System Identification by Cellular Neural Networks (CNN): Linear Interpolation of Nonlinear Weight Functions

Michael Reinisch, Gunter Geis and Ronald Tetzlaff

Institute of Applied Physics
Johann Wolfgang Goethe University,
Max-von-Laue-Str. 1, 60438 Frankfurt / Main, Germany

ABSTRACT

Recently CNN with nonlinear weight functions are used for various problems. Thereby nonlinear weights are represented by polynomials or tabulated functions combined with a cubic spline interpolation.
In this paper a linear interpolation technique is considered to allow an accurate approximation of nonlinear weight functions in CNN. In a previous publication the Table Minimising Algorithm (TMA) was introduced and applied to the Korteweg-de Vries-equation (KdV). In this contribution new results obtained by applying the algorithm to additional partial differential equations (PDE) will be given and discussed.

Keywords: CNN, linear interpolation technique, identification, optimisation

1. INTRODUCTION

Over the past years the importance of system identification by CNN has been shown in different investigations.[1, 2] Tabulated functions combined with a cubic spline interpolation have been introduced and analysed in detail for different cases.[3] First results[2] reveal that using linear interpolation on tabulated nonlinear functions is practicable and reasonable for a representation of non-linearities in CNN. In this contribution a more detailed analysis of this procedure will be given including cases which are sensitive to parameter variations. In the following the Burgers equation and the Φ^4-equation are investigated as two examples for nonlinear partial differential equations (PDE), which have been already represented by polynomial CNN.[4]
Firstly, the linear interpolation approach is introduced followed by results for the Burgers equation and Φ^4-equation.

2. METHODS

As already shown[4] nonlinear PDE can be represented by CNN with polynomial weight functions. In order to identify a nonlinear spatio-temporal system by a CNN, it is necessary to determine an accurate representation of the network weight function. Especially a mostly calculation efficient approximation may be useful allowing an application of the identification procedure on CNN based programable electronic devices. In this paper CNN with the state equations

$$c\frac{\partial s_i(t)}{\partial t} = -\frac{1}{r}s_i(t) + \sum_{k\in\mathcal{N}(i)} a_k\left(o(s_k(t))\right) + \sum_{k\in\mathcal{N}(i)} b_k\left(u_k\right) + q \qquad (1)$$

– c is a capacity and $\frac{1}{r}$ denotes a conductance – and the neighbourhood $\mathcal{N}(i)$ are used. For the output function $o(s_k(t))$ the identity is chosen. The input u_k, the bias q and the conductance are set to zero, which results in a state equation of an autonomous CNN. Therefore the state equation

$$\frac{\partial s_i(t)}{\partial t} = \sum_{k\in\mathcal{N}(i)} a_k\left(s_k(t)\right) \qquad (2)$$

Further author information: (Send correspondence to Michael Reinisch)
Michael Reinisch: E-mail: Reinisch@iap.uni-frankfurt.de, Telephone: +49 69 798 47456
Gunter Geis: E-mail: Geis@iap.uni-frankfurt.de, Telephone: +49 69 798 47455
Ronald Tetzlaff: E-mail: r.tetzlaff@iap.uni-frankfurt.de, Telephone: +49 69 798 47442

follows with the polynomial feedback function

$$a_k\left(s_k(t)\right) \quad = \quad \sum_{k\geq 0}^{D} p_{a,i,k} s_k(t)^k \ , \tag{3}$$

where D denotes the polynomial order. Usually, the polynomial weights will be determined in a comparison of the CNN state equation to the discretised form of the considered PDE.[4] Hereby, a linear interpolation method is applied for these functions to obtain an accurate but mostly calculation efficient representation form. Therefore we applied the Table Minimising Algorithm (TMA).[2] The TMA supports two arbitrary parameters, a maximal acceptable error ϵ and a window of size R, which must be appropriately chosen.

2.1. Burgers equation

A well known PDE is the Burgers equation[5]

$$\frac{\partial f(x,t)}{\partial t} \quad = \quad \frac{\partial f(x,t)^2}{\partial x} + \frac{\partial^2 f(x,t)}{\partial x^2} \ , \tag{4}$$

which is often used to model one dimensional flow processes. An analytical solution is given by

$$f(x,t) \quad = \quad \frac{v_1 \exp(-v_1(x - x_{01} - v_1 t)) + v_2 \exp(-v_2(x - x_{02} - v_2 t))}{1 + \exp(-v_1(x - x_{01} - v_1 t)) + \exp(-v_2(x - x_{02} - v_2 t))} \ . \tag{5}$$

Two soliton waves – with initial positions x_{01}, x_{02} and velocities v_1, v_2 – which merge at time

$$t \quad = \quad \frac{x_{02} - x_{01}}{v_1 - v_2} \tag{6}$$

are modeled by this solution.
In order to allow an identification by CNN it is necessary to perform a spatial discretisation. Therefore the spatial derivations have been substituted by the difference quotients. This leads to

$$\frac{\partial f(x_i,t)}{\partial t} \quad = \quad \frac{f(x_{i-1},t)^2 - f(x_{i+1},t)^2}{2h} + \frac{f(x_{i-1},t) - 2f(x_i,t) + f(x_{i+1},t)}{h^2} \ , \tag{7}$$

with the spatial discretisation width h. The template

$1.0s + 0.5s^2$	$-2.0s$	$1.0s - 0.5s^2$

results from Eqn. (7) with $h = 1$. The obtained nonlinear template with polynomial elements is an approximative representation of Eqn. (4) using CNN.
The determination of approximative solutions of certain PDE by calculating output activities of CNN following the above mentioned procedure has been treated in a lot of network simulations.[6,7] Especially representing arbitrary nonlinearities in programable electronic devices is still an open problem. Tabulation and interpolation[3] to model arbitrary nonlinearities may be one possible solution. In this contribution a linear interpolation procedure and the TMA are used for the Burgers equation as well as for the Φ^4-equation.

2.2. Φ^4-equation

The Φ^4 equation[5] is given by

$$\frac{\partial^2 f(x,t)}{\partial t^2} \quad = \quad \frac{\partial^2 f(x,t)}{\partial x^2} + f(x,t)^3 - f(x,t) \ . \tag{8}$$

One type of solution which describes a kink-antikink – f_+ denotes the kink and f_- the antikink – wave collisions is

$$f_{\pm}(x,t) = \pm \tanh\left(\frac{x - v_{\pm}t - x_{0\pm}}{\sqrt{2(1-v_{\pm}^2)}}\right) , \tag{9}$$

with initial positions x_{0+}, x_{0-} and velocities v_+, v_-.
Due to the fact that Eqn. (8) contains the second order derivation of time, we consider the form derived by the method of substitution[8]

$$w(x,t) = \frac{\partial f(x,t)}{\partial t} \tag{10}$$

$$\frac{\partial w(x,t)}{\partial t} = \frac{\partial^2 f(x,t)}{\partial x^2} + f(x,t)^3 - f(x,t) , \tag{11}$$

which – after spatial discretisation – can be represented by a two layer CNN. For the Φ^4-equation the sensitivity to certain initial conditions has already been shown.[4] In this paper the templates of a Φ^4-equation representing CNN have been obtained in a parameter training. The achieved templates are

from layer 1 to layer 2

$27.96638s$	$-55.49913s - 0.43353s^3$	$27.96638s$

from layer 2 to layer 1

0.5002

The further procedure is the same as for the Burgers equation described in section 2.1.

3. RESULTS

The results obtained by applying the above given method and the Table Minimising Algorithm (TMA) will be discussed in the following. The polynomial representation of the weights is given as a reference.

3.1. Burgers equation

The initial state loaded into the cells of the CNN has been calculated using Eqn. (5) with the parameters $x_{01} = 25$, $x_{02} = 75$ and $v_1 = 2$, $v_2 = -1$. In Fig. 1 the solution of Burgers equation using a polynomial representation is given, where two propagating wave fronts are shown. For a tabulated representation using a linear interpolation

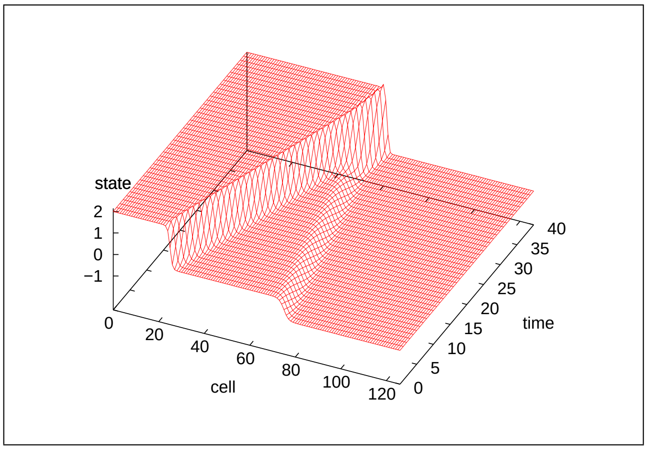

Figure 1. Burgers equation using a polynomial representation of the weight functions

of 650 equidistant sample points of the CNN feedback functions we obtain the solution shown in Fig. 2. In this

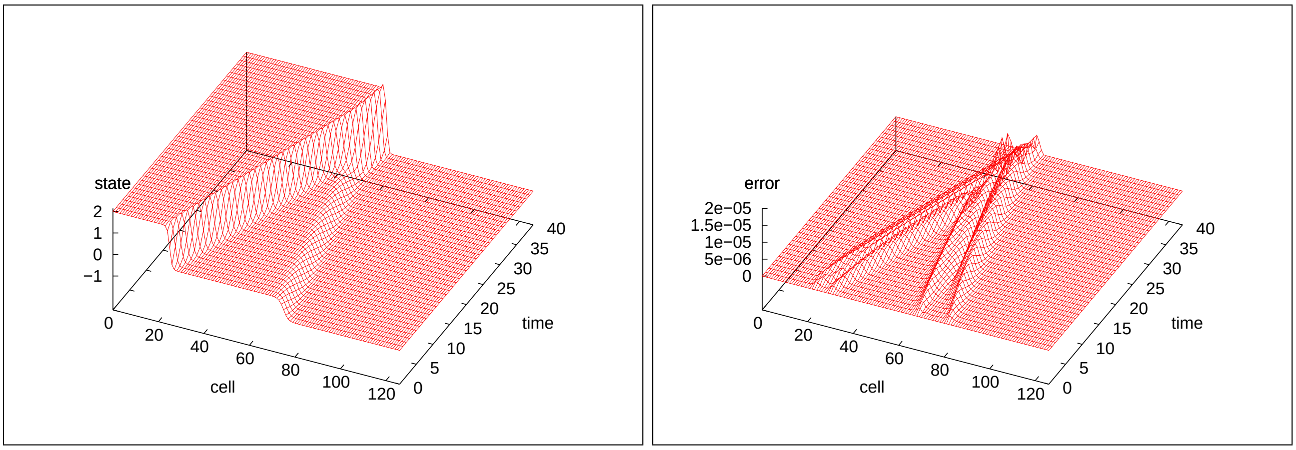

Figure 2. left: Solution of Burgers equation using a tabulated representation with 650 equidistant sample points for the weight functions; right: Difference of the solution obtained by a tabulated representation with 650 sample points to the reference solution in Fig. 1

case only small differences to the reference solution in Fig. 1 occur, which are caused by small differences in velocity of the propagating wave fronts in the two compared cases.
After using the TMA[2] with the settings $\epsilon = 0,1$ and $R = 50$ only 14 non-equidistant sample points for weight functions remain, which is a considerable reduction of the calculation complexity compared to the equidistant tabulated weight function. The obtained solution and the differences to the reference solution are shown in Fig. 3.

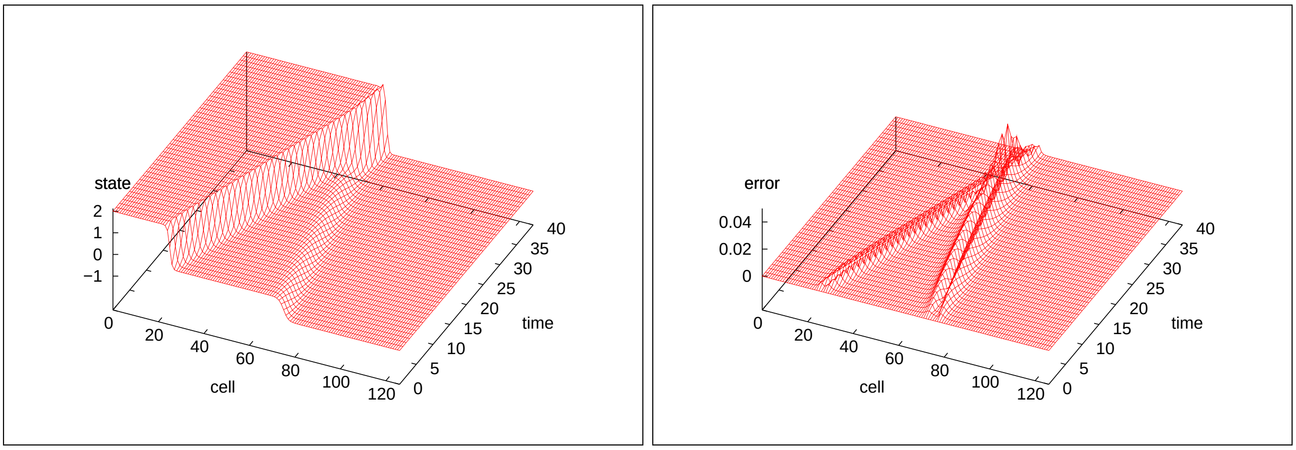

Figure 3. left: Solution of Burgers equation using a tabulated representation applying a TMA with 14 non-equidistant sample points for the weight functions; right: Difference of the solution obtained by a tabulated representation applying a TMA with 14 non-equidistant sample points for the weight functions to the reference solution in Fig. 1

3.2. Φ^4-equation

The initial state loaded into the cells of the CNN has been calculated using Eqn. (9) with the parameters $x_{0+} = 24$, $x_{0-} = 40$ and $v_+ = 0.1$, $v_- = -0.25$. In Fig. 4 a solution of Φ^4-equation using the polynomial representation is given. As already mentioned the shape of certain Φ^4-solutions depend sensetively on the initial conditions. Thus a special case is considered in this paper. In order to determine precise approximations of the reference solutions obtained with the polynomial CNN, a linear interpolation procedure with 1000 equidistant samples has been applied leading to the results which are given in Fig. 5.

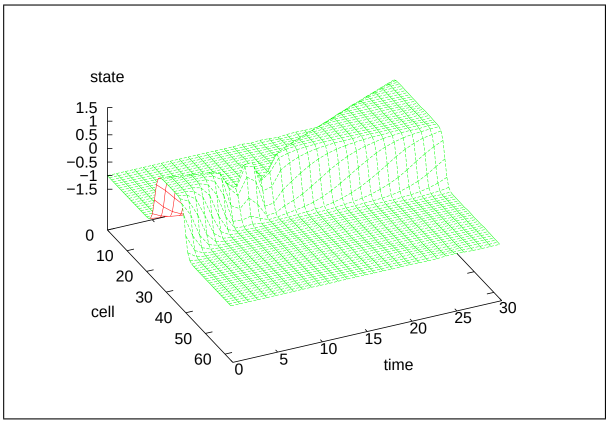

Figure 4. Solution Φ^4-equation using a polynomial representation of the weight functions

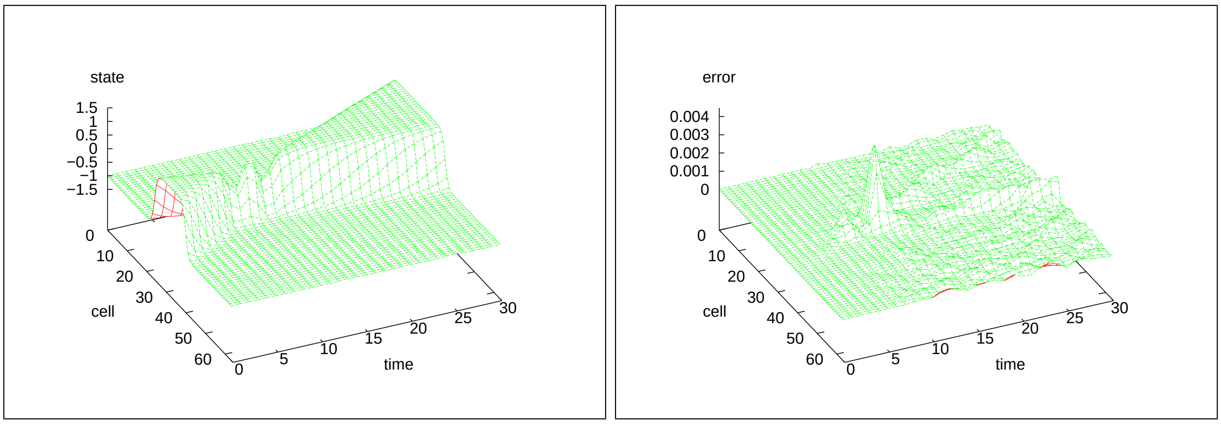

Figure 5. left: Solution of the Φ^4-equation using a tabulated representation with 1000 sample points for the weight functions; right: Difference of the solution obtained by a tabulated representation with 1000 sample points to the reference solution in Fig. 4

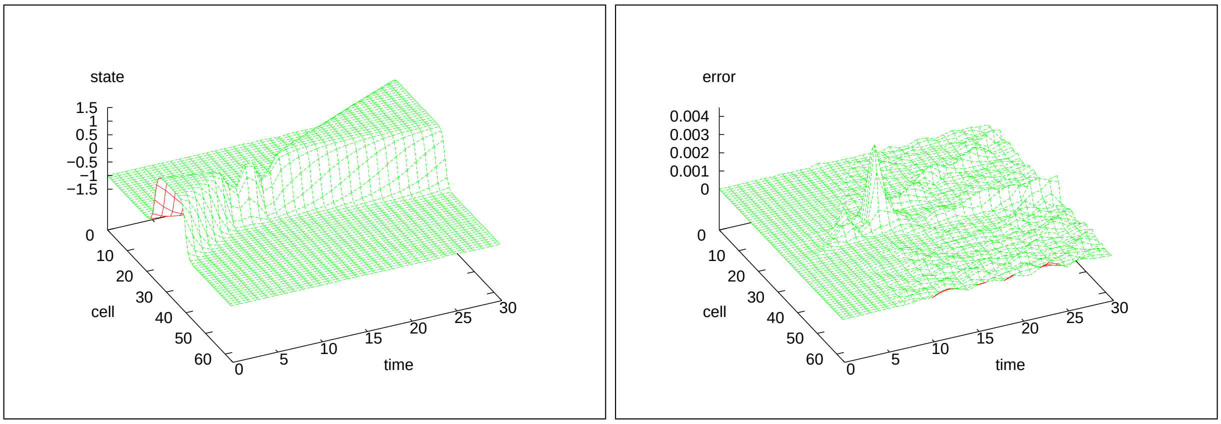

Figure 6. left: Solution of the Φ^4-equation using a tabulated representation applying a TMA with 744 sample points for the weight functions; right: Difference of the solution obtained by tabulated representation applying a TMA with 744 sample points for the weight functions to the reference solution in Fig. 4

Although, by using the TMA with $\epsilon = 0,000001$ and $R = 20$ the above given high number of samples decreased to 77%, this is still a considerable calculation complexity necessary for the linear interpolation of weight functions. In Fig. 6 the achieved result and the error are given. The obtained error is comparable to the error achieved with equidistant tabulated version.

4. CONCLUSION

The results of our investigations clearly show that the tabulated representation of nonlinear weight functions leads in all treated cases to accurate approximations of solutions obtained from different PDE. For the Burgers equation a considerable reduction of sample points could be achieved. For the more complex case of the Φ^4-equation it was shown that also a reduction of samples is obtained, nevertheless there is still a considerable calculation complexity compared to the Burgers equation. In further investigations a parameter training will be applied to obtain a tabulated representation of weight functions using a linear interpolation.

REFERENCES

1. F. Puffer, R. Tetzlaff, and D. Wolf, "A learning algorithm for cellular neural networks (cnn) solving nonlinear partial differential equations," in *ISSSE'95*, pp. 105–405, (San Francisco), 1995.
2. G. Geis, M. Reinisch, R. Tetzlaff, and F. Puffer, "Linear interpolation of nonlinearities in cellular neural networks," in *Proceedings of the* 8^{th} *IEEE International Biannual Workshop of CNNs and their Applications*, pp. 393–398, (Budapest), july 2004.
3. A. Loncar and R. Tetzlaff, "Cellular neural networks with nearly arbitrary nonlinear weight functions," in 6^{th} *CNNA'00*, Johann Wofgang Goethe University, (Seville), 2000.
4. F. Puffer, R. Tetzlaff, and D. Wolf, "Modeling nonlinear systems with cellular neural networks," in *ICASP'96*, pp. 3513–3516, 1996.
5. R. K. Dodd, J. C. Eilbeck, J. D. G. H. C., and Morris, *Solitons and Nonlinear Wave Equations*, Academic Press Inc. Ltd., London, 1984.
6. T. Roska, L. Chua, D. Wolf, T. Kozek, R. Tetzlaff, and F. Puffer, "Simulating nonlinear waves and partial differential equations via cnn - part i: Basic techniques," in *IEEE Trans. on Circuits and Systems*, **42,10**, pp. 807–815, 1995.
7. T. Kozek, L. Chua, T. Roska, D. Wolf, R. Tetzlaff, and F. Puffer, "Simulating nonlinear waves and partial differential equations via cnn - part ii: Typical examples," in *IEEE Trans. on Circuits and Systems*, **42,10**, pp. 816–820, 1995.
8. Bronstein, Semendjajew, Musiol, and Muehlig, *Taschenbuch der Mathematik*, Harri Deutsch, Frankfurt a.M., 4. ed., 1999.