

一般化優対角行列による緩和を用いた Pooling Problem に対する解法の構築

君塚 柁貴

東京工業大学情報理工学院数理・計算科学系（現：(株)日立製作所）
 指導教員：山下 真 東京工業大学 准教授

1. はじめに

石油は世界のエネルギー供給の中心的役割を果たしており、今後もその役割を担っていくことが考えられる。そのため、石油の原料である原油は重要な資源であるが、その埋蔵量は有限であることが知られている。したがって、原油を需要に合わせて効率良く使用していく必要がある。

本研究ではこうした実問題に対して、需要を満たす中で効率の良い原油の輸送スケジュールが解となる Pooling Problem (PP) [1, 2] に注目した。そして PP に対し、既存手法 [1] より高精度でかつ高速で解を求めることが可能な手法を構築した。特に、PP の SDP 緩和と同じ最適値が LP 緩和で得られることを理論的に示した。また、再スケジュールリング法により、効率的に解を得ることに成功した。

2. Pooling Problem の定式化と既存手法

表 1 に記載されている集合、変数、定数を用いて図 1 で表される PP を定式化すると次のようになる。この問題は非凸二次の MINLP で定式化されており NP 困難であることが知られている [2]。

$$\begin{aligned} \min_{a, p, q, u, v} & \sum_{t \in T} \left(\sum_{(i, j) \in A} CA_{ij} a_{ij}^t + \sum_{i \in V_P} CQ_i DC_i^t v_i^t \right) \\ \text{s.t.} & u_{ij}^t L_{ij} \leq a_{ij}^t \leq u_{ij}^t U_{ij}, ((i, j) \in A, t \in T), \\ & u_{ij}^t \in \{0, 1\}, \\ & \sum_{j \in I(i)} u_{ji}^t + \sum_{k \in E(i)} u_{ik}^t \leq 1, (i \in V, t \in T), \\ & p_i^{t+1} = p_i^t + SA_i^t - \sum_{k \in E(i)} a_{ik}^t, p_i^t \geq 0, \end{aligned}$$

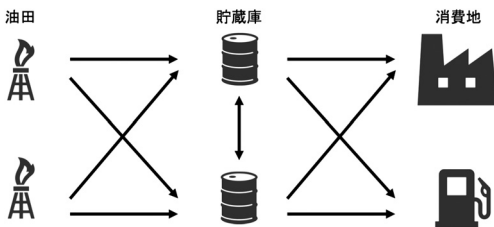


図 1 Pooling Problem の全体像

$$\begin{aligned} p^{M_T+1} &= 0, (i \in V_S, t \in T), \\ p_i^{t+1} q_i^{t+1} &= p_i^t q_i^t + SA_i^t SQ_i^t - \sum_{k \in E(i)} a_{ik}^t q_i^t, \\ &(i \in V_S, t \in T), \\ p_i^{t+1} &= p_i^t + \sum_{k \in I(i)} a_{ki}^t - \sum_{k \in E(i)} a_{ik}^t, \\ 0 < p_i^{\min} &\leq p_i^t \leq p_i^{\max}, (i \in V_B, t \in T), \\ p_i^{t+1} q_i^{t+1} &= p_i^t q_i^t + \sum_{k \in I(i)} a_{ki}^t q_k^t - \sum_{k \in E(i)} a_{ik}^t q_i^t, \\ &(i \in V_B, t \in T), \\ q_i^t &= \frac{1}{DC_i^t} \sum_{j \in I(i)} a_{ij}^t q_j^t, (i \in V_P, t \in T), \\ v_i^t &\geq \max\{0, DQ_i^t - q_i^t\}, (i \in V_P, t \in T). \end{aligned}$$

表 1 定式化で用いる集合、定数、変数

集合					
V_S	油田の集合	V_B	貯蔵庫の集合	V_P	消費地の集合
T	計画時間の集合	V	油田、貯蔵庫、消費地の集合		
(i, j)	i から j への連絡	$I(i)$	i に向かうバスの出発点の集合		
A	バス (i, j) の集合	$E(i)$	i から出るバスの行き先の集合		
定数					
M_S	油田の数	M_B	貯蔵庫の数	M_P	消費地の数
M_T	計画期間	SA_i^t	供給される量	SQ_i^t	供給される質
U_{ij}	流す際の最大量	L_{ij}	流す際の最小量	CA_{ij}	輸送コスト
DC_i^t	要求される量	DQ_i^t	要求される質	CQ_i	罰金係数の重み
p_i^{\max}	保存できる最大量	p_i^{\min}	保存すべき最小量		
変数					
a_{ij}^t	流す量	p_i^t	保有する量	q_i^t	保有する質
u_{ij}^t	binary 変数	v_i^t	罰金項		

既存手法 [1] では、このように定式化される PP に対し、PP を非凸二次の NLP に緩和した後、次のような三段階に分けて解く Three step method を提案した。

1. PP の SDP 緩和を解き緩和解 x_{rela} を得る
2. x_{rela} を用いて PP の実行可能解 x_{fea} を得る
3. x_{fea} を初期点として局所探索を行い局所最適解 x^* を得る

3. 提案手法と示した定理

本研究では、SDSOS, DSOS [3] 緩和を用いた手法を二つ提案した。また、提案手法の根拠となる定理の証明を行った。一つ目の提案手法は、[1] と SDSOS, DSOS 緩和を組合せた手法であり、SDSOS, DSOS 緩和はそれぞれ SOCP, LP 緩和の特殊形である。この手

法を提案する根拠となった定理 3.1 を示した (証明略).

表 2 問題の設定

問題	M_S	M_B	M_P	M_T	変数の数 (n)
3	2	2	1	10	176
8	2	4	2	28	1228

定理 3.1. 次のような QCQP を考える.

(QCQP-Problem)

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & x^T A_0 x + 2b_0^T x + c_0 \\ \text{s.t.} \quad & x^T A_j x + 2b_j^T x + c_j \leq 0, \\ & (j = 1, \dots, m). \end{aligned}$$

係数行列 A_0, A_1, \dots, A_m の対角成分は全て 0 とする. この時, QCQP-Problem の SDP 緩和と SDSOS 緩和の最適値は一致する. また, 次のように定義される集合 S が有界であると仮定する.

$$S := \left\{ x \in \mathbb{R}^n : \begin{pmatrix} c_j & b_j^T \\ b_j & A_j \end{pmatrix} \bullet \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \leq 0 \right. \\ \left. (j = 1, \dots, m), \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \succeq O, \exists X \succeq O \right\}.$$

(ここで S は, QCQP-Problem の制約集合を実対称行列 X を用いて SDP 緩和したものである.) この時, QCQP-Problem の SDP 緩和と DSOS 緩和の最適値は一致する.

二つ目の提案手法は, 輸送スケジュールを逐次修正していく手法であり, アルゴリズムは次のとおりである.

再スケジューリング法

1. $\hat{t} = 0$ とする
2. 時刻 $\hat{t} + 1, \dots, M_T$ までの PP を定式化し SDSOS (または DSOS) 緩和を用いて解いて緩和解 x_{rela} を得る
3. x_{rela} をもとに PP の実行可能解 x_{fea} を得る
4. x_{fea} が需要を満たすか調べる (満たしていれば x_{fea} を返して終了)
5. 需要をはじめて満たさない時刻を t^+ とし, $\hat{t} + 1, \dots, t^+$ までの輸送スケジュールを需要を満たすよう修正する (修正が不可能ならば x_{fea} を返して終了)
6. $\hat{t} = t^+$ とし 2. に戻る ($\hat{t} = M_T$ ならば終了)

再スケジューリング法の重要となる部分が Step 5 の操作である. Step 5 は各時刻毎に x_{fea} が需要を満たしているかどうかを確認し, 需要を満たしている場合は輸送スケジュールの効率化, 満たしていない場合は

表 3 数値実験結果: 問題 3, 8

問題 3 ($n = 176$)				
	満足度	緩和値	目的関数値	実行時間 [s]
Three step	98.66%	181.14	6594.55	2874.69
提案手法 1 (SDP)	99.99%	300.00	546.91	3404.74
提案手法 1 (SDSOS)	80.11%	300.00	89974.73	2901.57
提案手法 1 (DSOS)	99.99%	300.00	510.81	2906.73
再スケジュール (SDSOS)	100.00%	*	503.97	38.11
再スケジュール (DSOS)	99.70%	*	1852.02	34.52
問題 8 ($n = 1228$)				
	満足度	緩和値	目的関数値	実行時間 [s]
Three step	98.27%	458.22	22214.26	39700.32
提案手法 1 (SDP)			> 24 hours	
提案手法 1 (SDSOS)	99.99%	824.00	1714.70	22675.94
提案手法 1 (DSOS)	99.33%	824.00	8957.09	40951.62
再スケジュール (SDSOS)	100.00%	*	1636.82	14112.74
再スケジュール (DSOS)	100.00%	*	1634.99	9496.46

輸送スケジュールの再編成を行う.

4. 数値実験

先に紹介した手法の比較のため数値実験を行った (一部掲載). 問題 3, 8 の設定の一部は表 2 のとおりである. 得られた解は満足度で評価をする. 満足度は $(\sum_{t \in T} \sum_{i \in V_P} DQ_i^t - \sum_{t \in T} \sum_{i \in V_P} v_i^t) / \sum_{t \in T} \sum_{i \in V_P} DQ_i^t$ で定義され, 需要をどの程度満たしているかを示している.

表 3 より, SDP 緩和と同じ値を SDSOS, DSOS 緩和でも得られることを確認できた. また, 再スケジューリング法により, 実行時間を大幅に短縮することに成功した.

そして, 再スケジューリング法は既存手法 [1] と比較して大規模な問題を高速で解くことが可能であることを数値実験から確認した.

定理 3.1 の適用範囲の拡大, 再スケジューリング法のアルゴリズムの改良, そして実データを盛り込んだ問題に適用などが今後の課題として挙げられる.

参考文献

- [1] T. Nishi, "A semidefinite programming relaxation approach for the pooling problem," Master's thesis, Department of Applied Mathematics and Physics, Kyoto University, 2010.
- [2] M. Alfaki and D. Haugland, "Strong formulations for the pooling problem," *Journal of Global Optimization*, **56**, pp. 897–916, 2013.
- [3] A. A. Ahmadi and A. Majumbar, "DSOS and SD-SOS Optimization: LP and SOCP-Based Alternatives to Sum of Squares Optimization," In *Proceedings of the 48th Annual Conference on Information Sciences and Systems*, pp. 1–5, 2014.