Plug-in Direct Multi-Swarm Repetitive Controller for the Sine Wave Inverter – on Keeping Particles Diversified in a Dynamic and Noisy Environment

Bartlomiej Ufnalski
Institute of Control and Industrial Electronics
Warsaw University of Technology, Poland
URL: http://ufnalski.edu.pl/

Lech M. Grzesiak
Institute of Control and Industrial Electronics
Warsaw University of Technology, Poland
URL: http://www.ee.pw.edu.pl/

***Abstract*—The paper discusses selected modifications to a constricted, i.e. canonical, version of the particle swarm optimizer (PSO) enabling effective operation of a repetitive controller. The recently developed plug-in direct multi-swarm repetitive controller (PDMSRC) for the sine wave constant-amplitude constant-frequency (CACF) voltage-source inverter (VSI) with an output LC filter serves as a case study. Online dynamic optimization problems (DOPs), i.e. those solved without switching into some sort of offline/commissioning mode, impose restrictions that are challenging for population-based optimization techniques. In the PDMSRC samples of the control signal are directly stored in the particles. Solutions available for swarms operating in dynamic and noisy environments are briefly discussed and candidates for the PDMSRC are indicated. Geometric versus statistical and cumulative versus dimension-wise diversity measures are tested. The advantages of replacing the frequently used non-dimension-wise repulsion with a dimension-wise mechanism are demonstrated here. The effectiveness of the proposed approach is illustrated with the help of numerical experiments.**

***Index Terms*—Repetitive control, iterative learning control, particle swarm optimization, dynamic optimization problem, diversity measures, pure sine wave inverter, repetitive disturbance rejection.**

I. Introduction

In many industrial systems the repetitiveness of a control signal is clearly apparent. A family of power electronic converters designed to produce CACF sinusoidal voltage may serve as an example. Control schemes that harness the repetitiveness of the reference signal and the disturbance are then potentially promising solutions for such inverters. The task of maintaining a high quality output voltage despite nonlinear load is often tackled using the internal model principle. Probably the most popular approach introduces an internal model of anticipated dominant disturbance frequencies and is known as the multiresonant or multioscillatory controller [1–3]. The main difficulty with that scheme comes from the problematic implementation of oscillatory terms near the Nyquist limit, therefore the resulting controller usually has bandwidth lower by one order of magnitude than the sampling employed. Another approach is to introduce a universal internal model of any periodic signal

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

where u denotes the control signal, e is the control error, k_{RC} is the controller gain, k is the iteration (pass, trial, cycle, period) index and p is the time index along the pass ($1 \leq p \leq \alpha$, where α is the pass length). The control signal is shaped in the iterative manner from pass to pass, i.e. information on control errors from the previous pass is used to correct control signal in the current pass [4, 5]. However, most of classic iterative learning controllers (ILC) suffer from long term stability problems and additional filtering is essential to stabilize the system [6–12]. For example, the very basic P-type control law (1) has to be modified into

$$u(p,k) = \mathbf{Q}(z)\, u(p,k-1) + k_{\mathrm{RC}} \mathbf{L}(z)\, e(p,k-1) \ , \qquad (2)$$

where $\mathbf{Q}$ and $\mathbf{L}$ are usually non-causal low-pass zero-phase-shift filters. The formula (2) represents the uniformed framework for ILC and repetitive control (RC) [13]. In some designs $\mathbf{Q}$ 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 [14] that such forgetting schemes are equivalent to standard feedback control. The low-pass narrow-bandwidth filtering needed to stabilize the system compromises performance and may result in a disturbance rejection capability similar to the multioscillatory control schemes [15].

Low-pass filtering in the classic ILC prevents problematic learning for high frequencies and as such becomes similar, as far as periodic disturbance rejection is considered, to the lack of oscillatory terms in the upper band. The idea behind (2) is to cut off learning for prohibited frequencies. A prohibited harmonic in this context means any disturbance harmonic that cannot be completely rejected due to the limited DC-link voltage present in any practical VSI. Such a harmonic would have a destabilizing impact in the long run due to the integration in the k-direction present in (1). This problem has been already widely acknowledged by practitioners [6–12]. The motivation

This is the author's version of the article. Please consult the proceedings of CPE-POWERENG 2016 on IEEE Xplore for citation. The relevant presentation is available at www.ufnalski.edu.pl/proceedings/cpe2016/. More stuff can be found at www.mathworks.com/matlabcentral/profile/authors/2128309-bartlomiej-ufnalski.

is hence to develop iterative learning control schemes that natively have mechanisms preventing overlearning.

The resources offered by off-the-shelf microcontrollers encourage automation developers to formulate control tasks as DOP ones and even to solve them using evolutionary dynamic optimization (EDO) techniques [16]. The feasibility of such solutions in the case of repetitive deterministic gradient-based neurocontrollers for CACF VSIs has already been demonstrated in [17–20] and the relevant models have been shared under the BSD license [21, 22]. It has also been already shown that the repetitive control task can be solved in the iterative manner using a population-based stochastic gradient-free optimization approach to directly shape the control signal. The resulting plug-in direct particle swarm repetitive controller (PDPSRC) and its multi-swarm modification (PDMSRC) have been demonstrated in [23–25] and the relevant model has been shared under the BSD license [26]. The controller does not require any additional low-pass filtering and therefore all available bandwidth can be potentially exploited. The basic PDPSRC reported in [23] uses a cumulative diversity measure to keep the swarm alive, i.e. to prevent the particles from lumping into a single mass. In this paper an improvement in terms of the rate of convergence (the responsiveness) as well as the computational burden achieved by using a dimension-wise diversity measure is discussed.

The rest of the paper is organized as follows. Section II gives some details on the PDMSRC. Section III describes different diversity measures. Section IV briefly presents an accompanying along-the-pass controller and a repetitive disturbance feedforward path. Section V numerically compares the behaviour of the swarms governed using selected diversity measures. The paper is concluded in Section VI.

II. Plug-in direct particle swarm repetitive controller

Particle swarms have already gained broad acceptance in offline optimization tasks. For an introduction to PSO see e.g. [27]. The algorithm has been originally proposed for solving unimodal problems in stationary environments. Nevertheless, after applying some modifications, the technique of particle swarms can be employed to solve problems in dynamic environments. Further modifications allow also for simultaneous finding and tracking of more than one optimal solution (multimodal optimization tasks). For the representative set of swarm movement laws effective in the dynamic environments see [28] or [29].

In the developed controller the swarm performs an online optimization of the control signal shape according to the following performance index

$$\mathcal{J}(k,n) = \mathcal{J}_0 + \underbrace{\sum_{p=\alpha_{n-1}+1}^{\alpha_n} \left(u_{\mathrm{C}}^{\mathrm{ref}}(p) - u_{\mathrm{C}}^{\mathrm{m}}(p,k)\right)^2}_{\text{penalty for control error}} + \underbrace{\beta \sum_{p=\alpha_{n-1}+2}^{\alpha_n} \left(u_{\mathrm{PSO}}(p,k) - u_{\mathrm{PSO}}(p-1,k)\right)^2}_{\text{penalty for control signal dynamics}}, \tag{3}$$

where: k is again the reference signal pass index, p is again the sample index reset at each pass beginning, $n \in [1, N]$ denotes the subswarm identification index, $\alpha_n \in \{\alpha_1, \alpha_2, \ldots, \alpha_N\}$ defines junctions between subswarms and β is the subjective penalty factor. The term with β that penalizes for control signal dynamics has been introduced in order to prevent overlearning. The lack of such a penalty in the classic ILC scheme allows for oscillation build-up. The penalty factor β can also be interpreted as a smoothing factor for control signal. The bigger the positive value of β, the more the landscape of (3) favours smoothly shaped control signals. The superscript $\bullet^{\mathrm{m}}$ denotes measurement signal corrupted by the noise and $\alpha_N = \alpha$ is again the pass length equal to the single period of the reference voltage $u_{\mathrm{C}}^{\mathrm{ref}}$. More details on the multiswarm repetitive controller are provided in [24].

The challenging idea incorporated here is to use the physical plant itself as a critic and to perform optimization during the regular operation of the system without the need for implementing any mathematical model of the system. Such a solution is innately robust against identification errors. On the other hand, any evaluation of a solution proposed by a particle necessitates applying this solution to the physical plant. It is then essential to make the swarm behave in such a way that each particle produces acceptable output voltage waveform at any iteration of the swarm. The developed controller consists of the particles that directly store samples of the control signal. The name *direct* comes from the fact that samples of the control signal are directly stored in particles – the PSO does not serve as a tuning/adaptation algorithm for some other controller. After passing all α consecutive control signal values to the PWM (pulse width modulator), which enables to rate one particle from each subswarm, the swarm is updated asynchronously [25] and the process is repeated ceaselessly.

The optimization task considered here is assumed to be of the unimodal type. The optimization landscape is dynamic due to the varying load conditions. There are two main issues that have to be addressed in dynamic environments: outdated memory and diversity loss. A lack of adequate mechanisms to manage them would result in an inability to find and then track a moving optimum. In some very specific dynamic optimization problems (DOPs) it could be sufficient to implement only one mechanism. For example, if a high level of diversity is kept, PSO should be able to find a new optimum located within the area populated by the particles. However, if an abrupt change occurs, there is a good chance that the moving

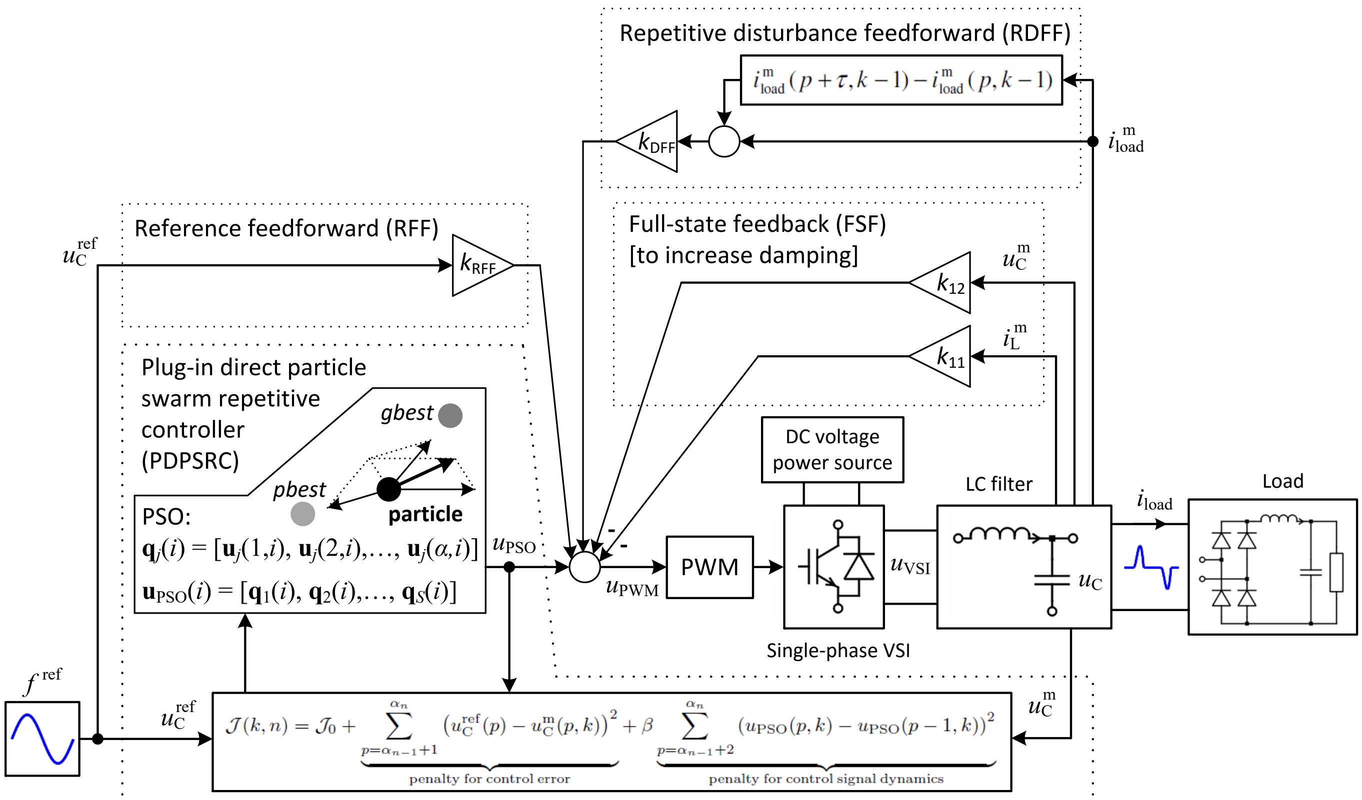


Fig. 1. A schematic diagram of the control system — the exemplary nonlinear load depicted for clarity.

optimum will outrun the swarm. Moreover, if a new optimum in the minimization problem has a higher value of $\mathcal{J}$, it is very likely that the swarm will be misled by the outdated attractors and will remain stuck in the outdated optimum. This brings on the need to deal with both aforementioned issues concurrently in most DOPs.

There are several techniques preventing diversity loss: randomization, repulsion, dynamic networks and multi-populations [29]. Whereas the problem of outdated memory is usually tackled with the help of re-evaluation or forgetting [29]. Some of these algorithms require an additional independent change detecting mechanism. As a consequence of the no free lunch (NFL) theorem(s) for optimization, there is no ultimate set of techniques for dynamic environments. The choice is always problem specific. This means that theoretically even a change in converter parameters or load type could render a selected mechanism governing the swarm movements ineffective. However, such an extreme behaviour has not been observed in the discussed system. This suggests that the problem is well-conditioned as far as (3) is considered. The wide variety of reported EDO techniques can be categorized as more or less suitable for the direct swarm repetitive controller. Unfortunately, many techniques that offer superior performance in benchmark tests are quite deleterious to the performance of the converter. For example, diversification by randomization would mean applying a random control signal to the plant for all passes related to the randomized particles. The process of randomization would then have to be controlled and the randomized values would have to be distributed in carefully arranged areas, e.g. near current attractors. The following four objectives have been assumed for the particle swarm repetitive controller:

i) all passes (≡particles) within $\mathbf{u}_{\mathrm{PSO}}(i)$ should produce similar quality of the output voltage waveform,
ii) the behaviour of the swarm should be similar in all search dimensions equivalent here to the control signal sample index along the pass,
iii) the transition from an outdated optimum to a new one should be smooth, i.e. a reasonable compromise between the time needed to get near the new optimum and the quality of the output voltage during this search has to be worked out,
iv) the tuning of the resulting repetitive controller should be as straightforward as possible, i.e. the modifications introduced to the basic PSO should entail setting only a couple of interpretation-intuitive parameters.

The condition (i) is hard to fulfil in swarms with specialization among the particles, e.g. in solutions with charged particles or quantum particles. A re-diversification by randomization should also be avoided. Preferably, the detection of the change in the landscape should not be done by re-evaluating fitness for the *pbest*s. Any re-evaluation means applying potentially outdated control signal to the plant. As far as condition (iii) is to be met, the penalty factor β in (3) plays the key role.

High values of β favour particles that carry samples arranged more like strings rather than sequences of independent values. The condition (iv) is probably the most subjective. However, some modifications to the standard PSO, although effective, require tuning of many additional parameters that usually interfere with each other in terms of the overall controller performance. The idea here is to keep the tuning procedure, which in this study involves the trial and error method, as simple as possible without sacrificing the quality of the output voltage. The standard PSO algorithm is recognized as robust and simple in tuning and thus the goal is not to ruin this with complicated modifications that are difficult to handle during the tuning stage. The condition (ii) will be addressed in Section III.

Taking into account the above objectives, the simple idea of placing repellers at the already-detected *gbest* and stored in the swarm memory *pbest*s has been chosen as a strong candidate for implementation. The mechanism has been described in [30, 31]. The speed and position update rules are as follows

$$\begin{aligned} \mathbf{v}_{nj}\left(i+1\right) &= c_1\mathbf{v}_{nj}\left(i\right) + c_2 r^{\text{pbest}}\delta_p^{\text{pbest}}\left(\mathbf{q}_{nj}^{\text{pbest}} - \mathbf{q}_{nj}\left(i\right)\right) \\ &\quad + c_3 r^{\text{gbest}}\delta_p^{\text{gbest}}\left(\mathbf{q}_{n}^{\text{gbest}} - \mathbf{q}_{nj}\left(i\right)\right) \qquad (4) \\ \mathbf{q}_{nj}\left(i+1\right) &= \mathbf{q}_{nj}\left(i\right) + \min\{ \\ &\quad \max\{-v_{\text{clmp}}, \mathbf{v}_{nj}\left(i+1\right)\}, v_{\text{clmp}}\}\ , \qquad (5) \end{aligned}$$

where $\mathbf{v}_{nj}$ and $\mathbf{q}_{nj}$ are the velocity and position of the j-th particle within the n-th subswarm, $\mathbf{q}_{nj}^{\text{pbest}}$ stores the best solution proposed so far by the j-th particle from the n-th subswarm, $\mathbf{q}_{n}^{\text{gbest}}$ denotes the best solution found so far by the n-th subswarm, and c_1, c_2 and c_3 are the inertia, cognitive and social weights, respectively. The random numbers r^{pbest} and r^{gbest} are uniformly distributed in the unit interval. Also, a fully connected network, which remains a popular choice for unimodal problems, has been adopted here. The *gbest* is then the best of all *pbest*s. The two switching variables (directions) δ_p^{pbest} and δ_p^{gbest} are bivalent (-1 or 1) and enable to switch between the attract and repel modes. It has been decided to modify the handling of these switching variables here. Originally in [30] or [31] they are not dimension-wise (p-wise), i.e. no individual control of diversity is possible in each search dimension. It will be shown later on that dimension-wise diversity measures are more effective than cumulative (i.e. aggregated over dimensions) measures in the case of the discussed direct swarm repetitive controller.

The strategy proposed in [30] defines two states of the swarm: attraction to both $\mathbf{q}_j^{\text{pbest}}$ and $\mathbf{q}^{\text{gbest}}$ or repulsion from both of them. This implies that $\delta_p^{\text{pbest}} = \delta_p^{\text{gbest}} = \delta_p$ changes its value from 1 to -1 if a given diversity threshold D_{thold} is crossed. On the other hand, the strategy proposed in [31] introduces one additional intermediate state in which particles are repelled by $\mathbf{q}^{\text{gbest}}$ but are still attracted by their $\mathbf{q}_j^{\text{pbest}}$-s. This introduces two different switching thresholds $D_{\text{thold}}^{\text{pbest}}$ and $D_{\text{thold}}^{\text{gbest}}$. As mentioned before, originally both strategies have been proposed in conjunction with the cumulative (i.e. non-p-wise) diversity measure

TABLE I
PARAMETERS OF THE SWARM

Parameter	Symbol	Value
Dimensionality of the problem	α	200
Number of particles (swarm size)	S	25
Swarm update frequency	$\frac{f^{\text{ref}}}{S}$	2 Hz
Evaporation constant	ρ	1.15
Diversity threshold	D_{thold}	2
Number of subswarms	N	10
Penalty factor	β	0.25

$$D_{\text{dist}}(i) = \frac{1}{S\sqrt{\alpha}} \sum_{j=1}^{S} \sqrt{\sum_{p=1}^{\alpha} \left(\mathbf{q}_j(p,i) - \overline{\mathbf{q}}(p,i)\right)^2}\ , \qquad (6)$$

where S is the swarm size, α is the dimensionality of the problem (here equal to the number of samples per period of the reference signal), $\overline{\mathbf{q}}$ is the average point and i is the swarm iteration index. In (6) the dimension index p does not occur on the left hand side, because this measure represents the mean Euclidean distance of all particles to the average point scaled by $\alpha^{-0.5}$.

To deal with an outdated memory, the gradual knowledge evaporation concept [32] has been used. This technique forces particles to gradually lose their fitness. Personal fitness, which is constant in the basic PSO if a better solution does not emerge, here evaporates at the constant rate ρ as described in [24]. Such a mechanism complies with the requirements (i)-(iv). The algorithm does not revisit $\mathbf{q}_j^{\text{pbest}}$ to detect change. It also does not introduce thresholds to detect stale memories, which is very desirable in noisy environments. It would be difficult to define such a threshold in a Gaussian-noise-polluted environment. Also, the formula contributes only one parameter to be tuned. The bigger the value of ρ, the faster the transition to the new optimum after a change in the shape of the load current. However, too big a value of ρ is detrimental to output voltage quality under repetitive disturbance due to the excessive evaporation of good solutions that in this particular situation do not become outdated. It is to be noted that the speed of forgetting is exponential. For example, for $\rho = 1.15$ the personal fitness of a particle is reduced more than twice after 5 swarm iterations.

III. Diversity measures

Diversity measures are sometimes used to assess the performance of the swarm and their effectiveness is mostly discussed in the context of static environments [33–35]. They can serve, e.g., as the stopping criterion. Among the most popular measures are the moments of distribution (statistical measures) and geometrical measures, e.g. the radius of the swarm. Initially, the repetitive swarm controller [23, 36] relied on non-dimension-wise measures such as (6). Several other cumulative measures could be considered. For example, standard deviations of proposed solutions can be merged into a single diversity index by calculating their mean value per dimension

$$D_{\mathrm{std}}(i)=\frac{1}{\alpha}\sum_{p=1}^{\alpha}\sqrt{\frac{1}{S}\sum_{j=1}^{S}\left(\mathbf{q}_j(p,i)-\overline{\mathbf{q}}(p,i)\right)^2}. \tag{7}$$

Identical merging can be used to transform any dimension-wise measure into a cumulative one. Both (6) and (7) produce very similar behavior of the swarm in the PDPSRC – the solution is operable but it lacks the desirable property stated in the condition (ii) from Section II. It has been observed that, although the quality of the voltage is similar for all particles, there are noticeable fluctuations in the quality along the pass. These discrepancies are often dimension specific, e.g. a specific p-th sample in each pass is clearly outdated, and they do not necessarily occur at the sample correlated with the occurrence of a high absolute value of the load current derivative. This could suggest that some level of improvement is achievable when implementing a dimension-wise (i.e. p-wise) diversity control. The goal is to comply with the requirement (ii) formulated in Section II. Moreover, the final decision on the measure to be employed should be driven by two more key aspects: real-time implementation and the rate of convergence. For example, the calculation of the square root should be evaded, the L^1-norm is preferable over the L^2-norm.

Three more candidate diversity measures are taken into account:

a) the dimension-wise (p-wise) counterpart for (7) simplified by omitting the square root

$$D_{\mathrm{var}}^{p-\mathrm{wise}}(p,i)=\frac{1}{S}\sum_{j=1}^{S}\left(\mathbf{q}_j(p,i)-\overline{\mathbf{q}}(p,i)\right)^2\ , \tag{8}$$

b) the position diversity on each dimension based on the L^1-norm (mean/average absolute deviation)

$$D_{\mathrm{abs}}^{p-\mathrm{wise}}(p,i)=\frac{1}{S}\sum_{j=1}^{S}\left|\mathbf{q}_j(p,i)-\overline{\mathbf{q}}(p,i)\right|\ , \tag{9}$$

c) the dimension-wise radius of the swarm

$$D_{\mathrm{radius}}^{p-\mathrm{wise}}(p,i)=\frac{1}{2}\left(\max\{\mathbf{q}_1(p,i),\ldots,\mathbf{q}_S(p,i)\}\right.$$
$$\left.-\min\{\mathbf{q}_1(p,i),\ldots,\mathbf{q}_S(p,i)\}\right)\ . \tag{10}$$

When comparing the above p-wise measures, the thresholds have to be specified individually. However, no perfectly fair comparison is possible. For example, to compare $D_{\mathrm{var}}^{p-\mathrm{wise}}$ with $D_{\mathrm{radius}}^{p-\mathrm{wise}}$, a certain level of statistical significance has to be chosen. The goal is then not to determine the absolute winner, but to test whether or not one of them manifests some convincing level of improvement over the others. The objective is to verify if p-wise measures are significantly different in the resulting performance of the PDMSRC in comparison to the non-dimension-wise measures employed originally. It should be noted that the p-wise measures are calculated identically for single-swarm and multi-swarm controllers, whereas any cumulative measure – such as (6) or (7) – can be calculated for each subswarm separately or additionally aggregated over all subswarms. Both approaches have been tested to draw conclusions in this study. As the PDMSRC acts only in the k-direction, the accompanying controller depicted in Fig. 1 has been proposed.

TABLE II
PARAMETERS OF THE CONVERTER

Parameter	Value
Filter inductance	300 µH
Filter capacitance	160 µF
Filter resistance R_{f}	0.2 Ω
Filter resonant frequency	726 Hz
Critical damping resistance R_{crit}	2.74 Ω
Reference frequency f^{ref}	50 Hz
Sampling/PWM frequency	10 kHz
Measurement noise*	see annotations
Pass length α	200
DC-link voltage	450 V
Rectifier power	ca. 6 kW
Rectifier current crest factor	ca. 2.5
Resistive load power	ca. 4 kW

* 98% confidence interval

IV. Full-state feedback controller and repetitive disturbance feedforward

Although the considered plant simplified to the RLC circuit with a controlled voltage source is inherently stable, natural damping is very low since it comes only from the parasitic resistance of the circuit (compare R_{f} with R_{crit} in Tab. II). The full-state feedback (FSF) has been implemented to shift continuous-time closed-loop poles to the left in the complex plane. The reference feedforward (RFF) path is also added to keep a unity gain for the zero frequency. Also the repetitive disturbance feedforward path described in [36] is used here to compensate the resistive voltage drop (for the zero frequency)

$$u_{\mathrm{RDFF}}=(\hat{R}_{\mathrm{f}}+k_{11})i_{\mathrm{load}}^{\mathrm{predicted}}, \tag{11}$$

where $\hat{R}_{\mathrm{f}}$ is the identified resistance of the output filter. The prediction is made based on the previous pass

$$i_{\mathrm{load}}^{\mathrm{predicted}}=i_{\mathrm{load}}^{\mathrm{m}}(p,k)+i_{\mathrm{load}}^{\mathrm{m}}(p+\tau,k-1)-i_{\mathrm{load}}^{\mathrm{m}}(p,k-1), \tag{12}$$

where τ represents the delay time in the delay-non-compensated DFF path due to the digital implementation of the controller and an inherent delay of the PWM.

V. Numerical experiment results

A comparison of the proposed dimension-wise repel and attract modes against the standard method has been performed using exactly the same test scenario as in [24]. All comparisons assume that $D_{\mathrm{var}}^{p-\mathrm{wise}}=\sigma^2$ is collated with $D_{\mathrm{abs}}^{p-\mathrm{wise}}=\sigma\sqrt{2\pi^{-1}}$ and $D_{\mathrm{radius}}^{p-\mathrm{wise}}=2\sigma$. It should be stressed that the discussed particles are not normally distributed. For more information on Gaussian PSO follow [37]. However, to make the comparison less subjective, the mean absolute deviations for Gaussian distribution and the 2-sigma threshold (more than

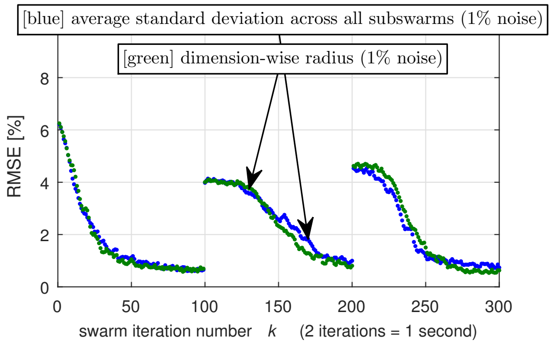


Fig. 2. Evolution of the root mean squared error under a cumulative versus a dimension-wise diversity control – the 10-subswarm case.

95% of the population) have been used here. For the sake of brevity, only selected results are included in the paper. More comparisons can be easily generated using the complete numerical model published on MATLAB Central [38].

It has been observed that the dimension-wise measures offer a slightly better convergence rate in comparison with the cumulative ones. This should be attributed to the evident property of (6) or (7) – their values can be high even if particles have collapsed in selected dimensions. However, the winning margin in terms of the convergence rate diminishes with the growing number of subswarms and for e.g. 10 subswarms it becomes negligible. This is illustrated in Fig. 2. On the other hand, if the output voltage waveform is inspected, a clear difference between the cumulative and the dimension-wise measures emerges. The averaging property of (7) makes it more prone to let particles collapse in some dimensions while still keeping the whole swarm above a given diversity threshold. This in turn impairs exploration in these dimensions and leads to bigger control errors. Such an outcome is exposed in Fig. 3. Moreover, the p-wise measures are preferable in view of their time-distributed calculations. Sticking to originally proposed cumulative measures would preclude effective real-time code design reported in [39, 40].

All three p-wise diversity measures manifest similar effectiveness in terms of the area under the RMSE curve (Fig. 4). However, the calculation of the radius outperforms other measures in respect of its fast numerical execution. Moreover, the threshold for the radius clearly renders the performance in terms of the output voltage waveform fluctuations from pass to pass. If all particles in all dimensions are below the threshold, all control signals are within the envelope no wider than $2D_{\text{radius}}^{p-\text{wise}}$. A similar conclusion cannot be drawn for $D_{\text{var}}^{p-\text{wise}}$ and $D_{\text{abs}}^{p-\text{wise}}$ because both of them can drop below a given threshold despite the presence of some outliers. Monitoring only the radius also has its drawbacks. A single particle easily forces this measure to cross the threshold whereas all other particles could get glued together which in turn could impair the swarm's ability to respond equally effectively in all dimensions. However, due to the gradual knowledge evaporation even this relatively unusual situation does not result in a disability to follow a moving optimum. Moreover, in a noisy measurement environment, which is the case in any CACF VSI, the likelihood of the mentioned behaviour is negligibly small.

Furthermore, it has been tested if there are clear benefits of introducing different repel/attract mode switching levels for the *gbest* and the *pbest*s. The goal here is then to try to find by guessing and checking two values that produce a clearly better behaviour of the repetitive controller in comparison to the swarm with $\delta_p^{\text{pbest}} = \delta_p^{\text{gbest}}$. Despite numerous trials no obvious improvement of the output voltage waveform or the convergence rate has been observed. Therefore, it can be roughly concluded that the single threshold formula is sufficient. The obtained results support the idea of using the dimension-wise geometric diversity measure, namely the radius, in conjunction with the one-level triggered repel/attract modes to synthesize the plug-in direct particle swarm repetitive controller.

VI. CONCLUSIONS

Several different diversity measures have been investigated in the context of the plug-in direct particle swarm repetitive controller for the true sine wave inverter. The dimension-wise radius has been identified as a strong candidate for the swarm-based repetitive controller. The originally proposed cumulative/averaging diversity measures have been abandoned. The proposed approach outperforms them in terms of: the calculations being easily distributable in time and thus a low computation burden being imposed on the digital signal controller, a better responsiveness of the swarm, and a better quality of the output voltage waveform. The very basic geometric measure in the form of radius is then recommended for real-time implementation.

ACKNOWLEDGMENT

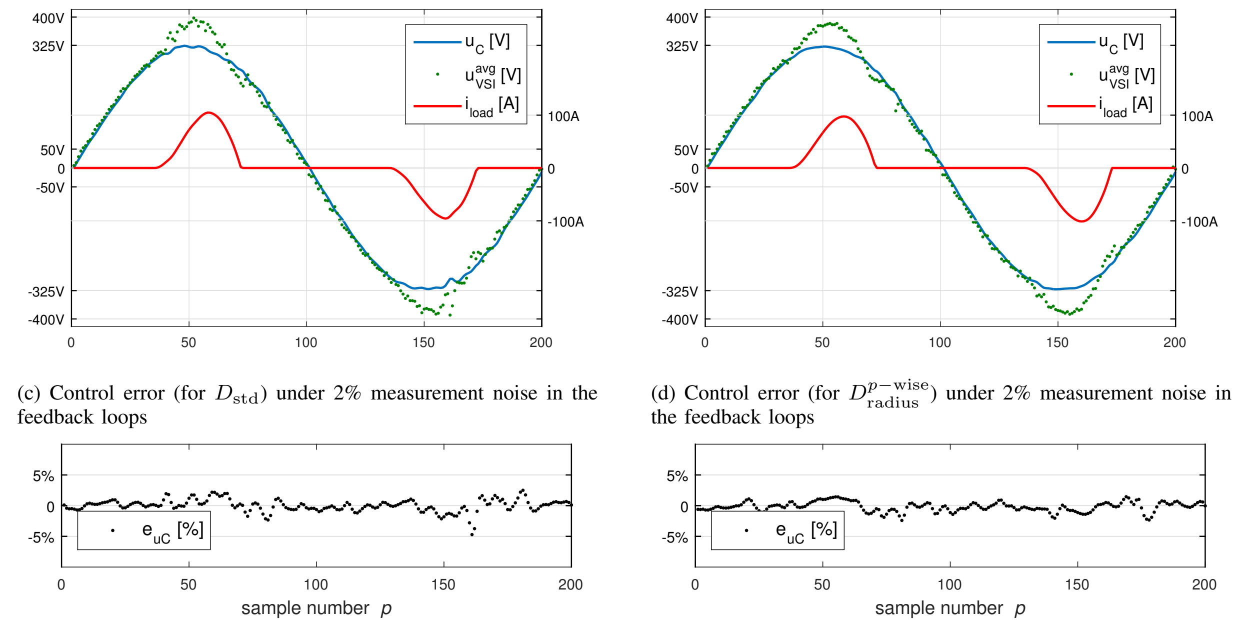
The research was partially supported by the statutory fund of the Electrical Drive Division within the Institute of Control and Industrial Electronics, Faculty of Electrical Engineering, Warsaw University of Technology, Poland.

REFERENCES

[1] G. Escobar, P. Mattavelli, M. Hernandez-Gomez, and P. R. Martinez-Rodriguez, "Filters with linear-phase properties for repetitive feedback," *IEEE Trans. Industrial Electronics*, vol. 61, no. 1, pp. 405–413, Jan. 2014.

[2] B. Ufnalski, A. Kaszewski, and L. M. Grzesiak, "Particle swarm optimization of the multi-oscillatory LQR for a three-phase four-wire voltage source inverter with an LC output filter," *IEEE Trans. Industrial Electronics*, vol. 62, no. 1, pp. 484–493, Jan. 2015. DOI: 10.1109/TIE.2014.2334669.

[3] L. F. Alves Pereira, J. Vieira Flores, G. Bonan, D. Ferreira Coutinho, and J. M. Gomes da Silva Jr., "Multiple resonant controllers for uninterruptible power supplies – a systematic robust control design approach," *IEEE Trans. Industrial Electronics*, vol. 61, no. 3, pp. 1528–1538, Mar. 2014.



(a) Output capacitor voltage, load current, and PWM converter average voltage (for D_{std})

(b) Output capacitor voltage, load current, and PWM converter average voltage (for $D_{\mathrm{radius}}^{p-\mathrm{wise}}$)

(c) Control error (for D_{std}) under 2% measurement noise in the feedback loops

(d) Control error (for $D_{\mathrm{radius}}^{p-\mathrm{wise}}$) under 2% measurement noise in the feedback loops

Fig. 3. Comparison of tracking performance under the diode rectifier load for two different diversity measures – the depicted control error calculated as $u_{\mathrm{C}}^{\mathrm{ref}} - u_{\mathrm{C}}$ instead of the less informative $u_{\mathrm{C}}^{\mathrm{ref}} - u_{\mathrm{C}}^{\mathrm{m}}$.

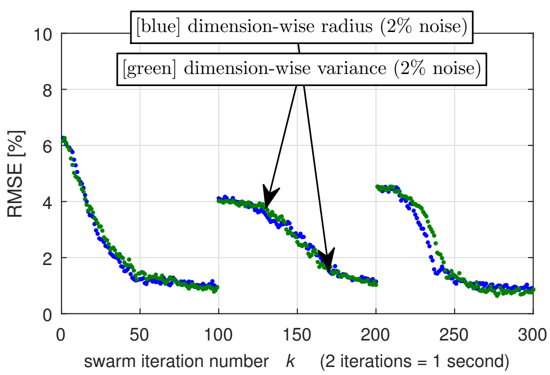


Fig. 4. Evolution of the root mean squared error (calculated per period of $u_{\mathrm{C}}^{\mathrm{ref}}$) for two selected dimension-wise mechanisms.

[4] E. Rogers, K. Galkowski, W. Paszke, K. L. Moore, P. H. Bauer, L. Hladowski, and P. Dabkowski, "Multidimensional control systems: Case studies in design and evaluation," *Multidimensional Systems and Signal Processing*, vol. 26, no. 4, pp. 895–939, 2015.

[5] S. Mandra, K. Galkowski, H. Aschemann, and A. Rauh, "Guaranteed cost iterative learning control – an application to control of permanent magnet synchronous motors," in *IEEE 9th International Workshop on Multidimensional (nD) Systems (nDS)*, 2015, pp. 1–6.

[6] 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," *IEEE Trans. Circuits and Systems I: Fundamental Theory and Applications*, vol. 49, no. 6, pp. 753–767, 2002.

[7] R. W. Longman, "On the interaction between theory, experiments, and simulation in developing practical learning control algorithms," *International Journal of Applied Mathematics and Computer Science*, vol. 13, no. 1, pp. 101–111, 2003.

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

[9] Z. Cai, "Iterative learning control: Algorithm development and experimental benchmarking," PhD thesis, University of Southampton, UK, 2009.

[10] Y. Shi, "Robustification in repetitive and iterative learning control," PhD thesis, Columbia University, USA, 2013.

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

[12] B. Ufnalski, L. M. Grzesiak, and K. Galkowski, "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*, vol. 61, no. 3, pp. 1–12, Sep. 2013. DOI: 10.2478/bpasts-2013-0069.

[13] Y. Wang, F. Gao, and F. J. Doyle III, "Survey on iterative learning control, repetitive control, and run-to-

run control," *Journal of Process Control*, vol. 19, no. 10, pp. 1589–1600, 2009.

[14] M. H. A. Verwoerd, "Iterative learning control – a critical review," PhD thesis, University of Twente, The Netherlands, 2005.

[15] B. Ufnalski. (2015). Multiresonant versus iterative learning control, [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/53967-multiresonant-versus-iterative-learning-control.

[16] T. T. Nguyen, S. Yang, and J. Branke, "Evolutionary dynamic optimization: A survey of the state of the art," *Swarm and Evolutionary Computation*, vol. 6, pp. 1–24, 2012.

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

[18] B. Ufnalski and L. M. Grzesiak, "Particle swarm optimization of an online trained repetitive neurocontroller for the sine-wave inverter," in *39th Annual Conference of the IEEE Industrial Electronics Society (IECON)*, Nov. 2013, pp. 6003–6009. DOI: 10.1109/IECON.2013.6700120.

[19] B. Ufnalski and L. M. Grzesiak, "Repetitive neurocontroller with disturbance dual feedforward – choosing the right dynamic optimization algorithm," in *17th European Conference on Power Electronics and Applications (EPE'15 ECCE-Europe)*, 2015, pp. 1–10. DOI: 10.1109/EPE.2015.7309263.

[20] B. Ufnalski and L. M. Grzesiak, "Repetitive neurocontroller with disturbance feedforward path active in the pass-to-pass direction for a VSI inverter with an output LC filter," *Bulletin of the Polish Academy of Sciences – Technical Sciences*, vol. 64, no. 1, pp. 115–125, 2016. DOI: 10.1515/bpasts-2016-0013.

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

[22] B. Ufnalski. (2015). Repetitive neurocontroller training algorithms, [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/53466-repetitive-neurocontroller-training-algorithms.

[23] 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, Jun. 2014. [Online]. Available: http://pe.org.pl/articles/2014/6/2.pdf.

[24] B. Ufnalski and L. M. Grzesiak, "Plug-in direct particle swarm repetitive controller with a reduced dimensionality of a fitness landscape — a multi-swarm approach," *Bulletin of the Polish Academy of Sciences – Technical Sciences*, vol. 63, no. 4, pp. 857–866, Dec. 2015. DOI: 10.1515/bpasts-2015-0098.

[25] B. Ufnalski and L. M. Grzesiak, "A performance study on synchronous and asynchronous update rules for a plug-in direct particle swarm repetitive controller," *Archives of Electrical Engineering*, vol. 63, no. 4, pp. 635–646, Dec. 2014. DOI: 10.2478/aee-2014-0044.

[26] B. Ufnalski. (2014). Plug-in direct particle swarm repetitive controller, [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/47847-plug-in-direct-particle-swarm-repetitive-controller.

[27] R. C. Eberhart, Y. Shi, and J. Kennedy, *Swarm Intelligence*, 1st, ser. The Morgan Kaufmann Series in Evolutionary Computation. Morgan Kaufmann Publishers, Apr. 2001.

[28] T. Blackwell, "Particle swarm optimization in dynamic environments," in *Evolutionary Computation in Dynamic and Uncertain Environments*, ser. Studies in Computational Intelligence, S. Yang, Y.-S. Ong, and Y. Jin, Eds., vol. 51, Springer Berlin Heidelberg, 2007, pp. 29–49.

[29] T. Blackwell, J. Branke, and X. Li, "Particle swarms for dynamic optimization problems," in *Swarm Intelligence*, ser. Natural Computing Series, C. Blum and D. Merkle, Eds., Springer Berlin Heidelberg, 2008, pp. 193–217.

[30] J. Riget and J. S. Vesterstrøm, "A diversity-guided particle swarm optimizer - the ARPSO," Aarhus Universitet, Denmark, Tech. Rep., 2002.

[31] M. Pant, T. Radha, and V. P. Singh, "A simple diversity guided particle swarm optimization," in *IEEE Congress on Evolutionary Computation (CEC)*, 2007, pp. 3294–3299.

[32] X. Cui, J. S. Charles, and T. E. Potok, "A simple distributed particle swarm optimization for dynamic and noisy environments," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2008)*, ser. Studies in Computational Intelligence, vol. 236, Springer Berlin Heidelberg, 2009, pp. 89–102.

[33] Y. Shi and R. C. Eberhart, "Population diversity of particle swarms," in *IEEE Congress on Evolutionary Computation (CEC), IEEE World Congress on Computational Intelligence*, 2008, pp. 1063–1067.

[34] S. Cheng and Y. Shi, "Diversity control in particle swarm optimization," in *IEEE Symposium on Swarm Intelligence (SIS)*, 2011, pp. 1–9.

[35] O. Olorunda and A. P. Engelbrecht, "Measuring exploration/exploitation in particle swarms using swarm diversity," in *IEEE Congress on Evolutionary Computation (CEC), IEEE World Congress on Computational Intelligence*, 2008, pp. 1128–1134.

[36] B. Ufnalski and L. M. Grzesiak, "Feedback and feedforward repetitive control of single-phase UPS inverters - an online particle swarm optimization approach," Scientific Reports of the Cologne University of Applied Sciences, Tech. Rep. 1, 2014.

[37] R. A. Krohling and L. dos Santos Coelho, "PSO-E: Particle swarm with exponential distribution," in *IEEE Congress on Evolutionary Computation (CEC)*, 2006, pp. 1428–1433.

[38] B. Ufnalski. (2016). How to keep your swarms diversified, [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/55606-how-to-keep-your-swarms-diversified.
[39] B. Ufnalski. (2015). Time-distributed particle swarm repetitive control algorithm, [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/48967-time-distributed-particle-swarm-repetitive-control-algorithm.
[40] P. Biernat, B. Ufnalski, and L. M. Grzesiak, "Real-time implementation of the multi-swarm repetitive control algorithm," in *9th International Conference on Compatibility and Power Electronics (CPE)*, 2015, pp. 119–125. DOI: 10.1109/CPE.2015.7231059.