Modified Brain Storm Optimization Algorithms Based on Topology Structures

Li Li[1], F.F. Zhang[1], Xianghua Chu[1(✉)], and Ben Niu[1,2(✉)]

[1] College of Management, Shenzhen University, Shenzhen 518060, China
x.chu@hotmail.com, Drniuben@gmail.com
[2] Department of Mechanism Engineering, Hong Kong Polytechnic University, Kowloon, Hong Kong

Abstract. An algorithm performs better often due to its communication mechanisms. Different types of topology structures denote various information exchange mechanisms. This paper incorporates topology structure concept into brain storm optimization (BSO) algorithm. Three types of topology structures, which are full connected, ring connected and star connected, are introduced. And three novel modified optimization algorithms based on topology structures are proposed (BSO-FC, BSO-RI, BSO-ST). Unimodal and multimodal criteria functions are employed to verify the effectiveness of the raised algorithms. In addition, both the original BSO algorithm and bacterial foraging optimization (BFO) algorithm are selected as contrastive algorithms to expose the optimization capacity of the proposed algorithms. Experimental results show that all of the modified algorithms have better performance than the original BSO algorithm, especially the BSO-ST algorithm.

Keywords: Brain storm optimization · Topology structures · Population-based optimization · Mutation operator · Gaussian mutation

1 Introduction

Optimization problem is an important branch of modern management. In many years, people are trying their best to find better ways to solve these problems. In recent years, optimization algorithms based on the population have been extensively investigated. In contrast to algorithms based on single-point, for example, hill-climbing algorithm, population-based optimization algorithms do its jobs by communicating and competing with each other. Nowadays, population-based optimization algorithms are generally classified as swarm intelligence algorithms.

There are many swarm intelligence algorithms, such as particle swarm optimization (PSO) [1], bacterial foraging optimization (BFO) [2], artificial bee colony optimization (ABC) [3], ant colony optimization (ACO) [4], etc. But all of them are just inspired by simple animals or insects, such as ants, bees, birds, etc.

As a new type of swarm intelligence, brain storm optimization (BSO) was first proposed by Shi in [5, 6]. BSO is motivated by the most intelligent organisms, the human being. After that, many scholars have conducted research on this algorithm because of its excellent performance.

Y. Tan et al. (Eds.): ICSI 2016, Part II, LNCS 9713, pp. 408–415, 2016.
DOI: 10.1007/978-3-319-41009-8_44

Some people do research on the parameters of this algorithm. According to the dynamic range in the iteration, reference [7] proposed a dynamic step-size strategy, which was dynamically changed in each iteration. Reference [8] made investigation on the parameters in the BSO to see how they affected the performance of BSO. In order to reduce the computation time, reference [9] implemented only several iterations of the k-means clustering method in each iteration of BSO, instead of achieving a convergent clustering by using k-means clustering method. Reference [10] adopted the concept of preda-tor-prey into BSO to make full use of the global knowledge.

In addition, the BSO algorithm is also applied to solve the practice problems. Reference [11] found the optimal solution of economic dispatch problem considering wind power based on the BSO algorithm. Reference [10] applied the modified BSO to optimally design a DC brushless motor to maximize its efficiency.

In the BSO algorithm described by Shi [5], it contains two interesting characters. One character is the clustering process that all the ideas are divided into several groups by k-mean clustering method. The other character is the individual generation process that produces new idea by mixing a Gaussian random noise in. As we all know, BSO algorithm has been studied by many researchers, but it is just in its infancy now, and lots of studies are needed.

In this present paper, we concentrate on the character of the individual generation process to come up with three modified BSO algorithms based on the topology structures. Experiments show that the raised algorithms can improve the capability of the original BSO, obviously.

The rest part of this paper is organized as follows. Section 2 demonstrates the BSO algorithm. The modified BSO algorithms are presented in Sect. 3. Benchmark functions are used to verify the proposed algorithms in Sect. 4. Finally, Sect. 5 summarizes the concluding remarks.

2 Brain Storm Optimization Algorithm

As a creative problem-solving method, brain storm process was systematized by Ostorn in 1939. In a brainstorming process, the idea generation often follows four principles. The detailed description is listed in Table 1. When facing the complicated problem which a single people cannot solve, human being get together to brain storm, especially people with different background. This is the central idea of this method and this process can improve the probability of solving the difficult problem.

Table 1. Osborn's original rules for idea generation in a brainstorming process

Rules	Content
Rule1	Suspend Judgment
Rule2	Anything Goes
Rule3	Cross-fertilize
Rule4	Go for Quantity

Inspired by the process of human being idea formation, Shi came up with a new algorithm called brain storm optimization (BSO) [5]. The pseudo code of the BSO algorithm is shown in Fig. 1.

In original BSO algorithm, new ideas are formed by mixing Gaussian noise in. The main formulas are enumerated as follows, where X_{i+1} denotes the new individual, X_i denotes the selected idea, $N(\mu, \sigma)$ is Gaussian function, *logsig()* is a logarithmic sigmoid transfer function, and k is a coefficient, $max_{iteration}$ is the maximum number of

```
Algorithm BSO
Begin
   Initialize variables and generate N ideas;
  While
    Cluster;
    Select cluster center;
    If(random(0,1)<p_replace)
       Selected a cluster randomly and replace
       the cluster center with a randomly
       generated idea;
    End
  FOR(each ideas)
     If(random(0,1)<p_one)
       Randomly select a cluster with a
       probability p;
        If(random(0,1)<p_one_center)
          Add random values to the selected
          cluster center to generate a new
          idea;
        Else
          Add random values to a random idea of
          the selected cluster to generate a
          new idea;
        End If
      Else
        Random select two clusters
        If(random(0,1))<p_two_center
          Combine the two selected cluster
          center and add with random values to
          generate a new idea;
        Else
         Combine two random ideas from the two
         selected clusters and add with random
         values to generate a new idea;
        End If
     End If
  End For
  End While
End
```

Fig. 1. Pseudo-code of the BSO algorithm

iterations, $current_{iteration}$ is the current iteration number, and *rand()* is a random value within (0,1).

$$X_{i+1} = X_i + \varepsilon \times \mathrm{N}(\mu, \sigma) \tag{1}$$

$$\varepsilon = logsig\left(\frac{0.5 \times max_{iteration} - current_{iteration}}{k}\right) \div rand() \tag{2}$$

3 The Modified Brain Storm Optimization Algorithms

The excellent performance of the swarm intelligence optimization often dues to the information communication mechanism, like the bees cooperate and compete with each other. The introduction of information exchange mechanisms is aimed to prevent population from local optimum. Each topology structure contains a special information communication strategy. Reference [12] introduced the topology structures in detail. Commutating with each other, population can get more information from others and share information to others, which can improve their behavior in the process of searching optimization solution. In this paper, we incorporates the concept of topology structure into individual generation process instead of the Gaussian function.

Three topology structures, including full connected topology, ring connected topology, star connected topology, are used in this paper. The elementary diagram of the topology structures are presented in Fig. 2 [13].

According to the topology structures, the proposed algorithms based on topology structures are named BSO-FC, BSO-RI, and BSO-ST, respectively.

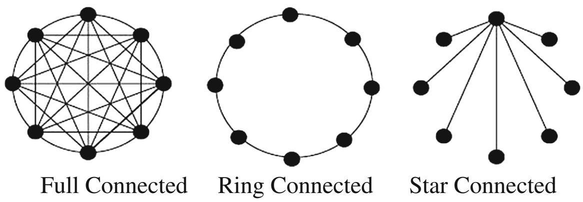


Fig. 2. Three types of topology structures

4 Experiments and Discussions

4.1 Benchmark Functions

Six benchmark functions are applied to test the virtue of the BSO-FC, BSO-RI and BSO-ST. Among them, Sphere, SumPowers, Schwefel221 are unimodal functions, while Apline, Rastrigin, Griewank are multimodal functions. These benchmark functions are employed to verify the capability of the proposed algorithms. The benchmark functions are introduced in Table 2.

Table 2. Search ranges of the benchmark functions

Attribute	Name	Function	Range
Unimodal	Sphere	$f(x) = \sum_{i=1}^{n} X_i^2$	[−100,100]
	SumPowers	$f(x) = \sum_{i=1}^{n} \lvert X_i \rvert^{i+1}$	[−10,10]
	Schwefel221	$f(x) = max\lvert X_i \rvert$	[−10000,10000]
Multimodal	Apline	$f(x) = \lvert X_i \times sin(X_i + 0.1 \times X_i \rvert$	[−10,10]
	Rastrigin	$f(x) = \sum_{i=1}^{n} (X_i^2 - 10cos(2\pi X_i)\ +\ 10)^2$	[−5.12,5.12]
	Griewank	$f(x) = \frac{1}{4000}\sum_{i=1}^{n} X_i^2 - \prod_{i=1}^{n} cos(\frac{X_i}{\sqrt{i}}) + 1$	[−600,600]

4.2 Parameter Settings

In the experiments, each test is run for 1000 iterations and 20 replications are conducted for each test to collect the statistics. All the functions mentioned above are tested with 20 dimensions. We set the population size is 20. The notations used are summarized in Table 3. The parameter settings primarily for algorithms are shown in Table 4.

Table 3. Parameters for the algorithms

Parameters	Description
Ped	The probability of demise
Fed	The interval iterations to perform demise operation
C	Chemotactic step
Ns	The number of executions when a bacteria looking for a good food source
p_replace	Probability to determine whether a dimension is disrupted or not
p_one	Probability for select one individual, not two, to generate new individual
p_one_center	Probability of using one cluster center
k	The slope for changing *logsig*() function
p_two_center	Probability of using two cluster centers

Table 4. Algorithm configurations

Alogithm	Parameter settings	
BFO	Ped = 0.2 Fed = 5	C = 0.05 Ns = 8
BSO		
BSO-FC	p_replace = 0.2	p_one = 0.8
BSO-RI	p_one_center = 0.4	k = 20
BSO-ST	p_two_center = 0.5	

4.3 Comparison on Solutions

The mean and standard deviation for the different functions given by BFO, BSO, BSO-FC, BSO-RI and BSO-ST are shown in Tables 5 and 6. The best values given by the five algorithms are signed as bold. As listed in Tables 5 and 6, BSO-ST indicates better performance than the other four algorithms on Ackley, Schwefel221, Griewank and Rastrigin functions, while the BSO-FC reveals better capability on Sphere and SumPowers functions slightly.

For the three unimodal functions, Table 5 shows that BSO-ST is the best algorithm for Schwefel221 and the second best algorithm for Sphere and SumPower. For the three mulmodal functions, Tables 6 tells that BSO-ST performs the best on Ackley, Rastrigin, and Griewank. Owing to the commutation mechanism, which is open minded, BSO-ST reveals excellent capability on multimodal functions. It is similar to the brainstorming process that any thought of your mind should not be ignored.

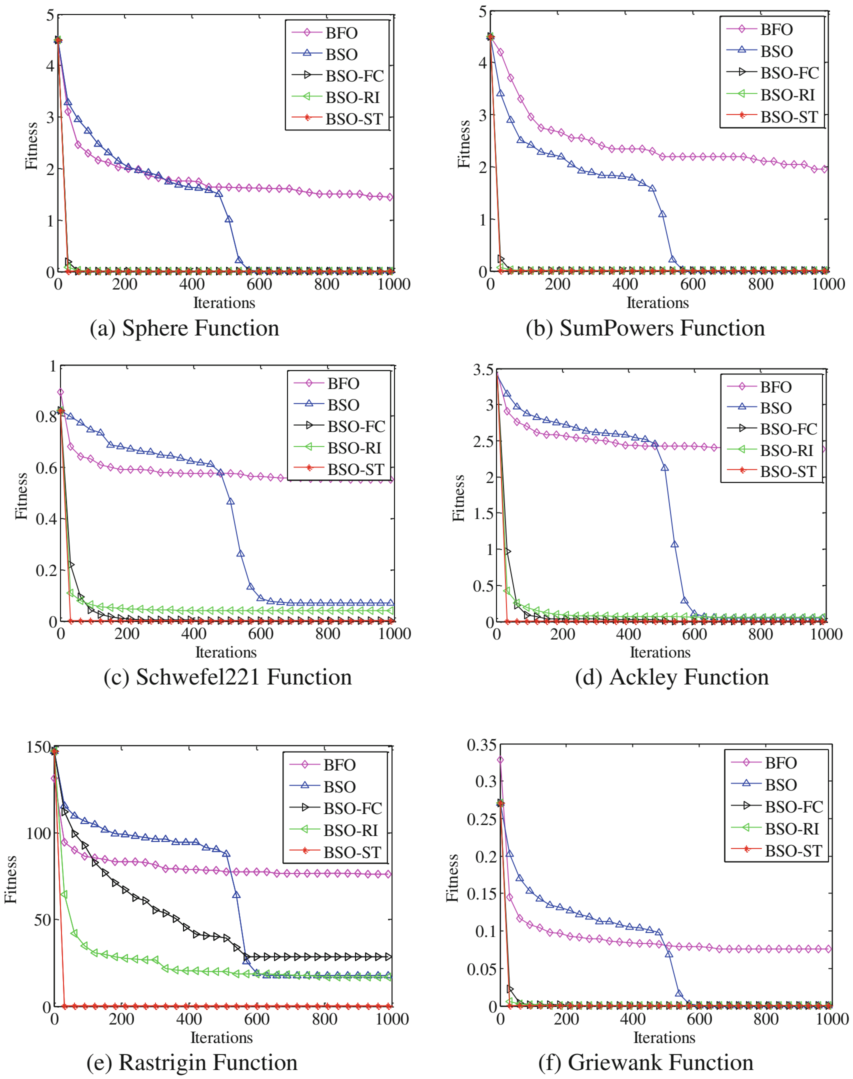
The convergence progress of the mean fitness values are shown in Fig. 3. It is obvious that for most functions, the BSO-ST converges the fastest at the beginning and finds the optimum solution soon.

Table 5. Test data based on unimodal criteria functions

Algorithm		Sphere	SumPowers	Schwefel221
BFO	Mean	1.44368e+000	1.68039e−002	5.45828e−001
	Std.	3.98885e−001	1.20481e−002	5.52406e−002
BSO	Mean	9.48757e−005	9.73018e−005	7.11372e−002
	Std.	1.0325e−004	1.7128e−004	1.84648e−002
BSO-FC	Mean	**3.11238e−008**	**4.01170e−008**	2.66992e−003
	Std.	1.39190e−007	1.79409e−007	9.57804e−003
BSO-RI	Mean	3.56433e−003	3.14258e−003	3.98185e−002
	Std.	7.64096e−003	8.54150e−003	4.00449e−002
BSO-ST	Mean	1.74583e−007	1.74377e−007	**9.53348e−005**
	Std.	**1.37360e−008**	**1.4302e−008**	**4.1704e−006**

Table 6. Test data based on multimodal criteria functions

Algorithm		Ackley	Rastrigin	Griewank
BFO	Mean	2.38442e+000	7.62865e+001	7.61453e−002
	Std.	1.90104e−001	8.92492e+000	2.07259e−002
BSO	Mean	4.87586e−002	5.00343e+000	2.50661e−004
	Std.	1.61261e−001	1.76937e+001	3.6427e−004
BSO-FC	Mean	1.12938e−003	2.83987e+001	2.26755e−006
	Std.	5.05073e−003	2.71628e+001	9.50192e−006
BSO-RI	Mean	5.55449e−002	1.68623e+001	1.94659e−004
	Std.	8.33758e−002	3.70785e+001	4.3508e−004
BSO-ST	Mean	**3.74755e−004**	**2.94473e−005**	**1.37943e−008**
	Std.	**1.2816e−005**	**3.0775e−006**	**1.6663e−009**



(a) Sphere Function (b) SumPowers Function

(c) Schwefel221 Function (d) Ackley Function

(e) Rastrigin Function (f) Griewank Function

Fig. 3. The optimal values with the change of iterations

5 Conclusions

This paper presents three novel modified brain storm optimization algorithms based on different types of topology structures. Topology structures are used to generate new individuals instead of the Gaussian Function. Because of the good commutation mechanism of the topology structure, we expect that the proposed algorithms can achieve better performance.

The capability of the raised algorithms are tested by criteria functions and compared with BFO and BSO. These functions contain three unimodal functions and three multimodal functions. Experiment results have indicated that the solutions obtained by BSO-FC, BSO-RI and BSO-ST are better than these produced by the original BSO and BFO, obviously, especially the BSO-ST.

Acknowledgments. This work is partially supported by The National Natural Science Foundation of China (Grants nos. 71571120, 71271140, 71461027, 71471158, 71501132) and the Natural Science Foundation of Guangdong Province (Grant nos. 1614050000376).

References

1. Kennedy, J., Eberhart, R., Shi, Y.: Swarm Intelligence. Morgan Kaufmann Publisher, Burlington (2001)
2. Passion, K.M.: Bacterial foraging optimization. Int. J. Swarm Intell. Res. **1**(1), 1–16 (2010)
3. Karaboga, D.: An idea based on honey bee swarm for numerical optimization. Technical report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department (2005)
4. Dorigo, M., Stützle, T.: Ant Colony Optimization. The MIT Press, Cambridge (2004)
5. Shi, Y.: Brain storm optimization algorithm. In: Tan, Y., Shi, Y., Chai, Y., Wang, G. (eds.) ICSI 2011, Part I. LNCS, vol. 6728, pp. 303–309. Springer, Heidelberg (2011)
6. Shi, Y.: An optimization algorithm based on brainstorming process. Int. J. Swarm Intell. Res. **2**, 35–62 (2011)
7. Zhou, D., Shi, Y., Cheng, S.: Brain storm optimization algorithm with modified step-size and individual generation. In: Tan, Y., Shi, Y., Ji, Z. (eds.) ICSI 2012, Part I. LNCS, vol. 7331, pp. 243–252. Springer, Heidelberg (2012)
8. Zhan, Z., Chen, W., Lin, Y., Gong, Y., Li, Y., Zhang, J.: Parameter investigation in brain storm optimization. In: 2013 IEEE Symposium on Swarm Intelligence, Singapore (2013)
9. Zhan, Z., Zhang, J., Shi, Y., Liu, H.: A modified brain storm optimization. In: Proceedings of Congress on Evolutionary Computation, pp. 1–8. Brisbane, Australia (2012)
10. Duan, H., Li, S., Shi, Y.: Predator-prey based brain storm optimization for DC brushless motor. IEEE Trans. Magn. **49**, 5336–5340 (2013)
11. Jadhav, H.T., Sharma, U., Patel, J., Roy, R.: Brain storm optimization algorithm based economic dispatch considering wind power. In: 2012 IEEE International Conference on Power and Energy, Kota Kinabalu Sabah, Malaysia (2012)
12. McNabb, A., Gardner, M., Seppi, K.: An exploration of topologies and communicational in large particle swarms. In: Proceedings of the IEEE Congress on Evolutionary Computation, pp. 712–719 (2009)
13. Niu, B., Liu, J., Bi, Y., Tan, L.J.: Improved bacterial foraging optimization algorithm with information communication mechanism. In: Computational Intelligence and Security (CIS), pp. 47–51 (2014)

Theorem 3.1. Let $f(x) \in F_q[x]$ be a polynomial over F_q of degree n and $\phi(q^n - 1) = q^n - q$. $f(x)$ is irreducible if and only if $\mathrm{ord}(f(x)) = q^n - 1$.

Lemma 3.2 [12]. For every finite field F_q and every $n \in N$, the product of all irreducible polynomials over F_q whose degree divides n is equal to $x^{q^n} - x$.

Lemma 3.3 [12]. Let c be a positive integer. Then the polynomial $f(x) \in F_q[x]$ with $f(0) \neq 0$ divides $x^c - 1$ if and only if $\mathrm{ord}(f(x))|c$.

Let $f(x)$ be a polynomials in $F_q[x]$ of degree n, and $n = p_1^{t_1} p_2^{t_2} \ldots p_s^{t_s}$ is the prime factor decomposition of n. Let $n_i = n/p_i$.

Theorem 3.4. Let $f(x)$ be a polynomial over F_q of degree n. If $f(x)$ satisfies the following properties:

(1) $\mathrm{ord}(f(x))|q^n - 1$;
(2) For every $c \in F_q, f(c) \neq 0$;
(3) $\gcd(\mathrm{ord}(f(x)), q^{n_i} - 1) = 1$, $(i = 1, 2, \ldots, s)$

then $f(x)$ is an irreducible polynomial over F_q.

Proof. Since $\mathrm{ord}(f(x))|q^n - 1$, we have $f(x)|x^{q^n-1} - 1$.

According to Lemma 3.3, $f(x)$ has no repeated factor. Suppose $f(x)$ were reducible over F_q. Then we have a factorization $f(x) = f_1(x)f_2(x)\ldots f_t(x)$, where each $f_j(x) (j = 1, 2, \ldots, t)$ are pairwise relatively prime. Since $f_j(x)|x^{q^n-1} - 1$, then

$$\deg(f_j(x))|n \quad (j = 1, 2, \ldots, t).$$

we claim that

$$\deg(f_j(x)) \nmid n_i \ (i = 1, 2 \ldots, s).$$

Suppose $\deg(f_j(x))|n_k$ for some $1 \leq k \leq s$. Then

$$f_j(x)|x^{q^{n_k}} - x \quad \text{and} \quad f_j(x)|x^{q^{n_k}-1} - 1.$$

By Lemma 3.2, since

$$\mathrm{ord}\big(f_j(x)\big)|q^{n_k} - 1 \text{ and } \mathrm{ord}\, f_j(x)|\mathrm{ord}\, f(x),$$

by Lemma 3.3, we have

$$\gcd(\mathrm{ord}(f(x)), q^{n_k} - 1) \neq 1.$$

a contradiction to (3). Therefore, $\deg(f_j(x)) = n$ and $f_j(x) = f(x)$. Hence, $f(x)$ is an irreducible polynomial over F_q.

Theorem 3.5. If $f(x)$ is an irreducible polynomial over F_q, then

(1) $\mathrm{ord}(f(x))|q^n - 1$;
(2) For every $c \in F_q, f(c) \neq 0$;
(3) $\mathrm{ord}(f(x)) \nmid q^{n_i} - 1$.

Proof. Since $f(x)$ is an irreducible polynomial over F_q of degree n, then $f(c) \neq 0$ for every $c \in F_q$, and $\mathrm{ord}(f(x))|q^n - 1$.

Suppose

$$\mathrm{ord}(f(x))|\ q^{n_k} - 1 \ \text{ for some } \ 1 \le k \le s.$$

According to Lemma 3.3,

$$f(x)|x^{q^{n_k}-1} - 1.$$

Then

$$f(x)|x^{q^{n_k}} - x.$$

Hence, $\deg(f(x))|n_k$, a contradiction to $\deg(f(x)) = n$ by Lemma 3.2. Therefore, $\mathrm{ord}(f(x)) \nmid q^{n_i} - 1 \quad (i = 1, 2, \ldots, s)$.

The following results can be implied by above two theorems.

Corollary 3.6. Let $f(x)$ be a polynomial over F_q of degree p, where p is a prime. Then $f(x)$ is irreducible if and only if:

(1) $\mathrm{ord}(f(x))|q^p - 1$;
(2) For every $c \in F_q, f(c) \neq 0$.

Corollary 3.7. Let $f(x)$ be a polynomial over F_q of degree $n = p_1 p_2$, where p_1 and p_2 are prime numbers. Then $f(x)$ is irreducible if and only if:

(1) $\mathrm{ord}(f(x))|q^n - 1$;
(2) For every $c \in F_q, f(c) \neq 0$;
(3) $\mathrm{ord}(f(x)) \nmid q^{p_1} - 1$ and $\mathrm{ord}(f(x)) \nmid q^{p_2} - 1$.

4 Conclusion

Irreducible polynomials over finite fields play an important role in computer algebra, coding theory and cryptography. This paper constructed the irreducible polynomials of arbitrary degree n over finite fields based on the number of the roots over the extension field and determined whether a polynomial over finite fields is irreducible or not in terms of the relationship between the order of a polynomial over finite fields and the order of the multiplicative group of the extension field. Furthermore, a relevant example was analyzed to show the specific construction procedures.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (61373150, 11501343), the Key Technologies R&D Program of Shaanxi Province (2013k0611), and the Fundamental Research Funds for the Central Universities (GK201603087).

References

1. Garefalakis, T., Kapetanakis, G.: A note on the Hansen-Mullen conjecture for self-reciprocal irreducible polynomials. Finite Fields Appl. **35**(4), 61–63 (2015)
2. Magamba, K., Ryan, J.A.: Counting irreducible polynomials of degree r over F_{q^n} and generating Goppa Codes using the lattice of subfields of $F_{q^{nr}}$. J. Discrete Math. **2014**, 1–4 (2014)
3. Ugolini, S.: Sequences of irreducible polynomials over odd prime fields via elliptic curve endomorphisms. J. Number Theor. **152**, 21–37 (2015)
4. Ri, W.H., Myong, G.C., Kim, R., et al.: The number of irreducible polynomials over finite fields of characteristic 2 with given trace and subtrace. Finite Fields Appl. **29**, 118–131 (2014)
5. Kaminski, M., Xing, C.: An upper bound on the complexity of multiplication of polynomials modulo a power of an irreducible polynomial. IEEE Trans. Inf. Theor. **59**(10), 6845–6850 (2013)
6. Fan, H.: A Chinese remainder theorem approach to bit-parallel GF(2^n) polynomial basis multipliers for irreducible trinomials. IEEE Trans. Comput. **65**(2), 1 (2016)
7. Kopparty, S., Kumar, M., Saks, M.: Efficient indexing of necklaces and irreducible polynomials over finite fields. In: Esparza, J., Fraigniaud, P., Husfeldt, T., Koutsoupias, E. (eds.) ICALP 2014. LNCS, vol. 8572, pp. 726–737. Springer, Heidelberg (2014)
8. Nechae, A.A., Popov, V.O.: A generalization of Ore's theorem on irreducible polynomials over a finite field. Discrete Math. Appl. **25**(4), 241–243 (2015)
9. Pollack, P.: Irreducible polynomials with several prescribed coefficients. Finite Fields Appl. **22**(7), 70–78 (2013)
10. Kyuregyan, M.K.: Recurrent methods for constructing irreducible polynomials over F q of odd characteristics. Finite Fields Appl. **12**(3), 357–378 (2006)
11. Kyuregyan, M.K., Kyureghyan, G.M.: Irreducible compositions of polynomials over finite fields. Des. Codes Cryptogr. **61**(3), 301–314 (2011)
12. Abrahamyan, S., Kyureghyan, M.: A recurrent method for constructing irreducible polynomials over finite fields. In: Gerdt, V.P., Koepf, W., Mayr, E.W., Vorozhtsov, E.V. (eds.) CASC 2011. LNCS, vol. 6885, pp. 1–9. Springer, Heidelberg (2011)
13. Abrahamyan, S., Alizadeh, M., Kyureghyan, M.K.: Recursive constructions of irreducible polynomials over finite fields. Finite Fields Appl. **18**(4), 738–745 (2012)
14. Kaliski, B.: Irreducible Polynomial. Springer, New York (2011)