ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-1555

A Novel Approach of Applying the Differential Evolution to Spatial Discrete Data

Vojtěch Uher and **Petr Gajdoš** and **Michal Radecký** and **Václav Snášel**[1]

Abstract. The Differential Evolution (DE) is a powerful bio-inspired algorithm searching optimal solutions. The actual DE modifications can handle the real, integer and discrete valued problems. The values of the discrete-valued variables represent the integer indices addressing the discrete samples in the ordered array. The optimization in unordered samples leads to a random search. This paper proposes a novel modification dealing with d-dimensional discrete vertices. A vertex hashing is used to strengthen the local properties of a dataset and to improve the spatial convergence of the evolution.

1 Introduction

Several DE modifications dealing with the real, integer and discrete valued problems have been published [6][4][3][1]. The discrete DE variants [1][4] iteratively optimize integer indices addressing the enumerative samples in the memory during g generations. The presented method is based on the DDE by Onwubolu and Davendra [1]. The DDE uses the integer/real value transformation and profits from the robustness of the real-coded DE [6]. The efficiency of the algorithm depends on the order of the data, because it significantly affects the convergence of the evolution [4]. This paper aims at spatial combinational problems, where a set of discrete vertices represents the solution. An analysis of the spacial data is a very common task in computer vision, robotics or pattern recognition. Such a problem can be classified as a discrete-valued problem, so that integer indices addressing the proposed vertices are iteratively optimized. However, the d-dimensional vertices cannot be simply ordered in memory. The vertices are often non-uniformly distributed, thus the linear combination of indices addressing unordered vertices leads to the random selection. This paper proposes a solution using a linearization of the spatial data with the space-filling curves (SFCs) [2][5].

2 Spatial optimization with DDE

The parameters of our modification are similar to the DDE [1]. P is the number of individuals of a population, F is the mutational factor, C is the crossover probability, g is the maximum number of generations, n is the number of individual variables, d is the dimension of the vertices, l is the total number of vertices and $f(X)$ is the objective function $f(X) : R_n \rightarrow R$, where $X = (x_1, \ldots, x_n)$ represents the vertex indices. The DDE defines so-called Forward Backward Transformation of the variable values. The values are transformed from integer to real values (Forward) before the DE strategy starts. The real values are transformed to integers (Backward) for the objective function evaluation. Our algorithm is explained on the DE/best/l/bin variant [1] and it is modified in the initialization, evaluation and mutation phases. The crossover and selection phases are traditional.

2.1 Initialization and evaluation

First, the l vertices are hashed and sorted in the memory according to the selected SFC (see e.g. [2][5]). The SFC makes the d-dimensional data partly sequenced, so that the spatially close vertices are stored in a row in the memory. Next, the DDE schematic is followed [1]. The input parameters are set and the initial population with random values within the bounds $\langle 0, l)$ is generated.

The objective function $f(X)$ evaluates the quality of the found individual. An individual consists of n variables storing the vertex indices. The variable values are regularly transformed from real to integer ones (Backward Transformation) to address the corresponding vertices for the evaluation. In our test case, the point-to-point distance function is tested, thus the distances of n vertices are computed and the total sum of the distances is used as the objective value.

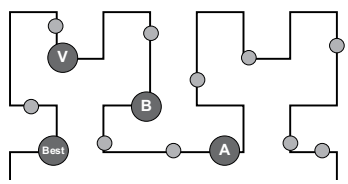

Figure 1. The random vertices are ordered by the Hilbert curve. A new curve index V is computed by the mutation operator from the three individual indices ($best$, A, B).

2.2 Mutation operator

The mutation operator computes a mutant vector as a linear combination of three different individuals: two from the current population and the best-known one (Figure 1). In this case, an individual is a set of vertex indices, which represents the potential solution of a discrete optimization. The integer values are transformed to the real ones before the mutation phase (Forward Transformation). For each index $x^G_{i,j}$ a mutant index $v^G_{i,j} = x^G_{best,j} + F \cdot (x^G_{A,j} - x^G_{B,j})$ is computed, where $i = 1, \ldots, P$, $j = 1, \ldots, n$, G is a generation counter and $i \neq A \neq B$. If $v^G_{i,j}$ is placed out of the interval $\langle 0, l)$, a random index from the interval is selected. An application of this operator to the indices addressing the unordered vertices is wasteful, because they do not provide any information about the course of the dataset. However, the SFCs systematically connect the nearby vertices, so that the vertex order represents the spatial character of the data. The SFCs better ensure that a mutant index $v^G_{i,j}$ addresses

[1] Department of Computer Science and National Supercomputing Center, VŠB-Technical University of Ostrava - Ostrava, Czech Republic, email: {vojtech.uher, petr.gajdos, michal.radecky, vaclav.snasel}@vsb.cz

a vertex that lies closer to the reference vertices with indices $x^G_{best,j}$, $x^G_{A,j}$ and $x^G_{B,j}$ (see Figure 1). The evolution thus better converges to the extremes. The Figure 2 shows a comparison of the heatmaps representing the distribution of mutation indices depending on F for the three different SFCs. It shows that the greater F leads to greater number of mutant indices placed out of the interval $\langle 0, l)$. The parameter F affects the coverage of the area. The smaller F leads to gentler sampling of the area. Thus, the specific F can improve the accuracy of the local search in the dataset. The dark areas represent the places with worse coverage. The wrong mutation indices that do not meet the constraints are replaced by random ones, thus they can additionally cover the dark areas.

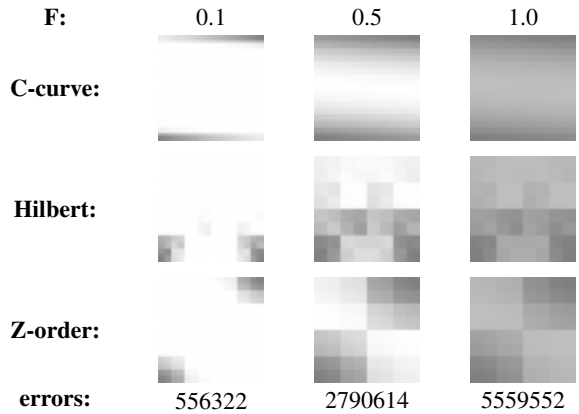

Figure 2. The heatmaps showing the distribution of the mutation indices for different SFCs (C-curve, Hilbert, Z-order) and factor F. The heatmaps were obtained by computation of the all 256^3 existing combinations of the complete level-4 SFC indices. The errors (dependent on F) represent the number of combinations leading to mutant indices placed out of bounds.

3 Experiments

The proposed method was tested on the nearest neighbors (NN) problem and five datasets (see Figure 3) to show the functioning and the improved spatial convergence of the DDE combined with the three selected SFCs (Z-odrer, Hilbert, C-curve). The C-curve was selected as a naive approach for comparison with the more sophisticated SFCs. A vertex $\vec{p}$ is randomly selected from the dataset for each measurement. The DDE algorithm searches the NN of the $\vec{p}$, so that the distance between the proposed vertices and the $\vec{p}$ is minimized. The DDE was tested with the following parameters: $n = 5$, $C = 0.95$, $F = 0.15$, $P = 30$. Stochastic search of the precise n-nearest neighbors is practically impossible. A sufficient result is searched, which is defined as a vertex with lower distance than the best analytically computed solution multiplied by the fitness rate $f_R = 2.0$. This pays for all the n vertices addressed by the individual indices. Figure 3 shows the comparison of the SFCs on the three standard Stanford datasets (buddha, bunny, dragon) and two artificial datasets with 10^6 vertices (uniform and standard normal distribution).

4 Conclusion

This paper introduced a novel variant of the Discrete Differential Evolution (DDE) which searches the optimal solutions in the spatial data. The problem with d-dimensional vertex ordering was solved by the space filling curves (SFCs) to strengthen the local properties of discrete datasets. Figure 3 proves that the DDE combined with the more sophisticated SFCs (Z-order, Hilbert) converges faster to the sufficient result than with the naive C-curve order. Thus, our modified DDE seems to be an efficient and functional method.

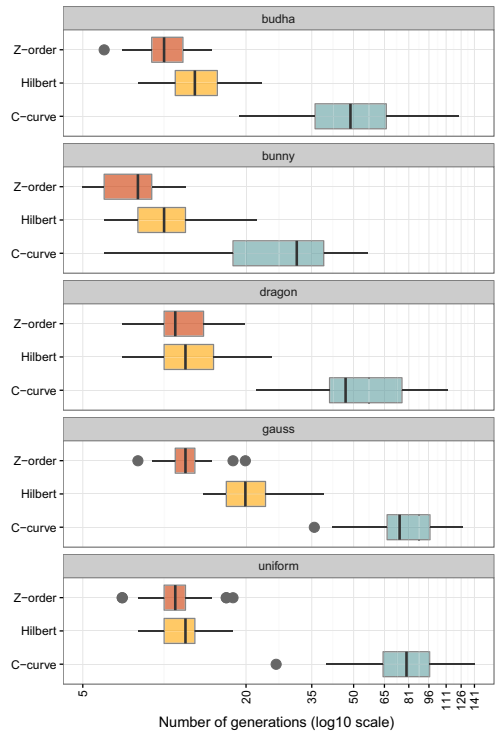

Figure 3. The box plot comparing the SFCs on the point-to-point distance minimization problems in the 3D space. The horizontal axis shows the number of generations needed to reach a sufficient result. Each measurement was done 50 times for the same parameters and datasets.

ACKNOWLEDGEMENTS

This work was supported by The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project "IT4Innovations excellence in science - LQ1602". This work was supported by SGS project, VSB-Technical University of Ostrava, under the grants no. SP2016/97 and no. SP2016/68.

REFERENCES

[1] Donald Davendra and Godfrey Onwubolu, *Differential Evolution: A Handbook for Global Permutation-Based Combinatorial Optimization*, chapter Forward Backward Transformation, 35–80, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[2] Jonathan K. Lawder and Peter J. H. King, 'Using space-filling curves for multi-dimensional indexing', in *Proceedings of the 17th British National Conferenc on Databases: Advances in Databases*, BNCOD 17, pp. 20–35, London, UK, UK, (2000). Springer-Verlag.

[3] Efrñn Mezura-Montes, Jesús Velázquez-Reyes, and Carlos A. Coello Coello, 'A comparative study of differential evolution variants for global optimization', in *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*, GECCO '06, pp. 485–492, New York, NY, USA, (2006). ACM.

[4] Godfrey C. Onwubolu and Donald Davendra, *Differential Evolution: A Handbook for Global Permutation-Based Combinatorial Optimization*, Springer Publishing Company, Incorporated, 1st edn., 2009.

[5] John Skilling, 'Programming the hilbert curve', *AIP Conference Proceedings*, **707**(1), 381–387, (2004).

[6] Rainer Storn and Kenneth Price, 'Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces', *J. of Global Optimization*, **11**(4), 341–359, (December 1997).