Inputs, Outputs, and Composition in the Logic of Information Flows

Bart Bogaerts

joint work with
Eugenia Ternovska (SFU)
Heba Aamer, Dimitri Surinx, Jan Van den Bussche (UHasselt)

July 17, 2020

ARTIFICIAL INTELLIGENCE RESEARCH GROUP

CONCLUSION

- Context: Logic of Information Flows
  $\pm$ first-order logic, with dynamic semantics and composition
- Q1: “what does it mean to be an input/output of a complex LIF expression?”
  - Semantic definition (undecidable)
  - Syntactic definition (optimal)
- Q2: “Is composition primitive in this logic?”
  Positive and negative results (using inputs and outputs)

CONTEXT: LOGIC OF INFORMATION FLOWS

- KR formalism for modeling (combination of) modules
- Module: a relation (input arguments, output arguments)
- Connecting: (extension of) first-order logic

(Also applications with higher-order relations, fixpoint logic)

DYNAMIC SEMANTICS (LAW OF INERTIA)

Binary relation ***Increment*** (1st argument: input; 2nd argument: output)
Standard (static) semantics, assignment ν:

$$D, \nu \models \mathit{Increment}(x, y) \quad \Leftrightarrow \quad \nu(y) = \nu(x) + 1$$

Dynamic semantics, pair of assignments (ν_1, ν_2):

$$D, (\nu_1, \nu_2) \models \mathit{Increment}(x; y) \quad \Leftrightarrow \quad \nu_2(y) = \nu_1(x) + 1 \text{ and } \nu_2 = \nu_1 \text{ elsewhere}$$

DYNAMIC SEMANTICS (LAW OF INERTIA)

Binary relation ***Increment*** (1st argument: input; 2nd argument: output)
Standard (static) semantics, assignment ν:

$$D, \nu \models \mathit{Increment}(x, y) \quad \Leftrightarrow \quad \nu(y) = \nu(x) + 1$$

Dynamic semantics, pair of assignments (ν_1, ν_2):

$$D, (\nu_1, \nu_2) \models \mathit{Increment}(x; y) \quad \Leftrightarrow \quad \nu_2(y) = \nu_1(x) + 1 \text{ and } \nu_2 = \nu_1 \text{ elsewhere}$$
$$D, (\nu_1, \nu_2) \models \mathit{Increment}(x; x) \quad \Leftrightarrow \quad \nu_2(x) = \nu_1(x) + 1 \text{ and } \nu_2 = \nu_1 \text{ elsewhere}$$

DYNAMIC SEMANTICS (LAW OF INERTIA)

Binary relation ***Increment*** (1st argument: input; 2nd argument: output)
Standard (static) semantics, assignment ν:

$$D, \nu \models \mathit{Increment}(x, y) \quad \Leftrightarrow \quad \nu(y) = \nu(x) + 1$$

Dynamic semantics, pair of assignments (ν_1, ν_2):

$$D, (\nu_1, \nu_2) \models \mathit{Increment}(x; y) \quad \Leftrightarrow \quad \nu_2(y) = \nu_1(x) + 1 \text{ and } \nu_2 = \nu_1 \text{ elsewhere}$$
$$D, (\nu_1, \nu_2) \models \mathit{Increment}(x; x) \quad \Leftrightarrow \quad \nu_2(x) = \nu_1(x) + 1 \text{ and } \nu_2 = \nu_1 \text{ elsewhere}$$

$$[\![\alpha]\!]_D = \{(\nu_1, \nu_2) \mid D, (\nu_1, \nu_2) \models \alpha\}$$

This is called a Binary Relation on Valuations (BRV)

OPERATIONS IN LIF (OPERATIONS ON BRVS)

- Boolean connectives (union, intersection, difference)
- Selection (equality), e.g., $\sigma^{l}_{x=y}M$
- Cylindrification (existential quantification/projection), e.g., $\mathrm{cyl}^{r}_{x}M$
- Composition: $R \,;\, S = \{(\nu_1, \nu_3) \mid \exists \nu_2 : (\nu_1, \nu_2) \in R \wedge (\nu_2, \nu_3) \in S\}$

Dynamic version of Codd's relational algebra plus composition

QUESTION 1

What are inputs and outputs of compound modules in LIF?
(given inputs and outputs of atomic building blocks)

WHAT ARE THE INPUTS, OUTPUTS OF AN EXPRESSION?

Atomic modules (relations):

- input arguments are specified in the vocabulary
- remaining arguments are outputs

E.g. relation *Increment* of input arity 1, total arity 2
Expression $\mathit{Increment}(x; y)$ has input var x, output var y
Expression $\mathit{Increment}(x; x)$ has input var x, output var x

For complicated expressions, not so obvious

- $R(x; y) \;;\; S(y; z)$
- $R(x; y) \cap S(y; z)$

SEMANTIC OUTPUTS

Definition

Variable x is an output of expression E if...
...there exists instance D, assignments ν_1, ν_2 such that

- $D, (\nu_1, \nu_2) \models E$
- $\nu_2(x) \neq \nu_1(x)$

Informally: E can change x.

SEMANTIC INPUTS

Definition

Variable x is an input of expression E if…
…there exists instance D, assignments ν_1, ν_2, ν_1' such that

- $D, (\nu_1, \nu_2) \models E$
- $\nu_1' = \nu_1$ except on x
- every ν_2' such that $D, (\nu_1', \nu_2') \models E$ differs from ν_2 on at least one output

Informally: changing x changes the behaviour of E on the outputs.

(UN)DECIDABILITY

Theorem

The problem "Is ***x*** *a semantic input (output) of* ***E****?" is undecidable.*

Gives rise to syntactic approximations

α	$I(\alpha)$	$O(\alpha)$
id	$\emptyset$	$\emptyset$
$M(\bar{x};\bar{y})$	$\{x_1,\ldots,x_n\}$ where $\bar{x}=x_1,\ldots,x_n$	$\{y_1,\ldots,y_n\}$ where $\bar{y}=y_1,\ldots,y_n$
$\alpha_1 \cup \alpha_2$	$I(\alpha_1)\cup I(\alpha_2)\cup(O(\alpha_1)\triangle O(\alpha_2))$	$O(\alpha_1)\cup O(\alpha_2)$
$\alpha_1 \cap \alpha_2$	$I(\alpha_1)\cup I(\alpha_2)\cup(O(\alpha_1)\triangle O(\alpha_2))$	$O(\alpha_1)\cap O(\alpha_2)$
$\alpha_1 - \alpha_2$	$I(\alpha_1)\cup I(\alpha_2)\cup(O(\alpha_1)\triangle O(\alpha_2))$	$O(\alpha_1)$
$\alpha_1 \,;\, \alpha_2$	$I(\alpha_1)\cup(I(\alpha_2)-O(\alpha_1))$	$O(\alpha_1)\cup O(\alpha_2)$
$\mathrm{cyl}^l_x(\alpha_1)$	$I(\alpha_1)-\{x\}$	$O(\alpha_1)\cup\{x\}$
$\mathrm{cyl}^r_x(\alpha_1)$	$I(\alpha_1)$	$O(\alpha_1)\cup\{x\}$
$\sigma^{lr}_{x=y}(\alpha_1)$	$\begin{cases} I(\alpha_1) & \text{if } x =_{\mathrm{syn}} y \text{ and } y \notin O(\alpha_1) \\ I(\alpha_1)\cup\{x,y\} & \text{if } x \neq_{\mathrm{syn}} y \text{ and } y \notin O(\alpha_1) \\ I(\alpha_1)\cup\{x\} & \text{otherwise} \end{cases}$	$\begin{cases} O(\alpha_1)-\{x\} & \text{if } x =_{\mathrm{syn}} y \\ O(\alpha_1) & \text{otherwise} \end{cases}$
$\sigma^{l}_{x=y}(\alpha_1)$	$\begin{cases} I(\alpha_1) & \text{if } x =_{\mathrm{syn}} y \\ I(\alpha_1)\cup\{x,y\} & \text{otherwise} \end{cases}$	$O(\alpha_1)$
$\sigma^{r}_{x=y}(\alpha_1)$	$\begin{cases} I(\alpha_1) & \text{if } x =_{\mathrm{syn}} y \\ I(\alpha_1)\cup(\{x,y\}-O(\alpha_1)) & \text{otherwise} \end{cases}$	$O(\alpha_1)$

SYNTACTIC APPROXIMATIONS OF INPUTS AND OUTPUTS

Our proposal is

- Sound:
  $O(E)$ contains all semantic outputs;
  $I(E)$ determines E on $O(E)$
  (and hence $I(E)$ contains all semantic inputs)
- Compositional:
  $I(E_1 \texttt{ op } E_2)$ and $O(E_1 \texttt{ op } E_2)$ depend only on $I(E_j)$, $O(E_j)$, and $\texttt{op}$
- Optimal:
  Most precise compositional and sound definition

INPUTS AND OUTPUTS

Example

Consider the LIF expression

$$\alpha := \sigma^{l}_{x=y}\sigma^{r}_{x=y}R(x;y)$$

In this case, $O^{\mathrm{sem}}(\alpha) = \emptyset$. However, it can be verified that $O^{\mathrm{syn}}(\alpha) = \{y\}$.

PRIMITIVITY OF COMPOSITION

- LIF is roughly first-order logic with a dynamic semantics and composition
- Do we really need composition?
- The answer: “Yes” and “No”

NON-PRIMITIVITY: INPUTS AND OUTPUTS

Theorem

Let α and β be LIF expressions such that $\mathbf{I}(\beta) \cap \mathbf{O}(\beta) = \emptyset$. Then, $\alpha \,;\, \beta$ is equivalent to

$$\gamma := \mathrm{cyl}^{r}_{\mathbf{O}(\beta)}(\alpha) \cap \mathrm{cyl}^{l}_{\mathbf{O}(\alpha)}(\beta).$$

NON-PRIMITIVITY: SUFFICIENT VARIABLES

Theorem

If the set of variables is infinite, then every LIF expression is equivalent to a LIF expression without composition.

Idea:

- Using the extra variables, we can "force" any expression to become IO-disjoint

PRIMITIVITY: FINITE SET OF VARIABLES

Theorem

If the set of variables is finite, then composition is primitive.

Idea:

- LIF with n variables can express existence of a $3n$-clique in a graph.
- Without composition, LIF with n variables can be translated into $FO(2n)$

RELATED & FUTURE WORK

- Inputs and outputs used to study relation with executable FO [Aamer et al, ICDT2020]
- Dynamic predicate logic [Groenendijk and Stokhof, 1991]: semantics in terms of pairs of valuations. Different dynamics:
  - LIF: underlying dynamic system
  - DPL: dynamics of parsing
- Exploit inputs and outputs for problem decomposition
- Consider other operations (principles remain), e.g., converse, fixpoints

CONCLUSION

- Context: Logic of Information Flows
  $\pm$ first-order logic, with dynamic semantics and composition
- Q1: "what does it mean to be an input/output of a complex LIF expression?"
  - Semantic definition (undecidable)
  - Syntactic definition (optimal)
- Q2: "Is composition primitive in this logic?"
  Positive and negative results (using inputs and outputs)