Hindawi Publishing Corporation
Journal of Optimization
Volume 2013, Article ID 438152, 16 pages
http://dx.doi.org/10.1155/2013/438152

Research Article

Physics-Inspired Optimization Algorithms: A Survey

Anupam Biswas, K. K. Mishra, Shailesh Tiwari, and A. K. Misra

Department of Computer Science & Engineering, Motilal Nehru National Institute of Technology Allahabad, Allahabad 211004, India

Correspondence should be addressed to K. K. Mishra; mishrakrishn@gmail.com

Received 7 February 2013; Revised 22 May 2013; Accepted 24 May 2013

Academic Editor: Qingsong Xu

Natural phenomenon can be used to solve complex optimization problems with its excellent facts, functions, and phenomenon. In this paper, a survey on physics-based algorithm is done to show how these inspirations led to the solution of well-known optimization problem. The survey is focused on inspirations that are originated from physics, their formulation into solutions, and their evolution with time. Comparative studies of these noble algorithms along with their variety of applications have been done throughout this paper.

1. Introduction

Leonid Kantorovich introduced linear programming for optimizing production in plywood industry in 1939 and probably it was the first time the term optimization of a process was used, though Fermat and Lagrange used calculus for finding optima and Newton and Gauss proposed methods for moving towards an optimum. Every technological process has to achieve optimality in terms of time and complexity and this led the researchers to design and obtain best possible or better solutions. In previous studies, several mathematical solutions were provided by various researchers such as LP [1], NLP [2] to solve optimization problems. The complexity of the proposed mathematical solutions is very high which requires enormous amount of computational work. Therefore, alternative solutions with lower complexity are appreciated. With this quest, nature-inspired solutions are developed such as GA [3], PSO [4], SA [5], and HS [6]. These nature-inspired metaheuristic solutions became very popular as the algorithms provided are much better in terms of efficiency and complexity than mathematical solutions. Generally, these solutions are based on biological, physical, and chemical phenomenon of nature.

In this paper, the algorithms inspired by the phenomenon of physics are reviewed, surveyed, and documented. This paper mainly focuses on the following issues:

(i) most inspirational facts and phenomena,

(ii) their formulation into a solution,

(iii) parameters considered for this formulation,

(iv) effectiveness of these parameters,

(v) variation with time in inspiration,

(vi) other biological influences,

(vii) convergence, exploration, and exploitation,

(viii) Various applications.

The rest of the paper is organized as follows. Section 2 overviews the history of physics-inspired algorithms and also the description of few major algorithms. In Section 3 a correlative study of these major algorithms is done on the basis of their inspirational theory and formulation method. Various parameters used in these algorithms along with their variants and respective applications are also discussed in this section. In Section 4, finally the overall study is concluded.

2. Historical Study

Both simplicity and efficiency attract researchers towards natural phenomenon, resulting in some popular algorithms such as GA [3] based on Darwin's principle of survival of the fittest, SA [5] in 1983 based on the annealing process of metal, PSO [4] in 1995 based on the behavior of fishes and birds swarms, and HS [6] in 2001 based on the way a musician adjusts instruments to obtain good harmony. Richard Feynman's proposal of quantum computing system [7, 8], inspired

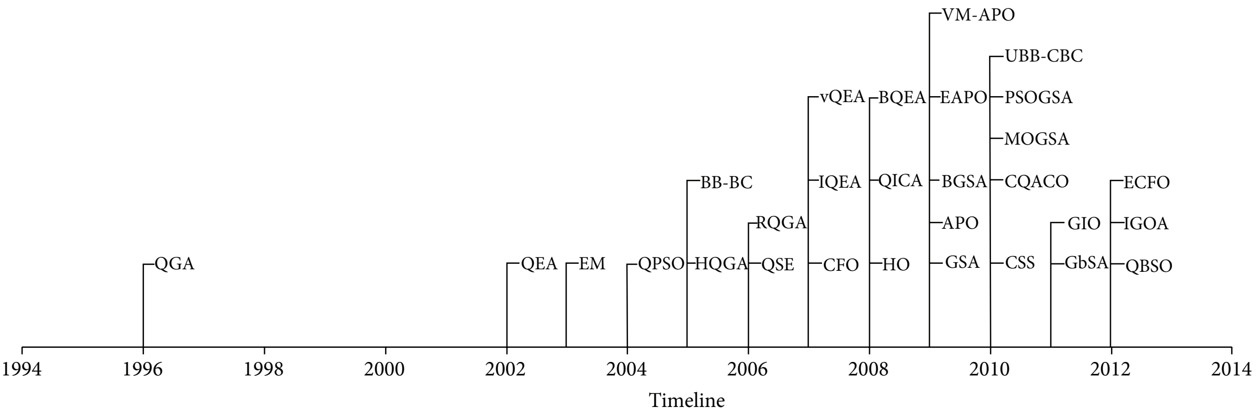


Figure 1: Evolution of physics-inspired optimization algorithms.

by quantum mechanics in 1982, paved way for physics-inspired optimization algorithms. With this, the concept of quantum computing was developed and in 1995 Narayanan and Moore [9] proposed Quantum-Inspired Genetic Algorithm (QGA). This is the beginning of physics-inspired optimization algorithms. After half a decade later, in 2002, Han and Kim proposed Quantum-Inspired Evolutionary Algorithm (QEA). In 2004 Quantum-Inspired Particle Swarm Optimization (QPSO) was proposed by Sun et al. [10] and in 2007 another swarm-based Quantum Swarm Evolutionary Algorithm (QSE) was proposed by Wang et al. [11]. Apart from the quantum mechanics, other principles and theorems of physics also begun to draw the attention of researchers. In 2003, Birbil and Fang [12] proposed Electromagnetism-like (EM) mechanism based on the superposition principle of electromagnetism. Big Bang-Big Crunch (BB-BC) [13] based on hypothetical theorem of creation and destruction of the universe was proposed in 2005. Based on Newton's gravitational law and laws of motion algorithms emerged such as CFO [14] by Formato in 2007, GSA by Rashedi et al. [15], APO by Xie et al. [16] in 2009, and GIO by Flores et al. [17] in 2011. Hysteretic Optimization (HO) [18] based on demagnetization process was proposed in 2008. In 2010, Kaveh and Talatahari proposed CSS [19] based on electrostatic theorems such as Coulomb's law, Gauss's law, and superposition principle from electrostatics and Newton's laws of motion. In 2011, Shah-Hosseini proposed Spiral Galaxy-Based Search Algorithm (GbSA) [20]. Jiao et al. [21] proposed QICA in 2008 based on quantum theory and immune system. Li et al. [22] proposed CQACO based on quantum theory and ant colony in 2010. Most recently in 2012, Zhang et al. [23] proposed IGOA based on gravitational law and immune system, and Jinlong and Gao [24] proposed QBSO based on quantum theory and bacterial forging. These major algorithms along with their modified, improved, and hybrid versions along with the year of proposal are shown in Figure 1. We have categorized these algorithms with their variants and their notion of inspiration as follows:

(A) Newton's gravitational law
  (i) Pure physics
    (1) CFO
      (a) Variant (pure physics)
        (1) ECFO
    (2) APO
      (a) Variant (pure physics)
        (1) EAPO
        (2) VM-APO
    (3) GSA
      (a) Variant (pure physics)
        (1) BGSA
        (2) MOGSA
      (b) Variant (Semiphysics)
        (1) PSOGSA
    (4) GIO
  (ii) Semiphysics
    (1) IGOA
(B) Quantum mechanics
  (i) Pure physics
    (1) QGA
      (a) Variant (pure physics)
        (1) RQGA
        (2) QGO
      (b) Variant (semiphysics)
        (1) HQGA
    (2) QEA
      (a) Variant (pure physics)
        (1) BQEA
        (2) vQEA
        (3) IQEA
  (ii) Semiphysics
    (1) QPSO
    (2) QSE
    (3) QICA
    (4) CQACO
    (5) QBSO
(C) Universe theory
  (i) Pure physics
    (1) BB-BC
      (a) Variant (pure physics)
        (1) UBB-CBC
    (2) GbSA

(D) Electromagnetism
(i) Pure physics
(1) EM
(E) Glass demagnetization
(i) Pure physics
(1) HO
(F) Electrostatics
(i) Pure physics
(1) CSS.

3. Algorithms

3.1. Newton's-Gravitation-Law-Based Algorithms

3.1.1. CFO. CFO [14] is inspired by the theory of particle kinematics in gravitational field. Newton's universal law of gravitation implies that larger particles will have more attraction power as compared to smaller particles. Hence, smaller ones will be attracted towards the larger ones. As a result, all smaller particles will be attracted towards the largest particle. This largest particle can be resembled as global optimum solution in case of optimization. To mimic this concept in CFO, a set of solutions is considered as probes on the solution space. Each probe will experience gravitational attraction due to the other. Vector acceleration experienced by probe p with respect to other probes at iteration t is given by the equation below:

$$a_t^p = G \sum_{\substack{k=1 \\ k \neq p}}^{N} U\left(F_t^k - F_t^p\right) \cdot \left(F_t^k - F_t^p\right)^{\alpha} \times \frac{\left(R_t^k - R_t^p\right)}{\left\| R_t^k - R_t^p \right\|^{\beta}}. \quad (1)$$

Here, G is CFO's gravitational constant, R_t^p and F_t^p are the position of a probe p and objective function value at that position, respectively, at iteration t, R_t^k and F_t^k are the position of all other probe, and objective function value at that position, respectively, at iteration t, $U(\cdot)$ is the unit step function. The CFO exponents α and β, by contrast, have no analogues in nature but these exponents provide flexibility to the algorithm. These parameters have drastic effect on overall exploration and convergence of the algorithm. The algorithm does not have any apparent mechanism for exploitation.

In this equation, $U(F_t^k - F_t^p) \cdot (F_t^k - F_t^p)^{\alpha}$ defines CFO's mass which is analogous to real objects mass in space.

The a_t^p causes the probe p to move from position R_t^p to R_{t+1}^p and the new location is obtained by the following equation:

$$R_{t+1}^p = R_t^p + \frac{1}{2} a_t^p \Delta t^2. \quad (2)$$

Here, Δt is the time interval between iterations. Recently, Ding et al. proposes an extended version of CFO, namely, ECFO [25]. Applications of this algorithm are neural network [26] and antenna applications [27, 28].

3.1.2. APO. APO [16] is based on the concept of artificial physics or physicomimetics [29], which was applied to robots. Analogous to Newton's gravitation law a new kind of force law is defined as follows:

$$F = G \frac{m_1 m_2}{r^p}, \quad (3)$$

where F is the force exerted between particles m_1 and m_2 in a hypothetical universe, G is gravitational constant, and r is the distance between particles m_1 and m_2. Unlike real universe, the value of p is not always equal to 2; instead it varies from −5 to +5.

Mass is defined in APO as follows:

$$m_i = e^{(f(x_{\text{best}}) - f(x_i))/(f(x_{\text{worst}}) - f(x_{\text{best}}))}. \quad (4)$$

Considering value of $p = -1$ in (3), the force in APO is defined as follows:

$$F_{ij}^k = \begin{cases} G m_i m_j \left(x_j^k - x_i^k\right) & \text{if } f\left(X_j\right) < f\left(X_i\right) \\ -G m_i m_j \left(x_j^k - x_i^k\right) & \text{if } f\left(X_j\right) \geq f\left(X_i\right), \end{cases} \quad (5)$$

where F_{ij}^k is the kth component of force exerted on particle i by particle j, x_i^k and x_j^k are the kth dimension of particles i and j, respectively. The kth component of the *total* force F_i^k exerted on particle i by all other particles is given by the following:

$$F_i^k = \sum_{j=1}^{N} F_{ij}^k \quad \forall i \neq \text{best}. \quad (6)$$

Velocity and positions of particles are updated with following equation:

$$v_i^k(t+1) = w v_i^k(t) + \frac{\lambda F_i^k}{m_i}, \quad (7)$$
$$x_i^k(t+1) = x_i^k(t) + v_i^k(t+1),$$

where λ is uniformly distributed random variable in $[0, 1]$, w is user-defined weight $0 < w < 1$.

Main exploitation and convergence component of APO algorithm is the computation of force exerted on each particle by others. Overall exploration of algorithm is controlled by the weight parameter w. The parameter λ is actually for putting limitation to convergence. But, due to randomness, it also serves for exploration. To overcome the lack of convergence component, an extended version of APO is proposed in [30], where individual particle's best position is tracked in iteration and utilized in velocity updating. A vector model of APO is defined in [31].

3.1.3. GSA. GSA [15] is inspired by Newton's law of universal gravitation and law of motion. In addition to this, another fact of physics is also considered, according to which the actual value of gravitational constant G depends on the actual age of the universe. So G at time t can be expressed as follows:

$$G(t) = G(t_0) \times \left(\frac{t_0}{t}\right)^{\beta}, \quad \beta < 1, \quad (8)$$

where $G(t_0)$ is the value of the gravitational constant at the first cosmic quantum-interval of time t_0, β is a time-dependent exponent.

In GSA, the solution space is considered as an imaginary universe. Every point in solution space is considered as an agent having mass. To compute mass of any agent i, a parameter $m_i(t)$ is computed. The parameter $m_i(t)$ and mass $M_i(t)$ of agent i are computed as follows:

$$m_i(t) = \frac{\mathrm{fit}_i(t) - \mathrm{worst}(t)}{\mathrm{best}(t) - \mathrm{worst}(t)}, \qquad M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)}, \tag{9}$$

where $\mathrm{fit}_i(t)$ is the fitness value of the agent i at time t.

Force exerted on each considered agent is computed as follows:

$$F_{ij}^{d} = G(t)\frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \varepsilon}\left(x_j^d(t) - x_i^d(t)\right), \qquad R_{ij} = \left\| X_i(t)\, X_j(t) \right\|_2, \qquad F_i^d(t) = \sum_{j=1, j \neq i}^{N} \mathrm{rand}_j F_{ij}^d(t), \tag{10}$$

where F_{ij}^d is the force acting on mass M_i and mass M_j at time t, M_{aj} is the active gravitational mass related to agent j, M_{pi} is the passive gravitational mass related to agent i, $G(t)$ is gravitational constant at time t, ε is a small constant, $R_{ij}(t)$ is the Euclidian distance between two agents i and j, $F_i^d(t)$ is the total force that acts on agent i in a dimension d at time t, and rand_j is a random number in the interval $[0, 1]$.

Acceleration of any agent i at time t in direction d is computed with equation given below:

$$a_i^d(t) = \frac{F_i^d}{M_i(t)}. \tag{11}$$

The next position of each agent and at which velocity they will move is calculated as follows:

$$v_i^d(t+1) = \mathrm{rand}_i \times v_i^d(t) + a_i^d(t), \qquad x_i^d(t+1) = x_i^d(t) + v_i^d(t+1). \tag{12}$$

The concept of variable gravitational constant G provides a good mechanism for convergence to the algorithm. As in subsequent iterations the value of G gradually increases, attraction force experienced by each agent also increases. Thus, agents converge towards the better agents with incremental attraction. However, the effect of attraction force is controlled by a random parameter. This random control of force ensures exploitation as well as exploration. Another random parameter used in velocity updating also implies exploration of search space.

In [32] binary version of GSA is proposed, multiobjective GSA [33] is proposed by Mirjalili and Hashim, and a hybrid of PSO and GSA is proposed in [34].

Applications of GSA algorithm are in power system [35–42], economic dispatch problem [43, 44], Wessinger's equation [45], fuzzy system [46, 47], forecasting of future oil demand [48], slope stability analysis [49], clustering [50–52], prototype classification [53], feature selection [54], web services [55], PID controller [56], antenna application [47], and so forth.

3.1.4. IGOA. IGOA [23] algorithm is an improved version of GSA [15]. The gravitation-law-based algorithm GSA can easily fall into local optimum solution and convergence rate is also comparatively slow [57]. To overcome these problems, IGOA introduces new operators which are inspired from biological immune system (BIS) [58]. In BIS mainly two kinds of activities take place, activities of antigens and activities of antibody. An antigen can only be negotiated with corresponding right antibody which comes from mother during birth. But for an unknown antigen, BIS also can act accordingly by learning. That means BIS has immune memory and antibody diversity. IGOA mimics this mechanism to avoid falling into local optimum. In this case, local optimum is similar to the unknown antigen in BIS. In IGOA vaccination and memory antibody replacement is used to improve the convergence speed and antibody diversity mechanism to catch the diversity of solution space along with GSA. IGOA is a newly proposed algorithm and not yet applied in any real-life application.

3.1.5. GIO. GIO [17] algorithm is similar to GSA [15] and CSS [19] where each point in the search space is assigned mass and charges, respectively. Although perspective of assigning masses or charges to each point is similar, the way of assignment and notion is different. CSS is inspired from electrostatic dynamics law, whereas GSA is inspired from Newton's gravitational laws and laws of motion. GIO is also inspired from Newton's law but unlike GSA this algorithm keeps hypothetical gravitational constant G as constant. Force exerted between two bodies is computed as follows:

$$F_{ij} = \frac{M\left(f\left(B_i\right)\right) \cdot M\left(f\left(B_j\right)\right)}{\left|B_i - B_j\right|^2}\widehat{B}_{ij}, \tag{13}$$

where B_i is the position of the ith body and B_j is that of jth body, F_{ij} is exerting force on the mass B_i; $|B_i - B_j|$ is the Euclidean distance and $\widehat{B}_{ij}$ is the unit vector between bodies B_i and B_j; $f(B_i)$ is the fitness of body B_i, $M(f(B_i))$ is corresponding mass of body B_i and is computed as follows:

$$M\left(f\left(B_i\right)\right) = \left(\frac{f\left(B_i\right) - \min f(B)}{\max f(B) - \min f(B)} \times \left(1 - \mathrm{mapMin}\right) + \mathrm{mapMin}\right)^2, \tag{14}$$

where $\min f(B)$ is the minimum fitness value of the positions of the bodies so far, $\max f(B)$ is the maximum fitness value of the positions of the bodies so far. mapMin is a constant used to limit the fitness value $f(B_i)$ to a mass in the interval [mapMin, 1). As each body are interacts with other bodies so resultant force acting on body B_i is computed as follows:

$$F_i = \sum_{j=1}^{n} \frac{M(f(B_i)) \cdot M\left(f\left(B_j^{\text{best}}\right)\right)}{\left|B_i - B_j^{\text{best}}\right|^2} B_i \widehat{B}_j^{\text{best}}. \tag{15}$$

Velocity with which a body will move to its new position is computed as follows:

$$V_{t+1} = \chi(V_t + R \cdot C \cdot B_k), \tag{16}$$

where V_t is the current velocity of B_i, R is a random real number generated in the range of [0, 1), C is the gravitational interaction coefficient, B_k is the displacement of body B_i and is computed with (17) and χ is the inertia constraint, and is computed with (18):

$$B_k = \sqrt{\frac{M(f(B_i))}{|F_i|}} \widehat{F}_i, \tag{17}$$

$$\chi = \frac{2k}{\left|2 - \phi - \sqrt{\phi^2 - 4\phi}\right|}. \tag{18}$$

In (18), k is an arbitrary value in the range (0, 1], $\phi = C_1 + C_2 > 4$, where C_1 and C_2 are the cognitive and the gravitational interaction constants, respectively.

New position of a body is obtained by adding the computed velocity corresponding to it. Formula given above for computing velocity is for unimodal optimization problems, which is further modified for multimodal optimization problems as follows:

$$V_{t+1} = \chi\left(V_t + C_1 \cdot R_1 \cdot \left(B^{\text{best}} - B\right) + C_2 \cdot R_2 \cdot B_k\right). \tag{19}$$

Here, R_1 and R_2 are real random numbers in the range [0, 1).

New position for next iteration is obtained by adding the updated velocity with current body as follows:

$$B_{t+1} = B_t + V_{t+1}. \tag{20}$$

Certain precaution has been taken during resultant force computation in order to avoid numerical errors. The force between masses $M(B_i)$ and $M(B_j^{\text{best}})$ is computed only if $|B_i - B_j| \geq 1 \times 10^{-5}$. In order to avoid division by 0, B_k is computed only if for a body B_i resultant force $F_i > 0$.

The concept of inertia constant χ is similar to the concept of constriction parameter in constricted PSO [59–61]. Exploration of a body in GIO is controlled by this parameter. Exploration, exploitation, and convergence are ensured by computation of mass and resultant force. The inertia constant χ also helps in convergence. Though Flores et al. [17] shows GIO's superiority over PSO in multimodal problems but it has not been yet applied in any real-life application.

3.2. Quantum-Mechanics-Based Algorithms

3.2.1. QGA. According to quantum mechanics, electrons are moving around the nucleus in an arc path, known as orbits. Depending on the angular momentum and energy level, electrons are located in different orbits. An electron in lower level orbit can jump to higher level orbit by absorbing certain amount of energy; similarly higher level electron can jump to lower energy level by releasing certain amount of energy. This kind of jumping is considered as discrete. There is no intermediate state in between two energy levels. The position where an electron lies on the orbit is unpredictable; it may lie at any position in orbit at a particular time. Unpredictability of electron's position is also referred as superposition of electron.

In classical computing, a bit is represented either by 0 or 1, but in quantum computing this is termed as qubit. State of a qubit can be 0 or 1 or both at the same time in superposition state. This superposition of qubit mimics the superposition of electrons or particles. State of qubit at any particular time is defined in terms of probabilistic amplitudes. The position of an electron is described in terms of qubits by a vector called quantum state vector. A quantum state vector can be described with the equation given below:

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{21}$$

where α and β are complex numbers that specify the probability amplitudes of obtaining the qubit in "0" state and in "1" state, respectively. In this case, the value of α and β always satisfies the equation $|\alpha|^2 + |\beta|^2 = 1$. For n positions of electrons, states can be described by n state vectors. These n positions of an electron can be known simultaneously.

QGA [9] utilized the concept of parallel universe in GA [3] to mimic quantum computing. According to this parallel universe interpretation, each universe contains its own version of population. All populations follow the same rules, but one universe can interfere in population of other universe. This interference occurs as in the form of a different kind of crossover called interference crossover, which provides good exploration capability to the algorithm. In QGA, all the solutions are encoded using superposition and all of these solutions may not be valid, which creates problems during implementation of crossover. Udrescu et al. propose RQGA [62], which provides a mechanism to overcome this problem. Hybrid versions [63] merge QGA with permutation-based GA and [64] merge QGA with real-valued GA. Malossini and Calarco propose QGOA [65] very similar to QGA with special quantum-based selection and fitness evaluation methods.

Many applications have been developed in recent years on the basis of this algorithm such as structural aligning [66], clustering [67, 68], TSP [69], combinatorial optimization problem [70], web information retrieval [71], computational grid [72], software testing [73], dynamic economic dispatch [74], area optimization [75], operation prediction [76], computer networking [77, 78], PID controller [79], multivariate problem [80], course timetabling [81], minimal redact [82], image applications [83–86], smart antenna [87], hardware [88], fuzzy system [89, 90], neural network [91], and robot application [92].

3.2.2. QEA. Quantum bit and superposition of states are the main basis of this algorithm. QEA [93] is originally inspired by quantum computing, which itself is inspired by the quantum mechanics. In QEA, the state of a qubit or Q-bit is represented as pair of numbers (α, β) in a column matrix $\left[\begin{smallmatrix}\alpha\\ \beta\end{smallmatrix}\right]$, where $|\alpha|^2 + |\beta|^2 = 1$ and $|\alpha|^2$ gives the probability that the Q-bit will be found in the "0" state and $|\beta|^2$ gives the probability that the Q-bit will be found in the "1" state.

A Q-bit individual which is a string of Q-bits is defined as follows:

$$\left[\begin{array}{c|c|c|c} \alpha_1 & \alpha_2 & \cdots & \alpha_m \\ \beta_1 & \beta_2 & \cdots & \beta_m \end{array}\right], \tag{22}$$

where $|\alpha_i|^2 + |\beta_i|^2 = 1$, $i = 1, 2, \ldots, m$. With this Q-bit representation, a population set is formulated and operations are performed on that population. Zhang and Gao further improved this algorithm as IQEA [94], by introducing probability amplitude ratio $\gamma_\alpha = |\beta|/|\alpha|$ if $\alpha \neq 0$ and $\gamma_\beta = |\alpha|/|\beta|$ if $\beta \neq 0$ to define relative relationship between α and β. As quantum rotation gate is unable to cover the entire search space since it outputs discrete values, a mechanism for calculating rotation angle of quantum rotation gate is defined. Platel et al. propose versatile QEA [95], with introducing new concept of hitchhiking phenomenon into QEA with little bit elitism in updating parameters and P. Li and S. Li propose Bloch QEA [96] based on Bloch coordinates depicted by qubits. Here α and β are defined as $\cos(\theta/2)$ and $e^{i\varphi}\sin(\theta/2)$, respectively. This θ and φ define bloch points.

Applications of QEA-related algorithms are combinatorial optimization [97, 98], image segmentation [99], Knapsack Problems [100–102], resource optimization [103, 104], numerical optimization [105, 106], extrusion [107], unit commitment problem [108, 109], power system [110, 111], signaling [112], face identification [113, 114], financial data analysis [115], Option pricing model calibration [116, 117], stock market prediction [118], and so forth.

3.2.3. *QSE.*

QSE [11] takes the concepts from both QEA [93] and PSO [4]. Similar to PSO's swarm intelligent concept, quantum swarms are represented using Q-bits. Unlike QEA, representation of Q-bit in QSE changes probabilistic parameters. α and β are replaced with angular parameters $\sin\theta$ and $\cos\theta$, here θ is quantum angle. Q-bit $[\theta]$ is represented as $\left[\begin{smallmatrix}\sin\theta\\ \cos\theta\end{smallmatrix}\right]$, where $|\sin\theta|^2 + |\cos\theta|^2 = 1$. For m Q-bits, this can be represented as $\left[\begin{smallmatrix}\sin\theta_1 & \sin\theta_1 & \cdots & \sin\theta_1\\ \cos\theta_1 & \cos\theta_1 & \cdots & \cos\theta_1\end{smallmatrix}\right]$. Each bit position x_{ij}^t of each individual, at time t, is determined with the following

$$x_{ij}^t = \begin{cases} 1 & \text{if random } [0,1] > \left|\cos\theta_{ij}\right|^2 \\ 0 & \text{otherwise.} \end{cases} \tag{23}$$

Velocity is updated as in PSO. Another quantum-swarm-based PSO called QPSO was proposed by Sun et al. [10]. Unlike QSE state of particle is not determined by the probabilistic angular parameters. Here, state of particle is determined by a wave function $\Psi(x)$ as follows:

$$\Psi(x) = \frac{1}{\sqrt{L}} e^{-(\|c-x\|/L)}. \tag{24}$$

Here, c and x are the center or current best and current location vector, L is called creativity or imagination parameter of particle. Location vector is defined as:

$$x(t) = c \pm \frac{L}{2} \ln\left(\frac{1}{R}\right) \tag{25}$$

Here, R is a random number in range $[0, 1]$. The creativity parameter L is updated as follows:

$$L(t+1) = 2 \times \alpha \times |c - x(t)|. \tag{26}$$

Here, α is the creative coefficient and acts as main ingredient for convergence towards the optima. Huang et al. [119] have improved this later on by considering global best instead of current best.

Applications of these algorithms are flow shop scheduling [120], unit commitment problem [121, 122], neural network [123], power system [124–126], vehicle routing problem [127–129], engineering design [130, 131], mining association rules [132], and so forth.

3.2.4. *QICA.*

Basic concept of QICA [21] is Artificial Immune System's clonal selection, which is hybridized with the framework of quantum computing. Basic quantum representational aspect is similar to QEA [93]. QICA introduces some new operators to deal with premature convergence and diverse exploration. The clonal operator Θ is defined as follows:

$$\Theta(Q(t)) = \left[\Theta(q_1)\Theta(q_2)\cdots\Theta(q_n)\right]^T, \tag{27}$$

where $Q(t)$ is quantum population and $\Theta(q_i) = I_i q_i$, I_i is the identity matrix of dimensionality D_i, which is given by the following:

$$D_i = \left\lceil N_c \times \frac{A(q_i)}{\sum_{i=1}^{n} A(q_i)} \right\rceil. \tag{28}$$

Here, $A(\cdot)$ is function for adaptive self-adjustment and N_c is a given value relating to the clone scale. After cloning these are added to population.

The immune genetic operator consists of two main parts, that is, quantum mutation and recombination. Before performing quantum mutation, population is guided towards the best one by using following equation:

$$\begin{bmatrix} \alpha_i^u \\ \beta_i^u \end{bmatrix} = U(\theta_i) \times \begin{bmatrix} \alpha_i^p \\ \beta_i^p \end{bmatrix}, \tag{29}$$

where α_i^u and β_i^u are updated values, α_i^p and β_i^p are previous values of probabilistic coefficients, $U(\theta_i) = \left[\begin{smallmatrix} rl\sin\theta_i & -\sin\theta_i \\ \cos\theta_i & \cos\theta_i \end{smallmatrix}\right]$ is quantum rotation gate and θ_i is defined as follows:

$$\theta_i = k \times f(\alpha_i, \beta_i), \tag{30}$$

where k is a coefficient which determines the speed of convergence and the function $f(\alpha_i, \beta_i)$ determines the search direction. This updated population is mutated using quantum NOT gate as:

$$Q^{u}(t) = \sqrt{1 - |Q^{u}(t)|^{2}}. \tag{31}$$

Quantum recombination is similar to interference crossover in QGA [9]. Finally, the clonal selection operator selects the best one from the population observing the mutated one and original population. Clonal operator of QICA increases explorative power drastically in contrast to QEA.

3.2.5. CQACO. CQACO [22] merges quantum computing and ACO [133]. Ant's positions are represented by quantum bits. This algorithm also represents qubits similar to QEA [93] and QICA [21] and uses the concept of quantum rotation gate as in QICA. Similar to CQACO, Wang et al. [134] proposed quantum ant colony optimization. Another variant is proposed by You et al. [135] in 2010. Quantum concept with ACO provides good exploitation and exploration capability to these algorithms. Applications of these algorithms are fault diagnosis [136], robot application [137], and so forth.

3.2.6. QBSO. QBSO [24] is the newest among all the quantum-based algorithms. This algorithm is semiphysics-inspired, as it incorporates concepts of both bacterial forging and quantum theory. In other words, QBSO is an improved version of BFO [138]. As BFO is unable to solve discrete problems, QBSO deals with this problem by using quantum theory to adapt the process of the BFO to accelerate the convergence rate. BFO consists of chemotaxis, swarming, reproduction, elimination, and dispersal processes whereas QBSO consists mainly of three of them, chemotaxis, reproduction, and elimination dispersal. In QBSO also the qubit is defined as in (21). Quantum bacterium of S bacteria is represented in terms of the three processes, that is, bit position, chemotactic step, and reproduction loop.

The ith quantum bacterium's quantum jth bit at the mth chemotactic step of the nth reproduction loop in the pth elimination dispersal event is updated as follows:

$$\Psi_{ij}(m+1,n,p) = \begin{cases} \sqrt{1 - \left(\Psi_{ij}(m,n,p)\right)^{2}}, \\ \qquad \text{if } \left(\theta_{ij}^{i+1} = 0 \text{ and } \rho(m+1,n,p) < c_{1}\right) \\ \text{abs}\begin{pmatrix} \Psi_{ij}(m,n,p) \times \cos\theta_{ij}^{i+1} \\ -\sqrt{1 - \left(\Psi_{ij}(m,n,p)\right)^{2}} \times \sin\theta_{ij}^{i+1} \end{pmatrix}, \\ \qquad \text{otherwise,} \end{cases} \tag{32}$$

where θ_{ij}^{t+1} is the quantum rotation angle, which is calculated through (33), t is the iteration number of the algorithm, ρ_{ij} is uniform random number in range $[0,1]$, and c_1 is mutation probability which is a constant in the range $[0,1]$.

After updating quantum bacterium, the corresponding bit position in the population x_{ij} is updated with (34), where γ_{ij} is uniform random number between 0 and 1:

$$\theta_{ij}^{i+1} = e_{1}\left(b_{j}(m,n,p) - x_{ij}(m,n,p)\right). \tag{33}$$

Here, e_1 is attracting effect factor and b_j is the jth bit position of global optimal bit:

$$x_{ij}(m+1,n,p) = \begin{cases} 1, & \text{if } \gamma_{ij}(m+1,n,p) > \left(\Psi_{ij}(m+1,n,p)\right)^{2} \\ 0, & \text{if } \gamma_{ij}(m+1,n,p) \leq \left(\Psi_{ij}(m+1,n,p)\right)^{2}. \end{cases} \tag{34}$$

Fitness value of each point solution in population is represented as the health of that particular bacterium.

3.3. Universe-Theory-Based Algorithms

3.3.1. BB-BC. BB-BC [13] algorithm is inspired mainly from the expansion phenomenon of Big Bang and shrinking phenomenon of Big Crunch. The Big Bang is usually considered to be a theory of the birth of the universe. According to this theory all space, time, matter, and energy in the universe were once squeezed into an infinitesimally small volume and a huge explosion was carried out resulting in the creation of our universe. From then onwards, the universe is expanding. It is believed that this expansion of the universe is due to Big Bang. However, many scientists believe that this expansion will not continue forever and all matters would collapse into the biggest black hole pulling everything within it, which is referred as Big Crunch.

BB-BC algorithm has two phases, namely, Big Bang phase and Big Crunch phase. During Big Bang phase, new population is generated with respect to center of mass. During Big Crunch phase, the center of mass is computed which resembles black hole (gravitational attraction). Big Bang phase ensures exploration of solution space. Big Crunch phase fullfills necessary exploitation as well as convergence.

BB-BC algorithm suffers botching all candidates into a local optimum. If a candidate with best fitness value converges to an optima at the very beginning of the algorithm, then all remaining candidates follow that best solution and trapped into local optima. This happens because the initial population is not uniformly distributed in the solution space. So, this algorithm provides a methodology to obtain uniform initial population in BB-BC. Initially, that is, at level 1, two candidates C_1 and C_2 are considered; at level 2, C_1 and C_2 are subdivided into C_3 and C_4; at level 3, C_1 and C_2 are again divided into C_4, C_5, C_6, C_7, C_8, C_9, and C_{10} and so on. This kind of division continues until we get the required numbers of candidates for initial population. In this way at nth level, C_1 and C_2 are subdivided into $2^n - 2$ candidates and include in population. In addition to this, in Big Crunch phase chaotic map is introduced, which improves convergence speed of algorithm. In this Chaotic Big Crunch phase, next position of each candidate is updated as follows:

$$x_{i}^{\text{new}} = x_{c} \pm \frac{\alpha(t)\left(x_{\max} - x_{\min}\right)}{t}, \tag{35}$$

where $\alpha_{t+1} = cf(\alpha(t))$, $0 < \alpha(t) < 1$, here $cf(x)$ is a chaotic map or function. BB-BC with uniform population is called UBB-BC and with chaotic map is called BB-CBC. If both are used, then it is called UBB-CBC.

Applications of this algorithm are fuzzy system [139–141], target tracking [142, 143], smart home [144], course timetabling [145], and so forth.

3.3.2. GbSA. GbSA [20] is inspired by spiral arm of spiral galaxies to search its surrounding. This spiral movement recovers from botching into local optima. Solutions are adjusted with this spiral movement during local search as well. This algorithm has two components:

(1) SpiralChaoticMove,

(2) LocalSearch.

SpiralChaoticMove actually mimics the spiral arm nature of galaxies. It searches around the current solution by spiral movement. This kind of movement uses some chaotic variables around the current best solution. Chaotic variables are generated with formula $x_{n+1} = \lambda x_n(1 - x_n)$. Here, $\lambda = 4$ and $x_0 = 0.19$. In this way, if it obtains a better solution than the current solution, it immediately updates and goes for *LocalSearch* to obtain more suitable solution around the newly obtained solution. GbSA is applied to Principle Component Analysis (PCA). *LocalSearch* ensures exploitation of search space and *SpiralChaoticMove* provides good exploration mechanism of search space ensuring reachability of algorithm towards the global optimum solution.

3.4. Electromagnetism-Based Algorithms

3.4.1. Electromagnetism-Like: EM. EM [12] algorithm is based on the superposition principle of electromagnetism, which states that the force exerted on a point via other points is inversely proportional to the distance between the points and directly proportional to the product of their charges. Points in solution space are considered as particles. The charge of each point is computed in accordance with their objective function value. In classical physics, charge of a particle generally remains constant, but in this heuristic the charge of each point is not constant and changes from iteration to iteration. The charge q^i of each point i determines its power of attraction or repulsion. This charge of a particle is evaluated as follows:

$$q^i = e^{(-n((f(x^i)-f(x^{\text{best}}))/\sum_{k=1}^{m}(f(x^k)-f(x^{\text{best}}))))}, \quad \forall i, \tag{36}$$

where m is the total number of points and n is the number of dimensions. This formula shows that points having better objective values will possess higher charges. This heuristic does not use signs to indicate positive or negative charge as in case of electric charge. So, direction of force (whether attractive or repulsive force) is determined by the objective function values (fitness) of two particular points. If point x^j has better value than x^i, then corresponding force is considered as attractive otherwise repulsive. That means that x^{best} attracts all other points towards it. The total force F^i (attractive or repulsive) exerted on point i is computed by the following equation:

$$F^i = \sum_{j \neq i}^{m} \begin{cases} \left(x^j - x^i\right) \dfrac{q^i q^j}{\left\|x^j - x^i\right\|^2} \\ \quad \text{if } f\left(x^j\right) < f\left(x^i\right) \\ \left(x^i - x^j\right) \dfrac{q^i q^j}{\left\|x^j - x^i\right\|^2} \\ \quad \text{if } f\left(x^j\right) \geq f\left(x^i\right). \end{cases}, \quad \forall i. \tag{37}$$

After evaluating the total force vector F^i, the point i is moved in the direction of the force with a random step length as given in (38). Here, RNG is a vector whose components denote the allowed feasible movement towards the upper bound or the lower bound

$$\begin{gathered} x^i = x^i + \lambda \frac{F^i}{\left\|F^i\right\|} (\text{RNG}) \\ i = 1, 2, \ldots, m. \end{gathered} \tag{38}$$

EM algorithm provides good exploration and exploitation mechanism with computation of charge and force. Exploration and convergence of EM are controlled by the random parameter λ. Exploration is also controlled with RNG, by limiting movements of particles.

Debels et al. [146] propose a hybrid version of EM combining the concept of GA with EM.

Numerous applications are developed on the basis of this algorithm such as scheduling problems [147–150], course timetabling [151], PID controller [152], fuzzy system [153–155], vehicle routing problem [156], networking [157], inventory control [158], neural network [159, 160], TSP [161, 162], feature selection [163], antenna application [164], robotics application [165], flow path designing [166], and vehicle routing [167].

3.5. Glass-Demagnetization-Based Algorithms

3.5.1. HO. HO [18] is inspired by the demagnetization process of a magnetic sample. A magnetic sample comes to a very stable low-energy state called ground state, when it is demagnetized by an oscillating magnetic field of slowly decreasing amplitude. After demagnetization, the system are shakeup repeatedly to obtain improved result. HO simulates these two processes of magnetic sample to get low-energy state by repeating demagnetization followed by a number of shakeups.

The process of demagnetization is mainly for exploration and convergence. After exploring, better solutions are searched by performing a number of shake-up operations. The algorithm possesses two kinds of stopping conditions, firstly fixed number of shakeups n_s for each instance of a given size N and secondly required number of shakeups M_{req} to obtain the current low-energy state or global optimum. Besides this M_{req} repetition of shakeup in current low-energy state, minimum number of shakeups $n_{s\,\text{min}}$ is set to ensure that algorithm does not accept suboptimum too early. Similarly, the maximum number of shakeups $n_{s\max}$ is also set to avoid wasting time on hard situation. HO algorithm is applied to TSP [168], spin glasses [169], vehicle routing problem [170], protein folding [171], and so forth.

3.6. Electrostatics-Based Algorithms

3.6.1. CSS. This algorithm inherits Coulomb's law, Gauss's law and superposition principle from electrostatics, and the Newtonian laws of mechanics. CSS [19] deploys each solution as a Charged Particle (CP). If two charged particles having charges q_i and q_j reside at distance r_{ij}, then according to Coulomb's law electric force F_{ij} exerted between them is as follows:

$$F_{ij} = k_e \frac{q_i q_j}{r_{ij}^2}. \tag{39}$$

Here, k_e is a constant called the Coulomb constant. Now if q_i amount of charges is uniformly distributed within a sphere of radius "a" then electric field E_{ij} at a point outside the sphere is as follows:

$$E_{ij} = k_e \frac{q_i}{r_{ij}^2}. \tag{40}$$

The electric field at a point inside the sphere can be obtained using Gauss's law as follows:

$$E_{ij} = k_e \frac{q_i}{a^3} r_{ij}. \tag{41}$$

The resultant force on a charge q_j at position r_j due to the electric field of a charge q_i at position r_i can be expressed in vector form as:

$$F_{ij} = E_{ij} q_j \frac{r_i - r_j}{\left\| r_i - r_j \right\|}. \tag{42}$$

For multiple charged particles, this equation can be expressed as follows:

$$\begin{aligned} F_j = k_e q_j \sum_{i,i \neq j} \left(\frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2 \right) \\ \times \frac{r_i - r_j}{\left\| r_i - r_j \right\|} \begin{cases} i_1 = 1,\ i_2 = 0 \Longleftrightarrow r_{ij} < a \\ i_1 = 0,\ i_2 = 1 \Longleftrightarrow r_{ij} \geq a. \end{cases} \end{aligned} \tag{43}$$

To formulate the concept of charges into CSS algorithm, a set of charged particles are considered. Each point in the solution space is considered as possible positions of any charged particle. Charges in each charged particle is computed as follows:

$$q_i = \frac{\text{fit}(i) - \text{fitworst}}{\text{firbest} - \text{fitworst}}. \tag{44}$$

Distance among particles is computed with the following:

$$r_{ij} = \frac{\left\| X_i - X_j \right\|}{\left\| \left(X_i + X_j \right)/2 - X_{\text{best}} \right\| + \varepsilon}. \tag{45}$$

Radius of particle is computed with

$$a = 0.10 \times \max\left(\left\{ x_{i,\max} - x_{i,\min} \mid i = 1, 2, \ldots, n \right\}\right). \tag{46}$$

The value of the resultant electrical force acting on a charged particle is determined as follows:

$$F_j = q_j \sum_{i,i \neq j} \left(\frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2 \right) P_{ij} \left(X_i - X_j \right). \tag{47}$$

Here, P_{ij} defines the attractiveness or repulsiveness of the force exerted. A good particle may attract a bad one and similarly bad one can also attract good one. So, if bad one attracts good one, then it is not suitable for an optimization problem. The parameter P_{ij} limits these kinds of attractions as follows:

$$P_{ij} = \begin{cases} 1, & \text{if } \left(\dfrac{\text{fit}(i) - \text{fitbest}}{\text{fit}(j) - \text{fit}(i)} > \text{rand} \right) \\ & \quad \text{or } \left(\text{fit}(j) > \text{fit}(i) \right) \\ 0, & \text{otherwise.} \end{cases} \tag{48}$$

Again in Newtonian mechanics or classical mechanics the velocity v of a particle is defined as follows:

$$v = \frac{r_{\text{new}} - r_{\text{old}}}{t_{\text{new}} - t_{\text{old}}} = \frac{r_{\text{new}} - r_{\text{old}}}{\Delta t}. \tag{49}$$

Displacement from r_{old} to r_{new} position along with acceleration a can be expressed as follows:

$$r_{\text{new}} = \frac{1}{2} a \cdot \Delta t^2 + v_{\text{old}} \cdot \Delta t + r_{\text{old}}. \tag{50}$$

Newton's second law states that "the acceleration of an object is directly proportional to the net force acting on it and inversely proportional to its mass;" that is, $F = ma$, so r_{new} can be expressed as follows:

$$r_{\text{new}} = \frac{1}{2} \frac{F}{m} \cdot \Delta t^2 + v_{\text{old}} \cdot \Delta t + r_{\text{old}}. \tag{51}$$

In CSS movements due to the electric force exerted among those particles are measured and accordingly new positions of particles are updated. New position ($X_{j,\text{new}}$) of CP and with which velocity ($V_{j,\text{new}}$) will reach the position ($X_{j,\text{new}}$) is computed as follows:

$$\begin{gathered} X_{j,\text{new}} = \text{rand}_{j1} \cdot k_a \cdot \frac{F_j}{m_j} \cdot \Delta t^2 + \text{rand}_{j2} \cdot k_v \cdot \Delta t + X_{j,\text{old}}, \\ V_{j,\text{new}} = \frac{X_{j,\text{new}} - X_{j,\text{old}}}{\Delta t}. \end{gathered} \tag{52}$$

Here, k_a is the parameter related to the attracting forces and k_v is velocity coefficient. The effect of the pervious velocity and the resultant force acting on a charged particle can be decreased with parameter k_v or increased with parameter k_a. These parameters can be computed as follows:

$$k_v = 0.5\left(1 - \frac{\text{iter}}{\text{iter}_{\max}}\right),$$
$$k_a = 0.5\left(1 + \frac{\text{iter}}{\text{iter}_{\max}}\right), \tag{53}$$

where iter is the current iteration number and $\text{iter}_{\max}$ is the maximum number of iterations.

The CSS algorithm possesses good exploring as well as exploiting capability of solution domain. Exploitation of CP is mainly ensured by the resulting electric force F_j of any particle j. Handling of attractiveness and repulsiveness of resulting force of any CP with the noble concept of parameter P_{ij} is very effective for exploitation. However, whether CP is going to explore or exploit the search space depends on the parameters k_a and k_v. Higher value of k_a implies higher impact on resulting electric force, which results exploitation of search space. Whereas higher value of k_v implies high exploration. Initially, values of k_a and k_v are almost same, but gradually k_a increases and k_v decreases. Hence, at the beginning, the algorithm explores the search space. As in successive iterations k_a increases, gradually the effect of attraction of good solutions also increases. Thus, the algorithm ensures convergence towards better solutions. The algorithm does not suffer from premature convergence due high exploration at the beginning of the algorithm. However, since good solution attracts others, if initial set of CPs not uniformly distributed over solution space, then the algorithm may be trapped into any local optima,.

Applications of this algorithm are mainly related to structural engineering designs [172–175] and geometry optimization [176].

4. Conclusion

In this paper, we have categorically discussed various optimization algorithms that are mainly inspired by physics. Major areas covered by these algorithms are quantum theory, electrostatics, electromagnetism, Newton's gravitational law, and laws of motion. This study shows that most of these algorithms are inspired by quantum computing and significant numbers of applications are developed on the basis of them. Parallel nature of quantum computing perhaps attracts researchers towards quantum-based algorithms. Another most attractive area of physics for inspiration is Newton's gravitational laws and laws of motion. We have realized that hybridization of quantum computing and biological phenomenon draws most attention these days. As biological phenomenon suggests best strategies and quantum computing provide simultaneity to those strategies; so merging of both into one implies better result. In this paper, we have studied formational aspects of all the major algorithms inspired by physics. We hope, this study will definitely be beneficial for new researchers and motivate them to formulate great solutions from those inspirational theorems of physics to optimization problems.

Abbreviations

ACO:	Ant colony optimization
APO:	Artificial physics optimization
BB-BC:	Big bang-big crunch
BFO:	Bacterial forging optimization
BGSA:	Binary gravitational search algorithm
BIS:	Biological immune system
BQEA:	Binary Quantum-inspired evolutionary algorithm
CFO:	Central force optimization
CQACO:	Continuous quantum ant colony optimization
CSS:	Charged system search
EAPO:	Extended artificial physics optimization
ECFO:	Extended central force optimization
EM:	Electromagnetism-like heuristic
GA:	Genetic Algorithm
GbSA:	Galaxy-based search algorithm
GIO:	Gravitational interaction optimization
GSA:	Gravitational search algorithm
HO:	Hysteretic optimization
HQGA:	Hybrid quantum-inspired genetic algorithm
HS:	Harmony search
IGOA:	Immune gravitation inspired optimization algorithm
IQEA:	Improved quantum evolutionary algorithm
LP:	Linear programming
MOGSA:	Multiobjective gravitational search algorithm
NLP:	Nonlinear programming
PSO:	Particle swarm optimization
PSOGSA:	PSO gravitational search algorithm
QBSO:	Quantum-inspired bacterial swarming optimization
QEA:	Quantum-inspired evolutionary algorithm
QGA:	Quantum-inspired genetic algorithm
QGO:	Quantum genetic optimization
QICA:	Quantum-inspired immune clonal algorithm
QPSO:	Quantum-behaved particle swarm optimization
QSE:	Quantum swarm evolutionary algorithm
RQGA:	Reduced quantum genetic algorithm
SA:	Simulated annealing
TSP:	Travelling salesman problem
UBB-CBC:	Unified big bang-chaotic big crunch
VM-APO:	Vector model of artificial physics optimization
vQEA:	Versatile quantum-inspired evolutionary algorithm.

References

[1] http://en.wikipedia.org/wiki/Linear_programming#CITEREF-Vazirani2001.

[2] D. P. Bertsekas, *Nonlinear Programmingby*, Athena Scientific, Belmont, Mass, USA, 2nd edition, 1999.

[3] J. H. Holland, "Genetic algorithms and the optimal allocation of trials," *SIAM Journal on Computing*, vol. 2, no. 2, pp. 88–105, 1973.

[4] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of the IEEE International Conference on Neural Networks. IV*, pp. 1942–1948, December 1995.

[5] S. Kirkpatrick and M. P. Vecchi, “Optimization by simulated annealing,” *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[6] Z. W. Geem, J. H. Kim, and G. V. Loganathan, “A new heuristic optimization algorithm: harmony search,” *Simulation*, vol. 76, no. 2, pp. 60–68, 2001.

[7] R. P. Feynman, “Simulating physics with computers,” *International Journal of Theoretical Physics*, vol. 21, no. 6-7, pp. 467–488, 1982.

[8] R. P. Feynman, “Quantum mechanical computers,” *Foundations of Physics*, vol. 16, no. 6, pp. 507–531, 1986.

[9] A. Narayanan and M. Moore, “Quantum-inspired genetic algorithms,” in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC ’96)*, pp. 61–66, May 1996.

[10] J. Sun, W. Xu, and B. Feng, “A global search strategy of quantum-behaved particle swarm optimization,” in *Proceedings of the 2004 IEEE Conference on Cybernetics and Intelligent Systems*, vol. 1, pp. 111–116, December 2004.

[11] Y. Wang, X. Feng, Y. Huang et al., “A novel quantum swarm evolutionary algorithm and its applications,” *Neurocomputing*, vol. 70, no. 4–6, pp. 633–640, 2007.

[12] S. I. Birbil and S. Fang, “An electromagnetism-like mechanism for global optimization,” *Journal of Global Optimization*, vol. 25, no. 3, pp. 263–282, 2003.

[13] O. K. Erol and I. Eksin, “A new optimization method: Big Bang-Big Crunch,” *Advances in Engineering Software*, vol. 37, no. 2, pp. 106–111, 2006.

[14] R. A. Formato, “Central force optimization: a new metaheuristic with applications in applied electromagnetics,” *Progress in Electromagnetics Research*, vol. 77, pp. 425–491, 2007.

[15] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, “GSA: a gravitational search algorithm,” *Information Sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.

[16] L. Xie, J. Zeng, and Z. Cui, “General framework of artificial physics optimization algorithm,” in *Proceedings of the World Congress on Nature and Biologically Inspired Computing (NaBIC ’09)*, pp. 1321–1326, IEEE, December 2009.

[17] J. Flores, R. López, and J. Barrera, “Gravitational interactions optimization,” in *Learning and Intelligent Optimization*, pp. 226–237, Springer, Berlin, Germany, 2011.

[18] K. F. Pál, “Hysteretic optimization for the Sherrington-Kirkpatrick spin glass,” *Physica A*, vol. 367, pp. 261–268, 2006.

[19] A. Kaveh and S. Talatahari, “A novel heuristic optimization method: charged system search,” *Acta Mechanica*, vol. 213, no. 3, pp. 267–289, 2010.

[20] H. Shah-Hosseini, “Principal components analysis by the galaxy-based search algorithm: a novel metaheuristic for continuous optimisation,” *International Journal of Computational Science and Engineering*, vol. 6, no. 1-2, pp. 132–140, 2011.

[21] L. Jiao, Y. Li, M. Gong, and X. Zhang, “Quantum-inspired immune clonal algorithm for global optimization,” *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 38, no. 5, pp. 1234–1253, 2008.

[22] W. Li, Q. Yin, and X. Zhang, “Continuous quantum ant colony optimization and its application to optimization and analysis of induction motor structure,” in *Proceedings of the IEEE 5th International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA ’10)*, pp. 313–317, September 2010.

[23] Y. Zhang, L. Wu, Y. Zhang, and J. Wang, “Immune gravitation inspired optimization algorithm,” in *Advanced Intelligent Computing*, pp. 178–185, Springer, Berlin, Germany, 2012.

[24] C. Jinlong and H. Gao, “A quantum-inspired bacterial swarming optimization algorithm for discrete optimization problems,” in *Advances in Swarm Intelligence*, pp. 29–36, Springer, Berlin, Germany, 2012.

[25] D. Ding, D. Qi, X. Luo, J. Chen, X. Wang, and P. Du, “Convergence analysis and performance of an extended central force optimization algorithm,” *Applied Mathematics and Computation*, vol. 219, no. 4, pp. 2246–2259, 2012.

[26] R. C. Green II, L. Wang, and M. Alam, “Training neural networks using central force optimization and particle swarm optimization: insights and comparisons,” *Expert Systems with Applications*, vol. 39, no. 1, pp. 555–563, 2012.

[27] R. A. Formato, “Central force optimization applied to the PBM suite of antenna benchmarks,” 2010, http://arxiv.org/abs/1003.0221.

[28] G. M. Qubati, R. A. Formato, and N. I. Dib, “Antenna benchmark performance and array synthesis using central force optimisation,” *IET Microwaves, Antennas and Propagation*, vol. 4, no. 5, pp. 583–592, 2010.

[29] D. F. Spears, W. Kerr, W. Kerr, and S. Hettiarachchi, “An overview of physicomimetics,” in *Swarm Robotics*, vol. 3324 of *Lecture Notes in Computer Science: State of the Art*, pp. 84–97, Springer, Berlin, Germany, 2005.

[30] L. Xie and J. Zeng, “An extended artificial physics optimization algorithm for global optimization problems,” in *Proceedings of the 4th International Conference on Innovative Computing, Information and Control (ICICIC ’09)*, pp. 881–884, December 2009.

[31] L. Xie, J. Zeng, and Z. Cui, “The vector model of artificial physics optimization algorithm for global optimization problems,” in *Intelligent Data Engineering and Automated Learning—IDEAL 2009*, pp. 610–617, Springer, Berlin, Germany, 2009.

[32] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, “BGSA: binary gravitational search algorithm,” *Natural Computing*, vol. 9, no. 3, pp. 727–745, 2010.

[33] H. R. Hassanzadeh and M. Rouhani, “A multi-objective gravitational search algorithm,” in *Proceedings of the 2nd International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN ’10)*, pp. 7–12, July 2010.

[34] S. Mirjalili and S. Z. M. Hashim, “A new hybrid PSOGSA algorithm for function optimization,” in *Proceedings of the International Conference on Computer and Information Application (ICCIA ’10)*, pp. 374–377, December 2010.

[35] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, and M. Farsangi, “Allocation of static var compensator using gravitational search algorithm,” in *Proceedings of the 1st Joint Congress on Fuzzy and Intelligent Systems*, pp. 29–31, 2007.

[36] B. Shaw, V. Mukherjee, and S. P. Ghoshal, “A novel opposition-based gravitational search algorithm for combined economic and emission dispatch problems of power systems,” *International Journal of Electrical Power and Energy Systems*, vol. 35, no. 1, pp. 21–33, 2012.

[37] S. Duman, U. Guvenc, Y. Sonmez, and N. Yorukeren, “Optimal power flow using gravitational search algorithm,” *Energy Conversion and Management*, vol. 59, pp. 86–95, 2012.

[38] P. Purwoharjono, M. Abdillah, O. Penangsang, and A. Soeprijanto, “Voltage control on 500 kV Java-Bali electrical power system for power losses minimization using gravitational search

algorithm," in *Proceedings of the 1st International Conference on Informatics and Computational Intelligence (ICI '11)*, pp. 11–17, December 2011.

[39] S. Duman, Y. Soonmez, U. Guvenc, and N. Yorukeren, "Optimal reactive power dispatch using a gravitational search algorithm," *IET Generation, Transmission & Distribution*, vol. 6, no. 6, pp. 563–576, 2012.

[40] S. Mondal, A. Bhattacharya, and S. Halder, "Solution of cost constrained emission dispatch problems considering wind power generation using gravitational search algorithm," in *Proceedings of the International Conference on Advances in Engineering, Science and Management (ICAESM '12)*, pp. 169–174, IEEE, 2012.

[41] A. Bhattacharya and P. K. Roy, "Solution of multi-objective optimal power flow using gravitational search algorithm," *IET Generation, Transmission & Distribution*, vol. 6, no. 8, pp. 751–763, 2012.

[42] S. Duman, Y. Sonmez, U. Guvenc, and N. Yorukeren, "Application of gravitational search algorithm for optimal reactive power dispatch problem," in *Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications (INISTA '11)*, pp. 1–5, IEEE, June 2011.

[43] S. Duman, U. Guvenc, and N. Yurukeren, "Gravitational search algorithm for economic dispatch with valve-point effects," *International Review of Electrical Engineering*, vol. 5, no. 6, pp. 2890–2895, 2010.

[44] S. Duman, A. B. Arsoy, and N. Yorukeren, "Solution of economic dispatch problem using gravitational search algorithm," in *Proceedings of the 7th International Conference on Electrical and Electronics Engineering (ELECO '11)*, pp. I54–I59, December 2011.

[45] M. Ghalambaz, A. R. Noghrehabadi, M. A. Behrang, E. Assareh, A. Ghanbarzadeh, and N. Hedayat, "A Hybrid Neural Network and Gravitational Search Algorithm (HNNGSA) method to solve well known Wessinger's equation," *World Academy of Science, Engineering and Technology*, vol. 73, pp. 803–807, 2011.

[46] R. Precup, R. David, E. M. Petriu, S. Preitl, and M. Radac, "Gravitational search algorithm-based tuning of fuzzy control systems with a reduced parametric sensitivity," in *Soft Computing in Industrial Applications*, pp. 141–150, Springer, Berlin, Germany, 2011.

[47] R. Precup, R. David, E. M. Petriu, S. Preitl, and M. Radac, "Fuzzy control systems with reduced parametric sensitivity based on simulated annealing," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 8, pp. 3049–3061, 2012.

[48] M. A. Behrang, E. Assareh, M. Ghalambaz, M. R. Assari, and A. R. Noghrehabadi, "Forecasting future oil demand in Iran using GSA (Gravitational Search Algorithm)," *Energy*, vol. 36, no. 9, pp. 5649–5654, 2011.

[49] M. Khajehzadeh, M. R. Taha, A. El-Shafie, and M. Eslami, "A modified gravitational search algorithm for slope stability analysis," *Engineering Applications of Artificial Intelligence*, vol. 25, 8, pp. 1589–1597, 2012.

[50] A. Hatamlou, S. Abdullah, and H. Nezamabadi-Pour, "Application of gravitational search algorithm on data clustering," in *Rough Sets and Knowledge Technology*, pp. 337–346, Springer, Berlin, Germany, 2011.

[51] M. Yin, Y. Hu, F. Yang, X. Li, and W. Gu, "A novel hybrid K-harmonic means and gravitational search algorithm approach for clustering," *Expert Systems with Applications*, vol. 38, no. 8, pp. 9319–9324, 2011.

[52] C. Li, J. Zhou, B. Fu, P. Kou, and J. Xiao, "T-S fuzzy model identification with a gravitational search-based hyperplane clustering algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 2, pp. 305–317, 2012.

[53] A. Bahrololoum, H. Nezamabadi-Pour, H. Bahrololoum, and M. Saeed, "A prototype classifier based on gravitational search algorithm," *Applied Soft Computing Journal*, vol. 12, no. 2, pp. 819–825, 2012.

[54] J. P. Papa, A. Pagnin, S. A. Schellini et al., "Feature selection through gravitational search algorithm," in *Proceedings of the 36th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '11)*, pp. 2052–2055, May 2011.

[55] B. Zibanezhad, K. Zamanifar, N. Nematbakhsh, and F. Mardukhi, "An approach for web services composition based on QoS and gravitational search algorithm," in *Proceedings of the International Conference on Innovations in Information Technology (IIT '09)*, pp. 340–344, IEEE, December 2009.

[56] S. Duman, D. Maden, and U. Guvenc, "Determination of the PID controller parameters for speed and position control of DC motor using gravitational search algorithm," in *Proceedings of the 7th International Conference on Electrical and Electronics Engineering (ELECO '11)*, pp. I225–I229, IEEE, December 2011.

[57] W. X. Gu, X. T. Li, L. Zhu et al., "A gravitational search algorithm for flow shop scheduling," *CAAI Transaction on Intelligent Systems*, vol. 5, no. 5, pp. 411–418, 2010.

[58] D. Hoffman, "A brief overview of the biological immune system," 2011, http://www.healthy.net/ .

[59] M. Cleric and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

[60] M. S. Innocente and J. Sienz, "Particle swarm optimization with inertia weight and constriction factor," in *Proceedings of the International conference on swarm intelligence (ICSI '11)*, 2011.

[61] R. Mendes, J. Kennedy, and J. Neves, "The fully informed particle swarm: simpler, maybe better," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 204–210, 2004.

[62] M. Udrescu, L. Prodan, and M. Vlăduţiu, "Implementing quantum genetic algorithms: a solution based on Grover's algorithm," in *Proceedings of the 3rd Conference on Computing Frontiers (CF '06)*, pp. 71–81, ACM, May 2006.

[63] B. Li and L. Wang, "A hybrid quantum-inspired genetic algorithm for multiobjective flow shop scheduling," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 37, no. 3, pp. 576–591, 2007.

[64] L. Wang, F. Tang, and H. Wu, "Hybrid genetic algorithm based on quantum computing for numerical optimization and parameter estimation," *Applied Mathematics and Computation*, vol. 171, no. 2, pp. 1141–1156, 2005.

[65] A. Malossini and T. Calarco, "Quantum genetic optimization," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 2, pp. 231–241, 2008.

[66] A. Layeb, S. Meshoul, and M. Batouche, "quantum genetic algorithm for multiple RNA structural alignment," in *Proceedings of the 2nd Asia International Conference on Modelling and Simulation (AIMS '08)*, pp. 873–878, May 2008.

[67] D. Chang and Y. Zhao, "A dynamic niching quantum genetic algorithm for automatic evolution of clusters," in *Proceedings of the 14th International Conference on Computer Analysis of Images and Patterns*, vol. 2, pp. 308–315, 2011.

[68] J. Xiao, Y. Yan, Y. Lin, L. Yuan, and J. Zhang, "A quantum-inspired genetic algorithm for data clustering," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '08)*, pp. 1513–1519, June 2008.

[69] H. Talbi, A. Draa, and M. Batouche, "A new quantum-inspired genetic algorithm for solving the travelling salesman problem," in *Proceedings of the IEEE International Conference on Industrial Technology (ICIT '04)*, vol. 3, pp. 1192–1197, December 2004.

[70] K.-H. Han, K.-H. Park, C.-H. Lee, and J.-H. Kim, "Parallel quantum-inspired genetic algorithm for combinatorial optimization problem," in *Proceedings of the 2001 Congress on Evolutionary Computation*, vol. 2, pp. 1422–1429, IEEE, May 2001.

[71] L. Yan, H. Chen, W. Ji, Y. Lu, and J. Li, "Optimal VSM model and multi-object quantum-inspired genetic algorithm for web information retrieval," in *Proceedings of the 1st International Symposium on Computer Network and Multimedia Technology (CNMT '09)*, pp. 1–4, IEEE, December 2009.

[72] Z. Mo, G. Wu, Y. He, and H. Liu, "quantum genetic algorithm for scheduling jobs on computational grids," in *Proceedings of the International Conference on Measuring Technology and Mechatronics Automation (ICMTMA '10)*, pp. 964–967, March 2010.

[73] Y. Zhang, J. Liu, Y. Cui, X. Hei, and M. Zhang, "An improved quantum genetic algorithm for test suite reduction," in *Proceedings of the IEEE International Conference on Computer Science and Automation Engineering (CSAE '11)*, pp. 149–153, June 2011.

[74] J. Lee, W. Lin, G. Liao, and T. Tsao, "quantum genetic algorithm for dynamic economic dispatch with valve-point effects and including wind power system," *International Journal of Electrical Power and Energy Systems*, vol. 33, no. 2, pp. 189–197, 2011.

[75] J. Dai and H. Zhang, "A novel quantum genetic algorithm for area optimization of FPRM circuits," in *Proceedings of the 3rd International Symposium on Intelligent Information Technology Application (IITA 09)*, pp. 408–411, November 2009.

[76] L. Chuang, Y. Chiang, and C. Yang, "A quantum genetic algorithm for operon prediction," in *Proceedings of the IEEE 26th International Conference on Advanced Information Networking and Applications (AINA '12)*, pp. 269–275, March 2012.

[77] H. Xing, X. Liu, X. Jin, L. Bai, and Y. Ji, "A multi-granularity evolution based quantum genetic algorithm for QoS multicast routing problem in WDM networks," *Computer Communications*, vol. 32, no. 2, pp. 386–393, 2009.

[78] W. Luo, "A quantum genetic algorithm based QoS routing protocol for wireless sensor networks," in *Proceedings of the IEEE International Conference on Software Engineering and Service Sciences (ICSESS '10)*, pp. 37–40, IEEE, July 2010.

[79] J. Wang and R. Zhou, "A novel quantum genetic algorithm for PID controller," in *Proceedings of the 6th International Conference on Advanced Intelligent Computing Theories and Applications: Intelligent Computing*, pp. 72–77, 2010.

[80] B. Han, J. Jiang, Y. Gao, and J. Ma, "A quantum genetic algorithm to solve the problem of multivariate," *Communications in Computer and Information Science*, vol. 243, no. 1, pp. 308–314, 2011.

[81] Y. Zheng, J. Liu, W. Geng, and J. Yang, "Quantum-inspired genetic evolutionary algorithm for course timetabling," in *Proceedings of the 3rd International Conference on Genetic and Evolutionary Computing (WGEC '09)*, pp. 750–753, October 2009.

[82] Y. J. Lv and N. X. Liu, "Application of quantum genetic algorithm on finding minimal reduct," in *Proceedings of the IEEE International Conference on Granular Computing (GRC '07)*, pp. 728–733, November 2007.

[83] X. J. Zhang, S. Li, Y. Shen, and S. M. Song, "Evaluation of several quantum genetic algorithms in medical image registration applications," in *Proceedings of the IEEE International Conference on Computer Science and Automation Engineering (CSAE '12)*, vol. 2, pp. 710–713, IEEE, 2012.

[84] H. Talbi, A. Draa, and M. Batouche, "A new quantum-inspired genetic algorithm for solving the travelling salesman problem," in *Proceedings of the IEEE International Conference on Industrial Technology (ICIT '04)*, pp. 1192–1197, December 2004.

[85] S. Bhattacharyya and S. Dey, "An efficient quantum inspired genetic algorithm with chaotic map model based interference and fuzzy objective function for gray level image thresholding," in *Proceedings of the International Conference on Computational Intelligence and Communication Systems (CICN '11)*, pp. 121–125, IEEE, October 2011.

[86] K. Benatchba, M. Koudil, Y. Boukir, and N. Benkhelat, "Image segmentation using quantum genetic algorithms," in *Proceedings of the 32nd Annual Conference on IEEE Industrial Electronics (IECON '06)*, pp. 3556–3562, IEEE, November 2006.

[87] M. Liu, C. Yuan, and T. Huang, "A novel real-coded quantum genetic algorithm in radiation pattern synthesis for smart antenna," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO '07)*, pp. 2023–2026, IEEE, December 2007.

[88] R. Popa, V. Nicolau, and S. Epure, "A new quantum inspired genetic algorithm for evolvable hardware," in *Proceedings of the 3rd International Symposium on Electrical and Electronics Engineering (ISEEE '10)*, pp. 64–69, September 2010.

[89] H. Yu and J. Fan, "Parameter optimization based on quantum genetic algorithm for generalized fuzzy entropy thresholding segmentation method," in *Proceedings of the 5th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '08)*, vol. 1, pp. 530–534, IEEE, October 2008.

[90] P. C. Shill, M. F. Amin, M. A. H. Akhand, and K. Murase, "Optimization of interval type-2 fuzzy logic controller using quantum genetic algorithms," in *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '12)*, pp. 1–8, June 2012.

[91] M. Cao and F. Shang, "Training of process neural networks based on improved quantum genetic algorithm," in *Proceedings of the WRI World Congress on Software Engineering (WCSE '09)*, vol. 2, pp. 160–165, May 2009.

[92] Y. Sun and M. Ding, "quantum genetic algorithm for mobile robot path planning," in *Proceedings of the 4th International Conference on Genetic and Evolutionary Computing (ICGEC '10)*, pp. 206–209, December 2010.

[93] K. Han and J. Kim, "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 6, pp. 580–593, 2002.

[94] R. Zhang and H. Gao, "Improved quantum evolutionary algorithm for combinatorial optimization problem," in *Proceedings of the 6th International Conference on Machine Learning and Cybernetics (ICMLC '07)*, vol. 6, pp. 3501–3505, August 2007.

[95] M. D. Platel, S. Sehliebs, and N. Kasabov, "A versatile quantum-inspired evolutionary algorithm," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '07)*, pp. 423–430, September 2007.

[96] P. Li and S. Li, "Quantum-inspired evolutionary algorithm for continuous space optimization based on Bloch coordinates of qubits," *Neurocomputing*, vol. 72, no. 1–3, pp. 581–591, 2008.

[97] K. Han and J. Kim, "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 6, pp. 580–593, 2002.

[98] P. Mahdabi, S. Jalili, and M. Abadi, "A multi-start quantum-inspired evolutionary algorithm for solving combinatorial optimization problems," in *Proceedings of the 10th Annual Genetic and Evolutionary Computation Conference (GECCO '08)*, pp. 613–614, ACM, July 2008.

[99] H. Talbi, M. Batouche, and A. Draao, "A quantum-inspired evolutionary algorithm for multiobjective image segmentation," *International Journal of Mathematical, Physical and Engineering Sciences*, vol. 1, no. 2, pp. 109–114, 2007.

[100] Y. Kim, J. Kim, and K. Han, "Quantum-inspired multiobjective evolutionary algorithm for multiobjective 0/1 knapsack problems," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 2601–2606, July 2006.

[101] A. Narayan and C. Patvardhan, "A novel quantum evolutionary algorithm for quadratic knapsack problem," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '09)*, pp. 1388–1392, October 2009.

[102] A. R. Hota and A. Pat, "An adaptive quantum-inspired differential evolution algorithm for 0-1 knapsack problem," in *Proceedings of the 2nd World Congress on Nature and Biologically Inspired Computing (NaBIC '10)*, pp. 703–708, December 2010.

[103] Y. Ji and H. Xing, "A memory storable quantum inspired evolutionary algorithm for network coding resource minimization," in *Evolutionary Algorithms*, InTech, Shanghai, China, 2011.

[104] H. Xing, Y. Ji, L. Bai, and Y. Sun, "An improved quantum-inspired evolutionary algorithm for coding resource optimization based network coding multicast scheme," *International Journal of Electronics and Communications*, vol. 64, no. 12, pp. 1105–1113, 2010.

[105] A. da Cruz, M. M. B. R. Vellasco, and M. Pacheco, "Quantum-inspired evolutionary algorithm for numerical optimization," in *Hybrid Evolutionary Algorithms*, pp. 19–37, Springer, Berlin, Germany, 2007.

[106] G. Zhang and H. Rong, "Real-observation quantum-inspired evolutionary algorithm for a class of numerical optimization problems," in *Proceedings of the 7th international conference on Computational Science, Part IV (ICCS '07)*, vol. 4490, pp. 989–996, 2007.

[107] R. Setia and K. H. Raj, "Quantum inspired evolutionary algorithm for optimization of hot extrusion process," *International Journal of Soft Computing and Engineering*, vol. 2, no. 5, p. 29, 2012.

[108] T. Lau, *Application of quantum-inspired evolutionary algorithm in solving the unit commitment problem [dissertation]*, The Hong Kong Polytechnic University, Hong Kong, 2011.

[109] C. Y. Chung, H. Yu, and K. P. Wong, "An advanced quantum-inspired evolutionary algorithm for unit commitment," *IEEE Transactions on Power Systems*, vol. 26, no. 2, pp. 847–854, 2011.

[110] J. G. Vlachogiannis and K. Y. Lee, "Quantum-inspired evolutionary algorithm for real and reactive power dispatch," *IEEE Transactions on Power Systems*, vol. 23, no. 4, pp. 1627–1636, 2008.

[111] U. Pareek, M. Naeem, and D. C. Lee, "Quantum inspired evolutionary algorithm for joint user selection and power allocation for uplink cognitive MIMO systems," in *Proceedings of the IEEE Symposium on Computational Intelligence in Scheduling (SCIS '11)*, pp. 33–38, April 2011.

[112] J. Chen, "Application of quantum-inspired evolutionary algorithm to reduce PAPR of an OFDM signal using partial transmit sequences technique," *IEEE Transactions on Broadcasting*, vol. 56, no. 1, pp. 110–113, 2010.

[113] J. Jang, K. Han, and J. Kim, "Face detection using quantum-inspired evolutionary algorithm," in *Proceedings of the 2004 Congress on Evolutionary Computation (CEC '04)*, vol. 2, pp. 2100–2106, June 2004.

[114] J. Jang, K. Han, and J. Kim, "Quantum-inspired evolutionary algorithm-based face verification," in *Genetic and Evolutionary Computation—GECCO 2003*, pp. 214–214, Springer, Berlin, Germany, 2003.

[115] K. Fan, A. Brabazon, C. O'Sullivan, and M. O'Neill, "Quantum-inspired evolutionary algorithms for financial data analysis," in *Applications of Evolutionary Computing*, pp. 133–143, Springer, Berlin, Germany, 2008.

[116] K. Fan, A. Brabazon, C. O'Sullivan, and M. O'Neill, "Option pricing model calibration using a real-valued quantum-inspired evolutionary algorithm," in *Proceedings of the 9th Annual Genetic and Evolutionary Computation Conference (GECCO '07)*, pp. 1983–1990, ACM, July 2007.

[117] K. Fan, A. Brabazon, C. OSullivan, and M. ONeill, "Quantum-inspired evolutionary algorithms for calibration of the VG option pricing model," in *Applications of Evolutionary Computing*, pp. 189–198, Springer, Berlin, Germany, 2007.

[118] R. A. de Araújo, "A quantum-inspired evolutionary hybrid intelligent approach for stock market prediction," *International Journal of Intelligent Computing and Cybernetics*, vol. 3, no. 1, pp. 24–54, 2010.

[119] Z. Huang, Y. Wang, C. Yang, and C. Wu, "A new improved quantum-behaved particle swarm optimization model," in *Proceedings of the 4th IEEE Conference on Industrial Electronics and Applications (ICIEA '09)*, pp. 1560–1564, May 2009.

[120] J. Chang, F. An, and P. Su, "A quantum-PSO algorithm for no-wait flow shop scheduling problem," in *Proceedings of the Chinese Control and Decision Conference (CCDC '10)*, pp. 179–184, May 2010.

[121] X. Wu, B. Zhang, K. Wang, J. Li, and Y. Duan, "A quantum-inspired Binary PSO algorithm for unit commitment with wind farms considering emission reduction," in *Proceedings of the Innovative Smart Grid Technologies—Asia (ISGT '12)*, pp. 1–6, IEEE, May 2012.

[122] Y. Jeong, J. Park, S. Jang, and K. Y. Lee, "A new quantum-inspired binary PSO for thermal unit commitment problems," in *Proceedings of the 15th International Conference on Intelligent System Applications to Power Systems (ISAP '09)*, pp. 1–6, November 2009.

[123] H. N. A. Hamed, N. Kasabov, and S. M. Shamsuddin, "Integrated feature selection and parameter optimization for evolving spiking neural networks using quantum inspired particle swarm optimization," in *Proceedings of the International Conference on Soft Computing and Pattern Recognition (SoCPaR '09)*, pp. 695–698, December 2009.

[124] A. A. Ibrahim, A. Mohamed, H. Shareef, and S. P. Ghoshal, "An effective power quality monitor placement method utilizing quantum-inspired particle swarm optimization," in *Proceedings of the International Conference on Electrical Engineering and Informatics (ICEEI '11)*, pp. 1–6, July 2011.

[125] F. Yao, Z. Y. Dong, K. Meng, Z. Xu, H. H. Iu, and K. Wong, "Quantum-inspired particle swarm optimization for power system operations considering wind power uncertainty and carbon tax in Australia," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 4, pp. 880–888, 2012.

[126] Z. Zhisheng, "Quantum-behaved particle swarm optimization algorithm for economic load dispatch of power system," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1800–1803, 2010.

[127] A. Chen, G. Yang, and Z. Wu, "Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem," *Journal of Zhejiang University*, vol. 7, no. 4, pp. 607–614, 2006.

[128] T. J. Ai and V. Kachitvichyanukul, "A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery," *Computers and Operations Research*, vol. 36, no. 5, pp. 1693–1702, 2009.

[129] Y. Marinakis, M. Marinaki, and G. Dounias, "A hybrid particle swarm optimization algorithm for the vehicle routing problem," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 4, pp. 463–472, 2010.

[130] S. N. Omkar, R. Khandelwal, T. V. S. Ananth, G. Narayana Naik, and S. Gopalakrishnan, "Quantum behaved Particle Swarm Optimization (QPSO) for multi-objective design optimization of composite structures," *Expert Systems with Applications*, vol. 36, no. 8, pp. 11312–11322, 2009.

[131] L. D. S. Coelho, "Gaussian quantum-behaved particle swarm optimization approaches for constrained engineering design problems," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1676–1683, 2010.

[132] M. Ykhlef, "A quantum swarm evolutionary algorithm for mining association rules in large databases," *Journal of King Saud University*, vol. 23, no. 1, pp. 1–6, 2011.

[133] M. Dorigo and T. Stiitzle, *Ant Colony Optimization*, pp. 153–222, chapter 4, MIT Press, Cambridge, Mass, USA, 1st edition, 2004.

[134] L. Wang, Q. Niu, and M. Fei, "A novel quantum ant colony optimization algorithm," in *Bio-Inspired Computational Intelligence and Applications*, pp. 277–286, Springer, Berlin, Germany, 2007.

[135] X. You, S. Liu, and Y. Wang, "Quantum dynamic mechanism-based parallel ant colony optimization algorithm," *International Journal of Computational Intelligence Systems*, vol. 3, no. 1, pp. 101–113, 2010.

[136] L. Wang, Q. Niu, and M. Fei, "A novel quantum ant colony optimization algorithm and its application to fault diagnosis," *Transactions of the Institute of Measurement and Control*, vol. 30, no. 3-4, pp. 313–329, 2008.

[137] Z. Yu, L. Shuhua, F. Shuai, and W. Di, "A quantum-inspired ant colony optimization for robot coalition formation," in *Chinese Control and Decision Conference (CCDC '09)*, pp. 626–631, June 2009.

[138] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 52–67, 2002.

[139] T. Kumbasar, I. Eksin, M. Güzelkaya, and E. Yeşil, "Big bang big crunch optimization method based fuzzy model inversion," in *MICAI 2008: Advances in Artificial Intelligence*, pp. 732–740, Springer, Berlin, Germany, 2008.

[140] T. Kumbasar, E. Yeşil, I. Eksin, and M. Güzelkaya, "Inverse fuzzy model control with online adaptation via big bang-big crunch optimization," in *2008 3rd International Symposium on Communications, Control, and Signal Processing (ISCCSP '08)*, pp. 697–702, March 2008.

[141] M. Aliasghary, I. Eksin, and M. Guzelkaya, "Fuzzy-sliding model reference learning control of inverted pendulum with Big Bang-Big Crunch optimization method," in *Proceedings of the 11th International Conference on Intelligent Systems Design and Applications (ISDA '11)*, pp. 380–384, November 2011.

[142] H. M. Genç, I. Eksin, and O. K. Erol, "Big Bang-Big Crunch optimization algorithm hybridized with local directional moves and application to target motion analysis problem," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '10)*, pp. 881–887, October 2010.

[143] H. M. Genç and A. K. Hocaoğlu, "Bearing-only target tracking based on Big Bang-Big Crunch algorithm," in *Proceedings of the 3rd International Multi-Conference on Computing in the Global Information Technology (ICCGI '08)*, pp. 229–233, July 2008.

[144] P. Prudhvi, "A complete copper optimization technique using BB-BC in a smart home for a smarter grid and a comparison with GA," in *Proceedings of the 24th Canadian Conference on Electrical and Computer Engineering (CCECE '11)*, pp. 69–72, May 2011.

[145] G. M. Jaradat and M. Ayob, "Big Bang-Big Crunch optimization algorithm to solve the course timetabling problem," in *Proceedings of the 10th International Conference on Intelligent Systems Design and Applications (ISDA '10)*, pp. 1448–1452, December 2010.

[146] D. Debels, B. De Reyck, R. Leus, and M. Vanhoucke, "A hybrid scatter search/electromagnetism meta-heuristic for project scheduling," *European Journal of Operational Research*, vol. 169, no. 2, pp. 638–653, 2006.

[147] P. Chang, S. Chen, and C. Fan, "A hybrid electromagnetism-like algorithm for single machine scheduling problem," *Expert Systems with Applications*, vol. 36, no. 2, pp. 1259–1267, 2009.

[148] A. Jamili, M. A. Shafia, and R. Tavakkoli-Moghaddam, "A hybridization of simulated annealing and electromagnetism-like mechanism for a periodic job shop scheduling problem," *Expert Systems with Applications*, vol. 38, no. 5, pp. 5895–5901, 2011.

[149] M. Mirabi, S. M. T. Fatemi Ghomi, F. Jolai, and M. Zandieh, "Hybrid electromagnetism-like algorithm for the flowshop scheduling with sequence-dependent setup times," *Journal of Applied Sciences*, vol. 8, no. 20, pp. 3621–3629, 2008.

[150] B. Naderi, R. Tavakkoli-Moghaddam, and M. Khalili, "Electromagnetism-like mechanism and simulated annealing algorithms for flowshop scheduling problems minimizing the total weighted tardiness and makespan," *Knowledge-Based Systems*, vol. 23, no. 2, pp. 77–85, 2010.

[151] H. Turabieh, S. Abdullah, and B. McCollum, "Electromagnetism-like mechanism with force decay rate great deluge for the course timetabling problem," in *Rough Sets and Knowl-edge Technology*, pp. 497–504, Springer, Berlin, Germany, 2009.

[152] C. Lee and F. Chang, "Fractional-order PID controller optimization via improved electromagnetism-like algorithm," *Expert Systems with Applications*, vol. 37, no. 12, pp. 8871–8878, 2010.

[153] S. Birbil and O. Feyzioğlu, "A global optimization method for solving fuzzy relation equations," in *Fuzzy Sets and Systems (IFSA '03)*, pp. 47–84, Springer, Berlin, Germany, 2003.

[154] P. Wu, K. Yang, and Y. Hung, "The study of electromagnetism-like mechanism based fuzzy neural network for learning fuzzy if-then rules," in *Knowledge-Based Intelligent Information and Engineering Systems*, pp. 907–907, Springer, Berlin, Germany, 2005.

[155] C. Lee, C. Kuo, H. Chang, J. Chien, and F. Chang, "A hybrid algorithm of electromagnetism-like and genetic for recurrent neural fuzzy controller design," in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1, March 2009.

[156] A. Yurtkuran and E. Emel, “A new hybrid electromagnetism-like algorithm for capacitated vehicle routing problems,” *Expert Systems with Applications*, vol. 37, no. 4, pp. 3427–3433, 2010.

[157] C. Tsai, H. Hung, and S. Lee, “Electromagnetism-like method based blind multiuser detection for MC-CDMA interference suppression over multipath fading channel,” in *2010 International Symposium on Computer, Communication, Control and Automation (3CA ’10)*, vol. 2, pp. 470–475, May 2010.

[158] C.-S. Tsou and C.-H. Kao, “Multi-objective inventory control using electromagnetism-like meta-heuristic,” *International Journal of Production Research*, vol. 46, no. 14, pp. 3859–3874, 2008.

[159] X. Wang, L. Gao, and C. Zhang, “Electromagnetism-like mechanism based algorithm for neural network training,” in *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*, pp. 40–45, Springer, Berlin, Germany, 2008.

[160] Q. Wu, C. Zhang, L. Gao, and X. Li, “Training neural networks by electromagnetism-like mechanism algorithm for tourism arrivals forecasting,” in *Proceedings of the IEEE 5th International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA ’10)*, pp. 679–688, September 2010.

[161] P. Wu and H. Chiang, “The Application of electromagnetism-like mechanism for solving the traveling salesman problems,” in *Proceeding of the 2005 Chinese Institute of Industrial Engineers Annual Meeting*, Taichung, Taiwan, December 2005.

[162] P. Wu, K. Yang, and H. Fang, “A revised EM-like algorithm + K-OPT method for solving the traveling salesman problem,” in *1st International Conference on Innovative Computing, Information and Control 2006 (ICICIC ’06)*, vol. 1, pp. 546–549, August 2006.

[163] C. Su and H. Lin, “Applying electromagnetism-like mechanism for feature selection,” *Information Sciences*, vol. 181, no. 5, pp. 972–986, 2011.

[164] K. C. Lee and J. Y. Jhang, “Application of electromagnetism-like algorithm to phase-only syntheses of antenna arrays,” *Progress in Electromagnetics Research*, vol. 83, pp. 279–291, 2008.

[165] C. Santos, M. Oliveira, V. Matos, A. Maria, A. C. Rocha, and L. A. Costa, “Combining central pattern generators with the electromagnetism-like algorithm for head motion stabilization during quadruped robot locomotion,” in *Proceedings of the 2nd International Workshop on Evolutionary and Reinforcement Learning for Autonomous Robot Systems*, 2009.

[166] X. Guan, X. Dai, and J. Li, “Revised electromagnetism-like mechanism for flow path design of unidirectional AGV systems,” *International Journal of Production Research*, vol. 49, no. 2, pp. 401–429, 2011.

[167] A. Yurtkuran and E. Emel, “A new hybrid electromagnetism-like algorithm for capacitated vehicle routing problems,” *Expert Systems with Applications*, vol. 37, no. 4, pp. 3427–3433, 2010.

[168] K. F. Pál, “Hysteretic optimization for the traveling salesman problem,” *Physica A*, vol. 329, no. 1-2, pp. 287–297, 2003.

[169] B. Gonçalves and S. Boettcher, “Hysteretic optimization for spin glasses,” *Journal of Statistical Mechanics*, vol. 2008, no. 1, Article ID P01003, 2008.

[170] X. Yan and W. Wu, “Hysteretic optimization for the capacitated vehicle routing problem,” in *Proceedings of the 9th IEEE International Conference on Networking, Sensing and Control (ICNSC ’12)*, pp. 12–15, April 2012.

[171] J. Zha, G. Zeng, and Y. Lu, “Hysteretic optimization for protein folding on the lattice,” in *Proceedings of the International Conference on Computational Intelligence and Software Engineering (CiSE ’10)*, pp. 1–4, December 2010.

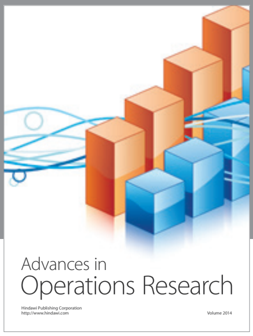
[172] A. Kaveh and S. Talatahari, “A charged system search with a fly to boundary method for discrete optimum design of truss structures,” *Asian Journal of Civil Engineering*, vol. 11, no. 3, pp. 277–293, 2010.

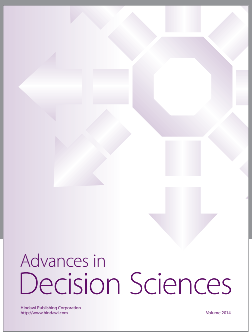
[173] A. Kaveh and S. Talatahari, “Optimal design of skeletal structures via the charged system search algorithm,” *Structural and Multidisciplinary Optimization*, vol. 41, no. 6, pp. 893–911, 2010.

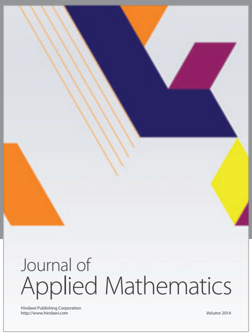
[174] A. Kaveh and S. Talatahari, “Charged system search for optimal design of frame structures,” *Applied Soft Computing Journal*, vol. 12, no. 1, pp. 382–393, 2012.

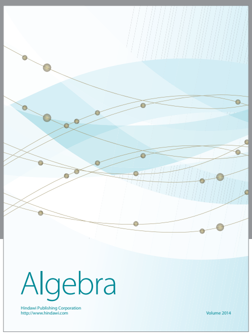
[175] A. Kaveh and S. Talatahari, “Charged system search for optimum grillage system design using the LRFD-AISC code,” *Journal of Constructional Steel Research*, vol. 66, no. 6, pp. 767–771, 2010.

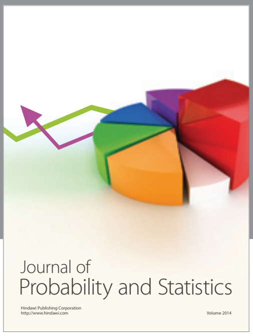
[176] A. Kaveh and S. Talatahari, “Geometry and topology optimization of geodesic domes using charged system search,” *Structural and Multidisciplinary Optimization*, vol. 43, no. 2, pp. 215–229, 2011.


Advances in Operations Research

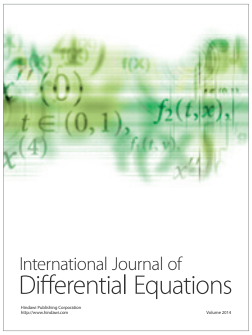

Advances in Decision Sciences

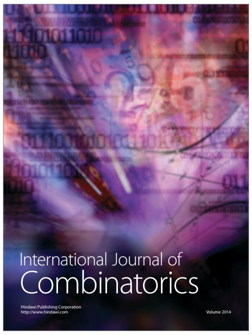

Journal of Applied Mathematics

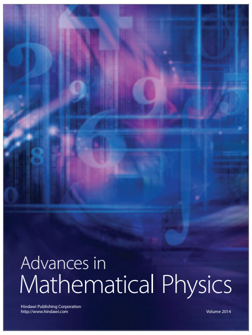

Algebra

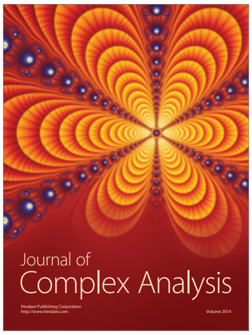

Journal of Probability and Statistics

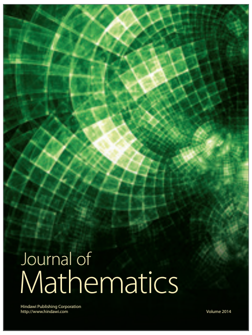
The Scientific World Journal

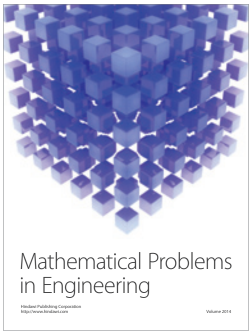

International Journal of Differential Equations

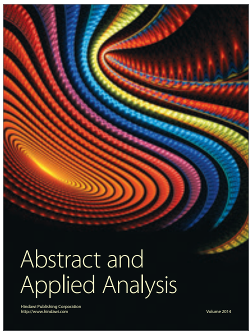

International Journal of Combinatorics

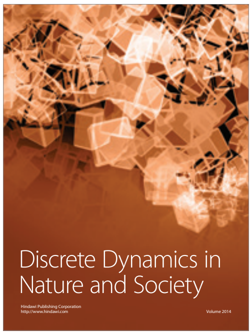

Advances in Mathematical Physics

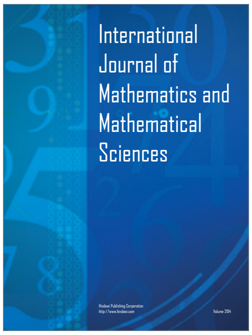

Journal of Complex Analysis


Journal of Mathematics


Mathematical Problems in Engineering

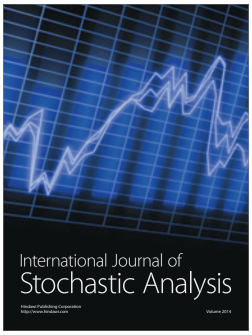

Abstract and Applied Analysis


Discrete Dynamics in Nature and Society


International Journal of Mathematics and Mathematical Sciences

Journal of Discrete Mathematics

Journal of Function Spaces


International Journal of Stochastic Analysis

Journal of Optimization