1

TV

Programming with Tiles

2006   PPoPP   Hierarchically Tiled Array

HTA

Hierarchically Tiled Arrays(HTAs) are arrays that may be partitioned into tiles. THese tiles can be conventional arays or lower level HTAs. Tiles can be distributed across processors in a distributed-memory machine or be stored in a single machine according to a user specified layout

The C++ implementation of the HTA class is a library with ~18000 lines of code. It only contains header files, as most classes in the library are C:: templates to facilitate inlining.

Dynamic partitioning

cache oblivious algorithms, FLAME require dynamic changes of the tile layout.
part/rmPart
    TBB requres more lines of code, variables, and data types than the HTA to express the same problem

Overlapped tiling

Stencil codes benefit from tiling, because they increase locality and determine data distribution when running in parallel.
    programmers create a shadow or ghost region around each tile that contains a copy of the elements of the neighbor tiles
    automatically or manually update

Evaluation

            sequential(LU   3D Jacobi)

parallel(Parallel Merge, MG/LU NAS)
Readability/Productivity
    the programmng effort[17]
    the cyclomatic number[22]
    lines of code

A Portable Runtime Interface For Multi-Level Mmeory Hierarchies

for moving data and computation through parallel machines with multi-level memory hierarchies

for multi-core/SMP, Cell B.E/

The Runtime Interface

adaptation of the Sequoia compiler

    initialize/setup of the machine, including communicaton resources and resources at all levels where tasks can be executed
    data transfers between memory levels using asynchronous bulk transfers between arrays
    task execution at specified levels of the machine

DISK
Top API   Bottom API

Multiscalar Processors

Multiscalar processors use a new, aggressive implementation paradigm for extractign large quantities of ILP from ordinary high level languages programs.