Session : Differential Evolution

Course Title: Computational Intelligence
Course Code: 19CSE422A

Course Leader:
Dr. Vaishali R. Kulkarni
Assistant Professor, Department of Computer Science and Engineering
Faculty of Engineering and Technology
Ramaiah University of Applied Sciences, Bengaluru
Email: vaishali.cs.et@msruas.ac.in
Tel: +91-804-906-5555 Ext:2325
Website: www.msruas.ac.in/staff/fet_cse#Vaishali

Objectives of this Session

I wish to:

1. Introduce the differential evolution (DE) algorithm
2. Provide the concepts of difference vector-based mutation operators in DE
3. Discuss the concept of recombination operator in DE
4. Provide working of DE with an example and
5. Discuss the applications of DE

Intended Outcomes of this Session

At the end of this session, the student will be able to:

1. Differentiate between GAs and DE
2. Implement basic DE for a continuous optimization problem
3. Apply varieties of mutation and crossover operators in DE implementation
4. Recommend good values of the various algorithm parameters based on the problem
5. Summarize the application potential of DE

Recommended Resources for this Session

1. Engelbrecht, A. P. (2007). *Computational intelligence: An introduction*. Chichester, England, John Wiley & Sons.

2. Konar, A. (2005). *Computational Intelligence: Principles, Techniques and Applications*. Secaucus, NJ, USA, Springer-Verlag New York, Inc.

Differential Evolution

Differential Evolution

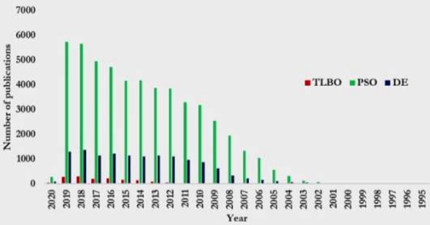
Journal of Global Optimization
December 1997, Volume 11, Issue 4, pp 341–359 | Cite as

Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces

Authors | Authors and affiliations

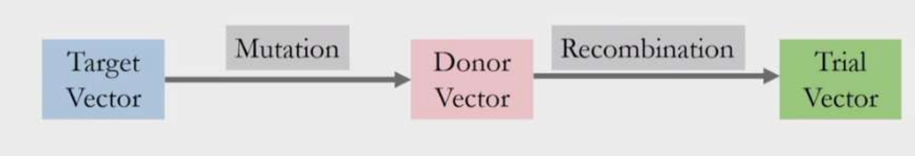
Rainer Storn, Kenneth Price

Article 2 Shares 23k Downloads 11k Citations



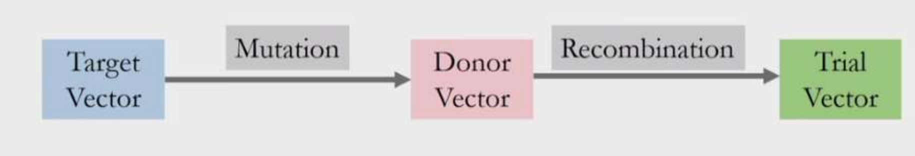
3

DE



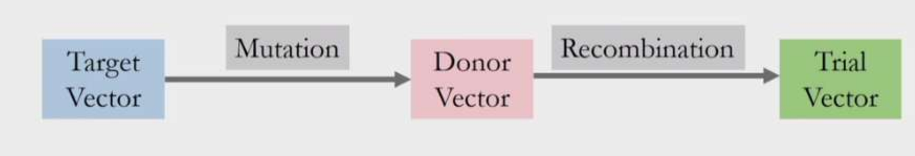
- DE is a stochastic population-based technique

DE



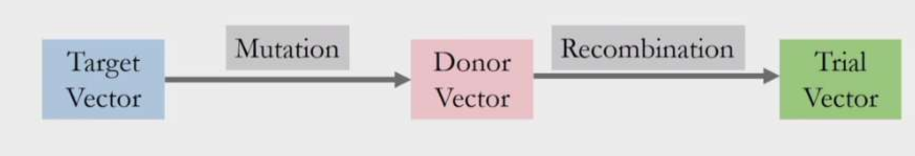
- DE is a stochastic population-based technique
- Each solution is known as Genome or Chromosome

DE



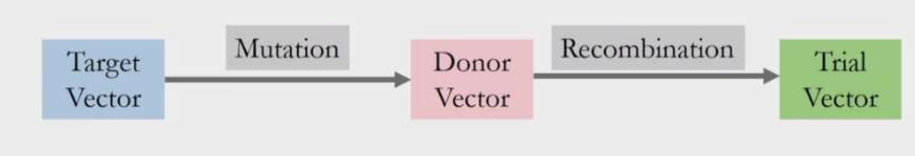
- DE is a stochastic population-based technique
- Each solution is known as Genome or Chromosome
- Each chromosome undergoes mutation followed by recombination

DE



- DE is a stochastic population-based technique
- Each solution is known as Genome or Chromosome
- Each chromosome undergoes mutation followed by recombination
- Selection of better solutions is performed only after generation of all trial vectors

DE



- DE is a stochastic population-based technique
- Each solution is known as Genome or Chromosome
- Each chromosome undergoes mutation followed by recombination
- Selection of better solutions is performed only after generation of all trial vectors
- Greedy selection is performed between target and trial vectors

Important Terms

- Target Vector
- Donor Vector
- Trial Vector

Mutation in DE

Donor vector (V) of chromosome (X_i) is created as:

$$V = X_{r1} + F(X_{r2} - X_{r3}) \tag{1}$$

where F is a scaling factor, a constant between 0 and 2
r_1, r_2 and r_3 are random solutions.
$r_1, r_2, r_3 \in 1, 2, 3, N$
$r_1 \neq r_2 \neq r_3$

Recombination: Binomial(Uniform) Crossover

It is performed to increase the diversity

Creation of trial vector can be:

$u^j = v^j$ if $r \leq p_c$ OR $j = \beta$

$u^j = x^j$ if rp_c AND $j \neq \beta$

p_C is a crossover probability

β is a randomly selected location in $\beta \in 1, 2, 3, .., D$

r is a random number between 0 and 1

u^j is j^{th} variable of trial vector

v^j is j^{th} variable of donor vector

x^j is j^{th} variable of target vector

Recombination: Exponential Crossover

- randomly choose an integer (n) between 1 and D
- Copy the nth variable from donor as nth variable of trial vector.
- For the subsequent variables, generate a random number between 0 and 1, till $r > p_c$
- if $r < p_c$, then copy the variable from donor to trial vector
- if $r > p_c$, copy the remaining variables from target to trial vectors.

General Differential Evolution Algorithm

- Population is initialized by selecting random values for the elements of each individual from the bounds defined for the problem being solved

General Differential Evolution Algorithm

- Population is initialized by selecting random values for the elements of each individual from the bounds defined for the problem being solved

- For each individual, $\mathbf{x}_i(t)$, $x_{ij}(t) \sim U(x_{\min,j}, x_{\max,j})$, where $x_{\min}$ and $x_{\max}$ define the search boundaries

General Differential Evolution Algorithm

- Population is initialized by selecting random values for the elements of each individual from the bounds defined for the problem being solved
- For each individual, $\mathbf{x}_i(t)$, $x_{ij}(t) \sim U(x_{\min,j}, x_{\max,j})$, where $x_{\min}$ and $x_{\max}$ define the search boundaries
- Many stopping conditions can be used to terminate the algorithm

General Differential Evolution Algorithm

Algorithm 3B.3. General Differential Evolution Algorithm

Set the generation counter $t = 0$;
Initialize the control parameters β and p_c;
Create and initialize the population $C(0)$ of n_s individuals;
while Stopping condition not true **do**
    for Each individual $x_i(t) \in C(t)$ **do**
        Evaluate the fitness, $f(\mathbf{x}_i(t))$;
        Create the trial vector $\mathbf{u}_i(t)$ through mutation;
        Create an offspring $\mathbf{x}'_i(t)$ through crossover;
        if $f(\mathbf{x}'_i(t))$ is better than $f(\mathbf{x}_i(t))$ **then**
            Add $\mathbf{x}'_i(t)$ to $C(t+1)$;
        else
            Add $\mathbf{x}_i(t)$ to $C(t+1)$;
        end if
    end for
end while

DE Control Parameters

- Population size n_s, the scale factor β, and the probability of recombination p_r influence the performance of DE

DE Control Parameters

- Population size n_s, the scale factor β, and the probability of recombination p_r influence the performance of DE
- **Population size:** The larger the population, the more differential vectors are available, and the more directions can be explored. But, the computational complexity increases with the size of the population

DE Control Parameters

- Population size n_s, the scale factor β, and the probability of recombination p_r influence the performance of DE
- **Population size:** The larger the population, the more differential vectors are available, and the more directions can be explored. But, the computational complexity increases with the size of the population
- Guideline: $n_s \approx 10n_x$. The mutation puts a lower bound as $n_s > 2n_v + 1$, where n_v is the number of differentials

DE Control Parameters

- Population size n_s, the scale factor β, and the probability of recombination p_r influence the performance of DE
- **Population size:** The larger the population, the more differential vectors are available, and the more directions can be explored. But, the computational complexity increases with the size of the population
- Guideline: $n_s \approx 10n_x$. The mutation puts a lower bound as $n_s > 2n_v + 1$, where n_v is the number of differentials
- **Scaling Factor:** $\beta \in (0, \infty)$ controls the amplification of the differential variations. The smaller the value of *beta*, the smaller the mutation step sizes, and the longer it will be for the algorithm to converge

DE Control Parameters

- Larger values for β facilitate exploration, but may cause the algorithm to overshoot good optima

DE Control Parameters

- Larger values for β facilitate exploration, but may cause the algorithm to overshoot good optima
- The value of β should be small enough to allow differentials to explore tight valleys, and large enough to maintain diversity

DE Control Parameters

- Larger values for β facilitate exploration, but may cause the algorithm to overshoot good optima
- The value of β should be small enough to allow differentials to explore tight valleys, and large enough to maintain diversity
- As the population size increases, the scaling factor should decrease. The more individuals in the population, the smaller the magnitude of the difference vectors, and the closer individuals will be to one another

DE Control Parameters

- Larger values for β facilitate exploration, but may cause the algorithm to overshoot good optima
- The value of β should be small enough to allow differentials to explore tight valleys, and large enough to maintain diversity
- As the population size increases, the scaling factor should decrease. The more individuals in the population, the smaller the magnitude of the difference vectors, and the closer individuals will be to one another
- Smaller step sizes can be used to explore local areas. More individuals reduce the need for large mutation step sizes

DE Control Parameters

- Larger values for β facilitate exploration, but may cause the algorithm to overshoot good optima
- The value of β should be small enough to allow differentials to explore tight valleys, and large enough to maintain diversity
- As the population size increases, the scaling factor should decrease. The more individuals in the population, the smaller the magnitude of the difference vectors, and the closer individuals will be to one another
- Smaller step sizes can be used to explore local areas. More individuals reduce the need for large mutation step sizes
- Large values for both n_s and β often result in premature convergence, and that $\beta = 0.5$ generally provides good performance

DE Control Parameters

DE Control Parameters

- **Recombination probability:** p_r has a direct influence on the diversity of DE

DE Control Parameters

- **Recombination probability:** p_r has a direct influence on the diversity of DE
- This parameter controls the number of elements of the parent, $\mathbf{x}_i(t)$, that will change

DE Control Parameters

- **Recombination probability:** p_r has a direct influence on the diversity of DE
- This parameter controls the number of elements of the parent, $\mathbf{x}_i(t)$, that will change
- The higher the p_r, the more variation is introduced in the new population, thereby increasing diversity and increasing exploration

DE Control Parameters

- **Recombination probability:** p_r has a direct influence on the diversity of DE
- This parameter controls the number of elements of the parent, $\mathbf{x}_i(t)$, that will change
- The higher the p_r, the more variation is introduced in the new population, thereby increasing diversity and increasing exploration
- Increasing p_r often results in faster convergence, while decreasing it increases search robustness

DE Control Parameters

- **Recombination probability:** p_r has a direct influence on the diversity of DE
- This parameter controls the number of elements of the parent, $\mathbf{x}_i(t)$, that will change
- The higher the p_r, the more variation is introduced in the new population, thereby increasing diversity and increasing exploration
- Increasing p_r often results in faster convergence, while decreasing it increases search robustness
- In most DE strategies, the control parameters are kept constant (DE convergence insensitive to different values of parameters)

DE Control Parameters

- **Recombination probability:** p_r has a direct influence on the diversity of DE
- This parameter controls the number of elements of the parent, $\mathbf{x}_i(t)$, that will change
- The higher the p_r, the more variation is introduced in the new population, thereby increasing diversity and increasing exploration
- Increasing p_r often results in faster convergence, while decreasing it increases search robustness
- In most DE strategies, the control parameters are kept constant (DE convergence insensitive to different values of parameters)
- Performance can be improved by finding the best values of parameters for each new problem

Variants of DE

DE is classified as DE/x/y/z. *x* is a vector to be mutated, *y* is a number of difference vector (random solutions) required for mutation and *z* is type of crossover scheme to be used (either binomial or exponential)

Mutation strategies (DE/x/y/z)

- DE: Differential Evolution
- x: Vector to be mutated
- y: number of difference vectors (random solutions) required for mutation
- z: type of crossover scheme to be used (can be either exponential or binomial crossover)

Strategy	Expression for donor vector	Minimum N_p
DE/rand/1	$V = X_{r_1} + F\left(X_{r_2} - X_{r_3}\right)$	4
DE/best/1	$V = X_{best} + F\left(X_{r_1} - X_{r_2}\right)$	3
DE/rand/2	$V = X_{r_1} + F\left(X_{r_2} - X_{r_3}\right) + F\left(X_{r_4} - X_{r_5}\right)$	6
DE/best/2	$V = X_{best} + F\left(X_{r_1} - X_{r_2}\right) + F\left(X_{r_3} - X_{r_4}\right)$	5
DE/target-to-best/1	$V = X_i + F\left(X_{best} - X_i\right) + F\left(X_{r_1} - X_{r_2}\right)$	3

Applications of DE

- Clustering
- Controllers
- Filter design
- Image analysis
- Integer-Programming
- Model selection
- NN training
- Scheduling
- System design

Session Summary

1. DE differs from GA in the sense that distance and direction information from the current population is used to guide the search process
2. Difference vectors, mutation and crossover operators decide the success of DE
3. Popular Crossover Operators: Binomial and Exponential
4. The control parameters of DE: Population size, Scaling Factor and Recombination probability
5. DE has multiple variants
6. The applications of DE: Clustering, Controllers, Filter design, Image analysis, Integer-Programming, Model selection, NN training, Scheduling, System design

Any Questions?

Thank You