

Automatic Parametrization on Divide-and-Conquer Algorithms¹

Murilo Do Carmo-Boratto^{*,a}, Domingo Giménez^a, Antonio M. Vidal^b

^a *Departamento de Informática y Sistemas, Universidad de Murcia, murilo@aomail.uab.es, Spain*

^b *Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, Spain*

Automatic tuning techniques have been used in the design of routines in recent years. The goal is to develop routines which automatically adapt to the conditions of the computational system, in such a way that efficient executions are obtained independently of the experience of the user. Techniques have been developed in different fields, and especially in linear algebra routines [1, 2, 3, 4].

In this work the possibility of applying automatic optimization techniques to divide-and-conquer algorithms is analyzed. The routines are developed together with their theoretical execution time, $t(s) = f(s, AP, SP)$, where s represents the problem size, SP are system parameters, and AP are algorithmic parameters. As an example, in a parallel mergesort algorithm system parameters are the costs of the basic arithmetic or communication operations, and algorithmic parameters could be: the level of recursion, the direct method used to solve the problem in the basic case, and the number of processors to use.

Different methods to estimate the value of the system parameters at installation time are analyzed. At execution time, the value of the algorithmic parameters with which the problem is solved are obtained by estimating the values which provide the lowest modelled execution time for the values of the system parameters obtained in the installation.

1. E.A. Brewer, Portable High-Performance Supercomputing: High Level Platform Dependent Optimization, Ph.D. thesis, Massachusetts Institute of Technology, 1994.
2. Z. Chen, J. Dongarra, P. Luszczek and K. Roche, Self Adapting Software for Numerical Linear Algebra and LAPACK for Clusters, *Parallel Computing* **29** (11-12) (2003) 1723-1743.
3. J. Cuenca, D. Giménez and J. González, Architecture of an Automatic Tuned Linear Algebra Library, *Parallel Computing* **30** (2) (2004) 187-220.
4. T. Katagiri, K. Kise, H. Honda and T. Yuba, ABCLibScript: a directive to support specification of an auto-tuning facility for numerical software, *Parallel Computing* **32** (2006) 92-112.

¹This work has been funded in part by the Spanish MCYT and FEDER under Grant TIC2003-08238-C02-02