SF1. COMPUTACIÓN EVOLUTIVA Y ALGORITMOS BIOINSPIRADOS

M. LOZANO, F. HERRERA

Dpto. Ciencias de la Computación e I.A.
Universidad de Granada
18071 - ESPAÑA
A Snapshot on the use of Evolutionary Algorithms for Parameter Optimization: Milestones and Current Trends

Francisco Herrera
Research Group on Soft Computing and Information Intelligent Systems (SCI$^2$S)
http://sci2s.ugr.es
Dept. of Computer Science and A.I.
University of Granada, Spain
Email: herrera@decsai.ugr.es
http://decsai.ugr.es/~herrera
Evolutionary Parameter Optimization: Introduction

Evolutionary Algorithms...

- Are function optimizers
- Inspired by natural evolution
- Population of individuals
- Are robust, hence preferred for real world problems
- Have little theory to explain how and why they work
- They come with various flavours
Evolutionary Parameter Optimization: Introduction
Evolutionary Parameter Optimization: Introduction

Evolutionary Algorithms don’t have this problem!!!
• The idea of using simulated evolution to solve engineering and design problems have been around since the 1950’s (Fogel, 2000).
  – Bremermann, 1962
  – Box, 1957
  – Friedberg, 1958

• However, it wasn’t until the early 1960’s that we began to see three influential forms of EC emerge (Back et al, 1997):
  – Evolutionary Programming (Lawrence Fogel, 1962),
  – Genetic Algorithms (Holland, 1962)
  – Evolution Strategies (Rechenberg, 1965 & Schwefel, 1968),
The designers of each of the EC techniques saw that their particular problems could be solved via simulated evolution.

- Fogel was concerned with solving programs evolution.

- **Rechenberg & Schwefel were concerned with solving parameter optimization problems.**

- Holland was concerned with developing robust adaptive systems.
Evolutionary Parameter Optimization: Introduction

We focus our attention on the problem of finding the global optimum of a function that is characterized by:

- multiple minima
- non-differentiable
- non-linear

\[ f(X_i) = D \cdot 10 + \sum_{j=1}^{D} [x_{ij}^2 - 10 \cdot \cos(2\pi x_{ij})] \]

it has many local minima and highly multimodal.
Problem Motivation

- There are a lot of applications where a scientist/engineer has to optimize a non-linear, non-differentiable function that has multiple minima.

- An example of such an application is found in the field of neural networks where one has to optimize the topology and weights of a neural network to solve a mapping problem.

- Neural networks have been extensively used in the literature to solve classification problems, regression problems, prediction problems.
Most Popular Real-Parameter Evolutionary Algorithms

- Real-coded (parameter) genetic algorithm (RCGAs)
- Evolution strategies (ES)
- Particle swarm optimization (PSO)
- Differential evolution (DE)
- Real coding memetic algorithms (RCMA)
I. Evolutionary Parameter Optimization: Introduction

II. Pioneers and outstanding work

III. Milestone: CEC’2005 Real Parameter Optimization Session and Benchmark

IV. Large Scale Optimization

V. Real-world Numerical Optimization Problems

VI. Non Rigorous Experiments: Local vs Global Comparison

VII. Current Trends

VIII. Final Comments
Pioneers and outstanding work

Binary Coding

CROSSOVER is the fundamental mechanism of genetic rearrangement for both real organisms and genetic algorithms. Chromosomes line up and then swap the portions of their genetic code beyond the crossover point.
Pioneers and outstanding work

**Binary GAs in Continuous Search Space**

**Difficulties with binary-coded EAs**

- Binary GAs make the search space discrete
- Hamming cliffs: (10000)'s neighbor (01111)
  - Gray coding isn't the solution
- Arbitrary precision impossible due to fixed-length coding
- Search restricted with variable boundaries
- Not all Holland's schemata are important
  - (1****) more important than (****1)

**Solution**: Redesign crossover which gives more importance to meaningful schemata in real space
Real Coding Genetic Algorithms

- Decision variables are coded directly, instead of using binary strings
- **Recombination and mutation** need structural changes

Recombination

\[ \Rightarrow ? \]

Mutation

\( (x_1 x_2 \ldots \ldots \ldots x_n) \Rightarrow ? \)

- Selection operator remains the same
- Simple exchanges are not adequate
Pioneers and outstanding work

Problems with real crossover: Neighbourhood and Crossover

**Crossover idea:** combining parents genotypes to get children genotypes “somewhere in between” them

**Interpretation & Generalization**

Traditional *mutation* & *crossover* have a natural interpretation in the neighbourhood structure in terms of *closeness* and *betweenness*
Pioneers and outstanding work

First Real Coding proposal: Linear/Arithmetical crossover

- **Linear Crossover**
  - From 2 parent points, 3 new points are generated:
    - \((1/2)p_1 + (1/2)p_2, (3/2)p_1 - (1/2)p_2, (-1/2)p_1+(3/2)p_2\)
    - \((1/2)p_1 + (1/2)p_2\) is the midpoint of \(p_1\) and \(p_2\)
    - The others are on the line determined by \(p_1\) and \(p_2\)
    - The best 2 of the 3 points are sent to the next generation
    - Disadvantage - Highly disrupted schemata. It is not compatible with the schema theorem described in the next slide.

[Diagram showing linear crossover]

Extended models: Arithmetical crossover (Michalewicz, 1992),
Max-Min Arithmetic operator (Herrera, Lozano, Verdegay, 1995)
Variable-wise recombination: Blend Crossover (BLX-α)

Exploration

\[ c_{\text{min}} - \alpha \cdot I \rightarrow I \]

\[ a_i \rightarrow c_1^i \rightarrow c_2^i \rightarrow b_i \]

Exploration

\[ c_{\text{max}} + \alpha \cdot I \]

Exploitation

- Uniform probability distribution within a bound controlled by \( \alpha \)
- Diversity in children proportional to that in parents
- The search is too wide if parents are distant
Pioneers and outstanding work

Real-coded Genetic Algorithms: First studies


Pioneers and outstanding work

Variable-wise recombination of Parents (RCGA first generation)

- Use a probability distribution to create offspring
- Different implementations since 1991:
  - Blend crossover (BLX-α), 1993
  - Simulated binary crossover (SBX-β), 1995
  - Fuzzy recombination (FR-d), 1995
  - Fuzzy connectives based operator (FCB), 1994

- **Main feature:** Difference between parents used to create children
  - Provides self-adaptive property

Pioneers and outstanding work

**BLX-α (Eshelman et al., 1993)**

**Fuzzy recombination (Voigt et al., 1995)**

**SBX (Deb et al., 1995)**

**Fuzzy Connectives based Operator (Herrera et al. 1994)**
Pioneers and outstanding work

Taxonomy of Crossover operators

- Discrete crossover
- Aggregation based Crossover
- Neighborhood based Crossover

Pioneers and outstanding work

Parent Center based Crossover operators

- FR (Voigt et al., 1995)
- SBX (Deb et al., 1995)
- XLM (Takahashi et al., 2001)
- PCX (Deb et al., 2002)
- vSBX (Ballester et al., 2003)
- PNX (Ballester et al., 2004)
- PBX-α (Lozano et al., 2004)

Similar behaviour than auto-adapted operators
Pioneers and outstanding work

**Vector-Wise Recombination Operators**

- Variable-wise recombination cannot capture nonlinear interactions

**Alternative:** Recombine parents as vectors (RCGA second generation)
  - Parent-centric recombination (PCX)
  - Unimodal normally-distributed crossover (UNDX)
  - Simplex crossover (SPX)

- Difference between parents is used to create offspring solutions (some models in this special issue).
Pioneers and outstanding work

Recombine parents as vectors
PCX, UNDX & SPX Operators

Pioneers and outstanding work

**Vector-Wise Recombination Operators**

- Variable-wise recombination cannot capture nonlinear interactions
- **Alternative:** Recombine parents as vectors (RCGA second generation)
  - Parent-centric recombination (PCX)
  - Unimodal normally-distributed crossover (UNDX)
  - Simplex crossover (SPX)
- Difference between parents is used to create offspring solutions (some models in this special issue).

**New algorithms (second EAs generation):** DE, PSO, CMA-ES
Evolution Strategies

Rechenberg & Schwefel (1964) were concerned with solving parameter optimization problems. Autoadaptation of parameters.

Mut: I $\rightarrow$ I

$\text{Mut} \ (x) = x' = (x_1 + z_1, \ldots, x_n + z_n)$

$z_i \sim N_i(0, \sigma^2)$

Pioneers and outstanding work

State of the art of the ES second generation: **CMA-ES**
Evolution Strategy with Covariance Matrix Adaptation (Hansen & Ostermeier, 1996)

- Selection-mutation ES is run for \( n \) iterations
- Successful steps are recorded
- They are analyzed to find uncorrelated basis directions and strengths
- Required \( O(n^3) \) computations to solve an eigenvalue problem
- Rotation invariant

---


---

Nikolaus Hansen
www.lri.fr/~hansen/
The PSO (Kennedy and Eberhart (1995)) starts from an initial population of solutions (particles) for the optimization problem. It finds new solutions by co-jointly exploring the space and exploiting the information provided by already found, good solutions.

Pioneers and outstanding work

Particle Swarm Optimization

Particles fly through the search space (biological inspiration)

Pioneers and outstanding work

Particle Swarm Optimization

- Kennedy and Eberhart, 1995
- Particles fly through the search space
- Velocity dynamically adjusted
- \[ x_i = x_i + v_i \]
- \[ v_i = v_i + c_1 \text{rand}() (p_{i, \text{best}} - x_i) + c_2 \text{rand}() (p_g - x_i) \]
- \( p_i \): best position of i-th particle
- \( p_g \): position of best particle so far
  - 1\text{st} term: momentum part (history)
  - 2\text{nd} term: cognitive part (private thinking)
  - 3\text{rd} term: social part (collaboration)
- \( c_1, c_2 \) in [0,2]
Differential Evolution

The DE approach (Storn and Price (1997)) starts from an initial population of solutions that are mutated and crossed over to eventually obtain better solutions for the optimization problem at hand.

Differential Evolution

1. Start with a pool of random solutions
2. Create a child $v$
3. $x_k$ and $v$ are recombined with $p$
4. Keep better of $y$ and $x^{(k)}$
   - Difference of parents in creating a child is important
   - A number of modifications exist

\[-v = x^{(1)} + \lambda(x^{(2)} - x^{(3)})\]
\[-y_i = \begin{cases} v_i, & \text{with a prob. } p \\ x_i^{(k)}, & \text{else} \end{cases}\]

Vector-Wise Recombination
I. Evolutionary Parameter Optimization: Introduction
II. Pioneer and outstanding work
III. Milestone: CEC’2005 Real Parameter Optimization Session and Benchmark
IV. Large Scale Global Optimization
V. Real-world Numerical Optimization Problems
VI. Non Rigorous Experiments: Local vs Global Comparison
VII. Current Trends
VIII. Final Comments
Milestone: CEC’2005 Real Parameter Optimization Session and Benchmark

Special Session on Real-Parameter Optimization.
Organizers: K. Deb and P.N. Suganathan.

Unimodal Functions
Success Performance Indices

Multimodal Functions
Solved in at least one run

Multimodal Functions
Never solved

The study was made with dimensions $D = 10$, $D = 30$, $D=50$.
The maximum number of fitness evaluations is $10,000 \cdot D$.
Each run stops when the maximal number of evaluations is achieved.

### Milestone: CEC’2005 Real Parameter Optimization Session and Benchmark

**Special Session on Real-Parameter Optimization.**


Organizers: K. Deb and P.N. Suganathan.

---

#### Unimodal Functions

**Success Performance Indices**

- 1 Sphere
- 2 Schwefel 1.2
- 3 Ellipsoid Condition 10\(^6\)
- 4 Schwefel 1.2 with Noise
- 5 Schwefel 2.6 on Bounds
- 6 Rosenbrock

**6 functions**

---

#### Multimodal Functions

**Solved in at least one run**

- 7 Griewank out Bounds
- 8 Rastrigin Separable
- 9 Rastrigin Rotated
- 10 Weierstrass
- 11 Schwefel 2.13
- 15 Hybrid Separable

**6 functions**

---

#### Multimodal Functions

**Never solved**

- 8 Ackley Condition 10\(^{-2}\)
- 13 Expanded 6&7
- 14 Expanded Schaffer F6
- 16 Hybrid Rotated
- 17 Hybrid with Noise
- 18 Hybrid F18
- 19 Hybrid Narrow
- 20 Hybrid on Bounds
- 21 Hybrid F21
- 22 Hybrid High Condition
- 23 Hybrid Non-Continuous
- 24 Hybrid F24
- 25 Hybrid out

**13 functions**
Milestone: CEC’2005 Real Parameter Optimization Session and Benchmark

- **Algorithms involved in the comparison:** (11 algorithms)
  - **BLX-GL50** (Garcia-Martinez & Lozano, 2005): Hybrid Real-Coded Genetic Algorithms with Female and Male Differentiation
  - **BLX-MA** (Molina et al., 2005): Adaptive Local Search Parameters for Real-Coded Memetic Algorithms
  - **CoEVO** (Posik, 2005): Mutation Step Co-evolution
  - **DE** (Ronkkonen et al., 2005): Differential Evolution
  - **DMS-L-PSO**: Dynamic Multi-Swarm Particle Swarm Optimizer with Local Search
  - **EDA** (Yuan & Gallagher, 2005): Estimation of Distribution Algorithm
  - **G-CMA-ES** (Auger & Hansen, 2005): A restart Covariance Matrix Adaptation Evolution Strategy with increasing population size
  - **K-PCX** (Sinha et al., 2005): A Population-based, Steady-State real-parameter optimization algorithm with parent-centric recombination operator, a polynomial mutation operator and a niched -selection operation.
  - **L-CMA-ES** (Auger & Hansen, 2005): A restart local search Covariance Matrix Adaptation Evolution Strategy
  - **L-SaDE** (Qin & Suganthan, 2005): Self-adaptive Differential Evolution algorithm with Local Search
  - **SPC-PNX** (Ballester et al., 2005): A steady-state real-parameter GA with PNX crossover operator
Milestone: CEC’2005 Real Parameter Optimization Session and Benchmark

<table>
<thead>
<tr>
<th>G-CMA-ES vs.</th>
<th>$R^+$</th>
<th>$R^-$</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLX-GL50</td>
<td>289.5</td>
<td>35.5</td>
<td>0.001</td>
</tr>
<tr>
<td>BLX-MA</td>
<td>295.5</td>
<td>29.5</td>
<td>0.001</td>
</tr>
<tr>
<td>CoEVO</td>
<td>301.0</td>
<td>24.0</td>
<td>0.000</td>
</tr>
<tr>
<td>DE</td>
<td>262.5</td>
<td>62.5</td>
<td>0.009</td>
</tr>
<tr>
<td>DMS-L-PSO</td>
<td>199.0</td>
<td>126.0</td>
<td>0.357</td>
</tr>
<tr>
<td>EDA</td>
<td>284.5</td>
<td>40.5</td>
<td>0.001</td>
</tr>
<tr>
<td>K-PCX</td>
<td>269.0</td>
<td>56.0</td>
<td>0.004</td>
</tr>
<tr>
<td>L-CMA-ES</td>
<td>273.0</td>
<td>52.0</td>
<td>0.003</td>
</tr>
<tr>
<td>L-SaDE</td>
<td>209.0</td>
<td>116.0</td>
<td>0.259</td>
</tr>
<tr>
<td>SPC-PNX</td>
<td>305.5</td>
<td>19.5</td>
<td>0.000</td>
</tr>
</tbody>
</table>

G-CMAES versus the remaining algorithms. D = 10
P-value obtained through normal approximation

Two recent algorithms with good ranking and similar statistical behaviour:

http://math.lanl.gov/~vruwt/software/

**AMALGAM - SO:** A Multi ALgorithm Genetically Adaptive Method for Single Objective Optimization. This method simultaneously merges the strengths of the Covariance Matrix Adaptation (CMA) evolution strategy, Genetic Algorithm (GA) and Particle Swarm Optimizer (PSO) for population evolution and implements a self-adaptive learning strategy to automatically tune the number of offspring these three individual algorithms are allowed to contribute during each generation.
Two recent algorithms with good ranking and similar statistical behaviour:


Figure 3: Example of LS chain. $p_{i+1}$ is the final parameter value reached by the LS algorithm when it started with a value of $p_i$. $p_0$ is the default value for the strategy parameter.

**MA-CMA-Chains:** Local search adaptation
Every time the LS algorithm is applied to refine a particular chromosome, a fixed LS intensity should be considered for it, which will be called *LS intensity stretch* (I_str). In this way, a LS chain formed throughout n_app LS applications and started from solution s_0 will return the same solution as the application of the continuous LS algorithm to s_0 employing n_app · I_str fitness function evaluations.

After the LS operation, the parameters that define the current state of the LS processing are stored along with the reached final individual (in the steady-state GA population). When this individual is latter selected to be improved, the initial values for the parameters of the LS algorithm will be directly available. For example, if we employ the Solis and Wets’ algorithm as LS algorithm, the stored strategy parameter may be the current value of the ρ parameter. For the more elaborate CMA-ES, the state of the LS operation may be defined by the covariance matrix (C), the mean of the distribution (~m), the size (σ), and some additional variables used to guide the adaptation of these parameters.
MA-CMA-Chains: Local search adaptation


1. Generate the initial population.
2. Perform the steady-state GA throughout $n_{freq}$ evaluations.
3. Build the set $S_{LS}$ with those individuals that potentially may be refined by LS.
4. Pick the best individual in $S_{LS}$ (Let’s $c_{LS}$ to be this individual).
5. if $c_{LS}$ belongs to an existing LS chain then
6. Initialise the LS operator with the LS state stored together with $c_{LS}$.
7. else
8. Initialise the LS operator with the default LS state.
9. Apply the LS algorithm to $c_{LS}$ with an LS intensity of $I_{str}$ (Let’s $c_{LS}^r$ to be the resulting individual).
10. Replace $c_{LS}$ by $c_{LS}^r$ in the steady-state GA population.
11. Store the final LS state along with $c_{LS}^r$.
12. If (not termination-condition) go to step 2.

Figure 4: Pseudocode algorithm for the proposed MACO model
MA-CMA-Chains: Local search adaptation


**MA-LSCh-CMA**

Steady-state GA.

*BLX-α.*

*Negative Assortative Mating.*

*BGA Mutation Operator.*

Standard replacement strategy

CMA-ES as Continuous LS algorithm.

**Parameter setting.** For the experiments, MA-LSCh-CMA applies BLX-α with α = 0.5. The population size is 60 individuals and the probability of updating a chromosome by mutation is 0.125. The n_ass parameter associated with the negative assortative mating is set to 3. The value of the L G ratio, r_L/G, was set to 0.5, which represents an equilibrated choice. Finally, a value of 1e-8 was assigned to the δmin LS threshold.
**MA-CMA-Chains: Local search adaptation**


---

**Figure 6:** Rankings obtained by MA-LSCh-CMA instances with different $I_{str}$ values

$I_{str} = 500$ is the best choice
Milestone: CEC’2005 Real Parameter Optimization Session and Benchmark

MA-CMA-Chains: Local search adaptation


Figure 7: Percentages of IS chains with different lengths ($D = 10$)
MA-CMA-Chains: Local search adaptation


Figure 8: Percentages of LS chains with different lengths ($D = 30$)
MA-CMA-Chains: Local search adaptation


Figure 9: Percentages of LS chains with different lengths ($D = 50$)
Milestone: CEC’2005 Real Parameter Optimization Session and Benchmark

MA-CMA-Chains: Local search adaptation


Comparison with State-of-the-Art MACOs

<table>
<thead>
<tr>
<th>D</th>
<th>R_{+} (MA-LSCh-CMA)</th>
<th>R (DEahcSPX)</th>
<th>Critical value</th>
<th>Sig. differences?</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>135</td>
<td>75</td>
<td>52</td>
<td>No</td>
</tr>
<tr>
<td>30</td>
<td>169.5</td>
<td>40.5</td>
<td>52</td>
<td>Yes</td>
</tr>
<tr>
<td>50</td>
<td>176.5</td>
<td>33.5</td>
<td>52</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 7: DEahcSPX versus MA-LSCh-CMA (Wilcoxon’s test with p-value = 0.05)

**MA-CMA-Chains: Local search adaptation**


Comparison with the Winner of the CEC2005 Competition: G-CMA-ES

<table>
<thead>
<tr>
<th>$D$</th>
<th>$R_+$ (MA-LSCh-CMA)</th>
<th>$R_-$ (G-CMA-ES)</th>
<th>Critical value (p=0.05/p=0.1)</th>
<th>Sig. dif.? (p=0.05)</th>
<th>Sig. dif.? (p=0.1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>32.5</td>
<td>177.5</td>
<td>52/60</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>30</td>
<td>139</td>
<td>71</td>
<td>52/60</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>50</td>
<td>154</td>
<td>56</td>
<td>52/60</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 8: G-CMA-ES versus MA-LSCh-CMA (Wilcoxon’s test with $p$-value = 0.05 and $p$-value=0.1)


Large Scale Global Optimization

Nowadays, the ability to tackle high-dimensional problems is crucial to many real problems (bio-computing, data mining, etc.), arising high-dimensional optimization problems as a very interesting field of research.

The ability of being scalable for high-dimensional problems becomes an essential requirement for modern optimization algorithm approaches.

G-CMA-ES presents good results with a low/medium number of variables but its drawback is associated to the scalability – More than 100 variables
Special Session & Competition on Large Scale Global Optimization at CEC 2008.

Workshop for Evolutionary Algorithms and other Metaheuristics for Continuous Optimization Problems - A Scalability Test at ISDA 2009.

Special Session & Competition on Large Scale Global Optimization at CEC 2010.

**Winner:** Algorithm: **MA-SW-Chains**

MA-SW-Chains: Memetic Algorithm Based on Local Search Chains for Large Scale Continuous Global Optimization

*D. Molina, M. Lozano, F. Herrera*

A set of 19 scalable function optimization problems were provided:


- **5 Shifted Functions:** Schwefel's Problem 2.22 (F7), Schwefel's Problem 1.2 (F8), Extended f10 (F9), Bohachevsky (F10), and Schaffer (F11). ([Description](http://www.optimize.org/cec2008/Docs/cec2008manual.pdf)) ([Source code](http://www.optimize.org/cec2008/Docs/cec2008manual.pdf)).

- **8 Hybrid Composition Functions (F12-F19):** They are non-separable functions built by combining two functions belonging to the set of functions F1-F11 ([Description](http://www.optimize.org/cec2008/Docs/cec2008manual.pdf)) ([Source code](http://www.optimize.org/cec2008/Docs/cec2008manual.pdf)).

The study was made with dimensions $D = 50$, $D = 100$, $D = 200$, $D = 500$, and $D = 1,000$. The maximum number of fitness evaluations is $5,000 \cdot D$. Each run stops when the maximal number of evaluations is achieved.
Large Scale Global Optimization

Special Issue of Soft Computing: Scalability of Evolutionary Algorithms and other Metaheuristics for Large Scale Continuous Optimization Problems Volume 15, Number 11, 2011 (7 DE approaches)

P01 - **SOUPDE**  Shuffle Or Update **Parallel Differential Evolution** for Large Scale Optimization

P02 - **DE-D^40+M^m**  Role Differentiation and Malleable Mating for **Differential Evolution**: An Analysis on Large Scale Optimisation

P03 - **GODE**  Enhanced Opposition-Based **Differential Evolution** for Solving High-Dimensional Continuous Optimization Problems

P04 - **GaDE**  Scalability of Generalized Adaptive **Differential Evolution** for Large-Scale Continuous Optimization

P05 - **jDElsCop**  Self-adaptive **Differential Evolution** Algorithm using Population Size Reduction and Three Strategies

P06 - **SaDE-MMTS**  Self-adaptive **Differential Evolution with Multi-trajectory Search** for Large Scale Optimization

P07 - **MOS**  A MOS-based Dynamic **Memetic Differential Evolution** Algorithm for Continuous Optimization A Scalability Test (best results)

P08 - **MA-SSW-Chains**  Memetic Algorithms Based on Local Search Chains for Large Scale Continuous Optimisation Problems: MA-SSW-Chains

P09 - **RPSO-vm**  Restart Particle Swarm Optimization with Velocity Modulation: A Scalability Test

P10 - **Tuned IPSOLS**  An Incremental **Particle Swarm** for Large-Scale Optimization Problems: An Example of Tuning-in-the-loop (Re)Design of Optimization Algorithms

P11 - **multi-scale PSO**  Multi-Scale **Particle Swarm Optimization** Algorithm

P12 - **EvoPROpt**  **Path Relinking** for Large Scale Global Optimization

P13 - **EM323**  EM323 : A Line Search based algorithm for solving high-dimensional continuous non-linear optimization problems

P14 – **VXQR**  VXQR: Derivative-free unconstrained optimization based on **QR factorizations**
The algorithm with best values is MOS, in the following Wilcoxon’s test we compare this one with the other algorithms,

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>MOS value</th>
<th>Other value</th>
<th>Critical value</th>
<th>p-value</th>
<th>5% error</th>
<th>Sig. differences?</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHC</td>
<td>189,5</td>
<td>0,5</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>DE</td>
<td>172</td>
<td>18</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>DE-D40+Mm</td>
<td>157</td>
<td>33</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>EM323</td>
<td>176</td>
<td>14</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>EvoPROpt</td>
<td>189,5</td>
<td>0,5</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>GADE</td>
<td>138</td>
<td>52</td>
<td>46</td>
<td></td>
<td></td>
<td>No</td>
</tr>
<tr>
<td>G-CMA-ES</td>
<td>166,5</td>
<td>23,5</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>GODE</td>
<td>167,5</td>
<td>22,5</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>IPSOLS</td>
<td>109</td>
<td>81</td>
<td>46</td>
<td></td>
<td></td>
<td>No</td>
</tr>
<tr>
<td>JDEIsco</td>
<td>143,5</td>
<td>46,5</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>MASSWChains</td>
<td>182,5</td>
<td>7,5</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>RPSOvm</td>
<td>176</td>
<td>14</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>SADEMMTS</td>
<td>132,5</td>
<td>57,5</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>SOUPDE</td>
<td>157</td>
<td>33</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>VXQR1</td>
<td>163,5</td>
<td>26,5</td>
<td>46</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>
Large Scale Global Optimization

Special Issue of Soft Computing: Scalability of Evolutionary Algorithms and other Metaheuristics for Large Scale Continuous Optimization Problems Volume 15, Number 11, 2011

The algorithm with best values is MOS, in the following Wilcoxon's test we compare this one with the other algorithms,

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>MOS value</th>
<th>Other value</th>
<th>Critical value</th>
<th>Sig. differences?</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHC</td>
<td>189,5</td>
<td>0,5</td>
<td>46</td>
<td>Yes</td>
</tr>
<tr>
<td>DE</td>
<td>176</td>
<td>14</td>
<td>46</td>
<td>Yes</td>
</tr>
<tr>
<td>DE D40+Mm</td>
<td>157</td>
<td>33</td>
<td>46</td>
<td>Yes</td>
</tr>
<tr>
<td>EvoPROpt</td>
<td>190</td>
<td>0</td>
<td>46</td>
<td>Yes</td>
</tr>
<tr>
<td>GADE</td>
<td>138</td>
<td>52</td>
<td>46</td>
<td>No</td>
</tr>
<tr>
<td>G-CMA-ES</td>
<td>170,5</td>
<td>19,5</td>
<td>46</td>
<td>Yes</td>
</tr>
<tr>
<td>GODE</td>
<td>159</td>
<td>31</td>
<td>46</td>
<td>Yes</td>
</tr>
<tr>
<td>IPSOLS</td>
<td>95</td>
<td>95</td>
<td>46</td>
<td>No</td>
</tr>
<tr>
<td>JDElscop</td>
<td>153</td>
<td>37</td>
<td>46</td>
<td>Yes</td>
</tr>
<tr>
<td>MASSWChains</td>
<td>163,5</td>
<td>26,5</td>
<td>46</td>
<td>Yes</td>
</tr>
<tr>
<td>RPSOvmm</td>
<td>178</td>
<td>18</td>
<td>46</td>
<td>Yes</td>
</tr>
<tr>
<td>SADEMM1S</td>
<td>136,5</td>
<td>55,5</td>
<td>46</td>
<td>No</td>
</tr>
<tr>
<td>SOUPDE</td>
<td>167,5</td>
<td>22,5</td>
<td>46</td>
<td>Yes</td>
</tr>
<tr>
<td>VXQRI</td>
<td>160,5</td>
<td>29,5</td>
<td>46</td>
<td>Yes</td>
</tr>
</tbody>
</table>

D = 1000
Real-world Numerical Optimization Problems


4. Special Sessions and Workshops: Problem definitions and contributions (pdf files) http://sci2s.ugr.es/EAMHCO/#SS

1. Parameter Estimation for Frequency-Modulated (FM) Sound Waves
2. Lennard-Jones Potential Problem
3. The Bifunctional Catalyst Blend Optimal Control Problem
4. Optimal Control of a Non-Linear Stirred Tank Reactor
5. Tersoff Potential Function Minimization Problem
7. Transmission Network Expansion Planning (TNEP) Problem
8. Large Scale Transmission Pricing Problem
9. Circular Antenna Array Design Problem
10. Dynamic Economic Dispatch (DED) Problem
11. Static Economic Load Dispatch (ELD) Problem
12. Hydrothermal Scheduling Problem
13. Messenger: Spacecraft Trajectory Optimization Problem
14. Cassini 2: Spacecraft Trajectory Optimization Problem

13 Algorithms participate in the Special Track
4. Special Sessions and Workshops: Problem definitions and contributions (pdf files)
http://sci2s.ugr.es/EAMHCO/#SS (9 DE approaches)

1. Algorithm: Hybrid DE-RHC
2. Algorithm: GA-MPC (GA with a New Multi-Parent Crossover)
3. Algorithm: SAMODE (Differential Evolution with Multiple Strategies)
4. Algorithm: Elite GA (Genetic Algorithm)
5. Algorithm: IADE (Adaptive Differential Evolution Algorithm)
7. Algorithm: EA-DE-MA (Hybrid EA-DE-Memetic Algorithm)
8. Algorithm: CDASA (Continuous Differential Ant-Stigmergy Algorithm)
9. Algorithm: SAPMCSBX (Modified SBX and Adaptive Mutation)
10. Algorithm: SACWIDE (Self Adaptive Cluster Based and Weed Inspired Differential Evolution)
11. Algorithm: DE-Acr (Hybrid DE Algorithm With Adaptive Crossover Operator)
12. Algorithm: EPSDE (Ensemble Differential Evolution)
13. Algorithm: CDELS (Modified Differential Evolution with Local Search)
Algorithm: GA-MPC

GA with a New Multi-Parent Crossover for Solving IEEE-CEC2011 Competition Problems

Saber M. Elsayed; Ruhul A. Sarker; Daryl L. Essam


STEP 4: For each three consecutive individuals. If $u \in [0,1] < cr$

i) Rank these three individuals from $f(x_i) \leq f(x_{i+1}) \leq f(x_{i+2})$

ii) If one of the selected individuals is the same to another, then replace one of them with a random individual from the selection pool.

iii) Calculate $\beta = N(\mu, \sigma)$

iv) Generate three offspring ($o_i$):

- $o_1 = x_1 + \beta \times (x_2 - x_3)$
- $o_2 = x_2 + \beta \times (x_3 - x_1)$
- $o_3 = x_3 + \beta \times (x_1 - x_2)$
The algorithm with best values is GA-MPC, in the following Wilcoxon's test we compare this one with the other algorithms.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>GA-MPC value</th>
<th>Other value</th>
<th>Critical value</th>
<th>Sig. differences?</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDASA</td>
<td>242.5</td>
<td>10.5</td>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>CDE-LS</td>
<td>239</td>
<td>14</td>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>DE-ACr</td>
<td>158.5</td>
<td>94.5</td>
<td>65</td>
<td>No</td>
</tr>
<tr>
<td>DE-RCH</td>
<td>242.5</td>
<td>10.5</td>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>EA-DE-MA</td>
<td>235.5</td>
<td>17.5</td>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>ED-DE</td>
<td>230.5</td>
<td>22.5</td>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>Elite GA</td>
<td>229.5</td>
<td>23.5</td>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>EPSDE</td>
<td>235</td>
<td>18</td>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>IADE</td>
<td>222.5</td>
<td>30.5</td>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>SACWIDE</td>
<td>224.5</td>
<td>28.5</td>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>SAMODE</td>
<td>202</td>
<td>51</td>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>SAPMCSBX</td>
<td>242.5</td>
<td>10.5</td>
<td>65</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Algorithm: GA-MPC:** GA with a New Multi-Parent Crossover for Solving IEEE-CEC2011 Competition Problems

*Saber M. Elsayed; Ruhul A. Sarker; Daryl L. Essam*

*Evolutionary Computation, 2011. IEEE Congress on Jun, 5-8, 2011 Page(s): 1034 - 1040*
Non Rigorous Experiments:  
Local vs Global Comparison

It is usual to find a paper with a novel proposal:

“Advanced xxx algorithm”

For example: Advanced PSO, advanced DE ....

Authors compare the new proposal “Advanced xxx algorithm” with the basic “xxx algorithm” or recent “xxx algorithms” that are far from the state of the art.

The proposal “Advanced xxx algorithm” is better than previous ones (of course) and authors claim on the “high quality of the proposal”

From the local point of view is good but ... But the proposal “Advanced xxx algorithm” is far from the state of the art (G-CMAES, MA-CMA-Chais, AMALGAM – SO)
Examples for comparison:


Non Rigorous Experiments:
Local vs Global Comparison

Table 1: Results obtained by the Wilcoxon test for algorithm G-CMA-ES (D=10)

<table>
<thead>
<tr>
<th>G-CMA-ES Vs</th>
<th>$R^+$</th>
<th>$R^-$</th>
<th>P-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frankenstein PSO</td>
<td>278.0</td>
<td>22.0</td>
<td>6.39E-5</td>
</tr>
<tr>
<td>OLPSSO Global</td>
<td>310.0</td>
<td>15.0</td>
<td>8.166E-6</td>
</tr>
<tr>
<td>SADE</td>
<td>263.0</td>
<td>37.0</td>
<td>6.498E-4</td>
</tr>
<tr>
<td>DEGL</td>
<td>325.0</td>
<td>0.0</td>
<td>5.960E-8</td>
</tr>
<tr>
<td>JADE</td>
<td>298.0</td>
<td>27.0</td>
<td>7.498E-5</td>
</tr>
</tbody>
</table>

Table 2: Results obtained by the Wilcoxon test for algorithm G-CMA-ES (D=30)

<table>
<thead>
<tr>
<th>G-CMA-ES Vs</th>
<th>$R^+$</th>
<th>$R^-$</th>
<th>P-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frankenstein PSO</td>
<td>286.5</td>
<td>38.5</td>
<td>4.030E-4</td>
</tr>
<tr>
<td>OLPSSO Global</td>
<td>325.0</td>
<td>0.0</td>
<td>5.960E-8</td>
</tr>
<tr>
<td>SADE</td>
<td>217.0</td>
<td>83.0</td>
<td>0.0564</td>
</tr>
<tr>
<td>DEGL</td>
<td>277.0</td>
<td>48.0</td>
<td>0.0013</td>
</tr>
<tr>
<td>JADE</td>
<td>216.5</td>
<td>108.5</td>
<td>0.1524</td>
</tr>
</tbody>
</table>

Table 3: Results obtained by the Wilcoxon test for algorithm G-CMA-ES (D=50)

<table>
<thead>
<tr>
<th>G-CMA-ES Vs</th>
<th>$R^+$</th>
<th>$R^-$</th>
<th>P-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frankenstein PSO</td>
<td>276.0</td>
<td>24.0</td>
<td>9.084E-5</td>
</tr>
<tr>
<td>OLPSSO Global</td>
<td>281.0</td>
<td>44.0</td>
<td>8.082E-4</td>
</tr>
<tr>
<td>SADE</td>
<td>205.0</td>
<td>120.0</td>
<td>0.2457</td>
</tr>
<tr>
<td>DEGL</td>
<td>276.0</td>
<td>49.0</td>
<td>0.0015</td>
</tr>
<tr>
<td>JADE</td>
<td>217.0</td>
<td>108.0</td>
<td>0.148</td>
</tr>
</tbody>
</table>
Non Rigorous Experiments: Local vs Global Comparison

Of course, the two following kind of studies are important:

A) To propose new advances inside of techniques (DE, PSO, ...), but authors must try to reach the state of the art.

B) New optimization frameworks, as a first idea on a new research branch, are welcome: (third generation?)

Estimation of Distribution Algorithms

Central Force Optimization
B) New optimization frameworks, as a first idea on a new research branch, are welcome:

**Artificial Bee Colony Optimization**

**Variable mesh optimization**

Now it is necessary to advance in the development of new/novel proposals inside of these frameworks, making them competitive with the state of the art.
I. Evolutionary Parameter Optimization: Introduction

II. Pioneer and outstanding work

III. Milestone: CEC’2005 Real Parameter Optimization Session and Benchmark

IV. Large Scale Optimization

V. Real-world Numerical Optimization Problems

VI. Non Rigorous Experiments: Local vs Global Comparison

VII. Current Trends

VIII. Final Comments
Current trends

There are different areas of research that focus the attention of researchers in “evolutionary parameter optimization”:

• The algorithms’ scalability: High dimensional problems

• Multi-modal problems with multiple solutions

Recent review

Real-parameter evolutionary multimodal optimization — A survey of the state-of-the-art
Swarm and Evolutionary Computation, 1:2 (2011), 71-88
Swagatam Das, Sayan Maity, Bo-Yang Qu, P.N. Suganthan
There are different areas of research that focus the attention of researchers in “evolutionary parameter optimization”:

- **Constraint optimization**

  **Recent event:** CEC10 Special Session / Competition on Evolutionary Constrained Real Parameter single objective optimization

- **Multi-objective optimization**

  **The last high quality algorithm (state of the art):** MOEA/D Homepage
  [http://dces.essex.ac.uk/staff/qzhang/webofmoead.htm](http://dces.essex.ac.uk/staff/qzhang/webofmoead.htm)


Current trends

There are different areas of research that focus the attention of researchers in “evolutionary parameter optimization”:

• **New frameworks for Evolutionary parameter optimization** and the development of advanced approaches to compete with the state of the art.

• **Memetic Algorithms as the extension of hybrid approaches** (new frameworks and local search).

**Recent high quality methods are MAs:** MA-CMA-Chains (Genetic Algorithm and CMAES as local search, standar dimension) MOS (Dynamic Memetic Differential Evolution, large scale optimization)
Many real-world problems may be formulated as optimization problems of parameters with variables in continuous domains (parameter optimization problems).

The development of high quality evolutionary algorithms (improving known or developing new algorithms) allows us to tackle a large number of real-world applications.

It is very important to understand stochastic search in continuous and high-dimensional search spaces to advance in the topic.
Website: Evolutionary Algorithms and other Metaheuristics for Continuous Optimization Problems
http://sci2s.ugr.es/EAMHCO/

This Website is devoted to a Evolutionary Algorithms and other Metaheuristics for Continuous Optimization Problems. It is maintained by M. Lozano, D. Molina, C. García-Martínez, F. Herrera following the next summary:

1. Introduction
2. Pioneer and Outstanding Contributions
3. Books and Special Issues
4. Special Sessions and Workshops
5. Large Scale Optimization Problems
6. Complementary Material: SOCO Special Issue on Large Scale Continuous Optimization Problems
7. Software
8. Slides
9. Test Functions and Results
10. Statistical Test Based Methodologies for Algorithm Comparisons
11. Future Events