Modeling Stable Matching Problems with Answer Set Programming

Sofie De Clercq[a]

Steven Schockaert[b], Martine De Cock[a], Ann Nowé[c]

[a] Dept. of Applied. Mathematics, Computer Science and Statistics, Ghent University, Belgium
[b] School of Computer Science and Informatics, Cardiff University, UK
[c] Computational Modeling Lab, Vrije Universiteit Brussel, Belgium

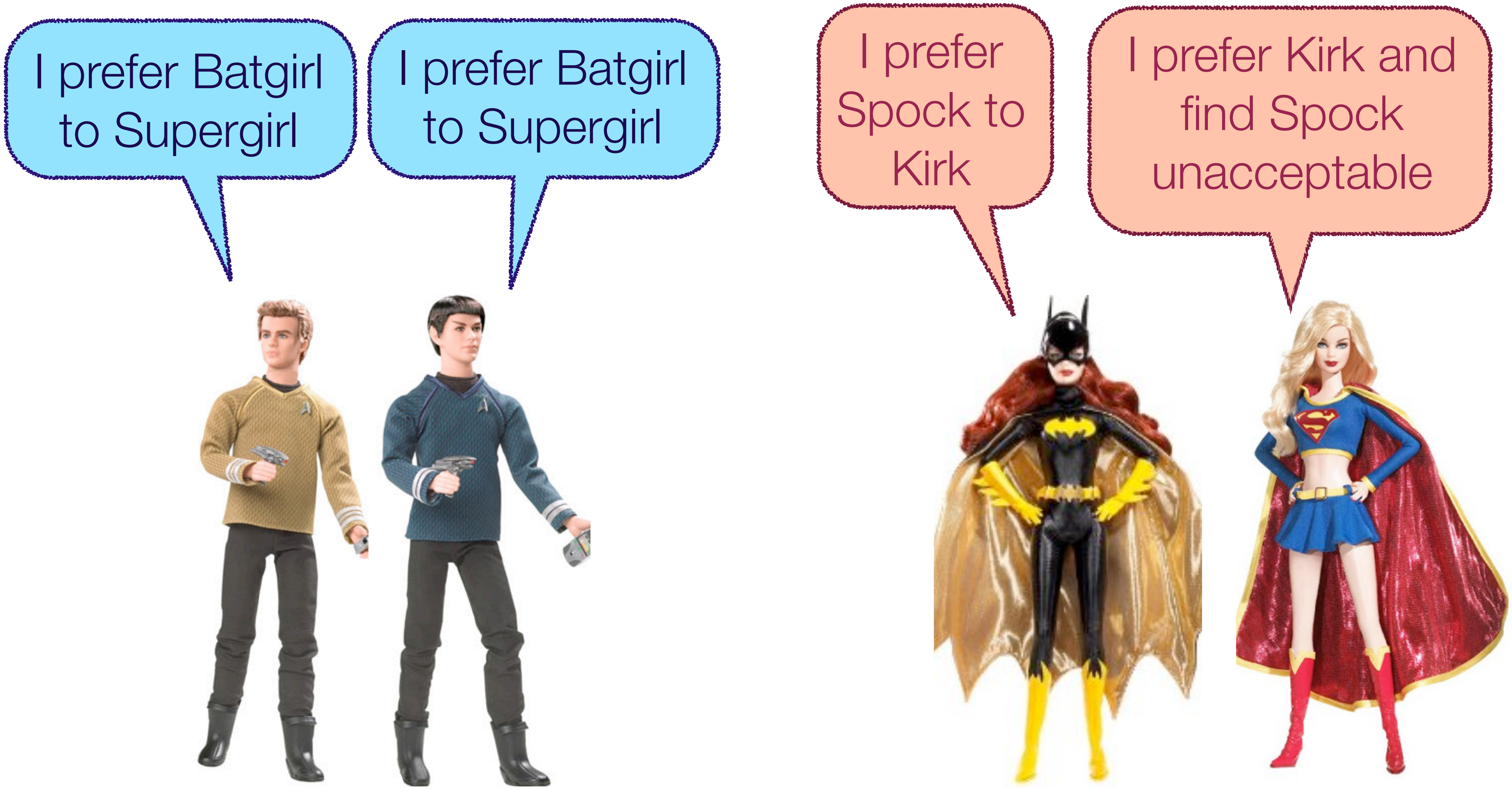
RuleML 2013, Seattle, Saturday, July 13th, 2013

UNIVERSITEIT GENT

CARDIFF UNIVERSITY PRIFYSGOL CAERDYDD

FWO VLAANDEREN
Fonds Wetenschappelijk Onderzoek
Research Foundation – Flanders

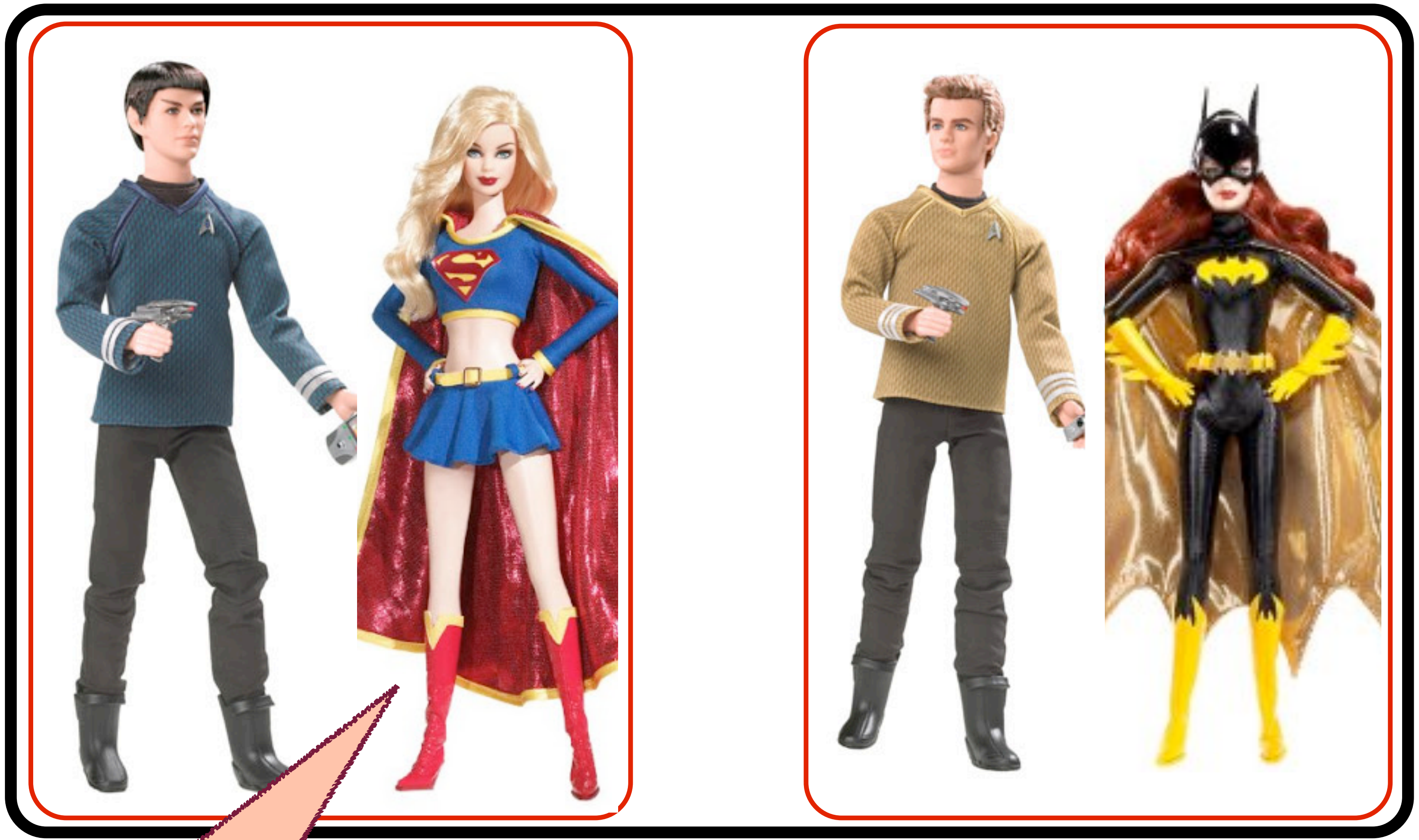
The Stable Marriage Problem (SMP)



Kirk Spock Batgirl Supergirl

Stability

$$\{(Spock, Supergirl), (Kirk, Batgirl)\}$$

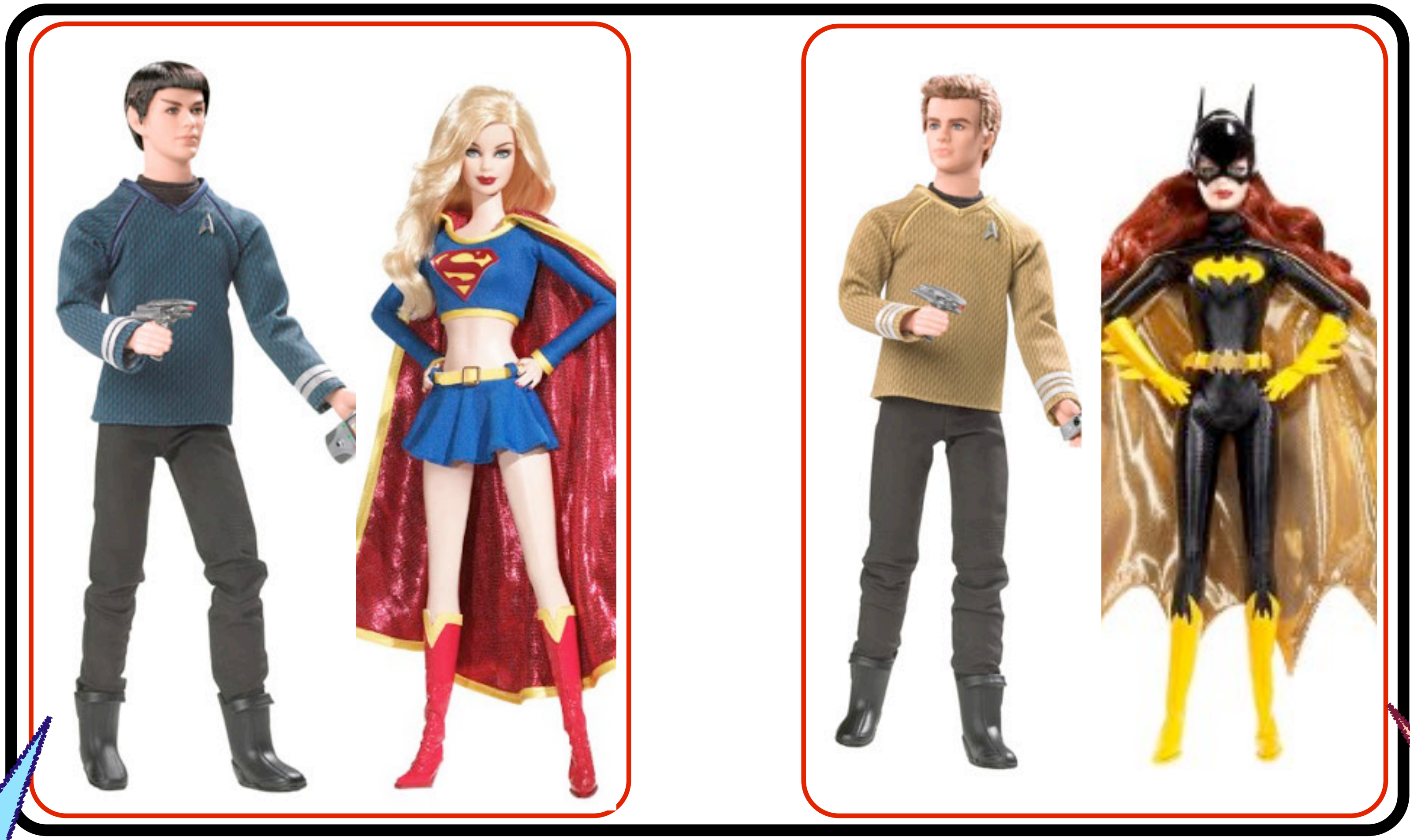


I found Spock unacceptable but got paired to him after all!

Supergirl blocks this set of marriages

Stability

$$\{(Spock, Supergirl), (Kirk, Batgirl)\}$$



I strictly prefer Batgirl to this woman

Batgirl and Spock block the set of marriages

I strictly prefer Spock to this guy

Stable set of marriages

↓

- no blocking individuals
- no blocking pairs

But... the notion of stability is too weak

to **distinguish good sets of marriages**

from **great sets of marriages**

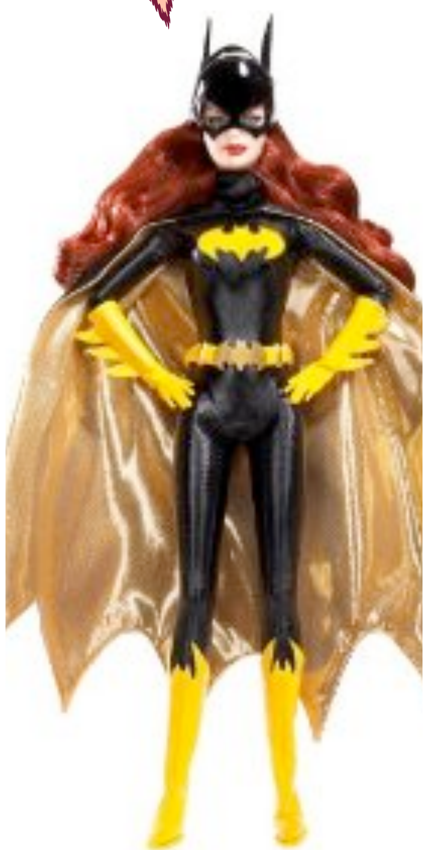
Optimality

I prefer Spock to Kirk

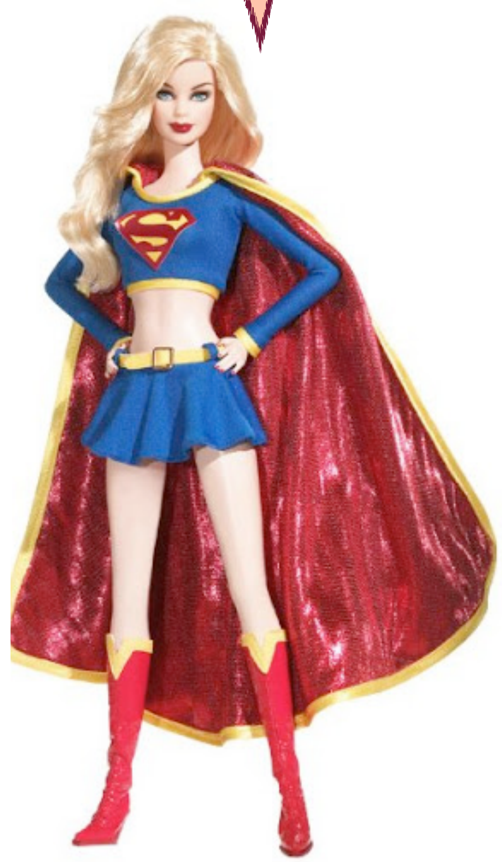
I am indifferent between Spock and Kirk

I prefer Batgirl to Supergirl

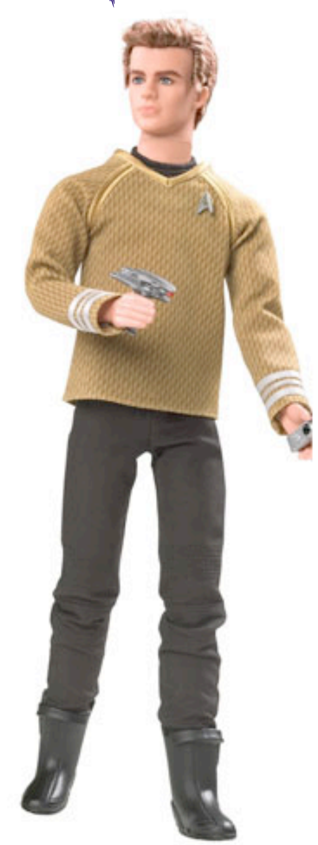
I prefer Supergirl to Batgirl



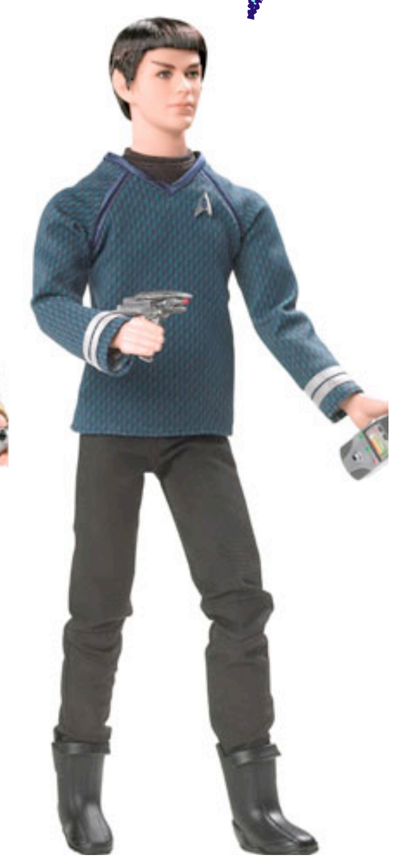
Batgirl



Supergirl

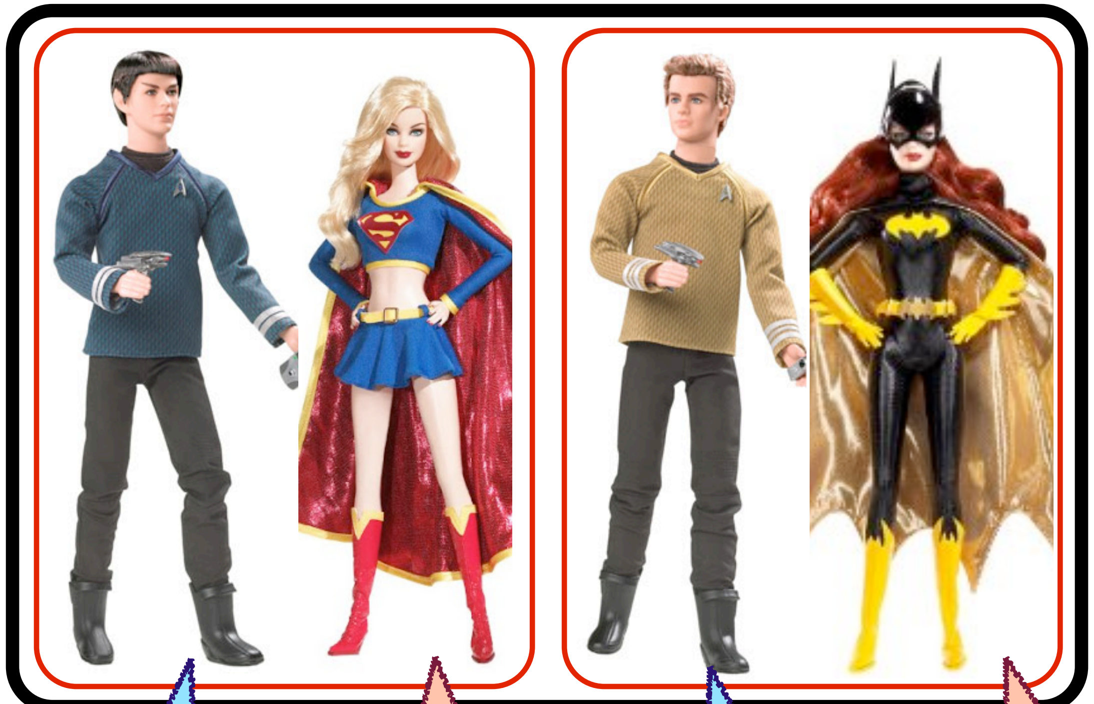


Kirk



Spock

Optimality



She is my first choice!

He is my first choice!

She is my first choice!

He is my second choice.

$$1 + 1 + 1 + 2 = 5$$

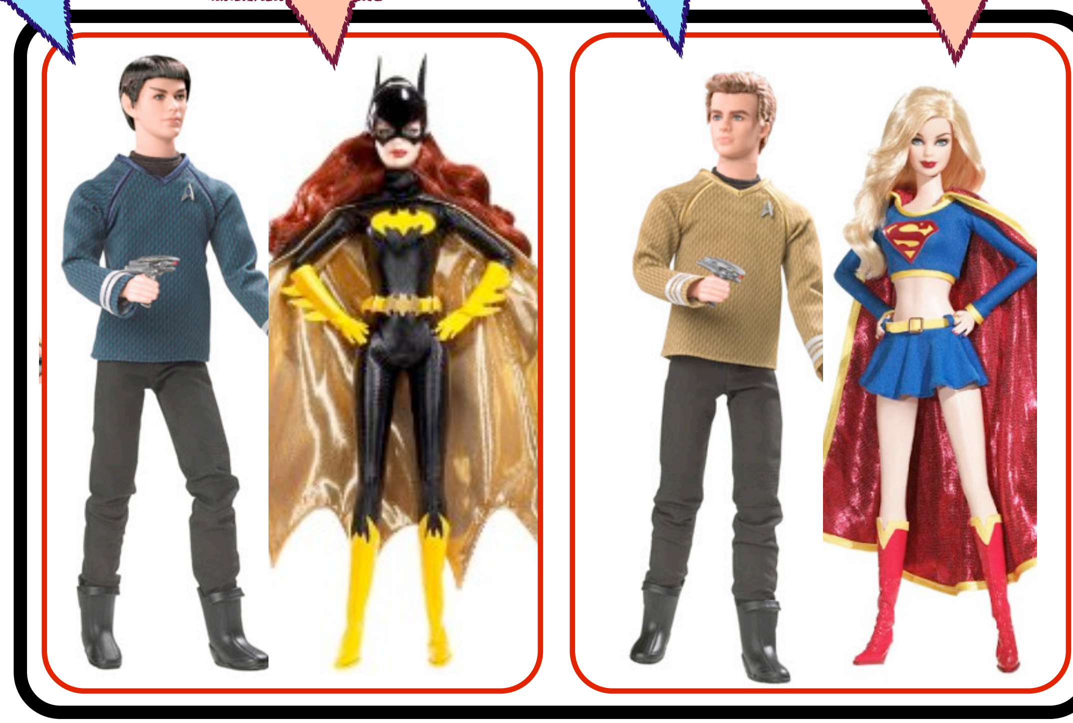
$$2 + 1 + 2 + 1 = 6$$

She is my second choice.

He is my first choice!

She is my second choice.

He is my first choice!



$5 < 6 \longrightarrow$ The first stable set is optimal based on egalitarity

The SMP in practice

Countless variants on the SMP have been investigated, e.g.:

- kidney-exchange,
- hospital-resident problem.

2012: Roth & Shapley win the Nobel Prize for Economics for their theory of stable allocations and the practice of market design.

Answer Set Programming

Answer Set Programming (ASP) uses **logical rules** to **describe** a **problem** and off-the-shelve solvers to compute its solutions, i.e. answer sets.

negation-as-failure (naf)

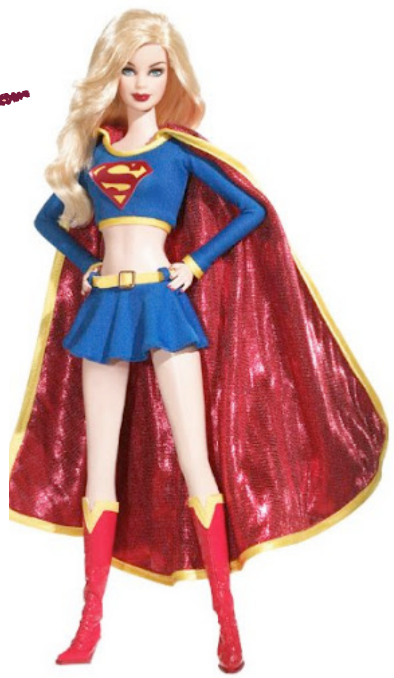
$$\underbrace{light\_red \vee light\_orange}_{\text{head}} \leftarrow \underbrace{cars\_stopped, not\ light\_broken}_{\text{body}}$$

Intuition: if we know that the cars have stopped and there is no evidence that the traffic light is broken, then the traffic light is either red or orange.

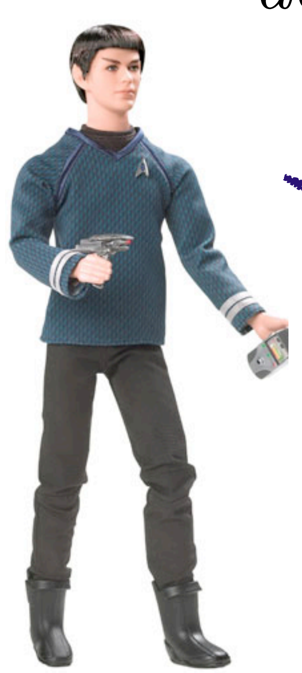
An answer set is a (kind of) minimal model of the program, satisfying every rule.

Obtaining stable sets of marriages with ASP

I am indifferent between Spock and Kirk



$$
\begin{aligned}
manpropose(Spock, Supergirl) &\leftarrow \\
manpropose(Spock, Batgirl) &\leftarrow not\ accept(Spock, Supergirl) \\
womanpropose(Spock, Supergirl) &\leftarrow not\ accept(Kirk, Supergirl) \\
womanpropose(Kirk, Supergirl) &\leftarrow not\ accept(Spock, Supergirl) \\
accept(Spock, Supergirl) &\leftarrow manpropose(Spock, Supergirl), \\
&\quad\ womanpropose(Spock, Supergirl)
\end{aligned}
$$



I prefer Supergirl to Batgirl

1-1 correspondence between answer sets and stable sets

Obtaining optimal stable sets of marriages with ASP

Our ASP program will consist of 3 parts:

- a 1st part describing a stable set,
- a 2nd part describing a stable set with another set of literals, denoted with accents,
- a 3rd part comparing the two previous and selecting optimal stable sets by saturation.

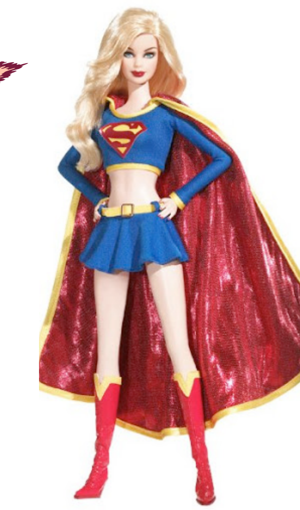
Saturation

$\Rightarrow$ 2nd program part, containing the literals to be saturated, should not contain naf

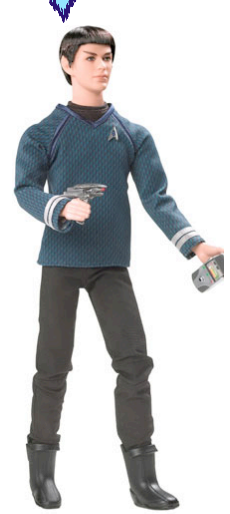
$\Rightarrow$ we compute the completion of the 1st program part and use a SAT translation to derive a disjunctive naf-free 2nd program part

Obtaining optimal stable sets of marriages with ASP

$$
\begin{aligned}
mancost(2,1) &\leftarrow accept(Spock, Supergirl)\\
mancost(2,2) &\leftarrow accept(Spock, Batgirl)\\
womancost(2,1) &\leftarrow accept(Spock, Supergirl)\\
womancost(2,1) &\leftarrow accept(Kirk, Supergirl)\\
manweight(Z) &\leftarrow \#sum\{B, A : mancost(A,B)\} = Z, \#int(Z)\\
womanweight(Z) &\leftarrow \#sum\{B, A : womancost(A,B)\} = Z, \#int(Z)\\
weight(Z) &\leftarrow manweight(X), womanweight(Y), Z = X + Y
\end{aligned}
$$



$$
\begin{aligned}
mansum'(n, X) &\leftarrow mancost'(n, X)\\
mansum'(J, Z) &\leftarrow mansum'(I, X), mancost'(J, Y),\\
&\qquad Z = X + Y, \#succ(J, I)\\
manweight'(Z) &\leftarrow mansum'(1, Z)
\end{aligned}
$$



Obtaining optimal stable sets of marriages with ASP

$$sat \leftarrow weight(X), weight'(Y), X \leq Y$$

$$\leftarrow not\ sat$$

$$\boxed{literal'} \leftarrow sat$$

every literal occuring in the 2nd program part

if an interpretation of the program does not correspond to an optimal stable set of the SMP instance, there will exist a model of the reduct w.r.t. that interpretation which does not contain sat

every answer set should contain sat

1-1 correspondence between answer sets and optimal stable sets

Proposition

For every answer set I of the ASP program $\mathcal{P}$ induced by an SMP instance with unacceptability and ties:

- $\{(x, y) \mid accept(x, y) \in I\}$ forms an egalitarian stable set of marriages,
- the optimal criterion value is the unique value v: $weight(v) \in I$.

Conversely for every egalitarian stable set $\{(x_1, y_1), \ldots, (x_k, y_k)\}$ with optimal criterion value v there exists an answer set I of $\mathcal{P}$ s.t.:

- $\{(x, y) \mid accept(x, y) \in I\} = \{(x_1, y_1), \ldots, (x_k, y_k)\}$,
- v is the unique value for which $weight(v) \in I$.

An analogous result holds for different optimality criteria, such as:

- minimal regret,
- sex-equalness,
- minimal or maximal cardinality.

Conclusion

Advantages of our approach:

- first exact method for general SMP instances with ties, regardless of the presence of unacceptability,
- easily adaptable (e.g. change optimality criterion, add constraints, switch to other SMP variant),
- can use generic, off-the-shelve solvers.

Any questions?

contact: SofieR.DeClercq@UGent.be

Answer Set Programming: technical details

literal

negation-as-failure (naf)

$$\underbrace{light\_red \vee light\_orange}_{\text{head}} \leftarrow \underbrace{cars\_stopped, not\ light\_broken}_{\text{body}}$$

Simple ASP program: only simple rules, i.e. without naf.

Aswer set of simple ASP program: minimal set of literals satisfying all rules, i.e. true head if true body.

Reduct of ASP program w.r.t. interpretation: ‘not literal’ is deleted from rule if ‘literal’ is not in the interpretation, otherwise the entire rule is deleted.

Aswer set of ASP program: interpretation which is answer set of the reduct w.r.t. that interpretation.

Technical details normal ASP to disjunctive naf-free ASP

The completion of $\mathcal{P}$ contains:

- $a \equiv \vee_i body'_i$ with $a \leftarrow body_i$ all rules in $\mathcal{P}$ with head a,
- $\bot \equiv \vee_i body'_i$ with $\leftarrow body_i$ all the constraints of $\mathcal{P}$,
- $a \equiv \bot$ for every a not occuring in any rule head,

with $body'_i$ equal to $body_i$ in which not is replaced with $\neg$.

E.g. $manpropose(Spock, Batgirl) \leftarrow not\, accept(Spock, Supergirl)$

$manpropose(Spock, Batgirl) \equiv \neg accept(Spock, Supergirl)$

completion

$$\neg manpropose(Spock, Batgirl) \vee \neg accept(Spock, Supergirl) \leftarrow$$
$$manpropose(Spock, Batgirl) \vee accept(Spock, Supergirl) \leftarrow$$

disjunctive naf-free ASP program

Literature results

	sex-equal	egalitarian	min. regret	max. card.
SMP	NP-hard	P ($O(n^2)$)	P ($O(n^2)$)	P ($O(n^2)$)
SMP + unacc.	NP-hard			P
SMP + ties		NP-hard	NP-hard	
SMP + unacc. + ties				NP-hard

The only exact algorithm tackling an NP-hard problem from this table finds a sex-equal stable set for an SMP instance in which the strict preference lists of men and/or women are bounded in length by a constant.

Optimality in practice

Egalitarian stable sets to optimally match virtual machines (VM) to servers in order to improve cloud computing by equalizing the importance of migration overhead in the data center network and VM migration performance.

Maximum cardinality in kidney-exchange.