数理解析研究所講究録 1252 巻 2002 年 69-74

強化学習を用いた確率的な配送経路決定問題の解法

早稲田大学　林准黙, 石川栄一, 吉本一穂
Department of Industrial and Management Systems Engineering,
Waseda University

Abstract

The Vehicle Routing Problem (VRP) plays a central role in the fields of physical distribution and logistics. Generally, the VRP consists of the following elements; one or several depots, customers' locations and their demands, travel times or costs between customers, and one or more vehicles and their capacities. The objective of the problem is to minimize the total distance (time) traveled by all the vehicles. Stochastic Vehicle Routing Problem (SVRP) arises whenever some elements of the problem are random. Common examples are stochastic demands and stochastic travel times. In this paper, the SVRP with stochastic travel times is considered. We present how the Q-learning technique, one of the reinforcement learning methods, can be applied to solve the SVRP with stochastic travel times. And then we propose a new learning based algorithm for the SVRP. The validity of the developed algorithm is examined with some example problems by computer simulation.

Keywords: *Stochastic Vehicle Routing Problem (SVRP), Q-learning*

1. 序　論

現実的な配送問題の欠点として, 末端輸送は自動車に依存せざるを得ず, 大都市物流問題はトラック配送をめぐる問題であるといつても過言ではない. 都市内輸送効率の低下はトラック自身がその原因となっている面もあり, それを補うため更に車両を増備すればする程道路事情が悪化するという悪循環の要素もある. このため大型自動車の都心部通行規制をはじめとする交通規制が強化されてきており, これがまた都市の輸送効率低下に拍車をかけている. このような輸送効率の低下を道路等施設の整備のみによって食い止めようとするのは現実的でなく, 物流業者としては効率的に回ることが可能な配送計画をたて, トラックの台数を抑える努力が必要となる. 現実的な配送計画においては, 渋滞の影響を考慮する必要がある. 過去に渋滞を時間帯別に確定情報として扱った技法はみられるが, より現実的なものにするためには需要地間の走行時間を確率分布で扱う問題を解く必要がある. このような問題においては同一ルート内において任意の需要地間の走行時間が次の需要地間の走行時間に影響を与えることより, 従来の VRP(Vehicle Rooting Problem)で用いられていたモダンヒューリスティック等の技法では解くことが不可能である. 本技法は環境に適応する学習制御の枠組で, 試行錯誤を通じて学習するため, 特に不確実性や計測が困難な未知パラメータが多い場合, 優れた解を発見する可能性がある強化学習の一手法である Q-learning を用いて, 各トラックの需要地の割当て, 配送順序の様々なパターンのシミュレーションを行うことにより, どのパターンを選択すれば短い時間で配送する可能性高いかを学習させたうえ配送経路決定問題の解決を目的とする. よって, 本技法で想定するような需要地間の走行時間が確率分布で与えられるような, 配送のたびごとに時間値が変わる問題を解くことが可能となる.

2. 強化学習(Reinforcement learning)

強化学習とは, 試行錯誤を通じて環境に適応する学習制御の枠組である. 教師付き学習(Supervised learning)とは異なり, 状態入力に対する正しい行動出力を明示的に示す教師が存在しない. かわりに報酬というスカラーの情報を手がかりに学習するが, 報酬にはノイズや遅れがある. そのため, 行動を実行した直後の報酬をみるだけでは, 学習主体はその行動が正しかったかどうかを判断できないという困難を伴う.
強化学習の枠組を図 1 に示す.

学習主体「エージェント」と制御対象「環境」は以下のやりとりを行う

Step1: エージェントは時刻 t において環境の状態観測S(t)に応じて 意志決定を行い,行動 a(t)を出力

Step2: エージェントの行動により,環境はS(t+1)へ状態遷移し,その遷移に応じた報酬r(t)をエージェントへ与える

Step3: 時刻 t を t+1 に進めてステップ1へ戻る

エージェントは報酬(Reward)の総計の最大化を目的として,状態観測から行動出力へのマッピング（方策（policy）と呼ばれる）を獲得する.[1]

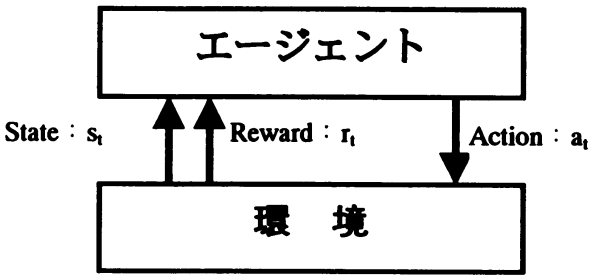

図1　強化学習の枠組み

3. 提案技法

3.1 本技法のモデル

本技法では現実の交通事情を踏まえ,図 2 に示したような,デポ－需要地,需要地－需要地に,幅をもった確率分布による走行時間を与えた際の配送問題の解法を提案する

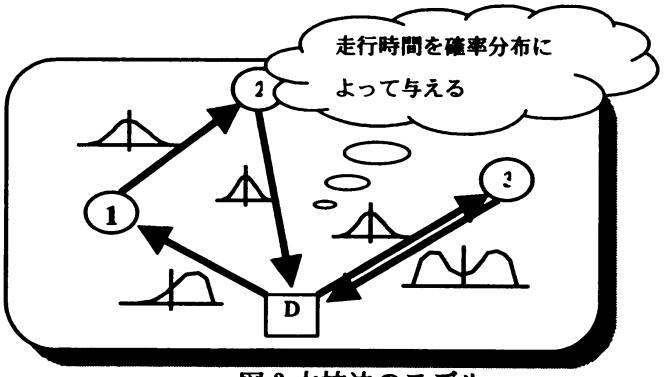

図2本技法のモデル

3.2 本技法の複雑性

本技法は走行時間を確率分布によって与えるため,同一ルート内において前の需要地間における走行時間の影響を受ける.このように従属した環境のもとで配送ルートを構築しなくてはならないことや,同じトラック割当て,配送順序であっても配送のたびごとに実際の走行時間が異なるといった状況は,従来のようなモダンヒューリスティックでは,解を得られるルートを探索することが不可能であることから,従来の確定したデポ－需要地,需要地－需要地の走行距離や走行時間によって解を求めていた技法とは異なった方法が必要となる.

そこで本技法ではシミュレーションを繰返しながら,試行錯誤を行って良い解を学習していくという,強化学習の手法の一つであるQ - Learning を用いることによって,前の配送した需要地の影響と幅をもった時間値を考慮することが可能となる技法を提案する

3.3 本技法の期待する結果

本技法の目的は従来のVRP 問題のように単に固定された距離もしくは時間値を評価関数にしてその総和の最短距離,最短時間のトラック割当て,配送ルートを導きだすといった類のものではなく,確

率分布によるデポ－需要地,需要地－需要地の走行時間の変動を考慮しながら,常に安定して短時間で配送を行う可能性が高いルートを導き出すことにある.

現在のように渋滞によってどの位の時間で配送を行うことが可能なのかを予測が困難であると,確実に配送を行うことができるようにするためには,トラックの積載率を下げ一台の担当需要地数を下げざるを得ない.そこで配送が完了するまでの時間値を正確に予測する事が可能になれば配送計画を遵守できる可能性が高まり,従来よりも効率的な配送計画を行うことが可能となる.よって本技法のように配送計画と実際の配送における時間値のずれ幅が小さいルートを構築することは重要である.

よって本技法では同じルートを繰返し選んだ場合,短時間で配送を行う可能性もあるが,長時間が掛かってしまう可能性もあるというルートではなく,短時間かつ,常に一定の時間値に収まり,分散が小さくなるようなトラック割当て,配送順序が良い解となる.

3.4 制約条件

① デポは1つである
② 1トラックは1ルートを配送する
③ トラックは積載制限をもつ
④ Soft-OTC（走行時間制限）をもつ
⑤ トラックの種類は1種類である
⑥ トラックは1需要地に1回しか立ち寄る事ができない
⑦ 各需要地は1トラックから配送される
⑧ 各需要地での需要量は確定的である

3.5 入力情報

① 需要地数
② 需要量（確定的）
③ 各アークの走行時間(確率分布)

3.6 出力情報

① 配送ルート
② トラック台数
③ 総走行時間の期待値

3.7 本技法におけるQ-learningの要素の定義を以下に示す

(1) 目標

目標は“走行所要時間 + 積載オーバーペナルティ + OTCオーバーのペナルティ”の総和の最小化

(2) 方策

総配送時間＋ペナルティの期待値を最小化するようなActionを選択すること.

(3) State

配送済み需要地: 全てのトラックにおいて,すでに配送し終えた需要地を示す.

対象となるトラックが訪問した需要地:現在配送エピソードのシミュレーション中のトラックが配送をし終えた需要地を示す.

現在地:現在シミュレーション中のトラックがいるデポ,または需要地を示す.

(4) Action

次の需要地を選択する

(5) Reward

走行所要時間 + 積載オーバーペナルティ + OTCオーバーのペナルティ

(6) Q-value

配送完了までのrewardの累積の期待値

確定的な場合

$$\hat{Q}(s,a) \leftarrow r(s,a) + \gamma \cdot \max_{a'} \hat{Q}(s',a') \quad \cdots (1)$$

$$s \leftarrow s'$$

確率的な場合

$$\hat{Q}_n(s,a) \leftarrow (1-\alpha)\hat{Q}_{n-1}(s,a) + \alpha \left[r(s,a) + \gamma \cdot \max_{a'} \hat{Q}_{n-1}(s',a') \right] \quad \cdots (2)$$

α:学習率　　γ:割引率

3.8 本技法のフローチャート

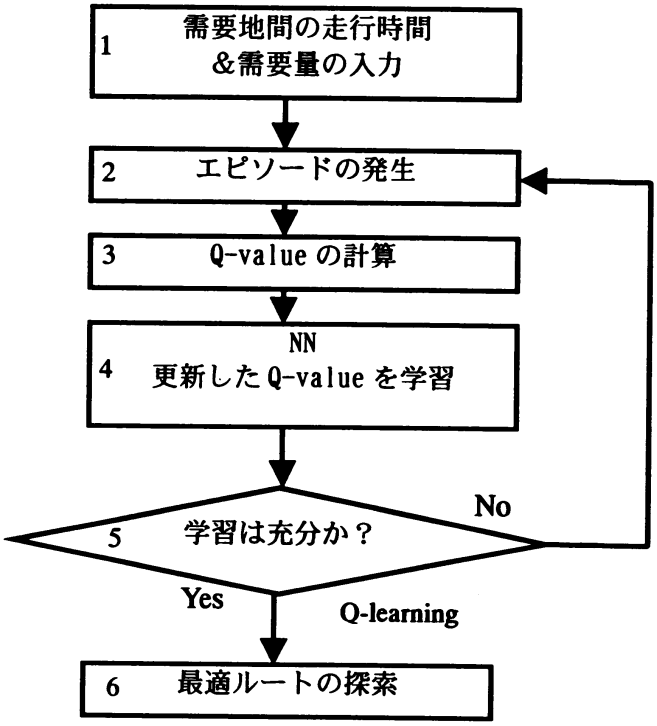

図 3 本技法のフローチャート

Step1:入力情報を与える
　需要地数,需要量,デポ - 需要地,需要地 - 需要地の確率分布による走行時間
Step2:ランダムにエピソードを発生させる(エピソード：トラックの割当て及び訪問順)
Step3:生成したエピソードにより,Q 学習を行う.エピソードに従い,Q - value を計算する
　Q-value の計算方法は以下の通り
　(1)エージェントは環境の状態 s_t を観測する
　(2)エージェントは任意の行動選択方法に従って行動 a_t を実行する
　(3)環境から報酬 r_t を受取る
　(4)状態遷移後の状態 s_{t+1} を観測する
　(5)以下の更新式により Q 値を更新：

$$Q(s_t,a_t) \leftarrow (1-\alpha)Q(s_t,a_t) + \alpha \left[r(s_t,a_t) + \gamma \max_a Q(s_{t+1},a) \right] \quad \cdots (3)$$

　ただし α は学習率($0 < \alpha \leq 1$)，γ は割引率($0 \leq \gamma < 1$)である.
　(6)時間ステップ t を t+1 へ進めて(1)へ戻る
　エピソードが完了するまで繰返す

Step4:Q-value を NN に記憶させておくため NN の学習を行う

Step5:学習が十分に行われたかを判定する.Yes は Step 6 , No は Step 2 に戻りさらに学習を行う
Step6:Q-table より最適ルートを探索する
第 1 期から順に最も小さいQ 値を取るものを選んでいく
第 2 期からは第 1 期の state に続く state の中から最小のQ 値をとるものを選択していく

3.9 Q-value 記憶のための NN

今まで Q-learning が VRP に用いられなかった理由として, 1 State に対して Q-value が 1 to 1 対応している為,様々な制約条件を持つ VRP では state の組合せ数が膨大になってしまい,Q-table の情報を記憶できないということが制約となっていた.本技法の場合で言えば,配送済み需要地の組合せ 2^{2N} 通り,現トラックの組合せ 2^{2N} 通り,現在地 N 通り,アクション N 通りと $N^2 \cdot 2^{3N}$ 通り (N は需要地数) となる. Q-table では,組合せ数をもつ State を表現するための記憶容量は膨大で,多くの需要地数への対応が不可能である.そこで本技法では NN を利用し Q-table に記憶するべき Q-value を state 情報から近似的に読み出せるようにすることで,Q-learning を VRP に適用することを可能にした.

4.実験結果

4.1 Q-Learning の検証

Q-learning と NN を組合せたアルゴリズムが実際,最適解を導き出すかを検証する.確定走行時間において需要地 5 で検証した.

表 1 需要地間の走行時間

	0	1	2	3	4	5
0	0	3	9	5	9	4
1	8	0	4	9	5	5
2	6	7	0	6	2	7
3	2	6	6	0	7	8
4	3	7	6	6	0	5
5	7	6	6	4	9	0

表 2 需要量

	需要量
需要地 1	1
需要地 2	2
需要地 3	2
需要地 4	1
需要地 5	3

需要地数 : 5
積載制限 : 5
Q-learning の繰返し数 : 10000 回
NN の繰返し数 : Q 値の更新 1 回あたり : 1 回
Q-learning の学習率 : 1.0 (確定情報のため)
NN の学習率 : 0.001
中間層数 : 入力層と同数

best_route 0-1-2-4-0-5-3-0 総走行時間 : 22 が最適解となり最適ルートを導き出す事ができた

4.2 確率分布で学習したルートと平均時間で学習したルートとの比較

本技法によって,確率分布で学習を行い算出したルートと平均時間により学習し算出した最適解ルートに,10 回ずつ確率分布による走行時間を与え,配送を行う.
需要地数 : 10
入力情報
需要地間走行時間

表3 確率分布用需要地間の走行時間　　走行時間の確率分布（μ, σ）

	0	1	2	3	4	5	6	7	8	9	10
0		(74,5)	(60,10)	(62,15)	(85,20)	(70,5)	(60,10)	(86,15)	(66,20)	(79,5)	(77,10)
1	(70,15)		(61,20)	(74,5)	(71,10)	(65,15)	(85,20)	(79,5)	(85,10)	(65,15)	(84,20)
2	(63,5)	(69,10)		(79,15)	(63,20)	(60,5)	(74,10)	(80,15)	(63,20)	(77,5)	(68,10)
3	(79,15)	(85,20)	(77,5)		(81,10)	(87,15)	(81,20)	(71,5)	(68,10)	(71,15)	(62,20)
4	(85,5)	(61,10)	(81,15)	(86,20)		(79,5)	(73,10)	(63,15)	(81,20)	(61,5)	(73,10)
5	(74,15)	(85,20)	(64,5)	(84,10)	(90,15)		(77,20)	(72,5)	(74,10)	(82,15)	(80,20)
6	(62,5)	(87,10)	(78,15)	(71,20)	(61,5)	(63,10)		(88,15)	(69,20)	(70,5)	(63,10)
7	(66,15)	(73,20)	(72,5)	(79,10)	(75,15)	(81,20)	(62,5)		(85,10)	(69,15)	(85,20)
8	(89,5)	(61,10)	(70,15)	(62,20)	(69,5)	(71,10)	(89,15)	(86,20)		(63,5)	(75,10)
9	(77,15)	(84,20)	(66,5)	(82,10)	(82,15)	(76,20)	(62,5)	(64,10)	(62,15)		(66,20)
10	(61,5)	(63,10)	(88,15)	(74,20)	(77,5)	(63,10)	(69,15)	(73,20)	(71,5)	(75,10)	

表4 走行時間シミュレーションの比較結果

	確率分布	平均時間
min	777	648
Max	793	960
平均	784.9	789.5
分散	23.0	10743.2

本技法を用い，確定走行時間で学習した最適ルートに確率分布による走行時間を与え算出したものと，確率分布により学習した最適ルートにシミュレーションを行ったものとの走行時間を比較したものが上の表＊である．平均時間の確定情報によって求めたルートは分散による変動が考慮されていないため，短時間で走行できる場合がある一方，分散が 10743.2 と大きく，時間が掛かってしまう可能性が高くなる．あらかじめ確率分布によってもとめたものは，様々なトラックの割当て，配送順の中で短時間で配送を行われたパターンの確率の高いものを学習していくため，分散が 23.0 と小さく安定している．よって，ある一定の時間内に配送を完了させるためには予め確率分布によって学習したパターンに従い配送を行えば，最大でも 793 で配送を完了させることができ，648 で配送完了するかもしれないが，960 かかってしまう可能性があるパターンより配送計画が行いやすくなる

5. 結論

Q learning を VRP に適用することにより，従来のモダンヒューリスティックでは解くことが出来なかった，走行時間が確率分布によって与えられるような幅をもった配送問題を解くことが可能となった．このような問題が解けるようになったのは Q-learning がシミュレーションによって，同じパターン(トラック割当て，配送順序)を何度も繰返し行うからであり，今までのようにいったん探索したルートを見直さない，すべてのパターンの探索を行わないヒューリスティック解法では解けなかったものである．

あらかじめ時間の幅を考慮して問題を解くことは，平均値で計算する場合に比べ，走行時間を大幅に短縮するわけではないが，確率分布の分散の違いによって次の需要地間の走行時間に与える影響を低く抑えるように短い走行時間で走行した確率の高いルートをなるべく選ぶようにパターンを選ぶため，分散値が大きく時間が掛かる可能性が高いパターンを選びにくくなり安定した時間で配送を行うことが可能になることが示された

〈参考文献〉

[1] Richard S. Sutton, Andrew G. Barto, **Reinforcement Learning.**
[2]萩原　将文, **ニューロ・ファジィ・遺伝的アルゴリズム.** 産業図書, 1994.
[3] Kaelbling, L. P., Littman, M. L. and Moore, A. W., "Reinforcement Learning: A Survey," ***Journal of Artificial Intelligence Research***, Vol. 4, 1996.