A-TOPSIS – An approach Based on TOPSIS for Ranking Evolutionary Algorithms

Summary

- Introduction – Problem description
- Multicriteria decision making
- TOPSIS
- A-TOPSIS
- Simulation Results
- Concluding

Introduction – Problem description

- In evolutionary computation, usually, the algorithms are applied several times to multiple benchmarks.
- To compare the performance between algorithms, the results are analyzed by means of statistical hypothesis tests.
- The statistical tests can detect if there are differences between the performances of the algorithms.
- The problem is if there are differences, which algorithm is the best one?

Introduction – Problem description

- Thus, it is necessary to make pairwise comparisons between the algorithms.
  - The number of tests required increases greatly with the number of algorithms being analyzed.
  - Tiresome work.
  - The probability of making a mistake increases.

- So, in this work, our goal is to modify TOPSIS to handle a decision matrix with ratings evaluated in terms of mean and standard deviations aiming selecting the best algorithm when applied to multiple benchmarks.

Multicriteria decision making

- Let us consider the decision matrix *D*, which consists of alternatives and criteria, described by:

$$
D = \begin{array}{c} \\ A_1 \\ \ldots \\ A_m \end{array}
\begin{array}{c} \begin{array}{ccc} C_1 & \ldots & C_n \end{array} \\ \begin{pmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{pmatrix} \end{array}
$$

- The weight vector $W = (w_1, w_2 \ldots, w_n)$ is composed of the individual weights $w_j\,(j = 1, \ldots, n)$ for each criterion C_j satisfying $\sum_{j=1}^{n} w_j = 1.$
- In general, the criteria are classified into two types: benefit and cost.

Multicriteria decision making

- The data of the decision matrix D come from different sources, so it is necessary to normalize it in order to transform it into a dimensionless matrix, which allow the comparison of the various criteria.

- In this work, we use the normalized decision matrix *R* :

$$R = \left[r_{ij} \right]_{mxn} \quad \rightarrow \quad r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}}, \text{with } i = 1, \ldots, m;\ j = 1, \ldots, n$$

- After normalization, we calculate the weighted normalized decision matrix:

$$P = \left[p_{ij} \right]_{mxn} \quad \rightarrow \quad p_{ij} = w_i \cdot r_{ij} \text{ with } i = 1, \ldots, m, \text{ and } j = 1, \ldots, n$$

TOPSIS

- The Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) is described in the following steps:

- **Step 1:** Identify the positive ideal solutions A^+ (benefits) and negative ideal solutions A^- (cost) as follows:

$$A^+ = (p_1^+, p_2^+, \ldots, p_m^+) \qquad \text{and} \qquad A^- = (p_1^-, p_2^-, \ldots, p_m^-)$$

- Where:

$$p_j^+ = \left(\max_i p_{ij}, j \in J_1; \min_i p_{ij}, j \in J_2 \right) \qquad \text{and} \qquad p_j^- = \left(\min_i p_{ij}, j \in J_1; \max_i p_{ij}, j \in J_2 \right)$$

Where J_1 and J_2 represent the criteria benefit and cost, respectively.

TOPSIS

- **Step 2:** Calculate the Euclidean distances from the positive ideal solution and negative ideal solution of each alternative A_i :

$$d_i^- = \sqrt{\sum_{j=1}^{n} (d_{ij}^-)^2} \quad \text{and} \quad d_i^+ = \sqrt{\sum_{j=1}^{n} (d_{ij}^+)^2}$$

where:

$$d_{ij}^+ = p_j^+ - p_{ij}, \text{with } i = 1, \ldots, m$$

$$d_{ij}^- = p_j^- - p_{ij}, \text{with } i = 1, \ldots, m.$$

- **Step 3:** Calculate the relative closeness ξ_i for each alternative A_i :

$$\xi_i = \frac{d_i^-}{d_i^+ + d_i^-}$$

A-TOPSIS

- A-TOPSIS is a novel approach based on TOPSIS for ranking algorithms.

- The decision matrix *D* consisting of alternatives and criteria is described by:

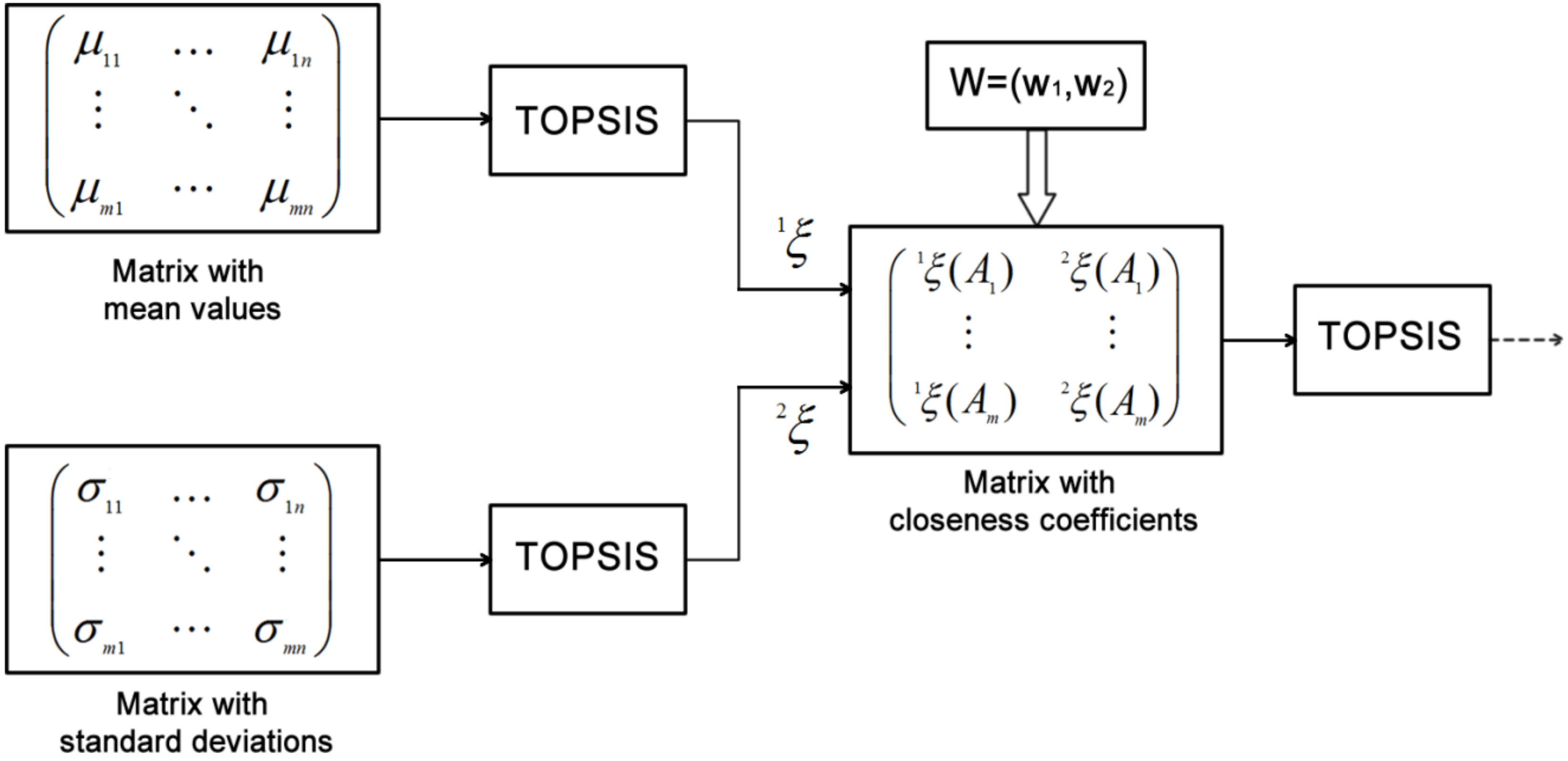
$$D = \begin{array}{c} \\ A_1 \\ \dots \\ A_m \end{array} \begin{array}{c} \begin{array}{ccc} C_1 & \dots & C_n \end{array} \\ \begin{pmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{pmatrix} \end{array} = \begin{pmatrix} (\mu_{11}, \sigma_{11}) & \cdots & (\mu_{1n}, \sigma_{1n}) \\ \vdots & \ddots & \vdots \\ (\mu_{m1}, \sigma_{m1}) & \cdots & (\mu_{mn}, \sigma_{mn}) \end{pmatrix}$$

- The problem consists of two decision matrices as given by $D = \{M_\mu, M_\sigma\}$.

$$M_\mu = \begin{pmatrix} \mu_{11} & \cdots & \mu_{1n} \\ \vdots & \ddots & \vdots \\ \mu_{m1} & \cdots & \mu_{mn} \end{pmatrix} \quad M_\sigma = \begin{pmatrix} \sigma_{11} & \cdots & \sigma_{1n} \\ \vdots & \ddots & \vdots \\ \sigma_{m1} & \cdots & \sigma_{mn} \end{pmatrix}$$

A-TOPSIS

- Illustration of the A-TOPSIS approach for ranking evolutionary algorithms in terms of mean values and standard deviations:



A-TOPSIS

- The steps of the A-TOPSIS algorithm are described as follows.
- **Step 1:** Normalize the matrices M_{μ} and M_{σ}.
- **Step 2:** Identify the positive ideal solutions $A^{+}=(p_1^{+}, p_2^{+}, ..., p_m^{+})$ (benefits) and negative ideal solutions $A^{-}=(p_1^{-}, p_2^{-}, ..., p_m^{-})$ (cost) for each matrix on the step 1.
- **Step 3:** Calculate the Euclidean distances from the positive ideal solution A^{+} and the negative ideal solution A^{-} of each alternative A_i in each matrix on the step 1.
- **Step 4:** Calculate the relative closeness ξ_i for each alternative A_i in each matrix on the step 1.

A-TOPSIS

- **Step 5:** : After calculating the vector ξ_i for both decision matrices we obtain a resulting decision matrix *G*, which is made up of the two vectors of the relative-closeness coefficients given by:

$$G = \begin{pmatrix} {}^{1}\xi(A_1) & {}^{2}\xi(A_1) \\ \vdots & \vdots \\ {}^{1}\xi(A_m) & {}^{2}\xi(A_m) \end{pmatrix}$$

- In this case, to each of the vector is assigned a weight $W = (w_1, w_2) = (w_\mu, w_\sigma)$ where represents the weight assigned to the criteria means, and standard deviations, respectively, which satisfies $w_\mu + w_\sigma = 1.$

- From this stage on our method continues by applying the standard TOPSIS to the resulting matrix in order to identify the global ranking.

Simulation results

- Let us consider the optimization with evolutionary algorithms.
- We used the G24 Benchmark Set of Dynamic Constrained Optimization Problems (DCOPs) which consists of a set of 18 benchmarks.
- The algorithms established in the literature for dynamic optimization problems are 21 different versions of Genetic Algorithms.
- As a standard procedure in evolutionary computation, the 21 different Genetic Algorithms versions have been applied to the 18 dynamic constrained minimization benchmarks and the experiment is repeated for each algorithm 50 times
- So, a statistic in terms of mean and standard deviation (stdDev) is calculated.

Simulation results

- The problem now is to determine the best algorithms in terms of effectiveness among the 21 algorithms analyzed.

- Then, we applying the A-TOPSIS

- In order to compare our approach, we present another different way to calculate the final ranking using the geometric mean between the closeness coefficients

$$\xi_G(A_i) = \sqrt{{}^1\xi(A_i) \cdot {}^2\xi(A_i)}$$

Simulation results

A-TOPSIS with $(w_1, w_2) = (0.5, 0.5)$			A-TOPSIS with $(w_1, w_2) = (1, 0)$			TOPSIS using geometric mean		
Ranking	Algorithm	ξ_i	**Ranking**	Algorithm	ξ_i	**Ranking**	Algorithm	ξ_i
1	A15	0.9531	**1**	A20	1.0000	**1**	A15	0.9026
2	A16	0.9522	**2**	A17	0.9767	**2**	A20	0.9021
3	A20	0.9510	**3**	A21	0.9630	**3**	A16	0.8947
4	A17	0.9444	**4**	A16	0.9470	**4**	A12	0.8935
5	A13	0.9363	**5**	A15	0.9386	**5**	A17	0.8925
6	A12	0.9357	**6**	A13	0.9186	**6**	A13	0.8898
7	A21	0.9269	**7**	A12	0.9156	**7**	A21	0.8809
8	A14	0.8931	**8**	A14	0.8819	**8**	A14	0.8574
9	A10	0.8800	**9**	A10	0.8624	**9**	A10	0.8501
10	A11	0.8737	**10**	A11	0.8519	**10**	A11	0.8470
11	A7	0.8317	**11**	A7	0.7848	**11**	A7	0.8272
12	A5	0.6442	**12**	A5	0.5526	**12**	A5	0.7017
13	A6	0.5701	**13**	A6	0.4766	**13**	A6	0.6493
14	A3	0.5528	**14**	A3	0.4447	**14**	A3	0.6367
15	A4	0.4774	**15**	A18	0.4411	**15**	A4	0.5843
16	A2	0.4353	**16**	A9	0.4178	**16**	A2	0.5262
17	A9	0.3566	**17**	A19	0.3830	**17**	A9	0.5049
18	A18	0.3535	**18**	A4	0.3802	**18**	A18	0.4968
19	A1	0.2944	**19**	A8	0.3600	**19**	A19	0.4528
20	A19	0.2941	**20**	A2	0.2066	**20**	A8	0.4074
21	A8	0.2570	**21**	A1	0	**21**	A1	0.3747

Concluding

- In this work, we present the A-TOPSIS to compare performance among algorithms in terms of mean values and standard deviations.
- This method allows finding the best algorithm, the second better and the worst.
- In order to illustrate the method a realistic case involving benchmarks of constrained dynamic optimization is presented. The results show the effectiveness of the method.
- In terms of computational burden, the A-TOPSIS consists of a very simple computation procedure.
- It is important to note that the TOPSIS is a well-stablished and reliable method.

Thank you for your attention

Contact:
krohling.renato@gmail.com
pacheco.comp@gmail.com

Acknowledgments:
The financial support of the of the Brazilian agency CNPq