

# Iterative Learning Control Based on Niche Shuffled Frog Leaping Algorithm Research

Xiaohong Hao, Dongjiang Wang

School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050,  
China

wangdj\_xhz@163.com

**Keywords:** Nonlinear system; Norm optimization; Parameter optimization; Niche shuffled frog leaping Algorithm.

**Abstract:** Aiming at slow convergence and low precision optimization in iterative learning control, a niche shuffled frog leaping algorithm was proposed in this paper, which combined memes algorithm and particle swarm algorithm, using the niche shuffled frog leaping algorithm based on the restrictive competition, avoiding the paedogenesis effectively improving the convergence speed and optimization accuracy. In order to achieve less error and monotone convergence in the iterative domain, get better transient tracking performance and establish a fast PID parameter optimization iterative learning control algorithm based on the discrete norm performance index, the PID controller was introduced into the iterative learning control parameter optimization algorithm to expand the algorithm's dimension, increase the degree of freedom in the optimal parameters, and ultimately promote the learning efficiency.

## 1. Introduction

Iterative learning control serves as a control technique for improving transient response and tracking of machines, devices or systems characterized by repetitive motion. Through modifying control signals short of expectation, it can optimize learning control algorithm and thus improve tracking performance of the system. Actual and expected output consistently apply iterative learning control algorithm into tracking trajectory of expected output. However, existing iterative learning control strategy suffers from defects such as low learning efficiency, slow convergence rate as well as repetitive iterative learning demanded for satisfactory tracking effects, which may then lead to substantial wastes in the system and thereby affect production efficiency. In order to further improve accuracy of learning control and boost learning efficiency, it is suggested to employ a combination between optimal algorithm and iterative learning control.

Amann, Owens et al [1] from the University of Sheffield have suggested an optimization-based iterative learning algorithm which has brought about new vitality to researches on iterative learning control algorithm. The evolution process of such an algorithm has already been systematically elaborated in the reference part: an optimal iterative learning control algorithm advanced by Amann et al in terms of the time-invariant linear discrete system, which can realize monotonic and geometrical convergence of tracking error [2]; Owens, Fang and Hätönen et al have put forward a parameter optimal iterative learning control (POILC) algorithm [3]; afterwards, Hätönen and Owens raised the high-order parameter optimal iterative learning control algorithm [4,5]. Although these algorithms are structurally simple and can realize monotonic and geometrical convergence of error to zero, the rate and efficiency of error convergence are still quite not satisfying. In engineering application, multiple iterations are necessary for achieving satisfactory tracking accuracy by using aforesaid methods.

Hatzikos and Owens then presented an optimal iterative learning control (GA-ILC) [6] based on genetic algorithm (GA) and obtained good control results. Later, Hatzikos et al extended this method [7] and achieved fine control effect in controlling nonlinear system. Nevertheless, this method

failed to encode useful prior information into the algorithm, which led to an extremely extensive searching space of the algorithm; moreover, weak searching capability of GA algorithm has also caused a low convergence rate of the algorithm.

Shuffled frog leaping algorithm (SFLA) is a cooperative-search swarm intelligent optimal algorithm proposed by Eusuff et al [8]. Individuals in the algorithm are characterized by memetic evolution and can realize global information exchanges by means of memetic algorithm. Suggested by Moscato in 1989, memetic algorithm (MA) is a swarm intelligent algorithm utilizing heuristic search in handling optimization problems; analogous to chromosome, “memetic” [9] refers to information deposited in human or animal brain for transmission as well as guidance of their own behaviors. (Particle swarm optimization PSO) algorithm acts as an evolutionary computation method adopted by Kennedy et al in 1995, which aims to cope with optimization problems by simulating foraging behavior of birds [10].

For the sake of further improving convergence effects of iterative learning algorithm, while also in terms of specific demands in iterative learning control, this paper improves shuffled frog leaping algorithm (SFLA) which has been successfully applied in solving varied optimization problems, and respectively puts forward a constrained linear system and nonlinear system that are capable of processing the input.

## 2. ILC problem description

Linear discrete system model

$$\begin{cases} x_k(t+1) = Ax_k(t) + Bu_k(t) \\ y_k(t) = Cx_k(t) \end{cases} \quad (1)$$

$x_k, u_k, y_k$  are the status variable output variable and input variable of the system running for the k-th time respectively. the desired trajectory on a given interval  $[0, T]$ , and the output error of the k-th time is  $e_k$ , then the learning law of the iterative learning control can be expressed as the following recurrence for:

$$u_{k+1} = f(u_k, u_{k-1}, \dots, u_{k-r}, e_{k+1}, e_k, \dots, e_{k-s}) \quad (2)$$

$$\lim_{k \rightarrow \infty} \|e_k\| = 0, \lim_{k \rightarrow \infty} \|u_k - u^*\| = 0 \quad (3)$$

The iterative learning control is convergent. The task of iterative learning control is to seek control inputs so that the trajectory of the controlled object achieves complete zero error tracking along the desired trajectory over a finite time interval  $[0, T]$ , then the deviation between the actual output and the desired output is zero, and requires the entire process to be completed quickly. Full tracking here refers to the system output from start to finish, whether transient or steady-state learning control, and to maintain the same target track.

## 3. Optimal Iterative Learning Control Based on Niche Shuffled Frog Leaping Algorithm Research

### 3.1 NSFLA-ILC convergence analysis and optimal solution

The ultimate goal of iterative learning algorithm is to solve the following optimization problem:  $\min_{u \in U} J_k(u_k)$ , its constraint condition is  $e = y_d - Pu$ . In order to facilitate the analysis, it is assumed that all the desired signals are in the range of the controlled object. Then there must be an optimal input, making the final tracking error.

$$e^* = y_d - Pu^* = 0 \quad (4)$$

The information transfer mode of Niche Shuffled Frog Leaping Algorithm (NSFLA) is conducted through classification of population and interaction between local search and global mixture process. Such kind of alternating method could effectively integrate the local information

with global information, making the algorithm get rid of being troubled by local optimal solution. This method could make the frog leaping algorithm have more efficient calculation performance and global search capacity.

The optimal iterative learning control algorithm based on NSFLA adopts the norm optimal index in NOILC: As the fitness function of Niche Shuffled Frog Leaping Algorithm, through search, the optimal solution  $u_{k+1}^*$  used for next iterative operation is acquired. Where,  $y_{k+1}(t) = [Pu_{k+1}](t)$ . Provided the input (it is assumed that there is one optimal solution for question (4) at least), then the next iterative computation could make the formula (4) true:

When the controlled object  $P$  in  $y_{k+1} = Pu_{k+1}$  is linear time invariant system (LTI), the optimal input could be acquired from formula (3.1):

$$u_{k+1} = u_k + P^* e_{k+1} \quad (5)$$

Where  $P^*$  is the adjoint operator of  $P$ . It is the non-causal realization of algorithm. There is following conclusion for the discrete linear time-invariant system:

$$\|e_{k+1}\| \leq \frac{1}{1+\sigma} \|e_k\| \quad (6)$$

Where,  $\sigma > 0$  is the minimum eigen value of  $P$ . Inequality (7) shows: in this case, the algorithm is uniformly convergent.

For the linear system, the above algorithm could be realized. While the nonlinear system does not exist possibly, which could not be realized with the adjoint operator  $P^*$ . As for the control problem of nonlinear system, this Paper deals with it through solving the optimization problem by using NSFLA in each iterative process.. There is a description that as long as SFLA could solve one optimal solution of question (3.1), at least, formula (5) is true. As for non-linear system, NSFLA-ILC could guarantee that its tracking error is monotonic convergent in iteration domain.

When the object is discrete system, we consider the following optimization problem:

$$\min_{u_{k+1} \in U} J(u_{k+1}) = \|e_{k+1}\|^2 + \|u_{k+1} - u_k\|^2 \quad (7)$$

Its constraint condition is  $y_{k+1} = G_e u_{k+1}$ ,  $G_e$  is the controlled object model defined as follows:

$$G_e = \begin{bmatrix} PB & 0 & 0 & \cdots & 0 \\ PAB & PB & 0 & \cdots & 0 \\ PA^2B & PAB & PB & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ PA^{N-1}B & PA^{N-2}B & \cdots & \cdots & PB \end{bmatrix} \quad (8)$$

$$u_k := [u_k(0), u_k(1), \dots, u_k(N-1)]^T \quad (9)$$

$$y_k := [y_k(0), y_k(1), \dots, y_k(N-1)]^T \quad (10)$$

In algorithm (3.3, the norm of input deviation is  $\|u_{k+1} - u_k\| = \gamma^2 \|e_k\|^2$ , namely, the input deviation is proportional to  $\gamma$ . Where

$$u_{k+1}(t) = u_k(t) + \gamma_{k+1} e_{k+1}(t+1) \quad (11)$$

Where,  $\gamma_{k+1}$  is chosen as the solution of following optimization problem

$$\begin{cases} \min_{\gamma_{k+1} \in R} J_{k+1}(\gamma_{k+1}) \\ J_{k+1}(\gamma_{k+1}) = \|e_{k+1}\|^2 + \omega \gamma_{k+1}^2 \end{cases} \quad (12)$$

Where,  $\omega > 0$ . Such kind of algorithm has same convergence characteristics with the norm optimal iterative learning control algorithm introduced in above section (the constraint condition is  $y_{k+1} = G_e u_{k+1}$ ).

### 3.2 Iterative learning control algorithm based on Niche Shuffled Frog Leaping Algorithm

#### 3.2.1 Niche Shuffled Frog Leaping Algorithm

While solving the optimization problem, the Niche Shuffled Frog Leaping Algorithm has excellent global searching ability and fast convergence speed. As for the typical shuffled frog-leaping algorithm, the local search will easily make it trap in the disadvantage of local optimum. The introduced niche technology limiting competition strategy could improve the global optimizing ability of algorithm and accelerate the convergence speed of algorithm. Niche technology can maintain the diversity of the solution, with a high global optimization and convergence rate. It divides every generation of individual population into several categories and each category has a group composed of excellent representatives. They represent the individual with the highest fitness. They hybridize and mutate in the same population and among different populations, and then generate new generation of population individual. Each individual survives in the specific environment. Competition exists in the same kind of individual and among different individuals. There are information exchanges between different kinds of individuals.

The traditional shuffled frog-leaping algorithm classifies the populations. Before the local search, it introduces RCS niche technology and classifies the initial population into the mutually exclusive sub-populations. It forms independent searching space dynamically, restraining the convergence due to community cooperation.

##### (1) Initialization

The sum of initialized population solution (frog) is  $F$ , the number of sub-population is  $m$ , the number of each sub-population solution is  $n$ , dimension is  $S$ , the number of hybrid iteration is  $G$ , the updating number of each sub-population local is  $N$ , the maximum moving step length of solution is  $D_{\max}$ , the radius of niche is  $R$  and fitness function. Within the feasible solution domain, the initial population composed of  $F$  solution  $X$  is constituted randomly, and then the  $fitness(x)$  of each solution is calculated. Where, the  $a(1 \leq a \leq F)$ <sup>th</sup> solution is represented as  $X_a = (x_a^1, x_a^2, \dots, x_a^S)$ .

##### (2) Generation of frogs

The initial frog population  $U = \{u_1(t), u_2(t), \dots, u_F(t)\}$  is generated randomly, the iteration value  $t = 0$ , given each frog  $u_i(t), i = 1, 2, \dots, F$ , and then the fitness function  $f_i(t) = f(u_i(t))$  (formula (11)) is calculated; the fitness of population is sorted in degressive mode, which shall be stored in the form of  $X_i(t) = \{u_i(t), f_i(t)\}$  and the optimal frog in the frog population is recorded as  $u_g(t)$ .

##### (3) Niche structure adopting RCS strategy

Step1: the  $i(1 \leq i \leq m-1)$  sub-population is compared with  $i(1 \leq i \leq m-1)$  sub-population. The optimal solution  $x_{ib}$  in  $i$  sub-population and the optimal solution  $x_{jb}$  in  $j$  sub-population are calculated, and then the Euclidean distance between the optimal solution  $x_{ib}$  in  $i$  sub-population and the optimal solution  $x_{jb}$  in  $j$  sub-population is calculated.  $d_{ij}$  is compared with niche radius  $R$ .

Step2: if  $d_{ij} < R$ , the  $fitness(x_{ib})$  and  $fitness(x_{jb})$  of  $x_{ib}$  and  $x_{jb}$  are compared, otherwise skip to step 6; if  $fitness(x_{ib}) < fitness(x_{jb})$ , set the  $fitness(x_{jb})$  as the infinity and initialize  $x_{jb}$  randomly, otherwise set  $fitness(x_{ib})$  as the infinity and initialize  $x_{ib}$  randomly.

Step3: if all niche population has optimal solution, continue step 4; otherwise skip to step 1, repeat pairwise comparison between the optimal solution of subpopulation.

##### (4) Local search

Step4: in each sub-population, update the solution  $x_w$  with worst fitness in the sub-population; the moving step length of solution is:

$$D_t = \sin\left(\frac{\pi}{2} \cdot \frac{t}{N}\right) \cdot (x_b - x_w) \quad (13)$$

The location of worst solution after updating is:

$$x_w = x_w + D_t, -D_{\max} \leq D_t \leq D_{\max}, t = 1, 2, \dots, n \quad (14)$$

Where,  $D_{\max}$  is the maximum step length,  $t$  is the algebra of current local search.

Step5: in each sub-population, compare the  $fitness'(x_w)$  of worst solution after updating with the original  $fitness(x_w)$  of worst solution. if  $fitness'(x_w) < fitness(x_w)$ , replace the original solution with the updated worst solution; otherwise, and calculate the moving step length of worst solution  $x_w$ .

$$D_t = \sin\left(\frac{\pi}{2} \cdot \frac{t}{N}\right) \cdot (x_b - x_w), t = 1, 2, \dots, n \quad (15)$$

And update the location of worst solution. In each sub-population, compare the  $fitness'(x_w)$  of updated worst solution with the  $fitness(x_w)$  of original worst solution, if  $fitness'(x_w) < fitness(x_w)$ , replace the original worst solution with the updated worst solution.

(5) Elimination mechanism of sub-population

After the completion of local search evolution, calculate the change value of fitness of optimal solution in each sub-population, if it is less than a smaller value within the preset continuous algebras, such sub-population is eliminated and it shall be initialized again randomly, otherwise, continue step 4.

Step6: mix the solutions of all sub-populations and constitute a complete population containing F solutions.

Step7: Judge whether it meets the preset mixing iteration number or end condition, if not, skip to step 8 and conduct next round of local search, otherwise, end it.

Niche Shuffled Frog Leaping Algorithm transfers information according to the classification of population, alternating the local evolution and re-mixing process and effectively integrating the global information interaction and local evolution search. It has highly efficient computation performance and excellent global searching ability. According to the prior information of controlled object, the advisable top and lower limit value  $u_j^{\max}$  and  $u_j^{\min}$  with different moment input could be set, which can greatly reduce the searching space of optimization algorithm and then reduce the computation cost.

#### 4. Simulation

For linear system with input constraint and expectation of square wave system, the iteration learning of Niche Shuffled Frog Leaping Algorithm shall be applied to verify the expectation of such improved algorithm in the non-linear system, which could reach a good effect.

The following non-linear control systems are adopted

$$\begin{cases} x_j(i+1) = A_D x_j(i) + B_D u_j(i) \\ y_j(i) = C_D x_j(i) \\ i = 0, 1, \dots, 100 \\ j = 0, 1, \dots \end{cases} \quad (16)$$

$$\text{Where, } A_D = \begin{bmatrix} 0.95 & 0 & 0 \\ 0.45 & 0.95 & 0 \\ 0.03 & 0.010 & 1 \end{bmatrix}; \quad B_D = \begin{bmatrix} 0 \\ 0.0197 \\ 0.0211 \end{bmatrix}; \quad C_D = [0 \quad 0 \quad 1] \quad (17)$$

Bring  $A_D$   $B_D$   $C_D$  into formula (3.1), obtain  $p = [p_1 \quad p_2 \quad \dots \quad p_{100}]^T$ ; the function  $p_c$  related to  $k_p$ ,  $k_i$ ,  $k_d$  could be obtained from  $p$ , and then  $J = \|p_c\|_1$  could be obtained.

The parameter setting of Niche Shuffled Frog Leaping Algorithm is shown as Table 4.1. Through

Niche Shuffled Frog Leaping Algorithm, the function value of fitness could be obtained through Niche Shuffled Frog Leaping Algorithm.

The parameters of the shuffled frog leaping algorithm are showed in the table 3.1. The fitness function values can be obtained by the algorithm.

$$J = \|p_c\|_1 = 1.1056, k_p = 0.9, k_I = -0.65, k_D = 13.6 \tag{18}$$

The value of  $k_p, k_I, k_D$  is substituted into PID iterative learning control in (19)

$$u_{j+1}(i) = u_j(i+1) + 0.9e_j(i) - 0.065 \sum_{m=1}^{i+1} e_j(m) + 13.6[e_j(i+1) - e_j(i)] \tag{19}$$

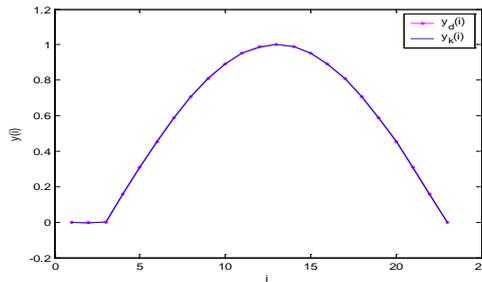


Fig. 1 Tracking performance of sine function

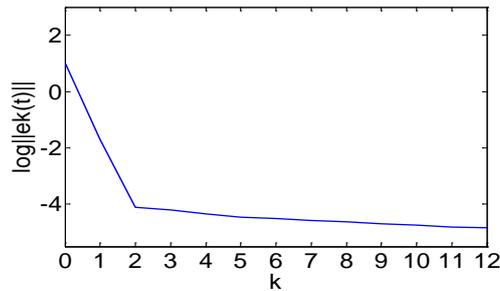


Figure 2 NSFLA-ILC Nonlinear System error convergence curve

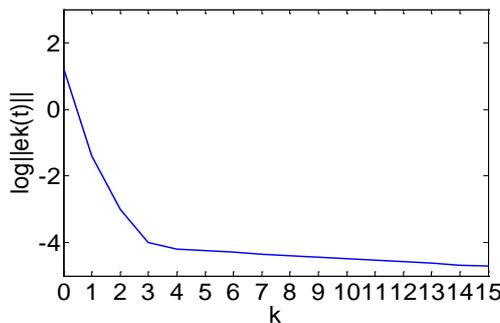


Figure 3 CSA-ILC Nonlinear System error convergence curve

The parameter setting of Niche Shuffled Frog Leaping Algorithm is shown as Figure 1. Through Niche Shuffled Frog Leaping Algorithm, the function value of fitness could be obtained through Niche Shuffled Frog Leaping Algorithm.

## 5. Conclusions

In terms of the improved Niche Shuffled Frog Leaping Algorithm, namely the application of Niche Shuffled Frog Leaping Algorithm into the iteration learning control, this Paper puts forward the optimal iteration-learning control algorithm based on Niche Shuffled Frog Leaping Algorithm. Compared with traditional optimization method, the advantage of NSFLA-ILC lies in that it could directly solve non-linear and constrained input problem. NSFLA-ILC combines the advantages of

meme algorithm based on genetic algorithm and the particle swarm optimization based on swarm foraging behavior, thus resulting in few algorithm parameters, fast computation speed and strong local searching ability, which could deal with the constrained input problem properly. The real number encoding of algorithm and the inflection filters of input acquired by improving NSFLA primarily remove the high-frequency part of input due to the algorithm and make NSFLA-ILC have better convergence effect.

## References

- [1] Jari Hätönen. Issues of algebra and optimality in iterative learning control[PhD. thesis]. Department of Process and Environmental Engineering, University of Oulu.2004.
- [2] V Hatzikos, J Hätönen, D H Owens. Genetic algorithms in norm-optimal linear and non-linear iterative learning control. International[J]. Journal of Control, 2004, 77(2): 188-197
- [3] D H Owens, J Hätönen. Iterative Learning control-an optimization paradigm[J]. Annual Reviews in Control, 2005(29), 57-70.
- [4] J Hätönen, D H Owens. Basis Function and Parameter Optimization in High-order Iterative Learning Control[J]. Automatic, 2005(42), 287-294.
- [5]Tayebi A. Analysis of two particular iterative learning control schemes in frequency and time domains[J]. Automatica, 2007, 43 (9), 1565-1572.
- [6] V Hatzikos, J Hätönen, D H Owens. Genetic algorithms in norm-optimal linear and non-linear iterative learning control. International[J]. Journal of Control, 2004, 77(2): 188-197.
- [7] Jari Hätönen. Issues of algebra and optimality in iterative learning control[PhD. thesis]. Department of Process and Environmental Engineering, University of Oulu. 2004.
- [8] Eusuff M M, Lansey K E. Optimization of water distribution network design using the shuffled frog leaping algorithm[J]. Journal of Water Resources Planning and Management, 2003, 129(3): 210-225.
- [9]Moscato P. On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms[J]. Caltech concurrent computation program, C3P Report, 1989, 826: 1989.
- [10]Kennedy J. Particle swarm optimization[M]Encyclopedia of Machine Learning. Springer US, 2010: 760-766