m^0) = \{\bot\}$. In fact, label$(m^0) = \{\bot\}$, by Lemma above, iff $P, \neg P \in$ label(m^0) for some $P \in N_C$, or there exist neighbourhoods $m^1 \in N(m^0)$,...,$m^l \in N(m^{l-1})$ in $\mathcal{G}^{\epsilon}(C)$ and r-edges $(v^0 r v^1)$,...,$(v^{l-1} r v^l) \in E$, $v^1 \in m^1$, $v^2 \in m^2$,..., $v^l \in m^l$ such that $P, \neg P \in l(v_i^l) \cup l(v_j^l)$, $v_i^l, v_j^l \in m^l$, $P \in N_C$ $(v_i^l \neq v_j^l)$ or $l(v_i^l) = \{\bot\}$ $(v_i^l = v_j^l)$. By Definition 7, this implies that there exists a 1-clash $[v^0]$, $v^0 \in n^0$. By Lemma 3, label$(n^0) = \{\bot\}$ iff there exists 1-clash $[v^0]$.
Assume that $\phi(n^k) = m^k$ and label$(n^k) =$ label$(m^k) \neq \{\bot\}$. Let $n^{k+1} \in N(n^k)$ be a r-neighbourhood ($\forall r$-neighbourhood) in $\mathcal{G}_C^{\epsilon}$. Assume that label$(n^{k+1}) \neq \{\bot\}$. By Lemma 3, there exists a r-neighbourhood ($\forall r$-neighbourhood) $m^{k+1} \in N(m^k)$ in $\mathcal{G}^{\epsilon}(C)$ such that $m^{k+1} = n^{k+1} \cap V_3$, and there does not exist any q-clash $[v_1^{k+1}, ..., v_q^{k+1}]$ such that $\{v_1^{k+1}, ..., v_q^{k+1}\} \subseteq m^{k+1}$. Definition 7 yields that there do not exist neighbourhoods

$m^{k+2} \in N(m^{k+1})$,..., $m^{l+1} \in N(m^l)$ $(l > k+1)$ in $\mathcal{G}^\epsilon(C)$ and r-edges $(v^{k+1}rv^{k+2})$, ..., $(v^{l-1}rv^l) \in E$, $v^{k+1} \in m^{k+1}$, ..., $v^l \in m^l$ such that $P, \neg P \in l(v_i^l) \cup l(v_j^l)$, $v_i^l, v_j^l \in m^l$, $P \in N_C$ or $\{\bot\} = l(v_i^l)$ $(v_i^l = v_j^l)$. By Lemma above, we have that the label of the node that corresponds to m^{k+1} is different from $\{\bot\}$, i.e., $\mathsf{label}(m^{k+1}) \neq \{\bot\}$ and thus $\mathsf{label}(n^{k+1}) = \mathsf{label}(m^{k+1})$.

Conversely, from this neighbourhood m^{k+1} we can determine uniquely $n^{k+1} = V^\forall(n^k)$ if m^{k+1} is a $\forall r$-neighbourhood (i.e. $m^{k+1} = n^{k+1} \cap V_3$), or $n^{k+1} = \{v^{k+1}\} \cup \{V^\epsilon_{v^{k+1}}\}$, $m^{k+1} = n^{k+1} \cap V_3$ (notations in the definition of neighbourhood) where $v^{k+1} \in m^{k+1}$ is a r-successor if m^{k+1} is a r-neighbourhood. Thus $\mathsf{label}(n^{k+1}) = \mathsf{label}(m^{k+1}) \neq \{\bot\}$. We can follows the schema : $\mathsf{label}(m^{k+1}) \neq \{\bot\}$ (by Lemma above) => no existence of sequence of neighbourhood => no existence of clash (Lemma 3) => $\mathsf{label}(n^{k+1}) = \mathsf{label}(m^{k+1}) \neq \{\bot\}$. Therefore, we set $\phi(n^{k+1}) = m^{k+1}$.

Assume that $\mathsf{label}(n^{k+1}) = \{\bot\}$. By Lemma 3, it is obvious that n^{k+1} is a $\forall r$-neighbourhood and there exists a q-clash $[v_1^{k+1}, ..., v_q^{k+1}]$ such that $\{v_1^{k+1}, ..., v_q^{k+1}\} \subseteq V^\forall(n^k)$. By Definition 7, there exist neighbourhoods $m^{k+2} \in N(m^{k+1})$,..., $m^{l+1} \in N(m^l)$ $(l > k+1)$ in $\mathcal{G}^\epsilon(C)$, $\{v_1^{k+1}, ..., v_q^{k+1}\} \subseteq m^{k+1}$ and r-edges $(v^{k+1}rv^{k+2})$, ..., $(v^{l-1}rv^l) \in E$, $v^{k+1} \in m^{k+1}$, ..., $v^l \in m^l$ such that $P, \neg P \in l(v_i^l) \cup l(v_j^l)$, $v_i^l, v_j^l \in m^l$, $P \in N_C$ or $\{\bot\} = l(v_i^l)$ $(v_i^l = v_j^l)$. By Lemma above, we have $\mathsf{label}(m^{k+1}) = \{\bot\}$ and m^{k+1} is a $\forall r$-neighbourhood. Conversely, from this neighbourhood m^{k+1} we can uniquely determine $n^{k+1} = V^\epsilon(n^k)$ and thus $\mathsf{label}(n^{k+1}) = \mathsf{label}(m^{k+1}) = \{\bot\}$. We set $\phi(n^{k+1}) = m^{k+1}$.

By consequent, we have constructed an isomorphism ϕ between $\mathcal{H}$ and $\mathbf{B}(\mathcal{G}^\epsilon_C)$. ■

Proposition 1 *Let C be an $\mathcal{ALE}$-concept description. There exists an isomorphism between $\mathbf{B}(\mathcal{G}^\epsilon_C)$ and $\mathcal{G}_C$.*

Proof: Assume that C is in the weak normal form. Let $\mathcal{G}(C) = (V, E, v^0, l)$ and its ϵ-tree $\mathcal{G}^\epsilon(C) = (V, E \cup E^\epsilon, l')$. Let $\mathbf{B}(\mathcal{G}^\epsilon(C)) = (V_1, E_1, u^0, l_1)$. Let $\mathcal{G}(C') = (V_2, E_2, w^0, l_2)$ be the description tree of the concept description C' obtained from C by applying rules 1, 2 in Definition 2. From Lemma 3, we only need to prove that there exists an isomorphism between the tree obtained by applying the rules in Lemma 3 to $\mathbf{B}(\mathcal{G}^\epsilon(C))$, and $\mathcal{G}_C$. First, we show that there exists an isomorphism between $\mathbf{B}(\mathcal{G}^\epsilon(C))$ and $\mathcal{G}(C')$.

We construct by induction on level k $(0 \leq k \leq |\mathcal{G}(C)|)$ a bijection ϕ: $V_2 \to V_1$ such that $l_2(w^0) = l_2(u^0)$, $l_2(w) = l_1(\phi(w))$ for all $w \in V_2$, and $(\phi(w_1)e\phi(w_2)) \in E_1$ for all $(w_1ew_2) \in E_2$.

Level $k = 0$. Since $\mathcal{G}^\epsilon(C)$ has unique 0-neighbourhood (v^0), we obtain the root u^0 of $\mathbf{B}(\mathcal{G}^\epsilon(C))$ where $l_1(u^0) = l'(v^0) = l(v^0)$. We obtain also the root w^0 of $\mathcal{G}(C')$ where $l_2(w^0) = l(v^0)$. We set $\phi(w^0) := u^0$.

Level $k > 0$. Let $w^k \in V_2$ be a node at level k of $\mathcal{G}(C')$. We have that w^k corresponds to a set of nodes $\{v_1^k, ..., v_m^k\}$, $v_1^k, ..., v_m^k \in V$ resulting from the normalization. Hence, $l_2(w^k) = \{l(v_1^k) \cup ... \cup l(v_m^k)\}$. By induction hypothesis, assume that $\phi(w^k) = u^k$ where the node u^k, which is obtained from executing Algorithm 1 for operator $\mathbf{B}$, corresponds to the k-neighbourhood $(v_1^k, ..., v_m^k)$ of $\mathcal{G}^\epsilon(C)$ and $l_1(u^k) = \mathsf{label}(u^k) = \{l(v_1^k) \cup ... \cup l(v_m^k)\}$ (Note that $\mathcal{G}(C')$ and $\mathcal{G}^\epsilon(C)$ share the set of nodes V). If there is not any confusion we write *neighbourhood* u^k for *node* u^k. We consider the following two cases:

1. Let $v_1^{k+1}, ..., v_l^{k+1}$ be all $\forall r$-successors of the nodes $v_1^k, ..., v_m^k$, *i.e.,* $\{v_1^{k+1}, ..., v_l^{k+1}\} = V^\forall(u^k)$. The application of normalization rule 1 yields a $(k+1)$-node $w^{k+1} = \{v_1^{k+1}, ..., v_l^{k+1}\}$ of $\mathcal{G}(C')$ and a $\forall r$-edge that connects w^k to w^{k+1}, *i.e.,* $(w^k \forall r w^{k+1}) \in E_2$. Let u^{k+1} be the $\forall r$-neighbourhood of u^k, *i.e.,* $u^{k+1} = V^\forall(u^k)$. If $\bot \in \{l(v_1^{k+1}) \cup ... \cup l(v_l^{k+1})\}$, we have that $\mathsf{label}(u^{k+1}) = \{\bot\}$ and $\bot \in l_2(w^{k+1})$. Otherwise, $\mathsf{label}(u^{k+1}) = \{l(v_1^{k+1}) \cup ... \cup l(v_l^{k+1})\} = l_2(w^{k+1})$. Therefore, the unique $\forall r$-successor w^{k+1} of w^k corresponds to the unique $\forall r$-successor u^{k+1} of u^k and $l_2(w^{k+1}) = l_1(u^{k+1})$. We set $\phi(w^{k+1}) := u^{k+1}$.
2. Let v_0^{k+1} be a r-successor of one of nodes $v_1^k, ..., v_m^k$ and $v_1^{k+1}, ..., v_l^{k+1}$ be all $\forall r$-successors of nodes $v_1^k, ..., v_m^k$. The application of the normalization rule 2 yields a $(k+1)$-node $w^{k+1} = \{v_0^{k+1}, v_1^{k+1}, ..., v_l^{k+1}\}$ of $\mathcal{G}(C')$ and a r-edge that connects w^k to w^{k+1}, *i.e.,* $(w^k r w^{k+1}) \in E_2$. Let u^{k+1} and $(u^k r u^{k+1})$ be a node and a r-edge that are generated from node u^k by Algorithm 1.

   Since v_i^{k+1} is either a $\forall r$-successor or a r-successor of one of nodes $v_1^k, ..., v_m^k$ for all

$i \in \{0,..,l\}$, we have $p(v_i^{k+1}) \in \{v_1^k,...,v_m^k\}$. Furthermore, since each node v_j^k where $v_j^k \in \{v_1^k,...,v_m^k\}$ is connected to a node $v_i^k \in \{v_1^k,...,v_m^k\}$ by an ϵ-edge, hence by Definition 5, two nodes v_0^{k+1}, v_j^{k+1} where $v_j^{k+1} \in \{v_1^{k+1},...,v_l^{k+1}\}$ are connected by an ϵ-edge. Hence, according to Algorithm 1 for operator $\mathbf{B}$, the node u^{k+1} corresponds to $(k+1)$-neighbourhood $(v_0^{k+1}, v_1^{k+1},...,v_l^{k+1})$. (Note that $V^{\epsilon}_{v_0^k} = \{v_0^{k+1}, v_1^{k+1},...,v_l^{k+1}\}$, $(v_0^k r v_0^{k+1}) \in E)$ and $l_1(u^{k+1}) = \mathsf{label}(u^{k+1})$. If $\perp \in \{l(v_0^{k+1}) \cup l(v_1^{k+1}) \cup ... \cup l(v_l^{k+1})\}$, we have that $\mathsf{label}(u^{k+1}) = \{\perp\}$ and $\perp \in l_2(w^{k+1})$. Otherwise, $\mathsf{label}(u^{k+1}) = \{l(v_0^{k+1}) \cup l(v_1^{k+1}) \cup ... \cup l(v_l^{k+1})\} = l_2(w^{k+1})$.
Conversely, from the $(k+1)$-neighbourhood $\{v_0^{k+1}, v_1^{k+1},...,v_l^{k+1}\}$ we can show that $\mathcal{G}(C')$ has a node $w^{k+1} = \{v_0^{k+1}, v_1^{k+1},...,v_l^{k+1}\}$ and a r-edge which connects w^k to w^{k+1}.
We set $\phi(w^{k+1}) := u^{k+1}$.

We have constructed a bijection ϕ as specified above from tree $\mathbf{B}(\mathcal{G}^{\epsilon}(C))$ into the tree $\mathcal{G}(C')$. According to Lemma 3, there exists an isomorphism between $\mathbf{B}(\mathcal{G}^{\epsilon}_C)$ and $\mathcal{H}$ where the tree $\mathcal{H}$ is obtained from $\mathbf{B}(\mathcal{G}^{\epsilon}(C))$ by applying the rules in Lemma 3. Therefore, it is sufficient to prove that the description tree $\mathcal{G}_C$ can be obtained from the tree $\mathcal{G}(C')$ by applying the rules in Lemma 3 (which correspond to rules 5, 6, 7 in Definition 2). In fact, each application of rules 5, 6, 7 to C' corresponds to each application of rules 5g, 6g, 7g to $\mathcal{G}(C')$ and conversely. Moreover, the application of rules 5, 6, 7 to C' allows us to obtain the strong normal form of C from which the description tree $\mathcal{G}_C$ is built. Let $\mathcal{G}'$ be the tree obtained by applying the rules 5g, 6g, 7g to $\mathcal{G}(C')$. Thus, $\mathcal{G}'$ is isomorph to $\mathcal{G}_C$. ■

Proposition 2 *Let C and D be $\mathcal{ALE}$-concept descriptions, and let $\mathcal{G}^{\epsilon}_C$ and $\mathcal{G}^{\epsilon}_D$ be their normalization graphs. Algorithm* 2 *applied to $\mathcal{G}^{\epsilon}_C$ and $\mathcal{G}^{\epsilon}_D$ can decide subsumption between C and D in polynomial space and exponential time.*

Proof: According to Remark 1, the transformation from $\mathcal{ALE}$-concept descriptions into the corresponding ϵ-trees takes a polynomial time in the size of input concept descriptions C and D (note that C and D must be transformed into weak normal form before building the corresponding ϵ-trees $\mathcal{G}^{\epsilon}(C)$ and $\mathcal{G}^{\epsilon}(D)$). Additionally, Remark 3 shows that adding nodes for stocking $\mathsf{clashes}$ increases polynomially the size of ϵ-trees. Thus, the size of normalization graphs $\mathcal{G}^{\epsilon}_C$ and $\mathcal{G}^{\epsilon}_D$ is polynomial in the size of C and D.
Algorithm 2 checks the existence of a homomorphism between between two description trees $\mathbf{B}(\mathcal{G}^{\epsilon}), \mathbf{B}(\mathcal{H}^{\epsilon})$. According to Theorem 1, Algorithm 2 allows us to decide subsumption between C and D.
According to Definition 9 (extended-neighbourhood), the number of $(k+1)$-neighbourhoods generated from a k-neighbourhood is polynomial in the size of $\mathcal{H}^{\epsilon}$ (or $\mathcal{G}^{\epsilon}$). Furthermore, since the height of $\mathcal{H}^{\epsilon}$ (or $\mathcal{G}^{\epsilon}$) is bounded by the size of tree and visited branches can be freed, the algorithm needs a piece of memory polynomial in the size of $\mathcal{H}^{\epsilon}$ (or $\mathcal{G}^{\epsilon}$) to store the neighbourhoods along the path $(w_0, w_{k+1}, ..., w_n)$ from root w_0 to leaf w_n. These paths are built by inductive calls in the algorithm. This implies that the algorithm takes an exponential time (*cf.* Remark 5) and a polynomial space. ■

Lemma 4 *Let $n_G^{k-1} = \{u_1,...,u_m\}$ and $n_H^{k-1} = \{w_1,...,w_n\}$ be $(k-1)$-neighbourhoods respectively in $\mathcal{G}^{\epsilon}$, $\mathcal{H}^{\epsilon} \in \mathcal{T}^{\mathcal{E}}_{\mathcal{ALE}}$. Let $n_{G\times H}^{k-1}$ be a $(k-1)$-neighbourhood in $\mathcal{G}^{\epsilon} \times \mathcal{H}^{\epsilon}$. Assume that $\{(u_1,w_1),...,(u_m,w_n)\} \subseteq n_{G\times H}^{k-1}$ and $l_{G\times H}(u_i,w_j) = \emptyset$, (u_i,w_j) does not have any r-successor and $\forall r$-successor for all $(u_i,w_j) \in n_{G\times H}^{k-1} \setminus \{(u_1,w_1),...,(u_m,w_n)\}$.*
It holds that there exist r-neighbourhoods ($\forall r$-neighbourhoods) $n_G^k = \{v_1,...,v_h\}$ and $n_H^k = \{z_1,...,z_l\}$ such that $n_G^k \in N(n_G^{k-1})$ and $n_H^k \in N(n_H^{k-1})$ iff there exists a r-neighbourhood ($\forall r$-neighbourhood) $n_{G\times H}^k \in N(n_{G\times H}^{k-1})$ such that $\{(v_1,z_1),...,(v_h,z_l)\} \subseteq n_{G\times H}^k$ and $l_{G\times H}(v_i,z_j) = \emptyset$, (v_i,z_j) does not have any r-successor and $\forall r$-successor for all $(v_i,z_j) \in n_{G\times H}^k \setminus \{(v_1,z_1),...,(v_h,z_l)\}$.

Proof: 1. Let $n_G^k = (v_1,...,v_a)$ and $n_H^k = (z_1,...,z_b)$ be $\forall r$-neighbourhoods respectively of n_G^{k-1} and n_H^{k-1}. Let $n_{G\times H}^k \in N(n_{G\times H}^{k-1})$ be the $\forall r$-neighbourhood. First, we show that $V_{G\times H}^{\forall}(n_{G\times H}^{k-1}) \neq \emptyset$ iff $V_G^{\forall}(n_G^{k-1}) \neq \emptyset$ and $V_H^{\forall}(n_H^{k-1}) \neq \emptyset$. Assume that $v_i \in V_G^{\forall}(n_G^{k-1})$ and $z_j \in V_H^{\forall}(n_H^{k-1})$, *i.e.*, $(u_h \forall r v_i) \in E_G$ and $(w_l \forall r z_j) \in E_H$ where $u_h \in n_G^{k-1}$ and $w_l \in n_H^{k-1}$. We have $\{(u_1,w_1),...,(u_m,w_n)\} \subseteq n_{G\times H}^{k-1}$. By Definition 10 (product), that means that $(v_i,z_j) \in V_{G\times H}^{\forall}(n_{G\times H}^{k-1})$. Conversely, assume that $(v_i,z_j) \in V_{G\times H}^{\forall}(n_{G\times H}^{k-1})$, *i.e.*, $((u_h,u_l)\forall r(v_i,z_j))$

$\in E_{G\times H}$ where $(u_h, w_l) \in n^{k-1}_{G\times H}$. We have (u_h, w_l) $\in \{(u_1, w_1), ..., (u_m, w_n)\}$ since
$\{(u_1, w_1), ..., (u_m, w_n)\} \subseteq n^{k-1}_{G\times H}$ and $(u_{h'}, w_{l'})$ does not have any $\forall r$-successor for all $(u_{h'}, w_{l'}) \in n^{k-1}_{G\times H} \setminus \{(u_1, w_1), ..., (u_m, w_n)\}$. By Definition 10 (product), that means that $v_i \in V^{\forall}_G(n^{k-1}_G)$ and z_j $\in V^{\forall}_H(n^{k-1}_H)$.
By consequent, we obtain there exists a $\forall r$-neighbourhood of $n^{k-1}_{G\times H}$ iff there exist $\forall r$-neighbourhoods of n^{k-1}_G and n^{k-1}_H.
1.1) Assume that $\mathsf{label}(n^k_G) = \{\bot\}$ and $\mathsf{label}(n^k_H)$ $= \{\bot\}$.
The definition of neighbourhood yields that $l_G(v_0)$ $= \{\bot\}$ for some $v_0 \in n^k_G$ and $l_H(z_0) = \{\bot\}$ for some $z_0 \in n^k_H$, and $n^k_G = V^{\epsilon}_G(n^{k-1}_G)$, $n^k_H = V^{\epsilon}_H(n^{k-1}_H)$ are the $\forall r$-neighbourhoods of n^{k-1}_G and n^{k-1}_H respectively.
We show that $n^k_{G\times H} = V^{\epsilon}_{GH}(n^{k-1}_{G\times H})$,
$V^{\epsilon}_G(n^{k-1}_G) \times V^{\epsilon}_H(n^{k-1}_H) \cup V^{\forall}_{GE} \cup V^{\forall}_{HE} = n^k_{G\times H}$,
$l_{G\times H}(v_i, z_j) = \{\bot\}$ for some $(v_i, z_j) \in n^k_{G\times H}$;
$l_{G\times H}(v_{i'}, z_{j'}) = \emptyset$, $(v_{i'}, z_{j'})$ does not have any r-successor and $\forall r$-successor for all $(v_{i'}, z_{j'}) \in n^k_{G\times H}$, $(v_{i'}, z_{j'}) \neq (v_i, z_j)$; and$l_{G\times H}(v_l, z_h) = \emptyset$, (v_l, z_h) does not have any r-successor and $\forall r$-successor for all (v_l, z_h) such that $(v_l, z_h) \in V^{\forall}(n^{k-1}_{G\times H})$, $(v_l, z_h)\epsilon(v_i, z_j) \in E^{\epsilon}_{G\times H}$, $(v_i, z_j) \in n^k_{G\times H}$ (*) . Note that
$V^{\forall}_{GE} := \{(v, z) \mid v{\in}V^{\forall}_G(n^{k-1}_G),\ (v\epsilon v') \in E^{\epsilon}_G,\ v' \in V^{\epsilon}_G(n^{k-1}_G),\ z \in V^{\epsilon}_H(n^{k-1}_H)\}$ and
$V^{\forall}_{HE} := \{(v, z) \mid v \in V^{\epsilon}_G(n^{k-1}_G),\ z \in V^{\forall}_H(n^{k-1}_H)\ ,\ (z\epsilon z') \in E^{\epsilon}_H,\ z' \in V^{\epsilon}_H(n^{k-1}_H)\}$.
Assume that $v_i \in V^{\epsilon}_G(n^{k-1}_G)$ and $z_j \in V^{\epsilon}_H(n^{k-1}_H)$. This means that $l_G(v_i) = \{\bot\}$ (unique) or $\emptyset$, $p(v_i) \in n^{k-1}_G$,$l_G(v_l) = \emptyset$, $(v_l\epsilon v_i) \in E^{\epsilon}_G$, $v_l \in V^{\forall}(n^{k-1}_G)$, $v_i \notin V^{\forall}(n^{k-1}_G)$, v_i does not have any $\forall r$-successor and r-successor, and $l_H(z_j) = \{\bot\}$ (unique) or $\emptyset$, $p(z_j) \in n^{k-1}_H$ ($\mathcal{P}(v_i)$ and $\mathcal{P}(z_j)$ are singleton), $l_H(z_h) = \emptyset$, $(z_h\epsilon z_j) \in E^{\epsilon}_H$, $z_h \in V^{\forall}(n^{k-1}_H)$, $z_j \notin V^{\forall}(n^{k-1}_H)$, z_j does not have any $\forall r$-successor and r-successor. Hence, we have $(v_l, z_h) \in V^{\forall}(n^{k-1}_{G\times H})$, $l_{G\times H}(v_l, z_h) = \emptyset$, $((v_l, z_h)\epsilon(v_i, z_j)) \in E^{\epsilon}_{G\times H}$, $p(v_i, z_j) = (p(v_i), p(z_j))$ $\in n^{k-1}_{G\times H}$, $(v_i, z_j) \notin V^{\forall}(n^{k-1}_{G\times H})$, $l_{G\times H}(v_i, z_j) = \{\bot\}$ (unique) or $\emptyset$. Therefore, according to Definition 9 (extended-neighbourhood), $(v_i, z_j) \in V^{\epsilon}_{GH}(n^{k-1}_{G\times H})$. Thus, $V^{\epsilon}_G(n^{k-1}_G) \times V^{\epsilon}_H(n^{k-1}_H) \subseteq n^k_{G\times H}$ and $n^k_{G\times H}$ $= V^{\epsilon}_{GH}(n^{k-1}_{G\times H})$.

Assume that $(v_i, z_j) \in V^{\forall}_{GE}$. By the definition, we have $((v_i, z_h)\epsilon(v_i, z_j)) \in E^{\epsilon}_{G\times H}$, $(v_i, z_h) \in V^{\forall}_{GH}(n^{k-1}_{G\times H})$, $(v_i, z_j) \notin V^{\forall}_{GH}(q^{k-1})$, $p(v_i, z_j) \in n^{k-1}_{G\times H}$ where $z_h \in V^{\forall}_H(n^{k-1}_H)$, $(z_h\epsilon z_j) \in E^{\epsilon}_H$, $l_G(v_i)$ $= \emptyset$, $l_H(z_h) = \emptyset$, $z_j \in V^{\epsilon}_H(n^{k-1}_H)$. This implies that $(v_i, z_j) \in V^{\epsilon}_{GH}(n^{k-1}_{G\times H})$ and $l_{G\times H}(v_i, z_h) = \emptyset$, $l_{G\times H}(v_i, z_j) = \emptyset$ and (v_i, z_j) does not have any r-successor and $\forall r$-successor. Similarly, if (v_i, z_j) $\in V^{\forall}_{HE}$ then $(v_i, z_j) \in V^{\epsilon}_{GH}(n^{k-1}_{G\times H})$ and $l_{G\times H}(v_l, z_j)$ $= \emptyset$, $l_{G\times H}(v_i, z_j) = \emptyset$ and (v_i, z_j) does not have any r-successor and $\forall r$-successor where $v_l \in V^{\forall}_G(n^{k-1}_G)$, $(v_l\epsilon v_i) \in E^{\epsilon}_G$, $v_i \in V^{\epsilon}_G(n^{k-1}_G)$.

Conversely, assume that $(v_i, z_j) \in n^k_{G\times H} = V^{\epsilon}_{GH}(n^{k-1}_{G\times H})$. According to Definition 6 (neighbourhood), we have that $(v_l, z_h) \in V^{\forall}_{GH}(n^{k-1}_{G\times H})$, $((v_l, z_h)\epsilon(v_i, z_j)) \in E^{\epsilon}_{G\times H}$, $(v_i, z_j) \notin V^{\forall}_{GH}(n^{k-1}_{G\times H})$, $p(v_i, z_j) \in n^{k-1}_{G\times H}$. From Definition 10 (product), we have $v_l \in V^{\forall}(n^{k-1}_G)$, $(v_l\epsilon v_i) \in E^{\epsilon}_G$ and $z_h \in V^{\forall}(n^{k-1}_H)$, $(z_h\epsilon z_j) \in E^{\epsilon}_G$, $p(v_i) \in m^{k-1}$, $p(z_j) \in z^{k-1}$, and either $v_i \in V^{\forall}(n^{k-1}_G)$, $z_j \in V^{\epsilon}_H(n^{k-1}_H)$ or $v_i \in V^{\epsilon}_G(n^{k-1}_G)$, $z_j \in V^{\forall}(n^{k-1}_H)$. This implies that $(v_i, z_j) \in V^{\epsilon}_G(n^{k-1}_G) \times V^{\epsilon}_H(n^{k-1}_H) \cup V^{\forall}_{GE} \cup V^{\forall}_{HE}$. It is obvious that for all $(u_i, v_j) \in V^{\epsilon}_G(n^{k-1}_G) \times V^{\epsilon}_H(n^{k-1}_H) \cup V^{\forall}_{GE} \cup V^{\forall}_{HE}$ it holds that $l_{G\times H}(v_i, z_j) = \{\bot\}$ (unique) or $\emptyset$ and (v_i, z_j) does not have any r-successor and $\forall r$-successor. Furthermore, $l_{G\times H}(v, z) = \emptyset$, (v, z) does not have any r-successor and $\forall r$-successor for all (v, z) such that $(v, z) \in V^{\forall}(n^{k-1}_{G\times H})$, $(v, z)\epsilon(v_i, z_j) \in E^{\epsilon}_{G\times H}$, (v_i, z_j) $\in n^k_{G\times H}$.
1.2) Assume that $\mathsf{label}(n^k_G) = \{\bot\}$ and $\mathsf{label}(n^k_H)$ $\neq \{\bot\}$ (or $\mathsf{label}(n^k_G) \neq \{\bot\}$ and $\mathsf{label}(n^k_H) = \{\bot\}$). The induction hypothesis (for $\mathcal{G}^{\epsilon}$ and $\mathcal{H}^{\epsilon}$) yields that $n^k_G = V^{\epsilon}_G(n^{k-1}_G)$ and n^k_G is the $\forall r$-neighbourhoods of n^{k-1}_G. This means that for all $v_i \in n^k_G$ we have $l_G(v_i) = \{\bot\}$ (unique) or $\emptyset$, $p(v_i)$ $\in n^{k-1}_G$ ($\mathcal{P}(v_i)$ is singleton),$l_G(v_l) = \emptyset$, $(v_l\epsilon v_i)$ $\in E^{\epsilon}_G$, $v_l \in V^{\forall}_G(n^{k-1}_G)$, $v_i \notin V^{\forall}_G(n^{k-1}_G)$, v_i does not have any $\forall r$-successor and r-successor; $l_G(v_l)$ $= \emptyset$, v_l does not have any $\forall r$-successor and r-successor for all v_l such that $v_l \in V^{\forall}_G(n^{k-1}_G)$, $(v_l\epsilon v_i) \in E^{\epsilon}_G$, $v_i \in V^{\epsilon}_G(n^{k-1}_G)$ (by the simplification of normalization graphs). Furthermore, $l_H(z)$ $\neq \{\bot\}$ for all $z \in n^k_H$. These imply that (v_l, z_h) $\in V^{\forall}_{GH}(n^{k-1}_{G\times H})$, $((v_l, z_h)\epsilon(v_i, z_h)) \in E^{\epsilon}_{G\times H}$, (v_i, z_h) $\notin V^{\forall}_{GH}(n^{k-1}_{G\times H})$, $p(v_i, z_h) \in n^{k-1}_{G\times H}$ where $z_h \in V^{\forall}_H(n^{k-1}_H)$ ($\mathcal{P}(v_l)$ and $\mathcal{P}(z_h)$ are singleton since v_l, z_h are $\forall r$-successors). Thus, by Definition 9 (extended-neighbourhood), $(v_i, z_h) \in V^{\epsilon}_{GH}(n^{k-1}_{G\times H})$ $\neq \emptyset$ and $n^k_{G\times H} = V^{\epsilon}_{GH}(n^{k-1}_{G\times H})$.

Let $v_0 \in n_G^k$ such that $l_G(v_0) = \{\bot\}$. According to the definition of product in [3] we obtain that the subtree $\mathbf{B}(\mathcal{G}^\epsilon)(n_G^k) \times \mathbf{B}(\mathcal{H}^\epsilon)(n_H^k)$ is equal to the subtree $\mathbf{B}(\mathcal{H}^\epsilon)(n_H^k)$. This implies that $(n_G^k, n_H^k) = n_H^k$. On the other hand, from Definition 10 (product), we have that the product graph $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$ contains the subgraph $(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon)((v_0, z_1), ..., (v_0, z_b))$ where this subgraph is obtained from the subgraph $\mathcal{H}^\epsilon(z_1, ..., z_b)$. We have to prove that $\{(v_0, z_1), ..., (v_0, z_b)\} \subseteq n_{G\times H}^k$ and $l_{G\times H}(v_{i'}, z_{j'}) = \emptyset$, $(v_{i'}, z_{j'})$ does not have any $\forall r$-successor and r-successor for all $(v_{i'}, z_{j'}) \in n_{G\times H}^k \setminus \{(v_0, z_1), ..., (v_0, z_b)\}$. Furthermore, we show that $l_{G\times H}(v_l, z_h) = \emptyset$, (v_l, z_h) does not have any $\forall r$-successor and r-successor for all (v_l, z_h) such that $(v_l, z_h) \in V_{GH}^\forall(n_{G\times H}^{k-1})$, $((v_l, z_h)\epsilon(v_i, z_j))$ and $(v_i, z_j) \in n_{G\times H}^k$.

Assume that $n_H^k = V_H^\forall(n_H^{k-1})$. Similar to above, we have $(v_l, z_h) \in V_{GH}^\forall(n_{G\times H}^{k-1})$, $((v_l, z_h)\epsilon(v_0, z_h)) \in E_{G\times H}^\epsilon$, $(v_0, z_h) \notin V_{GH}^\forall(n_{G\times H}^{k-1})$, $p(v_0, z_h) \in n_{G\times H}^{k-1}$ for all $z_h \in V_H^\forall(n_H^{k-1})$ where $v_l \in V_G^\forall(n_G^{k-1})$, $l_H(z_h) \neq \{\bot\}$, $l_G(v_0) = \{\bot\}$ such that $(v_l\epsilon v_0) \in E_G^\epsilon$, $v_0 \notin V_G^\forall(n_G^{k-1})$, $p(v_0) \in n_G^{k-1}$ ($\mathcal{P}(v_0)$ and $\mathcal{P}(z_h)$ are singleton). Therefore, $\{(v_0, z_1), ..., (v_0, z_b)\} \subseteq n_{G\times H}^k$. Let now $(v_{i'}, z_{j'}) \in n_{G\times H}^k \setminus \{(v_0, z_1), ..., (v_0, z_b)\}$. By Definition 9 (extended-neighbourhood), we have $(v_l, z_h) \in V_{GH}^\forall(n_{G\times H}^{k-1})$, $((v_l, z_h)\epsilon(v_{i'}, z_{j'})) \in E_{G\times H}^\epsilon$, $(v_{i'}, z_{j'}) \notin V_{GH}^\forall(n_{G\times H}^{k-1})$, $p(v_{i'}, z_{j'}) \in n_{G\times H}^{k-1}$ ($\mathcal{P}(v_{i'})$ and $\mathcal{P}(z_{j'})$ are singleton). This implies that $v_{i'} \in V_G^\epsilon(n_G^{k-1})$, $l_G(v_{i'}) = \emptyset$ or if $v_{i'}$is a $\forall r$-successor then $l_G(v_{i'}) = \emptyset$ and $v_{i'}$ does not have any $\forall r$-successor and r-successor (the property (*) of $V^\epsilon$is proven above). Therefore, $l_{G\times H}(v_{i'}, z_{j'}) = \emptyset$ and $(v_{i'}, z_{j'})$ does not have any $\forall r$-successor and r-successor. Furthermore, if $(v_l, z_h) \in V_{GH}^\forall(n_{G\times H}^{k-1})$, $((v_l, z_h)\epsilon(v_i, z_j))$ and $(v_i, z_j) \in n_{G\times H}^k$ then $v_l \in V_G^\forall(n_G^{k-1})$, $l_G(v_l) = \emptyset$, $(v_l\epsilon v) \in E_G^\epsilon$, $v \in V_G^\epsilon(n_G^{k-1})$. This implies that $l_{G\times H}(v_l, z_h) = \emptyset$, (v_l, z_h) does not have any $\forall r$-successor and r-successor.

Assume that $n_H^k = V_H^\epsilon(n_H^{k-1})$. We have $(v_l, z_h) \in V_{GH}^\forall(n_{G\times H}^{k-1})$, $((v_l, z_h)\epsilon(v_0, z_j)) \in E_{G\times H}^\epsilon$, $(v_0, z_j) \notin V_{GH}^\forall(n_{G\times H}^{k-1})$, $p(v_0, z_j) \in n_{G\times H}^{k-1}$ for all $z_j \in V_H^\epsilon(n_H^{k-1})$ where $v_l \in V_G^\forall(n_G^{k-1})$, $l_G(v_0) = \{\bot\}$ such that $(v_l\epsilon v_0) \in E_G^\epsilon$, $v_0 \notin V_G^\forall(n_G^{k-1})$, $p(v_0) \in n_G^{k-1}$ ($\mathcal{P}(z_j)$ is singleton since $(z_h\epsilon z_j) \in E_H^\epsilon$ and z_h is a $\forall r$-successor). This implies that $\{(v_0, z_1), ..., (v_0, z_b)\} \subseteq n_{G\times H}^k$. Let now $(v_{i'}, z_{j'}) \in n_{G\times H}^k \setminus \{(v_0, z_1), ..., (v_0, z_b)\}$. Similar to above, we have that $v_{i'} \in V_G^\epsilon(n_G^{k-1})$, $l_G(v_{i'}) = \emptyset$ or if $v_{i'}$is a $\forall r$-successor then $l_G(v_{i'}) = \emptyset$ and $v_{i'}$ does not have any successor (the property (*) of $V^\epsilon$is proven above). Therefore, $l_{G\times H}(v_{i'}, z_{j'}) = \emptyset$ and $(v_{i'}, z_{j'})$ does not have any $\forall r$-successor and r-successor. Furthermore, if $(v_l, z_h) \in V_{GH}^\forall(n_{G\times H}^{k-1})$, $((v_l, z_h)\epsilon(v_i, z_j))$ and $(v_i, z_j) \in n_{G\times H}^k$ then we can show that $l_{G\times H}(v_l, z_h) = \emptyset$, (v_l, z_h) does not have any $\forall r$-successor and r-successor.

Thus, we have shown $\{(v_0, z_1), ..., (v_0, z_b)\} \subseteq n_{G\times H}^k$ and $l_{G\times H}(v_{i'}, z_{j'}) = \emptyset$, $(v_{i'}, z_{j'})$ does not have any $\forall r$-successor and r-successor for all $(v_{i'}, z_{j'}) \in n_{G\times H}^k \setminus \{(v_0, z_1), ..., (v_0, z_b)\}$.

1.3) Assume that $\mathsf{label}(n_G^k) \neq \{\bot\}$ and $\mathsf{label}(n_H^k) \neq \{\bot\}$. We consider the three following cases:

i) Assume that $n_G^k = V_G^\forall(n_G^{k-1})$ and $n_H^k = V_H^\forall(n_H^{k-1})$.

It holds that $n_{G\times H}^k = V_{GH}^\forall(n_{G\times H}^{k-1})$ since if $n_{G\times H}^{k-1} = V_{GH}^\epsilon(n_{G\times H}^{k-1})$ then $n_G^k = V_G^\epsilon(n_G^{k-1})$ or $n_H^k = V_H^\epsilon(n_H^{k-1})$. Moreover, we have
$V_{GH}^\forall(n_{G\times H}^{k-1}) = \{(v_1, z_1), ..., (v_a, z_b)\}$.

ii) Assume that $n_G^k = V_G^\epsilon(n_G^{k-1})$ and $n_H^k = V_H^\forall(n_H^{k-1})$ (or $n_G^k = V_G^\forall(n_G^{k-1})$, $n_H^k = V_H^\epsilon(n_H^{k-1})$). We show that
$n_{G\times H}^k = V_{GH}^\epsilon(n_{G\times H}^{k-1}) = \{(v_1, z_1), ..., (v_a, z_b)\}$.

Let $(v_i, z_h) \in \{(v_1, z_1), ..., (v_a, z_b)\}$. We have that $(v_l, z_h) \in V_{GH}^\forall(n_{G\times H}^{k-1})$, $((v_l, z_h)\epsilon(v_i, z_h)) \in E_{G\times H}^\epsilon$, $(v_i, z_h) \notin V_{GH}^\forall(n_{G\times H}^{k-1})$, $p(v_i, z_h) \in n_{G\times H}^{k-1}$ for all $z_h \in V_H^\forall(n_H^{k-1})$ and for all $v_i \in V_G^\epsilon(n_G^{k-1})$ such that $v_l \in V_G^\forall(n_G^{k-1})$, $(v_l\epsilon v_i) \in E_G^\epsilon$, $v_i \notin V_G^\forall(n_G^{k-1})$, $p(v_i) \in n_G^{k-1}$ ($\mathcal{P}(v_i)$ and $\mathcal{P}(z_h)$ are singleton since z_h is a $\forall r$-successor and $(v_l\epsilon v_i) \in E_G^\epsilon$, and v_l is a $\forall r$-successor). Therefore, $(v_i, v_h) \in V_{GH}^\epsilon(n_{G\times H}^{k-1})$ and $\{(v_1, z_1), ..., (v_a, z_b)\} \subseteq n_{G\times H}^k$. Conversely, let $(v_i, z_j) \in V_{GH}^\epsilon(n_{G\times H}^{k-1})$. We have $(v_l, z_h) \in V_{GH}^\forall(n_{G\times H}^{k-1})$, $((v_l, z_h)\epsilon(v_i, z_j)) \in E_{G\times H}^\epsilon$, $(v_l, z_h) \in V_{GH}^\forall(n_{G\times H}^{k-1})$, $(v_i, z_j) \notin V_{GH}^\forall(n_{G\times H}^{k-1})$, $p(v_i, z_j) \in n_{G\times H}^{k-1}$. This implies that $v_l \in V_G^\forall(n_G^{k-1})$, $z_h \in V_H^\forall(n_H^{k-1})$, and $v_i \notin V_G^\forall(n_G^{k-1})$, $p(v_i) \in n_G^{k-1}$ or $z_j \notin V_H^\forall(n_H^{k-1})$, $p(z_j) \in n_H^{k-1}$. If $z_j \notin V_H^\forall(n_H^{k-1})$, $p(z_j) \in n_H^{k-1}$ then $z_j \in V_H^\epsilon(n_H^{k-1})$ and thus $n_H^k = V_H^\epsilon(n_H^{k-1})$. This is a contradiction. Therefore, $V_{GH}^\epsilon(n_{G\times H}^{k-1}) = \{(v_1, z_1), ..., (v_a, z_b)\}$. As above, we can show that if $(v_l, z_h) \in V_{GH}^\forall(n_{G\times H}^{k-1})$, $((v_l, z_h)\epsilon(v_i, z_j))$ and $(v_i, z_j) \in n_{G\times H}^{k-1}$ then
$l_{G\times H}(v_l, z_h) = \emptyset$, (v_l, z_h) does not have any $\forall r$-successor and r-successor.

iii) Assume that $m^k = V_G^\epsilon(n_G^{k-1})$ and $n^k = V_H^\epsilon(n_H^{k-1})$.

We show that $n^k_{G\times H} = V^\epsilon_{GH}(n^{k-1}_{G\times H})$ and $\{(v_1, z_1), ..., (v_a, z_b)\} \subseteq n^k_{G\times H}$ and $l_{G\times H}(v_{i'}, z_{j'}) = \emptyset$, $(v_{i'}, z_{j'})$ does not have any $\forall r$-successor and r-successor for all $(v_{i'}, z_{j'}) \in n^k_{G\times H} \setminus \{(v_1, z_1), ..., (v_a, z_b)\}$. Furthermore, we show that $l_{G\times H}(v_l, z_h) = \emptyset$, (v_l, z_h) does not have any $\forall r$-successor and r-successor for all (v_l, z_h) such that $(v_l, z_h) \in V^\forall_{GH}(n^{k-1}_{G\times H})$, $((v_l, z_h)\epsilon(v_i, z_j))$ and $(v_i, z_j) \in n^k_{G\times H}$.

We have $(v_l, z_h) \in V^\forall_{GH}(n^{k-1}_{G\times H})$, $((v_l, z_h)\epsilon(v_i, z_j)) \in E^\epsilon_{G\times H}$, $(v_i, z_j) \notin V^\forall_{GH}(n^{k-1}_{G\times H})$, $p(v_i, z_j) \in n^{k-1}_{G\times H}$ for all $v_i \in V^\epsilon_G(n^{k-1}_G)$ and $z_j \in V^\epsilon_H(n^{k-1}_H)$ ($\mathcal{P}(v_i)$ and $\mathcal{P}(z_j)$ are singleton). This implies that $\{(v_0, z_1), ..., (v_0, z_b)\} \subseteq n^k_{G\times H}$. Let now $(v_{i'}, z_{j'}) \in n^k_{G\times H} \setminus \{(v_1, z_1), ..., (v_1, z_b)\}$. Similar to above, we have that $v_{i'} \in V^\epsilon_G(n^{k-1}_G)$ or if $v_{i'}$is a $\forall r$-successor then $l_G(v_{i'}) = \emptyset$ and $v_{i'}$ does not have any $\forall r$-successor and r-successor (the property (*) of V^ϵ is proven above). Therefore, $l_{G\times H}(v_{i'}, z_{j'}) = \emptyset$ and $(v_{i'}, z_{j'})$ does not have any $\forall r$-successor and r-successor. Furthermore, if $(v_l, z_h) \in V^\forall_{GH}(n^{k-1}_{G\times H})$, $((v_l, z_h)\epsilon(v_i, z_j)) \in E^\epsilon_{G\times H}$ and $(v_i, z_j) \in n^k_{G\times H}$ then $v_l \in V^\forall_G(n^{k-1}_G)$, $z_h \in V^\forall_H(n^{k-1}_H)$ such that $(v_l\epsilon v_i) \in E^\epsilon_G$, $v_i \in V^\epsilon_G(n^{k-1}_G)$ or $(z_h\epsilon z_j) \in E^\epsilon_H$, $z_j \in V^\epsilon_H(n^{k-1}_H)$. This implies that $l_{G\times H}(v_l, z_h) = \emptyset$, (v_l, z_h) does not have any $\forall r$-successor and r-successor.

2. Let m^k_i and n^k_j be r-neighbourhoods respectively of n^{k-1}_G and n^{k-1}_H. Let q^k_{ij} be a r-neighbourhood of $n^{k-1}_{G\times H}$ such that

$m^k_i = \{v_i\} \cup V^\epsilon_{GE}$, $V^\epsilon_{GE} = \{v'_l \mid (v_i\epsilon v'_l) \in E^\epsilon_G, \mathcal{P}(v'_l) \cap n^{k-1}_G \neq \emptyset\}$, $(u_i r v_i) \in E_G$,

$n^k_j = \{z_j\} \cup V^\epsilon_{HE}$, $V^\epsilon_{HE} = \{z'_l \mid (z_j\epsilon z'_l) \in E^\epsilon_H, \mathcal{P}(z'_l) \cap n^{k-1}_H \neq \emptyset\}$, $(w_j r z_j) \in E_H$,

$q^k_{ij} = \{(v_i, z_j)\} \cup \bigcup_{(v_l, y_{l'})\in V^\epsilon}(v_l, y_{l'})$, where

$V^\epsilon = \{(v_l, z_{l'}) \mid (v_i, z_j)\epsilon(v_l, z_{l'}), \mathcal{P}(v_l, z_{l'}) \cap n^{k-1}_{G\times H} \neq \emptyset\}$

First, we show that there exists $(v_i, z_j) \in q^k_{ij}$ iff $v_i \in m^k_i$ and $z_j \in n^k_i$. Assume that $v_i \in m^k_i$ and $z_j \in n^k_j$, *i.e.*, $(u_h r v_i) \in E_G$ and $(w_l r z_j) \in E_H$ where $u_h \in n^{k-1}_G$ and $u_l \in n^{k-1}_H$. By the induction hypothesis, we have $\{(u_1, w_1), ..., (u_m, w_n)\} \subseteq n^{k-1}_{G\times H}$. By Definition 10 (product), we obtain $(v_i, z_j) \in q^k_{ij}$. Conversely, assume that $(v_i, z_j) \in q^k_{ij}$, *i.e.*, $((u_h, u_l) r (v_i, z_j)) \in E_{G\times H}$ where $(u_h, w_l) \in n^{k-1}_{G\times H}$. By the induction hypothesis, we have $(u_h, w_l) \in \{(u_1, w_1), ..., (u_m, w_n)\}$ since $\{(u_1, w_1), ..., (u_m, w_n)\} \subseteq n^{k-1}_{G\times H}$ and $(u_{h'}, w_{l'})$ does not have any r-successor for all $(u_{h'}, w_{l'}) \in q^{k-1} \setminus \{(u_1, w_1), ..., (u_m, w_n)\}$. By Definition 10 (product), that means that $v_i \in m^k_i$ and $z_j \in n^k_j$. We have to prove that $m^k_i \times n^k_j = q^k_{ij}$. We have that $\mathsf{label}(q^k_{ij}) \neq \{\perp\}$ if $\mathsf{label}(n^k_G) \neq \{\perp\}$ and $\mathsf{label}(n^k_H) \neq \{\perp\}$ (if $\mathcal{G}^\epsilon$ and $\mathcal{H}^\epsilon$ are normalization graphs then $\mathsf{label}(n^k_G) \neq \{\perp\}$ and $\mathsf{label}(n^k_H) \neq \{\perp\}$).

What remains to be shown is that i) if $(v_l, z_{l'}) \in V^\epsilon$ then $v_l \in V^\epsilon_{GE}$ and $z_{l'} \in V^\epsilon_{HE}$ ii) if $v_l \in V^\epsilon_{GE}$ and $z_{l'} \in V^\epsilon_{HE}$ then $(v_l, z_{l'}) \in V^\epsilon$.

1. For each $(v_l, y_{l'}) \in V^\epsilon$ we obtain that $v_l \in V^\epsilon_{GE}$ and $y_{l'} \in V^\epsilon_{HE}$.
   Let $(v_l, y_{l'}) \in V^\epsilon$. This yields that there exist edges $(v_i, y_j)\epsilon(v_l, y_{l'}) \in E^\epsilon_{G\times H}$, $(u_i, w_j) r (v_i, y_j) \in E_{G\times H}$ and $\mathcal{P}(v_l, y_{l'}) = (\mathcal{P}(v_l), \mathcal{P}(y_{l'})) \cap q^{k-1} \neq \emptyset$. This implies that $(v_i\epsilon v_l) \in E^\epsilon_G$, $(y_j\epsilon y_{l'}) \in E^\epsilon_H$ where $\mathcal{P}(v_l) \cap m^{k-1} \neq \emptyset$, $\mathcal{P}(y_{l'}) \cap n^{k-1} \neq \emptyset$. From $u_i \in m^{k-1}$, $(u_i r v_i) \in E_G$, $\mathcal{P}(v_l) \cap m^{k-1} \neq \emptyset$ and $(v_i\epsilon v_l) \in E^\epsilon_G$, we obtain $v_l \in V^\epsilon_{GE}$. From $w_j \in n^{k-1}$, $(w_j r y_j) \in E_H$, $\mathcal{P}(y_{l'}) \cap n^{k-1} \neq \emptyset$ and $(y_j\epsilon y_{l'}) \in E^\epsilon_H$, we obtain $y_{l'} \in V^\epsilon_{HE}$.
2. For each $v_l \in V^\epsilon_{GE}$ and $y_{l'} \in V^\epsilon_{HE}$ we obtain that $(v_l, y_{l'}) \in V^\epsilon$.
   Let $v_l \in V^\epsilon_{GE}$ and $y_{l'} \in V^\epsilon_{HE}$. This yields that there are ϵ-edges $(v_i\epsilon v_l) \in E^\epsilon_G$, $(y_j\epsilon y_{l'}) \in E^\epsilon_H$, $\mathcal{P}(v_l) \cap m^{k-1} \neq \emptyset$, $\mathcal{P}(y_{l'}) \cap n^{k-1} \neq \emptyset$ and $(u_i r v_i) \in E_G$, $(w_j r y_j) \in E_H$. This implies that $(v_i, y_j)\epsilon(v_l, y_{l'}) \in E^\epsilon_{G\times H}$, $((u_i, w_j) r (v_i, y_j)) \in E_{G\times H}$ and $\mathcal{P}(v_l, y_{l'}) = (\mathcal{P}(v_l), \mathcal{P}(y_{l'})) \cap x^{k-1} \neq \emptyset$. Thus, $(v_l, y_{l'}) \in V^\epsilon$.

■

Theorem 2 *Let* $\mathcal{G}^\epsilon, \mathcal{H}^\epsilon \in \mathcal{T}^{\mathcal{E}}_{\mathcal{ALE}}$. *There exists an isomorphism between* $\mathbf{B}(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon)$ *and* $\mathbf{B}(\mathcal{G}^\epsilon) \times \mathbf{B}(\mathcal{H}^\epsilon)$.

Lemma A. *Let* $\{u_1, ..., u_m\}$ *and* $\{w_1, ..., w_n\}$ *be* k*-neighbourhoods respectively on graphs* $\mathcal{G}^\epsilon_1 = (V_1, E_1 \cup E^\epsilon_1, l_1)$ *and* $\mathcal{G}^\epsilon_2 = (V_2, E_2 \cup E^\epsilon_2, l_2)$. *Let* $\{(u_1, w_1), ..., (u_m, w_n)\}$ *be the corresponding* k*-neighbourhood of the product graph* $\mathcal{G}^\epsilon_1 \times \mathcal{G}^\epsilon_2$. *If* $\mathsf{label}\{u_1, ..., u_m\} = \{\perp\}$ ($\mathsf{label}\{w_1, ..., w_n\} = \{\perp\}$) *then* $\mathsf{label}\{(u_1, w_1), ..., (u_m, w_n)\} = \mathsf{label}\{w_1, ..., w_n\}$ ($\mathsf{label}\{u_1, ..., u_m\}$). *Otherwise, it holds that*

$$\mathsf{label}\{(u_1, w_1), ..., (u_m, w_n)\} = \mathsf{label}\{u_1, ..., u_m\} \cap \mathsf{label}\{w_1, ..., w_n\}$$

Proof of the lemma. According to the definition of function label, $\mathsf{label}\{(u_1, w_1), ..., (u_m, w_n)\} = \{\perp\}$ iff $\mathsf{label}(u_i, w_j) = \{\perp\}$ for some $i \in \{1, ..., m\}$,

$j \in \{1,...,n\}$. From this, Definition 10 (product) yields that $\mathsf{label}\{u_1,...,u_m\}= \{\perp\}$ and $\mathsf{label}\{w_1,...,w_n\}= \{\perp\}$. Assume that $\mathsf{label}\{u_1,...,u_m\}\neq \{\perp\}$ and $\mathsf{label}\{w_1,...,w_n\} \neq \{\perp\}$. (if $\mathsf{label}\{u_1,...,u_m\}= \{\perp\}$ or $\mathsf{label}\{w_1,...,w_n\} = \{\perp\}$ the Lemma is obvious from Definition 10). We have that $\mathsf{label}\{(u_1, w_1),...,(u_m, w_n)\} = l(u_1, w_1) \cup ... \cup l(u_m, w_n) = (l(u_1) \cap l(w_1)) \cup ... \cup (l(u_m) \cap l(w_n))$;
On the other hand, according to the definition of function label, it is that $\mathsf{label}\{u_1,...,u_m\} = l(u_1) \cup ... \cup l(u_m)$ and $\mathsf{label}\{w_1,...,w_n\}= l(w_1) \cup ... \cup l(w_n)$. Therefore, $\mathsf{label}\{(u_1, w_1),...,(u_m, w_n)\} = \mathsf{label}\{u_1,...,u_m\} \cap \mathsf{label}\{w_1,...,w_n\}$.

Proof: (*Proof of the theorem*).
Let $\mathcal{G}^\epsilon = (V_G, E_G \cup E^\epsilon_G, l_G)$ and $\mathcal{H}^\epsilon = (V_H, E_H \cup E^\epsilon_H, l_H)$ and v^0, w^0 be the roots of $\mathcal{G}^\epsilon$, $\mathcal{H}^\epsilon$. We denote $|\mathcal{G}^\epsilon|$ as the depth of graph $\mathcal{G}^\epsilon$. Assume that $|\mathcal{G}^\epsilon| \leq |\mathcal{H}^\epsilon|$. We will construct by induction on the level of graph $\mathcal{G}^\epsilon$ an isomorphism ϕ from tree $\mathbf{B}(\mathcal{G}^\epsilon) \times \mathbf{B}(\mathcal{H}^\epsilon)$=$(V_2, E_2, z^0, l_2)$ to tree $\mathbf{B}(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon) = (V_1, E_1, x^0, l_1)$.

Level $k = 0$.
At level 0, since product $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$ has unique neighbourhood $\{(v^0, w^0)\}$ without outgoing or ingoing ϵ-edge (with the exception of ϵ-cycle), $\mathbf{B}(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon)$ has the root $x^0 = (v^0, w^0)$. Similarly, since $\mathcal{G}^\epsilon$ has unique node v^0 without outgoing or ingoing ϵ-edge (with the exception of ϵ-cycle) and $\mathcal{H}^\epsilon$ has unique node w^0 without outgoing or ingoing ϵ-edge at level 0 (with the exception of ϵ-cycle), thus $\mathbf{B}(\mathcal{G}^\epsilon) \times \mathbf{B}(\mathcal{H}^\epsilon)$ has the root $z^0 = (v^0, w^0)$.
Assume that $l_G(v^0) = \{\perp\}$ ($l_H(w^0) = \{\perp\}$). From Definition 10 we have that $\mathbf{B}(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon) = \mathbf{B}(\mathcal{H}^\epsilon)$ ($\mathbf{B}(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon) = \mathbf{B}(\mathcal{G}^\epsilon)$). On the other hand, we also have $\mathbf{B}(\mathcal{G}^\epsilon) = \{v^0\}$ ($\mathbf{B}(\mathcal{H}^\epsilon) = \{w^0\}$) where $l(v^0) = \{\perp\}$ ($l(w^0) = \{\perp\}$) (Algorithm 1), and thus $\mathbf{B}(\mathcal{G}^\epsilon) \times \mathbf{B}(\mathcal{H}^\epsilon) = \mathbf{B}(\mathcal{H}^\epsilon)$ ($\mathbf{B}(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon) = \mathbf{B}(\mathcal{G}^\epsilon)$) . If $l_G(v^0) \neq \{\perp\}$, $l_1(x^0) = \mathsf{label}(v^0,w^0) = (l_G(v^0) \cap l_H(w^0))$ and $l_2(z^0) = (l_G(v^0) \cap l_H(w^0))$. Thus, we set $\phi(z^0) := x^0$.

Level $k > 0$
Let $m^{k-1} = \{u_1,...,u_m\}$ be a $(k-1)$-neighbourhood of $\mathcal{G}^\epsilon$ and $n^{k-1} = \{w_1,...,w_n\}$ be a $(k-1)$-neighbourhood of $\mathcal{H}^\epsilon$. Let q^{k-1} be a $(k-1)$-neighbourhood of $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$ such that $\phi(m^{k-1}, n^{k-1}) = q^{k-1}$. If $\mathsf{label}(m^{k-1}) = \{\perp\}$ and $\mathsf{label}(n^{k-1}) = \{\perp\}$ then $\mathsf{label}(q^{k-1}) = \{\perp\}$ (since ϕ is an isomorphism) and $N^k(q^{k-1}) = N^k(m^{k-1}) = N^k(n^{k-1}) = \emptyset$.

Let $m^{k-1} \times n^{k-1} = \{(u_1, w_1),...,(u_m, w_n)\}$ and we denote (m^{k-1}, n^{k-1}) as a node in $\mathbf{B}(\mathcal{G}^\epsilon) \times \mathbf{B}(\mathcal{H}^\epsilon)$. As induction hypothesis, we assume that

1. $m^{k-1} \times n^{k-1} \subseteq q^{k-1}$,
2. One of two following conditions is satisfied:
   (a) $\mathsf{label}(m^{k-1}, n^{k-1}) = \mathsf{label}(q^{k-1}) = \{\perp\}$; $q^{k-1}= V^\epsilon$; and for all $(u_i, w_j) \in q^{k-1} \setminus m^{k-1} \times n^{k-1}$ it holds that $l(u_i, w_j) = \emptyset$ or $\{\perp\}$ (unique (u_i, w_j) such that $l(u_i, w_j) = \{\perp\}$) and (u_i, w_j) does not have any r-successor and $\forall r$-successor. Furthermore, $l_{G\times H}(u, w) = \emptyset$ and (u, w) does not have any r-successor and $\forall r$-successor for all $(u, w) \in V_{G\times H}$ such that $((u, v)\epsilon(u_i, w_j)) \in E^\epsilon_{G\times H}$, (v, w) is a $\forall r$-successor and $(u_i, w_j) \in q^{k-1}$.
   (b) $\mathsf{label}(m^{k-1}, n^{k-1}) = \mathsf{label}(q^{k-1}) \neq \{\perp\}$; and for all $(v_i, z_j) \in q^{k-1} \setminus m^{k-1} \times n^{k-1}$ it holds that $l(v_i, z_j) = \emptyset$ and (v_i, z_j) does not have any r-successor and $\forall r$-successor. Furthermore, if $q^{k-1}= V^\epsilon$ then $l_{G\times H}(u, w) = \emptyset$ and (u, w) does not have any r-successor and $\forall r$-successor for all $(u, w) \in V_{G\times H}$ such that $((u, v)\epsilon(u_i, w_j)) \in E^\epsilon_{G\times H}$, (v, w) is a $\forall r$-successor and $(u_i, w_j) \in q^{k-1}$.

Note that this hypothesis is verified if $\mathcal{G}^\epsilon$ and $\mathcal{H}^\epsilon$ are normalization graphs.
I) $\mathsf{label}(m^{k-1}) = \{\perp\}$ and $\mathsf{label}(n^{k-1}) \neq \{\perp\}$ (or $\mathsf{label}(m^{k-1}) \neq \{\perp\}$ and $\mathsf{label}(n^{k-1}) = \{\perp\}$).
By the induction hypothesis, we have that $l_G(u_i) = \{\perp\}$ for some $u_i \in m^{k-1}$ and, $l_G(u_{i'}) = \emptyset$, $u_{i'}$ does not have any $\forall r$-successor and r-successor for all $u_{i'} \in m^{k-1}$, $u_{i'} \neq u_i$. Furthermore, from $\mathsf{label}(n^{k-1}) \neq \{\perp\}$ the definition of function label yields that $l_H(w_j) \neq \{\perp\}$ for all $w_j \in n^{k-1}$. From Definition 10 (product), it is that the product graph $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$ contains the subgraph $(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon)((u_i, w_1),...,(u_i, w_n))$ which is obtained from the subgraph $\mathcal{H}^\epsilon(w_1,...,w_n)$ where $l_G(u_i) = \{\perp\}$. By the induction hypothesis, we have $q^{k-1}= V^\epsilon$, $\{(u_i, w_1),...,(u_i, w_n)\} \subseteq q^{k-1}$ such that $l_G(u_i) = \{\perp\}$, and $l_{G\times H}(u_{i'}, w_j) = \emptyset$, $(u_{i'}, w_j)$ does not have any r-successor and $\forall r$-successor for all $(v_{i'}, z_j) \in q^{k-1} \setminus \{(u_i, w_1),...,(u_i, w_n)\}$.
On the other hand, according to the definition of product in [3] we obtain that the subtree $\mathbf{B}(\mathcal{G}^\epsilon)(m^{k-1}) \times \mathbf{B}(\mathcal{H}^\epsilon)(n^{k-1})$ is equal to the subtree $\mathbf{B}(\mathcal{H}^\epsilon)(n^{k-1})$. This implies that $\mathbf{B}((\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon)(q^{k-1})) = \mathbf{B}(\mathcal{G}^\epsilon)\ (m^{k-1}) \times \mathbf{B}(\mathcal{H}^\epsilon)(n^{k-1})$.

II) $\mathsf{label}(m^{k-1}) \neq \{\bot\}$ and $\mathsf{label}(n^{k-1}) \neq \{\bot\}$.
1. Let $m^k = (v_1, ..., v_a)$ and $n^k = (z_1, ..., z_b)$ be $\forall r$-neighbourhoods (or r-neighbourhoods) respectively of m^{k-1} and n^{k-1}. By Lemma 4 and the induction hypothesis, there exists the $\forall r$-neighbourhood (or r-neighbourhood) q^k of q^{k-1} such that $m^k \times n^k \subseteq q^k$ and for all $(v_i, z_j) \in q^k \setminus m^k \times n^k$ it holds that $l(v_i, z_j) = \emptyset$ and (v_i, z_j) does not have any r-successor and $\forall r$-successor (*).
i) If $\mathsf{label}(m^k) = \{\bot\}$ and $\mathsf{label}(n^k) = \{\bot\}$ then there exist $v_h \in m^k$ and $z_l \in n^k$ such that $l_G(v_h) = \{\bot\}$ and $l_H(z_l) = \{\bot\}$. It is obvious that $(v_h, z_l) \in m^k \times n^k$ and $l_{G\times H}(v_h, z_l) = \{\bot\}$. Therefore, $\mathsf{label}(q^k) = \{\bot\}$.
ii) If $\mathsf{label}(m^k) = \{\bot\}$ and $\mathsf{label}(n^k) \neq \{\bot\}$ (or $\mathsf{label}(m^k) \neq \{\bot\}$ and $\mathsf{label}(n^k) = \{\bot\}$) then there exist $v_h \in m^k$ such that $l_G(v_h) = \{\bot\}$ and $l_H(z_j) \neq \{\bot\}$ for all $z_j \in n^k$. According to the definition of product in [3] we obtain that the subtree $\mathbf{B}(\mathcal{G}^\epsilon)(m^k) \times \mathbf{B}(\mathcal{H}^\epsilon)(n^k)$ is equal to the subtree $\mathbf{B}(\mathcal{H}^\epsilon)(n^k)$. This implies that (m^k, n^k)=n^k. On the other hand, from Definition 10 (product), we have that the product graph $\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon$ contains the subgraph $(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon)((v_h, z_1), ..., (v_h, z_b))$ where this subgraph is obtained from the subgraph $\mathcal{H}^\epsilon(z_1, ..., z_b)$. Moreover, according to (*), we have $\{(v_h, z_1), ..., (v_h, z_b)\} \subseteq q^k$ and $l_{G\times H}(v_{i'}, z_{j'}) = \emptyset$, $(v_{i'}, z_{j'})$ does not have any $\forall r$-successor and r-successor for all $(v_{i'}, z_{j'}) \in q^k \setminus \{(v_h, z_1), ..., (v_h, z_b)\}$. Therefore, $\mathsf{label}(q^k) = \mathsf{label}(n^k)$.
iii) If $\mathsf{label}(m^k) \neq \{\bot\}$ and $\mathsf{label}(n^k) \neq \{\bot\}$ then from (*) and Lemma A, we have $\mathsf{label}(q^k) = \mathsf{label}(m^k) \cap \mathsf{label}(n^k)$.
To sum up, the isomorphism is extended as follows: $\phi(m^k, n^k) := q^k$ where m^k, $n^k$and q^k are $\forall r$-neighbourhoods (r-neighbourhoods) respectively of m^{k-1}, $n^{k-1}$and q^{k-1}. The induction principle guarantees that ϕ is an isomorphism between trees $\mathbf{B}(\mathcal{G}^\epsilon \times \mathcal{H}^\epsilon)$ and $\mathbf{B}(\mathcal{G}^\epsilon) \times \mathbf{B}(\mathcal{H}^\epsilon)$. ■

Proposition 4 Let $C = C_1 \sqcup ... \sqcup C_n$ be an $\mathcal{ALC}$-concept description where $\bot \sqsubset C_1, ..., C_n$. The approximation of C by $\mathcal{ALE}$-concept description can be computed as follows:

$$approx_{\mathcal{ALE}}(C) \equiv \text{lcs}\{approx_{\mathcal{ALE}}(C_1), \ldots , approx_{\mathcal{ALE}}(C_n)\}$$

Proof: The proof is direct from the definitions of *lcs* and *approx*.

First, prove the proposition with $n = 2$. We have that
$C_1 \sqcup C_2 \sqsubseteq lcs\{approx_{\mathcal{ALE}}(C_1),\ approx_{\mathcal{ALE}}(C_2)\}$
since $C_1 \sqsubseteq approx_{\mathcal{ALE}}(C_1)$, $C_2 \sqsubseteq approx_{\mathcal{ALE}}(C_2)$,
$approx_{\mathcal{ALE}}(C_1) \sqsubseteq lcs\{approx_{\mathcal{ALE}}(C_1), approx_{\mathcal{ALE}}(C_2)\}$ and
$approx_{\mathcal{ALE}}(C_2) \sqsubseteq lcs\{approx_{\mathcal{ALE}}(C_1), approx_{\mathcal{ALE}}(C_2)\}$.
Assume that there exists an $\mathcal{ALE}$-concept description D such that
$C_1 \sqcup C_2 \sqsubseteq D \sqsubseteq lcs\{approx_{\mathcal{ALE}}(C_1), approx_{\mathcal{ALE}}(C_2)\}$ (**).
We show that $approx_{\mathcal{ALE}}(C_1) \sqcup approx_{\mathcal{ALE}}(C_2) \not\sqsubseteq D$ is impossible.
Indeed, there exist an interpretation $(\Delta, .^{\mathcal{I}})$ and an individual $d^{\mathcal{I}} \in \Delta$ such that $d^{\mathcal{I}} \in (approx_{\mathcal{ALE}}(C_1) \sqcup approx_{\mathcal{ALE}}(C_2))^{\mathcal{I}}$ ($\bot \sqsubset C_1, C_2$) and $d^{\mathcal{I}} \notin D^{\mathcal{I}}$.
There are the two following possibilities:

- If $d^{\mathcal{I}} \in (approx_{\mathcal{ALE}}(C_1))^{\mathcal{I}}$ and $d^{\mathcal{I}} \notin D^{\mathcal{I}}$, then $C_1 \sqsubseteq D \sqcap approx_{\mathcal{ALE}}(C_1) \sqsubset approx_{\mathcal{ALE}}(C_1)$, which contradicts the approximation definition since $D \sqcap approx_{\mathcal{ALE}}(C_1)$ is an $\mathcal{ALE}$-concept description.
- If $d^{\mathcal{I}} \in (approx_{\mathcal{ALE}}(C_2))^{\mathcal{I}}$ and $d^{\mathcal{I}} \notin D^{\mathcal{I}}$, then $C_2 \sqsubseteq D \sqcap approx_{\mathcal{ALE}}(C_2) \sqsubset approx_{\mathcal{ALE}}(C_{12})$, which contradicts the approximation definition since $D \sqcap approx_{\mathcal{ALE}}(C_2)$ is an $\mathcal{ALE}$-concept description.

Hence, we obtain that
$approx_{\mathcal{ALE}}(C_1) \sqcup approx_{\mathcal{ALE}}(C_2) \sqsubseteq D$. This implies that
$lcs(approx_{\mathcal{ALE}}(C_1),\ approx_{\mathcal{ALE}}(C_2)\} \equiv D$ since the hypothesis (**),
$approx_{\mathcal{ALE}}(C_1) \sqsubseteq D$, $approx_{\mathcal{ALE}}(C_2) \sqsubseteq D$ and $lcs\{approx_{\mathcal{ALE}}(C_1),\ approx_{\mathcal{ALE}}(C_2)\}$ is the least $\mathcal{ALE}$-concept description such that
$approx_{\mathcal{ALE}}(C_1) \sqsubseteq lcs\{approx_{\mathcal{ALE}}(C_1), approx_{\mathcal{ALE}}(C_2)\}$,
$approx_{\mathcal{ALE}}(C_2) \sqsubseteq lcs\{approx_{\mathcal{ALE}}(C_1), approx_{\mathcal{ALE}}(C_2)\}$.
By induction on n, the proposition can be proven for $n > 2$ by using the following property of the *lcs*:

$lcs\{C_1, ..., C_n\} \equiv lcs\{lcs\{C_1, ..., C_{n-1}\}, C_n\}$ ■