Representing & Reasoning with Qualitative Preferences: Tools and Applications

Ganesh Ram Santhanam[1], Samik Basu[1] & Vasant Honavar[2]
[1]Iowa State University, Ames, IA 50011
[2] Penn State University, University Park, PA 16802
gsanthan@iastate.edu

Outline

I. Qualitative Preference Languages
   - Representation : Syntax of languages CP-nets, TCP-nets, CI-nets, CP-Theories

II. Qualitative Preference Languages
   - Ceteris Paribus semantics: the induced preference graph (IPG)
   - Reasoning: Consistency, Dominance, Ordering, Equivalence & Subsumption
   - Complexity of Reasoning

III. Practical aspects: Preference Reasoning via Model Checking
   - From ceteris paribus semantics (IPG) to Kripke structures
   - Specifying and verifying properties in temporal logic
   - Translating Reasoning Tasks into Temporal Logic Properties

Outline

IV. Applications

- *Engineering*: Civil, Software (SBSE, RE, Services), Aerospace, Manufacturing
- *Security*: Credential disclosure, Cyber-security
- *Algorithms*: Search, Stable Marriage, Allocation, Planning, Recommender systems
- *Environmental applications*: Risk Assessment, Policy decisions, Environmental impact, Computational Sustainability

V. iPref-R Tool

- A tool that does well in practice for a known hard problem
- Architecture
- Demo
- Use of iPref-R in Security, Software Engineering

Broad view of Decision Theory

What is a *decision*?

Choosing from a set of *alternatives A*

Choice function: $\Phi(A) \subseteq A$

How are alternatives described?

What influences choice of an agent?

- preferences, uncertainty, risk

Can decisions be automated?

What happens if there are multiple agents?

- conflicting preferences and choices

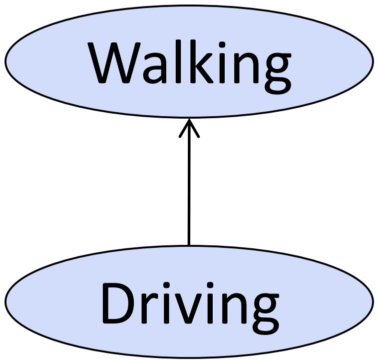
"I prefer walking over driving to work"

There is a 50% chance of snow. Walking may not be good after all.

Qualitative Preferences

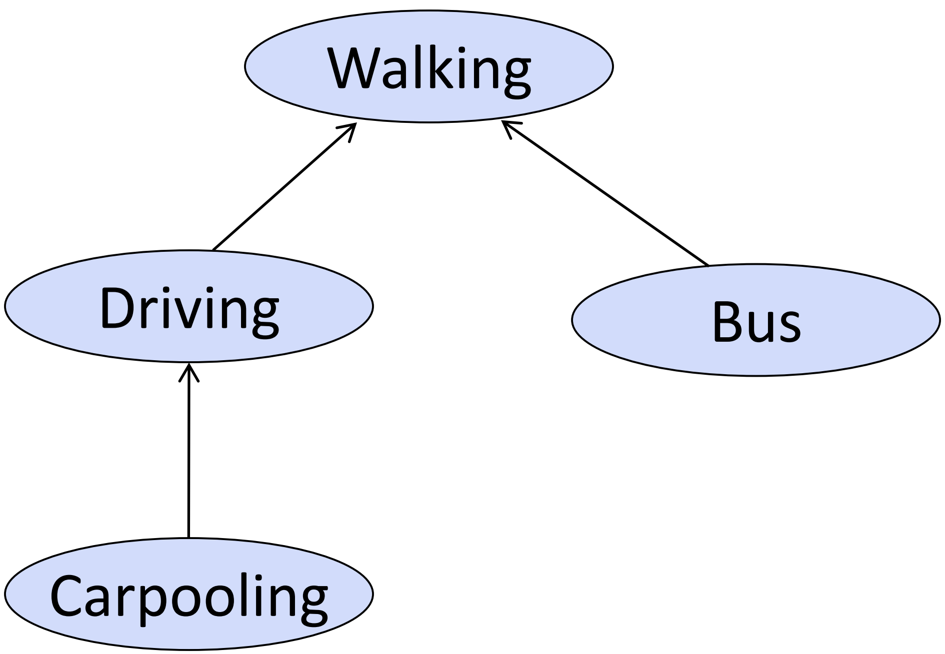
Qualitative

Quantitative



Walking = 0.7; Driving = 0.3

Walking = 0.6; Driving = 0.4



?

Loss of information regarding the incompleteness / imprecision of user preferences

Representation: Alternatives are Multi-attributed

Course selection - which course to take?

	572	509	586
Subject?	AI	SE	NW
Instructor?	Gopal	Tom	Bob
# Credits?	4	3	3

- Preference variables or attributes used to describe the domain
- Alternatives are assignments to preference variables
  - α = (instructor = Gopal, area = AI, credits = 3)
- $\alpha \succ \beta$ denotes that α is preferred to β

Qualitative Preference Languages

Qualitative preferences

- Unconditional Preferences
  - TUP-nets [Santhanam et al., 2010]
- Conditional Preferences
  - CP-nets [Boutilier et al. 1997,2002]
  - Models dependencies
- Relative Importance
  - TCP-nets [Brafman et al. 2006]
  - CI-nets [Bouveret et al. 2009]

AI $\succ_{area}$ SE

SE : Tom $\succ_{instructor}$ Gopal
AI : Gopal $\succ_{instructor}$ Tom

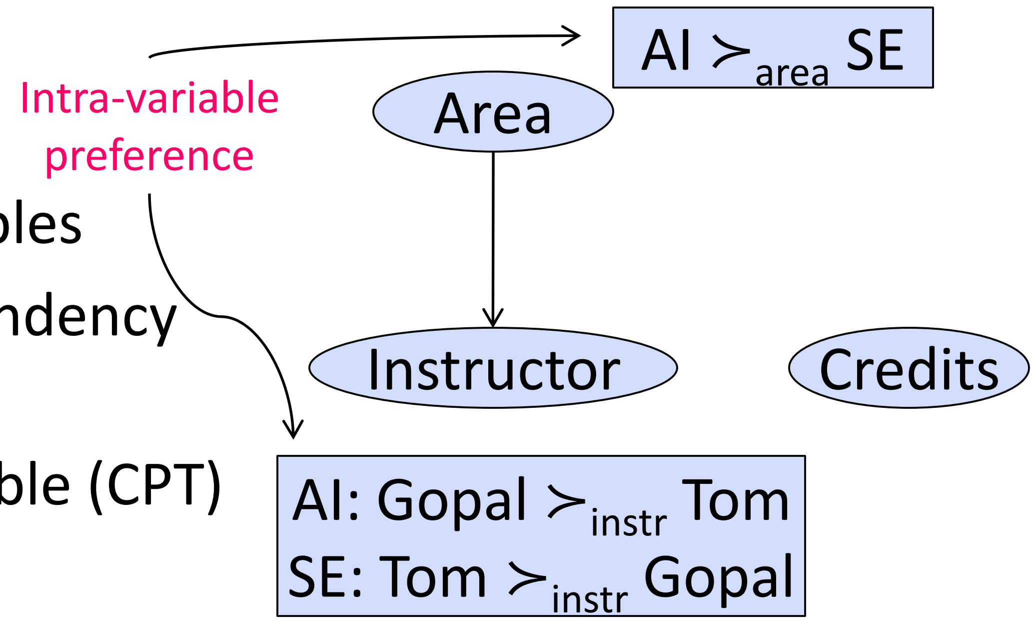
Instructor $\rhd$ Credits

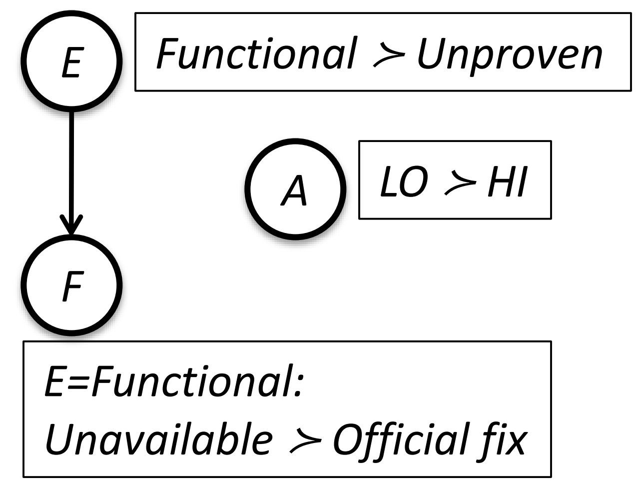
Idea is to represent *comparative* preferences

Conditional Preference nets (CP-nets) [Boutilier et al., 1997]

CP-nets

- Nodes – Preference Variables
- Edges – Preferential Dependency between variables
- Conditional Preference Table (CPT) annotates nodes
- CPT can be partially specified
- Comparative preferences over:
  - Pairs of values of an attribute

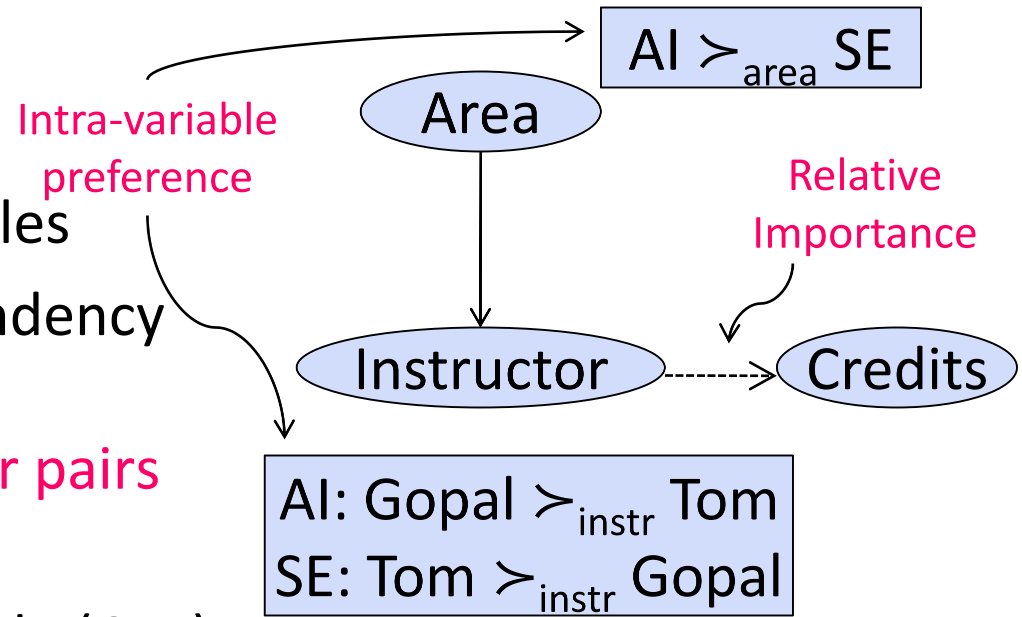


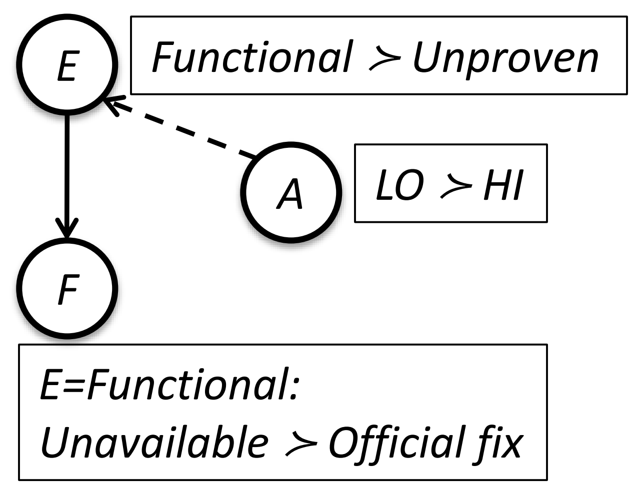


Trade-off enhanced CP-nets (TCP-nets) [Brafman et al., 2006]

TCP-nets

- Nodes – Preference Variables
- Edges – Preferential Dependency between variables
  & Relative Importance over pairs of variables
- Conditional Preference Table (CPT) annotates nodes
- CPT can be partially specified
- Comparative preferences over:
  - Pairs of values of an attribute
  - Pairs of attributes (importance)



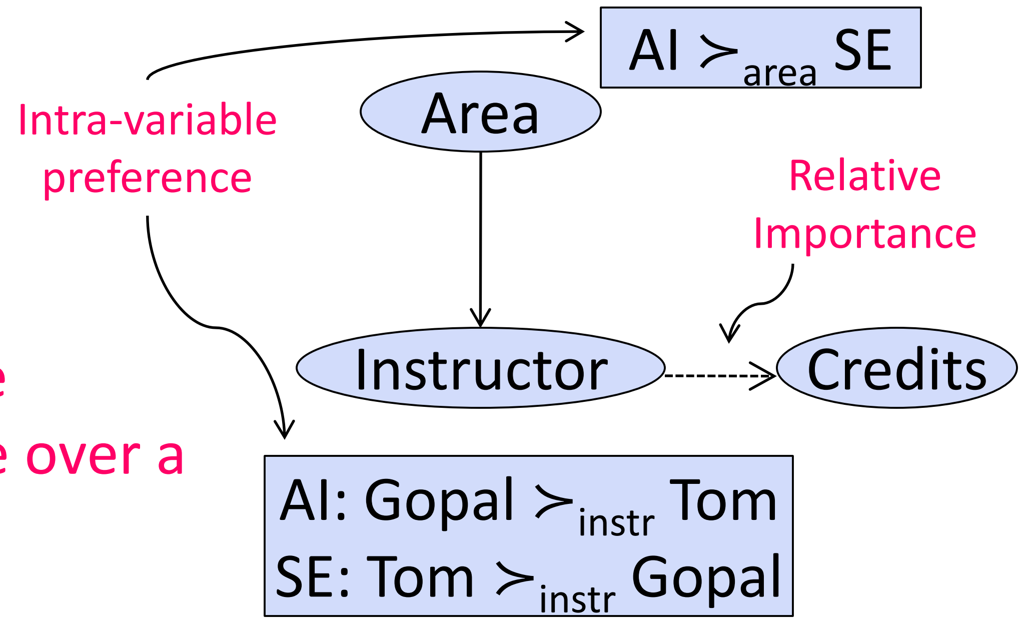


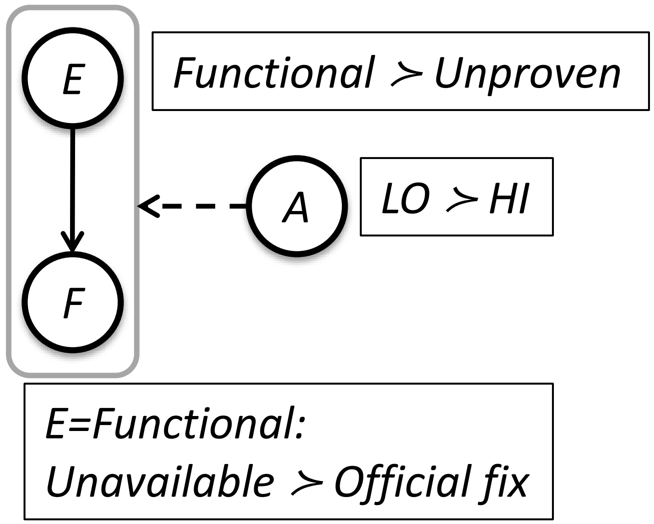
Conditional Preference Theories (CP-theories) [Wilson 2004,2006]

CP-Theories

- Similar to TCP-nets but..

  Possible to express relative Importance of one variable over a set of variables





Conditional Importance Networks (CI-nets) [Bouveret 2009]

CI-nets (fair division of goods among agents)

- Preference variables represent items to be included in a deal
- Preference variables are Binary (presence/absence of an item)
- Intra-variable Preference is monotonic ($0 \succ 1$ or $1 \succ 0$)
  - Subsets preferred to supersets (or vice versa) by default
- CI-net Statements are of the form $S^+, S^- : S_1 \succ S_2$
  - Represents preference on the *presence of one set of items over another set under certain conditions*
  - If all propositions in S^+ are true and all propositions in S^- are false, then the set of propositions S_1 is preferred to S_2

Conditional Importance Networks (CI-nets) [Bouveret 2009]

CI-nets (fair division of goods among agents)

- Example:

a = Name
b = Address
c = Bank Routing Number
d = Bank Account Number

P1. $\{d\}, \{\} : \{b\} \succ \{c\}$
P2. $\{b\}, \{a\} : \{c\} \succ \{d\}$
P3. $\{\}, \{d\} : \{a, b\} \succ \{c\}$

If I have to …
*disclose my **address** without having to disclose my **name**,*
then I would prefer …
*giving my **bank routing number***
over …
*my **bank account number***

Other Preference Languages

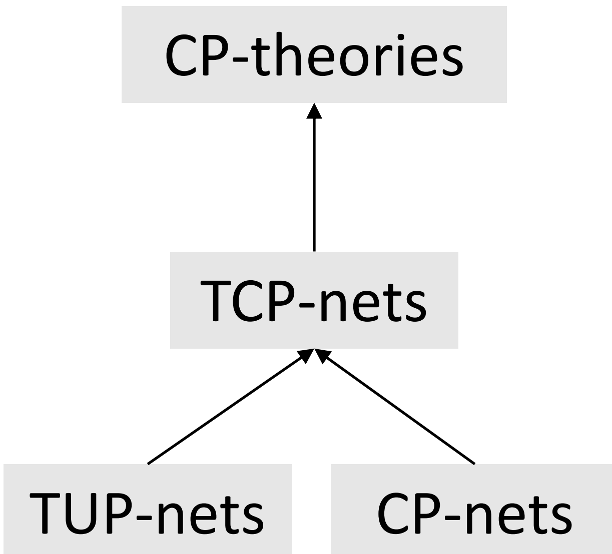
- Preference languages in Databases [Chomicki 2004]
- Preferences over Sets [Brafman et al. 2006]
- Preferences among sets (incremental improvement)[Brewka et al. 2010]
- Tradeoff-enhanced Unconditional Preferences (TUP-nets) [Santhanam et al. 2010]
- Cardinality-constrained CI-nets (C^3I-nets) [Santhanam et al. 2013]

In this tutorial …

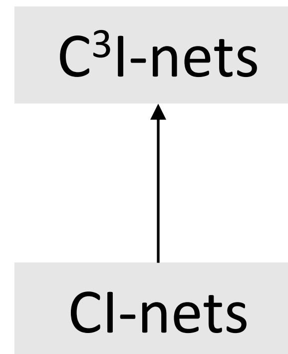
- We stick to CP-nets, TCP-nets and CI-nets.
- Overall approach is generic; extensible to all other ceteris paribus preference languages

Relative Expressivity of Preference Languages

Preferences over Multi-domain Variables



Preferences over (sets of) Binary Variables



Part II

Theoretical Aspects of Representing & Reasoning with Ceteris Paribus Preferences

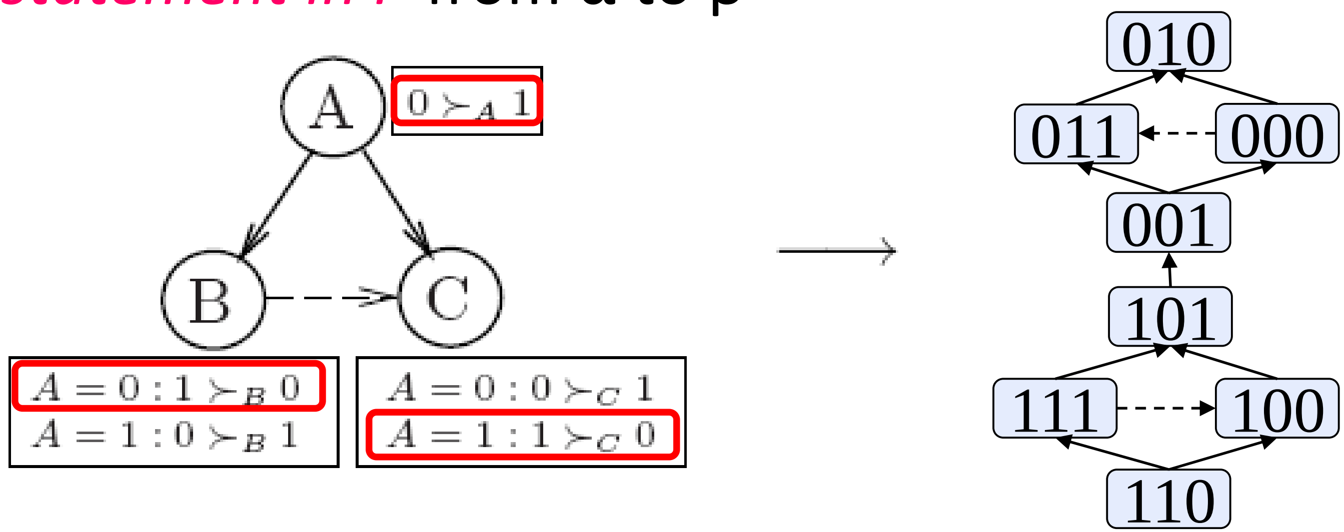
Theoretical Aspects

Part II – Outline

- Induced Preference Graph (IPG)
- Semantics in terms of flips in the IPG
- Reasoning Tasks
  - Dominance over Alternatives
  - Equivalence & Subsumption of Preferences
  - Ordering of Alternatives
- Complexity of Reasoning

Induced Preference Graph (IPG) [Boutilier et al. 2001]

- *Induced preference graph* δ(P)= *G*(*V*,*E*) of preference spec P:
  - Nodes *V* : set of alternatives
  - Edges *E* : (α , β) ∈ *E* iff there is a *flip induced by some statement in P* from α to β



- δ(N) is acyclic (dominance is a strict partial order)
- α ≻ β iff there is a *path* in δ(N) from α to β (serves as the *proof*)

Santhanam et al. AAAI 2010

Preference Semantics in terms of IPG

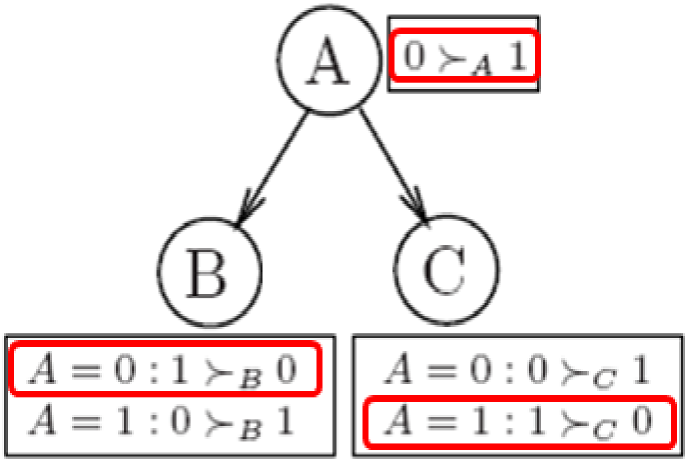
- $(\alpha, \beta) \in E$ iff there is a *flip* from α to β *"induced by some preference" in P*
- Types of flips
  - Ceteris Paribus flip – flip a variable, "all other variables equal"
  - Specialized flips
    - Relative Importance flip
    - Set based Importance flip
    - Cardinality based Importance flip
- Languages differ in the semantics depending on the specific types of flips they allow

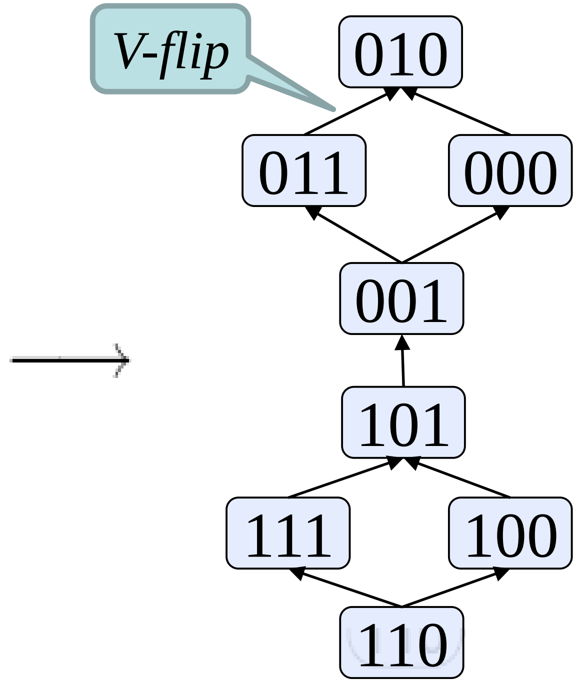
… Next: examples

Flips for a CP-net [Boutilier et al. 2001]

- $(\alpha, \beta) \in E$ iff there is a statement in CP-net such that $x_1 \succ_1 x'_1$ (x_1 is preferred to x'_1) and …
  - ***V-flip*** : *all other variables being equal*, $\alpha(X_1)=x_1$ and $\beta(X_1)=x'_1$

Ceteris paribus
(all else being equal)

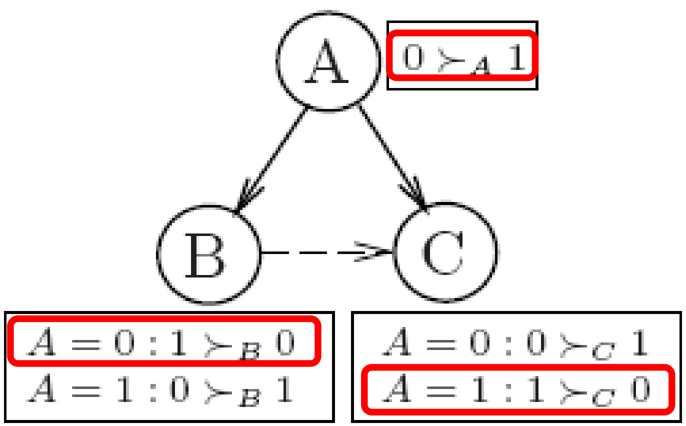


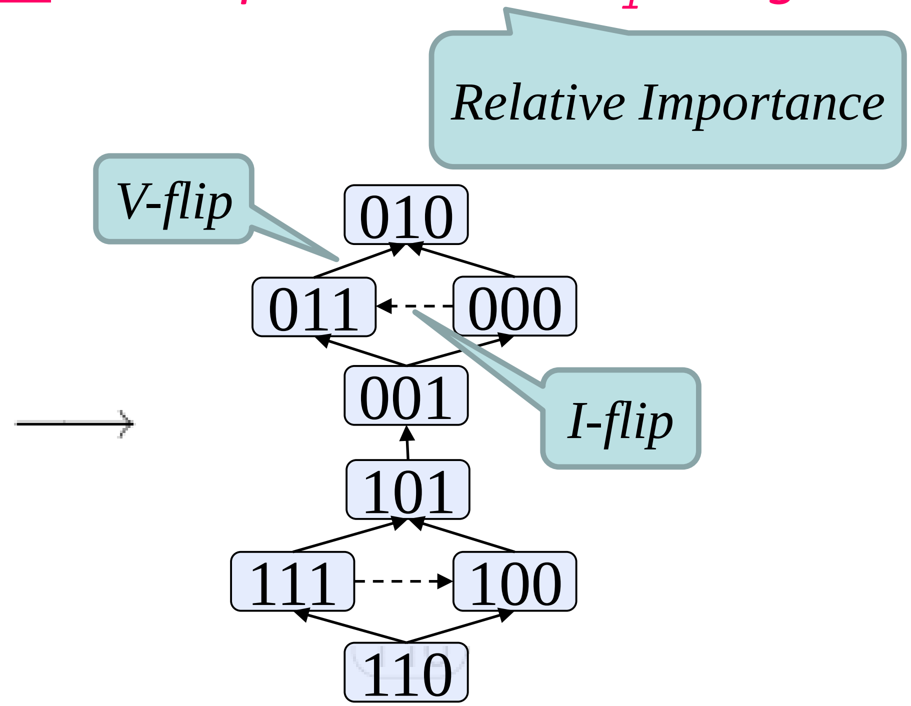


Single variable flip – change value of 1 variable at a time

Flips for TCP-nets & CP-theories [Brafman et al., Wilson 2004]

- (α , β) ∈ *E* iff there is a statement in TCP-net such that $x_1 \succ_1 x'_1$ (x_1 is preferred to x'_1) and …
  - ***V-flip*** : all other variables being equal, $\alpha(X_1)=x_1$ and $\beta(X_1)=x'_1$
  - ***I-flip*** : all variables *except **<u>those</u>** less important than X_1 being equal*, $\alpha(X_1)=x_1$ and $\beta(X_1)=x'_1$





<u>Multi-variable flip</u> – change values of multiple variables at a time

Flips for a CI-net [Bouveret 2009]

- *Recall*: CI-nets express *preferences over subsets* of binary variables X.
  - Truth values of X_i tells its presence/absence in a set
  - Nodes in IPG correspond to subsets of X
  - Supersets are always preferred to Strict Subsets (conventional)
  - $S^+, S^- : S_1 \succ S_2$ interpreted as …

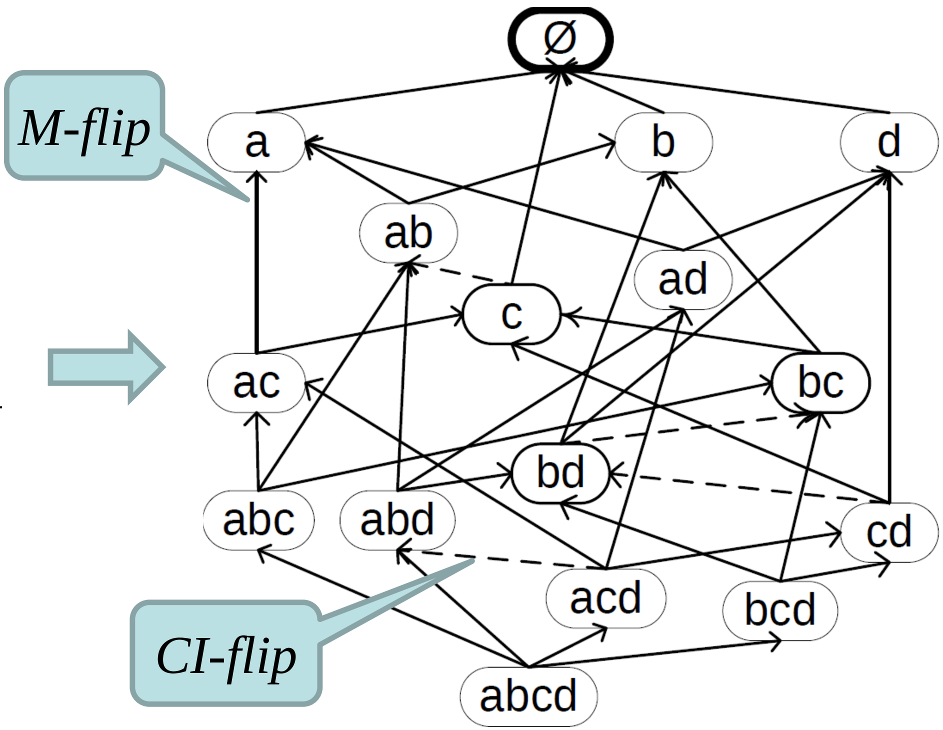
    If all propositions in S^+ are true and all propositions in S^- are false, then the set of propositions S_1 is preferred to S_2
- For α , $\beta \subseteq X$, $(\alpha, \beta) \in E$ (β preferred to α) iff
  - ***M-flip*** : all other variables being equal, $\alpha \subset \beta$
  - ***CI-flip*** : there is a CI-net statement s.t. $S^+, S^- : S_1 \succ S_2$ and α , β satisfy S^+, S^- and α satisfies S^+ and β satisfies S^-.

Flips for a CI-net [Bouveret 2009]

- For α , β ⊆ X, (α, β) ∈ *E* (β preferred to α) iff
  - ***M-flip*** : all other variables being equal, α ⊂ β
  - ***CI-flip*** : there is a CI-net statement $S^+, S^- : S_1 \succ S_2$ s.t. α , β satisfy S^+, S^- and α satisfies S^+ and β satisfies S^-.
- Example:

a = Name
b = Address
c = Bank Routing Number
d = Bank Account Number

P1. $\{d\}, \{\} \; : \; \{b\} \succ \{c\}$
P2. $\{b\}, \{a\} \; : \; \{c\} \succ \{d\}$
P3. $\{\}, \{d\} \; : \; \{a, b\} \succ \{c\}$



Oster et al. FACS 2012

Flips for a C^3I-net [Santhanam et al. 2013]

- C^3I-nets express *preference over subsets* similar to CI-net
  - Truth values of X_i tells its presence/absence in a set
  - Nodes in IPG correspond to subsets of X
  - Sets with *higher cardinality* are preferred (conventional)
  - $S^+, S^- : S_1 \succ S_2$ interpreted as …

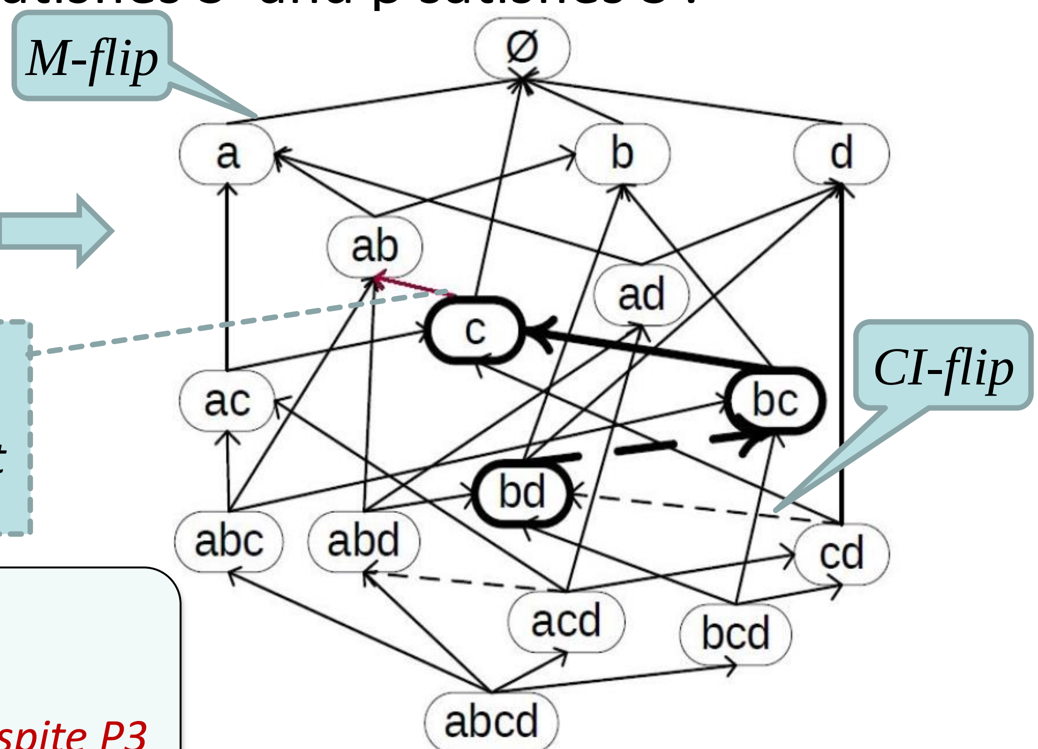
    If all propositions in S^+ are true and all propositions in S^- are false, then the set of propositions S_1 is preferred to S_2
- For α , $\beta \subseteq X$, $(\alpha, \beta) \in E$ (β preferred to α) iff
  - ***M-flip*** : all other variables being equal, $|\alpha| < |\beta|$
  - ***CI-flip*** : there is a CI-net statement s.t. $S^+, S^- : S_1 \succ S_2$ and α , β satisfy S^+, S^- and α satisfies S^+ and β satisfies S^-.
  - Extra cardinality constraint to enable dominance

Flips for a C³I-net [Santhanam et al. 2013]

- For α , β ⊆ X, (α, β) ∈ *E* (β preferred to α) iff
  - ***M-flip*** : α ⊂ β (all other variables being equal)
  - ***CI-flip*** : there is a CI-net statement $S^+, S^- : S_1 \succ S_2$ s.t. α , β satisfy S^+, S^- and α satisfies S^+ and β satisfies S^-.
  - ***C-flip*** : |α| < |β|

$$\text{P1. } \{d\}, \{\} \ : \ \{b\} \succ \{c\}$$
$$\text{P2. } \{b\}, \{a\} \ : \ \{c\} \succ \{d\}$$
$$\text{P3. } \{\}, \{d\} \ : \ \{a, b\} \succ \{c\}$$

*C-flip - present in the CI-net, but **not** in the C³I-net*



- *{c} ≻ {bc} due to Monotonicity*
- *{bc} ≻ {bd} due to P2*
- *{ab} ⊁ {c} due to Cardinality despite P3*

Santhanam et al. CSIIRW 2013

Reasoning Tasks

The semantics of any ceteris paribus language can be represented in terms of properties of IPG

- Now we turn to the Reasoning Tasks:
  - Dominance & Consistency
  - Equivalence & Subsumption
  - Ordering
- We describe reasoning tasks only in terms of verifying properties of the IPG

Reasoning Tasks

Dominance relation:

- $\alpha \succ \beta$ iff there exists a *sequence of flips* from β to α
- Property to verify: *Existence of path in IPG* from β to α

Consistency:

- A set of preferences is *consistent* if $\succ$ is a strict partial order
- Property to - verify: *IPG is acyclic*

Ordering: ?

semantics

- *Hint:* The non-dominated alternatives in the IPG are the best
- Strategy – Repeatedly Query IPG to get strata of alternatives

Equivalence (& Subsumption):

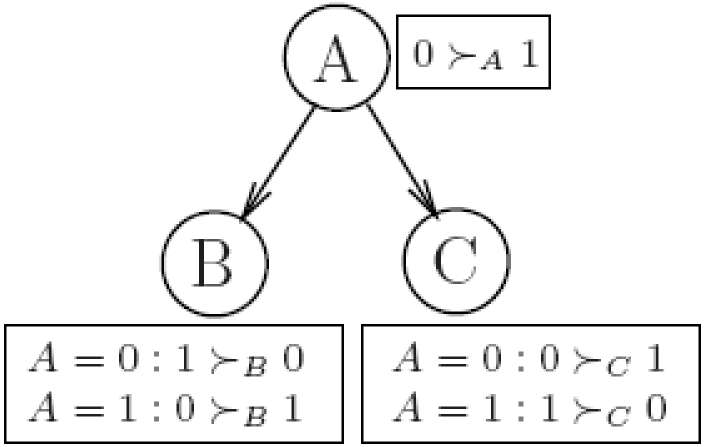
- A set P_1 of preferences is *equivalent* to another set P_2 if they induce the same dominance relation
- Property to verify: *IPGs are reachability equivalent*

Reasoning Tasks

Reasoning Task	**Computation Strategy: Property of IPG to check**	**Remarks**
Dominance: $\alpha \succ \beta$	Is β reachable from α ?	
Consistency of a set of preferences (P)	Is the IPG of P acyclic?	Satisfiability of the dominance relation; strict partial order
Equivalence of two sets of preferences P_1 and P_2	Are the IPGs of P_1 and P_2 reachability-equivalent?	
Subsumption of one set of preference (P_1) by another (P_2)	If β reachable from α in the IPG of P_1, does the same hold in the IPG of P_2?	
Ordering of alternatives	Iterative verification of the IPG for the non-existence of the non-dominated alternatives	Iterative modification of the IPG to obtain next set of non-dominated alternatives

Complexity of Dominance [Goldsmith et al. 2008]

Cast as a *search* for a flipping sequence, or *a path in IPG*



- α = (A = 1, B = 0, C = 0)
- β = (A = 0, B = 1, C = 1)
- α ≻ β – Why?

PSPACE-complete

Dominance testing reduces to STRIPS planning (Goldsmith et al. 2008)

Complexity of Reasoning Tasks

Reasoning Task	Complexity	Work by
Dominance: $\alpha \succ \beta$	PSPACE-complete	Goldsmith et al. 2008
Consistency of a set of preferences (P)	PSPACE-complete	Goldsmith et al. 2008
Equivalence of two sets of preferences P_1 and P_2	PSPACE-complete	Santhanam et al. 2013
Subsumption of one set of preference (P_1) by another (P_2)	PSPACE-complete	Santhanam et al. 2013
Ordering of alternatives	NP-hard	Brafman et al. 2011

Part III – Practical Aspects

<u>Part III</u>

Practical Aspects of Reasoning with Ceteris Paribus Preferences

Part III – Outline

- Two Sound and Complete Reasoning Approaches:
  - Logic Programming based
    - Answer Set Programming [Brewka et al.]
    - Constraint Programming [Brafman et al. & Rossi et al.]
  - Model Checking based
    - Preference reasoning can be reduced to verifying properties of the IPG [Santhanam et al. 2010]
    - Translate IPG into a Kripke Structure Model
    - Translate reasoning tasks into temporal logic properties over model
- Approximation & Heuristics
  - Wilson [Wilson 2006, 2011]

Preference Reasoning via Model Checking

- The *first practical solution to preference reasoning* in moderate sized CP-nets, TCP-nets, CI-nets, etc.
  - Casts dominance testing as reachability in an induced graph
  - Employs direct, succinct encoding of preferences using Kripke structures
  - Uses Temporal logic (CTL, LTL) for querying Kripke structures
  - Uses direct translation from reasoning tasks to CTL/LTL
    - Dominance Testing
    - Consistency checking (loop checking using LTL)
    - Equivalence and Subsumption Testing
    - Ordering (next-preferred) alternatives

Santhanam et al. (AAAI 2010, KR 2010, ADT 2013); Oster et al. (ASE 2011, FACS 2012)

Model Checking [Clark et al. 1986]

- Model Checking: Given a desired property φ, (typically expressed as a temporal logic formula), and a (Kripke) structure *M* with initial state s, decide if $M, s \models \varphi$
- Active area of research in formal methods, AI (SAT solvers)
- Broad range of applications: hardware and software verification, security..
- Temporal logic languages : CTL, LTL, μ-calculus, etc.
- Many model checkers available : SMV, NuSMV, Spin, etc.

Advantages of Model Checking:

1. Formal Guarantees
2. Justification of Results

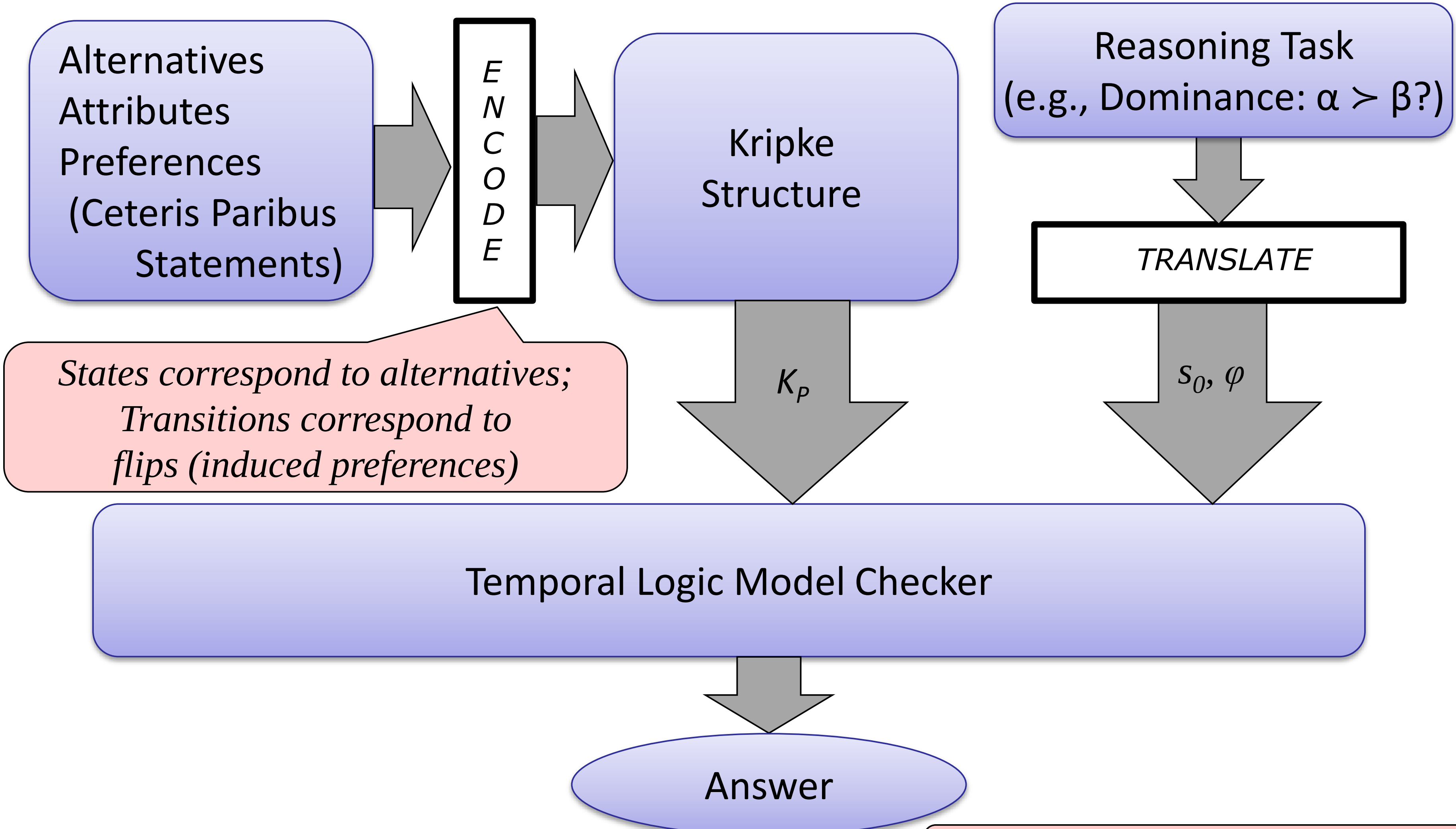
Preference Reasoning via Model Checking

- Key Idea:

> Preference reasoning can be reduced to verifying properties of the Induced Preference Graph [Santhanam et al. 2010]

- Overview of Approach
  1. Translate IPG into a Kripke Structure Model
  2. Translate reasoning tasks into temporal logic properties over model

Overview of Approach



Santhanam et al. AAAI 2010

Kripke Structure [Kripke, 1963]

A Kripke structure is a 4-tuple $K=(S, S_0, T, L)$ over variables V, where

- S represents the set of reachable states of the system
- S_0 is a set of initial states
- T represents the set of state transitions
- L is labeling (interpretation) function maps each node to a set of atomic propositions AP that hold in the corresponding state

Used to specify labeled transition systems describing states of the world w.r.t. flow of time

Computational tree temporal logic (CTL) is an extension of propositional logic

- Includes temporal connectives that allow specification of properties that hold over states and paths in K

Example

- $EF\varphi$ true in state s of K if φ holds in some state in some path beginning at s

Encoding Preference Semantics

Let $P = \{p_i\}$ be a set of ceteris paribus preference statements on a set of preference variables $X = \{x_1, x_2, \ldots\}$

Reasoning Strategy:

- Construct a Kripke model $K_P = (S, S_0, T, L)$ using variables Z
  - $Z = \{z_i \mid x_i \in X\}$, with each variable z_i having same domain D_i as x_i
  - K_P must mimic the IPG is some sense
- The State-Space of K_P
  - $S = \Pi_i D_i$: states correspond to set of all alternatives
  - T : transitions correspond to allowed changes in valuations according to flip-semantics of the language
  - L : labeling (interpretation) function maps each node to a set of atomic propositions *AP* that hold in the corresponding state
  - S_0 : Initial states assigned according to the reasoning task at hand

From Syntax to Semantics

Encode K_P such that paths in IPG are enabled transitions, and no additional transitions are enabled

- Let p be a conditional preference statement in P
- p induces a flip between two nodes in the IPG iff
  1. "Condition" part in the preference statement is satisfied by both nodes
  2. "Preference" part (less & more preferred valuations) in satisfied by both
  3. "Ceteris Paribus" part that ensures apart from (1 & 2) that all variables other than those specified to change as per (2) are equal in both nodes

- Create transitions in K_P with guard conditions
  – "Condition" part of statement is translated to the *guard* condition
  – "Preference" part of statement is translated to assignments of variables in the target state
  – How to ensure ceteris paribus condition?

From Syntax to Semantics

Encode K_P such that paths in IPG are enabled transitions, and no additional transitions are enabled

- Let p be a conditional preference statement in P
- p induces a flip between two nodes in the IPG iff
  1. "Condition" part in the preference statement is satisfied by both nodes
  2. "Preference" part (less & more preferred valuations) in satisfied by both
  3. "Ceteris Paribus" part that ensures apart from (1 & 2) that all variables other than those specified to change as per (2) are equal in both nodes

- Create transitions in K_P with guard conditions
  - "Condition" part of statement is translated to the *guard* condition
  - "Preference" part of statement is translated to assignments of variables in the target state

How to encode ceteris paribus condition in the guards?

From Syntax to Semantics

Recall: In temporal logics, destination states represent "*future*" state of the world

- Equality of source and destination states forbidden as part of the guard condition specification!
- Workaround: Use auxiliary variables h_i to label edges

$$h_i = \begin{cases} 0 & \Rightarrow \text{value of } z_i \textbf{ must not} \text{ change in a transition in the Kripke structure } K(P) \\ 1 & \Rightarrow \text{otherwise} \end{cases} \quad (1)$$

- Auxiliary edge labels don't contribute to the state space

From Syntax to Semantics

Guard condition specification

- *Recall:* *p* induces a flip between two nodes in the IPG iff
  1. "Condition" part in the preference statement is satisfied by both nodes
  2. "Preference" part (less & more preferred valuations) in satisfied by both
  3. "Ceteris Paribus" part that ensures apart from (1 & 2) that all variables other than those specified to change as per (2) are equal in both nodes
- For each statement *p* of the form $\varrho : x_i = v_i \succ_{x_i} x_i = v'_i$ where ϱ is the "condition" part, guard condition is

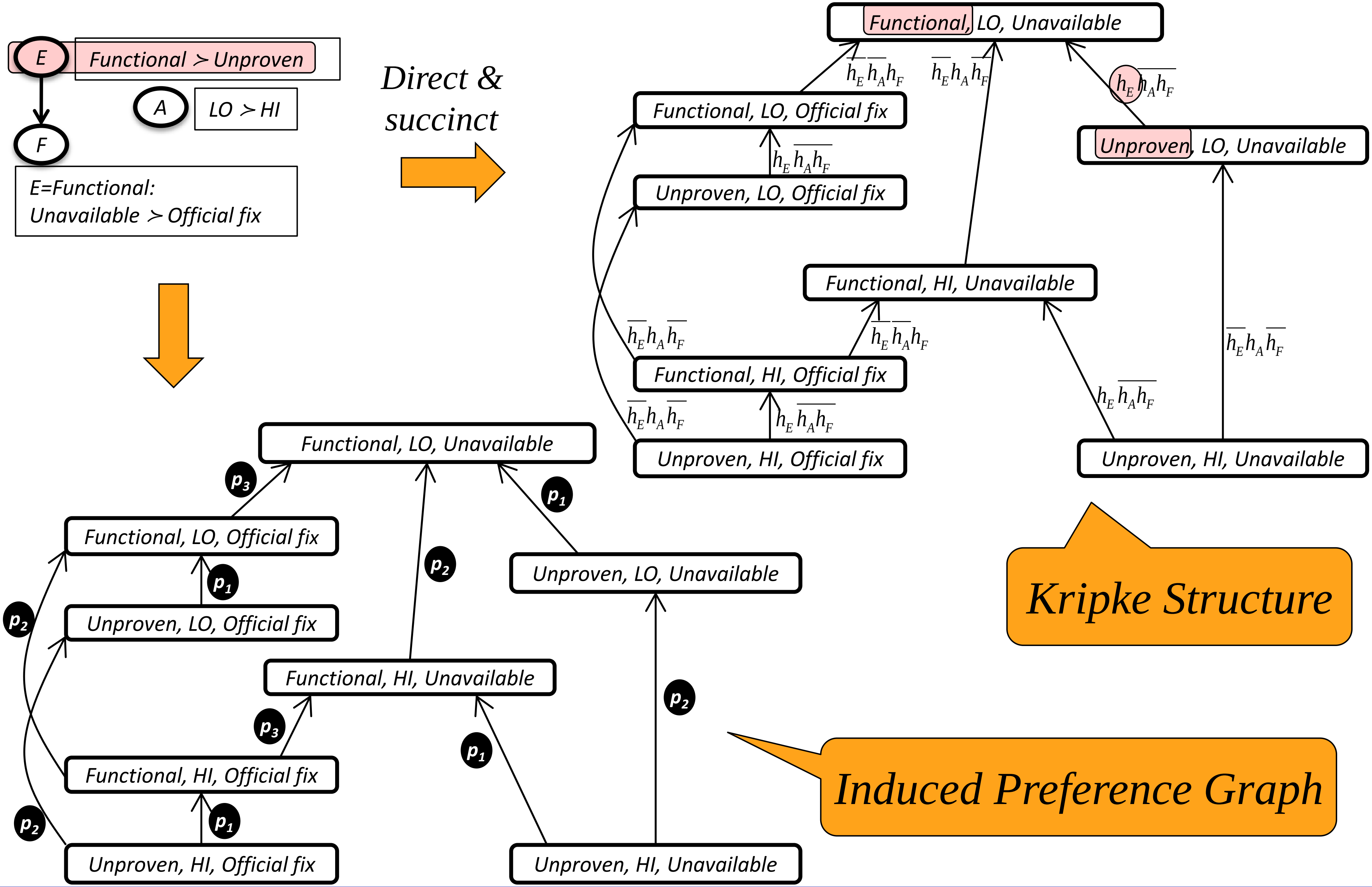
$$\mathcal{G}(p) = Allow(p) \wedge Restrict(p) \text{ s.t.}$$

condition preference

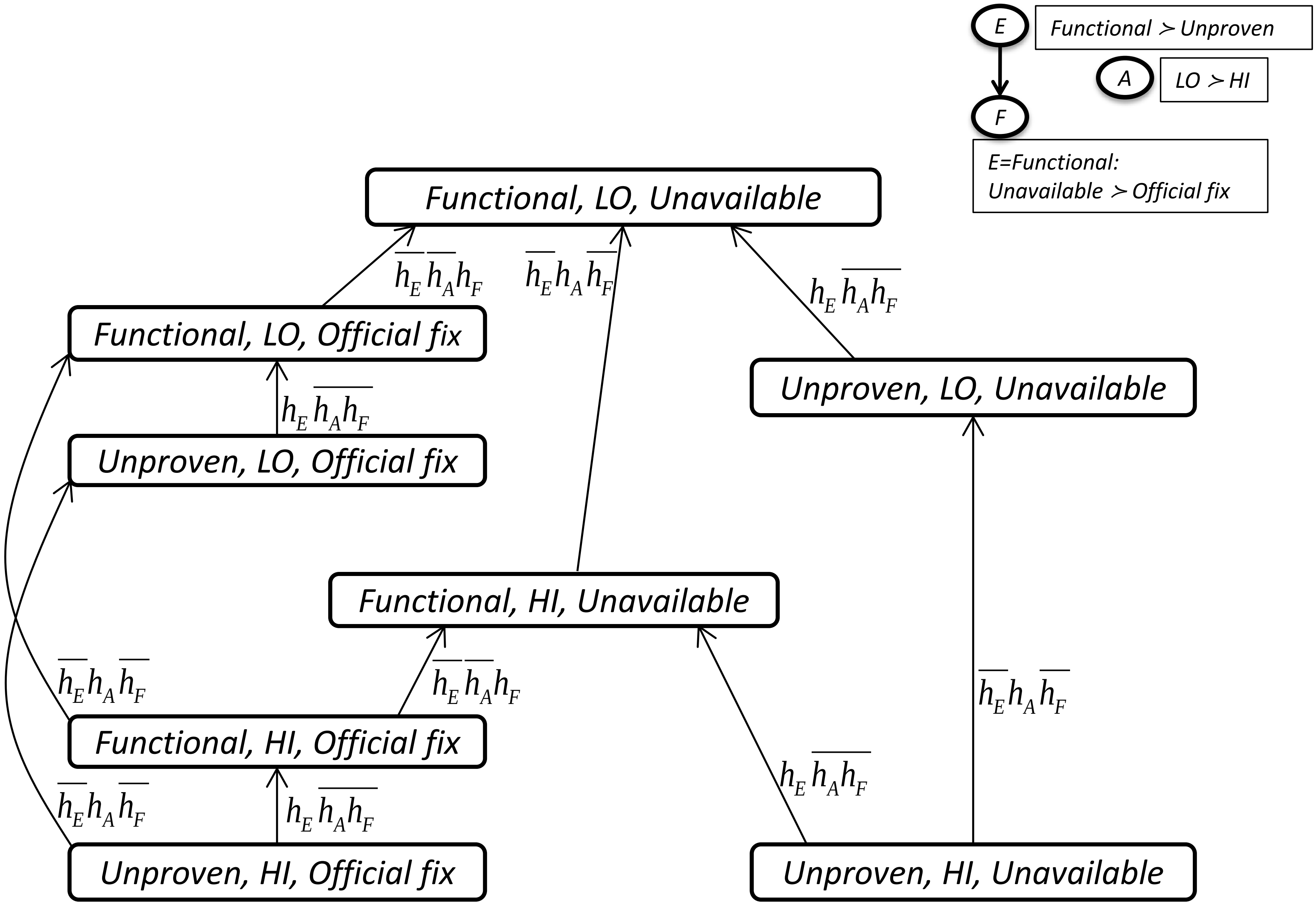
$$\begin{aligned} Allow(p) &:= \varrho \wedge z_i = v'_i \wedge h_i = 1 \\ Restrict(p) &:= \bigwedge_{x_j \in X \setminus \{x_i\}} h_j = 0 \end{aligned}$$

ceteris paribus

Encoding CP-net semantics



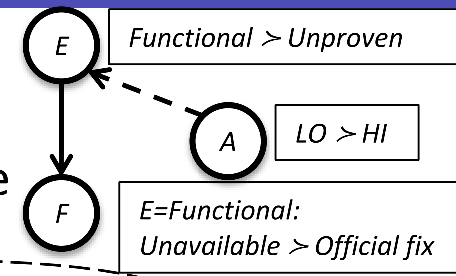
Encoding CP-net semantics

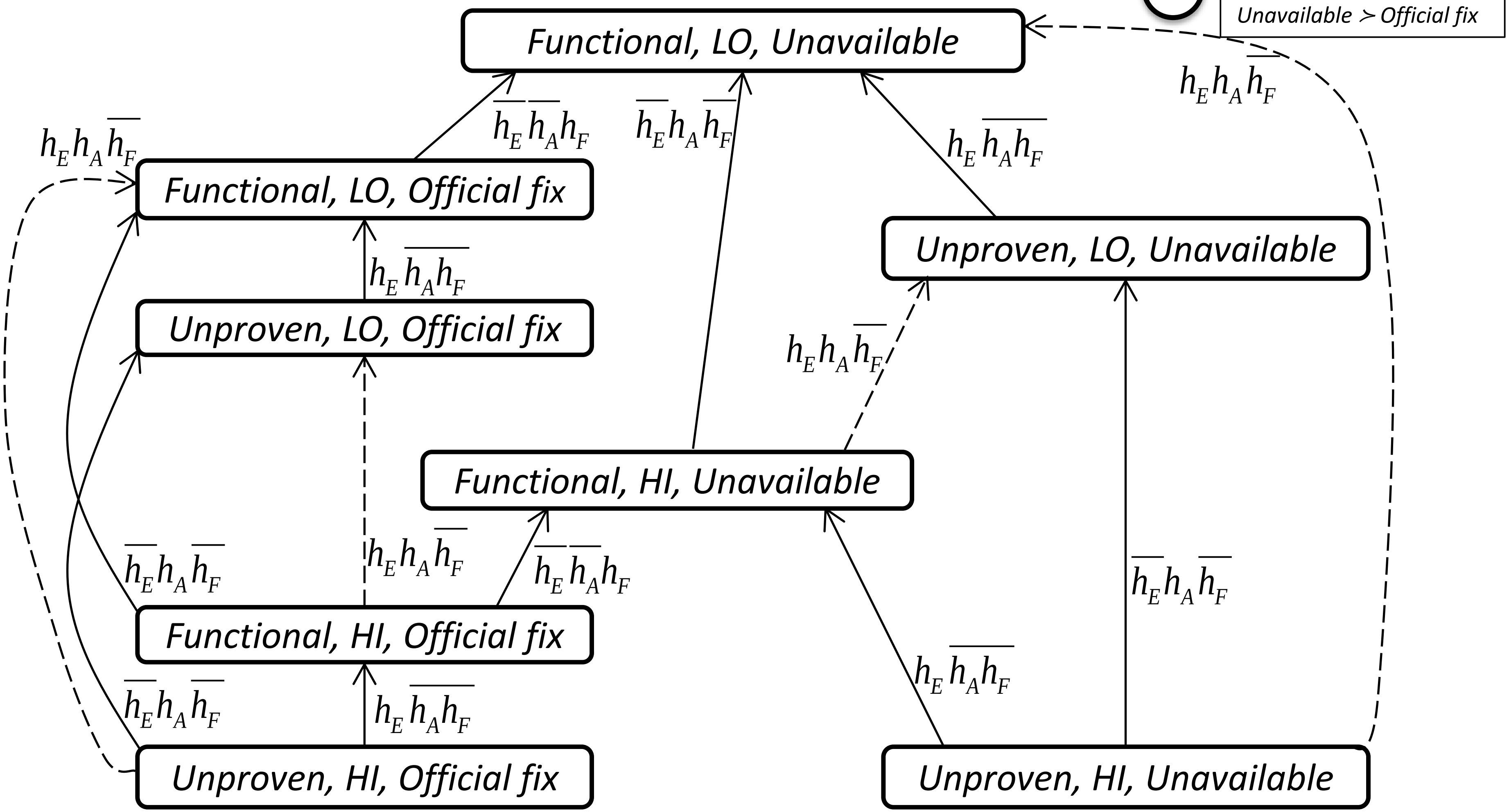


Encoding TCP-net Semantics

TCP-nets : Same overall idea as CP-nets

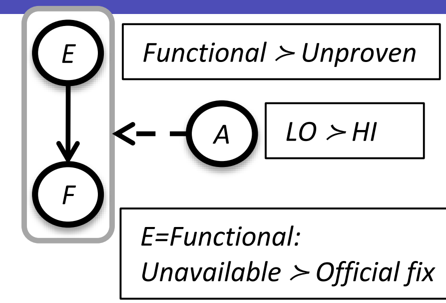
- Additional rule for encoding simple relative importance

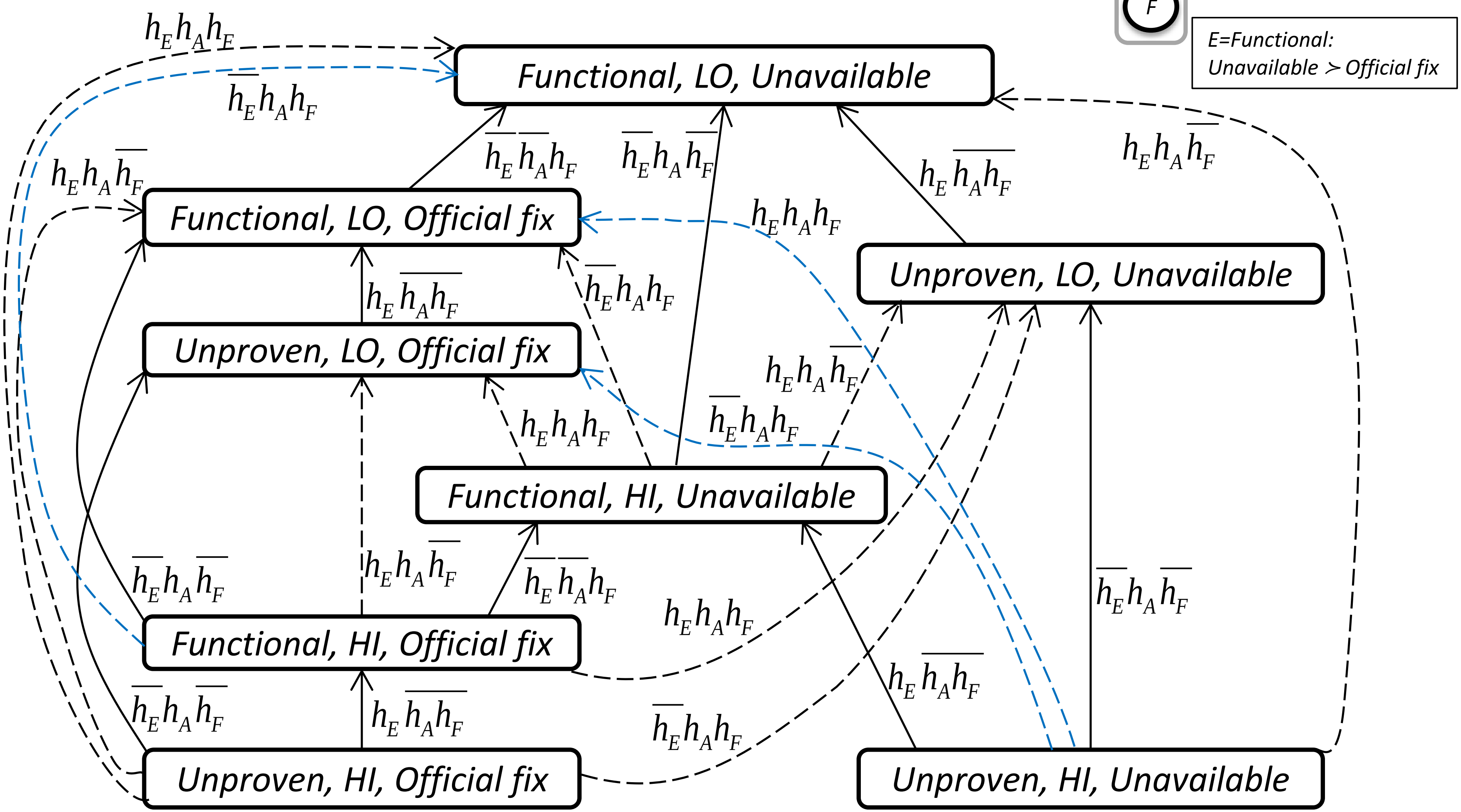




Encoding CP-theory Semantics

CP-theory: Same idea as TCP-net + Additional rule





Encoding Reasoning Tasks as Temporal Logic Properties

Next :

Specifying and Verifying Properties in Temporal Logic

Translating Reasoning Tasks into Temporal Logic Properties

Encoding Reasoning Tasks as Temporal Logic Properties

Computational tree temporal logic (CTL) [Clark et al. 1986] is an extension of propositional logic

- Includes temporal connectives that allow specification of properties that hold over states and paths in a Kripke structure
- CTL Syntax & Semantics

$\mathtt{EX}\ \psi$ if there exists a path $s = s_1 \rightarrow s_2 \ldots$ such that s_2 satisfies ψ

$\mathtt{AX}\ \psi$ if for all paths such that $s = s_1 \rightarrow s_2 \ldots$, s_2 satisfies ψ

$\mathtt{E}\ [\psi_1 \mathtt{U}\ \psi_2]$ if there exists a path $s = s_1 \rightarrow s_2 \ldots$ such that $\exists i \geq 1 : s_i$ satisfies ψ_2, and $\forall j < i : s_j$ satisfies ψ_1

- Translating Reasoning Tasks into Temporal Logic Properties
  - Dominance Testing
  - Consistency
  - Equivalence & Subsumption Testing
  - Ordering alternatives

NuSMV *[Cimatti et al. 2001]*:
Our choice of model checker

Dominance Testing (via NuSMV)

Given outcomes α and β, how to check if $\alpha \succ \beta$?

- Let φ_α be a formula that holds in the state corresponding to α
- Let φ_β be a formula that holds in the state corresponding to β

By construction, $\alpha \succ \beta$ wrt iff in the Kripke Structure K_N :

a state in which $\boldsymbol{\varphi_\beta}$ *holds is reachable from a state in which* $\boldsymbol{\varphi_\alpha}$ *holds*

- $\alpha \succ \beta$ iff the model checker NuSMV can verify $\varphi_\alpha \rightarrow \mathrm{EF}\varphi_\beta$ (SAT)
- When queried with $\neg(\varphi_\alpha \rightarrow \mathrm{EF}\varphi_\beta)$, if indeed $\alpha \succ \beta$, then model checker produces a proof of $\alpha \succ \beta$ (flipping sequence)
- Experiments show feasibility of method for 100 var. in seconds

Santhanam et al. AAAI 2010

Obtaining a Proof of Dominance

- 011 is preferred to 100

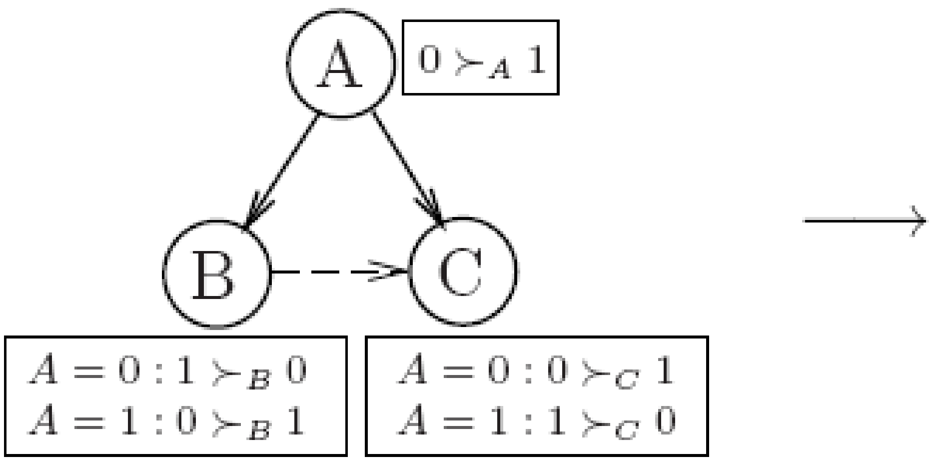
$$(a = 1 \wedge b = 0 \wedge c = 0)$$
$$\Rightarrow EF(a = 0 \wedge b = 1 \wedge c = 1)$$

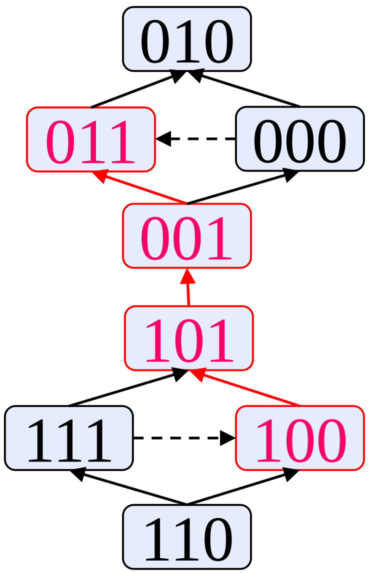
Improving flipping sequence:

$$100 \rightarrow 101 \rightarrow 001 \rightarrow 011$$

Proof : $011 \succ 001 \succ 101 \succ 100$

One of the proofs is chosen non-deterministically





Santhanam et al. AAAI 2010

Obtaining a Proof of Dominance

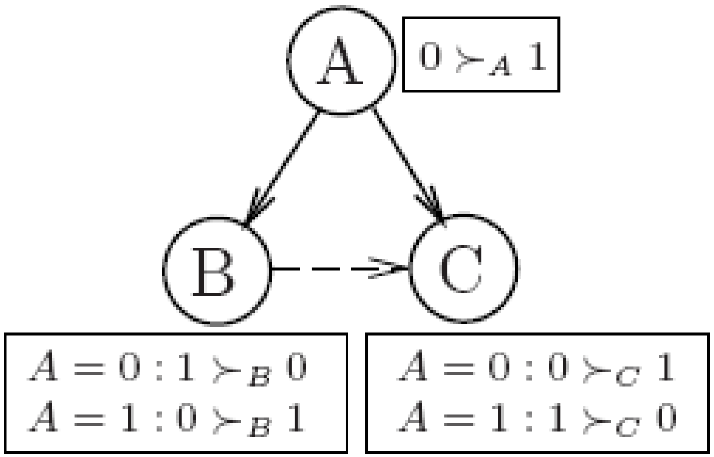
- 011 is preferred to 100

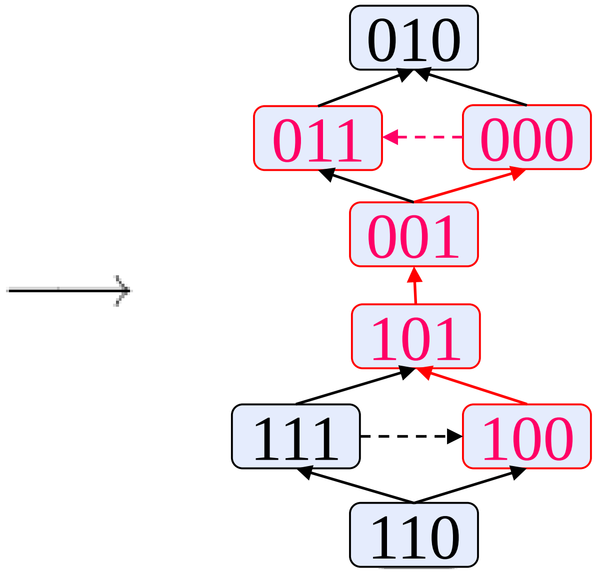
$$(a = 1 \wedge b = 0 \wedge c = 0) \Rightarrow EF(a = 0 \wedge b = 1 \wedge c = 1)$$

Improving flipping sequence:

100 → 101 → 001 → 000 → 011

Proof #2: 011 ≻ 000 ≻ 001 ≻ 101 ≻ 100

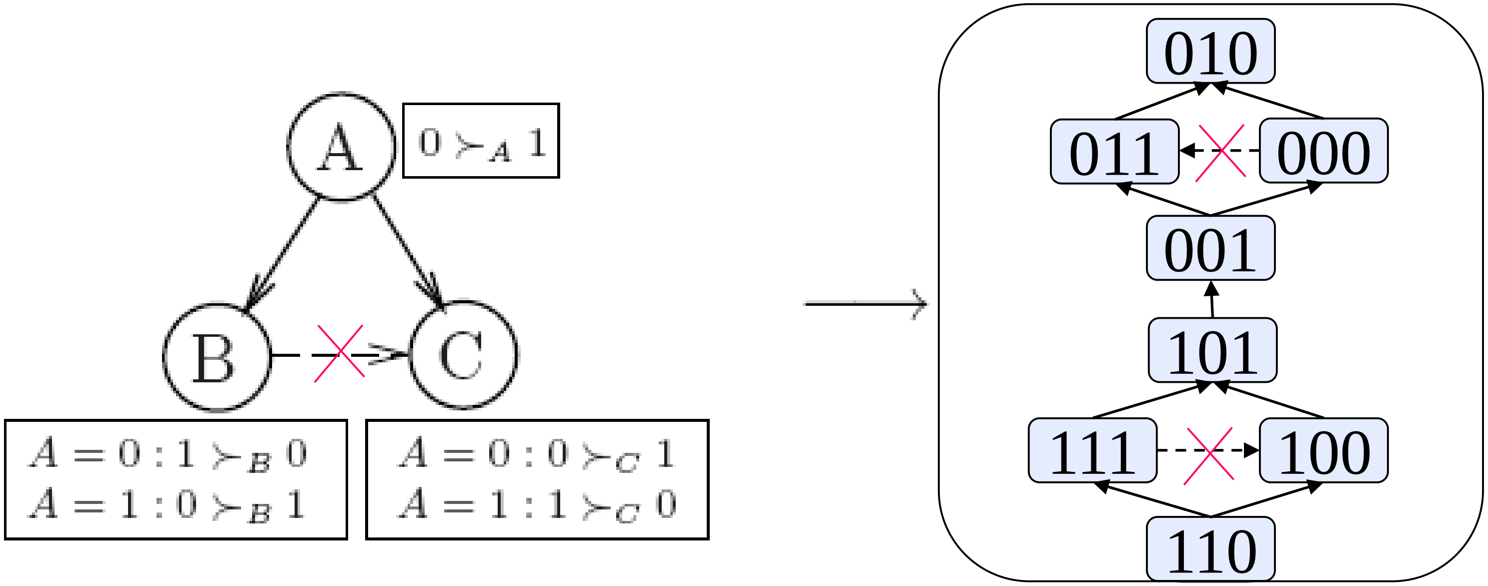




Santhanam et al. AAAI 2010

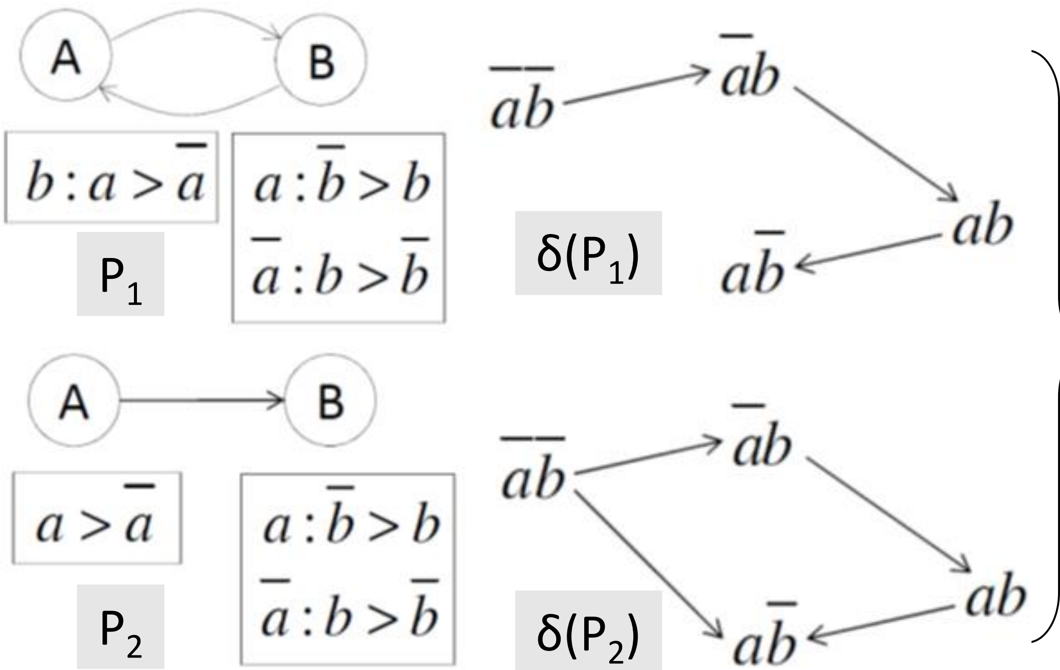
Non-dominance

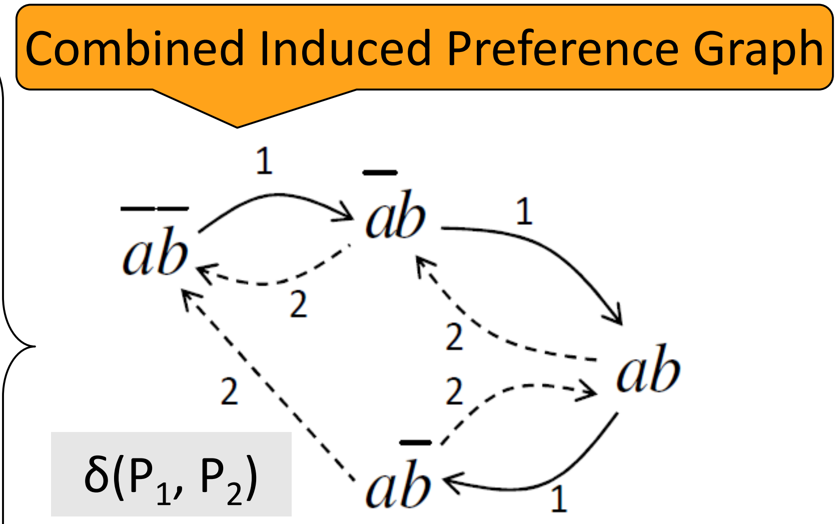
- 011 is not preferred to 000
  (if relative importance of B is not stated)

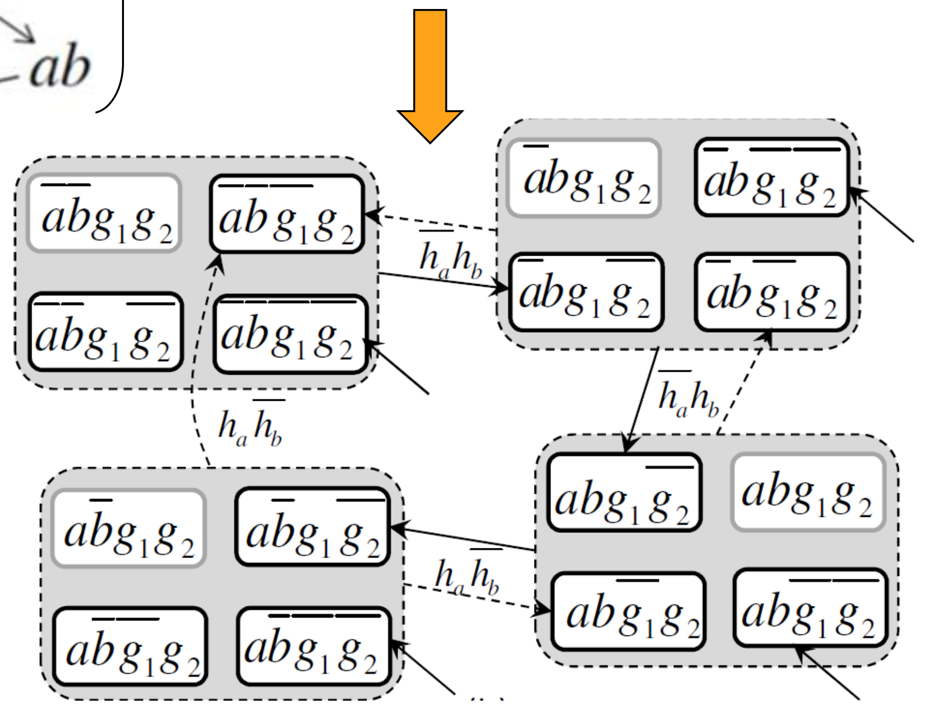


Santhanam et al. AAAI 2010

Equivalence and Subsumption Testing







$$\varphi : \mathbf{AX} \left(g_1 \Rightarrow \mathbf{EX}\ \mathbf{E} \left[g_2\ \mathbf{U}\ (\psi \wedge g_2) \right] \right)$$

CTL Model Checking

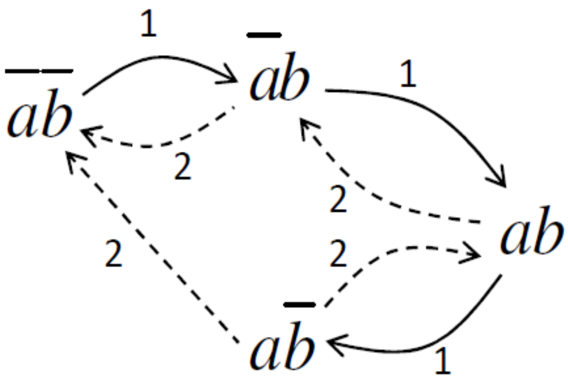
Answer

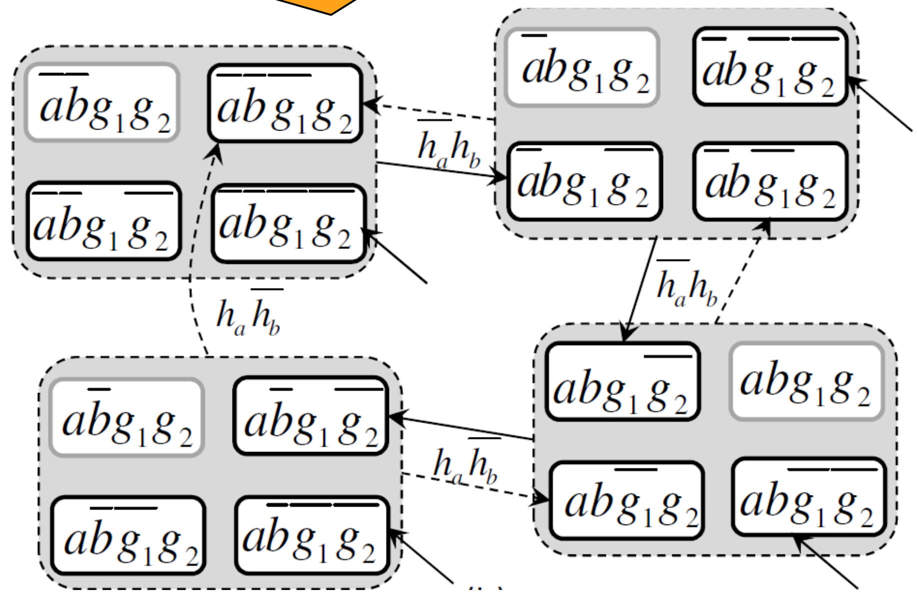
Santhanam et al. ADT 2013

Equivalence and Subsumption Testing

Combined Induced Preference Graph

Kripke Structure





State from which verification is done

$$\varphi : \mathbf{AX} \left(g_1 \Rightarrow \mathbf{EX}\ \mathbf{E} \left[g_2\ \mathbf{U}\ (\psi \wedge g_2) \right] \right)$$

$$\neg\varphi : \mathbf{EX} \left(g_1 \wedge \mathbf{AX}\ \neg\mathbf{E} \left[g_2\ \mathbf{U}\ (\psi \wedge g_2) \right] \right)$$

True $\Leftrightarrow P_1 \sqsubseteq P_2$

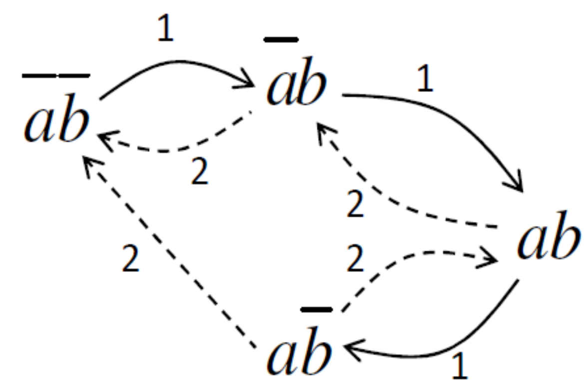
False $\Leftrightarrow P_2 \not\sqsubseteq P_1$

Santhanam et al. ADT 2013

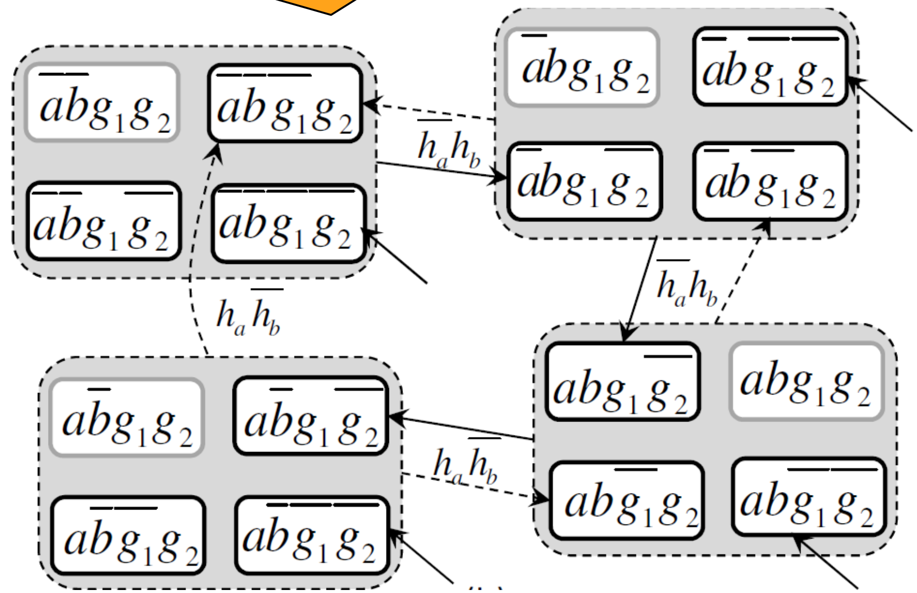
Model Checker returns $\overline{a}\,\overline{b} \rightarrow a\overline{b}$ *as proof*

Equivalence and Subsumption Testing

Combined Induced Preference Graph



Kripke Structure



$\varphi : \mathbf{AX} \left(g_1 \Rightarrow \mathbf{EX} \ \mathbf{E} \left[g_2 \ \mathbf{U} \ (\psi \wedge g_2) \right] \right)$

True ⇔ $P_1 \sqsubseteq P_2$

$\varphi' : \mathbf{AX} \left(g_2 \Rightarrow \mathbf{EX} \ \mathbf{E} \left[g_1 \ \mathbf{U} \ (\psi \wedge g_1) \right] \right)$

True ⇔ $P_2 \sqsubseteq P_1$

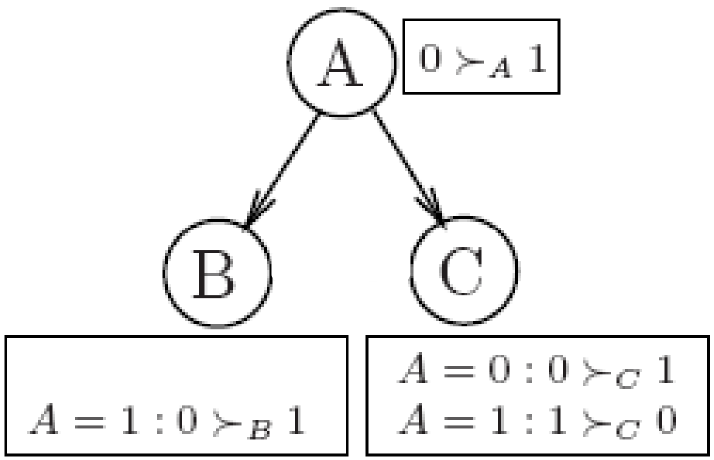
$P_1 \equiv P_2$

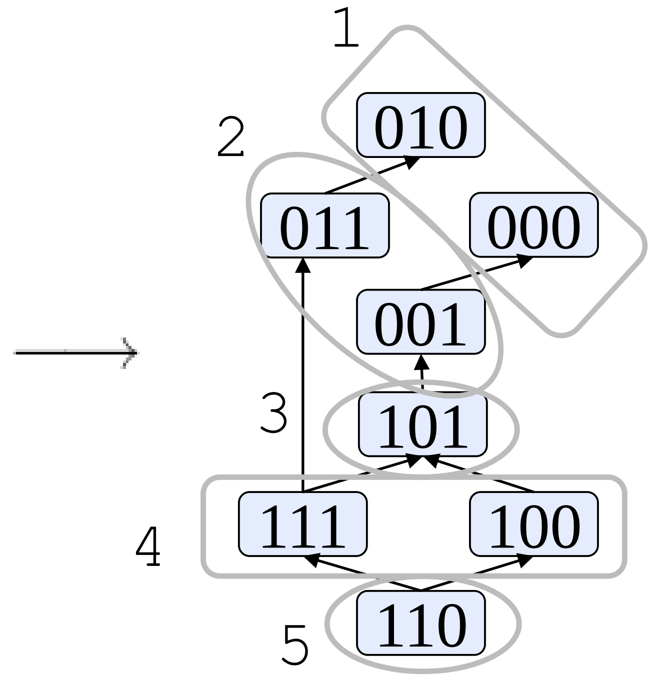
Santhanam et al. ADT 2013

Ordering : Finding the Next-preferred Alternative

- Which alternatives are most-preferred (non-dominated)?
- Can we enumerate all alternatives in order?
- Computing total and weak order extensions of dominance

How to deal with cycles?





We verify a sequence of reachability properties encoded in CTL

Acyclic Case: Oster et al. FACS 2012

Part IV – Applications

<u>Part IV</u>

Applications

Applications

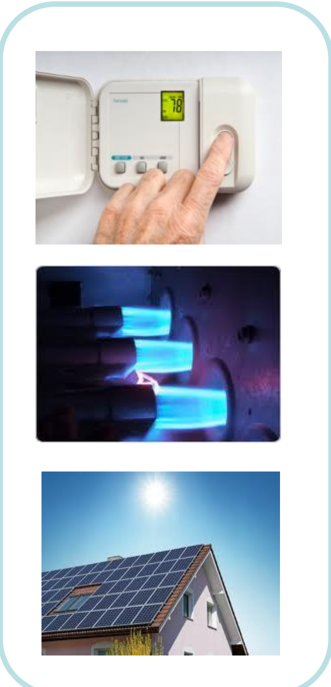
- Sustainable Design of Civil Infrastructure (e.g., Buildings, Pavements)
- Engineering Design (Aerospace, Mechanical)
- Strategic & mission critical decision making (Public policy, Defense, Security)
- Chemical and Nano-Toxicology
- Site Selection for Nuclear Waste and setting up new nuclear plants

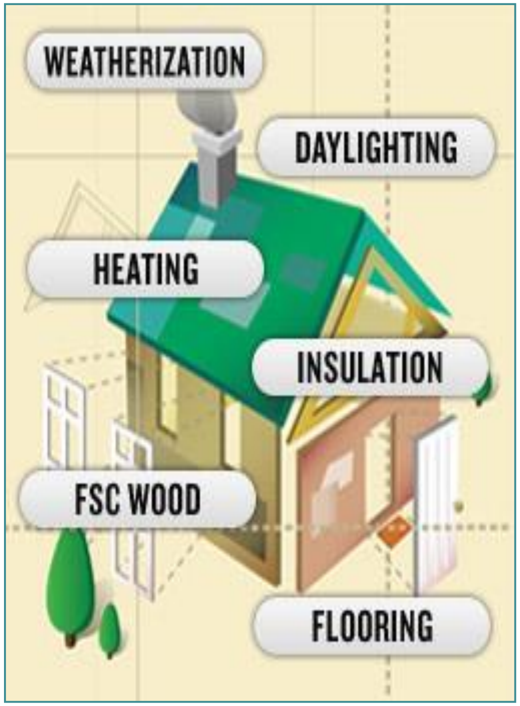
---

- Software Engineering
  - Semantic Search
  - Code Search, Search based SE
  - Program Synthesis, Optimization
  - Test prioritization
  - Requirements Engineering
- Databases – Skyline queries
- Stable Marriage problems
- AI Planning, configuration
- Recommender Systems

Applications

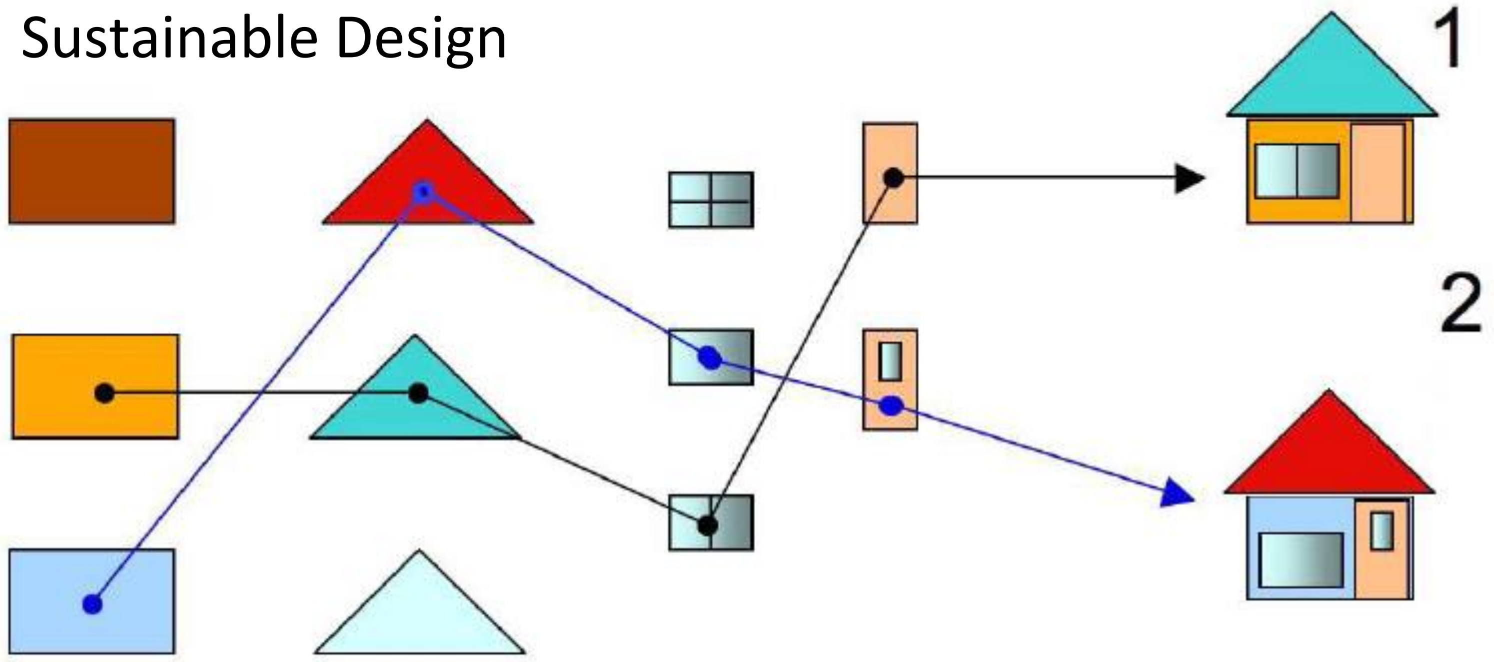
- Sustainable Design





Applications

- Sustainable Design



Function	Component	IC	FC	RE	TG
Heating	Electric	G	B	B	B
Heating	Gas	A	G	B	B
Heating	Solar	P	E	E	E
Flooring	Ceramic Tile	A	E	B	B
Flooring	Vinyl Tile	E	G	A	G
Flooring	Natural Cork	P	E	G	E
Siding	Brick&Mortar	P	E	P	B
Siding	Aluminum	G	G	G	A
Siding	Cedar	A	A	G	G

Table 1: Available Building Components in the Repository

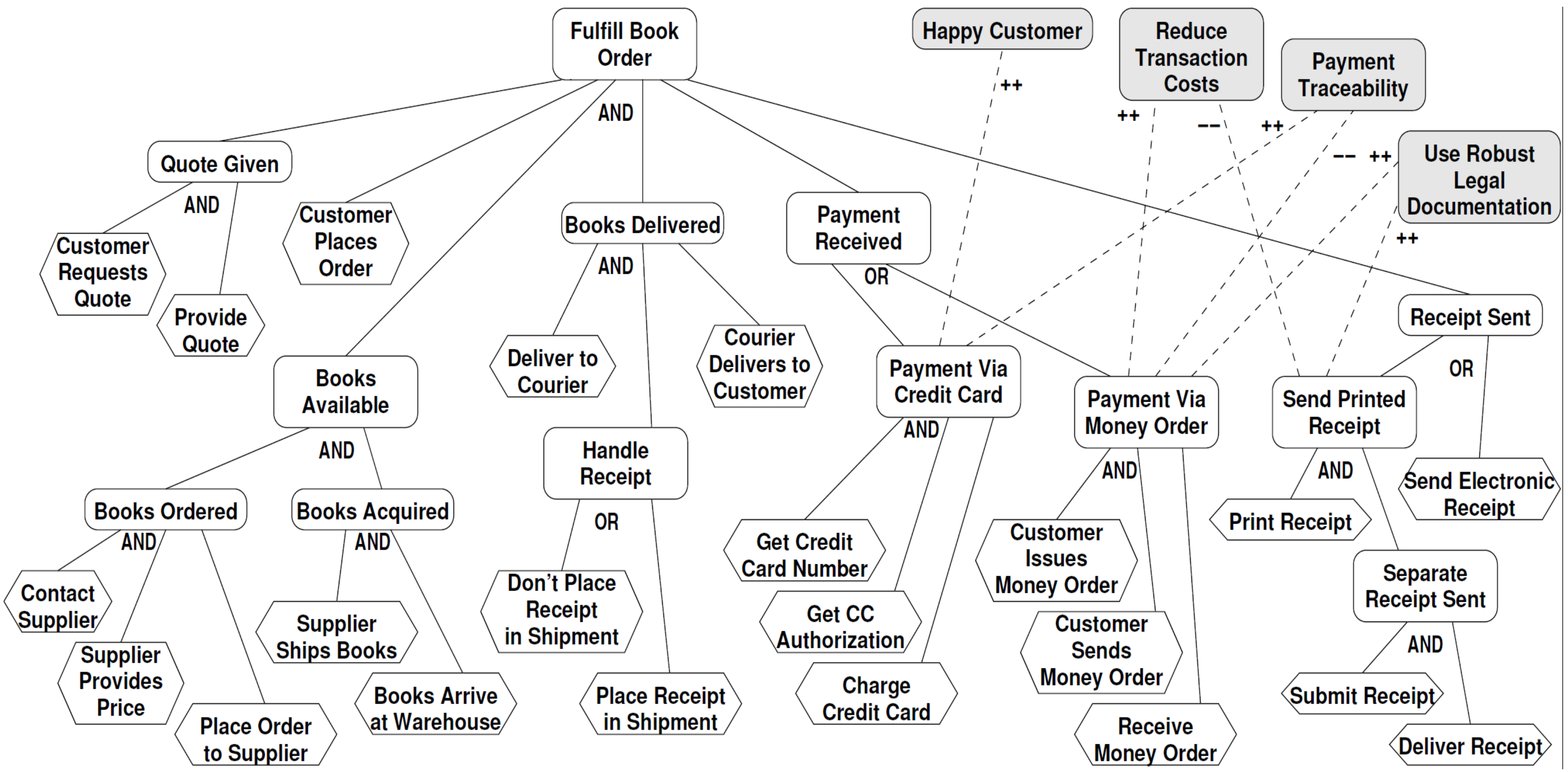
Design	Heating	Flooring	Siding
D_1	Electric	Vinyl Tile	Aluminum
D_2	Gas	Ceramic Tile	Brick&Mortar
D_3	Gas	Vinyl Tile	Aluminum
D_4	Solar	Ceramic Tile	Brick&Mortar
D_5	Solar	Natural Cork	Aluminum

Table 2: Candidate Building Designs

Applications

- Goal Oriented Requirements Engineering

Oster et al. ASE 2011

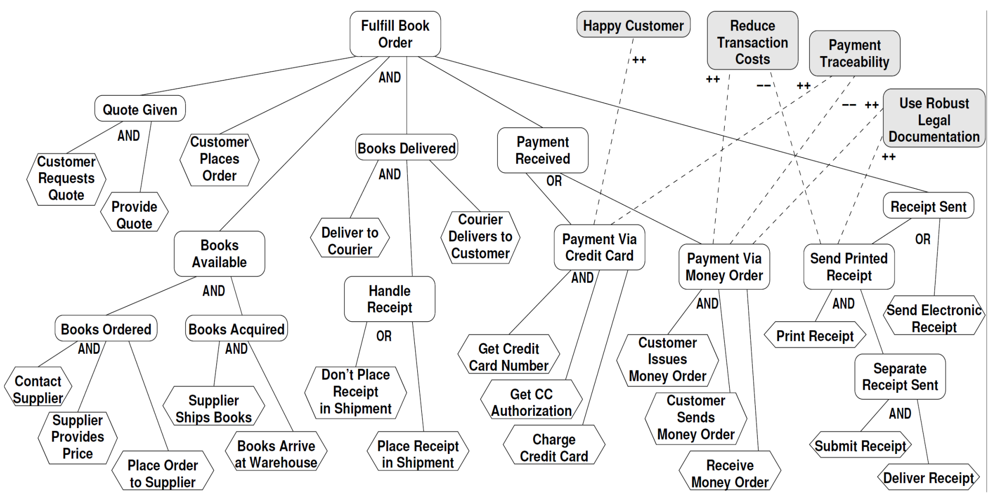


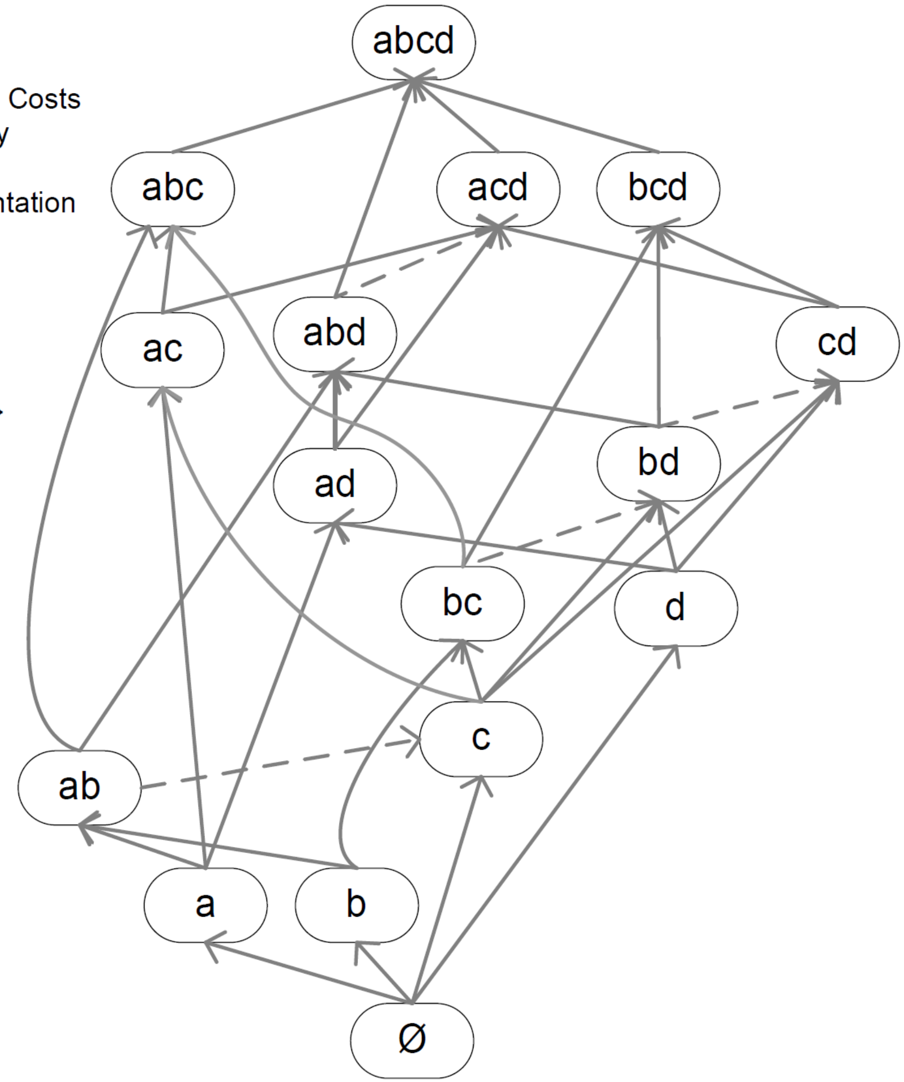
Goal oriented Requirements Engineering – CI-nets

a: Happy Customer
b: Reduce Transaction Costs
c: Payment Traceability
d: Use Robust Legal Documentation

<u>CI-net statements</u>

{d} ; {} : {c} ; {b}
{b} ; {a} : {d} ; {c}
{} ; {d} : {c} ; {a,b}





Oster et al. ASE 2011

Applications - Minimizing Credential Disclosure

- User needs renter's insurance for new apartment
  - Which service to choose to get a quote?
  - Privacy issue – disclosure of sensitive credentials
- All services do the same tasks (from user's perspective) info:

Oster et al. FACS 2012

#	*Name*	*Required Sensitive Information*
1	QuickQuote	Address, Bank Account #
2	InsureBest	Name, Address, Bank Routing #
3	EZCoverage	Name, Address
4	BankMatch	Bank Routing #

<u>User's Preferences:</u>

P1. If bank account number is disclosed, then I would rather give my address than bank routing number to the server

P2. If I have to disclose my address but not my name, then I would prefer to give my bank routing number rather than my bank account number

P3. If I don't need to disclose my bank account number, I will give my name and address instead of my bank routing number.

Applications - Minimizing Credential Disclosure

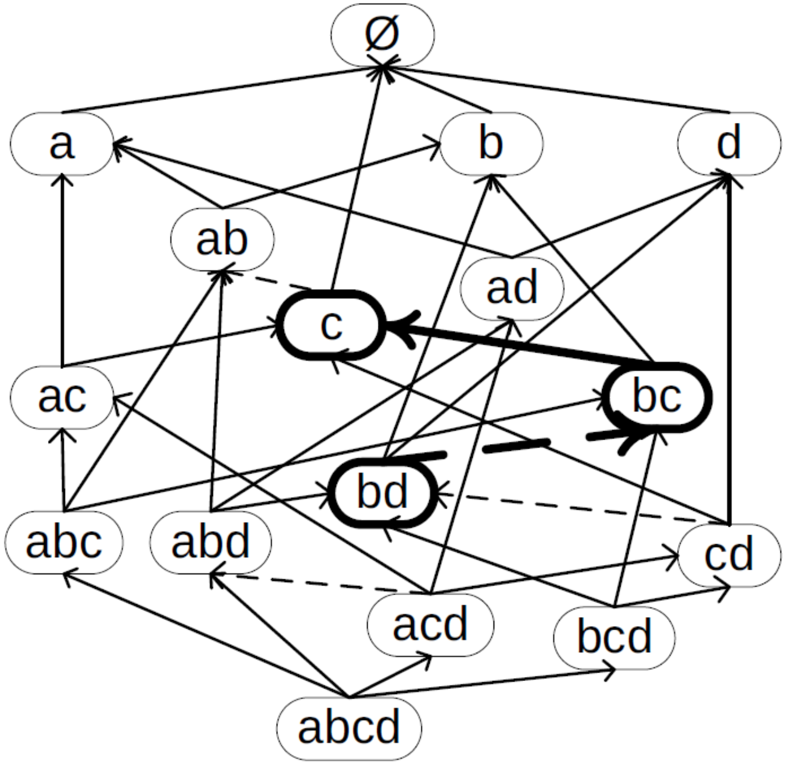
- Finding a sequence of next-preferred
  - Suboptimal sequence of preferred sets of credentials can compromise privacy,

    when it could have been avoided

Oster et al. FACS 2012

a = Name
b = Address
c = Bank Routing Number
d = Bank Account Number

P1. $\{d\}, \{\} \; : \; \{b\} \succ \{c\}$
P2. $\{b\}, \{a\} \; : \; \{c\} \succ \{d\}$
P3. $\{\}, \{d\} \; : \; \{a, b\} \succ \{c\}$



Part V – Tool

Part V

Tool

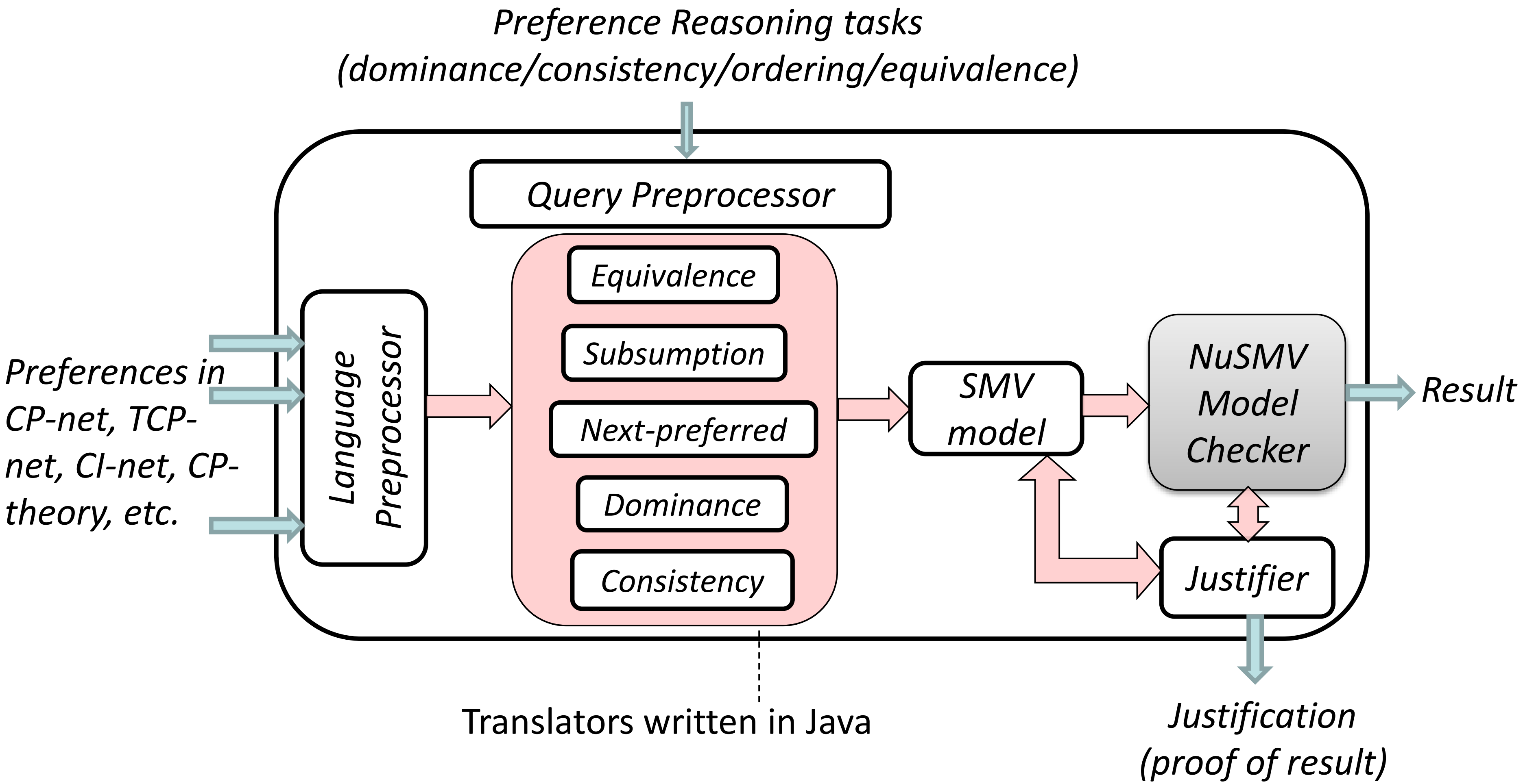
iPref-R Preference Reasoning Tool

- α-version of iPref-R freely available at
  - http://fmg.cs.iastate.edu/project-pages/preference-reasoner/
  - http://fmg.cs.iastate.edu/project-pages/GUI-iPref-R/
- Demo http://fmg.cs.iastate.edu/project-pages/GUI-iPref-R/video/demo.html
- Currently supports representing and reasoning with
  - CI-nets
  - CP-nets
  - Support for other languages in progress
- Reasoning tasks supported
  - Dominance Testing
  - Consistency
  - Next-preferred (for acyclic CP/CI-nets)
  - Support for Equivalence & Subsumption testing coming

iPref-R Architecture

- Architecture decouples preference reasoning from choice of
  - Model checker
  - Translation of preference
  - Preference languages
  - Modularization enables extension to other ceteris paribus languages, reasoning tasks and encodings

iPref-R Architecture



Summary

I. Qualitative Preference Languages
  - Representation : Syntax of languages CP-nets, TCP-nets, CI-nets, CP-Theories

II. Qualitative Preference Languages
  - Ceteris Paribus semantics: the induced preference graph (IPG)
  - Reasoning: Consistency, Dominance, Ordering, Equivalence & Subsumption
  - Complexity of Reasoning

III. Practical aspects: Preference Reasoning via Model Checking
  - From ceteris paribus semantics (IPG) to Kripke structures
  - Specifying and verifying properties in temporal logic
  - Translating Reasoning Tasks into Temporal Logic Properties

Summary

IV. Applications

- *Engineering*: Civil, Software (SBSE, RE, Services), Aerospace, Manufacturing
- *Security*: Credential disclosure, Cyber-security
- *Algorithms*: Search, Stable Marriage, Allocation, Planning, Recommender systems
- *Environmental applications*: Risk Assessment, Policy decisions, Environmental impact, Computational Sustainability

V. iPref-R Tool

- A tool that does well in practice for a known hard problem
- Architecture
- Demo
- Use of iPref-R in Security, Software Engineering

References

- Ganesh Ram Santhanam, Samik Basu and Vasant Honavar. ***Verifying Preferential Equivalence & Subsumption via Model Checking.*** 3rd International Conference on Algorithmic Decision Theory. 2013.
- Zachary Oster, Ganesh Ram Santhanam, Samik Basu and Vasant Honavar. ***Model Checking of Qualitative Sensitivity Preferences to Minimize Credential Disclosure.*** International Symposium on Formal Aspects of Component Software (FACS) 2012.
- Ganesh Ram Santhanam, Samik Basu, Vasant Honavar: ***Representing and Reasoning with Qualitative Preferences for Compositional Systems***. J. Artif. Intell. Res. (JAIR) 2011.
- Ganesh Ram Santhanam Samik Basu and Vasant Honavar. ***Identifying Sustainable Designs Using Preferences over Sustainability Attributes***. AAAI Spring Symposium: Artificial Intelligence and Sustainable Design 2011.
- Ganesh Ram Santhanam, Zachary J. Oster, Samik Basu***: Identifying a preferred countermeasure strategy for attack graphs.*** CSIIRW 2013
- Ganesh Ram Santhanam, Samik Basu and Vasant Honavar. ***Dominance Testing via Model Checking***. AAAI Conference on Artificial Intelligence 2010.
- Ganesh Ram Santhanam, Samik Basu and Vasant Honavar. ***Efficient Dominance Testing for Unconditional Preferences***. International Conference on the Principles of Knowledge Representation and Reasoning 2010.
- Ganesh Ram Santhanam and Kasthurirangan Gopalakrishnan. ***Pavement Life-Cycle Sustainability Assessment and Interpretation Using a Novel Qualitative Decision Procedure***. Journal of Computing in Civil Engineering 2013.
- Zachary J. Oster, Ganesh Ram Santhanam, Samik Basu: ***Automating analysis of qualitative preferences in goal-oriented requirements engineering***. ASE 2011.
- G. Brewka, M. Truszczynski; S. Woltran. ***Representing Preferences Among Sets***. AAAI 2010
- G. Brewka, I. Niemelä, M. Truszczynski. ***Preferences and Nonmonotonic Reasoning***, AI Magazine (special issue on preference handling) 2008.
- G. Brewka. ***Complex Preferences for Answer Set Optimization***, KR 2004.
- G. Brewka. ***Answer Sets and Qualitative Decision Making***, Synthese 2005.
- G. Brewka, I. Niemelä, T. Syrjänen. ***Logic Programs wirh Ordered Disjunction***, Computational Intelligence 2004.
- Ronen I. Brafman, Enrico Pilotto, Francesca Rossi, Domenico Salvagnin, Kristen Brent Venable, Toby Walsh: ***The Next Best Solution***. AAAI 2011.
- Ronen I. Brafman, Carmel Domshlak: ***Preference Handling - An Introductory Tutorial***. AI Magazine 2009.
- Maxim Binshtok, Ronen I. Brafman, Carmel Domshlak, Solomon Eyal Shimony: ***Generic Preferences over Subsets of Structured Objects***. J. Artif. Intell. Res. (JAIR) 2009.
- Ronen I. Brafman, Carmel Domshlak, Solomon Eyal Shimony: ***On Graphical Modeling of Preference and Importance***. J. Artif. Intell. Res. (JAIR) 2006.
- Ronen I. Brafman, Carmel Domshlak, Solomon Eyal Shimony, Y. Silver: ***Preferences over Sets.*** AAAI 2006.
- Ronen I. Brafman, Yuri Chernyavsky: ***Planning with Goal Preferences and Constraints***. ICAPS 2005.

References

- Craig Boutilier, Ronen I. Brafman, Carmel Domshlak, Holger H. Hoos, David Poole: ***CP-nets: A Tool for Representing and Reasoning with Conditional Ceteris Paribus Preference Statements***. J. Artif. Intell. Res. (JAIR) 2004.
- Carmel Domshlak, Ronen I. Brafman: CP-nets: ***Reasoning and Consistency Testing***. KR 2002.
- Ronen I. Brafman, Carmel Domshlak: ***Introducing Variable Importance Tradeoffs into CP-Nets***. UAI 2002.
- Sotirios Liaskos, Sheila A. McIlraith, Shirin Sohrabi, John Mylopoulos: ***Representing and reasoning about preferences in requirements engineering.*** Requir. Eng. 2011.
- Shirin Sohrabi, Jorge A. Baier, Sheila A. McIlraith: ***Preferred Explanations: Theory and Generation via Planning***. AAAI 2011.
- Shirin Sohrabi, Sheila A. McIlraith: ***Preference-Based Web Service Composition: A Middle Ground between Execution and Search***. International Semantic Web Conference (1) 2010.
- Jorge A. Baier, Sheila A. McIlraith: ***Planning with Preferences***. AI Magazine 2008.
- Zachary J. Oster: ***Reasoning with qualitative preferences to develop optimal component-based systems***. ICSE 2013.
- Judy Goldsmith, Jérôme Lang, Miroslaw Truszczynski, Nic Wilson: ***The Computational Complexity of Dominance and Consistency in CP-Nets***. J. Artif. Intell. Res. (JAIR) 2008.
- Judy Goldsmith, Ulrich Junker: Preference ***Handling for Artificial Intelligence***. AI Magazine 2008.
- Walid Trabelsi, Nic Wilson, Derek G. Bridge: ***Comparative Preferences Induction Methods for Conversational Recommenders***. ADT 2013
- Nic Wilson: ***Importance-based Semantics of Polynomial Comparative Preference Inference***. ECAI 2012.
- Walid Trabelsi, Nic Wilson, Derek G. Bridge, Francesco Ricci: ***Preference Dominance Reasoning for Conversational Recommender Systems: a Comparison between a Comparative Preferences and a Sum of Weights Approach.*** International Journal on Artificial Intelligence Tools 2011.
- Meghyn Bienvenu, Jérôme Lang, Nic Wilson: ***From Preference Logics to Preference Languages***, and Back. KR 2010
- Sylvain Bouveret, Ulle Endriss, Jérôme Lang: ***Conditional Importance Networks: A Graphical Language for Representing Ordinal, Monotonic Preferences over Sets of Goods.*** IJCAI 2009.
- Sylvain Bouveret, Ulle Endriss, Jérôme Lang***: Fair Division under Ordinal Preferences: Computing Envy-Free Allocations of Indivisible Goods***. ECAI 2010.
- Francesca Rossi, Kristen Brent Venable, Toby Walsh: ***A Short Introduction to Preferences: Between Artificial Intelligence and Social Choice.*** Synthesis Lectures on Artificial Intelligence and Machine Learning 2011.
- Ronen I. Brafman, Francesca Rossi, Domenico Salvagnin, Kristen Brent Venable, Toby Walsh: ***Finding the Next Solution in Constraint- and Preference-Based Knowledge Representation Formalisms***. KR 2010.
- Maria Silvia Pini, Francesca Rossi, Kristen Brent Venable, Toby Walsh***: Stable marriage problems with quantitative preferences.*** CoRR abs/1007.5120 2010.
- Francesca Rossi, Kristen Brent Venable, Toby Walsh: ***Preferences in Constraint Satisfaction and Optimization***. AI Magazine 2008.

Thank you

Collaborators

Dr. Vasant Honavar

Dr. Samik Basu

Dr. Giora Slutzki

Dr. Kasthurirangan Gopalakrishnan

Dr. Robyn Lutz

Dr. Zachary Oster

Carl Chapman

Katerina Mitchell

Acknowledgements

NSF Grants IIS 0711356, CCF 0702758, CCF 1143734, CNS 1116050