A Distributed Parallel Divide and Conquer Skeleton

J. R. González C. León C. Rodríguez *

Dpto. de Estadística, I.O. y Computación, Universidad de La Laguna,
E-38271 La Laguna, Canary Islands, Spain
email-(jrgonzal—cleon—casiano)@ull.es,
WWW home page: `http://nereida.deioc.ull.es`

Abstract

The MaLLBa library provides skeletons to solve combinatorial optimization problems. Its main objective is to simplify the implementation of algorithms based on some commonly used techniques such as Branch and Bound, Dynamic Programming or Divide and Conquer. This work is focused on the MaLLBa::DnC skeleton, which solves problems that fit in the Divide and Conquer paradigm. The user has to provide functions particularized to the problem he wants to solve. The MaLLBa::DnC skeleton needs a 'divide()' function which divides one problem into smaller subproblems. Also a 'combine()' function that combines partial solutions into a solution to its parent subproblem. To stop the partition process an 'easy()' function have to be provided which indicates when a problem is easy enough to be solved using the 'solve()' function for simple subproblems. Given that functions the skeleton encapsulates all remaining work and allows the problem to be solved either in a sequential or parallel way without bothering the user to write extra code.

In this work we will present a new MPI asynchronous peer-processor implementation of the MaLLBa:DnC skeleton where all processors are peers and behave the same way (except during the initialization phase) and where decisions are taken based only on local information. We will present the promising results obtained on a Linux cluster of PC for several well-known Divide and Conquer algorithms such as quicksort, mergesort, fft, convex hull and the Strassen algorithm to multiply matrices.

*This work has been supported by the EC (FEDER) and by the Spanish Ministry of Science and Technology inside the "Plan Nacional de I+D+I" with contract number TIC2002-04498-C05-05.