EE482C: Advanced Computer Organization
Spring 2001-2002

Project Proposal: Mapping vector codes to streams

Project members : Mehdi Baradaran Tahoori, Paul Wang Lee

1. Goal

Automatically translate vector codes into efficient stream code.

2. Near Term Goal

To develop methods to translate certain forms of simple vector or dense matrix code into snippets of Brook code that is relatively efficient. Writing a simple translator to demonstrate its effectiveness.

3. Approach

After reviewing relevant literature, we will start by manually translating sample vector/dense matrix codes into the high level stream programming language Brook. With this experience, we hope to find a number of translations that are conducive to automation. After formulating the candidate targets and the transformations, we will attempt to generalize these transformations to apply to more general cases. Finally, we will write a translator that uses these transformations to translates a small subset of the vector programming language into efficient stream code.

4. Anticipated sub-problems

- Identifying dependencies among the elements of the vectors between operations in order to determine the appropriate grouping of records and the division of streams.
- The division of a sequence of vector operations into kernels.
- How to recognize and handle large intermediate state; i.e. when to generate new streams vs. when to maintain in local memory.
- Determining the optimal record size for the tradeoff between reduction of extra-kernel communication and the size of intermediate state needed in the kernel.
- There may be (will be) code that is inherently very difficult or impossible to convert to efficient stream code. Find techniques to minimize the performance penalty when handling such classes of codes.