A Line-Search Method For Lagrangian Relaxation Ascent Algorithms

Gongyun Zhao and Zhuangwei Liu

Department of Mathematics
National University of Singapore
10 Kent Ridge Crescent, 0511
SINGAPORE

November 1994

Abstract

The Lagrangian relaxation strategy (or dualization) is one of the most important methodologies of optimization for solving structured large-scale mathematical programming problems. The line search procedure is very often encountered in solving the dual problem by using some ascent method, such as bundle method, interior point method, etc.. The existing line search methods, for example, Armijo's method, need to evaluate the dual objective function many times, in which each evaluation of the dual objective is to solve a relaxed subproblem, and therefore are time-consuming. In this short report we will propose an efficient line search method which is applicable to the Lagrangian relaxation ascent algorithms. The advantage of the new method is that the line search is embedded in evalution of the dual function.

Key Words: Line search, Lagrangian relaxation, Bundle method, Interior point method, Duality.

Abbreviated Title: Line search method.

AMS(MOS) subject classifications: 90C25, 49M35.

1. Lagrangian relaxation and line search

Before discussing the line search method, we first briefly describe two types of methods which can be used to solve structured mathematical programming problems. Let us, for instance, consider the following problem.

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & g(x) \leq 0 \\ & x \in X := \{x \in R^n \mid Bx \leq b\}, \end{aligned} \tag{1}$$

where $f_0 : R^n \to R$ and $g : R^n \to R^m$ are convex functions, $B \in R^{l \times n}$ has a simple structure, e.g., is a block diagonal matrix. The Problem (1) would be easy to solve without the presence of the constraint $g(x) \leq 0$. Applying Lagrangian relaxation with respect to the constraint $g(x) \leq 0$ gives the following dual problem

$$\begin{aligned} \max \quad & \theta(u) \\ \text{s.t.} \quad & u \geq 0 \end{aligned}$$

where $u \in R^m$ is a Lagrangian multiplier vector, and the dual objective function $\theta(u)$ is defined as

$$\begin{aligned} \theta(u) = \min \quad & f_0(x) + u^T g(x) \\ \text{s.t.} \quad & x \in X. \end{aligned} \tag{2}$$

Normally, the function $\theta(u)$ is non-smooth. Because of the non-smoothness, at each u we can only find a subgradient which may not be an ascent direction. For finding ascent directions, a variety of bundle methods have been developed, cf [1] [2] [5] [6] [7] [8]. At an iterative point u_k, a set of subgradients, called a bundle, is used to construct a direction. Then a line search is performed resulting either in a new point u_{k+1} if the direction is ascent, or in a new subgradient which will be added to the bundle if the direction is not ascent. In the latter case, no new point is found, but the bundle is enlarged hence is able to construct a better direction.

Another way of solving (1) uses a barrier function to avoid the non-smoothness. This strategy first adds a logarithmic barrier term to the objective, defining

$$f(t, x) = f_0(x) - t \sum_{i=1}^{l} \ln(b_i - B_i^T x),$$

where $(B_1, \ldots, B_l)^T = B$ and $(b_1, \ldots b_l)^T = b$, and then apply Lagrangian relaxation to $g(x) \leq 0$, resulting in the following dual function

$$\begin{aligned} \theta(t, u) = \min \quad & f(t, x) + u^T g(x) \\ \text{s.t.} \quad & x \in X. \end{aligned} \tag{3}$$

For each $t > 0$ and $u \geq 0$, by virtue of the logarithmic barrier term, the minimizer $x = x(t, u)$ in (3) is an interior point of X. Hence, $\theta(t, u)$ defined in (3) is a smooth function with respect to t and u. That means, for any fixed $t > 0$

$$\begin{aligned} \max \quad & \theta(t, u) \\ \text{s.t.} \quad & u \geq 0 \end{aligned} \tag{4}$$

is a smooth programming problem. Such a method, called the Lagrangian relaxed interior point method, solves (1) in the following way, cf [3] [4]. It consists of two loops, namely, outer loop and inner loop. At k-th outer iteration, a point u_k is given and a t_k is determined. Then an inner loop is performed to approximate the solution $u(t_k)$ of (4) from u_k, resulting in u_{k+1}. It then returns to the outer loop at $(k+1)$-th iteration, reducing t_k to t_{k+1} and performing an inner loop again with u_{k+1} and t_{k+1}. As t_k tends to zero, the approximate solution u_k will tend to the solution of (1). In this report we will focus on solving (4) for fixed $t_k > 0$, i.e. on the performance of the inner loop. At each inner iteration, an ascent direction should be computed and a line search be performed. If t_k is reduced by a small factor at each outer iteration, the line search is not required. However, in order to accelerate the algorithm, big reduction of t_k is usually be taken, hence the line search should be necessary.

It is worth noting that the line search in each of above methods is to find the maximum or approximate maximum of $\theta(u_k + \alpha\Delta u)$ along a direction Δu for $\alpha \geq 0$. Using any existing line search method, e.g. Armijo's method or line search methods presented in [1] [5], one should evaluate function θ many times, and each evaluation of θ is actually to solve a subproblem like (2) or (3) to find a minimum $x(u)$ or $x(t,u)$. Such a line search method is evidently time-consuming. In this report, we will propose an efficient line search method which is applicable to the Lagrangian relaxation ascent algorithms we mentioned above. The advantage of the new line search method is that the line search is embedded in evaluation of the dual function θ.

In this report, we are interested in line search methods. Hence, we do not go into details to discuss how to find an ascent direction from a bundle of subgradients in a bundle method, and how to reduce t_k in the interior point method, but supposing a fixed t_k. In what follows, we discuss the line search method based on the problem below,

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) \leq 0 \\ & x \in X \end{aligned} \tag{5}$$

where $f : R^n \to R$ and $g : R^n \to R^m$ are convex functions, and X is a convex set in R^n with a simple structure. In order to make use of the simple structure of X, we dualize the constraint $g(x) \leq 0$, resulting in the Lagrangian dual problem as follows.

$$\begin{aligned} \max \quad & \theta(u) \\ \text{s.t.} \quad & u \geq 0 \end{aligned} \tag{6}$$

where the dual objective function $\theta(u)$ is defined as

$$\begin{aligned} \theta(u) = \min \quad & f(x) + u^T g(x) \\ \text{s.t.} \quad & x \in X. \end{aligned} \tag{7}$$

We will always suppose that at a point u, an ascent direction Δu is found. Our goal is to derive a method to determine a step length as a solution to

$$\begin{aligned} \max \quad & \hat{\theta}(\alpha) \\ \text{s.t.} \quad & u + \alpha\Delta u \geq 0 \end{aligned} \tag{8}$$

where

$$\hat{\theta}(\alpha) := \theta(u + \alpha\Delta u) = \min \quad f(x) + u^T g(x) + \alpha \Delta u^T g(x) \qquad \text{s.t.} \quad x \in X. \tag{9}$$

The constraint of (8) is equivalent to

$$\alpha \leq \alpha_{max} := \min\{-\frac{u_i}{\Delta u_i} \mid \Delta u_i < 0, \, i = 1, \ldots m\}.$$

Let $\beta = \alpha_{max} - \alpha$, then (8) can be rewritten as

$$\begin{array}{ll} \max & \tilde{\theta}(\beta) \\ \text{s.t.} & \beta \geq 0 \end{array} \tag{10}$$

where

$$\begin{array}{rl} \tilde{\theta}(\beta) = \min & f(x) + (u + \alpha_{max}\Delta u)^T g(x) - \beta \Delta u^T g(x) \\ \text{s.t.} & x \in X. \end{array}$$

It is clear that (10) is the dual of the following problem

$$\begin{array}{ll} \min & f(x) + (u + \alpha_{max}\Delta u)^T g(x) \\ \text{s.t.} & \Delta u^T g(x) \geq 0 \\ & x \in X. \end{array} \tag{11}$$

If functions f and g satisfy certain constraint qualification, then by the Strong Duality Theorem, (10) and (11) have the same optimal objective value, and the optimal Lagrangian multiplier β to the constraint $\Delta u^T g(x) \geq 0$ in (11) is the optimal solution to (10), and therefore, it can be used to determine the optimal step length α defined by (8).

Summarizing the above discussion, our line search method is as follows: For a given u_k and Δu_k at iteration k, we solve problem (11) with any algorithm which returns a solution x_{k+1} and a Lagrangian multiplier β_k associated with the constraint $\Delta u_k^T g(x) \geq 0$, from which we obtain the point $u_{k+1} = u_k + \alpha_k \Delta u_k$, where $\alpha_k = \alpha_{max} - \beta_k$. It is worth noting that problem (11) returns x_{k+1} but not x_k, since at the iteration k, the solution from problem (11) is corresponding to u_{k+1}, i.e., it is the solution to $\theta(u_{k+1}) = \inf\{f(x) + u_{k+1}^T g(x) \mid x \in X\}$. Using x_{k+1}, we then compute an ascent direction Δu_{k+1} at u_{k+1}, (cf [4] [6] [7]). The above procedure is repeated until a near-optimal solution (x^*, u^*) is obtained.

We can also consider the convex programming problem with the constraint $g(x) = 0$, i.e.

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & g(x) = 0 \\ & x \in X \end{array} \tag{12}$$

The the Lagrangian dual of (12) is an unconstrained problem

$$\max \quad \theta(u),$$

where $\theta(u)$ is defined as (7). Since the dual variable u is unconstrained, the step length is then defined by the following unconstrained problem

$$\max \quad \hat{\theta}(\alpha), \tag{13}$$

where $\hat{\theta}(\alpha)$ is defined as (9). The problem (13) is a dual of the following problem.

$$\begin{aligned} \min \quad & f(x) + u^T g(x) \\ \text{s.t.} \quad & \Delta u^T g(x) = 0 \\ & x \in X \end{aligned} \tag{14}$$

The step length α, which is the Lagrangian multipler with respect to the constraint $\Delta u^T g(x) = 0$, can be determined by solving problem (14). Therefore, we call (11) and (14) the *line search subproblems.*

By an existing line search method, such as Armijo's rule, we have to evaluate function $\theta(u)$, i.e. to solve the subproblem (7) many times before we can obtain an approximately optimal step length. In contrast with this, our line search method needs only to solve the line search subproblem (11) or (14) once to find an optimal step length. On the other hand, even though the subproblem (11) (or (14)) looks more complicate than the subproblem (7), the former actually does not cost much more than the latter, because the former contains merely one more one-dimensional side constraint which will not incur great expense on calculations. To illustrate this, we discuss an example in the next section.

2. Solving line search subproblems

In this section, we use a concrete problem as an example to show the performance of our line search method discussed in Section 1.

We consider the following problem.

$$\begin{aligned} \min \quad & c^T x - t \sum_{i=1}^{n} \ln x_i \\ \text{s.t.} \quad & Ax = a \\ & x \in X, \end{aligned} \tag{15}$$

where $A \in R^{m \times n}$ and

$$X = \{x \in R^n \mid Bx = b, x \geq 0\}.$$

Here $B \in R^{k \times n}$ is a block diagonal matrix. This problem arises from using interior point method to solve a linear programming problem in the following form, cf [4].

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = a \\ & Bx = b \\ & x \geq 0. \end{aligned}$$

For each $t > 0$, (15) needs to be approximately solved. Since the matrix B has a block diagonal structure, the problem (15) with the absence of the constraint $Ax = a$ could be decomposed into independent subproblems and easily solved. Therefore, we relax the constraint $Ax = a$ and obtain the dual of (15) as follows,

$$\max \quad \theta(u), \tag{16}$$

where

$$\theta(u) = \inf\{c^T x - t\sum_{i=1}^{n} \ln x_i + u^T(Ax - a) \mid x \in X\}. \tag{17}$$

Given u and Δu (cf [4] Section 7 for determining Δu), the line search subproblem (14) becomes

$$\begin{aligned} \min \quad & (c + A^T u)^T x - t\sum_{i=1}^{n} \ln x_i \\ \text{s.t.} \quad & \Delta u^T A x = \Delta u^T a \\ & Bx = b \\ & x \geq 0. \end{aligned} \tag{18}$$

Here we omit $u^T a$ in the objective function, since it is a constant.

For the purpose of obtaining the optimal solution $\bar{x}$ and the optimal Lagrangian multiplier $\bar{\alpha}$ to the constraint $\Delta u^T Ax = \Delta u^T a$, we can solve KKT-conditions below instead directly solving (18). The KKT-condition of problem (18) is given by

$$\begin{aligned} & c + A^T u - tx^{-1} - \alpha A^T \Delta u - B^T y = 0 \\ & \Delta u^T A x = \Delta u^T a \\ & Bx = b \\ & x \geq 0. \end{aligned}$$

where the α and y are the Lagrangian multipliers with respect to the constraints $\Delta u^T Ax = \Delta u^T a$ and $Bx = b$, respectively.

For $x, s \in R^n$ and $v \in R$, we define

$$x \circ s = (x_1 s_1, \ldots, x_n s_n)^T, \qquad x^v = (x_1^v, \ldots, x_n^v)^T,$$

and we denote $e = (1, \ldots, 1)^T \in R^n$.

Let $s = tx^{-1}$ then we have

$$\begin{aligned} & x \circ s = te \\ & s + \alpha A^T \Delta u + B^T y = c + A^T u \\ & \Delta u^T A x = \Delta u^T a \\ & Bx = b \\ & x \geq 0, s \geq 0. \end{aligned} \tag{19}$$

Because of $x \circ s = te$ and $t > 0$, the solution (x, s) to (19) must be positive, i.e. an interior point of the nonnegative orthant. Hence we can use Newton's method to solve (19). The Newton direction $(\Delta x, \Delta s, \Delta y, \Delta\alpha)$ is defined by

$$\begin{aligned} & x \circ \Delta s + s \circ \Delta x = te - x \circ s \\ & \Delta s + B^T \Delta y + \Delta\alpha A^T \Delta u = 0 \\ & \Delta u^T A \Delta x = 0 \\ & B \Delta x = 0. \end{aligned} \tag{20}$$

We denote

$$D = \text{diag } (x \circ s^{-1})^{1/2}, \qquad \bar{B} = \begin{pmatrix} B \\ \Delta u^T A \end{pmatrix}, \qquad \eta = t(x \circ s)^{-1/2} - (x \circ s)^{1/2}.$$

Then one can verify that the solution of (20) is

$$\Delta s = D^{-1} P_{\bar{B}D} \eta, \qquad \Delta x = D(I - P_{\bar{B}D})\eta, \qquad \begin{pmatrix} \Delta y \\ \Delta\alpha \end{pmatrix} = -(\bar{B} D^2 \bar{B}^T)^{-1} \bar{B} D \eta, \tag{21}$$

where $P_{\bar{B}D}$ is the projection matrix associated with $\bar{B}D$ defined by

$$P_{\bar{B}D} = D\bar{B}^T (\bar{B} D^2 \bar{B}^T)^{-1} \bar{B} D.$$

We can represent $P_{\bar{B}D}$ in terms of P_{BD} and $DA^T \Delta u$.

$$P_{\bar{B}D} = \frac{1}{\Delta u^T A D (I - P_{BD}) D A^T \Delta u} (I - P_{BD}) D A^T \Delta u \Delta u^T A D (I - P_{BD}) + P_{BD} \tag{22}$$

where

$$P_{BD} = DB^T (BD^2 B^T)^{-1} BD.$$

Because the matrix B is well structured, e.g. block diagonal matrix, P_{BD} is easy to compute. More important, if P_{BD} is known, then since $DA^T \Delta u$ is a vector, the computation of the first term in (22) to obtain $P_{\bar{B}D}$ involves only matrix-vector multiplications, which costs even less than the computation of P_{BD} itself.

Now let us compare the cost of solving the subproblem (18) (or (19)) and the subproblem below,

$$\begin{aligned} \theta(u) = \min \quad & (c + A^T u)^T x - t \sum_{i=1}^{n} \ln x_i \\ \text{s.t.} \quad & Bx = b \\ & x \geq 0. \end{aligned} \tag{23}$$

We note that the evaluation of the dual function $\theta(u)$ is to solve (23) for a given u. The KKT-condition of problem (23) is equivalent to

$$\begin{aligned} & x \circ s = te \\ & s + B^T y = c + A^T u \\ & Bx = b \\ & x \geq 0, s \geq 0. \end{aligned} \tag{24}$$

The Newton direction for solving problem (24) is

$$\delta s = D^{-1} P_{BD} \eta, \qquad \delta x = D(I - P_{BD})\eta, \qquad \delta y = -(BD^2 B^T)^{-1} BD\eta. \tag{25}$$

The main expense in solving problems (19) and (24) is the computation of the Newton directions $(\Delta x, \Delta s)$ and $(\delta x, \delta s)$ for which $P_{\bar{B}D}$ and P_{BD} are used respectively. Observing the formula (22) and recalling the remark after that, we can see that the computation of $P_{\bar{B}D}$ does not cost more than the computation of two P_{BD}'s. Therefore, to find the optimal step length $\bar{\alpha}$ by our line search method which is to solve (19) once, does not cost more than to compute two values of the function $\theta(u)$ which are to solve (24) two times. Since any existing lines search method has to evaluate the function $\theta(u)$ many times to determine a step length, our line search method should be much more effective than them.

References

[1] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms I, II*, Springer-Verlag, Berlin, 1993.

[2] K.C. Kiwiel, "Approximation in proximal bundle methods and decomposition of convex programs," to appear in: *J. Optim. Theory Appl.*, 84 (1995).

[3] M. Kojima, N. Megiddo, and S.Mizuno, "A Lagrangian relaxation method for approximating the analytic center of a polytope," Technical Report, IBM Almaden Research Center, San Jose, CA 95120–6099, USA, 1992.

[4] M. Kojima, N. Megiddo, S.Mizuno, and S. Shindoh, "Horizontal and vertical decomposition in interior point methods for linear programs," Working paper, Dept of Information Sciences, Tokyo Institute of Technology, 2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152, Japan, 1993.

[5] M.M. Mäkelä and P. Neittaanmäki, *Nonsmooth Optimization, Analysis and algorithms with applications to optimal control*, World Scientific, New Jersey, 1992.

[6] D. Medhi, "Parallel bundle-based decomposition for large-scale structured mathematical programming problems," *Annals of Operations Research* 22, 101–127 (1990).

[7] S.M. Robinson, "Bundle-based decomposition: description and preliminary results," in *System Modelling and Optimization*, eds. A. Prékopa, J. Szelezsán and B. Strazicky, Lecture Notes in Control and Information Sciences, Vol. 84, 751–756 (1986).

[8] H. Schramm and J Zowe, "A version of the bundle idea for minimizing a nonsmooth function: Conception idea, convergence analysis, numerical results," *SIAM J. Optim.* 2, 121–152 (1992).