

UNIVERSITATEA “OVIDIUS” DIN CONSTANȚA
ȘCOALA DOCTORALĂ DE ȘTIINȚE APLICATE
DOMENIUL DE DOCTORAT MATEMATICĂ

TEZĂ DE DOCTORAT

REZUMAT

METODE ITERATIVE DE TIP PROIECTIV
PENTRU PROBLEME ÎN SENSUL CELOR MAI
MICI PĂTRATE

Coordonator Științific
Prof. univ. dr. Constantin POPA

Student Doctorand
Ioana PANTELIMON

CONSTANȚA
2013

Prefață

În domeniul algebrei liniare numerice, diverse aplicații, precum reconstrucția și procesarea de imagini, conduc la rezolvarea unor sisteme de ecuații liniare rare, rău condiționate, având dimensiuni mari. Din cauza măsurărilor inexacte sau a aproximărilor mașină, aceste probleme nu sunt întotdeauna consistente, i.e., termenul liber nu aparține imaginii matricei sistemului. În astfel de cazuri ne interesează să rezolvăm o problemă corespunzătoare, în sensul celor mai mici pătrate. Algoritmii iterativi, deși au o complexitate de calcul mai mare, oferă, în general, rezultate mai bune atunci când datele sunt incomplete sau perturbate.

În lucrarea [31], o metodă iterativă generală, care include algoritmii Kaczmarz (vezi, e.g., [49]), Cimmino (vezi, e.g., [31]), Diagonal Weighting (vezi, e.g., [42]) sau Landweber (vezi, e.g., [30, 43]), a fost propusă pentru a rezolva problema în sensul celor mai mici pătrate. Autorii au demonstrat faptul că o versiune extinsă a metodei generale converge chiar și în cazul inconsistent la o soluție a problemei menționate anterior.

Pentru rezolvarea unei probleme de reconstrucție de imagini, atunci când există informații a priori despre imaginea scanată, precum netezime sau apartenență la o mulțime închisă și convexă, se pot aplica strategii de constrângere care îmbunătățesc calitatea reconstrucției. Constrângerea proceselor iterative se realizează prin înmulțirea la stânga a operatorului algoritmic cu un operator de constrângere la fiecare pas iterativ. Procedura descrisă are ca scop nu doar rezolvarea problemei, dar și găsirea unei soluții care înglobează cunoștințe inițiale. Această abordare a fost aplicată cu succes în reconstrucția și procesarea de imagini prin utilizarea unui singur operator de constrângere [7, 11, 14, 24, 25, 26, 28, 47, 50].

Un alt subiect de interes este reprezentat de creșterea performanței algoritmilor iterativi. În [45], autorii au propus o procedură prin care clasicul algoritm Kaczmarz este transformat într-o metodă directă.

Considerând aceste aspecte, teza este organizată după cum urmează.

În primul capitol prezentăm câteva rezultate preliminare precum și metoda generală introdusă în [31].

În Capitolul 2 încercăm să folosim o familie de funcții de constrângere, în locul unui singur operator. În Secțiunea 2.1 prezentăm pe scurt metoda iterativă generală constrânsă din [31] și rezultatele de convergență corespunzătoare. În continuare, propunem o procedură nouă de constrângere adaptivă, care utilizează o familie de funcții strict neexpansive și idempo-

tente și demonstrăm convergența noului algoritm, sub ipoteze suplimentare. De asemenea, dăm un exemplu de familie de proiecții pe “cutii”, i.e., produse de intervale închise, care satisfac condițiile impuse. În Secțiunea 2.2, adaptăm câteva rezultate din [13] într-un cadru generalizat, pentru o familie de operatori strict neexpansivi având puncte fixe comune și o presupunere adițională, adecvată în contextul problemelor de reconstrucție de imagini, și demonstrăm convergența procesului iterativ corespunzător. Prezintă algoritmul constrâns folosind o familie de operatori (FCA) și demonstrăm că metodele de dezvoltare în serie și matricele de netezire din [13] verifică toate ipotezele date. Mai mult, arătăm că metoda iterativă generală [31] este de asemenea un operator de forma necesară convergenței algoritmului FCA și că exemplul de familie neliniară de operatori de constrângere menționat anterior satisface condițiile noastre.

În Capitolul 3 încercăm să demonstrăm convergența unor variante extinse, pe blocuri, ale metodelor proiective Kaczmarz și Jacobi. Scopul nostru este să confirmăm că acești algoritmi sunt cazuri particulare ale procesului iterativ general extins dat în [31]. Înlocuind operatorul de inversare al unei matrice cu pseudoinversa Moore-Penrose, generalizăm rezultate precedente din [41].

Cel de-al patrulea capitol este dedicat studiului unei metode de accelerare care se bazează pe ideea de a adăuga sistemului liniar inițial direcții pe linii, respectiv coloane, pentru proiecție. În lucrarea [45] s-a demonstrat faptul că o versiune a algoritmului Kaczmarz construiește într-o singură iterație soluția problemei noastre în sensul celor mai mici pătrate. Din păcate, ca orice metodă directă aplicată unei matrice rare și de mari dimensiuni, acest algoritm determină o creștere considerabilă a procentului de umplere al datelor noastre. Pentru a depăși această dificultate, propunem o versiune modificată a metodei directe Kaczmarz, în care transformările aplicate matricei sistemului sunt menite să conserve structura de raritate inițială. Aceste transformări sunt realizate printr-un algoritm de clustering folosind distanțele Jaccard și Hamming. Metoda rezultată va fi iterativă, însă se obține o accelerare a convergenței față de algoritmul Kaczmarz clasic.

În ultimul capitol construim un algoritm adaptiv care utilizează o familie de funcții de constrângere pentru a rezolva o problemă de reconstrucție de imagini de tip “Tomographic Particle Image Velocimetry” (vezi, e.g., [35] și [34]). Rezultatele numerice sunt ilustrate pentru diferite niveluri de perturbare. Prezintă de asemenea și experimente numerice referitoare la metoda de accelerare a schemei iterative Kaczmarz descrisă în capitolul anterior. Investigăm în continuare metode de a calcula numărul minim de clustere

necesare pentru a conserva gradul de umplere al datelor noastre sub un prag dat.

Contribuțiile originale prezentate în această teză sunt conținute în următoarele lucrări.

[10] Y. Censor, **I. Pantelimon**, and C. Popa. Family constraining of iterative algorithms. *Numerical Algorithms*, pages 1–16, 2013. doi: 10.1007/s11075-013-9736-5

[33] **I. Pantelimon** and C. Popa. Constraining by a family of strictly non-expansive idempotent functions with applications in image reconstruction. *BIT Numerical Mathematics*, 53(2):527–544, 2013. doi: 10.1007/s10543-012-0414-0

[36] **I. Pomparău**. On the acceleration of Kaczmarz projection algorithm. Presented at the 84th Annual Meeting of the International Association of Applied Mathematics and Mechanics, 18–22 March, 2013, Novi Sad, Serbia. Submitted for publication in *Proceedings in Applied Mathematics and Mechanics*, 2013

[37] **I. Pomparău**. On an accelerated version of the Kaczmarz algorithm with a priori clustering. In *Topics in mathematical modelling of life science problems*, pages 61–76. Matrix Rom, București, 2013. Presented at the Ninth Workshop on Mathematical Modeling of Environmental and Life Sciences Problems, 1–4 November, 2012, Constanța, România

[38] **I. Pomparău** and C. Popa. Supplementary projections for the acceleration of Kaczmarz algorithm. Submitted for publication, 2013

[39] **I. Pomparău** and C. Popa. Weaker assumptions for convergence of extended block Kaczmarz and Jacobi projection algorithms. Submitted for publication, 2013

1 Cadru teoretic

1.1 Noțiuni preliminare

Multe probleme reale conduc, după anumiți pași de discretizare, la rezolvarea unor sisteme de ecuații liniare de forma $Ax = b$, pentru o matrice A de dimensiuni $m \times n$ și un vector $b \in \mathbb{R}^m$. Dacă apar erori de modelare sau de măsurătoare, atunci este posibil ca sistemul să fie inconsistent. În acest caz se ajunge la a rezolva următoarea problemă în sensul celor mai mici pătrate: găsiți $x \in \mathbb{R}^n$ astfel încât

$$\|Ax - b\| = \min\{\|Az - b\|, z \in \mathbb{R}^n\}. \quad (1)$$

Vom folosi în continuare notațiile $A^T, A^\dagger, A_i, A^j, \mathcal{R}(A), \mathcal{N}(A), P_V, x_{LS}$ și $LSS(A; b)$ pentru transpusa, pseudoinversa (unică) Moore-Penrose, cea de-a i -a linie, cea de-a j -a coloană, imaginea și spațiul nul al lui A , proiecția pe un spațiu vectorial V , soluția (unică) de normă minimă și mulțimea de soluții a problemei (1); de asemenea, $\langle \cdot, \cdot \rangle$, $\|z\|$ și $\|M\|$ au sensul de produs scalar Euclidian, norma Euclidiană a vectorului z și norma spectrală a matricei M . Toți vectorii care apar în această lucrare vor fi considerați vectori coloană. Vom presupune în restul lucrării că

$$A_i \neq 0, A^j \neq 0, \text{ oricare ar fi } i \in \{1, 2, \dots, m\} \text{ și } j \in \{1, 2, \dots, n\}.$$

1.2 O metodă iterativă generală de tip proiectiv

Pentru a rezolva problema în sensul celor mai mici pătrate (1), următorul algoritm iterativ general, bazat pe proiecții, a fost propus în [31].

Algoritm Iterativ General(GEN)

Inițializare. $x^0 \in \mathbb{R}^n$;

Pas iterativ.

$$x^{k+1} = Qx^k + Rb, \quad (2)$$

cu Q și R matrice de dimensiuni $n \times n$ și respectiv $n \times m$, care satisfac următoarele proprietăți:

$$Q + RA = I, \quad (3)$$

$$\forall y \in \mathbb{R}^m, Ry \in \mathcal{R}(A^T), \quad (4)$$

$$\text{dacă } \tilde{Q} = QP_{\mathcal{R}(A^T)} \text{ atunci } \|\tilde{Q}\| < 1. \quad (5)$$

Observația 1 Formularea generală descrisă mai sus include aproape toți algoritmi de tip proiectiv folosiți în tomografia computerizată, cum ar fi Kaczmarz, Cimmino, Jacobi Proiectiv, Diagonal Weighting (vezi, e.g., [49], [31], [15] și respectiv [42]). Pentru detalii și demonstrații ale acestei afirmații vezi lucrarea [43].

Teorema 1 ([31, Teorema 2.2]) Dacă (3)-(5) au loc, pentru orice $x^0 \in \mathbb{R}^n$, șirul $(x^k)_{k \geq 0}$ generat cu (2) converge și

$$\lim_{k \rightarrow \infty} x^k = P_{\mathcal{N}(A)}(x^0) + x_{LS} + \Delta, \quad \text{cu} \quad \Delta = (I - \tilde{Q})^{-1} R P_{\mathcal{N}(A^T)}(b). \quad (6)$$

Observația 2 Dacă problema (1) este consistentă, avem că $\Delta = 0$ și limita din (6) este un element al mulțimii de soluții $S(A; b) = \{x \in \mathbb{R}^n \mid Ax = b\}$, pentru oricare ar fi $x^0 \in \mathbb{R}^n$. În cazul inconsistent, $\|\Delta\|$ reprezintă distanța de la punctul limită din (6) și mulțimea $LSS(A; b)$ (pentru detalii vezi, e.g., [44]).

Pentru matricele Q , R și U de dimensiuni $n \times n$, $n \times m$ și respectiv $m \times m$, autorii au definit în [31] următoarea metodă iterativă generală extinsă.

Algoritm General Extins (EGEN)

Inițializări. $x^0 \in \mathbb{R}^n$ este arbitrar ales și $y^0 = b$.

Pas iterativ. Oricare ar fi $k \geq 0$,

$$y^{k+1} = U y^k, \quad (7)$$

$$b^{k+1} = b - y^{k+1}, \quad (8)$$

$$x^{k+1} = Q x^k + R b^{k+1}. \quad (9)$$

Dacă Q și R satisfac (3)-(5) și U verifică următoarele presupuneri generale

$$\text{dacă } x \in \mathcal{N}(A^T) \text{ atunci } Ux = x, \quad (10)$$

$$\text{dacă } x \in \mathcal{R}(A) \text{ atunci } Ux \in \mathcal{R}(A), \quad (11)$$

$$\text{dacă } \tilde{U} = U P_{\mathcal{R}(A)} \text{ atunci } \|\tilde{U}\| < 1, \quad (12)$$

următorul rezultat de convergență a fost demonstrat.

Teorema 2 [31, Teorema 2.6] Presupunem că matricele Q și R satisfac ecuațiile (3)-(5) și că pentru U proprietățile (10)-(12) sunt îndeplinite. Atunci, oricare ar fi $x^0 \in \mathbb{R}^n$, șirul $(x^k)_{k \geq 0}$ generat cu algoritmul EGEN converge și

$$\lim_{k \rightarrow \infty} x^k = P_{\mathcal{N}(A)}(x^0) + x_{LS}. \quad (13)$$

2 Utilizarea unor familii de funcții de constrângere în algoritmi iterativi

În acest capitol ne concentrăm pe studiul tehnicilor de constrângere a proceselor iterative. Atunci când ne ocupăm de probleme reale, există situații în care cunoaștem a priori proprietăți ale soluției pe care o căutăm. Dacă este posibil, aceste cunoștințe pot fi formulate sub forma unor constrângeri suplimentare adăugate problemei. Însă, în anumite cazuri este mai avantajos să modificăm algoritmul și nu problema, astfel încât să ia în considerare condițiile impuse de constrângeri și în același timp să aproximeze o soluție a problemei inițiale. Numim acest procedeu, constrângere a algoritmului iterativ. Fiind dat un operator (algoritm) $\Gamma : R^n \rightarrow R^n$ între spații Euclidiene, procesul iterativ original poate fi scris sub forma

$$x^{k+1} = \Gamma(x^k), \text{ oricare ar fi } k \geq 0, \quad (14)$$

sub diferite ipoteze asupra lui Γ .

Prin a constrânge un astfel de algoritm folosind o familie de operatori înțelegem să înlocuim (14) cu următorul proces iterativ

$$x^{k+1} = C_k \Gamma(x^k), \text{ oricare ar fi } k \geq 0, \quad (15)$$

unde $\{C_k\}_{k=0}^\infty$ este o familie de operatori $C_k : R^n \rightarrow R^n$, pe care îi vom numi în restul acestei lucrări operatori de constrângere.

Ne interesează să studiem posibilitatea de a constrânge un algoritm folosind o familie de operatori și să analizăm comportamentul asimptotic al acestuia. Extindem rezultate anterioare, limitate la un singur operator de constrângere i.e., $C_k = C$, oricare ar fi $k \geq 0$, vezi, e.g., [7, 11, 14, 24, 25, 26, 28, 47, 50].

2.1 O metodă de constrângere a algoritmului GEN folosind o familie de funcții strict neexpansive și idempotente

În general, matricea A din (1) are dimensiuni mari, este rară, rău condiționată și nu are rang plin. Prin urmare, problema în sensul celor mai mici pătrate are o infinitate de soluții. O serie de algoritmi clasici folosiți pentru a rezolva (1) generează o aproximare a soluției (unice) de normă minimă. Însă, x_{LS}

nu este întotdeauna apropiată ca valoare de soluția pe care o căutăm (e.g., imaginea scanată în probleme de reconstrucție de imagini) și pe care o vom nota cu x^{ex} . Următoarea relație este cunoscută, vezi, e.g., [43],

$$x^{ex} = P_{N(A)}(x^{ex}) + x_{LS}. \quad (16)$$

În consecință, pentru a găsi o aproximare a soluției exacte mai bună decât cea furnizată de x_{LS} , se încearcă înglobarea informațiilor cunoscute a priori sub forma unor constrângeri suplimentare.

Un exemplu frecvent întâlnit este acela în care soluția exactă are componentele incluse în intervale cunoscute, i.e., $x_i^{ex} \in [a_i, b_i]$, cu $[a_i, b_i] \subset \mathbb{R}$, oricare ar fi $i \in \{1, 2, \dots, n\}$ (vezi, e.g., [20]). În acest caz, la fiecare pas al metodei iterative se va proiecta a i -a componentă a aproximației curente pe intervalul $[a_i, b_i]$. În lucrarea [28], autorii au implementat această idee algoritmului Kaczmarz. Aceștia au considerat o funcție de constrângere generală $C : \mathbb{R}^n \rightarrow \mathbb{R}^n$ cu o imagine închisă $\text{Im}(C) = \{y = Cx, x \in \mathbb{R}^n\} \subset \mathbb{R}^n$, având următoarele proprietăți

$$\|Cx - Cy\| \leq \|x - y\|, \quad (17)$$

$$\text{dacă } \|Cx - Cy\| = \|x - y\| \text{ atunci } Cx - Cy = x - y, \quad (18)$$

$$\text{dacă } y \in \text{Im}(C) \text{ atunci } y = Cy. \quad (19)$$

Spunem că un operator $C : \mathbb{R}^n \rightarrow \mathbb{R}^n$ este strict neexpansiv (SNE) dacă îndeplinește condițiile (17) și (18).

Observația 3 *Un exemplu de funcție pentru care (17)-(19) au loc este proiecția ortogonală pe “cutia” $[a, b] = [a_1, b_1] \times \dots \times [a_n, b_n] \subset \mathbb{R}^n$, definită astfel*

$$(Cx)_i = \begin{cases} x_i, & \text{dacă } x_i \in [a_i, b_i] \\ a_i, & \text{dacă } x_i < a_i \\ b_i, & \text{dacă } x_i > b_i. \end{cases} \quad (20)$$

În lucrarea [31], pentru o funcție $C : \mathbb{R}^n \rightarrow \mathbb{R}^n$ cu o imagine închisă $\text{Im}(C) \subset \mathbb{R}^n$ și care satisface proprietățile (17)-(19), autorii au propus un algoritm GEN constrâns. Însă aplicații practice indică faptul că o procedură “uniformă” de constrângere (se folosește un singur operator de constrângere) nu este întotdeauna eficientă. Câteodată este necesar să se ia în considerare un proces adaptiv de constrângere, actualizat de la o iterație la următoarea. În această secțiune, intenția noastră este să construim un asemenea procedeu, utilizând, la fiecare pas $k \geq 0$, o altă funcție $C_k : \mathbb{R}^n \rightarrow \mathbb{R}^n$ cu o imagine închisă $\text{Im}(C_k) \subset \mathbb{R}^n$ și care satisface (17)-(19).

2.1.1 O familie de funcții de constrângere

Fie $(C_k)_{k \geq 0}$ o familie de funcții de constrângere pentru care (17)-(19) și următoarea proprietate au loc

$$(A_1) \quad V = \bigcap_{k \geq 0} \text{Im}(C_k) \neq \emptyset.$$

Dacă presupunem că $\mathcal{V}_\infty^* = \{y \in V, y - \Delta \in LSS(A; b)\} \neq \emptyset$ și definim, pentru orice $k \geq 0$, $\mathcal{V}_k^* = \{z \in \text{Im}(C_k), z - \Delta \in LSS(A; b)\}$, obținem $\mathcal{V}_\infty^* = \bigcap_{k \geq 0} \mathcal{V}_k^*$.

Propunem următorul algoritm.

Algoritm General cu Familie de Constrângeri (CGENk)

Inițializare. $x^0 \in \text{Im}(C_0)$;

Pas iterativ. Oricare ar fi $k \geq 0$,

$$x^{k+1} = C_{k+1}[Qx^k + Rb]. \quad (21)$$

Considerăm în continuare două ipoteze suplimentare asupra familiei de funcții de constrângere $(C_k)_{k \geq 0}$.

(A₂) Mulțimile $V_k = \text{Im}(C_k)$ sunt închise;

(A₃) Pentru fiecare $\ell \in \mathbb{N}^*$, există $k(\ell) \geq \ell$ astfel încât oricare ar fi $y \in \mathcal{V}_\infty^*$ și $k \geq k(\ell)$ avem inegalitatea

$$\|C_{k+1}[Qx^k + Rb] - y\| \leq \|C_\ell[Qx^k + Rb] - y\|.$$

Teorema 3 În ipotezele de mai sus, șirul $(x^k)_{k \geq 0}$ generat cu (21) converge și

$$\lim_{k \rightarrow \infty} x^k \in \mathcal{V}_\infty^*. \quad (22)$$

2.1.2 O familie particulară de funcții de constrângere care satisface ipotezele suplimentare

Reamintim exemplul prezentat în Observația 3. Oricare ar fi $k \geq 0$, considerăm “cutia” $[a^k, b^k] = [a_1^k, b_1^k] \times \cdots \times [a_n^k, b_n^k] \subset \mathbb{R}^n$ și proiecția metrică pe cea de-a k -a “cutie” $[a^k, b^k]$. Este cunoscut faptul că un operator definit prin (20) verifică proprietățile (17)-(19) și (A₂). Vom încerca să construim o condiție suficientă pentru ca un astfel de operator să îndeplinească (A₁) și (A₃).

Propoziția 1 *Fiind dată o familie $(C_k)_{k \geq 0}$ de proiecții metrice pe cea de-a k -a “cutie” $[a^k, b^k]$ definite de (20) și două matrice Q și R având dimensiunile $n \times n$, respectiv $n \times m$, pentru care (3)-(5) au loc, presupunem că, oricare ar fi $\ell \geq 0$, există $k(\ell) \geq \ell$ astfel încât:*

$$\text{Im}(C_{k+1}) \subset \text{Im}(C_\ell), \quad \forall k \geq k(\ell). \quad (23)$$

Atunci proprietățile (A_1) și (A_3) sunt îndeplinite.

2.2 Constrângerea proceselor iterative într-un cadru general

În această secțiune vom introduce o familie de operatori strict neexpansivi $\{C_k\}_{k=0}^\infty$ și vom demonstra convergența unui algoritm căruia i se asociază o familie de funcții de constrângere într-un cadru general. Această abordare este aplicată cu succes unor probleme de reconstrucție de imagini, de netezire (vezi și [20, Subsecțiunea 12.3]), dar și constrângerii unor procese iterative liniare oarecare.

În ipoteza că o familie de operatori strict neexpansivi $\{T_k\}_{k=0}^\infty$ are puncte fixe comune și o condiție suplimentară, rezonabilă în domeniul reconstrucției de imagini, demonstrăm că șirul generat de schema iterativă

$$x^0 \in R^n \text{ și } x^{k+1} = T_{k+1}(x^k), \quad \forall k \geq 0,$$

converge la un punct fix comun al operatorilor $\{T_k\}_{k=0}^\infty$.

2.2.1 Convergența unei familii de operatori strict neexpansivi

Pentru o familie $\{T_k\}_{k=0}^\infty$ de operatori strict neexpansivi definim mai jos mulțimea punctelor fixe comune și intersecția acestora

$$\text{Fix}(T_k) = F_k := \{x \in R^n \mid T_k(x) = x\} \text{ și } F := \bigcap_{k=0}^\infty F_k. \quad (24)$$

Presupunem că

$$F \neq \emptyset. \quad (25)$$

Considerăm algoritmul

$$x^0 \in R^n \text{ și } x^{k+1} = T_{k+1}(x^k), \quad \forall k \geq 0, \quad (26)$$

Condiția 1 Fie $\{T_k\}_{k=0}^{\infty}$ o familie de operatori strict neexpansivi pentru care (25) are loc. Dacă $\{x^k\}_{k=0}^{\infty}$ este un șir generat de (26), atunci, oricare ar fi $\ell \geq 0$, există un index $k(\ell) \geq 0$ astfel încât

$$\|T_{k+1}(x^k) - z\| \leq \|T_{\ell}(x^k) - z\|, \quad (27)$$

pentru orice $z \in F$ și $k \geq k(\ell)$.

Teorema 4 Fie $\{T_k\}_{k=0}^{\infty}$ o familie de operatori strict neexpansivi pentru care (25) și Condiția 1 au loc. Orice șir $\{x^k\}_{k=0}^{\infty}$, construit conform (26), converge la un element din F .

Observația 4 Înlocuind strict neexpansivitatea operatorilor $\{T_k\}_{k=0}^{\infty}$ cu ipoteza mai slabă că aparțin clasei de operatori “paracontracting” (vezi [18, Definiția 1]), rezultatele enunțate în Teorema 4 rămân adevărate.

2.2.2 Algoritmul FCA

Dacă $\Gamma : R^n \rightarrow R^n$ și $C_k : R^n \rightarrow R^n$, cu $k \geq 0$, sunt strict neexpansive, definim mai jos operatorii $T_k : R^n \rightarrow R^n$ sub forma

$$T_k(x) := C_k \Gamma(x), \text{ oricare ar fi } k \geq 0, \quad (28)$$

și demonstrăm ca sunt de asemenea strict neexpansivi.

Propoziția 2 Oricare ar fi $k \geq 0$, un operator T_k definit prin (28), pentru care Γ și C_k sunt strict neexpansivi, are următoarele proprietăți:

$$\|T_k(x) - T_k(y)\| \leq \|x - y\|, \text{ oricare ar fi } x, y \in R^n, \quad (29)$$

și

$$\text{dacă } \|T_k(x) - T_k(y)\| = \|x - y\|, \text{ atunci } T_k(x) - T_k(y) = \Gamma(x) - \Gamma(y) = x - y. \quad (30)$$

Pentru $\{T_k\}_{k=0}^{\infty}$ definiți conform (28), cu $\{C_k\}_{k=0}^{\infty}$ și Γ strict neexpansivi, procesul iterativ (26) poate fi scris ca un algoritm cu constrângeri.

Algoritm FCA

Inițializare. $x^0 \in R^n$ este ales arbitrar.

Pas iterativ. Oricare ar fi $k \geq 0$, fiind dată aproximația curentă x^k calculați x^{k+1} astfel

$$x^{k+1} = C_{k+1}\Gamma(x^k). \quad (31)$$

Dacă ipotezele (25) și Condiția 1 sunt îndeplinite, atunci Propoziția 2 și Teorema 4 determină faptul că orice șir generat cu algoritmul FCA converge la un element din F .

Definiția 1 [13, Definiția 1] Fie $\mathcal{F}_1$ mulțimea de funcții continue $\Gamma : R^n \rightarrow R^n$ care satisfac

$$\|\Gamma(x) - \Gamma(y)\| \leq \|x - y\|, \quad \text{oricare ar fi } x, y \in R^n. \quad (32)$$

și

$$\begin{aligned} &\text{dacă } \|\Gamma(x) - \Gamma(y)\| = \|x - y\|, \text{ atunci} \\ &\Gamma(x) - \Gamma(y) = x - y \text{ și } \langle x - y, \Gamma(y) - y \rangle = 0. \end{aligned} \quad (33)$$

Definiția 2 [13, Definiția 2] Fie $\mathcal{F}_2$ mulțimea de operatori $\Gamma \in \mathcal{F}_1$ cu proprietatea că pentru orice $S \in R^{n \times n}$, funcția $g : R^n \rightarrow R$ definită prin $g(x) := \|x - S\Gamma(x)\|^2$ își atinge minimul global.

Propoziția 3 Dacă S este o matrice simetrică, stocastică, având diagonala strict pozitivă, atunci familia $\{C_k\}_{k=0}^\infty$ cu $C_k = S$, pentru oricare ar fi $k \geq 0$, este strict neexpansivă.

2.2.3 Rezolvarea unei probleme în sensul celor mai mici pătrate

Vom demonstra în continuare că metoda iterativă GEN [31], prezentată în Secțiunea 1.2, este strict neexpansivă și, în plus, că aparține mulțimii $\mathcal{F}_2$.

Propoziția 4 Dacă A și b sunt conform cu (1) și matricele Q , R și A satisfac proprietățile (3)–(5), atunci, operatorul afin $\Gamma : R^n \rightarrow R^n$ definit prin

$$\Gamma(\cdot) := Q(\cdot) + Rb \quad (34)$$

aparține mulțimii $\mathcal{F}_2$.

Observația 5 Algoritmul FCA, cu T_k definit conform (28), $C_k = I$, Γ din (34), cu Q, R îndeplinind (3)-(5), include algoritmii Kaczmarz (vezi, e.g., [49]), Cimmino (vezi, e.g., [31]) și Diagonal Weighting (vezi, e.g., [42]) (pentru detalii și demonstrații ale acestei afirmații vezi [43]). În următorul rezultat demonstrăm că un alt astfel de exemplu este reprezentat de metoda Landweber (vezi, e.g., [30, 43]).

Propoziția 5 Fie $\{\omega_k\}_{k=0}^\infty \subset \mathbb{R}^n$ astfel încât există un număr real ϵ cu $0 < \epsilon \leq \omega_k \leq \frac{2}{\rho(A)^2} - \epsilon$, unde prin $\rho(A)$ am notat norma spectrală a matricei A . Oricare ar fi $x^0 \in \mathbb{R}^n$ și $k \geq 0$, procesul iterativ Landweber este definit prin

$$x^{k+1} = (I - \omega_k A^T A)x^k + \omega_k A^T b. \quad (35)$$

Dacă notăm $I - \omega_k A^T A$ cu T_k și $\omega_k A^T$ cu R_k , atunci, oricare ar fi $k \geq 0$, proprietățile (3)-(5) sunt verificate.

Lema 1 Fie $\text{Fix}(\Gamma)$ mulțimea punctelor fixe ale operatorului Γ definit conform (34), cu Q și R matrice de dimensiuni $n \times n$, respectiv $n \times m$, și proprietățile (3)-(5). În aceste ipoteze, următoarea proprietate este adevărată

$$\text{Fix}(\Gamma) = \{x + \Delta \mid x \in LSS(A; b)\}, \text{ cu } \Delta = (I - \tilde{Q})^{-1} R P_{\mathcal{N}(A^T)}(b). \quad (36)$$

Lema 2 Fie $\{C_k\}_{k=0}^\infty$ familie de proiecții metrice pe a k -a “cutie” $[a_k, b_k] \subset \mathbb{R}^n$, conform definiției (20) și să presupunem că $\mathcal{V}_k^* \neq \emptyset$, oricare ar fi $k \geq 0$. Dacă pentru orice $\ell \geq 0$ există $k(\ell) \geq \ell$ astfel încât condiția (23) este verificată, i.e.,

$$\text{Im}(C_{k+1}) \subseteq \text{Im}(C_\ell), \text{ oricare ar fi } k \geq k(\ell),$$

atunci mulțimea

$$\mathcal{V}_\infty^* := \bigcap_{k=0}^\infty \mathcal{V}_k^*, \quad (37)$$

este nevidă.

Proiecțiile metrice definite conform (20) sunt frecvent folosite cu rol de constrângeri în probleme de reconstrucție de imagini exprimate sub forma (1). Scopul nostru este de a studia o procedură prin care funcția de constrângere să fie aleasă adaptiv la fiecare iterație a algoritmului aplicat, pentru a obține o aproximare mai bună a imaginii scanate. Semnificația relației (23) în practică este dată de faptul că imaginea fiecărei funcții de constrângere ar trebui să fie construită pe baza unor cunoștințe anterioare. Aceasta trebuie să conțină soluția exactă, iar $\{\text{Im}(C_k)\}_{k=0}^\infty$ nu trebuie să fie obligatoriu un șir descrescător.

Propoziția 6 Pentru o familie $\{C_k\}_{k=0}^{\infty}$ de proiecții metrice definite prin (20), având proprietățile $\mathcal{V}_k^* \neq \emptyset$ și (23), și un operator Γ definit conform (34), cu Q și R matrice care satisfac (3)–(5), ipoteza (25) și Condiția 1 sunt îndeplinite.

În concluzie, conform Propoziției 4, Lemei 1, Propoziției 6 și Teoremei 4, putem rezolva problema (1) în sensul celor mai mici pătrate folosind Algoritmul 2.2.2 cu Γ definit ca în (34), Q și R matrice pentru care au loc (3)–(5) și o familie $\{C_k\}_{k=0}^{\infty}$ de proiecții metrice din (20) satisfăcând proprietățile $\mathcal{V}_k^* \neq \emptyset$ și (23).

3 Ipoteze mai slabe pentru convergența metodelor proiective extinse și pe blocuri Kaczmarz și Jacobi

În acest capitol ne interesează să aproximăm o soluție a problemei (1) în cazul inconsistent, folosind algoritmi proiectivi pe blocuri Kaczmarz și Jacobi.

Versiuni extinse ale acestor două metode au fost introduse în [41] și rezultate de convergență au fost demonstrate în ipoteze de nesingularitate asupra matricelor rezultate din descompuneri pe blocuri de linii și coloane.

Vom arăta că, înlocuind operatorul de inversare al unei matrice cu pseudoinversa Moore-Penrose, rezultatele menționate anterior, vezi [41, Teorema 3.4 și Teorema 6.7] rămân adevărate în lipsa ipotezelor de nesingularitate.

3.1 Rezultate preliminare

Vom considera în continuare o descompunere pe blocuri de linii ale matricei A și ale vectorului corespunzător b . În acest sens, fie $p \geq 2$, $1 \leq m_i \leq m$, cu $i \in \{1, 2, \dots, p\}$, astfel încât $m_1 + m_2 + \dots + m_p = m$,

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_p \end{bmatrix} \quad \text{și} \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_p \end{bmatrix}, \quad (38)$$

unde A_i sunt matrice reale de dimensiune $m_i \times n$ și $b_i \in \mathbb{R}^{m_i}$.

Similar, pentru $q \geq 2$ și $n = n_1 + n_2 + \dots + n_q$, cu $1 \leq n_j < n$, oricare ar fi $j \in \{1, 2, \dots, q\}$, descompunerea în blocuri de coloane a matricei A este dată de

$$A^T = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_q \end{bmatrix}, \quad (39)$$

unde B_j sunt matrice reale de dimensiune $n_j \times m$.

În lucrarea [41], autorii au introdus versiuni extinse, pe blocuri, ale algoritmilor Kazmarz și Jacobi proiectiv cu parametri de relaxare. Aceștia au demonstrat că în ipotezele

$$\det(A_i A_i^T) \neq 0, \quad \forall i \in \{1, 2, \dots, p\} \quad (40)$$

și

$$\det(B_j B_j^T) \neq 0, \quad \forall j \in \{1, 2, \dots, q\} \quad (41)$$

cele două metode converg la un element al mulțimii de soluții în sensul celor mai mici pătrate ale problemei (1).

Fie aplicațiile liniare $f_0^i(b; \cdot), F_0(b; \cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ definite mai jos

$$f_0^i(b; x) = x + A_i^T (A_i A_i^T)^{-1} (b_i - A_i x), \quad \forall i \in \{1, 2, \dots, p\}, \quad (42)$$

$$F_0(b; x) = (f_0^1 \circ f_0^2 \circ \dots \circ f_0^p)(b; x) \quad (43)$$

și

$$\Phi_0 = \prod_{j=1}^q (I - A_j^T (A_j A_j^T)^{-1} A_j). \quad (44)$$

Următorul algoritm a fost introdus în [41].

Algoritmul pe Blocuri Kaczmarz Extins(EBK)

Inițializări. $x^0 \in \mathbb{R}^n$ este ales arbitrar și $y^0 = b$.

Pas iterativ. Oricare ar fi $k \geq 0$,

$$y^{k+1} = \Phi_0 y^k; \quad b^{k+1} = b - y^{k+1}; \quad x^{k+1} = F_0(b^{k+1}; x^k). \quad (45)$$

Pentru parametrii reali $\omega, \alpha \neq 0$, vom considera

$$Q_0^\omega = I - \omega \sum_{i=1}^p A_i^T (A_i A_i^T)^{-1} A_i, \quad (46)$$

$$R_0^\omega = \omega \text{col} \left[A_1^T (A_1 A_1^T)^{-1} \mid A_2^T (A_2 A_2^T)^{-1} \mid \dots \mid A_p^T (A_p A_p^T)^{-1} \right] \quad (47)$$

și

$$\Phi_0^\alpha = I - \alpha \sum_{j=1}^q B_j^T (B_j B_j^T)^{-1} B_j. \quad (48)$$

O versiune extinsă, pe blocuri, a metodei Jacobi a fost introdusă în [41].

Algoritmul pe Blocuri Jacobi Extins cu Parametri de Relaxare (EBJRP)

Inițializări. $x^0 \in R^n$ este ales arbitrar și $y^0 = b$.

Pas iterativ. Oricare ar fi $k \geq 0$,

$$y^{k+1} = \Phi_0^\alpha y^k; \quad b^{k+1} = b - y^{k+1}; \quad x^{k+1} = Q_0^\omega x^k + R_0^\omega b^{k+1}. \quad (49)$$

Din nefericire, în aplicații reale, condiții de tipul (40)-(41) sunt în general greu de verificat. Urmărind argumentele din [41], vom demonstra că înlocuind operatorul de inversare al unei matrice cu pseudoinversa Moore-Penrose, rezultatele de convergență rămân adevărate în absența presupunerilor (40)-(41). Realizăm acest lucru arătând că cei doi algoritmi sunt cazuri particulare ale metodei extinse generale EGEN [45], descrisă în Secțiunea 1.2.

3.2 Algoritmii EBK și EBJRP, cazuri particulare ale metodei EGEN

Vom nota în continuare cu f^i , F , Φ , Q^ω , R^ω și Φ^α , operatorii liniari f_0^i , F_0 , Φ_0 , Q_0^ω , R_0^ω și Φ_0^α definiți conform (42)-(44) și respectiv (46)-(48), înlocuind operatorul de inversare al unei matrice cu pseudoinversa Moore-Penrose.

Oricare ar fi $i \in \{1, 2, \dots, p\}$ și $j \in \{1, 2, \dots, q\}$, definim matricele

$$P_i = I - A_i^T (A_i A_i^T)^\dagger A_i, \quad \phi_j = I - B_j^T (B_j B_j^T)^\dagger B_j, \quad (50)$$

$$\bar{P}_i = A_i^T (A_i A_i^T)^\dagger A_i, \quad \bar{\phi}_j = B_j^T (B_j B_j^T)^\dagger B_j. \quad (51)$$

Considerăm aplicațiile liniare corespunzătoare

$$\Delta_i = A_i^T (A_i A_i^T)^\dagger, \quad Q_i = P_1 P_2 \dots P_i, \quad \forall i \in \{1, 2, \dots, p\}, \quad (52)$$

$$Q = P_1 P_2 \dots P_p, \quad (53)$$

$$R = \text{col} [\Delta_1 \mid Q_1 \Delta_2 \mid \dots \mid Q_{p-1} \Delta_p], \quad (54)$$

$$\Phi_j = \phi_1 \phi_2 \dots \phi_j, \quad \forall j \in \{1, 2, \dots, q\}, \quad (55)$$

$$\tilde{Q}^\omega = Q^\omega P_{\mathcal{R}}(A^T) \text{ and } \tilde{\Phi}^\alpha = \Phi^\alpha P_{\mathcal{R}}(A). \quad (56)$$

Prin urmare, avem

$$\Phi = \phi_1 \phi_2 \dots \phi_q, \quad Q^\omega = I - \omega \sum_{i=1}^p \bar{P}_i, \quad (57)$$

și

$$R^\omega = \omega \text{col} [\Delta_1 \mid \Delta_2 \mid \dots \mid \Delta_p], \quad \Phi^\alpha = I - \alpha \sum_{j=1}^q \bar{\phi}_j. \quad (58)$$

Următoarea teoremă garantează convergența algoritmului Kaczmarz extins pe blocuri definit utilizând pseudoinversa Moore-Penrose și eliminând ipotezele (40)-(41).

Teorema 5 *Dacă Q , R și Φ sunt definite conform (53), (54) și respectiv (57), atunci*

(i) *avem egalitatea*

$$F(b; x) = Qx + Rb, \quad (59)$$

(ii) *Q și R satisfac (3)-(5),*

(iii) *matricea Φ satisface proprietățile (10)-(12).*

În cazul algoritmului EBJRP vom confirma rezultate analoage. Afirmațiile (5) și (12) vor fi demonstrate folosind rezultate din lucrările [15, 41].

Teorema 6 *Următoarele proprietăți sunt îndeplinite*

(i) *Q^ω și R^ω verifică (3)-(4),*

(ii) *pentru matricea Φ^α , presupunerile (10)-(11) sunt adevărate.*

Corolarul 1 [41] *Dacă $0 < \omega < \frac{2}{\rho(E)}$ și $0 < \alpha < \frac{2}{\rho(D)}$, unde $D = \frac{1}{\alpha}(I - \Phi^\alpha)$ și $E = \frac{1}{\omega}R^\omega A$, atunci, pentru $\tilde{Q}^\omega$ și $\tilde{\Phi}^\alpha$, relațiile (5) și respectiv (12) au loc.*

Din Teorema 6 și Corolarul 1 rezultă că algoritmul EBJRP este un caz particular al metodei EGEN.

4 Proiecții suplimentare pentru accelerarea algoritmului Kaczmarz

În lucrarea [45], autorii au observat faptul că algoritmul Kaczmarz Extins calculează o soluție a problemei (1) într-o singură iterație, cu condiția ca matricea A să îndeplinească anumite condiții. Prin urmare, aceștia au studiat o procedură de a transforma problema originală adăugând o mulțime convenabilă de direcții pentru proiecție, construite sub forma unor combinații liniare de linii și coloane ale matricei sistemului. Deși matricele obținute astfel vor verifica ipotezele necesare, un dezavantaj este acela că vor avea în general un procent de umplere ridicat comparativ cu cel al matricei originale.

Obiectivul nostru este de a studia posibilitatea adăugării unei submulțimi de direcții suplimentare pentru proiecție astfel încât procentul de umplere al matricelor extinse să fie menținut aproape de cel al lui A . Deoarece în urma acestui procedeu se va obține o metodă iterativă, vom încerca să obținem o viteză de convergență mai bună față de algoritmul Kaczmarz clasic.

4.1 Algoritmul Kaczmarz Direct Extins

În [45], autorii au introdus algoritmul Kaczmarz Direct Extins (DEK) și au demonstrat convergența acestuia în cazul inconsistent al problemei (1). Ideea este de a se transforma problema originală prin construcția de direcții noi pe linii și coloane, după cum urmează. Oricare ar fi $i \in \{1, 2, \dots, m\}$ și $j \in \{1, 2, \dots, n\}$, fie aplicațiile $P_i : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $\phi_j : \mathbb{R}^m \rightarrow \mathbb{R}^m$, definite prin $P_i(x) = x - \frac{x^T A_i}{\|A_i\|^2} A_i$, $\phi_j(x) = x - \frac{x^T A^j}{\|A^j\|^2} A^j$. Considerăm mulțimile de vectori direcție $d_1, d_2, \dots, d_{m-1} \in \mathbb{R}^n$ și $\delta_1, \delta_2, \dots, \delta_{n-1} \in \mathbb{R}^m$ definite ([45]) mai jos

$$\begin{aligned} d_{m-1} &= P_m(A_{m-1}), \\ d_{m-2} &= P_{m-1} P^{d_{m-1}} P_m(A_{m-2}), \\ &\dots\dots\dots \\ d_1 &= P_2 P^{d_2} P_3 \dots P_{m-1} P^{d_{m-1}} P_m(A_1), \end{aligned} \tag{60}$$

$$\begin{aligned} \delta_{n-1} &= \phi_n(A^{n-1}), \\ \delta_{n-2} &= \phi_{n-1} \phi^{\delta_{n-1}} \phi_n(A^{n-2}), \\ &\dots\dots\dots \\ \delta_1 &= \phi_2 \phi^{\delta_2} \phi_3 \dots \phi_{n-1} \phi^{\delta_{n-1}} \phi_n(A^1), \end{aligned} \tag{61}$$

unde operatorii $P^{d_i} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ și $\phi^{\delta_j} : \mathbb{R}^m \rightarrow \mathbb{R}^m$ sunt dați de $P^{d_i}(x) = x - \frac{x^T d_i}{\|d_i\|^2} d_i$, $\phi^{\delta_j}(x) = x - \frac{x^T \delta_j}{\|\delta_j\|^2} \delta_j$. Definim în continuare matricele $\hat{A}$ și $\bar{A}$ de dimensiuni $(2m-1) \times n$, respectiv $m \times (2n-1)$

$$\hat{A} = \text{col} \left[A_1 \mid d_1 \mid A_2 \mid d_2 \mid \dots \mid d_{m-1} \mid A_m \right]^T, \quad (62)$$

$$\bar{A} = \text{col} \left[A^1 \mid \delta^1 \mid \delta^2 \mid \delta^2 \mid \dots \mid \delta^{n-1} \mid A^n \right]. \quad (63)$$

Oricare ar fi $i \in \{1, 2, \dots, m-1\}$, fiecare direcție d_i este (prin construcție) o combinație liniară a liniilor $A_i, A_{i+1}, \dots, A_m$ (pentru mai multe detalii vezi, e.g., [45]). Prin urmare, există scalarii $\beta_i^i, \beta_{i+1}^i, \dots, \beta_m^i \in \mathbb{R}$ astfel încât $d_i = \sum_{k=i}^m \beta_k^i A_k$. Fixăm $b(d_i) = \sum_{k=i}^m \beta_k^i b_k$ și

$$\hat{b} = (b_1, b(d_1), b_2, b(d_2), b_3, \dots, b(d_{m-1}), b_m)^T. \quad (64)$$

Metoda DEK, deși este directă, are dezavantajul că determină un procent ridicat de umplere al matricelor $\hat{A}$ și $\bar{A}$, chiar și atunci când A este rară.

4.2 Accelerarea algoritmului Kaczmarz în cazul consistent

În această secțiune încercăm să determinăm o procedură de a adăuga direcții pentru proiecție problemei (1) astfel încât atât performanța metodei Kaczmarz ([49, 40]) să fie sporită, cât și procentul de umplere al matricei rezultate să fie menținute la un nivel scăzut. Soluția noastră este de a efectua un clustering aglomerativ al liniilor și al coloanelor din punct de vedere al tiparului de raritate. Distanțele folosite vor fi *Jaccard* sau *Hamming* (pentru mai multe detalii vezi, e.g., [46]). Propunem următorul algoritm.

Algoritmul Kaczmarz Modificat cu Clustering (MKC)

Clustering. Folosind una dintre distanțele *Jaccard* sau *Hamming*, se calculează un clustering al liniilor matricei A cu maximum nc clustere; notăm numărul de linii din cel de-al k -lea cluster cu m_k și fie $\{A_1^k, A_2^k, \dots, A_{m_k}^k\}$ mulțimea de linii din care este format. Fie matricele B_k și A_c^{nc} de dimensiune $m_k \times n$, respectiv $m \times n$, definite astfel

$$B_k = \begin{pmatrix} (A_1^k)^T \\ (A_2^k)^T \\ \vdots \\ (A_{m_k}^k)^T \end{pmatrix} \text{ și } A_c^{nc} = \begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_{nc} \end{pmatrix}. \quad (65)$$

Matricea A_c^{nc} se obține printr-o permutare a liniilor matricei originale. Se aplică aceeași permutare lui b și se obține b_c^{nc} . Avem că $S(A; b) = S(A_c^{nc}; b_c^{nc})$.

Transformări ale matricei. Oricare ar fi $k \in \{1, 2, \dots, nc\}$, se calculează pentru B_k setul întreg de direcții $d_1^k, d_2^k, \dots, d_{m_k-1}^k \in \mathbb{R}^n$ conform (60) și se obține matricea $\widehat{B}^k$ potrivit relației (62). Noua matrice este definită mai jos.

$$\widehat{A}_c^{nc} = \begin{pmatrix} \widehat{B}^1 \\ \widehat{B}^2 \\ \vdots \\ \widehat{B}^{nc} \end{pmatrix}. \quad (66)$$

Vectorul corespunzător $\widehat{b}_c^{nc}$ se calculează din b_c^{nc} conform (64).

Iterații Kaczmarz. Se aplică algoritmul Kaczmarz pentru a rezolva problema (echivalentă cu (1)): să se găsească $x \in \mathbb{R}^n$ astfel încât

$$\|\widehat{A}_c^{nc}x - \widehat{b}_c^{nc}\| = \min\{\|\widehat{A}_c^{nc}z - \widehat{b}_c^{nc}\|, z \in \mathbb{R}^n\}.$$

Observația 6 Când $nc = m$, algoritmul MKC reprezintă metoda Kaczmarz iterativ aplicată problemei (1). În cazul unui singur cluster se adaugă matricei A întregul set de direcții și algoritmul MKC devine Kaczmarz Direct. Acesta din urmă aproximează într-o singură iterație o soluție pentru (1) (pentru detalii și demonstrație vezi [45, Teorema 3]).

4.3 Un algoritm extins pentru cazul inconsistent

În cazul inconsistent vom utiliza raționamentul din secțiunea anterioară. Vom obține în mod independent două transformări diferite ale matricelor A și A^T , notate prin $\widehat{A}_0$ și $\bar{A}_0$, și vom introduce o versiune modificată a algoritmului Kaczmarz Extins (EK). Menționăm că $\widehat{A}_0^T \neq \bar{A}_0$, iar noul algoritm nu este echivalent cu metoda EK.

Fiind dați scalarii $\beta_1, \beta_2, \dots, \beta_n \in \mathbb{R}$, $\alpha_1, \alpha_2, \dots, \alpha_n \in \mathbb{R}$ și indicii $i_0 \in \{2, 3, \dots, m\}$, $j_0 \in \{2, 3, \dots, n\}$ fixați arbitrar, definim vectorii d , δ astfel

$$d = \sum_{i=1}^m \beta_i A_i, \quad \delta = \sum_{i=1}^n \alpha_i A^i \quad (67)$$

și matricele de dimensiuni $(m+1) \times n$, respectiv $m \times (n+1)$,

$$\widehat{A}_0^T = \text{col}[A_1, \dots, A_{i_0-1}, d, A_{i_0}, \dots, A_m], \quad (68)$$

$$\bar{A}_0 = \text{col}[(A^1), \dots, (A^{j_0-1}), \delta, (A^{j_0}), \dots, (A^n)]. \quad (69)$$

Oricărui vector $z = (z_1, z_2, \dots, z_m)^T \in \mathbb{R}^m$ îi asociem vectorul $\widehat{z} \in \mathbb{R}^{m+1}$ dat de relația

$$\widehat{z} = (z_1, z_2, \dots, z_{i_0-1}, \sum_{i=1}^m \beta_i z_i, z_{i_0}, \dots, z_m)^T. \quad (70)$$

Considerăm aplicațiile $P^d : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $\phi^\delta : \mathbb{R}^m \rightarrow \mathbb{R}^m$, $\widehat{Q}_0 : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $\bar{\Phi}_o : \mathbb{R}^m \rightarrow \mathbb{R}^m$ definite prin $P^d(x) = x - \frac{x^T d}{\|d\|^2} d$, $\phi^\delta(x) = x - \frac{x^T \delta}{\|\delta\|^2} \delta$,

$$\widehat{Q}_0 = (P_1 \circ \dots \circ P_{i_0} \circ P^d \circ P_{i_0+1} \circ \dots \circ P_m), \quad (71)$$

$$\bar{\Phi}_o = (\phi_1 \circ \dots \circ \phi_{j_0} \circ \phi^\delta \circ \phi_{j_0+1} \circ \dots \circ \phi_n), \quad (72)$$

și matricea de dimensiune $n \times (m+1)$, $\widehat{R}_0 = \text{col}[\widehat{R}_0^1, \widehat{R}_0^2, \dots, \widehat{R}_0^{m+1}]$ dată de (vezi [45])

$$\begin{aligned} \widehat{R}_0^k &= \frac{1}{\|A_k\|^2} P_1 P_2 \dots P_{k-1}(A_k), \quad \forall k \in \{1, 2, \dots, i_0 - 1\}, \\ \widehat{R}_0^{i_0} &= \frac{1}{\|d\|^2} P_1 P_2 \dots P_{i_0-1}(d), \\ \widehat{R}_0^k &= \frac{1}{\|A_{k-1}\|^2} P_1 P_2 \dots P_{i_0-1} P^d \dots P_{k-2}(A_{k-1}), \quad \forall k \in \{i_0 + 1, \dots, m\}. \end{aligned} \quad (73)$$

Algoritmul Kaczmarz Extins Modificat(EKM)

Inițializări. $x^0 \in \mathbb{R}^n$ este ales arbitrar și $y^0 = b$.

Pas iterativ. Oricare ar fi $k \geq 0$, fie

$$y^{k+1} = \bar{\Phi}_0(y^k); \quad x^{k+1} = \widehat{Q}_0 x^k + \widehat{R}_0(b - y^{k+1}). \quad (74)$$

Teorema 7 Șirurile $(x^k)_{k \geq 0}$ și $(y^k)_{k \geq 0}$, generate folosind algoritmul EKM, converg și

$$\lim_{k \rightarrow \infty} y^k = P_{\mathcal{N}(A^T)}(b), \quad \lim_{k \rightarrow \infty} x^k = P_{\mathcal{N}(A)}(x^0) + x_{LS}. \quad (75)$$

Teorema 8 *Oricare ar fi matricea A de dimensiune $m \times n$ și vectorul $b \in \mathbb{R}^m$, se vor defini matricea $\widehat{A}_0$ de dimensiune $(m+1) \times n$ conform (68), vectorul $\widehat{b} \in \mathbb{R}^{(m+1)}$ din (70) și aplicațiile $\widehat{Q}_0 : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $\widehat{R}_0 : \mathbb{R}^{(m+1)} \rightarrow \mathbb{R}^n$, $\bar{\Phi}_0 : \mathbb{R}^m \rightarrow \mathbb{R}^m$ date de (71), (73), și respectiv (72). Dacă*

$$\begin{aligned}\widehat{Q}_0(A_i) &= 0, \quad \forall i \in \{1, 2, \dots, m\}, \quad \widehat{Q}_0(d) = 0, \\ \bar{\Phi}_0(A^j) &= 0, \quad \forall j \in \{1, 2, \dots, n\}, \quad \bar{\Phi}_0(\delta) = 0,\end{aligned}\tag{76}$$

atunci algoritmul Kaczmarz Extins Modificat converge într-o singură iterație la o soluție a problemei (1).

Pentru cazul inconsistent propunem următorul algoritm

Algoritmul Kaczmarz Extins Modificat cu Clustering (MEKC)

Clustering. Se execută clustering al coloanelor matricei A cu maximum ncc clustere folosind una dintre distanțele *Jaccard* sau *Hamming*. Pentru orice cluster k cu $k \in \{1, 2, \dots, ncc\}$, fie $A_k^1, A_k^2, \dots, A_k^{n_k}$ mulțimea de coloane din care este format, unde n_k reprezintă dimensiunea clusterului. Matricele B^k și A_{ncc}^c de dimensiuni $n_k \times m$, respectiv $n \times m$, sunt definite astfel

$$B^k = \text{col} [A_k^1, A_k^2, \dots, A_k^{n_k}], \tag{77}$$

$$A_{ncc}^c = \text{col} [B^1, B^2, \dots, B^{ncc}]. \tag{78}$$

Matricea A_{ncc}^c se obține în urma unei permutări de coloane ale lui A . Se realizează un clustering al liniilor matricei A cu maximum ncr clustere pentru a obține matricea A_c^{ncr} conform (65) și se aplică aceeași permutare termenului liber b , rezultând b_c^{ncr} .

Transformări ale matricelor. Oricare ar fi $k \in \{1, 2, \dots, ncc\}$, se construiesc direcțiile $\delta_k^1, \delta_k^2, \dots, \delta_k^{n_k} \in \mathbb{R}^n$ (61) pentru blocul de coloane B^k și se calculează $\bar{B}^k$ conform (63). Noua matrice este astfel definită

$$\bar{A}_{ncc}^c = \text{col} [\bar{B}^1, \bar{B}^2, \dots, \bar{B}^{ncc}]. \tag{79}$$

Se calculează pentru ncr clustere matricea $\widehat{A}_c^{ncr}$ dată de (66) și vectorul core-spunzător $\widehat{b}_c^{ncr}$.

Iterații Kaczmarz Extins Modificat. Se aproximează o soluție a problemei (1) folosind algoritmul EKM cu noile matrice $\bar{A}_{ncc}^c$ și $\widehat{A}_c^{ncr}$.

5 Experimente numerice

5.1 Tehnici de constrângere

În această secțiune încercăm să dezvoltăm și să analizăm un algoritm pentru problema (1) bazat pe informații a priori despre soluția exactă. Scopul experimentelor noastre este de a arăta că folosind o familie de funcții de constrângere construită adaptiv se obțin rezultate mai bune decât aplicând o constrângere constantă. Vom studia o versiune simplificată a unei probleme reale. Aceasta se referă la reconstrucția unui vector binar. Ne concentrăm asupra determinării numărului și locației aproximative a componentelor nenule.

5.1.1 O versiune adaptivă a algoritmului general CGENk

Propunem următorul algoritm cu o familie de funcții de constrângere, unde prin operatorul $K : \mathbb{R}^n \rightarrow \mathbb{R}^n$ am notat o iterație a metodei Kaczmarz.

Algoritmul Kaczmarz Constrâns cu o Familie (FCK)

Inițializări. Fie $x^0 = 0$, n_0 un număr întreg pozitiv fixat, ϵ_0 și δ două numere reale pozitive și C_k proiecția metrică pe “cutia” $[a^k, b^k] = [a_1^k, b_1^k] \times \dots \times [a_n^k, b_n^k] \subset \mathbb{R}^n$ (vezi (20)).

Pasul I. Oricare ar fi $k \in \{0, 1, \dots, n_0 - 1\}$, fie $[a^k, b^k] = [0, 1]^n$ și

$$x^{k+1} = C_{k+1}(Kx^k). \quad (80)$$

Pasul II. Fie $x^0 = x^{n_0}$ și pentru orice $k > n_0$ se calculează $\tilde{x}^k$ ca

$$\tilde{x}^k = Kx^{k-1}. \quad (81)$$

Oricare ar fi $i \in \{1, 2, \dots, n\}$, dacă $\tilde{x}_i^k \leq \epsilon_k$, atunci $[a_i^k, b_i^k] = [0, 10^{-4}]$; altfel $[a_i^k, b_i^k] = [a_i^{k-1}, b_i^{k-1}]$.

Se calculează x^k conform

$$x^k = C_k(\tilde{x}^k), \quad (82)$$

și se actualizează ϵ_{k+1} cu $\epsilon_{k+1} = \epsilon_k + \delta$.

Valorile parametrilor n_0 , ϵ_0 și δ sunt alese conform problemei de reconstrucție, în urma unor teste sistematice. Construim familia de funcții de constrângere astfel încât să satisfacă (23). Algoritmul este format din doi pași

și pornește cu vectorul nul ca aproximație inițială. Pe parcursul primei etape, pentru un număr fixat de iterații, folosim aceeași funcție de constrângere, proiecția metrică pe “cutia” $[a, b] = [0, 1]^n$. Raționamentul este de a păstra aproximația în intervalul preconizat și în același timp de a permite valorilor nenule să “crească”.

În cea de-a doua etapă vom presupune că, dacă după un număr suficient de iterații valoarea unei componente este în continuare apropiată de zero, atunci în soluția exactă valoarea acesteia este nulă. La fiecare iterație aplicăm Kaczmarz și “măsurăm” valoarea fiecărui pixel. Dacă este sub un prag dat, ϵ_k , atunci intervalul corespunzător din “cutie” va fi micșorat la $[0, 10^{-4}]$. Am ales intervalul $[0, 10^{-4}]$ în favoarea lui $[0, 0]$ deoarece am observat experimental faptul că se obțin rezultate mai bune atunci când interiorul intervalelor de constrângere este nenul. Pentru a accelera convergența algoritmului, la fiecare iterație, se crește pragul cu δ .

În următoarele două secțiuni studiem o problemă particulară de reconstrucție de imagini (vezi [35] și [34]). Această problemă apare în 3D Tomographic Particle Image Velocimetry (TomoPIV), o metodă optică de a măsura viteză fluidelor.

5.1.2 Rezolvarea unei probleme de reconstrucție de imagini

5.1.3 Rezultate numerice

5.2 Adăugarea unor direcții pentru proiecție

În următoarea secțiune examinăm experimente numerice referitoare la algoritmi MKC și MEKC pentru două matrice rare.

5.2.1 Studii de caz

5.2.2 Conservarea procentului de umplere sub un prag dat

Atunci când ne ocupăm de o problemă (1) inconsistentă cu o matrice A rară, rău condiționată și de dimensiuni mari, metodele iterative au o performanță de timp scăzută, în timp ce algoritmi direcții conduc la pierderea proprietății de raritate a datelor. Atunci când aplicăm algoritmul MEKC, pe măsură ce numărul de clustere pe linii sau coloane crește, procentul de umplere al matricelor noi obținute dar și rata de convergență a procesului iterativ se diminuează. Prin urmare, problema este de a găsi valori mici ale lui ncr și ncc astfel încât numărul de elemente nenule ale matricelor $\widehat{A}_c^{ncr}$ și $\bar{A}_{ncc}^c$ să rămână sub un prag (threshold) dat.

Vom considera cazul consistent al problemei (1). Pentru o distanță și un *threshold* fixate, ne interesează să găsim o valoare ncr^* minimă astfel încât $\text{nz}(\widehat{A^{ncr^*}}) < \text{threshold}$

Soluție. Soluția noastră este de a executa o căutare binară a lui ncr^* în șirul ordonat $\text{nz}(\widehat{A}) \geq \text{nz}(\widehat{A^2}) \geq \dots \geq \text{nz}(\widehat{A^{m-1}}) \geq \text{nz}(A)$, unde prin $\text{nz}(M)$ am notat numărul de elemente nenule ale unei matrice M :

- inițial $ncr = \frac{m+1}{2}$ și se calculează $\text{nz}(\widehat{A^{ncr}})$;
- dacă $\text{nz}(\widehat{A^{ncr}}) > \text{threshold}$, se continuă căutarea binară în subșirul drept $\text{nz}(\widehat{A^{ncr+1}}) \geq \dots \geq \text{nz}(\widehat{A^{m-1}}) \geq \text{nz}(A)$;
- altfel, ncr este o valoare posibilă a lui ncr^* și se continuă căutarea unei valori mai mici în $\text{nz}(\widehat{A}) \geq \text{nz}(\widehat{A^2}) \geq \dots \geq \text{nz}(\widehat{A^{ncr-1}})$;
- stop atunci când se obține un subșir vid; ncr^* este ultima valoare găsită pentru care are loc $\text{nz}(\widehat{A^{ncr}}) < \text{threshold}$.

Observația 7 *La fiecare pas al căutării binare calculăm șirul de direcții doar atâta timp cât nu se depășește pragul dat.*

După calcularea valorii ncr^* comparăm algoritmul Kaczmarz clasic cu $\text{MKC}(\text{dist}, ncr^*)$.

Soluție. Determinăm numărul de flops (operații aritmetice în virgulă mobilă) efectuate de fiecare din acești algoritmi.

Metodă	Număr de flops
Kaczmarz Direct	$3 * \text{nz}(\widehat{A}) + 2 * (2 * m - 1)$
$\text{MKC}(\text{dist}, ncr^*)$	$\left(3 * \text{nz}(\widehat{A^{ncr^*}}) + 2 * (2 * m - ncr^*)\right) * \text{niter}_{ncr^*}$
Kaczmarz	$(3 * \text{nz}(A) + 2 * m) * \text{niter}_K$

Rezultatele ilustrate mai sus arată că atunci când nu se poate aplica Kaczmarz Direct, algoritmul MKC furnizează o alternativă mai eficientă decât metoda Kaczmarz. Aceleași raționamente pot fi aplicate și în cazul inconsistent.

Bibliografie

- [1] A. Aleyner and Y. Censor. Best approximation to common fixed points of a semigroup of nonexpansive operators. *J. Nonlinear Convex Anal.*, 6:137–151, 2005.
- [2] H. Bauschke. The approximation of fixed points of compositions of nonexpansive mappings in hilbert space. *J. Math. Anal. Appl.*, 202:150–159, 1996.
- [3] H. Bauschke and J. Borwein. On projection algorithms for solving convex feasibility problems. *SIAM Rev.*, 38:367–426, 1996.
- [4] H. Bauschke and P.L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, New York, 2011.
- [5] A. Ben-Israel and T. Greville. *Generalized Inverses: Theory and Applications, Second Edition*. Springer-Verlag, New York, 2003.
- [6] A. Björk. *Numerical Methods for Least Squares Problems*. SIAM, Philadelphia, 1996.
- [7] C. Byrne. A unified treatment of some iterative algorithms in signal processing and image reconstruction. *Inverse Probl.*, 20:103–120, 2004.
- [8] A. Cegielski. *Iterative Methods for Fixed Point Problems in Hilbert Spaces, Lecture Notes in mathematics 2057*. Springer-Verlag, Berlin, Heidelberg, 2012.
- [9] Y. Censor and A.Z. Stavros. *Parallel optimization: theory, algorithms and applications*, “Numer. Math. and Sci. Comp.” Series. Oxford Univ. Press, New York, 1997.
- [10] Y. Censor, **I. Pantelimon**, and C. Popa. Family constraining of iterative algorithms. *Numerical Algorithms*, pages 1–16, 2013. doi: 10.1007/s11075-013-9736-5.
- [11] C. Chaux, J.C. Pesquet, and N. Pustelnik. Nested iterative algorithms for convex constrained image recovery problems. *SIAM J. Imaging Sci.*, 2:730–762, 2009.

- [12] P.L. Combettes. *Fejér-monotonicity in convex optimization*, volume 2, pages 106–114. Kluwer, Boston, 2001.
- [13] A.R. De Pierro and A. Iusem. On the asymptotic behavior of some alternate smoothing series expansion iterative methods. *Linear Algebra Appl.*, 130:3–24, 1990.
- [14] B. Eicke. Iteration methods for convexly constrained ill-posed problems in hilbert space. *Numer. Funct. Anal. Optim.*, 13:413–429, 2007.
- [15] T. Elfing. Block-iterative methods for consistent and inconsistent linear equations. *Numer. Math.*, 35:1–12, 1980.
- [16] T. Elfing. A stationary iterative pseudoinverse algorithm. *BIT*, 38(2): 275–282, 1998.
- [17] T. Elfing, T. Nikazad, and P.C. Hansen. Semi-convergence and relaxation parameters for a class of sirt algorithms. *Electron. Trans. Numer. Anal.*, 37:321–336, 2010.
- [18] L. Elsner, I. Koltracht, and M. Neumann. Convergence of sequential and asynchronous nonlinear paracontractions. *Numer. Math.*, 62:305–319, 1992.
- [19] G. Gan, C. Ma, and J. Wu. *Data Clustering: Theory, Algorithms and Applications*. Society for Industrial and Applied Mathematics, Philadelphia, 2007.
- [20] G.T. Herman. *Image Reconstruction from Projections. The Fundamentals of Computerized Tomography*. Academic Press, New York, 1980.
- [21] G.T. Herman. *The Fundamentals of Computerized Tomography: Image Reconstruction from Projections, 2nd edition*. Springer-Verlag, London, UK, 2009.
- [22] S.A. Hirstoaga. Iterative selection methods for common fixed point problems. *J. Math. Anal. Appl.*, 324:1020–1035, 2006.
- [23] R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, NY, USA, 1990.

- [24] A. Katsaggelos, J. Biemond, R. Mersereau, and R. Schafer. A general formulation of constrained iterative restoration algorithms. *IEEE Int. Conf. on Acoustics, Speech, and Signal Process.*, 10:700–703, 1985.
- [25] A.K. Katsaggelos, J. Biemond, R. Mersereau, and R. Schafer. A regularized iterative image restoration algorithm. *IEEE Trans. Signal Process.*, 39:914–929, 1991.
- [26] S. Kawata and O. Nalcioglu. Constrained iterative reconstruction by the conjugate gradient method. *IEEE Trans. on Med. Imaging*, 4:65–71, 1985.
- [27] K.C. Kiwiel and B. L. opuch. Surrogate projection methods for finding fixed points of firmly nonexpansive mappings. *SIAM J. Optim.*, 7:1084–1102, 1997.
- [28] I. Koltracht and P. Lancaster. Constraining strategies for linear iterative processes. *IMA J. Numer. Anal.*, 10:555–567, 1990.
- [29] I. Koltracht, P. Lancaster, and D. Smith. The structure of some matrices arising in tomography. *Linear Algebra Appl.*, 130:193–218, 1990.
- [30] L. Landweber. An iterative formula for fredholm integral equations of the first kind. *Amer. J. Math.*, 73:615–624, 1951.
- [31] A. Nicola, S. Petra, C. Popa, and C. Schnörr. On a general extending and constraining procedure for linear iterative methods. *Int. J. Comput. Math.*, 89:231–253, 2012.
- [32] J.M. Ortega and W.C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several variables*. Academic Press, New York, NY, USA, 1970.
- [33] **I. Pantelimon** and C. Popa. Constraining by a family of strictly nonexpansive idempotent functions with applications in image reconstruction. *BIT Numerical Mathematics*, 53(2):527–544, 2013. doi: 10.1007/s10543-012-0414-0.
- [34] S. Petra and C. Schnörr. TomoPIV meets compressed sensing. *Pure Math. Appl.*, 20:49–76, 2009.

- [35] S. Petra, C. Popa, and C. Schnörr. Accelerating constrained sirt with applications in tomographic particle image reconstruction, 2009. URL <http://www.ub.uni-heidelberg.de/archiv/9477>. Preprint.
- [36] **I. Pomparău**. On the acceleration of Kaczmarz projection algorithm. Presented at the 84th Annual Meeting of the International Association of Applied Mathematics and Mechanics, 18–22 March, 2013, Novi Sad, Serbia. Submitted for publication in *Proceedings in Applied Mathematics and Mechanics*, 2013.
- [37] **I. Pomparău**. On an accelerated version of the Kaczmarz algorithm with a priori clustering. In *Topics in mathematical modelling of life science problems*, pages 61–76. Matrix Rom, București, 2013. Presented at the Nine-th Workshop on Mathematical Modeling of Environmental and Life Sciences Problems, 1–4 November, 2012, Constanța, România.
- [38] **I. Pomparău** and C. Popa. Supplementary projections for the acceleration of Kaczmarz algorithm. Submitted for publication, 2013.
- [39] **I. Pomparău** and C. Popa. Weaker assumptions for convergence of extended block Kaczmarz and Jacobi projection algorithms. Submitted for publication, 2013.
- [40] C. Popa. Least-squares solution of overdetermined inconsistent linear systems using Kaczmarz relaxation. *Intern. J. Comp. Math.*, 55:79–89, 1995.
- [41] C. Popa. Extensions of block-projections methods with relaxation parameters to inconsistent and rank-deficient least-squares problems. *BIT*, 38(1):151–176, 1998.
- [42] C. Popa. Extended and constrained diagonal weighting algorithm with application to inverse problems in image reconstruction. *Inverse Probl.*, 26, 2010. doi: 10.1088/0266-5611/26/6/065004.
- [43] C. Popa. *Projection algorithms - classical results and developments. Applications to image reconstruction*. Lambert Academic Publishing - AV Akademikerverlag GmbH & Co. KG, Saarbrücken, Germany, 2012.

- [44] C. Popa and R. Zdunek. Kaczmarz extended algorithm for tomographic image reconstruction from limited-data. *Math. Comput. Simulation*, 65: 579–598, 2004.
- [45] C. Popa, T. Preclik, H. Köstler, and U. Rüde. On Kaczmarz’s projection iteration as a direct solver for linear least squares problems. *Linear Algebra and its Applications*, 436:389–404, 2012.
- [46] A. Rajaraman and J.D. Ullman. *Mining of Massive Datasets*. Cambridge University Press, New York, 2012.
- [47] R.W. Schafer, R.M. Mersereau, and M.A. Richards. Constrained iterative restoration algorithms. *Proc. of the IEEE*, 69:432–450, 1981.
- [48] G.F. Simmons. *Introduction to Topology and Modern Analysis*. McGraw-Hill Book Company, New York, NY, USA, 1963.
- [49] K. Tanabe. Projection method for solving a singular system of linear equations and its applications. *Numer. Math.*, 17:203–214, 1971.
- [50] D.C. Youla and H. Webb. Image restoration by the method of convex projections: Part 1—theory. *IEEE Trans. on Med. Imaging*, 1:81–94, 1982.