REASONING ABOUT PARALLEL PROGRAMS WITH LOCAL VARIABLES

Stephen Brookes

Carnegie Mellon University
School of Computer Science

MFPS'94

SHARED VARIABLE PARALLELISM

- Parallel imperative programs reading and writing shared memory

- $C_1 \| C_2$ modelled by interleaving of atomic actions

- Synchronization using conditional atomic action: **await** B **then** C

Coarseness Assumption

Assignment and boolean expressions are atomic.

OPERATIONAL SEMANTICS

$$\frac{\langle C_1, s\rangle \rightarrow \langle C_1', s'\rangle}{\langle C_1 \| C_2, s\rangle \rightarrow \langle C_1' \| C_2, s'\rangle}$$

$$\frac{\langle C_2, s\rangle \rightarrow \langle C_2', s'\rangle}{\langle C_1 \| C_2, s\rangle \rightarrow \langle C_1 \| C_2', s'\rangle}$$

$$\frac{\langle C_1, s\rangle \mathrm{term} \quad \langle C_2, s\rangle \mathrm{term}}{\langle C_1 \| C_2, s\rangle \mathrm{term}}$$

$$\frac{\langle B, s\rangle \rightarrow^* \texttt{tt} \quad \langle C, s\rangle \rightarrow^* \langle C', s'\rangle \mathrm{term}}{\langle \textbf{await } B \textbf{ then } C, s\rangle \rightarrow \langle \textbf{skip}, s'\rangle}$$

cf. Hennessy and Plotkin (1979)

PROGRAM BEHAVIOR

- Partial correctness:

$$\mathcal{M}[\![C]\!] \;=\; \{(s,s') \mid \langle C,s\rangle \rightarrow^{*} \langle C',s'\rangle\text{term}\}$$

- Strong correctness:

$$\begin{aligned}\mathcal{M}[\![C]\!] \;=\;& \{(s,s') \mid \langle C,s\rangle \rightarrow^{*} \langle C',s'\rangle\text{term}\}\\ &\cup\{(s,\perp) \mid \langle C,s\rangle \rightarrow^{\omega}\}\end{aligned}$$

- Total correctness:

$$\begin{aligned}\mathcal{M}[\![C]\!] \;=\;& \{(s,s') \mid \langle C,s\rangle \rightarrow^{*} \langle C',s'\rangle\text{term}\}\\ &\cup\{(s,s') \mid s' \in S_{\perp} \;\&\; \langle C,s\rangle \rightarrow^{\omega}\}\end{aligned}$$

- Deadlock:

$$\mathcal{M}[\![C]\!] \;=\; \{(s,s') \mid \langle C,s\rangle \rightarrow^{*} \langle C',s'\rangle\text{dead}\}$$

FULL ABSTRACTION

Milner (1977):

A semantics is *fully abstract* if two phrases have the same meaning precisely when they induce the same behavior in all program contexts.

- A natural criterion for judging merit of a semantics.
- Often difficult to achieve.
- A fully abstract semantics supports compositional reasoning.
- Failure of full abstraction may suggest:
  - defective semantic model
  - missing language features

TRACE SEMANTICS

Transitions: $\Sigma = S \times S$
Transition traces: $\Sigma^\infty = \Sigma^+ \cup \Sigma^\omega$
Trace semantics: $\mathcal{T}[\![C]\!] \subseteq \Sigma^\infty$

- A trace represents a sequence of snapshots taken during computation, allowing for possible interruption.
- Operational definition:

$$\begin{aligned}\mathcal{T}[\![C]\!] = \{(s_0, s_0')(s_1, s_1') \ldots (s_k, s_k') \mid \\ \langle C, s_0\rangle \to^* \langle C_1, s_0'\rangle \ \& \\ \langle C_1, s_1\rangle \to^* \langle C_2, s_1'\rangle \ \& \\ \ldots\ldots\ldots\ldots\ldots\ldots \ \& \\ \langle C_k, s_k\rangle \to^* \langle C', s_k'\rangle \mathrm{term}\}\end{aligned}$$

- Partial correctness behavior corresponds to "interference-free" subset:

$$\mathcal{M}[\![C]\!] = \{(s, s') \mid (s, s') \in \mathcal{T}[\![C]\!]\}.$$

PROPERTIES

- $\mathcal{T}[\![C]\!]$ is closed under **stuttering**:

$$\alpha\beta \in \mathcal{T}[\![C]\!] \ \Rightarrow\ \alpha(s,s)\beta \in \mathcal{T}[\![C]\!].$$

- $\mathcal{T}[\![C]\!]$ is closed under **mumbling**:

$$\alpha(s,s')(s',s'')\beta \in \mathcal{T}[\![C]\!] \ \Rightarrow\ \alpha(s,s'')\beta \in \mathcal{T}[\![C]\!].$$

DEFINITION

For $T \subseteq (S \times S)^+$ let $T^\dagger$ be smallest closed set including T.

FACT

Closed sets of traces, ordered by inclusion, form a complete lattice.

DENOTATIONAL SEMANTICS

$$\mathcal{T}[\![\textbf{skip}]\!] \;=\; \{(s,s) \mid s \in S\}^\dagger$$

$$\mathcal{T}[\![I{:=}E]\!] \;=\; \{(s,[s|I=n]) \mid (s,n) \in \mathcal{E}[\![E]\!]\}^\dagger$$

$$\begin{aligned}\mathcal{T}[\![C_1;C_2]\!] \;&=\; \mathcal{T}[\![C_1]\!];\mathcal{T}[\![C_2]\!] \\ &= \{\alpha\beta \mid \alpha \in \mathcal{T}[\![C_1]\!] \;\&\; \beta \in \mathcal{T}[\![C_2]\!]\}^\dagger\end{aligned}$$

$$\begin{aligned}\mathcal{T}[\![C_1\|C_2]\!] \;&=\; \mathcal{T}[\![C_1]\!]\|\mathcal{T}[\![C_2]\!] \\ &=\; \bigcup\{\alpha\|\beta \mid \alpha \in \mathcal{T}[\![C_1]\!] \;\&\; \beta \in \mathcal{T}[\![C_2]\!]\}^\dagger\end{aligned}$$

$$\mathcal{T}[\![\textbf{await}\ B\ \textbf{then}\ C]\!] \;=\; \{(s,s') \in \mathcal{T}[\![C]\!] \mid (s,\texttt{tt}) \in \mathcal{B}[\![B]\!]\}^\dagger$$

$$\mathcal{T}[\![\textbf{if}\ B\ \textbf{then}\ C_1\ \textbf{else}\ C_2]\!] \;=\; \mathcal{T}[\![B]\!];\mathcal{T}[\![C_1]\!] \cup \mathcal{T}[\![\neg B]\!];\mathcal{T}[\![C_2]\!]$$

$$\mathcal{T}[\![\textbf{while}\ B\ \textbf{do}\ C]\!] \;=\; (\mathcal{T}[\![B]\!];\mathcal{T}[\![C]\!])^*;\mathcal{T}[\![\neg B]\!]$$

where $\mathcal{T}[\![B]\!] = \{(s,s) \mid (s,\texttt{tt}) \in \mathcal{B}[\![B]\!]\}$.

PROPERTIES

- Sequential and parallel composition are continuous operations on closed sets of traces.

- Loop semantics can be expressed as a least fixed point:

$$\mathcal{T}[\![\textbf{while } B \textbf{ do } C]\!] = \mu T.(\mathcal{T}[\![B]\!]; \mathcal{T}[\![C]\!]; T \ \cup \ \mathcal{T}[\![\neg B]\!])$$

- Denotational and operational definitions of $\mathcal{T}$ coincide.

INFINITE TRACES

- Let Σ be $S \times S$.
- $\mathcal{T}[\![C]\!] \subseteq \Sigma^\infty = \Sigma^* \cup \Sigma^\omega$
- $\mathcal{T}[\![C]\!]$ is closed under stuttering and mumbling.
- Extend *concatenation* to infinite traces:

$$\alpha\beta = \alpha \quad \text{if } \alpha \text{ is infinite.}$$

- Loop semantics as an "operational" fixed point:

$$\begin{aligned}\mathcal{T}[\![\mathbf{while}\ B\ \mathbf{do}\ C]\!] \ = \\ &(\mathcal{T}[\![B]\!]; \mathcal{T}[\![C]\!])^*; \mathcal{T}[\![\neg B]\!] \\ &\cup (\mathcal{T}[\![B]\!]; \mathcal{T}[\![C]\!])^\omega\end{aligned}$$

- Full abstraction for strong correctness.

FAIR PARALLELISM

"No parallel component is delayed forever"

- $\mathcal{T}[\![C_1 \| C_2]\!] = \mathcal{T}[\![C_1]\!] \| \mathcal{T}[\![C_2]\!]$
- $T_1 \| T_2 = \cup\{\alpha \| \beta \mid \alpha \in T_1 \ \&\ \beta \in T_2\}^{\dagger}$
- $\alpha \| \beta = \{\gamma \mid (\alpha, \beta, \gamma) \in fairmerge\}$
- $fairmerge = (L^* R R^* L)^\omega \cup (L \cup R)^* A$, where

  $L = \{(\sigma, \epsilon, \sigma) \mid \sigma \in \Sigma\}$
  $R = \{(\epsilon, \sigma, \sigma) \mid \sigma \in \Sigma\}$
  $A = \{(\epsilon, \alpha, \alpha), (\alpha, \epsilon, \alpha) \mid \alpha \in \Sigma^* \cup \Sigma^\omega\}$

  cf. Park (1979)
- Fair parallel composition is a *continuous function* on closed sets of traces.

RELATED WORK

- Abrahamson (1979), Park (1979):
  - no stuttering or mumbling
  - not fully abstract

- de Boer, Kok, Palamidessi, Rutten (1991):
  - only restricted mumbling
  - different language and behavior
  - ignores fairness

- Abadi, Plotkin (1993):
  - finite traces, stuttering, mumbling
  - closed under prefix
  - full abstraction for safety properties
  - ignores fairness

FURTHER RESEARCH

- Connection with logic:
  - generalized Hoare logics
  - safety and liveness
  - modal logics (e.g. LTL, CTL, μ-calculus)
- Laws of program equivalence, for:
  - program transformation, derivation
  - parallelization
  - simplification of program proofs
- Full abstraction for fairly *communicating processes*
  - Hoare's CSP
  - Milner's CCS with value-passing