The Optimus optimization package

This projects proposes a programming tool written in ANSI C+ for global optimization problems. The main components of the software are: a) Coding of the objective problem in a high level language such as ANSI C++ b) Incorporation of many global optimization techniques to tackle the objective problem c)Parameterization of global optimization methods using user-defined parameters.

Requirements

1. A C++ compiler (GNU GCC is recommended).
2. The GNU make utility
3. The OpenMP programming library
4. The QT programming library.

Linux/Openbsd installation

The steps to install the software in most Unix systems are

1. Download the Qt library (usually using the package manager of the distribution)
2. Unzip the software: guznip GlobalOptimus-master.zip
3. Issue the command: cd GlobalOptimus-master
4. Issue the command: qmake (or qmake-qt5 in some systems)
5. Issue the command: make

Windows installation

On Windows a simple solution for the installation could be the following steps:

1. Install the software installer Choco
2. Issue the command: choco install make
3. Issue the command: choco install mingw

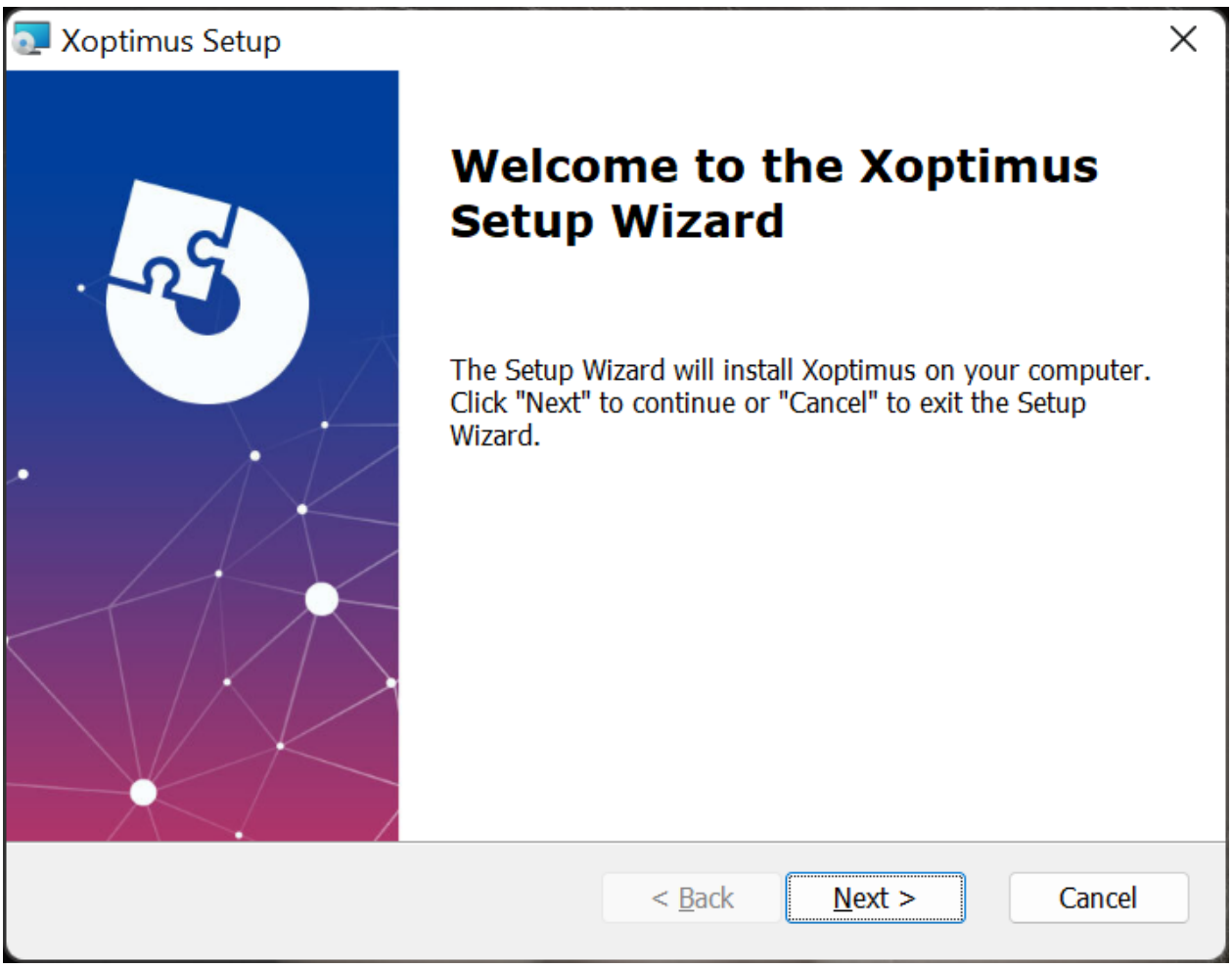
4. Install the Qt library
5. Unzip the software
6. Issue the command cd GlobalOptimus-master
7. Issue the command qmake
8. Issue the command make

Alternatively, in Windows the user can use the added Setup msi executable and issue the following:

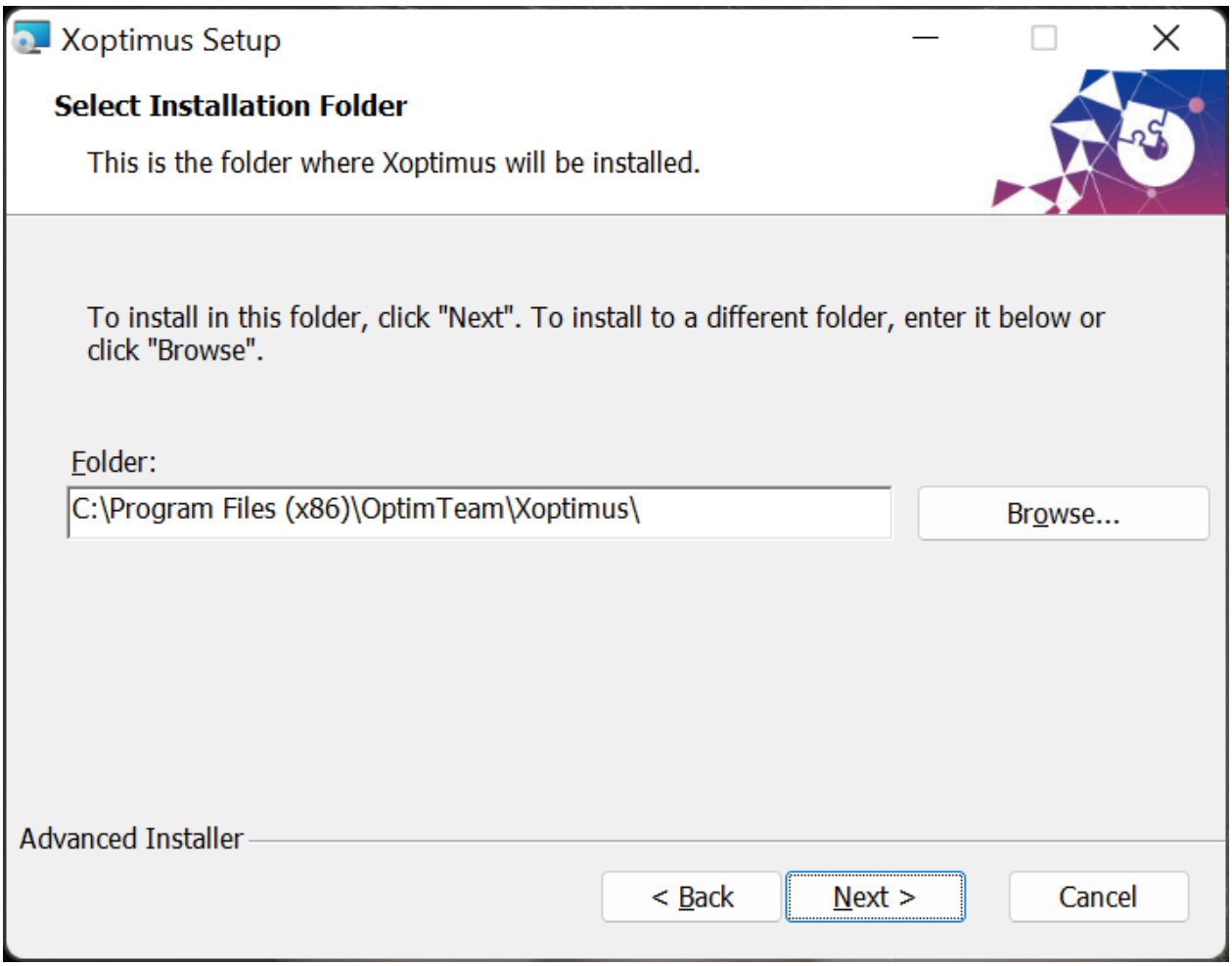
1. Install the package
2. Edit the 'settings' shortcut on the desktop and adjust the optimizer according to the comments
3. Open the command prompt (cmd) or PowerShell and to optimize, you can write the objective function you desire. For example, you can type 'runfunmin camel' or 'runfunmin potential 10' and so on.

The steps of the installation are shown in Figure 1. The user needs to select only the desired installation directory and the installer copies the necessary files to this directory.

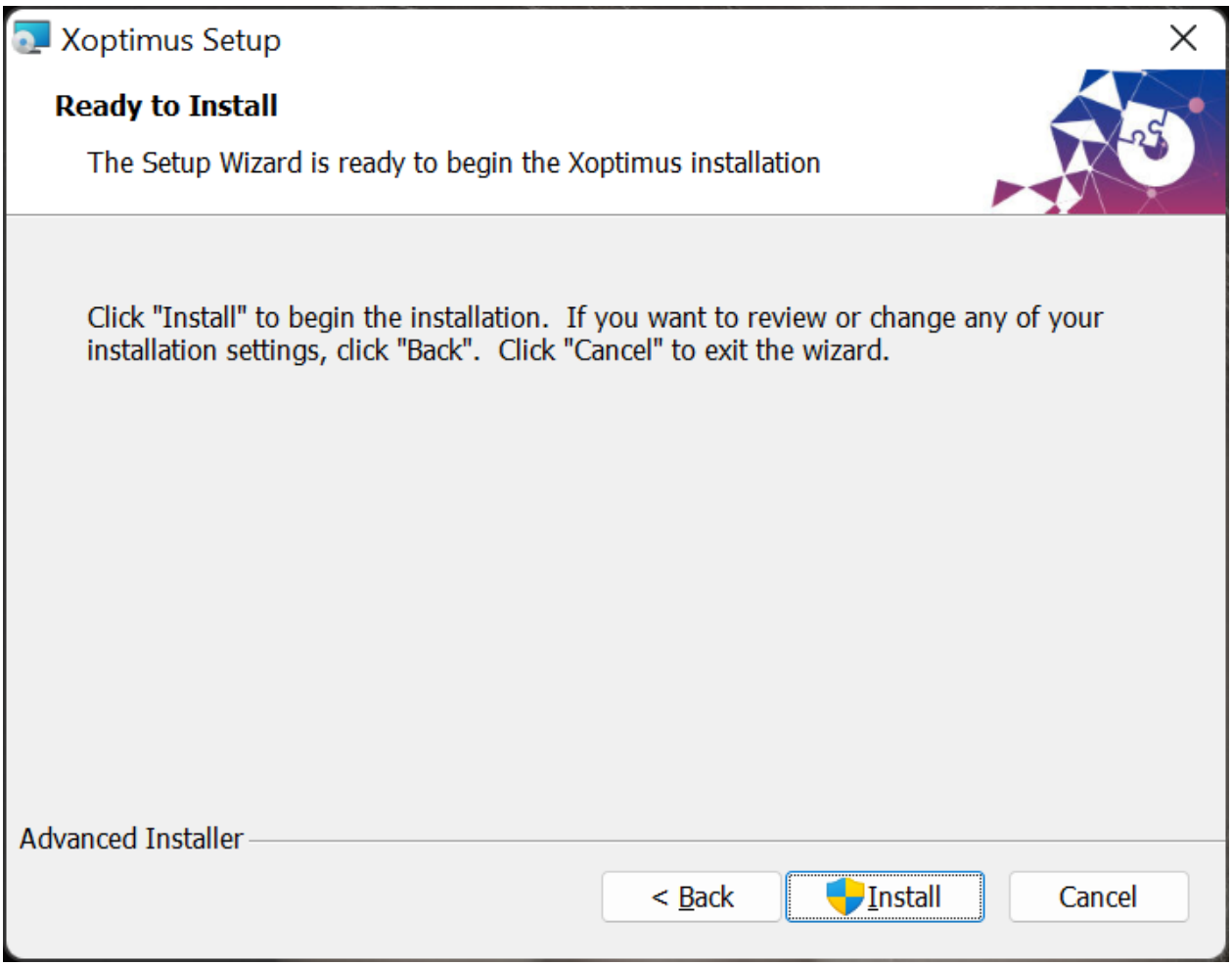
(a) First screen of the windows installation wizard.



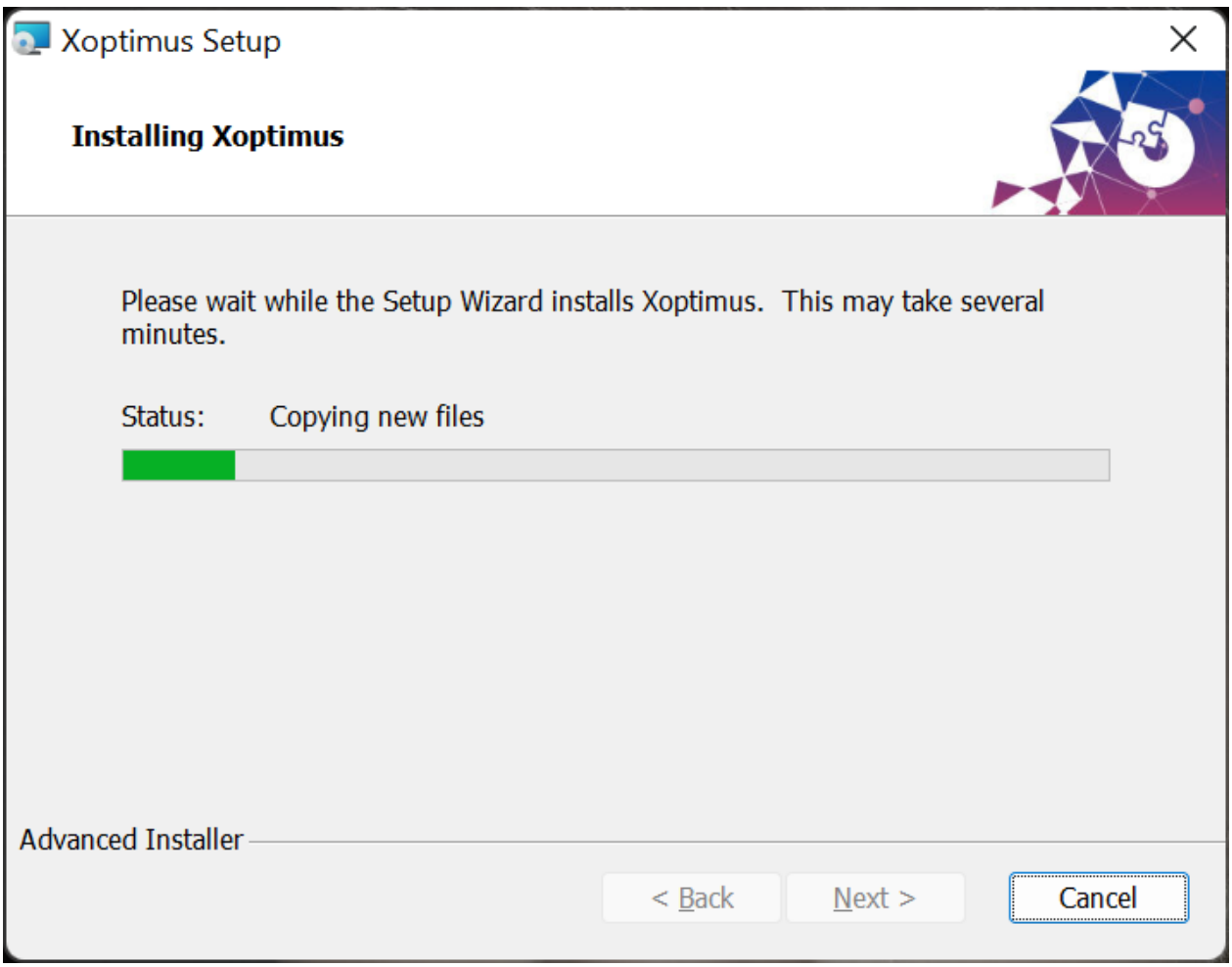
(b) The user selects the desired installation directory



(c) Prompt to install.



(d) Copying files.



(e) Finalizing installation.

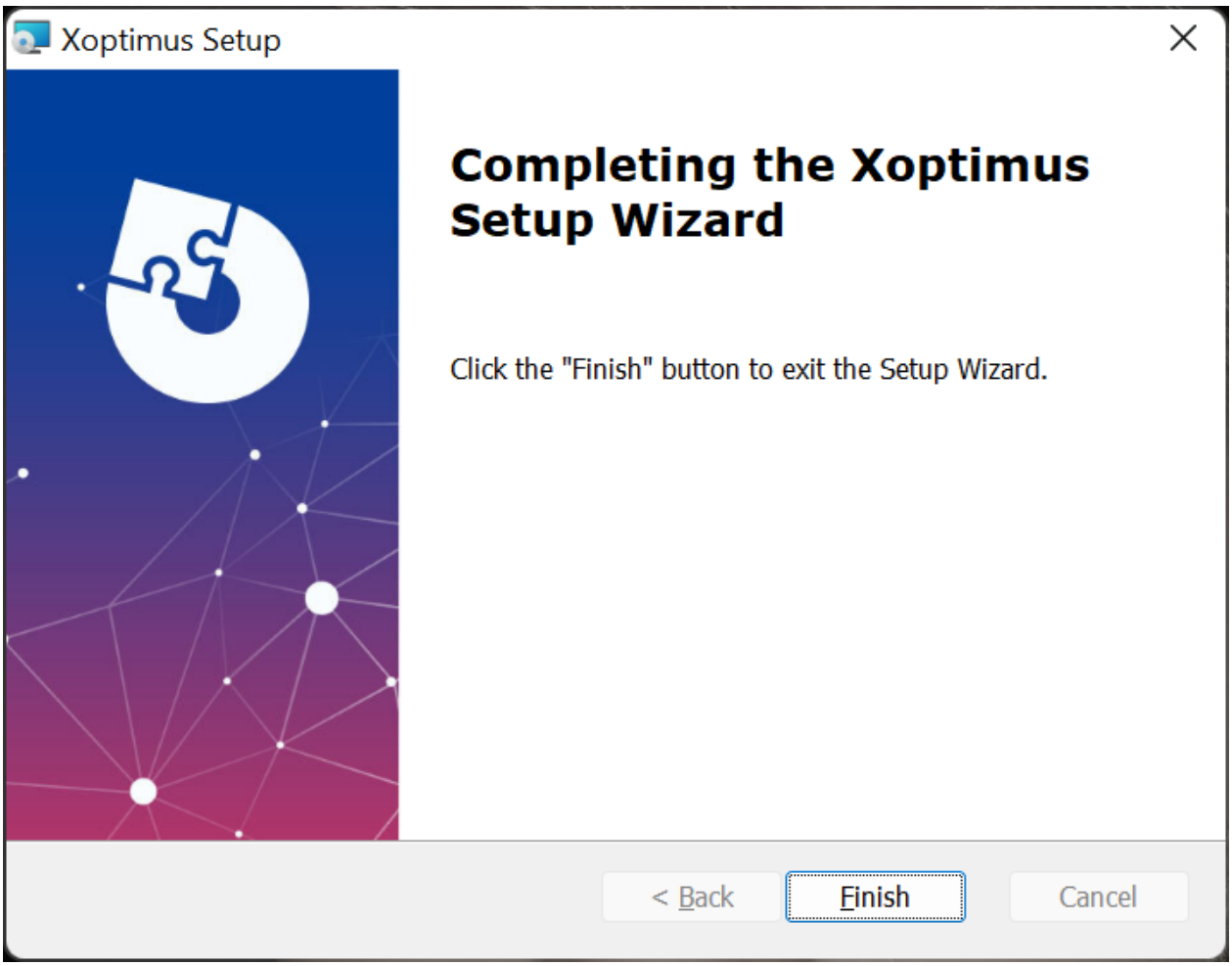


Figure 1. The steps of windows installation

The GUI application

The software also provides a graphical interface with which the user can easily select an objective function, optimization method and modify them by changing a number of available parameters. The software is called Xoptimus and the user can compile this program using the commands:

1. qmake Xoptimus.pro
2. make

Also, in Windows environments the Xoptimus.msi package will install this program. After invoking the program the first screen that is appeared is shown in Figure 6.

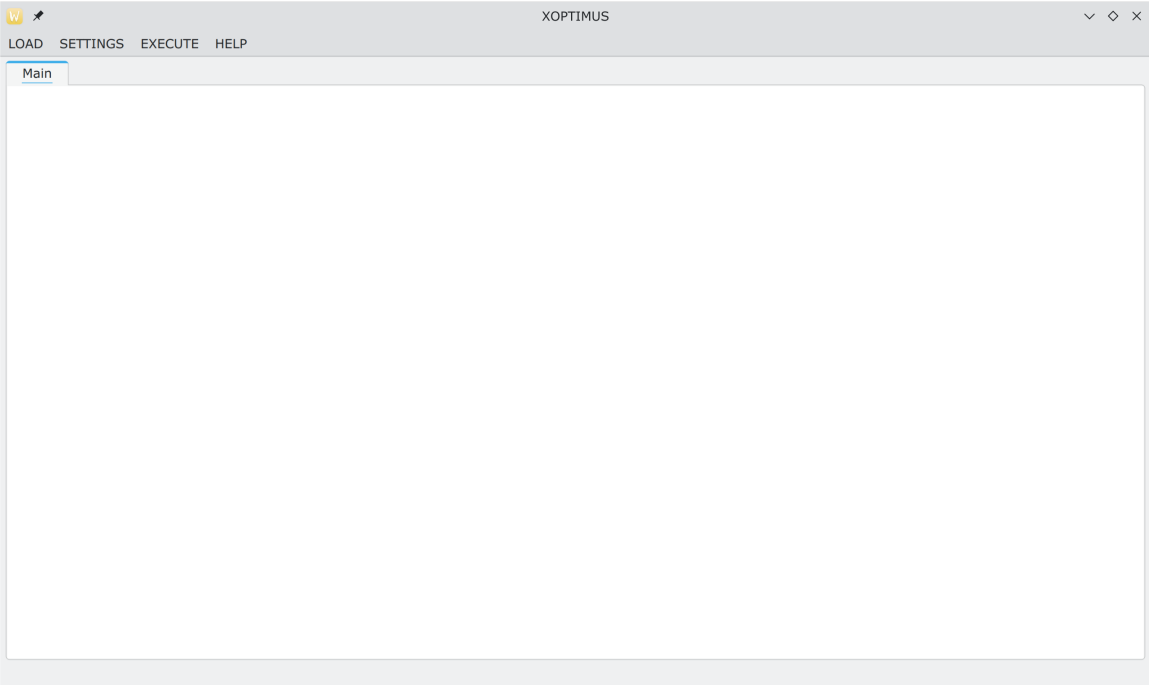


Figure 6. The start screen of Xoptimus program.

The user should select first the LOAD menu in order to define the objective problem and the desired global optimization method that should be used. The screen of Figure 7 is shown when the user select the option PROBLEM from the LOAD menu.

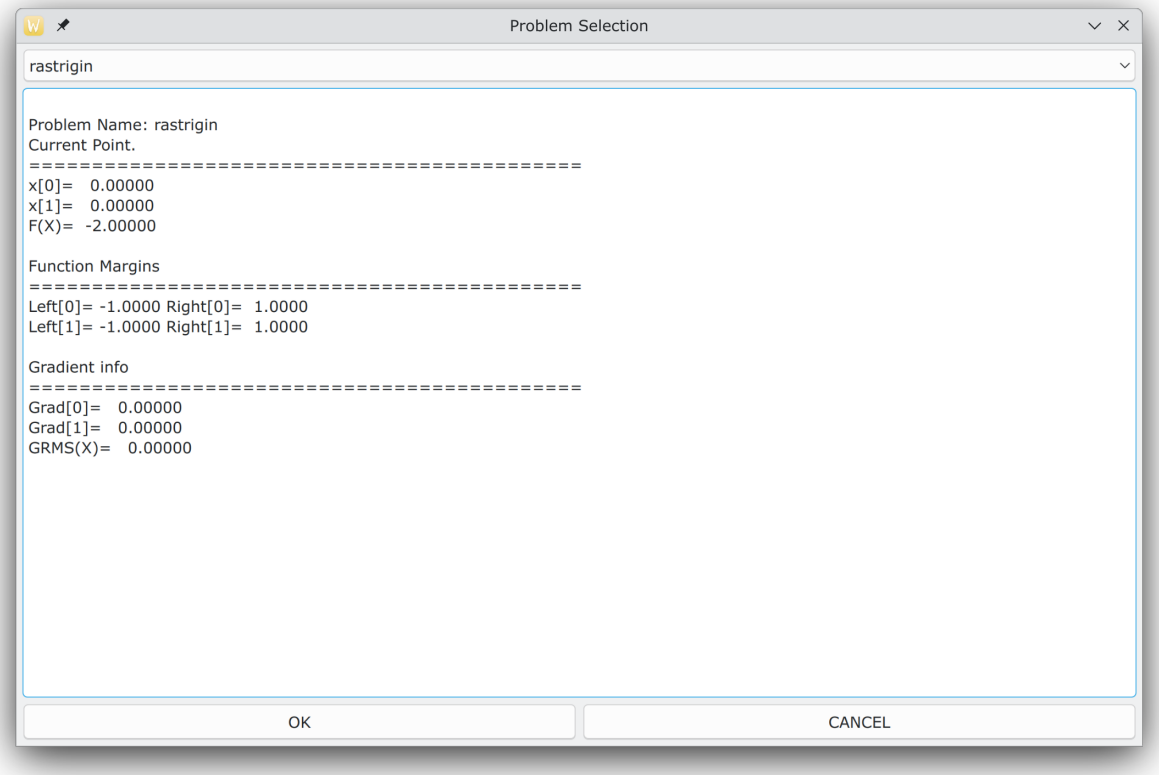


Figure 7. Selecting an optimization problem.

The user can select from a variety of offered objective functions from the dropdown menu. For each selected function the bounds are shown in the screen. Having selected the objective problem, the user should also selects the desired global optimization method from the option METHOD of the LOAD menu. This choice shows the dialog of Figure 8 in the screen.

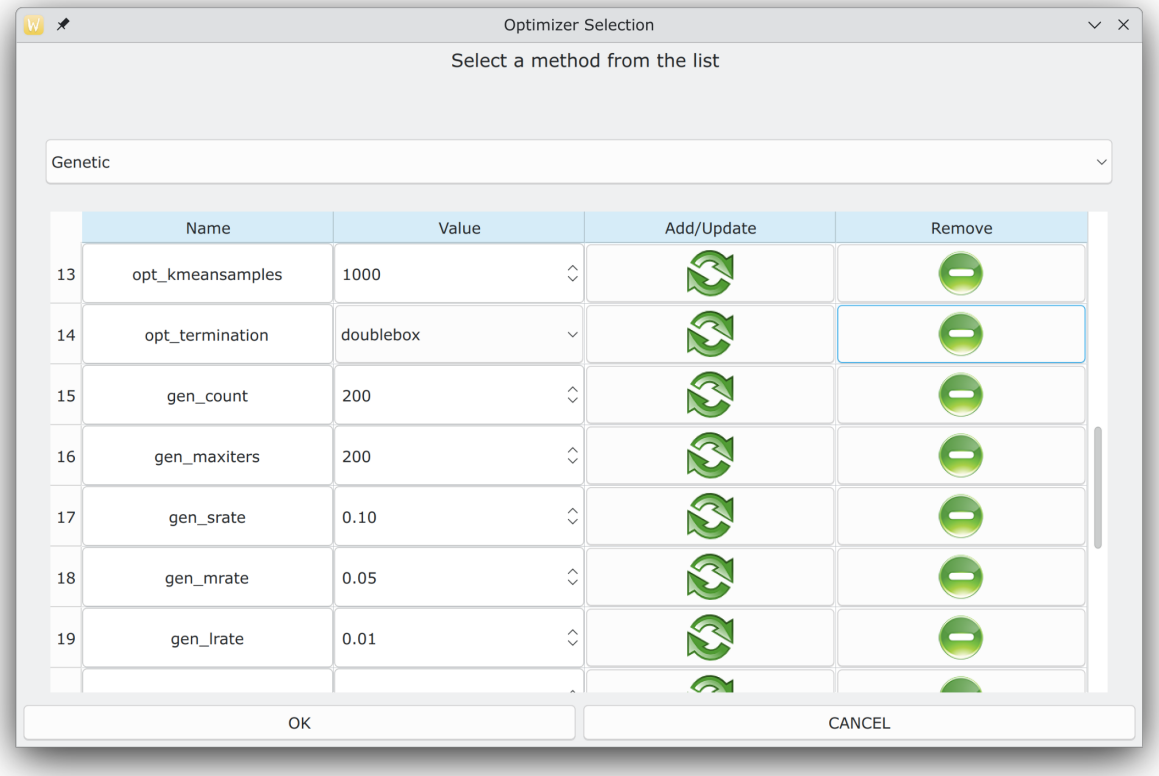


Figure 8. Selecting an optimization method.

From this dialog, the user can select the global optimization method from a dropdown menu and the user can also modify some of the parameters of the method, such as the number of chromosomes in the Genetic algorithm. Having selected the function and the desired global optimization method, the user can start the optimization process by invoking the option RUN from the menu EXECUTE. The selected global optimization method initiates and an example of the execution is shown in Figure 9.

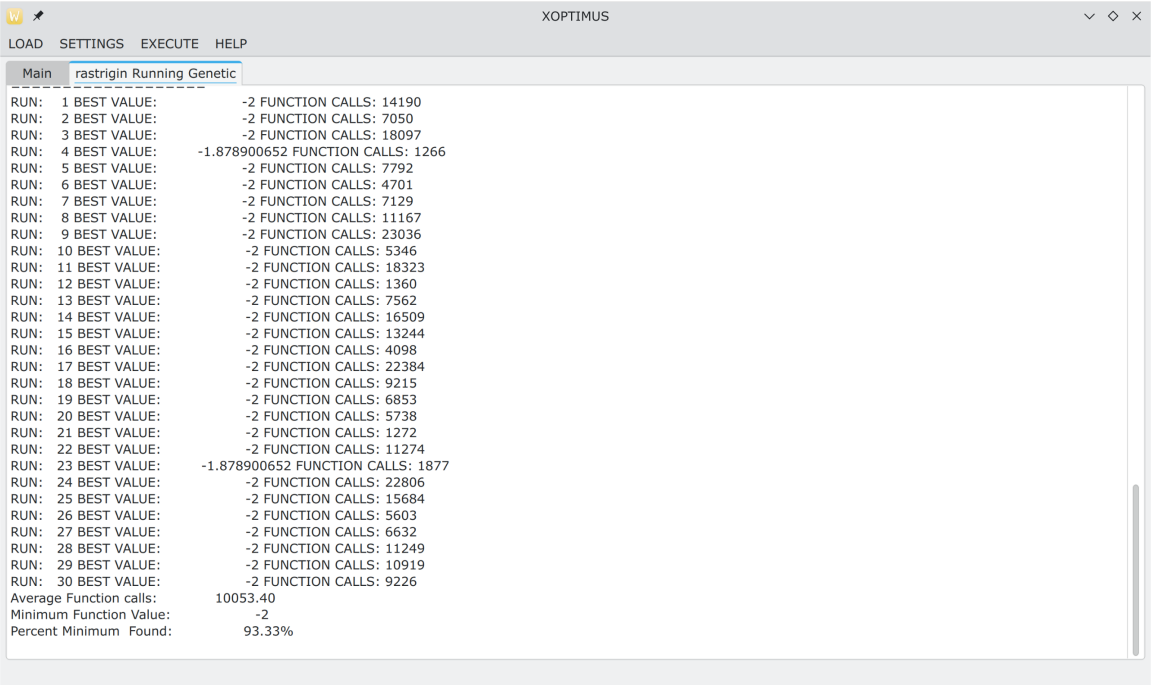


Figure 9. Executing the selected optimization method.

Available Global Optimization methods

The implemented Global Optimization methods are:

1. The Differential Evolution denoted as DifferentialEvolution.
2. The Parallel Differential Evolution method (utilizing the OpenMP library), denoted as ParallelDe.
3. A genetic algorithm, denoted as Genetic in the package.
4. An improved version of the Particle Swarm Optimizer, denoted as iPSO
5. A simple Multistart method, which initiates many local optimization procedures to discover the global optimum of a function.
6. The NeuralMinimizer method, which constructs an estimated function of the objective problem using Neural Networks.
7. A parallel Particle Swarm Optimizer (using the OpenMP library), denoted as ParallelPso.
8. A Simulated Annealing optimizer, denoted as Simman.
9. The optimal foraging algorithm (OFA), donated as Ofa.
10. Bio-inspired metaheuristic algorithm Giant Armadillo Optimization (GAO), donated as armadillo1.
11. The Gray Wolf Optimizer (GWO), donated as Gwo.

Available Local Optimization methods

All global optimization methods can be enhanced by applying a local minimization method after they are terminated. The parameter used to determine the used local optimization procedure is the --opt_localsearch parameter. The implemented local optimization methods are the following:

1. The bfgs method. The Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm was implemented using a variant of Powell.
2. The limited memory BFGS method (lbfgs).
3. The Gradient descent method. This method is denoted as gradient.
4. The Nelder Mead method (denoted as nelderMead).
5. The Adam optimizer (denoted as adam).

The script runfunmin.sh

It is a scrip to simplify the optimization tasks, used in Unix environments. A similar script for Windows environments is called runfunmin.bat