B-spline based repetitive controller revisited: error shift, higher-order polynomials and smooth pass-to-pass transition

Michal Malkowski, Bartlomiej Ufnalski and Lech M. Grzesiak
Institute of Control and Industrial Electronics
Faculty of Electrical Engineering, Warsaw University of Technology
75 Koszykowa Str., Warsaw 00-662, Poland
{michal.malkowski, bartlomiej.ufnalski, lech.grzesiak}@ee.pw.edu.pl

***Abstract*—The paper presents a modified version of a training algorithm for a B-spline network (BSN). The algorithm is used as a controller for a constant-amplitude constant-frequency voltage-source inverter (CACF VSI) with an LC output filter. The modification is divided into three parts. Firstly, there are B-spline functions of higher orders that are used especially functions of a second order. Secondly, the B-spline functions are described in cyclic manner which allows to achieve smooth control signal in steady state. Last modification involves introduction of a shift between the error and the control signal. It improves convergence of the algorithm and allows to achieve a long term stability without using the forgetting factor.**

***Index Terms*—repetitive control, iterative learning control, B-spline network, sine wave inverter, disturbance rejection, power quality.**

I. Introduction

In the case of plants in which reference signal and disturbances have repetitive characteristic, control schemes that take this property into account are very promising. One of them is iterative learning control (ILC) which is used in many applications: [1], [2], [3], [4], [5], [6]. The operating principle of ILC is described by the following equation:

$$u(p,\ k) = u(p,\ k-1) + k_{\mathrm{RC}} e(p,\ k-1), \tag{1}$$

where: u - control signal, e - control error, k - number of passes, p - number of time samples along a pass, k_{RC} - repetitive control gain. Classic ILC ensures the convergence of control error close to zero. However it is affected by the lack of long term stability [5], [7], [8]. This problem is often solved by filtering the control error and/or control signal, which may be written as

$$u(p,\ k) = Q(z^{-1})u(p,\ k-1) + L(z^{-1})e(p,\ k-1), \tag{2}$$

where Q and L denote the control signal filter and the control error filter respectively. However, such solution has some disadvantage. There is no straightforward method of designing the Q and L filters. Usually these are designed as low pass filters, but their exact parameters are produced by using a guess and check method. There are numerous ways of solving the long term stability problem [7], [9], [10]. One of them is to use iteratively trained B-spline network (BSN) as a repetitive controller (RC) [11]. The advantage of this approach lies in convenience of designing BSN parameters. Use of BSN RC is wide and involves applications in silicon wafer production [12], controlling ink jet printers [13], as well as controlling CACF VSIs [14].

This paper presents simulational and experimental results of modified version of BSN controller applied to CACF VSI with an output LC filter.

II. BSN controller

B-spline is a function piecewise-defined by polynomials. The zero order B-spline (defined by zero order polynomial function) is described by

$$\mu^1(t, t_s, d) = \begin{cases} 1 & \text{if } t_{\mathrm{s}} \le t < t_{\mathrm{s}} + d \\ 0 & \text{if otherwise} \end{cases} \tag{3}$$

where t denotes time, t_{s} is the start time of a B-spline and d is the duration of a B-spline. Higher order B-spline can be generated by

$$\begin{aligned} \mu^n(t, t_{\mathrm{s}}, d) = {} & \frac{(t - t_{\mathrm{s}})}{d} \mu^{n-1}(t, t_{\mathrm{s}}, d - \frac{d}{n}) + \\ & \frac{(t_{\mathrm{s}} + d - t)}{d} \mu^{n-1}(t, t_{\mathrm{s}} + \frac{d}{n}, d - \frac{d}{n}), \end{aligned} \tag{4}$$

where μ^n denotes n-th order B-spline. The examples of zero, first and second order B-splines with parameters $t_{\mathrm{s}} = 0.4$ and $d = 1$ are shown in Fig. 1.

The B-spline network is a function approximator. It consists of a finite (n) set of B-spline function generators (μ) which

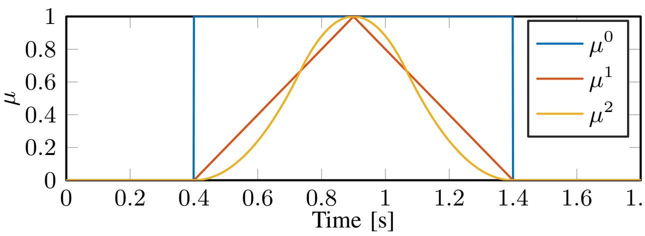
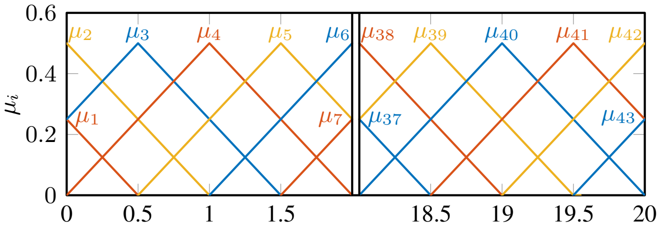
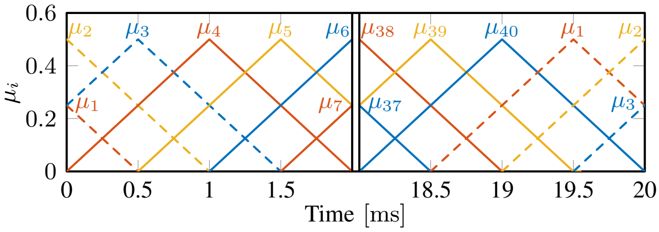


Fig. 1. B-splines of zero, first and second order



(a) B-splines arranged in noncyclic manner (first three functions have different weights than last three functions)



(b) B-splines arranged in cyclic manner (first three functions are connected with last three functions)

Fig. 2. Exemplary set of cyclic and noncyclic B-splines

have phase shift between each other and every one of them is linked with a scalar called weight (w). The output of BSN is calculated by multiplying each B-spline by corresponding weight and then summing them up, which is shown in

$$y_{\mathrm{BSN}} = \sum_{i=1}^{n} \mu_i w_i. \tag{5}$$

where y_{BSN} denotes output signal of the BSN. In the case of 75% overlap between B-splines the μ_i is defined as

$$\mu_i(t) \coloneqq \mu\left(t, \frac{(i-1)T}{n}, \frac{4T}{n}\right), \tag{6}$$

where T denotes time of an iteration. In the case of discrete realization (6) can be rewritten as

$$\mu_i(p) \coloneqq \mu\left(pT_{\mathrm{s}}, \frac{(i-1)NT_{\mathrm{s}}}{n}, \frac{4NT_{\mathrm{s}}}{n}\right), \tag{7}$$

where T_{s} is the sampling time and N is the length of pass in terms of samples.

The BSN RC produces control signal by training the B-spline network after each iteration of the process. The aim of the training procedure is to generate control signal which brings error close to zero. This can be achieved using the following control law [14]:

$$w_i(k) = (1-\lambda)w_i(k-1) - \gamma \sum_{p=0}^{N-1} \mu_i(p) e(p,\ k-1). \tag{8}$$

where λ is the forgetting factor and γ denotes the learning factor.

III. Modifications of the BSN controller

In order to improve steady-state performance, the algorithm of the BSN controller [14] was modified. The modification is divided into three main parts.

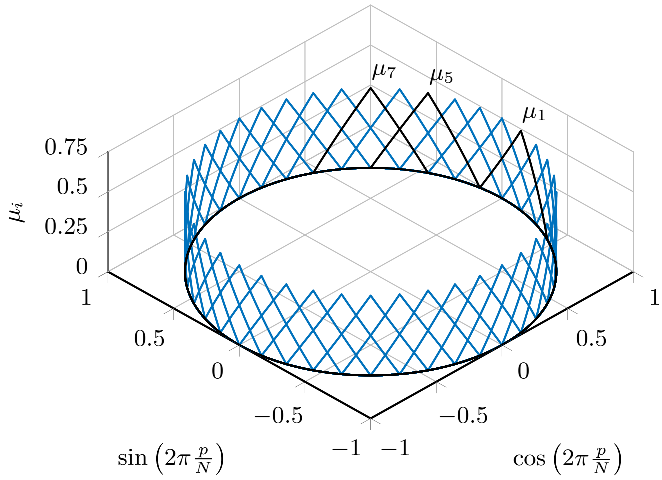


Fig. 3. Cyclic B-splines

A. Change of B-spline order

First modification involves change of the basis function to a second order B-splines. It was mentioned in [14] but the suggestion was that this would increase computational effort needed to calculate the output. This can be avoided by implementing basis function as a look-up table. Such approach makes the computational time dependent only on the length of the basis function, not on its shape, thus there is no increase in computational effort between the first and the second order based BSN RC. Performance comparison of controllers with different B-spline order are presented in Section V.

B. Use of cyclic B-splines

In order to achieve a smoother transition between passes, the B-spline functions are arranged in cyclic manner. This kind of approach is shown in [15]. Cyclic arrangement means that the set of starting B-spline functions (three in case of 75% overlapping) share its weights with the appropriate set of ending B-splines, thus they are connected (Fig. 2b). This is illustrated in a graphical way in Fig. 3 In the approach [14] the starting functions are separated from the ending functions (Fig. 2a). This makes the control signal jagged at the transition between passes. Example of such behavior is shown in Section V.

C. Introduction of shift between error and control signal

This modification is reported to improve the steady-state performance and the speed of error convergence in classical ILC [5], [16]. It can be applied to BSN controller by rewriting the (8) in the following way

$$w_i(k) = (1-\lambda)w_i(k-1) - \gamma \sum_{p=0}^{N-1} \mu_i(p) e(p+\alpha,\ k-1), \tag{9}$$

where parameter α denotes the shift between the error and the control signal in terms of samples. The addition of this shift is required by the fact that most of plants is proper, so the error in one sample does not correspond with the control in it, but only with the control from the previous samples. This

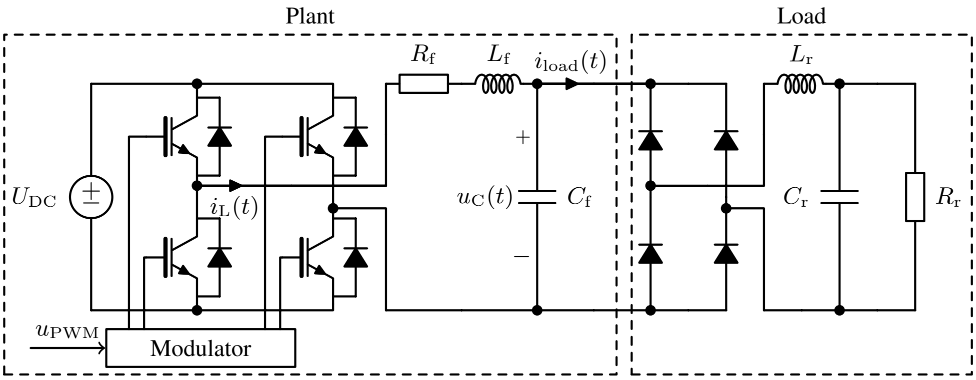


Fig. 4. Electrical scheme of the plant (CACF VSI) with an exemplary rectifier load

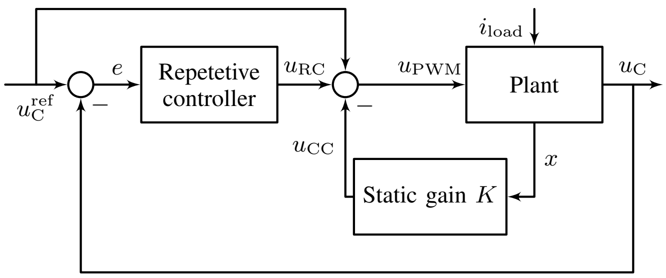


Fig. 5. Schematic diagram of the control system, where: u_CC - control signal form classic controller (full state feedback), u_RC - control signal from repetitive controller (BSN RC), $u_\mathrm{C}^\mathrm{ref}$ - reference signal

TABLE I
SYSTEM PARAMETERS

Parameter	Value	Description
R_f	$0.75\,\Omega$	Filter resistance
L_f	$0.48\,\mathrm{mH}$	Filter inductance
C_f	$100\,\mu\mathrm{F}$	Filter capacitance
R_r	$120\,\Omega$	Resistance of the rectifier load
L_r	$0.12\,\mathrm{mH}$	Inductance of the rectifier load
C_r	$4.3\,\mathrm{mF}$	Capacitance of the rectifier load
U_DC	$350\,\mathrm{V}$	Voltage in DC-link
T_s	$0.1\,\mathrm{ms}$	Sampling time
f_out	$50\,\mathrm{Hz}$	Output signal frequency
$U_\mathrm{out}^\mathrm{RMS}$	$230\,\mathrm{V}$	RMS value of output voltage

suggests that parameter α should be equal to or greater than one, which was confirmed in simulational and experimental studies. In fact, when the α is correctly chosen it speeds up the error convergence and in most cases allows to achieve a long term stability without using the forgetting factor.

Initially the selection of α was done by a guess and check method. However, an interesting observation was made. It appears that the best results are produced when α is equal to the sample number in which the impulse response of the plant reaches the maximum absolute value. Fig. 6 shows example in which control impulse has the biggest effect on the fourth output sample.

IV. Classic control

The classic controller is used also beside the repetitive controller. It consist of: full state feedback and feed-forward.

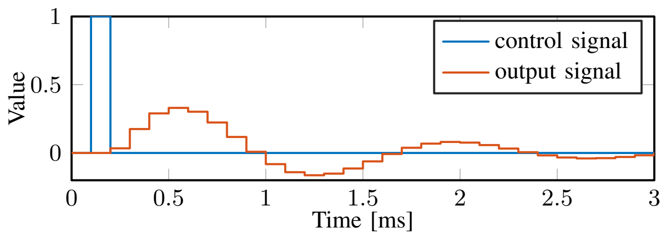


Fig. 6. Example impulse response of a plant

Full state feedback is used to increase the damping of a plant. Feed-forward is used to improve responsiveness of the system.

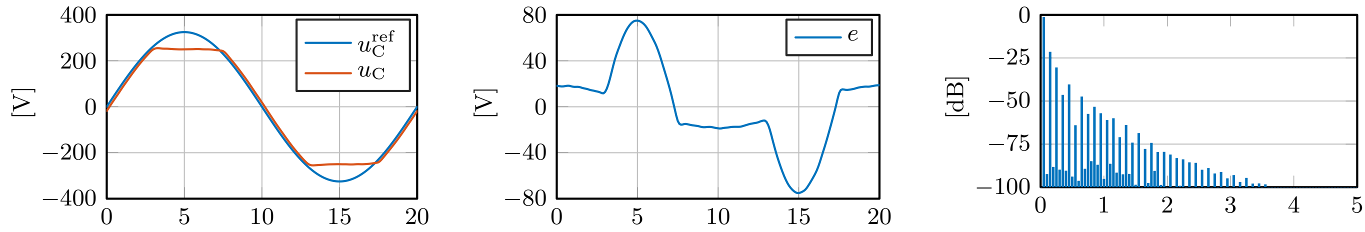
V. Results

The presented controller was tested on CACF VSI with an LC output filter. Exact control law is described by:

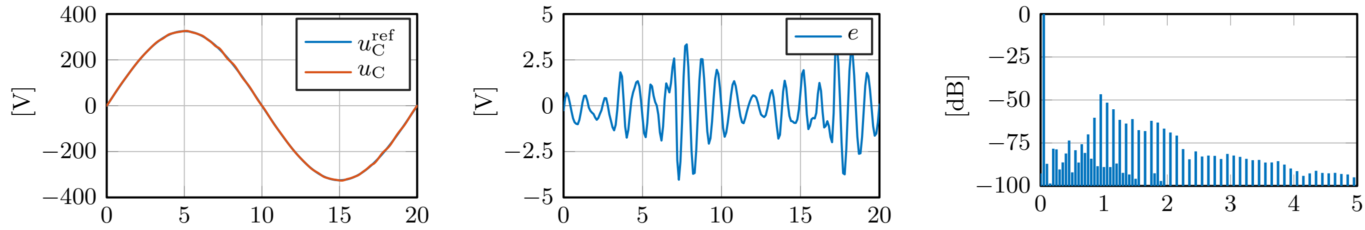
$$w_i(k) = w_i(k-1) - 0.2\sum_{p=0}^{199} \mu_i(p)e(p+4,\ k-1), \qquad (10)$$

Value of the learning factor was choosen by guess and check method. Electrical diagram of the plant is presented in Fig. 4. Full bridge diode rectifier with an capacitor filter is used as load. It is done so because this kind of load is non-linear and it places a heavy demand on the controller. Block diagram of the whole control system is presented in Fig. 5. Parameters of plant are shown in Table I. Tests were performed in simulations [17], [18] and on a real converter. Simulations were done using Matlab and Plecs [19] software. For the tests on the real converter the controller was implemented on TMS320F2812 micro-controller.

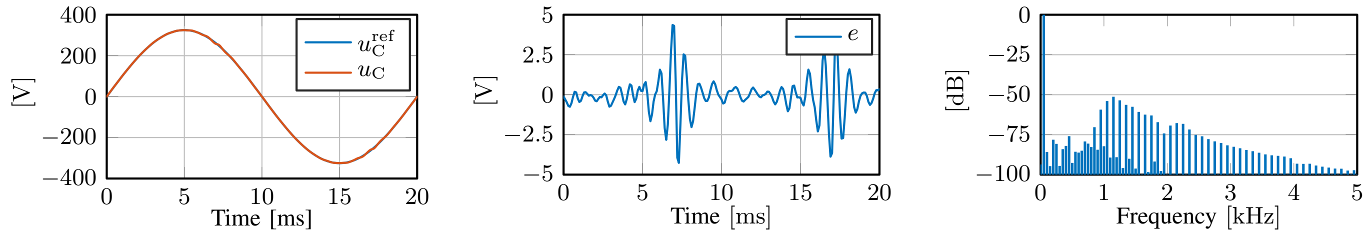
The first set of plots (Fig. 7) illustrates the harmonic analysis of output signal in three cases: without repetitive control (Fig. 7a), with BSN controller based on first order B-splines (Fig. 7b), with BSN controller based on second order B-splines (Fig. 7c). Usage of BSN improves THD (Total Harmonic



(a) Without repetitive controller, first harmonic amplitude is equal to 87.32% and THD = 10.37%



(b) With BSN controller based on the first order b-splines, first harmonic amplitude is equal to 100.00% and THD = 0.60%



(c) With BSN controller based on the second order b-splines, first harmonic amplitude is equal to 100.00% and THD = 0.49%

Fig. 7. CACF VSI output signal in steady state and its harmonic analysis

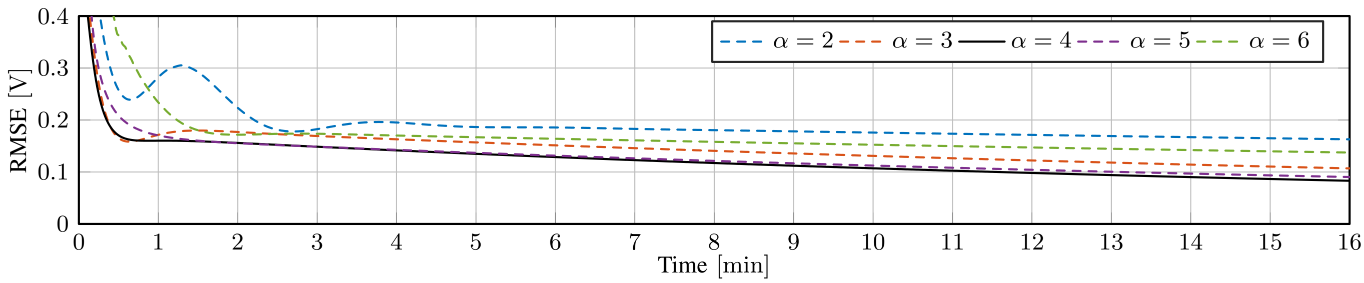


Fig. 8. Comparision of convergence of error signal for different shift beetween the error and the control signal (α) values. Best result is acheived for $\alpha = 4$.

Distortion) coefficient if the signal, and changing order of B-splines leads further improvement of THD.

Fig. 10 compares the control signal produced by the BSN controller with the non-cyclic B-splines against the control signal produced by the BSN with cyclic B-splines. In the case of the non-cyclic B-splines the control signal has discontinuity between passes (in tenth millisecond). This discontinuity causes oscillations in the output signal, which are presented in Fig. 11. They rises and without forgetting lead to instability of the system.

Fig. 8 presents convergence of the root mean square error of the output signal of a CACF VSI controlled by a modified BSN controller with a different α value. There is a correlation between the rate of convergence and the value of the impulse response of a system for different shift (α) between the error and the control signal values (compare Fig. 6 and Fig. 8).

Simulational results of a modified BSN controller are presented in Fig. 9.

Oscillations caused by BSN RC with no shift between the error and the control signal are presented in Fig. 13 and Fig. 14. The oscillations rises in time and eventually causes instability of the system.

Experimental results (done with $U_{\mathrm{DC}} = 350\,\mathrm{V}$ and $U_{\mathrm{out}}^{\mathrm{RMS}} = 70\,\mathrm{V}$) present waveforms of the CACF VSI output signal and the load current: without repetitive control (Fig. 12) and with classic ILC stabilized by introduction of forgetting factor

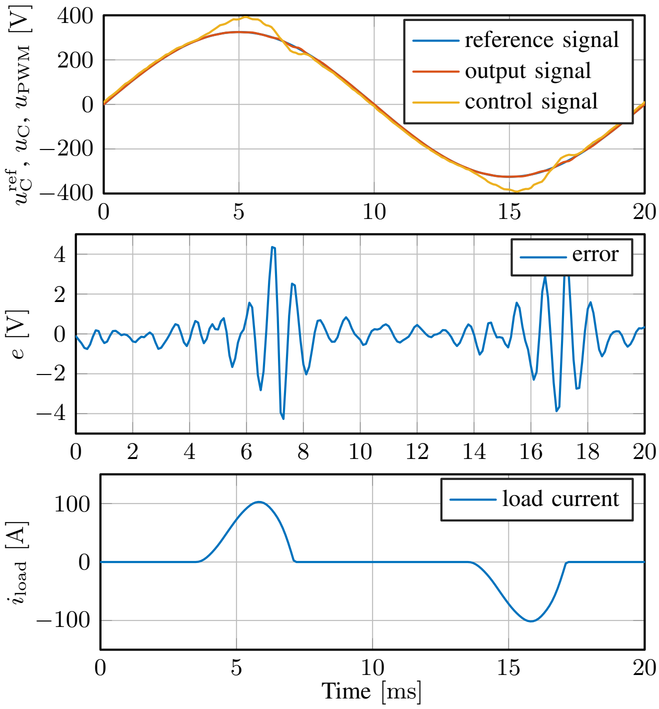
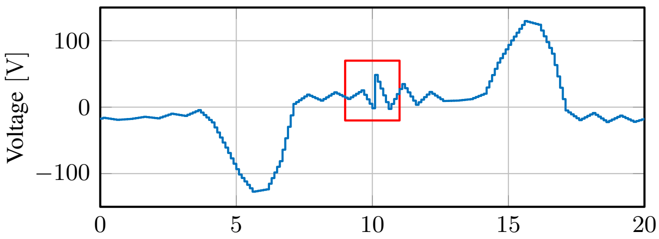
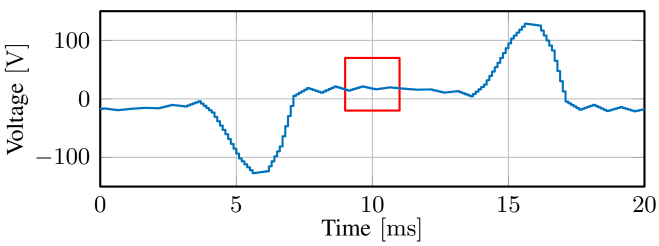


Fig. 9. Simulation results of modified BSN controller



(a) With using noncyclic b-splines



(b) With using cyclic b-splines

Fig. 10. Waveform of control voltage produced by BSN controller

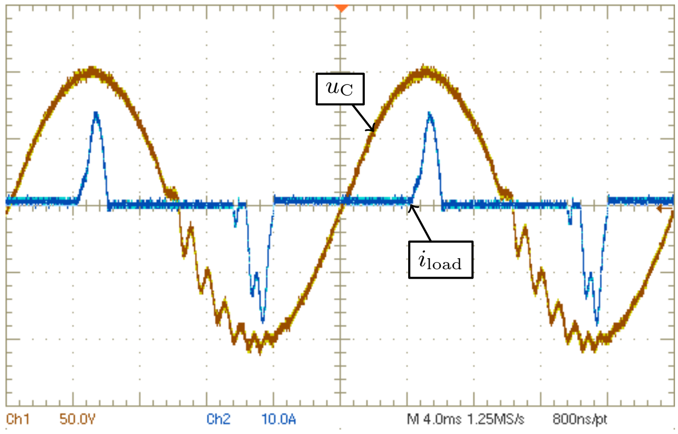


Fig. 11. Waveforms of the CACF VSI output signal (brown) and the load current (blue) controlled by RNC controller based on noncyclic B-splines

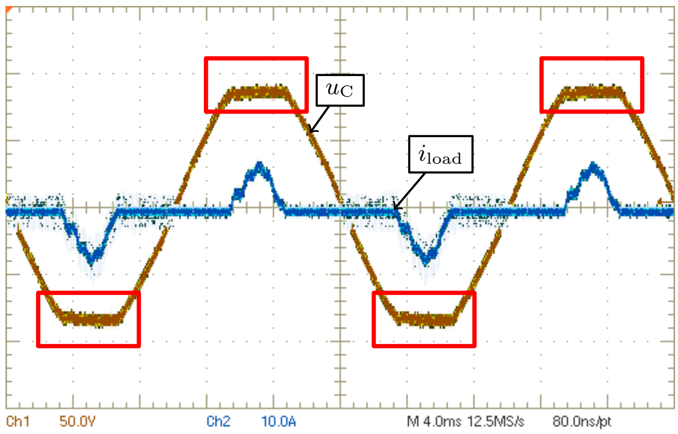


Fig. 12. Waveforms of the CACF VSI output voltage (brown) and the load current (blue) without the repetitive controller

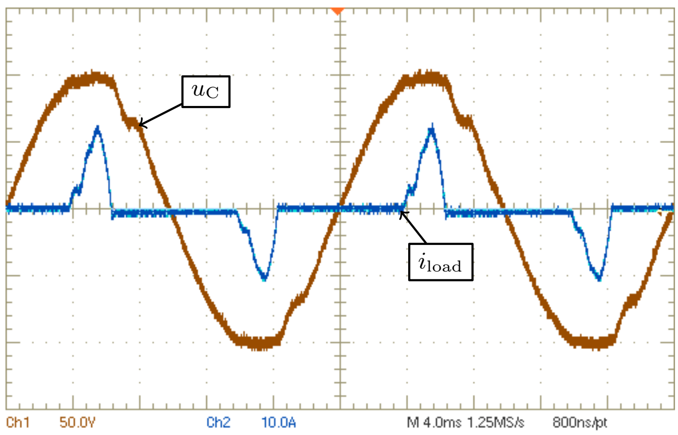


Fig. 13. Waveforms after approximately 3 seconds of work of the CACF VSI with the repetitive controller with no shift between the error and the control signal α

(Fig. 15). Usage of classic ILC improves output signal but because of forgetting factor the first harmonic has not full desired value.

Experimental results of the CACF VSI with modified BSN controller after approximately 20 seconds of work are presented in (Fig. 16). The performance is good however after 3 minutes of work there are visible small oscillations (Fig. 17). The oscillations do not increases in time even after 30 minutes of work (Fig. 18). The oscillations are probably caused by inappropriate frequency characteristic of B-splines and we are currently working on overcoming this problem by using another kind of basis functions.

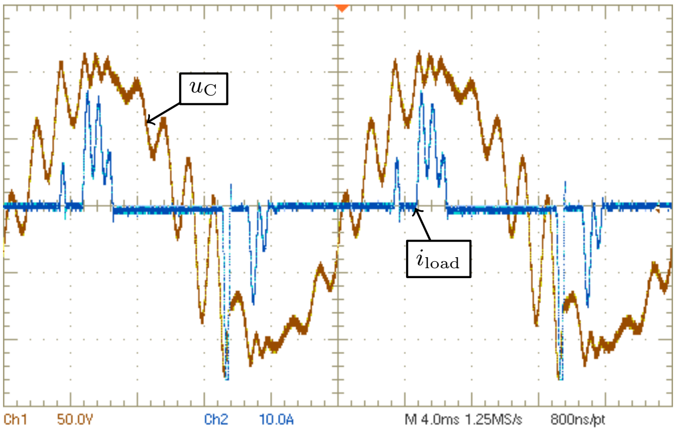


Fig. 14. Waveforms after approximately 20 seconds of work of the CACF VSI with the repetitive controller with no shift between the error and the control signal α

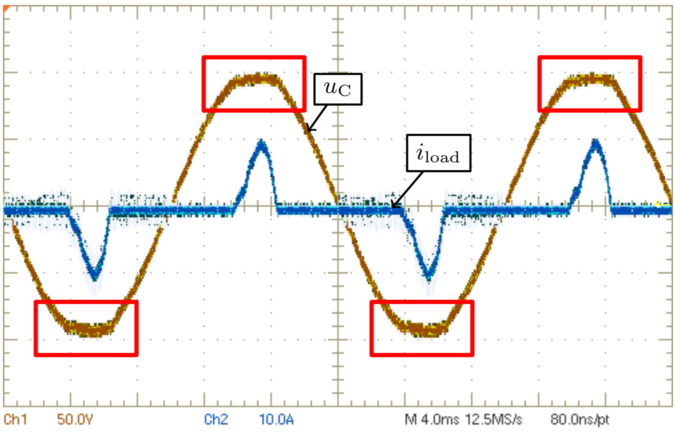


Fig. 15. Waveforms of the CACF VSI output voltage (brown) and the load current (blue) controlled by classical ILC with forgetting factor $= 0.008$

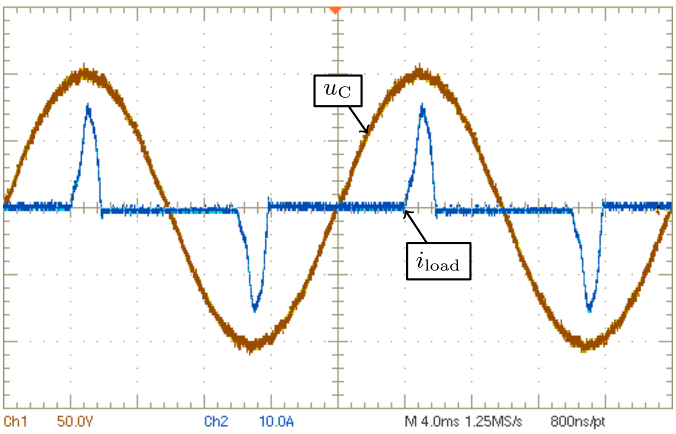


Fig. 16. Waveforms of the CACF VSI output voltage (brown) and the load current (blue) controlled by modified RNC controller after 20 seconds of work

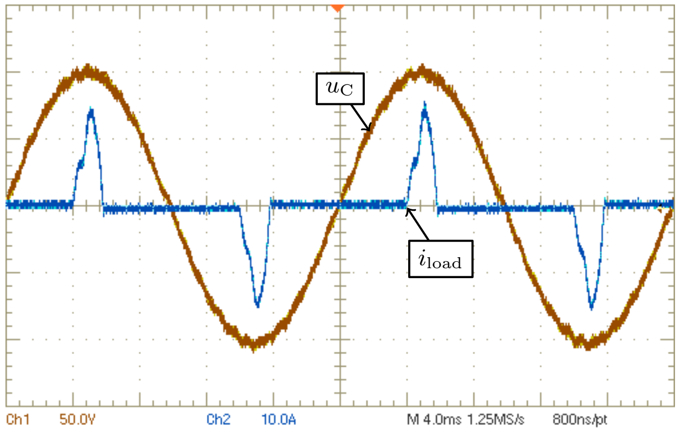


Fig. 17. Waveforms of the CACF VSI output voltage (brown) and the load current (blue) controlled by modified RNC controller after 3 minutes of work

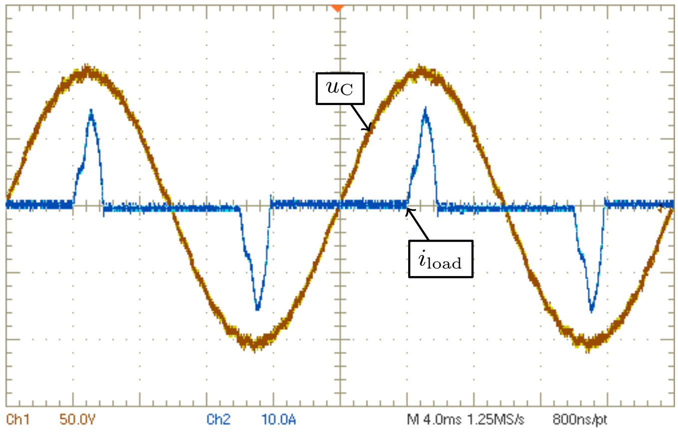


Fig. 18. Waveforms of the CACF VSI output voltage (brown) and the load current (blue) controlled by modified RNC controller after 30 minutes of work

VI. Conclusions

The modified B-spline based voltage controller for CACF VSI with an LC output filter has been proposed and tested. The controller consists of the full state feedback, the feed-forward and the BSN which is trained online to minimize the control error. The training algorithm was modified by changing the B-splines to the second order, arranging them in the cyclic manner and introducing a shift between the error and the control signal. The steady state and the transient performance of the proposed control technique has been tested by simulations and experiments. It has also been shown that the proposed modifications do not involve any additional computational effort.

Acknowledgment

This study was financed by the funds designated for the statutory activity of the Electrical Drive Division, Institute of Control and Industrial Electronics, Faculty of Electrical Engineering, Warsaw University of Technology, Warsaw, Poland.

References

[1] Z. Bien and J.-X. Xu, *Iterative learning control: analysis, design, integration and applications*. Kluwer Academic Publishers, 1998.

[2] A. Tayebi, S. Abdul, M. Zaremba, and Y. Ye, "Robust iterative learning control design: application to a robot manipulator," *Mechatronics, IEEE/ASME Transactions on*, vol. 13, no. 5, pp. 608–613, 2008.

[3] A. Tayebi, "Adaptive iterative learning control for robot manipulators," *Automatica*, vol. 40, no. 7, pp. 1195–1203, 2004.

[4] H. Deng, R. Oruganti, and D. Srinivasan, "Analysis and design of iterative learning control strategies for ups inverters," *Industrial Electronics, IEEE Transactions on*, vol. 54, no. 3, pp. 1739–1751, 2007.

[5] Y. Wang, F. Gao, and F. J. Doyle, "Survey on iterative learning control, repetitive control, and run-to-run control," *Journal of Process Control*, vol. 19, no. 10, pp. 1589–1600, 2009.

[6] K. L. Moore, "Iterative learning control: an expository overview," in *Applied and computational control, signals, and circuits*. Springer, 1999, pp. 151–214.

[7] Y. Shi, "Robustification in repetitive and iterative learning control," Ph.D. dissertation, Columbia Univeristy, 2013.

[8] H. Elci, R. W. Longman, M. Q. Phan, J.-N. Juang, and R. Ugoletti, "Simple learning control made practical by zero-phase filtering: Applications to robotics," *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, vol. 49, no. 6, pp. 753–767, 2002.

[9] B. Ufnalski and L. Grzesiak, "Artificial neural network based voltage controller for the single phase true sine wave inverter–a repetitive control approach," *Electrical Review (Przeglad Elektrotechniczny)*, vol. 89, no. 4, pp. 14–18, 2013.

[10] B. Ufnalski and L. M. Grzesiak, "A plug-in direct particle swarm repetitive controller for a single-phase inverter," *Electrical Review (Przeglad Elektrotechniczny)*, vol. 90, no. 6, pp. 6–11, 2014.

[11] Y. Q. Chen, K. L. Moore, and V. Bahl, "Learning feedforward control using a dilated b-spline network: Frequency domain analysis and design," *IEEE Transactions on Neural Networks*, vol. 15, no. 2, pp. 355–366, 2004.

[12] D. Yu, Y. Zhu, K. Yang, C. Hu, and M. Li, "Combined nonlinear feedback and cascaded iterative learning control for an ultra-precision dual-stage actuated wafer stage."

[13] P. Nagelmaeker, M. G. Wassink, and O. Bosgra, "Improving the drop consistency of an inkjet printhead by iterative learning control," *Eindhoven University of Technology Masters Thesis*, 2006.

[14] H. Deng, R. Oruganti, and D. Srinivasan, "Neural controller for ups inverters based on b-spline network," *Industrial Electronics, IEEE Transactions on*, vol. 55, no. 2, pp. 899–909, 2008.

[15] Z. Lin, J. Wang, and D. Howe, "A learning feed-forward current controller for linear reciprocating vapor compressors," *Industrial Electronics, IEEE Transactions on*, vol. 58, no. 8, pp. 3383–3390, 2011.

[16] R. W. Longman, "Iterative/repetitive learning control: learning from theory, simulations, and experiments," *Encyclopedia of the Sciences of Learning*, pp. 1652–1657, 2012.

[17] B. Ufnalski. (2015) B-spline based repetitive neurocontroller. [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/49023-b-spline-based-repetitive-neurocontroller

[18] M. Malkowski. (2015) B-spline network based repetitive controller (c-code). [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/49077-b-spline-network-based-repetitive-controller–c-code-

[19] Plexim. Plecs. [Online]. Available: http://www.plexim.com/