Derivative-free Optimization Solver for Calibration Problems

J. Fiala, B. Marteau (Numerical Algorithm Group)
L. Roberts, C. Cartis (University of Oxford)

InFoMM
Industrially Focused
Mathematical Modelling

Why derivative-free optimization (DFO)?

Classical optimization relies on provided derivatives:

- explicitly written derivatives
- finite differencing (FD, bumping), $\frac{\partial f}{\partial x_i} \approx \frac{f(x+he_i)-f(x)}{h}$
- algorithmic differentiation (see NAG AD tools)

Advantages of derivative-free optimization

- **black box** models – AD is not possible
- **noisy** models – FD are inaccurate or wrong
- **expensive** models – FD are impractical
- if high-accuracy is not required, DFO requires **fewer function evaluations**

NAG implements a DFO solver (e04ff) able to exploit the structure of calibration problems.

Robust noisy calibration

Noise can have undesired effects for optimization solvers. Model-based DFO solvers naturally present some **resilience to noise**, further enhanced in the NAG Library by:

- Specific choice of algorithmic parameters
- Automatic detection of early convergence and smart "soft" restart procedures

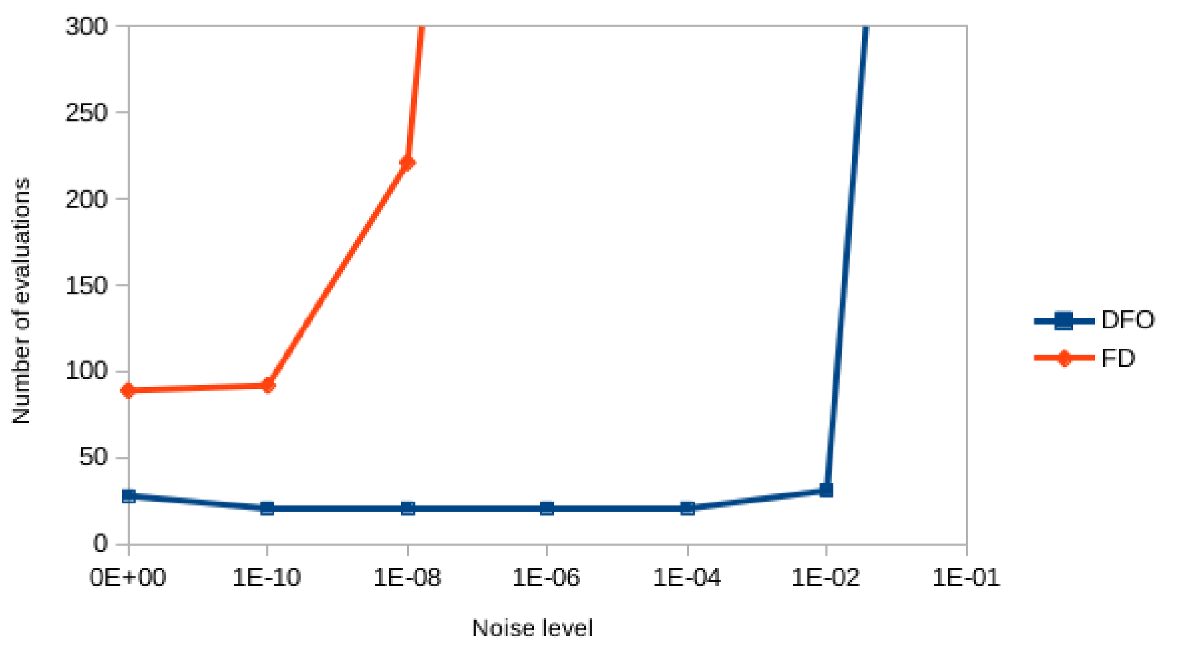
Ilustration. A noisy calibration via nonlinear least squares with uniform noise $\epsilon \in [-\nu, \nu]$

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^{m} (r_i(x) + \epsilon)^2$$

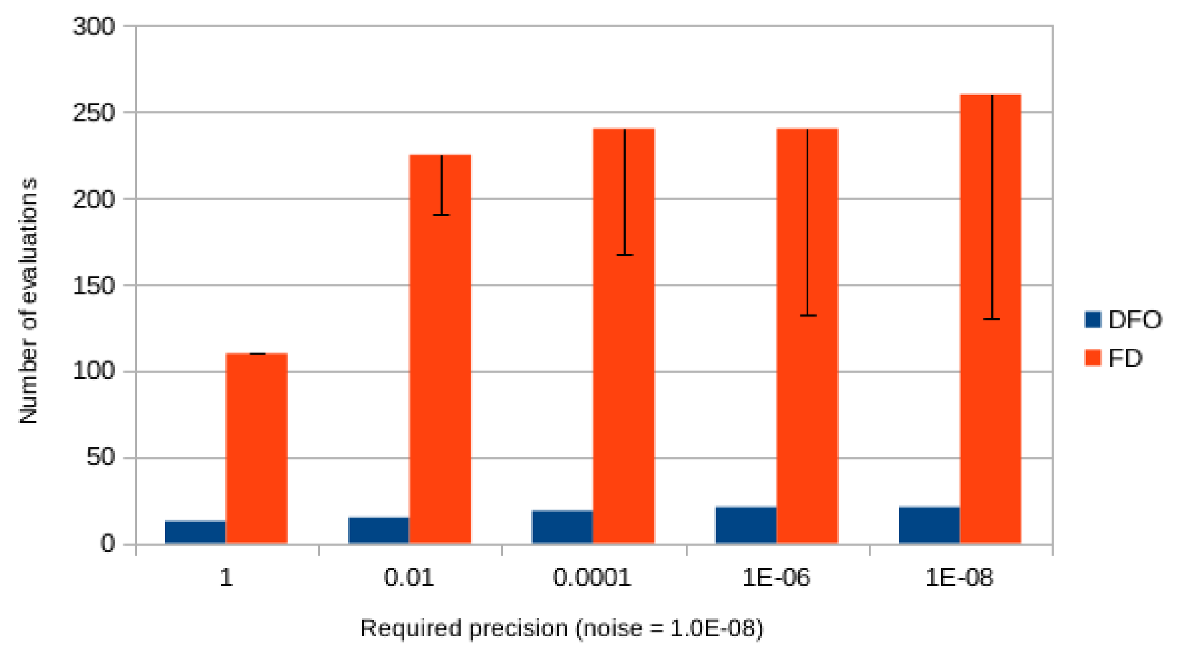
Comparison between quasi-Newton method combined with finite differences (e04fc) and the new derivative-free solver (e04ff) on the Rosenbrock test function:

Level of noise ν	0.0e00	1.0e-10	1.0e-08	1.0e-06	1.0e-04	1.0e-02	1.0e-01
Finite differences	89	92	221	∞	∞	∞	∞
DFO	**28**	**21**	**21**	**21**	**21**	**31**	∞

Table 1: Number of objective evaluations required to reach a point within 10^{-5} of the solution without noise



Evaluations to reach $F(x) \leq 10^{-5}$ w.r.t noise level



Avg. number of evaluations needed to reach given precision over 20 runs. Error bars represent the proportion of runs that fail.

Typical sources of noise in finance

- Objective funtion F might be evaluated in the interest of speed with **low precision algorithms** (such as low precision quadrature) which generate noise.
- Sometimes F is computed via **Monte Carlo** (for example hybrid models): as paths move in and out of the money, noise is introduced into the objective function.
- Often the market prices y_i have finite precision, for example prices are quoted to one basis point. Calibration is **not performed to high accuracy**: it can be stopped when model prices are within one basis point (1BP) of observed prices.

The third case is often acceptable when calibrating the Heston stochastic volatility model with term structure for European options.

A use case in finance: the Heston model calibration

Let $C^{model}(\rho_t, \alpha_t, \sigma_t, \lambda_t)$ represent the market price of a European option estimated by a Heston model with the following piecewise-constant time-dependant internal parameters:

ρ_t, correlation parameters of Brownian motions in the Heston model
α_t, volatility of the scaled volatility
σ_t, scaling parameter
λ_t, mean reversion rate

Under some assumptions, λ_t can be discarded from the calibration and set to a constant value.

We defined seven time intervals on which the parameters $(\rho_t, \alpha_t, \sigma_t)$ are constant and must be fitted to real market quotes C_i^{market}. Calibrating the model therefore amounts to sequentially solving seven three-dimensional nonlinear least squares problems with box constraints:

$$\min \sum_{i=1}^{m} (C_i^{model}(\rho_t, \alpha_t, \sigma_t) - C_i^{market})^2$$
$$\text{s.t.} \ -1 \leq \rho_t \leq 1, \alpha_t \geq 0, \sigma_t \geq 0$$

The table below shows a calibration of the Heston with term structure model to an accuracy of one basis point. **DFO-based solution delivers on average a 31% speedup.**

	Average number of evaluations	Average CPU time(s)
Finite differences	331	15.2
DFO	**223**	**10.5**

Table 2: Comparison of DFO (e04ff) and derivative based solvers (e04un) on 1355 Heston calibrations

DFO clearly outperforms derivative-based solvers when exact derivatives are lacking, or a high-accuracy calibration is not required. DFO is absolutely essential when the objective function evaluations are noisy.