ADMM for the SDP relaxation of the QAP and QKP

Daniela Cristina Lubke, Marcia Fampa[1], Fei Wang, Henry Wolkowicz, Danilo Elias Oliveira[2] and Yangyang Xu[4]

[1]Dept. of Systems Engineering and Computer Science, Federal University of Rio de Janeiro (UFRJ).
[2]Dept. of Combinatorics and Optimization, University of Waterloo.
[4]Institute for Mathematics and its Applications (IMA), University of Minnesota.

Introduction

The *quadratic assignment problem*, **QAP**, in the trace formulation is

$$(\mathbf{QAP}) \qquad p^* := \min_{X\in\Pi_n} \langle AXB - 2C, X\rangle, \qquad (1)$$

where $A, B \in \mathbb{S}^n$ are real symmetric $n\times n$ matrices, C is a real $n\times n$ matrix, $\langle \cdot,\cdot\rangle$ denotes the *trace inner product* and Π_n denotes the set of $n\times n$ permutation matrices. A typical objective of the **QAP** is to assign n facilities to n locations while minimizing total cost. The assignment cost is the sum of costs using the flows in A_{ij} between a pair of facilities i, j multiplied by the distance in B_{st} between their assigned locations s, t and adding on the location costs of a facility i in a position s given in C_{is}.

The new derivation

We start the derivation from the following equivalent quadratically constrained quadratic problem

$$\begin{array}{ll} \min\limits_X & \langle AXB - 2C, X\rangle \\ \text{s.t.} & X_{ij}X_{ik} = 0,\ X_{ji}X_{ki} = 0,\ \forall i, \forall j \neq k, \\ & X_{ij}^2 - X_{ij} = 0,\ \forall i,j, \\ & \sum\limits_{i=1}^n X_{ij}^2 - 1 = 0,\ \forall j,\ \sum\limits_{j=1}^n X_{ij}^2 - 1 = 0,\ \forall i. \end{array} \qquad (2)$$

The Lagrangian for (2) is

$$\begin{aligned} \mathcal{L}_0(X,U,V,W,u,v) = \langle AXB - 2C, X\rangle + \sum_{i=1}^n \sum_{j\neq k} U_{jk}^{(i)} X_{ij}X_{ik} \\ + \sum_{i=1}^n \sum_{j\neq k} V_{jk}^{(i)} X_{ji}X_{ki} + \sum_{i,j} W_{ij}(X_{ij}^2 - X_{ij}) \\ + \sum_{j=1}^n u_j \left(\sum_{i=1}^n X_{ij}^2 - 1\right) + \sum_{i=1}^n v_i \left(\sum_{j=1}^n X_{ij}^2 - 1\right). \end{aligned}$$

The dual problem is a maximization of the dual functional d_0,

$$\max\ d_0(U,V,W,u,v) := \min_X \mathcal{L}_0(X,U,V,W,u,v). \qquad (3)$$

To simplify the dual problem, we homogenize $\mathcal{L}_0$ by multiplying the degree-one terms in X by a scalar variable x_0 and adding the single constraint $x_0^2 = 1$ to the dual functional. We add the additional dual variable w_0 and let

$$\begin{aligned} \mathcal{L}_1(X,x_0,U,V,W,w_0,u,v) = \langle AXB - 2x_0C, X\rangle \\ + \sum_{i=1}^n \sum_{j\neq k} U_{jk}^{(i)} X_{ij}X_{ik} + \sum_{i=1}^n \sum_{j\neq k} V_{jk}^{(i)} X_{ji}X_{ki} + \sum_{i,j} W_{ij}(X_{ij}^2 - x_0X_{ij}) \\ + \sum_{j=1}^n u_j \left(\sum_{i=1}^n X_{ij}^2 - 1\right) + \sum_{i=1}^n v_i \left(\sum_{j=1}^n X_{ij}^2 - 1\right) + w_0(x_0^2 - 1). \end{aligned}$$

This homogenization technique is the same as that in [3].
The new dual problem is

$$\max\ d_1(U,V,W,w_0,u,v) := \min_{X,x_0} \mathcal{L}_1(X,x_0,U,V,W,w_0,u,v).$$

Definition: Given n^2 matrices $\bar{Y}_{ij}$ for $i = 1,\dots,n$ and $j = 1,\dots,n$ that satisfy $\bar{Y}_{ij} = \bar{Y}_{ji}^T$, let $\bar{Y}$ be the $n\times n$ block matrix with $\bar{Y}_{ij}$ as the (i,j)-th block. We form the symmetric block matrix

$$Y = \begin{bmatrix} y_{00} & y_0^T \\ y_0 & \bar{Y} \end{bmatrix}, \qquad (4)$$

where y_{00} is a scalar, and y_0 is a vector in $\mathbb{R}^{n^2}$. The **SDP** relaxation of (2) is:

$$\begin{array}{ll} \min & \langle L_Q, Y\rangle \\ \text{s.t.} & \mathcal{G}_J(Y) = E_{00} \\ & \operatorname{diag}(\bar{Y}) = y_0 \\ & \operatorname{trace}(\bar{Y}_{ii}) = 1,\ \forall i \\ & \sum_{i=1}^n \bar{Y}_{ii} = I \\ & Y \succeq 0, \end{array} \qquad (5)$$

where

$$\mathcal{G}_J(Y)_{ij} = \begin{cases} Y_{ij} & \text{if } (i,j)\in J \text{ or } (j,i)\in J \\ 0 & \text{otherwise.} \end{cases}$$

$$L_Q = \begin{bmatrix} 0 & -\operatorname{vec}(C)^T \\ -\operatorname{vec}(C) & B\otimes A \end{bmatrix}.$$

The **SDP** relaxation of **QAP** presented in [3] uses facial reduction to guarantee strict feasibility is:

$$\begin{array}{ll} p_R^* := \min_R & \langle L_Q, \hat{V}R\hat{V}^T\rangle \\ \text{s.t.} & \mathcal{G}_J(\hat{V}R\hat{V}^T) = E_{00} \\ & R \succeq 0. \end{array} \qquad (6)$$

A New ADMM Algorithm for the SDP Relaxation

We can write (6) equivalently as

$$\min_{R,Y} \langle L_Q, Y\rangle \ \text{ s.t. } \ \mathcal{G}_J(Y) = E_{00},\ Y = \hat{V}R\hat{V}^T,\ R \succeq 0. \qquad (7)$$

The following theorem from [3] shows the equivalence between (5) and (7).

Theorem: A matrix Y is feasible for (5) *if, and only if*, it is feasible for (7). □

Therefore we can work with (7). The *augmented Lagrange* of (7) is

$$\mathcal{L}_A(R,Y,Z) = \langle L_Q, Y\rangle + \langle Z, Y - \hat{V}R\hat{V}^T\rangle + \frac{\beta}{2}\|Y - \hat{V}R\hat{V}^T\|_F^2. \qquad (8)$$

Recall that (R,Y,Z) are the primal reduced, primal, and dual variables respectively. We denote (R,Y,Z) as the *current iterate*. Our new algorithm, an application of **ADMM**, uses the augmented Lagrangian in (8) and performs the following updates to obtain a new iterate (R_+, Y_+, Z_+):

$$R_+ = \arg\min_{R\in\mathbb{S}_+^n} \mathcal{L}_A(R,Y,Z), \qquad (9a)$$

$$Y_+ = \arg\min_{Y\in\mathcal{P}_i} \mathcal{L}_A(R_+,Y,Z), \qquad (9b)$$

$$Z_+ = Z + \gamma\cdot\beta(Y_+ - \hat{V}R_+\hat{V}^T), \qquad (9c)$$

where the simplest case for the polyhedral constraints $\mathcal{P}_i$ is the linear manifold from the *gangster constraints*:

$$\mathcal{P}_1 = \{Y \in \mathbb{S}^{n+2} : \mathcal{G}_J(Y) = E_{00}\}.$$

We use this notation as we add additional simple polyhedral constraints. The second case is the polytope:

$$\mathcal{P}_2 = \mathcal{P}_1 \cap \{0 \leq Y \leq 1\}.$$

Let $\hat{V}$ be normalized such that $\hat{V}^T\hat{V} = I$. Then the R-subproblem can be explicitly solved by

$$\begin{aligned} R_+ &= \arg\min_{R\succeq 0} \langle Z, Y - \hat{V}R\hat{V}^T\rangle + \tfrac{\beta}{2}\|Y - \hat{V}R\hat{V}^T\|_F^2 \\ &= \arg\min_{R\succeq 0} \left\|Y - \hat{V}R\hat{V}^T + \tfrac{1}{\beta}Z\right\|_F^2 \\ &= \arg\min_{R\succeq 0} \left\|R - \hat{V}^T(Y + \tfrac{1}{\beta}Z)\hat{V}\right\|_F^2 \\ &= \mathcal{P}_{\mathbb{S}_+}\left(\hat{V}^T(Y + \tfrac{1}{\beta}Z)\hat{V}\right), \end{aligned}$$

where $\mathbb{S}_+$ denotes the **SDP** cone, and $\mathcal{P}_{\mathbb{S}_+}$ is the orthogonal projection onto $\mathbb{S}_+$. For any symmetric matrix W, we have $\mathcal{P}_{\mathbb{S}_+}(W) = U_+\Sigma_+U_+^T$, where (U_+, Σ_+) contains the positive eigenpairs of W; we let (U_-, Σ_-) be for the negative eigenpairs.

If $i = 1$ in (9b), the Y-subproblem also has a closed-form solution:

$$\begin{aligned} Y_+ &= \arg\min_{\mathcal{G}_J(Y)=E_{00}} \langle L_Q, Y\rangle + \langle Z, Y - \hat{V}R_+\hat{V}^T\rangle + \frac{\beta}{2}\|Y - \hat{V}R_+\hat{V}^T\|_F^2 \\ &= \arg\min_{\mathcal{G}_J(Y)=E_{00}} \left\|Y - \hat{V}R_+\hat{V}^T + \frac{L_Q + Z}{\beta}\right\|_F^2 \\ &= E_{00} + \mathcal{G}_{J^c}\left(\hat{V}R_+\hat{V}^T - \frac{L_Q + Z}{\beta}\right). \end{aligned}$$

One major advantage of using **ADMM** is that the complexity increases marginally when we add constraints to (6) and tighten the **SDP** relaxation. If $0 \leq \hat{V}R\hat{V}^T \leq 1$ is added in (6), then we simply add the constraints $0 \leq Y \leq 1$ to (7). This yields the new problem

$$p_{RY}^* := \min_{R,Y} \{\langle L_Q, Y\rangle : \mathcal{G}_J(Y) = E_{00},\ 0 \leq Y \leq 1,\ Y = \hat{V}R\hat{V}^T,\ R \succeq 0\}.$$

The **ADMM** for solving p_{RY}^* has the same R-update and Z-update as those in (9). The Y-update is changed to

$$Y_+ = E_{00} + \min\left(1, \max\left(0, \mathcal{G}_{J^c}(\hat{V}R_+\hat{V}^T - \frac{L_Q + Z}{\beta})\right)\right).$$

The nonnegativity constraint means that the ≤ 1 constraint is redundant. But the inclusion makes the algorithm converge faster and avoid roundoff error.

Result and discussions

The proposed methods was code in MATLAB on a PC with 16 Gigabyte memory running Window 7.

0. Instance name	1. opt value	2. Bundle LowBnd	3. HKM-FR LowBnd	4. ADMM LowBnd	5. feas UpBnd	6. ADMM %gap	7. ADMM vs Bundle %Impr LowBnd
Esc16a	68	59	50	64	70	8.82	7.35
Esc16b	292	288	276	290	294	1.37	0.68
Had18	5358	5317	5287	5358	5358	0.00	0.77
Had20	6922	6885	6848	6922	6930	0.12	0.53
Kra30a	88900	77647	-1111	86838	105650	21.16	10.34
Kra30b	91420	81156	-1111	87858	102370	15.87	7.33
Kra32	88700	79659	-1111	85773	103070	19.50	6.89
Nug21	2438	2323	2386	2382	2748	15.01	2.42
Nug22	3596	3440	3396	3529	3860	9.20	2.47
Nug28	5166	4901	-1111	5026	5492	9.02	2.42
Nug30	6124	5803	-1111	5950	6720	12.57	2.40
Rou15	354210	333287	323235	350217	367782	4.96	4.78
Rou20	725522	663833	642856	695181	765390	9.68	4.32
Scr15	51140	48836	42204	51140	55760	9.03	4.51
Scr20	110030	94998	83302	106801	124522	16.11	10.73
Tai20a	703482	637300	619092	671675	750450	11.20	4.89
Tai25a	1167256	1041337	-1111	1096657	1271696	15.00	4.74
∗Tai30a	1818146	1652186	-1111	1706871	1942086	12.94	3.01
Tho30	149936	136059	-1111	143576	162882	12.88	5.01

Table 1: Results of lower and upper bounds for each instance in **QAPLIB** Instances I. Failure of an algorithm is marked by -1111 and the optimal value of the instance marked by $*$ is still unknown.

Summary and conclusions

We have shown the efficiency of using the **ADMM** approach in solving the **SDP**relaxation of the **QAP** problem. In particular, we have shown that we can obtain high accuracy solutions of the **SDP** relaxation in less significantly less cost than current approaches. In addition, the **SDP** relaxation includes the nonnegativity constraints at essentially no extra cost. This results in both a fast solution and improved lower and upper bounds for the **QAP**. In future work we plan to apply **ADMM** to the *binary quadratic knapsack problem*:

$$(\mathbf{QKP}) \qquad \begin{array}{ll} \max & x^TQx \\ \text{s.t.} & w^Tx \leq c \\ & x \in \{0,1\}^n, \end{array}$$

where $Q \in \mathbb{S}^n$ is a symmetric $n\times n$ nonnegative integer matrix indicating profit for selected items, $w \in \mathbb{Z}_+^n$ are positive integer weights for the items, and $c \in \mathbb{Z}_+$ is a positive, integer knapsack capacity. The binary variable x indicates whether an item is chosen or not.

References

[1] Oliveira D.E., Wolkowicz, H., Xu Y., ADMM for the SDP relaxation of the QAP, *Submitted on 17 Dec 2015 in arXiv:1512.05448.*

[2] Rendl, F. and Sotirov, R., Bounds for the quadratic assignment problem using the bundle method. *Mathematical Programming 2(1):71-109,1998.*

[3] Zhao, Qing and Karisch, Stefan E. and Rendl, Franz and Wolkowicz, Henry, Semidefinite Programming Relaxations for the Quadratic Assignment Problem.*Journal of Combinatorial Optimization, 2(1):71-109,1998.*