LETTER

Design of a constant loop bandwidth phase-locked loop based on artificial neural network

Rui Guo¹, YuanCong Wu¹, ShaoGang Hu¹, Qi Yu¹, TuPei Chen², and Yang Liu^{1, a)}

Abstract An LC-VCO based phase-locked loop (PLL) frequency synthesizer which incorporates loop bandwidth tracking technique based on neural network is presented in this work. In order to minimize loop bandwidth variations, the proposed PLL employs an averaging varactor based split-tuned LC-VCO and a controllable phase error scaling module. Biasing voltages of varactor and the control word of the phase error scaling module are trained by the neural network and updated. The proposed technique can maintain a constant loop bandwidth over operating frequencies from 1.6-2GHz with variation varying from -2.66~3%.

Keywords: neural networks, MLP, PLL, bandwidth

Classification: Integrated circuits (memory, logic, analog, RF, sensor)

1. Introduction

To satisfy the growing demand of RF circuits especially for the multiband multimode transceiver, PLL is always hoped to have a wide tuning range [1, 2, 3, 4]. Compared with ring oscillator, LC-VCO which has the characteristics of low phase noise and wide tuning ranges optimized by the capacitor bank is always preferred [5, 6]. The main challenge of designing high performance PLL is the trade-off among these parameters such as phase noise, lock time, tuning range and VCO gain (K_{VCO}). When the tuning range is divided into several sub-bands, the tuning range is widened and K_{VCO} should be reduced to optimize the phase noise. However, PLL synthesizer bandwidth affects most of those parameters directly or indirectly [7, 8, 9, 10, 11, 12, 13, 14]. A lower loop bandwidth suppresses the in-band phase noise but extend the settling time. For LC-VCO, when the tuning range is widened or Process, Voltage, and Temperature (PVT) change, many parameters above may show variations. Keeping the loop bandwidth constant should be quite serious in the PLL design.

Many efforts have been made to keep the PLL loop bandwidth constant. Ref. [15] adjusts the current of charge pump (CP) to be proportional to the current of CCO. The variation of the CCO gain (K_{CCO}) decreases the loop bandwidth tracking ability. Ref. [16] adds the divider ratio N to improve

the tracking ability of the ring oscillator. Though both of these methods are effective to optimize the loop bandwidth, the application range is limited. Ref. [17] utilizes K_{VCO} calculator module to estimate the current of CP. In [18], loop bandwidth is optimized from two aspects: (1) the linearity of the VCO tuning curve and (2) the ratio of the CP current and divider ratio N . The variation of the loop bandwidth is limited to less than 4% for the whole lock range.

Artificial Neural Network (ANN), which is inspired by biological neural networks, can also be used in circuit design. Reference [19] proposed an ANN to improve FOM of ADC. Reference [20] realized a Power-Efficient Transmitter based on ANN to classify the input data. In [21], ANN also enlightened RF Power Amplifier. And [22] utilized the brain-like behavior of ANN to realize the gesture recognition.

In this work, neural network algorithm is applied in a LC-VCO PLL to keep the loop bandwidth being constant. Phase error of PFD is widened to adjust current. The linearity of the VCO and the final loop bandwidth are optimized by ANN. The variation of the loop bandwidth is limited to less than 3%.

2. Analysis of LC-VCO PLL loop bandwidth

Fig. 1 shows the structure of LC-VCO based charge-pump PLL optimized by the proposed ANN. When the PLL operates in the overdamped case, the closed loop bandwidth can be approximated by

$$\omega_{-3db} \approx \frac{I_{CP}}{2\pi} \cdot R_2 \cdot \frac{K_{VCO}}{N}, \quad (1)$$

where R_2 is the resistor of the loop filter; I_{CP} is the current of the charge pump; N is the divider ratio; and K_{VCO} is the VCO gain.

Considering the varactor, the gain K_{VCO} can be describes

Fig. 1 Schematic illustration of the proposed constant loop bandwidth PLL with ANN

¹ State Key Laboratory of Electronic Thin Films and Integrated Devices, University of Electronic Science and Technology of China, China

² School of Electrical and Electronic Engineering, School of Electrical and Electronic Engineering, Nanyang, China

^{a)} yliu1975@uestc.edu.cn

as

$$K_{VCO} = \frac{L}{2} \cdot \omega_{osc}^3 \cdot \left| \frac{\partial C_{var}}{\partial V_{ctrl}} \right|, \quad (2)$$

where L is the inductor; ω_{osc} is the operating frequency; and V_{ctrl} is the control voltage. Considering (1) and (2),

$$\begin{aligned} \omega_{-3db} &\approx \frac{I_{cp}}{2\pi} \cdot R_2 \cdot \frac{1}{N} \cdot \frac{L}{2} \cdot \omega_{osc}^3 \cdot \left| \frac{\partial C_{var}}{\partial V_{ctrl}} \right| \\ &= \frac{I_{cp}}{2\pi} \cdot R_2 \cdot \frac{L}{2} \cdot f_{ref}^3 \cdot N^2 \cdot \left| \frac{\partial C_{var}}{\partial V_{ctrl}} \right|, \end{aligned} \quad (3)$$

where f_{ref} is the reference frequency of the PFD. To keep the loop bandwidth constant, equations

$$\left| \frac{\partial C_{var}}{\partial V_{ctrl}} \right| = \text{constant} \quad (4)$$

and

$$I_{cp} \cdot N^2 = \text{constant} \quad (5)$$

should be maintained.

The variation of ΔV_{ctrl} on the control voltage caused by the phase error $\Delta\phi$ can be described as

$$\Delta V_{ctrl} = \frac{\Delta\phi}{2\pi} \cdot T_{PFD} \cdot \frac{I_{cp}}{C}, \quad (6)$$

where C is the capacitor of the loop filter. The variation of the current I_{cp} could be replaced by adjusting the width of the phase error. In this work, numerator “ M ” is introduced to further calibrate the width of the phase error as following

$$\begin{aligned} \Delta V_{ctrl_{scaling}} &= \left(\frac{M^2}{N^2} \cdot \Delta\phi \right) \frac{T_{PFD}}{2\pi C} \cdot I_{cp} \\ &= \Delta\phi \cdot \frac{T_{PFD}}{2\pi C} \left(\frac{M^2}{N^2} \cdot I_{cp} \right). \end{aligned} \quad (7)$$

The remaining of this paper focuses on achieving constant loop by improving the linearity of VCO and calibrating the phase error width with the proposed ANN.

3. Design of MLP-based constant loop bandwidth PLL

Building varactor arrays is the most common way to balance the VCO gain for the whole tuning range. Three sets of varactor pairs and biasing voltages are introduced into this design as shown in Fig 2(a). According to Eq. (1), a suitable set of biasing voltages can improve the linearity of varactor. Ref. [23] applied a neural network to optimize the biasing voltages. Compared with result in [23], more parameters like VCO gain, divider ratio and phase scaling should be considered to optimize loop bandwidth, according to Eqs. (3) (5) and (7). As shown in Figure 2(b), when a set of bias voltages [VB1, VB2, VB3] are confirmed, all of locking points $[f_0, f_1, \dots, f_n]$ and VCO gain $[K_{vcof0}, K_{vcof1}, \dots, K_{vcofn}]$ can be obtained. With the variation of [VB1, VB2, VB3], a mass data of K_{vco} and locking points are generated for next training process.

The method of calibrating the current of the CP is replaced by scaling the phase error. At the same time, individual phase error scaling module is introduced in this design as

Fig. 2 The schematic of (a) varactor arrays and (b) sub-bands distribution provided by split-tuned LC-VCO.

shown in Fig. 1. The width of phase error is counted and enlarged by the factor M^2/N^2 . According to the different M and N , the phase error could be widened and shrunken.

As shown in Fig. 3, when the value of N is higher than M , one extra period of reference frequency is needed to calculate the scaling. In this design, the reference frequency is 10MHz. An extra period (0.1 μ s) is added in the locking process. Compared with the locking time, the extra time cost could be accepted.

Fig. 3 Timing diagram of phase error scaling module.

Corresponding to the lock frequency points $[f_0, f_1, \dots, f_n]$, the numerator M is written as $[M_0, M_1, \dots, M_n]$. The value of M could be selected from

$$\begin{aligned} M &= [M_0, M_1, \dots, M_n] \\ &= [m \pm 1, m \pm 1, \dots, m \pm 1], \end{aligned} \quad (8)$$

where m is an integer. The loop bandwidth for different frequency points can be accurately compensated. Then the neural network is used to provide not only the optimized varactor bias voltages, but also a suitable set of M . Eqs. (1), (7) and (8) can be recombined as

$$\omega \approx \frac{I_{cp}}{2\pi} \cdot R_2 \cdot \frac{K_{VCO}}{N} \left(\frac{M}{N} \right)^2 \quad (9)$$

$$\omega_{fn} \approx \frac{I_{cp}}{2\pi} \cdot R_2 \cdot \frac{K_{VCO}}{f_n} \cdot f_{ref} \left(\frac{M_n}{f_n} \cdot f_{ref} \right)^2 \quad (10)$$

The loop bandwidth of each frequency point in the lock range can be obtained.

Considering that the Multilayer Perceptron (MLP) algorithm takes advantage in solving the regression problem [24], it is introduced into the constant loop bandwidth design. The MLP consists of one input layer, at least one hidden layer and one output layer. As shown in Fig. 4, bias voltages and different numerators are assigned to the input layer neurons. Several hidden layers are included to transform the current input into output based on the weights. This operation is also known as forward propagation. As VCO linearity and

constant loop bandwidth all should be considered in this design, two neurons are utilized at the output layer. The loss function is obtained by the label of the training dataset and the output of the forward propagation. In the backward propagation (BP), the layer weights are updated by reducing the loss [25, 26]. In this work, two steps are needed to complete the training. In Step A, a MLP is constructed to learn the concept of linearity and loop bandwidth coefficient of variation. This operation is a kind of fitting process. When Step A is finished with all the weights and bias confirmed, input neurons are then updated in the Step B with all weights and bias fixed as show in Fig. 5.

Fig. 4 The structure of MLP in this design.

Fig. 5 The workflow of the MLP to train and update inputs.

In this design, a 1.6-2 GHz LC-VCO PLL has been designed. When the reference frequency is chosen as 10MHz, the value of loop bandwidth for twenty-one frequency points are utilized to assess the variation. Corresponding to the frequency points, the divider ratio N varies from 80 to 100. To keep the expansion closed to contraction scaling, m is assigned to 90. Then each factor of M can only be set with 89 or 91. Then bias voltages VBIAS1, VBIAS2 and VBIAS3 are selected from {0, 0.15, 0.3, 0.45, 0.6, 0.75, 0.9, 1.05, 1.2} to obtain the whole samples for training linearity. At the same time, the capacitor array (5-bit) is utilized to generate 32 sub-bands for bandwidth calculation and calibration.

4. The detail of off-line training based on MLP

MLP is utilized to optimize the input voltage and M . The process can be described in two steps. Firstly, the network is trained to fit labels containing information for linearity and

coefficient of variation. As shown in Fig. 5, network takes bias voltages and M as input and output corresponding to linearity and coefficient of variation. Secondly, propagation is used to update the input voltage and M to keep the bandwidth constant. The detail of these two steps is described in Algorithm:

Step A:

```

Initialize MLP with parameter W
Set learning rate  $\alpha 1$ 
for each epoch: do
  if epoch mod 50 == 0
    do learning rate decay  $\alpha 1 \leftarrow \alpha 1/10$ 
  forward propagation to get output l and c
  calculate MSE loss L1
  back propagate the gradient to update the parameter of
  the network
   $W \leftarrow W - \alpha 1 * \partial L1 / \partial W$ 
end for

```

In step A, data are divided into training set and cross validation set at ratio of 4:1. The network is trained to fit the linearity and coefficient of variation with the guidance of relative error in cross validation set. Parameters such as learning rate, batch size, the number of epochs and the architecture of the network are updated continuously to achieve the best PLL performance. By using the back-propagation algorithm with gradient descent [27, 28, 29], all of weights and bias are updated. Finally, we set the learning rate $\alpha 1$ to be 0.0008 and batch size to be 32. Two hundred epochs are experienced until the final parameters are determined. All of the labels and loss could be calculated by following equations:

$$grad_i = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} \quad (11)$$

$$grad_{n-1} = \frac{y_n - y_{n-1}}{x_n - x_{n-1}} \quad (12)$$

$$E(grad) = \frac{1}{n-1} \sum_{i=1}^{n-1} grad_i \quad (13)$$

$$Label_l = D(grad) = \frac{1}{n-1} \sum_{i=1}^{n-1} [grad_i - E(grad)]^2 \quad (14)$$

$$E(\omega) = \frac{1}{21} \sum_{i=1}^{21} \omega_i \quad (15)$$

$$S(\omega) = \sqrt{\frac{1}{21} \sum_{i=1}^{21} [\omega_i - E(\omega)]^2} \quad (16)$$

$$Label_c = \frac{S(\omega)}{E(\omega)} \quad (17)$$

$$L1 = \frac{\sum_{i=1}^{21} (|predict_l - label_l|_i + 100 |predict_c - label_c|_i)}{m} \quad (18)$$

Step B:

Set learning rate α_2
 for each input vector X :
 for each epoch: do
 forward propagation to get output l and c
 calculate loss $L_2 = l^{(5/4)} + c^{(5/2)}$
 backpropagate the gradient to update the input
 vector $X \leftarrow X - \alpha_2 * \partial L_2 / \partial X$
 Select $x = \text{argmin}_x (5l + c)$
 end for
 end for

In step B, we take two inputs as a group. For each group, we update the inputs as described in Algorithm 1. After all inputs have been updated, we examine which one can achieve the minimum sum of output. In this step, we set the learning rate α_2 to be 0.00005 and 50 epochs for training.

5. Result and discussion

In the training process, the loss is used to describe the learning ability of the MLP as shown in Fig. 6. Loss is reduced gradually and finally tend to be stable. At the 200th epoch, there is no more improvement in loss. Then the MLP can be used to update the input bias voltage in the second step.

Fig. 6 The evolution of the loss in the training process.

When the training process is done and the input is updated, bias voltages are assigned to varactor. R-squared (R^2) [30] is used to evaluate the fitting result for the model. $R^2 = 1$ indicates a perfectly fitting. In the traditional varactor design, the bias voltages are set with 0, 0.6 and 1.2V, separately. R^2 is set to 0.99927. The bias voltages are updated with 0.46, 0.03 and 0.77V. Then R^2 of the new tuning curve is 0.99931. The linearity of the VCO tuning curve is kept and improved as shown in Fig. 7. K_{vco} of the tuning curve is shown as Fig. 8.

Besides the bias voltage, the scaling and sampling error also should be considered for VCO. When the numerator is higher than denominator, the value of scaling is decreased. On the other hand, the value of scaling is increased. The value of scaling can be obtained from

$$Scaling = Round \left(\left(\frac{Round \left(\frac{M}{N} \cdot 128 \right)}{128} \right)^2 \cdot \frac{f_{sample}}{f_{ref}} \right). \quad (19)$$

The area and power consumption of phase error scaling module can be estimated by back-end tool. For 130 nm process, the area cost is $43237 \mu m^2$. The power consumption is 5.206mw at 1.2V power supply.

In this design, the sample clock is assigned with 370MHz. Then the scaling and estimated scaling is shown in Fig. 9. When M has been updated by the MLP network, the scaling is updated as “op-ac” and “op-est” as shown in Fig. 9.

When the scaling and sampling error are count into the loop bandwidth, the normalized bandwidth is shown in Fig. 10. When scaling and sampling error are not considered, only varactor bias voltages affect the linearity of VCO and loop bandwidth as shown in red line in Figure 10. When M is a constant, the variation of the normalized bandwidth is shown with “M constant” line in Fig. 10. The variation of the bandwidth is controlled within $-2.606 \sim 3.317\%$. When M is updated by the MLP and assigned to scaling module, the optimized normalized bandwidth is shown as the yellow line in Fig. 10. Then the normalized bandwidth variation

Fig. 7 Tuning curve and optimized by the MLP compared with the traditional method.

Fig. 8 The K_{vco} of the tuning curve.

Fig. 9 The generation of the scaling error.

Fig. 10 The improvement of constant bandwidth after updating M.

Table I Comparison of bandwidth variation between references and this work

reference	[15]	[17]	[18]	This
Band-width Variation	$\pm 7.5\%$	$\pm 2\%$	$\pm 4\%$	$\pm 3\%$
oscillator type	RCCO	LC-VCO	LC-VCO	LC-VCO
control algorithm	Adjusts CP	Kvco calculator module	VCO linearity & divider ratio	Neural Network Algorithm
frequency multiplication ratio	0.9-2.2 GHz	1.9-3.8 GHz	3.1-3.9 GHz	1.6-2 GHz

varies from 2.66~3%.

In this design, only a small digital circuit is used in the PLL to scale the phase error. Compared with other works in Table I, the variation of the loop bandwidth is quietly limited with tiny cost. The tuning curve linearity of CCO and LC-VCO are always different. The improvement is more

apparent when compared with work [18].

6. Conclusion

A technique based on MLP artificial neural network to maintain a constant loop bandwidth of an LC-VCO PLL frequency synthesizer is presented. Considering power consumption and chip area, only a small digital module is needed to calibrate the bandwidth. The use of MLP network can maintain a constant loop bandwidth over operating frequencies from 1.6-2GHz with variation varying from $-2.66\sim 3\%$.

Acknowledgments

This work was supported by NSFC under project No. 61774028 and 61771097. (Corresponding author: Yang Liu)

References

- [1] H. Yoon, *et al.*: "A 0.56–2.92 GHz wideband and low phase noise quadrature LO-generator using a single LC-VCO for 2G-4G multi-standard cellular transceivers," *IEEE J. Solid-State Circuits* **51** (2016) 614 (DOI: [10.1109/JSSC.2015.2510019](https://doi.org/10.1109/JSSC.2015.2510019)).
- [2] M.K. Hati and T.K. Bhattacharyya: "A fast and efficient constant loop bandwidth with proposed PFD and pulse swallow divider circuit in $\Delta\Sigma$ fractional-N PLL frequency synthesizer," *Microelectronics Journal* **61** (2017) 21 (DOI: [10.1016/j.mejo.2016.12.014](https://doi.org/10.1016/j.mejo.2016.12.014)).
- [3] Y. Peng, *et al.*: "Low-phase-noise wideband mode-switching quad-core-coupled mm-wave VCO using a single-center-tapped switched inductor," *IEEE J. Solid-State Circuits* **53** (2018) 3232 (DOI: [10.1109/JSSC.2018.2867269](https://doi.org/10.1109/JSSC.2018.2867269)).
- [4] M.K. Hati and T.K. Bhattacharyya: "A fast automatic frequency and amplitude control LC-VCO circuit with noise filtering technique for a fractional-N PLL frequency synthesizer," *Microelectronics Journal* **52** (2016) 134 (DOI: [10.1016/j.mejo.2016.03.014](https://doi.org/10.1016/j.mejo.2016.03.014)).
- [5] D. Ponton, *et al.*: "LC-VCO in the 3.3- to 4-GHz band implemented in 32-nm low-power CMOS technology," *IEEE Trans. Circuits Syst. II, Express Briefs* **58** (2011) 467 (DOI: [10.1109/TCSII.2011.2160033](https://doi.org/10.1109/TCSII.2011.2160033)).
- [6] E. Zailer, *et al.*: "8-GHz, 6.6-mW LC-VCO with small die area and FOM of 204 dBc/Hz at 1-MHz offset," *IEEE Microw. Wireless Compon. Lett.* **26** (2016) 936 (DOI: [10.1109/LMWC.2016.2615024](https://doi.org/10.1109/LMWC.2016.2615024)).
- [7] C. Fang, *et al.*: "Impacts of phase-locked loop bandwidth on millimeter wave three-dimensional imaging system," *IEICE Electron. Express* **16** (2019) 20190187 (DOI: [10.1587/elex.16.20190187](https://doi.org/10.1587/elex.16.20190187)).
- [8] P. Ye, *et al.*: "Novel optimal bandwidth design in INS-assisted GNSS phase lock loop," *IEICE Electron. Express* **8** (2011) 650 (DOI: [10.1587/elex.8.650](https://doi.org/10.1587/elex.8.650)).
- [9] M. Ugajin: "Proposal of channel-grouping wireless-transceiver architecture for suppressing local-oscillator phase noise," *IEICE Electron. Express* **9** (2012) 86 (DOI: [10.1587/elex.9.86](https://doi.org/10.1587/elex.9.86)).
- [10] Z. Xian, *et al.*: "A low jitter phase-locked loop based on self-biased techniques," *IEICE Electronics Express* **12** (2015) 20150597 (DOI: [10.1587/elex.12.20150597](https://doi.org/10.1587/elex.12.20150597)).
- [11] Y. Feng, *et al.*: "A fast-locking fractional-N frequency synthesizer using a new variable bandwidth method," *IEICE Electron. Express* **10** (2013) 20130373 (DOI: [10.1587/elex.10.20130373](https://doi.org/10.1587/elex.10.20130373)).
- [12] K.-C. Kuo: "A 2.4GHz low phase noise frequency synthesizer for WiMAX applications," *IEICE Electron. Express* **8** (2011) 938 (DOI: [10.1587/elex.8.938](https://doi.org/10.1587/elex.8.938)).
- [13] Z. Zhang, *et al.*: "A 1.25-to-6.25 GHz -237.2-dB FOM wideband self-biased PLL for multi-rate serial link data transmitter," *IEICE Electron. Express* **14** (2017) 20170422 (DOI: [10.1587/elex.14.20170422](https://doi.org/10.1587/elex.14.20170422)).
- [14] W. Zou, *et al.*: "A wideband low-jitter PLL with an optimized Ring-VCO," *IEICE Electron. Express* **17** (2020) 20190703 (DOI: [10.1587/elex.17.20190703](https://doi.org/10.1587/elex.17.20190703)).
- [15] Z. Zhang, *et al.*: "A 0.9–2.25-GHz sub-0.2-mW/GHz compact low-voltage low-power hybrid digital PLL with loop bandwidth-tracking technique," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **26**

- (2018) 933 (DOI: [10.1109/TVLSI.2018.2797280](https://doi.org/10.1109/TVLSI.2018.2797280)).
- [16] J. Kim, *et al.*: “Design of CMOS adaptive-bandwidth PLL/DLLs: a general approach,” *IEEE Trans. Circuits Syst. II. Exp. Briefs* **50** (2003) 860 (DOI: [10.1109/TCSII.2003.819120](https://doi.org/10.1109/TCSII.2003.819120)).
 - [17] J. Shin and H. Shin: “A 1.9–3.8 GHz $\Delta\Sigma$ fractional-N PLL frequency synthesizer with fast auto-calibration of loop bandwidth and VCO frequency,” *IEEE J. Solid-State Circuits* **47** (2012) 665 (DOI: [10.1109/JSSC.2011.2179733](https://doi.org/10.1109/JSSC.2011.2179733)).
 - [18] T. Wu, *et al.*: “Method for a constant loop bandwidth in LC-VCO PLL frequency synthesizers,” *IEEE J. Solid-State Circuits* **44** (2009) 427 (DOI: [10.1109/JSSC.2008.2010792](https://doi.org/10.1109/JSSC.2008.2010792)).
 - [19] L. Danial, *et al.*: “Breaking through the speed-power-accuracy tradeoff in ADCs using a memristive neuromorphic architecture,” *IEEE Trans. Emerg. Topics Comput. Intell.* **2** (2018) 396 (DOI: [10.1109/TETCI.2018.2849109](https://doi.org/10.1109/TETCI.2018.2849109)).
 - [20] D.Y. Kong, *et al.*: “Realization of a power-efficient transmitter based on integrated artificial neural network,” *IEEE Access* **6** (2018) 68773 (DOI: [10.1109/ACCESS.2018.2880033](https://doi.org/10.1109/ACCESS.2018.2880033)).
 - [21] F. Mkadem and S. Boumaiza: “Physically inspired neural network model for RF power amplifier behavioral modeling and digital pre-distortion,” *IEEE Trans. Microw. Theory Techn.* **59** (2011) 913 (DOI: [10.1109/TMTT.2010.2098041](https://doi.org/10.1109/TMTT.2010.2098041)).
 - [22] G.Y. Liu, *et al.*: “Smart electronic skin having gesture recognition function by LSTM neural network,” *Appl. Phys. Lett.* **113** (2018) 084102 (DOI: [10.1063/1.5040413](https://doi.org/10.1063/1.5040413)).
 - [23] R. Guo, *et al.*: “Design of a neural network-based VCO with high linearity and wide tuning range,” *IEEE Access* **7** (2019) 60120 (DOI: [10.1109/ACCESS.2019.2915335](https://doi.org/10.1109/ACCESS.2019.2915335)).
 - [24] R.P. Brent: “Fast training algorithms for multilayer neural nets,” *IEEE Trans. Neural Netw.* **2** (1991) 346 (DOI: [10.1109/72.97911](https://doi.org/10.1109/72.97911)).
 - [25] Y. Liu, *et al.*: “A stochastic computational multi-layer perceptron with backward propagation,” *IEEE Trans. Comput.* **67** (2018) 1273 (DOI: [10.1109/TC.2018.2817237](https://doi.org/10.1109/TC.2018.2817237)).
 - [26] K. Sun, *et al.*: “Design and application of a variable selection method for multilayer perceptron neural network with LASSO,” *IEEE Trans. Neural Netw. Learn. Syst.* **28** (2017) 1386 (DOI: [10.1109/TNNLS.2016.2542866](https://doi.org/10.1109/TNNLS.2016.2542866)).
 - [27] D.W. Ruck, *et al.*: “The multilayer perceptron as an approximation to a Bayes optimal discriminant function,” *IEEE Trans. Neural Netw.* **1** (1990) 296 (DOI: [10.1109/72.80266](https://doi.org/10.1109/72.80266)).
 - [28] J.W. Watterson: “An optimum multilayer perceptron neural receiver for signal detection,” *IEEE Trans. Neural Netw.* **1** (1990) 298 (DOI: [10.1109/72.80267](https://doi.org/10.1109/72.80267)).
 - [29] Y.-R. Chien, *et al.*: “A multilayer perceptron-based impulsive noise detector with application to power-line-based sensor networks,” *IEEE Access* **6** (2018) 21778 (DOI: [10.1109/ACCESS.2018.2825239](https://doi.org/10.1109/ACCESS.2018.2825239)).
 - [30] A.C. Cameron and F.A.G. Windmeijer: “An R-squared measure of goodness of fit for some common nonlinear regression models,” *J. Econometrics* **77** (1997) 329 (DOI: [10.1016/s0304-4076\(96\)01818-0](https://doi.org/10.1016/s0304-4076(96)01818-0)).