Sequential quadratic programming (**SQP**) is an iterative method for constrained nonlinear optimization. SQP methods are used on mathematical problems for which the objective function and the constraints are twice continuously differentiable.

SQP methods solve a sequence of optimization subproblems, each of which optimizes a quadratic model of the objective subject to a linearization of the constraints. If the problem is unconstrained, then the method reduces to Newton's method for finding a point where the gradient of the objective vanishes. If the problem has only equality constraints, then the method is equivalent to applying Newton's method to the first-order optimality conditions, or Karush–Kuhn–Tucker conditions, of the problem.

Algorithm basics

Consider a nonlinear programming problem of the form:

$$\begin{array}{rl}\min\limits_{x} & f(x)\\ \text{s.t.} & b(x)\geq 0\\ & c(x)=0.\end{array}$$

The Lagrangian for this problem is[1]

$$\mathcal{L}(x,\lambda,\sigma)=f(x)-\lambda^T b(x)-\sigma^T c(x),$$

where λ and σ are Lagrange multipliers. At an iterate x_k, a basic sequential quadratic programming algorithm defines an appropriate search direction d_k as a solution to the quadratic programming subproblem

$$\begin{array}{rl}\min\limits_{d} & f(x_k)+\nabla f(x_k)^T d+\frac{1}{2}d^T\nabla_{xx}^2\mathcal{L}(x_k,\lambda_k,\sigma_k)d\\ \mathrm{s.\,t.} & b(x_k)+\nabla b(x_k)^T d\geq 0\\ & c(x_k)+\nabla c(x_k)^T d=0.\end{array}$$

Note that the term $f(x_k)$ in the expression above may be left out for the minimization problem, since it is constant.

Alternative approaches

- Sequential linear programming
- Sequential linear-quadratic programming
- Augmented Lagrangian method

Implementations

SQP methods have been implemented such well known numerical environments as MATLAB and GNU Octave. There also exist numerous software libraries, including open source

- SciPy (de facto standard for scientific Python) has scipy.optimize.minimize(method='SLSQP') solver
- NLopt (C/C++ implementation, numerous interfaces including Python, R, MATLAB/Octave)

and proprietary/commercial ones

- LabVIEW
- KNITRO[2] (C, C++, C#, Java, Python, Fortran)
- NPSOL (Fortran)
- SNOPT (Fortran)
- NLPQL (Fortran)
- SuanShu (Java)

See also

- Josephy-Newton algorithm
- Newton's method
- Secant method

Notes

1. Jorge Nocedal and Stephen J. Wright (2006). *Numerical Optimization*. Springer. ISBN 0-387-30303-0.
2. KNITRO User Guide: Algorithms

References

- Bonnans, J. Frédéric; Gilbert, J. Charles; Lemaréchal, Claude; Sagastizábal, Claudia A. (2006). *Numerical optimization: Theoretical and practical aspects*. Universitext (Second revised ed. of translation of 1997 French ed.). Berlin: Springer-Verlag. pp. xiv+490. doi:10.1007/978-3-540-35447-5. ISBN 3-540-35445-X. MR 2265882.
- Jorge Nocedal and Stephen J. Wright (2006). *Numerical Optimization*. Springer. ISBN 0-387-30303-0.

External links

- Sequential Quadratic Programming at NEOS guide

Last edited 2 months ago by an anonymous user