

What's Decidable About Arrays?

Speaker: Philip Busch

Supervisor: Viorica Sofronie-Stokkermans

Motivation

Verification Process:

- Generate verification conditions.
- Prove that each verification condition is valid.

Motivation

Verification Process:

- Generate verification conditions.
- Prove that each verification condition is valid.

How do we prove that verification conditions are valid?

Motivation

Verification Process:

- Generate verification conditions.
- Prove that each verification condition is valid.

How do we prove that verification conditions are valid?

Decision Procedures!

Motivation

A decision procedure for a theory of arrays is of interest for applications in formal verification, program analysis and automated theorem proving.

Motivation

A decision procedure for a theory of arrays is of interest for applications in formal verification, program analysis and automated theorem proving.

Satisfiability:

Motivation

A decision procedure for a theory of arrays is of interest for applications in formal verification, program analysis and automated theorem proving.

Satisfiability:

- Full: undecidable

Motivation

A decision procedure for a theory of arrays is of interest for applications in formal verification, program analysis and automated theorem proving.

Satisfiability:

- Full: undecidable
- Quantifier-free: decidable

Definitions

Definition (Theory of Arrays): The theory of arrays uses Presburger arithmetic for array indices and the parameter element theories $T_{elem}^1, \dots, T_{elem}^m$, for $m > 0$, for its elements.

$$\Sigma_A = \Sigma_{\mathbb{Z}} \cup \bigcup_k \Sigma_{elem}^k \cup \{read, write\}$$

Definitions

Definition (Array Property): A formula of the form

$$(\forall \bar{i}) (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))$$

\bar{i} vector of index variables, $\phi_I(\bar{i})$ index guard, $\phi_V(\bar{i})$ value constraint

Definitions

Definition (Array Property): A formula of the form

$$(\forall \bar{i}) (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))$$

\bar{i} vector of index variables, $\phi_I(\bar{i})$ index guard, $\phi_V(\bar{i})$ value constraint

Form of index guard $\phi_I(\bar{i})$ constrained according to the grammar

iguard \rightarrow iguard \wedge iguard $|$ iguard \vee iguard $|$ atom

atom \rightarrow expr \leq expr $|$ expr $=$ expr

expr \rightarrow *uvar* $|$ pexpr

pexpr \rightarrow \mathbb{Z} $|$ $\mathbb{Z} \cdot$ *evar* $|$ pexpr $+$ pexpr

Definitions

Definition (Array Property): A formula of the form

$$(\forall \bar{i}) (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))$$

\bar{i} vector of index variables, $\phi_I(\bar{i})$ index guard, $\phi_V(\bar{i})$ value constraint

Examples (index guard): $i = 3, i \leq j$

Definitions

Definition (Array Property): A formula of the form

$$(\forall \bar{i}) (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))$$

\bar{i} vector of index variables, $\phi_I(\bar{i})$ index guard, $\phi_V(\bar{i})$ value constraint

Examples (index guard): $i = 3, i \leq j$

Counter-Examples (index guard): $i + 2 \leq j, i < j$

Definitions

Definition (Array Property): A formula of the form

$$(\forall \bar{i}) (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))$$

\bar{i} vector of index variables, $\phi_I(\bar{i})$ index guard, $\phi_V(\bar{i})$ value constraint

The form of a value constraint $\phi_V(\bar{i})$ is also restricted:

Definitions

Definition (Array Property): A formula of the form

$$(\forall \bar{i}) (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))$$

\bar{i} vector of index variables, $\phi_I(\bar{i})$ index guard, $\phi_V(\bar{i})$ value constraint

The form of a value constraint $\phi_V(\bar{i})$ is also restricted:

1. Any occurrence of a quantified index variable in $\phi_V(\bar{i})$ must be as a read into an array.

Definitions

Definition (Array Property): A formula of the form

$$(\forall \bar{i}) (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))$$

\bar{i} vector of index variables, $\phi_I(\bar{i})$ index guard, $\phi_V(\bar{i})$ value constraint

The form of a value constraint $\phi_V(\bar{i})$ is also restricted:

1. Any occurrence of a quantified index variable in $\phi_V(\bar{i})$ must be as a read into an array.
2. Array reads may not be nested.

Definitions

Definition (Array Property): A formula of the form

$$(\forall \bar{i}) (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))$$

\bar{i} vector of index variables, $\phi_I(\bar{i})$ index guard, $\phi_V(\bar{i})$ value constraint

Examples (value constraint): $a[i] = b[i]$, $a[i] \leq b[i]$

Definitions

Definition (Array Property): A formula of the form

$$(\forall \bar{i}) (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))$$

\bar{i} vector of index variables, $\phi_I(\bar{i})$ index guard, $\phi_V(\bar{i})$ value constraint

Examples (value constraint): $a[i] = b[i]$, $a[i] \leq b[i]$

Counter-Examples (value constraint): $(\forall i) [\dots] a\{i \leftarrow 4\}, a[b[i]] \leq j$

Examples of Array Properties

Equality $a = b \Leftrightarrow (\forall i) (a[i] = b[i])$

Bounded Equality $beq(l, u, a, b) \Leftrightarrow (\forall i) (l \leq i \leq u \rightarrow a[i] = b[i])$

Sorted $sorted(l, u, a) \Leftrightarrow (\forall i) (l \leq i \leq j \leq u \rightarrow a[i] \leq b[j])$

$(\exists \text{ array } a) (\exists w, x, y, z, k, l, n \in \mathbb{Z})$

$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$

$\wedge sorted(0, n - 1, a \{k \leftarrow w\} \{l \leftarrow x\}) \wedge sorted(0, n - 1, a \{k \leftarrow y\} \{l \leftarrow z\})$

Examples of Array Properties

Equality $a = b \Leftrightarrow (\forall i) (a[i] = b[i])$

Bounded Equality $beq(l, u, a, b) \Leftrightarrow (\forall i) (l \leq i \leq u \rightarrow a[i] = b[i])$

Sorted $sorted(l, u, a) \Leftrightarrow (\forall i) (l \leq i \leq j \leq u \rightarrow a[i] \leq b[j])$

$(\exists \text{ array } a) (\exists w, x, y, z, k, l, n \in \mathbb{Z})$

$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$

$\wedge sorted(0, n - 1, a \{k \leftarrow w\} \{l \leftarrow x\}) \wedge sorted(0, n - 1, a \{k \leftarrow y\} \{l \leftarrow z\})$

Examples of Array Properties

Equality $a = b \Leftrightarrow (\forall i) (a[i] = b[i])$

Bounded Equality $beq(l, u, a, b) \Leftrightarrow (\forall i) (l \leq i \leq u \rightarrow a[i] = b[i])$

Sorted $sorted(l, u, a) \Leftrightarrow (\forall i) (l \leq i \leq j \leq u \rightarrow a[i] \leq b[j])$

$(\exists \text{ array } a) (\exists w, x, y, z, k, l, n \in \mathbb{Z})$

$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$

$\wedge sorted(0, n - 1, a \{k \leftarrow w\} \{l \leftarrow x\}) \wedge sorted(0, n - 1, a \{k \leftarrow y\} \{l \leftarrow z\})$

Examples of Array Properties

Equality $a = b \Leftrightarrow (\forall i) (a[i] = b[i])$

Bounded Equality $beq(l, u, a, b) \Leftrightarrow (\forall i) (l \leq i \leq u \rightarrow a[i] = b[i])$

Sorted $sorted(l, u, a) \Leftrightarrow (\forall i) (l \leq i \leq j \leq u \rightarrow a[i] \leq b[j])$

$(\exists \text{ array } a) (\exists w, x, y, z, k, l, n \in \mathbb{Z})$

$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$

$\wedge sorted(0, n - 1, a \{k \leftarrow w\} \{l \leftarrow x\}) \wedge sorted(0, n - 1, a \{k \leftarrow y\} \{l \leftarrow z\})$

Examples of Array Properties

Equality $a = b \Leftrightarrow (\forall i) (a[i] = b[i])$

Bounded Equality $beq(l, u, a, b) \Leftrightarrow (\forall i) (l \leq i \leq u \rightarrow a[i] = b[i])$

Sorted $sorted(l, u, a) \Leftrightarrow (\forall i) (l \leq i \leq j \leq u \rightarrow a[i] \leq b[j])$

$(\exists \text{ array } a) (\exists w, x, y, z, k, l, n \in \mathbb{Z})$

$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$

$\wedge sorted(0, n - 1, a \{k \leftarrow w\} \{l \leftarrow x\}) \wedge sorted(0, n - 1, a \{k \leftarrow y\} \{l \leftarrow z\})$

More Definitions

Definition (Array Property Fragment): The array property fragment of T_A consists of all existentially-closed Boolean combinations of array property formulae and quantifier-free T_A -formulae.

More Definitions

Definition (Array Property Fragment): The array property fragment of T_A consists of all existentially-closed Boolean combinations of array property formulae and quantifier-free T_A -formulae.

Definition (Goal of this talk): Present a decision procedure SAT_A which verifies the satisfiability of formulae containing the alternate use of existential and universal quantification (in a restricted manner).

The main idea is to replace universal quantification by a finite conjunction of literals.

Oh No, More Definitions

Definition (Read Set): $\mathcal{R} := \{t \mid a[t] \in \phi \wedge t \text{ not univ. quantified} \}$

Oh No, More Definitions

Definition (Read Set): $\mathcal{R} := \{t \mid a[t] \in \phi \wedge t \text{ not univ. quantified} \}$

Definition (Bounds Set): The bounds set \mathcal{B} for formula ϕ is the set of Presburger arithmetic terms that arise as root pexpr during the parsing of all index guards in ϕ .

Oh No, More Definitions

Definition (Read Set): $\mathcal{R} := \{t \mid a[t] \in \phi \wedge t \text{ not univ. quantified} \}$

Definition (Bounds Set): The bounds set \mathcal{B} for formula ϕ is the set of Presburger arithmetic terms that arise as root pexpr during the parsing of all index guards in ϕ .

Definition (Index Set): $\mathcal{I}_\phi = \begin{cases} \{0\} & \text{if } \mathcal{R} = \mathcal{B} = \emptyset \\ \mathcal{R} \cup \mathcal{B} & \text{otherwise} \end{cases}$

Decision Procedure SAT_A

1. Insert predicate definitions and convert to negation normal form.

2.

$$\frac{\psi[a \{i \leftarrow e\}]}{\psi[b] \wedge b[i] = e \wedge (\forall j)(j \neq i \rightarrow a[j] = b[j])} \text{ for fresh } b \text{ (write)}$$

3.

$$\frac{\psi[(\exists \bar{i})(\phi_I(\bar{i}) \wedge \neg \phi_V(\bar{i}))]}{\psi[\phi_I(\bar{j}) \wedge \neg \phi_V(\bar{j})]} \text{ for fresh } \bar{j} \text{ (exists)}$$

4.

$$\frac{\psi[(\forall \bar{i})(\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))]}{\psi \left[\bigwedge_{\bar{i} \in \mathcal{I}_{\psi_3}^n} (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i})) \right]} \text{ (forall)}$$

5. Replace array reads by uninterpreted function symbols.

Decision Procedure SAT_A

1. Insert predicate definitions and convert to negation normal form.

2.

$$\frac{\psi[a \{i \leftarrow e\}]}{\psi[b] \wedge b[i] = e \wedge (\forall j)(j \neq i \rightarrow a[j] = b[j])} \text{ for fresh } b \text{ (write)}$$

3.

$$\frac{\psi[(\exists \bar{i})(\phi_I(\bar{i}) \wedge \neg \phi_V(\bar{i}))]}{\psi[\phi_I(\bar{j}) \wedge \neg \phi_V(\bar{j})]} \text{ for fresh } \bar{j} \text{ (exists)}$$

4.

$$\frac{\psi[(\forall \bar{i})(\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))]}{\psi \left[\bigwedge_{\bar{i} \in \mathcal{I}_{\psi_3}^n} (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i})) \right]} \text{ (forall)}$$

5. Replace array reads by uninterpreted function symbols.

Decision Procedure SAT_A

1. Insert predicate definitions and convert to negation normal form.

2.

$$\frac{\psi[a \{i \leftarrow e\}]}{\psi[b] \wedge b[i] = e \wedge (\forall j)(j \neq i \rightarrow a[j] = b[j])} \text{ for fresh } b \text{ (write)}$$

3.

$$\frac{\psi[(\exists \bar{i})(\phi_I(\bar{i}) \wedge \neg \phi_V(\bar{i}))]}{\psi[\phi_I(\bar{j}) \wedge \neg \phi_V(\bar{j})]} \text{ for fresh } \bar{j} \text{ (exists)}$$

4.

$$\frac{\psi[(\forall \bar{i})(\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))]}{\psi \left[\bigwedge_{\bar{i} \in \mathcal{I}_{\psi_3}^n} (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i})) \right]} \text{ (forall)}$$

5. Replace array reads by uninterpreted function symbols.

Decision Procedure SAT_A

1. Insert predicate definitions and convert to negation normal form.

2.

$$\frac{\psi[a \{i \leftarrow e\}]}{\psi[b] \wedge b[i] = e \wedge (\forall j)(j \neq i \rightarrow a[j] = b[j])} \text{ for fresh } b \text{ (write)}$$

3.

$$\frac{\psi[(\exists \bar{i})(\phi_I(\bar{i}) \wedge \neg \phi_V(\bar{i}))]}{\psi[\phi_I(\bar{j}) \wedge \neg \phi_V(\bar{j})]} \text{ for fresh } \bar{j} \text{ (exists)}$$

4.

$$\frac{\psi[(\forall \bar{i})(\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))]}{\psi \left[\bigwedge_{\bar{i} \in \mathcal{I}_{\psi_3}^n} (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i})) \right]} \text{ (forall)}$$

5. Replace array reads by uninterpreted function symbols.

Decision Procedure SAT_A

1. Insert predicate definitions and convert to negation normal form.

2.

$$\frac{\psi[a \{i \leftarrow e\}]}{\psi[b] \wedge b[i] = e \wedge (\forall j)(j \neq i \rightarrow a[j] = b[j])} \text{ for fresh } b \text{ (write)}$$

3.

$$\frac{\psi[(\exists \bar{i})(\phi_I(\bar{i}) \wedge \neg \phi_V(\bar{i}))]}{\psi[\phi_I(\bar{j}) \wedge \neg \phi_V(\bar{j})]} \text{ for fresh } \bar{j} \text{ (exists)}$$

4.

$$\frac{\psi[(\forall \bar{i})(\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))]}{\psi \left[\bigwedge_{\bar{i} \in \mathcal{I}_{\psi_3}^n} (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i})) \right]} \text{ (forall)}$$

5. Replace array reads by uninterpreted function symbols.

Decision Procedure SAT_A

1. Insert predicate definitions and convert to negation normal form.

2.

$$\frac{\psi[a \{i \leftarrow e\}]}{\psi[b] \wedge b[i] = e \wedge (\forall j)(j \neq i \rightarrow a[j] = b[j])} \text{ for fresh } b \text{ (write)}$$

3.

$$\frac{\psi[(\exists \bar{i})(\phi_I(\bar{i}) \wedge \neg \phi_V(\bar{i}))]}{\psi[\phi_I(\bar{j}) \wedge \neg \phi_V(\bar{j})]} \text{ for fresh } \bar{j} \text{ (exists)}$$

4.

$$\frac{\psi[(\forall \bar{i})(\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))]}{\psi \left[\bigwedge_{\bar{i} \in \mathcal{I}_{\psi_3}^n} (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i})) \right]} \text{ (forall)}$$

5. Replace array reads by uninterpreted function symbols.

Decision Procedure SAT_A

1. Insert predicate definitions and convert to negation normal form.

2.

$$\frac{\psi[a \{i \leftarrow e\}]}{\psi[b] \wedge b[i] = e \wedge (\forall j)(j \neq i \rightarrow a[j] = b[j])} \text{ for fresh } b \text{ (write)}$$

3.

$$\frac{\psi[(\exists \bar{i})(\phi_I(\bar{i}) \wedge \neg \phi_V(\bar{i}))]}{\psi[\phi_I(\bar{j}) \wedge \neg \phi_V(\bar{j})]} \text{ for fresh } \bar{j} \text{ (exists)}$$

4.

$$\frac{\psi[(\forall \bar{i})(\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))]}{\psi \left[\bigwedge_{\bar{i} \in \mathcal{I}_{\psi_3}^n} (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i})) \right]} \text{ (forall)}$$

5. Replace array reads by uninterpreted function symbols.

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge \text{sorted}(0, n - 1, a \{k \leftarrow w\} \{l \leftarrow x\})$$

$$\wedge \text{sorted}(0, n - 1, a \{k \leftarrow y\} \{l \leftarrow z\})$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge \text{sorted}(0, n - 1, a \{k \leftarrow w\} \{l \leftarrow x\})$$

$$\wedge \text{sorted}(0, n - 1, a \{k \leftarrow y\} \{l \leftarrow z\})$$

Is this array property formula satisfiable?

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

$$c[k + 1] \leq c[l]$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

$$c[k + 1] \leq c[l]$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

$$c[k + 1] \leq c[l] = x$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

$$c[k + 1] \leq c[l] = x$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

$$c[k + 1] \leq c[l] = x < y$$

Example

Consider this array property formula from the examples:

$$\begin{aligned} & w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1 \\ & \wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j]) \\ & \wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j]) \\ & \wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x \\ & \wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w \\ & \wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z \\ & \wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y \end{aligned}$$

$$c[k + 1] \leq c[l] = x < y$$

Example

Consider this array property formula from the examples:

$$\begin{aligned} & w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1 \\ & \wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j]) \\ & \wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j]) \\ & \wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x \\ & \wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w \\ & \wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z \\ & \wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y \end{aligned}$$

$$c[k + 1] \leq c[l] = x < y = d[k]$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

$$c[k + 1] \leq c[l] = x < y = d[k]$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

$$c[k + 1] \leq c[l] = x < y = d[k] \leq d[k + 1]$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

$$c[k + 1] \leq c[l] = x < y = d[k] \leq d[k + 1]$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow c[i] \leq c[j])$$

$$\wedge (\forall i, j)(0 \leq i \leq j \leq n - 1 \rightarrow e[i] \leq e[j])$$

$$\wedge (\forall i)(i \neq l \rightarrow b[i] = c[i]) \wedge c[l] = x$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = b[i]) \wedge b[k] = w$$

$$\wedge (\forall i)(i \neq l \rightarrow d[i] = e[i]) \wedge e[l] = z$$

$$\wedge (\forall i)(i \neq k \rightarrow a[i] = d[i]) \wedge d[k] = y$$

$$c[k + 1] \leq c[l] = x < y = d[k] \leq d[k + 1] \not\downarrow$$

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$

$$\wedge \text{sorted}(0, n - 1, a \{k \leftarrow w\} \{l \leftarrow x\})$$

$$\wedge \text{sorted}(0, n - 1, a \{k \leftarrow y\} \{l \leftarrow z\})$$

Is this array property formula satisfiable?

Example

Consider this array property formula from the examples:

$$w < x < y < z \wedge 0 < k < l < n \wedge l - k > 1$$
$$\wedge \text{sorted}(0, n - 1, a \{k \leftarrow w\} \{l \leftarrow x\})$$
$$\wedge \text{sorted}(0, n - 1, a \{k \leftarrow y\} \{l \leftarrow z\})$$

Is this array property formula satisfiable? No.

Proof Sketch

on the board

Decision Procedure SAT_A

1. Insert predicate definitions and convert to negation normal form.

2.

$$\frac{\psi[a \{i \leftarrow e\}]}{\psi[b] \wedge b[i] = e \wedge (\forall j)(j \neq i \rightarrow a[j] = b[j])} \text{ for fresh } b \text{ (write)}$$

3.

$$\frac{\psi[(\exists \bar{i})(\phi_I(\bar{i}) \wedge \neg \phi_V(\bar{i}))]}{\psi[\phi_I(\bar{j}) \wedge \neg \phi_V(\bar{j})]} \text{ for fresh } \bar{j} \text{ (exists)}$$

4.

$$\frac{\psi[(\forall \bar{i})(\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i}))]}{\psi \left[\bigwedge_{\bar{i} \in \mathcal{I}_{\psi_3}^n} (\phi_I(\bar{i}) \rightarrow \phi_V(\bar{i})) \right]} \text{ (forall)}$$

5. Replace array reads by uninterpreted function symbols.

Complexity

Theorem (NP-Complete): If satisfiability of quantifier-free ($T_{EUF} \cup T_{\mathbb{Z}} \cup \bigcup_k T_{elem}^k$)-formulae is in NP, then for the subfragment of the array property fragment of $T_A^{\{elem_k\}_k}$ in which all array property formulae have height at most N, satisfiability is NP-complete.

Remarks

Theorem (Fragment Extension): Extending the array property fragment with any of

- nested reads (e.g. $a_1[a_2[i]]$, where i is universally quantified)
- array reads by a universally quantified variable in the index guard
- general Presburger arithmetic expressions over universally quantified index variables (even just addition of 1, e.g. $i + 1$) in the index guard or in the value constraint

results in a fragment of $T_A^{\mathbb{Z}}$ for which satisfiability is undecidable.

Related Topics

- Maps (array theory with uninterpreted indices)

Related Topics

- Maps (array theory with uninterpreted indices)
- πVC (“Prove It” Verifying Compiler)

Related Topics

- Maps (array theory with uninterpreted indices)
- πVC (“Prove It” Verifying Compiler)
- πVC implementation of SAT_A

Related Topics

- Maps (array theory with uninterpreted indices)
- πVC (“Prove It” Verifying Compiler)
- πVC implementation of SAT_A

Literature:

Aaron R. Bradley, Zohar Manna, and Henny B. Sipma.

What’s Decidable About Arrays?, Stanford, CA 94305-9045

<http://theory.stanford.edu/~arbrad/>

Google: “Aaron Bradley”

Game Over

Any questions?