Planning of Axiom Absorptions

Jiewen Wu and Volker Haarslev

Concordia University,
Montréal, Québec, H3G 1M8 Canada
{w_jiewen,haarslev}@cs.concordia.ca

Abstract. Absorptions are generally employed in Description Logics (DL) reasoners in a uniform way regardless of the structure of an input knowledge base. In this paper we present an approach to encode some state-of-the-art absorption techniques into a state space planner, aiming to achieve a better solution. The planner applies appropriate operators to general axioms and produces a solution with a minimized cost to automatically organize these absorptions in a certain sequence to facilitate DL reasoning. Compared to predetermined or fixed applications of established absorptions, such a solution is more flexible and probable to absorb more general axioms into an unfoldable TBox.

1 Introduction

For reasoning over a DL knowledge base, general axioms are internalized by causing nondeterministic disjunctions added to every node during a tableaux expansion. In this paper, general axioms are defined in $\mathcal{T}_g$, as can be seen in the division of a TBox at the beginning of Sect. 3. To resolve the nondeterminism introduced by general axioms in $\mathcal{T}_g$, absorptions can perform syntactic transformations on these axioms, and move them to an unfoldable TBox, which contains only axioms suitable for lazy unfolding ([HT00,Hor03]). Intuitively, an absorption technique tries to relocate general axioms from $\mathcal{T}_g$ to some unfoldable TBox $\mathcal{T}_u$. This rewriting technique is widely employed in DL reasoning to enhance the performance of reasoning.

Although various absorption techniques have been presented in the literature ([Hor03,HW06,ZH06,THPS07]), few of them are universally applicable and effective alone. Some absorptions can be applied to almost all known ontologies, however their effectiveness deteriorates when ontologies become large and complicated, especially ontologies from biomedical domains. For this reason, a *best* absorption to resolve all general axioms is desirable, whereas the precise definition of a *best* absorption is still an open question. Some empirical studies suggested absorption techniques that are capable of rewriting as many general axioms as possible from $\mathcal{T}_g$ to $\mathcal{T}_u$ outperform the others ([HT00,THPS07]). In this paper, we consider the possibility that not every general axiom can be absorbed for all knowledge bases, due to the characteristics of the domains to be modeled. To our knowledge, one typical example is the BCS 5 ontology ([ABdR99]) modeling feature interaction in the telecommunication domain, whose general axioms always fail any currently known absorption technique.

Instead of inventing a new absorption technique, this paper presents research to apply existing absorptions to ontologies in a fashion that depends on the structures of these ontologies. The basic idea is as follows. Initially some features of the input ontology affecting its absorptions are put into a cost analysis. A planning system then presents corresponding strategies on the categories and the sequence of absorptions to be applied. Finally, among these possibilities a strategy with the minimal estimated cost is selected.

Section 2 presents some background knowledge about planning. Following that Sect. 3 will describe those absorptions selected as operators. Section 4 presents an overview of the planning procedure and the chosen metrics to approximate costs. Empirical evaluations are discussed in Sect. 5. Section 6 summarizes the paper with a conclusion and discussion.

2 Preliminaries

Description logics (DL) is a family of well-studied decidable subset of First Order Logic (FOL). We present a brief introduction to the syntax, semantics and reasoning services to DL $\mathcal{ALCI}$.

Syntax Let A be a concept name, C, D be any concept, R be some role. R^- is an inverse role of R. The set of concepts is defined as follows.
$C := \top \mid \bot \mid A \mid \neg A \mid C \sqcap C \mid C \sqcup C \mid \forall R.C \mid \exists R.C$

Semantics An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of the domain of $\mathcal{I}$, a non-empty set $\Delta^{\mathcal{I}}$, and a mapping function $\cdot^{\mathcal{I}}$. The mapping function $\cdot^{\mathcal{I}}$ maps every role to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ s.t. $\langle x, y \rangle \in R^{\mathcal{I}}$ iff $\langle y, x \rangle \in R^{-^{\mathcal{I}}}$; it maps, additionally, every concept to a subset of $\Delta^{\mathcal{I}}$ s.t.
$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$, $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$, $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$, $(\exists R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(x, C) \neq \emptyset\}$, $(\forall R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(x, \neg C) = \emptyset\}$.

Inference $C \sqsubseteq D$ is a general concept inclusion (GCI), and a set of GCIs forms a TBox. An interpretation $\mathcal{I}$ satisfies a GCI $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. $\mathcal{I}$ satisfies a TBox $\mathcal{T}$ if it satisfies every GCI in $\mathcal{T}$, and such an interpretation is called a model of $\mathcal{T}$. C is *satisfiable* w.r.t. $\mathcal{T}$ if there exists a model $\mathcal{I}$ of $\mathcal{T}$ with $C^{\mathcal{I}} \neq \emptyset$. D *subsumes* C w.r.t. $\mathcal{T}$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ holds in every model of $\mathcal{T}$. $\mathcal{T}$ is consistent if there is a model of $\mathcal{T}$.

This paper employs the classical representation scheme to describe planning problems. Formal definitions are adapted from [GNT04]. The classical planning requires several restrictive assumptions. Particularly, a classical planner does an *offline planning* in the sense that the planner ignores any dynamic that may occur in the state-transition system Σ during the planning.

A planning *operator* can be represented as a triple $O = \langle \texttt{Name}, \texttt{Prec}, \texttt{Effc} \rangle$, where `Name` is a name for this operator, and `Prec` and `Effc` are sets of literals (i.e. atoms and negations of atoms of the representation language L) to generalize the preconditions and effects respectively.

A classical *planning domain* is a state-transition system $\Sigma = \langle \mathtt{S}, \mathtt{A}, \gamma \rangle$, where $\mathtt{S}$ is a finite set of states (a state s contains a set of *ground* atoms), $\mathtt{A}$ consists of all actions, i.e. *ground* instances of operators in O, and γ is a state-transition function which changes one state to another upon the application of some action. $\mathtt{S}$ is closed under γ. A classical *planning problem* is described as $P = \langle \Sigma, s_0, g \rangle$, where Σ is the planning domain, s_0 is the initial state, and g is the goal. A *plan* is a sequence of actions $\pi = \langle a_1, \ldots, a_i \rangle$, where $i \geq 0$, $\forall$ i, $a_i \in \mathtt{A}$. A plan π is a *solution* for a planning problem P if the goal g can be reached from s_0 along π.

3 From Absorptions to Operators

The intuition is that every absorption technique is a planning operator. In this section, we study and adapt some existing absorption techniques to our specific planning problem. *Basic absorption* (Sect. 3.1), *conjunctive absorption* (Sect. 3.2), and *inverse role based absorption* (Sect. 3.3) are presented, and a formulation of operators ends this section.

The following definitions and notations are used throughout the paper. A, B and A_i ($1 \leq i \leq n$ for some integer n) denote concept names, C and D are arbitrary concept expressions, and R denotes some role. A TBox $\mathcal{T}$ is divided such that $\mathcal{T} \equiv \mathcal{T}_u^A \cup \mathcal{T}_u^{\neg A} \cup \mathcal{T}_u^{\sqcap} \cup \mathcal{T}_e \cup \mathcal{T}_g$, where $\mathcal{T}_u^A$ and $\mathcal{T}_u^{\neg A}$ contain axioms of the forms $A \sqsubseteq C$ and $\neg A \sqsubseteq C$ respectively, $\mathcal{T}_u^{\sqcap}$ consists of axioms in the form $A_1 \sqcap \ldots \sqcap A_i \sqsubseteq C$ where $i \geq 2$, $\mathcal{T}_e$ is composed of definitional axioms of the form $A \equiv C$, and $\mathcal{T}_g$ consists of all other axioms, i.e. the remaining general axioms.

3.1 Basic Absorption

We first refer readers to [Hor03] for fundamental absorptions. A general axiom has options to be absorbed as a definitional axiom of the form $A \sqsubseteq C$, or $\neg A \sqsubseteq C$, or $A \equiv C$ ([HT00]). In our paper, three forms of axioms are admissible in unfoldable TBoxes: $A \sqsubseteq C$, $A \equiv C$, and $\neg A \sqsubseteq C$. However, we impose a condition on above fundamental absorptions: absorptions cannot define both $A \sqsubseteq C$ and $\neg A \sqsubseteq D$ in $\mathcal{T}_u^A$ and $\mathcal{T}_u^{\neg A}$ resp. unless these two axioms can form a definition as $A \equiv C$. Although it is possible to relax this condition by applying resolution-based absorption as given in [ZH06], that absorption, breaking the offline restriction, could hardly be formulated in a classical planner. An absorption complying with these restrictions is named *basic absorption* in this paper.

3.2 Conjunctive Absorption

Conjunctive absorption has been derived from the binary absorption ([HW06]), which was motivated to utilize axioms of the form $A_1 \sqcap A_2 \sqsubseteq C$ where only concept names are allowed on the LHS. In [HW06], new concept names will be introduced when necessary in order to take advantage of binary absorptions. However, new concepts should be avoided in the classical planning to meet the requirement of offline planning.

Faithfully following [HW06], we have extended binary absorption to conjunctive absorption. [HW06] demonstrated that the binary absorption is an elegant form of n-ary absorption, we directly use an absorption to utilize axioms of the form $A_1 \sqcap A_2 \sqcap \ldots \sqcap A_i \sqsubseteq C$ (a formal proof can be easily derived from Lemma 4.1 in [HW06]). Notably, the concept expression on the left-hand side (LHS) of these axioms is in a conjunctive form, which provides this absorption with the name *conjunctive absorption*. Further, we assume that not all the negations of these conjuncts on the LHS are defined in $\mathcal{T}_u^{\neg A}$, otherwise supplementary measures have to be taken to ensure the correctness of the absorption.

Conjunctive absorption, in contrast to binary absorption, does not add new concepts to the classical planning domain. Additionally, a tableau expansion rule is required to unfold axioms in $\mathcal{T}_u^{\sqcap}$. A similar rule is needed to deal with axioms absorbed by binary absorptions in [HW06]. Our experiences indicated that in tableau reasoning the former rule is not necessarily more expensive than the latter. The choice of a better one between the conjunctive absorption and the binary absorption largely depends on characteristics of axioms of the input ontology. From another perspective, both conjunctive absorption and binary absorption resemble special forms of the basic absorption that disallows negated named concepts on the LHS. Intuitively, any general axiom that can be absorbed by conjunctive or binary absorptions is a candidate for basic absorption as well. This viewpoint indicates that the conjunctive absorption actually cannot absorb more general axioms than the basic absorption, but the former has the advantage of reducing the number of disjunctions in a more straightforward way.

3.3 Inverse Role Based Absorption

In the absorption framework presented in [HW06], an auxiliary step of applying transformations to concept expressions having inverse roles was shown. Later, [DHW07] formally proved that the elimination of inverse roles is feasible via a similar technique.

The basic transformation rule presented in both papers is as follows: $\top \sqsubseteq C \sqcup \forall R.D$ is equivalent to $\top \sqsubseteq D \sqcup \forall R^-.C$ where R^- is an inverse relation of R. Observe that the inverse role elimination approach itself does not move a general axiom from a $\mathcal{T}_g$ to a $\mathcal{T}_u$, that is, it is not a typical absorption technique.

In our implementation, the inverse role elimination approach has been extended to a direct absorption technique. Following the above transformation rule, $\top \sqsubseteq C \sqcup \forall R.D$ is directly absorbed into $\neg D \sqsubseteq \forall R^-.C$ in $\mathcal{T}_u^{\neg A}$ when a (negated) concept name can be extracted[1] from D. An observation is that the qualification of a universal restriction may play the same role as a named concept. Consequently, the capability of extracting qualifications that are (negated) named concepts in universal restrictions is the key for inverse role based absorptions. This absorption slightly differs from the processing step in [HW06] because no new concept names are necessary (and thus can be formulated in a classical

[1] Either D is a (negated) concept name or recursive extractions of D can reveal some (negated) named concept. See literal `extract` in InverseRoleAbs in Sect. 3.4.

planner). Practically, preference can be given to extracted qualifications that are negated concept names when the inverse role transformation is applicable more than once to some general axiom.

3.4 Operator Formulation

The literals to define the operators are self-explanatory. Due to lack of space, a simplified formulation of absorptions is given below.

BasicAbs1 move a general axiom gci from $\mathcal{T}_g$ to $\mathcal{T}_u^A$
Prec: hasDisjunct(gci,$\neg A$); $\neg$definedIn(A,$\mathcal{T}_e$); $\neg$definedIn($\neg A$,$\mathcal{T}_u^{\neg A}$)
Effc: definedIn(A,$\mathcal{T}_u^A$)

BasicAbs2 move a general axiom gci from $\mathcal{T}_g$ to $\mathcal{T}_u^{\neg A}$
Prec: hasDisjunct(gci,A); $\neg$definedIn(A,$\mathcal{T}_e$); $\neg$definedIn(A,$\mathcal{T}_u^A$);
when definedIn($A \sqcap A_1 \sqcap \ldots \sqcap A_n, \mathcal{T}_u^{\sqcap}$) *then* $\exists$ i: $\neg$definedIn($\neg A_i, \mathcal{T}_u^{\neg A}$), $1 \leq i \leq n$
Effc: definedIn($\neg A$,$\mathcal{T}_u^{\neg A}$)

ConjunctiveAbs move a general axiom gci from $\mathcal{T}_g$ to $\mathcal{T}_u^{\sqcap}$
Prec: hasDisjunct(gci,$\neg A_1$); ... ; hasDisjunct(gci,$\neg A_n$) where $n \geq 2$;
$\forall$i, $\neg$definedIn(A_i,$\mathcal{T}_e$), $1 \leq i \leq n$; $\exists$j, $\neg$definedIn($\neg A_j$,$\mathcal{T}_u^{\neg A}$), $1 \leq j \leq n$
Effc: definedIn($A_1 \sqcap \ldots \sqcap A_n, \mathcal{T}_u^{\sqcap}$)

InverseRoleAbs move a general axiom gci from $\mathcal{T}_g$ to $\mathcal{T}_u^A$[*or* $\mathcal{T}_u^{\neg A}$]
Prec: hasDisjunct(gci,$\forall R.C$); extract($\forall R.C$, A[or $\neg A$]); $\neg$definedIn(A,$\mathcal{T}_e$);
$\neg$definedIn(A[or $\neg A$],$\mathcal{T}_u^A$[or $\mathcal{T}_u^{\neg A}$])
Effc: definedIn($\neg A$[or A],$\mathcal{T}_u^{\neg A}$[or $\mathcal{T}_u^A$])

4 Applying Planning to Absorptions

Before devising a new planner, we decided to study known AI planners to analyze their feasibility and scalability in the DL domain. One of the most competitive planners, SGPlan 5[2], is selected. We first constructed a simple TBox with seven axioms, all of which were represented in PDDL 3.0 ([GL05]). Then certain copies of these axioms are replicated to increase the size of the problem. The result showed that a replication of 60 copies (180 general axioms) could substantially lead to a failure of this planner. From our experience, sophisticated classical planners do not scale up well in the DL domain because a TBox with hundreds or even thousands of general axioms is not uncommon. To avoid the issue on scalability, we implemented our own planner as described in the next section.

4.1 Planner Architecture

Initial State and the Goal. In the state-transition system, a state is a collection of ground atoms, for example $s_1 = \{$definedIn(A,$\mathcal{T}_e$)$\}$ is a typical state.

[2] The winner of IPC2006, see http://manip.crhc.uiuc.edu/programs/SGPlan/

Notice that negative literals are prohibited in any state because the closed-world assumption is usually used, i.e. literals, unless explicitly defined positively in a state, are considered negative in that state.

An *initial state* contains all the necessary ground atoms to express the relevant information on the TBox and general axioms. The ideal goal in our case should be to absorb all general axioms for any ontology. However, as claimed from the very beginning, we are aware that such a plan does not always exist, more precisely, there exist ontologies lacking feasible plans to absorb all their general axioms. To bypass this problem, the *goal* is set to allow *a very small percentage of* total general axioms not absorbed in $\mathcal{T}_g$.

Group Planning. As stated at the beginning of Sect. 4, to deal with large planning domains, i.e. ontologies with a large $\mathcal{T}_g$, the planner was implemented in an unconventional manner. During the planning, an action, i.e., a ground instance of an operator, is applied to a group of candidate general axioms rather than a single one, which can cause more than one general axioms to be absorbed. An example is provided in Sect. 4.3. In contrast, a planner whose actions only work on single general axiom during a transition may be more effective, but it is not easily scalable in practice.

Planning Procedure. When an ontology is fed into the planning system, it is preprocessed to initialize the initial state. Any operator, if its preconditions are met in this state, will be applied to the current state. Consequently, another state will be created or selected by a state-transition function. On the one hand, any of the operators, if successfully applied to some general axiom, is contributing to a solution. On the other hand, such an operator, as implied from the formulation of operators, is not necessarily the sole operator applicable to this general axiom. For example, if the conjunctive absorption is applicable to some general axiom, then this general axiom can be transformed by the basic absorption as well. State-selection heuristics can then be used to resolve possible nondeterministic choices, as shown in Sect. 4.4. Additionally, cost metrics are presented in Sect. 4.2 to approximate an optimal solution.

4.2 Cost Metrics

Assuming that some operator is applied to a typical general axiom, the costs of both the LHS and the right-hand side (RHS) of an absorbed axiom usually have to be estimated differently. Thus, the cost metrics are presented in two parts. Observe that a reasonable approximation of costs in some state may greatly improve the quality of state-selections. Technically, one thorough cost analysis of an axiom will result in a numerical value identifying the *approximated cost* of that axiom.

Cost Analysis on the LHS. The principle of the cost analysis on the LHS, roughly speaking, is to assess the gains and byproducts after applying an action instead of other actions. This action, which is associated with some sequent states in general, is called an **associated action** w.r.t these states. For instance, if an application of the basic absorption absorbs a general axiom to B in $\mathcal{T}_u^A$, then one of the gains is that this general axiom is resolved, however one of its negative impacts is that other general axioms lose the chances to be absorbed to $\neg B$ in $\mathcal{T}_u^{\neg A}$ according to the assumption presented in Sect. 3.1. We now show how to estimate costs for absorbed axioms with different concepts on the LHS.

- Concept A on the LHS: General axioms that have been successfully absorbed by the associated action are called **processed** (general) axioms. Axioms generated from these processed axioms define A in $\mathcal{T}_u^A$. At the same time, some general axioms are blocked by the associated action, i.e. they cannot be absorbed to axioms of the form $\neg A \sqsubseteq C$ by other actions due to the restriction discussed in Sect. 3.1. These general axioms are called **unprocessed** (general) axioms. Formally the cost in this category is represented by the product of an assigned weight and the difference between the number of processed and unprocessed axioms.
- Concept $\neg A$ on the LHS: In this case, processed axioms have $\neg A$ defined in $\mathcal{T}_u^{\neg A}$, which block the following two kinds of unprocessed axioms. First, general axioms which can define A in $\mathcal{T}_u^A$ are blocked. Second, some general axioms that can be conjunctively absorbed may be blocked as well. Considering a general axiom that can define $A_1 \sqcap A_2$ in $\mathcal{T}_u^{\sqcap}$, if $\neg A_1$ is already defined in $\mathcal{T}_u^{\neg A}$, then BasicAbs2 in Sect. 3.4 keeps this general axiom from being conjunctively absorbed. As a result, this general axiom is blocked too. The cost is computed in the similar manner as above.
- Concept $A_1 \sqcap \ldots \sqcap A_n$ on the LHS: Suppose there are n general axioms GCI_i ($1 \leq i \leq n$) to define $\neg A_1$ to $\neg A_n$ in $\mathcal{T}_u^{\neg A}$. If some other general axiom will be absorbed to $A_1 \sqcap \ldots \sqcap A_n$ in $\mathcal{T}_u^{\sqcap}$, then at least one of GCI_i needs to be blocked and becomes an unprocessed axiom. The probability to block such an unprocessed general axiom is also considered in the cost estimation.

Cost Analysis on the RHS. Suppose an associated action absorbs some general axiom to a new axiom, the following two properties are studied on the RHS of this new axiom to estimate how "difficult" it could be.

- The number of disjunctions. Disjunctions are one of the main concerns of optimizations. Generally, the fewer disjunctions on the RHS of an axiom, the fewer branchings in tableau expansion. Thus, the number of disjunctions left on the RHS of generated axioms is used to examine the associated action, i.e. some absorption.
- Saturation of concepts. Considering an axiom generated by an absorption with A on the RHS, A may have several super-concepts that are eventually introduced during unfolding. To terminate the cyclic introduction of A itself, a procedure called **saturation** of A is introduced. The cost on the RHS is determined after all concept names have been saturated.

Apart from these cost metrics on a specific axiom after absorption, estimated costs are attached to absorptions themselves. Every absorption has a different cost to apply. For example, the conjunctive absorption is assigned a high cost when applied because it generates axioms to be dealt with by a relatively expensive tableau expansion rule.

4.3 Example

This section gives an example to show what is group planning (Sect. 4.1) and how to calculate part of the costs in state s_2 w.r.t an associated action (BasicAbs1).

(1)$\top \sqsubseteq A_1 \sqcup \neg A_2$	$A_2 \sqsubseteq A_1$ (*processed axiom*)
(2)$\top \sqsubseteq A_1 \sqcup A_3$	$\top \sqsubseteq A_1 \sqcup A_3$
(3)$\top \sqsubseteq A_1 \sqcup A_2 \sqcup A_3$	$\top \sqsubseteq A_1 \sqcup A_2 \sqcup A_3$ (*unprocessed axiom*)
(4)$\top \sqsubseteq \neg A_1 \sqcup \neg A_2 \sqcup \neg A_3$	$A_2 \sqsubseteq \neg A_1 \sqcup \neg A_3$ (*processed axiom*)
state s_1	state s_2

Suppose that the whole TBox has only four general axioms as shown in state s_1. Then, an action instantiated with BasicAbs1 is applied to general axioms in this state. Assuming that this associated action will absorb all general axioms that have $\neg A_2$ as a disjunct on the RHS, i.e. (1) and (4). Observe that the associated action absorbs the *group* rather than only one of these candidate axioms. After the application of this action, both (2) and (3) remain unabsorbed in s_2, but only axiom (3) is an unprocessed general axiom since it loses the opportunity to be absorbed to $\neg A_2$ due to this action while axiom (2) is never affected. Notice that axiom (3) may still be absorbed to other concepts or by other operators.

4.4 Search Algorithm

Classical planning algorithms are either based on searching state space or plan space. Although plan-space planning, for a while, outperformed state-space planning, the former ignores the notion of explicit states along the plan, as a result, domain-specific heuristics are difficult to apply ([GNT04]). Regarding our planning domain, a state-space planning algorithm is chosen to allow efficient use of heuristics.

The search algorithm in our implementation is A^*, one of the best-established forward search algorithms. The cost along one plan is calculated on the generated axioms using the cost metrics described in Sect. 4.2, and the heuristic $h(n)$ we used is the number of remaining general axioms in node n.

5 Empirical Studies

The aforementioned planning system was implemented in a prototype tableau based reasoner with a very limited number of implemented optimization techniques. Our reasoner, implemented in a straightforward way, contains several preprocessors and a planning system. The preprocessors provide some common optimization techniques including simplification and normalization. This reasoner is limited to satisfiability testing on ontologies of the expressivity $\mathcal{ALCI}$. Other than that, a special rule to expand axioms in $\mathcal{T}_u^{\sqcap}$ has been added. Since few reasoning optimizations were implemented, the reasoner is not comparable to any existing popular reasoner in terms of reasoning services or efficiency. The planner will be activated after some necessary preprocessing.

The planner has been tested with several ontologies on a standard PC: Dual Core (2.40 GHz) Pentium processor and 3 GB of physical memory. The DM ontologies ([BDTW07]) are a series of bounded model checking ontologies. There are two digits in the name, where the first one shows the size of the model in terms of a "cell" that includes 17 state-variables, i.e. "5 cells" indicates 85 state-variables, and the second digit stands for the the bound. The larger the size and the bound, the harder the ontology.

We compared the absorption effectiveness and reasoning performance when the planner is turned on or off. When the planner is turned off, basic absorptions (BasicAbs1, BasicAbs2) will be triggered. For experiments on DM ontologies, a timeout (TO) of 1 000 seconds was used. Times (in seconds) given in Table 1 are the average of five independent runs.

Table 1. Coherence Check Times /# of general axioms *not* absorbed

KB Name	DM5-5	DM5-10	DM5-15	DM16-5	DM16-10
Planner On	0.14/0	0.17/0	0.27/0	1.89/0	2.92/0
Planner Off	14.3/110	16.3/110	16.5/110	TO/352	TO/325

In the preprocessing phase, all general axioms are normalized to disjunctive normal form, thus the number of general axioms is counted in a way that may differ from other reasoners. The original DM ontologies contain more than 500 general axioms. When the planner was not used, basic absorptions could not absorb all general axioms. Moreover, we also tried conjunctive absorptions (ConjunctiveAbs) in this case, yet more general axioms remained. On the contrary, general axioms can be completely absorbed when the planner is used.

The runtime performance shown in Table 1 may suggest that the planning method is not competitive. However, the reasoning performance is not completely determined by the number of general axioms. The reasoning optimization techniques adapted as well as other factors may greatly affect runtime performances of a reasoner from one ontology to another. When this evolving field produces ontologies with a considerable amount of general axioms, we expect that general axioms may become one of the dominant factors for reasoning performance.

6 Conclusion and Discussion

We have presented a prototype of a planning system for applying absorption techniques in DL. Compared to applying absorptions in an almost fixed and predetermined way to all ontologies, our planner organizes present absorptions in a more reasonable way for different ontologies and tends to be more adaptable and effective. While planning is feasible, we also revealed some limitations.

Offline planning, as shown in Sect. 2, is the prime limitation for the classical planning. In this case, a planner plans for the given initial and goal states regardless of current dynamics, for example creation of new objects. An online planning may relax this assumption, but it is difficult to deal with. Alternatively, we restrained the operators to comply with this offline planning requirement, for instance we adapted the inverse role based absorption rather than the inverse role elimination. Further, binary absorptions, as given in [HW06], introduce new concepts when necessary, and can not be dealt with by restrictive classical planners. Instead, we formulated the conjunctive absorption as an operator. Actually, the first kind of normal forms presented in [BLS06] has the identical conjunctive form on the LHS of an axiom with a special completion rule for the reasoner CEL.

Basic absorptions, if no distinctions are made between cases where concept names or their negations appear on the LHS, could be extended to allow heuristic choices. Similar to what was given in [ZH06], concepts and their negations are sorted according to their number of occurrences in all general axioms. Suppose there is some general axiom that can be absorbed to either A or $\neg A$. Now the heuristic is used to select the most promising candidate (negated) concept name. Namely, if $\neg A$ occurs more frequently than A does, priority is given to $\neg A$ for the basic absorptions so as to block less general axioms probably. On the contrary, our implementation distinguishes between two basic absorptions so that the heuristic on occurrence is already generalized in the planning itself.

Treatment for definitional equivalence axioms of the form $A \equiv C$ is at the discretion of the designer, for example, these axioms can be directly moved to $\mathcal{T}_e$. Further preprocessing on them is also acceptable. Equivalence axioms could be reverted to two general axioms, i.e., $A \sqsubseteq C$ and $C \sqsubseteq A$ to be considered by the planner. The planner, in turn, could be equipped with an additional operator that pieces candidate general axioms together to form definitional equivalences. It is not difficult to observe that such an operator proceeds in the same manner as the resolution-based absorption with the exception that trivial general axioms of the form $\top \sqsubseteq C \sqcup \neg C$ that would have been introduced are omitted. Our previous experiments implemented both treatment, i.e., directly move equivalences to $\mathcal{T}_e$ or introduce an extra operator, do not evidence that one is more advantageous than the other for our test ontologies.

Other absorption techniques, though practically tested, are not discussed in this paper. The resolution-based absorption ([ZH06]) and the domain and range absorption ([THPS07,HM01]) are part of our implementation. The former brings in new general axioms using an approach resembling resolution techniques, which violates the offline restriction of the classical planning. For the domain absorption, the inverse role based absorption can act as a substitute. Consequently, both of them are not covered in this planner.

To refine a planner, cost metrics deserve attention as well. Cost metrics for generated axioms only estimate the real cost. The intuition of a cost estimation is to map as many features of various ontologies as possible. However, a full mapping of a TBox could result in complicated planning. Thus, how to balance the granularity of cost mapping and the performance gain needs to be further investigated.

Acknowledgments. We would like to thank Chih-Wei Hsu, the author of SG-Plan 5, for his insightful discussions and Francis Gasse for his contribution to the DL reasoner. We also thank anonymous reviewers for their useful feedback.

References

[ABdR99] Carlos Areces, Wiet Bouma, and Maarten de Rijke. Feature interaction as a satisfiability problem. In *Proceedings of* MASCOTS'99, pages 339–346, October 1999.

[BDTW07] Shoham Ben-David, Richard J. Trefler, and Grant Weddell. Bounded model checking with description logic reasoning. In *TABLEAUX*, pages 60–72, 2007.

[BLS06] F. Baader, C. Lutz, and B. Suntisrivaraporn. CEL—a polynomial-time reasoner for life science ontologies. In U. Furbach and N. Shankar, editors, *IJCAR'06*, volume 4130 of *LNAI*, pages 287–291, 2006.

[DHW07] Yu Ding, Volker Haarslev, and Jiewen Wu. A new mapping from $\mathcal{ALCI}$ to $\mathcal{ALC}$. In *Proceedings of* DL2007, pages 53–64, Italy, 2007.

[GL05] Alfonso Gerevini and Derek Long. Plan constraints and preferences in PDDL3. Technichal Report, Department of Electronics for Automation, University of Brescia, Italy. August 2005.

[GNT04] Malik Ghallab, Dana Nau, and Paolo Traverso. *Automated Planning: Theory and Practice*. Morgan Kaufmann, 1st edition, 2004.

[HM01] Volker Haarslev and Ralf Möller. High performance reasoning with very large knowledge bases: A practical case study. In *Proceedings of* IJCAI'2001, pages 161–168, 2001.

[Hor03] Ian Horrocks. Implementation and optimization techniques. In *Description Logic Handbook*, pages 306–346, 2003.

[HT00] Ian Horrocks and Stephan Tobies. Reasoning with axioms: Theory and practice. In *Proceedings of KR'00*, San Francisco, CA, 2000.

[HW06] Alexander K. Hudek and Grant Weddell. Binary absorption in tableaux-based reasoning for description logics. In *Proceedings of* DL2006, UK, 2006.

[THPS07] Dmitry Tsarkov, Ian Horrocks, and Peter F. Patel-Schneider. Optimizing terminological reasoning for expressive description logics. *Journal of Automated Reasoning*, 39(3), 2007.

[ZH06] Ming Zuo and Volker Haarslev. High performance absorption algorithms for terminological reasoning. In *Proceedings of* DL2006, 2006.