Working Paper

Perturbation Methods for Saddle Point Computation

Markku Kallio and Andrzej Ruszczyński

WP-94-38
May 1994

IIASA International Institute for Applied Systems Analysis ▫ A-2361 Laxenburg ▫ Austria
Telephone: +43 2236 71521 ▫ Telex: 079 137 iiasa a ▫ Telefax: +43 2236 71313

Perturbation Methods for Saddle Point Computation

Markku Kallio and Andrzej Ruszczyński

WP-94-38
May 1994

IIASA International Institute for Applied Systems Analysis □ A-2361 Laxenburg □ Austria
Telephone: +43 2236 71521 □ Telex: 079 137 iiasa a □ Telefax: +43 2236 71313

Abstract

A general class of iterative methods for saddle point seeking is developed. The directions used are subgradients evaluated at perturbed points. Convergence of the methods is proved and alternative strategies for implementation are discussed. The procedure suggests scalable algorithms for solving large-scale linear programs via saddle points. For illustration, some encouraging tests with the standard Lagrangian of linear programs from the *Netlib* library are reported.

Key words: Saddle point, linear programming, convex programming.

Perturbation Methods for Saddle Point Computation

Markku Kallio and Andrzej Ruszczyński

1. Introduction

Let $L : R^n \times R^m \to R$ be a finite convex-concave function and let $X \subset R^n$ and $Y \subset R^m$ be closed convex sets. The objective of this paper is to develop a class of methods for finding a *saddle point* of L over $X \times Y$, i.e., a point $(\hat{x}, \hat{y}) \in X \times Y$ such that

$$L(\hat{x}, y) \le L(\hat{x}, \hat{y}) \le L(x, \hat{y}), \quad \forall x \in X, \ \forall y \in Y. \tag{1.1}$$

This is one of fundamental problems of convex programming and game theory (for a thorough treatment of the theory of saddle functions see Rockafellar [13]). There were many attempts to develop saddle point seeking procedures; the simplest algorithm (see, e.g., Arrow, Hurwicz and Uzawa [1]) has the form

$$\begin{aligned} x^{k+1} &= \left[x^k - \tau_k L_x(x^k, y^k)\right]_X \\ y^{k+1} &= \left[y^k + \tau_k L_y(x^k, y^k)\right]_Y, \quad k = 1, 2, \ldots, \end{aligned}$$

where $L_x(x^k, y^k)$ and $L_y(x^k, y^k)$ are some subgradients of L at (x^k, y^k) with respect to x and y, and $[\cdot]_X$ and $[\cdot]_Y$ denote orthogonal projections on X and Y, respectively. Such methods are convergent only under special conditions (like strict convexity-concavity) and with special stepsizes for primal and dual updates: $\tau_k \to 0$, $\sum_{k=0}^{\infty} \tau_k = \infty$ (cf. Nemirovski and Yudin [11]).

One possibility to overcome these difficulties is the use of the *proximal point method* introduced by Martinet [10] and further developed by Rockafellar [14]. Its idea is to replace (1.1) by a sequence of regularized saddle-point problems. A variation of this approach is the *alternating direction method* of Gabay [5] (see the recent analysis of Eckstein and Bertsekas [3]).

The main purpose of this paper is to develop a class of iterative methods for (1.1) which do not have saddle-point subproblems. The key idea, which unifies earlier works of ours [6, 7, 15], is to calculate the directions to be used at (x^k, y^k) at perturbed points (x^k, η^k) and (ξ^k, y^k), with appropriately generated ξ^k and η^k. We shall develop this concept in Section 2. In Section 3 strong convergence of the resulting methods to a saddle point of L is proved. In Section 4 some examples of applications are discussed. Finally, a numerical illustration is given to solve linear programs via saddle points of the standard Lagrangian.

The scalar product and the corresponding norm are denoted $\langle \cdot, \cdot \rangle$ and $\| \cdot \|$. We use $\partial_x L(x, y)$ and $\partial_y L(x, y)$ to denote the subdifferentials of $L(x, y)$ with respect to x and y. Elements of these subdifferentials (subgradients) will be denoted by $L_x(x, y)$ and $L_y(x, y)$. The set of all subsets of X is denoted 2^X.

2. The method

The methods that will be developed here are characterized by perturbation mappings $\xi : X \times Y \to 2^X$ and $\eta : X \times Y \to 2^Y$. We shall impose on them the following conditions.

(A1) The sets $\xi(x,y)$ and $\eta(x,y)$ are bounded on bounded subsets of $X \times Y$.

(A2) For every $(x,y) \in X \times Y$, if there is a sequence $\left\{(x^k, y^k)\right\} \subset X \times Y$ such that $(x^k, y^k) \to (x,y)$ and $L(x^k, \eta^k) - L(\xi^k, y^k) \to 0$ for some $\eta^k \in \eta(x^k, y^k)$ and $\xi^k \in \xi(x^k, y^k)$, then (x,y) is a saddle point of L on $X \times Y$.

In particular, these conditions are satisfied when the mappings ξ and η are bounded, have closed graphs and $L(x, \eta(x,y)) - L(\xi(x,y), y) = 0$ only at saddle points (x,y). In section 4 we discuss examples of mappings ξ and η that satisfy this condition.

Let us now describe in detail a method for finding a saddle point of L.

Initialization. Choose $x^0 \in X$, $y^0 \in Y$ and $\gamma \in (0,2)$. Set $k = 0$.

Perturbation. Find perturbed points $\eta^k \in \eta(x^k, y^k)$ and $\xi^k \in \xi(x^k, y^k)$.

Stopping test. Define the gap $E_k = L(x^k, \eta^k) - L(\xi^k, y^k)$. If $E_k = 0$, then stop.

Direction finding. Find subgradients $L_x(x^k, \eta^k)$ and $L_y(\xi^k, y^k)$ and define

$$\begin{aligned} d_x^k &= \left[-L_x(x^k, \eta^k)\right]_{C_X^k} \\ d_y^k &= \left[L_y(\xi^k, y^k)\right]_{C_Y^k}, \end{aligned}$$

where C_X^k and C_Y^k are closed convex cones containing the cones of feasible directions for x and y, respectively, at (x^k, y^k).

Update. Define

$$\begin{aligned} x^{k+1} &= \left[x^k + \tau_k d_x^k\right]_X \\ y^{k+1} &= \left[y^k + \tau_k d_y^k\right]_Y, \end{aligned}$$

where the stepsize

$$\tau_k = \frac{\gamma E_k}{\|d^k\|^2}. \tag{2.1}$$

Increase k by one and go to Perturbation.

As in [7], for practical definitions of the directions, the cones C_X and C_Y may be employed to take efficiently into account simple bounds (and polyhedral constraints) on vectors x and y, but we can always set $C_X = R^n$ and $C_Y = R^m$.

3. Convergence

Theorem 1. *Assume that a saddle point of L on $X \times Y$ exists. If the perturbation mappings satisfy conditions (A1)-(A2) then the method generates a sequence $\left\{(x^k, y^k)\right\}_{k=0}^{\infty}$ convergent to a saddle point of L on $X \times Y$.*

Proof. Let (x^*, y^*) be a saddle point of L on $X \times Y$. We define

$$W_k = \|x^k - x^*\|^2 + \|y^k - y^*\|^2. \tag{3.1}$$

Our proof, basically, uses the general line of argument developed for iterative methods based on abstract Fejér mappings (see Eremin and Astafiev [4]). We shall prove that our algorithmic mapping decreases the distance W_k whenever (x^k, y^k) is not a solution.

First of all, let us note that (A2) implies that the method can stop only at a saddle point. Therefore one can assume from now on that the sequence (x^k, y^k) is infinite.

Since the projection on X is non-expansive,

$$\begin{aligned}\|x^{k+1} - x^*\|^2 &\le \|x^k + \tau_k d_x^k - x^*\|^2 \\ &= \|x^k - x^*\|^2 + 2\tau_k \langle d_x^k, x^k - x^*\rangle + \tau_k^2 \|d_x^k\|^2.\end{aligned} \tag{3.2}$$

Since $x^* - x^k \in C_X^k$

$$\langle L_x(x^k, \eta^k) + d_x^k, x^* - x^k\rangle \ge 0.$$

By convexity,

$$\langle L_x(x^k, \eta^k), x^k - x^*\rangle \ge L(x^k, \eta^k) - L(x^*, \eta^k).$$

Employing the above inequalities in (3.2) yields

$$\|x^{k+1} - x^*\|^2 \le \|x^k - x^*\|^2 - 2\tau_k \left[L(x^k, \eta^k) - L(x^*, \eta^k)\right] + \tau_k^2 \|d_x^k\|^2.$$

Likewise, by obvious symmetry, one obtains

$$\|y^{k+1} - y^*\|^2 \le \|y^k - y^*\|^2 + 2\tau_k \left[L(\xi^k, y^k) - L(\xi^k, y^*)\right] + \tau_k^2 \|d_y^k\|^2.$$

Adding the last two inequalities one concludes that

$$W_{k+1} \le W_k - 2\tau_k \left[L(x^k, \eta^k) - L(x^*, \eta^k) - L(\xi^k, y^k) + L(\xi^k, y^*)\right] + \tau_k^2 \|d^k\|^2. \tag{3.3}$$

The saddle point conditions imply that $L(x^*, \eta^k) \le L(\xi^k, y^*)$. Therefore (3.3) implies:

$$W_{k+1} \le W_k - 2\tau_k E_k + \tau_k^2 \|d^k\|^2. \tag{3.4}$$

Substituting (2.1) one gets

$$W_{k+1} \le W_k - \frac{\gamma(2-\gamma) E_k^2}{\|d^k\|^2}. \tag{3.5}$$

Thus the sequence $\{W_k\}$ is non-increasing and

$$\lim_{k\to\infty} \frac{E_k^2}{\|d^k\|^2} = 0. \tag{3.6}$$

Since W_k is bounded, the sequence $\{(x^k, y^k)\}$ has an accumulation point $(\hat{x}, \hat{y})$. By (A1), $\{d^k\}$ is bounded and, by (3.6), $\lim_{k\to\infty} E_k = 0$. Therefore, by (A2), $(\hat{x}, \hat{y})$ is a saddle point of L and one can use it instead of (x^*, y^*) in (3.1). Then, from (3.5) one concludes that the distance to $(\hat{x}, \hat{y})$ is non-increasing. Consequently, $(\hat{x}, \hat{y})$ is the only accumulation point of the sequence $\{(x^k, y^k)\}$. □

4. Examples of perturbation

There are many ways of specifying perturbations which satisfy (A1) and (A2).

Example 1: *Gradient steps*

If L has Lipschitz continuous gradients, we can define the perturbations via gradient steps

$$\xi(x,y) = [x - \alpha \nabla_x L(x,y)]_X \tag{4.1}$$

$$\eta(x,y) = [y + \alpha \nabla_y L(x,y)]_Y \, , \tag{4.2}$$

with $\alpha \in (0, 1/\lambda)$, where λ is the Lipschitz constant of the gradients of L. Such a rule is also employed in [6, 9].

Example 2: *Gap maximization*

If the feasible sets X and Y are compact, we can define

$$\begin{aligned} \xi(x,y) &= \arg\min_{\xi \in X} L(\xi, y) \\ \eta(x,y) &= \arg\max_{\eta \in Y} L(x, \eta). \end{aligned}$$

This method can be then interpreted as a subgradient method [4] for minimizing the function

$$F(x,y) = \max_{(\xi,\eta) \in X \times Y} [L(x,\eta) - L(\xi,y)] \, .$$

Example 3: *Proximal mappings*

An alternative way, which does not require compactness, is to define the perturbations employing regularization:

$$\begin{aligned} \xi(x,y) &= \arg\min_{\xi \in X} \left[L(\xi, y) + \frac{\rho}{2} \|\xi - x\|^2 \right], \\ \eta(x,y) &= \arg\max_{\eta \in Y} \left[L(x, \eta) - \frac{\rho}{2} \|\eta - y\|^2 \right], \end{aligned}$$

where $\rho > 0$. This case is analysed in detail in [15], together with potential applications to decomposition methods in convex programming. Obviously, alternative regularizations may be employed.

Verification of (A1) and (A2) in each of these examples is easy and will therefore be omitted.

5. An application to linear programming

For a computational illustration, consider the standard linear programming problem [2]:

$$\min_{l \leq x \leq u} \{c^T x \mid Ax \hat{=} b\} \tag{5.1}$$

where $A \in R^{m \times n}$, $b \in R^m$, $x, c, l, u \in R^n$, and $\hat{=}$ refers to equations or inequalities. Define the Lagrangian function

$$L(x,y) = c^T x + y^T(b - Ax) \tag{5.2}$$

where $y \in R^m$ is the vector of dual multipliers. Let $X = \{x \in R^n \mid l \leq x \leq u\}$ and let $Y \subset R^m$ account for the sign constraints for the dual vector y. It is well known, that saddle points of L correspond to the optimal solutions of (5.1).

For implementing the method of Section 2, we employ (4.1) and (4.2) with $\alpha = 1$. For the update step, we define cones C_X^k and C_Y^k by simple bounds (l and u for the primal variables and sign constraints for the dual variables), which are active at (x^k, y^k).

An experimental computer code *Saddle* was developed by means of revising an earlier code reported in [8]. To begin the first iteration, we set primal and dual variables to zero and project onto the sets X and Y. The iterations end when the gap E_k relative to the absolute value of the objective function value is smaller than an optimality tolerance ϕ (we use $\phi = 10^{-6}$).

It is important to note, that both primal and dual variables can be processed in parallel. Simulations of parallel runs were performed in a HP9000/720 serial computer. Denoting by t_s the serial run time obtained with *Saddle* and assuming one processor for each column and row, a measure of the parallel run time is given by $t_p = t_s/(n+m)$. As communication time and the impact of an uneven distribution of tasks among processors is omitted, such parallel times should be regarded as (rather optimistic) potential run times in a massively parallel computer.

Problem	Rows	Columns	Iterations	Serial Time	Parallel Time	Relative Error
stocfor3	16675	15695	3564	736	0.023	6.E-05
80bau3b	2263	9799	4222	337	0.028	2.E-05
stocfor2	2158	2031	2386	64	0.015	2.E-04
degen3	1504	1818	17525	626	0.188	5.E-05
sctap3	1481	2480	613	17	0.004	2.E-04
pilot	1442	3652	25851	1492	0.292	4.E-03

Table 1: Number or rows and columns for *Netlib* test problems, number of iterations, serial and parallel run time (sec), and relative error in the objective function.

The six problems from the *Netlib* library [12] reported in [7] are also chosen for illustration here. Dimensions of these problems are given in Table 1. Iteration counts, serial and parallel run times, and relative errors in the optimal objective function value are given in Table 1 as well.

Acknowledgement

The authors are grateful to Yuri Ermoliev for bringing to their attention the close relationship of their results and the results of I.I. Eremin concerning Fejér sequences.

References

[1] K.J. Arrow, L. Hurwicz and H. Uzawa, *Studies in Linear and Nonlinear Programming* (Stanford University Press, Stanford, 1958).

[2] G.B.Dantzig, *Linear programming and extensions*, Princeton University Press, Princeton, 1963.

[3] J. Eckstein and D.P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming* 55 (1992) 293-318.

[4] I.I. Eremin and N.N. Astafiev, *Introduction to the Theory of Linear and Convex Programming* (Nauka, Moscow, 1976).

[5] D. Gabay, "Application de la méthode des multiplicateurs aux inéquations variationelles," in: M. Fortin and R. Glowinski (eds.), *Méthodes de Lagrangien Augmenté* (Dunod, Paris, 1982) pp. 279-307.

[6] M. Kallio, "A method for saddle point computation with application to linear programming," IIASA, Laxenburg, April 1994.

[7] M. Kallio and A. Ruszczyński, "Parallel solution of linear programs via Nash equilibria," Working Paper WP-94-15, IIASA, Laxenburg, 1994.

[8] M. Kallio, A. Ruszczyński and S. Salo, "A regularized Jacobi method for large-scale linear programming,'" Working Paper WP-93-61, IIASA, Laxenburg, 1993.

[9] G.M. Korpelevich, "The extragradient method for finding saddle points and other problems," *Ekonomika i Matematicheskie Metody* 12 (1976) 747-756.

[10] B. Martinet, "Regularisation d'inéquations variationelles par approximations successives," *Rev. Francaise Inf. Rech. Oper.* 4 (1970) 154-159.

[11] A.S. Nemirovski and D.B. Yudin, "Cesaro convergence of the gradient method for approximation of saddle points of convex-concave functions," *Doklady AN SSSR* 239 (1978) 1056-1059.

[12] Netlib, LP Test Problems, Bell Laboratories.

[13] R.T. Rockafellar, *Convex Analysis* (Princeton University Press, Princeton, 1970).

[14] R.T. Rockafellar, "Monotone operators and the proximal point algorithm," *SIAM J. Control and Optimization* 14 (1976) 977-898.

[15] A. Ruszczyński, "A partial regularization method for saddle point seeking", working paper WP-94-20, IIASA, Laxenburg, 1994.