連結制約と被覆制約を持つ施設配置問題に対する発見的解法

山之内 亮介
南山大学大学院数理情報研究科数理情報専攻
（現所属：東海旅客鉄道株式会社）
指導教員：佐々木美裕 准教授

1. はじめに

本研究では，連結制約と被覆制約を持つ施設配置問題 (CCFLP) をネットワーク上で考え，貪欲算法とけちけち法に基づく発見的解法を用いて，大規模な問題の解を求める．CCFLP では，配置される p 個の施設によってすべての需要点が被覆されること（被覆制約）に加え，任意の需要点間（OD ペア）が施設を経由して連結となること（連結制約）が要求される．目的は，OD ペア間の距離の総和を最小化することである．施設を電気自動車 (EV) の急速充電器と考えると，対象地域内のすべての OD ペア間を EV で往来可能となるように p 個の急速充電器を配置する問題となる [1]．他の応用例として，センサネットワーク構築の最適化などが挙げられる．

2. モデルの説明

V_C, V_D を $V_C \cap V_D = \emptyset$ を満たす点集合，2 つの枝集合 E_C^{all} と E_D^{all} を $E_C^{all} = \{(u,v)|u \in V_C, v \in V_C\}$, $E_D^{all} = \{(u,v)|u \in V_C, v \in V_D\}$ とし，$E_C \subseteq E_C^{all}$, $E_D \subseteq E_D^{all}$ と定義する．$V = V_C \cup V_D$ と $E = E_C \cup E_D$ に対し，無向グラフ $G = (V, E)$ を考える．$W \subseteq V_C$ に対し，$W \cup V_D$ による G の誘導部分グラフを $G_W = (W \cup V_D, E_W)$ とする．任意の無向グラフ $G = (V, E)$ と G の各枝の長さ $l : E \to \mathbb{R}$ で定義されるネットワーク $\mathcal{N} = (V, E)$ 上において，$d_G(u,v) (u \in V, v \in V)$ を $\mathcal{N} = (V, E)$ 上の最短経路の長さとする．ただし，u から v へ到達不可能な場合は，$d_G(u,v) = \infty$ と仮定する．また，すべての OD ペアの集合を $\Pi^{all} = \{(u,v)|u \in V_D, v \in V_D\}$ とする．OD ペアの集合 $\Pi \subseteq \Pi^{all}$ に対して，CCFLP は次のように定式化できる．

[CCFLP]

$$\text{minimize} \quad \sum_{(u,v)\in\Pi} d_{G_W}(u,v) \qquad (1)$$

$$\text{subject to} \quad W \subseteq V_C,\ |W| = p. \qquad (2)$$

G_W が非連結の場合，目的関数値が ∞ となるので，G_W が連結とならなければならない．よって，CCFLP は，W による $G_C = (V_C, E_C)$ の誘導部分グラフが連結であること（連結制約）と，すべての $v \in V_D$ について $\{(u,v)|(u,v) \in E_W, u \in W\} \neq \emptyset$ とすること（被覆制約）を満たす $W \subseteq V_C$ の中で目的関数 (1) を最小化するものを見つける問題となる．ここで，V_C を急速充電器の配置候補点集合，1 回の充電で EV が走行できる限界距離を L として，

$$E_C = \{e \in E_C^{all} \mid l(e) \leq L\}, \qquad (3)$$

$$E_D = \{e \in E_D^{all} \mid l(e) \leq L/2\}, \qquad (4)$$

$$\Pi = \{(u,v) \in \Pi^{all} \mid d_{G_L}(u,v) \leq L/2\} \qquad (5)$$

とすれば，CCFLP は EV 専用急速充電器の配置問題 [1] と等価となる．ただし，G_L は，需要点集合 V_D と枝集合 $\{(u,v)|u \in V_D, v \in V_D\}$ で定義される無向グラフである．

3. 解法

はじめに，貪欲算法とけちけち法に基づく解法を用いて初期解を生成する．$\mathcal{N} = (V, E)$ 上において，対象とする OD ペア (u,v) の数は (5) より $|\Pi|$ であるので，最短経路は全部で $|\Pi|$ 本存在する．このとき，$u \in V_C \subseteq V$ がこれら $|\Pi|$ 本の最短経路を構成する点として使われた回数を $w(u)$ とし，これを各反復において追加または削除する施設の選択基準として使用する．

貪欲算法では，すべての $u \in V_C$ の中で，重要度 $w(u)$ の高い順に W に追加し，連結制約，被覆制約および $|W| \geq p$ を満たすまで W を更新する．

けちけち法では，はじめに $W := V_C$ とし，重要度 $w(u)$ の低い順に，W から $u \in W$ を削除し，$|W| = p$ を満たすまで W を更新する．ただし，$u \in W$ を W から削除することにより誘導部分グラフ G_W が非連結になる場合は削除しない．

貪欲算法の各反復において，最良の評価値を持つ解 1 つだけでなく複数の解を保持し，さらに，配置する施設 $u \in V_C$ を選ぶ基準として，重要度 $w(u)$ に加え

て，被覆数 $c(u)$ を用いる．$c(u)$ は，ネットワーク上で $u \in V_C$ に隣接している $v \in V_D$ の数 $c(u)$ で定義する．以上をまとめると，貪欲算法は以下のように記述できる．けちけち法の詳細については省略する．

[貪欲算法]

入力：$\mathcal{N} = (V, E)$; 出力：$W_{ik} \subseteq V_C$

ステップ 0: $i := 1$ とする．また，すべての $v \in V_D$ に対して $d(v) := |\{(u, v)|(u, v) \in E_D, u \in V_C\}|$ を求め，$F := \{u \in V_C|(u, v') \in E_D, d(v') = 1\}$ とする．

ステップ 1: $W_{ik} := F(k = 1, \ldots, \sigma)$，$C'_i := V_C \backslash F$，$D'_i := V_D \backslash \{v \in V_D|(u', v) \in E_D, u' \in F\}$ とする．また，すべての $u_i \in C'_i$ に対して $c(u_i) := |\{(u_i, v)|(u_i, v) \in E_D, v \in D'_i\}|$ を求める．

ステップ 2: すべての $u_i \in C'_i$ を $c(u_i)$ の大きい順に並べ替え，大きい順に $u_i^1, \ldots, u_i^\sigma$ とする．ただし，$c(u_i)$ の値が同じ $u_i \in C'_i$ が複数ある場合，$w(u_i)$ の値が大きいものを優先する．

ステップ 3: $W_{ik} := W_{ik} \cup \{u_i^k\}$，$C_{ik} := C'_i \backslash \{u_i^k\}$，$D_{ik} := D'_i \backslash \{v \in D'_i|(u_i^k, v) \in E_D\}(k = 1, \ldots, \sigma)$ とする．

ステップ 4: $k := 1$ とする．

ステップ 5: $(\mathcal{N}, W_{ik}, C_{ik}, D_{ik})$ を入力として，[被覆制約を満たす点集合の探索] へ．

ステップ 6: (G_C, W_{ik}) を入力として，[連結制約を満たす点集合の探索] へ．また，$k = \sigma$ ならば，ステップ 7 へ進む．そうでなければ，$k := k + 1$ とし，ステップ 5 へ戻る．

ステップ 7: $W_{ik}(k = 1, \ldots, \sigma)$ の目的関数値をそれぞれ求める．目的関数値の 1 番小さい W_{ik} を保持し，$F := F \cup \{u_i^k\}$ とする．$i = p$ ならば W_{ik} を出力して終了．そうでなければ，$i := i + 1$ とし，ステップ 1 へ戻る．

被覆制約を満たす点集合の探索および，連結制約を満たす点集合の探索では，それぞれの制約を満たすように $u \in V_C$ の中から被覆数 $c(u)$ と重要度 $w(u)$ の高い順に W に追加していく．探索方法の詳細は省略する．

次に，貪欲算法またはけちけち法で得られた初期解を改善するために，W の近傍 $N(W)$ を

$$N(W) = \{W' | W' = W \cup \{u_1\} \backslash \{u_2\}, u_1 \notin W, u_2 \in W, (u_1, v) \in E_D, (u_2, v) \in E_D\}.$$

で定義し，近傍探索とアニーリング法を用いる．アニーリング法の詳細は省略する．

4. 計算実験

需要点と候補点をランダムに配置した 2 つの例題を用いて，提案解法の評価を行う．各例題の需要点数 $|V_D|$，候補点数 $|V_C|$，OD ペア数 $|\Pi|$ は，例題 1 が 30，50，403 であり，例題 2 が 50，80，1159 である．

提案解法は，Microsoft Visual C++ 2010 Express で実装した．また，山之内，保田 [1] のモデルを用いて，最適化ソフトウェア IBM ILOG CPLEX Optimization Studio 12.5 で厳密解を求めた．使用した計算機に搭載されている CPU は Intel Core i7-960 Processor，メモリは 24GB である．

初期解の相対誤差が大きかった例題 1 の $p = 12$，14 と例題 2 の $p = 16$，18 の場合の結果を表 1 に示す．表中の N/A は，解を求められなかったことを表す．

厳密解を求める際にかかった実行時間は，例題 1 の $p = 12$ のとき約 2500 秒，$p = 14$ のとき約 500 秒，例題 2 の $p = 16$ のとき約 18 時間，$p = 18$ のとき約 15 時間であった．いずれの例題においてもアニーリング法を用いることにより解が大幅に改善されたことが確認できる．

表 1　例題 1，2 の実行結果

相対誤差 (%)									
例題	p	初期解				アニーリング法適用後			
		けちけち法	貪欲算法			けちけち法	貪欲算法		
			$\sigma=1$	$\sigma=3$	$\sigma=5$		$\sigma=1$	$\sigma=3$	$\sigma=5$
1	12	2.59	8.48	0.58	0.58	0.19	0.00	0.00	0.00
	14	8.92	6.90	0.43	2.32	0.00	0.00	0.18	0.06
2	16	N/A	38.66	10.51	9.53	N/A	4.60	6.90	3.63
	18	14.74	13.87	0.93	0.26	0.37	1.26	0.31	0.18

実行時間（秒）									
例題	p	初期解				アニーリング法適用後			
		けちけち法	貪欲算法			けちけち法	貪欲算法		
			$\sigma=1$	$\sigma=3$	$\sigma=5$		$\sigma=1$	$\sigma=3$	$\sigma=5$
1	12	0.02	0.01	0.17	0.23	3.79	5.05	4.51	4.52
	14	0.02	0.01	0.20	0.28	4.68	5.84	4.92	5.02
2	16	N/A	0.03	0.47	0.72	N/A	7.37	12.12	7.81
	18	0.02	0.03	0.51	0.83	10.45	8.93	15.92	17.48

参考文献

[1] 山之内亮介，保田将弘，“電気自動車専用急速充電器の最適配置問題，”南山大学数理情報学部 2011 年度卒業論文，2012.