Alcino Cunha

SPECIFICATION AND MODELING

RELATIONAL LOGIC

Universidade do Minho & INESC TEC

2019/20

RELATIONAL LOGIC

- Adds to first-order logic combinators to build more complex predicates (aka *relational expressions*) out of simpler ones
- Also adds closure operators, technically enhancing the expressive power of first-order logic
- It can considerably simplify the specification of some constraints
- First-order logic specifications tend to be *point-wise* (lots of quantifiers) while relational logic favours a more *point-free* style

SYNTAX

Category	Identifier
…	…
Unary (relational) expressions	A, B, …
Higher-arity (relational) expressions	Φ, Ψ, …

SYNTAX

$$
\begin{array}{rcll}
\phi & ::= & \Phi(t_1, \ldots, t_{\mathrm{ar}(\Phi)}) & \\
& | & \Phi \subseteq \Psi & \mathrm{ar}(\Phi) = \mathrm{ar}(\Psi) \\
& | & \Phi = \Psi & \mathrm{ar}(\Phi) = \mathrm{ar}(\Psi) \\
& | & |\Phi| \leq 1 & \\
& | & |\Phi| \geq 1 & \\
& | & |\Phi| = 0 & \\
& | & |\Phi| = 1 & \\
& | & \ldots &
\end{array}
$$

SYNTAX

$$
\begin{array}{lcll}
\Phi & ::= & P & \\
& | & X & \\
& | & \Phi_1 \cup \Phi_2 & \mathrm{ar}(\Phi) = \mathrm{ar}(\Psi) \\
& | & \Phi_1 \cap \Phi_2 & \mathrm{ar}(\Phi) = \mathrm{ar}(\Psi) \\
& | & \Phi_1 \setminus \Phi_2 & \mathrm{ar}(\Phi) = \mathrm{ar}(\Psi) \\
& | & \Phi_1 \times \Phi_2 & \\
& | & \Phi_1 \,.\, \Phi_2 & \mathrm{ar}(\Phi) + \mathrm{ar}(\Psi) > 2 \\
& | & A \triangleleft \Phi & \\
& | & \Phi \triangleright A & \\
& | & \Phi^{\circ} & \mathrm{ar}(\Phi) = 2 \\
& | & \Phi^{+} & \mathrm{ar}(\Phi) = 2 \\
& | & \Phi^{\star} & \mathrm{ar}(\Phi) = 2 \\
& | & \{x_1, \ldots, x_n \mid \phi\} &
\end{array}
$$

SEMANTICS

$$
\begin{aligned}
\mathcal{M}, \mathcal{A} \models \Phi \subseteq \Psi \quad \text{iff} \quad & \mathcal{M}, \mathcal{A} \models \forall x_1, \ldots, x_{\mathrm{ar}(\Phi)}. \\
& \qquad \Phi(x_1, \ldots, x_{\mathrm{ar}(\Phi)}) \rightarrow \Psi(x_1, \ldots, x_{\mathrm{ar}(\Phi)}) \\
\mathcal{M}, \mathcal{A} \models \Phi = \Psi \quad \text{iff} \quad & \mathcal{M}, \mathcal{A} \models \Phi \subseteq \Psi \wedge \Psi \subseteq \Phi \\
\mathcal{M}, \mathcal{A} \models |\Phi| \leq 1 \quad \text{iff} \quad & \mathcal{M}, \mathcal{A} \models \forall x_1, \ldots, x_{\mathrm{ar}(\Phi)}, y_1, \ldots, y_{\mathrm{ar}(\Phi)}. \\
& \qquad \Phi(x_1, \ldots, x_{\mathrm{ar}(\Phi)}) \wedge \Phi(y_1, \ldots, y_{\mathrm{ar}(\Phi)}) \rightarrow \\
& \qquad x_1 = y_1 \wedge \ldots \wedge x_{\mathrm{ar}(\Phi)} = y_{\mathrm{ar}(\Phi)} \\
\mathcal{M}, \mathcal{A} \models |\Phi| \geq 1 \quad \text{iff} \quad & \mathcal{M}, \mathcal{A} \models \exists x_1, \ldots, x_{\mathrm{ar}(\Phi)}. \Phi(x_1, \ldots, x_{\mathrm{ar}(\Phi)}) \\
\mathcal{M}, \mathcal{A} \models |\Phi| = 0 \quad \text{iff} \quad & \mathcal{M}, \mathcal{A} \models \neg(|\Phi| \geq 1) \\
\mathcal{M}, \mathcal{A} \models |\Phi| = 1 \quad \text{iff} \quad & \mathcal{M}, \mathcal{A} \models |\Phi| \leq 1 \wedge |\Phi| \geq 1
\end{aligned}
$$

SEMANTICS

$$
\begin{aligned}
\mathcal{M}, \mathcal{A} \models P(t_1, \ldots, t_n) \quad &\text{iff} \quad (\llbracket t_1 \rrbracket_{\mathcal{M},\mathcal{A}}, \ldots, \llbracket t_n \rrbracket_{\mathcal{M},\mathcal{A}}) \in \mathcal{I}(P) \\
\mathcal{M}, \mathcal{A} \models x(t) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models x = t \\
\mathcal{M}, \mathcal{A} \models (\Phi \cup \Psi)(t_1, \ldots, t_n) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models \Phi(t_1, \ldots, t_n) \vee \Psi(t_1, \ldots, t_n) \\
\mathcal{M}, \mathcal{A} \models (\Phi \cap \Psi)(t_1, \ldots, t_n) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models \Phi(t_1, \ldots, t_n) \wedge \Psi(t_1, \ldots, t_n) \\
\mathcal{M}, \mathcal{A} \models (\Phi \setminus \Psi)(t_1, \ldots, t_n) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models \Phi(t_1, \ldots, t_n) \wedge \neg\Psi(t_1, \ldots, t_n) \\
\mathcal{M}, \mathcal{A} \models (\Phi \times \Psi)(t_1, \ldots, t_n) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models \Phi(t_1, \ldots, t_{\mathrm{ar}(\Phi)}) \wedge \Psi(t_{\mathrm{ar}(\Phi)+1}, \ldots, t_n) \\
\mathcal{M}, \mathcal{A} \models (\Phi \,.\, \Psi)(t_1, \ldots, t_n) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models \exists x.\Phi(t_1, \ldots, t_{\mathrm{ar}(\Phi)-1}, x) \wedge \\
&\qquad\qquad\qquad\qquad \Psi(x, t_{\mathrm{ar}(\Phi)}, \ldots, t_n) \\
\mathcal{M}, \mathcal{A} \models (A \triangleleft \Phi)(t_1, \ldots, t_n) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models A(t_1) \wedge \Phi(t_1, \ldots, t_n) \\
\mathcal{M}, \mathcal{A} \models (\Phi \triangleright A)(t_1, \ldots, t_n) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models \Phi(t_1, \ldots, t_n) \wedge A(t_n) \\
\mathcal{M}, \mathcal{A} \models (\Phi^\circ)(t, u) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models \Phi(u, t) \\
\mathcal{M}, \mathcal{A} \models (\Phi^+)(t, u) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models (\Phi \cup \Phi.\Phi \cup \Phi.\Phi.\Phi \cup \ldots)(t, u) \\
\mathcal{M}, \mathcal{A} \models (\Phi^\star)(t, u) \quad &\text{iff} \quad \mathcal{M}, \mathcal{A} \models (\Phi^+)(t, u) \vee t = u \\
\mathcal{M}, \mathcal{A} \models (\{x_1, \ldots, x_n \mid \phi\})(t_1, \ldots, t_n) \quad &\text{iff} \quad \phi[t_1/x_1, \ldots, t_n/x_n]
\end{aligned}
$$

SEMANTICS

- It is more intuitive to assume the existence of a function $[\![\cdot]\!]_{\mathcal{M},\mathcal{A}}$ that computes the value of a relational expression in a model

$$\mathcal{M}, \mathcal{A} \models \Phi(t_1, \ldots, t_n) \quad \text{iff} \quad ([\![t_1]\!]_{\mathcal{M},\mathcal{A}}, \ldots, [\![t_n]\!]_{\mathcal{M},\mathcal{A}}) \in [\![\Phi]\!]_{\mathcal{M},\mathcal{A}}$$

- The following slides explain this function by example

SEMANTICS BY EXAMPLE

$$
\begin{aligned}
\mathcal{A}(x) &= \texttt{File1} \\
\mathcal{I}(\texttt{Trash}) &= \{(\texttt{File1}), (\texttt{File2})\} \\
\mathcal{I}(\texttt{Protected}) &= \{(\texttt{File2}), (\texttt{File3})\} \\
\mathcal{I}(\texttt{link}) &= \{(\texttt{File1}, \texttt{File1}), (\texttt{File1}, \texttt{File2}), (\texttt{File2}, \texttt{File3})\}
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{Trash} \cup \texttt{Protected}]\!] &= \{(\texttt{File1}), (\texttt{File2}), (\texttt{File3})\} \\
[\![\texttt{Trash} \cap \texttt{Protected}]\!] &= \{(\texttt{File2})\} \\
[\![\texttt{Trash} \setminus \texttt{Protected}]\!] &= \{(\texttt{File1})\} \\
[\![x \times \texttt{Trash}]\!] &= \{(\texttt{File1}, \texttt{File1}), (\texttt{File1}, \texttt{File2})\} \\
[\![\texttt{Protected} \triangleleft \texttt{link}]\!] &= \{(\texttt{File2}, \texttt{File3})\} \\
[\![\texttt{link} \triangleright \texttt{Trash}]\!] &= \{(\texttt{File1}, \texttt{File1}), (\texttt{File1}, \texttt{File2})\} \\
[\![\texttt{link}^\circ]\!] &= \{(\texttt{File1}, \texttt{File1}), (\texttt{File2}, \texttt{File1}), (\texttt{File3}, \texttt{File2})\} \\
[\![\{z, w \mid z = x \wedge \texttt{Trash}(w)\}]\!] &= \{(\texttt{File1}, \texttt{File1}), (\texttt{File1}, \texttt{File2})\}
\end{aligned}
$$

SEMANTICS BY EXAMPLE

$$
\begin{aligned}
\mathcal{A}(x) &= \texttt{File1} \\
\mathcal{I}(\texttt{Trash}) &= \{(\texttt{File1}), (\texttt{File2})\} \\
\mathcal{I}(\texttt{link}) &= \{(\texttt{File1}, \texttt{File1}), (\texttt{File1}, \texttt{File2}), (\texttt{File2}, \texttt{File3})\} \\
\mathcal{I}(\texttt{name}) &= \{(\texttt{File1}, \texttt{Name1}), (\texttt{File2}, \texttt{Name2}), (\texttt{File3}, \texttt{Name3})\}
\end{aligned}
$$

$$
\begin{aligned}
[\![x\ .\ \texttt{link}]\!] &= \{(\texttt{File1}), (\texttt{File2})\} \\
[\![\texttt{link}\ .\ x]\!] &= \{(\texttt{File1})\} \\
[\![\texttt{Trash}\ .\ \texttt{name}]\!] &= \{(\texttt{Name1}), (\texttt{Name2})\} \\
[\![\texttt{link}\ .\ \texttt{name}]\!] &= \{(\texttt{File1}, \texttt{Name1}), (\texttt{File1}, \texttt{Name2}), (\texttt{File2}, \texttt{Name3})\} \\
[\![\texttt{link}\ .\ \texttt{link}^{\circ}]\!] &= \{(\texttt{File1}, \texttt{File1}), (\texttt{File2}, \texttt{File2})\} \\
[\![\texttt{link}^{+}]\!] &= \{(\texttt{File1}, \texttt{File1}), (\texttt{File1}, \texttt{File2}), (\texttt{File2}, \texttt{File3}), (\texttt{File1}, \texttt{File3})\}
\end{aligned}
$$

RELATIONAL LOGIC SYNTAX IN ALLOY

Alloy	Math
Φ **in** Ψ	$\Phi \subseteq \Psi$
Φ = Ψ	$\Phi = \Psi$
lone Φ	$\lvert\Phi\rvert \leq 1$
some Φ	$\lvert\Phi\rvert \geq 1$
no Φ	$\lvert\Phi\rvert = 0$
one Φ	$\lvert\Phi\rvert = 1$

RELATIONAL LOGIC SYNTAX IN ALLOY

Alloy	Math
Φ + Ψ	$\Phi \cup \Psi$
Φ & Ψ	$\Phi \cap \Psi$
Φ - Ψ	$\Phi \setminus \Psi$
Φ -> Ψ	$\Phi \times \Psi$
Φ . Ψ	$\Phi . \Psi$
A <: Φ	$A \triangleleft \Psi$
Φ :> A	$\Phi \triangleright A$
~ Φ	Φ°
^ Φ	Φ^{+}
* Φ	$\Phi^{\star}$
{ x : A | ϕ }	$\{x \mid x \in A \wedge \phi\}$

BACK TO THE THRASH

```
sig Name {}
sig File {
  name : set Name,
  link : set File
}
sig Trash in File {}
sig Protected in File {}
```

FORMULA EXAMPLES

```
-- The trash is empty
no Trash

-- Every file is either in the trash or protected
File = Trash + Protected
no Trash & Protected

-- There are no links
no link

-- Every file has at least one name
all x : File | some x.name

-- Every file has at most one name
all x : File | lone x.name
```

A QUESTION OF STYLE

```
-- File names are unique, point-wise style
all x, y : File, n : Name | x->n in name and y->n in name implies x = y

-- File names are unique, navigational (mixed) style
all n : Name | lone name.n

-- File names are unique, point-free style
name.~name in iden
```