

# Gamma-filter self-organising neural networks for unsupervised sequence processing

P.A. Estévez, R. Hernández, C.A. Perez and C.M. Held

Adding  $\gamma$ -filters to self-organising neural networks for unsupervised sequence processing is proposed. The proposed  $\gamma$ -context model is applied to self-organising maps and neural gas networks. The  $\gamma$ -context model is a generalisation that includes as a particular example the previously published merge-context model. The results show that the  $\gamma$ -context model outperforms the merge-context model in terms of temporal quantisation error and state-space representation.

**Introduction:** Self-organising feature maps (SOMs) [1] have been used extensively for clustering and data visualisation of static data. Recently, SOMs have been extended to sequence processing by using recurrent connections and temporal or spatial context representations; see [2] for a review. Engineering applications include the analysis of temporally or spatially connected data, such as DNA chains, biomedical signals, speech and image processing, and time series in general. In the merge SOM (MSOM) model [3], each neuron is associated with a weight vector and a context vector. The merge context model can be combined with neural gas (NG) [4], creating the merge neural gas (MNG) model [5]. In this Letter, adding  $\gamma$ -filters [6] to self-organising neural networks for unsupervised sequence processing is proposed. The  $\gamma$ -context model is applied to extend SOM and NG networks.

**Method:** We describe our method using the NG network model. A reference vector  $w^i \in \mathbb{R}^d$  is associated with the  $i$ th neuron, for  $i = 1, \dots, M$ , where  $d$  is the dimensionality of the input space and  $M$  is the number of neurons. A set of  $\gamma$ -filter vector contexts  $C = \{c_1^i, c_2^i, \dots, c_K^i\}$  is added to each  $i$ th neuron, where  $c_k^i \in \mathbb{R}^d$  for  $k = 1, \dots, K; i = 1, \dots, M$ , and  $K$  is the highest order  $\gamma$ -filter. When a data vector  $x(n)$  is presented to the network at time  $n$ , the best matching unit (BMU),  $I_n$ , is determined by computing the closest neuron according to the following recursive distance criterion or distortion error:

$$d_i(n) = \alpha_0 \|x(n) - w^i\|^2 + \sum_{k=1}^K \alpha_k \|C_k(n) - c_k^i\|^2 \quad (1)$$

where the parameters  $\alpha_k$ , for  $k = 0, \dots, K$ , control the contribution of reference and context vectors. Because the context construction is recursive, setting  $\alpha_0 > \alpha_1 > \dots > \alpha_K > 0$  is recommended. The  $K$  descriptors of the current context,  $C_k(n)$ , are recursively computed as the linear combination of the contexts of the  $k$ th and  $(k-1)$ th order  $\gamma$ -filter associated with the BMU obtained at the previous time step,  $I_{n-1}$ , i.e.

$$C_k(n) = \beta c_k^{I_{n-1}} + (1 - \beta) c_k^{I_{n-2}} \quad \forall k = 1, \dots, K \quad (2)$$

where  $c_k^{I_{n-1}} = w^{I_{n-1}}$ , and the initial conditions  $c_k^0, \forall k = 1, \dots, K$ , are set randomly. The parameter  $\beta \in (0, 1)$  provides a mechanism to decouple depth,  $D$ , and resolution,  $R$ , from the filter order. Depth indicates how far into the past the memory stores information. Resolution indicates the degree to which the information relative to the individual elements of the sequence is preserved. The mean memory depth for a  $\gamma$ -filter of order  $K$  is  $D = K/(1 - \beta)$ , and its resolution is  $R = 1 - \beta$  [6]. By increasing  $K$ , we can achieve a greater depth without compromising resolution. It is easy to verify that for  $K = 1$  in (2), i.e. when only a single  $\gamma$ -filter stage is used, the merge context model used in MSOM [3] and MNG [5] is recovered.

When an input vector  $x$  is presented, we determine the BMU, the second closest neuron, and so on. The ranking index associated with each  $i$ th neuron is denoted as  $r_i \in \{0, \dots, M-1\}$ . The neighbourhood function is defined in input space as  $h_\lambda(r_i) = \exp(-r_i/\lambda) \forall i = 1, \dots, M$ , where  $\lambda$  defines the neighbourhood size. The cost function for the  $\gamma$ -NG model is defined as:

$$E_{\gamma-ng}(w, c_1, \dots, c_K, \lambda) = \frac{1}{2H(\lambda)} \sum_{i=1}^M \int_V P(x) h_\lambda(r_i) d_i(x, w^i, c_1^i, \dots, c_K^i) dx \quad (3)$$

where  $V \subseteq \mathbb{R}^d$  is the input space,  $P(x)$  is the probability distribution of data vectors over the manifold  $V$ ;  $h_\lambda$  is the neighbourhood function;  $d_i$  is the distortion error, and  $H(\lambda)$  is a normalisation factor,  $H(\lambda) =$

$\sum_{i=1}^M h_\lambda(r_i)$ . When  $\alpha_k = 0, \forall k = 1, \dots, K$ , in (1), the  $\gamma$ -NG cost function (3) reduces to the original NG cost function [4]. Minimising (3) by stochastic gradient descent we obtain the following update rules for reference and context vectors:

$$\begin{aligned} \Delta w^i &= \varepsilon_w(n) h_\lambda(k_i) (x(n) - w^i) \\ \Delta c_k^i &= \varepsilon_c(n) h_\lambda(k_i) (C_k(n) - c_k^i) \end{aligned} \quad (4)$$

where the learning rates  $\varepsilon_w, \varepsilon_c$ , as well as  $\lambda$ , decrease monotonically from an initial value,  $v_0$ , to a final value,  $v_f$ , as  $v(n) = v_0(v_f/v_0)^{n/n_{\max}}$ , where  $n_{\max}$  is the maximum number of iterations.

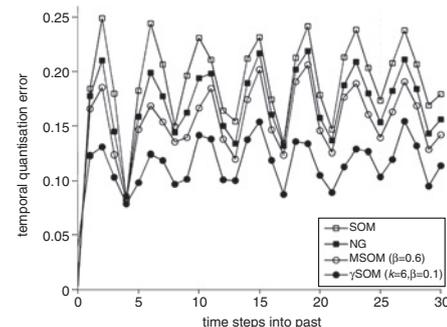
Temporal quantisation error (TQE) [7] is used to assess the representation of temporal dependencies in the map. The quantisation error of a neuron corresponds to the standard deviation associated with its receptive field. The TQE of the entire map for  $\tau$  time-steps back into the past is defined as the average quantisation error measured over all neurons. Recurrence plots are useful tools for visualising recurrences of dynamical systems in the phase space [8]. A recurrence matrix is defined where the  $ij$ th component is one if the states of the system at times  $i$  and  $j$  are similar up to an error  $\varepsilon$ , otherwise the corresponding entry in the matrix is zero. The parameter  $\varepsilon$  is fixed so that the recurrence rate is 0.02 [8]. The  $\gamma$ -NG model can be interpreted as a quantised reconstruction of the phase space in the  $K+1$  dimensional space by defining the vector  $r(t) = [w^{I_n}, c_1^{I_n}, \dots, c_K^{I_n}]$ .

*$\gamma$ -NG algorithm:*

1. Initialise randomly reference vectors  $w^i$  and context vectors  $c_k^i, i = 1, \dots, M; k = 1, \dots, K$ .
2. Present input vector,  $x(n)$ , to the network
3. Calculate context descriptors  $C_k(n)$  using (2)
4. Find the BMU,  $I_n$ , using (1)
5. Update reference and context vectors using (4).
6. Set  $n \rightarrow n+1$
7. If  $n < L$  go back to step 2, where  $L$  is the cardinality of the data set.

The  $\gamma$ -context model can easily be added to SOM, by replacing the neighbourhood function  $h_\lambda(k_i)$  in (4) by a neighbourhood function defined in the 2D output grid, typically a Gaussian centred in the winner unit  $i^*, h_\sigma(d_G(i, i^*)) = \exp(-d_G(i, i^*)/\sigma)$ , where  $d_G$  is the Manhattan distance, and  $\sigma$  is the neighbourhood size; see [9] for more details on  $\gamma$ -SOM.

Experiments were carried out with two data sets: Mackey-Glass time series [3] and Rössler's chaotic system [8]. Parameter  $\beta$  in (2) was varied from 0.1 to 0.9 with 0.1 steps. Likewise, the number of filter stages,  $K$ , was varied from 1 to 9. The number of neurons for both SOM (square grid) and NG variants was set to  $M = 0.1L$  where  $L$  is the length of the time series. Training is done in two stages lasting 100 epochs each. In the first stage, the parameters are set to  $\alpha_0 = 0.5$  and  $\alpha_k = 0.5/K, \forall k = 1, \dots, K$  in (1). This is done to favour a good convergence of the reference vectors in the first stage. In the second training stage, the parameters are set to  $\alpha_k = (K+1-k)/A, \forall k = 0, \dots, K$ , where  $A$  is determined by imposing the constraint  $\sum_k \alpha_k = 1$ . The initial and final values of parameters  $\lambda, \varepsilon_w, \varepsilon_c$  were set as follows:  $\lambda_0 = 0.1M, \lambda_f = 0.01, \varepsilon_{w0} = 0.3, \varepsilon_{wf} = 0.001, \varepsilon_{c0} = 0.3, \varepsilon_{cf} = 0.001$ .  $\sigma$  was set in the same way as  $\lambda$ .  $\varepsilon_w$  was fixed to 0.001 during the second training phase.



**Fig. 1** Temporal quantisation error against time-steps into past for Mackey-Glass time series, using SOM, NG, MSOM and  $\gamma$ -SOM

Figs. 1 and 2 illustrate the TQE as a function of the time-steps into the past for the Mackey-Glass time series, with  $L = 1000$ . Fig. 1 shows that  $\gamma$ -SOM obtained a much lower TQE than both SOM and MSOM; in addition, NG obtained a lower TQE than SOM. Fig. 2 shows that  $\gamma$ -NG with  $K > 1$  obtained a lower TQE than MNG ( $K = 1$ ). In addition MNG and  $\gamma$ -NG obtained, respectively, similar TQEs to MSOM and  $\gamma$ -SOM. For the 3D Rössler's chaotic system, we draw 4000 points from the first principal component. Fig. 3 illustrates the recurrence plot obtained with the  $\gamma$ -NG state-space reconstruction from the 1D Rössler time series. The latter produces an error of 3.74% when compared point to point with the recurrence plot of the original 3D Rössler system. In contrast, the error using NG is 6.85%, and using MNG is 5.31%.

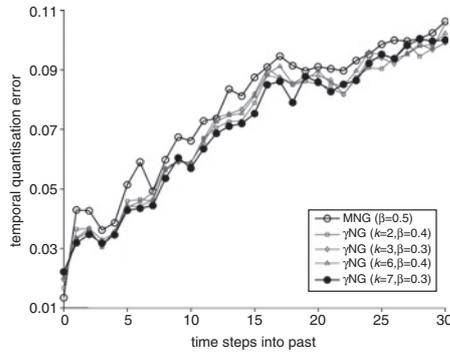


Fig. 2 Temporal quantisation error against time-steps into past for Mackey-Glass time series, using MNG, and  $\gamma$ -NG with different  $\beta$  and  $K$  parameters

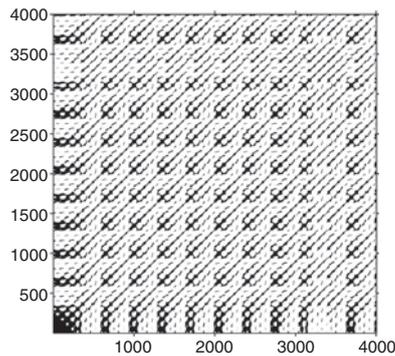


Fig. 3 Recurrence plot for  $\gamma$ -NG state-space reconstruction from 1D Rössler time series

**Conclusion:** Adding  $\gamma$ -filters to neurons allows extending self-organising neural networks to deal with unsupervised sequence processing. The proposed  $\gamma$ -context yielded significantly fewer temporal quantisation errors and better state-space representations than merge context models. Potential engineering applications of the proposed model include time series analysis, spatio-temporal maps, and the visualisation of temporally or spatially connected data.

**Acknowledgment:** This work was funded by Conicyt-Chile under grant Fondecyt no. 1080643.

© The Institution of Engineering and Technology 2011  
13 January 2011  
doi: 10.1049/el.2011.0115

P.A. Estévez, R. Hernández, C.A. Perez and C.M. Held (Department of Electrical Engineering and Advanced Mining Technology Center, Universidad de Chile, Casilla, Santiago 412-3, Chile)

E-mail: pestevez@cec.uchile.cl

### References

- 1 Kohonen, T.: 'Self-organizing maps' (Springer-Verlag, Berlin, Germany, 1995)
- 2 Salhi, M.S., Arous, N., and Ellouze, N.: 'Principal temporal extensions of SOM: overview', *Int. J. Signal Process. Image Process. Pattern Recognit.*, 2009, **2**, (3), pp. 121–144
- 3 Strickert, M., and Hammer, B.: 'Merge SOM for temporal data', *Neurocomputing*, 2005, **64**, pp. 39–72
- 4 Martinetz, T.M., Bercovich, S.G., and Schulten, K.J.: "'Neural-gas" network for vector quantization and its applications to time-series prediction', *IEEE Trans. Neural Netw.*, 1993, **4**, pp. 558–569
- 5 Strickert, M., and Hammer, B.: 'Neural gas for sequences'. Proc. Workshop on Self-Organizing Networks, (WSOM), Kyushu, Japan, 2003, pp. 53–58
- 6 Principe, J.C., de Vries, B., and de Oliveira, P.G.: 'The gamma filter – a new class of adaptive IIR filters with restricted feedback', *IEEE Trans. Signal Process.*, 1993, **41**, pp. 649–656
- 7 Strickert, M., Hammer, B., and Blohm, S.: 'Unsupervised recursive sequence processing', *Neurocomputing*, 2005, **63**, pp. 69–97
- 8 Marwan, N., Romano, M.C., Thiel, M., and Kurths, J.: 'Recurrence plots for the analysis of complex systems', *Phys. Rep.*, 2007, **438**, pp. 237–329
- 9 Estevez, P.A., and Hernandez, R.: 'Gamma SOM for temporal sequence processing', *LNCIS*, 2009, **5629**, pp. 63–71