stellarator news

Published by Fusion Energy Division, Oak Ridge National Laboratory
Building 5700 P.O. Box 2008 Oak Ridge, TN 37831-6169, USA

Editor: James A. Rome Issue 123 December 2009
E-Mail: jar@ornl.gov Phone (865) 482-5643
On the Web at http://www.ornl.gov/sci/fed/stelnews

Optimization of stellarators using metaheuristics and grid computing

Stellarator optimization requires many different executions of an equilibrium code such as VMEC and the applications that estimate the target functions. Often, it is very difficult to get the best configuration because of the huge parameter space. Trying to explore this vast solution space by using brute force algorithms or by specification of all of the different configurations becomes an impossible task. In the case of optimization problems, the use of metaheuristics such as genetic or evolutionary algorithms has been traditionally considered one of the best solutions. Nevertheless, stellarator optimization using these kind of tools requires huge computational resources; grid computing technologies could be a good solution for this task. The different computations that must be performed in order to evaluate the selected individuals, i.e., the selected stellarator configurations, can be executed in parallel in the computational resources of the infrastructure, and the management of the algorithm can be centralized in a single machine in a master-slave process.

In our case and as a starting point, we focused on optimization of the neoclassical transport of particles by estimating the average $\boldsymbol{B} \times \nabla B$ drift and searching for configuration equilibria that minimize this quantity (see Fig. 1). We are using VMEC to estimate the magnetic configuration and different metaheuristics to perform this optimization. In all our cases, an individual is a magnetic configuration (an input for VMEC, that is, an execution of the code in a remote resource of the grid); a chromosome is a single configuration parameter of the input (the Fourier harmonics of the magnetic field), and an individual is based on several chromosomes. The population consists of all individuals that are being evaluated at a giving moment. We have developed three different types of algorithms.

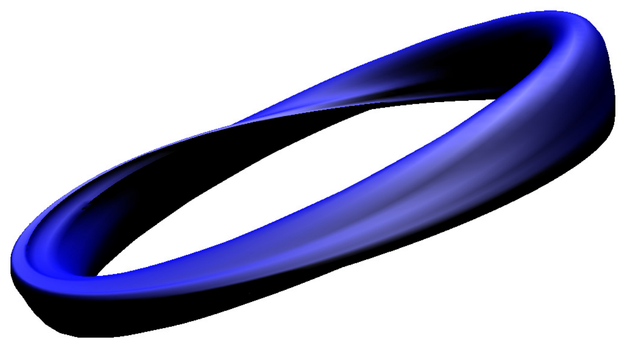

Fig. 1. Example of a two-period stellarator optimized with the condition of minimum average drift.

- **Genetic Algorithms**: These algorithms imitate evolution strategies such as crossover or mutation of chromosomes to create new individuals in a population. They are iterative models that require a large population (a few thousand individuals) to obtain good results. The wall clock time for an optimization process using this technique can be as much as one month, representing more than one year of CPU time.
- **Scatter Search**: Scatter search (SS) is a metaheuristic process using formulations that date back to the 1960s

In this issue . . .

for combining decision rules and problem constraints. SS works over a set of solutions, combining them to get new ones that improve the original set. In contrast to other evolutionary methods, such as genetic algorithms, SS is based not on large random populations but on strategic selections among small populations. SS also follows an iterative model that uses a small population and several improvement processes over a set of reference solutions. It is based on the idea of keeping a good balance between convergence and dispersion to find approximate solutions.

- **Artificial Bee Colony Algorithm**: This optimization process is modeled on behavior observed in bee colonies: "rover" bees that find a desirable food source return to the colony and communicate the location of the food source by "dancing." Once the information is exchanged, more bees are allocated to similar "flowers" (see Fig. 2). In other words, some processes search for approximate solutions for a given problem; and, when an optimized solution is found, more processes are allocated to look for approximate solutions using the new optimized one as a base element to explore. The exchange of information is, in fact, a master process managing the overall execution. It is not based on the idea of iteration and, although it is more difficult to develop and deploy, it makes better use of grid resources and offers the fastest results of the different metaheuristics.

All these algorithms have been developed using a generic schema, so any other optimization process suitable for execution on the grid can be solved by means of these techniques without substantial effort. Also, the scientist doesn't need to know many things about the technology being used, since these algorithms deal with the challenges that a distributed infrastructure can present.

All three types of algorithm are suitable to be run on the grid. Grid computing offers a large number of computational resources and a distributed paradigm that creates a perfect environment for large simulations where the communication requirements between the different elements involved in the simulation are low. These computational resources, usually devoted to scientific calculations, offer a high quality of service when in production state, with large storage capability, and are easily accessible. They create an attractive venue where scientists can carry out their simulations. Grid computing is, therefore, especially appropriate to run serial codes that can be executed in parallel, with different data, to get different results. This is the case of the stellarator optimization problem.

The next step is to introduce more target functions to be optimized, beyond just neoclassical transport. Mercier and ballooning stability criteria are the next optimization functions that will be considered in our algorithm. The Mercier criterion is estimated by the latest version of VMEC, which is already running in the grid, while calculating ballooning stability criterion requires porting the COBRA code to the grid. This task is now under way.

In this way we are able to optimize any given configuration by taking into account these three criteria for the moment. The inclusion of further optimization criteria can be done easily in this kind of algorithm.

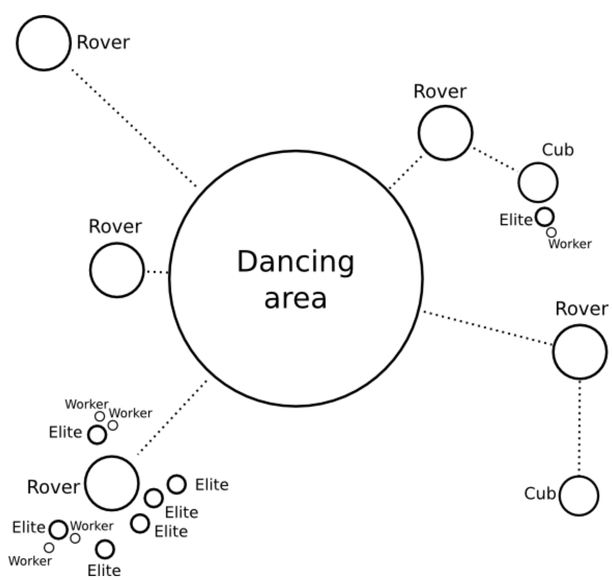

Fig. 2. Distribution in the parameter space of several types of bees and view of their explored areas.

Antonio Gómez and Francisco Castejón
Laboratorio Nacional de Fusión
Asociación Euratom/Ciemat para Fusión
E-mail antonio.gomez@ciemat.es