TECHNIQUES TO MAPPING HOMOGENEOUS PROCESSES IN HETEROGENEOUS ENVIRONMENTS
INTRODUCTION

- Until now, a lot of software has been developed for homogeneous systems.
- Now, heterogeneous systems.

- Two options
  - To rewrite software
  - Mapping routines in heterogeneous systems.

- Complex Problems
  - To Parallell

- To Parallell
  - To Estimate
  - To Build mathematical parameters
  - To build model execution time
Execution Time Estimation:

\[ t_{\text{exe}} = f(n, SP, AP) \]

- \( n \) = problem size
- \( SP \) = system parameters
- \( AP \) = algorithm parameters

Autooptimization studied in homogeneous
INTRODUCTION

- Heterogenous system algorithms

- HoHe
  - * 1 process/processor
  - * processes with different load work

- HeHo
  - * > 1 process/processor
  - * processes with equal load work

- HeHo better: no to rewrite routines
THE PROBLEM

- In parallel computing, assign processes to processors in order to obtain a mapping which reduces execution time.
- Efficient and optimal solutions for some particular problems.
- For other problems approximate solutions are obtained with heuristic methods.
- The assignation was studied for a parallel dynamic programming scheme and for dense linear algebra factorizations.
- The problem is solved by searching through an assignation tree.
THE PROBLEM

∀node ∃ \( t_{\text{modell}} \)

Mapping processes (1..n) to processors (1..P).
THE PROBLEM

- The search through the tree:
  - Backtracking
  - Backtracking by pruning nodes
  - Branch and Bound
  - Greedy methods
  - .....

- A large amount of time is consumed (Backtracking)

- Solution obtained far from the optimum (greedy method, backtracking by pruning nodes, ...)

THE PROBLEM

- We consider parallel iterative schemes.
- A large variety of problems: dynamic programming, graph algorithms, genetic algorithms…
- Additionally, the method can be applied to other problems and schemes, as for example linear algebra factorizations.
- The theoretical model requires a precise knowledge of system parameters like basic arithmetic operations time, or sending and start-up times of a communication.
Modelled execution time

\[ t(s,D) = t_c \cdot t_{\text{comp}}(s,D) + t_{\text{comm}}(s,D) \]

- \( s \): problem size
- \( D \): number of processes used in the solution
- \( t_c \): cost of a basic arithmetic operation
- \( t_{\text{comp}} \): number of basic arithmetic operations
- \( t_{\text{comm}} \): communication cost

- It is easy to represent computation part.
- It is frequently difficult to represent communication part: some adjust by least square.
POSSIBLE SOLUTIONS

- To use traditional techniques in trees to achieve a good solution, but this consumes a large amount of time.
  - Even with strategies to prune nodes in the tree.

- To use heuristic techniques successfully applied to solve other hard optimization problems, like Scatter Search.
  - **Scatter Search**: heuristic technique by which different generations of an initial population is possible to achieve a good solution of a problem.
Basic Scatter Search Scheme:

1. generate initial population
2. while convergence is not reached
3. element selection to be combined
4. combining of selected elements
5. inclusion of the most promising elements in the population
6. inclusion in the population of the most scattered elements with respect to the most promising elements
7. endwhile
HEURISTIC APPROACHES

- A large number of different schemes of scatter search in order to apply our problem.
HEURISTIC APPROACHES

- A large number of different schemes of scatter search in order to apply our problem.

  A.- Initial population

  Initial assignation of processes to processors randomly: AA
  Initial assignation of processes to processors considering number of processors: AC
HEURISTIC APPROACHES

✓ A large number of different schemes of scatter search in order to apply our problem.

A.- Initial population

B.- To select elements to combine

- Initial assignation of processes to processors randomly: AA
- Initial assignation of processes to processors considering number of processors: AC

- All elements of population vs all elements: TT
- Best elements of population vs worst elements: MP
HEURISTIC APPROACHES

C.- To include the most scattered respect to the most promising

To include elements with longest distance to most promising elements: MF
To include elements which are more “different” with respect to promising elements: MD
HEURISTIC APPROACHES

D.- Convergence reached?

C.- To include the most scattered respect to the most promising

To include elements with longest distance to most promising elements: **MF**
To include elements which are more “different” with respect to promising elements: **MD**

The best of the new solutions are not better than the best of the older ones: **MM**
The average of new solutions is not better than old ones: **MA**
RESULTS

- Best scheme with options of initial reference set: **AC** and convergence reached: **MA**.

DIFFERENT OPTIONS OF SCATTER SEARCH SCHEME VERSUS BACKTRACKING WITH PRUNING

<table>
<thead>
<tr>
<th></th>
<th>TT</th>
<th>MP</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>C.- Include</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MF</td>
<td>90%</td>
<td>85%</td>
</tr>
<tr>
<td>MD</td>
<td>91%</td>
<td>86%</td>
</tr>
</tbody>
</table>

- Best scheme with options: **TT-MD**
RESULTS

**REAL TIMES IN KIPLING**

<table>
<thead>
<tr>
<th>SIZE</th>
<th>COMPLEXITY</th>
<th>ANOTHER TECHNIQUES</th>
<th>SCATTER SEARCH</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMALL [1..100.000]</td>
<td>SMALL</td>
<td>82%</td>
<td>18%</td>
</tr>
<tr>
<td></td>
<td>BIG</td>
<td>80%</td>
<td>20%</td>
</tr>
<tr>
<td>MEDIUM [100.000..500.000]</td>
<td>SMALL</td>
<td>63%</td>
<td>37%</td>
</tr>
<tr>
<td></td>
<td>BIG</td>
<td>59%</td>
<td>41%</td>
</tr>
<tr>
<td>BIG [500.000…1.000.000]</td>
<td>SMALL</td>
<td>52%</td>
<td>48%</td>
</tr>
<tr>
<td></td>
<td>BIG</td>
<td>47%</td>
<td>53%</td>
</tr>
</tbody>
</table>

TABLE WITH COMPARISONS AMONG TRADITIONAL TECHNIQUES AND SCATTER SEARCH TO ESTIMATE THE BEST MAPPING OF PROCESSES TO PROCESSORS
RESULTS

SIMULATIONS TIMES

<table>
<thead>
<tr>
<th>SIZE</th>
<th>COMPLEXITY</th>
<th>ANOTHER TECHNIQUES</th>
<th>SCATTER SEARCH</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMALL [1..100.000]</td>
<td>SMALL</td>
<td>91%</td>
<td>9%</td>
</tr>
<tr>
<td></td>
<td>BIG</td>
<td>73%</td>
<td>27%</td>
</tr>
<tr>
<td>MEDIUM [100.000..500.000]</td>
<td>SMALL</td>
<td>42%</td>
<td>58%</td>
</tr>
<tr>
<td></td>
<td>BIG</td>
<td>40%</td>
<td>60%</td>
</tr>
<tr>
<td>BIG [500.000...1.000.000]</td>
<td>SMALL</td>
<td>22%</td>
<td>78%</td>
</tr>
<tr>
<td></td>
<td>BIG</td>
<td>5%</td>
<td>95%</td>
</tr>
</tbody>
</table>

TABLE WITH COMPARISONS AMONG TRADITIONAL TECHNIQUES AND SCATTER SEARCH TO ESTIMATE THE BEST MAPPING OF PROCESSES TO PROCESSORS
FUTURE WORKS

- Scatter search is a promising technique to mapping in heterogeneous systems.
- To apply another heuristic techniques, like tabu search, genetic algorithms, and to study applications to another parallel algorithms schemes.
- Our final goal is the software must decide how to mapping an homogeneous algorithm in an heterogeneous systems, depending on different algorithms and system parameters.