Solving the Stable Marriage Problem using Communicating Answer Set Programming

Sofie De Clercq\textsuperscript{a}

Steven Schockaert\textsuperscript{b}, Martine De Cock\textsuperscript{a}, Ann Nowé\textsuperscript{c}

\textsuperscript{a} Dept. of Appl. Mathematics and Computer Science, Ghent University, Belgium
\textsuperscript{b} School of Computer Science and Informatics, Cardiff University, UK
\textsuperscript{c} Computational Modeling Lab, Vrije Universiteit Brussel, Belgium
Content of the talk

1. The Stable Marriage Problem (SMP)
   - Classical SMP
   - SMP with unacceptability
   - SMP with unacceptability and indifference

2. Communicating Answer Set Programming (CASP)

3. Modelling and solving the SMP in CASP
The Stable Marriage Problem

I prefer Barbie 1 to Barbie 2 and Barbie 2 to Barbie 3

I prefer Spock to Kirk and Kirk to Ken

...
The Stable Marriage Problem

I preferred Barbie 1 to Barbie 2 but got paired to Barbie 2

I preferred Spock to Kirk but got paired to Kirk

Blocking pair
The Stable Marriage Problem

Goal of the SMP: find a **stable** set of marriages without blocking pairs always exists

**Classical SMP**: as many men as women, strict and complete preferences

**SMP with unacceptability**: strict but incomplete preferences allowing the men and women to point out unacceptable partners
SMP with unacceptability

I prefer Barbie 1 to Barbie 2, but Barbie 3 is unacceptable

I prefer Spock, but Kirk and Ken are unacceptable

1 2 3
SMP with unacceptability

Blocking individual

I found Kirk unacceptable but got paired to Kirk
The Stable Marriage Problem

Goal of the SMP: find a stable set of marriages without blocking pairs

Classical SMP: as many men as women, strict and complete preferences

SMP with unacceptability: strict but incomplete preferences allowing the men and women to point out unacceptable partners

Goal of the SMP with unacceptability: find a stable set of marriages without blocking pairs and blocking individuals always exists
SMP with unacceptability and indifference: incomplete preferences, not necessarily strict

Goal of the SMP with unacceptability and indifference: find a stable set of marriages

without blocking individuals and blocking pairs i.e. a man and woman who strictly prefer each other to their actual partner

always exists
1. The Stable Marriage Problem (SMP)
   - Classical SMP
   - SMP with unacceptability
   - SMP with unacceptability and indifference

2. Communicating Answer Set Programming (CASP)

3. Modelling and solving the SMP in CASP
CASP = Communicating Answer Set Programming

ASP program: finite set of rules of the form

\[ a \leftarrow b_1, \ldots, b_n, \textit{not} c_1, \ldots, \textit{not} c_m \]

- head
- positive body
- negative body

negation-as-failure: \textit{not} c_m is true if no proof for c_m

Goal of ASP: program rules describe the problem and answer sets provide the solutions of the problem
## Answer sets of ASP programs

**Example: answer sets of an ASP program**

\[
\begin{align*}
a & \leftarrow \text{not } b \\
b & \leftarrow \text{not } a
\end{align*}
\]

<table>
<thead>
<tr>
<th>guess</th>
<th>derived from ASP program</th>
<th>result</th>
</tr>
</thead>
<tbody>
<tr>
<td>{}</td>
<td>≠ {a, b}</td>
<td>no answer set</td>
</tr>
<tr>
<td>{a}</td>
<td>= {a}</td>
<td>answer set</td>
</tr>
<tr>
<td>{b}</td>
<td>= {b}</td>
<td>answer set</td>
</tr>
<tr>
<td>{a, b}</td>
<td>≠ {}</td>
<td>no answer set</td>
</tr>
</tbody>
</table>
CASP program: unit of communicating ASP programs (components), i.e. finite set of rules of the following form:

\[ A: a \leftarrow B_1:b_1, \ldots, B_n:b_n, not C_1:c_1, \ldots, not C_m:c_m \]

There is proof that \( a \) is true in component \( A \)

There is no proof that \( c_m \) is true in component \( C_m \)

This rule belongs to component \( A \)
Example: answer sets of a CASP program \( \mathcal{P} \)

\[
\begin{align*}
Q:a & \leftarrow R:a, \text{not } Q:b & R:b & \leftarrow \text{not } Q:a \\
Q:b & \leftarrow Q:a, R:b & R:a & \leftarrow Q:b
\end{align*}
\]

\( I = \{Q:b, R:a\} \) answer set of \( \mathcal{P} \)?
Example: answer sets of a CASP program $\mathcal{P}$

\[
\begin{align*}
Q &: a \leftarrow R : a, \text{not } Q : b \\
Q &: b \leftarrow Q : a, R : b \\
R &: b \leftarrow \text{not } Q : a \\
R &: a \leftarrow Q : b
\end{align*}
\]

$I = \{Q : b, R : a\}$ answer set of $\mathcal{P}$?

$\iff I_{\downarrow Q} = \{b\}$ answer set of $Q^I$ and $I_{\downarrow R} = \{a\}$ answer set of $R^I$

reduct \hspace{1cm} \text{reduct}
Example: answer sets of a CASP program $\mathcal{P}$

\[ Q:a \leftarrow R:a, \text{not } Q:b \quad R:b \leftarrow \text{not } Q:a \]
\[ Q:b \leftarrow Q:a, R:b \quad R:a \leftarrow Q:b \]

$I = \{Q:b, R:a\}$ answer set of $\mathcal{P}$?

$\iff I_{\downarrow Q} = \{b\}$ answer set of $Q^I$ and $I_{\downarrow R} = \{a\}$ answer set of $R^I$

- **reduct**
  - delete this rule because $Q:b$ is in $I$
  - delete this rule because $R:b$ is not in $I$

- **reduct**
  - remove this part because $Q:a$ is not in $I$
  - remove this part because $Q:b$ is in $I$
Example: answer sets of a CASP program $\mathcal{P}$

$Q: a \leftarrow R: a, \neg Q: b$
$Q: b \leftarrow Q: a, R: b$
$R: b \leftarrow \neg Q: a$
$R: a \leftarrow Q: b$

$I = \{Q: b, R: a\}$ answer set of $\mathcal{P}$?

$\iff I_{\downarrow Q} = \{b\}$ answer set of $Q^I$ and $I_{\downarrow R} = \{a\}$ answer set of $R^I$

\text{reduct} \quad \text{reduct}

delete this rule because $Q: b$ is in $I$

delete this rule because $R: b$ is not in $I$

$\Rightarrow Q^I$ contains no rules

$\Rightarrow I_{\downarrow Q}$ is no answer set of $Q^I$

$\Rightarrow I$ is no answer set of $\mathcal{P}$

$\Rightarrow R^I = \{b \leftarrow, a \leftarrow\}$

$\Rightarrow I_{\downarrow R}$ is no answer set of $R^I$
Example: answer sets of a CASP program $\mathcal{P}$

\[
\begin{align*}
Q & : a \leftarrow R : a, \neg Q : b & R & : b \leftarrow \neg Q : a \\
Q & : b \leftarrow Q : a, R : b & R & : a \leftarrow Q : b \\
\end{align*}
\]

$I = \{R : b\}$ answer set of $\mathcal{P}$?
Example: answer sets of a CASP program $\mathcal{P}$

\[
Q: a \leftarrow R: a, \text{not } Q: b \\
Q: b \leftarrow Q: a, R: b \\
R: b \leftarrow \text{not } Q: a \\
R: a \leftarrow Q: b
\]

$I = \{R: b\}$ answer set of $\mathcal{P}$?

$\Leftrightarrow \quad I_{\downarrow Q} = \{\} \quad \text{answer set of } Q^I \quad \text{and} \quad I_{\downarrow R} = \{b\} \quad \text{answer set of } R^I$

reduce

reduce
Example: answer sets of a CASP program $P$

$I = \{ R: b \}$ answer set of $P$?

$\Leftrightarrow I \downarrow Q = \{ \} \text{ answer set of } Q^I \text{ and } I \downarrow R = \{ b \} \text{ answer set of } R^I$

- reduce
- delete this rule because $R: a$ is not in $I$
- remove this part because $R: b$ is in $I$

- reduce
- remove this part because $Q: a$ is not in $I$
- delete this rule because $Q: b$ is not in $I$
Example: answer sets of a CASP program $\mathcal{P}$

\[ Q:a \leftarrow R:a, \neg Q:b \]
\[ Q:b \leftarrow Q:a, R:b \]
\[ R:b \leftarrow \neg Q:a \]
\[ R:a \leftarrow Q:b \]

$I = \{R:b\}$ answer set of $\mathcal{P}$?

$\Leftrightarrow I_{\downarrow Q} = \{\} \text{ answer set of } Q^I \text{ and } I_{\downarrow R} = \{b\} \text{ answer set of } R^I$

\[ Q^I = \{b \leftarrow a\} \]
\[ I_{\downarrow Q} \text{ is an answer set of } Q^I \]
\[ I \text{ is an answer set of } \mathcal{P} \]

.delete this rule because $R:a$ is not in $I$

.remove this part because $R:b$ is in $I$

$\Rightarrow R^I = \{b \leftarrow\}$
\[ I_{\downarrow R} \text{ is an answer set of } R^I \]

.remove this part because $Q:a$ is not in $I$

.delete this rule because $Q:b$ is not in $I$
Content of the talk

1. The Stable Marriage Problem (SMP)
   - Classical SMP
   - SMP with unacceptability
   - SMP with unacceptability and indifference

2. Communicating Answer Set Programming (CASP)

3. Modelling and solving the SMP in CASP
SMP in CASP

Gale-Shapley algorithm to calculate a stable set of marriages

Using CASP to model and solve the SMP and its variants
  - transparent model
  - easily extending to other variants
  - easily adding constraints

Three components for CASP Program $\mathcal{P}$:
  - $M$: to model the intentions of the men,
  - $W$: to model the intentions of the women,
  - $A$: to model the acceptance of each other as partners.
I equally prefer Spock and Kirk, but Ken is unacceptable.

I prefer Barbie 1 to Barbie 2 and Barbie 2 to Barbie 3.

\[
\begin{align*}
M: \text{propose} (sp, b_1) & \leftarrow \\
M: \text{propose} (sp, b_2) & \leftarrow \text{not } A: \text{accept} (sp, b_1) \\
M: \text{propose} (sp, b_3) & \leftarrow \text{not } A: \text{accept} (sp, b_1), \text{not } A: \text{accept} (sp, b_2) \\
M: \text{accept} (sp, sp) & \leftarrow \text{not } A: \text{accept} (sp, b_1), \text{not } A: \text{accept} (sp, b_2), \text{not } A: \text{accept} (sp, b_3) \\
A: \text{accept} (sp, b_1) & \leftarrow M: \text{propose} (sp, b_1), W: \text{propose} (sp, b_1) \\
W: \text{propose} (sp, b_1) & \leftarrow \text{not } A: \text{accept} (ki, b_1) \\
W: \text{propose} (ki, b_1) & \leftarrow \text{not } A: \text{accept} (sp, b_1) \\
W: \text{accept} (b_1, b_1) & \leftarrow \text{not } A: \text{accept} (sp, b_1), \text{not } A: \text{accept} (ki, b_1)
\end{align*}
\]
Connection between the answer set of a CASP program induced by an SMP instance and the stable sets of marriages of that SMP instance:

- For every stable set $I = \{(m_1, w_1), (m_2, w_2), \ldots\}$ of marriages of a certain SMP instance, the CASP program induced by it has an answer set which contains $A: accept(m, w)$ for every $(m, w)$ in $I$.

- Conversely, for every answer set $I$ of the CASP program induced by a certain SMP instance, the set of marriages consisting of the couples $(m, w)$ for which $A: accept(m, w)$ is in $I$ is a stable set of marriages.

The CASP program induced by an SMP instance solves the SMP instance.
Future work

- Using CASP to find ‘optimal’ stable matches (matches that are more preferred than others)

- Extending CASP with utilities to fit cooperative games

Any questions?

contact: SofieR.DeClercq@UGent.be