Ph.D. Research Seminar on

High Performance Numerical Linear Algorithms

by

Gaurav Trivedi

under the guidance of

Prof. H. Narayanan

and

Prof. Madhav P. Desai

Department of Electrical Engineering

Indian Institute of Technology

Mumbai

May, 2000

Presentation Outline

I will talk about

- Motivation
- Fundamentals of Parallel Processing
- Implementation Details
- Performance Analysis
- Linear Algorithms
- Proposed future work

Motivation

- Limitation of the traditional computers.
- Advancement in hardware and architecture enables to exploit parallelism.
- Lagging in software development relatively to hardware development.

Fundamentals of Parallel Processing

- Models of multiunit processor:
  - Tightly Coupled Model
  - Loosely Coupled Model

1. Advantages of Tightly coupled model
   - Increased throughput
   - Less cost effective
   - Less communication overhead
   - Increased reliability
2. Advantages of Loosely coupled model
   - Resource sharing
   - Computation speedup
   - Less cost effective
   - Reliability

Classification of parallel processors

- Types of parallel processors
  - Single Instruction Stream Single Data Stream (SISD)
  - Single Instruction Stream Multiple Data Stream (SIMD)
  - Multiple Instruction Stream Single Data Stream (MISD)
  - Multiple Instruction Stream Multiple Data Stream (MIMD)

Different Architectures

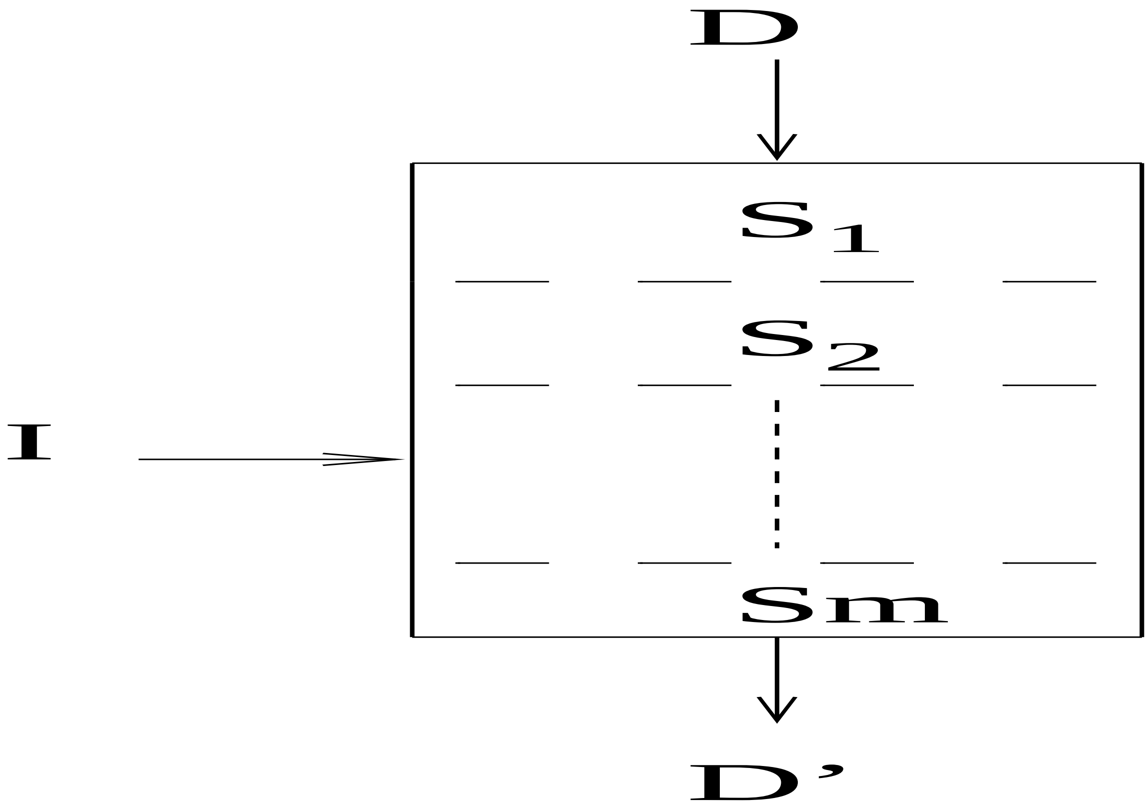

Figure 1: SISD Organization

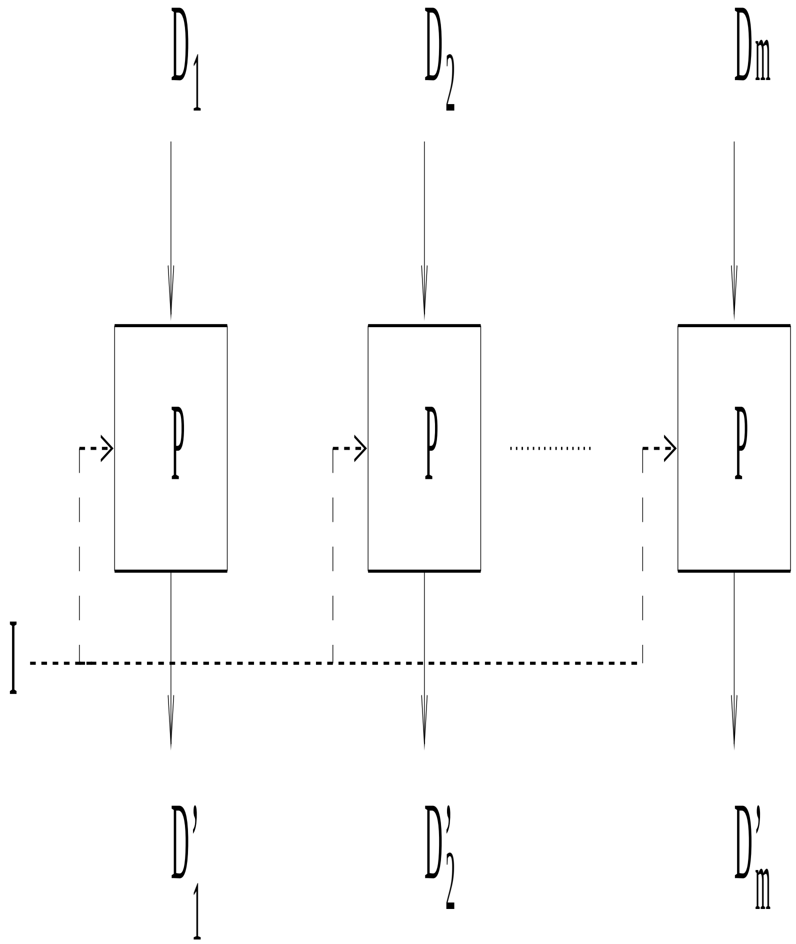

Figure 2: SIMD Organization

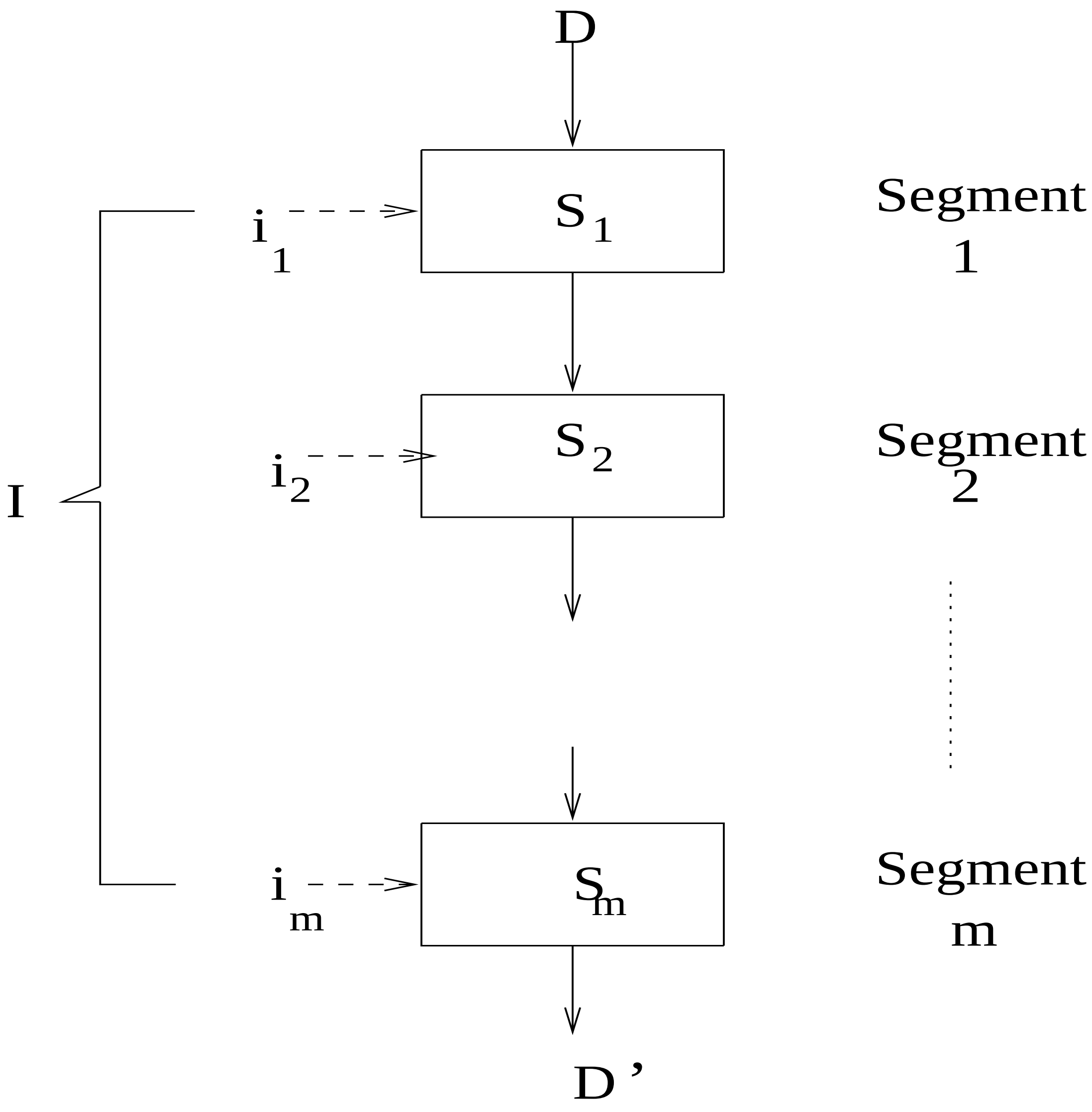

Figure 3: MISD Organization

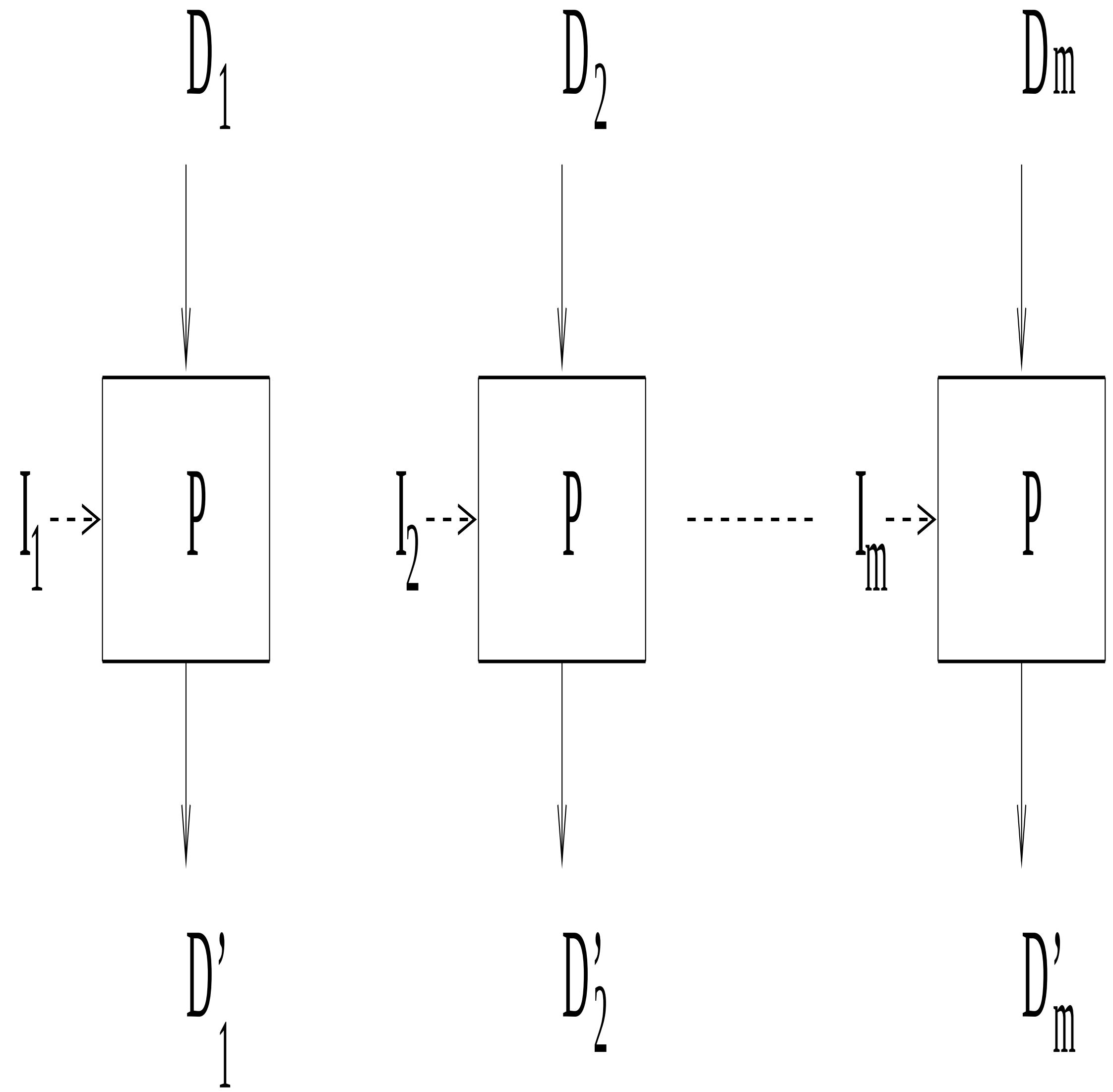

Figure 4: MIMD Organization

Pipelining

These are the basic types of pipelining,which appear in the pipeline processors:

1. Instruction pipeline
2. Arithmetic pipeline.

General Issues of parallel processing

Some other issues, that affect performance are:

1. Virtual Shared Memory
2. Memory Issues: Example given
3. Interconnection Issues
4. Message Passing

Example of Memory Management

The example itself concerns the dense matrix-matrix multiplication $C = AB + C$. The matrices are square and of order 1024. Assume a page contains 65,536 elements; then it follows that 64 columns of a matrix can be stored on 1 page and that all three matrices can be stored on 48 pages in total. Let us assume that the available main memory space can contain 16 pages. This implies that if we have access to successive columns of a matrix, we have a page fault after every 64 columns. Here it is also assumed that a page fault takes about 0.5 seconds of I/O time. Different ways to compute $C = AB + C$ are given below:

1. Inner-product approach - ijk

```
      DO 30 I = 1, N
         DO 20 J = 1, N
            DO 10 K = 1, N
               C(I,J) = A(I,K) * B(K,J) + C(I,J)
10          CONTINUE
```

```
 20       CONTINUE
 30    CONTINUE
```

It will 93 days for the I/O!!!!

2. Column-wise update of C - jki

```
       DO 30 J = 1, N
          DO 20 K = 1, N
             DO 10 I = 1, N
               C(I,J) = A(I,K) * B(K,J) + C(I,J)
 10          CONTINUE
 20       CONTINUE
 30    CONTINUE
```

It will 2.25 Hrs for the I/O!!

3. Partitioning the matrices into block-columns - blocked jki

```
       DO 40 J = 1, N, NB
          DO 30 K = 1, N, NB
             DO 20 JJ = J, J+NB-1
                DO 10 KK = K, K+NB-1
               C(:, JJ) = A(:, KK) * B(KK, JJ) + C(:, JJ)
 10             CONTINUE
 20          CONTINUE
 30       CONTINUE
 40    CONTINUE
```

It will 70 seconds for the I/O.

Implementation Details

1. Parallel Decomposition and Data Dependency Graphs
2. Synchronization
3. Load Balancing

Performance Consideration

1. Speedup
$$Sp(n) = \frac{T(l)}{T(n)} \tag{1}$$
2. Efficiency
$$E(n) = \frac{S(n)}{n} \tag{2}$$
3. Amdahl's Law
$$t = \frac{(1-f)N}{S} + \frac{fN}{V} \tag{3}$$
Here, f is the fractional work, which can be done in parallel
4. General form of Amdahl's Law
$$r = \frac{N}{\frac{N_1}{r_1} + \frac{N_2}{r_2} + ... + \frac{N_n}{r_n}} Mflops \tag{4}$$
5. Amdahl's Law for pineline processors
$$r_N = \frac{r_\infty}{n_{1/2}/N + 1} \tag{5}$$

6. Amdahl's Law for parallel processing

$$S_p = \frac{p}{(f + (1 - f)p)} \tag{6}$$

7. Gustafson's Model

$$S_{p,f} = p + (1 - p)(1 - f) \tag{7}$$

Linear Algorithm

1. Point Algorithm

for $j = 1$ step 1 until n

$$b \leftarrow L_j^{-1} P_j b;$$

end;

Solve $Ux = b$;

Linear Algorithms continued....

1. Right-Looking Algorithm

STEP 1:

$$PA = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & I & 0 \\ L_{31} & 0 & I \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} & U_{13} \\ 0 & \hat{A_{22}} & \hat{A_{23}} \\ 0 & \hat{A_{32}} & \hat{A_{33}} \end{bmatrix} \quad (8)$$

where L_{11} and U_{11} are kxk matrices and P is a permutation matrix representing the effects of pivoting. The blocks labeled $\hat{A_{ij}}$ are the updated portion of A and has not yet been factored, and will be referred to as the active sub-matrix.

STEP 2:

$$\begin{bmatrix} I & 0 \\ 0 & P_2 \end{bmatrix} PA = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & I \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} & U_{13} \\ 0 & U_{22} & U_{23} \\ 0 & 0 & \hat{A_{33}} \end{bmatrix} \quad (9)$$

where P_2 is the permutation matrix of the order $n - k$.

STEP 3:

$$P_2 \begin{bmatrix} \hat{A_{22}} \\ \hat{A_{32}} \end{bmatrix} = \begin{bmatrix} L_{22} \\ L_{32} \end{bmatrix} U_{22} \quad (10)$$

STEP 4:

$$\begin{bmatrix} \hat{A_{22}} \\ \hat{A_{32}} \end{bmatrix} \leftarrow P_2 \begin{bmatrix} \hat{A_{22}} \\ \hat{A_{32}} \end{bmatrix} \quad (11)$$

and

$$\begin{bmatrix} L_{21} \\ L_{31} \end{bmatrix} \leftarrow P_2 \begin{bmatrix} L_{21} \\ L_{31} \end{bmatrix} \quad (12)$$

STEP 5: Solve the triangular system

$$U_{23} = L_{22}^{-1} \hat{A_{23}} \tag{13}$$

to compute the next block row of U.
STEP 6: Update $\hat{A_{33}}$

$$\hat{A_{33}} \leftarrow \hat{A_{33}} - L_{32} U_{23} \tag{14}$$

STEP 7: *repeat* the STEP 2 - STEP 6, *until* $A = LU$

Linear Algorithms continued....

1. Left-Looking Algorithm

STEP 1: Assume that

$$PA = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & I & 0 \\ L_{31} & 0 & I \end{bmatrix} \begin{bmatrix} U_{11} & A_{12} & A_{13} \\ 0 & A_{22} & A_{23} \\ 0 & A_{32} & A_{33} \end{bmatrix} \tag{15}$$

STEP 2:

$$\begin{bmatrix} I & 0 \\ 0 & P_2 \end{bmatrix} PA = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & I \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} & A_{13} \\ 0 & U_{22} & A_{23} \\ 0 & 0 & A_{33} \end{bmatrix} \tag{16}$$

STEP 3: Solve the triangular system

$$U_{12} = L_{11}^{-1} A_{12} \tag{17}$$

STEP 4: Update the rest of the middle block column of U,

$$\begin{bmatrix} \hat{A_{22}} \\ \hat{A_{32}} \end{bmatrix} \leftarrow \begin{bmatrix} A_{22} \\ A_{32} \end{bmatrix} - \begin{bmatrix} L_{21} \\ L_{31} \end{bmatrix} U_{12} \tag{18}$$

STEP 5: Perform the factorization

$$P_2 \begin{bmatrix} \hat{A_{22}} \\ \hat{A_{32}} \end{bmatrix} = \begin{bmatrix} L_{22} \\ L_{32} \end{bmatrix} U_{22} \tag{19}$$

STEP 6: Lastly the pivoting

$$\begin{bmatrix} A_{23} \\ A_{33} \end{bmatrix} \leftarrow P_2 \begin{bmatrix} A_{23} \\ A_{33} \end{bmatrix} \tag{20}$$

and

$$\begin{bmatrix} L_{21} \\ L_{31} \end{bmatrix} \leftarrow P_2 \begin{bmatrix} L_{21} \\ L_{31} \end{bmatrix} \tag{21}$$

Linear Algorithms continued....

1. Crout Algorithm

STEP 1: Assume that

$$PA = \begin{bmatrix} L_{11} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} I & U_{12} & U_{13} \\ L_{21} & A_{22} & A_{23} \\ L_{31} & A_{32} & A_{33} \end{bmatrix}$$

$$\begin{bmatrix} U_{11} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \tag{22}$$

STEP 2: For factorization,solve

$$\begin{bmatrix} A_{22} & A_{23} \end{bmatrix} \leftarrow \begin{bmatrix} A_{22} & A_{23} \end{bmatrix} - L_{12} \begin{bmatrix} U_{12} & U_{13} \end{bmatrix} \tag{23}$$

and

$$A_{32} \leftarrow A_{32} - L_{31} U_{12} \tag{24}$$

STEP 3: Do factorization

$$P_2 \begin{bmatrix} A_{22} \\ A_{32} \end{bmatrix} = \begin{bmatrix} L_{22} \\ L_{32} \end{bmatrix} U_{22} \tag{25}$$

STEP 4: Replace

$$\begin{bmatrix} A_{23} \\ A_{33} \end{bmatrix} \leftarrow P_2 \begin{bmatrix} A_{23} \\ A_{33} \end{bmatrix} \tag{26}$$

and

$$\begin{bmatrix} L_{21} \\ L_{31} \end{bmatrix} \leftarrow P_2 \begin{bmatrix} L_{21} \\ L_{31} \end{bmatrix} \tag{27}$$

and put

$$U_{23} = L_{21}^{-1} A_{23} \tag{28}$$

STEP 5:

$$P_2 P A = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} I & 0 & U_{13} \\ 0 & I & U_{23} \\ L_{31} & L_{32} & A_{33} \end{bmatrix}$$

$$\begin{bmatrix} U_{11} & U_{12} & 0 \\ 0 & U_{22} & 0 \\ 0 & 0 & I \end{bmatrix} \tag{29}$$

STEP 6: *repeat* STEP 2 - STEP 5 *until* LU decomposition complete.

Future Work

- After discussing various aspects for the parallel processing and high performance computing, the future work will involve the following:
  - A detailed survey of computer architecture, specifically for pipelined processors,is to be done.
  - Since optimization of the code is of utmost importance, so the in-depth knowledge is to be gained in the field of compiler theory.
  - Message passing strategies need to be explored in a greater detail.
  - Linear Algebra algorithms for the uniprocessor computation and its extensions for the multiprocessor paradigm are to be devised.
  - There is a need for the general algorithms

to be made compatible to the multiprocessor systems.

– Algorithms have to be discovered or devised, which are best suitable for shared memory systems.