Errata: Model-checking Task Parallel Programs for Data-race

Radha Nakade Eric Mercer Peter Aldous Kyle Storey Benjamin Ogles
Joshua Hooker Sheridan Jacob Powell Jay McCarthy

May 29, 2019

1 join semantics

The published version contains an error in the semantics of the join function; it reverses the order of the edge added to the graph.

1.1 Original version

The *join* function makes clear in the graph where completed tasks synchronize at an **await**-statement: $G' = \text{join}(G, R, n_2, n)$ joins the nodes in R and the node from the recently completed task n_2 to the new node n in the computation graph G to n':

$$\begin{aligned} G' &= G[N \overset{\cup}{\mapsto} \{n\}] \\ &\quad [E \overset{\cup}{\mapsto} \{(n, n_2)\} \cup \{(n, n_i) \mid n_i \in R\}] \end{aligned}$$

1.2 Corrected version

The *join* function makes clear in the graph where completed tasks synchronize at an **await**-statement: $G' = \text{join}(G, R, n_2, n)$ joins the nodes in R and the node from the recently completed task n_2 to the new node n in the computation graph G:

$$\begin{aligned} G' &= G[N \overset{\cup}{\mapsto} \{n\}] \\ &\quad [E \overset{\cup}{\mapsto} \{(n_2, n)\} \cup \{(n_i, n) \mid n_i \in R\}] \end{aligned}$$

2 Await-done

The published version calls join incorrectly; the second argument is $\Gamma(R)$ but should be $R(r)$. The correct version of the rule for Await-done is as follows:

Await-done

$$\frac{\begin{array}{c} m_1 = m \setminus (r \mapsto ((\ell_2, S[\textbf{done } v], d_2, n_2), m_2)) \\ s = d(v) \qquad (m_1 \cup m_2)(r) = \emptyset \qquad n' = \text{fresh}() \\ \Gamma' = \Gamma[R(r) \mapsto \emptyset][G \mapsto \text{join}(\Gamma(G), R(r), n_2, n')] \end{array}}{\begin{array}{c} (\Gamma, \sigma, C[(\ell, S[\textbf{await } r], d, n), m]) \to \\ (\Gamma', \sigma, C[(\ell, S[s], d, n'), m_1 \cup m_2]) \end{array}}$$