A Syntax of V^-

Roger Burtonpatel

October 22, 2023

We present a grammar of V^-:

⟨*program*⟩ ::= {⟨*def*⟩}

⟨*def*⟩ ::= val ⟨*name*⟩ ⟨*exp*⟩
| ⟨*exp*⟩

⟨*name*⟩ ::= any token that is not an *int-lit*, does not contain whitespace, and is not a ⟨*value-constructor-name*⟩ or a reserved word. ⟨*value-constructor-name*⟩ ::= `cons` | ⟨*exp*⟩ ::= ⟨*integer-literal*⟩
| ⟨*name*⟩
| ⟨*name*⟩
| ⟨*guarded-if*⟩
| ⟨*lambda*⟩

⟨*integer-literal*⟩ ::= token composed only of digits, possibly prefixed with a + or −.

⟨*lambda*⟩ ::= λ{⟨*name*⟩}. ⟨*exp*⟩

⟨*guarded-if*⟩ ::= `if` [⟨*guarded-exp*⟩ { [] ⟨*guarded-exp*⟩ }] `fi`

⟨*guarded-exp*⟩ ::= ⟨*exp*⟩
| `E` {⟨*logical-var*⟩}. ⟨*guarded-exp*⟩
| ⟨*exp*⟩`;` ⟨*guarded-exp*⟩
| ⟨*logical-var*⟩ = ⟨*exp*⟩`;` ⟨*guarded-exp*⟩
| ⟨*exp*⟩ = ⟨*exp*⟩`;` ⟨*guarded-exp*⟩
| ⟨*guarded-exp*⟩ | ⟨*guarded-exp*⟩
| `one`({⟨*guarded-exp*⟩})
| `all`({⟨*guarded-exp*⟩})

⟨*logical-var*⟩ ::= a fresh name (cannot be `lam`- or `E`-bound in this scope).

Add patterns, value constructors (application), choice, one, all.

1 Forms of Judgement for V^-:

Metavariables:
$v,\ v'$: value
eq : equation
`?t` : a temporarily-stuck equation
fail : failure
$\rho,\ \hat{\rho}$: environment: $name \to \mathcal{V}_\perp$
$\mathcal{T}$: Context of all temporarily stuck equations
e : An expression
ge : A guarded expression

Forms of judgement on equations:

$$\langle \rho, eq \rangle \rightarrowtail \hat{\rho} \quad \text{(EQUATIONSUCCESS)}$$

$$\langle \rho, eq \rangle \rightarrowtail \texttt{?t} \quad \text{(EQUATIONTEMPSTUCK)}$$

$$\langle \rho, eq \rangle \rightarrowtail \textbf{fail} \quad \text{(EQUATIONFAIL)}$$

Forms of judgement on expressions:

$$\langle \rho, e \rangle \Downarrow v \quad \text{(EVALSUCC)}$$

$$\langle \rho, e \rangle \Downarrow \textbf{fail} \quad \text{(EVALFAIL)}$$

$$\langle \rho, \mathcal{T}, ge \rangle \Downarrow v \quad \text{(EVALGESUCC)}$$

$$\langle \rho, \mathcal{T}, ge \rangle \Downarrow \textbf{fail} \quad \text{(EVALGEFAIL)}$$

$$\langle \rho, \texttt{?t}; \mathcal{T} \rangle \rightarrowtail \langle \hat{\rho}, \mathcal{T} \rangle \quad \text{(CTXTORHO)}$$

In english:
An equation is either solved to produce bindings that extend an environemnt, gets temporarily stuck, or fails.
An expression either evaluates to produce a value or fails.

Other important guidelines (where do we put these?):

$$\rho \subseteq \rho'$$
$$\text{when dom } \rho \subseteq \text{dom } \rho'$$
$$\text{and } \forall x \in \text{dom } \rho : \rho(x) \subseteq \rho'(x)$$

When $\langle \rho, e \rangle \rightarrowtail \rho'$, then $\rho \subseteq \rho'$.

2 Rules (Big-step Operational Semantics) for V^-:

$$\text{(GE-CTX-STUCK)} \quad \frac{\langle \rho, eq \rangle \rightarrowtail \texttt{?t} \qquad \langle \rho, \texttt{?t}; \mathcal{T} \rangle \rightarrowtail \langle \hat{\rho}, \mathcal{T} \rangle \qquad \langle \rho, \mathcal{T}, ge \rangle \Downarrow v}{\langle \rho, eq; \mathcal{T}, ge \rangle \Downarrow v}$$

$$\text{(GE-CTX-STUCK)} \quad \frac{\dfrac{\langle \rho, eq \rangle \rightarrowtail \texttt{?t}}{\langle \rho, \texttt{?t}; \mathcal{T} \rangle \rightarrowtail \langle \hat{\rho}, \mathcal{T} \rangle} \qquad \langle \rho, \mathcal{T}, ge \rangle \Downarrow v}{\langle \rho, eq; \mathcal{T}, ge \rangle \Downarrow v}$$

Which of the above two do you prefer?

$$\begin{array}{l}\text{(GE-CTX-STUCK)} \\ \text{(GE-EQ-STUCK)}\end{array} \quad \frac{\dfrac{\langle \rho, eq \rangle \rightarrowtail \texttt{?t} \qquad \langle \rho, \texttt{?t}; \mathcal{T} \rangle \rightarrowtail \langle \hat{\rho}, \mathcal{T} \rangle \qquad \langle \rho, \mathcal{T}, ge \rangle \Downarrow v}{\langle \rho, eq; \mathcal{T}, ge \rangle \Downarrow v}}{\langle \rho, \mathcal{T}, eq; ge \rangle \Downarrow v}$$