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

Bartlomiej Ufnalski and Lech M. Grzesiak

Institute of Control and Industrial Electronics
Warsaw University of Technology, POLAND

{bartlomiej.ufnalski, lech.grzesiak}@ee.pw.edu.pl

SENEN 2013, November 20–22, Łódź, Poland
Introduction to control techniques for repetitive processes

Particle swarm optimization for SOPs and DOPs

Direct particle swarm repetitive controller

Results and conclusion(s)
Some known techniques and the novelty

- Non-repetitive controllers — the repetitiveness of the process is being neglected.
Some known techniques and the novelty

- Non-repetitive controllers — the repetitiveness of the process is being neglected.

Bartłomiej Ufnalski and Lech M. Grzesiak
Plug-in direct particle swarm repetitive controller (PDPSRC)
Some known techniques and the novelty

- Non-repetitive controllers — the repetitiveness of the process is being neglected.
- Repetitive control (iterative learning control, ILC) — a 2D approach with the along-the-pass and the pass-to-pass directions to be considered (PAN Bulletin 3’2013).
Some known techniques and the novelty

- Non-repetitive controllers — the repetitiveness of the process is being neglected.
- Repetitive control (iterative learning control, ILC) — a 2D approach with the along-the-pass and the pass-to-pass directions to be considered (PAN Bulletin 3’2013).
- Repetitive neurocontrollers — a fairly new concept within the frame of 2D systems (Przegląd Elektrotechniczny 4’2013, IEEE IECON 2013).
Some known techniques and the novelty

- Non-repetitive controllers — the repetitiveness of the process is being neglected.
- Repetitive control (iterative learning control, ILC) — a 2D approach with the along-the-pass and the pass-to-pass directions to be considered (PAN Bulletin 3’2013).
- Repetitive neurocontrollers — a fairly new concept within the frame of 2D systems (Przegląd Elektrotechniczny 4’2013, IEEE IECON 2013).
- **Direct particle swarm repetitive controller** — a novelty within the frame of 2D systems (SENE 2013).
The very basic P-type control law

\[ u(p, k) = u(p, k - 1) + K_{RC} e(p, k - 1), \]

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 \( \alpha \) is the pass length).

Theoretically perfect tracking could be achieved if only...
Long-term stability issues

... this formula had been stable!

It then has to be modified into

\[ u(p, k) = Q\left(z^{-1}\right) u(p, k - 1) + L\left(z^{-1}\right) e(p, k - 1) \]

where \(Q\) and \(L\) are usually non-causal low-pass zero-phase-shift filters. This compromises the performance and hence there still is plenty of room for new iterative learning techniques.
Two-dimensional (2D) behavior of a controller

The objective

Rejection of a repetitive disturbance load current in a constant-amplitude constant-frequency VSI
✓ a measurement white noise
✓ a control signal white noise to acknowledge system noise
✗ a pulse width modulator, a dead time and switches
The control objective

The cost function for the controller

\[ J(k) = \left( \sum_{p=1}^{\alpha} \left( \frac{\left( u^\text{ref}_C(p) - u^m_C(p, k) \right)^2}{\text{penalty for control error}} + \frac{\beta \left( u^\text{PSO}_{C}(p, k) - u^\text{PSO}_{C}(p-1, k) \right)^2}{\text{penalty for control signal dynamics}} \right) \right)^{-1} \]

The reciprocal is introduced just to turn the optimization problem into the maximization one.

A population based optimizer has been employed to perform this optimization in the online mode.
PSO for static optimization problems

\[ v_j(i+1) = c_1 v_j(i) + c_2 r_p (q_j^{pbest} - q_j(i)) + c_3 r_g (q_g^{gbest} - q_j(i)) \]
\[ q_j(i+1) = q_j(i) + v_j(i+1), \]

where

- \( j \) is the particle identification number,
- \( i \) denotes the swarm iteration number,
- \( v_j \) and \( q_j \) are speed and position of the \( j \)-th particle,
- \( q_j^{pbest} \) stores the best solution proposed so far by the \( j \)-th particle,
- \( q_g^{gbest} \) 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,
- \( r_p \) and \( r_g \) are random numbers uniformly distributed in the unit interval.
Particle swarm optimization for SOPs and DOPs

Direct particle swarm repetitive controller

Results and conclusion(s)

PSO for dynamic optimization problems – the diversity

Swarm diversity control:

$$\mathbf{v}_j(i+1) = c_1 \mathbf{v}_j(i) + c_2 r_p \delta (\mathbf{q}_j^{p\text{best}} - \mathbf{q}_j(i)) + c_3 r_g \delta (\mathbf{q}_g^{g\text{best}} - \mathbf{q}_j(i))$$

with

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

and

$$D_{\text{dist}} = \frac{1}{N_p \sqrt{N_d}} \sum_{j=1}^{N_p} \left( \sum_{n=1}^{N_d} (q_{jn} - \bar{q}_n)^2 \right)$$

where $N_p$ is the swarm size, $N_d$ is the dimensionality of the problem and $\bar{q}$ is the average point.
PSO for dynamic optimization problems – the outdated memory

The evaporation constant $\rho$ for the personal fitness value $P_j = \mathcal{J}(q_j^{pbest})$ stored in each particle’s memory forces particles to gradually forget previous best solution. The formula of forgetting is as follows

$$
\begin{bmatrix}
P_j(i+1) \\
q_j^{pbest}
\end{bmatrix}
=\begin{cases}
\begin{bmatrix}
\rho P_j(i) \\
q_j^{pbest}
\end{bmatrix} & \text{if } \eta_{j+1} \leq \rho P_j(i) \\
\begin{bmatrix}
\eta_{j+1} \\
q_j(i+1)
\end{bmatrix} & \text{if } \eta_{j+1} > \rho P_j(i)
\end{cases}
$$

where $\eta_{j+1} = \mathcal{J}(q_j(i+1))$ is the current fitness of the $j$-th particle and $\rho$ has a positive value less than 1 for any positive-definite functional $\mathcal{J}$ and an optimization task formulated as the maximization one.
The idea of a **direct** particle swarm controller

The particle is a vector of $\alpha$ consecutive samples of the control signal along the pass

$$q = [u_{PSO}(1), u_{PSO}(2), u_{PSO}(3), \ldots, u_{PSO}(\alpha)]$$
The idea of a particle swarm repetitive controller

Is it feasible to organize the control task in the following manner:

1. apply control signal proposed by the particle,
Is it feasible to organize the control task in the following manner:

1. apply control signal proposed by the particle,
2. calculate fitness value for the particle,
Is it feasible to organize the control task in the following manner:

1. apply control signal proposed by the particle,
2. calculate fitness value for the particle,
3. repeat 1-2 until all particles are rated (without resetting the physical process),
The idea of a particle swarm repetitive controller

Is it feasible to organize the control task in the following manner:

1. apply control signal proposed by the particle,
2. calculate fitness value for the particle,
3. repeat 1-2 until all particles are rated (without resetting the physical process),
4. update particles, i.e. update the swarm,
Is it feasible to organize the control task in the following manner:

1. apply control signal proposed by the particle,
2. calculate fitness value for the particle,
3. repeat 1-2 until all particles are rated (without resetting the physical process),
4. update particles, i.e. update the swarm,
5. repeat 1-4 until the system is in operation (without resetting the physical process)?
## Selected parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LC output filter</td>
<td>300 µH, 160 µF, 0.2Ω</td>
</tr>
<tr>
<td>Reference</td>
<td>$f_{\text{ref}} = 50,\text{Hz}$, $U_{\text{RMS}}^{\text{ref}} = 230,\text{Vrms}$, sinusoidal</td>
</tr>
<tr>
<td>Sampling time</td>
<td>$T_s = 100,\mu\text{s}$ ($\alpha = N_d = 200$ point per pass)</td>
</tr>
<tr>
<td>Measur. noise</td>
<td>ca. 1%</td>
</tr>
<tr>
<td>DC-link voltage</td>
<td>450V</td>
</tr>
<tr>
<td>Load-1</td>
<td>Diode rectifier: 500 µH, 3 mF, 6 kW, crest factor of 2.5</td>
</tr>
<tr>
<td>Load-2</td>
<td>Resistive: 4 kW</td>
</tr>
<tr>
<td>Dimensionality of the problem $N_d$, $\alpha$</td>
<td>200</td>
</tr>
<tr>
<td>Number of particles $N_p$</td>
<td>25</td>
</tr>
<tr>
<td>Swarm update frequency $f_{\text{ref}}/N_p$</td>
<td>2 Hz</td>
</tr>
<tr>
<td>Evaporation constant $\rho$</td>
<td>0.97</td>
</tr>
<tr>
<td>Diversity threshold $D_{\text{thold}}$</td>
<td>0.5 · 325⁻¹</td>
</tr>
<tr>
<td>Diversity hysteresis $h$</td>
<td>0</td>
</tr>
<tr>
<td>Penalty factor $\beta$</td>
<td>2.5</td>
</tr>
</tbody>
</table>
Open-loop behavior of the plant (only the RFF is implemented)
The PDPSRC has been turned on
The idea of a plug-in particle swarm controller

The full-state feedback (FSF) is introduced and poles of the plant seen by the PDPSRC are shifted to the left. In this particular experiment the closed-loop conjugate poles are designed to increase damping 5 times.
Closed-loop behavior of the plant (only the RFF and the non-augmented FSF are implemented)
Introduction to control techniques for repetitive processes
Particle swarm optimization for SOPs and DOPs
Direct particle swarm repetitive controller
Results and conclusion(s)

The PDPSRC has been turned on

sample number $p$
p-direction
along the pass direction

swarm iteration $k$
$\rightarrow$-direction
pass to pass direction

$u_c$ [V]

$u_VSI$ average

$i_{\text{load}}$

RMSE [V]

$u_C$

$t$ [s]

Bartłomiej Ufnalski and Lech M. Grzesiak
Plug-in direct particle swarm repetitive controller (PDPSRC)
Stability, exploitation and responsiveness of the swarm

Stability

A stability proof is not available at the moment. It will be hard or even impossible to provide such a proof. The situation is similar to the lack of stability proofs for most predictive control schemes that incorporate cost function online minimization.

\[ \rho = 0.85 \text{ instead of } 0.97 \text{ and } D_{\text{thold}} = 1.5 \cdot 325^{-1} \text{ instead of } 0.5 \cdot 325^{-1} \] — better responsiveness at the cost of worse exploitation.
Conclusions

- The novel swarm based repetitive controller has been developed and tested numerically.
Conclusions

- The novel swarm based repetitive controller has been developed and tested numerically.
- The proposed algorithm does not suffer from long term stability issues that often appear in classic iterative learning control schemes. No low-pass filtering is needed to robustify the control scheme.
The novel swarm based repetitive controller has been developed and tested numerically.

The proposed algorithm does not suffer from long term stability issues that often appear in classic iterative learning control schemes. No low-pass filtering is needed to robustify the control scheme.

It has been illustrated that the PSO technique can be used in online mode to directly shape the control signal for the repetitive process.
The novel swarm based repetitive controller has been developed and tested numerically.

The proposed algorithm does not suffer from long term stability issues that often appear in classic iterative learning control schemes. No low-pass filtering is needed to robustify the control scheme.

It has been illustrated that the PSO technique can be used in online mode to directly shape the control signal for the repetitive process.

The system manifests an acceptable trade-off between exploration and exploitation capabilities in the pass-to-pass direction, i.e. is able to iteratively compensate for the repetitive disturbance load current.
Questions?

Thank you for your kind attention!

And please do not hesitate to contact us at

✉️ *bartlomiej.ufnalski@ee.pw.edu.pl*

if new questions arise after reading ANY of our papers.

This presentation will be available at

🌐 *www.ufnalski.edu.pl.*

Please cite the accompanying paper using the following B_{\text{IB}}T_{\text{E}}X entry

```latex
@INPROCEEDINGS{PDPSRC_Ufnalski_SENE_2013,
author={Ufnalski, B. and Grzesiak, L. M.},
booktitle={XI Conference on Control in Power Electronics and Electrical Drives, \url{sene.p.lodz.pl}, SENE 2013},
title={A plug-in direct particle swarm repetitive controller for a single-phase inverter},
month={Nov.}, year={2013}, pages={1--6},}
```

if you find this technique inspiring.