TẠP CHÍ KHOA HỌC & CÔNG NGHỆ ĐẠI HỌC DUY TÂN
DTU Journal of Science and Technology 4(47) (2021) 8-13
TRƯỜNG ĐẠI HỌC DUY TÂN
DUYTAN UNIVERSITY

Optimizing cantilever retaining wall design using feasibility rule-based evolutionary algorithm developed with Visual C# .NET

Tối ưu hóa thiết kế tường chắn đất sử dụng thuật toán tiến hóa được kết hợp quy tắc khả thi và phát triển với ngôn ngữ C# .NET

Tran Xuan Linh[a,b], Nguyen Quoc Lam[b], Hoang Nhat Duc[a,b*]
Trần Xuân Linh[a,b], Nguyễn Quốc Lâm[b], Hoàng Nhật Đức[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, Trường Đại Học Duy Tân, Đà Nẵng*
[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: 03/4/2021, ngày phản biện xong: 8/5/2021, ngày chấp nhận đăng: 20/8/2021)

Abstract

Designing cantilever retaining walls is an important task in various construction projects. This study aims at constructing an evolutionary-algorithm-based cantilever retaining wall design approach. Differential Evolution (DE) and the feasibility rule-based constraint-handling (FRBCH) method are integrated to achieve the research objective. A DE based software program incorporating FRBCH has been developed with Visual C# .NET to facilitate its implementation. A case study of cantilever retaining wall design has been used to validate the capability of the FRBCH-DE integration.

Keywords: Differential Evolution; Cantilever retaining wall design; Constrained handling; Evolutionary algorithm.

Tóm tắt

Thiết kế tường chắn đất là một nhiệm vụ quan trọng trong nhiều dự án xây dựng. Nghiên cứu của chúng tôi xây dựng một chương trình thiết kế tối ưu kết cấu này dựa trên thuật toán tiến hóa. Thuật toán tiến hóa vi phân (DE) và các quy tắc khả thi dùng cho xử lý ràng buộc (FRBCH) được kết hợp để xây dựng chương trình này. Một phần mềm dựa trên thuật toán DE và FRBCH đã được lập trình với Visual C# .NET để tăng cường tính ứng dụng của các thuật toán. Một ví dụ tính toán tường chắn đất đã được sử dụng để minh chứng khả năng của chương trình FRBCH-DE.

Từ khóa: Tiến Hóa Vi Phân; Thiết Kế Tối Ưu Tường Chắn Đất; Tối Ưu Hóa Có Ràng Buộc; Thuật Toán Tiến Hóa.

1. Introduction

Cantilever walls are widely used to support earth backfills in various construction projects [1]. The main function of these structures is to support deep excavation in basement construction, road construction, bridge abutment construction, etc. Therefore, design an optimal cantilever retaining wall is an important task in civil engineering [2-8]. It is desired to obtain an optimal shape of cantilever

[*]*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

retaining walls, which results in a low material cost and satisfaction of all safety requirements including safety against overturning/sliding and safety of bearing capacity [9].

This study aims at establishing an evolutionary algorithm based approach for optimizing the external stability of cantilever retaining wall. The Differential Evolution (DE), a powerful evolutionary algorithm, is selected in this study to achieve the aforementioned research objective. In addition, since the problem of interest involves the constraints regarding the safety of the structure against overturning/sliding and safety of bearing capacity, the feasibility rule based constraint-handling (FRBCH) method is applied [10]. The DE algorithm integrated with the FRBCH approach has been developed with Visual C#.NET in Microsoft Visual Studio by the authors. This optimization method is then applied to optimize the design of a cantilever retaining wall structure adopted from the previous work of [9].

2. Differential Evolution (DE) and the Feasibility Rule-Based Constraint-Handling (FRBCH) Method

The DE algorithm [11, 12] is a simple yet effective method for dealing with unconstrained optimization problems. The operation of DE involves four main stages: (i) population initialization, (ii) mutation, (iii) crossover, and (iv) selection. In the first stage, a set of searching agents is randomly generated within the search space. The second and the third stages, a mutation-crossover operation is used to perturb the current population members and generate new members. In the last stage, newly created trial solutions compete with existing ones to determine the members of a new DE population. DE has been demonstrated to be highly effective and efficient evolutionary algorithms which can attain good candidate solution with acceptable computational cost [13-18].

However, the original DE algorithm is designed to tackle unconstrained optimization problems, to deal with constrained optimization tasks which are ubiquitous in civil engineering, it is necessary to incorporate DE with a constraint handling method [19, 20]. This study selects the FRBCH method proposed in by Deb [10] and integrates it into the structure of the original DE algorithm. Using the FRBCH method, the objective function of the standard DE is modified as follows:

$$F(X) = \begin{cases} F(X) \ \ if \ g_j(x) \geq 0 \ \ \forall j \\ f_{\max} + \sum_{j=1}^{m} g_j(x) \end{cases} \quad (1)$$

where f_{max} is the objective function value of the worst feasible candidate. $g_j(x)$ denotes the jth constraint.

Based on the stated definition of the FRBCH based DE evolutionary algorithm, this paper has developed an optimization method, named as FRBCH-DE, used for cantilever retaining wall design. FRBCH-DE has been constructed in Microsoft Visual Studio Visual with C#.NET programming language. **Fig. 1** demonstrates the interface of FRBCH-DE. The revised objective function calculation is illustrated in **Fig. 2**. Herein, Pop_p denotes the p^{th} member of the current population. ConVio is a Boolean variable stating the constraint violation status of a member. The function 'ObjectiveFunction' is used to compute the value of the original objective function.

```
//Run optimization
string SaveLoc = "D:/ConstrainedOptimizationResult/" + AlgorithmName + '_' + ProblemName + '/';
MyMatrix.CreateFolder(SaveLoc);

var Results = Optimize(ObjectiveFunction, ObjFunWithConstraints,
    ConstraintFunction, LB_Function, UB_Function, CheckContraintViolation,
    ConstraintViolationDegree, 50, 1000);

var X = Results[0];
Console.WriteLine("Best Found Solution = "); MyMatrix.PrintMatrix(X);

var TrackBestCostFunVal = Results[1];
var TrackBestSol = Results[2];
var ConstraintValOfBestSol = Results[3];
var ComputeTime = Results[4];
```

Fig. 1 Interface of FRBCH-DE

```
// Update cost function values of pop. members
for (int p = 0; p < PopSize; p++)
{
    double[,] Pop_p = MyMatrix.ExtractMatrixRow(Pop, p);
    bool ConVio = CheckContraintViolation(Pop_p);
    if ((ConVio == false) && (Found_FeaSol == false))
    {
        Found_FeaSol = true;
        f_p = ObjectiveFunction(Pop_p);
        fmax = f_p;
    }

    if ((ConVio == false) && (Found_FeaSol == true))
    {
        f_p = ObjectiveFunction(Pop_p);
        if (f_p > fmax)
        {
            fmax = f_p;
        }
    }
}
```

Fig. 2 The revised objective function calculation of FRBCH-DE

3. Case Study

In this section of the article, FRBCH-DE is employed to design a cantilever retaining wall structure demonstrated in **Fig. 4**. The problem definition coded in C# is shown in **Fig. 5.** The objective function of the cantilever retaining wall design problem is illustrated in **Fig. 6**. Herein, PA denotes the earth force per unit length of the wall. PH and PV are the horizontal and vertical components of PA. The parameters of the backfill are as follows: $\gamma_1^{'} = 18$, $\phi_1^{'} = 30^o$, and $c_1^{'} = 0$. The parameters of the soil beneath the footing are as follows: $\gamma_2^{'} = 17.3$, $\phi_2^{'} = 20^o$, and $c_2^{'} = 38.3\ kPa$. The parameter H is 6 m and H1 is $CD \times \tan(10^o)$.

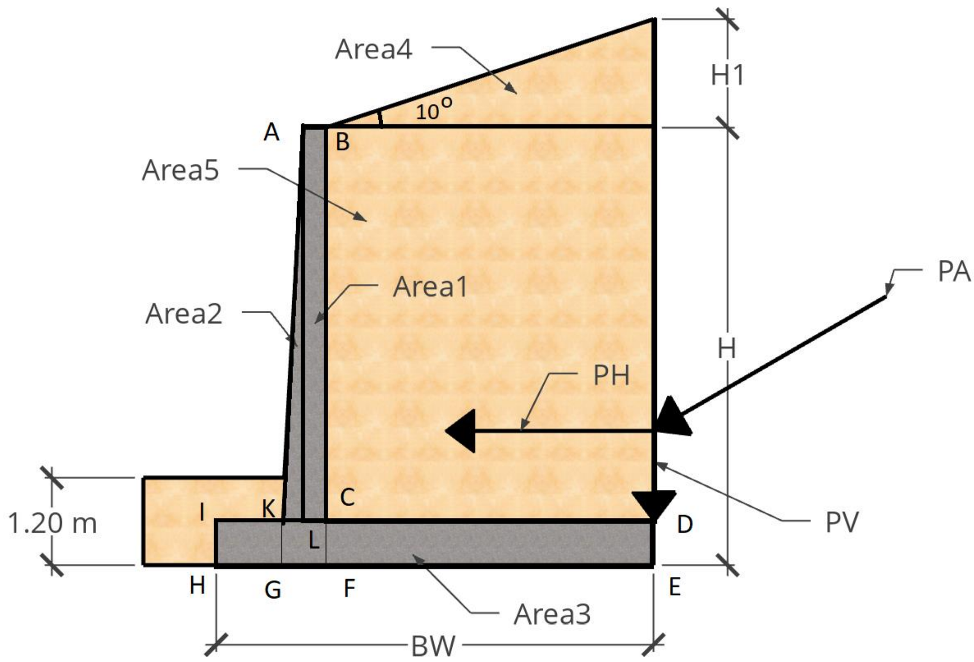

Fig. 3 Illustration of the cantilever retaining wall structure

```
class COP_ExternalStabilityCantileverRetainWall
{

    public readonly double alpha = 10; // degree
    public readonly double H0 = 6; // m
    public readonly double g2 = 17.3; // kN/m3
    public readonly double phi2 = 20; // degree
    public readonly double c2 = 38.3; // kPa
    public readonly double g1 = 18.1; // kN/m3
    public readonly double phi1 = 30; // degree
    public readonly double c1 = 0; // kPa
    public readonly double ConcreteGama = 23.56; // kN/m3
    public readonly double EmbDepth = 1.2; // m

    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
```

Fig. 4 The optimization problem parameters

```
public double ComputeObjFun(double[,] X)
{

    double AB = X[0, 0];
    double BC = X[0, 1];
    double CD = X[0, 2];
    double HE = X[0, 3];
    double HG = X[0, 4];

    double DE = H0 - BC;
    double KC = HE - CD - HG;

    double ObjFun = (AB*BC + 0.5*BC*(KC-AB) + DE*HE)*ConcreteGama; // Total weight
    // KC > AB
    return ObjFun;
}
```

Fig. 5 The objective function of the problem

Herein, there are five decision variables which determine the shape of the cantilever retaining wall (AB, BC, CD, HE, and HG). The objective function is basically the total weight of the structure (refer to **Fig. 5**). This objective function is given by:

$$\text{Min. } f = (\text{AB}\times \text{BC} + 0.5\,\text{BC}\times(\text{KC} - \text{AB}) + \text{DE}\times HE)\gamma_{\text{Concrete}} \quad (1)$$

where γ_{Concrete} =23.56 kN/m^3 denotes the mass density of concrete.

The FRBCH-DE method is utilized to find a set of the five decision variables which minimizes the total weight of the structure and satisfy all of the constraints regarding the safety against sliding, overturning, and safety regarding bearing capacity. For the details of those constraints, readers are guided to the previous work of [9]. Using 300 generations and a population size of 50, the best found cost function value is 22.95 and the design variables are 0.100 5.900 3.235 3.768, and 0.432. Additionally, all of the required constraints are satisfied. The computation time of the FRBCH-DE method is 6445 (ms).

4. Concluding remarks

This study has constructed and verified a cantilever retaining wall design approach based on the utilization of the DE evolutionary algorithm and the FRBCH method. The integrated approach, denoted as FRBCH-DE, has been developed with Visual C#.NET. A case study of cantilever retaining wall design involving the determination of five decision variables has been employed to verify the capability of FRBCH-DE. Experimental result shows that FRBCH-DE is able to find a good set of decision variables that feature a low objective function and satisfy all the required constraints.

References

[1] R. Sheikholeslami, B. G. Khalili, A. Sadollah, and J. Kim, "Optimization of reinforced concrete retaining walls via hybrid firefly algorithm with upper bound strategy," *KSCE Journal of Civil Engineering*, vol. 20, pp. 2428-2438, September 01 2016.

[2] V. Yepes, J. Alcala, C. Perea, and F. González-Vidosa, "A parametric study of optimum earth-retaining walls by simulated annealing," *Engineering Structures*, vol. 30, pp. 821-830, 2008/03/01/ 2008.

[3] I. Aydogdu, "Cost optimization of reinforced concrete cantilever retaining walls under seismic loading using a biogeography-based optimization algorithm with Levy flights," *Engineering Optimization*, vol. 49, pp. 381-400, 2017/03/04 2017.

[4] C. V. Camp and A. Akin, "Design of Retaining Walls Using Big Bang-Big Crunch Optimization," *Journal of Structural Engineering*, vol. 138, pp. 438-448, 2012.

[5] A. H. Gandomi, A. R. Kashani, D. A. Roke, and M. Mousavi, "Optimization of retaining wall design using evolutionary algorithms," *Structural and Multidisciplinary Optimization*, vol. 55, pp. 809-825, March 01 2017.

[6] E. N. Ghaleini, M. Koopialipoor, M. Momenzadeh, M. E. Sarafraz, E. T. Mohamad, and B. Gordan, "A combination of artificial bee colony and neural network for approximating the safety factor of retaining walls," *Engineering with Computers,* June 22 2018.

[7] B. Gordan, M. Koopialipoor, A. Clementking, H. Tootoonchi, and E. Tonnizam Mohamad, "Estimating and optimizing safety factors of retaining wall through neural network and bee colony techniques," *Engineering with Computers,* pp. 1-10, September 18 2018.

[8] A. Kaveh and A. F. Behnam, "Charged System Search Algorithm for the Optimum Cost Design of Reinforced Concrete Cantilever Retaining Walls," *Arabian Journal for Science and Engineering,* vol. 38, pp. 563-570, March 01 2013.

[9] M. Xiao, *Geotechnical Engineering Design*: John Wiley & Sons, ISBN: 9780470632239, 2015.

[10] 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.

[11] K. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution - A Practical Approach to Global Optimization*: Springer-Verlag Berlin Heidelberg, 2005.

[12] 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.

[13] T. V. Dinh, H. Nguyen, X.-L. Tran, and N.-D. Hoang, "Predicting Rainfall-Induced Soil Erosion Based on a Hybridization of Adaptive Differential Evolution and Support Vector Machine Classification," *Mathematical Problems in Engineering,* vol. 2021, p. 6647829, 2021/02/20 2021.

[14] N.-D. Hoang and Q.-L. Nguyen, "A Novel Approach for Automatic Detection of Concrete Surface Voids Using Image Texture Analysis and History-Based Adaptive Differential Evolution Optimized Support Vector Machine," *Advances in Civil Engineering,* vol. 2020, p. 4190682, 2020/07/28 2020.

[15] N.-D. Hoang, Q.-L. Nguyen, and Q.-N. Pham, "Optimizing Construction Project Labor Utilization Using Differential Evolution: A Comparative Study of Mutation Strategies," *Advances in Civil Engineering,* vol. 2015, p. 8, 2015.

[16] Bilal, M. Pant, H. Zaheer, L. Garcia-Hernandez, and A. Abraham, "Differential Evolution: A review of more than two decades of research," *Engineering Applications of Artificial Intelligence,* vol. 90, p. 103479, 2020/04/01/ 2020.

[17] O. Tarkhaneh and H. Shen, "An adaptive differential evolution algorithm to optimal multi-level thresholding for MRI brain image segmentation," *Expert Systems with Applications,* vol. 138, p. 112820, 2019/12/30/ 2019.

[18] M. Kaur, H. K. Gianey, D. Singh, and M. Sabharwal, "Multi-objective differential evolution based random forest for e-health applications," *Modern Physics Letters B,* vol. 33, p. 1950022, 2019.

[19] 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.

[20] R. M. John, G. R. Robert, and B. F. David, "A Survey of Constraint Handling Techniques in Evolutionary Computation Methods," in *Evolutionary Programming IV: Proceedings of the Fourth Annual Conference on Evolutionary Programming*, ed: MITP, 1995, p. 1.