

## 1 Origins

We now discuss *linearly constrained Lagrangian methods* (LCL methods) for solving the general optimization problem

|     |  |
|-----|--|
| NCB | minimize $\phi(x)$<br>$x \in \mathbb{R}^n$<br>subject to $c(x) = 0, \quad \ell \leq x \leq u.$ |
|-----|--|

LCL methods seem to have evolved as an alternative to augmented Lagrangian methods, with the early developers being unaware that each method needed features of the other. (As we shall see, some features of the  $\ell_1$  penalty function are also needed.)

In the augmented Lagrangian approach, the penalty term  $\frac{1}{2}\rho_k \|c(x)\|^2$  provides positive curvature to the Lagrangian, allowing a new estimate of the optimal  $x$  to be obtained as a *minimizer* of the augmented Lagrangian  $L(x, y_k, \rho_k)$  using unconstrained or bound-constrained solvers. The risk is ill-conditioning of the Hessian  $\nabla_{xx}^2 L(x, y_k, \rho_k)$  if  $\rho_k$  is too large or only just large enough to create a local minimizer. A disadvantage is having to optimize within a large space (depending on the number of active bounds).

The methods of Robinson [10] and Robinson and Kreuser [11] made use of *linearized constraints* within the subproblems. They avoided ill-effects from the penalty term(!), and they offered the prospect of optimizing in a smaller space. Suppose  $(x^*, y^*)$  solves problem NCB, and suppose the constraints are linearized at  $x^*$  to give the linearly constrained problem

|     |  |
|-----|--|
| LC* | minimize $\phi(x) - y^{*T}c(x)$<br>$x \in \mathbb{R}^n$<br>subject to linearized $c(x) = 0, \quad \ell \leq x \leq u.$ |
|-----|--|

A key observation is that  $x^*$  solves LC\* as well as NCB. LCL methods build on this observation by defining subproblems like LC\* involving the Lagrangian and the linearization of  $c(x)$  at a sequence of points  $\{(x_k, y_k)\}$ .

The dual vector for LC\* proves to be  $\Delta y^* = 0$ , implying that the linearized constraints are not constraining the solution when the objective is the “optimal” Lagrangian  $L(x, y^*, 0)$ . The same is true when the objective is  $L(x, y^*, \rho)$  for any  $\rho \geq 0$ . This gives us scope for satisfying the second-order conditions for optimality.

Early views of LCL methods are well described by Fletcher [1]. The name SLC has been used in the past to indicate *sequential linearly constrained* subproblems that give rise to the next point  $(x_{k+1}, y_{k+1})$ . We use the term LCL to highlight the presence of the Lagrangian (or augmented Lagrangian) and the relationship to the BCL approach of LANCELOT.

## 2 Robinson’s Method

For a given point  $(x_k, y_k)$ , define the following linear and nonlinear functions:

*Linear approximation to  $c(x)$ :*

$$\bar{c}_k(x) = c(x_k) + J(x_k)(x - x_k).$$

*Departure from linearity:*

$$d_k(x) = c(x) - \bar{c}_k(x).$$

*Modified Lagrangian:*

$$M_k(x) = \phi(x) - y_k^T d_k(x).$$

Robinson's method obtains  $(x_{k+1}, y_{k+1})$  by solving the subproblem

|        |  |
|--------|--|
| $LC_k$ | $\begin{aligned} &\underset{x \in \mathbb{R}^n}{\text{minimize}} && M_k(x) \\ &\text{subject to} && \bar{c}_k(x) = 0, \quad \ell \leq x \leq u. \end{aligned}$ |
|--------|--|

Conceptually, the objective  $M_k(x)$  could be the normal Lagrangian because  $d_k(x)$  and  $c(x)$  are the same when the linearized constraints are satisfied ( $\bar{c}_k(x) = 0$ ). However, the solution to  $LC_k$  would then have to be regarded as  $(x_{k+1}, \Delta y_k)$  with  $y_{k+1} = y_k + \Delta y_k$ .

Under suitable conditions, Robinson [10] proves that the sequence of subproblem solutions converges *quadratically* to a solution of NCB.

A strength of the method is that any convenient solver may be used as a “black box” for the subproblems  $LC_k$ . An obvious example is the Reduced-Gradient method in MINOS. The solver may benefit from working within the reduced subspace defined by the constraint linearization. In addition, rapid convergence is obtainable without the use of second derivatives.

In practice, Robinson's method need not solve  $LC_k$  accurately until a solution is approached. It has been observed to succeed remarkably often, in the sense of converging to a local optimum on a wide range of convex and non-convex problems. Note that a good  $x_0$  with  $y_0 = 0$  generates a good  $(x_1, y_1)$  under favorable conditions.

The choice  $y_k = 0$ ,  $\rho_k = 0$  is known to converge for certain *reverse-convex* problems (see references in [9]), but in general, convergence is assured only if  $(x_0, y_0)$  is sufficiently close to  $(x^*, y^*)$ .

### 3 MINOS

In order to generalize the RG method in MINOS, Robinson's LCL approach was regarded as a promising alternative to GRG. As described in Murtagh and Saunders [9], the penalty term of the *augmented* Lagrangian is included in the subproblem objective in an attempt to improve convergence from arbitrary starting points. A *modified* augmented Lagrangian is used in  $LC_k$ :

$$M_k(x) = \phi(x) - y_k^T d_k(x) + \frac{1}{2} \rho_k \|d_k(x)\|^2,$$

and again it is equivalent to the normal augmented Lagrangian when  $\bar{c}_k(x) = 0$ . An important benefit is that if  $c(x)$  involves only *some* of the variables nonlinear, then  $M_k(x)$  has the same property, whereas  $\|c(x)\|^2$  appears to be more nonlinear.

#### 3.1 Inexact Solution of $LC_k$

MINOS uses Simplex or Reduced-Gradient iterations to satisfy the linearized constraints for each subproblem. It then limits the number of “minor iterations” performed on  $LC_k$  as a heuristic way to avoid excessive optimization within the wrong subspace.

By analogy with other augmented Lagrangian methods, it would be wise to check  $\|c(x)\|$  when each subproblem is terminated. If  $\|c(x)\|$  has increased substantially,  $\rho_k$  could be increased.

Instead,  $\|x_{k+1} - x_k\|$  and  $\|y_{k+1} - y_k\|$  are monitored in MINOS and if they seem large, the step towards  $(x_{k+1}, y_{k+1})$  is heuristically shortened.

#### 3.2 Reducing $\rho_k$

In practice it is readily observed that if the LCL subproblems converge, they generally do so more quickly if  $\rho_k$  is not large. By default,  $\rho_0$  is initially moderate ( $100/m_1$  for problems with  $m_1$  nonlinear constraints). When  $\|c(x)\|$  and  $\|y_{k+1} - y_k\|$  become reasonably small, MINOS reduces  $\rho_k$  in stages. It was originally thought that  $\rho_k = 0$  was needed to retain the quadratic convergence of Robinson's method, but a small positive value  $\rho_k = \bar{\rho}$  is currently retained after several reductions. As observed by Friedlander [2], Robinson's convergence

analysis applies to problem NCB with objective  $\bar{\phi}(x) \equiv \phi(x) + \frac{1}{2}\bar{\rho}\|c(x)\|^2$  (and  $c(x) = 0$  at a solution).

### 3.3 Infeasible Subproblems

Perhaps the greatest source of difficulty with Robinson's method (including its implementation in MINOS) is the occurrence of infeasible linearizations. Heuristically, MINOS relaxes the linearized constraints in stages. If the constraints remain infeasible after five consecutive relaxations, the original problem is declared infeasible. Otherwise, the relaxed  $LC_k$  is optimized as usual. Future subproblems are not relaxed unless they are also infeasible.

The relaxation process is successful sometimes, but in practice it has been more effective for users to add their own "elastic slacks" on troublesome nonlinear constraints, with a linear cost (penalty) on the elastic slacks. The subproblems are then automatically feasible in a relaxed form where necessary, and the relaxation is kept to a minimum by the penalties.

A similar process is automated within SNOPT [4]. All nonlinear constraints are made equally elastic if  $LC_k$  is infeasible, or if  $\|y_k\|$  grows large (a sign of impending infeasibility).

Elastic constraints are an even more integral part of the *stabilized LCL method* developed in Friedlander's thesis research [2]. Indeed, all of the heuristic maneuvers listed above are dealt with methodically by Friedlander's LCL algorithm. The resulting insights came 20 years after the first LCL version of MINOS, and 30 years after Robinson's tantalizing analysis of the local convergence properties of his *Newton-like* (but not exactly Newton) process.

## 4 A Stabilized LCL Method

In Friedlander and Saunders [3], the elastic LC subproblem is written in terms of the normal augmented Lagrangian

$$L_k(x) = \phi(x) - y_k^T c(x) + \frac{1}{2}\rho_k \|c(x)\|^2$$

as follows:

|   |
|---|
| $\begin{aligned} \text{(ELC}_k) \quad & \underset{x, v, w}{\text{minimize}} && L_k(x) + \sigma_k e^T(v + w) \\ & \text{subject to} && \bar{c}_k(x) + v - w = 0, \quad \ell \leq x \leq u, \quad v, w \geq 0, \end{aligned}$ |
|---|

where  $e$  is a vector of 1s and  $\sigma_k > 0$ . This subproblem is *always feasible*. Its solution is of the form  $(x_k^*, \Delta y_k^*, v_k^*, w_k^*)$ , with  $\sigma_k$  having the effect of enforcing  $\|\Delta y_k^*\|_\infty \leq \sigma_k$ .

The subproblem is also stated in terms of the  $\ell_1$  penalty function:

|   |
|---|
| $\begin{aligned} \text{(ELC}'_k) \quad & \underset{x}{\text{minimize}} && L_k(x) + \sigma_k \ \bar{c}_k(x)\ _1 \\ & \text{subject to} && \ell \leq x \leq u. \end{aligned}$ |
|---|

This form reveals a strong connection with the BCL approach. Indeed, the strategies for setting  $\rho_k$  and for controlling the accuracy of the subproblem solutions closely follow the BCL algorithm in LANCELOT. The primary innovation is the  $\ell_1$  penalty term. Far from a solution, this term allows the method to deviate from the constraint linearizations. Near a solution, it keeps the iterates close to the linearizations. For values of  $\sigma_k$  above a certain threshold, the linearized constraints are satisfied exactly (assuming the original constraints are feasible), thus permitting the rapid convergence of Robinson's method.

Experience with MINOS and SNOPT suggests that  $\rho_k$  should be *reduced* a finite number of times as convergence takes place, to make the subproblems easier to solve.

Returning to  $(ELC_k)$ , we may subtract the elastic linearized constraints from both occurrences of  $c(x)$  in  $L_k(x)$ . We now have the following functions:

Linear approximation to  $c(x)$ :

$$\bar{c}_k(x) = c(x_k) + J(x_k)(x - x_k).$$

Elasticized departure from linearity:

$$d_k(x, v, w) = c(x) - \bar{c}_k(x) - v + w.$$

Modified augmented Lagrangian:

$$M_k(x, v, w) = \phi(x) - y_k^T d_k(x, v, w) + \frac{1}{2} \rho_k \|d_k(x, v, w)\|^2.$$

With these definitions, a sequence  $\{(x_k, y_k)\}$  may be defined from increasingly accurate solutions  $(x_k^*, y_k^*)$  to elastic linearized subproblems of the form

|   |
|---|
| $\begin{aligned} (\text{ELC}_k'') \quad & \underset{x, v, w}{\text{minimize}} && M_k(x, v, w) + \sigma_k e^T(v + w) \\ & \text{subject to} && \bar{c}_k(x) + v - w = 0, \quad \ell \leq x \leq u, \quad v, w \geq 0. \end{aligned}$ |
|---|

#### 4.1 The sLCL Algorithm

The stabilized LCL algorithm is virtually the same as the BCL algorithm except for the elastic subproblem  $(\text{ELC}_k'')$  and manipulation of the associated parameter  $\sigma_k$ . The subproblem dual variables  $y_k^*$  and  $z_k^* \equiv \nabla_x M_k(x_k^*, v_k^*, w_k^*) - J_k^T y_k^*$  give  $y_{k+1}$  directly.

Key features are that the subproblems are solved inexactly (they are always feasible) and the penalty parameter  $\rho_k$  is increased only finitely often. The quantity  $\|\Delta y_k^*\|$  is bounded by  $\sigma_k + \omega_k$  and hence is uniformly bounded for large enough  $k$ . Under certain conditions, eventually  $v_k = w_k = 0$ , the  $\rho_k$ 's remain constant (and could be reduced to improve efficiency), the iterates converge quadratically as in Robinson's method, and the algorithm terminates in a finite number of iterations.

---

**Algorithm 1:** sLCL (Stabilized LCL Method).

---

**Input:**  $x_0, y_0, z_0$

**Output:**  $x^*, y^*, z^*$

Set  $\bar{\sigma} \gg 1$  and scale factors  $\tau_\rho, \tau_\sigma > 1$ .

Set penalty parameters  $\rho_0 > 0$  and  $\sigma_0 \in [1, \bar{\sigma}]$ .

Set positive convergence tolerances  $\omega_*, \eta_* \ll 1$  and infeasibility tolerance  $\eta_0 > \eta_*$ .

Set constants  $\alpha, \beta > 0$  with  $\alpha < 1$ .

$k \leftarrow 0$

converged  $\leftarrow$  false

**repeat**

Choose optimality tolerance  $\omega_k > 0$  such that  $\lim_{k \rightarrow \infty} \omega_k \leq \omega_*$ .

Find a point  $(x_k^*, v_k^*, w_k^*, y_k^*, z_k^*)$  that solves  $(\text{ELC}_k'')$  within tolerance  $\omega_k$ .

**if**  $\|c(x_k^*)\| \leq \max(\eta_*, \eta_k)$  **then**

$x_{k+1} \leftarrow x_k^*, y_{k+1} \leftarrow y_k^*, z_{k+1} \leftarrow z_k^*$  [update solution estimates]

**1 if**  $(x_{k+1}, y_{k+1}, z_{k+1})$  solves NCB **then** converged  $\leftarrow$  true

$\Delta y_k^* \leftarrow y_k^* - y_k + \rho_k c_k(x_k^*)$

**2**  $\rho_{k+1} \leftarrow \rho_k$  [keep  $\rho_k$ ]

**3**  $\sigma_{k+1} \leftarrow \frac{1}{1+\rho_k} \min(1 + \|\Delta y_k^*\|_\infty, \bar{\sigma})$  [reset  $\sigma_k$ ]

$\eta_{k+1} \leftarrow \eta_k / (1 + \rho_{k+1}^\beta)$  [decrease  $\eta_k$ ]

**else**

$x_{k+1} \leftarrow x_k, y_{k+1} \leftarrow y_k, z_{k+1} \leftarrow z_k$  [keep solution estimates]

$\rho_{k+1} \leftarrow \tau_\rho \rho_k$  [increase  $\rho_k$ ]

$\sigma_{k+1} \leftarrow \sigma_k / \tau_\sigma$  [decrease  $\sigma_k$ ]

$\eta_{k+1} \leftarrow \eta_0 / (1 + \rho_{k+1}^\alpha)$  [may increase or decrease  $\eta_k$ ]

**end**

$k \leftarrow k + 1$

**until** converged

$x^* \leftarrow x_k, y^* \leftarrow y_k, z^* \leftarrow z_k$

---

## 4.2 Infeasible Problems

As given above, the sLCL algorithm might not terminate if problem NCB has no solution. An additional test is needed to force a “Problem is infeasible” exit if  $\rho_k$  is above a certain threshold and  $\|c(x_k^*)\|$  is consistently not decreasing.

In [2, 3] it is shown that on infeasible problems, the sLCL algorithm converges to a local minimizer or stationary point for the function  $\|c(x)\|_2^2$ . In contrast, SNOPT minimizes  $\|c(x)\|_1$ .

## 4.3 Looking Ahead

A Fortran 90 implementation of the sLCL algorithm based on subproblem (ELC $_k''$ ) has been developed. It is called Knossos (home of King Minos). It allows for a choice of LC subproblem solvers: the reduced-gradient method of MINOS, the SQP method of SNOPT, and perhaps future solvers that make use of second derivatives.

Recall that for over 20 years, MINOS has been an important nonlinear solver for GAMS and AMPL, even though it may fail on problems that are highly nonlinear or infeasible. (CONOPT has been especially important for such cases.) Also, SNOPT has demonstrated great robustness by solving about 900 of the 1000 CUTER test problems [5], using remarkably few function and gradient evaluations [4]. A concern is that SNOPT sometimes needs rather many major iterations (and hence linear algebra) compared to MINOS. Knossos/MINOS and Knossos/SNOPT promise the best of both worlds.

At the same time, nonlinear interior methods such as IPOPT, KNITRO, and LOQO [6, 7, 8] also show considerable promise for applications where second derivatives are available.

## References

- [1] R. Fletcher. Methods related to Lagrangian functions. In P. E. Gill and W. Murray, editors, *Numerical Methods for Constrained Optimization*, pages 219–239. Academic Press, London, 1974.
- [2] M. P. Friedlander. *A Globally Convergent Linearly Constrained Lagrangian Method for Nonlinear Optimization*. PhD thesis, Dept of Management Science and Engineering, Stanford University, 2002.
- [3] M. P. Friedlander and M. A. Saunders. A globally convergent linearly constrained Lagrangian method for nonlinear optimization. *SIAM J. Optim.*, 15(3):863–897, 2005.
- [4] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Review*, 47(1):99–131, 2005. SIGEST article.
- [5] N. I. M. Gould, D. Orban, and Ph. L. Toint. CUTER and SifDec: A constrained and unconstrained testing environment, revisited. *ACM Trans. Math. Softw.*, 29(4):373–394, 2003.
- [6] IPOPT open source NLP solver. <https://projects.coin-or.org/Ipopt>.
- [7] KNITRO optimization software. <http://www.ziena.com>.
- [8] LOQO optimization software. <http://orfe.princeton.edu/~loqo>.
- [9] B. A. Murtagh and M. A. Saunders. A projected Lagrangian algorithm and its implementation for sparse nonlinear constraints. *Math. Program. Study*, 16:84–117, 1982.
- [10] S. M. Robinson. A quadratically-convergent algorithm for general nonlinear programming problems. *Math. Program.*, 3:145–156, 1972.
- [11] J. B. Rosen and J. Kreuser. A gradient projection algorithm for nonlinear constraints. In F. A. Lootsma, editor, *Numerical Methods for Nonlinear Optimization*, pages 297–300. Academic Press, London and New York, 1972.