

Fast Runcurve Optimization based on Markov Decision Process

Fujii, S.; Yoshimoto, K.; Ueda, K.; Takahashi, S.; Nikovski, D.

TR2013-003 January 2013

Abstract

This paper proposes a fast algorithm to solve runcurve optimization problem. We model the problem with Markov decision process in which cost function is defined by a weighted sum of energy consumption and running time, select appropriate weight values, and compute optimal runcurve with dynamic programming. We have confirmed that the proposed algorithm could optimize runcurve of 2,000m within around 1 sec by simulation results.

Symposium of the Society of Instrumentation and Control Engineers of Japan (SICE)

This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.

マルコフ決定過程を用いた列車走行曲線最適化の高速化

○三菱電機 藤井 彩恵, 吉本 剛生, 上田 健詞, 高橋 理

MERL Daniel Nikovski

Fast Runcurve Optimization based on Markov Decision Process

Sae Fujii, Koki Yoshimoto, Kenji Ueda, Satoru Takahashi, Mitsubishi Electric

Daniel Nikovski, MERL

Abstract This paper proposes a fast algorithm to solve runcurve optimization problem. We model the problem with Markov decision process in which cost function is defined by a weighted sum of energy consumption and running time, select appropriate weight values, and compute optimal runcurve with dynamic programming. We have confirmed that the proposed algorithm could optimize runcurve of 2,000m within around 1 sec by simulation results.

Keywords: Runcurve optimization, Markov decision process, Dynamic programming

1 はじめに

近年、鉄道の省電力化、低炭素化の必要性が世界各国で求められている。例えば、日本国内では、国土交通省がエコレールラインプロジェクトにおいて、2030年に鉄道の電力消費量を2010年比で2割削減することを目標とすることを発表している。また、欧州ではERRAC（欧州鉄道研究諮問評議会）が2030年までに1990年比で電力消費量を3割削減することを目標としており、RailEnergyプロジェクト¹⁾などで研究開発が進められている。

鉄道の省エネを実現する上で、駅間の走行時分を守りつつ、消費電力量が最小となる列車の運転を実現するような走行曲線（以降、最適走行曲線と呼ぶ）を導出する走行曲線最適化問題が注目されている。ここで、走行曲線とは、列車の位置と速度の関係を表す曲線である。走行曲線最適化問題の解法には解析によるアプローチと、数値計算によるアプローチとに分類される²⁾。解析によるアプローチでは、最適走行曲線となる条件を満たす方程式を解くことで、走行曲線を最適化する^{3,4)}。これらの手法は加速、惰行、減速のみで走行する、トンネルを考慮しないといったシンプルな列車モデルや路線モデルを仮定しており、複雑な列車モデルや路線モデルには適用できない。数値計算によるアプローチでは、走行時分と消費電力量を重み付けした評価関数を最小化する走行曲線を遺伝アルゴリズムにより求める手法⁵⁾や、多目的進化型アルゴリズムを用いる手法⁶⁾、走行状態遷移のタイミングを勾配法により導出する手法⁷⁾などが提案されている。これらの手法は高速に走行曲線を求めることができるが、最適走行曲線とならない可能性がある。一方、数値計算によるアプローチで最適走行曲線を求める方法として、動的計画法を用いた手法が提案されている^{8,9)}。この手法で

は、走行曲線最適化問題を多段決定問題で定式化し、時間、位置、速度で定義される状態空間を格子状に分割する。そして各状態の最適な制御入力（ノッチ）を導出する。状態空間の分割数を大きくすればするほど、より最適に近い走行曲線が得られるが、演算の対象となる状態数が膨大となり、計算時間を要するといった課題がある。

本稿では、状態空間の分割数を小さくし高速に最適走行曲線を導出するために、走行曲線最適化問題をマルコフ決定過程でモデル化し、動的計画法を用いて最適走行曲線を高速に導出する手法を提案する。マルコフ決定過程は、一般に確率システムの動的な最適化問題の定式化に用いられるモデルである。提案方式では、ある状態で加速、等速、惰行、減速のいずれかの走行モードで遷移する状態を複数保持し、これらの状態への遷移を確率的に表現する。また、提案手法では、最適化の対象となる関数を消費電力量と走行時分の重み付き和で表現し、適切な重みを探索することで、状態空間を位置と速度のみで定義し、演算の対象となる状態数を削減する。提案手法をシミュレーションにより評価し、駅間約2,000mの最適走行曲線を約1秒以内に最適化できることを確認した。

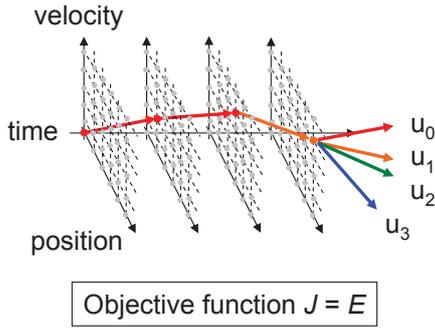
2 走行曲線最適化問題

時刻 t における列車の位置を $x(t)$ 、列車の速度を $v(t)$ とすると、列車の運動は式 (1), (2) により定義される。

$$\frac{dx(t)}{dt} = v(t) \quad (1)$$

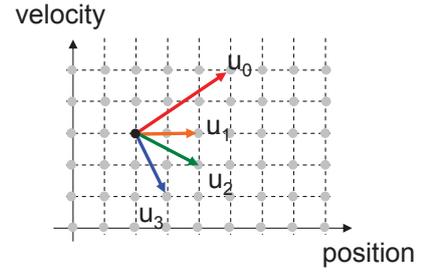
$$\frac{dv(t)}{dt} = \frac{f(v(t), u(t))}{W} - r(v(t)) - g(x(t)) \quad (2)$$

ここで、 $u(t)$ は時刻 t における走行モードを表す。本稿では、走行モードを加速、等速、惰行、減速の4つとし、



Objective function $J = E$

(a) 従来方式



Objective function $J = \mu E + (1 - \mu)T \quad 0 \leq \mu \leq 1$

(b) 提案方式

図 1: 提案手法における走行曲線最適化問題のモデル化

u_0, u_1, u_2, u_3 で表す。よって, $u(t) \in \{u_0, u_1, u_2, u_3\}$ である。また, W は列車重量, 関数 f はモータやブレーキによる駆動力および制動力, 関数 r は走行抵抗による加速度, 関数 g は勾配による加速度を表す。ある時刻 t における瞬間消費電力を次式で定義する。

$$p(v(t), u(t)) = \begin{cases} \frac{f(v(t), u(t))v(t)}{e_a(v(t))} & (\text{力行時}) \\ e_b(v(t)) \cdot f(v(t), u(t))v(t) & (\text{回生時}) \end{cases}$$

ただし, $e_a(v(t))$ はモータ効率, $e_b(v(t))$ は回生効率を表す。

よって, 時刻 0 からある時刻 T までの列車の消費電力量は次式により定義される。

$$E(T) = \int_0^T p(v(t), u(t)) dt \quad (3)$$

本稿で扱う走行曲線最適化問題は, 目標走行時分 T と走行距離 L のもとで, 消費電力量を最小化するような走行モード $u(t) (0 \leq t \leq T)$ を求める問題と定義できる。

$$\min_{u(t) (0 \leq t \leq T)} E(T)$$

subject to formula (1), (2), $x(0) = 0, x(T) = L,$

$$v(0) = 0, v(T) = 0, 0 \leq v(t) \leq V_{max}$$

ただし, V_{max} は列車の最高速度とする。

3 マルコフ決定過程

マルコフ決定過程は, 確率システムの動的な最適化問題の定式化に用いられるモデルであり, 次の要素により構成される。

- S : 状態の集合
- U : 行動の集合
- $Pr(s'|s, u)$: 状態 $s \in S$ で行動 $u \in U$ により状態 $s' \in S$ へ遷移する確率
- $R(s, u)$: 状態 $s \in S$ において行動 $u \in U$ で遷移する際のコスト

最適化の対象をコストの合計値で定義し, コストの合計

値を最小化する各時刻での行動を求める。まず, すべての状態 $s \in S$ について次式で表される価値関数 $V(s)$ と, 価値関数を最小化する行動 u を求める。

$$V(s) = \min_{u \in U} \{R(s, u) + \sum_{s' \in S} Pr(s'|s, u)V(s')\} \quad (4)$$

$V(s)$ を求める方法には, 政策反復法や価値反復法がある。本稿で用いる価値反復法では, すべての状態 $s \in S$ について $V(s)$ を初期化し, 動的計画法を用いて $V(s)$ の値が収束するまで式 (4) を繰り返し計算する。次に, 初期状態から価値関数を最小化する行動と, その行動により遷移した状態の導出を, 終了状態に到達するまで繰り返す。

4 提案手法

4.1 走行曲線最適化問題のモデル化 図 1 に本稿における走行曲線最適化問題のモデル化の概要を示す。本稿では, 状態空間の離散化が荒い場合においても最適走行曲線の導出するために, ある状態である走行モード (加速, 等速, 惰行, 減速のいずれか) で走行した場合に遷移する状態を複数保持し, これらの状態への遷移を確率的に表現したマルコフ決定過程により走行曲線最適化問題をモデル化する。直感的に考えれば, 最適化の対象が消費電力量であるため, 図 1(a) のようにコスト関数を消費電力量とし, 状態空間を位置, 速度, 時刻で定義することが考えられる。しかし, 状態数が位置方向, 時間方向および速度方向の分割数の積となり, 状態空間の離散化が粗い場合でも状態遷移確率や価値関数の収束計算に非常に時間がかかる。

ここで, 一般に走行時分と消費電力量は図 2 に示すように, 走行時分が長いほど消費電力量が減少する関係となる。提案手法では, 図 1(b) のように走行曲線最適化問題を消費電力量と走行時分の重み付け和を最小化するようなモデル化により, 目標走行時分と一致するような適切な重みを探る必要があるが, 探索回数が数回程度に抑えられるならば, 状態空間を時刻と位置と速度で定義する場合と比べて, 演算時間の短縮が期待できる。

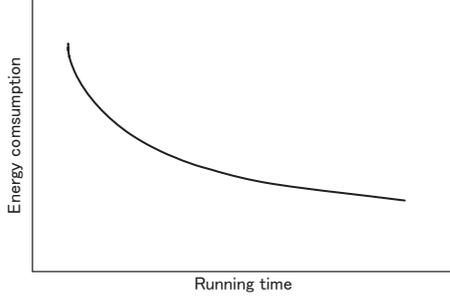


図 2: 走行時分と消費電力量の関係

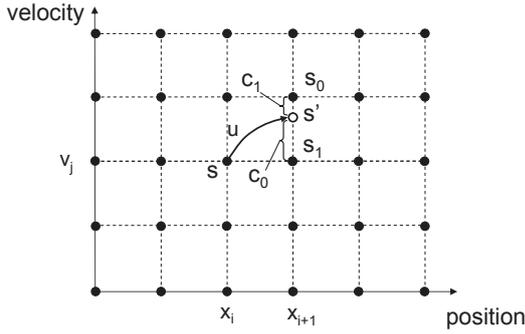


図 3: 提案手法における状態遷移

以下、提案手法におけるマルコフ決定過程に基づく走行曲線最適化問題のモデル化を述べる。まず、状態空間を位置と速度で定義し、各状態を位置方向および速度方向で状態空間を $M \times N$ に格子状に分割することで定義する。そして、状態 $s_{i,j}$ を $s_{i,j} = [x_i, v_j]$ ($0 \leq i \leq M, 0 \leq j \leq N$) とする。行動は列車の走行モードとし、 $U = \{u_0, u_1, u_2, u_3\}$ とする。ある状態 $s_{i,j} = [x_i, v_j]$ ($0 \leq i < M, 0 \leq j \leq N$) において行動 $u \in U$ での遷移先の状態 s' は、 $x = x_{i+1}$ へ到達するまで走行モード u で走行した場合の到達した状態とし、図 3 に示すように $x = x_{i+1}$ 上の 2 つの状態 $s_0, s_1 \in S$ の凸結合とし次式で表す。

$$\begin{aligned} s' &= c_0 s_0 + c_1 s_1 \\ c_0 + c_1 &= 1, 0 \leq c_0 \leq 1, 0 \leq c_1 \leq 1 \end{aligned} \quad (5)$$

そして、状態 s_0, s_1 への遷移確率 $Pr(s_0|s, u)$, $Pr(s_1|s, u)$ を係数 c_0, c_1 とする。状態 s において行動 u で遷移した際のコストは $R(s, u)$ は、次式のように状態 s' に遷移するのに要した消費電力量 $E(s, u)$ と走行時分 $T(s, u)$ の重み付け和とする。

$$R(s, u) = \mu E(s, u) + (1 - \mu) T(s, u) \quad (6)$$

なお、消費電力量 $E(s, u)$ は次式により定義される。

$$E(s, u) = \int_0^{T(s, u)} p(v(t), u(t)) dt \quad (7)$$

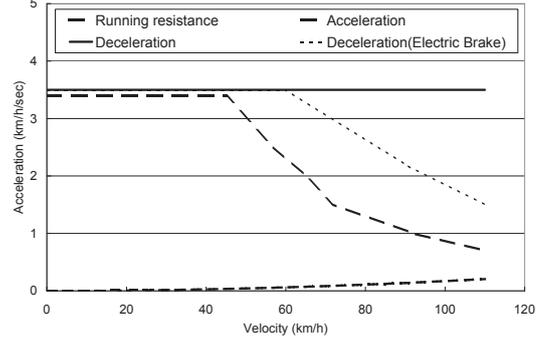


図 4: 加減速性能と走行抵抗

ただし、 $s = [x_i, v_i]$ とした場合、 $x(0) = x_i, x(T(s, u)) = x_{i+1}, v(0) = v_i$ とする。

このようにモデル化することで、 $x = x_i$ 上の状態 s の価値関数 $V(s)$ は $x = x_{i+1}$ 上の状態 s' の価値関数 $V(s')$ にのみ依存して決定されるため、 $x = x_M, x_{M-1}, \dots, x_0$ 上の各状態の価値関数を順に計算することで、各状態の価値関数の値の計算は 1 回となる。

4.2 提案手法のアルゴリズム 前節でモデル化した走行曲線最適化問題を動的計画法により解く。導出の対象となる走行曲線を状態列 $\{s^{(0)}, s^{(1)}, \dots, s^{(l)}\}$ ($s^{(k)} = [x, v]$) で定義する。ただし、 $s^{(0)} = [0, 0]$, $s^{(l)} = [L, 0]$ とする。

1. 状態 $s_{i,j} \in S$ の遷移先の状態 $s' \in S$ と、状態 s' への状態遷移確率 $Pr(s'|s_{i,j}, u)$ を計算する。
2. 状態 $s_{i,j} \in S$ の価値関数 $V(s_{i,j})$ を ∞ で初期化する。ただし $V(M, N)$ のみ 0 とする。ある $\mu (0 \leq \mu \leq 1)$ のもとで、各状態 $s_{i,j}$ の価値関数の値 $V(s_{i,j})$ と $V(s_{i,j})$ を最小化するモード (最適走行モード) $u \in U$ を計算する。
3. ある状態 $s^{(k)}$ の最近傍の状態 $s_{i,j} \in S$ の最適走行モード $u \in U$ で $x = x_{i+1}$ 上に遷移した場合の状態 $s^{(k+1)}$ を求める。以上の処理を終了状態 $s^{(l)}$ に到達するまで繰り返す。
4. 走行曲線 $\{s^{(0)}, s^{(1)}, \dots, s^{(l)}\}$ の走行時分が与えられた目標走行時分 T と一致する場合は処理を終える。一方、異なる場合には、新しい $\mu (0 \leq \mu \leq 1)$ を設定し、処理 2 へ戻る。

なお、目標走行時分 T と一致するような μ の探索方法として、二分探索法など¹⁰⁾が利用できる。

5 提案手法の性能評価

提案手法の性能をシミュレーションにより評価した。計算機の CPU スペックは Intel Core-i5 (2.7GHz)、メモリは 4GB とする。はじめに、文献⁹⁾と同様のパラメータ設定のもとで、走行曲線を作成した。列車総重量は 587t、列車長は 200m とし、加速性能、減速性能ならびに走行

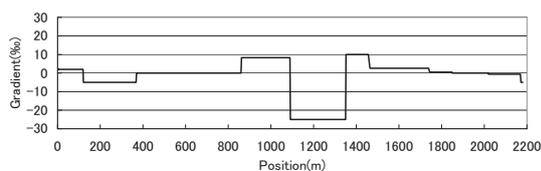


図 5: 勾配

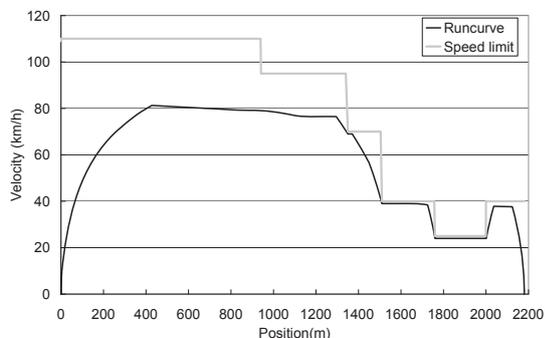


図 6: 走行曲線と制限速度

表 1: 走行曲線作成結果

	走行時分	消費電力量	計算時間
提案方式	168 sec	14.51 kWh	0.7 sec
従来方式 ⁹⁾	168 sec	14.97 kWh	34 sec

抵抗は図 4 に示すとおりとした。また、モータ効率 e_a および回生効率 e_b は定数とし、それぞれ 0.917, 0.849 とする。状態空間は 500×100 (位置方向, 速度方向) で分割する。走行距離 2,180m を目標走行時分 168sec で走行した場合の勾配を図 5, 走行曲線と制限速度を図 6 に示し、従来方式⁹⁾ との比較結果を表 1 に示す。

文献⁹⁾ と列車走行模擬および計算機スペックが異なるため、厳密な比較ではないが、表 1 に示す結果より、ほぼ同等の消費電力量の走行曲線を高速に作成できることを確認した。また、提案方式では走行時分と消費電力量の重みの探索が 5 回であった。以上より、提案方式では走行曲線最適化問題をマルコフ決定過程でモデル化し、評価関数を走行時分と消費電力量の重み付け和で定義することで、演算対象となる状態数を削減し、高速な最適走行曲線の導出を実現している。

6 まとめと今後の課題

本稿では、走行曲線最適化問題をマルコフ決定過程によりモデル化し、状態空間の離散化が粗い場合においても高速に最適な走行曲線を作成するアルゴリズムを提案した。また、提案手法では、価値関数を消費電力量と走行時分の重み付け和とすることで、演算対象となる状態数を削減している。提案手法をあるシナリオのもとで評価

し、1秒以内で約 2,000m の最適な走行曲線を導出できることを確認した。

提案手法の課題は、状態空間を位置と速度で定義しているため、時間によって変動する制約(例えば、ある時間帯での臨時速度制限など)が考慮できない点である。今後はこれらの課題を解決するようにアルゴリズムを改良する予定である。

参考文献

- 1) Railenergy: <http://www.railenergy.eu/>
- 2) Y. Wang, et al: A survey on optimal trajectory planning for train operations, In Proc. IEEE SORI, 589–594 (2011)
- 3) P. G. Howlett and J. Cheng: Optimal driving strategies for a train on a track with continuously varying gradient, J. Aust. Math. Soc., Series B, vol.38, 388–410 (1997)
- 4) P. G. Howlett and A. Leizarowitz: Optimal strategies for vehicle control problems with finite control sets, Dynamics of Continuous, Discrete and Impulsive Systems Series B: Applications & Algorithms, vol.8, 41–69 (2001)
- 5) Y. Bocharnikov, et al: Optimal driving strategy for traction energy saving on suburban railways, IET Electr. Power Appl., vol.1, no.5, 675–682 (2008)
- 6) R. Chevrier: An evolutionary multi-objective approach for speed tuning optimization with energy saving in railway management, In Proc. IEEE ITSC, 19–22 (2010)
- 7) T. Albrecht, et al: Dealing with operational constraints in energy efficient driving, In Proc. IET RTS, 1–7 (2010)
- 8) H. Ko, et al: Application of dynamic programming to optimization of running profile of a train, In Proc. COMPRAIL, 103–112 (2004)
- 9) 高英聖, 古関隆章, 宮武昌史: 動的計画法を用いた列車運転曲線最適化問題の求解法, 電気学会産業応用部門大会, 271–276 (2006)
- 10) S. A. Teukolsky, et al: Numerical recipes 3rd edition: The art of scientific computing, William H. Press