A SYSTOLIC APPROACH TO LOOP PARTITIONING AND MAPPING INTO FIXED SIZE DISTRIBUTED MEMORY ARCHITECTURES

Ioannis Drositis, Nektarios Koziris, George Papakonstantinou and Panayotis Tsanakas
National Technical University of Athens
Department of Electrical and Computer Engineering
Division of Computer Science
Zografou Campus, 157-73 Zografou, Athens, Greece.
{jdros, nkoziris}@cslab.ece.ntua.gr

ABSTRACT

The most important issue in sequential program parallelization is the efficient assignment of computations into different processing elements. Nested loops are the most extensive part of a program, in terms of time execution. Loop parallelization involves two subsequent steps: “time” partitioning of the index space, into disjoint groups of concurrent computations (partitioning) and group assignment into the target parallel architecture (mapping). This paper presents a new method for the problem of mapping of nested FOR-loops, with uniform dependencies, into mesh-connected parallel architectures. The proposed partitioning is based on loop mapping for systolic arrays. We consider the virtual array of systolic cells, produced when applying a bijective linear transformation $T$ onto the index space $J^n$. Matrix $T$ includes the time hyperplane $\Pi$, which determines the optimal time schedule for systolic arrays, and $S$, which projects the $n$-dimensional index space onto an $(n-1)$-dimensional virtual array of systolic cells. Our proposed method divides the virtual array of cells into a fixed number of clusters, equal to the number of available real processors. Inside every cluster, the neighboring virtual cells are grouped together. The virtual array cutting is performed along all possible boundary directions, so as to minimize inter-cluster communication links and equilibrate the number of virtual cells for every cluster. After selecting the optimal cut, in terms of total link cost, along every dimension, each formed cluster is assigned to a different processor. The proposed space mapping, cuts down overall communication delays, while using a fixed number of processors from an $(n-1)$-dimensional mesh-connected distributed architecture.

Keywords: Loop partitioning, loop mapping, hyperplane method, virtual array of processors, space cuts, distributed architectures.