

FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
UNIVERSITATEA BABEȘ-BOLYAI



Erika Nagy

**Metode numerice de aproximare
a zerourilor unor operatori
și de rezolvare a inegalităților variaționale
cu aplicații**

Rezumatul tezei de doctorat

Conducător de doctorat: **Prof. Dr. Gábor Kassay**

CLUJ-NAPOCA

2013

Cuprins

1	Introducere	3
2	Noțiuni din teoria operatorilor monotoni	6
2.1	Operatori	6
2.2	Funcții	7
3	Technica primal-duală de divizibilitate	9
3.1	Formularea problemei cu incluziuni monotone	9
3.2	Algoritmul primal-dual de divizare	11
3.3	Aplicații la probleme de minimizare convexe	13
4	Experimente numerice	23
4.1	Procesarea imaginilor	23
4.1.1	Claritatea imaginilor	23
4.1.2	Cadru experimental	26
4.2	Probleme de localizare	30
4.2.1	Prezentarea problemei	32
4.2.2	Experiență computațională	33
4.3	Consens în media rețelelor colorate	36
4.3.1	Algoritmul propus	36
4.3.2	Evaluarea performanței	38
4.4	Clasificare cu ajutorul mașinilor de vector-suport	41
4.4.1	Hiperplanul de separare	41
4.4.2	Experimente cu recunoașterea cifrelor	43
5	Aplicația interdisciplinară a algoritmului	46

5.1	Optimizare în biologie	46
5.2	Technici de recunoașterea modelelor	48
	Bibliografie	50

Capitolul 1

Introducere

Existența metodelor de optimizare începe cu matematicienii Newton, Lagrange și Cauchy. Dezvoltarea metodelor de calcul diferențial pentru optimizare a fost posibilă datorită contribuțiilor lui Newton și Leibniz. Bazele calculului de variații, cu privire la minimizarea funcțiilor, au fost puse de Bernoulli, Euler, Lagrange și Weierstrass.

Metoda de optimizare pentru probleme limitate, care implică adăugarea de multiplicatori necunoscuți, a devenit cunoscută sub numele autorului ei, Lagrange. Cauchy a făcut primul pas în rezolvarea problemelor de optimizare fără restricții. În mijlocul secolului al XX-lea, calculatoarele digitale de mare viteză au făcut posibilă punerea în aplicare a procedurilor de optimizare complexe și au stimulat cercetări suplimentare privind metodele noi. Au urmat avansuri spectaculoase, producând o literatură masivă pentru tehnici de optimizare. De asemenea, acest progres a dus la apariția unor noi domenii bine definite în teoria optimizării [67].

În ultimii ani, au apărut mai mulți algoritmi de divizare [43] pentru rezolvarea problemelor de incluziuni monotone, implicând sume paralele și compoziții cu operatori liniari continui, care în cele din urmă sunt reduse pentru a găsi zerourile sumei unui operator maximal monoton și unul cocoerciv sau monoton și Lipschitz. Aceste probleme au fost rezolvate prin angajarea unor algoritmi forward-backward sau forward-backward-forward într-un produs de spații adecvate și a dat naștere la metodele primal-duale de divizare ([9, 16, 20, 23, 61] și referințele aferente).

Recent, se poate remarca un interes al cercetătorilor în rezolvarea sistemelor

de probleme de incluziuni monotone [1, 5, 6, 22]. Acest lucru este motivat de faptul că problemele de optimizare convexe care rezultă, de exemplu, în domenii cum ar fi procesarea imaginilor [13], probleme de localizare [18, 31], consens în media rețelelor colorate [46, 47] și clasificare cu mașini cu vectori de suport [27, 28] urmează să fie rezolvate în raport cu mai multe variabile, adesea legate în diferite moduri, cum ar fi prin ecuații liniare.

Prezenta cercetare este motivată de investigațiile efectuate în [1]. Autorii propun un algoritm pentru rezolvarea problemelor de incluziuni monotone cuplate, în care variabilele sunt legate de operatori care satisfac împreună o proprietate de cocoercivitate. Scopul nostru este de a depăși necesitatea de a avea diferențiabilitate pentru unele funcții care apar în problemele de optimizare convexe din [1]. În acest scop, avem în vedere un sistem de incluziuni monotone mai general, pentru care operatorul de cuplare satisface o proprietate de continuitate Lipschitz, împreună cu sistemul dual de incluziuni monotone în sensul de dualitate Attouch-Théra (cf. [2]). Rezolvarea simultană a sistemului primal și dual de incluziuni monotone se reduce la problema de a găsi zerourile sumei unui operator maximal monoton și un operator monoton și Lipschitz continuu într-un produs de spații adecvat. Problema din urmă este rezolvată de un algoritm forward-backward-forward, fapt care ne permite să stabilim pentru schema iterativă atât rezultate slabe cât și puternice de convergență.

Teza este organizată în cinci capitole și o bibliografie. În primul capitol afirmăm problema inițială care a stimulat această cercetare. H. Attouch, L.M. Briceno-Arias și P.L. Combettes propun în [1] o metodă de divizare paralelă pentru rezolvarea sistemelor de incluziuni monotone cuplate în spații Hilbert, stabilirea convergenței se întâmplă cu presupunerea că există soluții. Metoda poate gestiona un număr arbitrar de variabile și scheme de cuplare neliniare.

În al doilea capitol sunt prezentate câteva notații necesare și rezultate preliminare, în scopul de a facilita citirea manuscrisului. Sunt introduse noțiuni cum ar fi operatori nonexpansivi, sumă paralelă, cocoercivitate, conjugare, dualitate Fenchel, subdiferențiabilitate și operatori maximal monotoni împreună cu niște algoritmi de divizare esențiale, cum ar fi algoritmul Douglas-Rachford, algoritmul forward-backward și algoritmul lui Tseng. Acești algoritmi au fost studiate în detaliu în [4].

Capitolul trei este dedicat algoritmului primal-dual de divizare pentru rezolvarea problemei luate în considerare în această teză și investighează convergența algoritmului. Operatorii care apar în fiecare dintre incluziuni din sistem sunt procesate în fiecare iterație separat, și anume, operatorii univoci sunt evaluate în mod explicit, echivalent cu un pas forward, în timp ce operatorii multivoci sunt evaluate prin intermediul rezolvenților lor, ceea ce este echivalent cu un pas backward. Un avantaj este că pașii din schema iterativă pot fi executate simultan, acest lucru făcând metoda aplicabilă la o varietate de probleme de minimizare convexe. Rezolvarea problemelor de optimizare convexe cu mai multe variabile este, de asemenea, prezentată în acest capitol.

În capitolul patru, performanțele numerice și eficacitatea algoritmului de divizare propuse sunt evidențiate prin mai multe experimente numerice, cum ar fi procesarea imaginilor, adică curățarea unei imagini neclare; probleme de localizare, unde obiectivul este de a minimiza calea de energie dintre facilitățile existente și mai multe facilități noi, care trebuie situate între cele existente; consens în media rețelelor colorate, care constă în calcularea mediei pornind de la valoarea fiecărui nod într-un mod recursiv și distribuit; clasificarea imaginilor prin intermediul vectorilor suport, unde obiectivul este de a construi o funcție de decizie, cu ajutorul datelor de formare, care atribuie corect o dată nouă, cu o rată de clasificare eronată scăzută.

Ultima parte a tezei de doctorat prezintă o colaborare cu o echipă de cercetare în biologie, care studiază prezența picoalgelor în lacurile sărate din Transilvania. Scopul este de a adapta algoritmul în procesarea imaginilor și recunoașterea formelor în imagini microscopice și electroforetogramme. Procesul de dezvoltare a inclus imagini microscopice epifluorescente de picoalge, scopul fiind acela de a număra și distinge picoalge de zgomotul de bază din imagini.

Cuvinte cheie

optimizare convexă, sistem de incluziuni, algoritm forward-backward-forward, continuitate Lipschitz, operator maximal monoton, clasificarea imaginilor

Capitolul 2

Noțiuni din teoria operatorilor monotoni

Noțiunea matematică a unui spațiu Hilbert a fost numită după matematicianul german David Hilbert. La început, spațiile Hilbert au fost studiate ca spații de funcții infinit-dimensionale de David Hilbert, Frigyes Riesz și Erhard Schmidt, în primul deceniu al secolului 20. De-a lungul acestei lucrări \mathbb{R}_+ denotă setul de numere reale pozitive, \mathbb{R}_{++} setul de numere reale strict pozitive și $\overline{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$ linia reală extinsă. Considerăm spații Hilbert dotate cu produsul scalar $\langle \cdot | \cdot \rangle$ și norma asociată $\|\cdot\| = \sqrt{\langle \cdot | \cdot \rangle}$. Pentru a evita confuzia, atunci când este necesar, se vor folosi indicii adecvate la produsul scalar și la normă. Simbolurile \rightharpoonup și \rightarrow denotă convergența slabă și puternică, respectiv.

2.1 Operatori

Fie \mathcal{H} un spațiu Hilbert real și fie $2^{\mathcal{H}}$ setul de putere al lui \mathcal{H} , i.e., familia tuturor subseturilor lui \mathcal{H} . Notăția $M : \mathcal{H} \rightarrow 2^{\mathcal{H}}$ înseamnă că M este un operator *multivoc*. Notăm cu zer $M = \{x \in \mathcal{H} : 0 \in Mx\}$ setul zerourilor lui M și cu gra $M = \{(x, u) \in \mathcal{H} \times \mathcal{H} : u \in Mx\}$ graful lui M . Domeniul lui M este

$$\text{dom } M = \{x \in \mathcal{H} : Mx \neq \emptyset\}.$$

Inversa lui M , denotată cu $M^{-1} : \mathcal{H} \rightarrow 2^{\mathcal{H}}$, este definită după cum urmează

$$\text{gra } M^{-1} = \{(u, x) \in \mathcal{H} \times \mathcal{H} : (x, u) \in \text{gra } M\}.$$

Prin urmare, pentru fiecare $(x, u) \in \mathcal{H} \times \mathcal{H}$, $u \in Mx \Leftrightarrow x \in M^{-1}u$.

Suma și suma paralelă a doi operatori multivoci $M, N : \mathcal{H} \rightarrow 2^{\mathcal{H}}$ este definită ca

$$M + N : \mathcal{H} \rightarrow 2^{\mathcal{H}}, (M + N)(x) = M(x) + N(x), \quad \forall x \in \mathcal{H} \quad (2.1.1)$$

și

$$M \square N : \mathcal{H} \rightarrow 2^{\mathcal{H}}, M \square N = (M^{-1} + N^{-1})^{-1}. \quad (2.1.2)$$

Fie \mathcal{G} un spațiu Hilbert real și considerăm operatorul *univoc* $L : \mathcal{H} \rightarrow \mathcal{G}$. Norma operatorului linear continuu L este $\|L\| = \sup\{\|Lx\| : x \in \mathcal{H}, \|x\| \leq 1\}$, și $L^* : \mathcal{G} \rightarrow \mathcal{H}$, definită astfel încât $\langle Lx|y \rangle = \langle x|L^*y \rangle$ pentru fiecare $(x, y) \in \mathcal{H} \times \mathcal{G}$, denotă operatorul adjunct al lui L .

2.2 Funcții

Notăm cu $\Gamma(\mathcal{H})$ setul de funcții convexe, corespunzătoare și inferior semicontinuu $f : \mathcal{H} \rightarrow \overline{\mathbb{R}}$.

Funcția indicator $\delta_C : \mathcal{H} \rightarrow \overline{\mathbb{R}}$ a unui set $C \subseteq \mathcal{H}$ este definită după cum urmează

$$\delta_C(x) = \begin{cases} 0, & \text{for } x \in C; \\ +\infty, & \text{altfel.} \end{cases} \quad (2.2.1)$$

Observăm că funcția indicator δ_C este inferior semicontinuu dacă și numai dacă C este închis.

Definiție 2.2.1. Fie f și g două funcții corespunzătoare din \mathcal{H} la $\overline{\mathbb{R}}$. Convoluția infi-

mală a funcțiilor f și g este definită ca

$$f \square g : \mathcal{H} \rightarrow \overline{\mathbb{R}}, f \square g(x) = \inf_{y \in \mathcal{H}} \{f(y) + g(x - y)\}.$$

Definiție 2.2.2. Fie $f : \mathcal{H} \rightarrow \overline{\mathbb{R}}$. Funcția conjugată (sau conjugata Fenchel) a funcției f este $f^* : \mathcal{H} \rightarrow \overline{\mathbb{R}}$, $f^*(u) = \sup \{\langle u, x \rangle - f(x) : x \in \mathcal{H}\}$ pentru $u \in \mathcal{H}$ și biconjugata lui f este $f^{**} = (f^*)^*$.

Observăm că, dacă $f \in \Gamma(\mathcal{H})$, atunci și $f^* \in \Gamma(\mathcal{H})$.

Teorema următoare legată de dualitatea lui Fenchel a fost obținută de R. T. Rockafellar [53].

Theorem 2.2.1. Fie \mathcal{H} un spațiu Hilbert. Fie $f, g : \mathcal{H} \rightarrow \mathbb{R} \cup +\infty$ funcții convexe corespunzătoare.

(i) Dacă $\text{dom } f \cap \text{dom } g$ conține un punct în care f sau g este continuu, atunci

$$(f + g)^* = f^* \square g^*$$

cu o convoluție infimală exactă.

(ii) Presupunem că f sau g aparțin lui $\Gamma(\mathcal{H})$. Dacă $\text{dom } f^* \cap \text{dom } g^*$ conține un punct în care f^* sau g^* este continuu, atunci

$$f \square g = (f^* + g^*)^*$$

cu o convoluție infimală exactă.

Capitolul 3

Technica primal-duală de divizibilitate

Noțiunea matematică a unui spațiu Hilbert a fost numită după matematicianul german David Hilbert. La început, spațiile Hilbert au fost studiate ca spații de funcții infinit-dimensionale de David Hilbert, Frigyes Riesz și Erhard Schmidt, în primul deceniu al secolului 20. De-a lungul acestei lucrări \mathbb{R}_+ denotă setul de numere reale pozitive, \mathbb{R}_{++} setul de numere reale strict pozitive și $\overline{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$ linia reală extinsă. Considerăm spații Hilbert dotate cu produsul scalar $\langle \cdot | \cdot \rangle$ și norma asociată $\|\cdot\| = \sqrt{\langle \cdot | \cdot \rangle}$. Pentru a evita confuzia, atunci când este necesar, se vor folosi indicii adecvate la produsul scalar și la normă. Simbolurile \rightharpoonup și \rightarrow denotă convergența slabă și puternică, respectiv.

Următoarele rezultate prezentate în teză se bazează pe articolul [10], care a rezultat din colaborarea mea cu profesorul Radu Ioan Boț și Ernő Robert Csetnek.

3.1 Formularea problemei cu incluziuni monotone

Problema în cauză este un sistem mai general de incluziuni monotone, decât cel prezentat în [1], pentru care operatorul satisface o proprietate de Lipschitz continuitate, împreună cu sistemul său dublu de incluziuni monotone în sensul dualității Attouch-Théra [2].

Problema 3.1.1. Fie $m \geq 1$ un număr pozitiv, $(\mathcal{H}_i)_{1 \leq i \leq m}$ spații Hilbert reale și, pentru $i = 1, \dots, m$ fie $B_i : \mathcal{H}_1 \times \dots \times \mathcal{H}_m \rightarrow \mathcal{H}_i$ operatori μ_i -Lipschitz continuu cu $\mu_i \in \mathbb{R}_{++}$, care satisfac împreună o proprietate de monotonitate

$$\begin{aligned} & \forall (x_1, \dots, x_m) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m, \forall (y_1, \dots, y_m) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m \\ & \sum_{i=1}^m \langle x_i - y_i | B_i(x_1, \dots, x_m) - B_i(y_1, \dots, y_m) \rangle_{\mathcal{H}_i} \geq 0. \end{aligned} \quad (3.1.1)$$

Fie \mathcal{G}_i spații Hilbert reale, pentru fiecare $i = 1, \dots, m$ și fie $A_i : \mathcal{G}_i \rightarrow 2^{\mathcal{G}_i}$ operatori maximal monotoni, $C_i : \mathcal{G}_i \rightarrow 2^{\mathcal{G}_i}$ operatori monotoni, astfel încât C_i^{-1} să fie ν_i -Lipschitz continuu cu $\nu_i \in \mathbb{R}_+$ și $L_i : \mathcal{H}_i \rightarrow \mathcal{G}_i$ operatori liniari continui. Problema propusă este aceea de a rezolva sistemul de incluziuni monotone (a se vedea 2.1.2 pentru definiția sumei paralele pentru doi operatori)

$$\text{căutăm } \bar{x}_1 \in \mathcal{H}_1, \dots, \bar{x}_m \in \mathcal{H}_m \text{ astfel încât } \begin{cases} 0 \in L_1^*(A_1 \square C_1)(L_1 \bar{x}_1) + B_1(\bar{x}_1, \dots, \bar{x}_m) \\ \vdots \\ 0 \in L_m^*(A_m \square C_m)(L_m \bar{x}_m) + B_m(\bar{x}_1, \dots, \bar{x}_m) \end{cases} \quad (3.1.2)$$

împreună cu sistemul dual

$$\begin{aligned} & \text{căutăm } \bar{v}_1 \in \mathcal{G}_1, \dots, \bar{v}_m \in \mathcal{G}_m \text{ astfel încât} \\ & \exists x_1 \in \mathcal{H}_1, \dots, \exists x_m \in \mathcal{H}_m \end{aligned} \begin{cases} 0 = L_1^* \bar{v}_1 + B_1(x_1, \dots, x_m) \\ \vdots \\ 0 = L_m^* \bar{v}_m + B_m(x_1, \dots, x_m) \\ \bar{v}_1 \in (A_1 \square C_1)(L_1 x_1) \\ \vdots \\ \bar{v}_m \in (A_m \square C_m)(L_m x_m) \end{cases} \quad (3.1.3)$$

Spunem că $(\bar{x}_1, \dots, \bar{x}_m, \bar{v}_1, \dots, \bar{v}_m) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m \times \mathcal{G}_1 \times \dots \times \mathcal{G}_m$ este o soluție primal-duală la Problema 3.1.1, dacă

$$0 = L_i^* \bar{v}_i + B_i(\bar{x}_1, \dots, \bar{x}_m) \text{ și } \bar{v}_i \in (A_i \square C_i)(L_i \bar{x}_i), \quad i = 1, \dots, m. \quad (3.1.4)$$

Dacă $(\bar{x}_1, \dots, \bar{x}_m, \bar{v}_1, \dots, \bar{v}_m) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m \times \mathcal{G}_1 \times \dots \times \mathcal{G}_m$ este o soluție primal-duală la Problema 3.1.1, atunci $(\bar{x}_1, \dots, \bar{x}_m)$ este o soluție la (3.1.2) și $(\bar{v}_1, \dots, \bar{v}_m)$ este o soluție la (3.1.3). Observăm deasemenea că

$$(\bar{x}_1, \dots, \bar{x}_m) \text{ rezolvă (3.1.2)} \Leftrightarrow 0 \in L_i^*(A_i \square C_i)(L_i \bar{x}_i) + B_i(\bar{x}_1, \dots, \bar{x}_m), \quad i = 1, \dots, m \Leftrightarrow \\ \exists \bar{v}_1 \in \mathcal{G}_1, \dots, \bar{v}_m \in \mathcal{G}_m \text{ astfel încât } \begin{cases} 0 = L_i^* \bar{v}_i + B_i(\bar{x}_1, \dots, \bar{x}_m), \quad i = 1, \dots, m \\ \bar{v}_i \in (A_i \square C_i)(L_i \bar{x}_i), \quad i = 1, \dots, m. \end{cases}$$

Astfel, dacă $(\bar{x}_1, \dots, \bar{x}_m)$ este o soluție la (3.1.2), atunci există $(\bar{v}_1, \dots, \bar{v}_m) \in \mathcal{G}_1 \times \dots \times \mathcal{G}_m$ astfel încât $(\bar{x}_1, \dots, \bar{x}_m, \bar{v}_1, \dots, \bar{v}_m)$ să fie o soluție primal-duală la Problema 3.1.1 și, dacă $(\bar{v}_1, \dots, \bar{v}_m) \in \mathcal{G}_1 \times \dots \times \mathcal{G}_m$ este o soluție la (3.1.3), atunci există $(\bar{x}_1, \dots, \bar{x}_m) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m$ astfel încât $(\bar{x}_1, \dots, \bar{x}_m, \bar{v}_1, \dots, \bar{v}_m)$ este o soluție primal-duală la Problema 3.1.1.

3.2 Algoritmul primal-dual de divizare

Scopul acestei secțiuni este acela de a asigura un algoritm pentru rezolvarea Problemei 3.1.1 și pentru a furniza rezultate de convergență slabe și puternice pentru secvențele generate de acesta. Schema iterativă propusă are proprietatea că fiecare operator univoc este procesat în mod explicit, în timp ce fiecare operator multivoc este evaluat prin intermediul rezolventului său. Secvențe absolut sumabile fac algoritmul să fie tolerant la erori.

Algoritmul 3.2.1.

Fie $(a_{1,i,n})_{n \geq 0}$, $(b_{1,i,n})_{n \geq 0}$, $(c_{1,i,n})_{n \geq 0}$ secvențe absolut sumabile în \mathcal{H}_i , pentru fiecare $i = 1, \dots, m$ și $(a_{2,i,n})_{n \geq 0}$, $(b_{2,i,n})_{n \geq 0}$, $(c_{2,i,n})_{n \geq 0}$ secvențe absolut sumabile în \mathcal{G}_i . De asemenea, setăm

$$\beta = \max \left\{ \sqrt{\sum_{i=1}^m \mu_i^2}, \nu_1, \dots, \nu_m \right\} + \max_{i=1, \dots, m} \|L_i\|, \quad (3.2.1)$$

fie $\varepsilon \in]0, 1/(\beta + 1)[$ și $(\gamma_n)_{n \geq 0}$ o secvență din $[\varepsilon, (1 - \varepsilon)/\beta]$. Pentru fiecare $i = 1, \dots, m$, punctele inițiale $x_{i,0} \in \mathcal{H}_i$ și $v_{i,0} \in \mathcal{G}_i$ vor fi alese arbitrar și setăm

$$\forall n \geq 0 \quad \left[\begin{array}{l} \text{For } i = 1, \dots, m \\ y_{i,n} = x_{i,n} - \gamma_n(L_i^* v_{i,n} + B_i(x_{1,n}, \dots, x_{m,n}) + a_{1,i,n}) \\ w_{i,n} = v_{i,n} - \gamma_n(C_i^{-1} v_{i,n} - L_i x_{i,n} + a_{2,i,n}) \\ p_{i,n} = y_{i,n} + b_{1,i,n} \\ r_{i,n} = J_{\gamma_n A_i^{-1}} w_{i,n} + b_{2,i,n} \\ q_{i,n} = p_{i,n} - \gamma_n(L_i^* r_{i,n} + B_i(p_{1,n}, \dots, p_{m,n}) + c_{1,i,n}) \\ s_{i,n} = r_{i,n} - \gamma_n(C_i^{-1} r_{i,n} - L_i p_{i,n} + c_{2,i,n}) \\ x_{i,n+1} = x_{i,n} - y_{i,n} + q_{i,n} \\ v_{i,n+1} = v_{i,n} - w_{i,n} + s_{i,n}. \end{array} \right.$$

Convergența algoritmului 3.2.1 se stabilește prin demonstrarea faptului că schema ei iterativă poate fi redusă la versiunea algoritmului tolerant de erori forward-backward-forward al lui Tseng (cf. [59] pentru cazul fără erori), recent prevăzut în [16]. Cadrul nostru algoritmic va depinde de următorul algoritm de divizare, foarte interesant în aplicațiile lui.

Theorem 3.2.1. [16] *Fie \mathcal{H} un spațiu Hilbert real, fie $M : \mathcal{H} \rightarrow 2^{\mathcal{H}}$ maximal monoton și fie $L : \mathcal{H} \rightarrow \mathcal{H}$ monoton. Presupunem că $\text{zer}(M + L) \neq \emptyset$ și că L este β -Lipschitz continuu, pentru $\beta \in]0, +\infty[$. Fie $(a_n)_{n \in \mathbb{N}}$, $(b_n)_{n \in \mathbb{N}}$ și fie $(c_n)_{n \in \mathbb{N}}$ o secvență în \mathcal{H} astfel încât*

$$\sum_{n \in \mathbb{N}} \|a_n\| < +\infty, \quad \sum_{n \in \mathbb{N}} \|b_n\| < +\infty \quad \text{și} \quad \sum_{n \in \mathbb{N}} \|c_n\| < +\infty,$$

fie $x_0 \in \mathcal{H}$, fie $\varepsilon \in]0, 1/(\beta + 1)[$, fie $(\gamma_n)_{n \in \mathbb{N}}$ o secvență din $[\varepsilon, (1 - \varepsilon)/\beta]$, și fie

$$\forall n \in \mathbb{N} \quad \left[\begin{array}{l} y_n = x_n - \gamma_n(Lx_n + a_n) \\ p_n = J_{\gamma_n M} y_n + b_n \\ q_n = p_n - \gamma_n(Lp_n) + c_n \\ x_{n+1} = x_n - y_n + q_n. \end{array} \right.$$

Atunci următoarele condiții sunt îndeplinite pentru $\bar{x} \in \text{zer}(M + L)$:

(i) $\sum_{n \in \mathbb{N}} \|x_n - p_n\|^2 < +\infty$ și $\sum_{n \in \mathbb{N}} \|y_n - q_n\|^2 < +\infty$.

(ii) $x_n \rightharpoonup \bar{x}$ și $p_n \rightharpoonup \bar{x}$.

(iii) Presupunem că una dintre condițiile următoare este îndeplinită:

- (a) $M + L$ este demiregular în \bar{x} .
- (b) M sau L este uniform monoton în \bar{x} .
- (c) $\text{int zer}(M + L) \neq \emptyset$.

Atunci $x_n \rightarrow \bar{x}$ și $p_n \rightarrow \bar{x}$.

Următoarea teoremă stabilește convergența Algoritmului 3.2.1:

Theorem 3.2.2. [10] Presupunem că Problema 3.1.1 are o soluție primal-duală. În cazul secvențelor generate de Algoritmul 3.2.1, următoarele afirmații sunt adevărate:

- (i) $\forall i \in \{1, \dots, m\} \quad \sum_{n \geq 0} \|x_{i,n} - p_{i,n}\|_{\mathcal{H}_i}^2 < +\infty$ și $\sum_{n \geq 0} \|v_{i,n} - r_{i,n}\|_{\mathcal{G}_i}^2 < +\infty$.
- (ii) Există o soluție primal-duală $(\bar{x}_1, \dots, \bar{x}_m, \bar{v}_1, \dots, \bar{v}_m)$ la Problema 3.1.1 astfel încât:

- (a) $\forall i \in \{1, \dots, m\} \quad x_{i,n} \rightarrow \bar{x}_i, p_{i,n} \rightarrow \bar{x}_i, v_{i,n} \rightarrow \bar{v}_i$ și $r_{i,n} \rightarrow \bar{v}_i$ pentru $n \rightarrow +\infty$.
- (b) Dacă $C_i^{-1}, i = 1, \dots, m$, este uniform monoton și există o funcție crescătoare $\phi_B : \mathbb{R}_+ \rightarrow \mathbb{R}_+ \cup \{+\infty\}$ care dispare numai la valoarea 0 și satisface

$$\begin{aligned} & \forall (x_1, \dots, x_m) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m, \forall (y_1, \dots, y_m) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m \\ & \sum_{i=1}^m \langle x_i - y_i | B_i(x_1, \dots, x_m) - B_i(y_1, \dots, y_m) \rangle_{\mathcal{H}_i} \geq \\ & \phi_B \left(\|(x_1, \dots, x_m) - (y_1, \dots, y_m)\| \right), \end{aligned} \quad (3.2.2)$$

atunci $\forall i \in \{1, \dots, m\} \quad x_{i,n} \rightarrow \bar{x}_i, p_{i,n} \rightarrow \bar{x}_i, v_{i,n} \rightarrow \bar{v}_i$ și $r_{i,n} \rightarrow \bar{v}_i$ pentru $n \rightarrow +\infty$.

3.3 Aplicații la probleme de minimizare convexe

În această secțiune ne vom îndrepta atenția spre rezolvarea problemelor de minimizare convexe cu variabile multiple, prin intermediul algoritmului primal-dual prezentat și investigat în această teză.

Problema 3.3.1. Fie $m \geq 1$ și $p \geq 1$ numere pozitive, $(\mathcal{H}_i)_{1 \leq i \leq m}$, fie $(\mathcal{H}'_i)_{1 \leq i \leq m}$ și $(\mathcal{G}_j)_{1 \leq j \leq p}$ spații Hilbert reale, $f_i, h_i \in \Gamma(\mathcal{H}'_i)$ astfel încât h_i sunt ν_i^{-1} -puternic convexe

cu $\nu_i \in \mathbb{R}_{++}$, $i = 1, \dots, m$, și $g_j \in \Gamma(\mathcal{G}_j)$ pentru $i = 1, \dots, m$, $j = 1, \dots, p$. De asemenea, fie $K_i : \mathcal{H}_i \rightarrow \mathcal{H}'_i$ și $L_{ji} : \mathcal{H}_i \rightarrow \mathcal{G}_j$, $i = 1, \dots, m$, $j = 1, \dots, p$ operatori liniari continui. Considerăm următoarea problemă de optimizare convexă

$$\inf_{(x_1, \dots, x_m) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m} \left\{ \sum_{i=1}^m (f_i \square h_i)(K_i x_i) + \sum_{j=1}^p g_j \left(\sum_{i=1}^m L_{ji} x_i \right) \right\}. \quad (3.3.1)$$

În cele ce urmează, vom arăta că sub o condiție de calificare corespunzătoare, rezolvarea problemei de optimizare convexă (3.3.1) poate fi redusă la rezolvarea unui sistem de incluziuni monotone de tip (3.1.2).

Definim următoarea funcție convexă adecvată și inferior semicontinuuă

$$f : \mathcal{H}'_1 \times \dots \times \mathcal{H}'_m \rightarrow \overline{\mathbb{R}}, \quad (y_1, \dots, y_m) \mapsto \sum_{i=1}^m (f_i \square h_i)(y_i),$$

și operatorul liniar continuu

$$K : \mathcal{H}_1 \times \dots \times \mathcal{H}_m \rightarrow \mathcal{H}'_1 \times \dots \times \mathcal{H}'_m, \quad (x_1, \dots, x_m) \mapsto (K_1 x_1, \dots, K_m x_m),$$

împreună cu adjuncta

$$K^* : \mathcal{H}'_1 \times \dots \times \mathcal{H}'_m \rightarrow \mathcal{H}_1 \times \dots \times \mathcal{H}_m, \quad (y_1, \dots, y_m) \mapsto (K_1^* y_1, \dots, K_m^* y_m).$$

De asemenea, considerăm operatorii liniari continui

$$L_j : \mathcal{H}_1 \times \dots \times \mathcal{H}_m \rightarrow \mathcal{G}_j, \quad (x_1, \dots, x_m) \mapsto \sum_{i=1}^m L_{ji} x_i, \quad j = 1, \dots, p,$$

împreună cu adjunctele

$$L_j^* : \mathcal{G}_j \rightarrow \mathcal{H}_1 \times \dots \times \mathcal{H}_m, \quad y \mapsto (L_{j1}^* y, \dots, L_{jm}^* y), \quad j = 1, \dots, p,$$

respectiv. Avem

$$\begin{aligned} & (\bar{x}_1, \dots, \bar{x}_m) \text{ este o soluție optimă la (3.3.1)} \\ \Leftrightarrow & (0, \dots, 0) \in \partial \left(f \circ K + \sum_{j=1}^p g_j \circ L_j \right) (\bar{x}_1, \dots, \bar{x}_m). \end{aligned} \quad (3.3.2)$$

În scopul de a împărții subdiferențialele menționate mai sus, într-o sumă de subdiferențiale, trebuie îndeplinite niște condiții de calificare. În acest context, considerăm următoarele condiții de calificare de tip interioritate:

$$(QC_1) \left| \begin{array}{l} \text{există } x'_i \in \mathcal{H}_i \text{ astfel încât} \\ K_i x'_i \in (\text{dom } f_i + \text{dom } h_i) \text{ și } f_i \square h_i \text{ este continuu în } K_i x'_i, i = 1, \dots, m, \\ \text{și } \sum_{i=1}^m L_{ji} x'_i \in \text{dom } g_j \text{ și } g_j \text{ este continuu în } \sum_{i=1}^m L_{ji} x'_i, j = 1, \dots, p \end{array} \right.$$

și

$$(QC_2) \left| \begin{array}{l} (0, \dots, 0) \in \text{sqr} \left(\prod_{i=1}^m (\text{dom } f_i + \text{dom } h_i) \times \prod_{j=1}^p \text{dom } g_j \right. \\ \left. - \{ (K_1 x_1, \dots, K_m x_m, \sum_{i=1}^m L_{1i} x_i, \dots, \sum_{i=1}^m L_{pi} x_i) : \right. \right. \\ \left. \left. (x_1, \dots, x_m) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m \} \right). \end{array} \right.$$

Se observă că $(QC_1) \rightarrow (QC_2)$, aceste implicații fiind stricte, și facem referință la [4, 7, 8, 29, 58, 68, 69] pentru alte condiții de calificare în optimizare convexă.

Remarcă 3.3.1. După cum am subliniat deja, pentru $i = 1, \dots, m$, $f_i \square h_i \in \Gamma(h'_i)$, prin urmare, este continuă pe $\text{int}(\text{dom } f_i + \text{dom } h_i)$, presupunând că acest set nu este vid ([29, 69]). Pentru alte rezultate referitoare la continuitatea convoluției infimale a funcțiilor convexe invităm cititorul să consulte [57].

Remarcă 3.3.2. În spații finit-dimensionale, condiția de calificare (QC_2) este echivalentă cu

$$(QC_2) \left| \begin{array}{l} \text{există } x'_i \in \mathcal{H}_i \text{ astfel încât } K_i x'_i \in \text{ri dom } f_i + \text{ri dom } h_i, i = 1, \dots, m, \\ \text{și } \sum_{i=1}^m L_{ji} x'_i \in \text{ri dom } g_j, j = 1, \dots, p. \end{array} \right.$$

Presupunând că una dintre condițiile de calificare menționate mai sus este

îndeplinită, avem

$$\begin{aligned}
& (\bar{x}_1, \dots, \bar{x}_m) \text{ este o soluție optimală la (3.3.1)} \\
& \Leftrightarrow (0, \dots, 0) \in K^* \partial f(K(\bar{x}_1, \dots, \bar{x}_m)) + \sum_{j=1}^p L_j^* \partial g_j(L_j(\bar{x}_1, \dots, \bar{x}_m)) \\
& \Leftrightarrow (0, \dots, 0) \in \left(K_1^* \partial(f_1 \square h_1)(K_1 \bar{x}_1), \dots, K_m^* \partial(f_m \square h_m)(K_m \bar{x}_m) \right) \\
& \quad + \sum_{j=1}^p L_j^* \partial g_j(L_j(\bar{x}_1, \dots, \bar{x}_m)). \tag{3.3.3}
\end{aligned}$$

Convergența puternică a funcțiilor h_i implică $\text{dom } h_i^* = \mathcal{H}'_i$, astfel $\partial(f_i \square h_i) = \partial f_i \square \partial h_i, i = 1, \dots, m$. Prin urmare, (3.3.3) este echivalentă cu

$$(0, \dots, 0) \in \left(K_1^* (\partial f_1 \square \partial h_1)(K_1 \bar{x}_1), \dots, K_m^* (\partial f_m \square \partial h_m)(K_m \bar{x}_m) \right) + \sum_{j=1}^p L_j^* v_j,$$

unde

$$\bar{v}_j \in \partial g_j(L_j(\bar{x}_1, \dots, \bar{x}_m)) \Leftrightarrow \bar{v}_j \in \partial g_j\left(\sum_{i=1}^m L_{ji} \bar{x}_i\right) \Leftrightarrow \sum_{i=1}^m L_{ji} \bar{x}_i \in \partial g_j^*(\bar{v}_j), j = 1, \dots, p.$$

Atunci $(\bar{x}_1, \dots, \bar{x}_m)$ este o soluție optimală la (3.3.1) dacă și numai dacă $(\bar{x}_1, \dots, \bar{x}_m, \bar{v}_1, \dots, \bar{v}_p)$ este o soluție la

$$\begin{cases}
0 \in K_1^* (\partial f_1 \square \partial h_1)(K_1 \bar{x}_1) + \sum_{j=1}^p L_{j1}^* \bar{v}_j \\
\vdots \\
0 \in K_m^* (\partial f_m \square \partial h_m)(K_m \bar{x}_m) + \sum_{j=1}^p L_{jm}^* \bar{v}_j \\
0 \in \partial g_1^*(\bar{v}_1) - \sum_{i=1}^m L_{1i} \bar{x}_i \\
\vdots \\
0 \in \partial g_p^*(\bar{v}_p) - \sum_{i=1}^m L_{pi} \bar{x}_i.
\end{cases} \tag{3.3.4}$$

Se observă că (3.3.4) este un sistem de incluziuni cuplate de tip (3.1.2), prin

$$A_i = \partial f_i, C_i = \partial h_i, L_i = K_i, i = 1, \dots, m,$$

$$A_{m+j} = \partial g_j^*, \quad C_{m+j}(x) = \begin{cases} \mathcal{G}_j, & x = 0 \\ \emptyset, & \text{altfel} \end{cases},$$

$$L_{m+j} = \text{Id}_{\mathcal{G}_j}, \quad j = 1, \dots, p,$$

și, pentru $(x_1, \dots, x_m, v_1, \dots, v_p) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m \times \mathcal{G}_1 \times \dots \times \mathcal{G}_p$, operatorii de cuplare sunt

$$B_i(x_1, \dots, x_m, v_1, \dots, v_p) = \sum_{j=1}^p L_{ji}^* v_j, \quad i = 1, \dots, m,$$

și

$$B_{m+j}(x_1, \dots, x_m, v_1, \dots, v_p) = - \sum_{i=1}^m L_{ji} x_i, \quad j = 1, \dots, p.$$

Definim

$$B(x_1, \dots, x_m, v_1, \dots, v_p) = (B_1, \dots, B_{m+p})(x_1, \dots, x_m, v_1, \dots, v_p) \quad (3.3.5)$$

$$= \left(\sum_{j=1}^p L_{j1}^* v_j, \dots, \sum_{j=1}^p L_{jm}^* v_j, - \sum_{i=1}^m L_{1i} x_i, \dots, - \sum_{i=1}^m L_{pi} x_i \right).$$

$$(3.3.6)$$

Rezultă că $C_i^{-1} = (\partial h_i)^{-1} = \partial h_i^* = \{\nabla h_i^*\}$ este ν_i -Lipschitz continuu pentru $i = 1, \dots, m$. Pe de altă parte, C_{m+j}^{-1} este operatorul zero pentru $j = 1, \dots, p$, prin urmare 0-Lipschitz continuu.

De asemenea, operatorii $B_i, i = 1, \dots, m + p$ sunt liniari și Lipschitz continuu, având constantele Lipschitz

$$\mu_i = \sqrt{\sum_{j=1}^p \|L_{ji}\|^2}, \quad i = 1, \dots, m, \quad \text{și} \quad \mu_{m+j} = \sqrt{\sum_{i=1}^m \|L_{ji}\|^2}, \quad j = 1, \dots, p,$$

respectiv. Pentru fiecare $(x_1, \dots, x_m, v_1, \dots, v_p), (y_1, \dots, y_m, w_1, \dots, w_p) \in \mathcal{H}_1 \times \dots \times$

$\mathcal{H}_m \times \mathcal{G}_1 \times \dots \times \mathcal{G}_p$ rezultă

$$\begin{aligned} & \sum_{i=1}^m \langle x_i - y_i | B_i(x_1, \dots, x_m, v_1, \dots, v_p) - B_i(y_1, \dots, y_m, w_1, \dots, w_p) \rangle_{\mathcal{H}_i} \\ & + \sum_{j=1}^p \langle v_j - w_j | B_{m+j}(x_1, \dots, x_m, v_1, \dots, v_p) - B_{m+j}(y_1, \dots, y_m, w_1, \dots, w_p) \rangle_{\mathcal{G}_j} \\ & = \sum_{i=1}^m \left\langle x_i - y_i \left| \sum_{j=1}^p L_{ji}^* v_j - \sum_{j=1}^p L_{ji}^* w_j \right. \right\rangle_{\mathcal{H}_i} - \sum_{j=1}^p \left\langle v_j - w_j \left| \sum_{i=1}^m L_{ji} x_i - \sum_{i=1}^m L_{ji} y_i \right. \right\rangle_{\mathcal{G}_j} = 0, \end{aligned}$$

prin urmare (3.1.1) este îndeplinită. Astfel am demonstrat și faptul că operatorul liniar continuu B este oblic (i.e. $B^* = -B$).

Remarcă 3.3.3. Datorită faptului că operatorul B este oblic, nu este cocoerciv, prin urmare, metoda prezentată în [1] nu poate fi aplicată în acest context. Pe de altă parte, în scopul de a determina o soluție optimă a problemei de optimizare (3.3.1) (și o soluție optimă a dualei de tip Fenchel), se pot utiliza algoritmi primal-duali de divizare proximali care au fost introduse recent în [23, 61]. Aceste abordări au particularitatea de a face față în mod eficient sumelor de compoziții cu funcții corespunzătoare, convexe și inferior semicontinuu și cu operatori liniari continuu, evaluând separat fiecare funcție printr-un pas backward și fiecare operator liniar continuu (și adjuncta acestuia) prin intermediul unui pas forward. Cu toate acestea, sistemul iterativ pe care îl propunem în această secțiune pentru rezolvarea (3.3.1), are avantajul de a exploata structura problemei separabile.

Să amintim, de asemenea, că problema duală de incluziuni la (3.3.4) este (a se vedea (3.1.3))

$$\begin{array}{l}
 \text{căutăm } \bar{w}_1 \in \mathcal{H}'_1, \dots, \bar{w}_m \in \mathcal{H}'_m, \\
 \bar{w}_{m+1} \in \mathcal{G}_1, \dots, \bar{w}_{m+p} \in \mathcal{G}_p \text{ astfel încât} \\
 \exists x_1 \in \mathcal{H}_1, \dots, \exists x_m \in \mathcal{H}_m, \exists v_1 \in \mathcal{G}_1, \dots, \exists v_p \in \mathcal{G}_p
 \end{array}
 \left\{ \begin{array}{l}
 0 = K_1^* \bar{w}_1 + \sum_{j=1}^p L_{j1}^* v_j \\
 \vdots \\
 0 = K_m^* \bar{w}_m + \sum_{j=1}^p L_{jm}^* v_j \\
 0 = \bar{w}_{m+1} - \sum_{i=1}^m L_{1i} x_i \\
 \vdots \\
 0 = \bar{w}_{m+p} - \sum_{i=1}^m L_{pi} x_i \\
 \bar{w}_1 \in (\partial f_1 \square \partial h_1)(K_1 x_1) \\
 \vdots \\
 \bar{w}_m \in (\partial f_m \square \partial h_m)(K_m x_m) \\
 \bar{w}_{m+1} \in \partial g_1^*(v_1) \\
 \vdots \\
 \bar{w}_{m+p} \in \partial g_p^*(v_p).
 \end{array} \right. \quad (3.3.7)$$

Atunci $(\bar{x}_1, \dots, \bar{x}_m, \bar{v}_1, \dots, \bar{v}_p, \bar{w}_1, \dots, \bar{w}_m, \bar{w}_{m+1}, \dots, \bar{w}_{m+p})$ este o soluție primal-duală la (3.3.4) - (3.3.7), dacă

$$\begin{array}{l}
 \bar{w}_i \in (\partial f_i \square \partial h_i)(K_i \bar{x}_i), \bar{w}_{m+j} \in \partial g_j^*(\bar{v}_j), \\
 0 = K_i^* \bar{w}_i + \sum_{j=1}^p L_{ji}^* \bar{v}_j \text{ și } 0 = \bar{w}_{m+j} - \sum_{i=1}^m L_{ji} \bar{x}_i, i = 1, \dots, m, j = 1, \dots, p.
 \end{array}$$

Cu condiția că $(\bar{x}_1, \dots, \bar{x}_m, \bar{v}_1, \dots, \bar{v}_p, \bar{w}_1, \dots, \bar{w}_m, \bar{w}_{m+1}, \dots, \bar{w}_{m+p})$ este o soluție primal-duală la (3.3.4) - (3.3.7), rezultă că $(\bar{x}_1, \dots, \bar{x}_m)$ este o soluție optimală la (3.3.1) și $(\bar{w}_1, \dots, \bar{w}_m, \bar{v}_1, \dots, \bar{v}_p)$ este o soluție optimală la problema duală de tip *Fenchel*

$$\sup_{\substack{(w_1, \dots, w_m, w_{m+1}, \dots, w_{m+p}) \in \mathcal{H}'_1 \times \dots \times \mathcal{H}'_m \times \mathcal{G}_1 \times \dots \times \mathcal{G}_p \\ K_i^* w_i + \sum_{j=1}^p L_{ji}^* w_{m+j} = 0, i=1, \dots, m}} \left\{ - \sum_{i=1}^m (f_i^*(w_i) + h_i^*(w_i)) - \sum_{j=1}^p g_j^*(w_{m+j}) \right\}. \quad (3.3.8)$$

Algoritmul 3.2.1 dă naștere următoarei scheme iterative pentru rezolvarea (3.3.4) - (3.3.7).

Algoritmul 3.3.1.

Pentru fiecare $i = 1, \dots, m$ și fiecare $j = 1, \dots, p$ fie $(a_{1,i,n})_{n \geq 0}$, $(b_{1,i,n})_{n \geq 0}$, $(c_{1,i,n})_{n \geq 0}$, secvențe absolut sumabile în \mathcal{H}_i , $(a_{2,i,n})_{n \geq 0}$, $(b_{2,i,n})_{n \geq 0}$, $(c_{2,i,n})_{n \geq 0}$ secvențe absolut sumabile în \mathcal{H}'_i și $(a_{1,m+j,n})_{n \geq 0}$, $(a_{2,m+j,n})_{n \geq 0}$, $(b_{1,m+j,n})_{n \geq 0}$, $(b_{2,m+j,n})_{n \geq 0}$, $(c_{1,m+j,n})_{n \geq 0}$ și $(c_{2,m+j,n})_{n \geq 0}$ secvențe absolut sumabile în \mathcal{G}_j . De asemenea, fie

$$\beta = \max \left\{ \sqrt{\sum_{i=1}^{m+p} \mu_i^2}, \nu_1, \dots, \nu_m \right\} + \max \{ \|K_1\|, \dots, \|K_m\|, 1 \}, \quad (3.3.9)$$

unde

$$\mu_i = \sqrt{\sum_{j=1}^p \|L_{ji}\|^2}, \quad i = 1, \dots, m, \quad \text{și} \quad \mu_{m+j} = \sqrt{\sum_{i=1}^m \|L_{ji}\|^2}, \quad j = 1, \dots, p, \quad (3.3.10)$$

fie $\varepsilon \in]0, 1/(\beta + 1)[$ și fie $(\gamma_n)_{n \geq 0}$ o secvență din $[\varepsilon, (1 - \varepsilon)/\beta]$. Fie punctele inițiale $(x_{1,1,0}, \dots, x_{1,m,0}) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m$, $(x_{2,1,0}, \dots, x_{2,m,0}) \in \mathcal{H}'_1 \times \dots \times \mathcal{H}'_m$ și

$(v_{1,1,0}, \dots, v_{1,p,0}), (v_{2,1,0}, \dots, v_{2,p,0}) \in \mathcal{G}_1 \times \dots \times \mathcal{G}_p$ alese arbitrar și setăm

$$\forall n \geq 0 \left[\begin{array}{l} \text{For } i = 1, \dots, m \\ \left[\begin{array}{l} y_{1,i,n} = x_{1,i,n} - \gamma_n \left(K_i^* x_{2,i,n} + \sum_{j=1}^p L_{ji}^* v_{1,j,n} + a_{1,i,n} \right) \\ y_{2,i,n} = x_{2,i,n} - \gamma_n (\nabla h_i^* x_{2,i,n} - K_i x_{1,i,n} + a_{2,i,n}) \\ p_{1,i,n} = y_{1,i,n} + b_{1,i,n} \\ p_{2,i,n} = \text{Prox}_{\gamma_n f_i^*} y_{2,i,n} + b_{2,i,n} \end{array} \right. \\ \text{For } j = 1, \dots, p \\ \left[\begin{array}{l} w_{1,j,n} = v_{1,j,n} - \gamma_n (v_{2,j,n} - \sum_{i=1}^m L_{ji} x_{1,i,n} + a_{1,m+j,n}) \\ w_{2,j,n} = v_{2,j,n} - \gamma_n (-v_{1,j,n} + a_{2,m+j,n}) \\ r_{1,j,n} = w_{1,j,n} + b_{1,m+j,n} \\ r_{2,j,n} = \text{Prox}_{\gamma_n g_j} w_{2,j,n} + b_{2,m+j,n} \end{array} \right. \\ \text{For } i = 1, \dots, m \\ \left[\begin{array}{l} q_{1,i,n} = p_{1,i,n} - \gamma_n \left(K_i^* p_{2,i,n} + \sum_{j=1}^p L_{ji}^* r_{1,j,n} + c_{1,i,n} \right) \\ q_{2,i,n} = p_{2,i,n} - \gamma_n (\nabla h_i^* p_{2,i,n} - K_i p_{1,i,n} + c_{2,i,n}) \\ x_{1,i,n+1} = x_{1,i,n} - y_{1,i,n} + q_{1,i,n} \\ x_{2,i,n+1} = x_{2,i,n} - y_{2,i,n} + q_{2,i,n} \end{array} \right. \\ \text{For } j = 1, \dots, p \\ \left[\begin{array}{l} s_{1,j,n} = r_{1,j,n} - \gamma_n (r_{2,j,n} - \sum_{i=1}^m L_{ji} p_{1,i,n} + c_{1,m+j,n}) \\ s_{2,j,n} = r_{2,j,n} - \gamma_n (-r_{1,j,n} + c_{2,m+j,n}) \\ v_{1,j,n+1} = v_{1,j,n} - w_{1,j,n} + s_{1,j,n} \\ v_{2,j,n+1} = v_{2,j,n} - w_{2,j,n} + s_{2,j,n} \end{array} \right. \end{array} \right.$$

Următorul rezultat de convergență pentru algoritmul 3.3.1 este o consecință a Teoremei 3.2.2.

Theorem 3.3.1. [10] *Presupunem că problema de optimizare (3.3.1) are o soluție optimală și că una dintre condițiile de calificare (QC_i) , $i = 1, 2$, este îndeplinită.*

Următoarele afirmații sunt adevărate pentru secvențele generate de Algoritmul 3.3.1:

$$(i) \forall i \in \{1, \dots, m\} \quad \sum_{n \geq 0} \|x_{1,i,n} - p_{1,i,n}\|_{\mathcal{H}_i}^2 < +\infty, \quad \sum_{n \geq 0} \|x_{2,i,n} - p_{2,i,n}\|_{\mathcal{H}_i}^2 < +\infty \text{ și}$$

$$\forall j \in \{1, \dots, p\} \quad \sum_{n \geq 0} \|v_{1,j,n} - r_{1,j,n}\|_{\mathcal{G}_j}^2 < +\infty, \quad \sum_{n \geq 0} \|v_{2,j,n} - r_{2,j,n}\|_{\mathcal{G}_j}^2 < +\infty.$$

(ii) *Există o soluție optimală $(\bar{x}_1, \dots, \bar{x}_m)$ la (3.3.1) și o soluție optimală*

$(\bar{w}_1, \dots, \bar{w}_m, \bar{w}_{m+1}, \dots, \bar{w}_{m+p})$ la (3.3.8), astfel încât $\forall i \in \{1, \dots, m\}$ $x_{1,i,n} \rightarrow \bar{x}_i$, $p_{1,i,n} \rightarrow \bar{x}_i$, $x_{2,i,n} \rightarrow \bar{w}_i$ și $p_{2,i,n} \rightarrow \bar{w}_i$ și $\forall j \in \{1, \dots, p\}$ $v_{1,j,n} \rightarrow \bar{w}_{m+j}$ și $r_{1,j,n} \rightarrow \bar{w}_{m+j}$ pentru $n \rightarrow +\infty$.

Remarcă 3.3.4. Recent, a fost propus un alt sistem iterativ pentru rezolvarea sistemelor de incluziuni monotone în [22], care, de asemenea, este capabil să rezolve probleme de optimizare de tip (3.3.1), în cazul în care funcțiile $g_j, j = 1, \dots, p$, nu sunt neapărat diferentiabile. Metoda este diferită de abordarea noastră, care presupune că variabilele sunt cuplate de către operatorul univoc B , în [22] cuplarea se face de către unele compoziții ale sumelor paralele de operatori maximal monotoni cu cele liniare continue.

Capitolul 4

Experimente numerice

Rezultatele teoretice obținute sunt foarte aplicabile în diferite domenii ale matematicii. În această secțiune vom prezenta cele patru experimente numerice, care pun accentul pe performanța algoritmului primal-dual și pe unele dintre variantele sale pentru sisteme de incluziuni monotone cuplate, cum ar fi prelucrarea imaginilor, probleme de localizare, consensul mediei în cazul rețelelor colorate și clasificarea imaginilor prin intermediul vectorilor suport.

4.1 Procesarea imaginilor

Primul experiment numeric rezolvă problema de procesare a imaginilor prin algoritmul de divizare primal-dual dezvoltat în această teză. Scopul este de a estima imaginea originală necunoscută din imaginea neclară, cu ajutorul algoritmului.

4.1.1 Claritatea imaginilor

Fie \mathcal{H} și $(\mathcal{H}'_i)_{1 \leq i \leq m}$ spații Hilbert reale, unde $m \geq 1$. Pentru fiecare $i \in \{1, \dots, m\}$, fie $f_i \in \Gamma(\mathcal{H}'_i)$ și fie $K_i : \mathcal{H} \rightarrow \mathcal{H}'_i$ operatori liniari continui. Considerăm problema inițială de optimizare convexă

$$\inf_{x \in \mathcal{H}} \left\{ \sum_{i=1}^m f_i(K_i x) \right\}. \quad (4.1.1)$$

În cazul variabilelor multiple, (4.1.1) poate fi formulată ca:

$$\inf_{x_1, \dots, x_{m+1}} \left\{ \sum_{i=1}^m f_i(K_i x_i) + \sum_{i=1}^m \delta_{\{0\}}(x_i - x_{m+1}) \right\}. \quad (4.1.2)$$

Acesta este un caz special al Problemei 3.3.1 cu $(m, p) := (m + 1, m)$, $f_{m+1} = 0$, $K_{m+1} = 0$ și $\forall i \in \{1, \dots, m + 1\}$, $h_i = \delta_{\{0\}}$ și $\forall j \in \{1, \dots, m\}$, $\mathcal{G}_j = \mathcal{H}$, $g_j = \delta_{\{0\}}$ și

$$L_{ji} : \mathcal{H}_i \rightarrow \mathcal{H}_j \mapsto \begin{cases} \text{Id, if } i = j; \\ -\text{Id, if } i = m + 1; \\ 0, \text{ altfel.} \end{cases} \quad (4.1.3)$$

Fie $(x_{1,1,0}, \dots, x_{1,m+1,0})$, $(x_{2,1,0}, \dots, x_{2,m+1,0}) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_{m+1}$ și $(v_{1,1,0}, \dots, v_{1,m,0})$, $(v_{2,1,0}, \dots, v_{2,m,0}) \in \mathcal{H}_1 \times \dots \times \mathcal{H}_m$. În cazul acesta, din (3.3.9) obținem

$$\beta = 2\sqrt{m} + \max \left\{ \|K_1\|, \dots, \|K_{m+1}\|, 1 \right\}. \quad (4.1.4)$$

Algoritmul 4.1.1.

Prin urmare, schema iterativă din Algoritmul 3.3.1 se schimbă în

$$\forall n \geq 0 \quad \left[\begin{array}{l} \text{For } i = 1, \dots, m \\ \quad \left[\begin{array}{l} y_{1,i,n} = x_{1,i,n} - \gamma_n (K_i^* x_{2,i,n} + v_{1,i,n}) \\ w_{1,i,n} = v_{1,i,n} - \gamma_n (v_{2,i,n} - x_{1,i,n} + x_{1,m+1,n}) \\ y_{2,i,n} = x_{2,i,n} + \gamma_n K_i x_{1,i,n} \\ w_{2,i,n} = v_{2,i,n} + \gamma_n v_{1,i,n} \end{array} \right. \\ y_{1,m+1,n} = x_{1,m+1,n} + \gamma_n \sum_{i=1}^m v_{1,i,n} \\ \text{For } i = 1, \dots, m \\ \quad \left[\begin{array}{l} x_{1,i,n+1} = x_{1,i,n} - \gamma_n (K_i^* \text{prox}_{\gamma_n f_i^*} y_{2,i,n} + w_{1,i,n}) \\ v_{1,i,n+1} = v_{1,i,n} + \gamma_n (y_{1,i,n} - y_{1,m+1,n}) \\ x_{2,i,n+1} = x_{2,i,n} - y_{2,i,n} + \text{prox}_{\gamma_n f_i^*} y_{2,i,n} + \gamma_n K_i y_{1,i,n} \\ v_{2,i,n+1} = v_{2,i,n} - w_{2,i,n} + \gamma_n w_{1,i,n} \end{array} \right. \\ x_{1,m+1,n+1} = x_{1,m+1,n} + \gamma_n \sum_{i=1}^m w_{1,i,n} \end{array} \right.$$

Atunci $(x_{1,1,n}, \dots, x_{1,m+1,n}) \rightarrow (\bar{x}, \dots, \bar{x})$.

Considerăm matricea $A \in \mathbb{R}^{n \times n}$, care reprezintă *operatorul neclar* și vectorul $b \in \mathbb{R}^n$, care descrie *imaginea neclară*. Scopul este de a estima $x^* \in \mathbb{R}^n$, care este de fapt *imaginea originală necunoscută* și care îndeplinește

$$Ax = b.$$

Problema care trebuie rezolvată este echivalentă cu

$$\inf_{x \in S} \{f_1(x) + f_2(Ax)\}, \quad (4.1.5)$$

pentru $f_1 : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$, $f_1(x) = \lambda \|x\|_1 + \delta_S(x)$ și $f_2 : \mathbb{R}^n \rightarrow \mathbb{R}$, $f_2(y) = \|y - b\|^2$. Prin urmare f este corespunzător, convex și inferior semicontinuu cu domeniu mărginit și g este o funcție 2-puternic convexă cu domeniul complet, diferențiabil și cu un gradient Lipschitz continuu, care are constanta Lipschitz egală cu 2. $\lambda > 0$ reprezintă parametrul de regularizare și $S \subseteq \mathbb{R}^n$ este un cub n -dimensional reprezentând gama de pixeli. Deoarece fiecare pixel din tonurile de gri furnizează o valoare care este între 0 și 255, o alegere naturală pentru setul convex S ar fi cubul n -dimensional $[0, 255]^n \subseteq \mathbb{R}^n$. În scopul de a reduce constanta Lipschitz care apare în abordarea dezvoltată, am scalat imaginile folosite pentru a exemplifica aplicației astfel încât fiecare pixel se află în intervalul $[0, \frac{1}{10}]$.

4.1.2 Cadru experimental

Vom lucra în mod concret cu imaginile de teste *Cameraman*, *Lena* și *Barbara* cu dimensiunile 256×256 .

Imaginea de test *Cameraman* face parte din setul de instrumente de prelucrare a imaginii din Matlab și dimensiunea imaginii vectorizate și scalate este egală cu $n = 2^{256} = 65536$. Prin utilizarea funcțiilor Matlab `imfilter` și `fspecial`, această imagine este făcută neclară după cum urmează:

```

1 H=fspecial('gaussian',9,4); % zgomot gaussian de marime 9
2                               % deviatie standard 4
3 B=imfilter(X,H,'conv','symmetric'); % B=imaginea neclara
4                               % X=imaginea originala

```

În rândul 1, funcția `fspecial` returnează un filtru de rotație simetrică de tip Gauss de mărime 9×9 cu deviația standard 4. Parametrii lui H sunt pozitive și suma lor este 1. În rândul 3, funcția `imfilter` folosește filtrul H pentru imaginea $X \in \mathbb{R}^{256 \times 256}$ și furnizează o imagine neclară $B \in \mathbb{R}^{256 \times 256}$. Opțiunea de limită "simetric" corespunde condițiilor reflexive de limită. Pentru mai multe aplicații interesante în domeniul imaginilor neclare, clarificate cu tehnici diferite, facem referință la [11–15, 19].

Datorită filtrului de rotație simetrică H , operatorul liniar $A \in \mathbb{R}^{n \times n}$, calculat cu ajutorul funcției Matlab `imfilter`, este la rândul său simetric. După adăugarea unui zgomot alb de tip Gauss, de medie zero, cu deviația standard egală cu 10^{-4} , obținem imaginea neclară $b \in \mathbb{R}^n$.

În cazul acesta particular $m := 2$, $K_1 = \text{Id}$, $K_2 = A$ și β este egal cu

$$\beta = 2\sqrt{2} + \max \{ \|A\|, 1 \}. \quad (4.1.6)$$

Folosind descompunerea spectrală reală a lui A , rezultă că $\|A\|^2 = 1$. Prin urmare $\beta = 2\sqrt{2} + 1$.

Algoritmul 4.1.2.

Algoritmul 4.1.1 poate fi scris ca

$$\forall n \geq 0 \left[\begin{array}{l} y_{1,1,n} = x_{1,1,n} - \gamma_n(x_{2,1,n} + v_{1,1,n}) \\ w_{1,1,n} = v_{1,1,n} - \gamma_n(v_{2,1,n} - x_{1,1,n} + x_{1,3,n}) \\ y_{2,1,n} = x_{2,1,n} + \gamma_n x_{1,1,n} \\ w_{2,1,n} = v_{2,1,n} + \gamma_n v_{1,1,n} \\ y_{1,2,n} = x_{1,2,n} - \gamma_n(A^* x_{2,2,n} + v_{1,2,n}) \\ w_{1,2,n} = v_{1,2,n} - \gamma_n(v_{2,2,n} - x_{1,2,n} + x_{1,3,n}) \\ y_{2,2,n} = x_{2,2,n} + \gamma_n A x_{1,2,n} \\ w_{2,2,n} = v_{2,2,n} + \gamma_n v_{1,2,n} \\ y_{1,3,n} = x_{1,3,n} + \gamma_n(v_{1,1,n} + v_{1,2,n}) \\ x_{1,1,n+1} = x_{1,1,n} - \gamma_n(\text{prox}_{\gamma_n f_1^*} y_{2,1,n} + w_{1,1,n}) \\ v_{1,1,n+1} = v_{1,1,n} + \gamma_n(y_{1,1,n} - y_{1,3,n}) \\ x_{2,1,n+1} = x_{2,1,n} - y_{2,1,n} + \text{prox}_{\gamma_n f_1^*} y_{2,1,n} + \gamma_n y_{1,1,n} \\ v_{2,1,n+1} = v_{2,1,n} - w_{2,1,n} + \gamma_n w_{1,1,n} \\ x_{1,2,n+1} = x_{1,2,n} - \gamma_n(A^* \text{prox}_{\gamma_n f_2^*} y_{2,2,n} + w_{1,2,n}) \\ v_{1,2,n+1} = v_{1,2,n} + \gamma_n(y_{1,2,n} - y_{1,3,n}) \\ x_{2,2,n+1} = x_{2,2,n} - y_{2,2,n} + \text{prox}_{\gamma_n f_2^*} y_{2,2,n} + \gamma_n A y_{1,2,n} \\ v_{2,2,n+1} = v_{2,2,n} - w_{2,2,n} + \gamma_n w_{1,2,n} \\ x_{1,3,n+1} = x_{1,3,n} + \gamma_n(w_{1,1,n} + w_{1,2,n}). \end{array} \right.$$

Atunci $(x_{1,1,n}, x_{1,2,n}, x_{1,3,n}) \rightarrow (\bar{x}, \bar{x}, \bar{x})$.

Imaginea 4.1 arată poza reconstruită cu 500 și 1000 de iterații.

O altă imagine de test, foarte populară și des folosită în procesarea imaginilor este *Lena*, cu dimensiunile 256×256 . Poza este supusă aceluși procedeu de neclaritate, descrisă în cazul precedent. Imaginea 4.2 afișează imaginile reconstruite cu 500 și 1500 de iterații.

Un al treilea exemplu în procesarea imaginilor este imaginea de test *Barbara* cu dimensiunile 256×256 . Am rulat algoritmul nostru și pe această imagine de test pentru a arăta aplicabilitatea metodei pe diferite imagini. Imaginea 4.3 arată poza

(a) Imaginea neclară



(b) 500 de iterații



(c) 1000 de iterații



(d) Imaginea originală



Imaginea 4.1: Imaginea (a) este obținută după ce a fost înmulțită cu operatorul neclar și adăugat la zgomotul alb Gaussian, (b) prezintă secvența medie generată de Algoritmul 4.1.2 după 500 de iterații. Imaginea (c) este imaginea reconstruită după 1000 de iterații, (d) arată imaginea de test clară Cameraman cu dimensiunile 256×256 .

reconstruită în cazul iterațiilor 400 și 800.

Un instrument valoros pentru măsurarea calității acestor imagini este *raportul semnal-zgomot (ISNR)*, definit ca

$$\text{ISNR}(k) = 10 \log_{10} \left(\frac{\|x - b\|^2}{\|x - x_k\|^2} \right)$$



Imaginea 4.2: Imaginea (a) este obținută după ce a fost înmulțită cu operatorul neclar și adăugat la zgomotul alb Gaussian, (b) prezintă secvența medie generată de Algoritmul 4.1.2 după 500 de iterații. Imaginea (c) este imaginea reconstruită după 1500 de iterații, (d) arată imaginea de test clară Lena cu dimensiunile 256×256 .

unde x , b și x_k denotă imaginea originală, observată și estimată, respectiv, la iterația k . Imaginea 4.4 arată evoluția numărului ISNR în cazul celor trei poze prezentate.

(a) Imaginea neclară



(b) 400 de iterații



(c) 800 de iterații



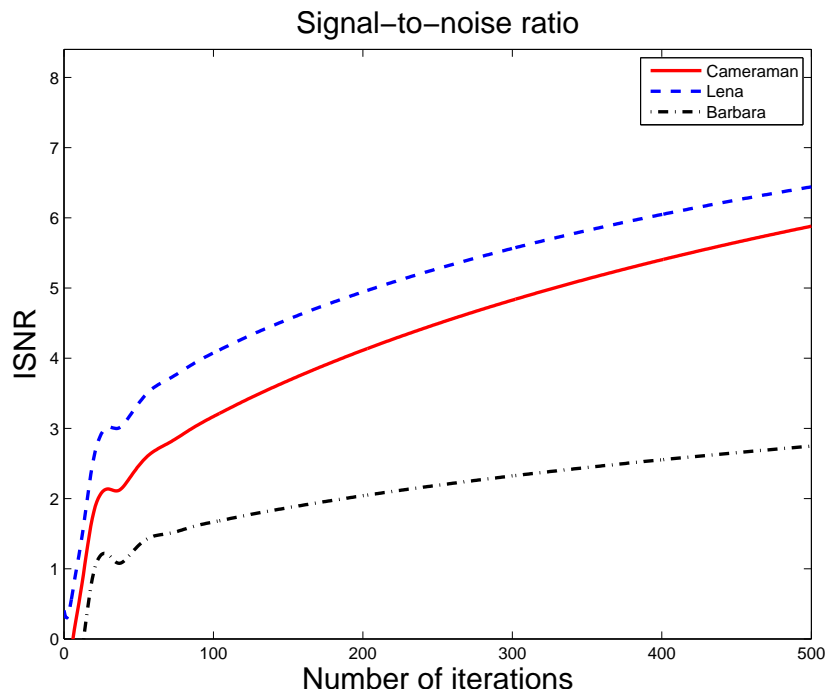
(d) Imaginea originală



Imaginea 4.3: Imaginea (a) este obținută după ce a fost înmulțită cu operatorul neclar și adăugat la zgomotul alb Gaussian, (b) prezintă secvența medie generată de Algoritmul 4.1.2 după 400 de iterații. Imaginea (c) este imaginea reconstruită după 800 de iterații, (d) arată imaginea de test clară Barbara cu dimensiunile 256×256 .

4.2 Probleme de localizare

Al doilea experiment numeric rezolvă problema locației cu mai multe facilități. La începutul secolului al 17-lea matematicianul Fermat era interesat de problemele de localizare cu o singură facilitate. Totuși, doar în secolul al 20-lea, erau adresate abordările normative pentru rezolvarea modelelor simbolice ale acestora în literatura de specialitate. Fiecare dintre aceste tehnici se referă la determinarea locului unei noi



Imaginea 4.4: Evoluția raportului semnal-zgomot (ISNR) în cazul imaginilor de test Cameraman, Lena și Barbara.

facilități, sau chiar mai multe facilități, cu privire la amplasarea între facilitățile existente, astfel încât să reducă la minimum funcția de cost bazată pe distanța ponderată dintre facilități [17, 18, 49, 51, 52, 64–66].

Dacă studiem lista de referințe ale lucrărilor realizate în ultimul deceniu care implică problemele de localizare cu facilități, este evident că există un interes interdisciplinar puternic în domeniile matematicii aplicate, cercetării operaționale, ingineriei civile, arhitecturii, științei managementului, ingineriei sistemelor, logisticii, economiei, științei regionale, sistemelor de transport, designului urban și ingineriei industriale, printre altele. Ca urmare, termenul ”facilitate” a căpătat o conotație foarte largă, pentru a se potrivi cu aplicații din fiecare dintre aceste domenii. De exemplu, o facilitate poate fi un spital, o ambulanță, un aeroport, o echipă de poliție, o școală, un terminal de computer, o comunitate planificată, un depozit, o clădire de birouri, o pompă într-o conductă, și așa mai departe [31, 33, 37]. Francis și Goldstein [34] oferă o bibliografie largă și recentă a problemei de localizare din literatura de specialitate. Una dintre clasificările cele mai complete ale acestor probleme este prevăzută într-o carte de Francis și White [35].

4.2.1 Prezentarea problemei

Continuăm cu prezentarea unei proceduri eficiente pentru rezolvarea unei probleme de localizare cu mai multe facilități, care pot fi formulate matematic după cum urmează:

$$\inf_{x_1, x_2} \left\{ \sum_{i=1}^k \lambda_i \|x_1 - c_i\|_2 + \sum_{i=1}^k \gamma_i \|x_2 - c_i\|_2 + \alpha \|x_1 - x_2\| \right\}, \quad (4.2.1)$$

unde folosim notațiile următoare

x_j =vector de locații pentru noile facilități $j, 1 \leq j \leq 2$;

c_i =vector de locații pentru facilitățile existente $i, 1 \leq i \leq k$;

λ_i =greutate pozitivă între o facilitate nouă x_1 și una existentă i ,

$$1 \leq i \leq k;$$

γ_i =greutate pozitivă între o facilitate nouă x_2 și una existentă i ,

$$1 \leq i \leq k;$$

α =greutate pozitivă între două facilități noi x_1 și x_2 .

Problema de localizare cu mai multe facilități 4.2.1 discutată în acest capitol este de fapt un caz special al Problemei 3.3.1 cu $(m, p) := (2, 2k + 1)$, $f_i = 0$, $h_i = \delta_{\{0\}}$, $K_i = 0$, unde $i = 1, 2$, și

$$g_j(x) = \begin{cases} \lambda_j \|x - c_j\|_2, & \text{dacă } j = 1, \dots, k; \\ \gamma_j \|x - c_j\|_2, & \text{dacă } j = k + 1, \dots, 2k; \\ \alpha \|x\|, & \text{dacă } j = 2k + 1, \end{cases}$$

$$L_{j1} = \begin{cases} \text{Id}, & \text{dacă } j = 1, \dots, k; \\ 0, & \text{dacă } j = k + 1, \dots, 2k; \\ \text{Id}, & \text{dacă } j = 2k + 1, \end{cases} \quad \text{și} \quad L_{j2} = \begin{cases} 0, & \text{dacă } j = 1, \dots, k; \\ \text{Id}, & \text{dacă } j = k + 1, \dots, 2k; \\ -\text{Id}, & \text{dacă } j = 2k + 1. \end{cases}$$

Algoritmul 4.2.1.

Prin urmare, schema iterativă din Algoritmul 3.3.1 devine

$$\forall n \geq 0 \left\{ \begin{array}{l} \text{For } i = 1, 2 \\ \left[\begin{array}{l} y_{1,i,n} = x_{1,i,n} - \gamma_n(x_{2,i,n} + \sum_{j=1}^{2k+1} L_{ji}^* v_{1,j,n}) \\ y_{2,i,n} = x_{2,i,n} + \gamma_n x_{1,i,n} \end{array} \right. \\ \text{For } j = 1, \dots, 2k + 1 \\ \left[\begin{array}{l} w_{1,j,n} = v_{1,j,n} - \gamma_n(v_{2,j,n} - L_{j1}x_{1,1,n} - L_{j2}x_{1,2,n}) \\ w_{2,j,n} = v_{2,j,n} + \gamma_n v_{1,j,n} \\ r_{2,j,n} = \text{prox}_{\gamma_n g_j} w_{2,j,n} \end{array} \right. \\ \text{For } i = 1, 2 \\ \left[\begin{array}{l} x_{1,i,n+1} = x_{1,i,n} - \gamma_n(\sum_{j=1}^{2k+1} L_{ji}^* w_{1,j,n}) \\ x_{2,i,n+1} = x_{2,i,n} - y_{2,i,n} + \gamma_n y_{1,i,n} \end{array} \right. \\ \text{For } j = 1, \dots, 2k + 1 \\ \left[\begin{array}{l} v_{1,j,n+1} = v_{1,i,n} - \gamma_n(r_{2,j,n} - L_{j1}y_{1,1,n} - L_{j2}y_{1,2,n}) \\ v_{2,j,n+1} = \gamma_n(w_{1,j,n} - v_{1,j,n}) + r_{2,j,n}. \end{array} \right. \end{array} \right.$$

Algoritmul implementat în limbajul de programare MatLab lucrează cu parametrul $\beta = 2 * \sqrt{k + 1} + 1$. Alegem ε astfel încât să fie o valoare cât de mică posibilă din intervalul $]0, 1/(\beta + 1)[$ și alegem valoarea parametrului γ egală cu $(1 - \varepsilon)/\beta$, pentru o soluție optimă.

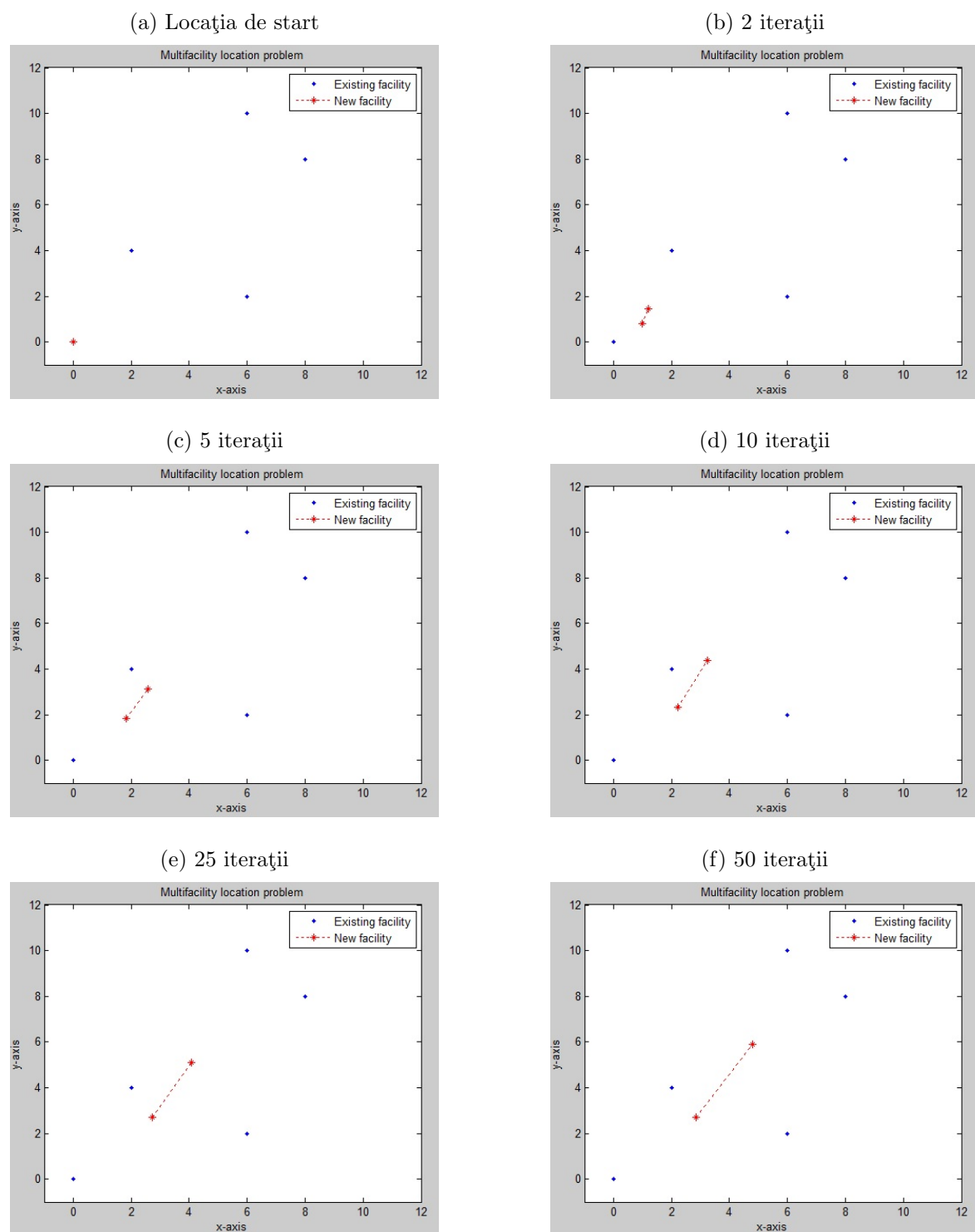
4.2.2 Experiență computațională

Ca o ilustrare a Algoritmului 4.2.1, considerăm problema de localizare, care implică două facilități noi și cinci facilități existente. Acest exemplu este folosit și în procedura de aproximare hiperboloidă prezentată în [31]. Presupunem că facilitățile existente se află în punctele de coordonate:

$$P_1 = (0, 0); P_2 = (2, 4); P_3 = (6, 2); P_4 = (6, 10) \text{ și } P_5 = (8, 8).$$

De asemenea, fie $\lambda_i = (4, 2, 3, 0, 0)$, $\gamma_i = (0, 2, 1, 3, 2)$, $i = 1, \dots, 5$ și $\alpha = 2$. Rezultatul algoritmului este ilustrat în Imaginea 4.5, având locațiile inițiale de $x_1^0 = (0, 0)$ și $x_2^0 = (0, 0)$.

Cele mai recente tehnici folosite pentru probleme de localizare, inclusiv Algoritmul 4.2.1, definesc un criteriu de oprire bazată pe schimbarea succesivă a valorilor funcției obiective, care este mult mai eficientă decât cea bazată pe schimbarea succesivă a locației facilităților noi (soluție de convergență). Technica de proiectare folosită în metoda propusă se folosește de toate avantajele structurii problemei de localizare și poate fi ușor extinsă pentru a rezolva probleme care implică mișcări de unități.



Imagina 4.5: O problemă de localizare cu mai multe facilități, implicând două facilități noi și cinci facilități existente. Facilitățile existente au punctele de coordonate $P_1 = (0,0)$, $P_2 = (2,4)$, $P_3 = (6,2)$, $P_4 = (6,10)$ și $P_5 = (8,8)$. Greutatea pozitivă α dintre cele două facilități noi este egală cu 2 și greutățile dintre facilitățile noi și cele existente sunt egale cu $\lambda_i = (4,2,3,0,0)$ și $\gamma_i = (0,2,1,3,2)$, $i = 1, \dots, 5$.

4.3 Consens în media rețelelor colorate

Al treilea experiment numeric se preocupă cu problema consensului în media rețelelor colorate. Având o rețea, în cazul în care fiecare nod posedua o valoare în formă de număr real, problema consensului în media rețelelor colorate constua în calcularea mediei acestor valori într-un mod recursiv și distribuit, permițând nodurilor să comunice, informațiile circulând numai pe muchii disponibile în rețea.

4.3.1 Algoritmul propus

Considerăm o rețea conectată $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, unde \mathcal{V} reprezintă un set de *noduri* și \mathcal{E} reprezintă un set de *muchii*. Fiecare muchie este reprezentat ca o pereche de noduri (i, j) , unde $i < j$. Nodurile i și j pot comunica valorile lor, dacă sunt vecini, cu alte cuvinte, dacă $(i, j) \in \mathcal{E}$. Setul de vecini ai nodului k este reprezentat cu \mathcal{N}_k și gradul nodului cu $D_k = |\mathcal{N}_k|$. Presupunem că fiecare nod are o valoare sub forma unui număr real, de asemenea numită *culoare*, și nici un nod vecin nu poate să aibă aceeași culoare. Fie C numărul de culori cu care este colorată rețeaua și \mathcal{C}_i setul de noduri care au culoarea i , $i = 1, \dots, C$. Fără a afecta generalitatea problemei, presupunem că primele C_1 noduri sunt în setul \mathcal{C}_1 , următoarele C_2 noduri sunt în setul \mathcal{C}_2 , etc. Mai mult, presupunem că o schemă de colorat este disponibilă. Pentru mai multe detalii cu privire la modelarea matematică a problemei de consens în media rețelelor colorate, facem referință la [46, 47].

Fie P și E numărul de noduri și muchii din rețea, respectiv. Prin urmare, $\sum_{i=1}^C C_i = P$. Denotăm cu θ_k valoarea atribuită nodului k , $k = 1, \dots, P$, problema pe care dorim să o rezolvăm este

$$\min_{x \in \mathbb{R}} \left\{ \sum_{k=1}^P \frac{1}{2} (x - \theta_k)^2 \right\}. \quad (4.3.1)$$

Soluția optimală unică la problema (4.3.1) este $\theta^* = \frac{1}{P} \sum_{k=1}^P \theta_k$, adică media valorilor atribuite setului de noduri din rețea. Scopul este de a face această valoare medie valabilă în fiecare nod într-un mod distribuit și recursiv. În acest scop, vom reproduce copii ale lui x în întreaga rețea, mai exact, pentru $k = 1, \dots, P$, nodul k va avea copia cu numărul

k , denotat cu x_k , care v-a fi actualizat iterativ în timpul algoritmului. La sfârșit, trebuie să garantăm că toate copiile sunt egale și exprimăm constrângerea, solicitând ca $x_i = x_j$ pentru fiecare $(i, j) \in \mathcal{E}$. Acest fapt dă naștere următoarei probleme de optimizare

$$\min_{\substack{\bar{x}=(x_1, \dots, x_P) \in \mathbb{R}^P \\ x_i=x_j, \forall \{i,j\} \in \mathcal{E}}} \left\{ \sum_{k=1}^P \frac{1}{2} (x_k - \theta_k)^2 \right\}. \quad (4.3.2)$$

Fie $A \in \mathbb{R}^{P \times E}$ matricea de incidentă nod-muchie a rețelei, având fiecare coloană asociată cu o muchie în următorul mod: coloana asociată muchiei $(i, j) \in \mathcal{E}$ are valoarea 1 în rândul i și -1 în rândul j , celelalte rânduri fiind egale cu zero. În consecință, constrângerile din (4.3.2) pot fi scrise cu ajutorul transpusei matricei de incidentă nod-muchie ca $A^T \bar{x} = 0$. Având în vedere ordinea de noduri și schema de colorare, putem scrie $A^T \bar{x} = A_1^T \bar{x}_1 + \dots + A_C^T \bar{x}_C$, unde $\bar{x}_i \in \mathbb{R}^{C_i}$, $i = 1, \dots, C$, colectează copiile nodurilor în C_i , i.e.

$$\bar{x} = \left(\underbrace{x_1, \dots, x_{C_1}}_{\bar{x}_1}, \dots, \underbrace{x_{P-C_C+1}, \dots, x_P}_{\bar{x}_C} \right).$$

Prin urmare, problema de optimizare (4.3.2) devine

$$\min_{\substack{\bar{x}=(\bar{x}_1, \dots, \bar{x}_C) \\ A_1^T \bar{x}_1 + \dots + A_C^T \bar{x}_C = 0}} \left\{ \sum_{i=1}^C f_i(\bar{x}_i) \right\}, \quad (4.3.3)$$

unde, pentru $i = 1, \dots, C$, funcția $f_i : \mathbb{R}^{C_i} \rightarrow \mathbb{R}$ este definită ca $f_i(\bar{x}_i) = \sum_{l \in C_i} \frac{1}{2} (x_l - \theta_l)^2$.

Observăm cu ușurință că problema (4.3.3) este un caz particular a problemei de optimizare (3.3.1), atunci când

$$m = C, p = 1, h_i = \delta_{\{0\}}, K_i = \text{Id}, L_{1i} = A_i^T \in \mathbb{R}^{E \times C_i}, i = 1, \dots, C, \text{ și } g_1 = \delta_{\{0\}}.$$

Algoritmul 4.3.1.

Folosind faptul că $h_i^* = 0, i = 1, \dots, C$, și $\text{Prox}_{\gamma g}(x) = 0$ pentru $\gamma > 0$ și $x \in \mathbb{R}^E$,

schema iterativă din Algoritmul 3.3.1 în cazul fără erori, devine:

$$\forall n \geq 0 \left[\begin{array}{l} \text{For } i = 1, \dots, C \\ \left[\begin{array}{l} y_{1,i,n} = x_{1,i,n} - \gamma_n(x_{2,i,n} + A_i v_{1,1,n}) \\ y_{2,i,n} = x_{2,i,n} + \gamma_n x_{1,i,n} \\ p_{2,i,n} = \text{Prox}_{\gamma_n f_i^*} y_{2,i,n} \end{array} \right. \\ w_{1,1,n} = v_{1,1,n} - \gamma_n(v_{2,1,n} - \sum_{i=1}^C A_i^T x_{1,i,n}) \\ \text{For } i = 1, \dots, C \\ \left[\begin{array}{l} q_{1,i,n} = y_{1,i,n} - \gamma_n(p_{2,i,n} + A_i w_{1,1,n}) \\ q_{2,i,n} = p_{2,i,n} + \gamma_n y_{1,i,n} \\ x_{1,i,n+1} = x_{1,i,n} - y_{1,i,n} + q_{1,i,n} \\ x_{2,i,n+1} = x_{2,i,n} - y_{2,i,n} + q_{2,i,n} \end{array} \right. \\ v_{1,1,n+1} = v_{1,1,n} + \gamma_n \sum_{i=1}^C A_i^T y_{1,i,n} \\ v_{2,1,n+1} = \gamma_n^2 \left(\sum_{i=1}^C A_i^T x_{1,i,n} - v_{2,1,n} \right). \end{array} \right.$$

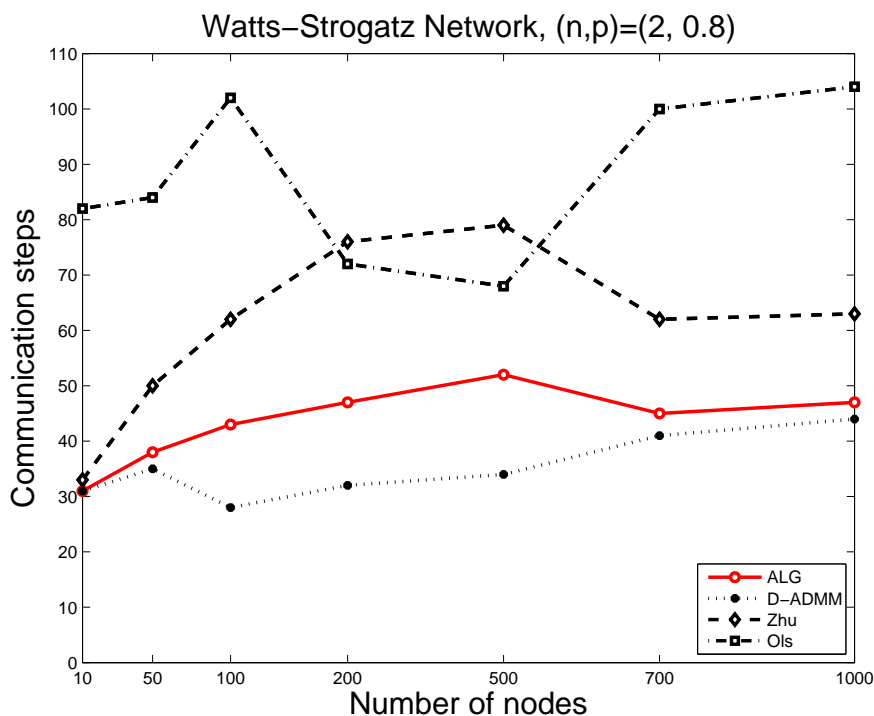
Observăm că $i = 1, \dots, C$ și $\gamma > 0$ presupune $\text{Prox}_{\gamma f_i^*}(\bar{x}_i) = (1 + \gamma)^{-1}(\bar{x}_i - \gamma \bar{\theta}_i)$, unde $\bar{\theta}_i$ este un vector în \mathbb{R}^{C_i} al cărui componenți sunt θ_l , cu $l \in C_i$.

4.3.2 Evaluarea performanței

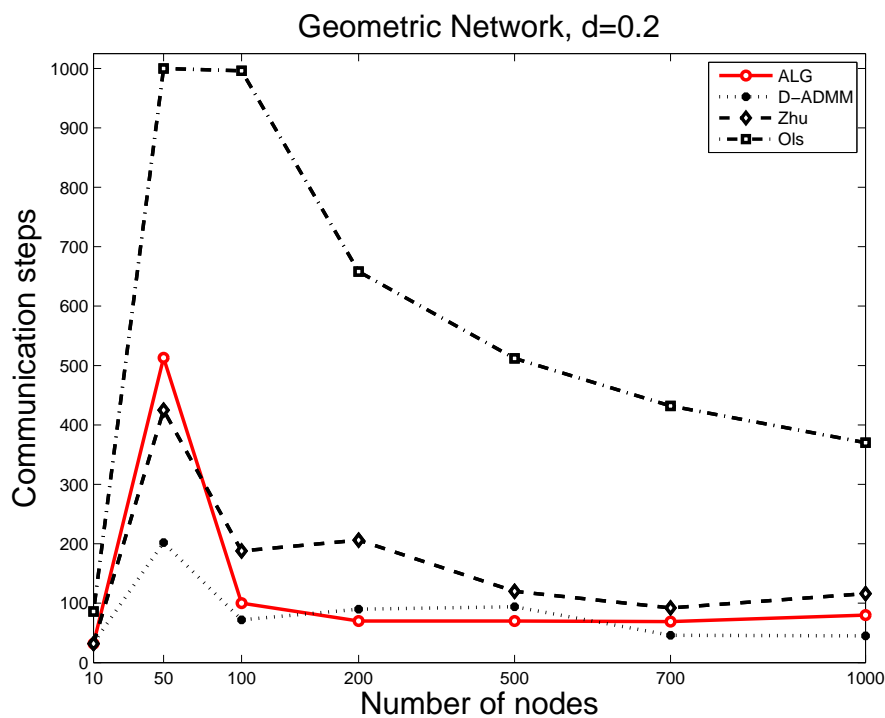
În scopul de a compara performanțele metodei noastre cu alte algoritmi deja existente în literatura de specialitate, am folosit rețelele generate în [47] cu un număr de noduri cuprinse între 10 și 1000. Valoarea θ_k , asociată fiecărui nod, a fost generată aleator și independent cu o distribuție normală cu abaterea medie de 10 și 100. Am lucrat cu rețele având 10, 50, 100, 200, 500, 700 și 1000 de noduri și am măsurat performanța algoritmului nostru din punct de vedere al numărului de comunicare, care coincide de fapt cu numărul de iterații. Ca și criteriu de oprire am considerat

$$\frac{\|x_n - \mathbb{1}_P \theta^*\|}{|\sqrt{P} \theta^*|} \leq 10^{-4},$$

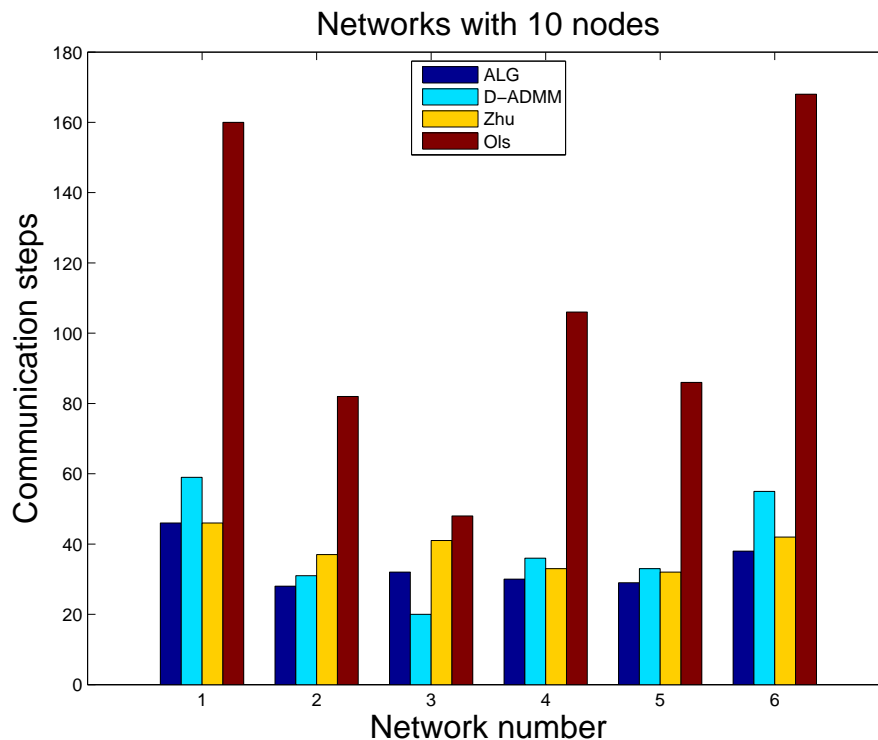
unde $\mathbb{1}_P$ denotă vectorul din \mathbb{R}^P care are toate valorile egale cu 1.



Imaginea 4.6: Graficul arată pașii de comunicare necesari, de către patru algoritmi, în cazul rețelei Wass-Strogatz, cu un număr diferit de noduri. ALG reprezintă algoritmul primal-dual propus în această lucrare.



Imaginea 4.7: Graficul arată pașii de comunicare necesari, de către patru algoritmi, în cazul rețelei Geometrice, cu un număr diferit de noduri. ALG reprezintă algoritmul primal-dual propus în această lucrare.



Imaginea 4.8: Compararea celor patru algoritmi în cazul a șase rețele cu 10 noduri. Aici, ALG reprezintă algoritmul primal-dual propus.

În Imaginea 4.6 și Imaginea 4.7 prezentăm numărul de comunicații necesare pentru rețeaua *Watts-Strogatz* cu parametrii $(2,0.8)$ și pentru rețeaua *Geometrică* cu parametrul 0.2 . Așa cum arată Imaginea 4.6 și Imaginea 4.7, algoritmul nostru are o performanță comparabilă cu D-AMM, prezentat în [47], dar s-a comportat mult mai bine decât algoritmi prezentați în [48] și [70].

În scopul de a observa comportamentul algoritmului pe diferite rețele, l-am testat pe 6 modele, prezentate în Tabelul 4.1, cu un număr diferit de noduri. Observând pașii de comunicare necesare, putem conchide că algoritmul nostru este eficientă în ceea ce privește comunicarea nodurilor și are o performanță mai bună sau similară decât algoritmi din [47], [48] și [70] (exemplificat în Imaginea 4.8).

Număr	Model	Parametrii
1	Erdős-Rényi	0.25
2	Watts-Strogatz	(2, 0.8)
3	Watts-Strogatz	(4, 0.6)
4	Barabasi-Albert	—
5	Geometric	0.2
6	Lattice	—

Tabel 4.1: Parametrii rețelelor.

4.4 Clasificare cu ajutorul mașinilor de vector-suport

Al patrulea experiment numeric prezentat în acest capitol este clasificarea imaginilor cu ajutorul mașinilor de vector-suport.

Având un set de date de formare $a_i \in \mathbb{R}^n$, $i = 1, \dots, k$, aparținând uneia dintre cele două clase de date, notată cu ”-1” și ”+1”, scopul este de a construi o funcție de decizie dată sub forma unui hiperplan de separare, care trebuie să atribuie fiecare dată nouă în unul dintre cele două clase, cu o rată de clasificare eronată scăzută.

4.4.1 Hiperplanul de separare

Construim o matrice $A \in \mathbb{R}^{k \times n}$ astfel încât fiecare rând să corespundă unui punct de date a_i , $i = 1, \dots, k$ și un vector $d \in \mathbb{R}^k$ astfel încât, pentru $i = 1, \dots, k$, valoarea de pe locul i să fie egală cu -1 , dacă a_i aparține clasei ”-1” și să fie egală cu ”1”, altfel. Se consideră cazul în care datele de formare nu pot fi separat fără erori. În acest caz, se poate separa setul de date cu un număr minim de erori. Pentru a acoperi situația în care separarea nu poate fi făcută exact, considerăm variabilele pozitive $\xi_i \geq 0$, $i = 1, \dots, k$. Astfel obiectivul va fi de a găsi $(s, r, \xi) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}_+^k$ ca soluție optimă pentru problema următoare de optimizare

$$\min_{\substack{(s,r,\xi) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}_+^k \\ D(As+1_k r) + \xi \geq 1_k}} \{ \|s\|^2 + C \|\xi\|^2 \}, \quad (4.4.1)$$

unde $\mathbb{1}_k$ denotă vectorul din \mathbb{R}^k care are toate valorile egale cu 1, inegalitatea $z \geq \mathbb{1}_k$ pentru $z \in \mathbb{R}^k$ reprezintă $z_i \geq 1, i = 1, \dots, k$, $D = \text{Diag}(d)$ este matricea diagonală având ca diagonală vectorul d și C este un parametru de compromis. Fiecare dată nouă $a \in \mathbb{R}^n$ va fi distribuit în una dintre clasele cu ajutorul funcției de decizie $z(a) = s^T a + r$, adică, a v-a fi distribuit în clasa “-1”, dacă $z(a) < 0$, și în clasa “+1”, altfel. Pentru mai multe perspective teoretice în ceea ce privește mașinile de vector-suport, facem referință la [28, 32].

Problema clasificării cu ajutorul mașinilor de vector-suport (4.4.1) poate fi scris ca un caz special al problemei de optimizare (3.3.1), prin

$$m = 3, p = 1, f_1(\cdot) = \|\cdot\|^2, f_2 = 0, f_3(\cdot) = C \|\cdot\|^2 + \delta_{\mathbb{R}_+^k}(\cdot), h_i = \delta_{\{0\}}, K_i = \text{Id}, i = 1, 2, 3,$$

$$g_1 = \delta_{\{z \in \mathbb{R}^k : z \geq \mathbb{1}_k\}}, L_{11} = DA, L_{12} = D\mathbb{1}_k \text{ și } L_{13} = \text{Id}.$$

Algoritmul 4.4.1.

Astfel, Algoritmul 3.3.1 dă naștere, în cazul fără erori, următoarei scheme iterative:

$$\forall n \geq 0 \left\{ \begin{array}{l} \text{For } i = 1, 2, 3 \\ \left\{ \begin{array}{l} y_{1,i,n} = x_{1,i,n} - \gamma_n(x_{2,i,n} + L_{1i}^T v_{1,1,n}) \\ y_{2,i,n} = x_{2,i,n} + \gamma_n x_{1,i,n} \\ p_{2,i,n} = \text{Prox}_{\gamma_n f_i^*} y_{2,i,n} \end{array} \right. \\ w_{1,1,n} = v_{1,1,n} - \gamma_n(v_{2,1,n} - \sum_{i=1}^3 L_{1i} x_{1,i,n}) \\ w_{2,1,n} = v_{2,1,n} + \gamma_n v_{1,1,n} \\ r_{2,1,n} = \text{Prox}_{\gamma_n g_1} w_{2,1,n} \\ \text{For } i = 1, 2, 3 \\ \left\{ \begin{array}{l} q_{1,i,n} = y_{1,i,n} - \gamma_n(p_{2,i,n} + L_{1i}^T w_{1,1,n}) \\ q_{2,i,n} = p_{2,i,n} + \gamma_n y_{1,i,n} \\ x_{1,i,n+1} = x_{1,i,n} - y_{1,i,n} + q_{1,i,n} \\ x_{2,i,n+1} = x_{2,i,n} - y_{2,i,n} + q_{2,i,n} \end{array} \right. \\ s_{1,1,n} = w_{1,1,n} - \gamma_n(r_{2,1,n} - \sum_{i=1}^3 L_{1i} p_{1,i,n}) \\ s_{2,1,n} = r_{2,1,n} + \gamma_n w_{1,1,n} \\ v_{1,1,n+1} = v_{1,1,n} - w_{1,1,n} + s_{1,1,n} \\ v_{2,1,n+1} = v_{2,1,n} - w_{2,1,n} + s_{2,1,n} \end{array} \right.$$

Observăm, de asemenea, că pentru punctele proximale necesare în algoritm, avem $\gamma > 0$ și $(s, r, \xi, z) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}^k \times \mathbb{R}^k$ următoarea formulă exactă:

$$\text{Prox}_{\gamma f_1^*}(s) = (2 + \gamma)^{-1} 2s, \text{Prox}_{\gamma f_2^*}(r) = 0, \text{Prox}_{\gamma f_3^*}(\xi) = \xi - \gamma P_{\mathbb{R}_+^k}((2C + \gamma)^{-1} \xi)$$

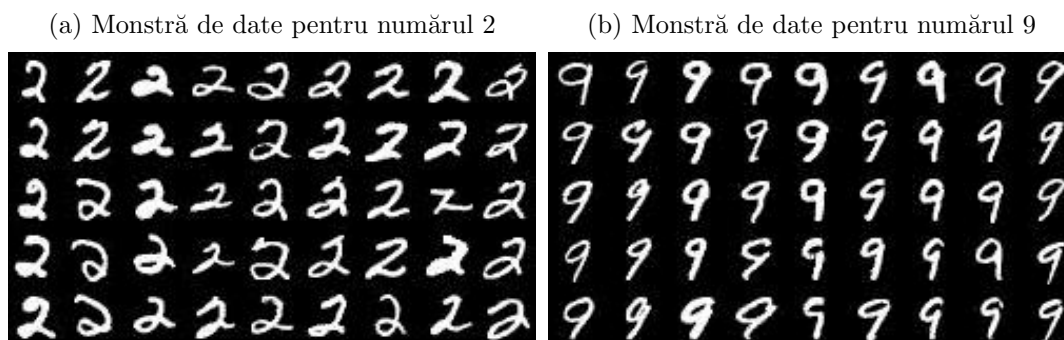
și

$$\text{Prox}_{\gamma g_1}(z) = P_{\{x \in \mathbb{R}^k : x_{\geq 1_k}\}}(z).$$

4.4.2 Experimente cu recunoașterea cifrelor

Am folosit o serie de date de 11907 imagini de formare și 2041 de imagini de testare de dimensiune 28×28 de la site-ul <http://www.cs.nyu.edu/~roweis/data>.

html. Problema a constat în determinarea funcției de decizie bazată pe un fond de cifre scrise cu mâna, arătând fie numărul doi sau numărul nouă, etichetate cu -1 și 1 , respectiv (a se vedea Imaginea 4.9). Am evaluat calitatea funcției de decizie pe un set de date de test prin calcularea procentajului de imagini clasificate greșit. Notăm că folosim doar o jumătate din imaginile disponibile din setul de date de antrenare, pentru a reduce efortul de calcul.



Imaginea 4.9: Monstră de imagini aparținând claselor -1 și $+1$, respectiv.

În ceea ce privește setul de date considerate, notăm cu $\mathcal{D} = \{(X_i, Y_i), i = 1, \dots, 6000\} \subseteq \mathbb{R}^{784} \times \{+1, -1\}$ setul de date de formare disponibile în 3000 de imagini din clasa -1 și 3000 de imagini din clasa 1 . O mostră din fiecare tip de imagini, este prezentată în Imaginea 4.9. Imaginile au fost vectorizate și normalizate prin împărțirea fiecăruia dintre ele de cantitatea $(\frac{1}{6000} \sum_{i=1}^{6000} \|X_i\|^2)^{\frac{1}{2}}$.

Număr de iterații	100	1000	2000	3000	5000
Eroarea datelor de formare	2.95	2.6	2.3	1.95	1.55
Eroarea datelor de test	2.95	2.55	2.45	2.15	2

Tabel 4.2: Rata de clasificare eronată în procentaj pentru numere diferite de iterații, atât pentru datele de formare cât și pentru datele de testare.

Am oprit algoritmul primal-dual după un număr diferit de iterații și am evaluat performanțele funcțiilor de decizie rezultate. În Tabelul 4.2 prezentăm rata de clasificări greșite în procentaj pentru datele de formare și pentru datele de test (eroarea datelor de formare este mai mică decât cel pentru datele de testare) și observăm că calitatea de clasificare crește cu numărul de iterații. Cu toate acestea, chiar și pentru un număr mic de iterații, rata clasificărilor greșite surclasează pe cele raportate în literatura de

specialitate care se ocupă cu metode numerice de clasificare de tip vector-suport. Să amintim, de asemenea, că rezultatele numerice sunt date pentru cazul $C = 1$. Am testat, de asemenea, alte opțiuni pentru C , cu toate acestea nu am observat un mare impact asupra rezultatelor.

Capitolul 5

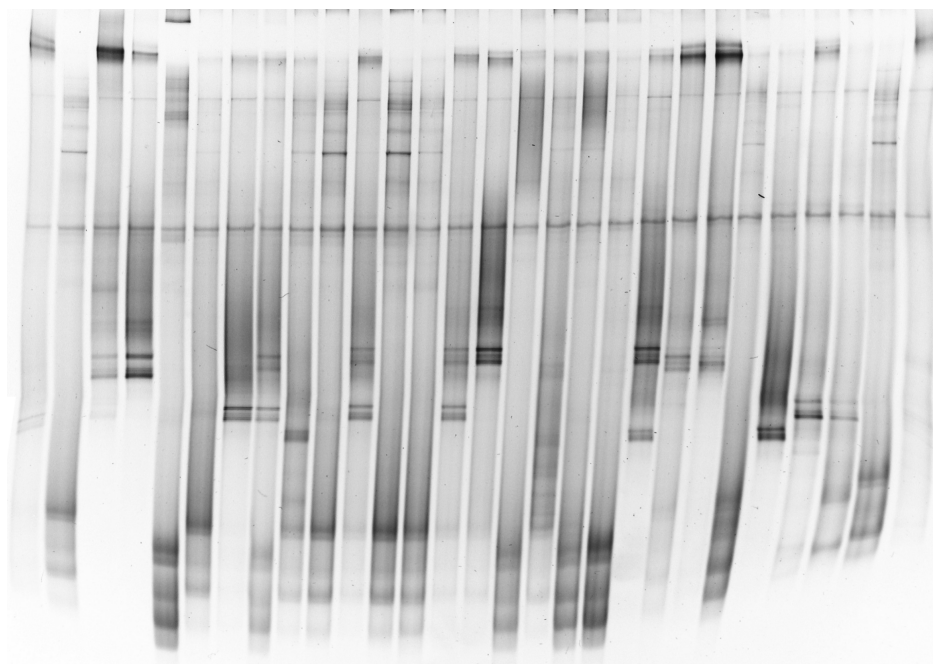
Aplicația interdisciplinară a algoritmului

În cursul studiilor mele, am colaborat cu un grup de cercetători în biologie, care studiau diversitatea picoalgelor în lacurile suarate din România. Am investigat comunitățile de fitoplancton din mai multe lacuri hipersaline din bazinul Transilvaniei, cu ajutorul celor mai noi tehnici de microscopie și biologie moleculară [38–40]. Scopul nostru a fost să adaptăm algoritmul primal-dual în procesarea imaginilor și recunoașterea modelelor, în cazul imaginilor microscopice și în cazul electrophoretogramelor. De-a lungul procesului de dezvoltare am lucrat cu imagini microscopice epifluorescente de picoalge, cu scopul de a le număra și distinge din zgomotul de fundal.

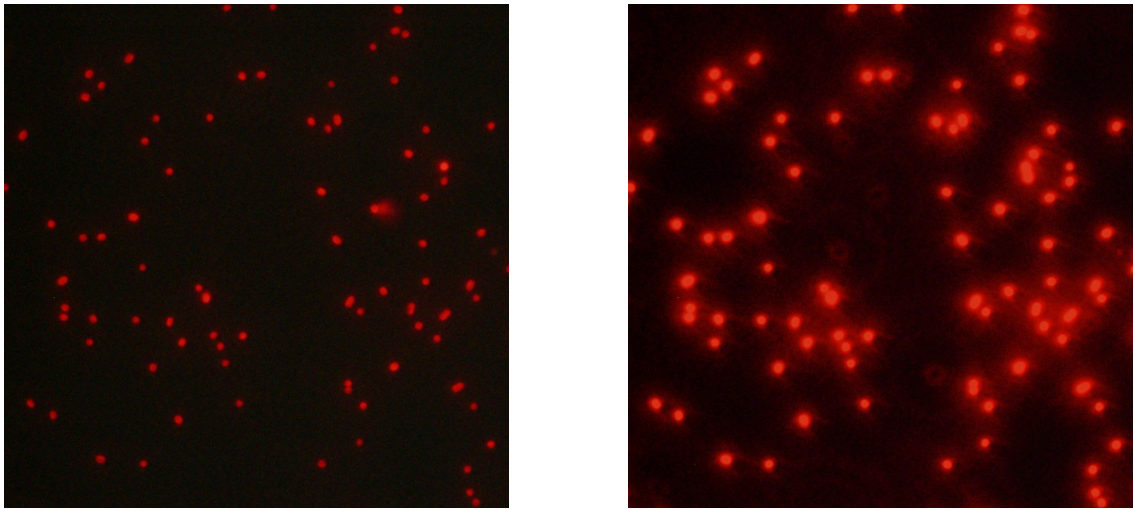
5.1 Optimizare în biologie

Cele mai multe tehnici moleculare duc la metoda de electrophoreză, în care, folosind electricitatea, ADN-ul sau proteinele pot fi separate într-un gel. În cazul nostru, fragmentele de ADN aparținând diferitelor organisme colectate din lacurile sărate au fost migrate, folosind gel de electrophoreză de tip agaros, cu scopul de a separa fragmentele de ADN. Agarosul este un material polizaharid polimer, de obicei extras din alge, și este folosit des în biologia moleculară, biochimia și chimia clinică, pentru separarea moleculelor mai mari, în special în cazul unei populații mixte de ADN, într-un matrix de agaros.

După obținerea modelelor de ADN migrate (Imaginea 5.1), acestea sunt analizate, și diferitele eșantioane sunt comparate pentru a stabili diferențele dintre lacurile sărate. Modelele pot fi folosite și pentru a stabili diferențele într-un lac la diferite adâncimi ale apei [38]. Procesarea moleculară, de exemplu clonarea și secvențierea, pot fi costisitoare și consumatoare de timp. Scopul nostru este de a optimiza algoritmul primal-dual astfel încât să găsească modelele specifice, pentru a face procesul descris mai rapid și mai rentabil. Modelele pot fi distinse cu metoda numită ”recunoașterea modelelor”. Acesta este de fapt o clasificare, care încearcă să atribuie pentru fiecare valoare de intrare o clasă specificată de la început. Procedura este una non-probabilistică binară liniară clasificatoare, foarte similară cu metoda de clasificare prin intermediul mașinilor cu vector-suport (a se vedea 4.4). În cazul nostru au fost luate imagini epifluorescente microscopice pentru a număra celulele cianobacteriale din monstra de apă, așa cum se vede în Imaginea 5.2.



Imaginea 5.1: Gel gradient denaturat de electrophoreză cu eșantioane ADN din lacurile sărate aflate în bazinul Transilvaniei. Coloanele reprezintă eșantioane de apă diferite și benzile orizontale din coloane reprezintă specii diferite.



Imaginea 5.2: Imagini microscopice epifluorescente (magnificare = x 1000) cu picocianobacterii. Prima imagine arată picocianobacteria cu emiterie albastru-violet, și în a doua imagine ele sunt prezente cu emiterie verde.

5.2 Technici de recunoașterea modelelor

Enumerarea conținutului de alge dintr-o monstră de apă, de obicei implică numărarea manuală cu ajutorul microscopului. Recentele software-uri avansate de recunoașterea modelelor, permit ca aceste numărări să fie automatizate. Technicile noi au potențialul de a avansa sensibilitatea sistemelor de monitorizare prin furnizarea mai multor informații privind starea apei, aproape în timp real, cu mai multă importanță statistică decât microscopia convențională. Disponibilitatea acestor informații pot genera considerabil reducerea costurilor, prin permiterea operatorilor de sistem să trateze proactiv apa.

În general, observate cu ochiul uman, diferitele specii de picoalge pot părea clar diferite, dar matematic nu sunt ușor de diferențiate. În urmă cu mai mulți ani, recunoașterea automată a modelelor în imagini, a fost un domeniu complex de cercetare în informatică, deoarece foarte multe diferențieri făcute cu ochiul și creierul uman, nu sunt ușor de realizat computațional.

Activitatea noastră în acest domeniu are progrese mari cu rezultate foarte bune.

În viitor, ne propunem să dezvoltăm tehnica de optimizare pentru recunoașterea modelelor, cu scopul de a elimina particulele neclasificate și de a obține rezultate de clasificare mai bune pentru imaginile microscopice.

Bibliografie

- [1] H. Attouch, L.M. Briceño-Arias și P.L. Combettes. A parallel splitting method for coupled monotone inclusions. *SIAM Journal on Control and Optimization*, 48(5):3246–3270, 2010.
- [2] H. Attouch și M. Théra. A general duality principle for the sum of two operators. *Journal of Convex Analysis*, 3:1–24, 1996.
- [3] A. Barabasi și R. Albert. Emergence of scalling in random networks. *Science*, 286:509–512, 1999.
- [4] H.H. Bauschke și P.L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. CMS Books in Mathematics, Springer-Verlag New York, 2011.
- [5] H.H. Bauschke, P.L. Combettes și S. Reich. The asymptotic behavior of the composition of two resolvents. *Nonlinear Analysis*, 60(2):283–301, 2005.
- [6] A. Beck și M. Teboulle. Smoothing and first order methods: a unified framework. *SIAM Journal on Optimization*, 22(2):557–580, 2012.
- [7] R.I. Boț. *Conjugate Duality in Convex Optimization*. Lecture Notes in Economics and Mathematical Systems, Vol. 637, Springer-Verlag Berlin Heidelberg, 2010.
- [8] R.I. Boț și E.R. Csetnek. Regularity conditions via generalized interiority notions in convex optimization: new achievements and their relation to some classical statements. *Optimization*, 61(1):35–65, 2012.
- [9] R.I. Boț, E.R. Csetnek și A. Heinrich. A primal-dual splitting algorithm for finding zeros of sums of maximally monotone operators. *To appear in Siam Journal on Optimization*, 2012.

- [10] R.I. Boţ, E.R. Csetnek și E. Nagy. Solving systems of monotone inclusions via primal-dual splitting techniques. *To appear in Taiwanese Journal of Mathematics*, 2013.
- [11] R.I. Boţ și C. Hendrich. A double smoothing technique for solving unconstrained nondifferentiable convex optimization problems. *Computational Optimization and Applications*, 54(2):239–262, 2013.
- [12] R.I. Boţ și C. Hendrich. A Douglas-Rachford type primal-dual method for solving inclusions with mixtures of composite and parallel-sum type monotone operators. 2012. [arXiv:1212.0326v1\[math.OC\]](#)
- [13] R.I. Boţ și C. Hendrich. A variable smoothing algorithm for solving convex optimization problems. *Optimization*, 2012. [arXiv:1207.3254\[math.OC\]](#)
- [14] R.I. Boţ și C. Hendrich. Convergence analysis for a primal-dual monotone + skew splitting algorithm with applications to total variation minimization. 2012. [arXiv:1211.1706v1\[math.OC\]](#)
- [15] R.I. Boţ și C. Hendrich. On the acceleration of the double smoothing technique for unconstrained convex optimization problems. *Optimization*, 2012. <http://dx.doi.org/10.1080/02331934.2012.745530>
- [16] L.M. Briceño-Arias și P.L. Combettes. A monotone + skew splitting model for composite monotone inclusions in duality. *SIAM Journal on Optimization*, 21(4):1230–1250, 2011.
- [17] A. V. Cabot, R.L. Francis și M.A. Stary. A Network Flow Solution to a Rectilinear Distance Facility Location Problem. *American Institute of Industrial Engineers Transactions*, 2:132–141, 1970.
- [18] P. Calamai și C. Charalambous. Solving multifacility location problems involving Euclidean distances. *Naval Research Logistics Quarterly*, 27(4):609–620, 1980.
- [19] A. Chambolle și T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
- [20] P.L. Combettes. Solving monotone inclusions via compositions of nonexpansive averaged operators. *Optimization*, 53(5–6):475–504, 2004.

- [21] P.L. Combettes. Iterative construction of the resolvent of a sum of maximal monotone operators. *Journal of Convex Analysis*, 16(3–4):727–748, 2009.
- [22] P.L. Combettes. Systems of Structured Monotone Inclusions: Duality, Algorithms, and Applications. 2012. [arXiv:1212.6631v1\[math.OA\]](#)
- [23] P.L. Combettes și J.-C. Pesquet. Primal-dual splitting algorithm for solving inclusions with mixtures of composite, Lipschitzian, and parallel-sum type monotone operators. *Set-Valued and Variational Analysis*, 20(2):307–330, 2012.
- [24] L. Cooper. Location-Allocation Problems. *Operations Research*, 11:331–344, 1963.
- [25] L. Cooper. Heuristic Methods for Location-Allocation Problems. *SIAM Review*, 6:37–52, 1964.
- [26] L. Cooper. Solutions of Generalized Locational Equilibrium Problems. *Journal of Regional Science*, 7:1–18, 1967.
- [27] C. Cortes și V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.
- [28] N. Cristianini și J.S. Taylor. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press, Cambridge, 2000.
- [29] I. Ekeland și R. Temam. *Convex Analysis and Variational Problems*. North-Holland Publishing Company, Amsterdam, 1976.
- [30] P. Erdős și A. Rényi. On random graphs. *Publicationes Mathematicae*, 6:290–297, 1959.
- [31] J. W. Eyster, J.A. White și W.W. Wierwille. On Solving Multifacility Location Problems using a Hyperboloid Approximation Procedure. *American Institute of Industrial Engineers Transactions*, 5(1):01–06, 1973.
- [32] P.A. Forero, A. Cano și G.B. Giannakis. Consensus-based distributed support vector machines. *Journal of Machine Learning Research*, 11:1663–1707, 2010.
- [33] R.L. Francis și A.V. Cabot. Properties of a Multifacility Location Problem Involving Euclidean distances. *Naval Research Logistics Quarterly*, 19:335–353, 1972.
- [34] R.L. Francis și J.M. Goldstein. Location Theory: A Selective Bibliography. *Operations Research*, 22:400–410, 1974.

- [35] R.L. Francis și J. A. White. *Facility Layout and Location: An Analytic Approach*. Prentice-Hall, Englewood Cliffs, New Jersey 1974.
- [36] S.L.S. Jacoby, J.S. Kowalik și J.T. Pizzo. *Iterative Methods for Nonlinear Optimization Problems*. Prentice-Hall, Inc., 1972.
- [37] H. Juel și R.F. Love. An Efficient Computational Procedure for Solving the Multifacility Rectilinear Facilities Location Problem. *Operational Research Quarterly*, 27:697–703, 1976.
- [38] Zs.Gy. Keresztes, T. Felföldi, B. Somogyi, Gy. Székely, N. Dragoș, K. Márialigeti, Cs. Bartha și L. Vörös. First record of picophytoplankton diversity in Central European hypersaline lakes. *Extremophiles*, 16:759–769, 2012.
- [39] Zs.Gy. Keresztes, B. Somogyi, E. Boros, Gy. Székely, Cs. Bartha, **E. Nagy**, N. Dragoș și L. Vörös. Picoplankton in soda lakes of the Carpathian Basin. *Contrib. Bot.*, XLV:41–46, 2010.
- [40] Zs.Gy. Keresztes, **E. Nagy**, B. Somogyi, B. Németh, Cs. Bartha, Gy. Székely, N. Dragoș și L. Vörös. Trophic conditions of saline lakes in the Transylvanian-Basin. *Hung. Hydrobiol. Soc.*, 91(6):46–48, 2011.
- [41] H.W. Kuhn. A Note on Fermat’s Problem. *Mathematical Programming*, 4:98–107, 1973.
- [42] H.W. Kuhn și R.E. Kuenne. An Efficient Algorithm for the Numerical Solution of the Generalized Weber Problem in Spatial Economics. *Journal of Regional Science*, 4:21–33, 1962.
- [43] P.L. Lions și B. Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16(6):964–979, 1979.
- [44] R.F. Love. Locating Facilities in Three-Dimensional Space by Convex Programming. *Naval Research Logistics Quarterly*, 16:503–516, 1969.
- [45] W. Miehle. Link-Length Minimization in Networks. *Operations Research*, 6:232–243, 1958.
- [46] J.F.C. Mota, J.M.F. Xavier, P.M.Q. Aguiar și M. Püschel. ADMM for consensus on colored networks. *2012 IEEE 51st Annual Conference on Decision and Control (CDC)*, 5116–5121, 2012.

- [47] J.F.C. Mota, J.M.F. Xavier, P.M.Q. Aguiar și M. Püschel. D-ADMM: A communication-efficient distributed algorithm for separable optimization. 2012. [arXiv:1202.2805\[math.OC\]](#)
- [48] A. Olshevsky și J. Tsitsiklis. Convergence speed in distributed consensus and averaging. *SIAM Journal on Control and Optimization*, 48(1):33–55, 2009.
- [49] L.M. Ostresh. The Multifacility Location Problem: Applications and Descent Theorems. *Journal of Regional Science*, 17:409–419, 1977.
- [50] M. Penrose. *Random Geometric Graphs*. Oxford University Press, 2004.
- [51] A.A.B. Pritsker și P.M. Ghare. Locating New Facilities with Respect to Existing Facilities. *American Institute of Industrial Engineers Transactions*, 2(4):290–297, 1970.
- [52] M.R. Rao. *The Rectilinear Facilities Location Problem*. Working Paper Series No. F7215, System Analysis Program, The Graduate School of Management, University of Rochester, New York, 1972.
- [53] R.T. Rockafellar. Extension of Fenchel’s duality theorem for convex functions. *Duke Mathematical Journal*, 33(1):81–89, 1966.
- [54] R.T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, 1970.
- [55] R.T. Rockafellar. On the maximality of sums of nonlinear monotone operators. *Trans. Amer. Math. Soc.*, 149:75–88, 1970.
- [56] R.T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM Journal on Control and Optimization*, 14(5):877–898, 1976.
- [57] T. Strömberg. *The Operation of Infimal Convolution*. Dissertationes Math. (Rozprawy Math.), 352:58, 1996.
- [58] S. Simons. *From Hahn-Banach to monotonicity*. Springer-Verlag, Berlin, 2008.
- [59] P. Tseng. A modified forward-backward splitting method for maximal monotone mappings. *SIAM Journal on Control and Optimization*, 38(2):431–446, 2000.
- [60] R.C. Vergin și J.D. Rogers. An Algorithm and Computational Procedure for Locating Economic Activities. *Management Science*, 13:240–254, 1967.
- [61] B.C. Vũ. A splitting algorithm for dual monotone inclusions involving cocoercive operators. *Advances in Computational Mathematics*, 38(3):667–681, 2013.

- [62] D. Watts și S. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):409–419, 1998.
- [63] E. Weiszfeld. Sur le Point pour Lequel la Somme des Distances de n Points Donnes Est Minimum. *Tohoku Mathematical Journal*, 43:355–386, 1936.
- [64] G.O. Wesolowsky și R.F. Love. The Optimal Location of New Facilities Using Rectangular Distances. *Operations Research*, 19:355–386, 1971.
- [65] G.O. Wesolowsky și R.F. Love. A Nonlinear Approximation Method for Solving a Generalized Rectangular Distance Weber Problem. *Management Science*, 18(11):656–663, 1972.
- [66] J.A. White. *Facilities Layout and Location: an OR/MS/IE Interface*. Joint ORSA, TIMS, AIIE National Meeting, Atlantic City, New Jersey, November 1972.
- [67] D.J. Wilde și C.S. Beightler. *Foundations of Optimization*. Prentice-Hall, Inc., 1967.
- [68] C. Zălinescu. A comparison of constraint qualifications in infinite-dimensional convex programming revisited. *Journal of Australian Mathematical Society Series B*, 40(3):353–378, 1999.
- [69] C. Zălinescu. *Convex Analysis in General Vector Spaces*. World Scientific, Singapore, 2002.
- [70] H. Zhu, G. Giannakis și A.Cano. Distributed in-network channel decoding. *IEEE Transactions on Signal Processing*, 57(10):3970–3983, 2009.