

Soft constraint processing

Thomas Schiex

INRA – Toulouse
France

Pedro Mesequer

CSIC – IIIA –
Barcelona
Spain

Special thanks to:

Javier Larrosa

UPC – Barcelona
Spain

Overview

□ Introduction and definitions

- Why soft constraints and dedicated algorithms?
- Generic and specific models

□ Solving soft constraint problems

- By search (systematic and local)
- By complete inference... and search
- By incomplete inference... and search

□ Existing implementations

Why soft constraints?

- CSP framework: for *decision* problems
- Many problems are *optimization* problems

- Economics (combinatorial auctions)
 - Given a set G of goods and a set B of bids...
 - Bid (B_i, V_i) , B_i requested goods, V_i value
 - ... find the best subset of compatible bids

 - Best = maximize revenue (sum)

Why soft constraints?

- Satellite scheduling
 - Spot 5 is an earth observation satellite
 - It has 3 on-board cameras
 - Given a set of requested pictures (of different importance)...
 - Resources, data-bus bandwidth, setup-times, orbiting
 - ...select a subset of compatible pictures with max. importance (sum)

Why soft constraints

□ Multiple sequence alignment (DNA, AA)

■ Given k homologous sequences...

□ AATAATGTTATTGGTGGATCGATGA

□ ATGTTGTTTCGCGAAGGATCGATAA

■ ... find the best alignment (sum)

□ AATAATGTTATTGGTG---GATCGATGATTA

□ ----ATGTTGTTTCGCGAAGGATCGATAA---

Why soft constraints?

- Probabilistic inference (bayesian nets)
 - Given a probability distribution defined by a DAG of conditional probability tables
 - And some evidence
 - ...find the *most probable* explanation for the evidence (product)

Why soft constraints?

- Ressource allocation (frequency assignment)
 - Given a telecommunication network
 - ...find the best frequency for each communication link

 - Best can be:
 - Minimize the maximum frequency (max)
 - Minimize the global interference (sum)

Why soft constraints

- Even in decision problems, the user may have *preferences* among solutions.
- It happens in most real problems.

Experiment: give users a few solutions and they will find reasons to prefer some of them.

Overview

- **Introduction and definitions**
 - Why soft constraints and dedicated algorithms?
 - Generic and specific models
- **Solving soft constraint problems**
 - By search (systematic and local)
 - By complete inference... and search
 - By incomplete inference... and search
- **Existing implementations**

Soft constraints

- Two new difficulties:
 - How to express preferences in the model?
 - How to solve the resulting optimization problem?

Solving soft constraints as a CSP

- Using a global criteria constraint
 - New constraint S_k on all X
 - $S_k(X)$ = criteria value must be more than k
 - Number of variables having value *blue*

□ Apply:

```
 $k := n$   
Repeat  
  solve the original problem  $\cup S_k(X)$   
   $k := k-1$ ;  
Until solution
```

n-ary constraint
Inefficient!!



Combined local preferences

- Constraints are local *cost functions*
- *Costs* combine with a dedicated *operator* ⊕
 - *max*: priorities Fuzzy/min-max CSP
 - +: additive costs Weighted CSP
 - *: factorized probabilities... Probabilistic CSP, BN
- Goal: find an assignment with the optimal combined cost

Soft constraint network

□ (X, D, C)

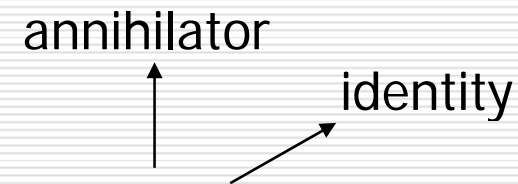
■ $X = \{x_1, \dots, x_n\}$ variables

■ $D = \{D_1, \dots, D_n\}$ finite domains

■ $C = \{c_1, \dots, c_e\}$ cost functions

□ $\text{var}(c_i)$ scope

□ $c_i(t) \rightarrow E$ (ordered cost domain, T, \perp)

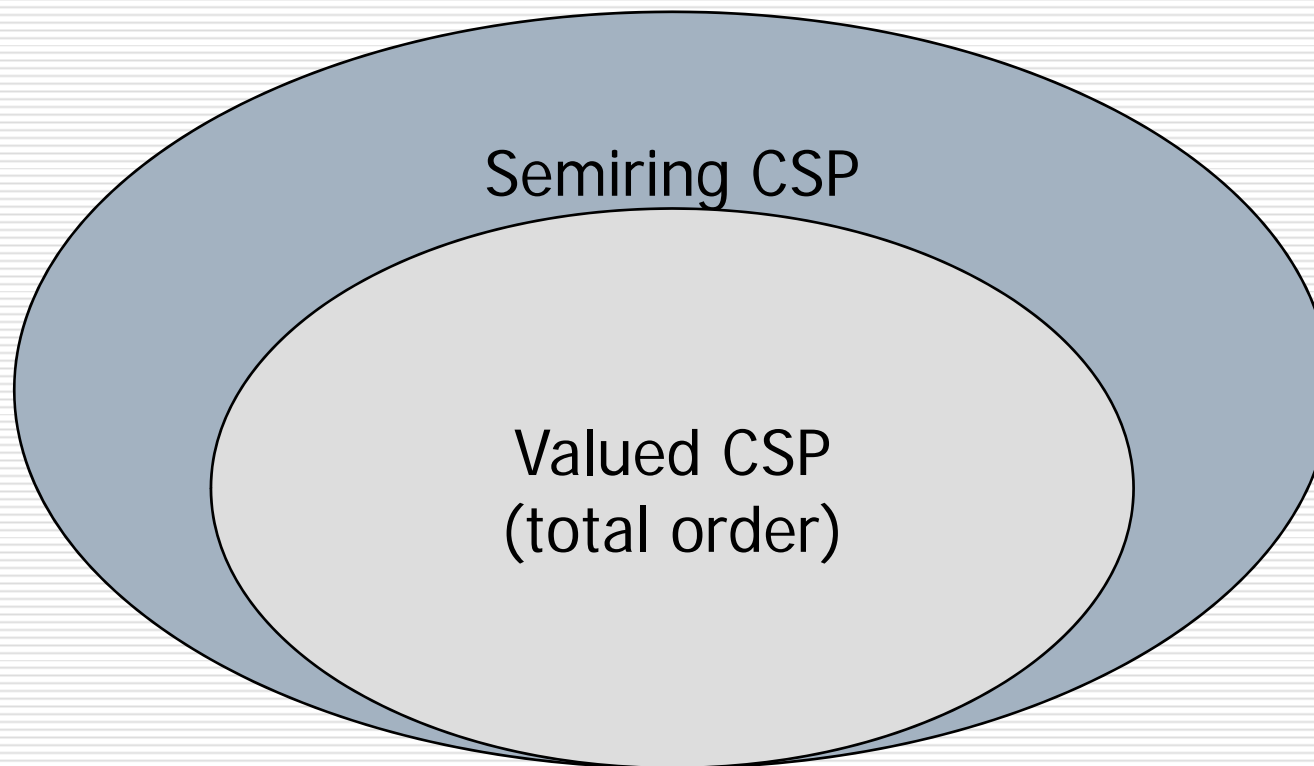


□ **Obj. Function:** $F(X) = \bigoplus c_i(X)$ → {
• commutative
• associative
• monotonic

□ **Solution:** $F(t) \neq T$

□ **Soft CN:** find minimal cost solution

Valued CSP and Semiring CSP

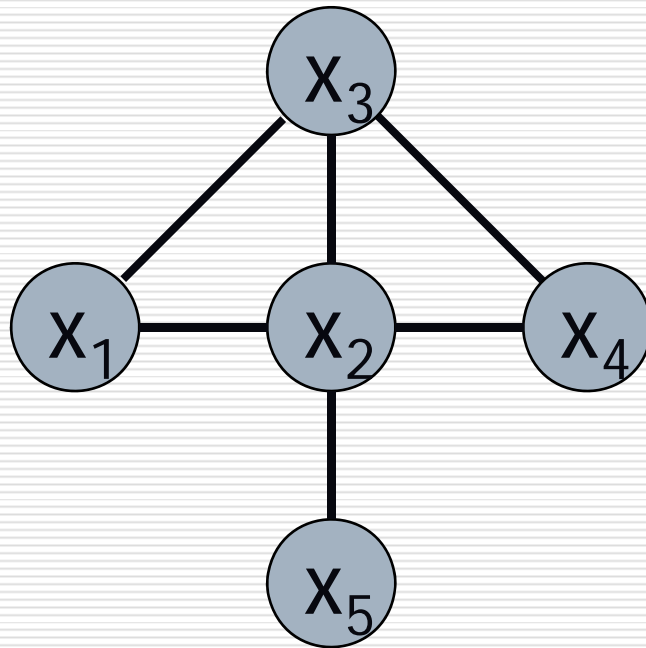


Specific frameworks

- $E = \{t, f\}$ $\oplus = \text{and}$ classical CSP
- $E = [0, 1]$ $\oplus = \text{max}$ \approx fuzzy CSP
- $E = \mathbb{N} \cup \{\infty\}$ $\oplus = +$ weighted CSP
- $E = [0, 1]$ $\oplus = *$ bayesian net

Lexicographic CSP, probabilistic CSP...

Weighted CSP example ($\oplus = +$)



For each vertex

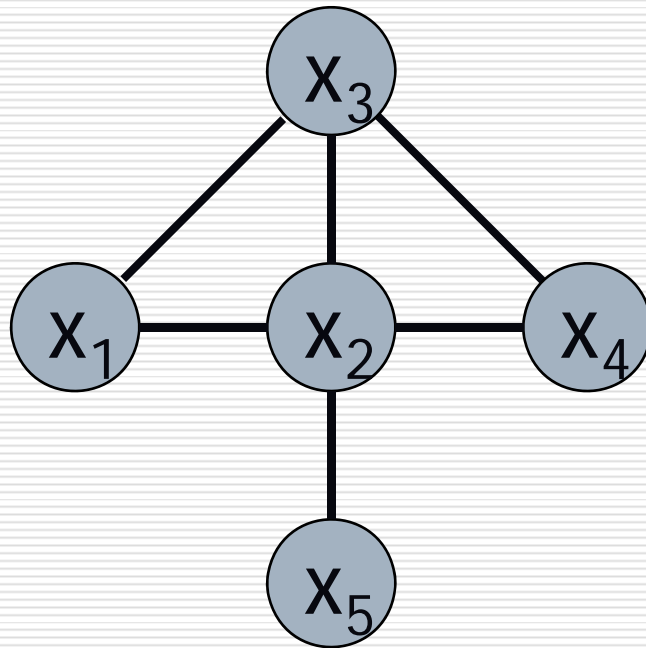
x_i	$c(x_i)$
b	0
g	1
r	1

For each edge:

x_i	x_j	$c(x_i, x_j)$
b	b	T
b	g	0
b	r	0
g	b	0
g	g	T
g	r	0
r	b	0
r	g	0
r	r	T

$F(X)$: number of non blue vertices

Fuzzy constraint network ($\oplus = \max$)



For each vertex

x_i	$c(x_i)$
b	0.2
g	0.4
r	0.6
y	0.8

Connected vertices must
have different colors

$F(X)$: highest color used ($b < g < r < y$)

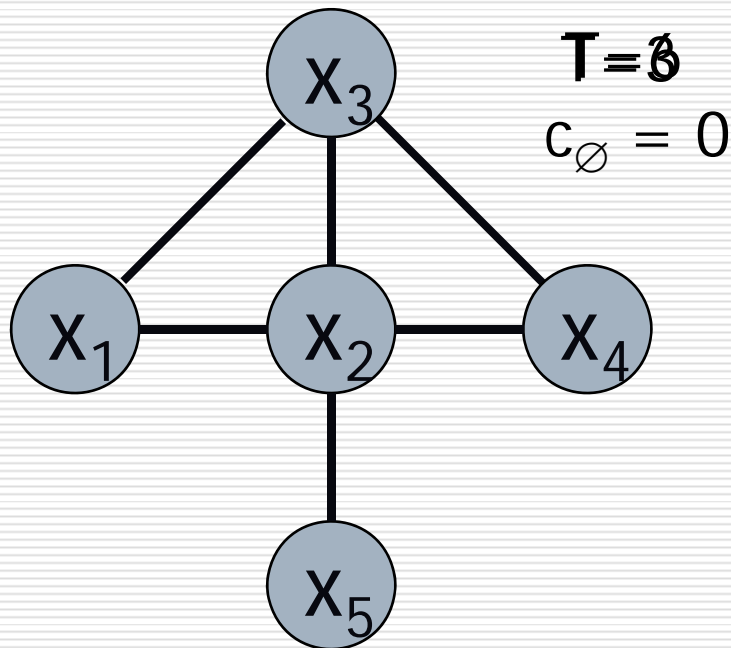
Some insight in unusual items

- T = maximum violation.
 - Can be set to a bounded max. violation k
 - Solution: $F(t) < k = Top$

- Empty scope soft constraint c_{\emptyset} (a constant)
 - Gives an obvious lower bound on the optimum
 - If you do not like it: $c_{\emptyset} = \perp$

Additional expression power

Weighted CSP example ($\oplus = +$)



For each vertex

x_i	$c(x_i)$
b	0
g	1
r	1

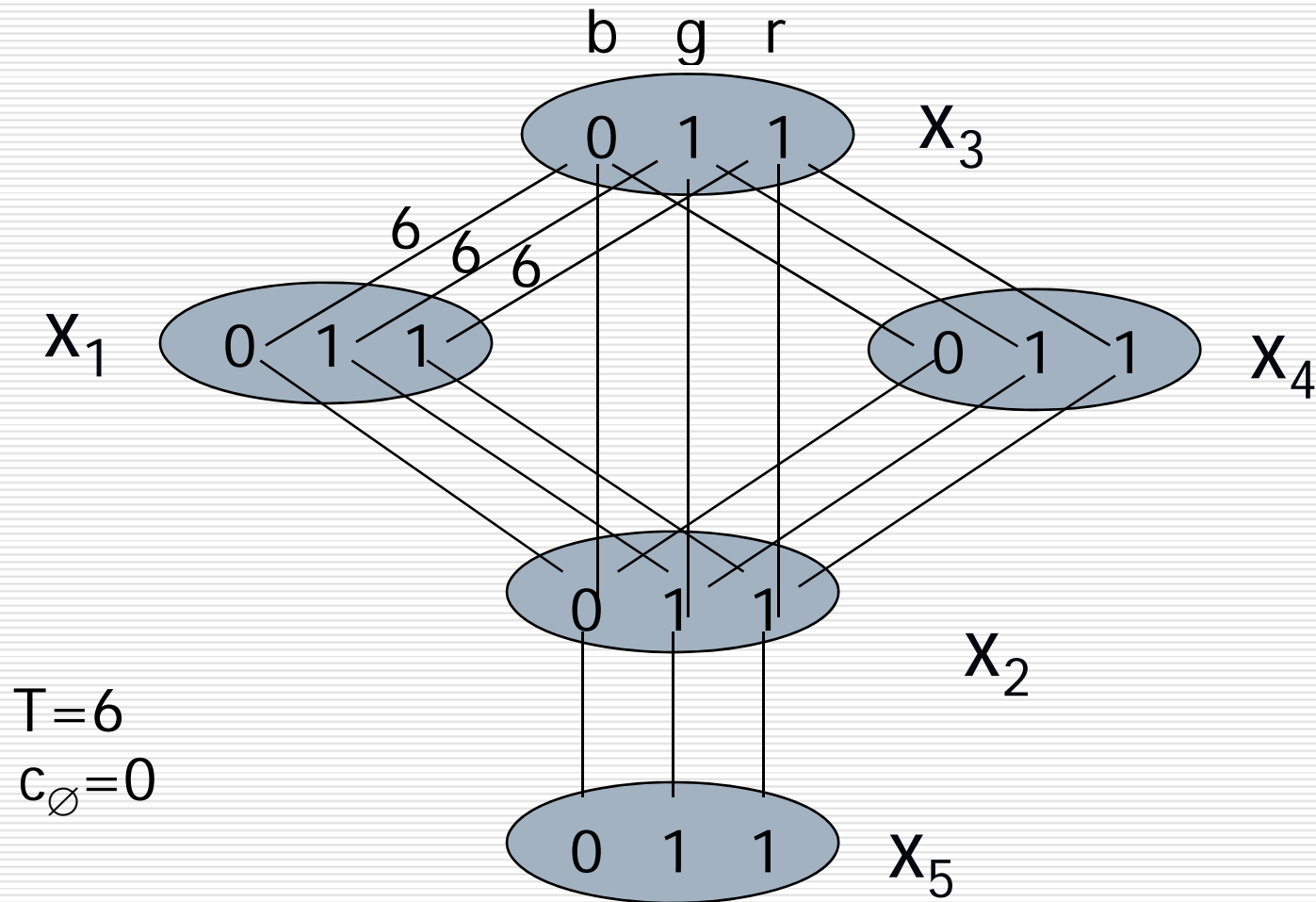
x_i	x_j	$c(x_i, x_j)$
b	b	T
b	g	0
b	r	0
g	b	0
g	g	T
g	r	0
r	b	0
r	g	0
r	r	T

For each edge:

$F(X)$: number of non blue vertices

Optimal coloration with less than 3 non-blue

Microstructural graph



Basic operations on cost functions

- I. Assignment (conditioning)
- II. Combination (join)
- III. Projection (elimination)

Assignment (conditioning)

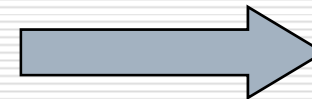
x_i	x_j	$c(x_i, x_j)$
b	b	T=6
b	g	0
b	r	0
g	b	0
g	g	T=6
g	r	0
r	b	0
r	g	0
r	r	T=6

Assign(c, x_i, b)



$f(x_j)$
0

Assign(f, x_j, g)



h_{\emptyset}

Combination (join with \oplus)

x_i	x_j	$c(x_i, x_j)$
b	b	6
b	g	0
g	b	0
g	g	6

\oplus

x_j	x_k	$c(x_j, x_k)$
b	b	6
b	g	0
g	b	0
g	g	6

x_i	x_j	x_k	$f(x_i, x_j, x_k)$
b	b	b	12
b	b	g	6
b	g	b	0
b	g	g	6
g	b	b	6
g	b	g	0
g	g	b	6
g	g	g	12

$= 0 \oplus 6$

Projection (elimination)

