From: **KR 2020** kr2020@easychair.org
Subject: KR 2020 notification for paper 98
Date: 3 June 2020 at 19:02
To: Alessandra Russo a.russo@imperial.ac.uk

****
This email originates from outside Imperial. Do not click on links and attachments unless you recognise the sender.
If you trust the sender, add them to your safe senders list https://spam.ic.ac.uk/SpamConsole/Senders.aspx to disable email stamping for this address.
****

Dear Alessandra,

We are pleased to inform you that your submission

98: Stable and Supported Semantics in Continuous Vector Spaces

has been accepted to the technical program of KR2020 as a full paper.
Congratulations!

We received 241 full and short paper submissions to the conference, of which 83 (34%) were accepted as full papers (including conditional acceptances) and 12 (5%) were accepted as short papers to the main technical program (including the Applications and Systems track, and the special sessions on KR & Machine Learning, and KR & Robotics). In addition, 29 papers were recommended for poster presentations. As we received a large number of high-quality submissions, the decisions were not easy and the threshold for acceptance was high. You can be proud of the success of your paper!

The KR2020 Program Committee worked hard to provide a careful evaluation of the submitted papers: each paper was reviewed by at least three experts, and discussed amongst them, the area/track chair(s), and the program chairs. The reviews for your paper are appended. Please take them carefully into account when preparing the camera-ready copy of your paper.

Please note:

1. The camera-ready copy of a full paper is restricted to 10 pages including references. Please address any comments or concerns raised by the program committee in the final version of your paper.

   The final, hard deadline for submissions of the camera-ready version of your paper and all source files, along with a signed copyright form, is

   ** Wednesday, July 1, 2020 AoE (23:59 UTC-12). **

   The accepted full papers and short papers will be published by IJCAI Inc. in the KR2020 proceedings.

2. Regarding the camera-ready copy submission, you will receive a separate email with an invitation link and instructions to submit your final version. Please stay tuned. If you do not receive an invitation in the next 48 hours, please contact kr.proceedings@confdna.com .

3. At least one author must register and attend the conference for the paper to be included in the proceedings (but of course we hope that all authors will attend KR2020).

   The deadline for registration (for at least one author) is

   ** Wednesday, June 24, 2020 AoE (23:59 UTC-12). **

   The registration page will open soon.

4. In addition to publication of your paper in the KR2020 proceedings, you will be given an oral presentation slot at the conference. The details will be provided in a separate email later on.

5. For the first time at a KR conference, we have a Recent Published Research (RPR) track, which provides a forum to discuss important results related to KR that appeared recently in selective journals and conferences but have not been previously presented at KR.

   The submission of papers to the RPR track is still open, with a deadline

   ** Monday, June 15, 2020 AoE (23:59 UTC-12). **

6. We have a strong workshop program, in addition to the usual tutorial
program, before the main conference:

-- Explainable Logic-Based Knowledge Representation (XLoKR 2020)
-- International Workshop on Applications of AI to Forensics (AI2Forensics)
-- KR for All Minds (KR4AMinds)
-- Models of Legal Reasoning (MLR)
-- Reasoning about ACtions and Events over Streams (RACES)

The submission of papers to all workshops is still open, with a deadline

** Monday, June 22, 2020 AoE (23:59 UTC-12). **

Please check the workshops' web pages at
https://kr2020.inf.unibz.it/page/workshops_and_tutorials
for details, and consider submitting papers.

7. The Doctoral Consortium is still accepting submissions till

** Friday, June 5, 2020 AoE (23:59 UTC-12). **

Information on the application can be found at
https://kr2020.inf.unibz.it/page/doctoral_consortium_call .

8. We have an excellent roster of invited speakers, as detailed on the KR2020
web site at https://kr2020.inf.unibz.it/page/invited_talks .

We assume that KR2020 will take place as planned, during September 12-18, 2020,
in Rhodes, Greece. Meanwhile, we are also planning on allowing authors of
accepted papers to participate remotely, should they not be able to travel. We
will send a separate email about this later on.

Congratulations again on your success with your paper.
We look forward to seeing you at KR2020.

Sincerely,
Diego Calvanese and Esra Erdem
KR2020 Program co-Chairs

SUBMISSION: 98
TITLE: Stable and Supported Semantics in Continuous Vector Spaces

---------- METAREVIEW ----------
This paper introduces a gradient-based search method for computing supported and stable models of normal logic programs. The method is novel and brings an interesting idea to bridge between KR and ML or other related areas using algebraic and neural computation. This would open up a new research direction in the area of KR.

Unfortunately, the experimental part in this paper is weak. The authors could compare with existing symbolic/non-differentiable or differentiable methods/solvers to compute supported and stable models in some way. Other suggestions can be found in each review (we have added one extra review).

---------- REVIEW 1 ----------
SUBMISSION: 98
TITLE: Stable and Supported Semantics in Continuous Vector Spaces
AUTHORS: Yaniv Aspis, Krysia Broda, Alessandra Russo and Jorge Lobo

---------- Overall evaluation ----------
SCORE: 0 (borderline paper)
---------- Reviewer's confidence ----------
SCORE: 3 (medium)
---------- Relevance of the paper to KR ----------
SCORE: 4 (good)
---------- Novelty ----------
SCORE: 4 (good)
---------- Significance ----------
SCORE: 3 (fair)
---------- Technical quality ----------
SCORE: 4 (good)
---------- Discussion of related work ----------
SCORE: 4 (good)
---------- Quality of the presentation ----------
SCORE: 4 (good)

---------- Review ----------
This paper presents a novel method to compute stable and supported semantics of logic programs based on Newton's method. To achieve this, a characterisation of these semantics in terms of differentiable operators is developed.

While it seems like there is value in the presented line of work. I do not believe the current paper is ready for publication. Various comments follow.

First of all the claim that this is a method to compute "supported and stable" models seems to be a bit exaggerated. I agree that this method allows computing supported models. However, for computing stable models this method simply boils down to a guess and check (as can be witnessed by the bad performance for "stable" in table 1.)

In the last paragraph of related work, a related paper is discussed that uses similar methods for SAT. However, since the current work is mainly about computing supported models of programs, i.e., classical models of Clark's completion, a more serious discussion of the relation with the existing work on gradient-based methods for SAT is required. I would also expect this to occur in the experimental section: what is gained or lost by using this operator-based approach compared to the existing work?
Additionally, since logic programs under stable semantics can be translated into sat (eg using lp2lp2 I lp2sat), a comparison between the here-presented operator-based method and the existing work on gradient-based methods for SAT is also in its place for stable semantics.

The experimental section seems a bit weak. One of the motivations for the current work is the fact that possible large computational gains can be made. It would be nice to see an indication of this. I don't expect new technology to outperform SOA solvers, but it would be nice to have at least an indication of promise in this respect. Why, and on what type of programs, can we at least hope to outperform SOA solvers in the future?
Additionally, the results showing that when programs grow a little bit, all hope of finding "real" interpretations is lost is discouraging.

Proposition 2 is wrong. It only holds for some values of of all the parameters, as can be seen scattered over the preceding discussion. This should not be phrased like this.
Also, the discussion above it was hard to follow.

Some more thoughts:

The title of the paper does not reflect the contents: it suggests that the paper is about semantics, while it is actually about computation. A better title might be "Computing supported models in continuous vector spaces"

It is claimed that a discrete matrix representation of propositional normal programs [based on the reduct] is developed. But this appears to not be true. There is no single matrix representation of a program. Instead, for each interpretation, I there is a representation of the reduct P^I

This paper in general appears to be targeted at people familiar with logic programming but the background section reads as if it is targeted at people unfamiliar with the stable semantics. This is not a big issue of course, but it might be better to align a bit more.

The notation D^{P^I} is very bad. P^I is a definite logic program, so D^{P^I} is already defined in (10).
In (13) it is defined differently.

Where do the magic numbers, e.g. 0,0008 and 0,179 in the experimental section come from?

* abstract "the application of the immediate consequence operator to a program reduct": the immediate consequence operator can only by applied on interpretations
* I don't know what the sentence "the underlying semantics of a normal logic program is preserved with respect to matrix multiplication and a differentiable operation" means
* clauses representing "background knowledge, constraints, or choices" -> It feels strange to see these three items in an enumeration since constraints are for instance a specific type of background knowledge, no?
* In the intro: it is unclear what "simulation of a choice rule" means.
* "a supported model and minimal" -> "a supported and minimal model"
* "the grounding of" should not be mentioned since the entire paper is propositional
* In equation (7), it might be better to use \mid instead of I
* The inclusion of the clause \true \if \true is suspicious. It suggests that also \true is (under the stable semantics) subject to minimisation, which it is not of course.
* The use of a different font for a vector when referring to the vector vs when referring to its elements is confusing. Why not use the same font (and only vary the boldness)? Similarly, the fact that the matrix D is non-italic while D_{ij} is italic is confusing.
* The statement "Rules can be embedded as NtimesN matrices" should be weakened since this is only done for definite rules.
* In section 3 some more motivation would be welcome (e.g. why divide by n^r if one will apply H_1 anyway?)
* Also, the intuition above (13) should really mention the MD condition. It is essential for this technique to work, as can also be seen in the proof. It took me a long while to figure this out.
* What happened with the spacing in Algorithm1?
* In (19), either M and I should be fixed before this or they should be included in the notation since x really depends on M and I.
* the sentence below (19) is incorrect

I was a bit surprised by the answer to my questions:

I was a bit surprised by the answer to my questions.
The authors propose a new gradient-based method for computing supported models (and combine this with a standard mechanism to check stability of a supported model).
It is obvious that two gradient-based methods for tackling this problem already exist (though have never been tested!)
* Take Clark's completion (using eg lp2sat); call a gradient-based SAT solver; check for stability
* Use any out-of the box tool to translate a logic program under stable semantics into SAT (using eg lp2lp2 | lp2sat); call a gradient-based SAT solver

In your reply you dismissed these suggestions as:
2.R1: "our contribution is not solving sat problems, hence we are not interested in the first option"
2.R2 "this is tangential to our line of research".

However, to me, these existing methods should be the baseline to be compared to. It should give us an idea whether the reduct-based method gives us any benefit over plain SAT translations.

Since the gradient-based methods for solving this type of problems is still in its infancy, I don't think that comparison to standard ASP solvers should happen necessarily (1.R2). One can indeed not expect a novel paradigm to outperform a decade of engineering. But.... comparing to other gradient-based approaches cannot just be dismissed; especially not if they solve the SAT problem which has always been very close to ASP solving.
The contribution of the paper is a new method for finding supported models (stable models follow from that). The task of finding supported models of a logic program is easily cast as a SAT problem and vice versa.
You propose an immediate-consequence-operator-based method for computing supported models of a logic program, thereby exploiting some more definitional structure of the input compared to what a SAT solver can do.
To me, it seems only logical that this new method would be compared to the baseline: do the new methods yield an advance or not?
My feeling is that you dismiss comparing with SAT a bit too easily "because it is a different problem", while actually to problems are not so different at all.
(Slightly simplifying, the main difference between an ASP solver and a SAT solver is a loop-handling mechanism, an unfounded set propagator, but that is exactly what becomes unnecessary when one wants to compute supported models.)

So, all in all, this results in a weak experimental section.

However, on the other hand, the following need to be taken into account:
* I don't know how advanced the existing gradient based methods for SAT are. I also don't know how easily extent gradient-based sat solvers can be used (are there any openly available?)
* The paper deviates well from established KR topics. A goal of KR is to attract this type of papers.

---------- REVIEW 2 ----------
SUBMISSION: 98
TITLE: Stable and Supported Semantics in Continuous Vector Spaces
AUTHORS: Yaniv Aspis, Krysia Broda, Alessandra Russo and Jorge Lobo

---------- Overall evaluation ----------
SCORE: 1 (weak accept)
---------- Reviewer's confidence ----------
SCORE: 4 (high)
---------- Relevance of the paper to KR ----------
SCORE: 4 (good)
---------- Novelty ----------
SCORE: 4 (good)
---------- Significance ----------
SCORE: 3 (fair)
---------- Technical quality ----------
SCORE: 4 (good)
---------- Discussion of related work ----------
SCORE: 4 (good)
---------- Quality of the presentation ----------
SCORE: 3 (fair)
---------- Review ----------
The paper introduces methods for computing supported/stable models of normal logic programs in vector spaces.
It represents propositional normal programs using discrete matrices and extends it to continuous vector space. It then computes supported/stable models of normal programs using a gradient-based search algorithm. Finally, it provides experimental results on success rates for computing supported/stable models.

As mentioned in the introduction, linear algebra offers a variety of algorithms and efficient computation on GPU hardware. There are several studies that aim at realizing symbolic reasoning in vector spaces.
The present paper provides a novel technique that has a good potential to realize logical inference in huge scale of knowledge bases.

There are several solvers for computing stable models. The present paper addresses no comparison with existing solvers, however. It needs benchmark testing to convince readers that the approach is competitive with the current state of the art.

For computing supported models, the following publication is related to the topic.

Sato, et al.: From 3-valued Semantics to Supported Model Computation for Logic Programs in Vector Spaces,
Proceedings of the 12th International Conference on Agents and Artificial Intelligence (ICAART), pages 758-765, Malta, 2020.

Minor comments:

(page 2, right)
two unique stable models --> two stable models

(page 3)
Please write that a vector v represents a column vector (so v^T represents a row vector).

(right, line -8)
only r is in P^I --> only r^I is in P^I

---------- REVIEW 3 ----------
SUBMISSION: 98
TITLE: Stable and Supported Semantics in Continuous Vector Spaces
AUTHORS: Yaniv Aspis, Krysia Broda, Alessandra Russo and Jorge Lobo

---------- Overall evaluation ----------
SCORE: 1 (weak accept)
---------- Reviewer's confidence ----------
SCORE: 3 (medium)
---------- Relevance of the paper to KR ----------
SCORE: 4 (good)
---------- Novelty ----------
SCORE: 5 (excellent)
---------- Significance ----------
SCORE: 4 (good)
---------- Technical quality ----------
SCORE: 4 (good)
---------- Discussion of related work ----------
SCORE: 4 (good)
---------- Quality of the presentation ----------
SCORE: 4 (good)
---------- Review ----------
The paper introduces an approach to compute supported and stable models of a normal logic program, using a gradient-based search algorithm in continuous vector spaces. In specific, the paper first extends the work in (Sakama et al. 2017) and presents a discrete matrix representation of an NLP based on the concept of program reduct. Then the paper provides a differentiable approximation of the discrete representation to continuous space and proves under what conditions two-valued semantics is preserved, which implies an algorithm for gradient-based computation of supported and stable models. At last, the experiments show that the rate of success of the algorithm for computing supported and stable models can be greatly affected by its parameter values.

Generally, the paper is well structured and easy to follow. As far as I know, this gradient-based search algorithm for computing supported and stable models of an NLP is novel, which has the potential to benefit efficient computation in huge scale of programs and provides a potential way to connect logic programming with neural networks. I think the result might interest researchers from several areas.

It's a pity that the rate of success of the algorithm is significantly affected by its parameter values, which reduces the reliability of computed models.
The efficiency of the approach is not discussed in the paper. More efficient implementations can be developed in future work.

---------- REVIEW 4 ----------
SUBMISSION: 98
TITLE: Stable and Supported Semantics in Continuous Vector Spaces
AUTHORS: Yaniv Aspis, Krysia Broda, Alessandra Russo and Jorge Lobo

---------- Overall evaluation ----------
SCORE: 1 (weak accept)
---------- Reviewer's confidence ----------
SCORE: 5 (expert)
---------- Relevance of the paper to KR ----------
SCORE: 5 (excellent)
---------- Novelty ----------
SCORE: 4 (good)
---------- Significance ----------
SCORE: 3 (fair)
---------- Technical quality ----------
SCORE: 3 (fair)
---------- Discussion of related work

---------- Discussion of related work ----------
SCORE: 4 (good)
---------- Quality of the presentation ----------
SCORE: 3 (fair)
---------- Review ----------
This paper proposes a novel approach to compute supported and stable models of normal logic programs in continuous vector spaces based on a gradient-based technique. The method is a continuous version of operations proposed by Sakama et al. as a non-continuous matrix method. The author used a sigmoid function to make the method continuous. Then the paper reports some results on experiments.

I found the novelty of this paper in its use of sigmoid and the derivation of related computation that is necessary to get supported models. This is highly technical, but the choice of the sigmoid function would be arguable. As in the trend of neural network research, one could consider LeRU and other functions that are considered better than sigmoid.

There are two theorems (propositions) in this paper.
(1) Proposition 1 is a variant of Sakama et al.'s theorem, in which the difference lies only in the use of reduct for normal programs. I cannot understand in the proof that two cases are considered depending on the condition that a rule is the only long rule or not. In fact there is no definition for a long rule in the paper.

(2) Proposition 2 is more important and essential in this paper. But the proof here is based on the derivation before the proposition appears, and the condition for this proposition is not explicitly stated. This part is essential in this paper so must be handled more carefully. Then after this, the authors introduce the Jacobi method with two sampling schemes, but unfortunately this description lacks the details and there is no formal property on this method. Again this part is technically important, so needs more easily understandable description.

On the experiments in Section 4, the authors seen the success rate to get supported and stable models by the proposed method. As the authors admit, this is not to demonstrate efficiency of the method, and in fact there is no such evaluation in the paper. This is rather disappointed, and I wonder here what is the motivation and goal of the whole work. Section 1 does not fully discuss why the differentiable method is important for computing supported and stable models of logic programs. The authors just say that performing symbolic reasoning in continuous vector spaces or via deep learning is interesting. It is not clear if they want to handle a huge amount of facts or noisy data, and indeed the experiments are not coordinated to support such claims. That is, a clear goal has not yet set in the beginning, so the contribution is not clear either.

Overall, I found this work is very interesting, while the performance evaluation is not enough. This paper tries to open up a novel research direction in the area of KR and establishes a bridge to a different discipline, but the work is not mature in this stage.

Reference:
"Sato, T., and Kojima, R. 2019. Logical inference as cost minimization in vector spaces." should be given its source: "Proc. of the Fourth International Workshop on Declarative Learning Based Programming (DeLBP'19), https://delbp.github.io/".