Set Constraints, Pattern Match Analysis, and SMT

Joseph Eremondi

University of British Columbia

TFP 2019

Introduction

Safe if **`tree`** is never `(Branch [])`

```
numNodes tree =
  case tree of
    Leaf data -> 1
    Branch [child] ->
      1+numNodes child
    Branch (h:t) ->
      1+foldMap numNodes t
```

Missing `Branch []`

Model set of values **`tree`** can safely take

Pattern Match Analysis (ctd.)

Return contains `1` iff
`tree` contains `Leaf _`

```
numNodes tree =
  case tree of
    Leaf data -> 1
    Branch [child] ->
      1+numNodes child
    Branch (h:t) ->
      1+foldMap numNodes t
```

Result set defined recursively

Reachable iff >2 children

Model set of values `numNodes` can return

```
case numNodes tree of
  1 ->
    let (Leaf data) = tree
    in ...
  _ -> ...
```

Know this is safe:
`numNodes` returns 1,
so input is `Leaf`

All can be modeled with (unrestricted) Set Constraints!

What We Need

- Need to reason about:
  - Sets of values
  - Constructor application and projection
- Must support:
  - Variables
  - Negative constraints
  - Boolean combinations (implication)

… Set Constraints!

Anatomy of Set Constraints

Constructor application

Negative constraint

Positive constraint

Projection

$$X \subseteq f(Y) \wedge Y \not\subseteq f^{-1}(Z)$$

Variables to solve for

Solve all the set constraints!

A Decision Procedure
for a Class of Set Constraints

Solving Classes of Set Constraints
with Tree Automata

Solving Systems of Set Constraints
(Extended Abstract)

Decidability of Systems of Set Constraints
with Negative Constraints*

Negative set constraints with equality*

Witold Charatonik† Leszek Pacholski‡
Institute of Computer Science Institute of Mathematics

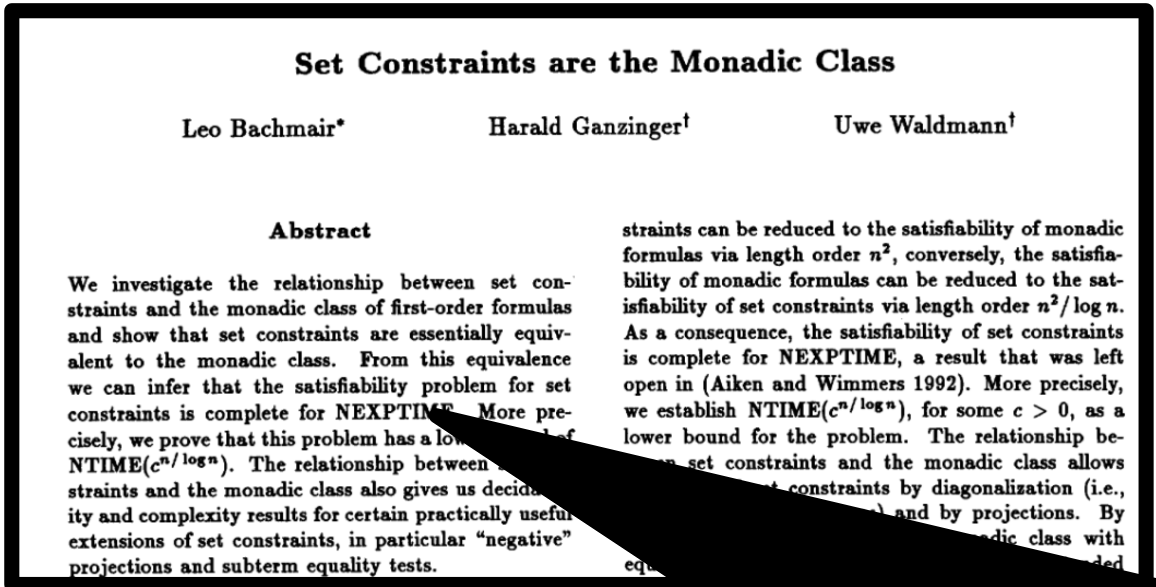
Set Constraints with Projections

WITOLD CHARATONIK AND LESZEK PACHOLSKI

University of Wrocław, Wrocław, Poland

Abstract. Set constraints form a constraint system where variables range over the domain of sets of trees. They give a natural formalism for many problems in program analysis. Syntactically, set constraints are conjunctions of inclusions between expressions built over variables, constructors (constants and function symbols from a given signature) and a choice of set operators that defines the specific class of set constraints. In this article, we are interested in the class of *set constraints with projections*, which is the class with all Boolean operators (union, intersection and complement) and *projections* that in program analysis directly correspond to type destructors. We prove that the problem of existence of a solution of a system of set constraints with projections is in NEXPTIME, and thus that it is NEXPTIME-complete.

Solve all the set constraints?



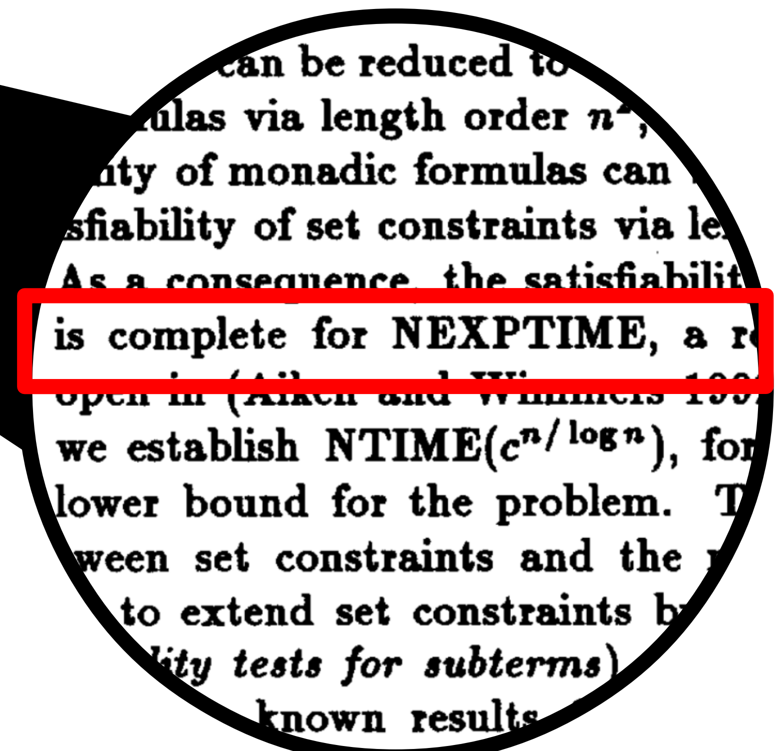
Set Constraints are the Monadic Class

Leo Bachmair* Harald Ganzinger† Uwe Waldmann†

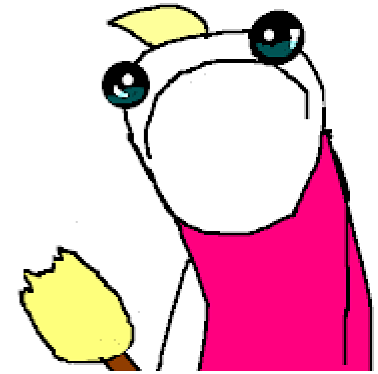
Abstract

We investigate the relationship between set constraints and the monadic class of first-order formulas and show that set constraints are essentially equivalent to the monadic class. From this equivalence we can infer that the satisfiability problem for set constraints is complete for NEXPTIME. More precisely, we prove that this problem has a lo... NTIME($c^{n/\log n}$). The relationship between ... straints and the monadic class also gives us decida... ity and complexity results for certain practically useful extensions of set constraints, in particular "negative" projections and subterm equality tests.

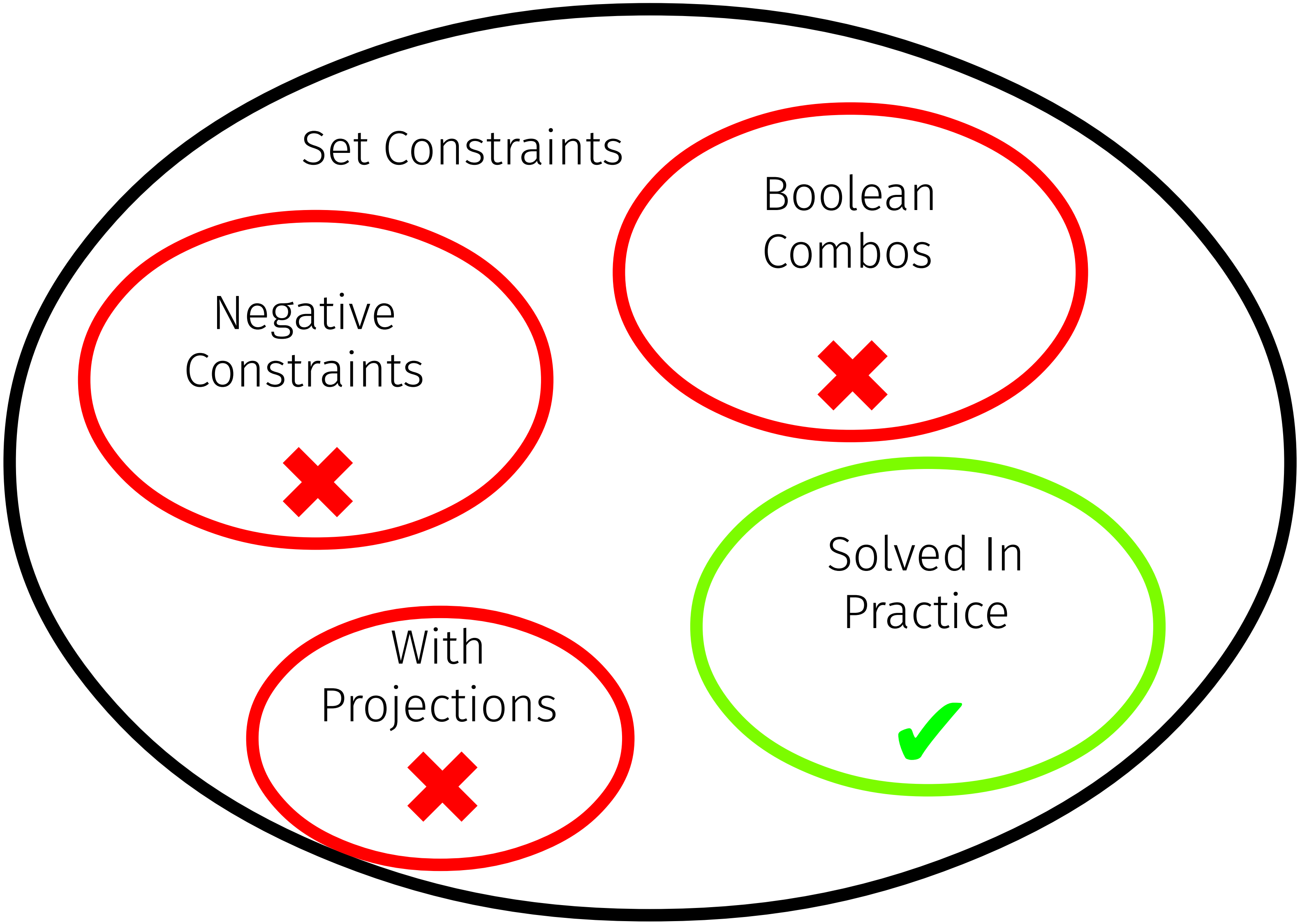
straints can be reduced to the satisfiability of monadic formulas via length order n^2, conversely, the satisfiability of monadic formulas can be reduced to the satisfiability of set constraints via length order $n^2/\log n$. As a consequence, the satisfiability of set constraints is complete for NEXPTIME, a result that was left open in (Aiken and Wimmers 1992). More precisely, we establish NTIME($c^{n/\log n}$), for some $c > 0$, as a lower bound for the problem. The relationship be... ...set constraints and the monadic class allows ... constraints by diagonalization (i.e., ...) and by projections. By ...dic class with ...



...can be reduced to
...ulas via length order n^2,
...ity of monadic formulas can
...sfiability of set constraints via le...
As a consequence, the satisfiabilit...
is complete for NEXPTIME, a r...
open in (Aiken and Wimmers 199...
we establish NTIME($c^{n/\log n}$), fo...
lower bound for the problem. T...
...ween set constraints and the ...
to extend set constraints b...
...ity tests for subterms)...
...known results...



Theory vs. Practice



Solving Set Constraints

$$x \in S \iff P_S(x)$$

Predicate for each sub-expr of constraint (à la Tseitin)

Takes exactly one argument

Semantics of sets as first-order formulas e.g.

$$\forall x.\, P_{\neg S}(x) \iff \neg P_S(x)$$

Set Constraint As Formula

$$S \subseteq T$$

$$\Downarrow$$

$$\forall x.\, P_S(x) \implies P_T(x)$$

$$S \not\subseteq T$$

$$\Downarrow$$

$$\exists x.\, P_S(x) \land \neg P_T(x)$$

Finite Model Property

$\boxed{n\text{-predicate formula holds}} \iff \boxed{\text{formula has model with } \leq 2^n \text{ elems}}$

✔Decidable... by trying 2^{2^n} models✖

- Objects in model?
  - Eq. Classes of Predicate values
- Model is:
  - Subset of possible Eq. classes
- Idea: model domain as a function
  - Type $\mathbb{B}^n \to \mathbb{B}$
  - Filters equivalence classes

Domain $D \subseteq \mathbb{B}^n$

$$0110011\ldots01001$$

Bit i set iff $P_i(x)$ holds for x

Uninterpreted function

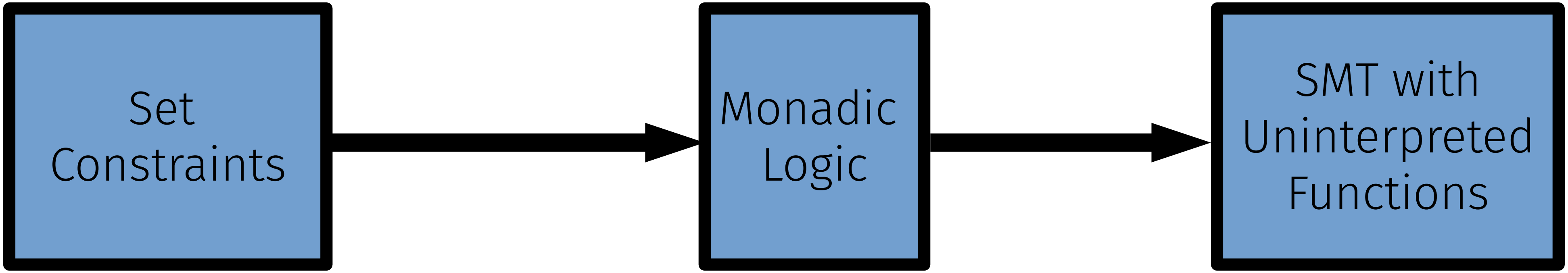
$$\mathsf{inDomain} : \mathbb{B}^n \to \mathbb{B}$$

Qualify formulas e.g.

$$S \subseteq T$$

$$\Downarrow$$

$$\forall x \in \mathbb{B}^n.\, \mathsf{inDomain}(x) \implies (P_S(x) \implies P_T(x))$$



Solver finds:

- impl. for **inDomain** $: \mathbb{B}^n \to \mathbb{B}$
- impl. for each constructor $f : (\mathbb{B}^n)^k \to \mathbb{B}^n$

Results

Translate to SMT

- written in Haskell
- Solved with Z3

Pattern Match Analysis

- Modified Elm compiler
- Emits set constraints

- Compile the rtfeldman/elm-css library
- Measure slowdown:
  - Exhaustiveness checking (default)
  - Pattern-match analysis

Results

- Bad news
  - Min slowdown factor 13 (273 ms)
  - Max slowdown factor 468753 (30 minutes)
- Good news
  - Worst case had domain $\subseteq \mathbb{B}^{50}$
  - 2^{50} ns $\approx$13 days > 30 minutes

Going Forward

Limitations

- It’s slow
  - But not heat-death of the universe slow
- Error messages not great
  - Need UNSAT core of set constraints

Open Questions

- Is Z3 the best option?
  - CVC4 more configurable?
  - QBF or OBDD?
- What causes specific case to blow-up?
  - Large number of projections?
- What heuristics can help with the speedup?

Set constraints can be solved in practice...
but require a bit of patience!

Preprint link at `eremondi.com`