Order-Value Optimization: Formulation and Solution by means of a Primal Cauchy Method

Roberto Andreani* Cibele Dunder† José Mario Martínez‡

December 19, 2002

Abstract

The Order-Value Optimization (OVO) problem is a generalization of the classical Minimax problem. Instead of the maximum of a set functions, the functional value that ranks in the $p-$th place is minimized. The problem seeks the application to (non-pessimistic) decision making and to model fitting in the presence of (perhaps systematic) outliers. A Cauchy-type method is introduced that solves the problem in the sense that every limit point satisfies an adequate optimality condition. Numerical examples are given.

Key words: Order-Value Optimization, iterative methods, global convergence, fitting parameters.

1 Introduction

Given m functions $f_1, \ldots, f_m$, defined in a domain $\Omega \subset \mathbb{R}^n$ and an integer $p \in \{1, \ldots, m\}$, the ($p-$) Order-Value (OVO) function f is given by

$$f(x) = f_{i_p(x)}(x),$$

*Department of Applied Mathematics, IMECC-UNICAMP, University of Campinas, CP 6065, 13081-970 Campinas SP, Brazil. This author was supported by FAPESP (Grant 01/05492-1) and CNPq (Grant 301115/96-6). E-Mail: andreani@ime.unicamp.br.

†Department of Applied Mathematics, IME-USP, University of Sao Paulo, CEP 05508-900 Sao Paulo, SP, Brazil. E-mail: dunder@ime.usp.br

‡Department of Applied Mathematics, IMECC-UNICAMP, University of Campinas, CP 6065, 13081-970 Campinas SP, Brazil. This author was supported by PRONEX, FAPESP (PT 2001-04597-4) E-Mail: martinez@ime.unicamp.br

where

$$f_{i_1(x)}(x) \leq f_{i_2(x)}(x) \leq \ldots \leq f_{i_p(x)}(x) \leq \ldots \leq f_{i_m(x)}(x).$$

If $p = 1$, $f(x) = \min\{f_1(x), \ldots, f_m(x)\}$ whereas for $p = m$ we have that $f(x) = \max\{f_1(x), \ldots, f_m(x)\}$.

We will show later that f is continuous. However, even if the functions f_i are differentiable, the OVO function is not smooth.

The OVO problem consists in the minimization of the Order-Value function:

$$\text{Minimize } f(x) \;\; s.t. \;\; x \in \Omega. \tag{1}$$

The definition of the OVO problem was motivated by two main applications.

(1) Assume that Ω is a space of decisions and, for each $x \in \Omega$, $f_i(x)$ represents the cost of decision x under the scenario i. The Minimax decision corresponds to choose x in such a way that the maximum possible cost is minimized. This is a very pessimistic alternative and decision-makers usually prefer to discard the worst possibilities in order to proceed in a more realistic way. For example, the decision maker may want to discard the 10 % more pessimistic scenarios. This corresponds to minimize the $p-$Order-Value function with $p \approx 0.9m$.

(2) Assume that we have a parameter-estimation problem where the space of parameters is Ω and $f_i(x)$ is the error in the observation i when we adopt the parameter x. The Minimax estimation problem corresponds to minimize the maximum error. As it is well-known this estimate is very sensitive to the presence of outliers. Many times, we want to eliminate (say) the 15% larger errors because the can represent wrong observations. This corresponds to minimize the $p-$Order-Value function with $p \approx 0.85 \times m$. The OVO strategy is especially designed to eliminate the influence of systematic errors.

In this paper we introduce a steepest descent type method for solving (1). We say that this method is "primal" because no auxiliar or dual variables are involved in it. The algorithm is presented in Section 2, where a convergence theorem is also given. In Section 3 we describe a way to solve approximately the subproblems, satisfying the theoretical requirements. Numerical examples are shown in Section 4. Conclusions are given in the last section of this paper.

2 The algorithm

Assume that $\Omega \subset I\!R^n$ is closed and convex, and $f_1, \ldots, f_m$ have continuous partial derivatives in an open set that contains Ω. We denote $g_j = \nabla f_j$ from now on.

For all $x, y \in \Omega$, $j = 1, \ldots, m$, we assume that

$$\|g_j(x)\|_\infty \leq c,$$

and

$$\|g_j(y) - g_j(x)\|_\infty \leq L\|y - x\|_\infty.$$

Consequently, for all $x, y \in \Omega$, $j = 1, \ldots, m$,

$$|f_j(y) - f_j(x)| \leq c\|y - x\|_\infty \tag{2}$$

and

$$f_j(y) \leq f_j(x) + g_j(x)^T(y - x) + \frac{L}{2}\|y - x\|_\infty^2. \tag{3}$$

Given $\varepsilon \geq 0$, $x \in \Omega$, we define

$$I_\varepsilon(x) = \{j \in \{1, \ldots, m\} \mid f(x) - \varepsilon \leq f_j(x) \leq f(x) + \varepsilon\}. \tag{4}$$

The following theorem shows the continuity of the OVO-function.

Theorem 2.1. *The $p-$order-value function f is continuous.*

Proof. Assume that $x^k \to x$, and suppose that, for all k in some subsequence,

$$|f(x^k) - f(x)| \geq \delta > 0. \tag{5}$$

For that subsequence, there exists an index $j \in \{1, \ldots, m\}$ such that

$$f(x^k) = f_j(x^k)$$

infinitely many times. Therefore,

$$f_j(x^k) \geq f_\ell(x^k) \tag{6}$$

for at least p indices $\ell \in \{1, \ldots, m\}$. Moreover,

$$f_j(x^k) \leq f_\ell(x^k) \tag{7}$$

for at least $m-p+1$ indices ℓ in the set $\{1,\ldots,m\}$. Since the number of subsets of $\{1,\ldots,m\}$ is finite, a set of indices ℓ that verify (6) is repeated infinitely many times, and the same happens with a set of indices ℓ that verify (7). Therefore, taking limits in (6) and (7), we obtain that

$$f_j(x) \geq f_\ell(x)$$

for at least p indices $\ell \in \{1,\ldots,m\}$ and

$$f_j(x) \leq f_\ell(x)$$

for at least $m-p+1$ indices $\ell \in \{1,\ldots,m\}$. Therefore,

$$f(x) = f_j(x).$$

But $f_j(x^k) \to f_j(x)$, so this contradicts (5). $\square$

The theorem below provides a necessary optimality condition for the OVO problem.

Definition. We say that x is $\varepsilon-$ optimal if

$$\mathcal{D} \equiv \{d \in I\!R^n \mid x+d \in \Omega \text{ and } g_j(x)^T d < 0 \;\; \forall\, j \in I_\varepsilon(x)\} = \emptyset. \tag{8}$$

Theorem 2.2. *If $x_* \in \Omega$ is a local minimizer of $f(x)$ subject to $x \in \Omega$ and $\varepsilon \geq 0$, then x_* is $\varepsilon-$ optimal.*

Proof. Suppose, by contradiction, that (8) is not true. Then, there exists $d \in I\!R^n$ and $\bar{\alpha} > 0$ such that $x_* + d \in \Omega$ and

$$f_j(x_* + \alpha d) < f_j(x_*) \;\; \forall \;\; \alpha \in (0, \bar{\alpha}] \tag{9}$$

for all $j \in I_\varepsilon(x_*)$.

So, for all $j \in \{1,\ldots,m\}$ with $f_j(x_*) = f(x_*)$, one has that $j \in I_\varepsilon(x_*)$ and then one gets, under the supposition of the proof, that (9) holds.

Define

$$\varepsilon_1 = \min_{f_j(x_*) < f(x_*)} \{f(x_*) - f_j(x_*)\} \in (0, \infty],$$

$$\varepsilon_2 = \min_{f_j(x_*) > f(x_*)} \{f_j(x_*) - f(x_*)\} \in (0, \infty],$$

Let $\tilde{\alpha} \leq \bar{\alpha}$ be such that

$$|f_j(x_* + \alpha d) - f_j(x_*)| < \frac{\min\{\varepsilon_1, \varepsilon_2\}}{2}$$

for all $j = 1, \ldots, m$, $\alpha \in (0, \tilde{\alpha}]$.

Therefore, for all $\alpha \in (0, \tilde{\alpha}]$, the index j such that

$$f(x_* + \alpha d) = f_j(x_* + \alpha d)$$

is one of the indices j such that $f(x_*) = f_j(x_*)$. In other words, this index belongs to $I_0(x_*) \subset I_\varepsilon(x_*)$. But, by (9),

$$f_j(x_* + \alpha d) < f_j(x_*)$$

for all $\alpha \in (0, \tilde{\alpha}]$. Therefore, x_* is not a local minimizer. $\square$

Below we describe the main algorithm. It is a primal method in the sense that only primal variables x are manipulated and updated at each iteration. The algorithm finds a decreasing sequence of functional values using search directions that come from the (inexact) resolution of a convex programming problem.

Algorithm 2.1

Let $x_0 \in \Omega$ an arbitrary initial point. Let $\theta \in (0, 1)$, $\Delta > 0$, $\varepsilon > 0$, $0 < \sigma_{min} < \sigma_{max} < 1$, $\eta \in (0, 1]$.

Given $x_k \in \Omega$ the steps of the $k-$th iteration are:

Step 1. (Solving the subproblem)

Define

$$M_k(d) = \max_{j \in I_\varepsilon(x_k)} g_j(x_k)^T d. \tag{10}$$

Consider the subproblem

$$\text{Minimize } M_k(d) \text{ s.t. } x_k + d \in \Omega, \ \|d\|_\infty \leq \Delta. \tag{11}$$

Note that (11) is equivalent to the convex optimization problem

$$\text{Minimize } w$$

$$g_j(x_k)^T d \leq w \ \forall \ j \in I_\varepsilon(x_k),$$

$$x_k + d \in \Omega, \quad \|d\|_\infty \le \Delta.$$

Let $\bar{d}_k$ be a solution of (11). (We will see later that we do not need to compute it.) Let d_k be such that $x_k + d_k \in \Omega$, $\|d_k\| \le \Delta$ and

$$M_k(d_k) \le \eta M_k(\bar{d}_k). \tag{12}$$

If $M_k(d_k) = 0$ stop.

Step 2. (Steplength calculation)

Set $\alpha \leftarrow 1$.

If

$$f(x_k + \alpha d_k) \le f(x_k) + \theta \alpha M_k(d_k) \tag{13}$$

set $\alpha_k = \alpha$, $x_{k+1} = x_k + \alpha_k d_k$ and finish the iteration. Otherwise, choose $\alpha_{new} \in [\sigma_{min}\alpha, \sigma_{max}\alpha]$, set $\alpha \leftarrow \alpha_{new}$ and repeat the test (13).

The following is a technical lemma which will be useful in the convergence proof.

Lemma 2.1. *Assume that $a_1, \ldots, a_r$ are real numbers such that*

$$a_1 \le a_2 \le \ldots \le a_q \le \ldots \le a_r.$$

Suppose that $\beta > 0$ and $b_1, \ldots, b_r \in I\!R$ are such that

$$b_j \le a_j - \beta \;\; \forall\, j = 1, \ldots, r$$

and

$$b_{i_1} \le b_{i_2} \le \ldots \le b_{i_q} \le \ldots \le b_{i_r}.$$

Then,

$$b_{i_q} \le a_q - \beta.$$

Proof. Clearly,

$$b_{i_q} \le a_{i_q} - \beta,\; b_{i_q} \le b_{i_{q+1}} \le a_{i_{q+1}} - \beta, \ldots, b_{i_q} \le b_{i_r} \le a_{i_r} - \beta.$$

Therefore,

$$b_{i_q} \le \min\{a_{i_{q+1}}, \ldots, a_{i_r}\} - \beta.$$

But, since $a_1 \le \ldots \le a_q \le \ldots \le a_r$, we have that

$$\min\{a_{i_{q+1}}, \ldots, a_{i_r}\} \le a_q.$$

Therefore,

$$b_{i_q} \leq a_q - \beta$$

as we wanted to prove. □

In the following theorem, we state that, if the iterate x_k is not ε-optimal, then x_{k+1} is well defined and α_k is bounded away from zero.

Theorem 2.3. *Assume that $x_k \in \Omega$ is the $k-th$ iterate of Algorithm 2.1. Then:*

(a) The algorithm stops at x_k if, and only if, x_k is $\varepsilon-$ optimal.

(b) If the algorithm does not stop at x_k, then the iteration is well-defined and

$$\alpha_k \geq \min\{\frac{2\sigma_{min}\gamma_k(1-\theta)}{L\Delta^2}, \frac{\varepsilon\sigma_{min}}{3c\Delta}\}, \tag{14}$$

where

$$\gamma_k = -\max_{j \in I_\varepsilon(x_k)} \{g_j(x_k)^T d_k\} > 0. \tag{15}$$

Proof. If the algorithm stops at x_k, then $M_k(d_k) = 0$. Therefore, by (12), $M_k(\bar{d}_k) = 0$. So, $M_k(d) \geq 0$ for all $d \in \mathcal{D}$ such that $\|d\|_\infty \leq \Delta$. Thus, $M_k(d) \geq 0$ for all $d \in \mathcal{D}$. This implies that x_k is ε-optimal.

Reciprocally, if x_k is ε-optimal, we must have that $M_k(\bar{d}_k) = 0$, so $M_k(d_k) = 0$ and the algorithm stops at x_k.

If the algorithm does not stop at x_k, then $M_k(d_k) < 0$. Therefore,

$$-\gamma_k = \max_{j \in I_\varepsilon(x_k)} \{g_j(x_k)^T d_k\} < 0.$$

Assume that

$$\alpha \in [0, \frac{2\gamma_k(1-\theta)}{L\Delta^2}].$$

Then,

$$\frac{L\alpha\Delta^2}{2} \leq (1-\theta)\gamma_k.$$

So,

$$\frac{L\alpha\Delta^2}{2} \leq (\theta - 1)g_j(x_k)^T d_k \ \ \forall \ \ j \in I_\varepsilon(x_k).$$

Therefore,

$$g_j(x_k)^T d_k + \frac{L\alpha\Delta^2}{2} \leq \theta g_j(x_k)^T d_k \ \ \forall \ \ j \in I_\varepsilon(x_k).$$

Therefore, since $\|\alpha d_k\|_\infty \leq \Delta$,

$$\alpha g_j(x_k)^T d_k + \frac{L\alpha^2 \|d_k\|_\infty^2}{2} \leq \alpha\theta g_j(x_k)^T d_k \;\; \forall \;\; j \in I_\varepsilon(x_k).$$

So,

$$f_j(x_k) + g_j(x_k)^T(\alpha d_k) + \frac{L}{2}\|\alpha d_k\|_\infty^2 \leq f_j(x_k) + \alpha\theta g_j(x_k)^T d_k \;\; \forall \;\; j \in I_\varepsilon(x_k).$$

Therefore, by (3)

$$f_j(x_k + \alpha d_k) \leq f_j(x_k) + \alpha\theta g_j(x_k)^T d_k \;\; \forall \;\; j \in I_\varepsilon(x_k).$$

So, we have proved that, if $\alpha \in [0, 2\gamma_k(1-\theta)/(L\Delta^2)]$,

$$f_j(x_k + \alpha d_k) \leq f_j(x_k) + \alpha\theta M_k(d_k) \;\; \forall \;\; j \in I_\varepsilon(x_k). \tag{16}$$

On the other hand, if $\alpha \in [0, \varepsilon/(3c\Delta)]$, we have that $\alpha c\Delta \leq \varepsilon/3$, so $\alpha c\|d_k\|_\infty \leq \varepsilon/3$, so $c\|\alpha d_k\|_\infty \leq \varepsilon/3$, therefore, by (2),

$$|f_j(x_k + \alpha d_k) - f_j(x_k)| \leq \frac{\varepsilon}{3} \;\; \forall \;\; j = 1, \ldots, m. \tag{17}$$

Therefore, for all $\ell = 1, \ldots, p$,

$$f_{i_\ell(x_k)}(x_k + \alpha d_k) \leq f_{i_\ell(x_k)}(x_k) + \frac{\varepsilon}{3} \leq f(x_k) + \frac{\varepsilon}{3}$$

and, for all $\ell = p, \ldots, m$,

$$f_{i_\ell(x_k)}(x_k + \alpha d_k) \geq f_{i_\ell(x_k)}(x_k) - \frac{\varepsilon}{3} \geq f(x_k) - \frac{\varepsilon}{3}.$$

This means that at least p elements of the set

$$\{f_1(x_k + \alpha d_k), \ldots, f_m(x_k + \alpha d_k)\}$$

are less than or equal to $f(x_k) + \varepsilon/3$ and that at least $m - p + 1$ elements of that set are greater than or equal to $f(x_k) - \varepsilon/3$.

Therefore,

$$f(x_k + \alpha d_k) = f_{i_p(x_k + \alpha d_k)}(x_k + \alpha d_k) \in [f(x_k) - \frac{\varepsilon}{3}, f(x_k) + \frac{\varepsilon}{3}] \tag{18}$$

Suppose that $j \notin I_\varepsilon(x_k)$. So, either $f_j(x_k) < f(x_k) - \varepsilon$ or $f_j(x_k) > f(x_k) + \varepsilon$. In the first case, by (17), we have that

$$f_j(x_k + \alpha d_k) < f(x_k) - \frac{2}{3}\varepsilon,$$

so, by (18),

$$f_j(x_k + \alpha d_k) < f(x_k + \alpha d_k).$$

Analogously, if $f_j(x_k) > f(x_k) + \varepsilon$, then

$$f_j(x_k + \alpha d_k) > f(x_k + \alpha d_k).$$

Therefore,

$$f(x_k + \alpha d_k) = f_j(x_k + \alpha d_k) \text{ for some } j \in I_\varepsilon(x_k).$$

Let us write

$$I_\varepsilon(x_k) = \{j_1, \ldots, j_\nu\} = \{j'_1, \ldots, j'_\nu\},$$

where

$$f_{j_1}(x_k) \leq \ldots \leq f_{j_\nu}(x_k)$$

and

$$f_{j'_1}(x_k + \alpha d_k) \leq \ldots \leq f_{j'_\nu}(x_k + \alpha d_k).$$

Clearly, there exists $q \in \{1, \ldots, \nu\}$ such that

$$i_p(x_k) = j_q.$$

Now, the indices $j \notin I_\varepsilon(x_k)$ such that $f_j(x_k) < f(x_k)$ are the same as the indices $j \notin I_\varepsilon(x_k)$ such that $f_j(x_k + \alpha d_k) < f(x_k)$ and, moreover, the indices $j \notin I_\varepsilon(x_k)$ such that $f_j(x_k) > f(x_k)$ are the same as the indices $j \notin I_\varepsilon(x_k)$ such that $f_j(x_k + \alpha d_k) > f(x_k)$. Then,

$$i_p(x_k + \alpha d_k) = j'_q.$$

Then, by (16) and Lemma 2.1, we have that, when

$$\alpha \in [0, \min\{\frac{2\gamma_k(1-\theta)}{L\Delta^2}, \frac{\varepsilon}{3c\Delta}\}] \tag{19}$$

we have:

$$f_{j'_q}(x_k + \alpha d_k) \leq f_{j_q}(x_k) + \alpha\theta M_k(d_k).$$

Therefore,

$$f_{i_p(x_k + \alpha d_k)}(x_k + \alpha d_k) \leq f_{i_p(x_k)}(x_k) + \alpha\theta M_k(d_k).$$

So,

$$f(x_k + \alpha d_k) \leq f(x_k) + \alpha\theta M_k(d_k).$$

Therefore, whenever (19) takes place, the test (13) must hold. This means that a value of α that does not satisfy (13) cannot be smaller than $\min\{2\gamma_k(1-\theta)/(L\Delta^2), \varepsilon/(3c\Delta)\}$. So, the accepted α must satisfy:

$$\alpha_k \geq \min\{\frac{2\sigma_{min}\gamma_k(1-\theta)}{L\Delta^2}, \frac{\varepsilon\sigma_{min}}{3c\Delta}\},$$

as we wanted to prove. □

The main convergence result is given in Theorem 2.4. First, we need to prove a simple preparatory lemma.

Lemma 2.2. *If $\{x_k\}$ is an infinite sequence generated by Algorithm 2.1, then either*

$$\lim_{k\to\infty} f(x_k) = -\infty \tag{20}$$

or

$$\lim_{k\to\infty} M_k(d_k) = 0. \tag{21}$$

Proof. If (20) does not hold, then, by (13), we have that

$$\lim_{k\to\infty} \alpha_k M_k(d_k) = 0.$$

By Theorem 2.3, this implies that (21) takes place. □

Theorem 2.4. *Suppose that $x_* \in \Omega$ is a limit point of a sequence generated by Algorithm 2.1. Then x_* is ε-optimal.*

Proof. Since $f(x_{k+1}) \leq f(x_k)$ for all $k = 0, 1, 2, \ldots$ and x_* is a limit point of $\{x_k\}$ then $f(x_k) \to f(x_*)$. Therefore, by Lemma 2.2, (21) takes place. Let K be an infinite sequence of indices such that

$$\lim_{k\in K} x_k = x_*.$$

If x_* is not ε-optimal, then there exists $\gamma > 0$ and $d \in I\!R^n$ such that $x_* + d \in \Omega$ and

$$g_j(x_*)^T d \leq -\gamma \ \ \forall \ \ j \in I_\varepsilon(x_*). \tag{22}$$

Without loss of generality we may assume that $\|d\| \leq \Delta/2$. By continuity, for k large enough, $k \in K$, defining

$$\hat{d}_k = d + x_* - x_k,$$

we have that $\|\hat{d}_k\|_\infty \leq \Delta$, $x_k + \hat{d}_k \in \Omega$ and

$$\lim_{k \in K} \hat{d}_k = d \tag{23}$$

By (21), we have that $\lim_{k\to\infty} M_k(\bar{d}_k) = 0$. Therefore, $\liminf_{k\to\infty} M_k(\hat{d}_k) \geq 0$. For all $k \in K$ there exists $j \in I_\varepsilon(x_k)$ such that $g_j(x_k)^T \hat{d}_k = M_k(\hat{d}_k)$. Since $I_\varepsilon(x_k)$ is finite, there exists j such that $g_j(x_k)^T \hat{d}_k = M_k(\hat{d}_k)$ infinitely many times. Therefore, for that particular j,

$$\liminf_{k \in K} g_j(x_k)^T \hat{d}_k = 0.$$

Therefore, taking limits,

$$g_j(x_*)^T d = 0. \tag{24}$$

But, for infinitely many indices, since $j \in I_\varepsilon(x_k)$,

$$f(x_k) - \varepsilon \leq f_j(x_k) \leq f(x_k) + \varepsilon,$$

therefore, taking limits,

$$f(x_*) - \varepsilon \leq f_j(x_*) \leq f(x_*) + \varepsilon,$$

so $j \in I_\varepsilon(x_*)$.

Therefore, (24) contradicts (22). □

3 Computing an approximate solution of the subproblem

If n is large or $I_\varepsilon(x_k)$ contains many elements, computing the exact solution of the (11) can be very costly. In these cases, an approximate solution d_k that satisfies (12) can be computed following the procedure described below. To simplify the description we assume that $Int(\Omega)$, the interior of the convex set Ω, is not empty. Choose $\eta' \in (\eta, 1)$. The successive iterates will x_k belong to $Int(\Omega)$. We assume that a procedure exists that computes a sequence $\{s_\nu\}$, where $s_\nu \in I\!R^n$, such that $s_\nu \to \bar{d}_k$ and a sequence of bounds c_ν such that $c_\nu \to M_k(\bar{d}_k)$ and

$$c_\nu \le M_k(\bar{d}_k)$$

for all $\nu = 0, 1, 2, \ldots$.

Define, for all $\nu = 0, 1, 2, \ldots$,

$$\lambda_\nu = \min\{1, \max\{\lambda \ge 0 \mid x_k + \lambda s_\nu \in \Omega, \ \|\lambda s_\nu\|_\infty \le \Delta\}\}.$$

Since Ω is convex and x_k is interior we have that

$$\lambda_\nu \to 1,$$

so,

$$x_k + \lambda_\nu s_\nu \to x_k + \bar{d}_k$$

and

$$M_k(\lambda_\nu s_\nu) \to M_k(\bar{d}_k).$$

Therefore, taking ν large enough, we get λ_ν such that

$$M_k(\lambda_\nu s_\nu) \le \eta' c_\nu \le \eta' M_k(\bar{d}_k). \tag{25}$$

So,

$$M_k(\frac{\eta}{\eta'}\lambda_\nu s_\nu) \le \eta M_k(\bar{d}_k).$$

This implies that, taking

$$d_k = \frac{\eta}{\eta'}\lambda_\nu s_\nu,$$

the condition (12) is satisfied, $x_k + \alpha d_k$ remains interior for all $\alpha \in [0, 1]$. So, x_{k+1} is interior and the process can be repeated in the next iteration.

4 Numerical example

To illustrate the behavior of the OVO approach, we consider here a fitting problem. We wish to fit the model

$$y(t, x) = x_1 + x_2 t + x_3 t^2 + x_4 t^3$$

to a set of data (t_i, y_i), $i = 1, \ldots, m$. The function to be minimized is obtained by the set of error functions

$$f_i(x) = (y(t_i, x) - y_i)^2$$

Given the "solution"

$$x^* = (x_1^*, x_2^*, x_3^*, x_4^*) = (0, 2, -3, 1),$$

we generate the data by

$$w_i = y(t_i, x^*),$$

$$t_i = -1 + 0.1i, \quad i = 0, \ldots, m, \quad m = 46,$$

$$y_i = 10 \quad \text{if} \quad i = 7, \ldots, 16,$$

$$y_i = w_i + r_i, \ \text{otherwise} \ ,$$

where r_i is random between -0.01 and 0.01. Therefore, $y_7, \ldots, y_{26}$ simulate wrong observations, or outliers. In Table 1 we give the data (t_i, y_i).

We consider the initial point $x = (-1.0, -2.0, 1.0, -1.0)$. Using the Matlab nonlinear least squares solvers POLYFIT and LSQNONLIN (using the bounds given below), we obtained the approximate solutions

$$x_{PFIT} = (6.4602, 2.7072, -7.5418, 2.1604)$$

and

$$x_{NLLSQ} = (6.4570, 2.7048, -7.5364, 2.1590),$$

which are quite far from x_*. We ran Algorithm 2.1, with the parameters

$$\Delta = 1.0, \quad \theta = 0.5, \quad \sigma_{min} = 0.1, \quad \sigma_{max} = 0.9, \quad \eta = 1.0, \quad \varepsilon = 0.001.$$

and the bounds

$$-10 \leq x_i \leq 10, \quad i = 1, 2, 3, 4.$$

The solutions obtained by the OVO algorithm for different values of p are shown in Table 2, where p define the p-order function, $n.iter$ is the number of iterations, $fobj$ is the value of the objective function. Observe that the "correct" p should be 36, because we generated 10 outliers.

The results were coherent and satisfactory. As expected, for $p > 36$ the solution given by the algorithm was a point far from x_* and the objective function value (OVO function) was large. For $p = 27$ and $p = 34$ the solution obtained was not a global minimizer of the OVO function. This is not surprising, since the algorithm is guaranteed to obtain only stationary points, therefore it can get nonglobal critical points in some cases. For all the other tested values of p the objective function value obtained was less than 0.05 and the solution obtained was close to x_*. The example suggests that in real-life situations, when one does not know the number of outliers, different values of p should be tested with different initial points and the solution must be accepted taking into account the values of the OVO function and p.

i	t_i	y_i	i	t_i	y_i	i	t_i	y_i
1	-1.0000	-5.8000	17	0.6000	0.5360	33	2.2000	0.3280
2	-0.9000	-5.1590	18	0.7000	0.4730	34	2.3000	1.0970
3	-0.8000	-4.2320	19	0.8000	-0.0080	35	2.4000	1.1440
4	-0.7000	-3.4130	20	0.9000	0.2990	36	2.5000	1.6750
5	-0.6000	-2.6960	21	1.0000	0.2000	37	2.6000	2.2960
6	-0.5000	-1.6750	22	1.1000	-0.2990	38	2.7000	3.4130
7	-0.4000	10.0000	23	1.2000	0.0080	39	2.8000	4.2320
8	-0.3000	10.0000	24	1.3000	-0.0730	40	2.9000	5.1590
9	-0.2000	10.0000	25	1.4000	-0.5360	41	3.0000	6.2000
10	-0.1000	10.0000	26	1.5000	-0.5750	42	3.1000	6.9610
11	0	10.0000	27	1.6000	-0.5840	43	3.2000	8.6480
12	0.1000	10.0000	28	1.7000	-0.5570	44	3.3000	9.6670
13	0.2000	10.0000	29	1.8000	-0.4880	45	3.4000	11.2240
14	0.3000	10.0000	30	1.9000	0.0290	46	3.5000	12.9250
15	0.4000	10.0000	31	2.0000	-0.2000			
16	0.5000	10.0000	32	2.1000	0.0310			

Table 1: Data for polynomial fitting with 4 parameters.

p	x_1	x_2	x_3	x_4	$fobj$	$n.iter$
20	-0.0060	1.9892	-2.9568	0.9748	0.0404	33
21	-0.1231	1.2809	-2.4353	0.8965	0.0369	47
22	0.3957	2.3052	-3.6787	1.1742	0.0297	85
23	0.1116	1.8346	-2.9223	0.9883	0.0401	49
24	-0.0081	2.0035	-2.9897	0.9943	0.0405	41
25	0.1033	1.8469	-2.9291	0.9896	0.0402	67
26	0.0261	1.9557	-2.9762	0.9958	0.0403	70
27	9.0443	-7.4778	-1.4899	1.2465	2.9491	78
28	-0.0190	2.0336	-3.0177	1.0028	0.0405	49
29	0.0015	1.9917	-2.9939	0.9989	0.0407	52
30	0.0096	1.9841	-2.9919	0.9987	0.0403	68
31	0.0030	2.0024	-3.0044	1.0011	0.0407	57
32	0.0001	2.0030	-3.0010	0.9999	0.0408	66
33	0.0021	2.0007	-3.0024	1.0007	0.0407	49
34	7.8217	5.8390	-10.0000	2.4904	14.1827	158
35	-0.0003	2.0004	-2.9997	0.9999	0.0403	56
36	0.0000	2.0003	-3.0002	1.0000	0.0403	67
37	7.3236	1.6767	-8.3264	2.4949	13.0175	187
38	9.6125	-9.6561	-0.5977	1.1134	10.9439	138
39	10.0000	-6.4565	-3.4642	1.6757	15.0919	130
40	6.2406	2.2166	-8.3576	2.4577	16.5412	295
41	6.2025	2.5359	-8.6241	2.5104	17.1385	283
42	6.3137	1.7981	-8.3196	2.4870	19.3482	207
43	7.7660	2.3983	-9.6980	2.8103	24.2539	152

Table 2: OVO solution for polynomial fitting with 4 parameters.

5 Conclusions

We have introduced the Order-Value Optimization problem, a continuous, nonsmooth and, in general, nonconvex optimization problem that applies to fitting models and, very likely, to decision making. A tentative algorithm, the implementation of which generally relies on Linear Programming subproblems has been defined which seems to work satisfactorily in computer generated problems.

The OVO algorithm has a good prospect of dealing with robust fitting problems. Many regression schemes, less sensitive to erroneous data than ordinary least squares, have been introduced in the statistical literature. See, for example, [4], [5] and references therein. A usual approach is to weight the differences by means of a filter that penalizes outliers. This requires some previous knowledge about the data that should be penalized. We think that the OVO algorithm could play the role of a universal filter in the sense that the user needs not know beforehand any kind of qualitative information about the outliers, but only inform the algorithm with a rough estimate of their cardinality in the original sample. The calibration of the model would depend only on the choice of a single 1-dimensional parameter, the whole penalization being done internally by OVO.

On the other hand, the OVO approach can also be used only to detect outliers, leaving the estimation procedure to other algorithm, after the elimination of the detected wrong measurements.

Finally, Order-Value optimization seems to be a challenging optimization problem which generalizes smooth optimization. A promising field of research is to adapt classical optimization algorithms (see, for example, [3]) to the OVO problem. The algorithm presented in this paper is a generalization of the steepest descent method, which, as it is well known, is the more classical algorithm for smooth optimization.

In a parallel research [1] we introduced a smooth reformulation of the OVO problem. This reformulation involves additional auxiliar and dual variables and is related to smooth reformulations of the minimax problem. See, for example, [2]. In general, both the primal method introduced in this paper and the smooth reformulation get the same solutions of the OVO problem. Using the reformulation we get, in theory, convergence of nonlinear programming algorithms to stricter stationary points than the ones defined in this paper. However, the complexity of algorithms based on the reformulation is increased.

Acknowledgement. We are indebted to an anonymous referee whose comments helped us to improve the final version of this paper.

References

[1] R. Andreani, C. Dunder and J. M. Martínez, *Nonlinear programming reformulation of the Order-Value Optimization problem*, Technical Report, Institute of Mathematics, University of Campinas, Brazil, 2001.

[2] G. Di Pillo, L. Grippo and S. Lucidi, *A smooth method for the finite minimax problem*, Mathematical Programming 60, 187-214, 1993.

[3] R. Fletcher, *Practical Methods of Optimization*, New York: Wiley, 1988.

[4] Huber, P.J., *Robust Statistics*, New York: Wiley, 1981.

[5] P. W. Holland and R. E. Welsch, *Robust regression using iteratively reweighted least-squares*, Communications in Statistics: Theory and Methods, A6, 813-827, 1977.