Robust stopping criteria for Dykstra's algorithm

Ernesto G. Birgin *   Marcos Raydan †

May 17, 2004

Abstract

Dykstra's algorithm is a suitable alternating projection scheme for solving the optimization problem of finding the closest point to a given one in the intersection of a finite number of closed and convex sets. It has been recently used in a wide variety of applications. However, in practice, the commonly used stopping criteria are not robust and could stop the iterative process prematurely at a point that does not solve the optimization problem. In this work we present a counter-example to illustrate the weakness of the commonly used criteria, and then we develop robust stopping rules. Additional experimental results are shown to illustrate the advantages of this new stopping criteria, including their associated computational cost.

Key words: Convex optimization, alternating projection methods, Dykstra's algorithm, stopping criteria.

1 Introduction

We consider Dykstra's algorithm for solving the optimization problem

$$\min_{x \in \Omega} \|x^0 - x\|, \tag{1}$$

where x^0 is a given point, Ω is a closed and convex set, and $\|z\|^2 = \langle z, z \rangle$ defines a real inner product in the space. The solution x^* is called the projection of x^0 onto Ω and is denoted by $P_\Omega(x^0)$. Dykstra's algorithm for solving (1) has been extensively studied since it fits in many different applications (see [1, 2, 4, 8, 9, 11, 12, 13, 18, 21, 23, 24, 26, 28, 29]).

Here, we consider the case

$$\Omega = \cap_{i=1}^{p} \Omega_i, \tag{2}$$

*Department of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, Ruga do Matão 1010 Cidade Universitária, 05508-090 São Paulo, SP - Brazil (egbirgin@ime.usp.br). Sponsored by FAPESP (Grants 01/04597-4 and 02/00094-0), CNPq (Grant 300151/00-4) and Pronex.

†Departamento de Computación, Facultad de Ciencias, Universidad Central de Venezuela, Ap. 47002, Caracas 1041-A, Venezuela (mraydan@reacciun.ve). Sponsored by the Center of Scientific Computing at UCV.

where Ω_i are closed and convex sets in $I\!R^n$, for $i = 1, 2, \ldots, p$, and $\Omega \neq \emptyset$. Moreover, we assume that for any $z \in I\!R^n$ the calculation of $P_\Omega(z)$ is not trivial; whereas, for each Ω_i, $P_{\Omega_i}(z)$ is easy to obtain as in the case of a box, an affine subspace, or a ball.

Roughly speaking, Dykstra's algorithm [2, 10] projects in a clever way onto each of the convex sets individually to complete a cycle which is repeated iteratively. We are mainly concerned with the criterion to stop the process within a certain previously established tolerance that indicates the distance of the current iterate to the unique solution.

This paper is organized as follows. In Section 2 we describe Dykstra's alternating projection method for solving (1)–(2), and discuss some of its properties. In Section 3 we discuss the difficulties with the typical and somehow informal stopping criteria that are frequently associated with Dykstra's algorithm. In Section 4 we introduce and analyze the new stopping criteria. In Section 5 we draw some conclusions.

2 Dykstra's algorithm

A suitable tool for solving (1) when Ω has the form (2) is Dykstra's alternating projection algorithm [2, 10], that will be described below. Dykstra's algorithm can also be obtained via duality [14, 20]. See also Hildreth [22], for the pioneer version on dual alternating projection methods for half spaces. Hildreth's algorithm has been extended for quadratic programming problems [24, 27].

Let us recall that for a given nonempty closed and convex set Ω of $I\!R^n$, and any $x^0 \in I\!R^n$, there exists a unique solution x^* to problem (1), which is called the projection of x^0 onto Ω, it is denoted by $P_\Omega(x^0)$, and it is characterized by the *Kolmogorov's criterion*:

$$\langle x^0 - x^*, x^* - x \rangle \geq 0 \text{ for all } x \in \Omega,\ x^* \in \Omega. \tag{3}$$

Dykstra's algorithm solves (1)–(2) by generating two sequences: the iterates $\{x_i^k\}$ and the increments $\{y_i^k\}$. These sequences are defined by the following recursive formulae:

$$\begin{array}{rcll} x_0^k & = & x_p^{k-1} & \\ x_i^k & = & P_{\Omega_i}(x_{i-1}^k - y_i^{k-1}) & ,\quad i = 1, 2, \ldots, p, \\ y_i^k & = & x_i^k - (x_{i-1}^k - y_i^{k-1}) & ,\quad i = 1, 2, \ldots, p, \end{array} \tag{4}$$

for $k = 1, 2, \ldots$ with initial values $x_p^0 = x^0$ and $y_i^0 = 0$ for $i = 1, 2, \ldots, p$.

Remarks

1. The increment y_i^{k-1} associated with Ω_i in the previous cycle is always subtracted before projecting onto Ω_i. Only one increment (the last one) for each Ω_i needs to be stored.

2. If Ω_i is a closed affine subspace, then the operator P_{Ω_i} is linear and it is not required, in the k^{th} cycle, to subtract the increment y_i^{k-1} before projecting onto Ω_i. Thus, for affine subspaces, Dykstra's procedure reduces to the alternating projection method of von Neumann [30]. To be precise, in this case, $P_{\Omega_i}(y_i^{k-1}) = 0$.

3. For $k = 1, 2, \ldots$ and $i = 1, 2, \ldots, p$, it is clear from (4) that the following relations hold

$$x_p^{k-1} - x_1^k = y_1^{k-1} - y_1^k, \tag{5}$$

$$x_{i-1}^k - x_i^k = y_i^{k-1} - y_i^k, \tag{6}$$

where $x_p^0 = x^0$ and $y_i^0 = 0$, for all $i = 1, 2, \ldots, p$.

For the sake of completeness we now present the key theorem associated with Dykstra's algorithm.

Theorem 2.1 (Boyle and Dykstra, 1986 [2]) *Let $\Omega_1, \ldots, \Omega_p$ be closed and convex sets of $I\!R^n$ such that $\Omega = \cap_{i=1}^p \Omega_i \neq \emptyset$. For any $i = 1, 2, \ldots, p$ and any $x^0 \in I\!R^n$, the sequence $\{x_i^k\}$ generated by (4) converges to $x^* = P_\Omega(x^0)$ (i.e., $\|x_i^k - x^*\| \to 0$ as $k \to \infty$).*

3 Difficulties with some commonly used stopping criteria

In some applications it is possible to obtain a *somehow* natural stopping rule, associated with the problem at hand. For example, when solving a linear system, $Ax = b$, by alternating projection methods [3, 15], the residual vector ($r(x) = b - Ax$) is usually available and yields some interesting and robust stopping rules. Another example appears in image reconstruction for which a *good and feasible* image tells the user that it is time to stop the process [5, 6]. Similar circumstances are present in some other specific applications (e.g. molecular biology [18, 19]).

However, in general, this is not the case, and we are left with the information produced only by the internal computations, i.e., the sequence of iterates and perhaps the sequence of increments, and some inner products. For this general case, a popular stopping rule is to monitor the subsequence of projections onto one particular convex set, Ω_i, and stop the process when the distance, in norm, of two consecutive projections is less than or equal to a previously established tolerance [16, 17, 21, 28].

Another commonly used criterion, that is claimed to improve the previous one (e.g. [2, 12, 18, 29]) is to somehow compute an average of all the projections at each cycle of projections, and then stop the process when the distance, in norm, of two consecutive of those average projections is less than or equal to a previously established tolerance.

Finally, we would like to mention that another criterion, that is also designed to improve any of the two criteria above, is to check any of the previously described rules during N consecutive cycles, where N is a fixed positive integer.

None of these stopping rules is a trustable choice. The example below establishes that they can fail even for a two dimensional problem (see Figures 1 and 2).

To illustrate the difficulties with the previously described stopping criteria, consider the closed and convex set $\Omega = \Omega_1 \cap \Omega_2$, where $\Omega_1 = \{x \in I\!R^2 \mid x_1 + x_2 \geq 10\}$ is a half space and $\Omega_2 = \{x \in I\!R^2 \mid 3 \leq x_1 \leq 10, 0 \leq x_2 \leq 4\}$ is a box. This closed and convex set in $I\!R^2$ is shown in Figure 1.

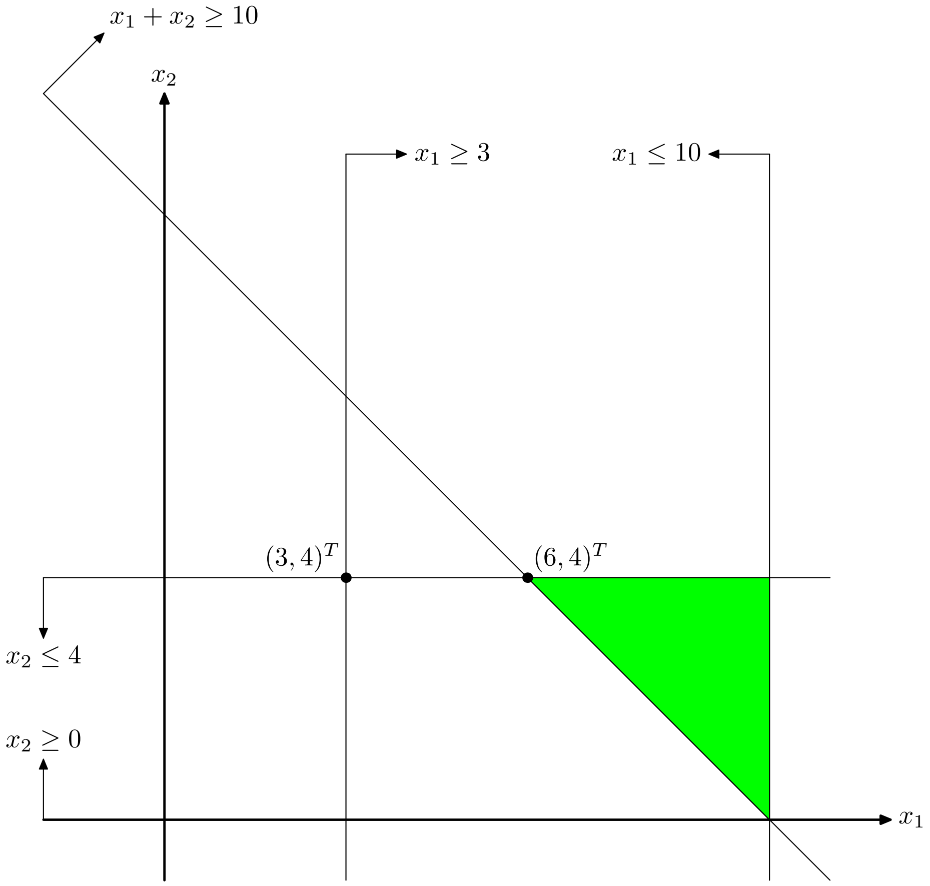

Figure 1: Feasible set $\Omega = \Omega_1 \cap \Omega_2$ in $I\!R^2$.

Let $x^0 = (-49, 50)^T$ and let us use Dykstra's algorithm to find the closest point to x^0 in Ω. In Figure 2 we can see the first two cycles of this convergent process. Since $y_1^0 = y_2^0 = 0$ (null initial increments) then for the first cycle we project x^0 onto Ω_1 to obtain $p_2 = x_1^1 = (-44.5, 54.5)^T$ and then we project p_2 onto Ω_2 to obtain $p_3 = x_2^1 = (3,4)^T$. For the second cycle, the increments are not null ($y_1^1 = (4.5, 4.5)^T$ and $y_2^1 = (47.5, -50.5)^T$), and we start from p_3. First we project $p_4 = p_3 - y_1^1$ onto Ω_1 to obtain $p_5 = x_1^2$. Then we project $p_6 = p_5 - y_2^1$ onto Ω_2 to obtain p_3 again. Hence $x_2^2 = x_2^1$. The increment associated with Ω_2 is large enough to take the iterate back to the quadrant where the projection onto the box is again p_3. As it can be seen in Table 1, this phenomenon will occur until cycle 32, i.e., $p_3 = x_2^1 = x_2^2 = \ldots = x_2^{32}$.

Moreover, by choosing x^0 far enough, we can guarantee that this misleading event can be repeated for as many cycles as any previously established positive integer N. Eventually the size of the increments will be reduced and convergence to x^* will be observed.

Current iterate		Proposed stopping criteria		
k	x_p^k	c^k	c_L^k	c_I^k
0	(−4.900E+01, 5.000E+01)	0.0000000E+00	0.0000000E+00	
1	(3.000E+00, 4.000E+00)	4.8470000E+03	4.8470000E+03	4.8470000E+03
2	(3.000E+00, 4.000E+00)	4.8560000E+03	4.8560000E+03	9.0000000E+00
3	(3.000E+00, 4.000E+00)	4.8650000E+03	4.8650000E+03	9.0000000E+00
4	(3.000E+00, 4.000E+00)	4.8740000E+03	4.8740000E+03	9.0000000E+00
5	(3.000E+00, 4.000E+00)	4.8830000E+03	4.8830000E+03	9.0000000E+00
6	(3.000E+00, 4.000E+00)	4.8920000E+03	4.8920000E+03	9.0000000E+00
7	(3.000E+00, 4.000E+00)	4.9010000E+03	4.9010000E+03	9.0000000E+00
8	(3.000E+00, 4.000E+00)	4.9100000E+03	4.9100000E+03	9.0000000E+00
9	(3.000E+00, 4.000E+00)	4.9190000E+03	4.9190000E+03	9.0000000E+00
10	(3.000E+00, 4.000E+00)	4.9280000E+03	4.9280000E+03	9.0000000E+00
11	(3.000E+00, 4.000E+00)	4.9370000E+03	4.9370000E+03	9.0000000E+00
12	(3.000E+00, 4.000E+00)	4.9460000E+03	4.9460000E+03	9.0000000E+00
13	(3.000E+00, 4.000E+00)	4.9550000E+03	4.9550000E+03	9.0000000E+00
14	(3.000E+00, 4.000E+00)	4.9640000E+03	4.9640000E+03	9.0000000E+00
15	(3.000E+00, 4.000E+00)	4.9730000E+03	4.9730000E+03	9.0000000E+00
16	(3.000E+00, 4.000E+00)	4.9820000E+03	4.9820000E+03	9.0000000E+00
17	(3.000E+00, 4.000E+00)	4.9910000E+03	4.9910000E+03	9.0000000E+00
18	(3.000E+00, 4.000E+00)	5.0000000E+03	5.0000000E+03	9.0000000E+00
19	(3.000E+00, 4.000E+00)	5.0090000E+03	5.0090000E+03	9.0000000E+00
20	(3.000E+00, 4.000E+00)	5.0180000E+03	5.0180000E+03	9.0000000E+00
21	(3.000E+00, 4.000E+00)	5.0270000E+03	5.0270000E+03	9.0000000E+00
22	(3.000E+00, 4.000E+00)	5.0360000E+03	5.0360000E+03	9.0000000E+00
23	(3.000E+00, 4.000E+00)	5.0450000E+03	5.0450000E+03	9.0000000E+00
24	(3.000E+00, 4.000E+00)	5.0540000E+03	5.0540000E+03	9.0000000E+00
25	(3.000E+00, 4.000E+00)	5.0630000E+03	5.0630000E+03	9.0000000E+00
26	(3.000E+00, 4.000E+00)	5.0720000E+03	5.0720000E+03	9.0000000E+00
27	(3.000E+00, 4.000E+00)	5.0810000E+03	5.0810000E+03	9.0000000E+00
28	(3.000E+00, 4.000E+00)	5.0900000E+03	5.0900000E+03	9.0000000E+00
29	(3.000E+00, 4.000E+00)	5.0990000E+03	5.0990000E+03	9.0000000E+00
30	(3.000E+00, 4.000E+00)	5.1080000E+03	5.1080000E+03	9.0000000E+00
31	(3.000E+00, 4.000E+00)	5.1170000E+03	5.1170000E+03	9.0000000E+00
32	(3.000E+00, 4.000E+00)	5.1260000E+03	5.1260000E+03	9.0000000E+00
33	(3.500E+00, 4.000E+00)	5.1347500E+03	5.1337500E+03	7.7500000E+00
34	(4.750E+00, 4.000E+00)	5.1394375E+03	5.1384375E+03	4.6875000E+00
35	(5.375E+00, 4.000E+00)	5.1406094E+03	5.1396094E+03	1.1718750E+00
36	(5.688E+00, 4.000E+00)	5.1409023E+03	5.1399023E+03	2.9296875E−01
37	(5.844E+00, 4.000E+00)	5.1409756E+03	5.1399756E+03	7.3242188E−02
38	(5.922E+00, 4.000E+00)	5.1409939E+03	5.1399939E+03	1.8310547E−02
39	(5.961E+00, 4.000E+00)	5.1409985E+03	5.1399985E+03	4.5776367E−03
40	(5.980E+00, 4.000E+00)	5.1409996E+03	5.1399996E+03	1.1444092E−03
41	(5.990E+00, 4.000E+00)	5.1409999E+03	5.1399999E+03	2.8610229E−04
42	(5.995E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	7.1525574E−05
43	(5.998E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	1.7881393E−05
44	(5.999E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	4.4703484E−06
45	(5.999E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	1.1175871E−06
46	(6.000E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	2.7939677E−07
47	(6.000E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	6.9849193E−08
48	(6.000E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	1.7462298E−08
49	(6.000E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	4.3655746E−09
50	(6.000E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	1.0913936E−09
51	(6.000E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	2.7284841E−10
52	(6.000E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	6.8212103E−11
53	(6.000E+00, 4.000E+00)	5.1410000E+03	5.1400000E+03	1.7053026E−11

Table 1: Dykstra's Algorithm for the projection of $x^0 = (-49, 50)^T$ onto $\Omega = \Omega_1 \bigcap \Omega_2$ (see Figures 1 and 2). Note that the sum of the distances among consecutive increments, c_I^k, is a strictly positive quantity which goes to zero when the method arrives to the solution. This fact warranties the monotonically increase of c_L^k and, as a consequence, of c^k.

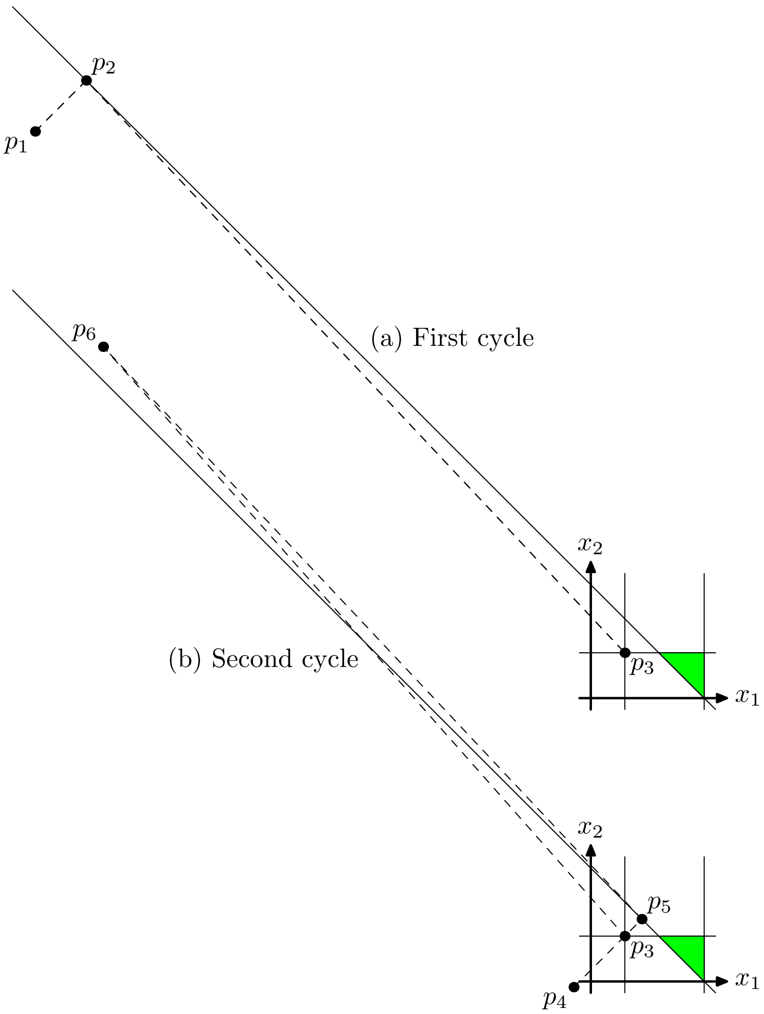

Figure 2: First two cycles of Dykstra's algorithm to find the projection of $x^0 = (-49, 50)^T$ onto $\Omega = \Omega_1 \cap \Omega_2$.

4 Robust stopping criteria

In order to develop robust stopping criteria for Dykstra's algorithm, we first need to establish an interesting inequality, that is obtained after a close inspection of the proof of the Boyle-Dykstra convergence theorem.

Theorem 4.1 *Let x^0 be any element of $\mathbb{R}^n$. Consider the sequences $\{x_i^k\}$ and $\{y_i^k\}$ generated by (4) and define c^k as*

$$c^k = \sum_{m=1}^{k} \sum_{i=1}^{p} \|y_i^{m-1} - y_i^m\|^2 + 2 \sum_{m=1}^{k-1} \sum_{i=1}^{p} \langle y_i^m, x_i^{m+1} - x_i^m \rangle. \tag{7}$$

Then, in the k^{th} cycle of Dykstra's algorithm,

$$\|x^0 - x^*\|^2 \geq c^k. \tag{8}$$

Moreover, at the limit when k goes to infinity, equality is attained in (8).

Proof. In the proof of Theorem 2.1, the following equation is obtained for $k > 1$ (Boyle and Dykstra [2]) (see also Deutsch [7], Lemma 9.19):

$$\begin{aligned} \|x^0 - x^*\|^2 = \|x_p^k - x^*\|^2 &+ \sum_{m=1}^{k}\sum_{i=1}^{p}\|y_i^{m-1} - y_i^m\|^2 \\ &+ 2\sum_{m=1}^{k-1}\sum_{i=1}^{p}\langle x_{i-1}^m - y_i^{m-1} - x_i^m, x_i^m - x_i^{m+1}\rangle \\ &+ 2\sum_{i=1}^{p}\langle x_{i-1}^k - y_i^{k-1} - x_i^k, x_i^k - x^*\rangle, \end{aligned} \tag{9}$$

where all terms involved are nonnegative for all k. Recall that $x_0^m = x_p^{m-1}$, and $y_i^0 = 0$ for all i. From (9) we obtain

$$\|x^0 - x^*\|^2 \geq \sum_{m=1}^{k}\sum_{i=1}^{p}\|y_i^{m-1} - y_i^m\|^2 + 2\sum_{m=1}^{k-1}\sum_{i=1}^{p}\langle x_{i-1}^m - y_i^{m-1} - x_i^m, x_i^m - x_i^{m+1}\rangle. \tag{10}$$

Finally, (8) is obtained by substituting (5) and (6) in (10).

Clearly, in (9) all terms in the right hand side are bounded. In particular, using (5) and (6), the fourth term can be written as $2\sum_{i=1}^{p}\langle y_i^k, x_i^k - x^*\rangle$, and using the Cauchy-Schwarz inequality and Theorem 2.1, we notice that it vanishes when k goes to infinity. Similarly, the first term in (9) tends to zero when k goes to infinity, and so at the limit equality is attained in (8). □

Let us now write c^k as follows:

$$c^k = c_L^k + c_S^k,$$

where

$$c_L^k = \sum_{m=1}^{k} c_I^m, \tag{11}$$

$$c_I^m = \sum_{i=1}^{p}\|y_i^{m-1} - y_i^m\|^2 \tag{12}$$

and

$$c_S^k = 2\sum_{m=1}^{k-1}\sum_{i=1}^{p}\langle y_i^m, x_i^{m+1} - x_i^m\rangle.$$

Both c_L^k and c_S^k are monotonically nondecreasing by definition. However, in the example shown in the previous section, it can be seen that the sequence of projections $\{x_i^m\}$ onto Ω_i

could remain constant for several consecutive cycles, and hence c_S^k could also remain constant for the same consecutive cycles. On the other hand, if the p increments $y_i^k, i = 1, 2, \ldots, p$, also remain constant for two consecutive cycles m and $m+1$, then, by (4), all the forthcoming projections and forthcoming increments (for all $k \geq m+1$) will remain the same, proving that we have already obtained the solution vector x^*. Hence, unless the solution has been attained, at least one of the increments must change (Table 1 illustrates this fact), and so c_I^{k+1} will be strictly positive and $c_L^{k+1} = c_L^k + c_I^{k+1} > c_L^k$, i.e., c_L^k must increase monotonically. This argument establishes the following result.

Theorem 4.2 *Consider the sequences $\{x_i^k\}$ and $\{y_i^k\}$ generated by (4), and c^k, c_L^k and c_I^k as defined in (7), (11) and (12), respectively. For any $k \in I\!N$, if $x^k \neq x^*$ then $c_I^{k+1} > 0$ and, hence, $c_L^k < c_L^{k+1}$ and $c^k < c^{k+1}$.*

We can combine the results established in Theorems 4.1 and 4.2 to propose robust stopping criteria. Notice that $\{c_L^k\}$ and $\{c^k\}$ are monotonically increasing and convergent, and also that $\{c_I^k\}$ converges to zero (these facts are illustrated in Table 1). Therefore we can stop the process when

$$c_I^k = \sum_{i=1}^{p} \|y_i^{k-1} - y_i^k\|^2 \leq \varepsilon$$

or, similarly, when

$$c^k - c^{k-1} = c_I^k + 2\sum_{i=1}^{p} \langle y_i^{k-1}, x_i^k - x_i^{k-1} \rangle \leq \varepsilon, \tag{13}$$

where $\varepsilon > 0$ is a sufficiently small tolerance. As c^k may grow fast, computing $c^k - c^{k-1}$ may give innacurate results due to loss of accuracy in floating point representation and, hence, cancellation. So, for the criterion in (13), it is recommendable to test convergence with the second expression.

In Table 1 we can observe the robustness of our proposed criteria for the example described in Figures 1 and 2. Notice that, indeed, c^k and c_L^k are monotonically increasing during the process, and that they only stop growing when the method arrives at the solution x^*, when c^k reveals the optimal Euclidean distance $\|x^0 - x^*\|^2$. Notice also that c_I^k tends to zero as k goes to ∞.

The computation of c_I^k involves the squared-norm $\|y_i^{k-1} - y_i^k\|^2$, for $i = 1, 2, \ldots, p$. By (6), $y_i^k = y_i^{k-1} + v$, where $v = x_i^k - x_{i-1}^k$ is a temporary n-dimensional array needed in the computation of Dykstra's algorithm. So, the computational cost involved in the calculation of c_I^k is just the cost of the extra inner product $\langle v, v \rangle$ at each iteration.

The computation of c^k involves the calculation of c_I^k plus an extra term. The computational of this extra term is also small and involves an inner product and the difference of two vectors per iteration. But, in contrast with the computation of c_I^k which does not require additional savings, the computation of the extra term requires to save p extra n-dimensional arrays (the same amount of memory required in Dykstra's algorithm to save the increments). So, the computation of c^k requires some additional calculations and memory savings, and hence it is more expensive. However, it also has the advantage of revealing the optimal distance: $\|x^0 - x^*\|^2$, that could be of interest in some applications.

We close this section with some comments concerning the behavior of the new stopping criteria when the problem is not feasible. First of all, due to errors or noise in the given data, it is a priori not always known whether the intersection set Ω is nonempty. Therefore, it is an interesting issue in real applications. In this case ($\Omega = \emptyset$), there is no solution and we know from Theorem 4.2 that the sequences $\{c_L^k\}$ and $\{c^k\}$ are monotonically increasing. Moreover, under some mild assumptions on the sets Ω_i, the sequences $\{x_i^k\}$ converge for $1 \leq i \leq p$, and there exists a real constant $\delta > 0$ such that $\sum_{i=1}^{p} \|x_{i-1}^k - x_i^k\|^2 \geq \delta$ for all k. A discussion on this topic is presented in [1, Section 6], including a notion of *distance* between all the sets Ω_i (see also [25]). Now using (6), we obtain

$$\sum_{i=1}^{p} \|x_{i-1}^k - x_i^k\|^2 = \sum_{i=1}^{p} \|y_i^{k-1} - y_i^k\|^2 = c_I^k.$$

Therefore, the sequence $\{c_I^k\}$ remains bounded away from zero, whereas $\{c_L^k\}$ and $\{c^k\}$ tend to infinity. Consequently, none of the new proposed stopping criteria will be satisfied for any k.

Regarding the mild assumptions discussed in [1, 25], for which the sequences $\{x_i^k\}$ converge for $1 \leq i \leq p$, and $c_I^k \geq \delta$ for all k, we can list the following cases that appear frequently in applications: (a) at least one of the sets Ω_i is bounded, (b) all of them are polyhedral, (c) there exists $z_i \in \Omega_i$ such that $\|z_i - z_j\|$ equals the distance between Ω_i and Ω_j for all possible $1 \leq i, j \leq p$. In other words, if any of these cases holds and one of the new proposed stopping rules is used, then Dykstra's algorithm stops only if a solution of (1) is reached. In that sense, they are robust stopping criteria. Nevertheless, there are cases, also discussed in [1, 25] for which the distance is not attained, and they establish that $\|x_i^k\|$ tend to $+\infty$ for $1 \leq i \leq p$. In the presence of one of these cases, the stopping rules may stop erroneously due to the numerical cancellation of very large numbers. For example, consider the following two convex sets: $\Omega_1 = \{(x, y)^T \in I\!R^2 \mid x > 0 \text{ and } y \geq M + 1/x\}$, and $\Omega_2 = \{(x, y)^T \in I\!R^2 \mid x > 0 \text{ and } y \leq -M - 1/x\}$, where $M > 0$ is a fixed real constant. None of the conditions above ((a), (b), or (c)) holds in this case, and in fact, the iterates tend to $(+\infty, M)^T$ and $(+\infty, -M)^T$, respectively. In theory, $c_I^k > 2M$ for all k, and our stopping criteria would not be satisfied. However, in practice, the *size* of the iterates could be very large, and cancellation might occur, producing a floating point representation of c_I^k very close to zero.

5 Conclusions

We pointed out that the frequently used stopping criteria for Dykstra's algorithm are not trustable and showed a two-dimensional example, using a box and a half space, in which these rules fail to detect convergence of the Dykstra's iterative procedure.

We introduced robust stopping criteria and applied them to an example in which the commonly used criteria failed. We proved that our criteria are well defined and that one of the sequences involved, $\{c^k\}$, converges to the distance among the point to be projected and its projection. We also establish that if there is no solution (empty intersection) then under mild assumptions the new criteria are not satisfied. Finally, we elaborated on the computational cost of the proposed stopping rules.

Acknowledgments.

We are indebted to two anonymous referees whose comments helped us to improve the previous version of this paper. In particular we thank one of the referees for suggesting the discussion about the behavior of the new stopping rules when the intersection set is empty. His comments motivated the last two paragraphs in Section 4.

References

[1] H. H. Bauschke and J.M. Borwein [1994], Dykstra's alternating projection algorithm for two sets, *J. Approx. Theory* 79, pp. 418-443.

[2] J. P. Boyle and R. L. Dykstra [1986], A method for finding projections onto the intersection of convex sets in Hilbert spaces, *Lecture Notes in Statistics* 37, pp. 28–47.

[3] R. Bramley and A. Sameh [1992], Row projection methods for large non symmetric linear systems, *J. Sci. Statist. Comp.* 13, pp. 168–193.

[4] L.M. Bregman, Y. Censor and S. Reich [1999], Dykstra's algorithm as the nonlinear extension of Bregman's optimization method, *Journal of Convex Analysis* 6, pp. 319–333.

[5] Y. Censor and G.T. Herman [1987], On some optimization techniques in image reconstruction from projections, *Applied Numerical Mathematics* 3, pp. 365–391.

[6] P.L. Combettes [1996], The convex feasibility problem in image recovery, *Advances in Imaging and Electron Physics* 95, pp. 155–270.

[7] F. Deutsch [2001], *Best Approximation in Inner Product Spaces*, Springer Verlag New York, Inc.

[8] F. Deutsch and H. Hundal [1994], The rate of convergence of Dykstra's cyclic projections algorithm: the polyhedral case, *Numer. Func. Anal. Optim.* 15, pp. 537-565.

[9] F. Deutsch [1995], Dykstra's cyclic projections algorithm: the rate of convergence, *Approximation Theory, Wavelets and Applications*, Kluwer Academic Publishers, Netherlands, Ed. S.P. Singh.

[10] R. L. Dykstra [1983], An algorithm for restricted least-squares regression, *Journal of the American Statistical Association* 78, pp. 837–842.

[11] M. G. Eberle and M. C. Maciel [2003], Finding the Closest Toeplitz Matrix, *Computational and Applied Mathematics* 22, pp. 1–18.

[12] R. Escalante and M. Raydan [1996], Dykstra's Algorithm for a Constrained Least-Squares Matrix Problem, *Numerical Linear Algebra and Applications* 3, pp. 459-471.

[13] R. Escalante and M. Raydan [1998], Dykstra's algorithm for constrained least-squares rectangular matrix problems, *Computers and Mathematics with Applications* 35, pp. 73-79.

[14] N. Gaffke and R. Mathar [1986], A cyclic projection algorithm via duality, *Metrika* 36, pp. 29–54.

[15] U. M. Garcia-Palomares [1999], Preconditioning projection methods for solving algebraic linear systems, *Numerical Algorithms* 21, pp. 157–164.

[16] W. Glunt [1995], An alternating projection method for certain linear problems in a Hilbert space, *IMA J. of Numer. Anal.* 15, pp. 291–305.

[17] W. Glunt, T. L. Hayden, S. Hong and J. Wells [1990], An alternating projection algorithm for computing the nearest Euclidean distance matrix, *SIAM J. Matrix Anal. Appl.* 11, pp. 589–600.

[18] W. Glunt, T. L. Hayden and M. Raydan [1993], Molecular conformations from distance matrices, *Journal of Computational Chemistry* 14, pp. 114–120.

[19] W. Glunt, T. L. Hayden and W. M. Liu [1991], The embedding problem for predistance matrices, *Bulletin of Mathematical Biology* 53, pp. 769–796.

[20] S. P. Han [1988], A successive projection method, *Mathematical Programming* 40, pp. 1–14.

[21] N. Higham [2002], Computing the nearest correlation matrix – a problem from finance, *IMA J. Numer. Anal.* 22, pp. 329–343.

[22] C. Hildreth [1957], A quadratic programming procedure, *Naval Res. Log. Quart.* 4, pp. 79-85.

[23] H. Hu and I. Olkin [1991], A numerical procedure for finding the positive definite matrix closest to a patterned matrix, *Statistics and Probability letters* 12, pp. 511-515.

[24] A. Iusem and A. De Pierro [1990], On the convergence properties of Hildreth's quadratic programming algorithm, *Math. Prog.* 47, pp. 37-51.

[25] A. Iusem and A. De Pierro [1991], On the convergence of Han's method for convex programming with quadratic objective, *Math. Prog.* 52, pp. 265-284.

[26] A. Iusem and B.F. Svaiter [1995], Primal-dual row-action method for convex programming, *J. Optimization Theory Appl.* 86, pp. 73-112.

[27] A. Lent and Y. Censor [1980], Extensions of Hildreth's row-action method for quadratic programming, *SIAM Journal on Control and Optimization* 18, pp. 444–454.

[28] M. Mendoza, M. Raydan and P. Tarazaga [1998], Computing the nearest diagonally dominant matrix, *Numerical Linear Algebra with Applications* 5, pp. 461-474.

[29] M. Raydan and P. Tarazaga [2002], Primal and polar approach for computing the symmetric diagonally dominant projection, *Numerical Linear Algebra with Applications* 9, pp. 333-345.

[30] J. von Neumann [1950], Functional operators vol. II. The geometry of orthogonal spaces, *Annals of Mathematical Studies* 22, Princeton University Press. This is a reprint of mimeographed lecture notes first distributed in 1933.