University of Maribor

Faculty of Electrical Engineering
and Computer Science

Incorporating a green component into the design and development of evolutionary algorithms

Iztok Fister, Damijan Novak, Vili Podgorelec, Iztok Fister Jr.

September 14, 2024

Agenda

Motivation

- The increasing computational demands of Artificial Intelligence (AI) are contributing to a growing carbon footprint.
- With the global push toward a green economy, reducing the environmental impact of AI is critical.
- Green Artificial Intelligence (Green AI) seeks to address this by focusing on more energy-efficient algorithms and reducing computational complexity.
- This work aligns with the green transition by proposing methods that aim to reduce both time complexity and environmental impact.
- To design a straightforward, modest algorithm incorporating a green component, with an aim of running this newly developed evolutionary algorithm on limited devices.

Contributions

- Proposal of Green Evolution Strategies aimed at minimizing time complexity and computational resource usage.
- Comparison with compact differential evolution on the CEC'14 benchmark suite.
- Findings show a marginal reduction in solution quality, offset by significant reductions in time complexity.
- A promising direction for future research in low-resource AI solutions.

Compact Evolutionary Algorithms (cEAs)

- Compact Evolutionary Algorithms (cEAs) replace real populations of individuals with a probability vector $\mathrm{PV} = [\mu_i, \sigma_i]$.
- The probability vector is dynamically sampled from distributions like:
  - Normal distribution $\mathcal{N}(\mu, \sigma)$
  - Uniform distribution $\mathcal{U}(a, b)$
- The Normal distribution has a probability density function (PDF):

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (1)$$

- For the Uniform distribution, values are sampled from the interval:

$$\mu - \sigma\sqrt{3} \leq x \leq \mu + \sigma\sqrt{3} \quad (2)$$

Evolution Strategies (ES)

- **Origin**: Developed in the 1960s by Rechenberg and Schwefel for numerical optimization.
- **Key Features**:
  - Operates on real-valued representations of individuals.
  - Excellent for global optimization, avoiding local optima.
  - Self-adaptation of mutation parameters.
- **Individual Representation**:

$$\langle x_1, \ldots, x_n, \sigma_1, \ldots, \sigma_n, \alpha_1, \ldots, \alpha_k \rangle \tag{3}$$

where σ are mutation step sizes and α are rotation angles.

- **Mutation Operator**:

$$\sigma' = \sigma \cdot \exp^{\tau \cdot \mathrm{N}(0,1)}, \quad x_i' = x_i + \sigma' \cdot \mathrm{N}(0,1) \tag{4}$$

- **Selection**: Typically (μ, λ) or $(\mu + \lambda)$ for green AI, using a single individual.

Green Evolutionary Algorithms (EAs)

- **Purpose of Green EAs**:
  1. Efficient operation on regular hardware to save power and reduce carbon footprint.
  2. Ability to run complex EAs on limited hardware.
- **Focus**: Algorithms that work with small or single populations.
- **Proposed Algorithms**:
  - **eXtended compact DE (XcDE)**: Combines features of cDE and UcDE.
  - **Green ES (gES)**: Incorporates **uncorrelated mutations** with **One Step Size** and n **Step Sizes** from ES.
- Both algorithms sample from **Normal**, **Uniform**, and **Cauchy** distributions.

Extended Compact Differential Evolution (XcDE)

- XcDE extends the compact DE approach by integrating various probability distributions:
  - **Normal**
  - **Uniform**
  - **Cauchy**
- Random vector generation:

$$y_i^t = \begin{cases} \mathcal{N}(\mu_i^{(t)}, \sigma_i^{(t)}), & \text{if } PD = \text{Normal}, \\ \mathcal{U}(a_i^{(t)}, b_i^{(t)}), & \text{if } PD = \text{Uniform}, \\ \mathcal{C}(\mu_i^{(t)}, \sigma_i^{(t)}), & \text{otherwise.} \end{cases} \tag{5}$$

- Pseudo-code highlights initialization and evolutionary steps.

XcDE Pseudo-Code

- **Key Parameters:**
  - Np: Virtual population size
  - σ_0: Initial standard deviation
  - D: Problem dimension
  - PD: Probability distribution
  - a, b: Problem boundaries
- Algorithm process:
  - Initialization of population and boundaries.
  - Loop through generations: Update variables and step sizes.
  - Mutation and crossover steps guided by selected probability distributions.
- Algorithm aims to efficiently evolve solutions by updating probability vectors.

Purpose of Experimental Work

- The goal: Compare the proposed gES algorithm with XcDE in terms of:
  - **Quality of results**
  - **Time complexity**
- Three algorithms were compared:
  - UcDE
  - gES (uncorrelated mutation with one step size)
  - gES+ (uncorrelated mutation with n step sizes)
- Each algorithm was tested with three probability distributions:
  - **Normal (-N)**, **Cauchy (-C)**, and **Uniform (-U)**
- Example: gES with Normal distribution is noted as **gES-N**.

Comparative Analysis Results

Algorithm	Fri.	Nemenyi CD	Nemenyi S.
XcDE-U	4.83	[4.02,5.64]	
XcDE-C	4.26	[3.45,5.07]	
XcDE-N	3.77	[2.96,4.58]	‡
gES-U	4.42	[3.61,5.23]	†
gES-C	4.69	[3.88,5.5]	†
gES-N	5.42	[4.61,6.23]	†
gES-U+	5.43	[4.62,6.24]	†
gES-C+	6.61	[5.80,7.42]	†
gES-N+	6.71	[5.90,7.52]	†

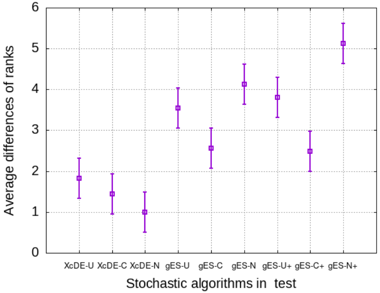

Figure: Nemenyi post-hoc statistical test results

Comparative Analysis: Time Complexity

Algorithm	Normal		Cauchy		Uniform		Total
	Tot.	Avg.	Tot.	Avg.	Tot.	Avg.	
XcDE	719.93	24.00	628.38	20.95	507.39	16.91	1,855.70
gES	527.36	17.58	504.94	16.83	446.75	14.89	1,479.05
gES+	639.80	21.33	574.44	19.15	483.50	16.12	1,697.74
Total	1,887.09	20.97	1,707.76	18.98	1,437.64	15.97	5,032.49

Conclusion

- A new way of incorporating the green component into the evolutionary algorithm was explored.
- Method reduces the algorithm's complexity and enable the algorithm to run on limited hardware.
- Comprehensive experiments performed on the CEC 14 benchmark revealed that the proposed method was less complex and thus encouraged us to widen this approach to application areas.

Future challenge: Our goal for future research is to tailor this algorithm to the problem of Numerical Association Rule Mining and apply it to the application in Smart Agriculture.

Questions and discussion