The Journal of Open Source Software

GALAHAD 4.0: an open source library of Fortran packages with C and Matlab interfaces for continuous optimization

Jaroslav M. Fowkes[1] **and Nicholas I. M. Gould**[1]

1 Science and Technology Facilities Council, Rutherford Appleton Laboratory, Harwell Campus, Didcot, Oxfordshire, OX11 0QX, UK

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor:

Submitted: 30 March 2022
Published: unpublished

Summary

The ability to solve continuous optimization problems is one of the cornerstones of computational mathematics. Such problems occur throughout science, engineering, planning and economics, since nature (and mankind) loves to optimize (minimize or maximize). Most real-life models of physical phenomena are nonlinear, and when discretised for computer solution they usually involve a large number of minimization variables (parameters) and/or constraints. Thus it is valuable to be able to rely on software specifically designed for optimization, and particularly that is designed to solve large, nonlinear problems.

Continuous optimization problems occur in a variety of formats. Problems may or may not involve constraints, least-squares fitting being a common but vital example of the latter. If there are constraints, they may simply be bounds on the values of the variables, or there may be linear or nonlinear relationships (both equations or inequalities) between sets of variables. In an ideal world, a global optimizer is sought, but often that is beyond current (and likely future) expectations particularly if there are a large number of variables involved; fortunately a local minimizer often suffices. There is also a natural hierarchy of problems, and the ability to solve one is useful if it occurs as a subproblem in a harder one—solving linear systems (sometimes approximately) is vital in linear or quadratic programming, quadratic programs are used within nonlinear programming methods, and local optimization is often a vital component of global optimization.

Thus ideally a comprehensive optimization library should address the different needs of its users by providing software tuned to a variety of commonly-occurring subclasses of problems. This is the aim of `GALAHAD`. `GALAHAD` provides packages for basic subproblem solvers (such as for linear systems, trust-region and regularization of quadratic and linear least-sqaures functions), linear and quadratic programming, unconstrained and bound-constrained optimization, nonlinear least-squares fitting, general nonlinear programming and both approximate univariate and multivariate global optimization, together with an array of attendant utilities packages (such as for polynomial fitting, hashing, presolves, and matrix approximation). It is also recognised that there are excellent external sources of relevant software, particular for solving linear systems, and `GALAHAD` provides uniform bridges to these if they are available.

Statement of need

The first release of the Fortran 90 `GALAHAD` library (Gould et al., 2003) aimed to expand the functionality of the earlier Fortran 77 `LANCELOT` package (Conn et al., 1992) for nonlinear optimization. Subsequent releases focused on increasing the scope of solvers provided, but aside from limited interfaces to Matlab and to the `CUTEst` modeling library (Gould et al., 2015), little effort was made to bridge the gap between Fortran and other, often more recent and popular, programming languages. `GALAHAD 4.0` addresses this deficiency.

Although `GALAHAD 4.0` contains an increased variety of new solvers, the principal motivation for the new release is to raise the profile of the library by increasing its potential userbase. While modern Fortran is an extremely flexible programming language, it is perceived as old fashioned in many circles. Rival open-source solvers such as `IPOPT` (Wächter & Biegler, 2006) and commercial ones such as `KNITRO` (Byrd et al., 2006) are written predominantly in C/C++, and this is attractive as there are often straightforward bridges from C to other popular languages such as Python and Julia. Thus, we have now provided interfaces from Fortran to C for a significant subset of the Fortran packages. This has been made possible using the standardised ISO-C bindings introduced in Fortran 2003, and enhanced in more modern revisions. Essentially an interface program binds Fortran types and functions to C equivalents, and a second C header file provides the C access.

A current list of major packages with C interfaces and their functionality is as follows:

package	purpose
uls	unsymmetric linear systems (external bridge)
sls	symmetric linear systems (external bridge)
sbls	symmetric block linear systems
psls	preconditioners for symmetric linear systems
fdc	determine consistency and redundancy of linear systems
lpa	linear programming using an active-set method (external bridge)
lpb	linear programming using an interior-point method
wcp	linear feasibility using an interior-point method
blls	bound-constrained linear least-squares problems using a gradient-projection method
bqp	bound-constrained convex quadratic programming using a gradient-projection method
bqpb	bound-constrained convex quadratic programming using an interior-point method
lsqp	linear and separable quadratic programming using an interior-point method
cqp	convex quadratic programming using an interior-point method
dqp	convex quadratic programming using a dual active-set method
eqp	equality-constrained quadratic programming using an iterative method
trs	the trust-region subproblem using matrix factorization
gltr	the trust-region subproblem using matrix-vector products
rqs	the regularized quadratic subproblem using matrix factorization
glrt	the regularized quadratic subproblem using matrix-vector products

package	purpose
dps	the trust-region and regularized quadratic subproblems in a diagonalising norm
lstr	the least-squares trust-region subproblem using matrix-vector products
lsrt	the regularized least-squares subproblem using matrix-vector products
l2rt	the regularized linear l_2 norm subproblem using matrix-vector products
qpa	general quadratic programming using an active-set method
qpb	general quadratic programming using an interior-point method
blls	bound-constrained linear-least-squares using a gradient-projection method
tru	unconstrained optimization using a trust-region method
arc	unconstrained optimization using a regularization method
nls	least-squares optimization using a regularization method
trb	bound-constrained optimization using a gradient-projection trust-region method
ugo	univariate global optimization
bgo	multivariate global optimization in a box using a multi-start trust-region method
dgo	multivariate global optimization in a box using a deterministic partition-and-bound method

Interfaces to other `GALAHAD` packages, such as `LANCELOT` and `FILTRANE` will be provided as time and demand permit. Future extensions to provide follow-on interfaces to Python and Julia using the C functionality are underway.

`GALAHAD` is easy to install using its own make -based system. Fortran documentation is provided in PDF via LaTeX, while both PDF and man documents for the C packages are available using Doxygen.

Acknowledgements

The authors are grateful for support provided by the Engineering and Physical Sciences Research Council (UK) grants EP/F005369/1, EP/I013067/1 and EP/M025179/1.

References

Byrd, R. H., Nocedal, J., & Waltz, R. A. (2006). KNITRO: An integrated package for nonlinear optimization. In *Large-scale nonlinear optimization* (Vol. 83, pp. 35–59). Springer, New York. https://doi.org/10.1007/0-387-30065-1/_4

Conn, A. R., Gould, N. I. M., & L. Toint, Ph. (1992). LANCELOT*: A Fortran package for large-scale nonlinear optimization (Release A)*. Springer-Verlag. https://doi.org/10.1007/978-3-662-12211-2

Gould, N. I. M., Orban, D., & L. Toint, Ph. (2003). GALAHAD—a library of thread-safe Fortran 90 packages for large-scale nonlinear optimization. *ACM Transactions on Mathematical Software*, *29*(4), 353–372. https://doi.org/10.1145/962437.962438

Gould, N. I. M., Orban, D., & L. Toint, Ph. (2015). CUTEst: A constrained and unconstrained testing environment with safe threads for mathematical optimization. *Computational Optimization and Applications*, *60*(3), 545–557. https://doi.org/10.1007/s10589-014-9687-3

Wächter, A., & Biegler, L. T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming Series A*, *106*(1), 25–57. https://doi.org/10.1007/s10107-004-0559-y