

自販機サプライチェーンに対する配送計画問題

名古屋工業大学大学院・社会工学専攻* 小島貢利(Mitsutoshi Kojima)

愛知工業大学・経営学部** 大野勝久(Katsuhisa Ohno), 田村隆善(Takayoshi Tamura)

Department of Industrial Management Engineering, Nagoya Institute of Technology*

Faculty of Management, Aichi Institute of Technology**

1. はじめに

日本国内における自販機の普及台数は、飲料向けだけでも 260 万台あり、年間自販金額は 2 兆 4 千億円である。たばこ自販機など全ての自販機を含めれば、それらの普及台数は約 400 万台、年間自販金額は 5 兆 3 千億円に達する[1]。各所に点在する自販機への商品の補充は配送車で行うが、補充量は現地に到着しないと確定せず、補充量に応じて積載量も減少し、途中でデポに戻る必要が生じる。したがって、需要が確率的に変動する場合に、配送車がどのような順番で自販機を巡回すれば良いか? は、配送コスト削減や環境負荷低減の観点から重要な問題となってくる。

この問題に類似した問題として SVRP(Stochastic Vehicle Routine Problem)が知られているが、マルコフ決定過程による厳密解法では、配送地点数や品種数の増加とともに、次元の呪いが発生し、実用規模の問題を解くことは事実上不可能であり、ニューロ DP, 近似 DP 等様々な解法が提案されている[2, 3, 4]。

本研究では、需要が確率的に変動する、多品種の商品を補充する自販機サプライチェーンに対する配送計画問題の定式化を行い、実用規模の問題に適用可能な最適化アルゴリズムを提案する。

2. 自販機サプライチェーンに対する配送計画問題

n 個所の自販機に G 品種の商品を配送する問題を考える。各自販機 i の商品 g の単位期間当たりの需要量 D_{ig} は確率的に変動し、 i における g の需要量が k となる確率 $D_{ig}(k)$, $i=1, \dots, n$, $g=1, \dots, G$, $k=1, \dots, K_{ig}$ が与えられるものとする。さらに、

1) 配送には各商品の積載容量 Q_g の車 1 台を使用する。

2) 自販機 $l \rightarrow m$ 間の配送費用 (距離等) は $d(l, m)$ である。ここで、

$l \neq m, l, m \in \{0, 1, \dots, n\}$ であり, 0 はデポを表す.

3) 各商品の需要量は互いに独立であり、配送に訪問したときに確定する.

4) 自販機への配送後, 次にどの自販機に行くか, または, 補充のためにデポに戻るかを決定する.

を仮定する. 問題は, 全ての自販機に配送し終わるまでの総配送費用を最小にする配送政策を求めることである.

ここで自販機 l に着いて補充した後の商品 g の積載量を $q_{lg} \in \{0, 1, \dots, Q_g\}$ とおき, $\mathbf{q}_l = (q_{l1}, \dots, q_{lG})$ とおく. さらに自販機 i における商品 g の残存需要量を $j_{ig} \in \{?, 0, 1, \dots, K_{ig}\}$ とおき, $\mathbf{j}_i = (j_{i1}, \dots, j_{iG})$ とおく. ただし, ? は 1 度も訪れていないため未知である状態を表す.

自販機 l に着いて補充した後のシステムの状態 x を $x = (l, \mathbf{q}_l, \mathbf{j}_1, \dots, \mathbf{j}_n)$ と表し, 状態空間を S とおく. このとき, $\mathbf{0} = (0, \dots, 0)$, $? = (?, \dots, ?)$, $\mathbf{Q} = (Q_1, \dots, Q_G)$ とおけば, 初期状態は $x_0 = (0, \mathbf{Q}, ?, \dots, ?)$, 最終状態は $x_f = (0, \mathbf{Q}, \mathbf{0}, \dots, \mathbf{0})$ と表すことができる. また, 状態 x が取り得る全状態数は $(n+1) \prod_{g=1}^G \left\{ (Q_g + 1) \prod_{i=1}^n (K_{ig} + 2) \right\}$ で与えられる.

4) の決定を表すために, l の次に訪問する自販機を m とおき, いずれかの商品の残存需要量が正のとき, 直接 $l \rightarrow m$ と移動する決定を $a=0$ とおき, 補充のため $l \rightarrow 0 \rightarrow m$ と移動する決定を $a=1$ とおけば, 状態 x で取りうる決定の集合 $U(x)$ は,

$$U(x) = \left\{ \{m \in \{1, 2, \dots, n\} | \mathbf{j}_m \neq \mathbf{0}\} \cup \{0\} \right\} \times \{a : a \in \{0, 1\}\}$$

で与えられる.

状態 x で決定 $u = (m, a)$ をとり, 状態 $x' = (m, \mathbf{q}_m, \mathbf{j}_1, \dots, \mathbf{j}'_m, \dots, \mathbf{j}_n)$ へ推移したときの費用を $g(x, u, x')$ とおけば,

$$g(x, u, x') = \begin{cases} d(l, m) & \text{if } u = (m, 0) \\ d(l, 0) + d(0, m) & \text{if } u = (m, 1) \end{cases} \quad (1)$$

であり, 状態 x' の q_{mg}, j'_{mg} は

$$q_{mg} = \begin{cases} [q_{lg} - D_{mg}]^+, & \text{if } u = (m, 0), j_{mg} = ? \\ [q_{lg} - j_{mg}]^+, & \text{if } u = (m, 0), j_{mg} \neq ? \\ [Q_g - D_{mg}]^+, & \text{if } u = (m, 1), j_{mg} = ? \\ [Q_g - j_{mg}]^+, & \text{if } u = (m, 1), j_{mg} \neq ? \end{cases} \quad (2)$$

$$j'_{mg} = \begin{cases} [D_{mg} - q_{lg}]^+, & \text{if } u = (m,0), j_{mg} = ? \\ [j_{mg} - q_{lg}]^+, & \text{if } u = (m,0), j_{mg} \neq ? \\ [D_{mg} - Q_g]^+, & \text{if } u = (m,1), j_{mg} = ? \\ [j_{mg} - Q_g]^+, & \text{if } u = (m,1), j_{mg} \neq ? \end{cases} \quad (3)$$

で与えられる。ただし、 $[q]^+ = \max(0, q)$ である。また、そのときの遷移確率は

$$P_{xx'}(u) = \prod_{g=1}^G P'_{xx'g}(u) \quad (4)$$

$$P'_{xx'g}(u) = \begin{cases} D_{mg}(k) & \text{if } \begin{cases} j_{mg} = ?, & a = 0, & q_{mg} = [q_{lg} - k]^+, & j'_{mg} = [k - q_{lg}]^+ \\ j_{mg} = ?, & a = 1, & q_{mg} = [Q_g - k]^+, & j'_{mg} = [k - Q_g]^+ \end{cases} \\ 1 & \text{if } \begin{cases} j_{mg} > 0, & a = 0, & q_{mg} = [q_{lg} - j_{mg}]^+, & j'_{mg} = [j_{mg} - q_{lg}]^+ \\ j_{mg} > 0, & a = 1, & q_{mg} = [Q_g - j_{mg}]^+, & j'_{mg} = [j_{mg} - Q_g]^+ \end{cases} \end{cases} \quad (5)$$

で与えられる。このとき、問題は最適性方程式

$$J^*(x) = \min_{u \in U(x)} \sum_{x' \in S} P_{xx'}(u) \{g(x, u, x') + J^*(x')\} \quad x \in S \quad (6)$$

をみたす最小費用 $J^*(x_0)$ と(6)式右辺を最小化する最適政策を求めることである。

3. SBMPIM

本研究では、生産システムの最適制御問題で有効性が示されている、SBMPIM(Simulation Based Modified Policy Iteration Method)[5, 6]を用いた最適性方程式(6)式を解くニューロ DP アルゴリズムを提案する。

Step1 : (初期設定)

初期状態 x_0 , 最終状態 x_f , シミュレーション回数 k_{max} , 収束判定のための $\varepsilon' \geq \varepsilon > 0$, 正整数 c_{max} を定める。 i 台の自販機への配送が完了していない状態の集合を S'_v とおき、シミュレーションで訪問済の状態の集合を $S_v = \{S_v^0, S_v^1, \dots, S_v^n\} = \emptyset$ とおく。さらに、補助的な状態の集合 $S_u = \emptyset$ とおく。

各自販機において需要を各商品の平均需要量の総和でおきかえた、単一品目の VRP を解き[7], 自販機の訪問順を

$$\tau = (0, i_1, i_2, \dots, i_N, 0) \quad (i_j = 0 \text{ デポ も含む}) \quad (7)$$

とおく。ここで自販機の番号を τ に現れる順に $i_1 = 1, i_2 = 2, \dots, i_N = n$ とデポをスキップしてつけ直す。初期政策を

$$f(i, \mathbf{q}_i, \mathbf{0}, \dots, \mathbf{0}, \mathbf{j}_i, \dots, \dots) = \begin{cases} (i, 1) & \mathbf{j}_i \neq \mathbf{0} \text{ のとき} \\ (i+1, 0) & \mathbf{j}_i = \mathbf{0} \text{ かつ } \mathbf{q}_i \neq \mathbf{0} \text{ のとき} \\ (i+1, 1) & \mathbf{j}_i = \mathbf{0} \text{ かつ } \mathbf{q}_i = \mathbf{0} \text{ のとき} \end{cases} \quad (8)$$

で与える.

政策反復の繰り返し回数 $r=1$, シミュレーションの繰り返し回数 $k=1$, 収束判定の繰り返し回数 $c=1$ とおく, さらに, 状態 $x=x_0$, $S_v = \{x\}$ とおく, Step2 のシミュレーションにおいて各反復で訪問する状態の集合を $S_T = \{S_T^1, \dots, S_T^{k_{\max}}\} = \emptyset$, $\bar{J}^0(x_0) = 0$ とおく.

Step2: (シミュレーション (政策 f のもとでの状態 x_0 から状態 x_f へのシミュレーションを k_{\max} 回行う))

k 回目のシミュレーションにおいて訪問した状態の集合を $S_T^k = \{x\}$, 政策 f のもとでの状態 x から状態 x_f への費用を $J^k(x) = 0$ とおく.

与えられた確率分布に従って, 各品目, 各自販機ごとに需要量の乱数を発生させる.

Step2-1 状態 $x = (l, \mathbf{q}_l, \mathbf{i}_1, \dots, \mathbf{i}_n)$ で決定 $f(x) = (m, a)$ をとり, $x' = (m, \mathbf{q}_m, \mathbf{i}'_1, \dots, \mathbf{i}'_n)$, $p(x') = \{\mathbf{i}'_j \neq \mathbf{0} \text{ となる数}\}$ を定める.

Step2-2 $y \in S_T^k$ にたいして費用を

$$J^k(y) = \begin{cases} J^k(y) + d(l, m), & \text{if } a = 0, \\ J^k(y) + d(l, 0) + d(0, m), & \text{if } a = 1. \end{cases} \quad (9)$$

と更新し, $J^k(x') = 0$, $S_T^k = S_T^k \cup \{x'\}$ とおく.

$x' \notin S_v, x' \notin S_u$ ならば $S_v^{p(x')} = S_v^{p(x')} \cup \{x'\}$, $v(x') = 1$, $J(x') = 0$, $f(x') =$ 初期政策 とおく, $x' \notin S_v, x' \in S_u$ ならば $S_v^{p(x')} = S_v^{p(x')} \cup \{x'\}$, $v(x') = 1$, $J(x') = 0$, $f(x') =$ 初期政策, $S_u = S_u - \{x'\}$ とおく.

$x' \in S_v$ ならば $v(x') = v(x') + 1$ と更新し, $x' \neq x_f$ ならば, $x = x'$ とおく, Step2-1 へ.

Step2-3 全ての状態 $y \in S_T^k$ にたいして,

$$J(y) = J(y) + \frac{1}{v(y)} (J^k(y) - J(y)) \quad (10)$$

と更新し, $k = k+1$ とおく, $k \leq k_{\max}$ ならば, $x = x_0$, $J^k(x) = 0$, $S_T^k = \{x\}$ とおく, Step2-1 へ.

Step2.4 k_{\max} 回のシミュレーションで求められた費用 $J^k(x_0)$ の平均値

$$\bar{J}^r(x_0) = \frac{1}{k_{\max}} \sum_{k=1}^{k_{\max}} J^k(x_0) \quad (11)$$

を計算する。

Step3 : (政策改良)

Step3-1 $p = 0, 1, \dots, n$ の順番に, 以下を計算する. $x \in S_v^p$ にたいして

$$J(x) = \min_{u \in N(x, f(x))} \left\{ \sum_{x' \in S} P_{xx'}(u) \{g(x, u, x') + J(x')\} \right\} \quad (12)$$

を計算する. ここで $N(x, f(x))$ は $U(x)$ における $f(x)$ の近傍であり, $P_{xx'}(u) > 0$ となる $x' \notin S_v \cup S_u$ にたいしては $S_u = S_u \cup \{x'\}$ とおき, $f(x')$ = 初期政策, $J(x')$ = 初期政策から決まる値を用いて計算する. $f(x)$ が $J(x)$ を与えなければ, $f(x)$ を(12)式右辺を最小化する決定に改良する.

Step3-2 $x \in S_u$ にたいして

$$J(x) = \min_{u \in N(x, f(x))} \left\{ \sum_{x' \in S} P_{xx'}(u) \{g(x, u, x') + J(x')\} \right\} \quad (13)$$

を計算する. $P_{xx'}(u) > 0$ となる $x' \notin S_v \cup S_u$ にたいしては $J(x')$ = 初期政策から決まる値を用いて計算する. $f(x)$ が $J(x)$ を与えなければ, $f(x)$ を(13)式右辺を最小化する決定に改良する.

Step4 : (収束判定)

$|\bar{J}^r(x_0) - \bar{J}^{r-1}(x_0)| < \varepsilon$ ならば $c = c + 1$ とおき, $c < c_{\max}$ ならば $r = r + 1$, $S_r = \emptyset$, $x = x_0$ とおいてステップ 2 へ. $c \geq c_{\max}$ ならば $\{\bar{J}^r(x_0), \dots, \bar{J}^{r-c+1}(x_0)\}$ の標本分散 σ_s^2 と平均 $\bar{J}(x_0) = \frac{1}{c} \sum_{c'=0}^{c-1} \bar{J}^{r-c'}(x_0)$ を計算し, 自由度 $c-1$ の t 分布の両側 α 点の値を $t_\alpha(c-1)$ としたとき, $t_\alpha(c-1)\sigma_s/\sqrt{c} < \varepsilon$ をみたせば停止. さもなければ, $c = c + 1$, $r = r + 1$, $S_r = \emptyset$, $x = x_0$ とおいてステップ 2 へ.

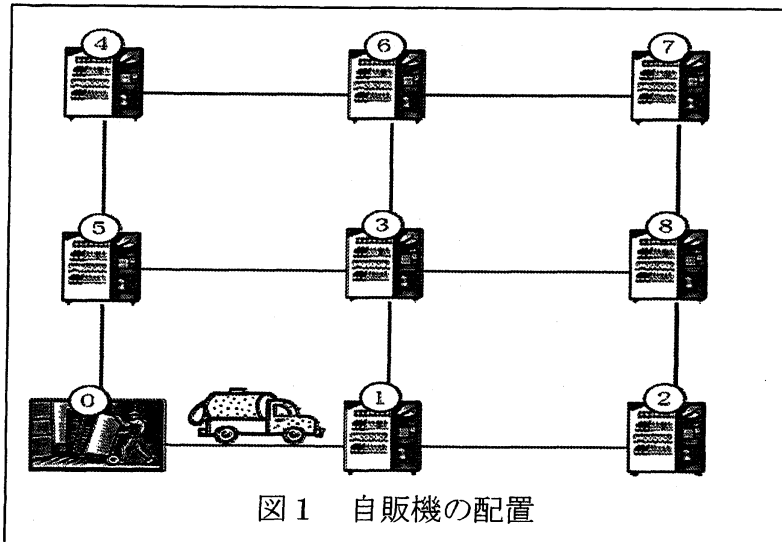
さもなければ $c = 1$, $r = r + 1$, $S_r = \emptyset$, $x = x_0$ とおいてステップ 2 へ.

4. 数値例

$n=8$, $G=1$, $Q_1=6$ とおき, 単一商品の自販機が図 1 のように配置されているものとする.

各自販機の需要量は平均 2 で, 項数 2 のシフトした二項分布に従い, $D_{i1}(1)=0.25$, $D_{i1}(2)=0.5$, $D_{i1}(3)=0.25$, $K_{i1}=3$, $i=1, \dots, 8$ である.

初期の政策および費用は, 文献[7]の CD-ROM に与えられる, VRP の Tabu-Search



プログラムの結果を利用した. 各自販機の需要量が2の場合, VRPの最良解をもとにした訪問順は以下のようになる.

$0 \rightarrow 1 \rightarrow 2 \rightarrow 0 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 0 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 0$

$k_{max}=10000$, $c_{max}=10$, $\varepsilon, \varepsilon' = 1.0 \times 10^{-3}$ とおく. 現在の政策による決定を $u = (m, a)$ とおくと, 以下の4条件の内, 最大3個の政策の近傍 N_i を定義する.

(1) $a=0$ (直接自販機へ移動) かつ配送後の積載量 $q_{11} < Q_1$ (最大積載量) の場合

$N_1 = (m, 1)$ 訪問箇所は同じで, デポを経由する.

(2) $a=1$ (デポ経由) かつ残余積載量 $q_{11} > 0$ の場合

$N_2 = (m, 0)$ 訪問箇所は同じで, 直接移動する.

(3) 配送を完了していない自販機台数が2以上かつ配送後の積載量 $q_{11} < Q_1$ の場合

$N_3 = (m', 1)$ デポを経由する.

ここで, m' は初期政策における, 配送を完了していない m の次の自販機とおく.

(4) 配送を完了していない自販機台数が2以上かつ配送後の積載量 $q_{11} > 0$ の場合

$N_4 = (m', 0)$ 直接移動する.

SBMPIMによる最良解の費用は $J(x_0) = 17.2712$, 政策反復回数は57回, シミュレーションで訪問した状態数は約3700状態であった. なお, この数値例における全状態数は $(n+1)(Q_1+1)(K_{11}+2)^n = 9 \times 7 \times 5^8 =$ 約2500万状態である.

各自販機の需要が2と固定したときのSBMPIMによる最良解の軌跡は以下のとおりである.

$0 \rightarrow 5 \rightarrow 3 \rightarrow 4 \rightarrow 0 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 0$

5. おわりに

本研究では、需要量が確率的に変動する、多品種の商品を補充する自販機サプライチェーンにたいして、品種毎の積載容量を考慮して、総配送費用（距離）を最小にする配送政策を求める配送計画問題の定式化を行った。さらに、SBMPIM[5,6]を用いた、最適な配送政策を決定するアルゴリズムを提案し、簡単な単品種の数値例でその有効性を検証した。

今後の課題は、多品種のより現実的な需要分布に対する検証と当アルゴリズムの現実の自販機サプライチェーンへの適用である。

参考文献

- [1] 日本自動販売機工業会:「自販機普及台数及び年間自販金額 2010年(平成22年)版」, <http://www.jvma.or.jp/information/fukyu2010.pdf> (2011).
- [2] Bertsekas, D.P. and Tsitsiklis, J.N.: *Neuro-Dynamic Programming*, Athena Scientific (1996).
- [3] Secomandi, N.: "Computing neuro-dynamic programming algorithms for the vehicle routine problem with stochastic demands," *Computers & Operations Research* 27 (2000) pp.1201-1225.
- [4] Powell, W.B.: *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, Wiley-Interscience, (2007).
- [5] Ohno, K.: "The optimal control of just-in-time-based production and distribution systems and performance comparisons with optimized pull systems," *European Journal of Operational Research* 213 (2011) pp.124-133.
- [6] 大野:「サプライチェーンの最適運用: かんぱん方式を超えて」, 朝倉書店(2011).
- [7] 大野, 田村, 伊藤:「Excelによるシステム最適化」, コロナ社(2001), pp.143-160.