Modernizing Parallel Code with Pattern Analysis

Supplementary Material: Additional Pattern Definitions

Roberto Castañeda Lozano
University of Edinburgh
School of Informatics
Edinburgh, United Kingdom
rcastae@inf.ed.ac.uk

Murray Cole
University of Edinburgh
School of Informatics
Edinburgh, United Kingdom
mic@inf.ed.ac.uk

Björn Franke
University of Edinburgh
School of Informatics
Edinburgh, United Kingdom
bfranke@inf.ed.ac.uk

This supplement completes the definitions provided in Section 4 of Castañeda Lozano *et al.*'s PPoPP '21 paper [1].

Conditional maps. Conditional maps are modeled as regular maps except that their last constraints are replaced by constraints requiring that only *some* components have outgoing arcs:

$$\begin{aligned} &\exists \mathrm{c} \in \mathrm{M}, \exists \langle u, v\rangle \in \mathit{arcs}(\mathit{DDG}) : \\ &\qquad u \in \mathit{nodes}(\mathrm{c}) \land v \notin \mathit{nodes}(\mathrm{c}) && \text{(1a)} \\ \land\ &\exists \mathrm{c} \in \mathrm{M}, \nexists \langle u, v\rangle \in \mathit{arcs}(\mathit{DDG}) : \\ &\qquad u \in \mathit{nodes}(\mathrm{c}) \land v \notin \mathit{nodes}(\mathrm{c}) && \text{(1b)} \end{aligned}$$

Linear map-reductions. A *linear map-reduction* within *DDG* is a tuple $\langle \mathrm{M}, \mathrm{R} \rangle$ where M is a map (2a), R is a linear reduction (2b), and both are subject to channeling constraints as specified in (2c-2d).

In a linear map-reduction, each map component produces an output data element that is only taken as input by its corresponding reduction component. This is modeled by requiring that all nodes in each map component can reach the nodes of its corresponding component in the linear reduction (2c), and forbidding all other arcs between the map and the linear reduction (2d).

$$\begin{aligned} &\mathit{linear\text{-}map\text{-}reduction}(\langle \mathrm{M}, \mathrm{R} \rangle, \mathit{DDG}) \iff \\ &\mathit{map}(\mathrm{M}, \mathit{DDG}) && \text{(2a)} \\ \land\ &\mathit{linear\text{-}reduction}(\mathrm{R}, \mathit{DDG}) && \text{(2b)} \\ \land\ &\forall i \in [1, n], \forall u \in \mathit{nodes}(\mathrm{M}_i), \forall v \in \mathit{nodes}(\mathrm{R}_i) : \\ &\qquad \mathit{reaches}(u, v) && \text{(2c)} \\ \land\ &\forall i, j \in [1, n] \mid i \neq j, \nexists \langle u, v \rangle \in \mathit{arcs}(\mathit{DDG}) : \\ &\qquad u \in \mathit{nodes}(\mathrm{M}_i) \land v \in \mathit{nodes}(\mathrm{R}_j) && \text{(2d)} \end{aligned}$$

Tiled map-reductions. A *tiled map-reduction* within *DDG* is a tuple $\langle \mathrm{M}, \langle \mathrm{RP}, \mathrm{RF} \rangle \rangle$ where M is a map (3a), $\langle \mathrm{RP}, \mathrm{RF} \rangle$ is a tiled reduction (3b), and both are subject to channeling constraints as specified in (3c-3e).

In a tiled map-reduction, each map component produces an output data element that is only taken as input by its corresponding partial reduction component. This is modeled by requiring that all nodes in each map component can reach the nodes of its corresponding partial reduction component (3c), and forbidding all other arcs between the map and the tiled reduction (3d-3e).

$$\begin{aligned} &\mathit{tiled\text{-}map\text{-}reduction}(\langle \mathrm{M}, \langle \mathrm{RP}, \mathrm{RF} \rangle \rangle, \mathit{DDG}) \iff \\ &\mathit{map}(\mathrm{M}, \mathit{DDG}) && \text{(3a)} \\ \land\ &\mathit{tiled\text{-}reduction}(\langle \mathrm{RP}, \mathrm{RF} \rangle, \mathit{DDG}) && \text{(3b)} \\ \land\ &\forall i \in [1, n], \forall u \in \mathit{nodes}(\mathrm{M}_i), \\ &\forall v \in \mathit{nodes}(\mathrm{RP}_{(i \operatorname{div} m + 1)(i \bmod m + 1)}) : \mathit{reaches}(u, v) && \text{(3c)} \\ \land\ &\forall i \in [1, n], \forall j \in [1, m], \\ &\forall k \in [1, p] \mid j \neq i \operatorname{div} m + 1 \land k \neq i \bmod m + 1, \\ &\nexists \langle u, v \rangle \in \mathit{arcs}(\mathit{DDG}) : \\ &\qquad u \in \mathit{nodes}(\mathrm{M}_i) \land v \in \mathit{nodes}(\mathrm{RP}_{jk}) && \text{(3d)} \\ \land\ &\forall i \in [1, n], \forall j \in [1, m], \nexists \langle u, v \rangle \in \mathit{arcs}(\mathit{DDG}) : \\ &\qquad u \in \mathit{nodes}(\mathrm{M}_i) \land v \in \mathit{nodes}(\mathrm{RF}_m) && \text{(3e)} \end{aligned}$$

References

[1] Roberto Castañeda Lozano, Murray Cole, and Björn Franke. 2021. Modernizing Parallel Code with Pattern Analysis. In *Principles and Practice of Parallel Programming*. ACM.