Preference Trees over Combinatorial Domains

Xudong Liu

Ph.D. Candidate
Advisor: Dr. Miroslaw Truszczynski

Department of Computer Science
College of Engineering
University of Kentucky
Lexington, KY, USA

Wednesday, 1/13/2016

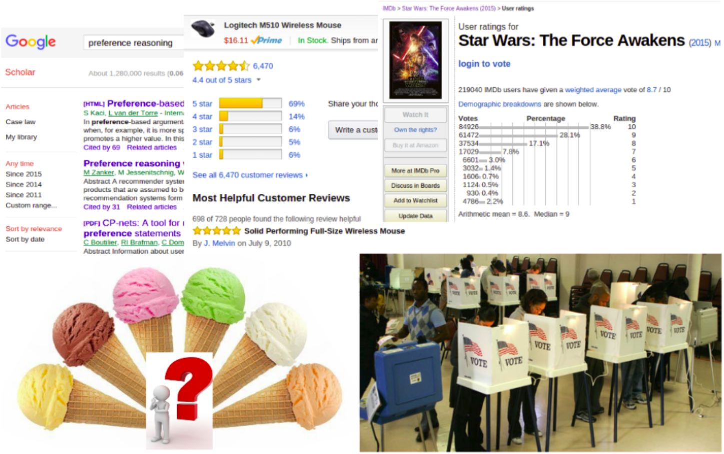


Figure : Preferences of different forms

Describing Preferences

Figure : How to express preferences?

1. On scale of 0 to 99, how will I rate these two cars?
    - I give Car1 44 points and Car2 78 points; thus, I prefer Car2 to Car1.
2. What are the desired properties I see in cars?
    - I prefer minivans to sedans.
    - If minivan, I prefer gray to blue; if sedan, I prefer blue to gray; ...

Describing Preferences

Car1 Car2

Figure : How to express preferences?

1. On scale of 0 to 99, how will I rate these two cars? (**Quantitative**)
   - I give Car1 44 points and Car2 78 points; thus, I prefer Car2 to Car1.
2. What are the desired properties I see in cars? (**Qualitative**)
   - I prefer minivans to sedans.
   - If minivan, I prefer gray to blue; if sedan, I prefer blue to gray; ...

Relations and Orderings

Binary Relations

Let O be a set of elements. A *binary relation* R over O is a collection of ordered pairs of elements in O; that is,

$$R \subseteq O \times O.$$

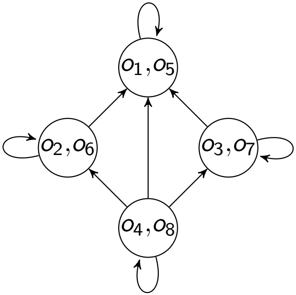
Properties of binary relations related to preferences:

1. Reflexivity: $\forall o \in O$, $(o, o) \in R$.
2. Irreflexivity: $\forall o \in O$, $(o, o) \notin R$.
3. Totality: $\forall o_1, o_2$, $(o_1, o_2) \in R$ or $(o_2, o_1) \in R$.
4. Transitivity: $\forall o_1, o_2, o_3$, if $(o_1, o_2) \in R$ and $(o_2, o_3) \in R$, then $(o_1, o_3) \in R$.
5. Symmetry: $\forall o_1, o_2$, if $(o_1, o_2) \in R$, then $(o_2, o_1) \in R$.
6. Antisymmetry: $\forall o_1, o_2$, if $(o_1, o_2) \in R$ and $(o_2, o_1) \in R$, then $o_1 = o_2$.

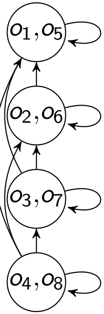
Relations and Orderings

Orderings

$\succeq$ is a *partial preorder* if it is reflexive and transitive, a *total preorder* if it is a partial preorder and total, a *partial order* if it is a partial preorder and antisymmetric, and a *total order* if it is a partial order and total.



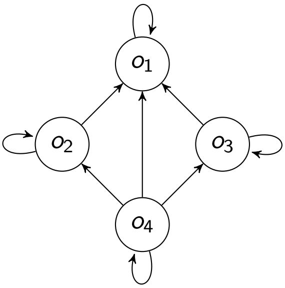
(a) partial preorder



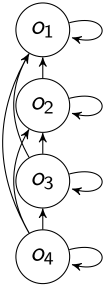
(b) total preorder

Orderings

$\succeq$ is a *partial preorder* if it is reflexive and transitive, a *total preorder* if it is a partial preorder and total, a *partial order* if it is a partial preorder and antisymmetric, and a *total order* if it is a partial order and total.



(a) partial order

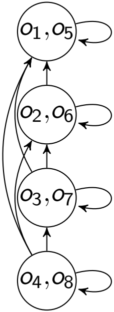


(b) total order

Relations and Orderings

Preference Relations

Let $\succeq$ be a preference relation that is a total preorder over O. We say that o_1 is *weakly preferred* to o_2 if $o_1 \succeq o_2$, that o_1 is *strictly preferred* ($\succ$) to o_2 if $o_1 \succeq o_2$ and $o_2 \not\succeq o_1$, and that o_1 is *indifferent* ($\approx$) from o_2 if $o_1 \succeq o_2$ and $o_2 \succeq o_1$.



(a) total preorder

$$o_1 \succeq o_5,$$
$$o_2 \succ o_4,$$
$$o_4 \approx o_8,$$

(b) preferences

Combinatorial Domains

Combinatorial Domains

Let V be a finite set of variables $\{X_1, \ldots, X_p\}$, associated with a set of finite domains $\{Dom(X_1), \ldots, Dom(X_p)\}$ for each variable X_i. A *combinatorial domain* $CD(V)$ is a set of *outcomes* described by combinations of values from $Dom(X_i)$:

$$CD(V) = \prod_{X_i \in V} Dom(X_i).$$

Combinatorial Domains: Example

Domain of cars over set V of p binary variables:

1. **BodyType**: {mvan, sedan}.
2. **Capacity**: {5, 7m}.
3. **Color**: {blue, gray}.

$\vdots$

$$CD(V) = \underbrace{\{\langle \text{sedan, 5, blue, } \ldots \rangle, \langle \text{mvan, 7m, gray, } \ldots \rangle, \ldots\}}_{2^p \text{ outcomes, too many!}}.$$

Combinatorial Domains: Example

Domain of cars:

1. **BodyType**: {mvan, sedan, sport, suv}.
2. **Capacity**: {2, 5, 7m}.
3. **Color**: {black, blue, gray, red, white}.
4. **LuggageSize**: {big, med, small}.
5. **Make**: {bmw, ford, honda, vw}.
6. **Price**: {low, med, high, vhigh}.
7. **Safety**: {low, med, high}.

Single Agent

Figure : Dominance and Optimization

Multi-Agent

Figure : Social Choice and Welfare

Research Problems of Interest

1. Preference representation formalisms to compactly model qualitative preferences over combinatorial domains.
2. Preference elicitation and learning methods to cast preferences of agents as a theory in a preference formalism.
3. Preference reasoning tasks:
   - Dominance and optimization
   - Manipulation: better off by misreporting preferences.

Q: How do we compactly represent qualitative preferences over combinatorial domains?

1. Preference Trees (P-trees)[1,13]
2. Partial Lexicographic Preference Trees (PLP-trees)[8]
3. Lexicographic Preference Trees (LP-trees)[4,12]

---

[1] Niall M Fraser. "Ordinal preference representations". In: Theory and Decision (1994)

[2] Xudong Liu and Miroslaw Truszczynski. "Preference Trees: A Language for Representing and Reasoning about Qualitative Preferences". In: Proceedings of the 8th Multidisciplinary Workshop on Advances in Preference Handling (MPREF). 2014

[3] Xudong Liu and Miroslaw Truszczynski. "Learning Partial Lexicographic Preference Trees over Combinatorial Domains". In: Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI). 2015

[4] Richard Booth et al. "Learning conditionally lexicographic preference relations". In: ECAI. 2010

[5] Xudong Liu and Miroslaw Truszczynski. "Aggregating Conditionally Lexicographic Preferences Using Answer Set Programming Solvers". In: Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT). 2013

Preference Learning

Q: How do we learn predictive qualitative preference models over combinatorial domains?

1. Partial Lexicographic Preference Trees (PLP-trees)[6,7,8]
   - Active and passive learning
   - Compute a (possibly small) PLP-tree consistent with all the data
   - Compute a PLP-tree that agrees with the data as much as possible
2. Preference Forests[9]
3. Preference Approximation[10]

---

[6] Michael Schmitt and Laura Martignon. "On the complexity of learning lexicographic strategies". In: The Journal of Machine Learning Research (2006)

[7] József Dombi, Csanád Imreh, and Nándor Vincze. "Learning lexicographic orders". In: European Journal of Operational Research (2007)

[8] Xudong Liu and Miroslaw Truszczynski. "Learning Partial Lexicographic Preference Trees over Combinatorial Domains". In: Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI). 2015

[9] Xudong Liu and Miroslaw Truszczynski. "Learning Preference Trees and Forests". In: Preparation

[10] Xudong Liu and Miroslaw Truszczynski. "Approximating Conditional Preference Networks Using Lexicographic Preference Trees". In: Preparation

Preference Reasoning

Q: How do we reason about preferences over combinatorial domains?

1. Preference Optimization[11,12,13,14]:
   - Dominance testing: $o_1 \succeq_P o_2$?
   - Optimality testing: $o_1 \succeq_P o_2$ for all $o_2 \neq o_1$?
   - Optimality computing: what is the optimal outcome wrt P?
   - Preference aggregation: which candidate wins the election?
2. Preference Misrepresentation[15,16]:
   - Manipulation

---

[11] Jérôme Lang, Jérôme Mengin, and Lirong Xia. "Aggregating Conditionally Lexicographic Preferences on Multi-issue Domains". In: CP. 2012

[12] Xudong Liu and Miroslaw Truszczynski. "Aggregating Conditionally Lexicographic Preferences Using Answer Set Programming Solvers". In: Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT). 2013

[13] Xudong Liu and Miroslaw Truszczynski. "Preference Trees: A Language for Representing and Reasoning about Qualitative Preferences". In: Proceedings of the 8th Multidisciplinary Workshop on Advances in Preference Handling (MPREF). 2014

[14] Xudong Liu and Miroslaw Truszczynski. "Reasoning with Preference Trees over Combinatorial Domains". In: Proceedings of the 4th International Conference on Algorithmic Decision Theory (ADT). 2015

[15] Felix Brandt, Vincent Conitzer, and Ulle Endriss. "Computational social choice". In: Multiagent systems (2012)

[16] Xudong Liu and Miroslaw Truszczynski. "Complexity of Manipulation in Elections Where Votes Are Lexicographic Preference Trees". In: Preparation

Preference Applications

Q: What fields can we apply preferences to?

1. Role-playing Games:
   - Hedonic games[17]
2. Automated Planning and Scheduling:
   - Trip planning[18]
3. Data-Driven Decision Making:
   - Predictive models[19]

[17] Matthew Spradling et al. "Roles and Teams Hedonic Game". In: Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT). 2013

[18] Xudong Liu et al. "On Personalizability and Extensibility of Multi-Modal Trip Planning". In: PARC Symposium. 2015

[19] Xudong Liu and Miroslaw Truszczynski. "Learning Preference Trees and Forests". In: Preparation

Outline

1. The languages of P-trees, PLP-trees, and LP-trees
2. Learning preference models in case of PLP-trees
3. Reasoning with preferences:
    - Preference optimization in case of P-trees
    - Computing winners and “strong” outcomes when votes are LP-trees
    - Application in trip planning
4. Future research directions

Outline

1. The languages of P-trees, PLP-trees, and LP-trees
2. Learning preference models in case of PLP-trees
3. Reasoning with preferences:
    - Preference optimization in case of P-trees
    - Computing winners and "strong" outcomes when votes are LP-trees
    - Application in trip planning
4. Future research directions

Preference Trees (P-Trees)

Let φ, ψ, and π be propositional formulas over the set $\mathcal{L}$ of literals that are values from $\bigcup_{X_i \in V} Dom(X_i)$.

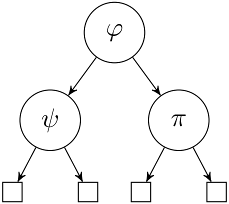


Figure : A P-tree

$$\varphi \wedge \psi \succ \varphi \wedge \neg\psi \succ \neg\varphi \wedge \pi \succ \neg\varphi \wedge \neg\pi.$$

Preference Trees (P-Trees)

Let φ, ψ, and π be propositional formulas over the set $\mathcal{L}$ of literals that are values from $\bigcup_{X_i \in V} Dom(X_i)$.

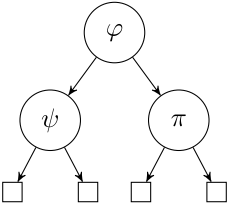


Figure : A P-tree

$$\varphi \wedge \psi \succ \varphi \wedge \neg\psi \succ \neg\varphi \wedge \pi \succ \neg\varphi \wedge \neg\pi.$$

Total preorder

Example: The Cars Domain

1. **BodyType**(X_1): {mvan($x_{1,1}$), sedan($x_{1,2}$), sport($x_{1,3}$), suv($x_{1,4}$)}.
2. **Capacity**(X_2): {2, 5, 7m}.
3. **Color**(X_3): {black, blue, gray, red, white}.
4. **LuggageSize**(X_4): {big, med, small}.
5. **Make**(X_5): {bmw, ford, honda, vw}.
6. **Price**(X_6): {low, med, high, vhigh}.
7. **Safety**(X_7): {low, med, high}.

Example: Preference Trees over Cars

BodyType(X_1): {mvan($x_{1,1}$), sedan($x_{1,2}$), sport($x_{1,3}$), suv($x_{1,4}$)}.
Color(X_3): {black, blue, gray, red, white}.
Price(X_6): {low, med, high, vhigh}.

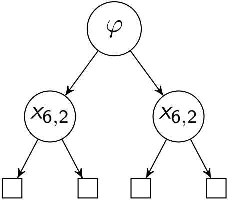


Figure : A P-tree over cars[20]

[20] $\varphi = (x_{1,1} \wedge x_{3,5}) \vee (x_{1,2} \wedge x_{3,2})$.

Example: Preference Trees over Cars

BodyType(X_1): {mvan($x_{1,1}$), sedan($x_{1,2}$), sport($x_{1,3}$), suv($x_{1,4}$)}.
Color(X_3): {black, blue, gray, red, white}.
Price(X_6): {low, med, high, vhigh}.

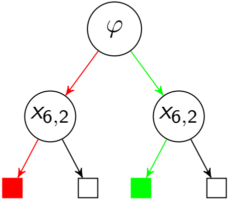


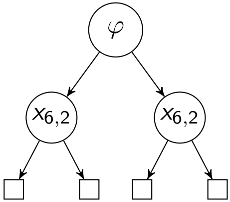
Figure : A P-tree over cars[20]

$Car2 \succ Car1$

[20] $\varphi = (x_{1,1} \wedge x_{3,5}) \vee (x_{1,2} \wedge x_{3,2})$.

Compact Representation of P-trees

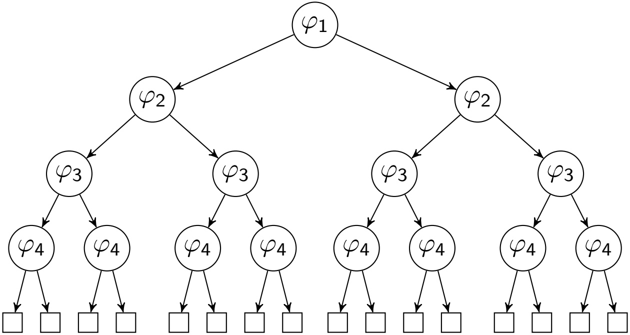
BodyType(X_1): {mvan($x_{1,1}$), sedan($x_{1,2}$), sport($x_{1,3}$), suv($x_{1,4}$)}.
Color(X_3): {black, blue, gray, red, white}.
Price(X_6): {low, med, high, vhigh}.



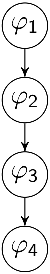
(a) Full (b) Compact

Figure : Compact P-trees

Compact Representation of P-trees



(a) Full



(b) Compact

Figure : Compact P-trees

Compact Representation of P-trees

A *compact P-tree* over $CD(\mathcal{I})$ is a binary tree where

1. every node is labeled with a Boolean formula over $\mathcal{I}$, and
2. every non-leaf node t labeled with φ has either two outgoing edges (Fig. (a)), or one outgoing edge pointing straight-down (Fig. (b)), left (Fig. (c)), or right (Fig. (d)).

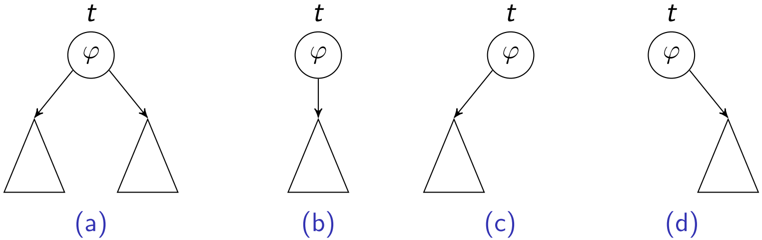
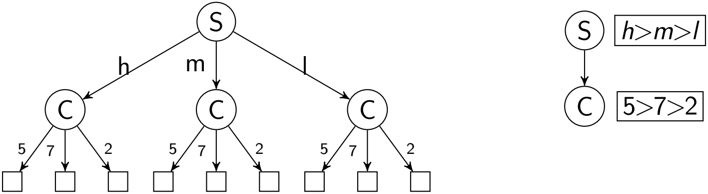


Figure : Compact P-trees

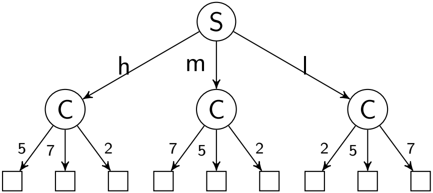
Partial Lexicographic Preference Trees (PLP-Tree)



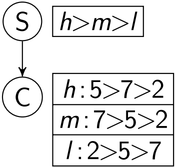
(a) Full (b) Compact

Figure : Unconditional Importance & Unconditional Preference (UIUP)

Partial Lexicographic Preference Trees (PLP-Tree)



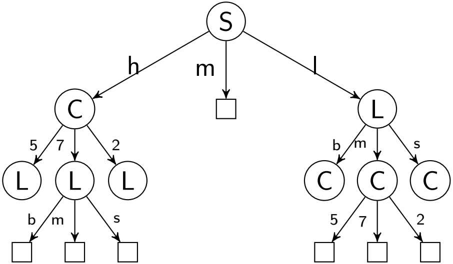
(a) Full



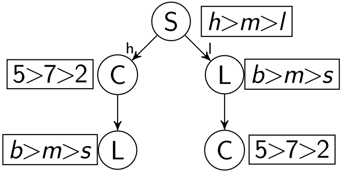
(b) Compact

Figure : Unconditional Importance & Conditional Preference (UICP)

Partial Lexicographic Preference Trees (PLP-Tree)



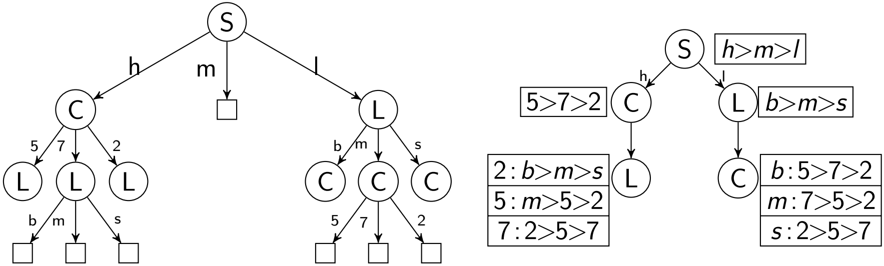
(a) Full



(b) Compact

Figure : Conditional Importance & Unconditional Preference (CIUP)

Partial Lexicographic Preference Trees (PLP-Tree)



(a) Full (b) Compact

Figure : Conditional Importance & Conditional Preference (CICP)

Lexicographic Preference Trees (LP-Trees)

1. An *LP-tree* $\mathcal{L}$ over $CD(\mathcal{I})$ is a PLP-tree, where
    - each attribute appears **exactly once** on every path from the root to a leaf.
    - Unlike P-trees and PLP-trees, an LP-tree induces a total order.

Outline

1. The languages of P-trees, PLP-trees, and LP-trees
2. Learning preference models in case of PLP-trees
3. Reasoning with preferences:
   - Preference optimization in case of P-trees
   - Computing winners and “strong” outcomes when votes are LP-trees
   - Application in trip planning
4. Future research directions

Learning PLP-trees

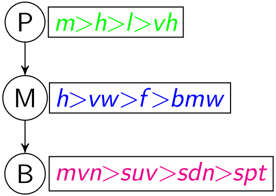
Consistent Learning (ConsLearn)

Given an example set $\mathcal{E}$, decide whether there exists a PLP-tree T (of a particular type) such that T is consistent with $\mathcal{E}$.

(<sdn,5,blk,m,h,m,m>,<suv,7m,wht,b,f,m,m>)

(<spt,2,wht,s,bmw,h,h>,<spt,2,wht,s,bmw,vh,h>)

(<mvn,7m,gry,b,f,m,m>,<sdn,5,bl,m,f,m,m>)



UIUP tree

Learning PLP-trees

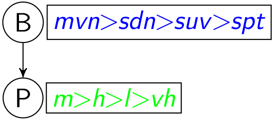
Small Learning (SmallLearn)

Given an example set $\mathcal{E}$ and a positive integer l ($l \leq |\mathcal{E}|$), decide whether there exists a PLP-tree T (of a particular type) such that T is consistent with $\mathcal{E}$ and $|T| \leq l$.

(<sdn,5,blk,m,h,m,m>,<suv,7m,wht,b,f,m,m>)

(<spt,2,wht,s,bmw,h,h>,<spt,2,wht,s,bmw,vh,h>)

(<mvn,7m,gry,b,f,m,m>,<sdn,5,bl,m,f,m,m>)



UIUP tree

Learning PLP-trees

Maixmal Learning (MaxLearn)

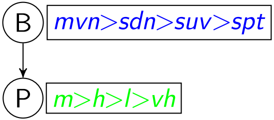
Given an example set $\mathcal{E}$ and a positive integer k ($k \leq m$), decide whether there exists a PLP-tree T (of a particular type) such that T satisfies at least k examples in $\mathcal{E}$.

(<sdn,5,blk,m,h,m,m>,<suv,7m,wht,b,f,m,m>)

(<spt,2,wht,s,bmw,h,h>,<spt,2,wht,s,bmw,vh,h>)

(<mvn,7m,gry,b,f,m,m>,<sdn,5,bl,m,f,m,m>)

(<suv,7m,gry,b,vw,vh,m>,<suv,7m,gry,b,vw,h,m>)



UIUP tree

Learning PLP-trees

Consistent Learning (ConsLearn)

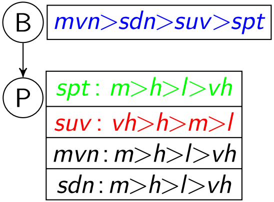
Given an example set $\mathcal{E}$, decide whether there exists a PLP-tree T (of a particular type) such that T is consistent with $\mathcal{E}$.

(<sdn,5,blk,m,h,m,m>,<suv,7m,wht,b,f,m,m>)

(<spt,2,wht,s,bmw,h,h>,<spt,2,wht,s,bmw,vh,h>)

(<mvn,7m,gry,b,f,m,m>,<sdn,5,bl,m,f,m,m>)

(<suv,7m,gry,b,vw,vh,m>,<suv,7m,gry,b,vw,h,m>)



UICP tree

Computational Complexity

1. P, NP, $coNP$: We typically believe that $P \subset NP$ and $P \subset coNP$.
2. Δ_2^P: P^{NP}, Σ_2^P: NP^{NP}, and Π_2^P: $coNP^{NP}$.
3. C-complete: hardest decision problems in class C.

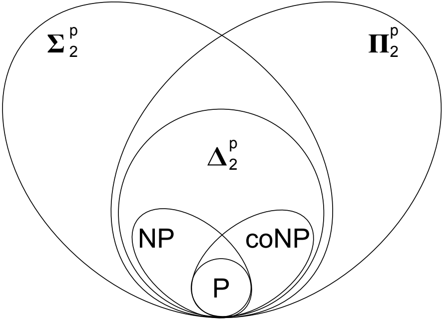


Figure : Computational complexity diagram

Complexity Results on PLP-trees

	UP	CP
UI	P	P
CI	NPC[21]	P

(a) ConsLearn

	UP	CP
UI	NPC	NPC
CI	NPC	NPC

(b) SmallLearn

	UP	CP
UI	NPC[22]	NPC
CI	NPC	NPC

(c) MaxLearn

Figure : Complexity results for learning PLP-trees

[21] Booth et al., *Learning Conditionally Lexicographic Preference Relations*, 2010.

[22] Schmitt and Martignon, *On the Complexity of Learning Lexicographic Strategies*, 2006.

Experimentation

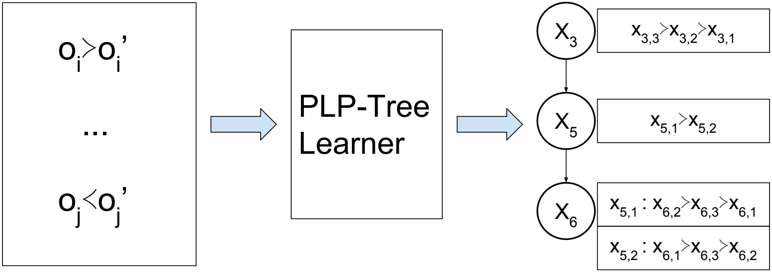


Figure : PLP-tree learning system

Datasets

Dataset	#Attributes	#Outcomes	#Examples
BreastCancerWisconsin	9	270	9009
CarEvaluation	6	1728	682721
CreditApproval	10	520	66079
GermanCredit	10	914	172368
Ionosphere	10	118	3472
MammographicMass	5	62	792
Mushroom	10	184	8448
Nursery	8	1266	548064
SPECTHeart	10	115	3196
TicTacToe	9	958	207832
Vehicle	10	455	76713
Wine	10	177	10322

Figure : Preference Learning Library[23]

[23] http://www.cs.uky.edu/~liu/preflearnlib.php

PLP-Trees To Learn

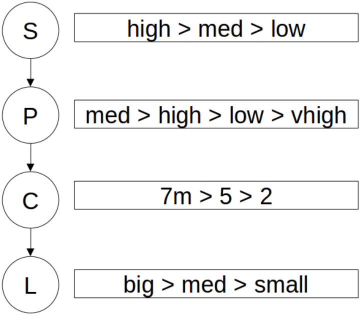


Figure : Unconditional Importance & Unconditional Preference (UIUP)

PLP-Trees To Learn

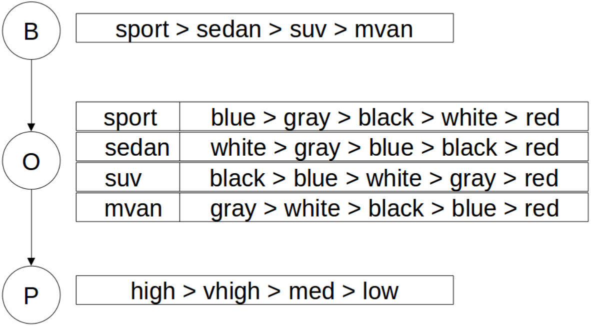


Figure : UICP with at most 1 parent (UICP-1)

PLP-Trees To Learn

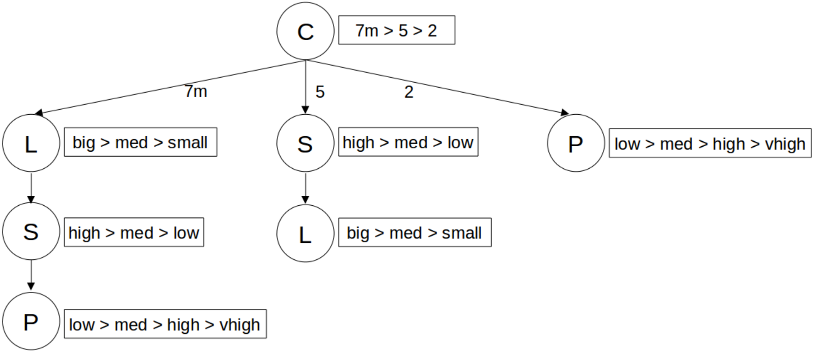


Figure : CIUP with 1 split at the root (CIUP-1)

PLP-Trees To Learn

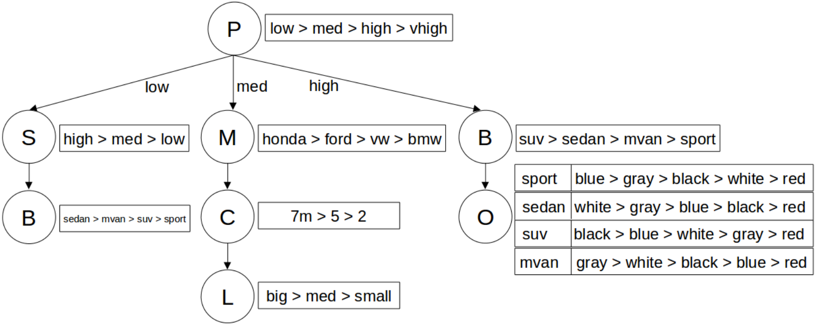
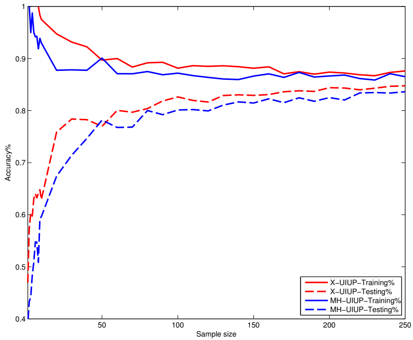


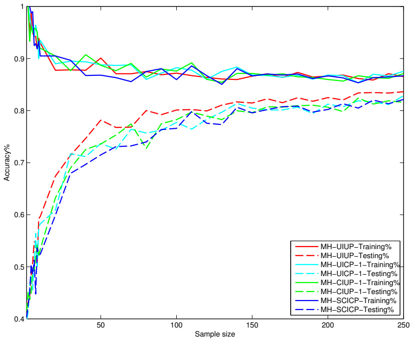
Figure : Simple CICP (SCICP)

Experimental Results: CarEvaluation[24]

#attributes:6, #outcomes:1728, #examples:682721



(a) Compare exact & greedy heuristic

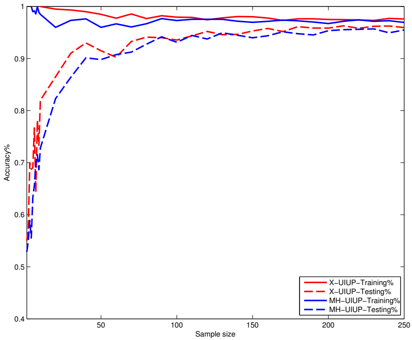


(b) Greedy heuristic

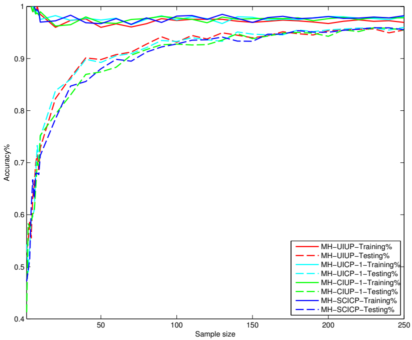
Figure : Learning curves solving MaxLearn

[24] http://www.cs.uky.edu/~liu/preflearnlib.php

#attributes:10, #outcomes:177, #examples:10322



(a) Compare exact & greedy heuristic



(b) Greedy heuristic

Figure : Learning curves solving MAXLEARN

[25] http://www.cs.uky.edu/~liu/preflearnlib.php

Outline

1. The languages of P-trees, PLP-trees, and LP-trees
2. Learning preference models in case of PLP-trees
3. Reasoning with preferences:
   - Preference optimization in case of P-trees
   - Computing winners and “strong” outcomes when votes are LP-trees
   - Application in trip planning
4. Future research directions

Computational Complexity Results for P-trees

Dominance-testing ($\textsc{DomTest}$): $o_1 \succ_T o_2$?
Optimality-testing ($\textsc{OptTest}$): o optimal w.r.t T?
Optimality-with-property ($\textsc{OptProp}$): is there optimal o with property α?

1. $\textsc{DomTest} \in P$
2. $\textsc{OptTest} \in coNP$-complete:
   - The complement problem is reduced from the SAT problem.
3. $\textsc{OptProp} \in \Delta_2^P$-complete:
   - The problem is reduced from the Maximum Satisfying Assignment (MSA) problem.

Outline

1. The languages of P-trees, PLP-trees, and LP-trees
2. Learning preference models in case of PLP-trees
3. Reasoning with preferences:
   - Preference optimization in case of P-tree
   - Computing winners and “strong” outcomes when votes are LP-trees
   - Application in trip planning
4. Future research directions

Positional Scoring Rules

- k-approval: $(1, \ldots, 1, 0, \ldots, 0)$ with k being the number of 1's and $m - k$ the number of 0's where $m = 2^p$.
- (k, l)-approval: $(a, \ldots, a, b, \ldots, b, 0 \ldots, 0)$, where a and b are constants $(a > b)$ and the numbers of a's and b's equal to k and l, respectively.
- b-Borda: $(b, b - 1, \ldots, 0)$, where if $b > m - 1$, b-Borda is reduced to the regular Borda rule with $(m - 1, m - 2, \ldots, 1, 0)$.

The Evaluation and Winner Problems

The Evaluation Problem

Let r be a positional scoring rule with a scoring vector w, $\mathcal{C}$ a class of LP-trees. Given a $\mathcal{C}$-profile P of n LP-trees over p attributes and a positive integer R, the *evaluation* problem is to decide whether there exists an alternative $o \in \mathcal{X}$ such that $s_w(o, P) \geq R$.

The Winner Problem

Let r be a positional scoring rule with a scoring vector w, $\mathcal{C}$ a class of LP-trees. Given a $\mathcal{C}$-profile P of n LP-trees over p attributes, the *winner* problem is to compute an alternative $o \in \mathcal{X}$ with the maximum score $s_w(o, P)$.

Complexity of the Evaluation Problem: k-Approval

	UP	CP
UI	P	P
CI	P	P

(a) $k = 2^{p-1} \pm f(p)$, $f(p)$ is a poly

	UP	CP
UI	NPC	NPC
CI	NPC	NPC

(b) $k = 2^{p-c}$, $c > 1$ is a const

Figure : k-Approval

Complexity of the Evaluation Problem: (k, l)-Approval

	UP	CP
UI	P	P
CI	P	P

(a) $k = l = 2^{p-1}$

	UP	CP
UI	NPC	NPC
CI	NPC	NPC

(b) $k = l = 2^{p-c}$, $c > 1$ is a const

Figure : (k, l)-Approval

Complexity of the Evaluation Problem: *b*-Borda

	UP	CP
UI	P	NPC
CI	NPC	NPC

(a) $b = 2^p - 1$

	UP	CP
UI	NPC	NPC
CI	NPC	NPC

(b) $b = 2^{p-c} - 1$, $c \geq 1$ is a const

Figure : *b*-Borda

Modeling the Problems in ASP

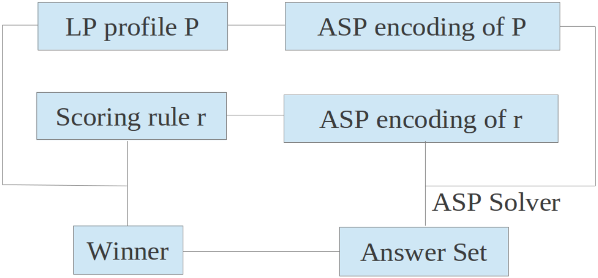


Figure : The winner problem

- Solvers: *clingo*[26], *clingcon*[27]

[26] M. Gebser et al. "Potassco: The Potsdam Answer Set Solving Collection". In: AI Communications (2011)

[27] Max Ostrowski and Torsten Schaub. "ASP modulo CSP: The clingcon system". In: TPLP (2012)

Modeling the Problems in W-MAXSAT

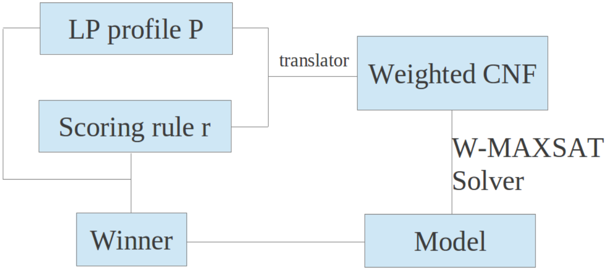


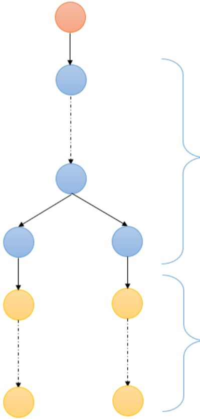
Figure : The winner problem

- Solver: *toulbar*[28]

[28] M Sanchez et al. “Max-CSP competition 2008: toulbar2 solver description”. In: the Third International CSP Solver Competition (2008)

Random LP Profiles

- To experiment with LP profiles, we developed methods to randomly generate *encodings* of a special type of CI-CP LP-tree of size linear in the number of attributes

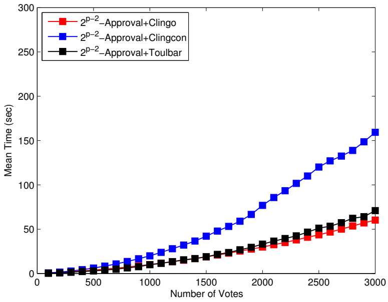


0_i	$0_r > 1_r$
1_i	$1_r > 0_r$

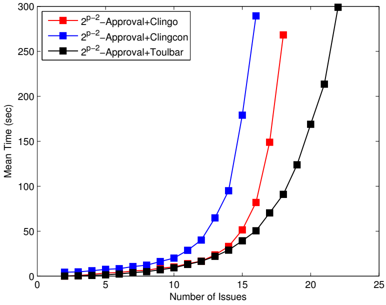
$0_j 0_k$	$0_s > 1_s$
$0_j 1_k$	$1_s > 0_s$
$1_j 0_k$	$0_s > 1_s$
$1_j 1_k$	$1_s > 0_s$

Figure : Random LP-tree

Varying p and n: 2^{p-2}-approval



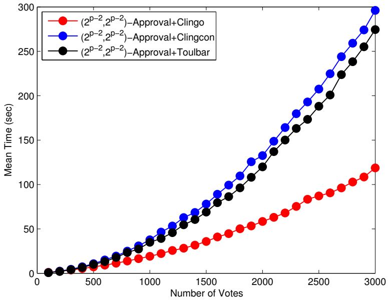
(a) Fixed #attributes (10)



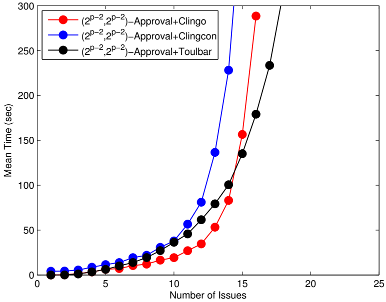
(b) Fixed #votes (1000)

Figure : Solving the winner problem

Varying p and n: $(2^{p-2}, 2^{p-2})$-approval [29]



(a) Fixed #attributes (10)

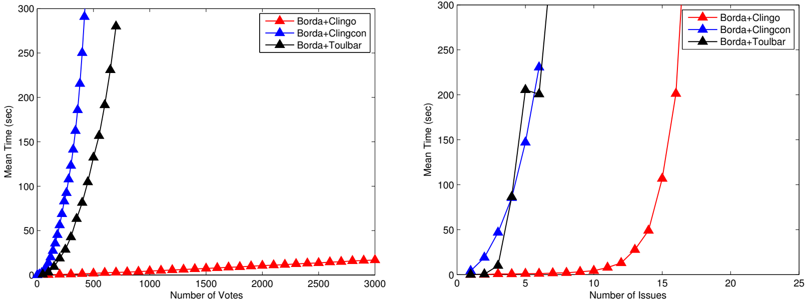


(b) Fixed #votes (1000)

Figure : Solving the winner problem

[29] scoring vector: $(2, \ldots, 2, 1, \ldots, 1, 0, \ldots, 0)$ with the numbers of 2's and 1's equal to 2^{p-2}

Varying p and n: Borda



(a) Fixed #attributes (10)   (b) Fixed #votes (1000)

Figure : Solving the winner problem

Outline

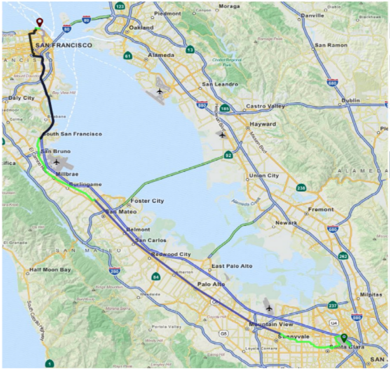
1. The languages of P-trees, PLP-trees, and LP-trees
2. Learning preference models in case of PLP-trees
3. Reasoning with preferences:
   - Preference optimization in case of P-tree
   - Computing winners and "strong" outcomes when votes are LP-trees
   - Application in trip planning
4. Future research directions

Personalization in Trip Planning

1. Important to incorporate user constraints and preferences into trip planning systems.
2. Collaboration with experts (in AI, planning, optimization, multi-agent systems) at PARC.
3. Developed a hipergraph-based trip planner that accommodates constraints specified as *linear temporal logic* and preferences expressed as *preferential cost function* to compute optimal routes using A*[30].
4. Available later for trip planning in the Bay Area, LA, and Denver.

[30] Xudong Liu et al. "On Personalizability and Extensibility of Multi-Modal Trip Planning". In: PARC Symposium. 2015

1. From SJC, to Pier 39, Monday, 9am.
2. Constraints: never drive a car, and bike for 1 to 2 hours.
3. Preferences: bike $=$ public $(0.25) >$ wait$(2) >$ walk(3), and 30\$/hr.



From:
San Jose Airport, San Jose, CA
To:
Pier 39, San Francisco, CA
Get directions

Route Summary
Start point: Terminal Drive
End point:
Driving time: 58 minutes
Distance: 48.6 miles

Segment	Mode	Time	Cost
1	bike	1 h 0 min 46 sec	\$0
2	transit	42 min	\$5.25
3	bike	56 min	\$0
4	uberx	15 min	\$23.92
5	bike	2 min	\$0
	Total:	2 h 57 min	\$29.17

Outline

1. The languages of P-trees, PLP-trees, and LP-trees
2. Learning preference models in case of PLP-trees
3. Reasoning with preferences:
   - Computing winners and "strong" outcomes when votes are LP-trees
   - Application in trip planning
4. Future research directions

Data-Driven Preference Engineering

1. Recommender Systems[31]:
    1. Collaborative
    2. Content-based
    3. Hybrid
2. Machine Learning:
    1. Supervised learning (e.g., decision trees, random forests)
    2. Label ranking[32]
3. Preference Elicitation (Human-in-the-Loop):
    1. Context-based
4. Preference Learning:
    1. Conditional Preference Networks, Preference Trees
    2. Stochastic Models (e.g., Choquet integral[33], TOPSIS-like models[34])

---

[31] Gediminas Adomavicius and Alexander Tuzhilin. "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions". In: Knowledge and Data Engineering, IEEE Transactions on (2005)

[32] Eyke Hüllermeier et al. "Label ranking by learning pairwise preferences". In: Artificial Intelligence (2008)

[33] Agnes Leroy, Vincent Mousseau, and Marc Pirlot. "Learning the parameters of a multiple criteria sorting method". In: Algorithmic decision theory. 2011

[34] Manish Agarwal, Ali Fallah Tehrani, and Eyke Hüllermeier. "Preference-based Learning of Ideal Solutions in TOPSIS-like Decision Models". In: Journal of Multi-Criteria Decision Analysis (2014)

Preference Reasoning and Applications

1. Social Choice and Welfare[35]:
   1. Voting
   2. Fair division
   3. Strategyproof Social Choice
2. Automated Planning and Scheduling:
   1. Travel scheduling
   2. Manufacturing
   3. Traffic control
3. Computer Vision and Image Processing:
   1. Image retrieval
   2. Image and video understanding

[35] Kenneth J Arrow, Amartya Sen, and Kotaro Suzumura. Handbook of Social Choice and Welfare. Vol. 1 & 2. 2010

Publications

1. Xudong Liu. "Modeling, Learning and Reasoning with Qualitative Preferences". Algorithmic Decision Theory, 2015.
2. Xudong Liu and Miroslaw Truszczynski. "Reasoning with Preference Trees over Combinatorial Domains". Algorithmic Decision Theory, 2015.
3. Xudong Liu and Miroslaw Truszczynski. "Learning Partial Lexicographic Preference Trees over Combinatorial Domains". AAAI Conference on Artificial Intelligence, 2015.
4. Xudong Liu and Miroslaw Truszczynski. "Preference Trees: A Language for Representing and Reasoning about Qualitative Preferences". Multidisciplinary Workshop on Advances in Preference Handling, 2014.

Publications

5. Matthew Spradling, Judy Goldsmith, Xudong Liu, Chandrima Dadi, and Zhiyu Li. "Roles and Teams Hedonic Game". Algorithmic Decision Theory, 2013.
6. Xudong Liu and Miroslaw Truszczynski. "Aggregating Conditionally Lexicographic Preferences Using Answer Set Programming Solvers". Algorithmic Decision Theory, 2013.
7. Xudong Liu. "Aggregating Lexicographic Preference Trees Using Answer Set Programming: Extended Abstract". International Joint Conference on Artificial Intelligence Doctoral Consortium, 2013.
8. Xudong Liu and Miroslaw Truszczynski. "Learning Preference Trees and Forests". (In Preparation).

Publications

9. Xudong Liu and Miroslaw Truszczynski. "Approximating Conditional Preference Networks Using Lexicographic Preference Trees". (In Preparation).
10. Xudong Liu and Miroslaw Truszczynski. "Complexity of Manipulation in Elections Where Votes Are Lexicographic Preference Trees". (In Preparation).
11. Xudong Liu and Miroslaw Truszczynski. "Reasoning About Lexicographic Preferences Over Combinatorial Domains". (In Preparation).
12. Xudong Liu and Christian Fritz. "On Personalizability and Extensibility of Multi-Modal Trip Planning". (In Preparation).

Related Work

1. Quantitative:
   - Utility/Cost Functions
   - Possibilistic Logic[36]
   - Fuzzy Preference Relations[37]
   - Penalty Logic[38]
2. Qualitative:
   - Answer-Set Optimization Theories[39]
   - Ceteris Paribus Networks (e.g., CP-nets[40], TCP-nets[41], CI-nets[42])
   - Conditional Preference Theories[43]

---

[36] Didier Dubois, Jérôme Lang, and Henri Prade. "A Brief Overview of Possibilistic Logic". In: ECSQARU. 1991

[37] SA Orlovsky. "Decision-making with a fuzzy preference relation". In: Fuzzy sets and systems (1978)

[38] Gadi Pinkas. Propositional non-monotonic reasoning and inconsistency in symmetric neural networks. 1991

[39] Gerhard Brewka, Ilkka Niemelä, and Miroslaw Truszczynski. "Answer Set Optimization". In: IJCAI. 2003

[40] C. Boutilier et al. "CP-nets: A Tool for Representing and Reasoning with Conditional Ceteris Paribus Preference Statements". In: Journal of Artificial Intelligence Research (2004)

[41] Ronen I. Brafman and Carmel Domshlak. "Introducing Variable Importance Tradeoffs into CP-Nets". In: UAI. 2002

[42] Sylvain Bouveret, Ulle Endriss, and Jérôme Lang. "Conditional importance networks: A graphical language for representing ordinal, monotonic preferences over sets of goods". In: (2009)

[43] Nic Wilson. "Extending CP-Nets with Stronger Conditional Preference Statements". In: AAAI-04. 2004

Questions?

Thank you!