Optimization with the Distributed Execution Framework

Thomas Feilhauer⋆ (Fachhochschule Vorarlberg)
Steffen Finck (Fachhochschule Vorarlberg)

A central task of the Distributed Execution Framework (DEF) [1] is to supply efficient library routines across applications for simulation and optimization. The routines can then be included by the users in their applications and can be executed in parallel on the DEF. A key issue that is important for the acceptance of the DEF among users and routine developers is the independence of the programming language and the computer platform. Developers can implement their routines in any programming language, and the DEF users can also write their client applications in any programming language independent of the library routines used therein. The routines are called from the client application via an API developed for the DEF and the DEF, which is connected to the client application via a web service interface, takes care of the processing of the library routine calls.

The other important feature of the DEF is support for parallel processing of DEF applications. The DEF permits to create clusters with any number of worker nodes. For this purpose, the DEF relies on cloud technologies that enables the clusters to be operated both On- and Off-Premise. Finally, at runtime, the DEF distributes the invocation of library routines from the client application to the worker instances provided by the cluster.

The DEF's capabilities are exemplified on an expensive optimization problem. The aim is to perform a simulation-based optimization on a model of a bank's balance sheet to determine the worst case scenario [2]. The problem contains various bound constraints, a non-linear constraint, and is high-dimensional. Two approaches are used for the optimization, a simple random search approach and a Evolution Strategy. Next to the results obtained by these approaches, the respective run times and the scalability of the DEF are investigated.

[1] T. Feilhauer and M. Sobotka, *DEF-a programming language agnostic framework and execution environment for the parallel execution of library routines*, Journal of Cloud Computing, 5(1) (2016)

[2] S. Finck, *Worst Case Search over a Set of Forecasting Scenarios Applied to Financial Stress-Testing*, GECCO 19 Companion, ACM, (2019)