Bartlomiej UFNALSKI, Lech M. GRZESIAK
Warsaw University of Technology, Institute of Control and Industrial Electronics

A plug-in direct particle swarm repetitive controller for a single-phase inverter

Abstract. *The paper presents an online particle swarm optimizer (PSO) as an iterative learning controller for the single phase inverter with an output LC filter. The novelty of the solution lies in the fact that the swarm directly stores samples of the control signal. The swarm optimizes, according to a user-defined performance index, in online mode the control signal to reject the repetitive disturbance (the load current drawn, for example, by the diode rectifier). The concept of the direct swarm controller is investigated with the help of numerical simulations.*

Streszczenie. *W artykule opisano regulator rojowy realizujący sterowanie z uczeniem iteracyjnym dla jednofazowego falownika napięcia z wyjściowym filtrem LC. Oryginalność rozwiązania polega na fakcie bezpośredniego przechowywania próbek sygnału sterującego przez rój cząstek. Rój optymalizuje w trybie on-line sygnał sterujący eliminując wpływ okresowego zakłócenia (prądu obciążenia pobieranego, na przykład, przez prostownik diodowy z filtrem pojemnościowym) na jakość napięcia wyjściowego. Sygnał sterujący jest optymalny z uwagi na zdefiniowany przez użytkownika wskaźnik jakości. Koncepcja bezpośredniego regulatora rojowego została zbadana przy użyciu technik modelowania numerycznego.* ***(Sterowanie powtarzalne napięciem wyjściowym falownika przy użyciu bezpośredniego regulatora rojowego)***

Keywords: iterative learning control, particle swarm optimization, single-phase inverter, repetitive disturbance rejection
Słowa kluczowe: sterowanie powtarzalne z uczeniem, optymalizacja metodą roju cząstek, falownik jednofazowy, eliminacja wpływu okresowego zakłócenia

doi:10.12915/pe.2014.06.02

Introduction

Control schemes that explicitly take into account the repetitiveness of the reference signal and the disturbance are very promising solutions for constant-amplitude constant-frequency (CACF) inverters. Proposed solutions originate from iterative learning control (ILC), repetitive control (RC) and run-to-run (R2R) control techniques. All mentioned techniques can be analyzed in the uniform framework discussed in [1]. The main idea behind this is to analyze a process as two-dimensional. This provides the opportunity to design a separate controller that shapes the behavior along the pass (in the p-direction) and a second controller that influences pass to pass dynamics (in the k-direction). For a repetitive process a desired control signal is constant in shape at the steady state in pass-to-pass direction. This in turn implies that the very basic P-type control law

$$u\left(p,k\right)=u\left(p,k-1\right)+K_{\mathrm{RC}}e\left(p,k-1\right), \quad (1)$$

where u denotes the control signal, e is the control error, K_{RC} is the controller gain, k is the iteration (pass, trial, cycle) index and p is the time index along the pass ($1\leq p\leq\alpha$, where α is the pass length), theoretically is able to force the control error to converge toward zero for a very large class of systems. It should be noticed that (1) constitutes an integral action in the k-direction. The situation is then equivalent to a non-repetitive process with a constant reference and a constant disturbance. However, it is widely known that (1) is impractical because of long term stability issue and additional filtering is essential to stabilize the system [2, 3, 4, 5, 6, 7, 8]. It should be noted that the long term stability problems were identified numerically and experimentally. There is no clear theoretical insight into the roots of this phenomenon. A low pass filtering prevents the controller from learning at higher frequencies. The cut-off frequency is usually determined experimentally by inspecting the control error frequency spectrum and/or using trial and error method. Moreover, it often happens that the cut-off frequency has to be relatively small, e.g. in [2] ca. one fourth of the Nyquist frequency was necessary to stabilize the repetitive part of the controller. This significantly limits the performance of the controller. One should be also aware that due to lack of theoretical insight into the nature of this problem, it is not sure whether such an aggressive filtering action solves the problem or only shifts it in time expressed in number of passes. The problem could still hold due to lack of filters with ideal attenuation in the stop-band. This in turn implies that the learning for high frequencies is only slowed down instead of being stopped permanently. This could be not critical for an assembly line robotic arm that repeats the task at e.g. 0.1Hz and will make ca. 60 thousand passes during 7-day-long continuous operation. However, it is usually critical for power electronic converters that in the similar time horizon will make ca. 30 million passes. Practical control laws have then the form of

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

where **Q** and **L** are usually low-pass zero-phase-shift filters. In some designs **Q** is assumed to be a positive scalar smaller than 1 serving as a forgetting factor. The forgetting is then equally active for all frequencies. It has been shown in [9] that such schemes with the forgetting are equivalent to standard feedback control. On the other hand, aggressive filtering compromises performance and in the case of CACF sine wave inverters results in disturbance rejection similar to multi-oscillatory (also known as multiresonant) control schemes [10, 11]. Multi-oscillatory controllers do not suffer from long term stability problems and as such are one of the best alternatives for high-performance CACF inverters.

There have been successful attempts to accomplish the repetitive control task using universal function approximators, e.g. artificial neural networks (ANN). The goal for any repetitive controller is to construct in an iterative manner a desired control signal along the pass that fulfills assumed performance requirements. This task can be put into the frame of ANN online training. The examples of such solutions for CACF inverters are described in [12, 13]. These solutions impose usually significant computational burden and are very non-straightforward in tuning.

Our idea is to formulate the problem of finding a perfect shape of the control signal along the pass as an online iterative optimization problem. The behavior of a controller at the steady state (in the p-direction) would be then optimal according to a user-defined cost function. The main goal is to develop a control scheme that does not need low-pass filtering to circumvent long term stability issues. Some inspiration came from predictive control systems. In those systems behavior is shaped solely by the user-defined cost function (the

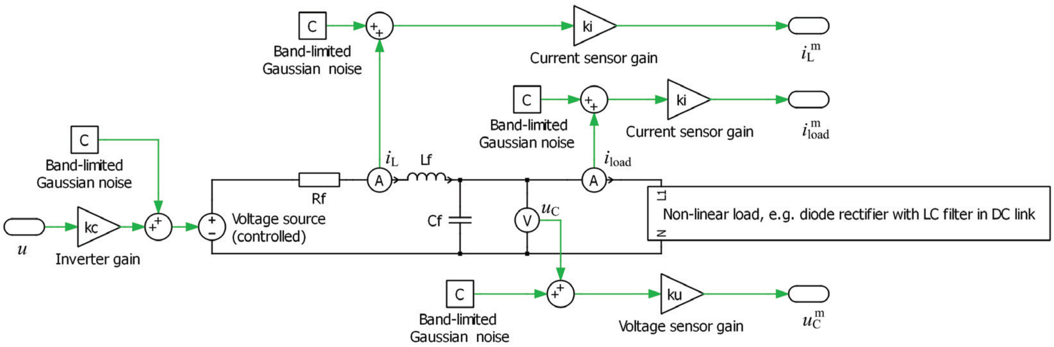

Fig. 1. Simplified model of the single-phase inverter

performance index). The predictive controller determines its next step by choosing it from all available moves tested using mathematical model of the plant and rated according to the user-defined performance index. The goal for the repetitive controller is to construct the optimal shape of the control signal along the pass. Due to the infinite number of possible shapes, this cannot be completed by testing all of them. An evolutionary search will be then employed for on-the-fly optimization. Our objective is also to develop the repetitive controller that does not require mathematical model of the plant (a model-free approach for black-boxed system) and is gradient-free (no derivative of cost function is calculated). To meet these objectives a particle swarm optimization technique has been employed here. There have been attempts reported in the literature to use online PSO for solving real-time control tasks [14]. Swarms have been used to optimize in online mode gains in controllers with predefined structure (e.g. of PID controller). The originality of the proposed solution lies in the fact that the swarm is itself a repetitive controller and directly stores all samples of the control signal along the pass. That is why the term "direct" has been used to name the solution. Moreover, the controller can be connected in parallel to an existing p-direction controller. This justifies the therm "plug-in" proposed as a part of the name. Also, from now on the acronym PDPSRC is used for short.

Model of the plant

The repetitive controller under consideration does not require modeling of the plant. However, the model of the plant has been used for testing purposes and for designing a feedback/feedforward controller that acts in the p-direction. The simplified model of the single-phase PWM inverter with the LC output filter is as follows

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}_\mathrm{f}(t) = \mathbf{A}_\mathrm{f}\mathbf{x}_\mathrm{f}(t) + \mathbf{B}_\mathrm{f}u(t) + \mathbf{E}_\mathrm{f}i_\mathrm{load}(t), \tag{3}$$

with

$$\mathbf{A}_\mathrm{f} = \begin{bmatrix} -\frac{R_\mathrm{f}}{L_\mathrm{f}} & -\frac{k_i}{k_u L_\mathrm{f}} \\ \frac{k_u}{k_i C_\mathrm{f}} & 0 \end{bmatrix} \tag{4}$$

$$\mathbf{B}_\mathrm{f} = \begin{bmatrix} \frac{k_\mathrm{c} k_i}{L_\mathrm{f}} \\ 0 \end{bmatrix} \tag{5}$$

$$\mathbf{E}_\mathrm{f} = \begin{bmatrix} 0 \\ -\frac{k_u}{C_\mathrm{f}} \end{bmatrix} \tag{6}$$

and

$$\mathbf{x}_\mathrm{f} = \begin{bmatrix} i_\mathrm{L}^\mathrm{m} \\ u_\mathrm{C}^\mathrm{m} \end{bmatrix}, \tag{7}$$

Table 1. Parameters of the converter

Parameter	Symbol	Value
Filter inductance	L_f	300μH
Filter capacitance	C_f	160μF
Filter resistance	R_f	0.2Ω
Filter resonant frequency	f_res	726Hz
Critical damping resistance	R_crit	2.74Ω
Reference frequency	f^ref	50Hz
Sampling/PWM frequency	f_s	10kHz
Pass length	α	200
DC-link voltage	k_c	450V
Measurement noise	–	ca. 1%
Voltage measurement gain	k_u	$1/325$ [1/V]
Current measurement gain	k_i	$1/200$ [1/A]
Rectifier power	–	ca. 6kW
Rectifier current crest factor	–	ca. 2.5
Resistive load power	–	ca. 4kW

where L_f, R_f, C_f denote inductance, resistance and capacitance of the output filter. The load current $i_\mathrm{load}(t)$ represents the disturbance. This description already accommodates k_u, k_i, k_c gains that model current and voltage transducers, and the voltage source inverter (VSI), respectively. The PWM VSI is approximated using a controlled voltage source. The superscript $\bullet^\mathrm{m}$ denotes the output signal of a measurement transducer. A schematic diagram of the simplified model is depicted in Fig. 1. Selected parameters of the converter are given in Table 1. Proposed swarm controller acts only in the k-direction. The considered plant is inherently stable. However, its natural damping is very low since resistance of the circuit is far below the critical one (see Table 1). The only damping comes from the stray resistance of the components. A separate controller acting in the p-direction should be implemented if high quality of the voltage waveform is needed also in transient states caused by load type and level variations.

Particle swarm optimizer for non-stationary environment

Particle swarm optimization is attributed to Kennedy, Eberhart and Shi [15]. It is an iterative gradient-free search algorithm inspired by biological swarming such as bird flocking, fish schooling, herding of land animals or collective behavior of insects. The search starts with a random set (here called swarm) of solutions (here called particles). The particles travel through a search space and are rated according to a user defined objective function $\mathcal{J}$. The basic speed update rule for a stationary environment is as follows

$$\begin{aligned} \mathbf{v}_j(i+1) = c_1\mathbf{v}_j(i) &+ c_2 r_\mathrm{p}\left(\mathbf{q}_j^\mathrm{pbest} - \mathbf{q}_j(i)\right) + \\ &+ c_3 r_\mathrm{g}\left(\mathbf{q}^\mathrm{gbest} - \mathbf{q}_j(i)\right), \end{aligned} \tag{8}$$

where j is the particle identification number, i denotes the swarm iteration number, $\mathbf{v}_j$ and $\mathbf{q}_j$ are speed and position of the j-th particle, $\mathbf{q}_j^\mathrm{pbest}$ stores the best solution proposed so

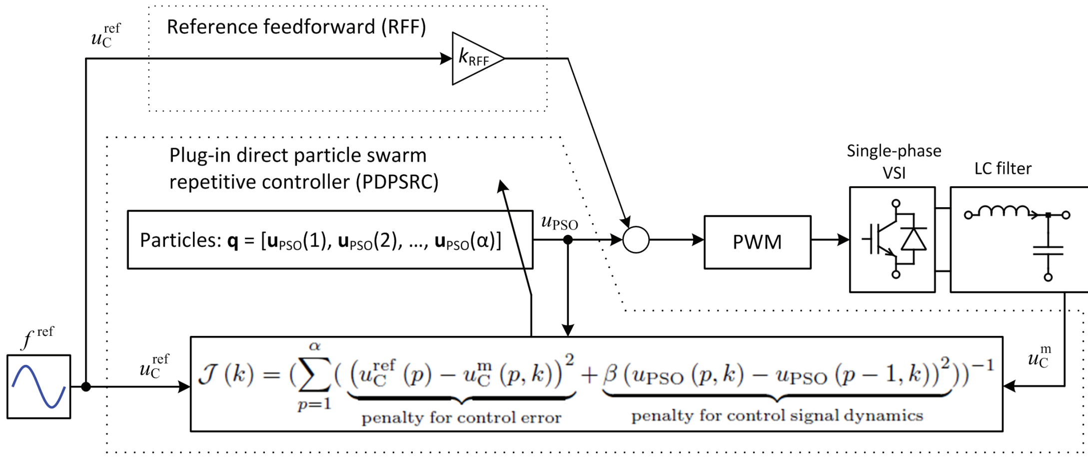

Fig. 2. Proposed direct swarm controller for the CACF inverter

far by the j-th particle, $\mathbf{q}^{\mathrm{gbest}}$ denotes the best solution found so far by the swarm, c_1, c_2 and c_3 are the explorative factor (inertia weight), the individuality factor and the social factor, respectively. The random numbers r_{p} and r_{g} are uniformly distributed in the unit interval. In all experiments described in this paper the c_1, c_2 and c_3 factors have been calculated using the constricted PSO formula [15] and are 0.73, $0.73 \cdot 2.05$ and $0.73 \cdot 2.05$, respectively. The swarm whose movements are governed by the rule (8) is not capable of tracking a moving optimum. Two modifications have been implemented to enable for optimization also in a non-stationary environment, which is the case here due to anticipated variations of load current.

The first modification introduces the evaporation constant ρ for the personal fitness value $P_j = \mathcal{J}\left(\mathbf{q}_j^{\mathrm{pbest}}\right)$ stored in each particle's memory. This forces particles to gradually forget previous best solution. The formula of forgetting is as follows [16]

(9)
$$\begin{bmatrix} P_j(i+1) \\ \mathbf{q}_j^{\mathrm{pbest}} \end{bmatrix} = \begin{cases} \begin{bmatrix} \rho P_j(i) \\ \mathbf{q}_j^{\mathrm{pbest}} \end{bmatrix} & \text{if } \eta_{j+1} \le \rho P_j(i) \\ \begin{bmatrix} \eta_{j+1} \\ \mathbf{q}_j(i+1) \end{bmatrix} & \text{if } \eta_{j+1} > \rho P_j(i), \end{cases}$$

where $\eta_{j+1} = \mathcal{J}\left(\mathbf{q}_j(i+1)\right)$ is the current fitness of the j-th particle and ρ has a positive value less than 1 for any positive-definite functional $\mathcal{J}$ and an optimization task formulated as the maximization one.

The second modification prevents the swarm from converging to zero speed. This is done by monitoring the diversity of the swarm and introducing the repelling mode into the speed update rule (8). In this study a commonly used swarm diversity measure has been employed [17]

$$D_{\mathrm{dist}} = \frac{1}{N_{\mathrm{p}}\sqrt{N_{\mathrm{d}}}} \sum_{j=1}^{N_{\mathrm{p}}} \sqrt{\sum_{n=1}^{N_{\mathrm{d}}} \left(q_{jn} - \overline{q}_n\right)^2}, \tag{10}$$

where N_{p} is the swarm size, N_{d} is the dimensionality of the problem and $\overline{q}$ is the average point. Originally this diversity measure uses the length of the longest diagonal in the search space instead of $\sqrt{N_{\mathrm{d}}}$. In the discussed solution the particle stores N_{d} samples of the normalized into $[-1, 1]$ control signal and often the positive and negative halves of the control signal are symmetric about time axis (a glide reflection). Therefore, it seems to be more intuitive to use radius $\sqrt{N_{\mathrm{d}}}$ of the search space as the normalizing coefficient in (10). If the diversity of the swarm drops below a given threshold, repulsion of particles is activated. In this study simple formula described in [17] has been adopted. The speed update rule (8) has been then modified as follows

$$\begin{aligned} \mathbf{v}_j(i+1) = c_1\mathbf{v}_j(i) \quad &+ \quad c_2 r_{\mathrm{p}} \delta \left(\mathbf{q}_j^{\mathrm{pbest}} - \mathbf{q}_j(i)\right) + \\ &+ \quad c_3 r_{\mathrm{g}} \delta \left(\mathbf{q}^{\mathrm{gbest}} - \mathbf{q}_j(i)\right) \end{aligned} \tag{11}$$

with

$$\delta = \begin{cases} 1 & \text{if } \delta < 0 \wedge D_{\mathrm{dist}} > D_{\mathrm{thold}} + \frac{h}{2} \\ -1 & \text{if } \delta > 0 \wedge D_{\mathrm{dist}} < D_{\mathrm{thold}} - \frac{h}{2}, \end{cases} \tag{12}$$

where D_{thold} is the diversity threshold and h is the hysteresis width, usually both problem specific.

Both modifications described in this section have been implemented in the developed swarm repetitive controller.

Swarm repetitive controller

The main idea behind the online direct optimization is to use the real plant itself as the critic without switching the system into some sort of offline mode. If population based optimization (e.g. PSO) is to be used, there is a need for performing fair comparison of solutions proposed by agents (patricles in PSO). As the performance index value $\mathcal{J}$ strongly depends on the reference signal and the disturbance, ideally both of them should be identical when rating each agent. Otherwise, no fair comparison would be possible. This condition is easily met in batch repetitive processes that can be, and usually are, reseted after each pass. However, this condition is also roughly met in continuous repetitive processes that cannot be reseted after each pass, as in the case of the discussed inverter. The main objective here is then to verify if it is feasible to organize the control task in the following manner:
a) apply control signal proposed by the agent,
b) calculate fitness value for the agent,
c) repeat a)-b) until all agents are rated (without resetting the physical process),
d) update agents,
e) repeat a)-d) until the system is in operation (without resetting the physical process).
In this study, the PSO has been chosen to perform d). Each particle is a vector of α consecutive samples of the control signal along the pass

$$\mathbf{q} = \left[\mathbf{u}_{\mathrm{PSO}}(1), \mathbf{u}_{\mathrm{PSO}}(2), \mathbf{u}_{\mathrm{PSO}}(3), \ldots, \mathbf{u}_{\mathrm{PSO}}(\alpha)\right]. \tag{13}$$

Table 2. Parameters of the swarm

Parameter	Symbol	Value
Dimensionality of the problem	N_{d}, α	200
Number of particles	N_{p}	25
Swarm update frequency	$\frac{f^{\mathrm{ref}}}{N_{\mathrm{p}}}$	2Hz
Evaporation constant	ρ	0.97
Diversity threshold	D_{thold}	$0.5 \cdot 325^{-1}$
Diversity hysteresis	h	0
Penalty factor	β	0.25

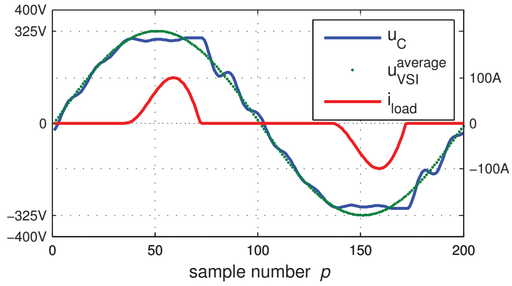

Fig. 3. Open-loop behavior of the plant (only the RFF is implemented)

Particles are rated according to the following performance index

$$\mathcal{J}(k) = \Big(\sum_{p=1}^{\alpha}\Big(\underbrace{\left(u_{\mathrm{C}}^{\mathrm{ref}}(p) - u_{\mathrm{C}}^{\mathrm{m}}(p,k)\right)^2}_{\text{penalty for control error}} +$$

$$+ \underbrace{\beta\left(u_{\mathrm{PSO}}(p,k) - u_{\mathrm{PSO}}(p-1,k)\right)^2}_{\text{penalty for control signal dynamics}}\Big)\Big)^{-1}, \tag{14}$$

where β is the subjective penalty factor. The reciprocal in (14) is introduced just to turn the optimization problem into the maximization one. The term with β that penalizes for control signal dynamics has been introduced to make sure that no overfitting will occur. Lack of such a penalty in the classic ILC scheme allows for oscillation build-up. The classical ILC (1) tends to over fit the control signal, i.e. it tries to find the perfect control signal even if it does not exist due to physical limitations of the plant. If (14) with appropriate $\beta > 0$ is used to rate the solutions, a compromise is worked out by particles between perfect disturbance rejection and control signal dynamics. Parameters of the swarm controller are given in Table 2 and its schematic diagram is provided in Fig. 2. The reference feedforward (RFF) has been kept from the standard open-loop control topology. The RFF introduces unity gain for zero frequency. It should be noted that the RFF is not essential for correct operation of the system. The part of the control signal contributed by the RFF has constant shape and could be incorporated into the swarm as the initial mean position of the particles. However, here the RFF is kept to make this initialization more straightforward, i.e. all control signal values stored in the particles are initialized near zero value.

Numerical experiment results

The dynamical behavior of the plant before plugging in the swarm controller is illustrated in Fig. 3. All tests presented here involve a load current imposed using the controlled current source. This makes test conditions even more challenging, e.g. a peak value of the load current does not decreases due to distorted voltage in transient states. Next the swarm repetitive controller has been plugged in and tested without any assisting controller in the p-direction. Such test conditions are demanding due to a very low natural damping in the discussed plant. The test scenario is as follows:

a) the swarm is initialized with near zero control signal,

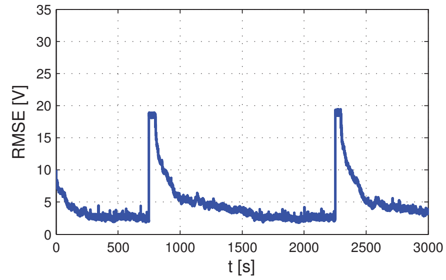

Fig. 4. Evolution of the root mean square error (the RFF and the PDPSRC are implemented)

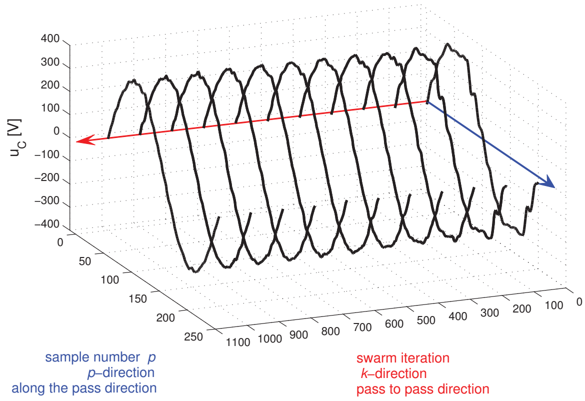

Fig. 5. Evolution of the output voltage after connecting the diode rectifier (the RFF and the PDPSRC are implemented)

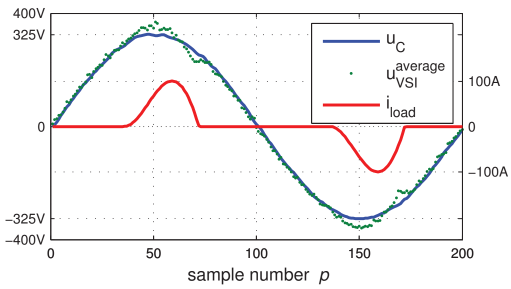

Fig. 6. Quality of the output voltage at test diode rectifier load (the RFF and the PDPSRC are implemented)

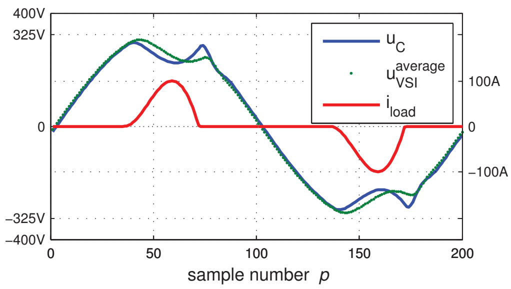

Fig. 7. Closed-loop behavior of the plant (only the RFF and the FSF are implemented)

b) the resistive load (see Table 1) is applied for 750s,

c) the resistive load is switched off and the diode rectifier (see Table 1) is switched on for 1500s,

d) the diode rectifier is switched off and the initial resistive load is applied once again.

The evolution of the root mean square error (RMSE) over the pass is demonstrated in Fig. 4. The RMSE is calculated for u_{C} instead of $u_{\mathrm{C}}^{\mathrm{m}}$ (see Fig. 1). Consequently, its value applies to the voltage waveform seen by the load. The evolution of the output voltage after switching on the nonlinear load

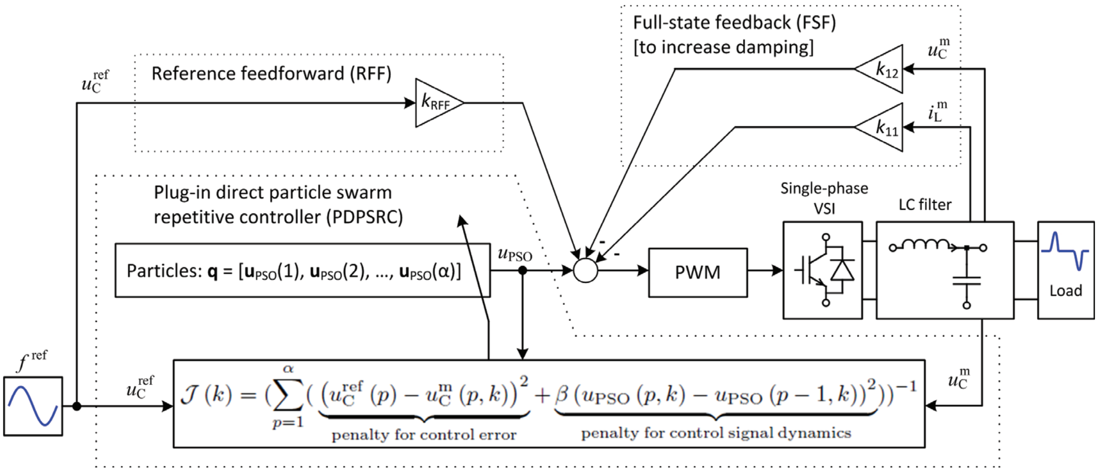

Fig. 8. Proposed direct swarm controller plugged into the system with the FSF

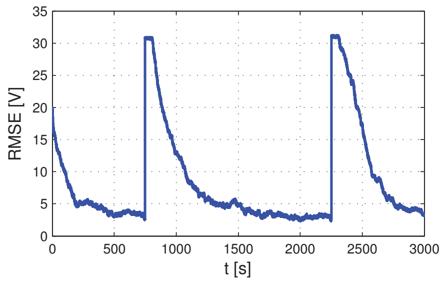

Fig. 9. Evolution of the root mean square error — the PDPSRC is plugged into the system with the FSF

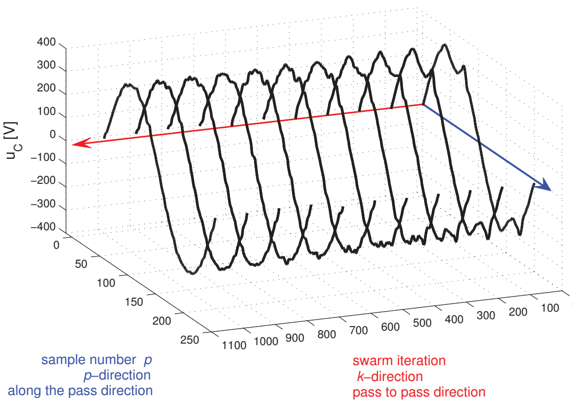

Fig. 10. Evolution of the output voltage after connecting the diode rectifier — the PDPSRC is plugged into the system with the FSF

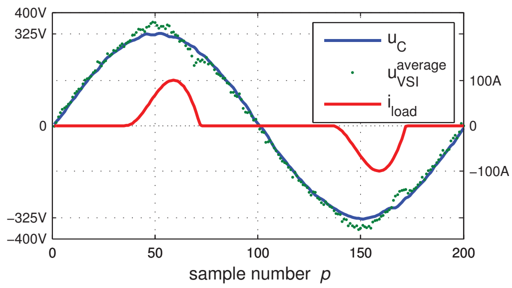

Fig. 11. Quality of the output voltage at test diode rectifier load (at the steady state of the PDPSRC)

is illustrated in Fig. 5. Quality of the voltage for the swarm near its equilibrium is depicted in Fig. 6. This illustrates the swarm's ability to cope with a significantly underdamped system. Since the PDPSRC acts only in the k-direction, which is typical for any plug-in type repetitive controller, transients

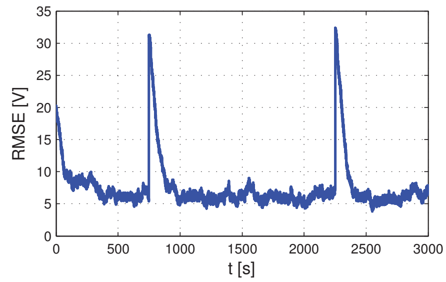

Fig. 12. Evolution of the RMSE for evaporation $\rho = 0.85$ and $D_{\text{thold}} = 1.5 \cdot 325^{-1}$ — better responsiveness at the cost of worse exploitation — to be compared to Fig. 9

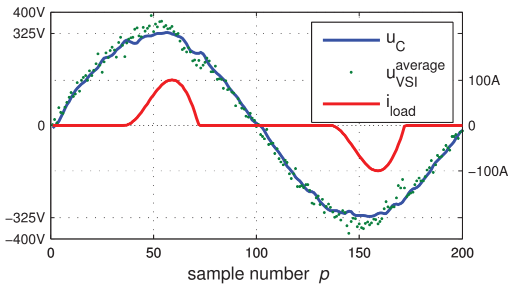

Fig. 13. Low quality of the output voltage due to faster swarm knowledge evaporation and higher diversity threshold

in the p-direction have to be shaped by a separate controller assumed as being out of the scope of this paper. However, it will be demonstrated that an additional controller acting along the pass can appear to be beneficial also for the performance of the PDPSRC. It has been observed that the swarm is able to find better solution in terms of the RMSE ripples near the equilibrium if the damping is increased. To illustrate this, the full-state feedback (FSF) is added and poles of the plant seen by the PDPSRC are shifted to the left. All design relevant formulas are given, e.g., in [18]. In this particular experiment the closed-loop conjugate poles are designed to increase damping 5 times. The dynamics of the closed-loop system in the p-direction is shown in Fig. 7. This of course deteriorates response of the system along the pass in terms of the voltage drop (in comparison to the situation from Fig. 3). The increased damping comes from the increased virtual resistance of the closed-loop system. The swarm controller plugged into such a system is depicted in Fig. 8. The better effectiveness of the control strategy in terms of the RMSE is presented in Fig. 9. It is noticeable that the consecutive passes are more repeatable from its RMSE point of view — the ripples are much lower than the ones in Fig. 4. The transient behavior of the system after connecting the nonlinear load is shown in Fig. 10. The quality of the voltage at the steady state is given in Fig. 11. Unsurprisingly, the online optimization problem in the k-direction is better conditioned if controllers are shaping the control signal in the p-direction in such a way that the landscape of the functional (14) is less rugged surface in $\mathbf{u}_{\mathrm{PSO}[1\times\alpha]}$ space.

Stability, exploitation and responsiveness of the swarm

No stability proof for the proposed control system has been elaborated yet. Taking into account the no free lunch theorem for optimization [19] one can conclude that such a proof, if ever derived, would be always objective function specific. By the objective function it is meant here (14) along with the plant and the disturbance which are essential to determine the cost. At the moment, numerical simulations are the only way to verify the stability. Nevertheless, such an approach to stability verification is quite common in online optimization problems, e.g. most predictive control schemes elaborated for power electronic converters are not accompanied by any relevant proof of stability. There is a belief among control practitioners that performance index online optimization based control methods are strong candidates for future control schemes [20].

In the case of the discussed controller the evaporation constant ρ and the diversity threshold D_{thold} are main parameters that are used to tailor the responsiveness of the swarm to load current shape variations. For comparison purposes the dynamics of the system with faster knowledge evaporation and higher diversity is illustrated in Fig. 12. This clearly sacrifices quality of the output voltage (Fig. 13). It is not uncommon for stand alone power generation units to operate under a moderate load variations for hours. A UPS for an office building can serve as the example. In such an application the fine voltage shaping that requires several minutes to reach new optimum seems to be acceptable.

Conclusions

The novel swarm based repetitive controller has been developed and tested numerically. Extensive tests indicate that the proposed algorithm does not suffer from long term stability issues — the common phenomenon encountered in the classic iterative learning controllers. An online optimization concept has been employed to iteratively shape the control signal along the pass. The particle swarm optimizer modified for non-stationary environments has been used. The cost function explicitly takes into account the dynamics of the control signal. This intrinsically prevents control signal build-up in the presence of a repetitive disturbance. No low-pass filtering is needed to robustify the control scheme. Thus, the available controller bandwidth can be fully exploited. Only minor modifications, from computation burden point of view, to the standard PSO have been identified as necessary for tracking the moving optimum. It has been illustrated that the PSO technique can be used in online mode to directly shape the control signal for the repetitive process. Obtained results indicate that the computational intelligence techniques of gradientless population-based optimization can be helpful in overcoming limiting issues of the traditional ILC schemes. Experimental verification is planned.

The research has been partially financed from the statutory funds of the Electrical Drive Division for 2013.

REFERENCES

[1] Wang Y., Gao F., Doyle III F.J., Survey on iterative learning control, repetitive control, and run-to-run control, *Journal of Process Control*, 19(10), pp. 1589–1600, Dec. 2009
[2] Ufnalski B., Grzesiak L.M., Galkowski K., Particle swarm optimization of an iterative learning controller for the single-phase inverter with sinusoidal output voltage waveform, *Bulletin of the Polish Academy of Sciences – Technical Sciences*, 61(3), 2013
[3] Elci H., Longman R.W., Phan M.Q., Je-Nan Juang, Ugoletti R., Simple learning control made practical by zero-phase filtering: applications to robotics, *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 49(6), pp. 753–767, Jun. 2002
[4] Heng Deng, Oruganti R., Srinivasan D., Analysis and design of iterative learning control strategies for UPS inverters, *IEEE Transactions on Industrial Electronics*, 54(3), pp. 1739–1751, Jun. 2007
[5] Cai Z., *Iterative learning control: algorithm development and experimental benchmarking*, Ph.D. thesis, University of Southampton, UK, 2009
[6] Shi Y., *Robustification in repetitive and iterative learning control*, Ph.D. thesis, Columbia University, USA, 2013
[7] Longman R.W., On the interaction between theory, experiments, and simulation in developing practical learning control algorithms, *International Journal of Applied Mathematics and Computer Science*, 13(1), pp. 101–111, 2003
[8] Longman R.W., Iterative/repetitive learning control: learning from theory, simulations, and experiments, *Encyclopedia of the Sciences of Learning*, pp. 1652–1657, 2012
[9] Verwoerd M.H.A., *Iterative learning control – a critical review*, Ph.D. thesis, University of Twente, The Netherlands, 2005
[10] Escobar G., Valdez A.A., Leyva-Ramos J., Mattavelli P., Repetitive-based controller for a UPS inverter to compensate unbalance and harmonic distortion, *IEEE Transactions on Industrial Electronics*, 54(1), pp. 504–510, Feb. 2007
[11] Kaszewski A., Ufnalski B., Grzesiak L.M., An LQ controller with disturbance feedforward for the 3-phase 4-leg true sine wave inverter, *IEEE International Conference on Industrial Technology (ICIT)*, pp. 1924–1930, Feb. 2013
[12] Deng H., Oruganti R., Srinivasan D., Neural controller for UPS inverters based on B-spline network, *IEEE Transactions on Industrial Electronics*, 55(2), pp. 899–909, Feb. 2008
[13] Ufnalski B., Grzesiak L., Artificial neural network based voltage controller for the single phase true sine wave inverter – a repetitive control approach, *Electrical Review (Przeglad Elektrotechniczny)*, open access at `pe.org.pl`, 89(4), pp. 14–18, Apr. 2013
[14] Wai R.-J., Lee J.-D., Chuang K.-L., Real-time PID control strategy for maglev transportation system via particle swarm optimization, *IEEE Transactions on Industrial Electronics*, 58(2), pp. 629–646, Feb. 2011
[15] Eberhart R.C., Shi Y., Kennedy J., *Swarm Intelligence*, 1st Edition, The Morgan Kaufmann Series in Evolutionary Computation, Morgan Kaufmann Publishers, 2001
[16] Cui X., Charles J.St., Potok T.E., A simple distributed particle swarm optimization for dynamic and noisy environments, *Studies in Computational Intelligence: Nature Inspired Cooperative Strategies for Optimization (NICSO 2008)*, 236, pp. 89–102, 2009
[17] Riget J., Vesterstrøm J.S., *A diversity-guided particle swarm optimizer - the ARPSO*, EVALife Technical Report no. 2002-02, Aarhus Universitet, Denmark, 2002
[18] Franklin G., Powell D., Workman M., *Digital control of dynamic systems*, 3rd Edition, Prentice Hall, Dec. 1997
[19] Wolpert D., Macready W., No free lunch theorems for optimization, *IEEE Transactions on Evolutionary Computation*, 1(1), pp. 67–82, 1997
[20] Kennel R., *Predictive control - the powerful method to control power converters and drives in the future*, Distinguish Lecturer IEEE Power Electronics Society Seminar, Warsaw, Poland, 22.05.2012

***Authors**: Ph.D. Bartlomiej Ufnalski, Prof. Lech M. Grzesiak, Institute of Control and Industrial Electronics, Faculty of Electrical Engineering, Warsaw University of Technology, ul. Koszykowa 75, 00-662 Warszawa, Poland, email: bartlomiej.ufnalski@ee.pw.edu.pl*