Subject: TACAS 2017 notification for paper 143
Date: Thursday, 22 December 2016 at 20:15:29 Greenwich Mean Time
From: TACAS 2017
To: Thomas Melham

Dear Authors,

we regret to inform you that your submission N. 143 with title
Model Search in Relational Domains
was not accepted for TACAS 2017.

TACAS had a very strong field of submissions and the PC could only accept the allotted number of papers.
We hope that you will find the feedback and comments contained in the reviews useful.

It would be nevertheless our great pleasure to meet you at ETAPS 2017 as attendee, and we'd welcome your f
submissions to TACAS again in the coming years.

Kind regards
Tiziana Margaria and Axel Legay

----------------------- REVIEW 1 ---------------------
PAPER: 143
TITLE: Model Search in Relational Domains
AUTHORS: Rajdeep Mukherjee, Peter Schrammel, Daniel Kroening and Tom Melham

Overall evaluation: -1
Reviewer's confidence: 2

----------- Review -----------
the paper is about extending Abstract Conflict-Driven Clause Learning (ACDCL) (a generalisation of the CDCL algorithm used in SAT solvers) to relational domains.

This paper is unfortunately needlessly difficult to read, at least for people non working in this subfield. Even the explanation of the trivial program 1 is confused, and seemingly reliant on implicit assumptions (also the other figures in the paper seem to confuse more than help); the authors claim that the paper is self-contained but for example

- even the basic concept of relational domain is never defined (“octagon is a relations domain” is not a definition)
- 2LS is the tool used for the experiments but no description of its features is given…
- the programs analysed in experiments are not described in any details allowing an assessment whether the contribution is relevant

about figures:
-fig 2 : the SSA transformation seems incorrect:
(x0 = v) should be guarded by c but it isn’t , also is an “and” missing at the end of first line in the figure?
- “Figure 4 shows a snapshot of an abstract conflict”: any explanation how is that figure depicting a conflict?

A central concept for defining the conflict learning in this work is the one of meet irreducible (already used in the literature) and a sort of complement of meet irreducible which is defined in terms of concretisation from an abstraction.
The main algorithm refines concatenations of meet irreducible (abstract values) and “deduce” new ones based on current decision. This process of decision and deduction is alternated until unsafe is returned or a conflict is found. If a conflict is found it is analysed and the analysis allows to remove part of the trail (the constituent of an abstract value): this is the learning phase, the ACDCL part.
Beyond this general overview, I couldn’t really work out the details of the algorithm(s) and as said the narrative seems more confusing than helping. Also some expressions seem plain wrong, e.g. “[[x = y +

z]]^Itvs[{y,z}]_Itvs[{x,y,z}] = a ⊓ ⊤ “:   where is the “a” there coming from?

The paper builds on [12,13], which are well written papers both in the background and in illustrating the concepts.
This work is not up those standards of conceptual clarity and on the achievements it is seemingly rather incremental.

No doubt there is lot of technical work involved, but the concepts and the algorithms should be better explained, and the experiments, with no adequate explanation of what these benchmarks look like, are unconvincing.

----------------------- REVIEW 2 ---------------------
PAPER: 143
TITLE: Model Search in Relational Domains
AUTHORS: Rajdeep Mukherjee, Peter Schrammel, Daniel Kroening and Tom Melham

Overall evaluation: 1
Reviewer's confidence: 2

----------- Review -----------
The authors describe an extension of the Abstract Conflict Driven Clause Learning (ACDCL) approach to so-called relational domains. ACDCL adapts the conflict-driven clausing learning approaches underpinning modern SAT solvers to non-boolean domains, and may thus be seen as edging into SMT solving, albeit not at the generality of traditional SMT solvers, but with the promise of greater efficiency. Existing ACDCL techniques only supported domains in which constraints are applied to single variables (i.e. bounds); the current paper extends the technique to constraints involving multiple variables (e.g. general convex polyhedra). Algorithms for the extension, and an experimental evaluation of the technique is given on testbeds coming from various sources vis a vis the software model checker CBMC and the commercial tool Astree.

The paper is a bit difficult to follow in places, primarily because intuitions about what "relational domains" are, exactly, are not given. A precise mathematical definition is presented, and there it is clear that in fact polyhedra (i.e. linear constraints) are the abstract domains considered. This is fine, but it would be nicer to spell this out up front, as one can imagine many other kinds of relational domains. That said, the experimental evaluation is impressive, showing excellent results for the new technique.

----------------------- REVIEW 3 ---------------------
PAPER: 143
TITLE: Model Search in Relational Domains
AUTHORS: Rajdeep Mukherjee, Peter Schrammel, Daniel Kroening and Tom Melham

Overall evaluation: 1
Reviewer's confidence: 2

----------- Review -----------
The paper describes program analysis based on a generalization of conflict-driven clause learning. In this approach, programs are translated into a formula that is unsatisfiable iff the program is safe. The formulas are obtained from abstract transformers for the individual program components. Models of the formula are then searched for using various abstract domains. The model search involves a step of decision in which an abstract element is decomposed into meet-irreducibles which are then candidates for refining the current model in the search for a counterexample.

The paper is well written and reports work that improves on the authors' 2LS verification tool.

One concern is the programs to which this approach is applicable. It is stated that this applies to bounded programs that are obtained by a program transformation that unfolds loops". Since the paper does not address loops at all, I take that this means that loops are removed completely and replaced by a finite unfolding. In this case, the comparison with a static analyzer such as Astree becomes questionable. The paper should clarify this point.