Combination of Theories

Seminar „Decision Procedures“

Dennis Schmidt

08. July 2016

Agenda

- Introduction
- Basic definitions
- Nelson-Oppen Procedure
  - Algorithms
  - Proof of Correctness

Introduction

- Deciding upon satisfiability of the combination of multiple theories:
  - Linear arithmetic and uninterpreted functions
    $$(f(x_1, 0) \geq x_3) \wedge (f(x_2, 0) \leq x_3)$$
  - Bit-vectors and uninterpreted functions
    $$f(a[32], b[1]) = f(b[32], a[1]) \wedge a[32] = b[32]$$
  - Arrays and linear arithmetic
    $$x = v\{i \leftarrow e\}[j] \wedge y = v[j] \wedge x > e \wedge x > y$$

Basic definitions

- First-order logic defines theories, with:
  - Shared by all theories
    - Logical symbols: e.g. $\wedge, \vee, \neg, \rightarrow, \leftrightarrow, =, \neq, \exists, \forall, (,)$
    - Logical axioms: Define the logical symbols
  - Theory specific
    - Nonlogical symbols: e.g. $+, -, *, f, g$
    - Nonlogical axioms: Define the non logical symbols
  - Variables
  - Syntax

Basic definitions

- A signature ∑, is a set of nonlogical symbols
- A theory T which is defined over a signature ∑ is called ∑-theory
- (Assume T is a ∑-theory)

Basic definitions

- **Definition (theory combination)**
  Given two theories T_1 and T_2 with signatures Σ_1 and Σ_2, respectively, the theory combination $T_1 \oplus T_2$ is a $(\Sigma_1 \cup \Sigma_2)$-theory defined by the axiom set $T_1 \cup T_2$.

Basic definitions

- ***T*-satisfiable**: There **exists** an **interpretation** that satisfies both φ and T.

- ***T*-valid** ($T \models \varphi$): **All interpretations** that satisfy T also satisfy φ.

Basic definitions – Example T-satisfiable/valid

- $\Sigma := \{0,1,+\}$
- $\varphi := \exists x.\, x + 0 = 1$
- Axioms for the Σ-theory T
  1. $\forall x, y.\, x + y = y + x$

→T-satisfiable

→Not T-valid

- $Structure\ S$:
  - $0\ and\ 1$ are interpreted as $0\ and\ 1$ in $\mathbb{N}_0$.
  - $+$ **means addition**

- $Structure\ S'$:
  - $0\ and\ 1$ are interpreted as $0\ and\ 1$ in $\mathbb{N}_0$.
  - $+$ **means multiplication**

Basic definitions – Example T-satisfiable/valid

- $\Sigma := \{0,1,+\}$
- $\varphi := \exists x.\, x + 0 = 1$
- Axioms for the Σ-theory T
  1. $\forall x, y.\, x + y = y + x$
  2. $\forall x.\, 0 + x = x$

→T-satisfiable

→T-valid

- $Structure\ S$:
  - $0\ and\ 1$ are interpreted as $0\ and\ 1$ in $\mathbb{N}_0$.
  - $+$ **means addition**

- $Structure\ S'$:
  - $0\ and\ 1$ are interpreted as $0\ and\ 1$ in $\mathbb{N}_0$.
  - $+$ **means multiplication**

Basic definitions

- **Definition (The theory combination problem)**
  Let φ be a $\Sigma_1 \cup \Sigma_2$ formula. The theory combination problem is to decide whether φ is $T_1 \oplus T_2$ valid:

$$T_1 \oplus T_2 \vDash \varphi$$

- **Definition (convex theory)**
  A Σ-theory T is convex if for every conjunctive Σ-formula φ
  $(\varphi \Rightarrow \bigvee_{i=1}^{n} x_i = y_i)$ is T-valid for some finite $n > 1 \Rightarrow$
  $(\varphi \Rightarrow x_i = y_i)$ is T-valid for some $i \in \{1, \dots, n\}$
  where $x_i, y_i, for\ i \in \{1, \dots, n\}$, are some variables.

Basic definitions – Example Convex Theory

- Linear arithmetic over $\mathbb{R}$ is convex

$$x \leq 3 \wedge x \geq 3 \Rightarrow x = 3$$

- Linear arithmetic over $\mathbb{Z}$ is not convex: while

$$x_1 = 1 \wedge x_2 = 2 \wedge 1 \leq x_3 \wedge x_3 \leq 2 \Rightarrow (x_3 = x_1 \vee x_3 = x_2)$$

is valid, neither

$$x_1 = 1 \wedge x_2 = 2 \wedge 1 \leq x_3 \wedge x_3 \leq 2 \Rightarrow x_3 = x_1$$

nor

$$x_1 = 1 \wedge x_2 = 2 \wedge 1 \leq x_3 \wedge x_3 \leq 2 \Rightarrow x_3 = x_2$$

is valid.

Nelson-Oppen – Restrictions

- Theories must meet the following restrictions to be decidable in combination:

1. $T_1, \ldots, T_n$ are quantifier-free first-order theories with equality.
2. There is a decision procedure for each of the theories $T_1, \ldots, T_n$.
3. The signatures are disjoint, i.e., for all $1 \leq i < j \leq n, \Sigma_i \cap \Sigma_j = \emptyset$.
4. $T_1, \ldots, T_n$ are theories that are interpreted over an infinite domain

Nelson-Oppen – Algorithm 1

- Input: A convex formula φ that **combines convex theories**, with previous restrictions.
- Output: “Satisfiable“ if φ is satisfiable, and “Unsatisfiable” otherwise.

1. **Purification**: Purify φ into $F_1, \dots, F_n$.

   e.g. $x_1 \leq f(x_1) \equiv x_1 \leq a \wedge a = f(x_1)$

2. Apply the decision procedure for T_i to F_i. If there exists i such that F_i is unsatisfiable in T_i, return “Unsatisfiable”.
3. **Equality propagation**: If there exist i, j such that F_i T_i-implies an equality between variables of φ that is not T_i-implied by F_j, add this equality to F_j and go to step 2.
4. Return “Satisfiable”

Nelson-Oppen – Example 1 (Purification)

Consider the formula:

$$(f(x_1, 0) \geq x_3) \wedge (f(x_2, 0) \leq x_3) \wedge (x_1 \geq x_2) \wedge (x_2 \geq x_1) \wedge (x_3 - f(x_1, 0) \geq 1)$$

Purification results:

$$\left(\underbrace{f(x_1, 0) \geq x_3}_{(a_1 \geq x_3) \wedge (a_0 = 0) \wedge (a_1 = f(x_1, a_0))}\right) \wedge \left(\underbrace{f(x_2, 0) \leq x_3}_{(a_2 \leq x_3) \wedge (a_3 = 0) \wedge (a_2 = f(x_2, a_3))}\right) \wedge (x_1 \geq x_2)$$
$$\wedge (x_2 \geq x_1) \wedge \left(\underbrace{x_3 - f(x_1, 0) \geq 1}_{(a_4 = 0) \wedge (a_5 = f(x_1, a_4)) \wedge (x_3 - a_5 \geq 1)}\right)$$

$$\equiv (a_1 \geq x_3) \wedge (a_0 = 0) \wedge (a_2 \leq x_3) \wedge (x_1 \geq x_2) \wedge (x_2 \geq x_1) \wedge (x_3 - a_1 \geq 1)$$
$$\wedge (a_1 = f(x_1, a_0)) \wedge (a_2 = f(x_2, a_0))$$

Nelson-Oppen – Example 1 (Equality Prop.)

$$(a_1 \geq x_3) \wedge (a_0 = 0) \wedge (a_2 \leq x_3) \wedge (x_1 \geq x_2) \wedge (x_2 \geq x_1) \wedge (x_3 - a_1 \geq 1)$$
$$\wedge \big(a_1 = f(x_1, a_0)\big) \wedge \big(a_2 = f(x_2, a_0)\big)$$

$\boldsymbol{F_1}$ $(\boldsymbol{Arithmetic\ over}\ \mathbb{R})$	$\boldsymbol{F_2}$ $(\boldsymbol{EUF})$
$a_1 \geq x_3$ $a_0 = 0$ $a_2 \leq x_3$ $x_1 \geq x_2$ $x_2 \geq x_1$ $x_3 - a_1 \geq 1$	$a_1 = f(x_1, a_0)$ $a_2 = f(x_2, a_0)$
$\boldsymbol{x_1 = x_2}$ $a_1 = a_2$ $\boldsymbol{a_1 = x_3}$ $\rightarrow \boldsymbol{unsatisfiable}$	$x_1 = x_2$ $\boldsymbol{a_1 = a_2}$ $a_1 = x_3$

Nelson-Oppen – Algorithm 2

- Input: A formula φ that **combines theories**, with previous restrictions.
- Output: "Satisfiable" if φ is satisfiable, and "Unsatisfiable" otherwise.

1. **Purification**: Purify φ into $F_1, \dots, F_n$.
2. Apply the decision procedure for T_i to F_i. If there exists i such that F_i is unsatisfiable in T_i, return "Unsatisfiable".
3. **Equality propagation**: If there exist i, j such that F_i T_i-implies an equality between variables of φ that is not T_i-implied by F_j, add this equality to F_j and go to step 2.
4. **Splitting**: If there exists i such that
   - $F_i \Rightarrow (x_1 = y_1 \vee \cdots \vee x_k = y_k)$ and
   - $\forall j \in \{1, \dots, k\}. F_i \not\Rightarrow x_j = y_j$,

   Then apply Nelson-Oppen recursively to: $\varphi' \wedge x_1 = y_1, \dots, \varphi' \wedge x_k = y_k$

   If any of these subproblems is satisfiable, return "Satisfiable". Otherwise return "Unsatisfiable"
5. Return "Satisfiable"

Nelson-Oppen – Example 2 (Purification)

Consider the formula:

$$(1 \leq x) \wedge (x \leq 2) \wedge p(x) \wedge \neg p(1) \wedge \neg p(2)$$

Purification:

$$(1 \leq x) \wedge (x \leq 2) \wedge p(x) \wedge \underbrace{\neg p(1)}_{\neg p(a_0) \wedge (a_0 = 1)} \wedge \underbrace{\neg p(2)}_{\neg p(a_1) \wedge (a_1 = 2)}$$

$$\equiv (1 \leq x) \wedge (x \leq 2) \wedge (a_0 = 1) \wedge (a_1 = 2) \wedge p(x) \wedge \neg p(a_0) \wedge \neg p(a_1)$$

Nelson-Oppen – Example 2 (Splitting)

$$(1 \leq x) \wedge (x \leq 2) \wedge (a_0 = 1) \wedge (a_1 = 2) \wedge p(x) \wedge \neg p(a_0) \wedge \neg p(a_1)$$

$\boldsymbol{F_1}$ $(\boldsymbol{Arithmetic\ over}\ \mathbb{Z})$	$\boldsymbol{F_2}$ $(\boldsymbol{EUF})$
$1 \leq x$ $x \leq 2$ $a_0 = 1$ $a_1 = 2$	$p(x)$ $\neg p(a_0)$ $\neg p(a_1)$
$\boldsymbol{x = 1 \vee x = 2}$	

Nelson-Oppen – Example 2 (Equality Prop.)

$\boldsymbol{F_1}$ $\boldsymbol{(Arithmetic\ over\ \mathbb{Z})}$	$\boldsymbol{F_2}$ $\boldsymbol{(EUF)}$
$1 \leq x$ $x \leq 2$ $a_0 = 1$ $a_1 = 2$	$p(x)$ $\neg p(a_0)$ $\neg p(a_1)$
$\boldsymbol{x = 1 \vee x = 2}$	

$\boldsymbol{F_1}$ $\boldsymbol{(Arithmetic\ over\ \mathbb{Z})}$	$\boldsymbol{F_2}$ $\boldsymbol{(EUF)}$
$1 \leq x$ $x \leq 2$ $a_0 = 1$ $a_1 = 2$	$p(x)$ $\neg p(a_0)$ $\neg p(a_1)$
$x = 1$ $\boldsymbol{x = a_0}$	$x = a_0$ $\rightarrow \boldsymbol{unsat.}$

$\boldsymbol{F_1}$ $\boldsymbol{(Arithmetic\ over\ \mathbb{Z})}$	$\boldsymbol{F_2}$ $\boldsymbol{(EUF)}$
$1 \leq x$ $x \leq 2$ $a_0 = 1$ $a_1 = 2$	$p(x)$ $\neg p(a_0)$ $\neg p(a_1)$
$x = 2$ $\boldsymbol{x = a_1}$	$x = a_1$ $\rightarrow \boldsymbol{unsat.}$

Nelson-Oppen – Proof of Correctness (⇒)

- We proof the correctness of Algorithm 1 for convex theories and for conjunctions of theory literals.
- Without proof: $\varphi \equiv \bigwedge_i F_i$
- **Theorem 1:** Algorithm 1 returns "unsatisfiable" if and only if its input formula φ is unsatisfiable in the combined theory.
  - Soundness: Assume φ is satisfiable in the combined theory.
    (⇒) Let α be a satisfying assignment of φ.
    Let A be the set of auxiliary variables added during purification.
    → As $\varphi \equiv \bigwedge_i F_i$ in the combined theory, we can extend α to an assignment α' that includes also the variables A.

Nelson-Oppen – Proof of Correctness (⇒)

- **Theorem 1:** Algorithm 1 returns "unsatisfiable" if and only if its input formula φ is unsatisfiable in the combined theory.
  - Soundness: (⇒) Assume φ is satisfiable in the combined theory.
    Let α be a satisfying assignment of φ.
    Let A be the set of auxiliary variables added during purification.
    → As $\varphi \equiv \bigwedge_i F_i$ in the combined theory, we can extend α to an assignment α' that includes also the variables A.

$$(x_1 \geq x_2) \wedge (x_2 \geq x_1) \wedge (f(x_1, 0) \leq x_3) \wedge (x_3 - f(x_1, 0) \geq 1)$$

$$\alpha = \{x_1 = 1, x_2 = 1, x_3 = 3, f(x_1, 0) = 2\}$$

$$(1 \geq 1) \wedge (1 \geq 1) \wedge (2 \leq 3) \wedge (3 - 2 \geq 1) \rightarrow true$$

$$(x_1 \geq x_2) \wedge (x_2 \geq x_1) \wedge (\boldsymbol{a_1} \leq x_3) \wedge (\boldsymbol{a_0} = \boldsymbol{0}) \wedge (\boldsymbol{a_1} = f(x_1, a_0)) \wedge (x_3 - \boldsymbol{a_1} \geq 1)$$

$$\boldsymbol{A} = \{\boldsymbol{a_0}, \boldsymbol{a_1}\} \qquad \alpha' = \{x_1 = 1, x_2 = 1, x_3 = 3, f(x_1, 0) = 2, \boldsymbol{a_0} = \boldsymbol{0}, \boldsymbol{a_1} = \boldsymbol{2}\}$$

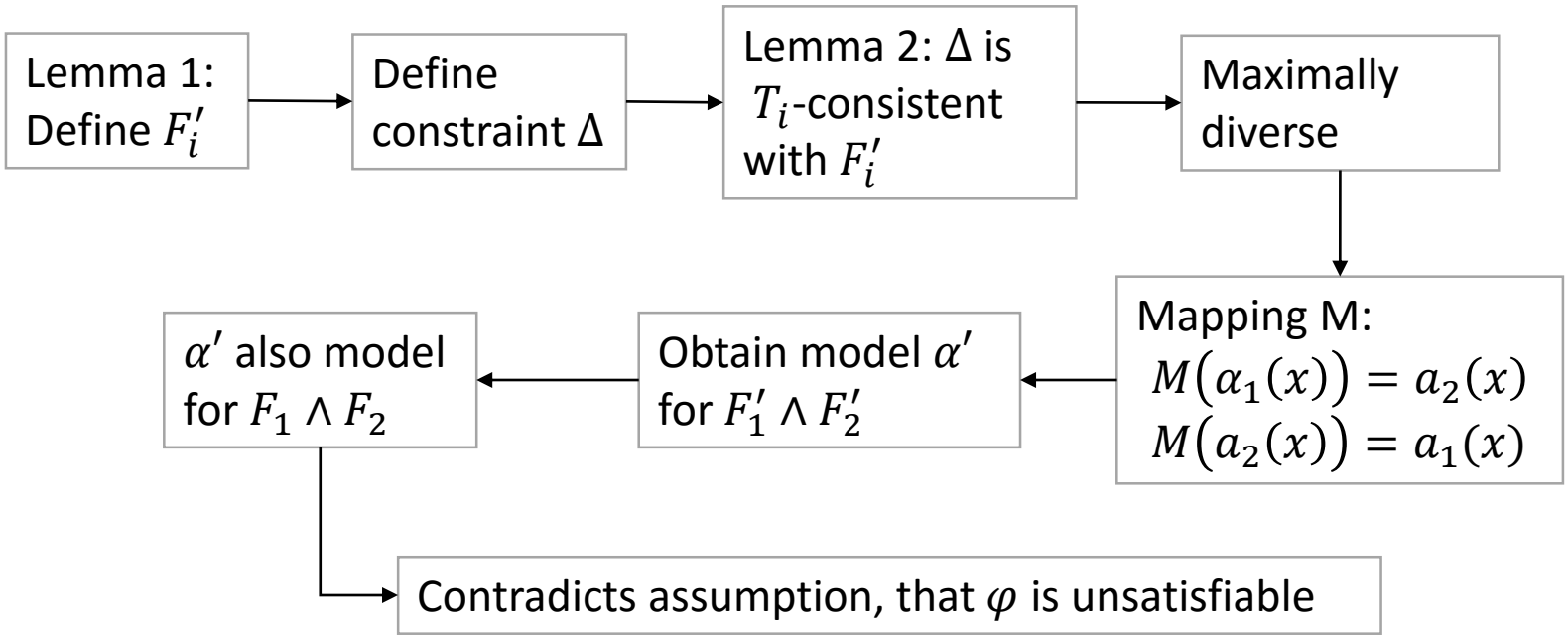
$$(1 \geq 1) \wedge (1 \geq 1) \wedge (2 \leq 3) \wedge (0 = 0) \wedge (2 = 2) \wedge (3 - 2 \geq 1) \rightarrow true$$

Nelson-Oppen – Proof of Correctness

- The algorithm always terminates, as there are only finite many equalities over the variables in the formula → finite many iterations.
- After showing the soundness (⇒) we need to show the completeness (⇐) of the algorithm.

Nelson-Oppen – Proof of Correctness (⇐)

- We show: the algorithm returns “unsatisfiable” if φ is unsatisfiable.
- Assume falsely the Algorithm returns “satisfiable”



Nelson-Oppen – Proof of Correctness (⇐)

- **Lemma 1:** Let F_i' denote the formula F_i upon termination of Algorithm 1. Upon termination with the answer “satisfiable”, any equality between φ's variables that is implied by any of the F_i' is also implied by all F_j' for any j.
  - Follows from equality propagation

Nelson-Oppen – Proof of Correctness (⇐)

- Let $E_1, \dots, E_m$ be a set of equivalence classes of variables in φ, such that x and y are in the same class iff F_1' implies $x = y$ in T_1.
  Due to *lemma 1*, $x, y \in E_i$ for some i iff $x = y$ is T_2-implied by F_2'.
  For $i \in \{1, \dots, m\}$, let r_i be an element of E_i.
  We now define a constraint **Δ** that forces all variables that are not implied to be equal to be different:

$$\Delta \doteq \bigwedge_{i \neq j} r_i \neq r_j$$

Example: $F_1' \coloneqq x_1 = y_1 \wedge x_2 = y_2 \wedge x_2 = y_3 \wedge x_3 < y_4$

$F_2' \coloneqq x_1 = y_1 \wedge x_2 = y_2 \wedge x_2 = y_3 \wedge f(x_3) = g(y_3)$

$E_1 = \{x_1, y_1\}, E_2 = \{x_2, y_2, x_3\}, E_3 = \{x_3\}, E_4 = \{y_4\}$

$\Delta \equiv x_1 \neq x_2 \wedge x_1 \neq x_3 \wedge x_1 \neq y_4 \wedge x_2 \neq x_3 \wedge x_2 \neq y_4 \wedge x_3 \neq y_4$

Nelson-Oppen – Proof of Correctness (⇐)

- **Lemma 2:** Given that both T_1 and T_2 have an infinite domain and are convex, Δ is T_1-consistent with F_1' and T_2-consistent with F_2'.
  - Let x and y be two variables that are not implied to be equal.
  - Due to **convexity**, they do not have to be equal to satisfy F_i'.
  - As the **domain is infinite**, there are always values left in the domain that we can choose to make x and y different.

$F_1' \equiv x_1 = 1 \wedge x_2 = 2 \wedge 1 \leq x_3 \wedge x_3 \leq 2$
$\Delta \equiv x_1 \neq x_2 \wedge x_1 \neq x_3 \wedge x_2 \neq x_3$
$Non\ convex\ \mathbb{Z}: \mathrm{F}_1' \wedge \Delta \rightarrow unsatisfiable$
$Convex\ \mathbb{R}: F_1' \wedge \Delta \rightarrow satisfiable$

$F_1' \equiv x_1 > 4 \wedge x_2 > 4$
$\Delta \equiv x_1 \neq x_2$
$Domain\ D = \{1,2,3,4,5\}: F_1' \wedge \Delta \rightarrow unsat.$
$Domain\ \mathbb{R}: \ F_1' \wedge \Delta \rightarrow satisfiable$

Nelson-Oppen – Proof of Correctness (⇐)

- Using *lemma 2*, we say that there are satisfying assignments α_1 and α_2 for $F_1' \wedge \Delta$ and $F_2' \wedge \Delta$ in T_1 and T_2. These assignment are **maximally diverse**, only variables which are implied to be equal get equal values assigned by α_1 or α_2.

- Example:
$$F_1' \coloneqq x_1 = y_1 \wedge x_2 = y_2 \wedge x_2 = y_3 \wedge \cdots$$
$$F_2' \coloneqq x_1 = y_1 \wedge x_2 = y_2 \wedge x_2 = y_3 \wedge \cdots$$
$$E_1 = \{x_1, y_1\}, E_2 = \{x_2, y_2, x_3\}$$
$$\Delta \equiv x_1 \neq x_2$$
$$F_1' \wedge \Delta \equiv x_1 = y_1 \wedge x_2 = y_2 \wedge x_2 = y_3 \wedge x_1 \neq x_2 \wedge \cdots$$
$$F_2' \wedge \Delta \equiv x_1 = y_1 \wedge x_2 = y_2 \wedge x_2 = y_3 \wedge x_1 \neq x_2 \wedge \cdots$$

Nelson-Oppen – Proof of Correctness (⇐)

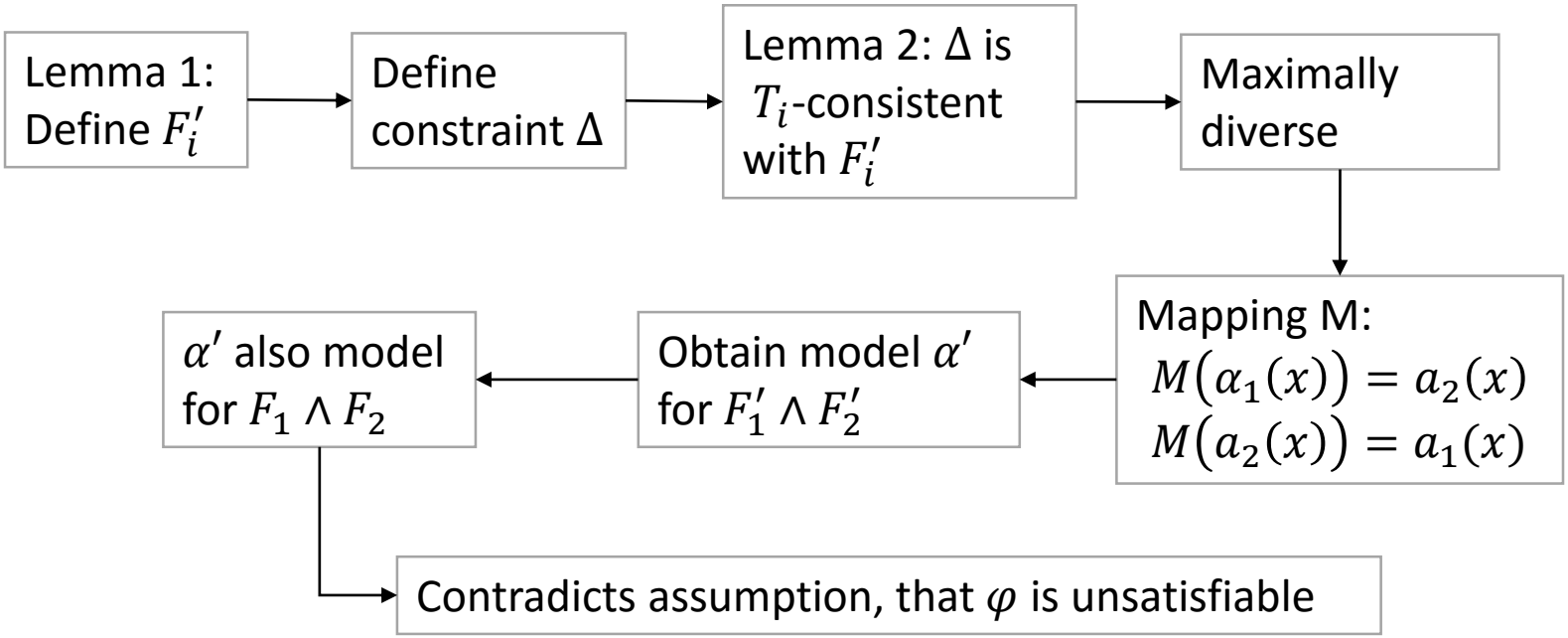
- We build a (isomorphism) mapping M from domain elements to domain elements. ($\alpha_2(x)\ mapped\ to\ a_1(x)$)
- For example:
  - $F_1 \equiv x = y,\ \mathrm{F}_2 \equiv f(x) = g(y)$
  - Implied (and propagated) equality: $x = y$
  - Possible variable assignment for $F_1' \wedge \Delta$ and $F_2' \wedge \Delta$:
    - $\alpha_1 = \{x \to D_1, y \to D_1\}$
      $\alpha_2 = \{x \to D_2, y \to D_2\}$
    - D_1 and D_2 are some domain elements

  → Isomorphism M: $M(D_1) = D_2$

Nelson-Oppen – Proof of Correctness (⇐)

- Using the mapping M, we can obtain a model α' for $F_1' \wedge F_2'$, by adjusting the symbols in F_2'.
  - This is possible as T_1 and T_2 do not share any non logical symbols.
- Continue example:
  - $F_1 \equiv x = y,\ \mathrm{F}_2 \equiv f(x) = g(y)$
  - Isomorphism M: $M(D_1) = D_2$
  - We can construct an interpretation for the non logical symbols f and g:
    - $F(D_1) = D_3, G(D_1) = D_3$

Nelson-Oppen – Proof of Correctness (⇐)



- As F_i' implies F_i in T_i, α' is also model for $F_1 \wedge F_2$ in the combined theory, which contradicts our assumption that φ is unsatisfiable.

Any questions?

Further knowledge

- **"Efficient satisfiability modulo theories via delayed theory combination.“** by Bozzano, Marco, et al.

  - Each pair of shared variables is encoded with a new Boolean variable.
  - Then, the SAT solver begins to assign values (arbitrary at first) to the new variables.
  - After every such assignment, the current partial assignment is sent to a theory solver.
  - If any one of the theory solvers finds a conflict with the current assignment to the other literals, it leads to backtracking.
  - Otherwise, the formula is declared satisfiable.

Further knowledge

- **"Efficient satisfiability modulo theories via delayed theory combination.“** by Bozzano, Marco, et al.
  - Advantage:
    - Each theory can be solved separately (no equality propagation)
    - Only a small amount of information has to be shared between the theory solvers

Further knowledge

- **"Model-based theory combination."** by de Moura, Leonardo, and Nikolaj Bjørner.

  - Making the equalities part of the model
  - Attempting to compute a consistent assignment to the theory variables that is as diverse as possible.
  - The equalities are then decided upon by following the assignment to the theory variables.

Thank You!