Research in Automated Reasoning

Alvaro del Val
E.T.S. Informática
Universidad Autónoma de Madrid
28049 Madrid
delval@ii.uam.es
http://www.ii.uam.es/∼delval

1 Introduction

A large part of Artificial Intelligence (AI) is based on the idea that the best way for a computer to produce intelligent behavior in many application domains is by explicitly encoding knowledge about the domain in some "knowledge base" KB, and then using reasoning algorithms to extract consequences from that knowledge. Knowledge representation deals with the issue of defining languages suitable for various domains and reasoning tasks; automated reasoning, with the design and implementation of efficient reasoning algorithms for these languages and tasks. Much of the work in this area deals with logical or probabilistic languages, including e.g. propositional and first order logic, constraint satisfaction, Bayes nets, Markov networks, etc. Automated reasoning can offer a substantial number of recent successes in applications: NASA's unmanned space missions, controlled by a propositional system with capabilities for diagnosis, repair and automated planning [28, 24]; applications of search procedures for constraint satisfaction problems for scheduling, production planning, telecommunications, logistics and hardware design and verification [19, 25]; advances in theorem proving in e.g. quasigroup completion, whose applications go from answering open problems in mathematics to drug experiment scheduling [22]; and applications in automated planning to problems much larger than it was ever possible before [21].

These successes illustrate that great progress has been made in scaling up available techniques to deal with larger and larger problems, which have now already reached real-world applications. Nevertheless, most interesting AI problems require exponential time algorithms. We have to find ways to make problems tractable, i.e. polynomial time. Here are a few in which I have been involved:

2 Knowledge compilation

A knowledge base KB is a store of useful, reusable knowledge. It is only to be expected that we will query the KB very often. In order to obtain much better query answering times you can *compile* KB to yield a compiled KB* such that reasoning is tractable relative to KB*. Knowledge compilation (KC), as it is called, can incur a significant cost, but this cost can be amortized over multiple queries, which are now guaranteed to take time polynomial in the size of KB*. This is similar to compiling a program: the preprocessing to obtain a binary image of the source program gives you much better running time when you actually use the compiled program. The challenge is to minimize the size of KB* and compilation time, while still ensuring that every query of interest can be answered in polynomial time.

KC may be exact or approximate, depending on whether all queries can be answered tractably, or only a subset thereof (see next section). I have made various contributions to approximate KC [7, 8] and exact KC [6, 11]. Current work is focused on scaling up both types of KC by developing new, more restrictive algorithms. My work on KC is surveyed in e.g. [1, 2].

3 Deduction with restricted query languages

The deductive task of finding consequences of KB can be made much more efficient when a well-defined restricted query language is used, as search effort can focus only on the desired subset of consequences of KB. Deduction focused on restricted sets of consequences has many applications in AI, ranging from plain query answering to a variety of tasks needed for automating common sense reasoning (see below, and also the survey [23]). This can be formally characterized in terms of *approximate KBs*, which are supposed to improve reasoning over the restricted language [7, 8, 26]. In [7, 8], and more recently in [11], I provide efficient procedures for restricted consequence-finding. The latter paper introduces kernel resolution, which despite its novelty is already treated in depth in a very recent survey [23]. Recent work [13] includes the characterization of the complexity of restricted consequence-finding, with applications such as identifying tractable "abduction" problems (finding hypothesis that could explain facts given KB).

Kernel resolution also has applications in KC. It unifies approximate and exact KC, permits incremental compilation, and allows us to focus compilation on restricted languages, which may be used to obtain tractable commom-sense reasoning through compilation, or to ensure that the compiled KB* has polynomial size.

4 Tractable satisfiability and deduction

Much work in propositional and first order reasoning deals with the problem of consistency (so called satisfiability) of the knowledge base KB, as logical consequence can be reduced to satisfiability. I'm also interested in satisfiability methods and more generally constraint satisfaction techniques. I have identified tractable classes of satisfiability problems based on the structure of the KB [15], and developed new, more efficient algorithms for some well-known tractable classes [12].

I also have results on mapping tractable satisfiability classes to tractable deduction classes, using a general notion of polynomial refutation completeness. In [14] I show how this can be used to define a hierarchy of increasingly complex but always tractable deduction problems.

5 Common sense reasoning

Much of what passes as automated common-sense reasoning in AI involves some sort of generation of explanations or abduction [23, 20]. Basically, common-sense reasoning can often be viewed as reasoning with assumptions, often more technically phrased as default or non-monotonic reasoning. Assumption-based reasoning can in turn be formulated as consequence-finding. I examine some aspects of abduction in [14, 13], identifying tractable abduction problems; current work involves extending these results to various forms of default reasoning, again identifying polynomial classes. Note that these problems are much harder computationally than plain classical reasoning, hence the identification of easy cases is specially relevant.

This is also a good place to mention that much of my less recent work involved knowledge representation issues and algorithms for common-sense reasoning tasks such as updating a KB that represents a dynamically changing world and thus must deal with temporal information (see e.g. [17] and related papers), and revising a KB in the face of new information that contradicts previous knowledge [9, 18, 5]. Work on implementation of these reasoning strategies can be found in [3, 4]. More recent work on knowledge representation involve qualitative reasoning about perception and belief [16, 27], and connections of AI with Psychology [10].

6 Applications

As said, my research to date focuses mostly on basic research issues in AI. Nevertheless, I'm interested in applications of search techniques developed for logical reasoning or constraint programming to the wide variety of problems to which it has been applied, some of which are mentioned in the introduction. I believe in having real-world problems as main drivers of research. So, if you have an interesting application in logistics, planning, etc., feel free to ask me about possible collaborations. The web pages for my two graduate courses[1] can also give you an idea of the technologies I would be interested in deploying for real applications.

References

[1] M. Cadoli. Panel on knowledge compilation and approximation: Terminology, questions, references. In *Proc. 4th Int. Symp. on Artificial Intelligence and Mathematics*, 1996.

[2] M. Cadoli and F. M. Donini. A survey on knowledge compilation. *AI Communications*, 10:137–150, 1997.

[3] A. del Val. Computing knowledge base updates. In *KR'92, Proc. 3rd Int. Conf. on Knowledge Representation and Reasoning*, pp. 740–750. Morgan Kaufmann, 1992.

[4] A. del Val. Syntactic characterizations of belief change operators. In *IJCAI'93, Proc. 13th Int. Joint Conf. on Artificial Intelligence*, pp. 540–545, 1993.

[5] A. del Val. On the relation between the coherence and foundations theories of belief revision. In *AAAI'94, Proc. 12th (U.S.) Nat. Conf. on Artificial Intelligence*, pp. 909–914, 1994.

[6] A. del Val. Tractable databases: How to make propositional unit resolution complete through compilation. In *KR'94, Proc. 4th Int. Conf. on Principles of Knowledge Representation and Reasoning*, pp. 551–561. Morgan Kaufmann, 1994.

[7] A. del Val. An analysis of approximate knowledge compilation. In *IJCAI'95, Proc. 14th Int. Joint Conf. on Artificial Intelligence*, pp. 830–836, 1995.

[1] Automated reasoning: www.ii.uam.es/~delval/doctorado.html; Automated Planning: www.ii.uam.es/~jsierra/doct/doctorado-99.html.

[8] A. del Val. Approximate knowledge compilation: The first order case. In *AAAI'96, Proc. 13th (U.S.) Nat. Conf. on Artificial Intelligence*, pp. 498–503. AAAI Press/MIT Press, 1996.

[9] A. del Val. Non-monotonic reasoning and belief revision: Syntactic, semantic, foundational, and coherence approaches. *J. of Applied Non-Classical Logics*, 7(2):213–240, 1997.

[10] A. del Val. How can psychology help artificial intelligence. In *Interfaces of Psychology*, Evora, Portugal, 1999. Universidade d'Evora.

[11] A. del Val. A new method for consequence finding and compilation in restricted languages. In *AAAI'99, Proc. 16th (U.S.) Nat. Conf. on Artificial Intelligence*, pp. 259–264. AAAI Press/MIT Press, 1999.

[12] A. del Val. On 2-SAT and Renamable Horn. In *AAAI'2000, Proc. 17th (U.S.) Nat. Conf. on Artificial Intelligence.* AAAI Press/MIT Press, to appear, 2000.

[13] A. del Val. The complexity of restricted consequence finding and abduction. In *AAAI'2000, Proc. 17th (U.S.) Nat. Conf. on Artificial Intelligence.* AAAI Press/MIT Press, to appear, 2000.

[14] A. del Val. On some tractable classes in deduction and abduction. *Artificial Intelligence*, 116:297–313, 2000.

[15] A. del Val. Tractable classes for directional resolution. In *AAAI'2000, Proc. 17th (U.S.) Nat. Conf. on Artificial Intelligence.* AAAI Press/MIT Press, to appear, 2000.

[16] A. del Val, P. Maynard–Reid and Y. Shoham. Qualitative reasoning about perception and belief. In *IJCAI'97, Proc. 15th Int. Joint Conf. on Artificial Intelligence*, pp. 508–513, 1997.

[17] A. del Val and Y. Shoham. Deriving properties of belief update from theories of action. *Journal of Logic, Language, and Information*, 3:81–119, 1994. Based on papers by same authors in AAAI'92 and IJCAI'93.

[18] A. del Val and Y. Shoham. A unified view of belief revision and update. *Journal of Logic and Computation*, 4:797–810, 1994.

[19] M. Ginsberg, D. Etherington, and B. Drabble. Can search play a role in practical applications? In *Proc. of AI Meets the Real World*, 1998.

[20] K. Inoue. Linear resolution for consequence-finding. *Artificial Intelligence*, 56:301–353, 1992.

[21] H. Kautz and B. Selman. BLACKBOX: A new approach to the application of theorem proving to problem solving. In *Workshop on Planning as Combinatorial Search, in conjunction with AIPS-98 (Conference on Artificial Intelligence Planning Systems)*, 1998.

[22] D. W. Loveland. Automated deduction. *AI Magazine*, 20(1 (Spring)):77–98, 1999.

[23] P. Marquis. Consequence-finding algorithms. In D. Gabbay and Ph. Smets, editors, *Handbook of Defeasible Reasoning and Uncertainty Management Systems.* Kluwer, 1999.

[24] NASA. Remote agent. http://rax.arc.nasa.gov, 1999.

[25] JF. Puget. Constraint programming: A great AI success. In *ECAI'98, Proc. 13th European Conf. on Artificial Intelligence*, pp. 698–705, 1998.

[26] B. Selman and H. Kautz. Knowledge compilation and theory approximation. *Journal of the ACM*, 43(2):193–224, March 1996.

[27] Y. Shoham and A. del Val. A logic for perception and belief. Technical Report STAN-CS-91-1391, Dept. Computer Science, Stanford Univ., 1991.

[28] B. C. Williams and P. Pandurang Nayak. Immobile robots: Artificial intelligence in the new millenium. *AI Magazine*, 17(3 (Fall)):16–35, 1996.