Parallel Skeleton for Multi-Objective Optimization *

Coromoto León
Dpto. Estadistica, I. O. y Computación
Universidad de La Laguna
38271, Tenerife, Spain
cleon@ull.es

Gara Miranda
Dpto. Estadistica, I. O. y Computación
Universidad de La Laguna
38271, Tenerife, Spain
gmiranda@ull.es

Carlos Segura
Dpto. Estadistica, I. O. y Computación
Universidad de La Laguna
38271, Tenerife, Spain
csegura@ull.es

ABSTRACT

Many real-world problems are based on the optimization of more than one objective function. This work presents a tool for the resolution of multi-objective optimization problems based on the cooperation of a set of algorithms. The invested time in the resolution is decreased by means of a parallel implementation of an evolutionary team algorithm. This model keeps the advantages of heterogeneous island models but also allows to assign more computational resources to the algorithms with better expectations. The elitist scheme applied aims to improve the results obtained with single executions of independent evolutionary algorithms. The user solves the problem without the need of knowing the internal operation details of the used evolutionary algorithms. The computational results obtained on a cluster of PCs for some tests available in the literature are presented.

Categories and Subject Descriptors

I.2.8 [**Computing Methodologies**]: Artificial Inteligence—*Problem Solving, Control Methods and Search* Heuristic Methods; D.1.3 [**Software Engineering**]: Programming Techniques—*Concurrent Programming* Parallel Programming

General Terms

Evolutionary Multi-objective Optimization

Keywords

Evolution Strategies, Multi-objective Optimization, Parallelization, Software Tools

1. PARALLEL SCHEME

The library provides a C++ implementation of a selection of the literature best-known evolutionary algorithms introducing the novelty of the algorithms cooperation. The user interface makes possible to properly configure the tool for solving a particular problem in an easy and intuitive way [1].

*This work has been supported by the EC (FEDER) and by the Spanish Ministry of Education and Science inside the 'Plan Nacional de I+D+i' with contract number TIN2005-08818-C04-04. The work of G. Miranda has been developed under the grant FPU-AP2004-2290.

GECCO'07, July 7–11, 2007, London, England, United Kingdom.
ACM 978-1-59593-697-4/07/0007.

The team algorithm model consists of a *coordinator* process and as many *islands* or *slaves* processes as specified by the user. The coordinator starts on every idle slave the execution of an algorithm configuration and manages the global solution. Slaves execute different algorithm instances, maintaining partial solutions in their *local Pareto front*. When an island execution finishes the local Pareto front is unified into the *global Pareto front*. A migration scheme allows slave processes to share part of their local solutions. Considering the obtained results the coordinator gives more opportunities to the algorithms with better expectations.

Figure 1 shows the average errors achieved with the ZDT tests. ZDT tests were solved with the single evolutionary algorithms: NSGA2, SPEA2 and SPEA fixing the number of evaluations to 50,000. Execution results were merged using the NSGA2 crowding operator, labelling this results as MERGE. Also, each test problem was solved using a team algorithm constituted by three islands. The number of evaluations with this method was fixed to 150,000, that is, the sum of the evaluations executed by the individual algorithms. The solutions obtained by this method were named TEAM. The experimental studies conclude that the parallel cooperation between algorithms leads to a considerable improvement of the results quality.

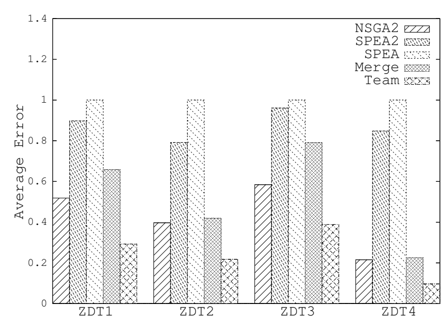

Figure 1: Average Error in ZDT Problems

2. REFERENCES

[1] C. León, G. Miranda, and C. Segura, ULL::A-Team. A Parallel Tool for Multi-Objective Optimization based on an Evolutionary Team Algorithm, DEIOC-07-01, Dpto. Estadística, I.O. y Computación, Universidad de La Laguna, April 2007, http://www.oplink.ull.es