

災害復旧期の交通需要予測のための 時間変化するパラメータの リアルタイム推定手法

望月 陽介¹・浦田 淳司²

¹学生会員 東京大学 工学部社会基盤学科 (〒113-8654 東京都文京区本郷 7-3-1)

E-mail: mochizuki@bin.t.u-tokyo.ac.jp

²正会員 東京大学大学院 工学系研究科社会基盤学専攻 (〒113-8654 東京都文京区本郷 7-3-1)

E-mail: urata@bin.t.u-tokyo.ac.jp

本研究では、災害復旧期の交通需要予測に向けたリアルタイムのパラメータ推定手法を開発した。最適化問題において計算時間を大幅に短縮可能な surrogate-based optimization のアルゴリズムをベースに、アクティビティシミュレータのパラメータ推定を行うアルゴリズムを開発し、実装を行った。提案アルゴリズムでは、微分値を学習して勾配降下法の採用、Online Transfer-learning の採用、学習データの範囲制限の三つの拡張を導入した。本研究では、提案アルゴリズムの有効性を、簡易なアクティビティシミュレータにより、示した。また、2016年熊本地震の状況を想定した地方都市圏を対象としたアクティビティシミュレータに、提案アルゴリズムを適用し、ごく短時間でパラメータ推定を行えることを確認した。

Key Words: 数理最適化, サロゲート最適化, アクティビティシミュレータ, 災害復旧期制御

1. はじめに

(1) 背景と目的

行動を継続的に観測できる常時観測機器の登場により、近年では災害時や復旧期の行動も継続的かつパッシブに観測することが可能になっている。こうした観測から桑原ら¹⁾は、近年の大規模な災害時では平常時と比べて、その発生時だけでなく発生直後から数日間の復旧期に渡り、人々の行動が変化することを明らかにした。こういった変化は、避難や物資の補給といった、災害による直接的な行動需要の発生その他、道路や線路の寸断、渋滞によるネットワーク状態の変化としても現れ、復旧期では日々時間経過により状況が変化していく。これに対し、物資の配分や代替交通手段の用意、沿道利用の制御といった、復旧期の交通制御施策を、復旧期の変化する行動に適した形で動的に行うことが検討・実施されている。適切な動的制御や施策実施のためには、災害復旧期において、時々刻々と変化する活動需要を予測し、対応した制御・施策を行うことが必要である。

被災者と非被災者双方の行動を予測することで、交通や物資の制御に必要な、復旧期の交通需要を予測しよう

と試みたのが、浦田ら²⁾が開発した、災害復旧期におけるアクティビティシミュレータである。本シミュレータでは、復旧期特有の行動を生成するシミュレータと、日常行動を生成するシミュレータ³⁾の二つを統合的に用いることで、復旧期の行動需要を既存シミュレータより高精度の予測を実現した。復旧期の行動予測や行動観測から得られた数値に基づき、上記の交通制御施策を行うことで、その施策の効果を高めることができる。一方で、シミュレータの行動生成はパラメータにより記述されており、復旧期を通して一つの値のパラメータのみを用いる限り、同じ行動傾向から発生する行動需要を出力する。日々行動選好が変わりうる災害復旧期の再現には、変化を取り込んだパラメータインプットが望ましい。本研究の目的は、復旧期の交通制御の際にシミュレータを用いた予測を行うことを念頭においた、パラメータ推定手法の開発である。

(2) 本研究の特徴

高計算コストのシミュレータを用いた最適化を高速化するアルゴリズムとして、Surrogate-based Optimization⁴⁾のような機械学習モデルを用いた最適化アルゴリズムが挙

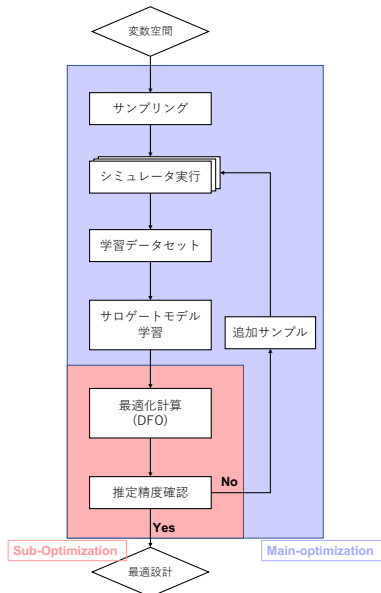


図-1 一般的な Surrogate-based Optimization のフロー (Zhong-Hua Han & ke-shi Zhang (2012)⁹⁾を一部改変)

げられる。しかしオンラインで繰り返し最適化を行うことを想定しておらず、かつ観測データを用いるアクティビティシミュレータのパラメータ推定には対応していない。本研究では Surrogate-based Optimization の構造を用いて災害復旧期にアクティビティシミュレータのパラメータ推定をリアルタイムに行える推定アルゴリズムを開発する。Surrogate-based Optimization を用いた一般的な最適化フローを図-1 に示す。最適化対象の関数を近似する機械学習モデル(サロゲートモデル)の構築と、モデルを用いた最適化を行う。サロゲートモデルを用いて最適化の計算を行う一連の操作を Main-optimization とする。Main-optimization では、サロゲートモデルの近似精度が低い場合に、追加サンプルによる再学習を行う過程を繰り返すことで精度保証を行う。しかし、本研究では、リアルタイムでのパラメータ推定を目指しており、追加サンプルによる精度保証アルゴリズムを導入できない。

提案するパラメータ推定手法では、災害復旧期のアクティビティシミュレータに適用するため、パラメータ推定への有効性を検証した上で3つの拡張を施す。拡張はそれぞれ、扱う問題に Surrogate-based Optimization を適用する際の課題を解決する。

- 拡張1 損失関数微分値を機械学習モデルで予測し、勾配降下法を採用
- 拡張2 Online Transfer-learning⁹⁾による機械学習モデルの更新
- 拡張3 機械学習モデルの学習データ範囲制限

課題1. 交通行動シミュレータのランダム性

課題2. 毎時のパラメータ推定

課題3. 観測値を得た直後に、高精度の推定パラメータを得るためには追加サンプリングを行えない

開発したアルゴリズムはアクティビティシミュレータ SPACE²⁾での実装を行うことで、復旧期の実用可能性を検証する。

2. 提案アルゴリズム

本章では、提案アルゴリズムを示す。はじめに全体のフローを示してアルゴリズムで行う操作の流れを整理した後、個別の操作について具体的な処理を説明する。

(1) 全体の推定フロー

提案アルゴリズムの適用先では、時間的に連続してパラメータ推定を繰り返す状況を想定している。そのため提案手法では機械学習モデルと推定パラメータをオンラインで更新することで、新しい推定パラメータを得る。この流れを示したのが図-2 である。第 t 期の機械学習モデル $Model_t$ と推定パラメータ β_t^* が得られている状態を想定し、第 $t+1$ 期のパラメータ推定を行う流れを示している。ここで1期は観測値を入手する時間間隔を表し、4章で行う復旧期シミュレータのパラメータ推定では1期は1時間に相当する。機械学習モデルを $Model_t$ から $Model_{t+1}$ へと新しく学習させる段階を学習段階、推定パラメータを β_t^* から β_{t+1}^* へと更新する段階を更新段階と呼ぶ。ここで、train data は機械学習モデルを学習させるために生成するデータ群であり、 $Loss_{t+1}$ はタイムステップ $t+1$ 時の観測値 $observe_{t+1}$ を正解値とする損失関数である。提案手法では推定に損失関数の微分値を用いるため、機械学習モデルからは $\frac{\partial Loss(\beta, observe)}{\partial \beta}$ を出力することを示している。

実際に行う手順は、図-3 に示した。提案手法は、生成したデータ群を学習データに使い機械学習モデルの学習を行う学習段階と、機械学習モデルの出力を用いてパラメータの更新を行う更新段階の二つで構成する。提案アルゴリズムの構成は、図-1 に示した Surrogate-based Optimization の構成の、追加サンプルを行わない場合と同じであることがわかる。学習段階と更新段階を、観測値の入手と同じ期間で每期繰り返すことで毎期の推定パラメータを得る。本アルゴリズムは、機械学習モデルやモデルの学習対象、パラメータ推定に用いる最適化アルゴリズムに関わらず推定を行うことができる。

- (2) 個別操作
 - a) 学習段階

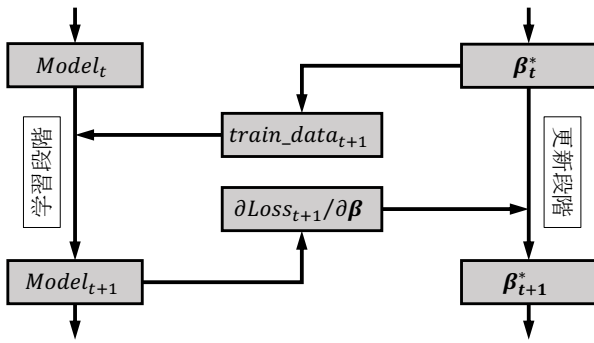


図-2 t期におけるパラメータ・学習モデルの更新

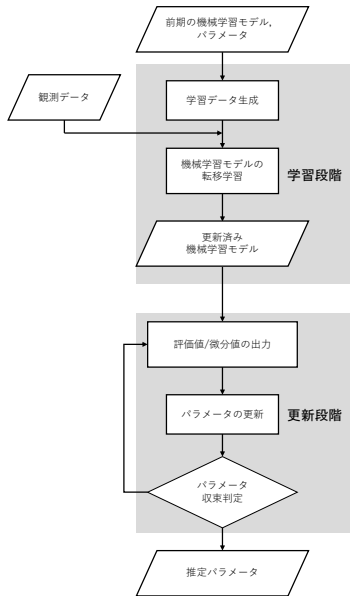


図-3 提案アルゴリズムのフロー

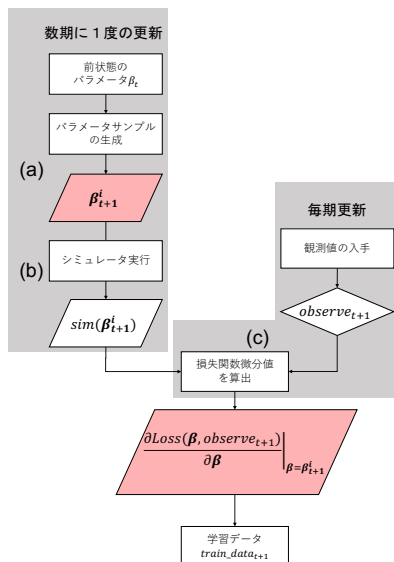


図-4 学習データ作成手順

学習段階では、 t 期の機械学習モデル $Model_t$ を、 $t + 1$ 期の機械学習モデル $Model_{t+1}$ に更新する。前期の機械

学習モデルは、前期での観測データ $observe_t$ を正解値とする損失関数の微分係数 $\frac{\partial Loss(\beta, observe_t)}{\partial \beta}$ を出力する、

$t + 1$ 期の観測データ $observe_{t+1}$ は前期の観測データから変化していることが想定され、その場合損失関数も変化している。 $t + 1$ 期の観測データを正解値とする損失関数の微分係数 $\frac{\partial Loss(\beta, observe_{t+1})}{\partial \beta}$ を精度良く予測するため、

機械学習モデルの更新を行う。機械学習モデルの更新は、既存のモデルを新しい学習データに対して学習させる転移学習と、時間的に連続して入手できる学習データに対して学習させるオンライン学習の、両方を取り入れる Online Transfer-learning の枠組みで捉えることができる。機械学習モデルを初めから構築せず、既存のモデルを用いることで、学習に要する時間を削減し、モデルが出力する値を安定させる。

Online Transfer-learning を行うにあたり、 $Model_t$ を $Model_{t+1}$ へと転移させるための学習データ $train_data_{t+1}$ の作成を行う。学習データの生成で行う操作は三つ、(a)パラメータサンプルの生成、(b)シミュレータの実行、(c)損失関数微分値の計算である。図-4で $t + 1$ 期の学習データ $train_data_{t+1}$ の生成フローを示した。機械学習モデルの入力と出力は赤で示している。

本アルゴリズムでは、観測データ入手から推定パラメータ入手までの時間を短縮するために、Surrogate-based Optimization で一般的に行われる追加サンプルを行わない。追加サンプルを行わずとも機械学習モデルの精度を高く保つため、(a)パラメータサンプルの生成では学習データ密度をパラメータ推定値周りで高める。得られている前期の推定パラメータ β_{t+1}^* の周りに $t + 1$ 期の真のパラメータが存在していると仮定し、パラメータサンプルを生成する範囲を制限することで密度を高める。学習サンプルが取りうるパラメータ範囲は $t - 1$ 期の推定値 β_{t-1}^* とパラメータ範囲を決定する実数 a 、 β と同次元の単位ベクトル e を用いて $[\beta^* - a \cdot e, \beta^* + a \cdot e]$ と表す。図-5は毎期の学習データサンプル範囲を表す。

操作(b)はシミュレータを全学習データサンプルのパラメータに対して実行するため、大きな計算時間を要する。毎期学習データサンプルを変更する必要はないため、操作(a)と(b)は図-3で示した推定とは別に計算を進めておき、数日に一度、操作(b)が完了した段階で学習データサンプルを変更する。なお、このアルゴリズムは4.(1)で詳述する。一方で、損失関数の算出に用いる観測値 $observe_{t+1}$ は毎期変更する必要があるため、操作(c)は毎期行う。

b) 更新段階

提案手法では、パラメータ推定において微分値を用いた

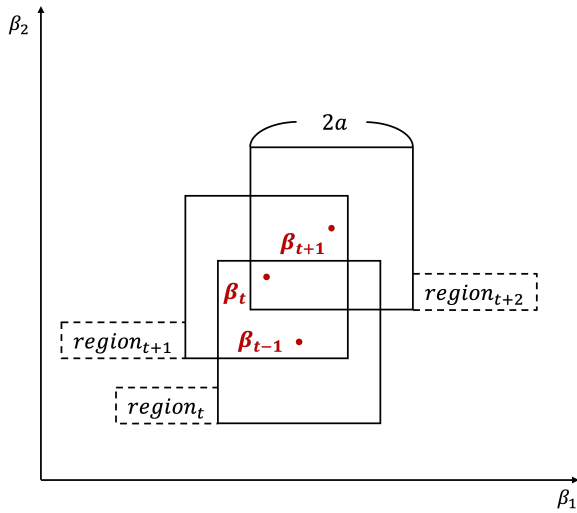


図-5 学習データサンプル範囲

最適化手法である勾配降下法を採用する。機械学習モデルが予測した損失関数微分値方向へパラメータの更新を繰り返すことによって、推定パラメータに近づく。微分値学習による最適化を採用することにより、ノイズに頑健な推定を行うことを目的とする。交通行動には異質性があり、交通行動シミュレータは行動の異質性を考慮する結果、実行ごとに異質性に起因するランダム性を反映する。ランダム性は、サロゲートモデルに対しては学習データに加わるノイズとして働き、推定精度を低下させる。微分値を用いることでノイズの影響を抑えることができる。なお、具体的な数値検証は次章で示す。なお、最適パラメータの探索には、勾配降下法のうち、ステップサイズ可変のアルゴリズム（最急降下法、Adam など）を適用する。

3. 有効性検証

本章では、提案アルゴリズムを用いたパラメータ推定の数値実験を行い、提案アルゴリズムの妥当性を示す。それぞれの数値実験では提案アルゴリズムで採用したそれぞれの拡張に焦点を当て、拡張がパラメータ推定に及ぼす影響を分析する。

(1) 微分値学習のノイズ頑健性検証

a) 実験目的

本数値実験は、微分値を学習して勾配降下法によりパラメータを推定する場合のノイズへの頑健性を検証する。加えて、微分値を用いない従来の Surrogate-based Optimization とパラメータ更新過程を比較し、微分値学習による頑健性のメカニズムを考察する。

表-1 微分値学習の有効性検証のための推定条件

方法	方法 1	方法 2
モデルの予測対象	損失関数微分値	損失関数値
最適化手法	勾配降下法	GA

b) 実験設定

表-1 に示す二つの方法でパラメータ推定を行う。方法 1 は微分値学習で、最適化手法には勾配降下法を採用する。方法 2 は従来の Surrogate-based Optimization と同じく損失関数を学習し、最適化手法には GA(遺伝的アルゴリズム)を採用する。方法 1 と 2 での推定精度を比較することで、微分値学習のノイズへの頑健性を検証する。

パラメータ推定では、式(1)に示す関数 $f(x_1, x_2)$ を損失関数として、パラメータ x_1 と x_2 を最適化する。ノイズへの頑健性を検証するため、学習データにノイズを加える。式(2)と(3)を用いて学習データにノイズ e を加える。関数 $scale$ は学習データの標準化、 inv_scale はその逆関数である。ノイズ e が従う正規分布の分散 σ は 0, 1/16, 1/8, 1/4 の 4 つの場合で実験を行う。実験では機械学習モデルに Gaussian Process モデルを用いる。

$$f(x_1, x_2) = x_1^2 + x_2^2$$

$$-1 \leq x_1 \leq 1, -1 \leq x_2 \leq 1 \quad (1)$$

$$f = inv_scale(scale(x_1^2 + x_2^2) + e)$$

$$e \sim N(0, \sigma) \quad (2)$$

$$\frac{\partial f}{\partial x_1} = inv_scale(scale(2x_1) + e)$$

$$\frac{\partial f}{\partial x_2} = inv_scale(scale(2x_2) + e) \quad (3)$$

$$e \sim N(0, \sigma)$$

c) 実験結果

ノイズが従う正規分布の分散 σ ごとに、50 回の推定を行い、真のパラメータ $(x_1, x_2) = (0, 0)$ とのマンハッタン距離を図-6 に示す。ノイズの分散が大きくなるにつれ推定精度が低下することが確認できる。微分値学習の手法 1 は、損失関数を学習した手法 2 に比べてノイズによる推定精度低下の影響が小さいことが読み取れ、ノイズへの頑健性が示唆される。ノイズの分散スケールにより頑健性は異なることも図からは読み取れるが、少なくとも実値の 1/4 程度以下のスケールでは微分値学習が優位である。

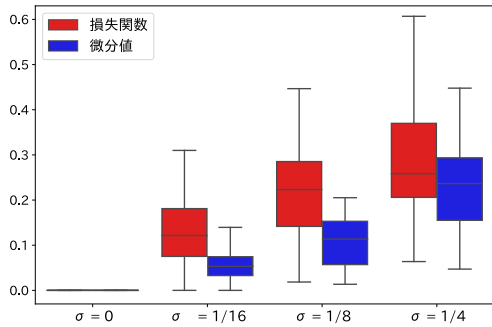


図-6 ノイズ下の推定精度比較

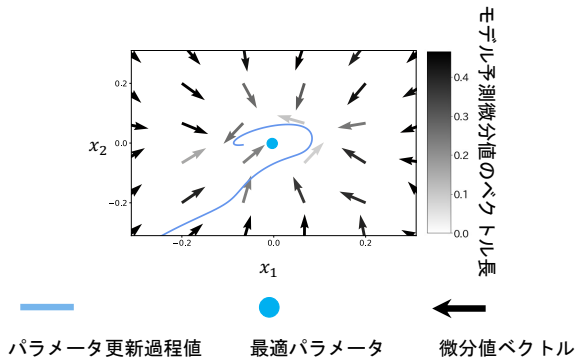


図-7 微分値学習時のパラメータ更新過程

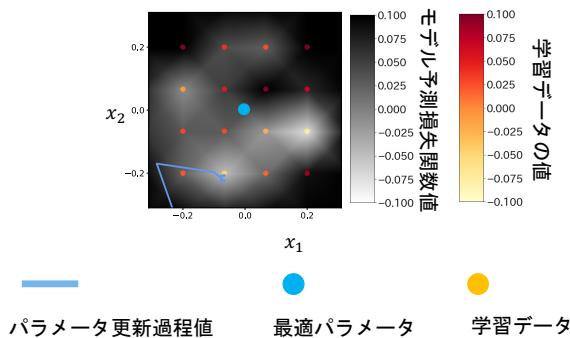


図-8 損失関数学習時のパラメータ更新過程

図-7 と図-8 は微分値学習と損失関数学習での最適化アルゴリズムのパラメータ更新過程の一例を示している。学習サンプルは 256、ノイズの分散は1/8である。両方とも青で示した真のパラメータ $(x_1, x_2) = (0, 0)$ に近づくほど高推定精度である。損失関数学習では、機械学習モデルが予測する損失関数値(ヒートマップで白黒で示した)が、ノイズが加わった黄色の一つの学習データの影響を受けた結果、その学習データ周りを最適パラメータと推定してしまっている。微分値学習でもモデルが予測する微分値(矢印でベクトルを示した)にノイズが加わり、パラメータの更新方向が安定していないが、更新先で正しい更新方向を得ることで最終的に最適値に近い値

表-2 オンライン推定の有効性検証のための推定条件

手法	手法 1	手法 2	手法 3
モデル出力	損失関数微分値	損失関数値	推定パラメータ
モデル入力	パラメータ	パラメータ	観測値
推定手法	勾配降下法	GA	モデル出力

を推定している。ノイズがパラメータの評価値(損失関数)に加わるか、パラメータの更新値(損失関数微分値)に加わるかの違いにより、微分値学習ではノイズに頑健な推定を行なえると考えられる。

(2) 簡易シミュレータでのオンライン推定の精度検証

a) 実験目的

本数値実験では、オンラインでの推定と学習範囲制限の有効性を検証する。パラメータ推定には、アクティビティシミュレータと類似の操作を行う行動シミュレータを用いる。実験目的の項目は以下の通りである。

- ア. Online Transfer-learning により、変化する毎期のパラメータを適切に推定可能なことを確認
- イ. 学習範囲制限により、パラメータ推定精度が向上することを確認

また、学習範囲の制限により危惧される、

- ウ. 学習範囲外に真のパラメータが存在する場合の推定精度が著しく低下しないことを確認

b) 実験設定

表-2 に示す3つの手法でパラメータ推定を行う。手法1と2はそれぞれ(1)で行った推定法と同じく、微分値学習を行う推定法と損失関数学習を行う推定法である。手法3は観測値から推定パラメータを直接予測する機械学習モデルで推定を行う。2章で述べた提案アルゴリズムでは手法1を採用している。機械学習モデルにはDNN(Deep Neural Network)を用いた。DNNの層構造は付録に示す。

実験では、二つの離散選択モデルを組み合わせた行動シミュレータのパラメータ推定を行った。メッシュ m にいる人が移動を行う効用 V_{dep}^m 、メッシュ m から l に移動する効用 $V_{move}^{m,l}$ は式(4)、(5)のように書ける。変数 X はメッシュごとの事前情報であり、パラメータ β を推定する。シミュレータにより推移するメッシュの人口 a^m と観測人口 a^m から損失関数を式(6)のように計算する。

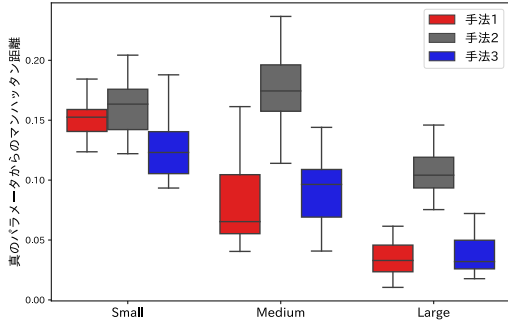


図-9 推定精度の手法比較 (Case 1)

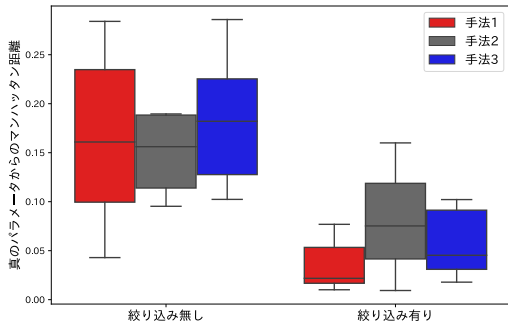


図-10 学習データ範囲制限の影響比較 (Case 2)

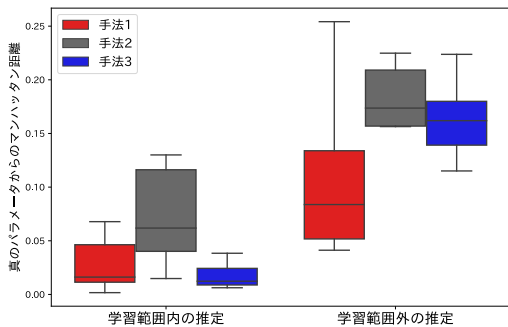


図-11 学習範囲外に変化した場合の推定精度 (Case 3)

$$V_{dep}^m = \beta_{dep^1} \cdot X_{dep}^{m,1} + \beta_{dep^2} \cdot X_{dep}^{m,2}$$

$$V_{move}^{l,m} = \beta_{cost} \cdot X_{cost}^{l,m} + \beta_{appeal^1} \cdot X_{appeal}^{m,1} + \beta_{appeal^2} \cdot X_{appeal}^{m,2}$$

$$Loss(\beta) = \frac{1}{M} \sum_m^M (a^m - \hat{a}^m)^2 \quad (6)$$

実験では、3つの Case でパラメータ推定を行った。各 Case では提案アルゴリズムを用いて、モデル入出力と推定手法を表-2に示すように変更した3つの手法で推定を行う。

- Case 1 学習データ数を3段階用意し、オンライン推定の精度を3手法で確認
- Case 2 学習データ範囲制限の有無で精度の変化を確認
- Case 3 パラメータが学習範囲外まで変化した場合の推定精度を確認

c) 実験結果

各 Case について、手法ごとに10回の推定を行い、推定パラメータと真のパラメータとのマンハッタン距離を算出した。推定精度を図-9, 10, 11に示す。

Case 1 では学習データ数の増加に伴い推定精度の向上が図-9 から読み取れる。オンライン推定が精度良く行えていることがわかり、実験目的(ア)が検証された。特にデータ数が大きい状況では手法1と3による推定精度が高い。データ数が小さい状況では手法3での推定精度が最も高い。

Case 2 では、学習データ範囲制限の有無で推定精度の向上が図-10 から読み取れる。実験目的(イ)が確認された。特に手法1と3で推定精度が大きく向上した。

Case 3 では学習範囲外のパラメータに対して推定精度の低下を図-11 で確認したが、手法1では学習範囲外でも推定精度を大きく落とすことなく推定が行われている。実験目的(ウ)の検証を行なった。

以上の結果から手法1と3においてオンライン推定が精度良く行えていることを確認した。学習サンプル数が少ない場合には手法3での推定精度が高いが、学習範囲の制限により推定精度の向上が確認でき、学習範囲制限を行う場合は、手法1を用いることで学習範囲外での推定精度低下を免れることができる。

4. シミュレータでのパラメータ毎時推定

本章では、2016年4月に発生した熊本地震後の熊本県をシミュレーション対象として、復旧期アクティビティシミュレータ SPACE のパラメータ推定を行う。パラメータ推定の操作を述べた後、パラメータ推定精度を確認するための数値実験を行う。

(1) 大規模計算における推定フローの工夫

3章の数値実験で得られた結果を踏まえて、2章で提案したアルゴリズムを復旧期シミュレータに適用するために組み合わせた。4.2節の手法1と手法3を組み合わせ、パラメータ推定を行う。パラメータ推定の際の、推定対象パラメータやパラメータ探索範囲、推定に用いる観測データは元論文に準拠する。観測データは毎時入手できると想定し、毎時のパラメータ推定を行う。

パラメータ推定のフローを図-12に示す。推定対象のパラメータは12個あり、その中でも特に感度の高いパ

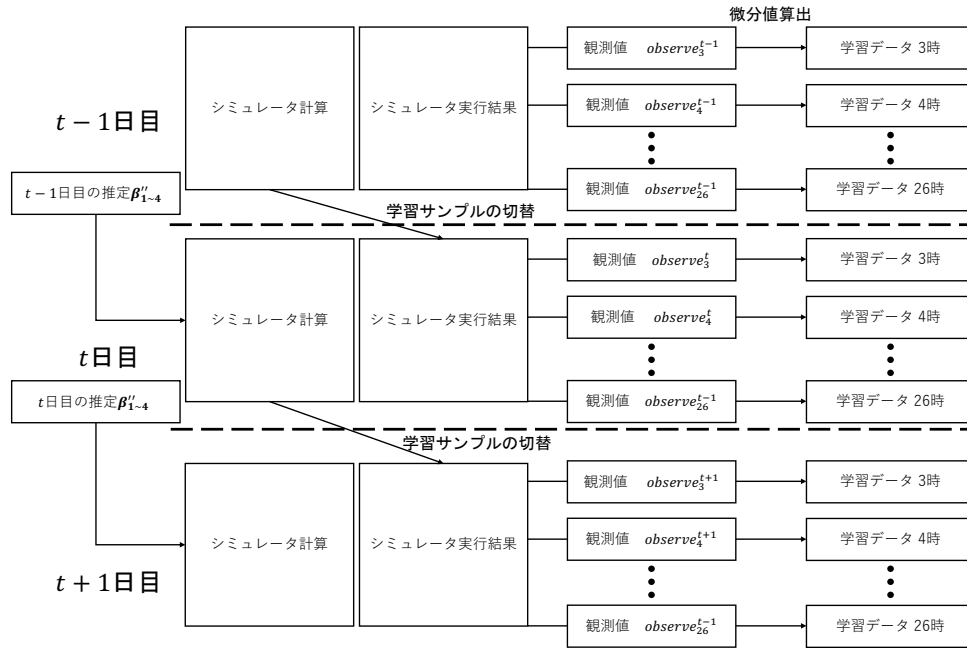


図-12 提案アルゴリズムを大規模シミュレータに適用する際の操作フロー

表-3 シミュレータでの提案アルゴリズム所要時間

操作	実行時期	所要時間
手法3 学習サンプル作成	推定開始 1 日目	1 サンプル 40 秒×1200 サンプル =13.3 時間
手法1 学習サンプル作成 前日推定値周りでのサンプル 逆関数近似推定	推定開始 1 日目以降 前日推定値入手後 推定開始 1 日目 最終観測値入手時	1 サンプル 40~80 秒×625 サンプル =7~14 時間 15.0 分
微分値学習推定	逆関数近似推定後 毎時観測値入手時	10.5 分

ラメータを $\beta_{1\sim4}$ として抽出した。初めに、少数の学習データでパラメータ推定を行える手法3を用いる。初めに、観測データから12個の推定パラメータを、機械学習モデルにより直接推定する。得られる推定結果を $\hat{\beta}_{1\sim12}$ とする。次に、以降では感度が高い $\beta_{1\sim4}$ に対して、より高精度な推定パラメータを毎時得るため、提案アルゴリズムである手法1によりリアルタイム推定を行う。

推定に必要な操作の所要時間を表-3に示す。オンライン推定の学習サンプルの作成には7~14時間程度を要するため、学習サンプルは1日に1回更新する。一方で、オンライン推定は観測データを入手してから10分程度で行うことができ、毎時推定パラメータを得るのに十分なリアルタイム性が保証されている。学習サンプル作成の際に用いる、シミュレータ実行の計算環境は表-4に示す。

(2) シミュレータでの推定精度確認

本節では、提案アルゴリズムを用いて復旧期シミュレ

表-4 計算環境 TokyoMarkov

GPU	Xeon(R) Gold 6128 @ 3.40GHz
OS	Ubuntu 18.04.3
メモリ	81GB
コンパイラ	g++ ver.7.4.0
オプション	-O3 -std=c++11 (-fopenmp)

表-5 パラメータ真値

	β_1	β_2	β_3	β_4
1 日目	1	5	5	5
2 日目	1.2	3	6.5	3.5
3 日目	1.4	1	8	2

ータのパラメータ推定をオンラインで行う際の、パラメータ推定精度を確認する。確認項目は以下の二つ。である。

1. 提案アルゴリズムで、シミュレータのパラメータをオンラインで推定した際の精度
2. 提案アルゴリズムである微分値学習を行うことで、機械学習モデルによる直接の推定パラメータから

推定精度が改善されるか

推定実験では、3 日間にわたり変化するパラメータを作成し、作成したパラメータを入力してシミュレータを実行した結果を、毎時得られる観測データとして用いた。感度が特に強いパラメータ $\beta_{1\sim4}$ を変化させ、他のパラメータは一定とした。真のパラメータ変化は表-5 に示す。機械学習モデル (3 と同様に Deep Neural Network を適用) による直接のパラメータ推定を 3 日間に渡り行った場合と、提案アルゴリズムによる毎時のパラメータ推定を行った場合とで推定精度を比較する。

得られた推定パラメータの精度を図-13 に示す。真のパラメータ(黒線)、提案アルゴリズムで推定したパラメータ(赤線)、機械学習モデルで直接推定したパラメータ(青線)の 3 つの値の毎時推移を $\beta_{1\sim4}$ について折れ線グラフで表した。真のパラメータ値については、シミュレータが 24 時間分の予測を行うことを加味して、24 時間の平均値を計算してプロットした。提案アルゴリズムでは、真のパラメータ変化を追ってオンラインに推定値を返すことができている。 β_4 のみ変化を追えていないが、推定値は近い値を返している。一方、機械学習モデルで直接推定したパラメータは、推定値は真のパラメータに近いが、真のパラメータの変化に対して推定値を変化させることができている。避難人口やネットワークの情報が変化の中で、観測値から直接変化するパラメータを推定するには、より複雑な推定構造や学習データが必要であると考えられる。単純な機械学習モデルでのパラメータ推定に対して、提案アルゴリズムで精度高い推定を行えることを確認した。

5. おわりに

本研究では、大規模交通シミュレータの毎時変化するパラメータを、リアルタイムに推定するアルゴリズムを提案した。大きい計算時間を要するシミュレータに適用される Surrogate-based Optimization を元に、復旧期の観測データを用いた交通シミュレータのパラメータ推定で現れる特有の課題を、解決するような拡張を実装し、提案アルゴリズムとした。従来と比べて新たに導入した拡張と、その効果を以下に示す。

- 拡張1. 損失関数微分値を機械学習モデルで予測し、勾配降下法を採用
- 拡張2. Online Transfer-learning⁶⁾による機械学習モデルの更新
- 拡張3. 機械学習モデルの学習データ範囲制限

- 効果1. ノイズに頑健なパラメータ推定
- 効果2. 真のパラメータ変化に応じた推定

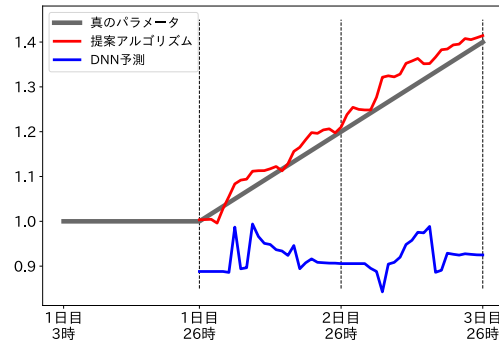


図-13-a パラメータ β_1

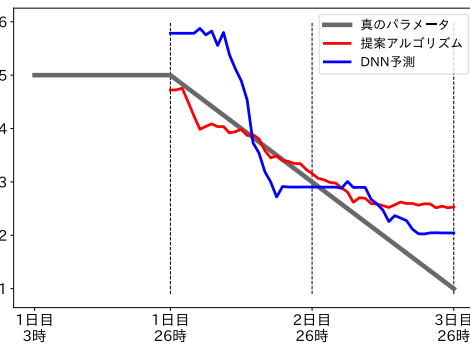


図-13-b パラメータ β_2

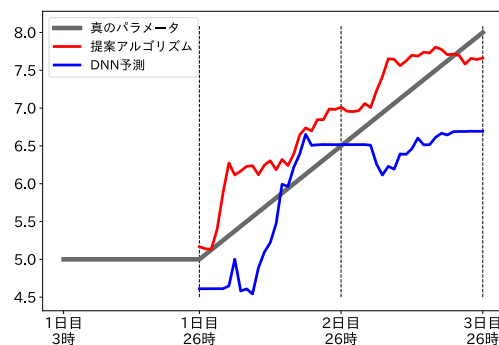


図-13-c パラメータ β_3

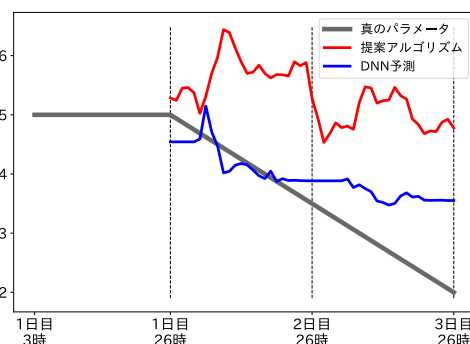


図-13-c パラメータ β_4

図-13 シミュレータでの毎時推定パラメータの推移

効果3. 観測値を得た直後に、高精度の推定パラメータ入手

複数の数値実験により、拡張の効果を検証した。それぞれの拡張は、上に示した効果が数値実験上で確認できた。検証については、より一般性を高める設定で行うことで、異なる条件下での成立を確認することもできよう。

本研究の交通分野での貢献は、本アルゴリズム開発により、従来は行えなかったリアルタイムのパラメータ推定を大規模交通シミュレータで可能にした点にある。4章では復旧期シミュレータを用い、提案アルゴリズムによって毎時にパラメータ推定を十分な精度で行えることを示した。行動の特徴が時々刻々と変化することが明らかになっている災害復旧期において、真の行動パラメータは1時点で推定したパラメータから時間と共に離れていき、シミュレーションの精度が損なわれる。災害復旧期の施策検討において、施策効果の計算を常に正確に行うために、交通シミュレータと共に提案アルゴリズムの活用が望まれる。

謝辞：東京大学の羽藤英二教授には、アルゴリズムの発展方針について多くのご意見をいただきました。また、藤原章正教授をはじめ、広島大学の交通工学研究室の方々から、災害についての情報提供を多くいただきました。感謝申し上げます。本研究の一部は、JSPS 科研費 20H00265 の助成を受けたものです。

参考文献

- 1) 桑原雅夫・原祐輔・三谷卓摩・川崎洋輔・竹之内篤・井料隆雅, 浦田淳司. 熊本地震における都市内交通, 避難の実態と課題. 土木計画学研究・講演集, pp. 54–58, 11 2016.
- 2) 浦田 淳司・佐々木 泰・井料隆雅. 災害復旧期におけるアクティビティシミュレータの開発と適用. 第 61 回土木計画学研究発表会, 2020.
- 3) 佐々木 泰・浦田 淳司・井料隆雅. 災害復旧期における日常活動需要計算のためのアクティビティシミュレータの構築. 第 60 回土木計画学研究発表会, 2019.
- 4) Alexander I. J. Forrester, Andr as Sobester, and Andy J. Keane. Engineering design via surrogate modelling. *Engineering Design via Surrogate Modelling*, 7 2008.
- 5) Zhong-Hua Han and Ke-Shi Zhang. Surrogate-based optimization. *Real-World Applications of Genetic Algorithms*, 3 2012.
- 6) Peilin Zhao, Steven C.H. Hoi, Jialei Wang, and Bin Li. Online transfer learning. *Artificial Intelligence*, Vol. 216, pp. 76–102, 2014.
- 7) Jack P.C. Kleijnen. An overview of the design and analysis of simulation experiments for sensitivity analysis. *European Journal of Operational Research*, Vol. 164, pp. 287–300, 7 2005.
- 8) Ding Yang, Yimin Sun, Danilo Di Stefano, Michela Turin, and Sevil Sariyildiz. Impacts of problem scale and sampling strategy on surrogate model accuracy: An application of surrogate-based optimization in building design. 2016 IEEE Congress on Evolutionary Computation, CEC 2016, pp. 4199–4207, 11 2016.
- 9) Atharv Bhosekar and Marianthi Ierapetritou. Advances in surrogate based modeling, feasibility analysis, and optimization: A review. *Computers and Chemical Engineering*, Vol. 108, pp. 250–267, 1 2018.
- 10) Wim C.M. Van Beers and Jack P.C. Kleijnen. Kriging interpolation in simulation: A survey. *Proceedings - Winter Simulation Conference*, Vol. 1, pp. 113–120, 2004.
- 11) Xinhai Chen, Rongliang Chen, Qian Wan, Rui Xu, and Jie Liu. An improved data-free surrogate model for solving partial differential equations using deep neural networks. *Scientific Reports*, Vol. 11, 12 2021.
- 12) Sun Hye Kim and Fani Boukouvala. Surrogate-based optimization for mixed-integer nonlinear problems. *Computers and Chemical Engineering*, Vol. 140, 9 2020.
- 13) Atılım Güneş, Güneş Baydin, Barak A Pearlmutter, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, Vol. 18, pp. 1–43, 2018.
- 14) Aatila Mustafa, Lachgar Mohamed, and Kartit Ali. An overview of gradient descent algorithm optimization in machine learning: Application in the ophthalmology field. *Communications in Computer and Information Science*, Vol. 1207 CCIS, pp. 349–359, 2020.
- 15) 北野宏明. 遺伝的アルゴリズム. *人工知能学会誌*, Vol. 7, pp. 26–37, 1992.
- 16) Luis Miguel Rios and Nikolaos V. Sahinidis. Derivative-free optimization: A review of algorithms and comparison of software implementations. *Journal of Global Optimization*, Vol. 56, pp. 1247–1293, 7 2013.
- 17) Jorge Nocedal and Strehpan J. Wright. *Numerical Optimization*. Springer New York, 2006.
- 18) Robert Hecht-Nielsen. Theory of the backpropagation neural network. pp. 593–605, 1989.
- 19) Sebastian Ruder. An overview of gradient descent optimization algorithms. 9 2016.
- 20) Karl Weiss, Taghi M. Khoshgoftaar, and Ding Ding Wang. A survey of transfer learning. *Journal of Big Data*, Vol. 3, 12 2016.
- 21) Rich Caruana, Lorien Pratt, and Sebastian Thrun. Multitask learning. *Machine Learning* 1997, 28:1, Vol. 28, pp. 41–75, 1997.
- 22) Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. pp. 1126–1135, 7 2017.
- 23) Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 22, pp. 1345–1359, 2010.
- 24) Steven C.H. Hoi, Doyen Sahoo, Jing Lu, and Peilin Zhao. Online learning: A comprehensive survey. *Neurocomputing*, Vol. 459, pp. 249–289, 10 2021.
- 25) Martin Zinkevich. Online convex programming and generalized in-

- tesimal gradient ascent. Proceedings of the Twentieth International Conference on International Conference on Machine Learning, pp. 928–935, 2003.
- 26) Carlos A Coello Coello. Theoretical and numerical constraint-handling techniques used with evolutionary algorithm : a survey of the state of the art. *Comput. Methods Appl. Mech. Engrg.*, Vol. 191, pp. 1245–1287, 2002.
- 27) Soora Rasouli and Harry Timmemans. Activity-based models of travel demand: Promises, progress and prospects. *International Journal of Urban Sciences*, Vol. 18, pp. 31–60, 1 2014.
- 28) Quentin Le Lidec, Igor Kalevatykh, Ivan Laptev, Cordelia Schmid, and Justin Carpentier. Differentiable simulation for physical system identification. *IEEE Robotics and Automation Letters*, Vol. 6, pp. 3413–3420, 4 2021.
- ?
?

Real-time estimation method for time-varying behavioral parameters of traffic demand prediction in disaster restoration period

Yosuke MOCHIZUKI and Junji URATA

This paper proposed a parameter estimation method by which one can continuously estimate updated parameters in the disaster restoration period. We extended surrogate-based optimization in three ways, in order to solve a problem respectively. Three extensions, which are using derivatives for optimization, applying online transfer-learning and putting restrictions on the range of training data, are implemented after verification of their effect. We estimated parameters of an activity simulator SPACE online using the data collected during the earthquake happened in 2016 Kumamoto, and the proposed method.