Relational Formal Modelling

Formal Methods I, 2002/03

J.N. Oliveira

Functions are not enough

Partiality:

```
vdm> p tl [ ]
l. 1, c. 4:
  Run-Time Error 77: The sequence was empty
vdm>  p 2/0
l. 1, c. 3:
  Run-Time Error 76: Division with zero
vdm>
```

Functions such as `tl`, `/`, `hd` (and many others!) are **partial**

Functions are not enough

```
gets : set of nat -> nat * set of nat
gets(s) == let a in set s
           in mk_(a,s \ {a}) ;
```

is not only partial

```
vdm> p gets({})
/home/jno/work/x.vdm, l. 4, c. 25:
  Run-Time Error 53: The binding environment was empty
vdm>
```

but also **nondeterministic**:

$$gets\{a,b\} = \langle a, \{b\}\rangle \vee gets\{a,b\} = \langle b, \{a\}\rangle$$

Specifications as “properties”

- Specification of *square root*:

$$(sqrt\ x)^2 = x$$

that is

$$sq \cdot sqrt = id$$

(= $sqrt$ has left inverse sq)

- Specification of $sort$:

$$l' = sort\, l \ \Leftrightarrow\ (IsOrdered\, l') \wedge IsPermutation(l', l)$$

Relational approach

Need to model

- total/**partial** functions
- **nondeterminism**
- **properties**, datatype **invariants** and loop-invariants
- orders and **inductive** structures
- vagueness or **under-specification** . . .

Relational approach

Need to model

- total/**partial** functions
- **nondeterminism**
- **properties**, datatype **invariants** and loop-invariants
- orders and **inductive** structures
- vagueness or **under-specification** . . .

$\Rightarrow$ adoption of binary relations, which have a long tradition in the . . .

Pre/post specification style

```
Sort(l: seq of int) r: seq of int
post IsPermutation(r,l) and IsOrdered(r);

IsPermutation: seq of int * seq of int -> bool
IsPermutation(l1,l2) == ....................

IsOrdered: seq of int -> bool
IsOrdered(l) == ............
```

```
gets(s: set of nat) r: nat * set of nat
pre  card s > 0
post r.#1 in set s and r.#2 = s \ {r.#1} ;
```

Pre/post specification layout

```
Spec(a: A) r: B
pre Precond(a)
post Postcond(r,a);
```

where

$$Precond \quad : \quad A \longrightarrow 2$$

$$Postcond \quad : \quad B \times A \longrightarrow 2$$

leads to the **binary relation** approach:

$$Postcond \in 2^{B \times A} \quad \Leftrightarrow \quad Postcond \subseteq B \times A$$

Relations as Arrows

- From now on, an arrow

$$B \xleftarrow{R} A$$

means a **binary relation** from A (source) to B (target) and write bRa to denote that pair $\langle b, a \rangle$ is in R.

- **Ordering** on relations:

$$R \subseteq S \;\equiv\; bRa \Rightarrow bSa$$

$R \subseteq S$ means that R is either **less defined** or **more deterministic** than S.

Basic relational combinators

Given $C \xleftarrow{S} B$ and $B \xleftarrow{R} A$

- **Composition** $S \cdot R$ is s.t.

$$c(S \cdot R)a$$

holds wherever there exists some $b \in B$ such that $cSb \wedge bRa$.

- **Converse** R°

$$a(R^\circ)b \equiv bRa$$

- **Meet** $R \cap S$ — recall set-theoretical intersection

Basic Relation Calculus

Composition is associative:

$$R \cdot (S \cdot T) = (R \cdot S) \cdot T$$

Composition is monotonic:

$$\frac{\begin{array}{c} R \subseteq S \\ T \subseteq U \end{array}}{(R \cdot T) \subseteq (S \cdot U)}$$

Identity

$$R \cdot id = id \cdot R = R$$

Relational Equality

"Ping-pong" rule:

$$R = S \;\equiv\; R \subseteq S \wedge S \subseteq R$$

Indirect proof:

$$R = S \;\equiv\; \forall X.(X \subseteq R \equiv X \subseteq S)$$

Meet and converse

$\cap$-universal

$$X \subseteq (R \cap S) \;\equiv\; (X \subseteq R) \wedge (X \subseteq S)$$

Converse

Involution : $(R^\circ)^\circ = R$

Order-preserving : $R \subseteq S \equiv R^\circ \subseteq S^\circ$

Contravariance : $(R \cdot S)^\circ = S^\circ \cdot R^\circ$

Converse distributes over $\cap$ (proof in next slide):

$$(R \cap S)^\circ = R^\circ \cap S^\circ$$

Elegant (indirect) proofs

$$
\begin{array}{ll}
 & X \subseteq R^\circ \cap S^\circ \\
\equiv & \quad \{ \cap\text{-universal} \} \\
 & (X \subseteq R^\circ) \wedge (X \subseteq S^\circ) \\
\equiv & \quad \{ \text{involution} \} \\
 & (X^\circ \subseteq R) \wedge (X^\circ \subseteq S) \\
\equiv & \quad \{ \cap\text{-universal} \} \\
 & X^\circ \subseteq (R \cap S) \\
\equiv & \quad \{ \text{involution} \} \\
 & X \subseteq (R \cap S)^\circ \\
thus & \quad \{ \text{indirection} \} \\
 & R^\circ \cap S^\circ = (R \cap S)^\circ
\end{array}
$$

Dedekind's rule

also known as the **modular law**:

$$(R \cdot S) \cap T \;\subseteq\; R \cdot (S \cap (R^\circ \cdot T))$$

Dually (apply converses and rename):

$$(R \cdot S) \cap T \;\subseteq\; (R \cap (T \cdot S^\circ)) \cdot S$$

Symmetrical equivalent statement:

$$(R \cdot S) \cap T \;\subseteq\; (R \cap (T \cdot S^\circ)) \cdot (S \cap (R^\circ \cdot T))$$

= "weak right-distribution of meet over composition".

Properties (A)

- $\cap$-cancellation — from $X = R \cap S$ in $\cap$-universal infer:

$$R \cap S \subseteq R \ \wedge \ R \cap S \subseteq S$$

- $\cap$-abbreviation:

$$R \subseteq S \ \equiv \ R = R \cap S$$

- $\cap$-idempotency:

$$R \cap R \ = \ R$$

Properties (B)

$\cap$ is commutative:

$$R \cap S \;=\; S \cap R$$

$\cap$ is associative:

$$R \cap (S \cap T) \;=\; (R \cap S) \cap T$$

$\cap$-fusion:

$$\begin{aligned} T \cdot (R \cap S) &\subseteq (T \cdot R) \cap (T \cdot S) \\ (R \cap S) \cdot T &\subseteq (R \cdot T) \cap (S \cdot T) \end{aligned}$$

Orders and their taxonomy (A)

An order (or endo-relation) $A \xleftarrow{R} A$ is

reflexive:	iff $id_A \subseteq R$
coreflexive:	iff $R \subseteq id_A$
transitive:	iff $R \cdot R \subseteq R$
anti-symmetric:	iff $R \cap R^\circ \subseteq id_A$
symmetric:	iff $R \subseteq R^\circ (\equiv R = R^\circ)$
connected:	iff $R \cup R^\circ = \top$

where $A \xleftarrow{\top} A$ is the largest relation of its type.

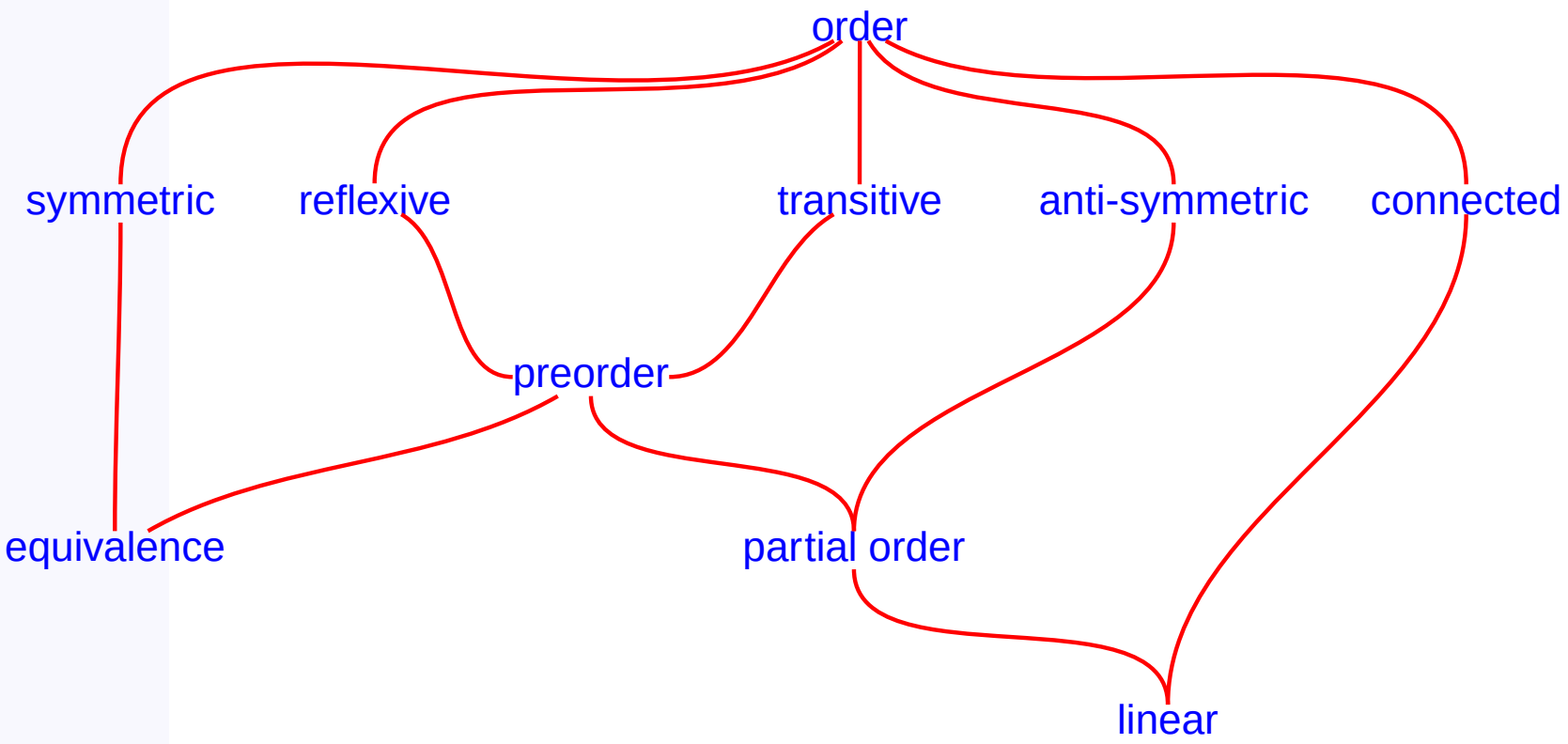
Order taxonomy (B)

- **Preorders** are reflexive and transitive orders.
- **Partial** orders are anti-symmetric preorders
- **Linear** orders are connected partial orders
- **Equivalences** are symmetric preorders
- **Predicates** are coreflexive orders: the "meaning" of a predicate $Bool \xleftarrow{\phi} A$ is a coreflexive relation $[\![\phi]\!]$ such that

$$\phi\, a \;=\; a[\![\phi]\!]a$$

i.é, it maps every a which validates ϕ onto itself.

Order taxonomy (C)



Meaning of Pre/post specs

```
Spec(a: A) r: B
pre Precond(a)
post Postcond(r,a);
```

means

$$Spec \stackrel{\text{def}}{=} Postcond \cdot Precond$$

where $Precond$ is the coreflexive equivalent of `Precond`.

Derived combinators

- **Kernel** of $B \xleftarrow{R} A$ is $A \xleftarrow{\mathit{ker}\ R} A$ defined by

$$\mathit{ker}\ R \stackrel{\text{def}}{=} R^\circ \cdot R$$

- **Image** of $B \xleftarrow{R} A$ is $B \xleftarrow{\mathit{img}\ R} B$ defined by

$$\mathit{img}\ R \stackrel{\text{def}}{=} R \cdot R^\circ$$

- Duality:

$$\begin{aligned} \mathit{ker}\ (R^\circ) &= \mathit{img}\ R \\ \mathit{img}\ (R^\circ) &= \mathit{ker}\ R \end{aligned}$$

Turning implicit specifications...

Sorting in VDM-SL notation:

```
Sort(l: seq of int) r: seq of int
post IsOrdered(r) and IsPermutation(r,l);
```

where

```
IsPermutation: seq of int * seq of int -> bool
IsPermutation(l1,l2) ==
  forall e in set (elems l1 union elems l2) &
    card {i | i in set inds l1 & l1(i) = e} =
    card {i | i in set inds l2 & l2(i) = e};
```

. . . into relational models

. . . abbreviates to

$$Sort \stackrel{\text{def}}{=} [\![IsOrdered]\!] \cdot IsPermutation$$

. . . into relational models

. . . abbreviates to

$$Sort \stackrel{\text{def}}{=} [\![IsOrdered]\!] \cdot (\textit{ker}\ seq2bag)$$

assuming

```
seq2bag: seq of int -> map int to nat1
seq2bag(l) ==
  { e |-> card { i | i in set inds l & l(i) = e } |
      e in set elems l };
```

. . . into relational models

. . . abbreviates to

$$Sort \stackrel{\text{def}}{=} [\![IsOrdered]\!] \cdot (\textit{ker}\ seq2bag)$$

assuming

```
seq2bag: seq of int -> map int to nat1
seq2bag(l) ==
  { e |-> card { i | i in set inds l & l(i) = e } |
      e in set elems l };
```

`IsPermutation` equivalence $\Leftarrow$ *ker* f is always reflexive, symmetric and transitive.

Properties of kernel and image

Order-preservation:

$$R \subseteq S \;\Rightarrow\; \mathit{ker}\ R \subseteq \mathit{ker}\ S$$
$$R \subseteq S \;\Rightarrow\; \mathit{img}\ R \subseteq \mathit{img}\ S$$

Symmetry:

$$(\mathit{ker}\ R)^\circ \;=\; \mathit{ker}\ R$$
$$(\mathit{img}\ R)^\circ \;=\; \mathit{img}\ R$$

Also:

$$R \subseteq R \cdot \mathit{ker}\ R \qquad (= \mathit{img}\ R \cdot R)$$

Entireness and simplicity

An **entire** (or total) relation is such that its kernel is reflexive:

$$R \text{ is entire} \equiv id \subseteq \mathit{ker}\ R$$

A **simple** (or functional) relation is such that its image is coreflexive:

$$R \text{ is simple} \equiv \mathit{img}\ R \subseteq id$$

Simplicity is the dual of entireness. Simple relations are also called **partial functions**.

(Total) functions

Functions are both simple and entire relations, usually denoted by lowercase letters f:

$$\underbrace{id \subseteq f^\circ \cdot f}_{\text{entire}} \;\wedge\; \underbrace{f \cdot f^\circ \subseteq id}_{\text{simple}}$$

Thus:

$$f \subseteq R \Rightarrow R \text{ is entire}$$
$$R \subseteq f \Rightarrow R \text{ is simple}$$

In general, *"larger than entire means entire"* and *"smaller than simple means simple"*

Surjectiveness and injectiveness

More taxonomy:

- R is **surjective** iff R° is entire
- R is **injective** iff R° is simple

Facts:

$$R \text{ is entire and injective} \equiv \mathit{ker}\ R = id$$
$$R \text{ is simple and surjective} \equiv \mathit{img}\ R = id$$

Bijections

f is **bijective** iff it is an injective and surjective function (thus simple and entire)

$$B \xleftarrow{f} A \text{ bijective} \equiv \mathit{ker}\ f = id \wedge \mathit{img}\ f = id$$

In this case

$$id = f^{\circ} \cdot f \ \wedge \ f \cdot f^{\circ} = id$$

Kernel of a function

Kernel of a total function is an equivalence: it is symmetric, reflexive and transitive:

$$
\begin{array}{ll}
 & a'(\mathit{ker}\ f)a \\
\equiv & \quad \{\ \mathit{ker}\ f = f^\circ \cdot f\ \} \\
 & \exists a''.(a'\, f^\circ\, a'') \wedge (a'' = f\, a) \\
\equiv & \quad \{\ \text{converse of a function, pointwise}\ \} \\
 & \exists a''.(a'' = f\, a') \wedge (a'' = f\, a) \\
\equiv & \quad \{\ \text{equality is transitive}\ \} \\
 & f\, a' = f\, a
\end{array}
$$

Properties of coreflexives

For any T and coreflexive R:

$$R \cdot T \subseteq T$$
$$T \cdot R \subseteq T$$

Coreflexives are symmetric and transitive:

$$R = R^{\circ} = R \cdot R = R \cap id$$

Meet of two coreflexives is composition:

$$R \cap S = R \cdot S$$