TẠP CHÍ KHOA HỌC & CÔNG NGHỆ ĐẠI HỌC DUY TÂN
DTU Journal of Science and Technology 4(47) (2021) 3-7

TRƯỜNG ĐẠI HỌC DUY TÂN
DUYTAN UNIVERSITY

Solving constrained optimization tasks in civil engineering using ε-Differential Evolution developed with Visual C#. NET

Giải các bài toán tối ưu hóa có ràng buộc trong ngành xây dựng sử dụng thuật toán ε - tiến hóa vi phân được phát triển với ngôn ngữ C# .NET

Hoang Nhat Duc[a,b*], Nguyen Quoc Lam[a,b]
Hoàng Nhật Đức[a,b*], Nguyễn Quốc Lâm[a,b]

[a]*Institute of Research and Development, Duy Tan University, Da Nang, 550000, Vietnam*
[a]*Viện Nghiên cứu và Phát triển Công nghệ Cao, Đại học Duy Tân, Đà Nẵng, Việt Nam*
[b]*Faculty of Civil Engineering, Duy Tan University, Da Nang, 550000, Vietnam*
[b]*Khoa Xây dựng, Trường Đại học Duy Tân, Đà Nẵng, Việt Nam*

(Ngày nhận bài: 25/3/2021, ngày phản biện xong: 31/3/2021, ngày chấp nhận đăng: 03/7/2021)

Abstract

This research work aims at constructing an evolutionary algorithm based approach for solving complex constrained optimization tasks. The ε-Differential Evolution (ε-DE) proposed by Takahama, et al. [1], is selected as the employed evolutionary approach. The ε-DE has been proven to be capable optimizer for tackling sophisticated problems in various engineering field. In this study, this optimization method is developed in Visual C#.NET to facilitate its implementations. The newly developed software program has been tested with a design problem in civil engineering which optimizes the spacing and the cross-section parameters of a wood beam system supporting a slab formwork.

Keywords: ε-Method; differential evolution; constrained optimization; evolutionary algorithm.

Tóm tắt

Nghiên cứu của chúng tôi xây dựng một công cụ tối ưu hóa trên thuật toán tiến hóa để giải quyết các bài toán tối ưu hóa chịu các ràng buộc phức tạp. Thuật toán tiến hóa vi phân (DE) kết hợp với phương pháp ε [1] được lựa chọn để xây dựng công cụ. Thuật toán ε-DE đã được chứng minh là một công cụ có tính hiệu quả cao trong việc giải quyết các vấn đề phức tạp trong nhiều lĩnh vực. Thuật toán này đã được chúng tôi xây dựng với ngôn ngữ Visual C# .NET để tạo điều kiện thuận lợi cho việc xây dựng các ứng dụng hoặc phần mềm. Chương trình tính toán dựa trên thuật toán ε-DE đã được sử dụng để giải một bài toán tối ưu hóa khoảng cách và các thông số mặt cắt của hệ thống dầm gỗ được sử dụng để đỡ ván khuôn sàn bê tông.

Từ khóa: Phương pháp ε; tiến hóa vi phân; tối ưu hóa có ràng buộc; thuật toán tiến hóa.

1. Introduction

Constrained optimization is widely encountered in various engineering fields [2, 3]. Civil engineers frequently need to solve design problems in which an objective function is either minimized or maximized and a set of

Corresponding Author: Hoang Nhat Duc; Institute of Research and Development, Duy Tan University, Da Nang, 550000, Vietnam; Faculty of Civil Engineering, Duy Tan University, Da Nang, 550000, Vietnam
Email: hoangnhatduc@duytan.edu.vn

constraints need to be satisfied [4, 5]. Such design problems can be very challenging since they may involve multiple decision variables and a large number of constraints. These facts create considerable difficulties for conventional optimization methods. Therefore, scholars have recently relied on metaheuristic to deal with constrained optimization problems [6-13]. Takahama, et al. [1] has proposed the ε-Differential Evolution (ε-DE) as an improvement of the standard Differential Evolution (DE). In ε-DE, the selection operation is modified by taking into account the constraint violation degree of each individual. Therefore, the ε-based approach can easily handle a large number of constraints.

In this article, an optimization model based on the ε-DE metaheuristic proposed by Takahama, et al. [1] is developed with Visual C#.NET. This model aims at solving constrained optimization problems in civil engineering. The newly developed tool is used to optimize the design of a wood beam system which supports concrete slab formwork.

2. ε-Differential Evolution (ε-DE)

The ε-DE, proposed in [1], is an effective method for dealing with constrained optimization problems. The ε method is integrated into the selection operator of the standard DE [14] to take into account the constraint satisfaction of population members. Similar to the original DE, the operational flow of ε-DE also consists of four phases: (i) population initialization, (ii) mutation, (iii) cross-over, and (iv) selection. The ε-DE metaheuristic inherits the powerful mutation-crossover of the DE algorithm in which mutated and trial vectors are generated from existing population members. The main difference between the standard DE and the ε-DE metaheuristic lies in the selection operator in which the ε constraint-handling method is applied. Employing this approach, the constraint violation degree $\phi(x)$ can be defined either as the maximum value of all constraints or the sum of all constraints:

$$\phi(x) = \max\{\max_j\{0, g_j(x)\}, \max_j |h_j(x)|\} \quad (1)$$

$$\phi(x) = \sum_j \|\max_j\{0, g_j(x)\|^2 + \sum_j \|\max_j |h_j(x)\|^2 \quad (2)$$

Using the computed values of $\phi(x)$, the ε selection operation is stated as follows:

$$(f_1, \phi_1) <_\varepsilon (f_2, \phi_2) = \begin{cases} f_1 < f_2 \text{ if } \phi_1, \phi_2 \le \varepsilon \\ f_1 < f_2 \text{ if } \phi_1 = \phi_2 \\ \phi_1 < \phi_2, otherwise \end{cases} \quad (3)$$

Based on the above definition of the ε-DE metaheuristic, this work has developed the ε-Constraint Handling DE (ε-CHDE) tool for dealing with constrained optimization problems in civil engineering. ε-CHDE has been developed in Microsoft Visual Studio integrated development environment with Visual C#.NET programming language. **Fig. 1** illustrates the function interface of the ε-CHDE. The ε-selection operation is demonstrated in **Fig. 2**. Additionally, the function implementing the ε-based individual comparison is illustrated in **Fig. 3**.

```
public static List<double[,]> Optimize(GeneralObjFun ObjectiveFunction,
    GeneralObjFunWithConstraints ObjFunWithConstraints,
    GeneralConstraintFun ConstraintFunction, GeneralLB_Fun LB_Function,
    GeneralUB_Fun UB_Function, GeneralCheckContraintViolation CheckContraintViolation,
    GeneralConstraintViolationDegree ConstraintViolationDegree,
    int PopSize, int MaxGeneration)
{
```

Fig. 1. Optimization function interface

```
if (Eps_Compare(CF_U, CF_Pop_p, Phi_U, Phi_Pop_p, eps) == true)
{
    // Console.WriteLine("Pop gets updated!");
    for (int d_sel = 0; d_sel < D; d_sel++)
    {
        Pop[p, d_sel] = U[0, d_sel];
    }

    if (Eps_Compare(CF_U, Best_CF, Phi_U, Phi_Best_Sol, eps) == true)
    {
        for (int d_sel = 0; d_sel < D; d_sel++)
        {
            Best_Sol[0, d_sel] = U[0, d_sel];
            Best_CF = CF_U;
            Phi_Best_Sol = Phi_U;
        }
    }
}
```

Fig. 2. ε- based selection operator in DE

```
public static bool Eps_Compare(double CF1, double CF2, double Phi1,
        double Phi2, double eps)
{
    bool Status = false;
    // true = Sol1 is better than Sol2
    if (
        ((CF1 < CF2) && (Phi1 < eps) && (Phi2 < eps))
          || ((CF1 < CF2) && (Phi1 == Phi2))
          || (Phi1 < Phi2)
         )
    {
        Status = true;
    }

    return Status;
}
```

Fig. 3. Function implementing ε- based comparison

3. Application of the ε-CHDE

The ε-CHDE is used to design a system of wood beam supporting concrete slab formwork (demonstrated in **Fig. 4**). The decision variables include the cross-sectional parameters (the depth *d* and the width *b*) and the number of required beams *n*. Thus, the beams are required to support the operation of constructing a reinforced concrete slab structure. The objective herein is to find a set of *d*, *b*, and *n* that minimizes the material cost of the beams.

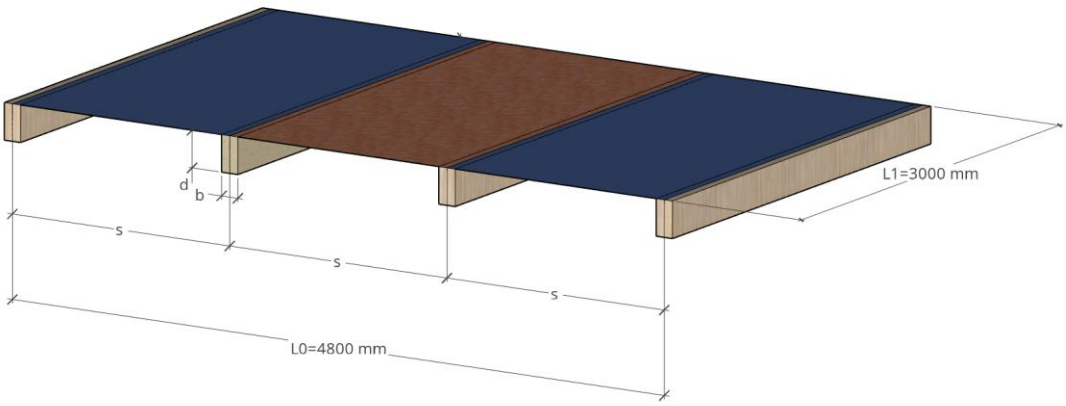

Fig. 4. Optimization problem

It is noted that the centre-to-centre spacing of the beams s is computed as:

$$s = L0/(n-1) \tag{4}$$

The load per unit length imposed on the slab formwork is given by:

$$w = s \times \lambda \tag{5}$$

where λ = 6390N/m^2 is the load caused by concrete weight that acts on the slab formwork.

The maximum bending moment in the beams caused by w is calculated as follows:

$$M_{\max} = wL1^2/8 \tag{6}$$

The maximum shear forced in the beams caused by w is calculated as follows:

$$S_{\max} = w \times L1/2 \tag{7}$$

The maximum shear stress in the beams caused by w is given by:

$$T_{\max} = 3S_{\max 2}/(2b \times d)$$

The constraints of this problem specify limitations on (i) bending stress, (ii) shear stress, and (iii) deflection of the beams [15-17]. The optimization problem is modeled as a class in Visual C# (refer to Fig. 5). Thus, this problem is mathematically formulated as follows:

$$Min. f = d \times b \times L1 \times \gamma_{Wood} \tag{8}$$

s.t.

$$\sigma_{Allow} - \frac{M_{\max}}{(bd^2/6)} \geq 0 \tag{9}$$

$$\tau_{Allow} - T_{\max} \geq 0 \tag{10}$$

$$\delta_{Allow} - 5wL1^4/(384\,\mathrm{EI}) \geq 0 \tag{11}$$

where $L1$ = 3 m is the length of a beam. Mass density of wood γ_{Wood} is 400 kg/m^3. σ_{Allow} = denotes the allowable bending stress = 10000000 N/m^2. τ_{Allow} =0.448 × 1000000 N/m^2. The modulus of elasticity of wood E = 1600000 × 0.00689476 × 1000000 N/m^2. The moment of inertia of the cross section about the centroidal axis I = bd^3/12. $\delta_{Allow} = L1/360$.

```
class COP_FloorCarriedByJoists
{

    readonly double L1 = 3; //m length of joist
    readonly double L0 = 4.8; //m  length of the other side of the slab
    readonly double Gama_Wood = 400; // mass density of wood kg/m3
    readonly double Sig_Allow_Bending_Wood = 10* 1000000; // N/m2
    readonly double Shear_Allow_Wood = 0.448 * 1000000; // N / m2;
    readonly double E_Wood = 1600000 * 0.00689476 * 1000000; //   MPa --> N/m2
    readonly double LoadOnSlabFormwork = 6390; //N/m2
    1 reference
    public double ComputeObjFun(double[,] X)[...]

    3 references
    public double[,] ComputeConstraints(double[,] X)[...]

    0 references
    public bool CheckContraintViolation(double[,] x)[...] // CheckContraintViolation

    0 references
    public double[,] ComputeConstraintViolationDegree(double[,] x)[...]

    0 references
    public double[] Get_LB()[...]

    0 references
    public double[] Get_UB()[...]

    0 references
    public double ComputeObjFunWithConstraint(double[,] x, double fmax)[...] // ComputeObjFunWithConstraint

    0 references
    public void ProblemAnalysis()[...]
}
```

Fig. 5. Optimization problem definition in Visual C# .NET

After 1000 generations and with the use of 50 searching agents, the best found design variables are: d = 0.167 m, b = 0.102 m, and n = 10. The objective function value is 205.393 kg. In addition, all of the aforementioned constraints have been satisfied.

4. Concluding remarks

This research work develops a metaheuristic based approach based on the powerful DE optimizer and the ε method for constraint handling. This metaheuristic approach has been constructed in Visual C#.NET to facilitate its implementation and development of desktop programs. The program, named as ε-CHDE, has been tested with a constrained optimization task in which a system of wood beam supporting concrete slab formwork is designed. Experimental result points out that ε-CHDE is capable of finding a good set of decision variables that lead to a low objective function and satisfaction of all the constraints. Therefore, ε-CHDE can be employed to construct various desktop programs used for solving constrained optimization problems in civil engineering.

References

[1] T. Takahama, S. Sakai, and N. Iwane, "Solving Nonlinear Constrained Optimization Problems by the ε Constrained Differential Evolution," in *2006 IEEE International Conference on Systems, Man and Cybernetics*, 2006, pp. 2322-2327.

[2] J. S. Arora, *Introduction to Optimum Design, Fourth Edition*: Academic Press, 2016.

[3] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*: Addison-Wesley Professional, ISBN 978-0201157673, 1989.

[4] S. M. Nigdeli, G. Bekdaş, and X.-S. Yang, "Metaheuristic Optimization of Reinforced Concrete Footings," *KSCE Journal of Civil Engineering,* vol. 22, pp. 4555-4563, November 01 2018.

[5] N. D. Hoang, "FR-DE Excel Solver: Differential Evolution with Deb's feasibility rules for solving constrained optimization problems in civil engineering," *DTU Journal of Science and Technology 04 (35),* 2019.

[6] X.-S. Yang, *Nature-Inspired Optimization Algorithms*: Elsevier, 2014.

[7] C. A. C. Coello, "Constraint-handling techniques used with evolutionary algorithms," presented at the Proceedings of the Genetic and Evolutionary Computation Conference Companion, Kyoto, Japan, 2018.

[8] C. A. Coello Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art," *Computer Methods in Applied Mechanics and Engineering,* vol. 191, pp. 1245-1287, 2002/01/04/ 2002.

[9] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanics and Engineering,* vol. 186, pp. 311-338, 2000/06/09/ 2000.

[10] N.-D. Hoang, "NIDE: A Novel Improved Differential Evolution for Construction Project Crashing Optimization," *Journal of Construction Engineering,* vol. 2014, p. 7, 2014.

[11] N. Đ. Hoàng, Q. L. Nguyễn, and Q. N. Phạm, "Tối ưu hóa tiến độ và chi phí cho dự án xây dựng sử dụng thuật toán tiến hóa vi phân," *Tạp Chí Khoa Học và Công Nghệ, Đại Học Duy Tân,* vol. 1, pp. 135–141, 2015.

[12] N. Đ. Hoàng and D. T. Vũ, "Tối ưu hóa kết cấu có điều kiện ràng buộc sử dụng thuật toán bầy đom đóm và các hàm phạt," *Tạp Chí Khoa Học và Công Nghệ, Đại Học Duy Tân,* vol. 2, pp. 75–84, 2015.

[13] H. Nhat-Duc and L. Cong-Hai, "Sử dụng thuật toán tiến hóa vi phân cho các bài toán tối ưu hóa kết cấu với công cụ DE-Excel solver," *DTU Journal of Science and Technology,* vol. 03, pp. 97-102, 2019.

[14] R. Storn and K. Price, "Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces," *Journal of Global Optimization,* vol. 11, pp. 341-359, December 01 1997.

[15] J. M. Gere and B. J. Goodno, *Mechanics of Materials, SI Edition*: Cengage Learning, 2013.

[16] J. L. Meriam , L. G. Kraige , and J. N. Bolton *Engineering Mechanics: Statics*: John Wiley & Sons, 2016.

[17] T. A. Philpot and J. S. Thomas, *Mechanics of materials an integrated learning system*: Wiley, 2018.