

深層学習を用いた 河川水位予測手法の開発

DEVELOPMENT OF THE REAL-TIME RIVER STAGE PREDICTION METHOD USING DEEP LEARNING

一言正之¹・櫻庭雅明²・清雄一³

Masayuki HITOKOTO, Masaaki SAKURABA and Yuichi SEI

¹正会員 工修 日本工営株式会社 中央研究所 (〒300-1259 茨城県つくば市稲荷原2304)

²正会員 博(工) 日本工営株式会社 中央研究所 (〒300-1259 茨城県つくば市稲荷原2304)

³非会員 博(情報理工) 電気通信大学大学院情報システム学研究所 (〒182-8585 東京都調布市調布ヶ丘1-5-1)

The real-time river stage prediction model is developed, using the artificial neural network model which is trained by the deep learning method. The model is composed of 4 layer feed-forward network. As a network training method, stochastic gradient descent method based on the back propagation method was applied. As a pre-training method, the denoising autoencoder was applied. The developed model is applied to the one catchment of the OYOYODO River, one of the first-grade river in Japan. Input of the model is hourly change of water level and hourly rainfall, output data is water level of HIWATASHI. To clarify the suitable configuration of the model, case study was done. The prediction result is compared with the other prediction models, consequently the developed model showed the best performance.

Key Words : River stage, real-time prediction, deep learning, artificial neural network

1. はじめに

洪水災害に対するソフト対策として、洪水予測技術の高度化が求められている。洪水予測手法として、降雨一流出過程を表現する物理型モデル¹⁾²⁾などが全国の河川で適用されているが、予測の過程に様々な誤差が内在する。一方、降雨や水位等の水文観測データに基づいたニューラルネットワーク (Artificial Neural Network: ANN) による洪水予測も、統計的手法の一種として既往の研究にて広く適用性が確認されている³⁾⁴⁾⁵⁾⁶⁾。ANNでは上流の水位や雨量を入力データとして下流の水位変化を学習することで水位予測を行うことができる。

ANNの課題として、特にネットワーク構造が大きい場合、過学習により特定のデータへの過度な適合が生じ、データに内在する規則性の獲得が容易ではなくなる事が挙げられる。したがって、ANNの精度向上のためには、入力データ候補を適切に選定することが必要となる⁹⁾。しかしながら、予測時間が長い場合などでは、限られた入力データから精度向上を図るのは難しい⁷⁾。ANNの新しい手法として深層学習と呼ばれる手法が提案されている⁸⁾。深層学習の特徴として、学習モデル自

ら、大量の入力データの中から本質的な情報の抽出を行うことが挙げられる。深層学習は、画像認識や音声認識、自然言語処理など様々な分野で研究が進められているが、河川の水位予測に深層学習を適用した事例は見られない。本研究では、深層学習を用いたANNによる河川水位予測手法を開発した。一級河川である大淀川水系の一流域を対象に、開発した手法の適用性を検証した。

2. 深層学習を用いた河川水位予測手法

深層学習とは、複数の中間層を用いたANNにより学習を行う手法であり、これまで様々な方法論が提案されている⁹⁾。本研究では、自己符号化器を積み重ねた階層型のネットワークを用いた。自己符号化器による事前学習の適用は、従来のANNとは大きく異なる点の一つである。本研究で適用した手法の概要を以下に示す。

(1) ニューラルネットワークの概要

a) ニューラルネットワークの基本構成

階層型ニューラルネットワークは、入力層・中間層・出力層から構成される (図-1)。ネットワークを構成す

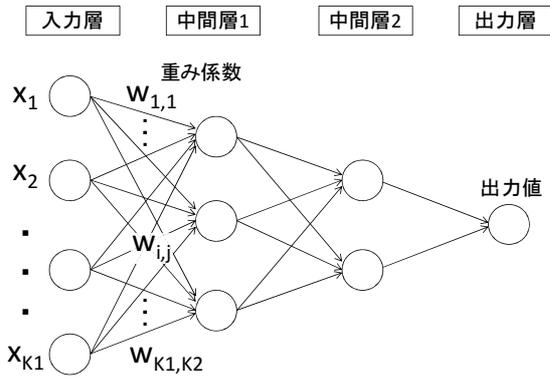


図-1 ANNの概念図.

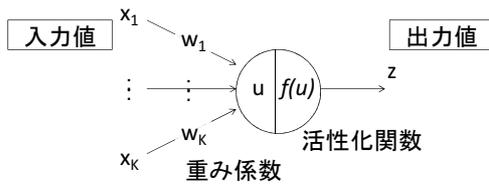


図-2 ANN構成素子の概念図.

る素子の模式図は図-2に示すとおりであり、各素子において以下のように計算を行う.

$$u = \theta_i + \sum_{i=1}^K w_i x_i \quad (1)$$

$$z = f(u) \quad (2)$$

ここで、 u は各素子の入力値、 x は入力値、 w は重み係数、 θ はバイアス、 K は各階層の構成素子数、 $f(u)$ は活性化関数、 z は素子の出力である. 本稿で以下はバイアスも含めたパラメータベクトル w を改めて重み係数と呼ぶ. なお活性化関数には様々な関数が使われるが、本検討では最も適用事例の多いシグモイド関数を用いた.

$$f(u) = \frac{1}{1 + \exp(-u)} \quad (3)$$

b) 勾配降下法によるネットワークの学習

ネットワークの学習では、出力層と目標出力 (実測値) との誤差が小さくなるように、各素子間の重みを勾配降下法によって最適化する. 出力と実績の誤差は以下の二乗誤差 E で評価する.

$$E(w) = \frac{1}{2} \sum_{n=1}^N (d_n - y(x_n; w))^2 \quad (4)$$

ここで、 N はサンプルデータ数、 d は目標出力、 y はネットワークの出力値である. ランダムに初期化された w に対して、学習データの各サンプルを用いて勾配降下法の計算を繰り返すことで、 E を極小化する w を得る. 勾配降下法では次式のように重みを更新する.

$$w^{(t+1)} = w^{(t)} - \epsilon \nabla E \quad (5)$$

ここで ϵ は学習係数であり、 w の更新量を決めるパラメータである. 添え字の t は学習のステップ数である.

c) 誤差逆伝搬法

勾配降下法の適用にあたっては、評価関数の勾配 ∇E を求める必要がある. 本検討では ∇E の算出は誤差逆伝搬法¹⁰⁾によって行った. 誤差逆伝搬法では、学習

データが与えられた時の各素子の誤差関数の勾配は次のように求められる.

$$\frac{\partial E_n}{\partial w_{ji}^{(l)}} = \delta_j^{(l)} z_i^{(l-1)} \quad (6)$$

ここで、 $\delta_j^{(l)} \equiv \frac{\partial E_n}{\partial u_j^{(l)}}$ はネットワーク第 l 層の u_j による E の微分である. $\delta_j^{(l)}$ は第 $l+1$ 層の諸量を用いて次式で表わされる.

$$\delta_j^{(l)} = \sum_k \delta_k^{(l+1)} (w_{kj}^{(l+1)} f'(u_j^{(l)})) \quad (7)$$

ここで、 f' は f の微分である. 以上より、式(6),(7)の適用により、出力層より順に計算することで、全てのネットワークにおける ∇E を算出することができる.

(2) 学習方法の詳細

a) 確率的勾配降下法

学習は確率的勾配降下法により行った. 確率的勾配降下法は、全学習サンプルに対して一部 (ミニバッチ) をランダムに取り出して誤差関数を評価するもの⁹⁾であり、本研究ではミニバッチのサイズは100とした.

b) 学習係数の設定

勾配降下法では、学習係数の設定が学習の成否を左右する重要な因子となる. 本検討での学習係数は、近年最もよく使われているAdaGrad¹¹⁾によって定めるものとした. AdaGradでは学習係数を以下のように設定する.

$$\epsilon = \frac{\epsilon_0}{\sqrt{\sum_{i=1}^T (\nabla E_i^{(t)})^2}} \quad (8)$$

ここで、 ϵ_0 は学習係数の初期値、 T は現時刻までの計算ステップ数である.

また、重みの修正量にモーメンタム⁹⁾を与えることにより、勾配降下法の収束性の向上を図った. モーメンタムを適用した際の重み係数の更新式は次式ようになる.

$$w^{(t+1)} = w^{(t)} - \epsilon \nabla E + \mu \Delta w^{(t-1)} \quad (9)$$

ここで、 μ は加算の割合を制御するパラメータである⁹⁾.

c) ドロップアウト

学習データの関連性を十分に学習し、かつ過学習による精度低下を避けるため、本研究ではドロップアウト¹²⁾を適用した. ドロップアウトは、学習計算時にネットワークの素子を確率 p の割合で無効化することで、ネットワークの自由度を強制的に小さくし過学習を避ける狙いがある. なお、学習終了時後の推論時には、素子からの出力を $(1-p)$ 倍することで、推論時に学習時よりも素子が増えることを補償する.

(3) ネットワークの事前学習

本研究では自己符号化器¹³⁾によって事前学習を行い、精度向上を図った. 自己符号化器では、図-3のように入力 x と出力 \hat{x} ができるだけ同じになるよう、重みの調

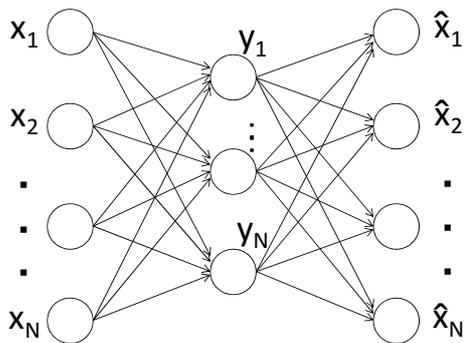


図-3 自己符号化器の概念図.

表-1 対象流域の観測所一覧.

種別	観測所数	観測所名
水位観測所	5	樋渡, 王子橋, 広瀬, 乙房, 岳下
雨量観測所	14	巢ノ浦, 野尻, 建設高原, 四家, 狭野, 霧島, 御池, 樋渡, 青井岳, 三股, 樺山, 岳下, 比曽木野, 末吉 (山田は未観測期間があるため除く)

整 (学習) を行う. また, 自己符号化器を拡張したデノイジング自己符号化器により, さらに精度の高い学習結果が報告されている¹⁴⁾. デノイジング自己符号化器では, 学習データに下記のようにノイズを加える.

$$\tilde{x} = x + \delta x \quad (10)$$

ここで, δ は平均0, 分散 σ^2 のガウス分布に従うランダムノイズである. 本研究ではケーススタディによりデノイジング自己符号化器の効果についても検討した.

3. 実河川への深層学習による水位予測の適用

(1) 対象流域

対象流域は, 大淀川水系の樋渡 (ひわたし) 地点流域とした (図-4). 流域面積は861km², 幹川流路延長は52 kmである. 周辺に雨量観測所が14箇所, 流域内に水位観測所が5箇所設置されている (表-1). 樋渡上流には大きな洪水調整施設がなく, また水位・雨量観測所が多数設置されているため, 検討に適している.

(2) 検討対象洪水

検討に用いた実績の水位と地上雨量データは, 水文・水質データベースより取得した. モデルの構築には, 1990年~2014年のはん濫注意水位 (6.0m) を超えた24洪水を用いた. 各洪水のピークから72時間前~48時間後までを1洪水とし, 全部で121時間×24洪水=2904セットの検討データを用意した. また, 1990年~2014年の間で氾濫危険水位 (9.2m) を超えた洪水は5事例あり, そのうち欠測データの多い1事例を除いた4事例 (1990年, 1993年, 2004年, 2005年) を精度評価の検討対象洪水とした.

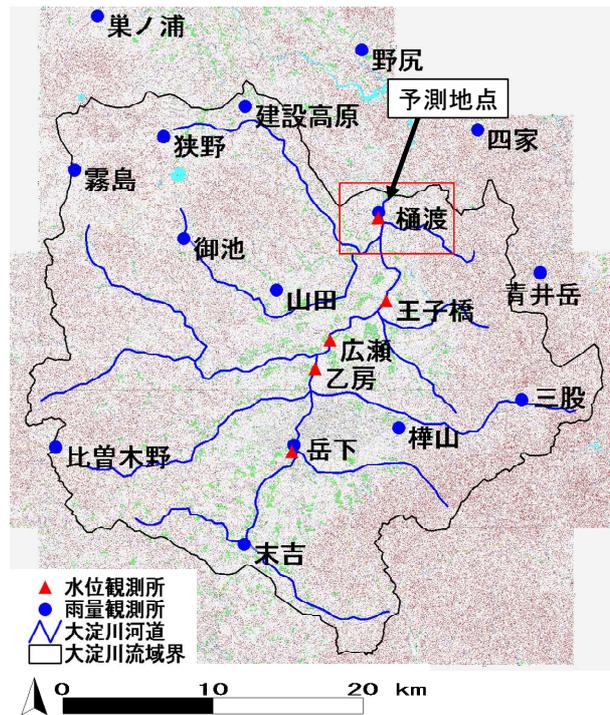


図-4 大淀川, 樋渡地点流域および観測所の位置図.

精度評価対象の4洪水の降雨の特徴をみると, 総雨量は検討対象24洪水中19位 (244mm), 2位 (608mm), 4位 (439mm), 1位 (759mm) となっており, 1990年洪水以外は総雨量が大きい. また時間雨量では8位 (31.8mm/h), 11位 (30.0mm/h), 10位 (30.1mm/h), 16位 (25.5mm/h) となっており, いずれも中位程度である. なおここでの雨量は検討対象14地点の平均雨量とし, 総雨量は検討対象の121時間の累積とした.

(3) 深層学習を用いた水位予測モデルの設定

a) ネットワーク構造

入力層, 2層の中間層および出力層からなる4層のネットワークとした. なお中間層が3層以上のネットワークも有力と考えられるが, 本研究では検討対象外とした.

b) 入出力データ

入出力層の組合せは表-2の通りとし, 1~6時間の予測時間ごとにネットワークを構築した. 入力データは, 水位予測に影響を及ぼしうる地点・時刻のデータをできる限り包含するように設定した. 著者らが相関分析によって調べた検討⁷⁾より, 樋渡水位変化に対する各水位観測所の水位変化の相関は, タイムラグが1,2時間の時に最大となったため, 入力層には現時刻から2時間前までのデータを設定した. また同様に各降雨観測所についてもタイムラグ1,2時間の時に相関が最大となったが, 累積雨量が影響する可能性を考慮して予測時刻から5時間前までのデータを設定した.

例えば3時間予測を行うネットワークでは, 出力層は現時刻から3時間後までの樋渡の水位変化, 入力層には樋渡自身の1時間前と現時刻の水位, 流域内5水位観測所

表-2 時刻0からt時間予測を行う場合の入出力層組合せ.

入力			出力
種別	地点	時刻	
時刻水位	樋渡	-1, 0	時刻0から時刻までにおける樋渡の水位変化
1時間の水位変化	流域内5地点	-2, -1, 0	
時刻雨量	流域内14地点	t-5, t-4, t-3, t-2, t-1	

表-3 学習の各種設定.

種別	観測所名
学習データ期間	1990~2014の上位24洪水
学習データ種類	樋渡水位, 上流水位変化, 時間雨量
データセット数	2904
バッチサイズ	100
予測時間	1時間~6時間
モーメンタム	0.5
学習率	初期値0.1としてAdaGradで自動設定
デノイズの加算ノイズ	平均0, 分散3%の正規乱数

表-4 学習の各種設定に関するスタディケース.

ケース番号	学習回数	各層の素子数	ドロップアウト率(ρ)
1	300	87-10-5-1	0.1
2	1000	87-20-10-1	0.3
3	3000	87-40-20-1	0.5
4	10000	87-60-30-1	

の2時間前~現時刻における1時間あたりの水位変化, 流域内14地点の2時間前~2時間後の時間雨量となる. 将来時刻の雨量には, 実測雨量を予測雨量に見立てた完全予測データを用いた. 将来時刻の水位は未知であるため, 水位変化は現時刻までのものを用いた.

以上より, 1つの学習データセットは87個のデータより構成される. 同様の組合せで, 全部で2904セットのサンプルを検証データとした.

c) 入力データの正規化

学習データが偏りを含む場合, 学習の効果が低下することが知られている. 本件検討では学習計算の実施前にデータの正規化を行った. データの正規化は次式で行う.

$$x_{ni} \leftarrow \frac{x_i - \bar{x}}{\sigma} \quad (11)$$

ここで, x_n は正規化後のデータ, \bar{x} はデータの平均, σ は標準偏差, 添え字の i はサンプルの組を表わす. なお, 計算の終了後, 式(11)の逆の手順を適用することにより求める予測値を得るものとした.

4. 予測計算の実施

(1) ケース設定

学習に関する各種の設定が, どのように予測精度に影響

するか不明である. 本研究ではケーススタディにより予測精度の比較を行った. 学習の基本設定は表-3の通りとし, ドロップアウト率(ρ), 学習回数, 各層の素子数については表-4の通りスタディケースを設定した. 各ケースについて自己符号化器にデノイズの設定あり・無しの場合を計算した.

(2) 予測計算の実施

検証対象4洪水に対してleave-one-out交差検証を行った. 具体的には, 対象24洪水のうち1洪水を検証データ, 残り23洪水を学習データとした計算を1セットとし, 同様の手順を検証対象4洪水について行うことで精度を評価した. なお精度評価の対象は, 水位が顕著に上昇し始めた時刻からピーク6時間後までの期間における水位のRMSEとした.

(3) ケーススタディの結果

ケーススタディの結果について, 1~6時間予測の平均RMSEをケースごとに整理したものを図-5に示す. 図中のケース番号は便宜的に表-4の番号に対応するものとし, 10の位が学習回数, 1の位が各層の素子数を表わすものとした. 例えばケース12は, 学習回数が300, 各層の素子数が(87-20-10-1)である. ドロップアウト率(ρ)およびデノイズの設定は, それぞれ異なる線で表示している. 図-5より, case33 ($\rho=0.1$, デノイズあり)の場合が最も再現性が高い結果となった. また, 各設定による予測精度への影響は下記の通りである.

学習回数については, 3000 (case31~34)とした場合が最も精度が高かった. ただし, 3000と10000 (case41~44)の差は明確でなく, 3000回程度で十分に学習が収束に近づいているものと推察される. 300 (case11~14)では精度が低く, 学習が不十分であると考えられる.

各層の素子数については, 素子数を増やすほど精度が高くなる傾向が見られた. ただし, (87-40-20-1)と(87-60-30-1)では明確な差が見られず, 例えばcase31~34では(87-40-20-1)のケースで誤差最小ケースが得られており, (87-60-30-1)では過学習の影響が表れ始めているものと推察される.

ドロップアウト率(ρ)については, 0.1にした場合が最も精度が高く, 次いで0.3, 0.5の順となった. ドロップアウト率は一般的に0.5程度に設定するケースが多いが, 本研究では異なる結果となり, 今後の検討が必要である.

デノイズの設定有無については, ほとんど差がなかったが, 設定した方が精度が良くなるケースがやや多い結果となった.

5. 他の水位予測手法との精度比較および考察

(1) 水位予測手法の比較対象

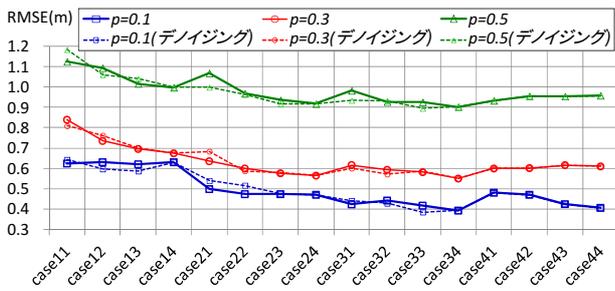


図-5 各ケースのRMSE (4洪水の1~6時間予測の平均)

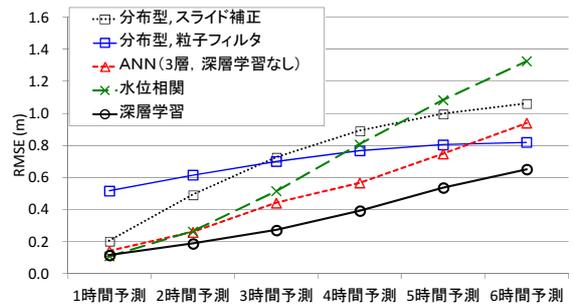


図-6 各予測モデルのRMSE (4洪水平均)

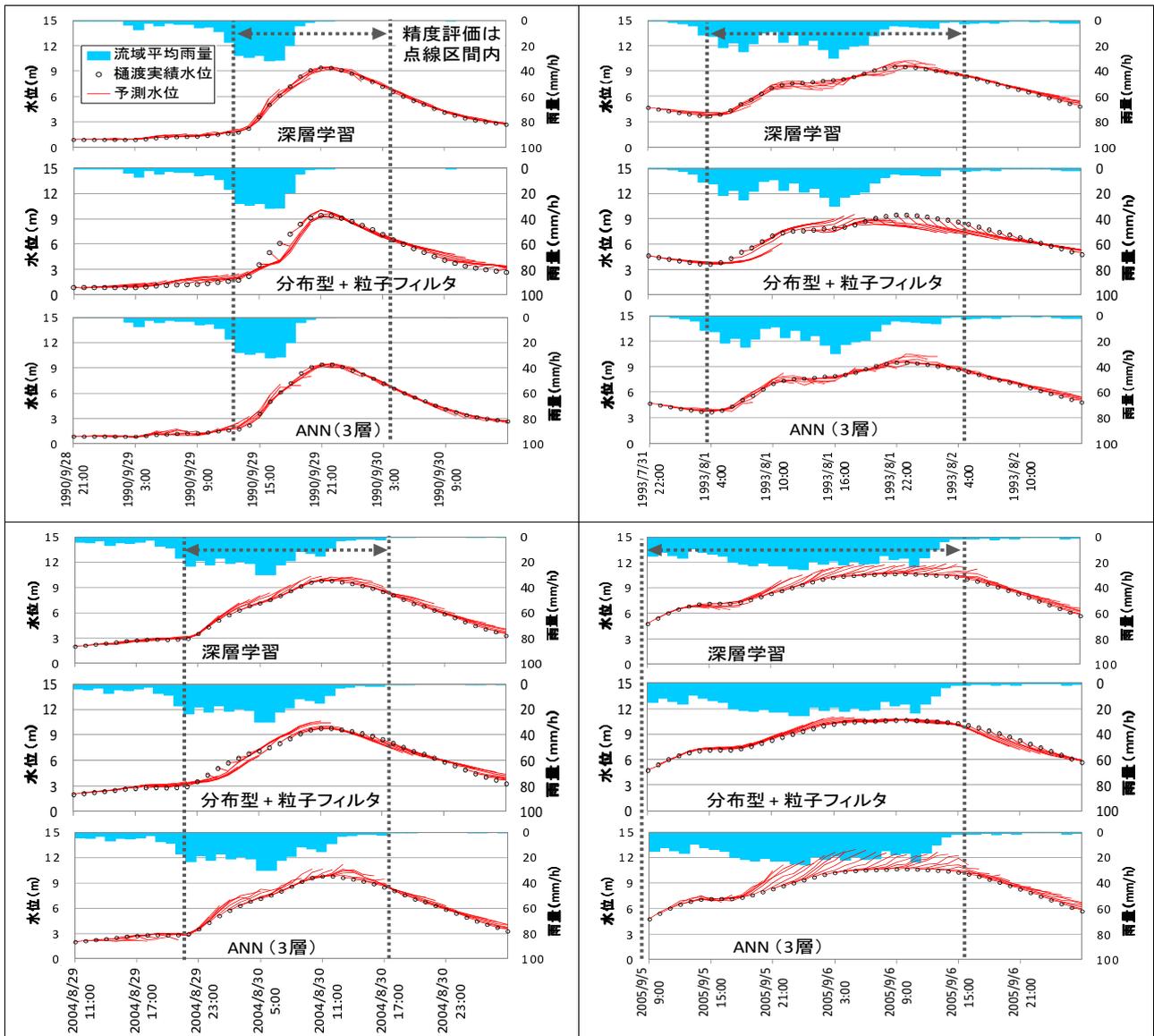


図-7 深層学習, 分布型+粒子フィルタ, およびANN(3層)による各洪水の水位予測結果 (1~6時間予測)

本研究で実施した計算結果と、他の手法による計算結果との比較を行った。比較対象は、①分布型モデル+スライド補正、②分布型モデル+粒子フィルタ、③ANNモデル (3層)、④水位相関モデル、である。①②④については文献⁷⁾に詳述されているためここでは割愛する。

③は洪水予測において最も実績例の多い3層の階層型ANNとし、深層学習は行わない。入出力データや学習

パラメータは深層学習モデルと同様とし、中間素子数は40、精度検証はleave-one-out交差検証より対象4洪水の予測精度を求めた。

なお、深層学習モデルとANN (3層) については、学習の際の重み係数の初期値をランダムに設定して10回の計算を行い、結果の算術平均をとったものを精度評価対象とした。

(2) 精度評価

本検討で構築した深層学習による水位予測結果と、①分布型モデル+スライド補正、②分布型モデル+粒子フィルタ、③ANNモデル（3層）、④水位相関モデルについて、精度を比較した。4洪水平均での1時間～6時間予測のRMSEを図-6に示す。また、深層学習と②③の水位予測結果を合わせて図-7に示す。図-7中の点線は精度評価期間を表わす。深層学習による水位予測結果は、他の4手法に比べて各予測時間で高い再現性を示した。

ANN（3層）との比較では、4洪水の各予測時間において、ほとんどの場合に深層学習の方が精度向上している。これはネットワーク構造が大きくなったことで表現性が増したと同時に、事前学習を始めとした適用手法により過学習が抑えられているためと考えられる。

分布型モデル（粒子フィルタ）との比較では、短い予測時間に対しては深層学習モデルの方が明確に精度が高い。これは、深層学習モデルでは短時間の水位上昇と相関の高い上流の観測水位情報を入力値に利用しているためと考えられる。長い予測時間に対しても深層学習モデルが上回るが、差は小さくなっている。これは上流の水位情報の影響が無くなり、相対的に降雨-流出過程の影響がより大きくなっていることから、分布型の優位性が出やすくなったためと考えられる。また2005年の洪水は、深層学習モデルが分布型よりも精度が低くなっている。2005年洪水は期間最大であり、ピーク付近の水位が他の洪水よりも高いため、学習データの事例不足となった可能性がある。他の3洪水については、いずれも深層学習モデルが分布型モデルを上回る精度となった。

(3) 計算機環境と計算時間

用いた計算機は、CPUが2機搭載のintel® Xeon®X5690（6-Core 3.46GHz）、メモリが96GB（DDR3）である。コンパイラはInter Composer XE 2011 Linux版（C++）を用いた。粒子フィルタによる6時間予測の計算時間は、MPI（openmpi-1.4.3）により96個の粒子を12コアに8個ずつ割り当て、74.0秒であった。深層学習モデルでは、学習には数時間を要したが、学習は事前に実施可能である。予測計算は0.1秒程度であったため、リアルタイム予測計算に要するコストはわずかである。

6. おわりに

本研究では、深層学習を用いた河川水位予測手法の開発を行い、実河川における適用性の確認を行った。提案手法の結果は、他の手法と比較して高い精度を示しており、今後の活用に向けた有効性が示唆される。

今後の課題として、様々な条件下にて本手法の適用性検証、特に未経験の洪水規模に対する検証は重要と考える。また入力データを含め、学習に関する様々な計算設

定条件・パラメータ等について知見の蓄積が必要である。

参考文献

- 1) 立川康人, 永谷言, 寶馨: 飽和・不飽和流れの機構を導入した流量流積関係式の開発, 水工学論文集, 第48巻, pp.7-12, 2004.
- 2) 鈴木俊明, 寺川陽, 松浦達郎: 実時間洪水予測のための分布型モデルの開発, 土木技術資料, Vol.38-10, pp.26-31, 1996.
- 3) ASCE Task Committee on Application of Artificial Neural Networks in Hydrology: Artificial neural networks in hydrology. II : Hydrologic Applications, Journal of Hydrologic Engineering, Vol.5, No.2, 2000.
- 4) Dawson, C.W. and Wilby, R.L.: Hydrological modeling using artificial neural networks, Progress in Physical Geography, Vol.25, No.1, 2001.
- 5) Maier, H.R. and Dandy, G.C.: Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications, Environmental Modelling & Software, Vol.15, 2000.
- 6) Maier, H.R., Jain, A., Dandy, G.C. and Sudheer, K.P.: Methods used for the development of neural networks for the prediction of water resource variables in river systems: Current status and future directions, Environmental Modelling & Software, Vol.25, 2010.
- 7) 一言正之, 桜庭雅明, 坂本俊二: 流域内のリアルタイム観測情報を用いた洪水予測の不確実性低減方法の比較, 河川技術論文集, 第21巻, pp.431-436, 2015.
- 8) Hinton, G.E., Osindero, S. and Teh, Y.: A fast learning algorithm for deep belief nets, Neural Computation, Vol.18, pp.1527-1544, 2006.
- 9) 岡谷貴之: 深層学習, 講談社サイエンティフィック, 2015.
- 10) Rumelhart, D.E. and McClelland, J.: Parallel distributed processing: Explorations in the microstructure of cognition, MIT Press, 1986.
- 11) Duchi, J., Hazan, E. and Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization, Journal of machine learning research, Vol.12, pp.2121-2159, 2011.
- 12) Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting, Journal of machine learning research, Vol.15, pp.1929-1958, 2014.
- 13) Bengio, Y., Lamblin, P., Popovici, D. and Larochelle, H.: Greedy layer-wise training of deep networks, Advances in Neural Information Processing Systems 19, pp.153-160, 2007.
- 14) Vincent, P., Larochelle, H., Lajoie, I., Bengio Y. and Manzagol, P. A.: Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion, Journal of Machine Learning Research, Vol.11, pp.3371-3408, 2010.

(2015. 9. 30受付)