Solving Optimization Problems with Grid-Enabled Technologies

by Enrique Alba and Antonio J. Nebro

Research in the 'Networking and Emerging Optimization Research Line' at the University of Málaga aims at solving optimization problems through the utilization of grid–enabled technologies and large computer networks.

Grid computing is a recent discipline related to the utilization of large-scale distributed systems for a given purpose by taking advantage of the rich infrastructure provided by the Internet. To understand the basic idea that motivates our research, consider a grid of computers as a huge virtual multicomputer ready for processing, storage and communication. Since a grid can be made up of a set of geographically separate networks, enormous computer power is available for solving complex problems that are limited in CPU and that require long delays if solved in modern computer LANs.

Complex problems that can only be solved in non-polynomial time arise in most fields of research and are becoming common in many areas of our lives: telecommunications, economy, bioinformatics, industrial environments,and so on. For such a wide spectrum of problems, heuristics come to the rescue, since exact techniques are unable to locate any kind of solution. This area is known as Networking and Emerging Optimization (NEO), and the GISUM group at the University of Málaga (Spain) is working on just these techniques.

In short, using Grid technology, our aim is to solve optimization problems that are otherwise out of the scope of researchers dealing with parallel algorithms. Five types of applications are usually identified as related to Grid computing: distributed supercomputing, high-throughput computing, on-demand computing, data-intensive computing and collaborative computing. At this stage of our research we are primary involved with the two first topics, ie distributed and high-throughput computing. However, we plan to enter the other three domains by developing an open optimization service for the Internet, solving data-mining problems and facing software agent applications respectively. Figure 1 is an example of the goal of the present work: computing an exact Pareto front.

$$
\begin{aligned}
\text{Minimize} \quad F &= (f_1(x,y), f_2(x,y)) \\
f_1(x,y) &= 4x^2 + 4y^2 \\
f_2(x,y) &= (x-5)^2 + (y-5)^2 \\
0 &\geq (x-5)^2 + y^2 - 25 \\
0 &\geq -(x-8)^2 - (y+3)^2 + 7.7 \\
0 \leq\ & x \leq 5 \\
0 \leq\ & y \leq 3
\end{aligned}
$$

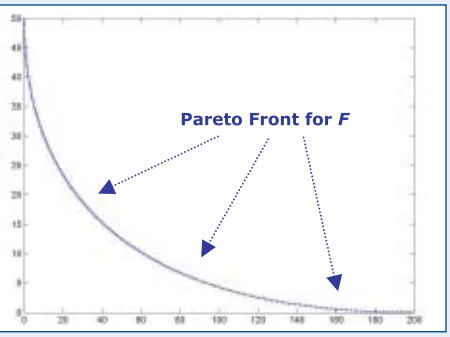

Figure 1: A simple example of a multi-objective optimization problem with the constraints (left), and its Pareto front (right).

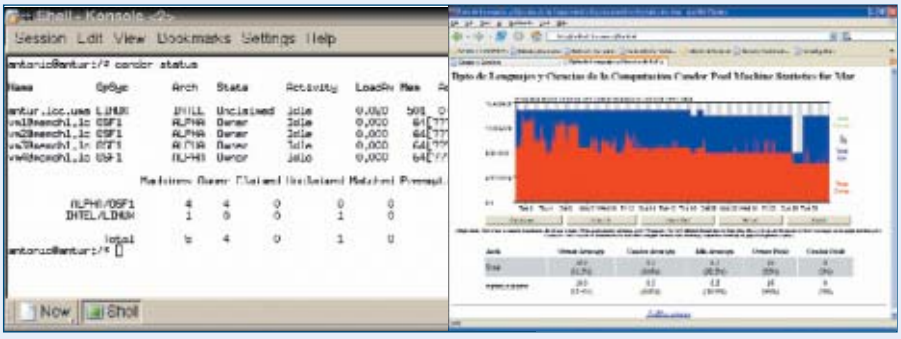

Figure 2: Tracking and managing the optimization algorithms in the grid with Condor.

Multi-criterion optimization in which several (non-dominated) optimum solutions must be found is a promising research field. At present, this field lacks algorithms that could ensure the computation of the exact Pareto front for a general problem. We avoid this inconvenience by using an enumerative-like search that computes all the non-dominated solutions in a grid. Later, researchers can use these results to find efficient heuristics achieving similar (optimal) results. We are in the first stage of research, and have solved problems at Málaga using a modest grid of around 110 processors. As simple as it sounds, finding the optimum Pareto front is extremely hard, even for small problems, and with most algorithms it is not guaranteed that the optimum front will be located for arbitrary problems as we do with our grid exact algorithm.

We are evaluating the performance of several grid-enabled technologies for our applications. Concretely, we have tested Globus, Condor, Legion, and Sun Grid Engine. Of these, the first two seem the most suitable for optimization in the grid. The simplicity and powerful process management of Condor were greatly appreciated in setting up a grid from scratch within a few weeks. Globus is in this sense more complex to use,

because users must deal with certificates, installation details and production tools that require a time-consuming learning curve. However, Globus actually eases the next step of our research in connecting to foreign grids. In addition, we have tested the network usage and execution times of heuristic algorithms and tools like MPI on Globus with very satisfactory results, and this suggests that we could achieve fruitful research with Globus in the future.

Most of this work has been undertaken within the on-going TRACER project (http://tracer.lcc.uma.es), which is developing advanced algorithms in close connection with the Internet. Two successful case studies in TRACER relating to Grid computing can be cited. Firstly, a grid algorithm was developed to compute the exact Pareto front of several hard multi-objective problems by using Condor in a grid with more than 110 computers (http://neo.lcc.uma.es/Software/ESaM). At present, we are extending the grid to other sites in Spain and Europe. Secondly, grid-aware multi-objective heuristics, initially run on Globus, were developed. We are currently testing new grid algorithms derived from well-known standard techniques like PAES or NSGA II.

We invite everyone to look at our results and contact us with any comments or suggestions for collaboration.

Link:
http://neo.lcc.uma.es

Please contact:
Enrique Alba, University of Málaga/
SpaRCIM, Spain
Tel: +34 952 132803
E-mail: eat@lcc.uma.es

Bioinformatics in the Semantic Grid

by Kai Kumpf and Martin Hofmann

One of the major challenges in Grid computing is the semantics-driven retrieval of Grid services and distributed data. DB-Annotator is an annotation tool for data in Grid services developed in a collaborative project between the bioinformatics department and the department for Web applications at the Fraunhofer Institute for Scientific Computing (SCAI). Fully annotated data in the Grid is particularly important in bioinformatics.

DB-Annotator was conceived for the Resource Description Framework (RDF) annotation of structured information sources such as relational data or XML-based service descriptions. Only semantically annotated Grid services (GS) provide a means of finding data or compute-services that are suitable for the task at hand. Furthermore, they make the building of workflows (coupled services within the grid) a realistic goal. DB-Annotator will support several levels of semantic annotation to enrich Grid services, ranging from the services themselves to the data within the Grid.

The project DB-Annotator (see Figure 2) was designed for universal annotation of data that can be represented in tabular form, ie data that can be qualified by unique keys.

The main thrust for the development came from the realization that most interesting data within biological databases resides within free-form text fields and is therefore not easily accesible for data mining. Relational databases (RDB) are usually the only choice for data retrieval within distributed organizations. Categorial description of the typed data residing in RDBs can be derived from the tables (entities) themselves, and from the columns (entity attributes) and the cell contents (instances of attributes). Still more fine-grained information comes from the notorious 'description' fields, but putting this implicit information to use for machine-reasoning is unrealistic at this point.

The goal was thus to provide easy navigation through existing ontologies and data-source-independent RDF annotation of RDB data via drag and drop (see Figure 2). This data categorization can

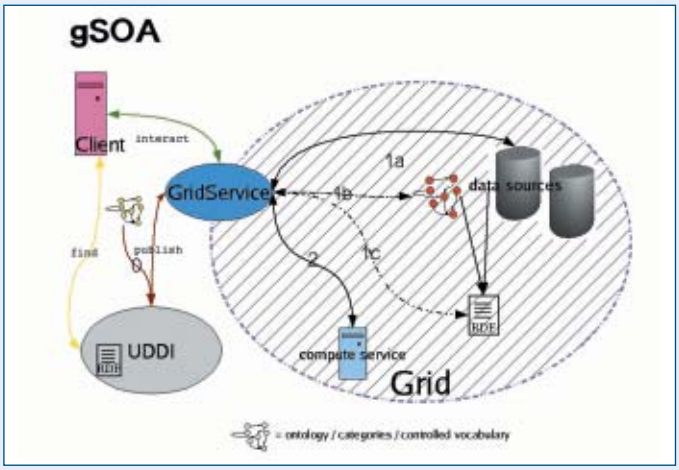

Figure 1: Grid Service- Oriented Architecture. The multiple levels of semantic annotation within a Grid-Service Oriented Architecture (gSOA). Path 0 corresponds to the semantic enhancement of the UDDI service registry via RDF. If no additional semantic glue is needed, direct access to either data- or compute-services is provided (1a, 2). When retrieving data, the semantic description can either stem from UDDI or, more finely -grained, from semantic annotation of structured (most often relational) data. 1b/c correspond to retrieval of annotated data; both are equivalent when there exists a central RDF annotation repository that stores n:m ontology-class -– (data) - relationships. Whereas 1a provides a complete view ofn the data via full-text search or keys, 1b/c allow querying by content.