Expressing Coarse-Grain Dependencies Among Tasks in Shared Memory Programs

Designers of embedded systems face tight constraints on resources, response time and cost. The ability to analyze embedded systems is essential to timely delivery of new designs. Many analysis techniques model parallel programs as task graphs. Task graphs capture the worst-case execution times of individual program tasks and the data dependencies among these. This paper introduces two compiler directives which let programmers annotate source code with data dependencies among tasks. Compiler analysis overapproximates the actual dependencies among tasks. The directives help eliminate potential data dependencies that do not occur at runtime. This lets tools compute more accurate task graphs from the annotated code. The correct use of the directives cannot be verified at compile time. Therefore, the check for correct use is done at runtime—not unlike dynamic array bounds checking in many languages. The overhead of verifying the correct use of the directives was measured on a set of benchmarks on two platforms. The overhead of runtime checks was found to be negligible in all instances.

General information
State: Published
Organisations: Department of Informatics and Mathematical Modeling
Contributors: Larsen, P., Karlsson, S., Madsen, J.
Pages: 652-660
Publication date: 2011
Peer-reviewed: Yes

Publication information
Journal: IEEE Transactions on Industrial Informatics
Volume: 7
Issue number: 4
ISSN (Print): 1551-3203
Ratings:
BFI (2019): BFI-level 1
Web of Science (2019): Indexed yes
BFI (2018): BFI-level 1
Web of Science (2018): Indexed yes
BFI (2017): BFI-level 1
Scopus rating (2017): CiteScore 8.37 SJR 1.599 SNIP 3.183
Web of Science (2017): Impact factor 5.43
Web of Science (2017): Indexed yes
BFI (2016): BFI-level 1
Scopus rating (2016): CiteScore 8.29 SJR 2.114 SNIP 3.418
Web of Science (2016): Impact factor 6.764
Web of Science (2016): Indexed yes
BFI (2015): BFI-level 1
Scopus rating (2015): CiteScore 7.57 SJR 2.381 SNIP 3.242
Web of Science (2015): Impact factor 4.708
BFI (2014): BFI-level 1
Scopus rating (2014): CiteScore 8.17 SJR 2.151 SNIP 3.883
Web of Science (2014): Indexed yes
BFI (2013): BFI-level 1
Scopus rating (2013): CiteScore 9.38 SJR 1.975 SNIP 4.424
Web of Science (2013): Impact factor 8.785
ISI indexed (2013): ISI indexed yes
Web of Science (2013): Indexed yes
BFI (2012): BFI-level 1
Scopus rating (2012): CiteScore 5.56 SJR 0.934 SNIP 3.078
Web of Science (2012): Impact factor 3.381
ISI indexed (2012): ISI indexed yes
Web of Science (2012): Indexed yes
BFI (2011): BFI-level 1
Scopus rating (2011): CiteScore 4.08 SJR 0.802 SNIP 2.412