OPHELIA Library
Reference manual

Fabian Bastin

July 3, 2011

Contents

Chapter 1

Introduction

Ophelia stands for "Optimization and Harmless Estimation Library for AMLET". The painting "Ophelia" of the front cover is due to Arthur Hughes in 1865.

The library is designed to be install on standard UNIX-like system, so that it is sufficient to use the following commands sequence:

```
./configure
make
make install
```

Optimization compilation options can be specified as arguments of `configure`, for instance:

```
./configure CFLAGS=''-O3 -funroll-loops'' FFLAGS="-O2"
```

You can regenerate the tools with command

```
./bootstrap
```

1.1 BLAS library

`Ophelia` make heavy use of `CBlas` routines, the C version of `Blas` (`http://www.netlib.org/blas`). Many implementations exist, most of them proprietary and restricited to some specific architectures. We only review three open-source implementations: GotoBlas2, GSL Blas, and ATLAS

1.1.1 GotoBlas2

GotoBlas2, available at `http://www.tacc.utexas.edu/tacc-projects/gotoblas2`, is a Blas implementation proposed by the University of Texas

at Austin. Praised as one of the best implementations, it has been released under BSD license in 2010. We therefore strongly recommend its use, and Ophelia is compiled by default with GotoBlas if detected. A local copy is hosted at `http://amlet.slashbin.net`, that has been tested on Arch Linux, Fedora 7, SuSE 11.3, and Mac OS X 10.6.8. On Arch Linux, the compilation required a slight modification of the script `f_check`, obtained by changing `($flags =~ /^\-l/)` with `($flags =~ /^\-l[a-z,A-Z]/)`.

1.1.2 GSL CBlas

GSL (`"http://www.gnu.org/software/gsl/"`) Blas is supported only for convenience reasons, as it is available as a package on all the major distributions. The GPL is however incompatible with OSL, used for many parts of the software. However, Ophelia cannot be viewed as a derived work of GSL CBlas as the code can be compiled without any modifications with other CBlas implementations.

1.1.3 ATLAS

`ATLAS` (`"http://math-atlas.sourceforge.net/"`) is another free implementation of Blas, far better than the GSL one, but which present various issues during the installation process. `Ophelia` will try to detect ATLAS is neither GotoBLAS2 nor GSL CBlas have been found. Note however the you have to install ATLAS as shared libraries in order to be able to run the code.

Chapter 2

Unconstrained optimization

We consider the general problem

$$\min_{x \in \mathcal{R}^n} f(x). \tag{2.1}$$

2.1 Linesearch methods

BFGS Linesearch

2.2 Trust-region methods

2.2.1 Unconstrained optimization

2.3 Hessian approximations

While the gradient $\nabla f(x)$ usually must be known accurately in minimization algorithms, we often can rely on approximations of the Hessian $\nabla^2 f(x)$. This is especially appealing since finite or central differences require $O(n^2)$ function evaluations, which is prohibitive in many applications. In this section, we will focus on traditional secant updates, by focusing on their definitions, but not on the underlying convergence theory. We refer the reader interested in such issues to Chapter 8 of Dennis and Schnabel [2].

Reefering to the original problem 2.1, suppose that we have just taken a step from x_c to x_+. Following Dennis and Schnabel, we consider the Hessian as the derivative matrix of the nonlinear equations system $\nabla f(x) = 0$. The secant equation for the unconstrained minimization problem is

$$H_+ s_c = y_c, \tag{2.2}$$

where

$$s_c = x_+ - x_c,$$
$$y_c = \nabla f(x_+) - \nabla f(x_c).$$

Since the Hessian of some function is symmetric, we seek an H_+ that satisfies $H_+ = H_+^T$ as well as (2.2). The first secant update successfully used in the litterature is the symmetric secand update

BFGS

A popular approximation is the BFGS (Broyden, Fletcher, Goldfarb and Shanno):

$$B_{k+1} = B_k + \frac{y_k y_k^T}{y_k^t d_k} + \frac{B_k d_k (B_k d_k)^T}{d_k^T B_k d_k},$$

where $y_k = \nabla_x f(x_{k+1}) - \nabla_x f(x_k)$.

SR1

2.3.1 Trust-region radius update

Initial trust-region radius

Various options again exist to choose the initial trust-region as well as to update it.

The initialization strategy can be defined by setting the `radius_init` element of the `BTR` structure to the desired user's function, or to NULL, if no initialization has to be performed. In this case, the initial trust-region radius is affected to `BTR_INITIAL_DELTA`, defined in the file `defaults.h`.

The default behaviour of the algorithm sets the initial trust-region to $\|\nabla f(x_0)\|$, as described by Lin and More [3], while more advanced strategies exist. On the first iteration, the predicted step is used to compute a more reasonable trust-region radius, using the formula:

$$\Delta := \min\{\Delta, \|s_0\|\}.$$

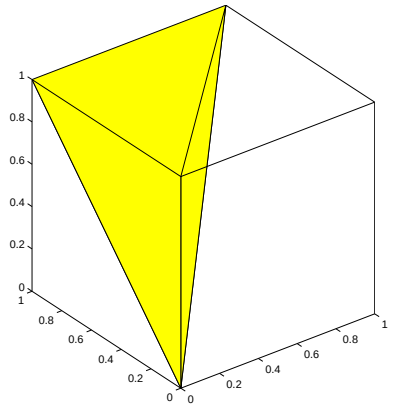
Chapter 3

Constrained optimisation

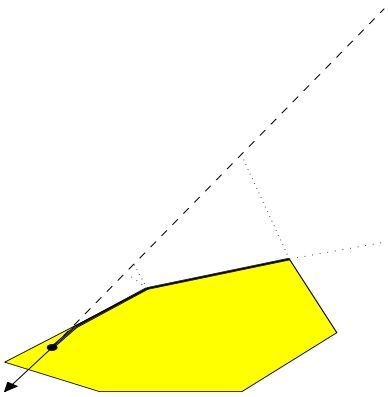
Projection methods

We consider the problem

$$\min_{x \in \mathcal{C}} f(x),$$



the order simplex in 3D



the projected path

3.1 Stopping criteria

3.1.1 Unconstrained optimisation

Relative gradient. Dennis and Schnabel [2], Page 160.

$$\max_i \frac{\nabla f(x)_i \max\{|x_i|, typ\ x_i\}}{\max\{|f(x)|, typ\ f\}},$$

where x_i is the i^{th} component of the vector x.

```
double nlp_relative_gradient(double val, double *x, double *grad,
                             double *typx, int n, double typf)
```

3.1.2 Constrained optimisation

Projection methods for trust-region methods

We use the criterion presented in Conn et al. [1], Section 12.1:

$$\|P[x - \nabla f(x)] - x\|_2 < \epsilon.$$

```
int btr_constrained_projection_stopping_test(BTR *btr, double rho)
```

$$\max_i \frac{(P[x - \nabla f(x)] - x)_i \max\{|x_i|, typ\ x_i\}}{\max\{|f(x)|, f(typ\ f\}},$$

where x_i is the i^{th} component of the vector x.

3.1.3 Specific projection algorithms

Isotonic regression

Isotonic regression (IR) involves finding a weighted least-squares fit $x \in \mathbb{R}^n$ to a vector $a \in \mathbb{R}^n$ with weights vector $w \in \mathbb{R}^n$ subject to a set of monotonicity constraints giving a simple or partial order over the variables.

Chapter 4

Metaheuristics methods

While some metaheuristics algorithms are proposed, it should be stressed that the library does not focus on such methods, and their support is therefore very limited.

4.1 Simulated annealing

```
NlpSa *nlp_sa_new();
```

Overview

Appendix A

Basic analysis

A.1 Banach spaces

A metric space is an ordered pair (M, d) where M is a set and d is a metric on M, that is, a function

$$d : M \times M \to \mathbb{R}$$

such that, for any x, y and z in M,

1. $d(x, y) \geq 0$;
2. $d(x, y) = 0$ if and only if $x = y$;
3. $d(x, y) = d(y, x)$;
4. $d(x, z) \leq d(x, y) + d(y, z)$.

The first condition follows from the other three, since:

$$2d(x, y) = d(x, y) + d(y, x) \geq d(x, x) = 0.$$

A more compact but equivalent definition replaces the above four axioms by the following two:

1. $d(x, y) = 0$ if and only if $x = y$;
2. $d(x, z) \leq d(x, y) + d(z, y)$.

The function d is also called distance function or simply distance. Often d is omitted and one just writes M for a metric space if it is clear from the context what metric is used.

A metric space M is said to be complete (or Cauchy) if every Cauchy sequence of points in M has a limit that is also in M or alternatively if every Cauchy sequence in M converges in M.

A seminormed vector space is a pair (V, p) where V is a vector space and p a seminorm on V. A normed vector space is a pair $(V, \|\cdot\|)$ where V is a vector space and $\|\cdot\|$ a norm on V. We often omit p or $\|\cdot\|$ and just write V for a space if it is clear from the context what (semi)norm we are using.

A Banach space is complete normed vector space. The familiar Euclidean spaces $\mathbb{R}^n$ and $\mathbb{C}^n$, where the Euclidean norm of $x = (x_1, \ldots, x_n)$ is given by

$$\|x\| = \sqrt{\sum_{i=1}^{n} x_i^2},$$

are Banach spaces.

A.2 Topological spaces

A topological space is a set X together with T, a collection of subsets of X, satisfying the following axioms:

1. The empty set and X are in T.
2. The union of any collection of sets in T is also in T.
3. The intersection of any finite collection of sets in T is also in T.

A topological space is called separable if it contains a countable dense subset; that is, there exists a sequence $\left\{_{x_n}\right\}_{n=1}^{\infty}\}$ of elements of the space such that every nonempty open subset of the space contains at least one element of the sequence.

Appendix B

License

All of the code is placed under the OSL. v2.1.

B.0.1 Open Software License v 2.1

This Open Software License (the "License") applies to any original work of authorship (the "Original Work") whose owner (the "Licensor") has placed the following notice immediately following the copyright notice for the Original Work:

Licensed under the Open Software License version 2.1

1) Grant of Copyright License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license to do the following:

   a) to reproduce the Original Work in copies;

   b) to prepare derivative works ("Derivative Works") based upon the Original Work;

   c) to distribute copies of the Original Work and Derivative Works to the public, with the proviso that copies of Original Work or Derivative Works that You distribute shall be licensed under the Open Software License;

   d) to perform the Original Work publicly; and

   e) to display the Original Work publicly.

2) Grant of Patent License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license, under patent

claims owned or controlled by the Licensor that are embodied in the Original Work as furnished by the Licensor, to make, use, sell and offer for sale the Original Work and Derivative Works.

3) Grant of Source Code License. The term "Source Code" means the preferred form of the Original Work for making modifications to it and all available documentation describing how to modify the Original Work. Licensor hereby agrees to provide a machine-readable copy of the Source Code of the Original Work along with each copy of the Original Work that Licensor distributes. Licensor reserves the right to satisfy this obligation by placing a machine-readable copy of the Source Code in an information repository reasonably calculated to permit inexpensive and convenient access by You for as long as Licensor continues to distribute the Original Work, and by publishing the address of that information repository in a notice immediately following the copyright notice that applies to the Original Work.

4) Exclusions From License Grant. Neither the names of Licensor, nor the names of any contributors to the Original Work, nor any of their trademarks or service marks, may be used to endorse or promote products derived from this Original Work without express prior written permission of the Licensor. Nothing in this License shall be deemed to grant any rights to trademarks, copyrights, patents, trade secrets or any other intellectual property of Licensor except as expressly stated herein. No patent license is granted to make, use, sell or offer to sell embodiments of any patent claims other than the licensed claims defined in Section 2. No right is granted to the trademarks of Licensor even if such marks are included in the Original Work. Nothing in this License shall be interpreted to prohibit Licensor from licensing under different terms from this License any Original Work that Licensor otherwise would have a right to license.

5) External Deployment. The term "External Deployment" means the use or distribution of the Original Work or Derivative Works in any way such that the Original Work or Derivative Works may be used by anyone other than You, whether the Original Work or Derivative Works are distributed to those persons or made available as an application intended for use over a computer network. As an express condition for the grants of license hereunder, You agree that any External Deployment by You of a Derivative Work shall be deemed a distribution and shall be licensed to all under the terms of this License, as prescribed in section 1(c) herein.

6) Attribution Rights. You must retain, in the Source Code of any Derivative Works that You create, all copyright, patent or trademark notices from the Source Code of the Original Work, as well as any notices of licensing and any descriptive text identified therein as an "Attribution Notice." You must cause the Source Code for any Derivative Works that You create to carry a prominent Attribution Notice reasonably calculated to inform recipients that You have modified the Original Work.

7) Warranty of Provenance and Disclaimer of Warranty. Licensor warrants that the copyright in and to the Original Work and the patent rights granted herein by Licensor are owned by the Licensor or are sublicensed to You under the terms of this License with the permission of the contributor(s) of those copyrights and patent rights. Except as expressly stated in the immediately proceeding sentence, the Original Work is provided under this License on an "AS IS" BASIS and WITHOUT WARRANTY, either express or implied, including, without limitation, the warranties of NON-INFRINGEMENT, MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY OF THE ORIGINAL WORK IS WITH YOU. This DISCLAIMER OF WARRANTY constitutes an essential part of this License. No license to Original Work is granted hereunder except under this disclaimer.

8) Limitation of Liability. Under no circumstances and under no legal theory, whether in tort (including negligence), contract, or otherwise, shall the Licensor be liable to any person for any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or the use of the Original Work including, without limitation, damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses. This limitation of liability shall not apply to liability for death or personal injury resulting from Licensor's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

9) Acceptance and Termination. If You distribute copies of the Original Work or a Derivative Work, You must make a reasonable effort under the circumstances to obtain the express assent of recipients to the terms of this License. Nothing else but this License (or another written agreement between Licensor and You) grants You permission to create Derivative Works based upon the Original Work or to exercise any of the rights

granted in Section 1 herein, and any attempt to do so except under the terms of this License (or another written agreement between Licensor and You) is expressly prohibited by U.S. copyright law, the equivalent laws of other countries, and by international treaty. Therefore, by exercising any of the rights granted to You in Section 1 herein, You indicate Your acceptance of this License and all of its terms and conditions. This License shall terminate immediately and you may no longer exercise any of the rights granted to You by this License upon Your failure to honor the proviso in Section 1(c) herein.

10) Termination for Patent Action. This License shall terminate automatically and You may no longer exercise any of the rights granted to You by this License as of the date You commence an action, including a cross-claim or counterclaim, against Licensor or any licensee alleging that the Original Work infringes a patent. This termination provision shall not apply for an action alleging patent infringement by combinations of the Original Work with other software or hardware.

11) Jurisdiction, Venue and Governing Law. Any action or suit relating to this License may be brought only in the courts of a jurisdiction wherein the Licensor resides or in which Licensor conducts its primary business, and under the laws of that jurisdiction excluding its conflict-of-law provisions. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any use of the Original Work outside the scope of this License or after its termination shall be subject to the requirements and penalties of the U.S. Copyright Act, 17 U.S.C. Â§101 et seq., the equivalent laws of other countries, and international treaty. This section shall survive the termination of this License.

12) Attorneys Fees. In any action to enforce the terms of this License or seeking damages relating thereto, the prevailing party shall be entitled to recover its costs and expenses, including, without limitation, reasonable attorneys' fees and costs incurred in connection with such action, including any appeal of such action. This section shall survive the termination of this License.

13) Miscellaneous. This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable.

14) Definition of "You" in This License. "You" throughout this License, whether in upper or lower case, means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with you. For purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

15) Right to Use. You may use the Original Work in all ways not otherwise restricted or conditioned by this License or by law, and Licensor promises not to interfere with or be responsible for such uses by You.

B.0.2 BSD License

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE

Bibliography

[1] Andrew R. Conn, Nicholas I. M. Gould, and Philippe L. Toint. *Trust-Region Methods*. SIAM, Philadelphia, USA, 2000.

[2] John E. Dennis and Robert B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Englewood Cliffs, New Jersey, USA, 1983.

[3] Chih-Jen Lin and Jorge J. Moré. Newton's method for large bound-constrained optimization problems. *SIAM Journal on Optimization*, 9(4):1100–1127, 1999.

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the definition; numbers in roman refer to the pages where the entry is used.