Improving Search Space
Exploration and Exploitation with
the Cross-Entropy Method and the
Evolutionary Particle Swarm Optimization

Leonel Carvalho, Vladimiro Miranda, Armando Leite da Silva,
Carolina Marcelino, Elizabeth Wanner

leonel.m.carvalho@inesctec.pt

08-08-2018
Methodological Approach

• Combination of two optimization methods
  – Cross-Entropy (CE) Method for exploration
  – Evolutionary Particle Swarm Optimization (EPSO) for exploitation
    • New recombination operator for multi-period constraints

• EPSO parameters were tuned using an iterative optimization algorithm based on a $2^2$ factorial design
  – Mutation Rate $\tau$
  – Communication Probability $P$
CE Method for Optimization

• Heuristic method proposed by Reuven Rubinstein for combinatorial and continuous non-linear optimization based on the Kullback–Leibler divergence
  – Kullback–Leibler divergence (relative entropy) is a measure of how one probability distribution diverges from a target one

• It is based on the concept that optimization problems can be defined as a rare-event estimation problem
  – The probability of locating an optimal or near optimal solution using random search is a rare-event probability
CE Method for Optimization

• Instead of random search, the **CE Method** sets an adaptive iterative importance sampling process
  – The sampling distribution (Bernoulli, Binomial, Gaussian, Multivariate Bernoulli) is modified in each iteration so that the rare-event becomes more likely to occur

• The optimization process starts by defining a sampling distribution for the variables followed by an iterated adjustment of the distribution parameters (e.g. mean, standard deviation) according to the performance of an elite subset of the samples
CE Method for Optimization

Select $\mu_0$ and $\sigma_0^2$, the number of samples per iteration $N$, the rarity parameter $\rho$, the smoothing parameter $\alpha$, $k := 0$

Do

\begin{align*}
& k := k + 1 \\
& \text{Generate} \text{ a sample of } X_1, ..., X_N \text{ from the sampling distribution } N(\mu_{k-1}, \sigma_{k-1}^2) \\
& \text{Compute } S(X_1), ..., S(X_N) \text{ and order the samples from the worst to the best performing ones, i.e. } S(X_1) < S(X_2) < ... < S(X_N) \\
& \text{Compute } \gamma_k \text{ as the } \rho^{th} \text{ quantile of the performance values and select } N^{\text{elite}} = \rho N; \text{ let } \psi \text{ be the subset from the ordered set of samples that contains all the } N^{\text{elite}} \text{ samples, i.e., the samples } S(X) < \gamma_k \\
& \text{For } j = 1 \text{ to } n \\
& \quad \mu_{kj} := \sum_{i \in \Psi} \frac{X_{ij}}{N^{\text{elite}}} \\
& \quad \sigma_{kj}^2 := \sum_{i \in \Psi} \frac{(X_{ij} - \mu_{kj})^2}{N^{\text{elite}}} \\
& \text{End For} \\
& \text{Apply smoothing} \\
& \quad \mu_k := \alpha \mu_k + (1 - \alpha) \mu_{k-1} \\
& \quad \sigma_k^2 := \alpha \sigma_k^2 + (1 - \alpha) \sigma_{k-1}^2 \\
& \text{Until } k < k^{\text{MAX}}
The mean and variance of the distribution is adjusted until there is a concentration of density at the optimum.
• **Evolutionary Particle Swarm Optimization (EPSO)** is an hybrid between **Evolutionary Strategies** and **Particle Swarm Optimization** proposed by Vladimiro Miranda in 2002
  
  – **Replication**: each individual is replicated $r$ times
  – **Mutation**: the $r$ clones have their weights $w$ mutated
  – **Recombination**: the $r+1$ individuals generate one offspring
  – **Evaluation**: each offspring has its fitness evaluated
  – **Selection**: the best particle out of the $r+1$ survives to be part of a new generation
Recombination in EPSO

• Movement Rule

\[ \mathbf{X}^{\text{new}} = \mathbf{X} + \mathbf{V}^{\text{new}} \]

\[ \mathbf{V}^{\text{new}} = w_I^* \mathbf{V} + w_M^* (\mathbf{X}_M - \mathbf{X}) + w_C^* \mathbf{P} (\mathbf{X}_G^* - \mathbf{X}) \]

– **Inertia**: movement in the same direction
– **Memory**: attraction towards the individual best solution
– **Cooperation**: attraction to a region near the global best position
EPSO Particularities

- Weights are mutated and subjected to selection
  \[ w^* = w[1 + \sigma N(0,1)] \]
  - The individuals are attracted to a region near the best solution found
  \[ X_G^* = X_G [1 + w_{GB}^* N(0,1)] \]
  - Matrix \( P \) acts as a communication barrier
  \[
  V_{\text{new}} = w_I^* V + w_M^* (X_M - X) + w_C^* P (X_G^* - X)
  \]
  \[
  P = \begin{bmatrix}
  1 & 0 & 0 \\
  0 & \ddots & 0 \\
  0 & 0 & 0
  \end{bmatrix}
  \]
  - If \( i \neq j \) then \( P_{ij} = 0 \)
  - Elseif \( U(0,1) > p \) then \( P_{ij} = 0 \)
  - Else \( P_{ij} = 1 \)
EPSO Recombination in the 2018 Competition

• **Test Bed B** consists of a **Dynamic OPF for 6 periods** that considers ramp constraints for generators (5 x 53 constraints)
  
  – Feasible solutions are difficult to obtain since the production in each period can be highly conditioned by the production in the adjacent periods

• **Idea**: to compute a new production for the current period based on the recombination of this production with the production of the next period
EPSO Recombination in the 2018 Competition

- Additional recombination rule for generators
  \[ X_{i,j}^{\text{NEW}} := X_{i,j}^{\text{NEW}} \delta + X_{i,j+1}^{\text{NEW}} (1 - \delta) \]

- 0 \leq \delta \leq 1 is random variable that follows a Beta(\(\alpha,\beta\)) pdf

- Beta(0.5,0.5) was selected
EPSO Recombination in the 2018 Competition

- In EPSO, integer variables are modeled as real variables and rounded for fitness computation.

- To force diversity, after the computation of a new position, each integer in the new position is compared with that of its predecessor and, if all integers in the two positions are equal, then one integer in the new solution is randomly selected to be increased/decreased by the value of 1 with probability 0.5.
Switch from CE Method to EPSO

When to switch from methods?
Switch from CE Method to EPSO

• Trial & error
  – Not very clever but a **practical approach** for the competition

• Track the **rate of improvement** of the best fitness over a number of iterations and switch when the rate becomes **inferior** a given **threshold**
  – A new parameter has to be defined

• **Monitor the variance of the CE Method sampling distributions**
  – The problem is that the variance can decrease very slowly for variables that can take a wide range of values without significantly affecting the fitness function
EPSO Parameter Tuning

- **Iterative method** based on $2^2$ factorial design and the **Two-way ANOVA** to guarantee the **best performance** for EPSO
  - Define maximum allowable interval for $\tau$ and $P$ (e.g. [0.2, 0.8])
  - Run a $2^2$ factorial design (**4 experiments**)
    - Perform $4 \times 31$ runs of EPSO for the 4 combinations of the limit values of $\tau$ and $P$
  - Do
    - Run **Two-way ANOVA** and **select** the variable with the **highest F-test statistic/the lowest p-value**
    - Compute the **main effect** to determine which limit should be updated
    - Update the limit to the **central value of the current interval** (e.g. if the output decreases with the variable increase, then set the lower limit to the central value)
    - Run a $2^2$ factorial design with the updated limits (**+2 experiments**)
  - **While** there is evidence that $\tau$ and $P$ affect the output ($p$-value less than 0.05) or if the difference between interval limits is greater than a value (e.g. 0.1)
Parameter Tuning for Minimization

Two-way ANOVA results

<table>
<thead>
<tr>
<th>Factor</th>
<th>F-test Statistic</th>
<th>P-value</th>
<th>Main Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Iteration</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \tau )</td>
<td>3.474</td>
<td>0.064</td>
<td>-350.73</td>
</tr>
<tr>
<td>( P )</td>
<td>6.675</td>
<td>0.011</td>
<td><strong>486.14</strong></td>
</tr>
<tr>
<td>( \tau \times P )</td>
<td>0.225</td>
<td>0.636</td>
<td>89.17</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Factor</th>
<th>F-test Statistic</th>
<th>P-value</th>
<th>Main Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>2nd Iteration</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \tau )</td>
<td>5.469</td>
<td>0.021</td>
<td>-548.35</td>
</tr>
<tr>
<td>( P )</td>
<td>6.022</td>
<td>0.015</td>
<td><strong>575.42</strong></td>
</tr>
<tr>
<td>( \tau \times P )</td>
<td>1.496</td>
<td>0.223</td>
<td>286.79</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Factor</th>
<th>F-test Statistic</th>
<th>P-value</th>
<th>Main Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>3rd Iteration</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \tau )</td>
<td>5.508</td>
<td>0.020</td>
<td><strong>448.7</strong></td>
</tr>
<tr>
<td>( P )</td>
<td>1.448</td>
<td>0.231</td>
<td>230.01</td>
</tr>
<tr>
<td>( \tau \times P )</td>
<td>0.958</td>
<td>0.329</td>
<td>187.14</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Factor</th>
<th>F-test Statistic</th>
<th>P-value</th>
<th>Main Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>4th Iteration</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \tau )</td>
<td>2.849</td>
<td>0.093</td>
<td>-306.78</td>
</tr>
<tr>
<td>( P )</td>
<td>2.497</td>
<td>0.116</td>
<td>287.21</td>
</tr>
<tr>
<td>( \tau \times P )</td>
<td>1.807</td>
<td>0.181</td>
<td>244.34</td>
</tr>
</tbody>
</table>

Greater than the threshold level of 0.05

Box-plots for the final range of \( \tau \) and \( P \)

<table>
<thead>
<tr>
<th>Factor</th>
<th>Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \tau ) (low)</td>
<td>1st 2nd 3rd 4th</td>
</tr>
<tr>
<td>( P ) (low)</td>
<td>0.2 0.2 0.2 0.5</td>
</tr>
<tr>
<td>( \tau ) (high)</td>
<td>0.8 0.8 0.8 0.8</td>
</tr>
<tr>
<td>( P ) (high)</td>
<td>0.2 0.5 0.35 0.35</td>
</tr>
</tbody>
</table>
Results

- Test Bed A – Stochastic OPF: 12 independent runs

<table>
<thead>
<tr>
<th>Fitness</th>
<th>Case 1</th>
<th>Case 2</th>
<th>Case 3</th>
<th>Case 4</th>
<th>Case 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best</td>
<td>80,732.46</td>
<td>67,709.06</td>
<td>55,245.86</td>
<td>84,382.21</td>
<td>71,044.22</td>
</tr>
<tr>
<td>Worst</td>
<td>81,547.19</td>
<td>68,923.91</td>
<td>56,683.60</td>
<td>84,880.76</td>
<td>71,128.74</td>
</tr>
<tr>
<td>Mean</td>
<td>81,077.07</td>
<td>68,473.43</td>
<td>55,935.62</td>
<td>84,442.94</td>
<td>71,065.91</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>240.41</td>
<td>306.05</td>
<td>418.30</td>
<td>138.71</td>
<td>22.97</td>
</tr>
</tbody>
</table>
Results

- Test Bed A – Stochastic OPF: 12 independent runs
Results

- Test Bed A – Stochastic OPF: 12 independent runs
Results

• Test Bed B – Dynamic OPF: 10 independent runs

<table>
<thead>
<tr>
<th></th>
<th>Objective</th>
<th>Fitness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best</td>
<td>753,071.99</td>
<td>773,193.77</td>
</tr>
<tr>
<td>Worst</td>
<td>797,029.49</td>
<td>823,684.44</td>
</tr>
<tr>
<td>Mean</td>
<td>766,841.73</td>
<td>789,719.58</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>15,175.79</td>
<td>15,618.73</td>
</tr>
</tbody>
</table>
Final Remarks

• **Combination of methods** to address different stages of the search can **greatly improve accuracy** and **robustness** of heuristic methods

• Systematic **parameter tuning** is essential to reduce the **information** required from the user

• Sometimes **tailor-made recombination rules** to deal with specific constraints of the optimization problems can lead to **better performances**
Thank you for your attention!

Leonel Carvalho
leonel.m.carvalho@inesctec.pt