

Abstract

Acyclic path profile is an abstraction of dynamic control flow paths of procedures and has been found to be useful in a wide spectrum of activities. Unfortunately, the runtime overhead of obtaining such a profile can be high, limiting its use in practice.

In this work, we present partitioned path profiling (P3) which runs K copies of the program in parallel, each with the same input but on a separate core, and collects the profile only for a subset of intra-procedural paths in each copy, thereby, distributing the overhead of profiling. P3 identifies “profitable” procedures and assigns disjoint subsets of paths of a profitable procedure to different copies for profiling. To obtain exact execution frequencies of a subset of paths, we design a new algorithm, called PSPP. All paths of an unprofitable procedure are assigned to the same copy. P3 uses the classic Ball-Larus algorithm for profiling unprofitable procedures. Further, P3 attempts to evenly distribute the profiling overhead across the copies. To the best of our knowledge, P3 is the first algorithm for parallel path profiling.

We have applied P3 to profile several programs in the SPEC 2006 benchmark. Compared to sequential profiling, P3 substantially reduced the runtime overhead on these programs averaged across multiple inputs. The reduction was 23%, 43% and 56% on average for 2, 4 and 8 cores respectively. P3 also performed better than a coarse-grained approach that treats all procedures as unprofitable and distributes them across available cores. For 2 cores, the profiling overhead of P3 was on an average 5% less compared to the coarse-grained approach across these programs. For 4 and 8 cores, it was respectively 18% and 25% less.