Abduction Guided Query Relaxation

(Extended Abstract)

Samy Sá
Universidade Federal do Ceará
Estrada do Cedro, Km 5
Quixadá, Brazil
samy@ufc.br

João Alcântara
Universidade Federal do Ceará
Campus do Pici, Bl 910
Fortaleza, Brazil
jnando@lia.ufc.br

ABSTRACT

We investigate how to improve cooperative communication between agents by representing knowledge bases as logic programs extended with abduction. In this proposal, agents try to provide explanations whenever they fail to answer a question. Query Relaxation is then employed to search for answers related to the query, characterizing cooperative behavior. Our contributions bring insightful improvements to relaxation attempts and the quality of related answers. We introduce rational explanations and use them to efficiently guide the search for related answers in a relaxation tree.

Categories and Subject Descriptors

F.4.1 [**Mathematical Logic**]: Logic and constraint programming; I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems

General Terms

Theory

Keywords

Query Relaxation, Abductive Logic Programming

1. INTRODUCTION

Cooperative Answering [2, 4] is a form of cooperative behavior in deductive databases. When the answer to a query is not satisfactory (such as in case of failure), an effort is made to return related information. This behavior can be imported to agents to improve communication or coordination. Deductive databases are a special kind of logic programs, which is also the kind of knowledge bases we consider for agents in this paper. Relaxation is presented by Gaasterland in [2] as a method for expanding both deductive databases and logic programming queries. Just as well, logic programs are suitable to build intelligent agents and multiagents systems, especially as an account for automated reasoning. We defend that cooperative answering can be of great use to MAS so agents can exhibit cooperative behavior in any information sharing situation.

Abduction is a kind of non-monotonic reasoning, usually defined as a search for the best explanation. We resort to abduction to improve the search for answers when there is need for relaxation. In our approach, abduction is used to produce explanations to failure and pinpoint the conditions of the query that should be worked on to guide relaxation. The author of a query can also help to guide the process by naming important conditions. We employ these clues and abduction to help relaxation return answers as close as possible to what is expected by the author of the query.

The paper is organized as follows: Section 2 introduces abductive logic programs, queries and relaxations. Section 3 presents the concepts we use to guide the search, which is discussed in section 4. Section 5 concludes the paper.

2. BACKGROUND

2.1 Abductive Logic Programs

We consider Abductive Logic Programs (ALPs) as in the abductive framework of *Extended Abduction* from Sakama and Inoue [6]. An abductive program is a pair $\langle P, H\rangle$, where P is an Extended Disjunctive Program [3] and H is a set of literals referred to as abducibles. If a literal $L \in H$ has variables, then all ground instances of L are abducibles. If P is consistent (does not prove L and $\neg L$ simultaneously), then $\langle P, H\rangle$ is consistent. Unless we state otherwise, a program is consistent. A conjunction $G = L_1, \ldots, L_m, not\ L_{m+1}, \ldots, not\ L_n$ is range restricted if every variable in $L_{m+1}, \ldots, L_n$ is also in $L_1, \ldots, L_m$. An *observation* over $\langle P, H\rangle$ is a conjunction G with all variables existentially quantified and range restricted. $\langle P, H\rangle$ satisfies an observation if $\{L_1\theta, \ldots, L_m\theta\} \subseteq S$ and $\{L_{m+1}\theta, \ldots, L_n\theta\} \cap S = \emptyset$ for some substitution θ and some answer set S of P.

Definition 1. *Let G be an observation over the ALP $\langle P, H\rangle$. The pair (E, F) is an* explanation *of G in $\langle P, H\rangle$ if (i) $(P \setminus F) \cup E$ has an answer set which satisfies G[1]; (ii) $(P \setminus F) \cup E$ is consistent; and (iii) E and F are sets of ground literals such that $E \subseteq H \setminus P$ and $F \subseteq H \cap P$ [6].*

Intuitively, an explanation (E, F) means that by adding (considering) the literals in E while retracting (falsifying) the literals in F from P, the resulting P' satisfies G. An explanation (E, F) is minimal if, for any explanation (E', F') such that $E' \subseteq E$ and $F' \subseteq F$, then $E' = E$ and $F' = F$. In general, only the minimal explanations are of interest.

Cite as: Abduction Guided Query Relaxation (Extended Abstract), Sá, S. and Alcântara, J., *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2011)*, Tumer, Yolum, Sonenberg and Stone (eds.), May, 2–6, 2011, Taipei, Taiwan, pp. 1275-1276.

---

[1] This definition is for *credulous* explanations. Its choice over *skeptical* explanations [5] allows for more explanations and a better chance of finding good related answers to a query.

2.2 Query Relaxation

The process of query relaxation is introduced in [2] to allow for cooperative query answering in deductive databases. We consider the relaxation methods as defined in [6], since they are already oriented to use with ALPs.

Definition 2. *A query G is a question to a logic program and has the same definition as observations to an ALP. We write $Lit(G)$ to refer to the set of literals in a query G. These literals are the* conditions *of the query.*

Definition 3. *A query G can be relaxed to a query G' by any combination of the methods: (*i*) Anti-Instantiation: Given a substitution θ if $G'\theta = G$, then G' is a relaxation of G by anti-instantiation; (*ii*) Dropping Conditions: If G' is a query and $Lit(G') \subset Lit(G)$ then G' is a relaxation of G where the conditions of $Lit(G) \setminus Lit(G')$ were dropped; or (*iii*) Goal Replacement: If G is a conjunction G_1, G_2 and there is a rule $L \leftarrow G'_1$ in P such that $G'_1\theta = G_1$, then $G' = L\theta, G_2$ is a relaxation of G by goal replacement.*

3. GUIDING QUERY RELAXATION

3.1 Useful Literals

A literal is useful towards relaxation if an explanation suggests a query relaxation that replaces it can succeed. Given the successful results of a query in $P' = (P \setminus F) \cup E$, the conditions satisfied by P' that are not satisfied by P are considered useful towards relaxation according to (E, F). $U_{E,F}(G)$ is the set of useful literals of G according to (E, F).

3.2 Query Author's Choice

Definition 4. *A restricted query is a pair (G, B) such that $B \subseteq Lit(G)$ and G is a query (as before).*

The set B contains the literals of G specified as the most important by the query author. These literals are treated as non-abducibles and are not replaced in relaxation attempts, so any related answers provided satisfy the conditions in B.

3.3 Rational Explanations

A substitution θ' such that no literals in $Lit(G\theta')$ are satisfied by P suggests all literals as useful. Any relaxation attempts based on such explanations will likely produce answers far from those expected or also lead to failure.

Definition 5. *An explanation (E, F) is a rational explanation iff $|Lit(G)| - |U_{E,F}(G)| \geq 1$. Otherwise, it is said to be a non-rational explanation.*

In case all possible relaxations of a query also fail, it is possible to still have explanations, but only non rational. We restrict relaxation attempts to those based on rational explanations and improve the quality of related results.

4. RESTRICTING THE SEARCH

Given an explanation (E, F) and query G, the search for related answers of G is restricted to those relaxations where at least one useful literal of G is replaced. For instance, dropping a condition (a literal) that is not an useful literal will not help satisfying the query (according to (E, F)). The same goes for all methods described in definition 3.

An explanation to the failure of a query G means it can be made consistent (not to fail) with the program P. For this reason, any rational explanation can guide relaxation, as it would suffice to drop all the conditions it suggests as useful. In order to retrieve answers as close as possible to those that would satisfy G, we should consider criteria to select the best explanations to guide relaxation.

4.1 The Best Explanations

Some explanations are better than others. For instance, some minimal explanations are related to the instances of G that P is the closest to satisfying, and, consequently, to the Maximal Succeeding Subqueries (MSS) of G [4]. However, MSS only consider the number of conditions satisfied. As a consequence, amongst the explanations related to MSS, some might require less changes to P than others. The explanations that require the lesser adaptation of P make the best candidates to guide relaxation. This way of qualifying explanations, resemble the Best-Small Plausibility Criterion [1]: more plausible explanations are better (less useful literals), but in case of two explanations of same plausibility, the smallest (less changes to P) should be preferred. The best explanations according to such criteria should lead relaxation to good neighborhood answers of a query.

5. CONCLUSION AND FUTURE WORK

Our work presents and discusses a novel approach to improve cooperative communication in multiagent systems. We employ query relaxation and focus the search for related answers on attempts supported by abductive reasoning with clues from the query author. We also also discuss how an explanation can be better than others. The best explanations are related to results to which the query fails minimally and that would require the less changes to the program. The results retrieved by relaxations based on this kind of explanations are the most likely to be useful to the query author. As for future work, we intend to expand this approach to deal with the case where the query succeeds, but the answer is not satisfactory. We also intend to investigate how this approach can improve group decision situations.

6. ACKNOWLEDGEMENTS

Research partially supported by CAPES (PROCAD).

7. REFERENCES

[1] T. Bylander, D. Allemang, M. C. Tanner, and J. R. Josephson. The computational complexity of abduction, 1991.

[2] T. Gaasterland, P. Godfrey, and J. Minker. Relaxation as a platform for cooperative answering. *J. Intell. Inf. Syst.*, 1(3/4):293–321, 1992.

[3] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Comput.*, 9(3/4):365–386, 1991.

[4] P. Godfrey. Minimization in cooperative response to failing database queries. *International Journal of Cooperative Information Systems*, 6:95–149, 1997.

[5] K. Inoue and C. Sakama. Abductive framework for nonmonotonic theory change. In *IJCAI*, page 204, 1995.

[6] C. Sakama and K. Inoue. Negotiation by abduction and relaxation. In *AAMAS*, page 242, 2007.