

# 2203 車体振動加速度を用いた軌道狂い推定に関する理論検討

正 [機] ○永沼 泰州 (日本機械保線) 学 [機] 小林 尊仁 (日大)  
正 [機] 綱島 均 (日大)

## Study on track geometry estimation from car-body acceleration

Yasukuni NAGANUMA, Nippon Kikai Hosen KK., 2-1-95 Konan, Minato-ku Tokyo  
Takahito KOBAYASHI, Nihon University  
Hitoshi TSUNASHIMA, Nihon University

To enhance the safety and reliability of railway transportation, it is one of the most important tasks to check the track condition frequently and accurately. This paper describes the track condition monitoring technique using car-body motions. In an inverse problem to estimate track irregularity from car-body acceleration and pitching rate measured by in-service vehicle, dynamic programming filter and Karman filter were applied. Study result confirmed that proposed solutions can be used to estimate track geometry in level with desired precision.

**Key word:** track condition monitoring, inverse problems, Kalman filter, dynamic programming filter

### 1 はじめに

軌道の不整(軌道狂い)の計測とこれに基づく補修は、安全で快適な鉄道システムの維持に不可欠である。通常、軌道狂いは特別な車体構造を持つ「軌道検測車」によって計測されており、その頻度は新幹線で3回/月、在来線では数回/年程度である。これに加えて新幹線や一部の在来線では、軌道状態監視を目的に複数編成の営業列車に計測装置が搭載され、車体振動加速度測定が高頻度に行われている。通常は乗り心地向上を目的とした軌道保守投入箇所の選定に活用されており、管理値を超える加速度が観測された場合には速度規制が実施される。車体振動加速度測定は、処理装置やセンサがシンプルかつ安価で保守に手が掛からないが、その測定波形は軌道狂い波形とかなり異なり、しかもその値は車両走行速度に大きく依存してしまう。もしも車体振動加速度測定の長所はそのままに短所を改善できるなら、軌道状態監視の精度向上とともに鉄道の安全性をさらに高めることができる。

以上の問題認識から、筆者らは車体振動加速度データから軌道狂いを逆算する手法を開発しており、既にインパルス応答を用いた手法を提案した<sup>1)</sup>。本稿はその後の研究成果として、「ダイナミックプログラミングフィルタ」および「カルマンフィルタ」を用いて、車体の鉛直加速度とピッチ角速度から10 m弦高低狂いを逆解析した結果を紹介する。

なお、本稿では、軌道狂いや線形を含んだ軌道の形状を「軌道形状(track geometry)」と呼び、設計線形からの不整量である「軌道狂い(track irregularity)」と区別する。また、車体振動加速度を単に「車体動揺」と呼ぶ。

### 2 理論検討の概要

本検討では鉄道車両の運動を状態空間モデルで表現する。車両運動シミュレーションやデジタルフィルタなど、時系列解析の多くの問題は、状態空間モデルを用いることによって状態推定の問題として定式化できる。状態空間モデルは、(1)式に示す状態方程式と(2)式に示す観測方程式によって表される。

$$\dot{\mathbf{x}}_n = \mathbf{F}\mathbf{x}_{n-1} + \mathbf{P}\mathbf{u}_n + \mathbf{G}\mathbf{w}_n \quad (\text{状態方程式}) \quad (1)$$

$$\mathbf{y}_n = \mathbf{H}\mathbf{x}_n + \mathbf{v}_n \quad (\text{観測方程式}) \quad (2)$$

ただし、 $\mathbf{x}_n$ は状態ベクトル、 $\mathbf{u}_n$ は入力ベクトル、 $\mathbf{y}_n$ は出力ベクトルであり、 $\mathbf{w}_n \sim N(0, Q)$ と $\mathbf{v}_n \sim N(0, R)$ はいずれもガウス型白色雑音で、それぞれシステムノイズ、観測ノイズと呼ばれる。また、 $\mathbf{F}$ は状態遷移行列、 $\mathbf{P}$ は入力行列、 $\mathbf{G}$ は駆動行列、 $\mathbf{H}$ は観測行列と呼ばれることがある。本検討は、逆解析手法の有効性のみを理論的に確認するため、以下(a)~(g)に示す手順で行った。

- 2階の微分方程式で表わされる車両の運動方程式を1階線形微分方程式に変換し、これを離散化して状態方程式を求める。
- 乱数を発生させ、実際の軌道の周波数特性に類似した軌道形状データ(波長6.0 m以上)を作成する。
- (b)の軌道形状データを(a)のモデルに入力し、後部台車直上の車体動揺(鉛直方向)および車体ピッチ角速度を計算する。
- (c)の車体動揺およびピッチ角速度に正規分布ノイズを加え、観測データとする。
- 逆解析の準備として、車両モデルを簡素化し、状態方程式を変更する。
- ダイナミックプログラミングフィルタ、カルマンフィルタを用いて、(d)の観測値から軌道形状を逆解析する。
- (b)の軌道形状と(f)で推定した軌道形状を比較し、各手法の有効性を確かめる。また、10 m弦高低狂いでの比較も行う。

### 3 車両モデルを用いた順解析

図1に鉛直方向の1車両モデルを示す。この6自由度モデルに、高低狂い $\mathbf{r}(t)$ を強制変位として入力した場合の運動方程式は(3)式となる。

$$\mathbf{M}\ddot{\mathbf{y}}(t) + \mathbf{C}\dot{\mathbf{y}}(t) + \mathbf{K}\mathbf{y}(t) = \mathbf{D}\dot{\mathbf{r}}(t) + \mathbf{E}\mathbf{r}(t) \quad (3)$$

ここで、

$$\mathbf{y}(t) = \begin{pmatrix} y_c \\ \alpha_c \\ y_{b,1} \\ \alpha_{b,1} \\ y_{b,2} \\ \alpha_{b,2} \end{pmatrix}, \quad \mathbf{r}(t) = \begin{pmatrix} y_{w,1} \\ y_{w,2} \\ y_{w,3} \\ y_{w,4} \end{pmatrix} \quad (4)$$

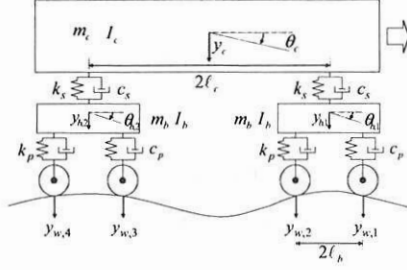


Fig. 1 Linear vehicle model

$$\alpha_c = l_c \theta_c, \quad \alpha_{b,1} = l_b \theta_{b,1}, \quad \alpha_{b,2} = l_b \theta_{b,2} \quad (5)$$

である。

2階の微分方程式である(3)式は、状態変数ベクトルを  $(\mathbf{y}, \dot{\mathbf{y}})^T$  とおけば、(6)式に示す1階の線形連立微分方程式に変換される。

$$\begin{pmatrix} \dot{\mathbf{y}} \\ \ddot{\mathbf{y}} \end{pmatrix} = \begin{pmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{pmatrix} \begin{pmatrix} \mathbf{y} \\ \dot{\mathbf{y}} \end{pmatrix} + \begin{pmatrix} 0 \\ M^{-1}E \end{pmatrix} \begin{pmatrix} 0 \\ \dot{\mathbf{r}} \end{pmatrix} \quad (6)$$

(6)式を数値積分法で離散化すれば、(1)式においてシステムノイズを考慮しない場合の状態方程式、

$$\mathbf{x}(n+1) = F\mathbf{x}(n) + P\mathbf{u}(n) \quad (7)$$

が得られる。状態ベクトルと外力ベクトルは(8)式に示す通りである。今回は離散化にPade 11法を用いた。

$$\mathbf{x}_n = \begin{pmatrix} \mathbf{y}_n \\ \dot{\mathbf{y}}_n \end{pmatrix} \in R^{12}, \quad \mathbf{u}_n = \begin{pmatrix} \mathbf{r}_n \\ \dot{\mathbf{r}}_n \end{pmatrix} \in R^8 \quad (8)$$

得られた状態方程式に入力した軌道形状、計算された鉛直方向の車体動揺、車体ピッチ角速度を図2に示す。

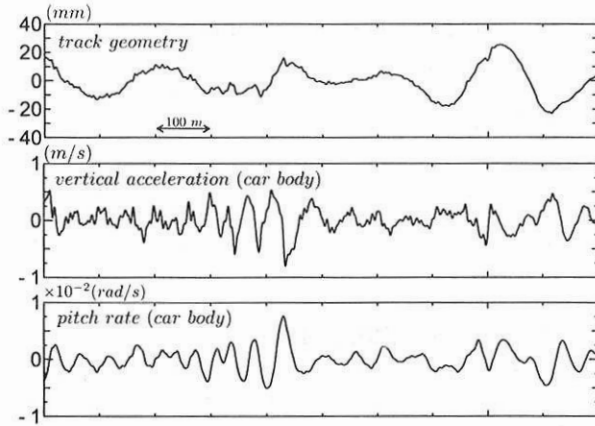


Fig. 2 Calculated acceleration and pitch rate

以降の検討では、得られた車体動揺とピッチ角速度に正規分布ノイズを加え、これらを観測値として用いた。車両動揺ノイズの標準偏差は  $3.0 \times 10^{-5} \text{ (m/s}^2\text{)}$ 、ピッチ角速度ノイズの標準偏差は  $3.0 \times 10^{-5} \text{ (rad/s)}$  とした。

#### 4 車体での観測値を用いた軌道狂いの推定(逆解析)

次に車両モデルを用いて、車体動揺と車体ピッチ角速度から軌道形状(および10 m弦高低狂い)を推定する。この

操作は、本来安定な物理現象において、その因果律を遡り出力から入力を逆解析する「逆問題」であるから一般に演算は不安定で、計算結果は測定ノイズの影響を大きく受ける。従ってノイズの影響を軽減する何らかの安定化が不可欠であり、今回私たちは「ダイナミックプログラミングフィルタ」と「カルマンフィルタ」の2つの手法を逆解析に利用した。

#### 4.1 逆解析のための数値モデルの簡素化

(6)式に示すとおり、順解析では外力ベクトルとして8成分(4つの車軸に時間差を持って入力される軌道形状とその1階微分)を与える。しかし、車体動揺と車体ピッチ角速度の観測データのみを使用してこれら8成分全てを正しく逆解析することはできない。なぜなら、車体での観測値には台車ピッチングの情報が含まれていないからである。車体には前後軸の平均軌道形状  $((y_{w,1} + y_{w,2})/2, (y_{w,3} + y_{w,4})/2)$  の影響しか及んでいないから、車体での観測値を用いて逆解析できるのは前後軸の平均軌道形状である。従って、軸間距離により決まる軌道形状の波長成分(例えば軸間距離2.5 mの車両では波長5 m, 1.66 m, 1 m, ...)を逆解析することはできない。ところが、これは保線実務の観点から見るとむしろ好都合である。通常、10 m弦軌道狂いは5 m以上の波長帯域で管理されるので、軸間距離の軌道形状を平均した値が得られれば十分である。

従って逆解析のための車両モデルは前後台車のピッチングを表現できる必要はなく、図1のモデルから2自由度を減じて4自由度とすることができる。逆解析のために簡略化した数値モデルを図3に示す。

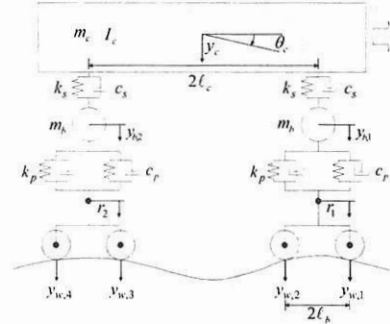


Fig. 3 Simplified model

このモデルの運動を記述するにあたり、以下の変数変換を行った。まず、前後台車の鉛直変位の和および差の平均をそれぞれ  $z_1, z_2$  とおく。

$$z_1 = \frac{y_{b,1} + y_{b,2}}{2}, \quad z_2 = \frac{y_{b,1} - y_{b,2}}{2} \quad (9)$$

また、車輪1,2の軌道形状の平均を  $r_1$ 、車輪3,4の軌道形状の平均を  $r_2$  とする。

$$r_1 = \frac{y_{w,1} + y_{w,2}}{2}, \quad r_2 = \frac{y_{w,3} + y_{w,4}}{2} \quad (10)$$

さらに、 $r_+$  and  $r_-$  を以下のように定義する。

$$r_+ = \frac{r_1 + r_2}{2}, \quad r_- = \frac{r_1 - r_2}{2} \quad (11)$$

(3)式の運動方程式における  $\mathbf{y}(t)$ ,  $\mathbf{r}(t)$  は自由度の縮小に

伴い下式で再定義される。

$$\mathbf{y}(t) = \begin{pmatrix} y_c \\ \alpha_c \\ z_1 \\ z_2 \end{pmatrix}, \mathbf{r}(t) = \begin{pmatrix} r_+ \\ r_- \end{pmatrix} \quad (12)$$

離散化した状態方程式の状態ベクトル $\mathbf{x}_n$ 、外力ベクトル $\mathbf{u}_n$ は以下のとおり。

$$\mathbf{x}_n = \begin{pmatrix} \mathbf{y}_n \\ \dot{\mathbf{y}}_n \end{pmatrix} \in R^8, \mathbf{u}_n = \begin{pmatrix} \mathbf{r}_n \\ \dot{\mathbf{r}}_n \end{pmatrix} \in R^4 \quad (13)$$

#### 4.2 ダイナミックプログラミングフィルタ

TrujilloとBusbyは、正則化を用いた動的計画法が動的システムの逆解析に有効であることを示した<sup>2)</sup>。このダイナミックプログラミングフィルタ (DPフィルタ) を用いた逆解析は、次式の最適化問題になる。

$$E_N(\mathbf{x}, \mathbf{r}_n) = \sum_{n=1}^N (\mathbf{y}_n - \mathbf{y}_n^*, w(\mathbf{y}_n - \mathbf{y}_n^*)) + (\mathbf{r}_n, \Lambda \mathbf{r}_n) \quad (14)$$

ここで $(\bullet, \bullet)$ は2つのベクトルの内積をあらわし、 $N$ は測定データ数である。 $w$ は重みマトリクスで今回は単位行列とした。 $\mathbf{y}_n^*$ は測定データで、 $\mathbf{y}_n$ は測定値と対応すべき状態変数である。 $\Lambda$ はTikhonov正則化パラメータで、ノイズの影響を抑制して解の値を制限する (逆算される軌道形状データの振幅を制御する) 働きをする。 $\mathbf{y}_n$ は状態変数 $\mathbf{x}_n$ と次式の観測方程式で関連付けられる。 $H$ は観測マトリクスである。

$$\mathbf{y}_n = H \mathbf{x}_n \quad (15)$$

最小2乗規範と動的計画法の原理を式(14)に適用し、Chandrasekharの式を用いれば、前進探索 (forward sweep) と後進探索 (backward sweep) という2セットの再帰式が得られる。最初のステップは以下に示す後進探索を行うことである。

$$D_{n-1}^{-1} = D_n^{-1} - 2G^T Y_n L_n Y_n^T G \quad (16)$$

$$K_{n-1} = K_n + (2K_n G^T Y_n L_n Y_n^T G - Y_n L_n Y_n^T G) D_{n-1} \quad (17)$$

$$Y_{n-1} = R^T (I - 2K_n G^T) Y_n \quad (18)$$

$$L_{n-1} = L_n + L_n Y_n^T 2G D_{n-1}^{-1} G^T Y_n L_n \quad (19)$$

$$\mathbf{s}_{n-1} = -2Q^T w \mathbf{y}_{n-1} + R^T (I - 2K_n G^T) \mathbf{s}_n \quad (20)$$

上記式は、計算開始点 $n = N$ における以下の初期条件のもとで計算される。

$$D_N^{-1} = 2\Lambda + 2G^T Q^T w Q G \quad (21)$$

$$K_N = Q^T w Q G D_N \quad (22)$$

$$Y_N = R^T Q^T \quad (23)$$

$$L_N = -w + w Q 2G D_N G^T Q^T w \quad (24)$$

$$\mathbf{s}_N = -2Q^T w \mathbf{y}_N \quad (25)$$

後進探索の過程において、ベクトル $D_n G^T \mathbf{s}_n$ とマトリクス $K_n^T R$ を全て記憶させておく。前進探索は式(26)および式(27)を $n = 1$ から順次計算すれば外部入力および状態ベクトルを得ることができる。

$$\mathbf{u}_n = -D_{n+1} P^T \mathbf{s}_{n+1} - 2K_n^T R \mathbf{x}_n \quad (26)$$

$$\mathbf{x}_{n+1} = F \mathbf{x}_n + P \mathbf{u}_n \quad (27)$$

DPフィルタでの逆解析にあたり、(6)式を変形して軌道形状を状態変数ベクトルに含める。これにより逆解析すべき外力ベクトル $\mathbf{u}_n$ を4成分から2成分に半減することができる。

$$\begin{pmatrix} \dot{\mathbf{y}} \\ \dot{\mathbf{r}} \end{pmatrix} = \begin{pmatrix} 0 & I & 0 \\ -M^{-1}K & -M^{-1}C & M^{-1}E \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{y} \\ \dot{\mathbf{y}} \\ \mathbf{r} \end{pmatrix} + \begin{pmatrix} 0 \\ M^{-1}D \\ I \end{pmatrix} \dot{\mathbf{r}}(t) \quad (28)$$

前進探索では外力として $\dot{\mathbf{r}}_n$ が求められ、同時に(27)式を解くことで、状態変数ベクトルの1成分として $\mathbf{r}_n = (r_+, r_-)^T$ が得られる。前後軸の平均軌道形状は下式で算出される。

$$\frac{y_{w,1} + y_{w,2}}{2} = r_+ + r_-, \quad \frac{y_{w,3} + y_{w,4}}{2} = r_+ - r_- \quad (29)$$

鉛直方向の車体動揺と車体のピッチ角速度を用いて得られた軌道形状、および軌道形状から計算した10 m高低狂いを図4に示す。Tikhonov正則化パラメータは $\Lambda = 1.0 \times 10^{-5}$ とした。

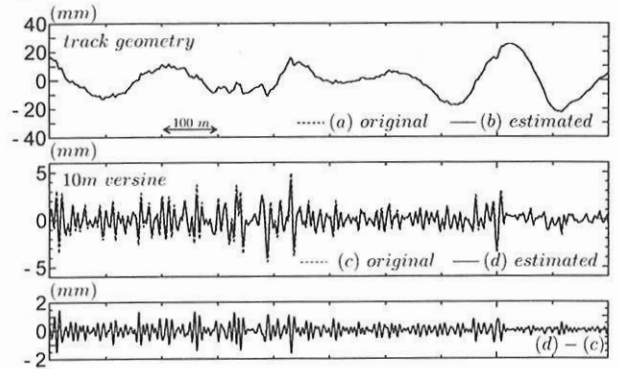


Fig. 4 Estimated track geometry by DP filter

推定された軌道形状および10 m弦軌道狂いは、いずれも真の軌道狂いを良く推定しており、軌道狂いの逆解析にDPフィルタが有効であることが確認された。10 m弦高低狂いの推定誤差は最大約1.5 mmであった。

DPフィルタは数学的にシンプルでプログラミングも容易である。しかも観測値および未知入力項のデータ長や観測箇所にも制約を受けないという利点がある。その反面、前進探索と後進探索の2つの計算ステップが必要となるので、オンラインのリアルタイム演算に適しているとは言い難い。

#### 4.3 カルマンフィルタ

カルマンフィルタは、(1),(2)式の状態空間モデルを用いて、誤差を含んだ観測値からシステムの状態を逐次推定する代表的な状態推定法であり、実時間処理に適している。外部入力がある場合のカルマンフィルタのアルゴリズムを以下に示す。

(1期先予測)

$$\mathbf{x}_{n|n-1} = F_n \mathbf{x}_{n-1|n-1} + P_n \mathbf{u}_n \quad (30)$$

$$V_{n|n-1} = F_n V_{n-1|n-1} F_n^T + G_n Q_n G_n^T \quad (31)$$

(フィルタ)

$$K_n = V_{n|n-1} H_n^T (H_n V_{n|n-1} H_n^T + R_n)^{-1} \quad (32)$$

$$\mathbf{x}_{n|n} = \mathbf{x}_{n|n-1} + K_n (\mathbf{y}_n - H_n \mathbf{x}_{n|n-1}) \quad (33)$$

$$V_{n|n} = (I - K_n H_n) V_{n|n-1} \quad (34)$$

通常、カルマンフィルタは、状態方程式における外部入力  $u_n$  を既知の確定入力として扱うので、これを逆解析することはできない。筆者らは先行研究において、入力である軌道形状をランダムウォークモデルで表現することによって、カルマンフィルタを逆解析に利用できることを示した<sup>1)</sup>。本研究も同様のアイデアに基づき、ランダムウォークモデル ( $u_{n+1} = u_n + e_n$ ) を、前節のDPフィルタで用いた状態方程式に組み入れる。具体的には、外力ベクトル (軌道形状の1階微分) を状態変数ベクトルに含めて(35)式とする。

$$\begin{pmatrix} x_{n+1} \\ u_{n+1} \end{pmatrix} = \begin{pmatrix} F & P \\ 0 & \Omega \end{pmatrix} \begin{pmatrix} x_n \\ u_n \end{pmatrix} + \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} w_n \\ e_n \end{pmatrix} \quad (35)$$

今回の研究では  $\Omega$  は単位ベクトルとし、システムノイズ  $w_n$  は0とした。観測方程式は(36)式の通りである。

$$d_n = \begin{pmatrix} H & 0 \end{pmatrix} \begin{pmatrix} x_n \\ u_n \end{pmatrix} + v_n \quad (36)$$

(35), (36)式の状態空間モデルによって、カルマンフィルタを用いた軌道形状の逆解析が可能になる。軌道形状は、状態変数ベクトル推定値の1成分として逐次的・安定的に与えられる。観測ノイズ ( $v_n$ ) の標準偏差を、2章(d)で加えた人工ノイズと同じ値である、 $3.0 \times 10^{-5} (m/s^2)$  および  $3.0 \times 10^{-5} (rad/s)$  にしたところ、逆算された結果には高周波ノイズが乗算してしまった。そこで、これを  $0.3 (m/s^2)$ ,  $0.3 (rad/s)$  としたところ良好な結果が得られた。逆解析した軌道形状と10 m弦高低狂いを図5に示す。10 m弦高低狂いの推定誤差は最大約1 mmであり、カルマンフィルタによる逆解析の有効性が確認された。

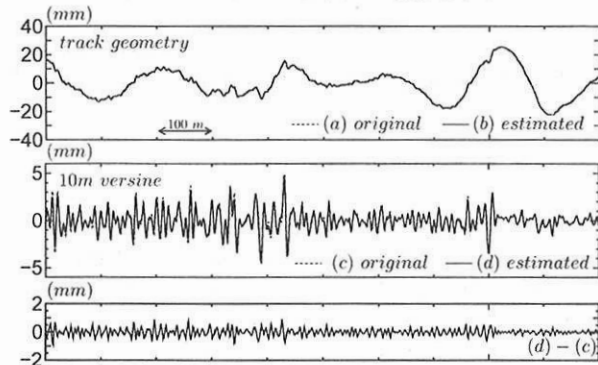


Fig. 5 Estimated track geometry by Kalman filter

## 5 おわりに

本稿では、測定が極めて容易な車体での観測値から、ダイナミックプログラミングフィルタおよびカルマンフィルタを用いて軌道狂いを推定する手法について検討した。本研究の成果を以下に要約する。

- (a) 1車体モデルの順解析では8入力 (4軸分の軌道形状とその微分) を与えるが、逆解析でこれらを全て正しく逆算することはできない。本研究では、保線実務を考慮した上で数値モデルを必要最小限に簡素化し、上記の問題を解決した。
- (b) ダイナミックプログラミングフィルタでは適切な正則化パラメータを与えることによって観測ノイズを抑制しながら安定的かつ実用的な精度で10 m弦高低狂いを推定できる。演算には後進探索と前進探索が必要なので実時間処理には適さない。

- (c) 軌道狂いをランダムウォークモデルで表現し、これを状態方程式に組み込むことでカルマンフィルタを逆解析に適用可能とした。計算の結果、ノイズを含む観測値から精度良く10 m弦高低狂いを推定できることがわかった。カルマンフィルタは実時間処理に適したアルゴリズムなので、軌道状態監視装置への応用が期待される。

ダイナミックプログラミングフィルタ、カルマンフィルタとも原理的には車速変化に対応できる。今後は実データでその有効性を確認するとともに、水準狂いなど他の軌道狂い推定にも取り組みたい。

また、本研究では、正則化パラメータの値や観測ノイズの分散の値を試行錯誤で求めたが、これら逆解析を安定化させる重要な役割を担うパラメータの選定指針を見出すことも今後の重要な課題の1つである。

## 参考文献

- 1) 永沼泰州, 小林尊仁, 網島均: 「状態空間モデルを用いた軌道の状態監視」, 機械学会関東支部第18期総会講演会 (2012).
- 2) Trujillo, D.M. and Busby, H.R.: Practical Inverse Analysis in Engineering, CRC Press, New York (1997).
- 3) 網島均, 松本陽, 水間毅, 中村英夫: 「プローブ車両による軌道の状態診断, 営業車両による軌道状態の常時監視システムの開発」, 検査技術, Vol.12, No.5 (2007), pp.50-55.
- 4) 永沼泰州, 吉村彰芳: 「状態空間モデルを用いた軌道狂い原波形の復元」, 第17回鉄道技術連合シンポジウム講演論文集 (2010).
- 5) 吉村彰芳: 「軌道狂い原波形の復元に関する理論的基礎の確立とその応用」, 鉄道技術研究報告, No.1336 (1987).
- 6) Y. Naganuma and Y. Sato: Track state control with use of real time digital data processing, International Journal of Heavy Vehicle Systems 2000, Vol. 7, No. 1, pp. 82-95 (2000).
- 7) Y. Naganuma and A. Yoshimura: Reconstruction and estimation of railway track geometry using regularization methods, Proceedings of IAVSD 2009 (2009).
- 8) 北川源四郎: 「時系列解析入門」, 岩波書店 (2005)
- 9) 中野道雄, 西山清: 「パソコンで解くカルマンフィルタ」, 丸善 (1993)