CrossMark

Genetic algorithms in wireless networking: techniques, applications, and issues

Usama Mehboob[1] · Junaid Qadir[3] · Salman Ali[1] · Athanasios Vasilakos[2]

Abstract In recent times, wireless access technology is becoming increasingly commonplace due to the ease of operation and installation of untethered wireless media. The design of wireless networking is challenging due to the highly dynamic environmental condition that makes parameter optimization a complex task. Due to the dynamic, and often unknown, operating conditions, modern wireless networking standards increasingly rely on machine learning and artificial intelligence algorithms. Genetic algorithms (GAs) provide a well-established framework for implementing artificial intelligence tasks such as classification, learning, and optimization. GAs are well known for their remarkable generality and versatility and have been applied in a wide variety of settings in wireless networks. In this paper, we provide a comprehensive survey of the applications of GAs in wireless networks. We provide both an exposition of common GA models and configuration and provide a broad-ranging survey of GA techniques in wireless networks. We also point out open research issues and define potential future work. While various surveys on GAs exist in the literature, our paper is the first paper, to the best of our knowledge, which focuses on their application in wireless networks.

✉ Junaid Qadir
  junaid.qadir@itu.edu.pk

  Usama Mehboob
  10beeumehboob@seecs.edu.pk

  Salman Ali
  09mscsesali@seecs.edu.pk

  Athanasios Vasilakos
  athanasios.vasilakos@ltu.se

[1] School of EE and CS, National University of Sciences and Technology (NUST), Islamabad, Pakistan

[2] Lulea University of Technology, Lulea, Sweden

[3] Department of Electrical Engineering, Information Technology University (ITU), Punjab, Pakistan, Lahore, Pakistan

1 Introduction

The study of nature is a rich source of inspiration for researchers working in the area of artificial intelligence and machine learning. In particular, the human brain and the biological process of evolution have helped develop and guide research in the neural network and the evolutionary algorithm communities. A genetic algorithm (GA) is a metaheuristic computational method (Hillier and Lieberman 2001), inspired from biological evolution (Ridley 2004), that aims to imitate the robust procedures used by various biological organisms to adapt as part of their natural evolution. GAs have been successfully used in fields as diverse as aircraft industry, chip design, computer animation, drug design, telecommunications, software creation, and financial markets (Mitchell 1998).

The field of GA was established by John Holland who investigated the evolution of complex adaptive systems (CAS) comprising interacting genes, starting in the early 1960s, and culminating in the publication in 1975 of his seminal book on this topic (Holland 1975). The idea of embedding computers with learning inspired from evolution goes back even further as Alan Turing proposed a "learning machine" which would parallel the principles of evolution in his landmark 1950 paper on machine intelligence (Turing 1950). Holland used the biological metaphor of chromosomes to refer to strings of binary symbols encoding a candidate solution to the given problem. Holland proposed using the computational analogues of the biological evolu-

tionary processes of random mutation, crossover, and natural selection (Goldberg and Holland 1988) to enable populations of chromosomes to get increasingly better at problem solving. The underlying premise is that given enough time, the process would converge towards a population that contains a chromosome (or chromosomes) that solves the given problem. Apart from the mutation and crossover operators, The design of a GA framework also involves other design issues such as genetic representation (encoding), population initialization, fitness function formulation, and a selection mechanism. More recently, researchers have also proposed the use of altruistic techniques in an evolutionary framework that involve cooperation as a fundamental primitive (Highfield and Nowak 2011).

1.1 Motivations for using GA

1.1.1 Generality and versatility

Genetic algorithms apply in a wide variety of settings and can be easily molded to particular problems. GAs constitute a very general meta-heuristic technique which can be thought of as the sledgehammer of the craft of algorithms, much like the technique of artificial neural networks (ANNs). In particular, GAs can be readily invoked in areas that do not yield readily to standard approaches or when more specialized techniques fail. GAs are capable of solving extremely large problems that have large search spaces. GAs are very good at navigating through huge search spaces to heuristically find near-optimal solutions in quick time. In addition, GAs can work even when the objective function is not exactly known since GAs rely only on an objective function's evaluation (without necessarily knowing the objective function explicitly). Although GAs do not guarantee optimality, GAs are generally useful in practice. Bhandari et al. 1996 showed that although GAs do not guarantee convergence to an optimal solution, GAs avoid local optimas with a high probability through the use of mutation and crossover operators.

1.1.2 Adaptiveness and online problem solving

An important feature of GAs is that they define online adaptive algorithms that can operate in unknown environments. Such an ability is crucial in various control settings in wireless networks in which decisions have to be made automatically in run-time to cater to dynamic channel parameters. In dynamic channel conditions that typically exist in most wireless networking configurations, optimization is an elusive moving target since the optimal solution keeps changing as the conditions change.

In online adaptive optimization algorithms, there is a fundamental tradeoff between exploration—which involves looking for potentially better previously unexplored solutions—and exploitation that implies the use of previously known good solutions. Exploration in effect is an attempt to find good adaptive building blocks and exploitation is the use and propagation of adaptations that are known to perform well. We can envision mutation and recombination of genes as analogous to exploration, whereas natural selection can be envisioned as a form of exploitation. In ever-changing dynamic environments, such as wireless networks, it is important to always keep on exploring. Holland's original GA work used schema analysis to show that, under qualifying assumptions, GAs can achieve near-optimal exploration and exploitation tradeoff.

1.1.3 Ability to find good building blocks

By working in terms of a population of candidate solutions, genetic algorithms can exploit the diversity of solutions to find building blocks—known as schemas in literature—which are substrings of chromosomes that denote high-performing elements of the overall solution (Holland 1975). In the context of biological genetics, schemas correspond to constellations of genes working together to help an organism adapt; evolution proceeds by selecting these constellations (or building blocks) through natural selection. Using genetic operators such as recombination and crossover, GAs can evolve better solutions through mixing of the high-performing basic building blocks.

1.1.4 Parallel nature and scalability

Traditional theory of GAs presumes that GAs accomplish the culmination by discovering, emphasizing, and recombining the good traits of chromosomes in a vastly parallel manner. GAs incorporate multiple solutions together in a 'population' of solutions. GAs use evolutionary techniques to test and improve the solutions using techniques such as mutation, crossover, selection, and recombination. The parallel nature of GAs renders it as a suitable optimization framework for scalable problem solving in a wide range of applications.

1.1.5 Support for multiobjective optimization

Many practical problems in wireless networks require optimizing for multiple parameters (that may potentially conflict with each other). An important characteristic of GAs is that it can easily support joint optimization of multiple objectives. Support for multi-objective optimization—in which the considered objectives may potentially conflict with each other—is of significant practical interest in wireless networks. Issues related to multi-objective optimization are introduced in Konak et al. (2006).

1.1.6 Support for global optimization

Unlike network models such as the multi-layered perceptrons (which make local changes so as to find the local minima), GAs are suited to finding the global optima due to a number of properties: (i) they search by means of a population of individuals; (ii) they work with an encoding of the multiple parameters; (iii) they use a fitness function that does not require the calculation of derivatives; and, finally (iv) they search probabilistically.

1.1.7 Easy implementation

Genetic algorithms are computationally simpler compared to other complementary AI techniques such as neural networks since they require only swapping and shifting of genes in chromosomes (unlike neural networks that require adders for its multiple hidden layers). GAs are easily implemented and can be rapidly prototyped in digital signal processors (DSPs) or field programmable gate arrays (FPGAs). In addition, GAs are highly amenable to successful implementation on modern parallel and cloud computing architectures due to its inherently parallel nature.

1.2 Contributions of this paper

This paper offers a comprehensive survey of the applications of GAs in wireless networking. In addition, we also provide a self-contained introduction to the GAs. The aim of this paper was not to provide an exhaustive survey,[1] but to provide a representative sample of important works along with a comprehensive listing of the various applications of GAs. We also highlight the application of GAs in wireless networking configurations such as wireless mesh networks (WMNs), mobile ad-hoc networks (MANETs), cellular networks (CNs), wireless sensor networks (WSNs), and cognitive radio networks (CRNs). While a lot of material exists on GAs—including various textbooks (Goldberg and Holland 1988; Davis et al. 1991; Pedrycz and Vasilakos 2010), as well as generic survey papers (Srinivas and Patnaik 1994; Man et al. 1996)—a focused survey article on the applications of GAs in wireless networking is missing in literature. This paper will be useful to networking researchers interested in applying GAs in particular, and metaheuristic and evolutionary techniques more generally, in the context of wireless networks. According to the best of our knowledge, this is the first survey paper that

---

[1] The sheer breadth of GA literature precludes an exhaustive survey; the exhaustive bibliography created by Goldberg et al. in 1997—which is a excellent reference to the pool of GA literature—amassed more than 4000 publications. With the unabated interest in GAs, it will be no surprise if an updated bibliography contains more than double the original references.

provides an extensive survey of the applications of GAs in general wireless networks.

1.3 Organization of this paper

We provide the necessary background on GA theory, working and common techniques used to carry out the evolutionary process in Sect. 2. The applications of GAs in wireless networks—such as routing, channel allocation, load balancing, localization, and quality of service (QoS)—are listed in Sect. 3. We highlight the various lessons learnt through years of GA research, in terms of common pitfalls as well as guidelines for successful implementation, in Sect. 4.3. In Sect. 5, we highlight open issues and suggest directions for future work. Finally, this paper is concluded in Sect. 6.

To facilitate the reader, acronyms used in this paper are collected in Table 1 as a convenient reference.

2 Background: genetic algorithms

A detailed GA taxonomy is provided in Fig. 1. This taxonomy diagram exhibits the breadth of GA's applications as described in this section.

2.1 GA terminology

Since the field of GA is inspired from the biological genetic evolution, the field uses a number of biological metaphors in its terminology. The purpose of this short subsection was to introduce these biological metaphors in the context of applications of GAs in wireless networking. We introduce the terminology through the following list:

- Organism: It represents the entity (e.g., a radio parameter, or a wireless resource) being optimized.
- Population: It refers to the ensemble or collection of organisms undergoing simulated genetic evolution.
- Chromosome: It encodes a particular solution to the problem under study. (Biologically, a chromosome contains an organism's genetic makeup).
- Fitness: It represents the utility of the current chromosome. (In evolutionary theory, fit chromosomes are passed on through heredity while unfit chromosomes die out due to the natural phenomenon of the 'survival of the fittest').
- Gene: It is the basic building block of the chromosome defining a particular feature of the simulated organism. (Each chromosome can contain a number of genes).
- Allele: Each gene can take several alternative forms, each of which is called an allele.
- Locus: It is the position on the chromosome containing a particular gene of interest.

Table 1 Acronyms used in this paper

Acronym	Expanded form
ACO	Ant colony optimization
AP	Access point
ANN	Artificial neural network
BER	Bit error rate
BFWA	Broadband fixed wireless access
BS	Base station
BSP	Broadcast scheduling problem
CAC	Call admission control
CBDT	Case base decision theory
CSM	Cognitive system module
CN	Cellular network
CDMA	Code division multiple access
CR	Cognitive radio
CRN	Cognitive radio network
DCA	Dynamic channel assignment
DSA	Dynamic spectrum access
DSPRP	Dynamic shortest path routing problem
FA	Forward ant
FCA	Fixed channel assignment
FPGA	Field programmable gate array
GA	Genetic algorithm
GP	Genetic programming
GPRS	General packet radio service
iGA	Island GA
HLB	Hybrid load balancing
ILP	Integer linear programming
LISP	List processing (programming language)
LLF	Least loaded first
LTE	Long-term evolution
MAC	Media access control
MANET	Mobile ad-hoc network
MDP	Markov decision process
MOGA	Multi-objective GA
MSC	Mobile switching center
NN	Neural network
NSGA	Non-dominated sorted genetic algorithm
NSGA-II	Non-dominated sorted genetic algorithm-II
PGA	Parallel GA
QoS	Quality of service
RNC	Radio network controller
RSS	Radio signal strength
RSSI	Received signal strength indicator
SA	Simulated annealing
SGSN	Serving GPRS node
SINR	Signal to interference and noise ratio
SNR	Signal-to-noise ratio
SOGA	Single objective GA

Table 1 continued

Acronym	Expanded form
SP	Shortest path
SWN	Software-defined wireless network
UMTS	Universal mobile telecommunications system
WLAN	Wireless local area network
WMN	Wireless mesh network
WSGA	Wireless system genetic algorithm

- Mutation: It represents a random change of an allele (or the value of a gene).
- Selection: The process through which the 'fittest' simulated organism survives while the unfit solutions are weeded out.

2.2 Number-of-objectives based classification

2.2.1 Single-objective GAs

Single objective GAs (SOGAs) are popular when it comes to optimize single objective with scalar fitness function. They are computationally less exhaustive but are required to run multiple times using different weight vectors to optimize multiple objectives as in case of multi-objective GAs (Ishibuchi et al. 2006).

2.2.2 Multi-objective GAs

In a multi-objective optimization, there are multiple objectives that need to be simultaneously optimized. For example, there can be multiple objectives in wireless networking—such as the signal to interference and noise ratio (SINR), the power consumption, and the bit error rate (BER)—that we could be interested in optimizing. In such a scenario, a single solution may not be suitable to multiple objectives. A solution may be best in one objective but worst in other cases. The overall system is optimized, in all directions, by multiple solutions. These categories of solutions are non-dominated solutions that lie on the Pareto front. Each point in Pareto front is optimal, since no single solution exists that may improve one objective vector without degrading at least one other objective. Almost all the solutions lying on the Pareto front would be compromises relative to multiple objectives.

Multi-objective GAs (MOGAs) can be employed in almost every optimization setting provided that chromosomes encoding and objective function are appropriately accounted for. Details and applications of MOGAs are discussed next in Sect. 3. In this section, we will describe one particular instance of a MOGA framework known as non-dominated sorting genetic algorithm (NSGA).
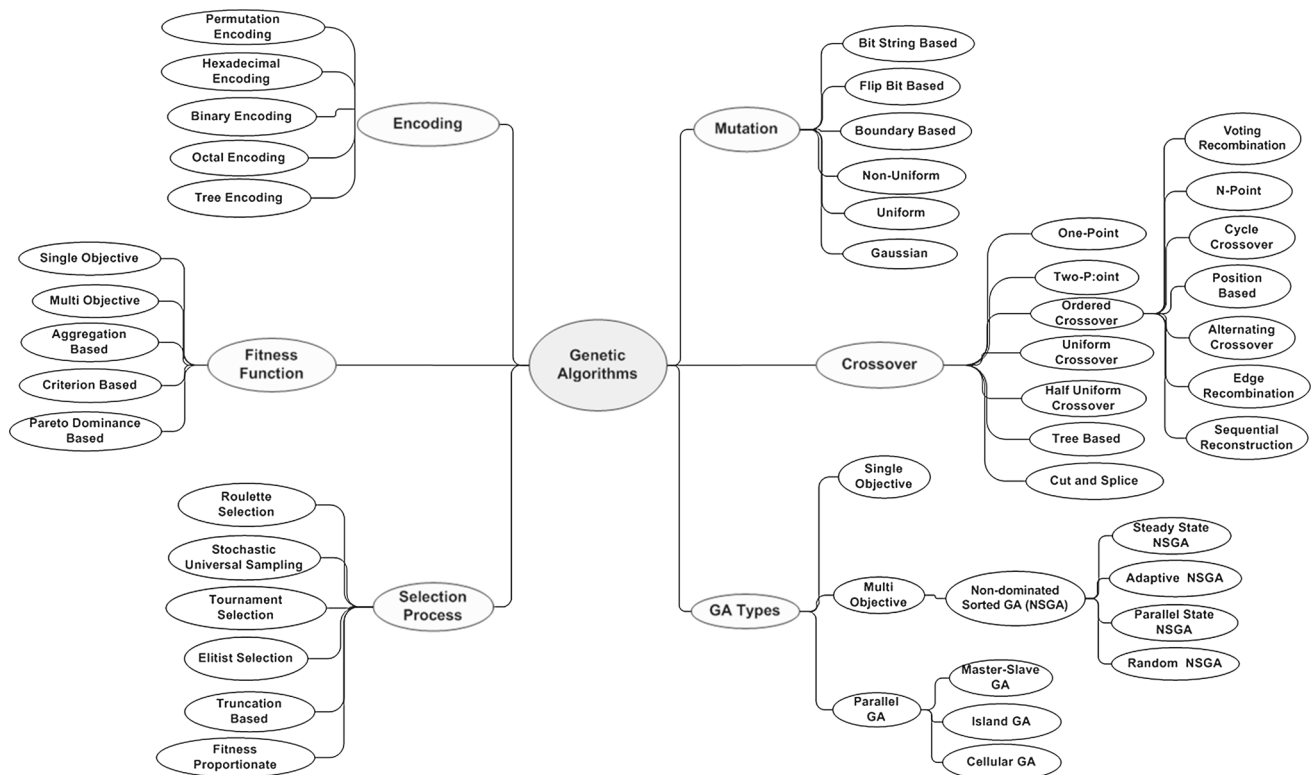
Fig. 1 Taxonomy of genetic algorithms

2.2.2.1 Non-dominated sorted genetic algorithm (NSGA)
Non-dominated sorted genetic algorithm is a class of multi-objective optimization algorithm that works at improving the adaptive fit of a population to the Pareto front constrained by a set of objective functions. NSGA can also be seen as a case of an evolutionary algorithm from the domain of evolutionary computation. The population within the algorithm is sorted into levels or sub-populations, which are based on the arrangement of the Pareto dominance. Major similarities between the population members of each sub division are calculated against the Pareto front, and the resulting class, along with its similarity measure, is then utilized to promote a solutions front that is non-dominated. Non-dominated sorting genetic algorithm-II (NSGA-II) is a widely used computationally fast and elitist MOGA approach (Deb et al. 2002).

2.3 Other types of GAs

2.3.1 Adaptive GA

To prevent GAs from suboptimally converging to a local optima, a number of adaptive techniques have been recommended for adjusting parameters such as crossover probability, mutation probability, and population size. These techniques constitute adaptive GAs. In contrast to simple GAs that rely on fixed control parameters, adaptive GAs utilize information about the population in each generation to adaptively adjust the probability of crossover and mutation.

2.3.2 Parallel GA

Parallel GA (PGA) can be categorized into three subcategories: (i) master–slave GA; (ii) island GA (also referred to as distributed GA or coarse-grained GA); and (iii) cellular GA (also called fine-grained GA). Amongst these three configurations, the Island PGAs (island GA, or iGA, for short) are best suited to wireless networking since they allow control of the migration policy unlike the cellular PGAs—which require local communication at all agents during each generation of the PGA—and the master–slave PGAs that require the master and slaves to constantly communicate.

2.3.3 Distributed or island GA

Distributed GAs divide a large population into several smaller subpopulations that interact weakly. Parallel GAs, on the other hand, are primed to speed up execution by implementing the sequential GA on several computation engines. Amongst the distributed GA approaches, the island GA approach is important and is popularly used in wireless networks (Friend et al. 2008; ElNainay et al. 2009). It works by separating the population into sub-populations,

called islands. These islands cooperate with other islands through the migration of individuals. In island GA, the GA runs on each node with the possibility of good solutions migrating between nodes. These migrations are performed in accordance with the migration policy that controls the 'when' and 'how' of the individual movement. In the island GAs, there is relatively infrequent migration between island models thereby implying reduced inter-agent communication requirements compared to master–slave GA and cellular GA Alba and Troya 1999.

2.4 GA components and operators

A flow chart of a typical GA is provided in Fig. 2. The operation of a GA-based framework starts by initially defining a population of candidate solutions. While it is acceptable, and indeed commonplace, to define the population randomly, the convergence process can be accelerated by starting with an appropriately defined population. Each candidate solution defined in the population is known as an individual, with each individual encoded in an abstract format named as a chromosome (in a metaphorical reference to the encoding role of chromosomes in biological genetics). Various evolutionary techniques (mutation, crossover, etc.) are then applied on the population. In every generation, multiple individuals are stochastically selected from the current population with fitter individuals being more likely selections (due to the genetic principle of 'survival of the fittest'). The selected individuals are then genetically modified (mutated or recombined) to form the next generation of the population. The usage of

genetic operators and stochastic selection allow a gradual improvement in the 'fitness' of the solution and allow GAs to keep away from local optima.

In the following subsections, we will provide more details about the various steps involved in problem solving through GAs.

2.4.1 Encoding into chromosomes

Genetic algorithms operate by initially defining a population of candidate solutions (each of which is called an *individual*). Individuals are encoded in an abstract representation, known as a chromosome, (which may be problem specific although representation in strings of 1s and 0s is common; in the related field of 'genetic programming', the encoding is in terms of snippets of computer code). The population size can be a significant factor in GA performance and efficiency. In particular, the performance of GAs can be poor when the population size is very small (Grefenstette 1986).

The encoding of candidate solutions in the form of chromosomes can also be of variable length. Such variable length encoding is well suited to problems such as shortest-path routing problem that compares paths of different lengths. As an example of variable length chromosome encoding, refer to the work of Ahn and Ramakrishna (2002) in which the routing path is encoded in the chromosome as a sequence of positive integers representing node IDs of the nodes enroute the end-to-end routing path. Each locus of the chromosome corresponds to the order of the node in the end-to-end routing path with the source node always represented by the gene of first locus. The length of the chromosome is variable but is upper bounded by N (the total number of network nodes).

Several encoding schemes for GA exist:

(a) Binary encoding: Binary encoding is the most common approach, mainly because initial GA-based works used this encoding. In binary encoding every chromosome is a string of bits, i.e. 0 or 1. Binary encoding provides many possible chromosomes even for a small amount of alleles. On the other hand, this encoding is in some cases not natural for many problems and in other cases corrections need to be made after mutation or crossover.

(b) Octal encoding: The chromosome is represented in octal numbers (0–7).

(c) Hexadecimal encoding: The chromosomes are represented using hexadecimal numbers. (0–9, A–F).

(d) Permutation encoding: Permutation encoding can be used in GA for ordering problems, such as task ordering problems, or the 'traveling salesman' problem. In permutation encoding, each chromosome is a represented as a string of numbers, which is actually a number in a sequence. In some problems, for some types of mutation and crossover, corrections must be inculcated to leave
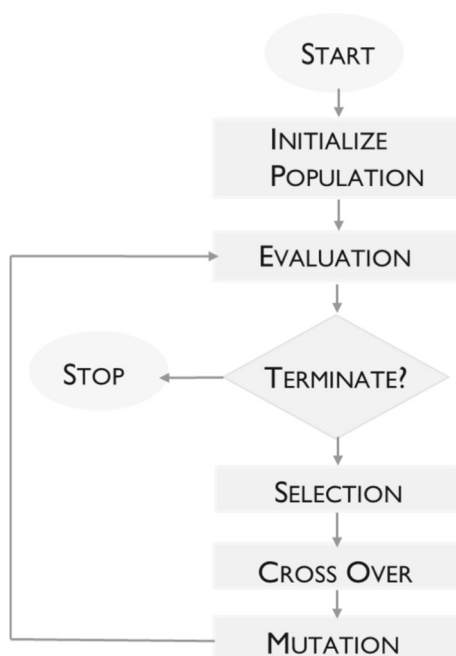
Fig. 2 A flow chart of a typical genetic algorithm (Qadir 2015)

the chromosome in a consistent form, viz. it should be corrected to ensure that it has a real sequence in it.

(e) Value encoding: Direct value encoding is useful in problems that deal with complicated values (such as a real number). Use of binary encoding method in this case would be very difficult. In value encoding, every chromosome is a string of some values. Values can be in any arbitrary form including numbers, characters, or any other complicated object. While value encoding is a good approach for some problems, it often requires new crossover and mutation operators defined specifically for the problem.

(f) Tree encoding: Tree encoding is used for expressions or evolving programs, specifically for genetic programming (GP). In tree encoding, every chromosome or gene is in a tree form of some objects, like commands in programming language or a function. Programming language like LISP is mostly used for this, because genetic programs in it are encoded in this form and can be easily parsed as a tree structure, thereby facilitating mutation and crossover to be performed relatively easily.

For illustrative purposes, an example binary encoding, permutation encoding, value encoding, and tree encoding is presented in Fig. 3.

Encoding example: Consider an example encoding of radio parameters in chromosomes and genes for GAs. The inherent

| Chromosome A | 101100101100101011100101 |
| Chromosome B | 111111100000110000011111 |

(a) Binary Encoding

Chromosome A	1.2324 5.3243 0.4556 2.3293 2.4545
Chromosome B	ABDJEIFJDHDIERJFDLDFLFEGT
Chromosome C	(back), (back), (right), (forward), (left)

(b) Permutation Encoding

| Chromosome A | 1 5 3 2 6 4 7 9 8 |
| Chromosome B | 8 5 6 7 2 3 1 4 9 |

(c) Value Encoding

Chromosome A   Chromosome B

$(+ \times (/ 5 y))$ (do_until step wall)
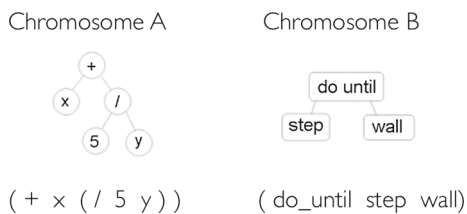
(d) Tree Encoding

Fig. 3 Illustration of basic types of encoding schemes used by GA (Adapted from Sivanandam and Deepa (2008))

difficulty in such a representation is the large relative difference between radio parameters. For example, frequency range of 1 MHz–6 GHz with step size of 1 Hz would require more than 30 bit genes, while modulation has only handful of techniques so it would require far less bits. Chromosomes can, therefore, utilize large number of bits for frequency while utilizing small number of bits for the modulation gene. To make GAs independent of which radio it is optimizing, the variable bit chromosome representation is suggested by Scaperoth et al. (2006) in the context of software-defined radios (SDR).

2.4.2 Fitness function

The fitness function, sometimes called the objective function, offers a mechanism for the evaluation of the individual chromosomes. Fitness function can be single objective, or multi-objective. The fitness function is devised in a way that the individual chromosome is its variable input parameter. Fitness function ensures that the chromosomes passed on to the next generation are not violating any constraints. It assigns a fitness score to each individual depending on the quality of the individual in terms of the optimization goals and objectives. In particular, for the fields of both GAs and genetic programming, each design solution is represented as a string of numbers referred to as the chromosome. In fitness evaluation, after each round of completed simulation or testing, N worst design solutions are deleted, and N new ones from the best design solutions are created. Each design solution thus needs to be awarded a figure of merit, which indicates how close it came to meet the overall specification. This is achieved by applying the fitness function to the simulation or test results obtained from that solution.

Genetic algorithms require much effort in designing a workable fitness function. Though a computer might generate the final form of the algorithm, it takes a human expert to define the fitness function since an inefficient function can result in convergence to an inappropriate solution, or no convergence at all. Speed of execution of the fitness function is very important, since a typical GA needs to be iterated many times to produce a usable result for a non-trivial problem. Hence, fitness function requirements include the following:

- Minimum fitness computation time of candidate solution.
- Precise model definition for fitness computation.
- Removal of noise from the fitness function values.

Two main categories of fitness functions exist: one where the fitness function does not change over time (as in the case of optimizing a fixed function, or testing with a fixed set of test cases) and the other where the fitness function is mutable (as in the case of co-evolving the set of test cases or in niche

differentiation). Another way of defining fitness functions is in terms of a fitness landscape that depicts the fitness level for each possible chromosome. Definition of the fitness function may not be straightforward in many cases. In some scenarios, it becomes very hard or impossible to come up even with a guess of what fitness function definition might be. This difficulty is solved by 'interactive GA', which outsources evaluation to external agents including humans. The function is calculated iteratively if the fittest solutions produced by GA are not in the desired range.

There are various strategies for assignment of fitness values to the subject population. The 'aggregation-based' strategy sums the objectives into a single parameterized objective value, e.g., an aggregation of weighted-sum. The 'criterion-based' method transitions between the objectives during the selection phase of the algorithm. Every time an individual is chosen for reproduction, a potentially different objective will dictate which member of the population will be copied into the mating pool. In the 'Pareto dominance based' methods, different approaches like dominance depth, dominance count, and dominance rank are used. The dominance rank dictates the number of individuals by which certain individual is dominated. In the dominance depth, the population is divided into several fronts and the dominance depth reflects the front to which an individual belongs. The dominance count is the number of individuals that are dominated by a certain individual.

Fitness function example: The evaluation of the solution is at the heart of GAs since it is on the basis of fitness values that GAs evolve solutions and remove poor individuals. It can be thought of as the performance metric for the optimization algorithm. As an example, for the routing problem, the fitness can be equated with high average throughput ($f_i = t_i$: where f_i is the fitness of ith individual and t_i is its average throughput), low delay ($f_i = -d_i$) where d_i is delay perceived by the ith individual. For classification problems, the fitness function can be some measure of the error. Realistic fitness functions are typically much more complex than this toy example, and can depend on many factors.

2.4.3 Mutation

The mutation operator is used for exploration with randomly altering genes in chromosomes to discover new horizons and new traits. Mutation helps to diversify the population, thus helping GAs to avoid local optima and pave the way towards global optima. Several types of mutation techniques exist including:

- Bit-string mutation: The mutation of bit strings occurs, through bit flips at random positions of chromosomes (as illustrated in Fig. 4).
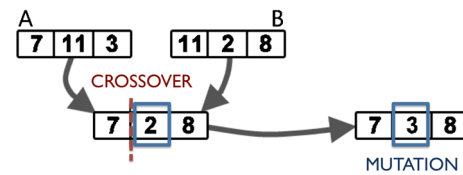

Fig. 4 Illustration of mutation and crossover operation

- Flip bit: This mutation operator takes the chosen genome chromosome and inverts the bits, i.e. if the genome bit is 0, it is changed to 1 and vice versa.
- Boundary: This mutation operator replaces the chromosome with either upper or lower bound in a random manner. This can be used for float and integer representation of chromosomes.
- Non-uniform: The probability that the extent of mutation will go down to 0 with the next generation of chromosome is increased by using non-uniform mutation. This keeps the population from stagnating at early stages of the evolution process. It also alters solution in later stages of evolution. The mutation operator can only be used for float and integer chromosome genes.
- Uniform: This operator replaces the score of the chosen chromosome with a uniform random value selected in the range of the user-specified upper and lower bounds. This mutation operator can only be used for integer and float genes.
- Gaussian: This operator adds a unit Gaussian distributed random value to the selected chromosome. If it falls outside of the boundary of user-specified upper or lower bounds for that chromosome, the new value is clipped. This mutation operator is used for integer and float strings.

2.4.4 Crossover

The crossover process is essentially an attempt to exploit the best traits of the current chromosomes and to mix them in a bid to improve their fitness. This operator randomly chooses a locus and exchanges the sub-sequences before and after that locus between two parent chromosomes to create a pair of offspring. A crossover point is randomly chosen. There can be multiple crossover points to aid more exploration. In biological systems, crossover is much more rampantly observed than mutation, often as much as a million times more frequent (Holland 1995). Crossover along with mutation provides the necessary 'evolutionary mix of small steps and occasional wild gambles' to facilitate robust search in complex solution spaces (Harford 2011).

In GAs, crossover is viewed as synthesis of best practices and mutation is viewed as a method of spontaneous inspiration and creativity. The evolution can start from a population

of completely random individuals and can evolve to better solutions through survival of the fittest after application of genetic operators. The selected individuals (or chromosomes) are genetically modified (mutated or recombined) to form the next generation of the population. The usage of genetic operators and stochastic selection allow a gradual improvement in the fitness of the solution and allow GAs to keep away from local optima.

Several methods exist for crossover of chromosomes which use different data structures (Goldberg and Holland 1988):

- One-Point crossover: A single crossover point on both parent's chromosome strings is selected. All data beyond that point, in either organism chromosome are swapped between the two parent strings. The resulting strings are the new children chromosome.
- Two-Point crossover: Two-point crossover requires two points to be selected on the parent chromosome strings. Everything between the two points is swapped between the parent chromosome strings, producing two child chromosomes.
- Uniform crossover and half uniform crossover: The uniform crossover approach uses a fixed mixing ratio between two parents strings. Unlike one-point and two-point crossover, this approach enables the parent chromosomes to contribute at the whole chromosome gene level rather than at the segment level. For example, if the mixing ratio is 0.5, the offspring chromosome has approximately half of the genes from first parent and the other half from other parent, although crossover points can also be randomly chosen instead. In the half uniform crossover method, exactly half of the non-matching bits are swapped between the chromosomes. First the number of differing bits, or the *Hamming distance*, is calculated. This number is then divided by two; the resulting number depicts the number of bits that do not match between the two parents, and will thus be swapped. An illustrative example comparing one point, two points, and uniform crossover is presented in Fig. 5.
- Cut and splice: In the crossover variant called the 'cut and splice' approach, a change in length of the children strings occurs. The reason for this difference in length is that each parent string has a separate choice of deciding crossover point.
- Three parent crossover: In this method, the child chromosome is derived from three parents which can be randomly chosen. Each bit of first parent is compared with bit of second parent to see if they are the same. If the bits are same then that corresponding bit is taken for the offspring; otherwise, the bit from the third parent is taken for the offspring chromosome.
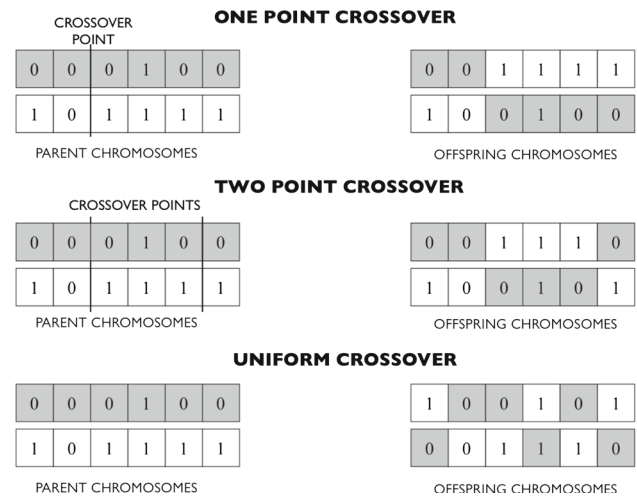
Fig. 5 Illustration of examples of one point, two points, and uniform crossover methods (Adapted from Sastry et al. (2005))

- Ordered chromosome crossover: In some cases, a direct swap may not be possible. One such case is when the chromosomes are in an ordered list. The N-point crossover can be used for ordered chromosomes, but always requires a corresponding repair process. Some of the approaches for ordered pair crossover include cycle crossover, position-based crossover, alternating position crossover, edge recombination crossover, sequential constructive crossover, and voting recombination crossover.

2.4.5 Selection

Selection in GA allows individual genomes to be chosen from a population for later recombination or crossover (breeding). This operator performs the selection of chromosomes from population for reproduction. The fitter the chromosomes, the more likely they are to be passed on to the next generation. A number of selection techniques has been employed in GAs to carry out the survival of the fittest from a pool of individuals (Fette 2009).

A common selection procedure may be implemented in the following way:

- The fitness function is evaluated for each individual and is then normalized for algorithm use. The fitness value of each individual chromosome is typically normalized by the aggregation of all fitness values to ensure that all the resulting fitness values sums up to 1.
- The population is then sorted in descending order of fitness values.
- Accumulated normalized fitness scores are calculated. Accumulated fitness value of an individual equals the sum of fitness scores of all the previous individuals and

its own fitness score. Accumulated fitness of the previous individual must always be 1—any other value can be used as indication of a wrong normalization procedure.

- A random number R is chosen in the range 0 to 1. The first individual whose summed up normalized value is greater than the number R is selected.

This selection method is called fitness proportionate selection or roulette-wheel selection, since the procedure is repeated until there are enough selected individuals. If instead of using a single pointer that is spun a number of times, multiple equally spaced pointers are used on a wheel that is spun once, the selection process will be called 'stochastic universal sampling'. The selection process will be a tournament selection process once there is repeated selection of the best-fitted individual from a randomly chosen subset pool. This will be further called truncation selection once we take the best half, third or some other proportion of the individuals.

There are also other selection mechanisms that do not take into account all individuals for the selection process, but only those that have a fitness value, greater than a given constant. Other algorithms select the individual from a restricted pool where only a constrained percentage of the individuals is allowed, based on their fitness scores. Retaining the best individual chromosomes in a generation unaltered, in the next set of generation, is called elitist selection or elitism. It is considered a successful variant of the general procedure of constructing a new population pool.

The main selection techniques used by GAs are summarized in the following list:

- Relative tournament evaluation: In relative tournament, two parents are compared randomly and the one with high fitness value is selected to be the parent of next generation. Winning chromosomes are awarded by adjusting their weight corresponding to the objective function. This weighting increases the importance of that chromosomes relative to others.
- Roulette wheel selection: Roulette wheel selection is a way of choosing mother and father chromosomes from the population in a way that is proportional to their fitness. The chromosomes with higher fitness have high chances of being selected using this technique.
- Relative pooling tournament evaluation: In this technique, each parent is thrown in a competition with the individual chromosomes in population, independently. The individual that is the winner of maximum number of tournaments gets a chance to be passed into next generation through reproduction.
- Elitism: Elitism is a method to store and use previously found best-suited solutions in subsequent population generations of GA. In a GA approach that uses elitism, the statistic related to the population's best solutions does

not degrade with generation. Maintenance of archives for non-dominated solutions is an important and critical issue. The final contents of archive usually represent the result returned by applying the optimization process. It is highly effective and common to employ the archive as a pool, from which guidance is taken for the generation of new solutions.

3 Applications of GAs in wireless networking

After providing background on GAs in the last section, we are now in a position to survey the applications of GAs in wireless networking. GAs have been applied in wireless networking in a wide variety of contexts. We will first describe the various applications of GAs in wireless networking in Sect. 3.1. We will then provide a network-type based classification of GA application in Sect. 3.2. Finally, we will discuss 'hybridization' proposals that combine GAs with other techniques in Sect. 3.3.

3.1 Application-based classification

In this section, we will discuss the applications of GAs in wireless networking. In particular, we will discuss routing, QoS, load balancing, channel allocation and assignment, localization, WLAN AP placement, and other general applications, in this particular order. The organization of this section is summarized in the Table 2.

3.1.1 Routing

The problem of shortest path routing problem is a well-studied problem in literature. GAs are well suited to the routing problem allowing convenient mechanisms for developing adaptive routing solutions. GAs have been used for developing routing solutions in a variety of wireless settings as summarized in Table 3. Ahn and Ramakrishna (2002)

Table 2 Applications of genetic algorithms in wireless networking (covered in Sect. 3.1)

Application	Discussed in
Routing	Section 3.1.1 and Table 3
Quality of service (QoS)	Section 3.1.2 and Table 4
Load balancing	Section 3.1.3 and Table 5
Bandwidth allocation	Section 3.1.4 and Table 6
Channel allocation and assignment	Section 3.1.5 and Table 7
Localization	Section 3.1.6 and Table 8
Wireless AP placement	Section 3.1.7 and Table 9
General applications	Section 3.1.8 and Table 10

Table 3 Using genetic algorithms for routing in wireless networks

Project and references	Network	No. of objectives	Technique used	Brief summary
Ahn and Ramakrishna (2002)	MANET	SOGA	Integer coded, variable length chromosomes	Unlike Dijkstra and Depth-First Search (DFS) algorithms, multiple solutions can be found for shortest path routing that have same cost using GAs
Cheng and Yang (2010)	MANET	MOGA	Distributed multipopulation GA	A distributed GA approach incorporating multi-population with random immigrant schemes solves the shortest path routing in MANET
Gen et al. (1997)	LANs	SOGA	Integer coded GA for random distributions	A priority encoding scheme is proposed using GAs to represent and optimize all possible paths in network graph
Davies and Lingras (2003)	WAN	SOGA	Simple GA with unbiased coding scheme	A GA approach continuously re-routes and assigns weight to traffic paths to implement adaptive control over dynamic routes and network topology
Yang et al. (2010)	MANET	SOGA	GA with immigrants and memory scheme	A GA proposed for investigating effectiveness and efficiency of GAs with immigrants and memory schemes in solving the dynamic shortest path routing problem in the real-world networks
Badia et al. (2009)	WMNs	SOGA	ILP formulation with GAs	GA addressed the routing and link scheduling problem when ILP cannot cope with computational complexity of large WMNs
Al-Karaki et al. (2009)	WSN	MOGA	Centralized GA and ILP	A metaheuristic algorithm that solves the joint problem of optimal routing with data aggregation to optimize the network lifetime while retaining the energy and latency efficiency
Lorenzo and Glisic (2013)	CN	MOGA	Sequential GA	A sequential GA solves the NP-hard problem by optimizing relay topology while eschewing the inter-cell interference
Apetroaei et al. (2011)	WSN	SOGA	Tree coded GA on spanning tree topology	GA inspired spanning tree topology for WSNs that adapts (according to the nodes, residual energy) to maximize the usage of the network
Yen et al. (2008)	MANET	SOGA	Prüfer number encoding	A GA strategy for energy efficient routing by maintaining the energy balanced nodes in entire MANET distribution
Chiang et al. (2007)	MANET	SOGA	Tree sequenced coding	A GA based on sequence and topology encoding employed topology crossovers to solve the multicast routing constraints efficiently

have tackled this problem through GAs. The routing table for each source–destination pair needs to be constructed first. Obviously, the number of possible routes between two nodes heavily depends on the network topology. If the network is densely connected or the size of the network is large, the number of possible routes of a source–destination pair becomes huge. Hence, it is impossible to list all the possible paths in the routing table. A predefined number of routes are randomly generated and fed to population pool, and chromosomes are then optimized. In the simulation, crossover and mutation are run on variable length chromosomes. Infeasible solutions are also kept in check. The paper discussed the issues of path-oriented encoding, and path-based crossover and mutation, which are relevant to the routing problem.

Yang et al. (2010) studied how GAs can be used for dynamic shortest path routing problem (DSPRP). Algorithms like Bellman-Ford and Dijkstra perform well in fixed infrastructure but on dynamic scale, high computational complexity renders them unacceptable. A DSPRP model is built

up in this paper using GAs. A specialized GA is designed for the shortest path problem in MANETs. Several immigrants and memory schemes that have been developed for GAs for general dynamic optimization problems are adapted and integrated into the specialized GA to solve the DSPRP in MANETs. The experimental results indicate that both immigrants and memory schemes enhance the performance of GAs for the DSPRP in MANETs.

In MANETs, all the nodes communicate over wireless links without any established infrastructure. MANETs are well suited to environments in which there is no fixed infrastructure or when the infrastructure is not trusted. In such networks, a common strategy is to form clusters where each node is linked to a clusterhead for efficient routing with other nodes that are not in its direct range. GAs have been used in such cluster-based routing schemes for MANETs. Al-Ghazal et al. (2004) have proposed a GA-based protocol named 'cluster gateway switch routing protocol' (CGSRP) to select the clusterhead to carry out communication between nodes.

Clusterhead must have enough resources, power, and bandwidth to avoid dangers of bottlenecks. This scheme works by encoding each node's unique ID in the chromosomes. The chromosomes have information about cluster-head, members, and number of links in each cluster-head. The encoded chromosomes are then evaluated against certain criteria as defined by the fitness function (which may incorporate features such as load balancing and bandwidth conservation). Each chromosome's fitness is then evaluated. The process of the survival of the fittest leads to an optimum selection of nodes as cluster-heads that optimally utilize resources.

Genetic algorithms have also been proposed for use in multicast and broadcast routing in wireless networks. A study on multicast routes optimization using GAs is done by Banerjee and Das (2001). An on demand routing scheme with GAs is discussed to meet bandwidth, delay constraints while maintaining the QoS in wireless networks. Chromosomes from different pools are randomly concatenated to obtain a multicast solution. In doing so, all possible routes between source and destination nodes are encoded and fed to GA in the form of population chromosomes. GA then uses evolutionary techniques to select the best chromosomes by evaluating them against a particular fitness function, which can be in the form of different constraints related to bandwidth, QoS, and channel allocation, etc. In another work, Salcedo-Sanz et al. (2003) proposed mixed neural-genetic algorithm for the 'broadcast scheduling problem' (BSP). The algorithm solves this optimization problem in two stages: first, it finds a feasible solution, and second it obtains maximal transmission throughput by employing fittest chromosomes through GAs. A track of interference and frame matrix are kept and the channel slots are assigned accordingly. Nodes that are 1-hop away are not assigned the same slot to avoid mutual interference. After finding feasible solutions, these solutions are encoded in chromosomes. Multiple GA operators—such as mutation, crossovers—are then applied leading to maximized throughput.

The use of hybrid GAs have been proposed for routing in wireless networks. NK and Viswanatha (2009) proposed using ant algorithms, combined with GAs, for route discovery. It was observed that the use of ant algorithm leads to reduction in the size of the routing table. We note here that ant-colony-optimization (ACO) algorithms belong to the broader class of AI known as swarm intelligence. While GAs cannot use global information of the network, the use of ACO and GA techniques in tandem leads to independent exploration of the network helping in effective computation of routes between any pair of nodes. The proposed algorithm creates an initial population, determines the forward-ants, generates new populations using genetic operators and fitness evaluation, and finally updates routing table. The hybrid algorithm performs well in not only generating routes between pair of nodes but also for achieving efficient load balancing.

3.1.2 Quality of service (QoS)

With increasing prevalence of multimedia applications (such as video conferencing and streaming), and the development of high-speed wireless communication technologies, efficient and reliable QoS provisioning has become increasingly important. Modern multimedia applications place rigorous QoS demands for different parameters like bandwidth, delay, and jitter (Malik et al. 2014). This motivates the development of effective resource allocation schemes that can ensure QoS while also satisfying these strict QoS constraints. In this section, we will describe the various GA-based QoS solutions that have been proposed in literature. This information is presented in summarized form in Table 4.

Various analytic approaches have been proposed in literature including those that rely on Semi-Markov decision process (SMDP) theory for optimal call admission control (CAC) (Choi et al. 2000). These methods unfortunately fail to scale to large-scale networks where the decision and action space is large. GAs can provide near optimal solutions in large-scale complex networks. Xiao et al. (2000) presented an effective near-optimal GA-based CAC approach for large-scale wireless/mobile networks. The chromosomes encoded are binary strings where 1 and 0 represent the acceptance and rejection of calls by CAC. These chromosomes are randomly generated and then evaluated against fitness function of channel utility and constraints. Evolution of best traits leads to near-optimal solutions for QoS. This scheme offers adaptive QoS to multi-class users while being computationally less intensive compared to the SMDP approach.

A multi-objective GA (MOGA) approach on multicast QoS routing is proposed by Roy et al. (2002). To deliver on-demand QoS, different solutions are kept in an optimized pool. Multiple solutions are offered depending upon the service demands. A MOGA approach optimizes the prioritized objectives as stated in the user QoS requirements. A similar approach for QoS-aware web-based service under strict constraints and optimizing the targeted QoS parameters is done by Canfora et al. (2005). It is shown in this study that the widely adopted optimization approach of linear integer programming is unable to offer non-linear QoS functions. GAs can be employed to work with non-linear QoS attributes. In yet another MOGA work, Roy et al. have proposed QM^2RP as a QoS-based multicast routing framework for mobile wireless cellular networks (Roy and Das 2004). The proposed protocol determines near-optimal routes on demand while working with potentially inaccurate network state information. The protocol considers a number of QoS optimization metrics including end-to-end delay, bandwidth, and residual capacity.

With the increase in number of service candidates, the traditional chromosome model for QoS optimization becomes

Table 4 Using genetic algorithms for QoS optimization in wireless networks

References	Goal	No. of objectives	Technique used	Brief summary
Resources scheduling for QoS				
Sherif et al. (2000)	Resource allocation with call admission control	SOGA	GA with reallocation of unused bandwidth	Accelerate allocation through crossover and mutation. Reduces computation of NP-hard problem. Dynamic assignment of network resources and bandwidth
Gözüpek and Alagöz (2011)	Scheduling under interference constraints	MOGA	Binary encoding with random seeding	GA-based sub-optimal schedulers for maximizing the throughput and improving the latency in CRNs
Cheng and Zhuang (2009)	Packet level resource allocation	SOGA	Karush-Kuhn Tucker (KKT) and GA driven approach	An effectual hybrid GA optimizes the QoS provisioning and resource allocation to keep a balance between performance and computation
Lopez et al. (2014)	Transmit power control for maximum utility	MOGA	GA with call admission control	MOGA approach adopted to maintain the transmit power of CRs, while maintaining the QoS and optimizing the spectrum access
Roy et al. (2002)	Energy-efficient resource scheduling	MOGA	Depth first search with NSGA	On demand QoS routing scheme has been proposed based on end user needs. A solution meeting the constraints is offered from an optimized pool of chromosomes
QoS for multimedia and web-services				
Jiang et al. (2011)	Multi-Constrained QoS for web-services	SOGA	Variable length chromosome GA	A scalable QoS aware web service composition is introduced using GA for diversified functionality through cut and splice operation of mutation and crossover
Canfora et al. (2005)	QoS service composition	SOGA	GA with ILP and elitism	GA approach for web based QoS services with non-linear aggregate functions while meeting a set of constraints
Li et al. (2006)	Video On Demand	MOGA	GA with interference model	GA is employed to leverage multi-source, multipath solution to design a concurrent video streaming environment in WMNs, along with route discovery
Energy-efficient QoS multicast routing				
EkbataniFard et al. (2010)	Energy efficient QoS routing in clusters	MOGA	NSGA-II	A hierarchical energy efficient routing protocol, for two tier WSN networks, under tight QoS constraints, is developed using MOGA optimization with NSGA-II (Deb et al. 2000)
GAMAN	Topological Multicasting	SOGA	Tree coded GA with multi-population	A GA for MANETs (GAMAN) is proposed by Barolli et al. (2003). The algorithm works locally on smaller population and offers promising QoS for ad-hoc networks
QM^{2}RP (Roy and Das 2004)	QoS-based multicast routing	MOGA	Tournament selection and niched Pareto optimization	A QoS-based mobile multicast routing protocol using multi-objective genetic algorithm
Cui et al. (2003)	Optimizing conflicting QoS parameters	MOGA	Depth-first search algorithm and GA	A MOGA approach is used to optimize multiple multicast routes lying on Pareto front
Lu and Zhu (2013)	Energy-efficient QoS multicast routing	MOGA	Tree structured coding	Proposed GA solves QoS by constructing a multicast tree with a minimum energy cost and bounded end-to-end delay
Yen et al. (2011)	Multicast routing through flooding	MOGA	Tree coded flooded GA	A GA-based method is proposed for multi-constrained QoS multicast routing, while incorporating elitism, to optimize the computation time
Huang and Liu (2010)	QoS Aware Multicast Routing	MOGA	Greedy and family competition with GA	A hybrid GA along with greedy algorithm mitigate premature convergence problems to avoid local optimas

Table 4 continued

References	Goal	No. of objectives	Technique used	Brief summary
Sun et al. (2008)	Multi-constrained optimization	MOGA	Spanning tree encoding	The evaluated GA can optimize the cost of the multicast tree, maximum link utilization, the average delay selection of the long-life path
Abdullah (2010)	Modified QoS Routing Protocol	MOGA	Variable length encoding	A GA approach look for the best QoS route, using route mutation, to optimize the design of MANET routing protocol by improving packet delay and throughput
Route discovery				
Gelenbe et al. (2006)	Autonomous route discovery with GA	SOGA	GA with distributed CPN protocol	Proposed autonomous route discovery in cognitive packet networks (CPNs) with GAs. Paths discovered by CPN are provided to GA which then produces new routes using evolutionary techniques
NK and Viswanatha (2009)	Route discovery through mobile agents	SOGA	GA with Ant algorithm	A hybrid algorithm (combining ant algorithms and GA) is proposed to reduce the size of routing table and help in routing among autonomous peers
Queue management at routers (AQM)				
Di Fatta (2003)	Designing FPI controller for AQM	SOGA	GA with fuzzy logic	Fuzzy proportional integer (FPI) controller is purposed and FPI parameters are controlled by GAs dynamically to optimize active queue management (AQM) policies
Selamat et al. (2004)	Query Retrieval for active AQM	SOGA	Integer coded distributed GA	A GA with mobile agents search system (MaSS) is proposed to minimize the query retrieval cost while maintaining a reasonable path delay

cumbersome in web services. To address this, Gao et al. (2007) studied a 'tree based GA' for optimizing QoS. This model is faster than traditional GA since it offers impulsive chromosome coding and decoding and present run time integration of QoS resources. Chromosomes are represented in series of nodes and all the leaf nodes execute the process. A tree characterizes the structured model for different QoS composition and requires less computation for large networks. Sub-tree crossover exchanges the task nodes and updates the QoS vector from the leaf nodes to the tree's root. A standardized and comprehensive fitness function verifies the whole QoS process plan. Results have showed that the tree-based models have lesser overhead than the traditional GAs with serial chromosomes while offering excellent replanning and service composition.

3.1.3 Load balancing

GAs have been proposed for use in various load balancing solutions for various wireless networking configurations in literature (as shown in Table 5).

There are three main types of load balancing categories: (i) strongest 'received signal strength indicator' (RSSI); (ii) least loaded first (LLF); (iii) hybrid load balancing (HLB) incorporating RSSI and LLB. These techniques fall under the category of network-centric load balancing, and are applied on a network scale to achieve global optima, unlike user-centered techniques that are more likely to achieve local optima only. Chromosomes consist of randomly assigned individuals to an access point. Crossover and mutation are performed on individuals and fitness evaluation is done. Both load balancing and congestion control problems have been addressed. QoS has also been addressed by assigning different weight to users, depending on its service requirement like streaming and media, etc. Experimental results and analysis showed that microGA outperforms macroGA at first but in the long run macroGA does better load balancing by pushing more congested users towards peripheral APs.

Ozugur et al. (2001) proposed the multi-objective hierarchical mobility management optimization for UMTS networks to distribute the load among different parts of network such as serving GPRS nodes (SGSN), radio network controllers (RNC), and mobile switching centers (MSC) in addition to performing the role of minimizing cost for location update in these intricate networks.

Selecting resource-rich nodes for trivial tasks creates an imbalance in energy distribution among nodes resulting in the demise of multicast service. To mitigate such a possibility, Yen et al. (2008) have proposed an energy-efficient GA routing in MANETs. The algorithms proposed in Yen et al. (2008) continuously observe the condition at various nodes and improve the energy balance by replacing the weaker nodes with energy-rich nodes. An extended Prüfer encoding scheme is proposed for energy-efficient exchange

Table 5 Using genetic algorithms for load balancing in wireless networks

References	No. of objectives	Technique used	Brief summary
Wireless mesh networks			
Hsu et al. (2008)	SOGA	GA with Dijkstra algorithm	A hybrid GA coupled with Dijkstra's algorithm look for the best available paths with bounded delays and low cost network configuration
Zeng and Chen (2008)	SOGA	GA with greedy algorithm and binary encoding	Load balanced disjoint clustering is done by a GA coupled with *greedy algorithm* for efficient distribution of network resources
Mobile Ad-hoc Networks			
Yen et al. (2008)	MOGA	GA with Prüfer number coding	In time-critical load balancing scenario, GAs inspired strategy is applied in multicast tree to balance the energy consumption, leading to prolonged network lifetime
Cheng (2013)	SOGA	Distributed GA with multi-population and immigrant schemes	A dynamic GA evaluates the fitness of each feasible clustering structure against load balancing metric
Wireless sensor networks			
He et al. (2012)	SOGA	1-to-1 mapping of nodes to genes in chromosomes	The application of GA to find a load-balanced connected dominating set (CDS) to be used as a virtual backbone of WSNs
Wireless local area networks			
Scully and Brown (2009)	SOGA	MicroGA & MacroGA	A comparison of MicroGA and MacroGA is done against QoS attributes like throughput and load balancing. MicroGA performs better in brisk load balancing under time constraints

of leaf nodes through mutation and crossover. In this way a substantial reduction in convergence time is achieved, thus prolonging the service duration efficiently.

Genetic algorithms have also been proposed for load balancing at APs in IEEE 802.11 WLANs for efficient distribution of bandwidth among users. Scully and Brown (2009) have discussed MicroGAs and MacroGAs to address the problem of load balancing. A comparison of MicroGAs and MacroGAs has been done by Krishnakumar (1990). Micro-GAs has a population of five individuals leading to brisk evolution.

3.1.4 Bandwidth allocation

The problem of bandwidth allocation under QoS constraints in wireless networks is typically a complex task. A proper bandwidth allocation scheme must be adopted to overcome the problem of limited wireless resources and utilize the channel allocation efficiently. Various GA-based approached have been proposed in literature as solutions to the problem of bandwidth allocation in wireless networks. We will describe a sample of these works (a tabulated summary is presented in Table 6.

An accelerated GA approach with dynamic adjusting of mutation and inheritance probability is proposed in Lin and Labeau (2012) as a solution to the bandwidth allocation problem in WLANs. The probability of mutation and crossover is made dependent on the individual chromosomes fitness.

This approach is shown to quickly converge to global optima while substantially reducing the number of generations as well as the computation time. In another work, Sherif et al. (2000) proposed a GA-based approach to analyze the network bandwidth allocation using a predetermined range of QoS levels (high, medium, low) declared by each multimedia substream. The proposed algorithm assigns the best available QoS level to each substream depending upon the availability of bandwidth resources in network. In the case of network congestion, the algorithm tries to preserve resources by degrading the quality of some admitted calls. Using this technique, the algorithm attempts maximum resource utilization and fair distribution of bandwidth in all the networks while keeping in view the network constraints.

3.1.5 Channel allocation and assignment

With the rapid proliferation of wireless and cellular networks, and the problem of limited frequency spectrum, the problem of channel assignment and reuse has become very important. The recent advancement in wireless technology has enabled many high-speed data services leading to an upsurge in number of wireless users. Since the frequency spectrum is a limited resource, there is a great emphasis on the reuse and sharing of frequency amongst the ever-increasing user base. The goal of channel allocation is to maximize frequency reuse by minimizing the number of channels required to cover all the network cells.

Table 6 Using genetic algorithms for bandwidth allocation in wireless networks

References	No. of objectives	Techniques used	Brief summary
Lin and Labeau (2012)	MOGA	GA with adaptive inheritance probability	A prioritized bandwidth allocation scheme for transmitting medical patients data is applied for multimedia sub-streams
Kandavanam et al. (2010)	MOGA	GA with variable neighborhood search approach	A hybrid GA optimizes the residual bandwidth allocation to routing links for multiclass traffic in all networks
Karabudak et al. (2004)	SOGA	GA with Markov decision model and elitism	The proposed GA scheme minimizes handoff latency by using elitism technique to allocate the maximum network resources to admitted calls
Vedantham and Iyengar (1998)	SOGA	GA with greedy algorithms	A GA approach to efficiently utilize the bandwidth for different incoming call streams to maximize the revenue generation
Riedl (2002)	SOGA	GA with local search heuristics	GA approach with bandwidth and delay metrics optimizes the IP traffic engineering in wireless networks
Kobayashi et al. (2004)	MOGA	Distributed GA	A distributed GA works by utilizing both local and global optimization in parallel on networks nodes to utilize the network resources and bandwidth
Al-Qahtani et al. (1998)	SOGA	GA for asynchronous transfer mode (ATM)	A genetic routing algorithm (GRA) is designed to allocate bandwidth to virtual path subnetworks in a wireless network

There has been a lot of GA-based work done in the area of channel allocation and assignment in wireless networks (a summary of which is provided in Table 7). We discuss some important GA-based channel allocation/ assignment works below.

Broadly speaking, there are three main constraints that channel assignment algorithms in wireless networks must consider: (i) co-channel interference constraints—which requires that the same frequency or channel cannot be assigned to two different radios simultaneously; (ii) adjacent channel interference constraints—which requires that adjacent frequencies not be assigned to adjacent channels simultaneously; and, (iii) co-site interference constraints—that requires minimum spacing between a pair of frequencies assigned to a channel must have minimum spacing.

Kassotakis et al. have proposed an approach based on hybrid genetic approach—that combines GAs with local improvement operator—for channel reuse in multiple access telecommunication networks Kassotakis et al. (2000). The proposed scheme aims to establish the maximal number of connections while minimizing the number of isochronous channels. Wong and Wassell (2002) investigated the channel allocation for a 'broadband fixed wireless access' (BFWA) networks. A channel allocation matrix keeps track of channel parameters for conflict-free assignments. It is observed that as the network grows, centralized scheme would offer unacceptable signaling overhead. GA is used for distributed channel where each access point (AP) transmits the interference power of the entire available channel to a control unit. Control unit employs GAs and ranks the chromosomes of channel priorities according to fitness value of interference. Crossover and mutation rate are also set to promote the best channels and explore the new chan-

nels. Experimental Statistics have revealed that better SNR gain (more than 14.5 dB) and throughput is achieved with GAs in comparison with the other two dynamic channel assignment methods: (i) least interference (Cheng and Chang 1999), and (ii) channel segregation (Akaiwa and Andoh 1993).

A study on GAs in channel assignment in cellular networks is performed by Kim et al. (1996). This paper reports that neural networks are more likely to get trapped on a local optima since a neural network's output critically depends on the initial values. In the case of GAs, they possess crossover and mutation as their key strength to escape local optimas. GAs, therefore, have far better chances to overcome the channel allocation problems such as (i) co-channel interference (ii) adjacent channel interference, and (iii) co-site interference. GAs are applied on clusters in cellular networks and it is showed that if the number of cells in one cluster are increased, the convergence rate is decreased leading to an increase in the number of generations due to the reuse of the same frequency among different cells. GA comes handy in keeping the cluster size optimum for efficient frequency distribution among cells.

Ngo and Li (1998) presented a modified GA to contribute in conflict free channel allocation and spectrum efficient assignment in cellular radio networks. There are generally two kinds of channel assignments: (i) fixed channel assignment (FCA)—where channel slots are pre-allocated to cells, and (ii) dynamic channel assignment (DCA)—where channels are assigned upon request. FCA generally outperforms DCA under heavy traffic load. This algorithm, called the genetic-fix algorithm, generates and manipulates individuals with fixed size, and hence greatly reduces the search space. Furthermore, using the minimum-separation encod-

Table 7 Using genetic algorithms for channel allocation/assignment in wireless networks

References	No. of objectives	Techniques used	Brief summary
Channel allocation in CRNs			
Zhao et al. (2009)	MOGA	Quantum computation with GA (Narayanan and Moore 1996)	A hybrid GA implemented to decrease the search space and ease the mapping process between the channel matrix and qubit chromosomes
Bhattacharjee et al. (2011)	MOGA	Comparison of GA with interference graphs	Channel assignment optimization for single cell cognitive network using GA
Friend et al. (2008)	SOGA	Distributed Island GA	An Island GA, solves the channel allocation through dynamic spectrum access in cognitive radio networks
ElNainay et al. (2009)	SOGA	Distributed Island GA	A localized island GA cooperates with other cognitive network nodes to utilize the available spectrum opportunistically and efficiently through channel allocation
Ali et al. (2012)	MOGA	Luby transform codes and GAs	A GA assisted resource management scheme is presented for reliable multimedia delivery
Channel allocation in cellular networks			
Zhenhua et al. (2010)	MOGA	Immuned GA with elitism	A modified immune GA, integrating immune operators, and minimum separation encoding to obtain the conflict free channel assignment in cellular radio networks
Pinagapany and Kulkarni (2008)	SOGA	Parallel GA with decimal encoding	A GA-based scheme used to solve both fixed and dynamic channel assignment problems in multi-cell cellular radio network
José-Revuelta et al. (2007)	SOGA	GA with Elitism and fitness entropy	Diversity guided Micro GA in cellular radios to optimize frequency/channel reuse in fixed channel assignment problem
Lima et al. (2007)	MOGA	GA with immigrants and memory schemes	Two adaptive GAs for locking and switching channel, solve the dynamic spectrum access problem through distribution of channels among cell in cellular networks
Channel allocation for broadband wireless networks			
Wong and Wassell (2002)	SOGA	GA-based dynamic channel allocation	GA-based technique incorporated channel segregation to maximize the throughput in broadband fixed wireless access networks
Fang and Chau (2006)	SOGA	GAs with video coding scheme	Evolutionary technique alongwith Reed-Solomon coding and SNR matrix allocate channel rates for scalable video transmission

ing scheme, the required number of bits for representing solutions is substantially reduced.

3.1.6 Localization

The task of determining the location of a wireless node is known as localization. Localization is important in many different settings (such as military monitoring services and disaster management) for providing location-aware services. GAs have been used to assist the localization process in various wireless settings. In this section, we will provide an overview of such work. A tabulated summary of some sample works is presented in Table 8.

A simple way to localize the nodes is to deploy GPS, but installing GPS in thousands of closely placed nodes is impractical. A better approach is to employ GPS in a few nodes, called anchor nodes, and apply techniques to locate all the other nodes relatively, with reference to multiple anchor nodes. Tam et al. (2006) proposed localization technique in which each node finds its position relative to three anchor nodes, thus enabling each node to locate itself through triangulation. This information is then broadcasted to base station where a centralized GA works as a post-optimizer, thus, leading to precise network graph with each node's location known. Zhang et al. presented the centralized GA with simulated annealing Zhang et al. (2008) to minimize the error in position estimation of nodes in WSNs. This approach is also based on anchor nodes localization scheme. Simulated annealing is local search technique and it avoids pre-mature convergence, by mixing inferior and sub-optimal solutions with the optimal ones. In the approach of centroid localization (Yun et al. 2008), proposed in Yun et al. (2008), GAs and fuzzy logics are used by each node to perform self-localization by estimating the edge weights based on the received signal strength (RSS).

3.1.7 WLAN AP placement

Since the arrival of IEEE 802.11, WLAN services are widely adopted with the predominant WLAN setting being

Table 8 Using genetic algorithms for localization in wireless networks

References	Goal	No. of objectives	Technique used	Brief summary
Zhang et al. (2008)	Node localization with random distributions	MOGA	GA with simulated annealing	A GA approach for centralized architecture to optimize the localization in WSN with least mean localization error
Yun et al. (2008)	Centroid localization	SOGA	GA with fuzzy logic	A hybrid GA with fuzzy membership is developed for edge weights to localize the nodes using received signal strength indicator (RSSI) information
Tam et al. (2006)	Localization with ad-hoc positioning system (APS)	SOGAs	MicroGA as post-optimizer with APS	A single objective MicroGA, using smaller population worked out best to improve the localization accuracy in APS for both isotropic and an-isotropic topologies
Nan et al. (2007)	Sensor nodes localization	SOGA	GA with elitism and partial map crossover (PMX)	A real-coded version of GA utilizes roulette wheel selection with Integer coding, thus, helping in achieving precise node localization in sensor networks
Yun et al. (2009)	Localization with RSSI	SOGA	GA and fuzzy logic system	Calculation of edge weight with reference of anchor nodes, followed by modeling through fuzzy logic functions and optimization by GA in the end

infrastructure mode (in which the topologies around APs). Optimizing the placement of WLAN APs plays a substantial role in maximizing the throughput and data rate of WLANs. In this section, we will summarize the various proposals for using GAs in WLAN AP placement. A tabulated summary is presented in Table 9.

A multi-objective GA (MOGA) approach for WLAN AP placement (with the objectives of minimizing the number of APs and maximizing SNR) has been described by researchers in Maksuriwong et al. (2003). GA-provided solutions have lesser variance implying that a larger number of area points within the network would have average SNR. Statistical analysis reveals that MOGA approach generate solutions with a high SNR data profile.

Optimizing the radio coverage inside buildings is as important as outdoors radio optimization. WLANs are mostly used inside homes and offices to provide mobile and Internet coverage to subscribers. Nagy and Farkas (2000) have studied the problem of optimizing the placement of indoor base stations (BS) while taking in account the effects of floors, ceiling, and walls on signal propagation and path losses. GA used the microwave propagation model to account for the effect of influence of brick, concrete, wood, furniture, and glass obstacles. Different weighting coefficients were assigned to each obstacle and parallel heuristic approach of GA is used to optimize the coverage results within indoor environment.

Another important application of GAs is in the optimization of AP placement in cellular and mobile networks. Since finding the set of coverage areas whose sum is the maximum covered area is an NP-hard problem, GAs come handy. Lieska et al. (1998) have explored in their study the combination of simulation and evolutionary techniques. Radio coverage areas are obtained by the series of BS with received power greater than −60 dbm. Algorithm works by selecting the chromosomes based on required BS. The chromosomes are than evaluated against constraints of power and SNR; afterwards, the healthy chromosomes are allowed to reproduce leading to global convergence of the entire network. This fitness function can be modified to include other constraints based on network requirement. A major advantage of a GA, therefore, is its ability to adapt to multiple situations and problems.

3.1.8 General applications

A survey for solving computationally difficult problem, through GAs, in wireless networks is presented by Das et al. (2006) in his book "Handbook of Bio-Inspired Algorithm and Applications". It has been shown that GAs have the potential to solve any optimization problem provided that proper modeling and chromosomes encoding has been performed. In this section, we will cover some of the other miscellaneous applications of GAs in wireless networking (that we have not covered in previous sections). A tabulated summary of these applications are presented in Table 10. We describe some of these applications here; some other applications are discussed when we cover network-configuration based categorization of GA applications in Sect. 3.2).

GAs have been proposed for energy efficient planning and management of cellular networks by enabling sleep modes in base stations (Chiaraviglio et al. 2012). Johnson and Rahmat-Samii (1995) have proposed a GA-based approach to find the best node distribution with optimized antenna pattern and SNR. The authors have employed simple binary coding scheme for a pool of randomly generated chromosomes for different nodes' distribution and evaluated their fitness

Table 9 Using genetic algorithms for wireless AP placement optimization

References	No. of objectives	Technique used	Brief summary
Wireless mesh networks			
Smadi et al. (2009)	Bi-Objective GA	GA with ILP	A Joint clustering and gateway problem is addressed by GAs for placement of hybrid free space opticals (FSO) to supplement the gateway capacity in WMNs
Xhafa et al. (2010)	Bi-objective GA	GA on grid area with rectangular mutation and crossover	Proposed GAs to optimize the connectivity and user coverage area of mesh network through optimal placement of nodes in the area
Mobile ad-hoc networks			
Kusyk et al. (2011)	MOGA	Distributed forced GA with evolutionary game theory	A GA alongwith traditional game theory, running at each node in autonomous network, helps nodes deciding their mobility and organization
Sahin et al. (2008)	SOGA	Distributed forced GA	A Distributed GA adjust speed, location, and direction of mobile nodes over a geographical terrain for military applications
Hoffmann et al. (2011)	SOGA	GA with TDMA table and time slot exchanges.	A GA approach to gateway allocation in aeronautical networks and TDMA resource scheduling to minimize the latency
Wireless sensor networks			
Youssef and Younis (2007)	SOGA	GAs on pre-calculation generic routing table	A GA places the WSN nodes for disjoint clustering and minimizes the number of hops between sensor and nodes to have efficient intra-cluster communication
Tripathi et al. (2011)	Bi-objective	GA and Genetic Programming (GP)	GP improves the deployment, while GA optimizes the node placement for maximum coverage in WSN
Pandey et al. (2007)	MOGA	GA with greedy algorithm and 'binary integer linear programming' (BILP)	A hybrid GAs along with greedy algorithm optimizes the node placements for traffic balancing, in two tier hierarchical heterogeneous WSNs
Xu and Yao (2006)	SOGA	GA with integer coding and order mapped crossover (OMX)	A GA approach worked on optimal placement of WSN nodes in grid area full of hindrances and obstacles to match in real-time environment

against the fitness function (in which N is the total number of nodes):

$$\text{Fitness} = \max - \frac{1}{N} \sum_i \text{Path\_length}_i \qquad (1)$$

The winner chromosomes (having minimum cost of path length) are more likely to be inherited in the next generation. This approach is also applied for random encoding of antenna pattern and transmission power.

Churn prediction and forecasting have application in almost all the real-world phenomenon including medical, finance, products sale, telecommunications and network. Forecasting the future sale and growth in turbulent markets like telecommunications is always a big issue for product managers. Even approximate insights can yield large profit return in the long run. Venkatesan and Kumar studied GAs for forecasting (Venkatesan and Kumar 2002) the magnitude of future sales, time period to peak sales, and the value of sales during peak time. Considering the significant advantages of accurate forecast of product lifecycle, GAs are used to estimate the diffusion model based on previous data points. GAs are fed with several critical factors like price, competitive

intensity, and network effects in obtaining better predictions during growth phase. Since GAs are inherently parallel, different scenarios can be taken care of while making predictions. Studies show that predictions by GAs are robust and accurate in predicting the future growth of a market. Pendharkar (2009) presented the hybrid GA with neural networks to forecast the churn in cellular network services. The hybrid algorithms optimize the objectives after learning and assigning weights to certain parameters. Results obtained from GA method are shown to be more accurate than statistical models but are computationally expensive.

3.2 Network type based classification

In this section, we will describe the applications of GAs in wireless networking categorized according to the type of wireless network. In particular, we will present GA applications in the networking configurations of wireless sensor networks (WSNs), wireless mesh networks (WMNs), cognitive radio networks (CRNs), cellular networks (CNs), and software-defined wireless networks (SWNs).

Table 10 Using genetic algorithms for general applications in wireless networks

References	Network type	No. of objectives	Technique used	Brief summary
Area coverage				
Xhafa et al. (2010)	WMNs	Bi-objective GA	Rectangular mutation & crossover	A GA approach is used to maximize the 'giant component' and coverage area in WMNs while using multiple distribution strategies of client nodes
Alba et al. (2007)	MANET	MOGA	Broadcasting updates from base station running GA	A hybrid broadcasting protocol using evolutionary algorithms. A MOGA approach is used to optimize metropolitan-scope MANET coverage and usage
Hu et al. (2010)	WSN	SOGA	GAs with schedule transition operations	Hybrid GA with scheduled transition and forward encoding scheme finds the maximum number of disjoint cover sets to enhance the disjoint covered sets in areas
Sahin et al. (2008)	MANET	MOGA	A Distributed forced GA	A forced GA is proposed letting the nodes decide their movement, direction and speed through molecular force equilibrium to get maximum area coverage
Lai et al. (2007)	WSN	SOGA	Integer coding	GA-based scheme with one-to-one mapping of genes and sensors to find maximum disjoint sensor covers to prolong the network life span
Quintao et al. (2005)	WSN	SOGA	GAs coupled with mathematical modeling and ILP	GA-based heuristic approach with integer linear programming (ILP) and mathematical formulation to solve dynamic network coverage problem
Network planning and design				
Bhondekar et al. (2009)	WSN	MOGA	GA model for grid deployment	A GA decides the transmission ranges and the active/ sleeping nodes in WSN topology while accounting for connectivity, energy, and other application specific parameters
Jourdan and de Weck (2004)	WSN	MOGA	GA for non-linear objectives	A MOGA approach design the Pareto optimal layouts for WSN nodes depending upon the sensing and communication range
Pries et al. (2009)	WMNs	SOGA	Tree coding	GA operators like subtree crossover and cell crossover, respectively, are employed in solving small and large-scale WMN gateway optimization problems
Ayyadurai et al. (2011)	CN	SOGA	Single cell optimization	An addaptive GA for dynamic system reconfiguration optimizes the resource allocation in cellular networks
Ghosh et al. (2005)	WMN	SOGA	Binary encoding of nodes	A dynamic mesh topology design, using GAs, while maintaining low network cost and path redundancy at base station for reliability
Sheen et al. (2010)	CN	MOGA	Joint optimization	A GA for maximizing joint multi-cell CN system efficiency, through optimizing relay stations (RS) placement, frequency reuse pattern, and maximizing throughput to enhance the end user experience
Chiaraviglio et al. (2012)	CN	MOGA	Hybrid GA with centralized sorting of base stations	GA-based switch off strategies for enabling sleep modes in base stations to design energy efficient cellular networks
Cluster optimization				
Zeng and Chen (2008)	WMN	MOGA	Greedy algorithm with GAs	A hybrid GA meets the QoS and load balancing constraints separately by constructing the graph and dividing the WMNs in disjoint clusters
Hussain et al. (2007)	WMN	SOGA	Cluster optimization through distributed broadcasting	GAs augmented the network life when compared with LEACH (Heinzelman et al. 2000) by employing energy efficient data dissemination and optimizing the cluster distances
Zhang et al. (2008)	WSN	SOGA	Simulated annealing (SA) (Stumpf et al. 1994)	A GA approach provided the energy efficient WSN topology by improving the data aggregation rate and minimizing the intra-cluster distances
Huruiala et al. (2010)	WSN	SOGA	Centralized GA	A GA running on BS coordinate with nodes to minimize latency and power consumption for routing under tight constraints

3.2.1 Wireless sensor networks (WSN)

Wireless sensor networks have emerged as an exciting technology that can be used to monitor the physical world. WSN is composed of self-organizing, miniaturized sensors nodes, running on batteries, and communicating over wireless links. The major challenge in WSNs is to provide area coverage with minimum WSN nodes while preserving the resources and energy consumption.

Jia et al. (2009) addressed this problem through MOGA for energy-efficient coverage control in WSNs. The algorithm provides a mechanism to cover the maximum deployed area with minimum number of WSN nodes. Since connectivity and coverage are both dependent on each other, this problem is addressed by MOGA approach by achieving the Pareto optimal front. This is a non-dominated problem, so it comprises of multiple solution individuals that are evaluated against two fitness function coverage area and sensor used rate. The MOGA approach applied performs well in maximizing the area covered with least computation. Another advantage is that it avoids partial solutions in whole search space and reset sensor nodes once in whole network simulation.

Ferentinos and Tsiligiridis (2007) studied an adaptive WSN design based on MOGA. Individuals with random placement of nodes are utilized as population and GAs decide which sensors would be active and which would act as clusterhead. Random designs are encoded and fed to GAs and the winner chromosomes are going to decide the network infrastructure including clusterheads and active or passive nodes along with signal strengths. From evolution of characteristics, it was shown that generally it is more favorable to have large number of sensor nodes with low energy rather than having lesser number of sensors with high energy consumption. This application of adaptive GA in WSN can lead to increase in network life span while maintaining network properties close to optimal value as well as it helps in adapting sophisticated decisions about operating modes (clusterheads, active nodes with optimal signal range).

Extending the life span of any WSN network is active area of research. Different schemes have been proposed like centralized monitoring GA for keeping active and passive nodes in network. A similar approach has been discussed by Jin et al. (2003) to cut the energy consumption in WSNs. A GA is used for independent cluster formation to reduce the communication distances thus bringing the route cost down. Experimental results showed that keeping the number of cluster to 10 % of total nodes results in 80 % reduction in path cost and communication distance as compared to random distribution of WSNs. In another work, Sengupta et al. (2012) have proposed an evolutionary multiobjective sleep-scheduling scheme for differentiated coverage in WSNs.

Zhang et al. (2008) used GA to extend the work of Niculescu and Nath on network localization in WSNs (Niculescu and Nath 2003). The central idea is to employ GPS-enabled nodes, called anchor nodes, in the network and to have all the other network nodes estimate their position through 1-hop distances to the anchor nodes. The locations are then flooded by each node to the clusterhead which uses the location information to construct the graph of the entire network. By using standard GA operators like mutation and crossover, the proposed GA-based approach was able to better estimate locations resulting in lesser mean position error than other algorithms in literature such as the 'semi-definite programming with gradient search localization' (SDPL) (Liang et al. 2004) and the 'simulated annealing based localization' (SAL) (Kannan et al. 2005).

In another work, Hussain et al. (2007) have studied and simulated GAs for centralized hierarchical WSNs. The proposed scheme comprises a hierarchy of layers with the BS being at the top layer and the clusterheads and the sensor nodes occupying the bottom layer. GAs are used to construct clusters with clusterheads and other nodes. The BS layer keeps track of the entire cluster through their clusterheads. A GA approach minimizes the average cluster distances while simultaneously increasing the total number of transmissions. After configuring sensor nodes in optimized formation, the data transmission phase follows.

3.2.2 Wireless mesh networks (WMNs)

Wireless mesh networks provide a popular mechanism for the provisioning of cost-effective wireless broadband connectivity. WMNs are composed of (potentially mobile) mesh clients and (typically fixed) mesh routers that connect over a relatively static multihop wireless network (Chou et al. 2006). To address the issue of efficient placement of mesh routers, Xhafa et al. (2010) employed GAs to optimize the user coverage area and connectivity of mesh network through optimal placement of nodes in an area. Chromosomes are designed in rectangular grids having information of both clients and mesh router nodes. Crossover is implemented to preserve the chromosome traits by randomly exchanging parent and child router nodes from one cell to another in grid area, thereby promoting high-scoring chromosomes. Mutation is then utilized to randomly migrate a router to another grid and recalculating the fitness. Various models can be adopted for distributing client nodes such as the normal, uniform, and exponential distribution. GAs are excellent in computing the mesh router placements and always succeed in achieving the connectivity, irrespective of client distribution; however, area coverage is subjected to client nodes in network area.

3.2.3 Cognitive radio networks (CRNs)

Cognitive radio networks (CRNs) offer a potential solution for improving the spectrum utilization in the world of limited spectrum. In dynamic spectrum access (DSA) based CRNs, the secondary users of a network aim to utilize the spectrum opportunistically to offer QoS for secondary users without causing any interference to licensed primary users. Such DSA-based CRNs utilize cognitive radios (CRs) that incorporate various AI techniques for intelligent decision making (Qadir 2015). The main benefits of applying GA in CRNs are as follows: (i) it is conceptually easy to understand; (ii) it is inherently amenable to parallel solutions and, therefore, can be easily distributed; and that (iii) GAs lend support to multi-objective optimization and perform well in noisy unknown environments.

An early application of GA techniques to CRNs is documented in a paper authored by Rondeau et al. (2004). This paper presented the adaptation mechanism of a cognitive engine, implemented by the authors, that used GAs to evolve radio parameters to make them best for the user's current needs. A 'wireless system genetic algorithm' (WSGA) is also proposed to realize adaptive waveform control and cross-layer optimization Rondeau et al. (2004). GAs have also been used for building distributed parallel solutions, using the technique of island GAs, to mitigate the channel assignment problem in CRNs (Friend et al. 2008). An island GA divides the overall population into subpopulations, known as islands, which then interact by migrating individuals to the other island. More details of parallel GAs, and of general parallel meta-heuristics, can be found in the survey Alba and Troya (1999) and Alba (2005), respectively.

(a) MOGAs for optimization of CRNs: Multi-objective optimization aims at estimating a true Pareto front optimization in wireless medium depending upon the external environment parameters. These radio environment parameters serve as genes and are encoded in binary strings, which form chromosomes. These chromosomes are evaluated against fitness function regularly in successive generations. After realizing sufficient fitness or completing predefined number of generation, these sets of transmission parameters are sent for updating new radio configuration. Due to variation in radio channel characteristics and spectrum availability, a cognitive radio has to maintain time-varying QoS while optimizing different parameters such as BER, throughput, power consumption, and interference.

Multi-objective genetic algorithms have been used extensively for optimization of cognitive radio networks (CRNs) (Fette 2009). The foremost aim of cognitive engine (CE) was to optimize the radio parameters for best QoS and the secondary goal to observe and learn for adaptation. To achieve these goals cognitive engines have (i) cognitive system module (CSM)—for modeling and learning (ii) wireless system genetic algorithm (WSGA)—for actions (iii) wireless channel genetic algorithm (WCGA)—for hardware implementation of radio parameters. These parameter and radio traits (called genes) represent chromosomes such as modulation, frequency, spreading, filtering, and FEC coding. Different objectives are optimized based on relative weights assigned to them using multi-objective approach. These objectives—including optimizing BER, power consumption, interference and MOGA approach—are applied on the population; the best-performing chromosomes are then chosen as the new set of radio parameters.

Case base decision theory (CBDT) for initialization of GAs: Since the secondary objective of CR is to adapt and evolve with changing environmental parameters by updating radio configurations, a CR system must be provided with a memory database that continuously monitors and learns GA outcomes in real-time optimization. As the system matures, with the passage of time, each new optimization would take less time to achieve global optima. To be more precise, CE keeps track of previous objectives and the winner chromosomes. So whenever a new objective or case is encountered, it is compared to previous cases in memory and chromosomes of the case/objective having maximum similarity and utility are used to partially initialize the GA population for current objective. Rest of the population is randomly generated to not only maintain diversity but also to help in achieving optima in lesser number of iterations. This adaptation scheme is carried out through algebraic manipulations with the independently constructed channel availability matrix, the interference matrix, and the channel reward matrix (which are each kept up to date about the external radio environment conditions) (Ye et al. 2010).

Rieser (2004) presented a biologically inspired cognitive radio model, using modern AI techniques. This model is based on adaptive cognitive engine that learns the radio environment, optimizes the radio parameters, and presents the best possible experience to secondary user. He built his foundation of GA-based technique on original Mitola (2006) proposed technique of software radios. He gave insights into all the popular AI Algorithms and how ineffective they are in evolving as compared to GAs, when faced with unanticipated radio channel environment. He applied GA to hidden Markov model for error modeling in uncertainties in radio channel environment. Wireless channel GA has its foundation on these radio environmental modeling.

The biologically inspired cognitive engine serves and evolves well in unanticipated radio channel environment, which makes it usable for military purpose and emergency situation. This whole cognitive architecture works in multiple steps. First, the engine gathers the channel information which is then subjected to machine learning techniques such

as learning classifiers. After learning and modeling, these information are taken up to alter the radio configurations for tweaking appropriate settings (Fette 2009). Wireless system GA treats the radio as the analogue of a biological creature and makes optimized discoveries to keep the appropriate balance of radio parameters.

(b) Cognitive radio (CR) testbeds: Rieser (2004) developed a biologically inspired cognitive radio CR testbed. A number of cognitive theories are developed and evaluated in parallel through the use of GA operators. This developed testbed consists of multilayered cognitive engine inspired by human brain evolution. The architecture is shown in figure. It provides multilayered scaled functionality with sensing radio parameters at both MAC and physical layers. It uses three algorithms that can be co-located in same radio or distributed to multiple radios. Another GA-based software testbed for cognitive engine (CE) is proposed by Kim et al. (2008). A two-step GA is used to avoid problem of optimizing the biased dominant parameters. To implement this, GA-based iteration is run for channel selection and transmitter parameter selection separately. Simulation of this testbed demonstrates that bandwidth and transmit power converge to optimal values with in few iterations of GAs.

3.2.4 Cellular networks (CNs)

Next-generation wireless networks have to cope with the massive increase in multimedia service along with cellular data. This motivates the development of self-consolidating networks that are capable of making intelligent decision to offer the best user quality of experience (QoE). Ayyadurai et al. (2011) presented an adaptive GA to optimize the cellular networks' spectral efficiency through dynamic allocation of resources, between the relay stations and the base stations. Population adjustment came along the way to improve the search capabilities leading to quick optimum. Chiaraviglio et al. (2012) presented the energy efficient deployment of cellular networks by leveraging GA to minimize the power consumption and number of transmitters in cells. This involves employing sleep modes, where a base station decides which nodes should be powered on, depending on number of active users in given cell.

A conflict-free channel assignment is an NP-hard problem. A powerful GA incorporating immune operators including immune clone, immune vaccination, and immune selection has been proposed by Zhenhua et al. (2010) to mitigate the interference constraints and obtain a conflict free channel assignment. Vaccination involves changing some bits of a gene to have an improved fitness value. Elitism keeps the population diversity and minimum separation encoding helps in keeping a frequency distance among individual channels. A similar approach has been studied by Pinagapany and

Kulkarni (2008) for solving both dynamic and fixed channel assignments in cellular networks. Channel compatibility matrix is constructed first to record the violation of individuals. After that, cell and bandwidth information is encoded in individual chromosomes and then subjected to crossover and mutation. José-Revuelta et al. (2007) offered low computational micro GA with elitism strategy to solve the fixed channel assignment (FCA) problem. The proposed algorithm spotlighted how to keep diversity by varying mutation probability to avoid local optima.

In cellular networks, it is desirable to minimize the call drop and call blocking probability. Mobile cellular networks are often evaluated against the criteria of dropping probability of current calls and the blocking probability of new calls. Lima et al. (2007) studied two adaptive GAs for locking and switching channel to solve the dynamic spectrum access problem in cellular networks. The proposed fitness function evaluates the capacity of each channel for new calls and also keeps track of existing calls. The algorithm takes care of randomness by allowing random immigrants and parameter adaptation to keep exploring new option for each channel.

3.2.5 Software-defined wireless networks (SWNs)

In recent times, software-defined networking (SDN) has been proposed as a new architecture for conveniently operating and managing diverse wireless networks. SDN is generally understood to entail (i) the separation of the control and data plane; (ii) the control of multiple data planes by a single controller; and the (iii) development of new sophisticated control abstractions (Qadir et al. 2014). Wireless networks using SDN principles are referred to as software-defined wireless networks (SWNs). Although this networking configuration is relatively new, the use of GA has been proposed in SWNs. For example, Bojic et al. have proposed a small-cell mobile backhaul resource manager, based on GA, that interworks with software defined management (Bojic et al. 2013).

3.3 Hybridization: GA with other techniques

Genetic algorithms is a useful framework that can be profitably used in a wide variety of settings. It is also possible to use GAs with other adaptive algorithmic approaches such as game theory, reinforcement learning (RL), graph theory, fuzzy logic-based, and quantum theory-based systems. Such a symbioses is not at all antithetical from the theme underlying the GA framework which encourages pluralism in preference to relying on any one solution; it is often the case that we can perform better by combining the best parts of different solutions as building blocks. In this section, we will describe hybrid approaches that combine various techniques with the principles of GAs. We will not only identify the GA hybrid technologies, but will also describe their application

specifically for wireless networks. We note here that some of the various applications of hybrid techniques involving GAs have already been presented in earlier sections (such as in Sect. 3.1) and tables in this paper. The interested reader can also refer to these places for specific pointers to hybrid techniques involving GA in wireless networking literature.

3.3.1 Combining GAs with game theory

Game theory is a mathematical decision framework through which we can study and analyze competitive interaction between two or more self-interested rational agents. The field of evolutionary game theory utilizes the concepts of evolutionary computing and GAs in the backdrop of a game-theoretic background (Weibull 1997).

Applications of GAs with game theory in wireless networks: Evolutionary games have been used in a wide variety of contexts in networks including cooperative sensing (Wang et al. 2010), multiple-access-control and power control (Tembine et al. 2010), routing games (Altman et al. 2007), network selection games (Niyato and Hossain 2009), congestion control (Altman et al. 2008), adaptive TCP (Anastasopoulos et al. 2010), and for dynamic bandwidth allocation (Zhu et al. 2010). The interested reader is referred to a chapter on this topic in a recently published book on game theory in wireless and communication networks (Han 2012) for a detailed treatment.

3.3.2 Combining GAs with reinforcement learning

Reinforcement learning (RL), unlike supervised learning, is a technique where agent has no prior knowledge of transitions and learns along the way through experience and interaction with the environment. We note that evolutionary algorithms like GAs are related to reinforcement learning as they rely on exploration and exploitation (Črepinšek et al. 2013).

More specifically, GA utilizes the genetic operators of mutation and recombination as procedures for exploration along with selection procedures to exploit known good solutions. RL can be carried out both by searching through the policy space or the value-function space. GAs can be thought of as a RL scheme of the policy space searching variety. Policies are encoded in chromosomes and deterministic procedure is used to evaluate the cumulative fitness of the agent that follows a given policy. According to evolutionary principle of survival of the fittest, only good policies are adopted while failed or weaker policies are relinquished (Moriarty et al. 1999).

Applications of GAs with RL in wireless networks: Generally speaking, RL algorithms are well suited for distributed problems like routing and localization and they can be used

with GA to solve NP-hard problems (that are difficult to solve optimally) heuristically. While there is limited deployment of RL-based GA solutions in wireless networks, in principle an RL-based GA can be implemented where RL is responsible to construct multi-individual crossover operator to facilitate search of global traits (Črepinšek et al. 2013).

3.3.3 Combining GAs with graph theory

In order to produce robust dynamic graph-theoretic solutions, graph theory can be combined with GAs in particular, and with evolutionary algorithms more generally. The combination of graph theory with evolutionary algorithms is known as evolutionary graph (EG) theory in literature.

Applications of GAs with graph theory in wireless networks: Ferreira et al. (2010) have applied EG theory in MANETs with known connectivity pattern for developing efficient practical routing protocols. The dynamic behavior of network is captured through evolving graphs. There are various metrics that can be used for determining optimal routes in EG-based models with known connectivity patterns, including the three metrics: shortest, fastest, and foremost as proposed in Xuan et al. (2003). These metrics determine, respectively, the minimum number of hops, the minimum time span, and the earliest arrival data. Ferreira et al. (2010) utilized the foremost journey metric along with the EG theory. The proposed algorithm can estimate the congested nodes and suggest the parallel paths. In another work, a bio-inspired graph theoretic solution for ensuring energy efficiency in cooperative communication networks has been proposed in Gajduk et al. (2014). The presented algorithm combines graph theory with evolutionary techniques to allow the nodes to intelligently operate in a decentralized fashion, and do forwarding by selecting nodes that conserve energy.

3.3.4 Combining GAs with fuzzy logic

Fuzzy logic is a logical system that can entertain logical variables that take continuous values between 0 and 1 instead of discrete values of either 0 or 1 (as in classical digital logic). Fuzzy logic is as an effective method for knowledge representation, control implementation, and cross-layer optimization in wireless networks (Baldo and Zorzi 2008). Fuzzy logic is well suited for knowledge representation in wireless networks since network conditions in wireless networks are often incompletely and imprecisely known. Fuzzy systems that utilize fuzzy logic are highly suited to problems with imprecise, noisy, and incomplete input information. GA can be used for both optimizing the fuzzy membership function as well as the fuzzy rules that drive the fuzzy control system. The intersection of hybrid technologies of GA, fuzzy logic, and other AI techniques (most prominently neural net-

works) is studied in the research theme of 'soft computing' Zadeh (1994). The trend of genetic fuzzy systems for soft computing applications is less developed than the trend of neuro-fuzzy systems, especially in the context of industrial control applications.

Applications of GAs with fuzzy logic in wireless networks: The use of genetic fuzzy techniques is popular for both communication systems (Wang 2003) and wireless networks. A localization scheme (Yun et al. 2009) based on RSS is designed using fuzzy logics and GAs. Fuzzy logic is used to model the edge weights and GAs play its part in optimizing the node placement. It is a distributed scheme since each node independently estimates its position. This scheme is particularly useful in large-scale networks. In a similar way, Saeedian et al. (2011) proposed a hybrid approach of fuzzy logics with GAs to select the clusterhead in WSN. This approach is a trade off between network-life and load. GA is used at base station to select nodes that have least energy metrics for their communication channel with clusterheads.

3.3.5 Combining GAs with quantum theory

Quantum theory derives its root from quantum mechanics, i.e., mechanics of atoms. A comparative study has been performed in Narayanan and Moore (1996) between quantum GA and classical GA. Quantum GA proves to be faster and more robust then classical GA for traveling sales person problem. Quantum GA operates on qubit as a smallest unit of information that serves as a linear superposition of two states, i.e., 1 and 0. A rotation gate is applied to keep the diversity of population in quantum GA. Quantum GA behaves more effectively than traditional GA for larger input set owing to its linear superposition states.

Applications of quantum GAs in wireless networks: A quantum GA is proposed in Luo (2010) to solve the NP-Complete QoS metrics problem in WSNs. The quantum GA performs more efficiently as compared to traditional GAs, especially for large-scale networks. To solve multicast QoS problem in WDM networks, a robust quantum GA is proposed (Xing et al. 2009) which incorporates multiple rotation angle steps (RAS) to foster the evolution process. Using multiple RAS schemes combined with repair methods helps faster convergence and elimination of unfits. In the cognitive radio domain, Zhao et al. (2009) used quantum GA along with particle swarms to solve the channel allocation problem.

3.3.6 Combining GAs with local search

A very common form of a hybrid GA, sometimes referred to as memetic algorithms (Moscato and Cotta 2010) in literature, is created by combining GAs with local search techniques and to incorporate domain-specific knowledge in the search process (Sastry et al. 2005). This can help produce stronger results but at the cost of higher computational costs. The local search operator is applied to each member of the population after each generation.

Applications of GAs with local search in wireless networks: Memetic algorithms have been extensively used in the context of wireless networks for various tasks such as energy-aware topology control (Konstantinidis et al. 2007) that solves the minimum energy network connectivity problem in WSNs better than spanning tree methods. The fitness function is designed to promote the least energy-consuming solution. Ting and Liao (2010) used memetic algorithms to increase the lifetime of WSN networks. The algorithm uses a compact operator that places the redundant sensors to get maximum coverage and minimum energy consumption. In cellular networks, assigning cells to switches is NP-hard problem. A multi-population memetic algorithm proposed by Quintero and Pierre (2002) outperforms the traditional tabu-search method with a considerable deduction in maintenance cost of WSN network for a 10-year period.

3.3.7 Combining GAs with other metaheuristics

A popular area of research is combining GAs with heuristic methods like ant colony optimization (ACO), artificial neural networks (ANN), and fuzzy logic to solve the dynamic constrained multi-objective optimization.

Applications of GAs and metaheuristics in wireless networks: GAs have been combined with various metaheuristic techniques in wireless networks. For example, there is ongoing research in wireless networks to use GAs to (1) discover the optimum mathematical models for grouping the cellular networks (Zhu and O'Connor 2013; Chiaraviglio et al. 2012); (2) train other algorithms like Recurrent Neural Networks (Syed 1995); and to (3) set optimal fuzzy parameters (Saeedian et al. 2011; Di Fatta 2003; Yun et al. 2008). In another work, Cauvery and Visvanatha (2009) have proposed a solution that combines GAs with ACO to reduce the size of routing table and help in routing among autonomous peers. ANNs, combined with GAs, have been used to forecast the churn in cellular network services (Pendharkar 2009) and for providing localization services (Chagas et al. 2012). There also has been a lot of work in applying GAs for adjusting the connection weights and the connectivity itself in ANNs (Whitley et al. 1990). In particular, a common application is to evolve ANN structures through the GA's fitness function to select optimum parameters for the underlying network (Kulkarni et al. 2011). There is also research interest in combining GAs with other metaheuristic techniques such as swarm intelligence to provide wireless solutions along with

limited computation overhead. Notwithstanding the amount of work done, there is a lot of scope for further optimization and improvement.

4 Practical issues and lessons learnt

In this section, we will summarize the main insights and lessons learnt through decades of GA research that can guide successful problem solving through GAs. We first describe insights about when and where to apply GAs in Sect. 4.1. We follow this up by a discussion on common pitfalls in Sect. 4.2 and finally describe guidelines and insights for successful GA implementations in Sect. 4.3.

4.1 When/where to apply GAs?

4.1.1 Comparison with other techniques

The metaheuristic technique of GAs can be compared to optimization techniques that guarantee globally optimal solutions. Optimization problems can be divided into continuous optimization problems and discrete optimization problems (that are also known as combinatorial optimization algorithms). Convex optimization problems are generally held to be tractable whereas nonlinear optimization problems can become intractable, especially when we consider large problems.

Apart from GA, there are a number of metaheuristic techniques, and many of them have been applied in the context of wireless networks. For instance, there are both nature-inspired algorithms—such as GAs, ant algorithms, simulated annealing—and non nature-inspired algorithms such as tabu search and iterated local search. Many of these metaheuristic techniques rely on iterative search. Metaheuristics can also be compared based on the number of solutions used at the same time. GAs use a population of solutions, whereas other techniques (known as trajectory methods) such as simulated annealing use a single solution at a time. A comparison of metaheuristic methods can be found readily in existing literature (Osman and Kelly 1996; Blum and Roli 2003; Glover and Kochenberger 2003). In this section, we will consider the restricted subset of metaheuristics comprising hill-climbing and simulated annealing.

Hill-climbing is trivial to program and does not require any memory (since no backtracking is involved), but such algorithms can get caught in local optima easily. Hill-climbing is a good option for finding local optima, which makes it optimal for convex problems (since for convex problems, the local optima is the global optima). For example, the simplex algorithm for the convex problem of linear programming uses hill climbing for exploring the solution space. Since hill-climbing can only guarantee finding of local optimas, hill-climbing is

often complemented with a scheme that uses memory-based search optimization using the tabu search, or memoryless stochastic modifications such as simulated annealing.

Simulated annealing like GA is a nature-inspired stochastic search technique. While GA is modeled on the natural evolution, SA takes inspiration from thermodynamics and statistical mechanics (Blum and Roli 2003). The fundamental idea of SA is to tolerate moves that result in temporarily worse-quality solutions compared to the current solution to escape the local minima. The probability of adopting such myopically suboptimal moves is reduced during the search. Simulated annealing and GAs both are well suited to high-dimensional spaces. Simulated annealing is used mostly when the search space is discrete. Simulated annealing has been applied in wireless networks for various problems such as localization (Kannan et al. 2005), power control (Montemanni et al. 2005), etc. Simulated annealing has also been used together with GA, and ant colony optimization, for high-performance multicast routing in QoS constraints that has a high convergence speed (Damanafshan et al. 2014).

Apart from GAs, ANNs are also widely deployed in wireless networks. Compared to GAs (which are essential search tools for metaheuristic optimization), ANNs are best suited for the task of classification and prediction of pattern (and find wide application in intrusion detection and spectrum sensing systems). ANNs, unfortunately, can be quite brittle in the presence of unknown environments that characterize wireless networks and can suffer from the problem of requiring extensive training.

In the particular context of wireless networks, while optimization has been extensively applied for resource allocation in wireless networks (Lin et al. 2006), optimization algorithms typically make the assumption of complete network information. In addition, optimization problems assume a static unchanging environment that is not realistic for wireless networks. A key benefit of GAs is that they are well suited to the optimization task in the ever-changing environments that characterize wireless networks (in which there is interaction between the various adaptive wireless nodes and interdependency between the various node's decisions). GAs are well suited for building metaheuristic adaptive algorithms that can provide satisfactory performance in changing network conditions. Compared to local optimization methods (e.g., gradient descent), GAs have the advantage that they can escape suboptimal local maximum/minimum of the target function. Since GAs rely on a population of solutions along with various genetic operators, GAs do not necessarily remain trapped in a suboptimal local region. As discussed above, various other metaheuristic techniques have also been proposed for use—some of them in concert with GAs—in wireless networks. Their deployment for wireless solutions should be planned in light of the pros and cons of these solutions (which has been briefly discussed above).

4.1.2 Recommendations for using GAs in wireless networks

Genetic algorithms are not ideally suited for a distributed implementation in wireless networks (especially in energy-conscious wireless networking configurations such as WSNs) since running GA on each node would cause energy overhead and reduction in network lifetime. In problems that require distributed network solutions (like localization and routing), reinforcement learning can be a better choice.

Genetic algorithms perform best in a centralized setting (e.g., at a cellular network's base station) for optimization of outputs and making optimal decisions while algorithms like Q-learning (Watkins and Dayan 1992) (a type of reinforcement learning) could be working in distributed manner on individual nodes. Use of GAs is better suited to MANETs than WSNs because of limited energy resources on individual WSN nodes. Using GAs in combination with other intelligent systems like ANN and fuzzy systems can also give better optimization results. Abu Alsheikh et al. (2014) have presented a comprehensive review article of all machine-learning (ML) techniques (including a discussion on GAs). While this paper focused on the use of ML techniques in WSNs, the interested reader can consult (Abu Alsheikh et al. 2014) for a comparative analysis of GAs with other comparable ML and metaheuristic techniques.

4.1.3 Tasks for which GAs are most suitable

The major strength of GAs, offsetting their inability to provide optimal solutions, is their great generality. GAs are the algorithmic workhorses that can be applied to a wide variety of problems. GAs can be applied wherever the corresponding search space is defined, and the desired solution could be quantified through an appropriate formulation of a fitness function. It has been said that "neural networks are the second best way of doing just about anything, and genetic algorithms are the third" (as quoted in Russell and Norvig (1995)). This highlights the fact that in most scenarios, optimal niche solutions will return better performance than GAs. Thus GAs should not adopted for problems that can be optimally solved by techniques such as linear programming, A-star search or constraint propagation.

In the context of wireless networks, GAs hold a lot of promise for efficiently solving complex (typically NP-hard) problems that are not amenable to efficient optimal solutions. It is worth pointing out that many problems in wireless networks—including those that focus on channel assignment (Mir et al. 2012) and routing [especially when we consider multi-rate (Qadir et al. 2006) and multi-interface wireless networks (Hassan et al. 2013)]—are NP-hard and thus obtaining the optimal solution is considered computationally complex. GAs are well suited to such complex problems for which an optimal solution is not known or is too complex; in

such circumstances, GAs may be the "best" practical solution. GAs—unlike traditional optimization algorithms—are also well suited to the dynamic environments that characterize wireless networks.

4.2 Common pitfalls

While GAs provide a useful framework for problem-solving in a wide variety of domains, they also suffer from various deficiencies that have been identified in literature. GA-based solutions for wireless networking also inherit these problems. In this section, we will highlight these general deficiencies that GA-based solutions will suffer from. In addition, we also identify and highlight common pitfalls regarding implementing GAs in wireless networks.

4.2.1 Genetic drift or bias

In GAs, there is a risk of slow convergence and settling for local minimas, especially with inappropriate choice of model parameters. While GAs can often converge to the global optima in the majority of applications, there are no mathematical guarantees to this effect. In particular, a GA can get stuck in the vicinity of a suboptimal point if it lacks population diversity, or if has an inappropriately small population. Such a phenomenon is known as 'genetic drift' or 'genetic bias'. Certain problems cannot be solved by means of simple genetic operations. This is mainly due to the choice of poor fitness functions that generate poor chromosome blocks. There cannot be an absolute assurance that a GA will be able to find a global optimum—this can especially be a problem when the population has a lot of subjects.

4.2.2 Using GAs for real-time applications

Like most other AI techniques, GAs are not well suited to real-time application that require guaranteed response times due to its stochastic and heuristic nature. In addition, the response time of GA can have significant variance depending on the various control parameters and population initialization. This potential pitfall should be kept in mind if GAs are used for online learning and optimization in wireless networks. In particular, Qo requirements in wireless networks may put constraints on achieving the optimum decision in limited time forcing GAs to terminate after predefined number of iterations.

4.2.3 GA deception

It has been reported in literature that certain objective functions, referred to as GA-deceptive functions, can be very difficult to optimize. With such deceptive functions, combination of good building blocks may result in very bad

chromosomes, causing the building block theory to fail. A potential solution to this problem is using messy GAs (Mitchell 1998).

4.3 Implementation insights

There are numerous challenges in devising an appropriate GA-based solution including suitable definitions of population size and evaluation function (which is typically non-trivial to define). In this section, we highlight some general implementation insights that have been gleaned from the body of knowledge representing GA research. Wireless practitioners can use the insights presented in this section for designing efficient GA-based algorithms.

4.3.1 Setting population size

The size of the population in a GA can be a major factor in determining its quality and convergence speed. Generally speaking, larger populations result in better solutions. A few studies have addressed the problem of determining exactly how large a population should be to ensure a certain quality (Goldberg 1989; Goldberg et al. 1991; Harik et al. 1999). For a moderately complex problem, a population of 50–100 chromosomes can serve as a good default value (Cox 2005). The precise choice of the population size depends on the problem complexity and the number of search variables. While it is typical for the population size to remain fixed from one generation to the next, it can also be made to grow or contract depending on the rate of convergence, and other factors such as the current genetic diversity.

4.3.2 Designing solution encoding and fitness function

The efficiency, processing power, and the convergence of GAs depend critically on how the solution is encoded in the form of chromosomes and on the formulation of an appropriate fitness function. This is the first necessary step towards an efficient implementation. There is no one-size-fits-all solution here, and the correct model choice depends on what is most useful for the problem at hand. The encoding of chromosomes should be carefully considered. While a majority of implementations utilize bit representation, due to the ease with which crossover and mutation can be implemented, other representations, utilizing data structures, may be appropriate for the considered problem and should also be considered (Sastry et al. 2005). Since there can be numerous encoding and formulations, this step involves considerable ingenuity. Some guidelines on these issues are provided in Grefenstette (1986), De Jong and Spears (1991).

4.3.3 Setting of mutation and crossover rates

The optimization of the control variables of GAs (such as the setting of mutation and crossover rates) plays an important role in the performance of a GA-based framework. A GA framework can avoid prematurely settling in a suboptimal rut by emphasizing exploration using mutation. Various solutions have been proposed to the problem of genetic drift/bias including fitness sharing (Fonseca and Fleming 1995), crowding, and pre-selection (Mahfoud 1992). Bhandari et al. (1996) showed that the probability of fast convergence increases by seeding the initial population with high-scoring chromosomes. As a general rule-of-thumb, crossover and mutation rates of 0.6 and 0.001 have been proposed for large population sizes (of ~100), with corresponding values of 0.9 and 0.01 proposed for small population sizes (of ~30) (Grefenstette 1986; De Jong and Spears 1991). Sastry et al. (2005) similarly recommend a mutation probability of 0.05 and a crossover rate of 0.6 with a population size of ~50. A tabulated summary of mutation and crossover rates generally recommended by the GA community and accepted as de facto 'standards' are presented in Table 11.

4.3.4 Computation of fitness values

In many problems, the computation of the fitness function can be extremely computationally intensive, thus motivating

Table 11 Summary of GA paramater settings guidelines generally accepted by the GA community

	De Jong and Spears (1991)	Grefenstette (1986)	Sastry et al. (2005)	Carroll (1996) (micro GA)
Population size	50	30	50	5
Number of generations	1000	Not specified	Not specified	100
Mutation rates	0.001	0.01	0.05	0.02; 0.04
Mutation types	Bit flip	Bit flip	Not specified	Jump and creep
Crossover rate	0.6	0.9	0.6	0.5
Crossover type	1 or 2 point (typically, 2 point)	1 or 2 point (typically, 2 point)	Not specified	Uniform

research in fitness approximation (Jin 2005). An important practical issue is to determine ways of modeling large complex systems so that the fitness value can be computed without executing the real-world system itself (which can be impractical in many situations).

4.3.5 Setting of termination criteria

Assuming the population, and the various GA control variables, have been set up properly, the GA will improve towards the optimal solution over successive generations. In light of potentially slow convergence, it is very important to for timely problem solving with GAs to devise a suitable termination criterion that defines a 'good enough' solution. Defining these criteria is non-trivial and context-dependent. While this criteria depends on the number of objective functions, and the number of variables, a default value of 2.5 times the population size can be used as a reasonable maximum generation count estimate (Cox 2005).

5 Open issues and future work

In this section, we provide the recommendation on scope of future research on GAs in the field of networking. In Sect. 5.1, we will highlight the common issues that GAs generally face. Since any progress towards resolving these open research problems will benefit all GA-based solutions, it is worth highlighting these problems. In Sect. 5.2, we will highlight open research issues and future work specific to the application of GAs for wireless networking.

5.1 General issues with GAs (not specific to wireless)

5.1.1 Theoretical results for GAs

A lot of effort has focused on providing theoretical understanding of why and how GAs work but rigorous results have been slow in coming. Some of the results, such as "No Free Lunch", have sobering implications. Simply stated, no free lunch implies that all search algorithms perform equally when averaged over all problems. This further entails that if we are comparing GAs with other search heuristics (such as hill climbing or simulation annealing), even if GA performs better for some class of problems, the other algorithms will necessarily perform better on some problems outside this class. GAs are also population-based heuristics; even as the entire population is improved by GAs, the same cannot be guaranteed for an individual existing within that population. There is a need for enhanced theoretical understanding of various facets of GAs to lay a strong foundation upon which the GA user's trust can be safely placed.

5.1.2 Efficient multi-objective and distributed GAs

The development of practical algorithms for wireless networks often requires the use of multi-objective optimization performed in a distributed fashion. Currently, the most popular approaches in wireless networks depend on centralized GA (Lai et al. 2007): e.g., in mobile cellular networks, the GA runs on the BS and helps decide the cluster size and nodes deployment. In wireless networks, there is often a need to distribute the computational load to various network nodes. This can be done by a locally run GA on each node combined with local search techniques for robustness.

5.1.3 Using big data techniques for scaling GAs

Big data techniques have the ability to scalably and reliably process massive amounts of data using a cluster of commodity servers (Qadir et al. 2014). Big data processing techniques, such as Google's MapReduce and other frameworks of its ilk, hold a lot of promise for processing large problems in quick time. This is done by horizontal scaling, or 'scaling out', through which more and more computers are employed to attack a given problem in parallel. There is potentially a lot of scope in employing these big data processing techniques to accelerate the convergence of GAs. A steady trickle of research has started emerging that is proposing the use of big data tools (such as the open-source Apache Hadoop platform which is built on Google's MapReduce paradigm) to scale up GAs (Verma et al. 2009; Llora et al. 2010). More research needs to be done to determine how to best utilize big data tools and techniques for improving GA performance.

5.2 Using GAs in wireless (wireless-specific issues)

5.2.1 Exploiting problem-specific knowledge

Genetic algorithms are often resorted to as black-box solutions to complex problems where the mechanisms to be used for problem solving are not well understood. Even without problem-specific knowledge, GAs can allow insights to develop thorough identification of similarities between highly fit individuals (Goldberg and Holland 1988). When the subject knowledge is available, in the form of good understanding of the underlying problem and plausible candidate solutions, it is desirable to exploit this information to expedite GA's progression towards a solution. The question of how to embed wireless-specific subject knowledge into the GA frameworks is very much an open research issue requiring further exploration. One way of embedding problem-specific information in the GA system is by seeing GAs with good structures and populations thereby accelerating search and learning progress. However, the use of

problem-specific knowledge to generate specialized heuristics or operators can entail some loss of generality of the GA system (Goldberg and Holland 1988).

5.2.2 Niche applications of GAs for wireless networks

GAs can be useful for optimizing new applications for wireless networks. As an example, there has been a lot of ongoing work in hierarchical clustering for scalable networking (Shurman et al. 2013; Pandey et al. 2007; Huruiala et al. 2010) in which GA has provided promising solutions; there is also an ongoing research on GA-based dynamic cluster design based on multiple metrics (Quintao et al. 2005). There is also room for improving the throughput by extending the GAs from single cell to multi-cell optimization and by working on inter-channel noise cancellation (Ayyadurai et al. 2011). Finally, there is a need to discover GA-based low-energy-consuming distributed schemes that can provide maximum disjoint covers to enhance the coverage area and network life.

6 Conclusions

We have provided a survey of the applications of genetic algorithms (GAs) in wireless networking. In addition to providing a self-contained introduction to common models and configurations of GAs, we have also provided a detailed survey of applications of GAs in wireless networking. We have categorized the applications of GAs in wireless networking both according to the wireless networking configuration and according to the different kinds of GA techniques. We have also highlighted pitfalls and challenges in successfully implementing GAs in wireless networks. Finally, we have highlighted a number of open issues and have identified potential directions for future work.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

Abdullah J (2010) Multiobjectives ga-based QoS routing protocol for mobile ad hoc network. Int J Grid Distrib Comput 3(4):57–68

Abu Alsheikh M, Lin S, Niyato D, Tan H-P (2014) Machine learning in wireless sensor networks: algorithms, strategies, and applications. In: Communications Surveys & Tutorials, IEEE, vol 16, no 4, pp 1996–2018

Ahn CW, Ramakrishna RS (2002) A genetic algorithm for shortest path routing problem and the sizing of populations. Evol Comput IEEE Trans 6(6):566–579

Akaiwa Y, Andoh H (1993) Channel segregation-a self-organized dynamic channel allocation method: application to tdma/fdma microcellular system. Sel Areas Commun IEEE J 11(6):949–954

Alba E (2005) Parallel metaheuristics: a new class of algorithms, vol 47. Wiley

Alba E, Dorronsoro B, Luna F, Nebro AJ, Bouvry P, Hogie L (2007) A cellular multi-objective genetic algorithm for optimal broadcasting strategy in metropolitan manets. Comput Commun 30(4):685–697

Alba E, Troya JM (1999) A survey of parallel distributed genetic algorithms. Complexity 4(4):31–52

Al-Ghazal M, El-Sayed A, Kelash H (2007) Routing optimlzation using genetic algorithm in ad hoc networks. In: Signal processing and information technology, 2007 IEEE International Symposium on. IEEE, pp 497–503

Ali S, Munir A, Qaisar SB, Qadir J (2012) A genetic algorithm assisted resource management scheme for reliable multimedia delivery over cognitive networks. In: Computational science and its applications-ICCSA 2012. Springer, pp 352–367

Al-Karaki JN, Ul-Mustafa R, Kamal AE (2009) Data aggregation and routing in wireless sensor networks: optimal and heuristic algorithms. Comput Netw 53(7):945–960

Al-Qahtani TA, Abedin MJ, Ahson SI (1998) Dynamic routing in homogenous atm networks using genetic algorithms. In: Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence, The 1998 IEEE International Conference on. IEEE, pp 114–119

Altman E, Elazouzi R, Hayel Y, Tembine H (2008) An evolutionary game approach for the design of congestion control protocols in wireless networks. In: Modeling and optimization in mobile, ad hoc, and wireless networks and workshops, 2008. WiOPT 2008. 6th International Symposium on. IEEE, pp 547–552

Altman E, Hayel Y, Kameda H (2007) Evolutionary dynamics and potential games in non-cooperative routing. In: Modeling and Optimization in mobile, ad hoc and wireless networks and workshops, 2007. WiOpt 2007. 5th International Symposium on. IEEE, pp 1–5

Anastasopoulos MP, Petraki DK, Kannan R, Vasilakos AV (2010) TCP throughput adaptation in WiMax networks using replicator dynamics. Syst Man Cybern Part B Cybern IEEE Trans 40(3):647–655

Apetroaei I, Oprea I-A, Proca B-E, Gheorghe L (2011) Genetic algorithms applied in routing protocols for wireless sensor networks. In: Roedunet international conference (RoEduNet), 2011 10th. IEEE, pp 1–6

Ayyadurai V, Moessner K, Tafazolli R (2011) Multihop cellular network optimization using genetic algorithms. In: Proceedings of the 7th international conference on network and services management. International Federation for Information Processing, pp 399–403

Badia L, Botta A, Lenzini L (2009) A genetic approach to joint routing and link scheduling for wireless mesh networks. Ad Hoc Netw 7(4):654–664

Baldo N, Zorzi M (2008) Fuzzy logic for cross-layer optimization in cognitive radio networks. IEEE Commun Mag 46(4):64

Banerjee N, Das SK (2001) Fast determination of qos-based multicast routes in wireless networks using genetic algorithm. In: Communications, 2001. ICC 2001. IEEE International Conference on, vol 8. IEEE, pp 2588–2592

Barolli L, Koyama A, Shiratori N (2003) A QoS routing method for ad-hoc networks based on genetic algorithm. In: Database and expert systems applications, 2003. Proceedings 14th International Workshop on. IEEE, pp 175–179

Bhandari D, Murthy C, Pal SK (1996) Genetic algorithm with elitist model and its convergence. Int J Pattern Recognit Artif Intell 10(06):731–747

Bhattacharjee S, Konar A, Nagar AK (2011) Channel allocation for a single cell cognitive radio network using genetic algorithm. In: Innovative mobile and internet services in ubiquitous computing (IMIS), 2011 fifth international conference on. IEEE, pp 258–264

Bhondekar AP, Vig R, Singla ML, Ghanshyam C, Kapur P (2009) Genetic algorithm based node placement methodology for wireless sensor networks. In: Proceedings of the international multiconference of engineers and computer scientists, vol 1. Citeseer, pp 18–20

Blum C, Roli A (2003) Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput Surv (CSUR) 35(3):268–308

Bojic D, Sasaki E, Cvijetic N, Wang T, Kuno J, Lessmann J, Schmid S, Ishii H, Nakamura S (2013) Advanced wireless and optical technologies for small-cell mobile backhaul with dynamic software-defined management. IEEE Commun Mag 51(9):86–93

Canfora G, Di Penta M, Esposito R, Villani ML (2005) An approach for qos-aware service composition based on genetic algorithms. In: Proceedings of the 7th annual conference on Genetic and evolutionary computation. ACM, pp 1069–1075

Carroll DL (1996) Chemical laser modeling with genetic algorithms. AIAA J 34(2):338–346

Chagas SH, Martins JB, De Oliveira LL (2012) An approach to localization scheme of wireless sensor networks based on artificial neural networks and genetic algorithms. In: New circuits and systems conference (NEWCAS), 2012 IEEE 10th International. IEEE, pp 137–140

Cheng H, Yang S, Cao J (2013) Dynamic genetic algorithms for the dynamic load balanced clustering problem in mobile ad hoc networks. Expert Syst Appl 40(4):1381–1392

Cheng M, Chang LF (1999) Wireless dynamic channel assignment performance under packet data traffic. Sel Areas Commun IEEE J 17(7):1257–1269

Cheng H, Yang S (2010) Multi-population genetic algorithms with immigrants scheme for dynamic shortest path routing problems in mobile ad hoc networks. In: Applications of evolutionary computation. Springer, pp 562–571

Cheng HT, Zhuang W (2009) Novel packet-level resource allocation with effective QoS provisioning for wireless mesh networks. Wirel Commun IEEE Trans 8(2):694–700

Chiang T-C, Liu C-H, Huang Y-M (2007) A near-optimal multicast scheme for mobile ad hoc networks using a hybrid genetic algorithm. Expert Syst Appl 33(3):734–742

Chiaraviglio L, Ciullo D, Koutitas G, Meo M, Tassiulas L (2012) Energy-efficient planning and management of cellular networks. In: Wireless on-demand network systems and services (WONS), 2012 9th Annual Conference on. IEEE, pp 159–166

Choi J, Kwon T, Choi Y, Naghshineh M (2000) Call admission control for multimedia services in mobile cellular networks: a markov decision approach. In: Computers and communications, 2000. Proceedings. ISCC 2000. Fifth IEEE Symposium on. IEEE, pp 594–599

Chou CT, Misra A, Qadir J (2006) Low-latency broadcast in multirate wireless mesh networks. Sel Areas Commun IEEE J 24(11):2081–2091

Cox E (2005) Fuzzy modeling and genetic algorithms for data mining and exploration. Academic Press

Črepinšek M, Liu S-H, Mernik M (2013) Exploration and exploitation in evolutionary algorithms: a survey. ACM Comput Surv (CSUR) 45(3):35

Cui X, Lin C, Wei Y (2003) A multiobjective model for QoS multicast routing based on genetic algorithm. In: Computer networks and mobile computing, 2003. ICCNMC 2003. 2003 International Conference on. IEEE, pp 49–53

Damanafshan M, Khosrowshahi-Asl E, Abbaspour M (2014) Gasant: An ant-inspired least-cost qos multicast routing approach based on genetic and simulated annealing algorithms. Int J Comput Commun Control 7(3):417–431

Das SK, Banerjee N, Roy A (2006) Solving optimization problems in wireless networks using genetic algorithms. In: Handbook of bioinspired algorithms and applications, p 219

Davies C, Lingras P (2003) Genetic algorithms for rerouting shortest paths in dynamic and stochastic networks. Eur J Oper Res 144(1):27–38

Davis L et al (1991) Handbook of genetic algorithms, vol 115. Van Nostrand Reinhold, New York

De Jong KA, Spears WM (1991) An analysis of the interacting roles of population size and crossover in genetic algorithms. In: Parallel problem solving from nature. Springer, pp 38–47

Deb K, Agrawal S, Pratap A, Meyarivan T (2000) A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. Lect Notes Comput Sci 1917:849–858

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: Nsga-ii. Evol Comput IEEE Trans 6(2):182–197

Di Fatta G, Hoffmann F, Lo Re G, Urso A (2003) A genetic algorithm for the design of a fuzzy controller for active queue management. Syst Man Cybern Part C Appl Rev IEEE Trans 33(3):313–324

EkbataniFard GH, Monsefi R, Akbarzadeh-T M-R, Yaghmaee M et al (2010) A multi-objective genetic algorithm based approach for energy efficient qos-routing in two-tiered wireless sensor networks. In: Wireless pervasive computing (ISWPC), 2010 5th IEEE International Symposium on. IEEE, pp 80–85

ElNainay MY, Ge F, Wang Y, Hilal AE, Shi Y, MacKenzie AB, Bostian CW (2009) Channel allocation for dynamic spectrum access cognitive networks using localized island genetic algorithm. In: Testbeds and research infrastructures for the development of networks & communities and workshops, 2009. TridentCom 2009. 5th International Conference on. IEEE, pp 1–3

Fang T, Chau L-P (2006) Gop-based channel rate allocation using genetic algorithm for scalable video streaming over error-prone networks. Image Process IEEE Trans 15(6):1323–1330

Ferentinos KP, Tsiligiridis TA (2007) Adaptive design optimization of wireless sensor networks using genetic algorithms. Comput Netw 51(4):1031–1051

Ferreira A, Goldman A, Monteiro J (2010) Performance evaluation of routing protocols for MANETs with known connectivity patterns using evolving graphs. Wirel Netw 16(3):627–640

Fette BA (2009) Cognitive radio technology. Access Online via Elsevier

Fonseca CM, Fleming PJ (1995) Multiobjective genetic algorithms made easy: selection sharing and mating restriction. In: Proceedings of the 1st International conference on genetic algorithms in engineering systems: innovations and applications (GALESIA)

Friend DH, ElNainay M, Shi Y, MacKenzie AB (2008) Architecture and performance of an island genetic algorithm-based cognitive network. In: Consumer communications and networking conference, 2008. CCNC 2008. 5th IEEE. IEEE, pp 993–997

Gajduk A, Utkovski Z, Basnarkov L, Kocarev L (2014) Energy-efficiency in decentralized wireless networks: a game-theoretic approach inspired by evolutionary biology, CoRR, vol abs/1405.3491

Gao C, Cai M, Chen H (2007) Qos-aware service composition based on tree-coded genetic algorithm. In: Computer software and applications conference, 2007. COMPSAC 2007. 31st Annual International, vol 1. IEEE, pp 361–367

Gelenbe E, Liu P, Laine J (2006) Genetic algorithms for autonomic route discovery. In: Distributed intelligent systems: collective intelligence and its applications, 2006. DIS 2006. IEEE Workshop on. IEEE, pp 371–376

Gen M, Cheng R, Wang Q (1997) Genetic algorithms for solving shortest path problems. In: Evolutionary computation, 1997, IEEE International Conference on. IEEE, pp 401–406

Ghosh S, Ghosh P, Basu K, Das SK (2005) GaMa: an evolutionary algorithmic approach for the design of mesh-based radio access networks. In: Local computer networks, 2005. 30th Anniversary. The IEEE Conference on. IEEE, p 8

Glover F, Kochenberger GA (2003) Handbook of metaheuristics. Springer

Goldberg DE (1989) Sizing populations for serial and parallel genetic algorithms. In: Proceedings of the 3rd international conference on genetic algorithms. Morgan Kaufmann Publishers Inc., pp 70–79

Goldberg DE, Deb K, Clark JH (1991) Genetic algorithms, noise, and the sizing of populations. Complex Syst 6:333–362

Goldberg DE, Holland JH (1988) Genetic algorithms and machine learning. Mach Learn 3(2):95–99

Goldberg DE, Zakrzewski K, Chang C, Gallego P (1997) Genetic algorithms: a bibliography. Urbana 51:61801

Gözüpek D, Alagöz F (2011) Genetic algorithm-based scheduling in cognitive radio networks under interference temperature constraints. Int J Commun Syst 24(2):239–257

Grefenstette JJ (1986) Optimization of control parameters for genetic algorithms. Syst Man Cybern IEEE Trans 16(1):122–128

Han Z (2012) Chapter 6, Game theory in wireless and communication networks: theory, models, and applications. Cambridge University Press

Harford T (2011) Adapt: why success always starts with failure. Macmillan

Harik G, Cantú-Paz E, Goldberg DE, Miller BL (1999) The gambler's ruin problem, genetic algorithms, and the sizing of populations. Evol Comput 7(3):231–253

Hassan MT, Ahmed E, Qadir J, Baig A (2013) Quantifying the multiple cognitive radio interfaces advantage. In: Advanced information networking and applications workshops (WAINA), 2013 27th International Conference on. IEEE, pp 511–516

He J, Ji S, Yan M, Pan Y, Li Y (2012) Load-balanced CDS construction in wireless sensor networks via genetic algorithm. Int J Sens Netw 11(3):166–178

Heinzelman WR, Chandrakasan A, Balakrishnan H (2000) Energy-efficient communication protocol for wireless microsensor networks. In: System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on. IEEE, p 10

Highfield R, Nowak M (2011) SuperCooperators: evolution, altruism and human behaviour (or why we need each other to succeed). Text Publishing

Hillier F, Lieberman G (2001) Introduction to operations research. McGraw Hill, New York

Hoffmann F, Medina D, Wolisz A (2011) Optimization of routing and gateway allocation in aeronautical ad hoc networks using genetic algorithms. In: Wireless communications and mobile computing conference (IWCMC), 2011 7th International. IEEE, pp 1391–1396

Holland J (1975) Genetic algorithms, computer programs that evolve in ways that even their creators do not fully understand. Sci Am, pp 66–72

Holland JH (1995) Hidden order: how adaptation builds complexity. Basic Books

Hsu C-Y, Wu J-LC, Wang S-T, Hong C-Y (2008) Survivable and delay-guaranteed backbone wireless mesh network design. J Parallel Distrib Comput 68(3):306–320

Hu X-M, Zhang J, Yu Y, Chung H-H, Li Y-L, Shi Y-H, Luo X-N (2010) Hybrid genetic algorithm using a forward encoding scheme for lifetime maximization of wireless sensor networks. Evol Comput IEEE Trans 14(5):766–781

Huang C, Liu Y (2010) Moeaq: a qos-aware multicast routing algorithm for manet. Expert Syst Appl 37(2):1391–1399

Huruiala P-C, Urzica A, Gheorghe L (2010) Hierarchical routing protocol based on evolutionary algorithms for wireless sensor networks. In: Roedunet international conference (RoEduNet), 2010 9th. IEEE, pp 387–392

Hussain S, Matin AW, Islam O (2007) Genetic algorithm for hierarchical wireless sensor networks. J Netw 2(5):87–97

Ishibuchi H, Nojima Y et al (2006) Comparison between single-objective and multi-objective genetic algorithms: Performance comparison and performance measures. In: Evolutionary Computation, 2006. CEC 2006. IEEE Congress on. IEEE, pp 1143–1150

Jia J, Chen J, Chang G, Tan Z (2009) Energy efficient coverage control in wireless sensor networks based on multi-objective genetic algorithm. Comput Math Appl 57(11):1756–1766

Jiang H, Yang X, Yin K, Zhang S, Cristoforo JA (2011) Multi-path qos-aware web service composition using variable length chromosome genetic algorithm. Inf Technol J 10(1):113–119

Jin Y (2005) A comprehensive survey of fitness approximation in evolutionary computation. Soft Comput 9(1):3–12

Jin S, Zhou M, Wu AS (2003) Sensor network optimization using a genetic algorithm. In: Proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics, pp 109–116

Johnson JM, Rahmat-Samii Y (1995) Genetic algorithm optimization of wireless communication networks. In: Antennas and propagation society international symposium, 1995. AP-S. Digest, vol 4. IEEE, pp 1964–1967

José-Revuelta S et al (2007) A new adaptive genetic algorithm for fixed channel assignment. Inf Sci 177(13):2655–2678

Jourdan D, de Weck OL (2004) Layout optimization for a wireless sensor network using a multi-objective genetic algorithm. In: Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th, vol 5. IEEE, pp 2466–2470

Kandavanam G, Botvich D, Balasubramaniam S, Jennings B (2010) A hybrid genetic algorithm/variable neighborhood search approach to maximizing residual bandwidth of links for route planning. In: Artifical evolution. Springer, pp 49–60

Kannan AA, Mao G, Vucetic B (2005) Simulated annealing based localization in wireless sensor network. In: Local computer networks, 2005. 30th Anniversary. The IEEE Conference on. IEEE, p 2

Karabudak D, Hung C-C, Bing B (2004) A call admission control scheme using genetic algorithms. In: Proceedings of the 2004 ACM symposium on applied computing. ACM, pp 1151–1158

Kassotakis IE, Markaki ME, Vasilakos AV (2000) A hybrid genetic approach for channel reuse in multiple access telecommunication networks. Sel Areas Commun IEEE J 18(2):234–243

Kim JM, Sohn SH, Han N, Zheng G, Kim YM, Lee JK (2008) Cognitive radio software testbed using dual optimization in genetic algorithm. In: Cognitive radio oriented wireless networks and communications, 2008. CrownCom 2008. 3rd International Conference on. IEEE, pp 1–6

Kim J-S, Park S, Dowd P, Nasrabadi N (1996) Channel assignment in cellular radio using genetic algorithms. Wirel Pers Commun 3(3):273–286

Kobayashi H, Munetomo M, Akama K, Sato Y (2004) Designing a distributed algorithm for bandwidth allocation with a genetic algorithm. Syst Comput Jpn 35(3):37–45

Konak A, Coit DW, Smith AE (2006) Multi-objective optimization using genetic algorithms: a tutorial. Reliab Eng Syst Saf 91(9):992–1007

Konstantinidis A, Yang K, Chen H-H, Zhang Q (2007) Energy-aware topology control for wireless sensor networks using memetic algorithms. Comput Commun 30(14):2753–2764

Krishnakumar K (1990) Micro-genetic algorithms for stationary and non-stationary function optimization. In: 1989 advances in intelligent robotics systems conference. International Society for Optics and Photonics, pp 289–296

Kulkarni RV, Förster A, Venayagamoorthy GK (2011) Computational intelligence in wireless sensor networks: a survey. Commun Surv Tutor IEEE 13(1):68–96

Kusyk J, Sahin CS, Umit M, Uyar E Urrea, Gundry S (2011) Self-organization of nodes in mobile ad hoc networks using evolutionary games and genetic algorithms. J Adv Res 2(3):253–264

Lai C-C, Ting C-K, Ko R-S (2007) An effective genetic algorithm to improve wireless sensor network lifetime for large-scale sur-

veillance applications. In: Evolutionary computation, 2007. CEC 2007. IEEE Congress on. IEEE, pp 3531–3538

Liang T-C, Wang T-C, Ye Y (2004) A gradient search method to round the semidefinite programming relaxation solution for ad hoc wireless sensor network localization. Sanford University, formal report, vol 5

Lieska K, Laitinen E, Lahteenmaki J (1998) Radio coverage optimization with genetic algorithms. In: Personal, indoor and mobile radio communications, 1998. The Ninth IEEE International Symposium on, vol 1. IEEE, pp 318–322

Lima MA, Araujo AF, Cesar AC (2007) Adaptive genetic algorithms for dynamic channel assignment in mobile cellular communication systems. Vehicular Technol IEEE Trans 56(5):2685–2696

Lin X, Shroff NB, Srikant R (2006) A tutorial on cross-layer optimization in wireless networks. Sel Areas Commun IEEE J 24(8):1452–1463

Lin D, Labeau F (2012) Accelerated genetic algorithm for bandwidth allocation in view of emi for wireless healthcare. In: Wireless communications and networking conference (WCNC), 2012 IEEE. IEEE, pp 3312–3317

Li D, Zhang Q, Chuah C-N, Ben Yoo S (2006) Multi-source multi-path video streaming over wireless mesh networks. In: Circuits and Systems, 2006. ISCAS 2006. Proceedings 2006 IEEE International Symposium on. IEEE, p 4

Llora X, Verma A, Campbell RH, Goldberg DE (2010) When huge is routine: scaling genetic algorithms and estimation of distribution algorithms via data-intensive computing. In: Parallel and distributed computational intelligence. Springer, pp 11–41

Lopez RB, Sanchez SM, Fernandez EM, Souza RD, Alves H (2014) Genetic algorithm aided transmit power control in cognitive radio networks. In: Cognitive radio oriented wireless networks and communications (CROWNCOM), 2014 9th International Conference on, IEEE, pp 61–66

Lorenzo B, Glisic S (2013) Optimal routing and traffic scheduling for multihop cellular networks using genetic algorithm. Mob Comput IEEE Trans 12(11):2274–2288

Luo W (2010) A quantum genetic algorithm based QoS routing protocol for wireless sensor networks. In: Software Engineering and service sciences (ICSESS), 2010 IEEE International Conference on. IEEE, pp 37–40

Lu T, Zhu J (2013) Genetic algorithm for energy-efficient QoS multicast routing. Commun Lett IEEE 17(1):31–34

Mahfoud SW (1992) Crowding and preselection revisited. Urbana 51:61801

Maksuriwong K, Varavithya V, Chaiyaratana N (2003) Wireless LAN access point placement using a multi-objective genetic algorithm. In: Systems, Man and Cybernetics, 2003. IEEE International Conference on, vol 2. IEEE, pp 1944–1949

Malik A, Qadir J, Ahmad B, Yau K-LA, Ullah U (2014) Qos in ieee 802.11-based wireless networks: a contemporary survey. arXiv:1411.2852

Man K-F, Tang K-S, Kwong S (1996) Genetic algorithms: concepts and applications. IEEE Trans Ind Electron 43(5):519–534

Mir AK, Akram A, Ahmed E, Qadir J, Baig A (2012) Unified channel assignment for unicast and broadcast traffic in cognitive radio networks. In: Local computer networks workshops (LCN Workshops), 2012 IEEE 37th Conference on. IEEE, pp 799–806

Mitchell M (1998) An introduction to genetic algorithms. MIT press

Mitola J III (2006) Cognitive Radio architecture: the engineering foundations of Radio XML. Wiley

Montemanni R, Gambardella LM, Das AK (2005) The minimum power broadcast problem in wireless networks: a simulated annealing approach. In: Wireless communications and networking conference, 2005 IEEE, vol 4. IEEE, pp 2057–2062

Moriarty DE, Schultz AC, Grefenstette JJ (1999) Evolutionary algorithms for reinforcement learning. J Artif Intell Res 11:241–276

Moscato P, Cotta C (2010) A modern introduction to memetic algorithms. In: Handbook of Metaheuristics. Springer, pp 141–183

Nagy L, Farkas L (2000) Indoor base station location optimization using genetic algorithms. In: Personal, Indoor and Mobile Radio Communications, 2000. PIMRC 2000. The 11th IEEE International Symposium on, vol 2. IEEE, pp 843–846

Nan G-F, Li M-Q, Li J (2007) Estimation of node localization with a real-coded genetic algorithm in wsns. In: Machine learning and cybernetics, 2007 International Conference on, vol 2. IEEE, pp 873–878

Narayanan A, Moore M (1996) Quantum-inspired genetic algorithms. In: Evolutionary computation, 1996, Proceedings of IEEE International Conference on. IEEE, pp. 61–66

Ngo CY, Li VO (1998) Fixed channel assignment in cellular radio networks using a modified genetic algorithm. Vehicular Technol IEEE Trans 47(1):163–172

Niculescu D, Nath B (2003) Ad hoc positioning system (aps) using aoa. In: INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies, vol 3. IEEE, pp 1734–1743

Niyato D, Hossain E (2009) Dynamics of network selection in heterogeneous wireless networks: an evolutionary game approach. Vehicular Technol IEEE Trans 58(4):2008–2017

NK FC, Viswanatha SDK (2009) Routing algorithm using mobile agents and genetic algorithm. Int J Comput Electr Eng, vol 1, no 3

Osman IH, Kelly JP (1996) Meta-heuristics: theory and applications. Springer Science & Business Media

Ozugur T, Bellary A, Sarkar F (2001) Multiobjective hierarchical 2G/3G mobility management optimization: niched Pareto genetic algorithm. In: Global telecommunications conference, 2001. GLOBECOM'01. IEEE, vol 6. IEEE, pp 3681–3685

Pandey S, Dong S, Agrawal P, Sivalingam K (2007) A hybrid approach to optimize node placements in hierarchical heterogeneous networks. In: Wireless communications and networking conference, 2007. WCNC 2007. IEEE. IEEE, pp 3918–3923

Pedrycz W, Vasilakos A (2010) Computational intelligence in telecommunications networks. CRC Press

Pendharkar PC (2009) Genetic algorithm based neural network approaches for predicting churn in cellular wireless network services. Expert Syst Appl 36(3):6714–6720

Pinagapany S, Kulkarni A (2008) Solving channel allocation problem in cellular radio networks using genetic algorithm. In: Communication Systems software and middleware and workshops, 2008. COMSWARE 2008. 3rd International Conference on. IEEE, pp 239–244

Pries R, Staehle D, Stoykova M, Staehle B, Tran-Gia P (2009) Wireless mesh network planning and optimization through genetic algorithms. In: Advances in mesh networks, 2009. MESH 2009. Second International Conference on. IEEE, pp 55–61

Qadir J (2015) Artificial intelligence based cognitive routing for cognitive radio networks. Springer Artificial Intelligence Review

Qadir J, Ahad N, Mushtaq E, Bilal M (2014) SDN, clouds, and big data: New opportunities. In: 12th International conference on frontiers of information (FIT)

Qadir J, Ahmed N (2014) Ahad N (2014) Building programmable wireless networks: an architectural survey. EURASIP J Wirel Commun Netw 1:172

Qadir J, Chou CT, Misra A (2006) Exploiting rate diversity for multicasting in multi-radio wireless mesh networks. In: Local computer networks, Proceedings 2006 31st IEEE Conference on. IEEE, pp 287–294

Quintao FP, Nakamura FG, Mateus GR (2005) Evolutionary algorithm for the dynamic coverage problem applied to wireless sensor networks design. In: Evolutionary computation, 2005. The 2005 IEEE Congress on, vol 2. IEEE, pp 1589–1596

Quintero A, Pierre S (2002) A memetic algorithm for assigning cells to switches in cellular mobile networks. Commun Lett IEEE 6(11):484–486

Ridley M (2004) Evolution. 3rd edn. Blackwell

Riedl A (2002) A hybrid genetic algorithm for routing optimization in ip networks utilizing bandwidth and delay metrics. In: IP operations and management, 2002 IEEE Workshop on. IEEE, pp 166–170

Rieser CJ (2004) Biologically inspired cognitive radio engine model utilizing distributed genetic algorithms for secure and robust wireless communications and networking. PhD thesis, Virginia Polytechnic Institute and State University

Rondeau TW, Le B, Rieser CJ, Bostian CW (2004) Cognitive radios with genetic algorithms: intelligent control of software defined radios. In: Software defined radio forum technical conference. Citeseer, pp C3–C8

Roy A, Banerjee N, Das SK (2002) An efficient multi-objective qos-routing algorithm for wireless multicasting. In: Vehicular technology conference, 2002. VTC Spring 2002. IEEE 55th, vol 3. IEEE, pp 1160–1164

Roy A, Das SK (2004) Qm2rp: a qos-based mobile multicast routing protocol using multi-objective genetic algorithm. Wirel Netw 10(3):271–286

Russell S, Norvig P (1995) Artificial intelligence: a modern approach, vol 74. Prentice hall Englewood Cliffs

Saeedian E, Torshiz MN, Jalali M, Tadayon G, Tajari MM (2011) Cfga: Clustering wireless sensor network using fuzzy logic and genetic algorithm. In: Wireless communications, networking and mobile computing (WiCOM), 2011 7th International Conference on. IEEE, pp 1–4

Sahin CS, Urrea E, Uyar MU, Conner M, Hokelek I, Bertoli G, Pizzo C (2008) Uniform distribution of mobile agents using genetic algorithms for military applications in manets. In: Military communications conference, 2008. MILCOM 2008. IEEE. IEEE, pp 1–7

Sahin CS, Urrea E, Uyar MU, Conner M, Hokelek I, Conner M, Bertoli G, Pizzo C (2008) Genetic algorithms for self-spreading nodes in manets. In: Proceedings of the 10th annual conference on genetic and evolutionary computation, GECCO '08, (New York). ACM, pp 1141–1142

Salcedo-Sanz S, Bousoño-Calzón C, Figueiras-Vidal AR (2003) A mixed neural-genetic algorithm for the broadcast scheduling problem. Wirel Commun IEEE Trans 2(2):277–283

Sastry K, Goldberg D, Kendall G (2005) Genetic algorithms. In: Search methodologies. Springer, pp 97–125

Scaperoth D, Le B, Rondeau T, Maldonado D, Bostian CW, Harrison S (2006) Cognitive radio platform development for interoperability. In: Military communications conference, 2006. MILCOM 2006. IEEE. IEEE, pp 1–6

Scully T, Brown KN (2009) Wireless LAN load balancing with genetic algorithms. Knowl Based Syst 22(7):529–534

Selamat A, Selamat MH et al (2004) Routing algorithm of mobile agents for query retrieval using genetic algorithm. Malays J Comput Sci 17(2):1–10

Sengupta S, Das S, Nasir M, Vasilakos AV, Pedrycz W (2012) An evolutionary multiobjective sleep-scheduling scheme for differentiated coverage in wireless sensor networks. Syst Man Cybern Part C Appl Rev IEEE Trans 42(6):1093–1102

Sheen W-H, Lin S-J, Huang C-C (2010) Downlink optimization and performance of relay-assisted cellular networks in multicell environments. Vehicular Technol IEEE Trans 59(5):2529–2542

Sherif MR, Habib IW, Nagshineh M, Kermani P (2000) Adaptive allocation of resources and call admission control for wireless atm using genetic algorithms. Sel Areas Commun IEEE J 18(2):268–282

Shurman MM, Al-Mistarihi MF, Mohammad AN, Darabkh K, Ababnah A et al (2013) Hierarchical clustering using genetic algorithm in wireless sensor networks. In: Information & Communication Technology Electronics & Microelectronics (MIPRO), 2013 36th International Convention on. IEEE, pp 479–483

Sivanandam S, Deepa S (2008) Introduction to genetic algorithms

Smadi MN, Ghosh SC, Farid AA, Todd TD, Hranilovic S (2009) Free-space optical gateway placement in hybrid wireless mesh networks. J Lightwave Technol 27(14):2688–2697

Srinivas M, Patnaik LM (1994) Genetic algorithms: a survey. Computer 27(6):17–26

Stumpf JD, Feng X, Kelnhofer RW (1994) An enhanced operator-oriented genetic search algorithm. In: Evolutionary computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on. IEEE, pp 235–238

Sun B, Pi S, Gui C, Zeng Y, Yan B, Wang W, Qin Q (2008) Multiple constraints QoS multicast routing optimization algorithm in manet based on ga. Progr Nat Sci 18(3):331–336

Syed O (1995) Applying genetic algorithms to recurrent neural networks for learning network parameters and architecture. PhD thesis, Case Western Reserve University

Tam V, Cheng K-Y, Lui K-S (2006) Using micro-genetic algorithms to improve localization in wireless sensor networks. J Commun 1(4):1–10

Tembine H, Altman E, El-Azouzi R, Hayel Y (2010) Evolutionary games in wireless networks. Syst Man Cybern Part B Cybern IEEE Trans 40(3):634–646

Ting C-K, Liao C-C (2010) A memetic algorithm for extending wireless sensor network lifetime. Inf Sci 180(24):4818–4833

Tripathi A, Gupta P, Trivedi A, Kala R (2011) Wireless sensor node placement using hybrid genetic programming and genetic algorithms. Int J Intell Inf Technol (IJIIT) 7(2):63–83

Turing AM (1950) Computing machinery and intelligence. Mind, pp 433–460

Vedantham S, Iyengar SS (1998) The bandwidth allocation problem in the atm network model is np-complete. Inf Process Lett 65(4):179–182

Venkatesan R, Kumar V (2002) A genetic algorithms approach to growth phase forecasting of wireless subscribers. Int J Forecast 18(4):625–646

Verma A, Llora X, Goldberg DE, Campbell RH (2009) Scaling genetic algorithms using mapreduce. In: Intelligent systems design and applications, 2009. ISDA'09. Ninth International Conference on. IEEE, pp 13–18

Wang L (2003) Soft computing in communications, vol 136. Springer

Wang B, Liu KR, Clancy TC (2010) Evolutionary cooperative spectrum sensing game: how to collaborate? Commun IEEE Trans 58(3):890–900

Watkins CJ, Dayan P (1992) Q-learning. Mach Learn 8(3–4):279–292

Weibull JW (1997) Evolutionary game theory. MIT press

Whitley D, Starkweather T, Bogart C (1990) Genetic algorithms and neural networks: optimizing connections and connectivity. Parallel Comput 14(3):347–361

Wong SH, Wassell I (2002) Dynamic channel allocation using a genetic algorithm for a tdd broadband fixed wireless access network, in IASTED International Conference in Wireless and Optical Communications. Banff, Canada

Xhafa F, Sánchez C, Barolli L (2010) Genetic algorithms for efficient placement of router nodes in wireless mesh networks. In: Advanced information networking and applications (AINA), 2010 24th IEEE International Conference on. IEEE, pp 465–472

Xiao Y, Chen CP, Wang Y (2000) A near optimal call admission control with genetic algorithm for multimedia services in wireless/mobile networks. In: National aerospace and electronics conference, 2000. NAECON 2000. Proceedings of the IEEE 2000. IEEE, pp 787–792

Xing H, Liu X, Jin X, Bai L, Ji Y (2009) A multi-granularity evolution based quantum genetic algorithm for qos multicast routing problem in wdm networks. Comput Commun 32(2):386–393

Xuan BB, Ferreira A, Jarry A (2003) Computing shortest, fastest, and foremost journeys in dynamic networks. Int J Found Comput Sci 14(02):267–285

Xu Y, Yao X (2006) A GA approach to the optimal placement of sensors in wireless sensor networks with obstacles and preferences. In: Consumer communications and networking conference, 2006. CCNC 2006. 3rd IEEE, vol 1. IEEE, pp 127–131

Yang S, Cheng H, Wang F (2010) Genetic algorithms with immigrants and memory schemes for dynamic shortest path routing problems in mobile ad hoc networks. Syst Man Cybern Part C Appl Rev IEEE Trans 40(1):52–63

Yen Y-S, Chan Y-K, Chao H-C, Park JH (2008) A genetic algorithm for energy-efficient based multicast routing on manets. Comput Commun 31(4):858–869

Yen Y-S, Chao H-C, Chang R-S, Vasilakos A (2011) Flooding-limited and multi-constrained QoS multicast routing based on the genetic algorithm for manets. Math Comput Model 53(11):2238–2250

Ye F, Yang R, Li Y (2010) Genetic algorithm based spectrum assignment model in cognitive radio networks. In: Information Engineering and Computer Science (ICIECS), 2010 2nd International Conference on. IEEE, pp 1–4

Youssef W, Younis M (2007) Intelligent gateways placement for reduced data latency in wireless sensor networks. In: Communications, 2007. ICC'07. IEEE International Conference on. IEEE, pp 3805–3810

Yun S, Lee J, Chung W, Kim E, Kim S (2009) A soft computing approach to localization in wireless sensor networks. Expert Syst Appl 36(4):7552–7561

Yun S, Lee J, Chung W, Kim E (2008) Centroid localization method in wireless sensor networks using tsk fuzzy modeling. In: International symposium on advanced intelligent systems, pp 971–974

Zadeh LA (1994) Fuzzy logic, neural networks, and soft computing. Commun ACM 37(3):77–84

Zeng F, Chen Z (2008) Load balancing placement of gateways in wireless mesh networks with QoS constraints. In: Young computer scientists, 2008. ICYCS 2008. The 9th International Conference for. IEEE, pp 445–450

Zhang J, Lin Y, Zhou C, Ouyang J (2008) Optimal model for energy-efficient clustering in wireless sensor networks using global simulated annealing genetic algorithm. In: Intelligent information technology application workshops, 2008. IITAW'08. International Symposium on. IEEE, pp 656–660

Zhang Q, Wang J, Jin C, Ye J, Ma C, Zhang W (2008) Genetic algorithm based wireless sensor network localization, In: Natural Computation, 2008. ICNC'08. Fourth International Conference on, vol 1. IEEE, pp 608–613

Zhang Q, Wang J, Jin C, Zeng Q (2008) Localization algorithm for wireless sensor network based on genetic simulated annealing algorithm. In: Wireless communications, networking and mobile computing, 2008. WiCOM'08. 4th International Conference on. IEEE, pp 1–5

Zhao Z, Peng Z, Zheng S, Shang J (2009) Cognitive radio spectrum allocation using evolutionary algorithms. Wirel Commun IEEE Trans 8(9):4421–4425

Zhenhua Y, Guangwen Y, Shanwei L, Qishan Z (2010) A modified immune genetic algorithm for channel assignment problems in cellular radio networks. In: Intelligent system design and engineering application (ISDEA), 2010 International Conference on, vol 2. IEEE, pp 823–826

Zhu K, Niyato D, Wang P (2010) Optimal bandwidth allocation with dynamic service selection in heterogeneous wireless networks. In: Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE. IEEE, pp 1–5

Zhu N, O'Connor I (2013) iMASKO: a genetic algorithm based optimization framework for wireless sensor networks. J Sens Actuator Netw 2(4):675–699