Exploiting Parallelism with Hypermatrices

José R. Herrero, Juan J. Navarro
{josepr,juanjo}@ac.upc.es

Computer Architecture Department
Universitat Politècnica de Catalunya
Barcelona
Spain
Outline

- Introduction
- Potential sources of parallelism
- Exploiting ILP
- Application
  - Sparse codes
  - Dense codes
- Conclusions
Hypermatrix structure

Matrix

HyperMatrix
Sources of Parallelism

Replication of resources:

- Machines in a network
- Processors within a machine
- Functional Units within a processor

Parallelism at different levels:

- Process
- Thread
- Instruction
Exploiting ILP

Goal: The sparse code requires efficient operation on small data submatrices

- Reduce data submatrix size while keeping good BLAS3 performance

Idea [Euro-Par’03]:

- Fix parameters at compilation time
- Choose best algorithm
  - Loop unrolling factors
  - Loop orders
Matrix multiplication performance on small matrices

\[ \mathbf{C} = \mathbf{C} - \mathbf{A} \times \mathbf{B}^\dagger \]

R10000 250 MHz (500 Mflops peak)

Exploiting Parallelism with Hypermatrices
José R. Herrero, Comp. Arch. Dept., UPC
Sparse codes: Overhead

Can store 0's within data submatrices

- Storage
- Computation

Trade-off in data submatrix size

- BLAS3 efficiency
- (Useless) operation on 0's
Reducing Overhead & Increasing Performance

- Efficient kernels which operate on small data submatrices
- Windows within data submatrices
Windows within data submatrices: Goal

Store and use only a part of a data submatrix

- Reduce unnecessary computation
- Reduce storage
Windows: Definition

Data Submatrix

left column
right column

top row

bottom row

Window: subset of data submatrix
Windows: Usage (I)

Operation can be reduced
Windows: Usage (II)

Operation can be skipped
Results: Context information

- MIPS R10000 @ 250 MHz (500 Mflops peak)
- Sequential code
- Data submatrices of fixed size
- Large problems solved In-Core
- Ordered using METIS
- Linear Programming problems
  - NetLib
  - Multicommodity Network Flow generators
Performance of several sparse Cholesky factorization codes.
R10000: Dense Matrix Multiplication

\[ C = C = A \times B^t \]
Exploiting Parallelism with Hypermatrices

José R. Herrero, Comp. Arch. Dept., UPC
Alpha 21264: Dense Matrix Multiplication

![Graph showing performance over dimension]

- **HM**
- **DGEMM**
- **DGEMM_ATLAS**
Alpha 21264: Dense Cholesky
Summary

Techniques which improve performance of Hypermatrix codes:

- Efficient kernels which operate on small data submatrices (ILP)
- Extension for sparse codes
  - Rectangular data submatrices
  - Windows within data submatrices