Particle Swarm Optimization

David Lehotzky

April 22, 2022

1 Concept

Particle swarm optimization (PSO) optimizes a problem by having a population (swarm) of candidate solutions (particles), and moving these particles around in the search-space according to simple mathematical formula over the particle's position and velocity. Each particle's movement is influenced by its local best known position, but is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions.

2 Properties

- **Gradient-free method:** it does not use the gradient of the optimized problem, hence the optimization problem does not need to be differentiable.
- **Metaheuristic method:**
  - does not guarantee an optimal solution is ever found,
  - makes few or no assumptions about the optimized problem.
- It can search very large spaces of candidate solutions.

3 Method

Let $f : \mathbb{R}^d \to \mathbb{R}$ be the cost function which must be minimized. The goal is to find a solution $\mathbf{a} \in \mathbb{R}^d$ for which $f(\mathbf{a}) \leq f(\mathbf{b})\ \forall \mathbf{b}$ in the search space, i.e., we are searching for a global minimum $\mathbf{a}$.

Let $\Omega(t)$ be a set of N particles in $\mathbb{R}^d$ at a discrete time step t. Then $\Omega(t)$ is said to be the particle swarm at time t. At time step t, particles $i = 1, \ldots, N$ in the swarm have position $\mathbf{x}_i(t) \in \mathbb{R}^d$ and velocity $\mathbf{v}_i(t) \in \mathbb{R}^d$. At time step t, let $\mathbf{y}_i(t)$ be the best known position of particle i and let $\hat{\mathbf{y}}(t)$ be the best known position of the entire swarm.

3.1 Basic PSO

In the Basic PSO Algorithm 1, parameters $\mathbf{b}_{\text{lo}}$ and $\mathbf{b}_{\text{up}}$ represent the lower and upper boundaries of the search-space, respectively. Parameter w is the inertia weight, while c_1 and c_2 are called cognitive and social coefficient. Operator $\otimes$ indicates component-wise multiplication of two vectors.

3.2 Canonical PSO

In the canonical PSO Algorithm 2, $\mathbf{y}_i(t)$ and $\hat{\mathbf{y}}_i(t)$ represent the personal and neighborhood best positions, respectively, of particle i at time step t. Note that contrary to the Basic PSO, here the position and velocity of the entire swarm are updated together after the update of personal and neighborhood best positions have been completed for all particles.

Algorithm 1 Basic Particle Swarm Optimization

for $i \leftarrow 1$ to N **do**
  $\mathbf{x}_i(0) \sim \text{Unif}(\mathbf{b}_{\text{lo}}, \mathbf{b}_{\text{up}})$ ▷ Initialize the particle's position with a uniformly distributed random vector.
  $\mathbf{y}_i(0) \leftarrow \mathbf{x}_i(0)$ ▷ Initialize the particle's best known position to its initial position.
  if $f(\mathbf{y}_i(0)) < f(\hat{\mathbf{y}}(0))$ **then**
    $\hat{\mathbf{y}}(0) \leftarrow \mathbf{y}_i(0)$ ▷ Update swarm's best known position.
  end if
  $\mathbf{v}_i(0) \sim \text{Unif}(-|\mathbf{b}_{\text{up}} - \mathbf{b}_{\text{lo}}|, |\mathbf{b}_{\text{up}} - \mathbf{b}_{\text{lo}}|)$ ▷ Initialize particle's velocity.
end for
$t \leftarrow 0$
while a termination criterion is not met **do**
  for $i \leftarrow 1$ to N **do**
    $\mathbf{r}_{i,1}(t), \mathbf{r}_{i,2}(t) \leftarrow \text{Unif}(\mathbf{0}, \mathbf{1})$ ▷ Pick random weights.
    $\mathbf{v}_i(t+1) = w\mathbf{v}_i(t) + c_1\mathbf{r}_1(t) \otimes (\mathbf{y}_i(t) - \mathbf{x}_i(t)) + c_2\mathbf{r}_2(t) \otimes (\hat{\mathbf{y}}(t) - \mathbf{x}_i(t))$ ▷ Update the particle's velocity.
    $\mathbf{x}_i(t+1) \leftarrow \mathbf{x}_i(t) + \mathbf{v}_i(t+1)$ ▷ Update the particle's position.
    $\mathbf{y}_i(t+1) \leftarrow \mathbf{y}_i(t)$
    $\hat{\mathbf{y}}(t+1) \leftarrow \hat{\mathbf{y}}(t)$
    if $f(\mathbf{x}_i(t+1)) < f(\mathbf{y}_i(t+1))$ **then**
      $\mathbf{y}_i(t+1) \leftarrow \mathbf{x}_i(t+1)$ ▷ Update the particle's best known position.
      if $f(\mathbf{y}_i(t+1)) < f(\hat{\mathbf{y}}(t+1))$ **then**
        $\hat{\mathbf{y}}(t+1) \leftarrow \mathbf{y}_i(t+1)$ ▷ Update the swarm's best known position.
      end if
    end if
  end for
  $t \leftarrow t+1$
end while

Algorithm 2 Canonical Particle Swarm Optimization [1]

for $i \leftarrow 1$ to N **do** ▷ Create and initialize a swarm $\Omega(0)$, of N particles.
  $\mathbf{x}_i(0) \sim \text{Unif}(\mathbf{b}_{\text{lo}}, \mathbf{b}_{\text{up}})$ ▷ Initialize the particle's position with a uniformly distributed random vector.
  $\mathbf{y}_i(0) \leftarrow \mathbf{x}_i(0)$ ▷ Initialize personal best position.
  $\hat{\mathbf{y}}_i(0) \leftarrow \mathbf{x}_i(0)$ ▷ Initialize neighborhood best position.
  $\mathbf{v}_i(0) \leftarrow \mathbf{0}$ ▷ Initialize particle's velocity.
end for
$t \leftarrow 0$
while a termination criterion is not met **do**
  for $i \leftarrow 1$ to N **do**
    if $f(\mathbf{x}_i(t)) < f(\mathbf{y}_i(t))$ **then**
      $\mathbf{y}_i(t) \leftarrow \mathbf{x}_i(t)$ ▷ Update personal best position.
    end if
    for all particles $\hat{i}$ with particle i in their neighborhood **do**
      if $f(\mathbf{y}_i(t)) < f\left(\hat{\mathbf{y}}_{\hat{i}}(t)\right)$ **then**
        $\hat{\mathbf{y}}_{\hat{i}}(t) \leftarrow \mathbf{y}_i(t)$ ▷ Update neighborhood best position.
      end if
    end for
  end for
  for $i \leftarrow 1$ to N **do**
    $\mathbf{r}_{i,1}(t), \mathbf{r}_{i,2}(t) \leftarrow \text{Unif}(\mathbf{0}, \mathbf{1})$
    $\mathbf{v}_i(t+1) = w\mathbf{v}_i(t) + c_1\mathbf{r}_{i,1}(t) \otimes (\mathbf{y}_i(t) - \mathbf{x}_i(t)) + c_2\mathbf{r}_{i,2}(t) \otimes (\hat{\mathbf{y}}_i(t) - \mathbf{x}_i(t))$ ▷ Update particle's velocity.
    $\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1)$ ▷ Update particle's position.
  end for
  $t \leftarrow t+1$
end while

3.2.1 Convergence

The results summarized below are adopted from [2].

Assumption 1: $\mathbf{y}_i(t)$, $\hat{\mathbf{y}}_i(t)$, $w(t)$, $\boldsymbol{\phi}_{i,1}(t) = c_1(t)\mathbf{r}_{i,1}(t)$ and $\boldsymbol{\phi}_{i,2}(t) = c_2(t)\mathbf{r}_{i,2}(t)$ are sampled from arbitrary random variables $\mathbf{p}$, $\mathbf{g}$, w, $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$, respectively, with given expectation and variance for all t.

For a 1D search space, under the above assumption, the canonical PSO algorithm can be represented by the stochastic recursion

$$x(t+1) = lx(t) - wx(t-1) + \phi_1 p + \phi_2 g, \tag{1}$$

where $l = 1 + w - \phi_1 - \phi_2$, p, g, ϕ_1, ϕ_2 and w are random variables with given expected values (μ) and standard deviations (σ) generated at each iteration. Under Assumption 1, the following results were proven:

- Whether the variance of the position of a particle is convergent is independent of how p and g are updated.
- The necessary and sufficient conditions for the convergence of the expectation of particle positions under Assumption 1 are

  $$-1 < \mu_w < 1 \quad \text{and} \quad 0 < \mu_{\phi_1} + \mu_{\phi_2} < 2\left(\mu_w + 1\right), \tag{2}$$

  and the fixed point of the expectation is given by

  $$E_x := \lim_{t\to\infty} \mathrm{E}\left[x(t)\right] = \frac{\mu_{\phi_1}\mu_p + \mu_{\phi_2}\mu_g}{\mu_{\phi_1} + \mu_{\phi_2}}. \tag{3}$$

- If all of the conditions

  i) $-1 < \mu_w < 1$,

  ii) $0 < \mu_{\phi_1} + \mu_{\phi_2} < 2\left(\mu_w + 1\right)$,

  iii) $k_2 < 0$,

  with

  $$k_1 = \left(\mu_{\phi_1} + \mu_{\phi_2}\right)^2,$$
  $$k_2 = k_1(1 - \mu_w) + 2\left(\mu_{\phi_1} + \mu_{\phi_2}\right)\left(\mu_w^2 + \sigma_w^2 - 1\right) + \left(\sigma_{\phi_1}^2 + \sigma_{\phi_2}^2\right)\left(\mu_w + 1\right),$$

  are satisfied then the variance of the particle's position converges and the fixed point of the variance can be computed by

  $$V_x := \lim_{t\to\infty} \mathrm{Var}[x(t)] = -\frac{k_3 + k_4}{k_1 k_2}\left(\mu_w + 1\right), \tag{4}$$

  with

  $$k_3 = k_1\left(\mu_{\phi_1}^2\sigma_p^2 + \mu_{\phi_2}^2\sigma_g^2 + \sigma_{\phi_1}^2\sigma_p^2 + \sigma_{\phi_2}^2\sigma_g^2\right), \tag{5}$$
  $$k_4 = \left(\mu_{\phi_1}^2\sigma_{\phi_2}^2 + \mu_{\phi_2}^2\sigma_{\phi_1}^2\right)\left(\mu_g - \mu_p\right)^2. \tag{6}$$

- It was experimentally shown [2] that the above conditions are also sufficient for the convergence of the variance.

3.2.2 Movement patterns

Assumption 2: Parameter w is a constant: $\mu_w = w$ and $\sigma_w = 0$. We re-parameterize the uniform distributions as $\mu_{\phi_1} = c/2$, $\mu_{\phi_2} = \alpha\mu_{\phi_1}$, consequently $\sigma_{\phi_1} = c/\sqrt{12}$ and $\sigma_{\phi_2} = \alpha\sigma_{\phi_1}$.

Under the above assumption, the conditions for convergence of expectation and variance of particle positions simplify to

i) $-1 < w < 1$,

ii) $0 < (1+\alpha)c < 4\left(w+1\right)$,

iii) $c(w^2 - 1)(1+\alpha) - \dfrac{c^2}{6}(-2 + w + w\alpha(3+\alpha) - \alpha(3+2\alpha)) < 0$, It can be shown that if iii) is satisfied then i) and ii) are also satisfied.

Correlation between subsequent steps:

Based on (1) and Assumption 2, the Pearson correlation between $x(t+1)$ and $x(t-i+1)$ at the equilibrium point of (1) for any $i \geq 0$ is calculated as

$$\rho_i = \begin{cases} 1 & i = 0, \\ 1 - \dfrac{(1+\alpha)c}{2+2w} & i = 1, \\ (1 + w - (1+\alpha)c/2)\,\rho_{i-1} - w\rho_{i-2} & i > 1. \end{cases} \tag{7}$$

This autocorrelation function captures the dependency between positions of a particle in different iterations.

Expected distance a particle moves at each step:

The expected movement distance for a particle at iteration $t+1$ is formulated as

$$\mathrm{E}\left[(x(t+1) - x(t))^2\right] = \ldots = 2V_x(1 - \rho_1) \tag{8}$$

For derivation, see [2]. For definition of V_x, see (4).

Expected search range:

The expectation of the distance a particle moves away from its fixed point E_x. This expectation can be computed as

$$\mathrm{E}\left[(x(t) - E_x)^2\right] \underset{t\to\infty}{\longrightarrow} V_x. \tag{9}$$

Consequently, V_x is a measure of the range a particle covers close to its fixed point.

Focus of the search:

We measure the search concentration around a point o as $(\mathrm{E}[x(t)] - \mathrm{E}[o])^2$, i.e., the distance of a particle's position in expectation from point o. We measure the focus of the search F as

$$\frac{(\mathrm{E}[x(t)] - \mathrm{E}[p])^2}{(\mathrm{E}[x(t)] - \mathrm{E}[g])^2} \underset{t\to\infty}{\longrightarrow} \frac{(E_x - \mu_p)^2}{(E_x - \mu_g)^2} = \ldots = \left(\frac{\mu_{\phi_2}}{\mu_{\phi_1}}\right)^2 = \alpha^2 := F. \tag{10}$$

Consequently, F measures the ratio between search concentration around the personal best and global best positions of a particle. When $F > 1$ the search is more concentrated around the global best, in contrast with $F < 1$ when the search is more concentrated on the local best.

3.2.3 Fixed point of variance

The variance of the position of a particle is

$$\mathrm{Var}[x(t+1)] = \mathrm{E}\left[x(t+1)^2\right] - \mathrm{E}[x(t+1)]^2, \tag{11}$$

where under Assumption 1 (see (1))

$$\mathrm{E}\left[x(t+1)^2\right] = \mathrm{E}\left[l^2 x(t)^2\right] + \mathrm{E}\left[w^2 x(t-1)^2\right] + \mathrm{E}\left[P^2\right] - 2\mathrm{E}[lwx(t)x(t-1)] - 2\mathrm{E}[wPx(t-1)] + 2\mathrm{E}[lPx(t)] \tag{12}$$

with $P = \phi_1 p + \phi_2 g$. Since w, ϕ_1, ϕ_2, p, g and $x(t)$ are independent, we have

$$\begin{aligned} \mathrm{E}\left[x(t+1)^2\right] = \mathrm{E}\left[l^2\right]\mathrm{E}\left[x(t)^2\right] + \mathrm{E}\left[w^2\right]\mathrm{E}\left[x(t-1)^2\right] + \mathrm{E}\left[P^2\right] \\ - 2\mathrm{E}[lw]\,\mathrm{E}[x(t)x(t-1)] - 2\mathrm{E}[wP]\,\mathrm{E}[x(t-1)] + 2\mathrm{E}[lP]\,\mathrm{E}[x(t)]. \end{aligned} \tag{13}$$

In (11)

$$\mathrm{E}[x(t+1)] = \mathrm{E}[l]\,\mathrm{E}[x(t)] - \mathrm{E}[w]\,\mathrm{E}[x(t-1)] + \mathrm{E}[P]. \tag{14}$$

To establish a recursion for the expectations of terms involving particle position we compute

$$\mathrm{E}[x(t+1)x(t)] = \mathrm{E}[l]\,\mathrm{E}\left[x(t)^2\right] - \mathrm{E}[w]\,\mathrm{E}[x(t)x(t-1)] + \mathrm{E}[P]\,\mathrm{E}[x(t)]\,. \tag{15}$$

Using equations (13)–(15), the following difference equation can be established:

$$\mathbf{z}(t+1) = \mathbf{M}\mathbf{z}(t) + \mathbf{b}, \tag{16}$$

with

$$\mathbf{z}(t) = \begin{bmatrix} \mathrm{E}[x(t)] \\ \mathrm{E}[x(t-1)] \\ \mathrm{E}\left[x(t)^2\right] \\ \mathrm{E}\left[x(t-1)^2\right] \\ \mathrm{E}[x(t)x(t-1)] \end{bmatrix}, \quad \mathbf{M} = \begin{bmatrix} \mathrm{E}[l] & -\mathrm{E}[w] & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 2\mathrm{E}[lP] & -2\mathrm{E}[wP] & \mathrm{E}\left[l^2\right] & \mathrm{E}\left[w^2\right] & -2\mathrm{E}[lw] \\ 0 & 0 & 1 & 0 & 0 \\ \mathrm{E}[P] & 0 & \mathrm{E}[l] & 0 & -\mathrm{E}[w] \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathrm{E}[P] \\ 0 \\ \mathrm{E}\left[P^2\right] \\ 0 \\ 0 \end{bmatrix} \tag{17}$$

where, under Assumption 2,

$$\begin{aligned}
\mathrm{E}[l] &= 1 + w - c\,(1+\alpha)\,/2 \\
\mathrm{E}[w] &= w \\
\mathrm{E}[P] &= c\,(\mu_p + \alpha\mu_g)\,/2 \\
\mathrm{E}\left[l^2\right] &= (1+w)^2 + c^2\left(2 + 3\alpha + 2\alpha^2\right)/6 - c(1+w)(1+\alpha) \\
\mathrm{E}\left[w^2\right] &= w^2 \\
\mathrm{E}\left[P^2\right] &= \frac{c^2}{6}\left(2\left(\sigma_p^2 + \mu_p^2 + \alpha^2\left(\sigma_g^2 + \mu_g^2\right)\right) + 3\alpha\mu_p\mu_g\right) \\
\mathrm{E}[lP] &= \mu_p c\left(\frac{w+1}{2} - \frac{c}{12}(4+3\alpha)\right) + \alpha\mu_g c\left(\frac{w+1}{2} - \frac{c}{12}(3+4\alpha)\right) \\
\mathrm{E}[wP] &= w\mathrm{E}[P] \\
\mathrm{E}[lw] &= w\mathrm{E}[l]
\end{aligned}$$

The fixed point $\hat{\mathbf{z}}$ of difference equation (16) can be computed as

$$\hat{\mathbf{z}} = (\mathbf{I} - \mathbf{M})^{-1}\,\mathbf{b}, \tag{18}$$

where $\hat{\mathbf{z}} := [\hat{z}_1, \ldots, \hat{z}_5]^\intercal$. After computing $\hat{\mathbf{z}}$, one can express the fixed point of the variance as

$$V_x = \hat{z}_3 - \hat{z}_1^2 = \frac{c(w+1)\left(\alpha^2\left(2(\alpha+1)^2\sigma_g^2 + (\mu_g - \mu_p)^2\right) + 2(\alpha+1)^2\sigma_p^2\right)}{(\alpha+1)^2\left(c(-\alpha(2\alpha+3) + \alpha(\alpha+3)w + w - 2) - 6(\alpha+1)\left(w^2 - 1\right)\right)}. \tag{19}$$

This variance can be factored such that

$$V_x(\alpha, c, w, \mu_p, \sigma_p, \mu_g, \sigma_g) = \gamma(\alpha, \mu_p, \sigma_p, \mu_g, \sigma_g)\, V_{\mathrm{c}}(\alpha, c, w)\,, \tag{20}$$

where

$$\gamma = \alpha^2\left(\mu_p - \mu_g\right)^2 + 2(\alpha+1)^2\left(\sigma_p^2 + \alpha^2\sigma_g^2\right) \tag{21}$$

is a factor that depends on unknown parameters $\mu_p, \sigma_p, \mu_g, \sigma_g$ whereas

$$V_{\mathrm{c}} = \frac{c(w+1)}{c\left(m_1 w - m_2\right) - 6(\alpha+1)^3\left(w^2 - 1\right)}, \tag{22}$$

which depends on only hyperparameters c, w, and α of the PSO. Here $m_1 = (\alpha+1)^2\left(1 + 3\alpha + \alpha^2\right)$, and $m_2 = (\alpha+1)^2\left(2 + 3\alpha + 2\alpha^2\right)$ Note that for $\alpha > 0$, factor γ is strictly monotonically increasing with respect to α, irrespective of unknown parameters μ_p, σ_p, μ_g and σ_g.

3.2.4 Parameter tuning (controlling movement patterns)

Our aim is to re-parameterize the PSO algorithm such that instead of c, w, α other parameters are used that are directly related to the correlation between subsequent steps, expected search range, expected distance and the focus of the search.

Since the expected search range, and expected distance are proportional to the controlled variance V_c, we choose this as a new parameter. To capture the dependency of two subsequent steps as a parameter, we chose correlation coefficient ρ_1 as another new parameter. Finally, to account for the focus of the search we chose F as the third new parameter.

Since $\tilde{\alpha} = \pm\sqrt{F}$ can be easily expressed, only w and c are need to be determined in terms of α, ρ_1, and V_c. By solving (7) and (22), these two hyperparameters can be expressed as

$$\tilde{w} = \frac{-1 + 3V_c(1+\alpha)^4 + m_2 V_c\,(\rho_1 - 1) + \rho_1}{1 + 3V_c(1+\alpha)^4 + m_1 V_c\,(\rho_1 - 1) - \rho_1} = \frac{m_1 + m_2\rho_1 - \dfrac{1-\rho_1}{V_c}}{m_2 + m_1\rho_1 + \dfrac{1-\rho_1}{V_c}}, \tag{23}$$

$$\tilde{c} = \frac{2\,(1-\rho_1)\,(\tilde{w}+1)}{\alpha + 1}. \tag{24}$$

We employ the movement pattern adaptation strategy presented in [2] and change the new set of hyper-parameters in time as follows, using 9 parameters $(V_{c,\min}, V_{c,\max}, \rho_{1,\min}, \rho_{1,\max}, F_{\min}, F_{\max}, r_{t_1}, r_{t_2}, t_{\max})$.

$$V_c(t) = \begin{cases} V_{c,\max} & 0 < t \le t_1, \\ V_{c,\max} - \frac{t-t_1}{t_2-t_1}\left(V_{c,\max} - V_{c,\min}\right) & t_1 < t \le t_2, \\ V_{c,\min} & t_2 < t \le t_{\max}, \end{cases} \tag{25}$$

$$\rho_1(t) = \begin{cases} \rho_{1,\min} & t \le t_1, \\ \rho_{1,\min} + 2\frac{t-t_1}{t_2-t_1}\left(\rho_{1,\max} - \rho_{1,\min}\right) & t_1 < t \le (t_2+t_1)/2, \\ \rho_{1,\max} - \frac{2t-(t_1+t_2)}{t_2-t_1}\left(\rho_{1,\max} - \rho_{1,\min}\right) & (t_2+t_1)/2 < t \le t_2, \\ \rho_{1,\min} & t_2 < t \le t_{\max}, \end{cases} \tag{26}$$

$$F(t) = \begin{cases} F_{\min} & t \le t_1, \\ 1 & t_1 < t \le t_2, \\ F_{\max} & t_2 < t \le t_{\max}, \end{cases} \tag{27}$$

where $t_1 = r_{t_1} t_{\max}, t_2 = r_{t_2} t_{\max}$.

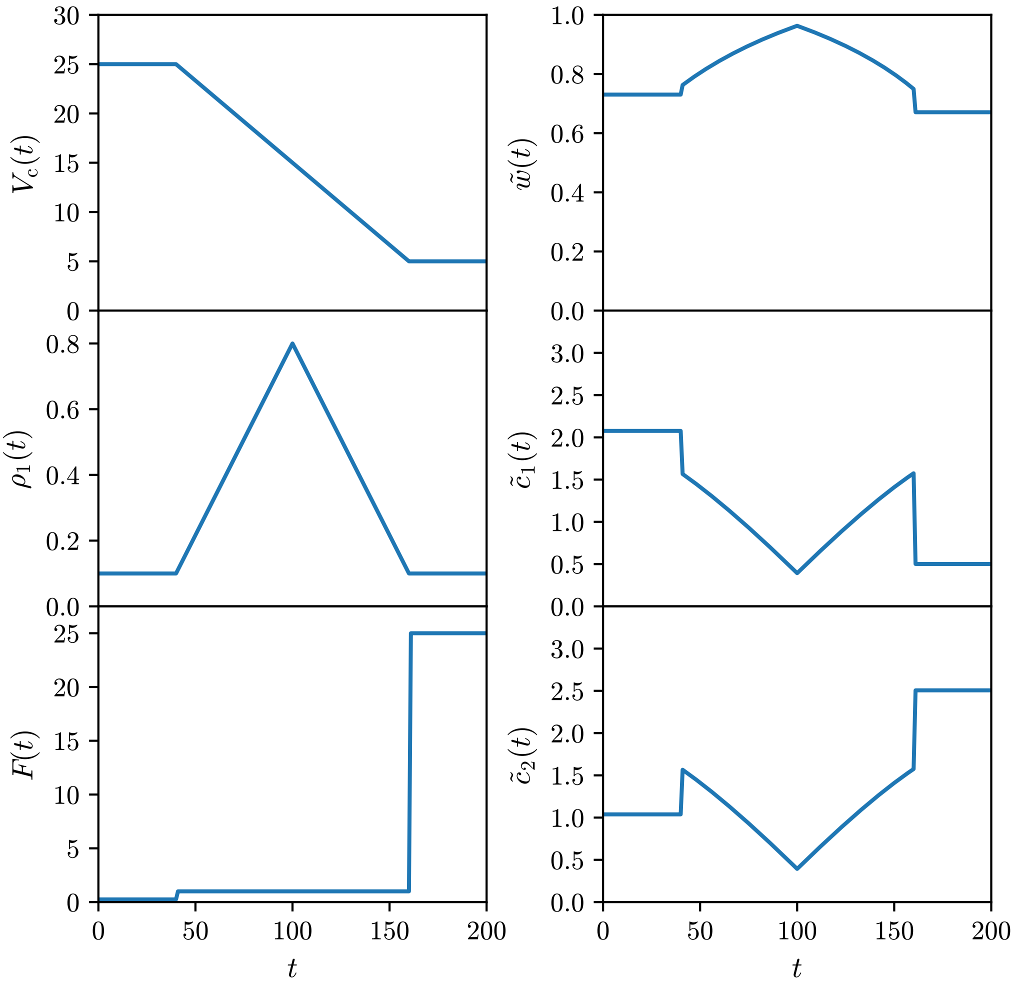

Figure 1: New (left) and original (right) set of hyper-parameters as a function of time step t. Time course of these parameters are defined by $(V_{\mathrm{c,min}}, V_{\mathrm{c,max}}, \rho_{1,\mathrm{min}}, \rho_{1,\mathrm{max}}, F_{\mathrm{min}}, F_{\mathrm{max}}, r_{t_1}, r_{t_2}, t_{\mathrm{max}}) = (5, 25, 0.1, 0.8, 0.25, 25, 0.2, 0.8, 200)$

References

[1] Christopher W. Cleghorn and Andries P. Engelbrecht. Particle swarm stability: a theoretical extension using the non-stagnate distribution assumption. *Swarm Intelligence*, 12:1–22, 2018.

[2] Mohammad Reza Bonyadi. A theoretical guideline for designing an effective adaptive particle swarm. *IEEE Transactions on Evolutionary Computation*, 24(1):57–68, 2020.