

CALCULUL GRADIENTULUI FUNCȚIEI DE PENALIZARE ÎN PROBLEMA MAX-CUT

V. Moraru

Universitatea Tehnică a Moldovei

INTRODUCERE

Problema determinării tăieturii maxime într-un graf constă în împărțirea vârfurilor în două mulțimi de vârfuri care se leagă între ele cu muchii și suma ponderilor muchiilor este maximă. Această problemă, numită și **problema MAX-CUT**, are diverse aplicații în practică (vezi, de ex.[1]). Problema MAX-CUT face parte în caz general din clasa problemelor *NP*-complete [2] și poate fi rezolvată într-un timp polinomial numai în cazuri speciale, de exemplu, când graful este planar (vezi [3]). O tehnică actuală de rezolvare constă în transformarea problemei originale, dată în spațiul vectorial \mathfrak{R}^n , într-o problemă de programare liniară în spațiul matricelor simetrice semidefinite de dimensiune $n \times n$. În lucrare problema determinării tăieturii maxime se formulează ca o problemă de programare pătratică cu restricții pătratice. Se prezintă relaxarea bazată pe programarea semidefinită. Se discută metoda punctului interior și se propune o procedură efektivă de calcul al gradientului funcției de penalizare.

1. FORMULAREA PROBLEMEI

Considerăm $G(V, U)$ un graf finit, ponderat și neorientat unde $V = (v_1, v_2, \dots, v_n)$ este mulțimea vârfurilor (nodurilor). Vom presupune în această lucrare că mulțimea muchiilor U este o mulțime care nu conține bucle și muchii multiple. Fiecărei muchii $(v_i, v_j) \in U$ i se asociază o valoare (ponderea) $a_{ij} = a_{ji} \geq 0$. Vom defini $a_{ij} = 0$ în cazul în care $(v_i, v_j) \notin U$, adică în cazul când v_i și v_j nu sunt adiacente. Matricea de adiacență asociată acestui graf va fi notată $A = (a_{ij})$.

Fie $S \subseteq V$ o submulțime nevidă a lui V . Prin definiție, o tăietură (notată în continuare prin $\delta(S)$) este o mulțime de muchii $(v_i, v_j) \in U$ cu proprietatea că dacă $v_i \in S$, atunci $v_j \in V \setminus S$. Se pune problema determinării tăieturii $\delta(S)$ cu

valoarea maximă a sumei ponderilor muchiilor, adică a partiționării mulțimii V în două părți S și $V \setminus S$, astfel că suma

$$\sum_{(v_i, v_j) \in \delta(S)} a_{ij}$$

este maxim posibilă oricare ar fi $S \subseteq V$.

Vom introduce variabilele $x_i, i = 1, 2, \dots, n$, cu $x_i = 1$ sau $x_i = -1$, după cum vârful v_i aparține sau nu mulțimii S :

$$x_i = \begin{cases} 1, & \text{pentru } \forall v_i \in S, \\ -1, & \text{pentru } \forall v_i \in V \setminus S. \end{cases}$$

Dacă vârfurile v_i și v_j aparțin mulțimii S sau mulțimii complementare $V \setminus S$, atunci $x_i = x_j$ și deci $x_i x_j = x_i^2 = x_j^2 = 1$. În cazul când $x_i = -x_j$, adică $v_i \in S$ și $v_j \in V \setminus S$ sau $v_i \in V \setminus S$ și $v_j \in S$, avem $x_i x_j = -1$. Ținând seama că $a_{ij} = a_{ji}$ și $x_i x_i = 1$, rezultă că

$$\begin{aligned} \sum_{(v_i, v_j) \in \delta(S)} a_{ij} &= \frac{1}{4} \sum_{i,j} a_{ij} (1 - x_i x_j) = \\ &= \frac{1}{4} \sum_{i=1}^n \left(\sum_{j=1}^n a_{ij} x_i x_i - \sum_{j=1}^n a_{ij} x_i x_j \right) = \\ &= \frac{1}{4} x^T (\text{Diag}(Ae) - A)x, \end{aligned}$$

unde $x = (x_1, x_2, \dots, x_n)^T$ și $e = (1, 1, \dots, 1)^T \in \mathfrak{R}^n$, iar $\text{Diag}(Ae)$ este matricea diagonală cu diagonala formată din vectorul Ae .

Matricea

$$L = \text{Diag}(Ae) - A$$

se numește **matricea Laplace**, asociată grafului G cu matricea adiacentă A . Elementele l_{ij} ale acestei matrice satisfac relațiile:

$$l_{ij} = -a_{ij} = l_{ji}, \quad \forall i \neq j,$$

$$l_{ii} = \sum_{j=1}^n a_{ij} = \sum_{j \neq i} |l_{ij}| > 0, \quad \forall i = 1, 2, \dots, n,$$

deci matricea lui Laplace L este o matrice pozitiv semidefinită.

Notăm $Q = \frac{1}{4}L$. Atunci problema determinării unei tăieturi maxime în grafuri neorientate poate fi reformulată astfel:

$$\left. \begin{array}{l} q(x) = x^T Q x \rightarrow \max \\ \text{în condițiile} \\ x_i^2 - 1 = 0, \quad i = 1, 2, \dots, n \end{array} \right\} \quad (1)$$

2. RELAXAREA SEMIDEFINITĂ

Problema (1) este o problemă NP -completă (vezi, de ex., [4,5]). Ea este un caz particular al unei probleme de programare pătratică cu restricții pătratice și este dificilă de rezolvat. În general, rezolvarea problemelor de optimizare combinatorială necesită o enumerare completă a soluțiilor admisibile. De aceea, considerăm relaxarea ei. Ideea de bază constă în relația $x^T Q x = \text{Tr}(Q x x^T)$, unde $\text{Tr}(\cdot)$ reprezintă urma

matricei, adică $\text{Tr}(A) = \sum_{i=1}^n a_{ii}$. Notăm $X = x x^T$.

Este clar că matricea $X = x x^T$ este pozitiv semidefinită și $\text{rang}(x x^T) = 1$. Din $x^2 = 1, i = 1, 2, \dots, n$ rezultă că $\text{diag}(x x^T) = e$, unde $\text{diag}(X)$ semnifică vectorul, ale cărui componente sunt elementele de pe diagonala principală a matricei X . Se demonstrează (vezi, de ex., [6]) că problema considerată (1) este echivalentă următoarei probleme:

$$\left. \begin{array}{l} \text{Tr}(QX) \rightarrow \max \\ \text{diag}(X) = e, \\ \text{rang}(X) = 1, \\ X \succeq 0. \end{array} \right\} \quad (2)$$

Inegalitatea $A \succeq B$ ($A \succ B$) semnifică că matricea $A-B$ este semipozitiv (respectiv pozitiv) definită. Renunțând în (2) la condiția $\text{rang}(X)=1$, se ajunge la problema relaxată

$$\left. \begin{array}{l} \text{Tr}(QX) \rightarrow \max \\ \text{diag}(X) = e, \\ X \succeq 0, \end{array} \right\} \quad (3)$$

Problema (3) este o problemă de programare

semidefinită și duală ei poate fi formulată în felul următor (vezi, de ex. [7-9]):

$$\left. \begin{array}{l} e^T u \rightarrow \min \\ \text{în condițiile} \\ \text{Diag}(u) - Q \succeq 0 \end{array} \right\} \quad (4)$$

Domeniul soluțiilor admisibile

$$T = \{u \in \mathbb{R}^n \mid \text{Diag}(u) - Q \succeq 0\}$$

pentru problema (4) este o mulțime convexă și, prin urmare, problema (3) este o problemă de programare convexă, care are o complexitate polinomială (vezi, de ex., [10, §5.3], [11]).

Așadar, problema duală (4) este mai “ușor” de rezolvat decât problema considerată (1), care este o problemă NP -completă.

Notăm $e_i^T = (0, \dots, 0, 1, 0, \dots, 0)$ vectorul cu unitatea pe poziția i , $i = 1, 2, \dots, n$. Fie matricea

$$E_i = e_i e_i^T. \text{ Atunci putem scrie: } \text{Diag}(u) = \sum_{i=1}^n u_i E_i.$$

Cu aceste notări problema (4) devine:

$$\left. \begin{array}{l} e^T u \rightarrow \min \\ \text{în restricțiile} \\ u_1 E_1 + u_2 E_2 + \dots + u_n E_n - Q \succeq 0. \end{array} \right\} \quad (5)$$

Goemans și Wiliamson [12] au estimat că

$$q^* = \max\{x^T Q x \mid x_i^2 = 1, \quad i = 1, 2, \dots, n\} \\ \leq 0.87856 \min\{e^T u \mid u \in T\}.$$

În cazul rezolvării problemelor generale de programare pătratică cu restricții pătratice estimări analogice au fost obținute în [12-15]. Alte relaxări echivalente pot fi găsite în [16-18]

3. METODA PUNCTULUI INTERIOR

Asociem problemei duale (4) funcția de barieră logaritmică [11]:

$$B(u) = \ln \det(\text{Diag}(u) - Q).$$

Aici \det înseamnă determinantul matricei considerate.

Notăm

$$C(u) = \text{Diag}(u) - Q = \sum_{i=1}^n u_i E_i - Q.$$

Din formula Cauchy-Benet:

$$C(u) \times \text{adj}(C(u)) = \det(C(u)) \times I$$

avem

$$\frac{\partial \det(C(u))}{\partial C} = \text{adj}(C(u)),$$

unde $\text{adj}(\bullet)$ este matricea adjunctă. Luând în considerare cele de mai sus, putem calcula

$$\begin{aligned} \frac{\partial B(u)}{\partial u_i} &= \frac{I}{\det(C(u))} \times \text{Tr}(E_i \times \text{adj}(C(u))) = \\ &= \text{Tr}(E_i \times C^{-1}(u)). \end{aligned}$$

Dar

$$C^{-1}(u) = \frac{I}{\det(C(u))} \text{adj}(C(u)).$$

Prin urmare

$$\nabla B(u) = \text{diag}(C^{-1}(u)). \quad (6)$$

Metoda punctului interior pentru problema (4) constă în rezolvarea șirului de probleme de optimizare necondiționată:

$$e^T u - \mu_k \ln \det(\text{Diag}(u) - Q) \rightarrow \min_{u \in \mathfrak{R}^n}, \quad (7)$$

unde parametrul de barieră μ_k este un număr real fixat și $\mu_k \downarrow 0$. Condiția de optimalitate pentru problema de optimizare necondiționată (7) este

$$e - \mu_k \text{diag}(C^{-1}(u)) = 0. \quad (8)$$

Problema (7) poate fi rezolvată cu ajutorul metodelor de gradient (metoda celei mai rapide descreșteri, metoda gradientului conjugat ș.a.), criteriul de stopare fiind îndeștularea aproximativă a condiției (8). Aceasta necesită calculul gradientului funcției de barieră $B(u)$ dat prin formula (6). După cum se vede din (6), avem nevoie de elementele de pe diagonala principală a matricei inverse $C^{-1}(u)$. Este bine cunoscut că inversarea matricelor este o operație costisitoare și trebuie evitată în practică. În continuare ne vom opri asupra unei proceduri efective de calcul al gradientului funcției de barieră. În această procedură, determinarea elementelor

diagonale ale matricei considerate se reduce la minimizarea necondiționată a unor funcții pătratice generate de una și aceeași matrice.

4. O PROCEDURĂ EFICIENTĂ DE CALCUL AL GRADIENTULUI

Punem problema determinării $\nabla B(u)$. Observăm că

$$z^T C^{-1}(u) z = \max_{z \in \mathfrak{R}^n} \{2z^T y - y^T C(u) y\}, \forall z \in \mathfrak{R}^n.$$

Într-adevăr, fie y_* soluția optimă, adică $2C(u)y_* = 2z$, ori $y_* = C^{-1}(u)z$. Rezultă, că

$$\begin{aligned} \max_y \{2z^T y - y^T C(u) y\} &= \\ &= 2z^T C^{-1}(u) z - z^T C^{-1}(u) C(u) C^{-1}(u) z = \\ &= z^T C^{-1}(u) z. \end{aligned}$$

Notăm elementele de pe diagonala principală a matricei $C^{-1}(u)$ prin $\bar{c}_{ii}(u)$. Așa cum $\bar{c}_{ii} = e_i^T C^{-1}(u) e_i$, din cele de mai sus rezultă că

$$\bar{c}_{ii} = -\min_{y \in \mathfrak{R}^n} \{y^T C(u) y - 2e_i^T y\}, i = 1, 2, \dots, n.$$

Prin urmare calculul elementelor $\bar{c}_{ii}(u)$ se reduce la minimizarea a n funcții pătratice cu una și aceeași matrice pozitiv definită $C(u)$.

Fie acum cunoscută factorizarea Cholesky pentru matricea $C(u)$, adică $C(u) = F^T(u)F(u)$, unde $F(u)$ este o matrice inferior triunghiulară.

Atunci

$$\begin{aligned} \bar{c}_{ii} &= -\min_y \{y^T F^T(u)F(u)y - 2e_i^T y\} = \\ &= -\min_y \{\|F(u)y\|_2^2 - 2e_i^T y\} \end{aligned}$$

Notăm prin $r_i \in \mathfrak{R}^n$ soluțiile sistemului de ecuații liniare:

$$F^T(u)r_i = e_i, i = 1, 2, \dots, n \quad (9)$$

și deci

$$r_i = F^{-T}(u)e_i.$$

De aici avem

$$\begin{aligned} r_i^T r_i &= e_i^T F^{-1}(u) F^{-T}(u) e_i = \\ &= e_i^T C^{-1}(u) e_i = \bar{c}_{ii}. \end{aligned}$$

Așa dar, având factorizarea Cholesky, componentele gradientului funcției de barieră $B(u)$ pot fi calculate astfel:

$$\frac{\partial B(u)}{\partial u_i} = \bar{c}_{ii} = r_i^T r_i, i = 1, 2, \dots, n,$$

unde vectorii r_i sunt soluțiile sistemului (9).

Remarcăm că dacă se cunoaște factorizarea Cholesky $C(u) = F^T F$ ușor poate fi calculată și valoarea funcției-scop în problema (7):

$$\begin{aligned} e^T u - \mu_k \ln \det(C(u)) &= \\ &= \sum_{i=1}^n u_i - \mu_k \ln(\det(F(u)))^2 = \\ &= \sum_{i=1}^n u_i - 2\mu_k \ln(f_{11} \cdot f_{22} \cdot \dots \cdot f_{nn}) = \\ &= \sum_{i=1}^n u_i - 2\mu_k (f_{11} + f_{22} + \dots + f_{nn}). \end{aligned}$$

Bibliografie

1. **Poljak S., Tuza Z.** The max-cut problem – a survey. In Lovasz L., Seymour P., editors, *Special Year on Combinatorial Optimizations. DIMACS Series in Discrete Mathematics and Theoretical Computer science. American Mathematical Society.* 1995.
2. **Garey M., Johnson D., Stochemeyer.** Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1: 237-267, 1976.
3. **Hadlock F.** Finding a maximum cut of a planar graph in polynomial time. *SIAM Journal on Computing*, 4,:221-225, 1975.
4. **Murty K. G.** Some NP-complete problems in quadratic and nonlinear programming. *Mathematical Programming*, 39: 117-129, 1987.
5. **Pardalos P.M., Vavasis S.A.** Quadratic programming with one negative eigenvalue is NP-hard. *J. Global Optim.*, 1(1):, 1991.
6. **Laurent M., Poljak S.** On a positive semidefinite relaxation of the cut polytope. *Linear Algebra and its Applications*, 223/224, 439-461, 1995.
7. **Wolcovicz H., Anjos M.F.** Semidefinite programming for discrete optimization and matrix completion problems. June 30, 2000. Research Report CORR 2000-38. University of Waterloo, Ontario, 88 pages. <http://orion.math.uwaterloo.ca/~hwolkowi/henry/reports>
8. **Vandenbergh L., Boyd S.** Semidefinite programming. *SIAM Review*, 38, 49-95, 1996.
9. **Lemarechal C., Oustry F.** Semidefinite relaxations and Lagrangian duality with application to combinatorial optimization. Technical report nr. 3710, Juine, 1999, INRIA, Montbonnot st. Martin, France. 40 pages.
10. **Ben- Tal A., Nemirovski A.** Convex optimization in engineering: Modeling, Analysis, Algorithms. Faculty of Industrial Engineering and Management, Technion- City, Haifa 32000, Israel., 1998. 299 p.
11. **Nesterov Y.E., Nemirovski A.S.** Interior point polynomial algorithms in convex programming. *SIAM Publications. SIAM, Philadelphia, USA,* 1994.
12. **Goemans M., Williamson D.P.** Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. Assoc. Comput. Mach.*, 42(6): 1115-1145, 1995.
13. **Nesterov Y.** Semidefinite relaxation and nonconvex quadratic optimization. *Optimization Methods and Software*, 9: 141`-160, 1998.
14. **Ye Y.** Approximation quadratic programming with bound and quadratic constraints. *Math. Programming*, 84(2): 219-226, 1999.
15. **Bertsimas D., Ye Y.** Semidefinite relaxations, multivariate normal distributions and order statistics. Working Paper, Department of Management Science, The University of Iowa, 1997.
16. **Poljak S., Wolcovicz H.** Convex relaxations of (0,1) quadratic programming. *Mathematics of Operations Research*, 20(3): 550-561, 1995.
17. **Poljak S., Rendl F.** Solving the max-cut problem using eigenvalues. *Discrete Applied Mathematics*, 62:249-278, 1995.
18. **Moraru V.** Determinarea tăieturii maxime într-un graf folosind programarea semidefinită. *Analele Academiei de Studii Economice din Moldova.* 498-504, 2001.