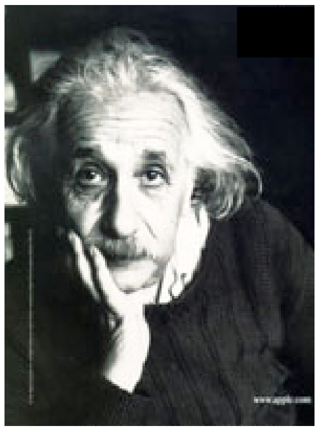
Graphical Modeling of Qualitative Preferences

Carmel Domshlak
(Cornell / Technion)

Intelligent Information
Systems Institute
CORNELL

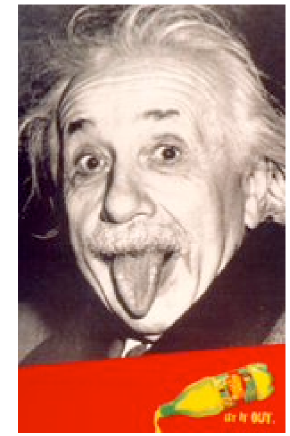
Overview

Use graphical models ...



30 min ...?!?!

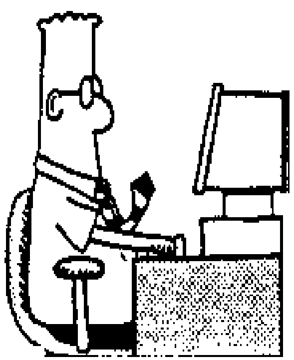
Drink graphical models!



- Motivating scenario for qualitative preferences.
- Graphical models in a nutshell - How and Why?
- Chronological perspective and bibliography.

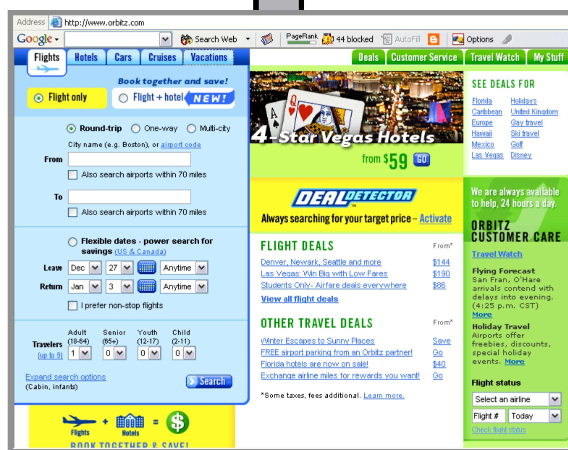
Motivating Scenario

With KLM I prefer to have a stop-over in Amsterdam ...



Preference Query

Query result

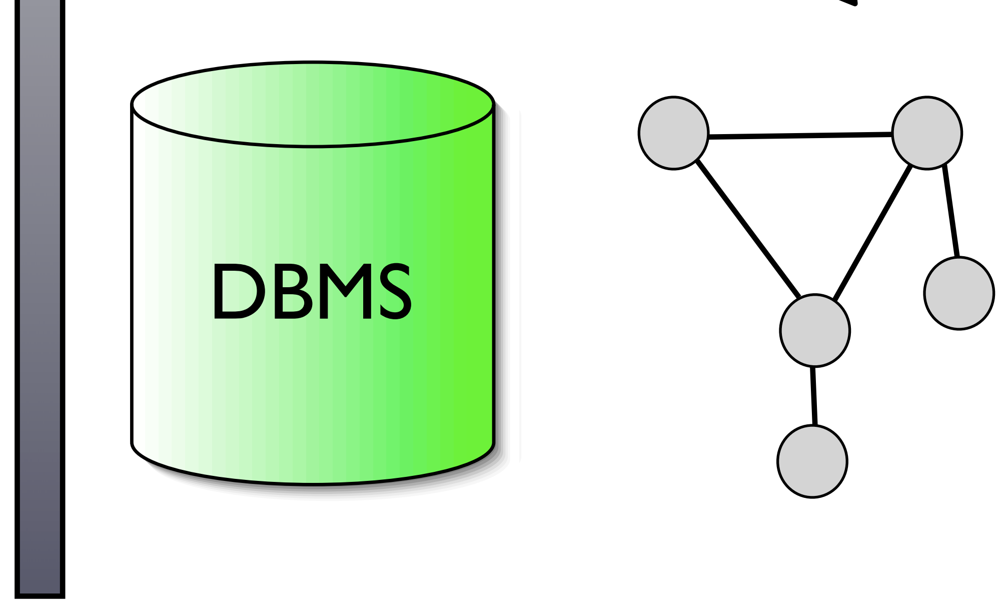


Query evaluation

Feasible alternatives

Motivating Scenario

- Natural, qualitative queries.
- Efficient consistency verification.
- Efficient query evaluation.



Characteristic Properties

1. Uncertainty is not an issue.
   - *Knowing the ordinal preferences suffices.*
2. Multi-attribute description of the alternatives.
   - *The DB tuples are described in terms of some n attributes.*

   Direct assessment of a preference ranking is typically infeasible as the size of the tuple space is exponential in n.
3. Lay users.
   - *Preference elicitation should be as effort-less as possible.*
4. Online Applications
   - *Query evaluation has to be computationally efficient.*

Query Language

Each query consists of a set of natural, qualitative preference statements interpreted as constraints on the ordinal preferences P of the user.

- (Conditional) preference

  (Among the flights leaving two days before the conference), I prefer to take an evening/night flight.

- (Conditional) relative importance

  (On British Airways, night flights), the absence of stop-overs is more important to me than getting a cheaper economy sit.

- ...

Query Language

Each query consists of a set of natural, qualitative preference statements interpreted as constraints on the ordinal preferences P of the user.

- Each statement provides us with a subset of P, and these subsets are composed into what we assume to be known about P.
- Semantic issues:
  - *How should we interpret each statement?*
  - *How should we compose the information provided by a set of statements?*

Computational Tasks

- Consistency Validation
  (... for an agreed notion of consistency).

- Optimization (i.e. generating one of the best tuples).

- Constrained Optimization
  (i.e. generating one of the best tuples).

- Filtering all the best tuples from a given list.

- Sorting a list of alternatives

Graphical Modeling - How?

1. Adopt a concrete semantics for statement interpretation.

2. Identify useful notions of purely qualitative, possibly conditional preferential independence.

3. Use preferential independence to define graphical models for preference representation.

4. Exploit the graphical core of the models to achieve computational efficiency.

Graphical Modeling

Qualitative statements of unconditional and conditional preference.

→ CP-nets

1. Adopt a concrete semantics for statement interpretation.
2. Identify useful notions of purely qualitative, possibly conditional preferential independence.
3. Use preferential independence to define graphical models for preference representation.
4. Exploit the graphical core of the models to achieve computational efficiency.

Interpretation Semantics

I prefer to fly British Airways rather than KLM.

Totalitarianism

Ignore the unmentioned attributes.

Any flight configuration with BA is preferred to *any* flight configuration with KLM.

Ceteris Paribus

Fix the unmentioned attributes.

A flight configuration with BA is preferred to a flight configuration with KLM, *provided* that these configurations are otherwise identical.

Interpretation Semantics

I prefer to fly British Airways rather than KLM.

Totalitarianism

Ignore the unmentioned attributes.

Any flight configuration with BA is preferred to *any* flight configuration with KLM.

Ceteris Paribus

Fix the unmentioned attributes.

A flight configuration with BA is preferred to a flight configuration with KLM, *provided* that these configurations are otherwise identical.

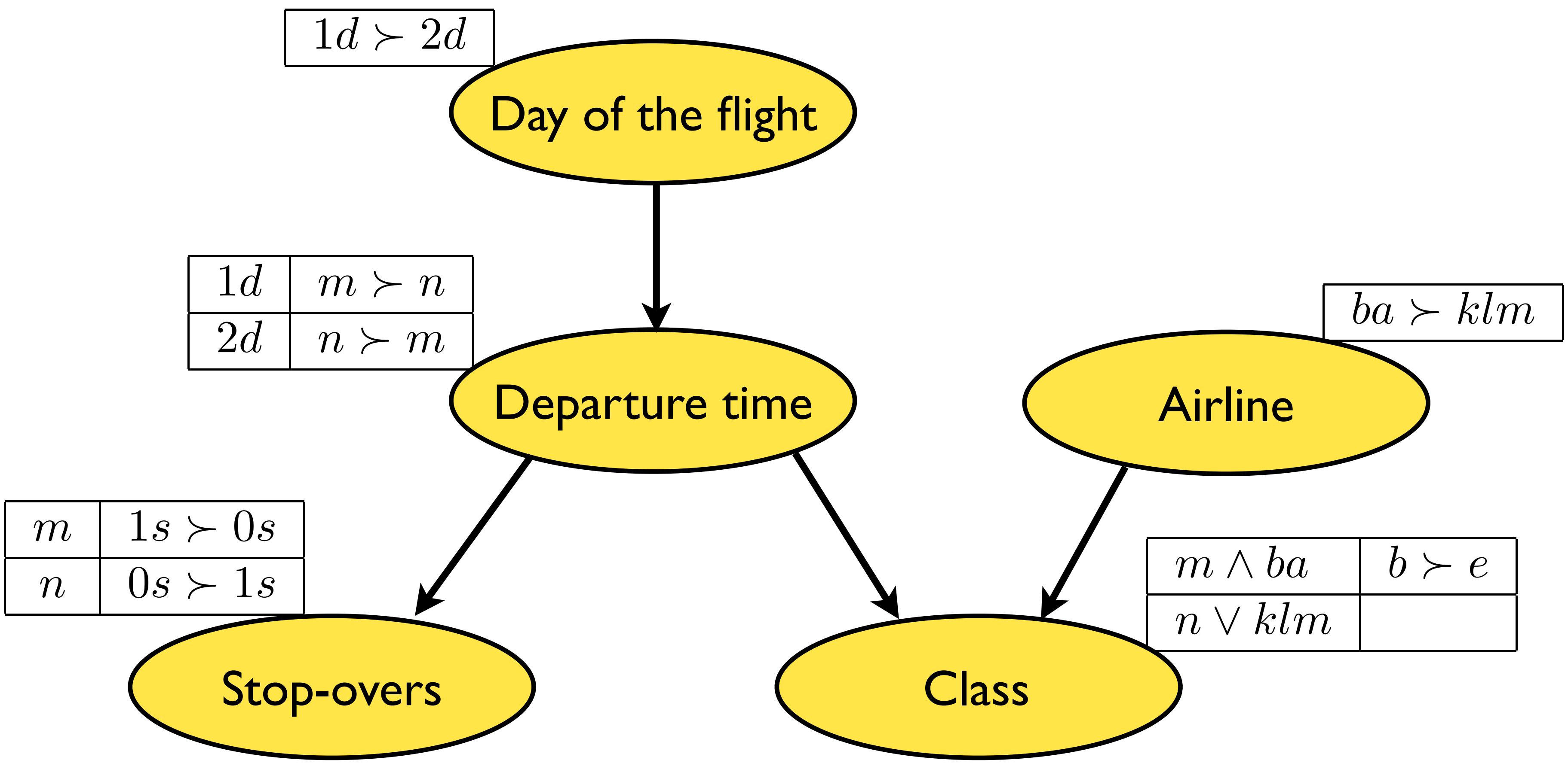
CP-nets

The model constitutes an annotated directed graph over the problem attributes $X_1, \ldots, X_n$.

- Each node represents a domain attribute X_i.
- The immediate predecessors of a node X_i in the graph are those variables that directly affect user's preference over the values of X_i.
- Each node is annotated with a conditional preference table (CPT), providing a partial ordering over the values of the node for each assignment to its immediate predecessors.

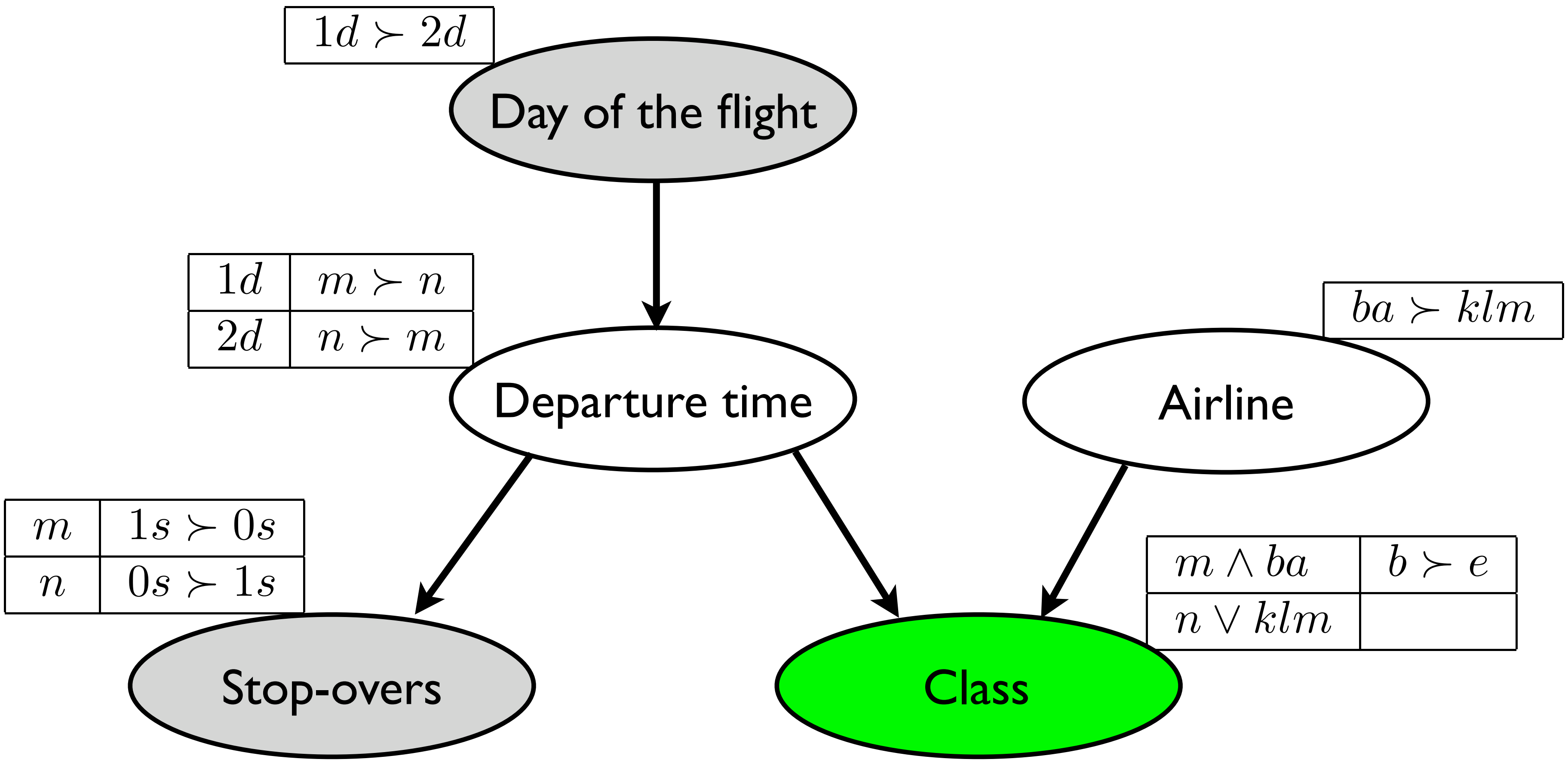
[BBGP-97][BBHP-uai99] ... [BBDHP-jair04]

CP-nets



$1d \succ 2d$
Day of the flight
$1d$ $m \succ n$
$2d$ $n \succ m$
Departure time
$ba \succ klm$
Airline
m $1s \succ 0s$
n $0s \succ 1s$
Stop-overs
$m \wedge ba$ $b \succ e$
$n \vee klm$
Class

Conditional Preferential Independence



Throughout most of this talk we assume a DAG structure.

DB-oriented Example

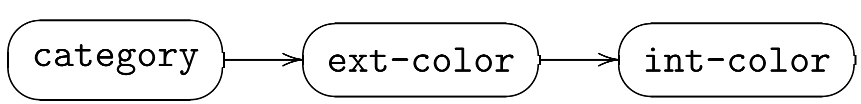
Query

s_1	*I prefer red minivans to white minivans.*
s_2	*I prefer white SUVs to red SUVs.*
s_3	*In white cars I prefer a dark interior.*
s_4	*In red cars I prefer a bright interior.*
s_5	*I prefer minivans to SUVs.*

Tuple space

	category	ext-color	int-color
t_1	minivan	red	bright
t_2	minivan	red	dark
t_3	minivan	white	bright
t_4	minivan	white	dark
t_5	SUV	red	bright
t_6	SUV	red	dark
t_7	SUV	white	bright
t_8	SUV	white	dark

CP-net

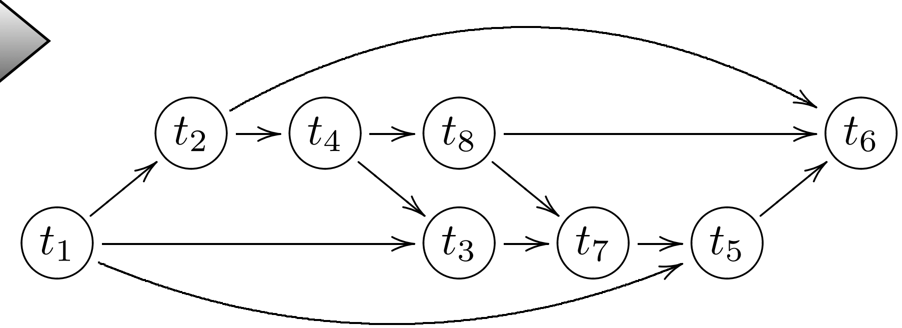


$C_{mv} \succ C_{suv}$

C_{mv}	$E_r \succ E_w$
C_{suv}	$E_w \succ E_r$

E_r	$I_b \succ I_d$
E_w	$I_d \succ I_b$

Preference ranking



Graphical Modeling - Why?

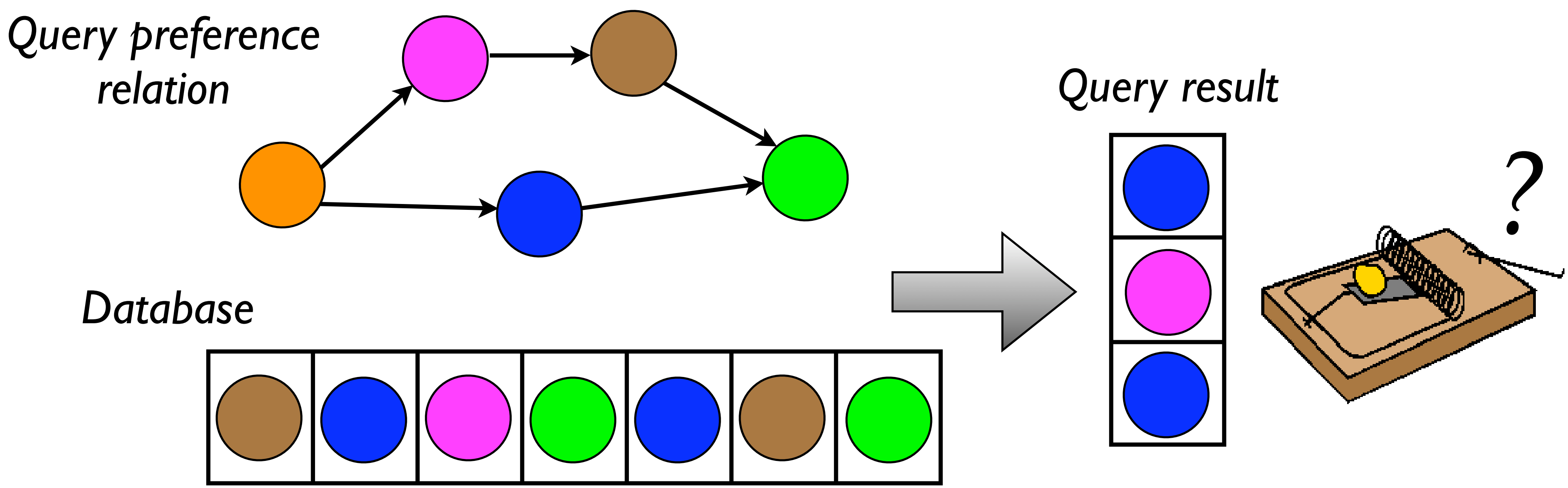
- Because graphical structure helps us in computational analysis of both consistency verification and inference.
  - Developing efficient algorithms for tractable cases.
  - Classifying hardness of the intractable cases, and developing informed heuristics for them.
-
- Because using graphical structure is critical for effective knowledge compilation.

Computational Tasks

- Consistency Validation.
- Optimization.

- Constrained Optimization.
- Filtering all the best tuples from a given list.
- Sorting a list of alternatives.

Filtering the best tuples
(Winnow, BMO, BEST)



Any implementation of filtering requires preferential comparison between pairs of alternatives:
Given a CP-net N and a pair of tuples α and β, determine whether

$$N \models \alpha \succ \beta$$

How hard can it be?

Dominance testing for binary-valued CP-nets

CP-net graph	Complexity	Remarks
Directed Tree	$O(n^2)$	*Lower bound*
Singly Connected (Polytree)	$O(2^{2k} n^{2k+3})$	*k - maximal node indegree*

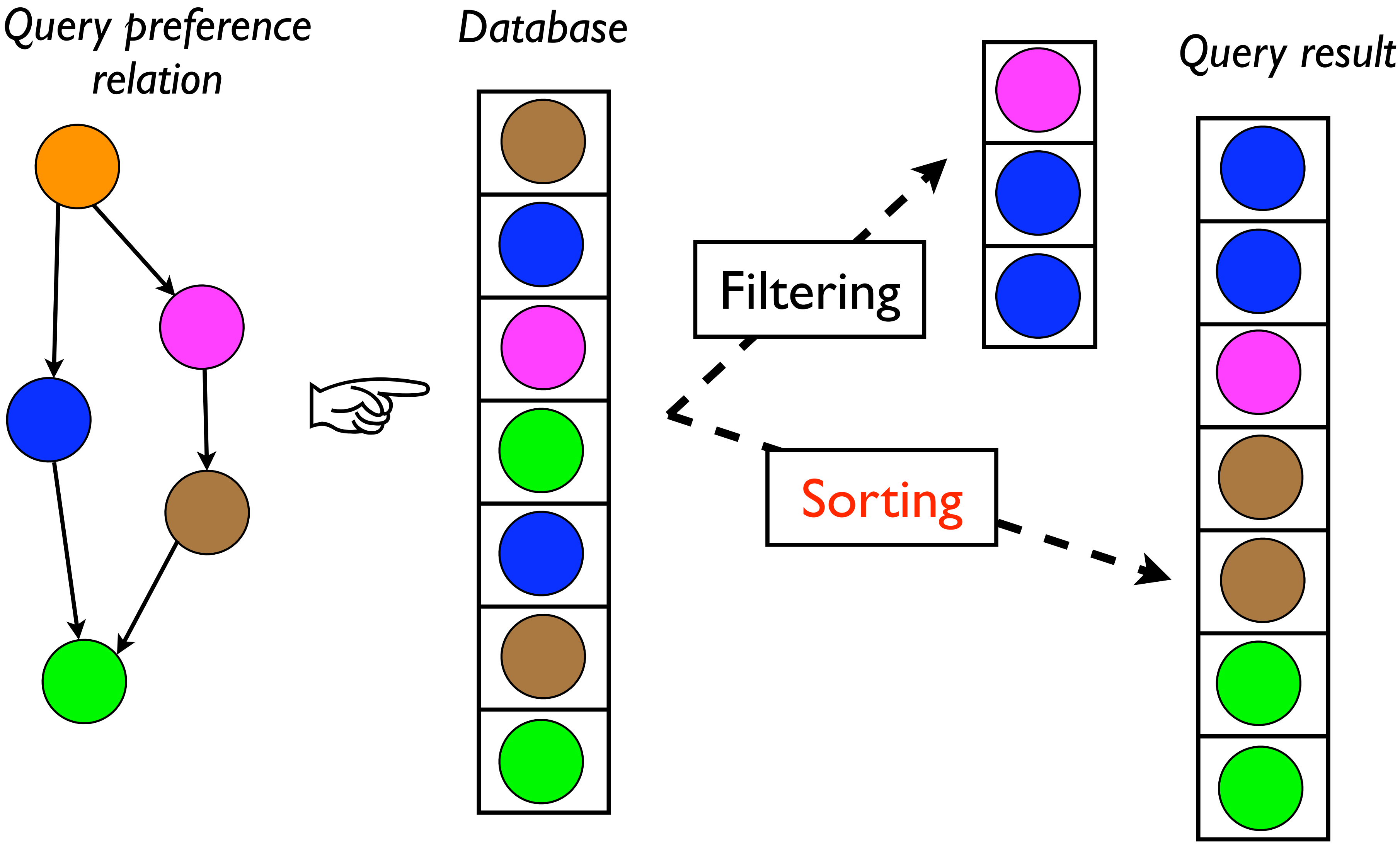
Dominance testing for binary-valued CP-nets

CP-net graph	Complexity	Remarks
Directed Tree	$O(n^2)$	*Lower bound*
Singly Connected (Polytree)	$O(2^{2k}n^{2k+3})$	*k - maximal node indegree*
Directed-path Singly Connected	NP-complete	*Reduction from 3SAT*

Dominance testing for binary-valued CP-nets

CP-net graph	Complexity	Remarks
Directed Tree	$O(n^2)$	*Lower bound*
Singly Connected (Polytree)	$O(2^{2^k} n^{2k+3})$	*k - maximal node indegree*
Directed-path Singly Connected	NP-complete	*Reduction from 3SAT*
Directed-path δ-connected	NP-complete	*Polynomial-size certificates*
General DAG	?	*Harder than NP?*

Alternatives for Query Evaluation?



Any computational difference?

Insights into preferential comparison

- Dominance Test

  Given a CP-net N and a pair of tuples α and β, determine whether

- $$N \models \alpha \succ \beta$$

  α **has to** be ordered before β

- Ordering Test

  Given a CP-net N and a pair of tuples α and β, determine whether

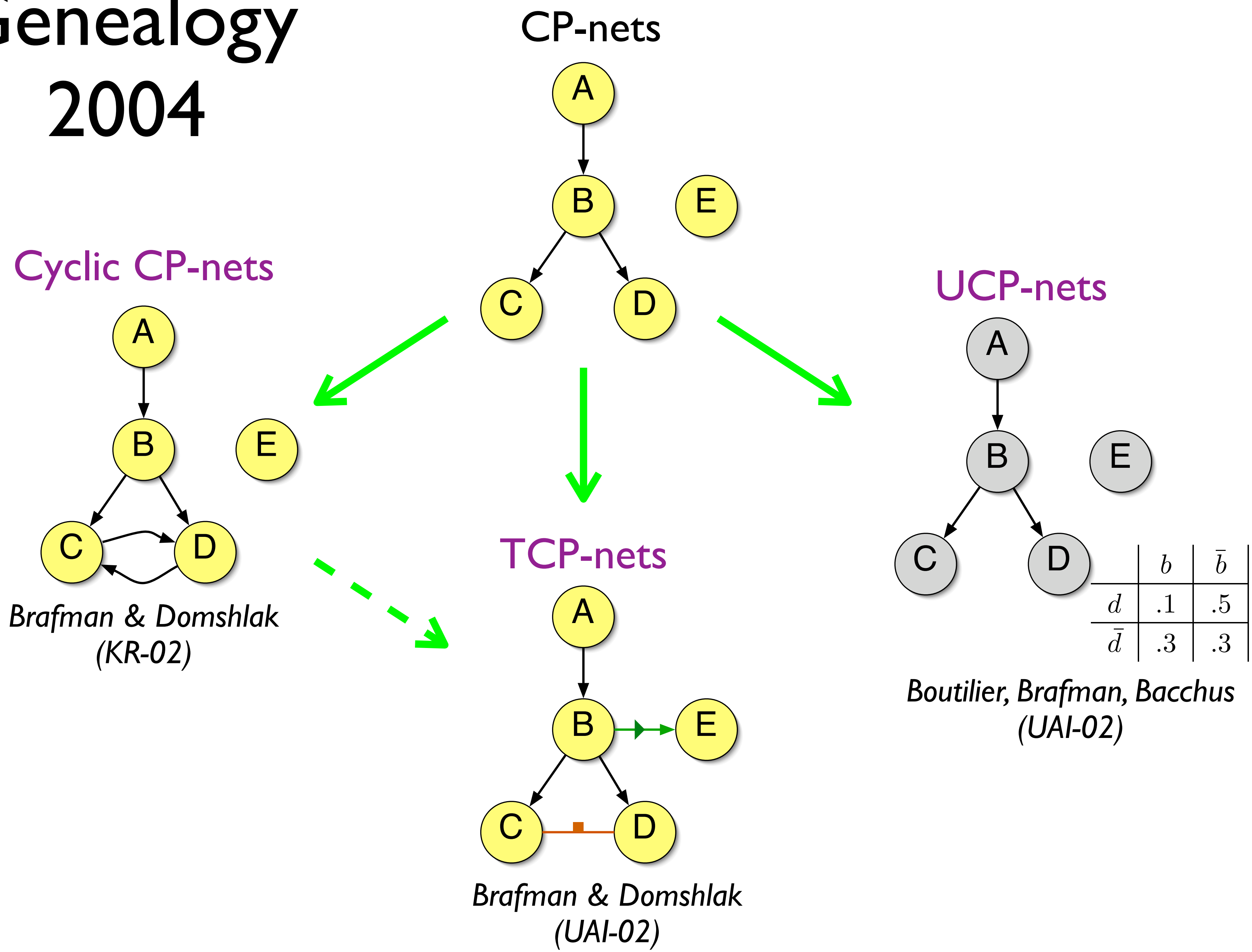
  $$N \not\models \beta \succ \alpha$$

  α **can** be ordered before β

Sorting vs. Filtering in a Nutshell

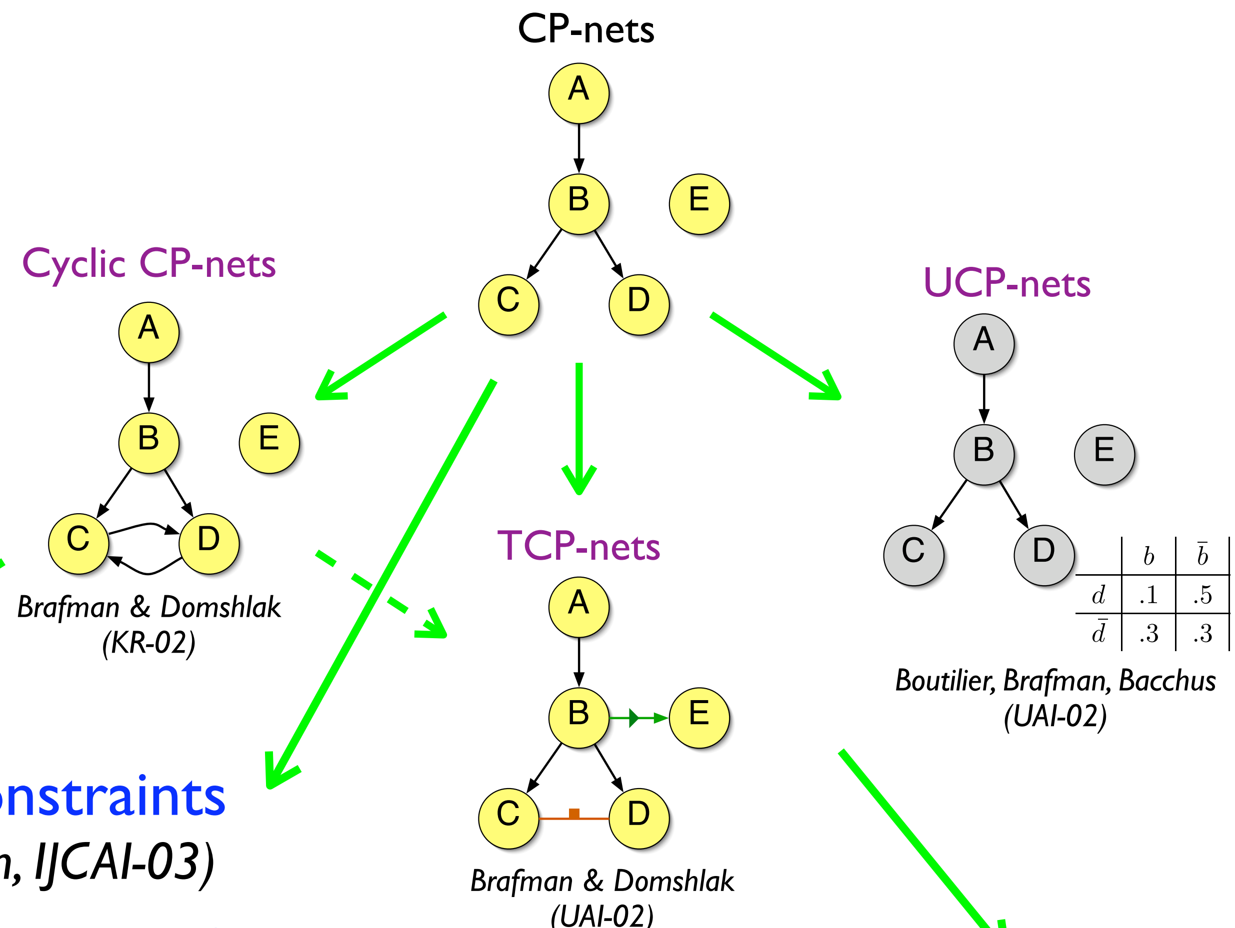
- There exists an ordering test procedure such that, for *any* acyclic CP-net and *any* pairs of tuples:
  - Time complexity of ordering is $O(n)$, and
  - The relation induced by this operator on the tuple space is a *total* order consistent with the preference query.
- Given a preference query inducing an acyclic CP-net, a dataset of D tuples can be properly sorted in time $O(nD \log D)$. (compare to filtering in time $O(\Phi \cdot D^2)$).
- Our conclusion (so far):
  - There is a principle difference between filtering and sorting, and it can have significant computational impacts.
  - Worth further investigation and enhancement.

Genealogy 2004



	b	$\bar{b}$
d	.1	.5
$\bar{d}$	.3	.3

Genealogy 2004



Connection with soft constraints
(Domshlak, Rossi, Venable, Walsh, IJCAI-03)

Weaker semantics for cyclic preferences
(Brafman, Dimopoulos, IJCAI-03)

Extending CP-nets to mixture of totalitarianism and *ceteris paribus*
(Wilson, AAAI-04, ECAI-04)

Multi-agent preferences (mCP-nets)
(Rossi, Venable, Walsh, AAAI-04, ECAI-04)

...

Value-function compilation
(Brafman, Domshlak, Kogan, UAI-04)

Bibliography to start with ...

Boutilier, Brafman, Domshlak, Hoos, Poole, "CP-nets: A tool for representing and reasoning with conditional ceteris paribus statements", *J. of AI Research*, 21, 2004.

Domshlak, "Modeling and Reasoning about Preferences with CP-nets", Ph.D. Dissertation, Ben-Gurion University, 2002.

Brafman, Domshlak, "Introducing Variable Importance Tradeoffs into CP-nets", *UAI*, 2002.

Boutilier, Brafman, Hoos, Poole, "Reasoning with conditional *ceteris paribus* preference statements", *UAI*, 1999.

...