Stud. Univ. Babeş-Bolyai Math. 63(2018), No. 3, 401–411
DOI: 10.24193/subbmath.2018.3.11

A comparative analysis of the convergence regions for different parallel affine projection algorithms

Irina Maria Artinescu

Abstract. This paper analysis the dimension and the shape of convergence regions of three algorithms used to solve the convex feasibility problem in bidimensional space: the *Parallel Projection Method (PPM)*, the classical *Extrapolated Method of Parallel Projections (EMOPP)* and a modified version of EMOPP.

Mathematics Subject Classification (2010): 52A10.

Keywords: Convex feasibility problem, parallel projection method, affine projection.

1. Introduction

The convex feasibility problem is one the classical problem in computational mathematics and can be simplify described as the problem to find a solution that satisfied a given set of inequalities. The projection methods were used in the past to solve some systems of linear equations in Euclidean spaces [4] and were modified to be applied to systems of linear inequalities in [1], [11], [12]. The algorithmic steps in these first algorithms consists of projections onto some affine subspaces or a half-spaces. Later, the method become more sophisticated [8], [9], [10], being adapted to solve the more general problem of finding a point in the intersection of a family of closed convex sets in a Hilbert space [2], [5].

The affine projection methods have numerous practical applications in data compression, in tomography, neural networks or in image filtering (see also [3]). While the mathematical analysis of week or strong convergence of different projection methods was largely studied in the past ([2], [5], [6], [7]), an explicit analysis of the convergence

This paper was presented at the 12th Joint Conference on Mathematics and Computer Science (MaCS 2018), June 14-17, 2014, Cluj-Napoca, Romania.

regions of the convex feasibility algorithms inspired by the projections methods where rarely approached.

In this paper we tested the convergence in finite number of steps for two of these projection methods: the *Parallel Projection Method (PPM)* and the classical *Extrapolated Method of Parallel Projections (EMOPP)*, and a variant of the EMOPP that uses variable affine combinations, in order to determine theirs convergence in finit number of steps and the shapes of theirs convergence regions, defined by the staring points for witch the algorithms converge in a given number of steps.

2. The convex feasibility problem and the parallel projections method

The convex feasibility problem (CFP) was formulated in [5] as : Given m closed convex sets $C_1, C_2, \ldots, C_m \subseteq \mathcal{R}^n$, with nonempty intersection, $\cap C_i \neq \emptyset$, defined by $C_i = \{x \in \mathcal{R}^n \mid f_i(x) \leq 0\}$, with $f_i : \mathcal{R}^n \to \mathcal{R}$ a convex function, the CFP is to find a point $x \in C = \bigcap_{i=1}^{m} C_i$.

Consider $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$ a convex polygon and consider the *Parallel Projection Method (PPM)*, which is governed by the iteration ([7]):

$$(\forall n \in \mathcal{N}) Q_{j+1} = Q_j + \Lambda\Big(\sum_{i \in 1..n} w_i P_i(Q_j) - Q_j\Big), \tag{2.1}$$

where $1 + \epsilon \leq \Lambda \leq 2 - \epsilon$ are the *relaxation* parameters, $0 < \epsilon < 1$ and $\sum_{i \in 1..n} w_i = 1$, with the fixed weight w_i.

A variation of the PPM, called *Extrapolated Method of Parallel Projections (EMOPP)* is obtained involving involving different classes of control index sets $\{I_n\}$ [5]. The iteration of this method are similar to (2.1):

$$(\forall j \in \mathcal{N}) Q_{j+1} = Q_j + \Lambda\Big(\sum_{i \in I_n} w_i P_i(Q_j) - Q_j\Big), \tag{2.2}$$

where the indices set $\{I_n\}$, called *control* sequence, are variable from one iteration to another. Many variants of the control sequences where studied in [5].

The *modified Extrapolated Method of Parallel Projections (mEMOPP)* was obtained introducing variable weight w_i that depend inverse proportionally to the distance from Q_n to his projections on the considered semi-planes. If we denote:

$$M_{j,i} = pr(Q_j, P_i P_{i+1}),$$

by convention $(P_{n+1} = P_1)$ and $d_{j,i} = dist(Q_j, P_i P_{i+1})$ and the weights are defined by

$$w_{j,i} = \frac{1/(d_{j,i} + 1)}{\sum_{i \in I_j} 1/(d_{j,i} + 1)}$$

where I_j are the set of indexes i for witch $P_i P_{i+1}$ separates Q_j and the interior of the polygon $\mathcal{P}$ for the case of mEMOPP.

The determination of the shape of convergence regions is equivalent to inverse the CFP: *determine for a given point Q of the plane the set of the points that are transported in Q using the different versions of the PPM associate transformation.*

For the simplification of the calculus, we choused as convex $\mathcal{P}$ a regular rectangle, defined by the relations $x = -a$, $x = a$, $y = -b$, $y = b$, where $a, b > 0$:

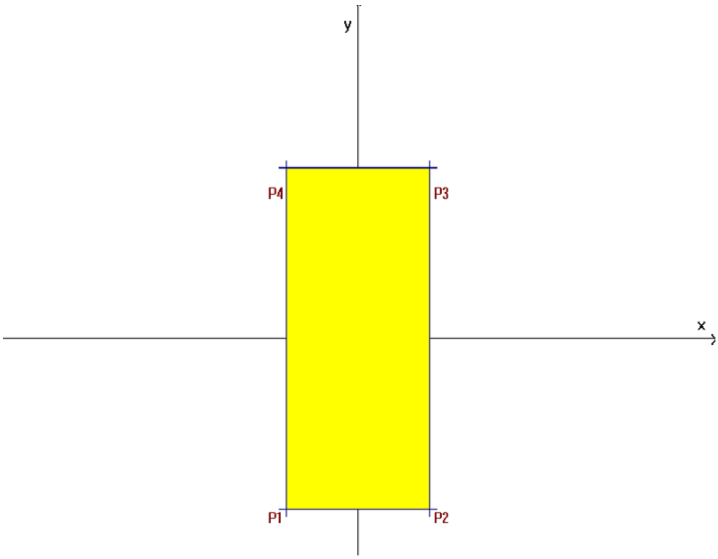

FIGURE 1. The initial polygon.

3. The convergence regions for parallel projection method

In the case of PPM algorithm, any point $Q(x, y)$ from the plane has the projection on the lines $y = -a$, $y = a$, $x = -b$, $x = b$ formed by the points $M_1(x, -b)$, $M_2(x, b)$, $M_3(-a, y)$ respectively $M_4(a, y)$. The transform $S = f_{PPM}(Q)$ move the point Q in

$$S(m, n) = (1 - \Lambda)(x, y) + \Lambda \left(\frac{x}{2}, \frac{y}{2}\right) = \frac{2 - \Lambda}{2}(x, y) \tag{3.1}$$

The transform f_{PPM} is a continuous bijection with the inverse:

$$Q(x, y) = f_{PPM}^{-1}(m, n) = \frac{2}{2 - \Lambda}(m, n) \tag{3.2}$$

The convergence regions formed by the starting points Q for which the algorithm stop in k steps, are given by

Theorem 3.1. *If we denote by L_k the rectangle defined with the relations $x \geq -\frac{2(k+1)}{2-\Lambda}b$, $x \leq \frac{2(k+1)}{2-\Lambda}b$, $y \geq -\frac{2(k+1)}{2-\Lambda}a$ and $y \leq \frac{2(k+1)}{2-\Lambda}a$, then the k convergence region for PPM algorithm are defined by $L_k \setminus L_{k-1}$.*

The proof is immediate from (3.2).

4. The convergence regions for extrapolated method of parallel projections

The first convergence region is formed by starting points for witch the algorithms converge in a one steps.

One consider the case of the EMOPP algorithm. The plane is separated in eight regions, function of the orientation of each point relatives to the sides of the rectangle, as in the Figure 2.

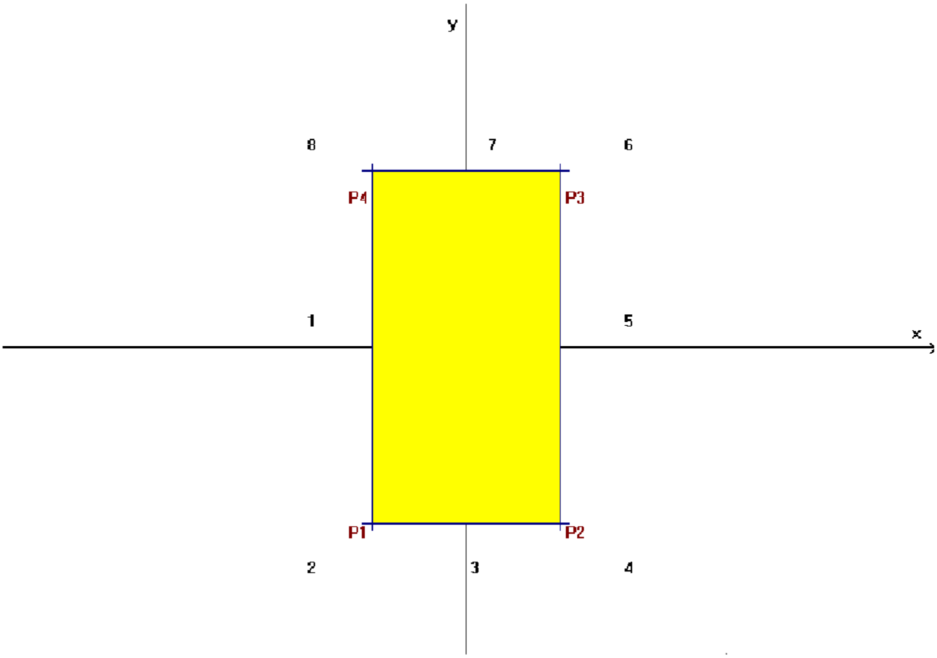

FIGURE 2. The eight sectors of the plane defined by the edges of the rectangle.

Consider a generic point $Q(x, y) \in ext(\mathcal{P})$. For Q in one of the sectors 1, 3, 5 and 7, the EMOPP step in involve only one projection $M = Pr(Q, P_iP_{i+1})$ on the nearest side $[P_iP_{i+1}]$ of the rectangle.

Defining the step of EMOPP as $Q(x, y) \to S(m, n) := f_{EMOPP}(Q)$, we have for Q in the sector 1:

$$S = f_{EMOPP}(Q) = Q + \Lambda(Pr(Q, P_1P_4) - Q). \tag{4.1}$$

With this notation we have (for the fist sector)

$$f_{EMOPP}(x, y) = (-x(\Lambda - 1) - \Lambda a, y) \tag{4.2}$$

The inverse transform is

$$f^{-1}_{EMOPP}(m, n) = \left(\frac{-\Lambda a - m}{\Lambda - 1}, n\right) \tag{4.3}$$

For the sector 5, one obtain

$$f_{EMOPP}(x, y) = (-x(\Lambda - 1) + \Lambda a, y) \tag{4.4}$$

The inverse transform is

$$f^{-1}_{EMOPP}(m, n) = \left(\frac{\Lambda a - m}{\Lambda - 1}, n\right) \tag{4.5}$$

For the sectors 3 and 7, similar formulas are obtained. For example, in the sector 3:

$$f_{EMOPP}(x, y) = (x, -y(\Lambda - 1) - \Lambda b) \tag{4.6}$$

The inverse transform is

$$f^{-1}_{EMOPP}(m, n) = \left(m, \frac{-\Lambda b - n}{\Lambda - 1}\right) \tag{4.7}$$

and in the sector 7:

$$f_{EMOPP}(x, y) = (x, -y(\Lambda - 1) + \Lambda b) \tag{4.8}$$

The inverse transform is

$$f^{-1}_{EMOPP}(m, n) = \left(m, \frac{\Lambda b - n}{\Lambda - 1}\right). \tag{4.9}$$

The preimage of each point inside the rectangle is formed then by at least four points from the sectors 1, 3, 5, 7. The preimage of entire rectangle in the sectors 1 and 5 is obtained by imposing the condition $-a < m < a$, and the preimage of the rectangle in the sectors 3 and 7 is obtained by imposing the condition $-b < n < b$. A direct calculus give that

$$\begin{aligned}
Q &\in \left[-a\frac{\Lambda+1}{\Lambda-1}, -a\right) \times (-b, b) \text{ for the sector 1;} \\
Q &\in (-a, a) \times \left[-b\frac{\Lambda+1}{\Lambda-1}, -b\right) \text{ for the sector 3;} \\
Q &\in \left(a, a\frac{\Lambda+1}{\Lambda-1}\right] \times (-b, b) \text{ for the sector 5;} \\
Q &\in (-a, a) \times \left(b, b\frac{\Lambda+1}{\Lambda-1}\right] \text{ for the sector 7;}
\end{aligned} \tag{4.10}$$

In the case of the other sectors, the transform f involve two projection at each step. Choosing, for example, the sector 6 (for $Q(x, y)$ having $x > a,\ y > b$), the projections of Q on the line $y = b$ give the point $M_1 = (x, b)$, the projections of Q on the line $x = a$ give the point $M_2 = (a, y)$ and the algorithm produces $Q(x, y) \to S(m, n)$ with

$$m = \frac{(2 + \Lambda)x - \Lambda a}{2}, \; n = \frac{(2 + \Lambda)y - \Lambda b}{2}. \tag{4.11}$$

The conditions $-a \le m \le a$ and $-b \le n \le b$ produce

$$\frac{\lambda - 2}{\lambda + 2} a \le x \le a \text{ and } \frac{\lambda - 2}{\lambda + 2} b \le y \le b \tag{4.12}$$

witch are in contradiction with the supposition $x > a,\ y > b$.

The image of every point of the sector 6 become to the sector 6 (see Figure 3).

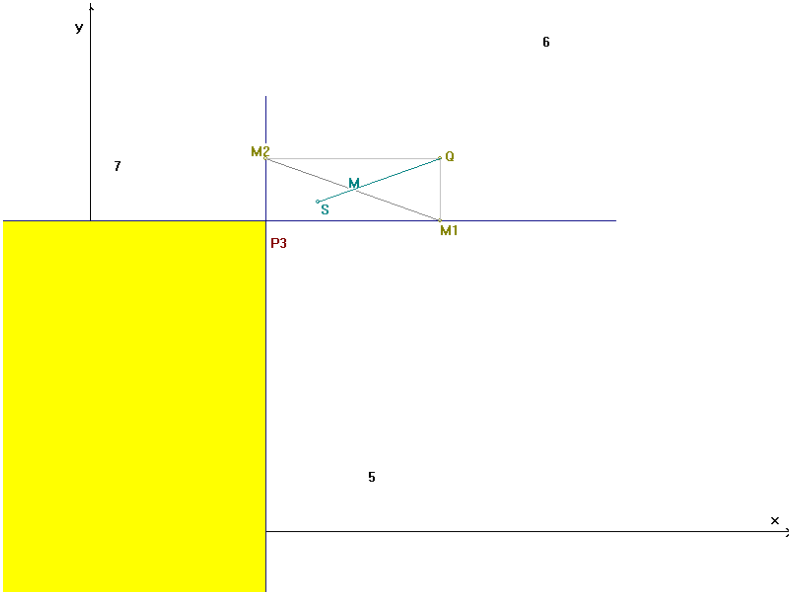

FIGURE 3. The EMOPP transform $Q \to S$ for a starting point Q inside the sector 6.

In conclusion, the EMOPP algorithm do not converges in finite number of steps for any starting point of the sector 6, if $\Lambda < 2$.

Similar results are obtained for the starting points of the sectors 2, 4, and 8. The infinite series defined by $Q_0 = Q$, $Q_{n+1} = f_{EMOPP}(Q)$ converges to the vertice P_3 of the rectangle.

The second order convergence regions, formed by the points for witch the algorithm stop in two steps ($Q(x,y) \to Q_1 \to S \in \mathcal{P}$), are determined by imposing the conditions

- $a < -x(\Lambda - 1) - \Lambda a < a\frac{\Lambda+1}{\Lambda-1}$ for Q in the sector 1,
- $b < -y(\Lambda - 1) - \Lambda b < b\frac{\Lambda+1}{\Lambda-1}$ for Q in the sector 3,
- $-a\frac{\Lambda+1}{\Lambda-1} < -x(\Lambda - 1) - \Lambda a < -a$ for Q in the sector 5,
- $-b\frac{\Lambda+1}{\Lambda-1} < -y(\Lambda - 1) - \Lambda b < -b$ for Q in the sector 7.

A direct calculus give:

$$\begin{aligned} Q &\in \left[-a\frac{\Lambda^2+1}{(\Lambda-1)^2}, -a\frac{\Lambda+1}{\Lambda-1}\right) \times (-b,b) \text{ for the sector 1;} \\ Q &\in (-a,a) \times \left[-b\frac{\Lambda^2+1}{(\Lambda-1)^2}, -b\frac{\Lambda+1}{\Lambda-1}\right) \text{ for the sector 3;} \\ Q &\in \left(a\frac{\Lambda+1}{\Lambda-1}, a\frac{\Lambda^2+1}{(\Lambda-1)^2}\right] \times (-b,b) \text{ for the sector 5;} \\ Q &\in (-a,a) \times \left(b\frac{\Lambda+1}{\Lambda-1}, b\frac{\Lambda^2+1}{(\Lambda-1)^2}\right] \text{ for the sector 7;} \end{aligned} \tag{4.13}$$

In general, we have:

Theorem 4.1. *The k-th convergence regions for the EMOPP algorithm (the regions of starting point for witch the algorithm stop in k steps) are defined by:*

$$\begin{aligned} Q &\in \left[-a\frac{R_k(\Lambda)}{(\Lambda-1)^k}, -a\frac{R_{k-1}(\Lambda)}{(\Lambda-1)^{k-1}}\right) \times (-b,b) \textit{ for the sector 1;} \\ Q &\in (-a,a) \times \left[-b\frac{R_k(\Lambda)}{(\Lambda-1)^k}, -b\frac{R_{k-1}(\Lambda)}{(\Lambda-1)^{k-1}}\right) \textit{ for the sector 3;} \\ Q &\in \left(a\frac{R_{k-1}(\Lambda)}{(\Lambda-1)^{k-1}}, a\frac{R_k(\Lambda)}{(\Lambda-1)^k}\right] \times (-b,b) \textit{ for the sector 5;} \\ Q &\in (-a,a) \times \left(b\frac{R_{k-1}(\Lambda)}{(\Lambda-1)^{k-1}}, b\frac{R_k(\Lambda)}{(\Lambda-1)^k}\right] \textit{ for the sector 7;} \end{aligned} \tag{4.14}$$

for any $k > 1$, where the polynomial R_k is given by:

$$R_k(\Lambda) = 1 + \frac{\Lambda}{2-\Lambda}[1-(\Lambda-1)^k] \tag{4.15}$$

Proof. He have $R_1 = 1 + \Lambda$ and the recursion

$$R_k = R_{k-1} + \Lambda(\Lambda-1)^{k-1} \tag{4.16}$$

can be proved by mathematical induction. Indeed, for $k = 2$ the formulas (4.15) was verified before. Now, supposing that the point $S(m,n)$ verifies (for the (k+1)-convergence region in sector 5)

$$a\frac{R_k(\Lambda)}{(\Lambda-1)^k} < m \le a\frac{R_{k+1}(\Lambda)}{(\Lambda-1)^{k+1}} \ , \ -b \le y \le b$$

a short computation give for $Q(x,y) = f^{-1}_{EMOPP}(S)$, using that $x = \frac{\Lambda a - m}{\Lambda - 1}$ and $y = n$:

$$-a\frac{R_k(\Lambda)}{(\Lambda-1)^k} \le x < -a\frac{R_{k-1}(\Lambda)}{(\Lambda-1)^{k-1}}.$$

witch is a point inside the k-convergence region of the sector 1. A similar formulas permit to pass from the (k+1)-convergence region of the sector 3 to the k- convergence region of the sector 7 and reciprocally.

Finally

$$R_k = 1 + \sum_{i=1}^{k} \Lambda(\Lambda - 1)^{i-1}$$

and the relation(4.15) is immediate. $\square$

5. The convergence regions of modified extrapolated method of parallel projections

In the case of modified Extrapolated Method of Parallel Projection, the weights w_i are not constants, but are inverse proportionates to the distances from starting points Q to the nearest edges of the rectangle. We consider now that the relaxation parameter $\Lambda \in (1, 2)$.

If the starting point Q become to one of the regions 1, 3, 5, or 7, there are a single projection involved, then the weight involved will be only $w_1 = 1$. In this case, the same formulas as in (4.14) can be deduced.

The situation of the sectors 2, 4, 6 and 8 are therefore different. Studying as example only the sector 6 (formed by the points $Q(x, y)$ with $x > a$ and $y > b$), the projection of nearest edges of the rectangle $y = b$, respectively $x = a$ give the points $M_1(x, b)$ and $M_2(a, y)$.

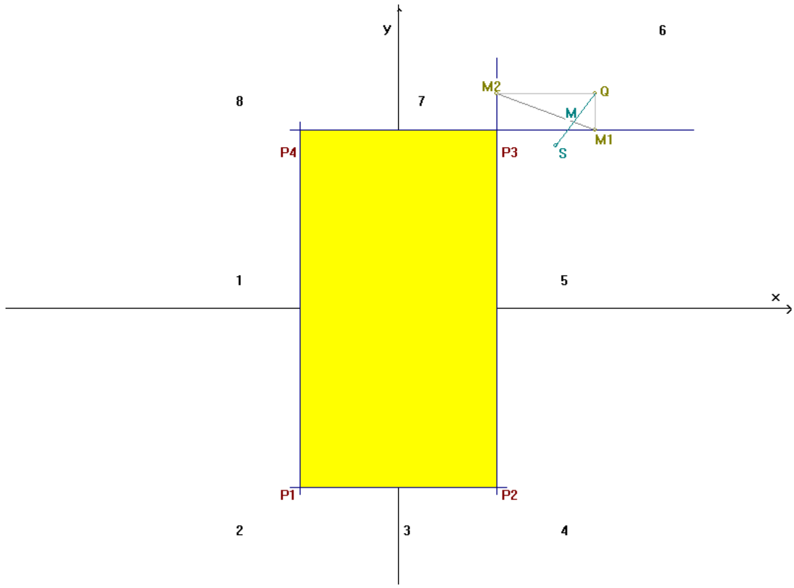

FIGURE 4. The mEMOPP transform $Q \to S$ for a starting point Q from the sector 6.

The weights are computed from $d_1 = QM_1 = y - b$ and $d_2 = QM_2 = x - a$:

$$\begin{cases} w_1 &= \frac{\frac{1}{d_1+1}}{\frac{1}{d_1+1}+\frac{1}{d_2+1}} = \frac{x-a+1}{x+y-a-b+2} \\ w_2 &= \frac{\frac{1}{d_2+1}}{\frac{1}{d_1+1}+\frac{1}{d_2+1}} = \frac{y-b+1}{x+y-a-b+2} \end{cases} \tag{5.1}$$

and we obtain the points

$$M = w_1M_1 + w_2M_2 = (w_1x + w_2a, w_1b + w_2y) \text{ and } S(m, n) = f_{mEMOPP}(Q):$$

$$\begin{cases} m &= (1-\Lambda)x + \Lambda(w_1x + w_2a) \\ n &= (1-\Lambda)y + \Lambda(w_1b + w_2y) \end{cases} \tag{5.2}$$

We supposed at this stage that the distances $d_1 < d_2$ (see the Figure 4). The point S can belong to the sector 6 or can jump to the sector 5. For any point in the sector 5, the mEMOPP algorithm converges in a finite number of steps, conform to the Theorem 4.1. We wish to determine the sub-region of the sector 6 that have the image from f_{mEMOPP} inside the sector 5.

The condition that S become to the sector 5 writes as $m > a$ and $-b \leq n \leq b$. The condition $m > a$ is always verified because $QS < QP_3$. The condition $n \leq b$ rewrites as:

$$\begin{aligned} (1-\Lambda)y + \Lambda(w_1b + (1-w_1)y) &\leq b \\ (1-\Lambda w_1)y &\leq b(1-\lambda w_1) \text{and because } y > b \Rightarrow \\ (1-\Lambda w_1) &\leq 0 \end{aligned} \tag{5.3}$$

The weight where computet in the formulas (5.1), then we have

$$\Lambda \frac{y-b+1}{x+y-a-b+2} \geq 1,$$

and finally

$$y \leq (x-a)(\Lambda-1) + b - 2. \tag{5.4}$$

The condition $-b \leq n$ become:

$$\begin{aligned} (1-\Lambda)y + \Lambda(w_1b + (1-w_1)y) &\geq -b \\ (1-\Lambda w_1)y &\geq -b(1+\lambda w_1) \text{and because } 1-\Lambda w_1 < 0 \Rightarrow \\ y &\leq b(1 + \frac{2}{\Lambda w_1 - 1}) \end{aligned} \tag{5.5}$$

and $\Lambda w_1 < 2$ give $y < 3b$.

In conclusion if the starting point $Q(x, y)$ verifies $b < y < 3b$ and

$$y \leq (x-a)(\Lambda-1) + b - 2,$$

the mEMOPP algorithm produce a point $S = f_{mEMOPP}(Q)$ inside the section 5. The algorithm will stop in $k+1$ steps if we have the supplementary condition

$$a\frac{R_{k-1}(\Lambda)}{(\Lambda-1)^{k-1}} < (1-\Lambda)x + \Lambda(w_1x + w_2a) \leq a\frac{R_k(\Lambda)}{(\Lambda-1)^k} \tag{5.6}$$

where

$$R_k(\Lambda) = 1 + \frac{\Lambda}{2-\Lambda}[1-(\Lambda-1)^k]$$

(from the Theorem 4.1).

With the notation

$$S_k = a\frac{1 + \frac{\Lambda}{2-\Lambda}[1 - (\Lambda - 1)^k]}{(\Lambda - 1)^k}$$

the limits between the $k + 1$ and the $k + 2$ -convergence regions inside the surface delimited by $b < y < 3b$ and $y \leq (x - a)(\Lambda - 1) + b - 2$ verifies the equation (from (5.6))

$$(1 - \lambda w_2)x + \Lambda w_2 a = S_k. \tag{5.7}$$

After some computations, the equation (5.7) become

$$\begin{aligned} x^2 - (\Lambda - 1)xy - [a + b - 2 + \Lambda(1 - b) + S_k]x - \\ -(S_k - \Lambda a)y + S_k(a + b - 2) + \Lambda a(b - 1) \end{aligned} \tag{5.8}$$

and the substitution $z = x - \frac{\Lambda-1}{2}y$ give the equation

$$\begin{aligned} z^2 - \frac{(\Lambda - 1)^2}{4}y^2 - [a + b - 2 + \Lambda(1 - b) + S_k](z + \frac{\Lambda - 1}{2}y) - \\ -(S_k - \Lambda a)y + S_k(a + b - 2) + \Lambda a(b - 1) \end{aligned} \tag{5.9}$$

The equation (5.9) is obviously a hyperbola equation, denoted by $\mathcal{H}_{k+1}$ in the next. We proved

Theorem 5.1. *In the case of the mEMOPP algorithm applied for the rectangle* $P_1P_2P_3P_4$ *, for any* $k > 1$ *there are a region of starting points inside the sector 6, delimited by* $b < y < 3b$ *and* $y \leq (x - a)(\Lambda - 1) + b - 2$ *and the hyperbolas* $\mathcal{H}_{k-1}$*,* $\mathcal{H}_k$*, from where the algorithm stop in exactly* k *steps.*

This result imply that the algorithm mEMOPP have a better convergence that the not-modified version EMOPP. Similar results can be obtained for the case of the sectors 2, 4 and 8.

6. Conclusions

In this paper we have shown that while the PPM algorithm that solve the convex feasibility problem converges always for a regular quadrilateral convex, his newest version EMOPP do not converges in finite number of steps for large regions of starting points. The modified version of EMOPP that involves variable weights in the affine combination used at each step, depending on the relative position of the point to the convex, permits to significative extend the convergence regions of the algorithm. Explicit determination of these regions where also presented.

References

[1] Agmon, S., *The relaxation method for linear inequalities*, Canad. J. Math., **6**(1954), 382-392.

[2] Bauschke, H.H., Borwein, J.M., *On projection algorithms for solving convex feasibility problems*, SIAM Rev., **38**(1996), no. 3, 367-426.

[3] Censor, Y., Chen, W., Combettes, P., David, R., Herman, G., *On the effectiveness of projection methods for convex feasibility problems with linear inequality constraints*, Comput. Optim Appl., Kluwer Academic Publishers Norwell, USA, 2012, 1065-1088.

[4] Cimmino, G., *Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari*, La Ricerca Scientifica 1, Roma, 1938, 326-333.

[5] Combettes, P.L., *The convex feasibility problem in image recovery*, Adv. Imag. Elect. Phys., **95**(1996), 155-270.

[6] Combettes, P.L., *Hilbertian convex feasibility problem: Convergence of projection methods*, Appl. Math. Optim., **35**(1997), 311-330.

[7] Combettes, P.L., *Convex set theoretic image recovery by extrapolated iterations of parallel subgradient projections*, IEEE Trans. Image Process., **6**(1997), no. 4, 493 -506.

[8] Gilbert, P., *Iterative methods for the three-dimensional reconstruction of an object from projections*, J. Theoret. Biol., **36**(1972), 105-117.

[9] Gordon, R., Bender, R., Herman, G.T., *Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography*, J. Theoret. Biol., **29**(1970), 471-481.

[10] Gubin, L.G., Polyak, B.T., Raik, E.V., *The method of projections for finding the common point of convex sets*, USSR Comput. Math. Math. Phys., **7**(1967), no. 6, 1-24.

[11] Merzlyakov, Y.I., *On a relaxation method of solving systems of linear inequalities*, USSR Comput. Math. Math. Phys., **2**(1963), 504-510.

[12] Motzkin, T.S., Schoenberg, I.J., *The relaxation method for linear inequalities*, Canad. J. Math., **6**(1954), 393-404.

Irina Maria Artinescu
West University of Timişoara
Faculty of Mathematics and Computer Science
V. Parvan 4 Street, 300223 Timişoara, Romania
e-mail: `irina.artinescu84@e-uvt.ro`