Artificial Intelligence 234 (2016) 219–271

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

Semi-equilibrium models for paracoherent answer set programs ☆

Giovanni Amendola [a], Thomas Eiter [b,*], Michael Fink [b], Nicola Leone [a], João Moura [c]

[a] *Department of Mathematics and Computer Science, University of Calabria, Via P. Bucci, Cubo 30b, 87036 Rende (CS), Italy*
[b] *Institute of Information Systems, Vienna University of Technology, Favoritenstraße 9-11, A-1040 Vienna, Austria*
[c] *NOVA LINCS, Departamento de Informática, Universidade Nova de Lisboa, 2829-516 Caparica, Portugal*

ARTICLE INFO

Article history:
Received 22 December 2014
Received in revised form 14 October 2015
Accepted 21 January 2016
Available online 4 February 2016

Keywords:
Answer set programming
Equilibrium logic
Paracoherent reasoning
Splitting sequences
Inconsistency management

ABSTRACT

The answer set semantics may assign a logic program to model, due to logical contradiction or unstable negation, which is caused by cyclic dependency of an atom on its negation. While logical contradictions can be handled with traditional techniques from paraconsistent reasoning, instability requires other methods. We consider resorting to a paracoherent semantics, in which 3-valued interpretations are used where a third truth value besides true and false expresses that an atom is believed true. This is at the basis of the semi-stable model semantics, which was defined using a program transformation. In this paper, we give a model-theoretic characterization of semi-stable models, which makes the semantics more accessible. Motivated by some anomalies of semi-stable model semantics with respect to basic epistemic properties, we propose an amendment that satisfies these properties. The latter has both a transformational and a model-theoretic characterization that reveals it as a relaxation of equilibrium logic, the logical reconstruction of answer set semantics, and is thus called the semi-equilibrium model semantics. We consider refinements of this semantics to respect modularity in the rules, based on splitting sets, the major tool for modularity in modeling and evaluating answer set programs. In that, we single out classes of canonical models that are amenable for customary bottom-up evaluation of answer set programs, with an option to switch to a paracoherent mode when lack of an answer set is detected. A complexity analysis of major reasoning tasks shows that semi-equilibrium models are harder than answer sets (i.e., equilibrium models), due to a global minimization step for keeping the gap between true and believed true atoms as small as possible. Our results contribute to the logical foundations of paracoherent answer set programming, which gains increasing importance in inconsistency management, and at the same time provide a basis for algorithm development and integration into answer set solvers.

☆ Some of the results were presented in preliminary form at KR 2010 [17] and JELIA 2014 [2]. This work was partially supported by Regione Calabria under the EU Social Fund and project PIA KnowRex POR FESR 2007–2013, by the Vienna Science and Technology Fund (WWTF) grant ICT 08-020, the Austrian Science Fund (FWF) grant P20841, and by the Italian Ministry of University and Research under PON project Ba2Know (Business Analytics to Know) S.I.-LAB n. PON03PE_0001. The work of J. Moura was supported by grant SFRH/BD/69006/2010 from Fundação para a Ciência e a Tecnologia (FCT) from the Portuguese Ministério do Ensino e da Ciência.

* Corresponding author.
E-mail addresses: amendola@mat.unical.it (G. Amendola), eiter@kr.tuwien.ac.at (T. Eiter), fink@kr.tuwien.ac.at (M. Fink), leone@mat.unical.it (N. Leone), joaomoura@yahoo.com (J. Moura).

http://dx.doi.org/10.1016/j.artint.2016.01.011
0004-3702/

1. Introduction

Answer Set Programming (ASP) is a premier formalism for nonmonotonic reasoning and knowledge representation, mainly because of the existence of efficient solvers and well-established relationships with common nonmonotonic logics. It is a declarative programming paradigm with a model-theoretic semantics, where problems are encoded into a logic program using rules, and its models, called answer sets (or stable models) [26], encode solutions; see [6,11,24].

As is well-known, not every logic program has some answer set. This can be due to different reasons: (1) an emerging logical contradiction, as e.g. for the program

$$P = \{\, locked(door) \leftarrow not\; open(door);\;\; -locked(door)\,\}$$

where "$-$" denotes strong (sometimes also called classical) negation and "*not*" denotes weak (or default) negation; according to the first rule, a door is locked unless it is known to be open, and according to the second rule it is not locked. The problem here is a missing connection from $-locked(door)$ to $open(door)$.[1] (2) Due to cyclic dependencies which pass through negation, as e.g. in the following simplistic program.

Example 1. Consider the barber paradox, which can be regarded as an alternative form of Russell's famous paradox in naive set theory[2]: in some town, the barber is a man who shaves all men in town, and only those, who do not shave themselves. The paradox arises when we ask "Who shaves the barber?". Assuming that Joe is the barber, the knowledge about who is shaving him is captured by the logic program

$$P = \{shaves(joe, joe) \leftarrow not\; shaves(joe, joe)\},$$

(where *joe* is the barber), which informally states that Joe shaves himself if we can assume that he is not shaving himself. Under answer set semantics, P has no model; the problem is a lack of stability, as either assumption on whether $shaves(joe, joe)$ is true or false cannot be justified by the rule.

In general, the absence of an answer set may be well-accepted and indicates that the rules cannot be satisfied under stable negation. There are nonetheless many cases when this is not intended and one might want to draw conclusions also from a program without answer sets, e.g., for debugging purposes, or in order to keep a system (partially) responsive in exceptional situations; in particular, if the contradiction or instability is not affecting the parts of a system that intuitively matter for a reasoning problem.

In order to deal with this, Inoue and Sakama [49] have introduced a paraconsistent semantics for answer set programs. While dealing with logical contradictions can be achieved with similar methods as for (non-) classical logic (cf. also [9,1,37]), dealing with cyclic default negation turned out to be tricky. We concentrate in this article on the latter, in presence of constraints, and refer to it as *paracoherent reasoning*, in order to distinguish reasoning under logical contradictions from reasoning on programs without strong negation that lack stability in models.

With the idea that atoms may also be possibly true (i.e., believed true), Inoue and Sakama defined a semi-stable semantics which, for the program in Example 1, has a model in which $shaves(joe, joe)$ is *believed true*; this is (arguably) reasonable, as $shaves(joe, joe)$ cannot be false while satisfying the rule. Note however that believing $shaves(joe, joe)$ is true does not provide a proof or founded justification that this fact is actually true; as a mere belief it is regarded to be weaker than if $shaves(joe, joe)$ would be known as a fact or derived from a rule.

In fact, semi-stable semantics *approximates* answer set semantics and coincides with it whenever a program has some answer set; otherwise, under Occam's razor, it yields models with a smallest set of atoms believed to be true. That is, the intrinsic *closed world assumption (CWA)* of logic programs is slightly relaxed for achieving stability of models.

In a similar vein, we can regard many semantics for non-monotonic logic programs that relax answer sets as *paracoherent semantics*, e.g. [4,19,39,43,44,47,48,51,56,59]. Ideally, such a relaxation meets for a program P the following properties:

(D1) Every (consistent) answer set of P corresponds to a model (*answer set coverage*).
(D2) If P has some (consistent) answer set, then its models correspond to answer sets (*congruence*).
(D3) If P has a classical model, then P has a model (*classical coherence*).

In particular, (D3) intuitively says that in the extremal case, a relaxation should renounce to the selection principles imposed by the semantics on classical models (in particular, if a single classical model exists).

Widely-known semantics, such as 3-valued stable models [47], L-stable models [19], revised stable models [43], regular models [59], and pstable models [39], satisfy only part of these requirements (see Section 8.2 for more details). Semi-stable models however, satisfy all three properties and thus have been the prevailing paracoherent semantics.

[1] Constraints (rules with empty head) may be considered as descriptions of cases when inconsistency arises, if $\perp$ (falsum) is added to the head; however, also an instability view is possible, cf. Section 6.2.

[2] Namely, that the set of all sets that are not members of themselves cannot exist.