SAND2011-3224C

International Conference on Supercomputing

Automatic Extraction of SST/macro Skeleton Models

Damian Dechev
Scalable Computing R&D Department
Sandia National Laboratories
ddechev@sandia.gov

Abstract

The utilization of large scale parallel event simulators such as SST/macro requires that skeleton models of underlying software systems and architectures be created. Implementing such models by abstracting the designs of large scale parallel applications requires a substantial manual effort and introduces the hazards of human errors. We outline an approach for the automatic extraction of SST/macro skeleton models from large scale parallel applications. Our methodology for deriving SST/macro skeleton models is based on the use of extensible and open-source ROSE compiler infrastructure. The SST/macro skeleton models are then combined with appropriate models of the network and hardware configurations.

Brief Description

Our motivation is to address the critical issue of interpolating application behavior for exascale machines when driving the SST/macro simulator by simplifying the process of creating skeleton applications. The key idea is to construct these reduced skeletons automatically from full applications by using program analysis modules provided by ROSE compiler. One aspect of compiler analysis to achieve this optimization is by abstracting away fragments of redundant computations and message data whose values do not affect the skeleton application's state. Another aspect is to identify the values that could affect program performance and to isolate the computations and communications that determine these values. These aspects of compiler analysis use a technique called program slicing which is done by extraction module of ROSE. The translation module of ROSE encodes the insertion and rewriting of actions from the input application code. The implementation of the translation module is achieved by matching expression patterns provided by SST/macro simulator against the sliced-out AST. This converts the ROSE AST annotated with input application instances into AST with SST/macro-specific nodes, thus creating the skeleton model.

Categories & Subject Descriptors: D.2.4 [Software Engineering]: Software/Program Verification---Statistical methods; D.2.4 [Software Engineering]: Processors---Translator writing systems and compiler generators; D.4.8 [Software Engineering]: Performance---Simulation;

General Terms: Algorithms, Performance, Design, Experimentation, Human Factors, Verification.