A multi-objective optimization model based on immune algorithm in wireless mesh networks

Jing Chen¹, Kun He¹, Ruiying Du¹, Fajiang Yu¹,*†, Quan Yuan², Lina Wang¹ and Cai Fu³

¹School of Computer, Wuhan University, Wuhan 430072, Hubei, China
²Department of Math and Computer Science, University of Texas-Permian Basin, TX, USA
³Computer Department, Huazhong University of Science and Technology, Wuhan 430074, China

SUMMARY

With the characteristics of high self-organized, dynamic, and interoperable, the wireless mesh network (WMN) is deemed as a potential technology to be applied widely for home, enterprise, and social public service. Many current optimization schemes usually focus on a single metric such as network deployment cost, throughput, QoS, and so on, but few schemes consider that the optimized metric may affect other metrics of WMN. In practice, the influence among the different metrics is often nonignorable. To optimize the performance from a global perspective, we propose a multi-objective optimization model based on immune algorithm (MOM-IA), which provides a paradigm to find the optimal solution satisfying some different restriction conditions. To simplify, MOM-IA mainly analyzes the restriction relationship of connectivity, redundancy, and throughput, which are the multiple objects. Considering the characteristic of dynamic and the discrete integer parameters in WMN, we design a longtime evolution immune algorithm to solve the MOM. Finally, the analysis of experiments presents that MOM-IA has good performance in solution set diversity and Pareto-front distribution, which means the probability to find the optimal solution in WMN. Copyright © 2014 John Wiley & Sons, Ltd.

KEY WORDS: WMNs; multi-objective optimization model; immune algorithm

1. INTRODUCTION

The wireless mesh network (WMN) is a potential distributed network with the characteristics of self-organized and rapid deployment [1]. Combining the advantages of the wireless local area network [2] and ad hoc network [3], WMN consists of independent and distributed wireless nodes where the access points do not have the energy limitation. Furthermore, it can interoperate with mobile cellular network [4], wireless sensor network, and so on. Hence, it was widely applied in many fields, such as home, enterprise, social public service, and military [5, 6].

With more and more service applied in WMN, the performance requirement of network becomes higher and higher [7]. Besides the throughput and transmission speed, the reliability and the maintenance cost of network are also very important for service. Obviously, it is hard to reach the best performance in all aspects at the same time. There are contradictions in some aspects. It means that the promotion in some aspects may induce the downgrade of other aspects. For example, the enhancement of connectivity often induces extra redundancy. As a result, the metric of redundancy is affected. Hence, it is significant to research the relationship of different factors, such as efficiency and reliability, and find the optimal method to improve the global performance of WMN.

*Correspondence to: Fajiang Yu, School of Computer, Wuhan University, Wuhan 430072, Hubei, China.
†E-mail: qshxyu@126.com

Copyright © 2014 John Wiley & Sons, Ltd.
Currently, researchers mainly focus on two options in network optimization. One option is the mathematic model construction and analysis, where the major task is to analyze the constraint conditions, solution set features, and research the relationship of network factors by the suitable mathematic model. The other option is the design and optimizing of solving process. Many algorithms are proposed such as classical linear weighting method [8], evolutionary algorithm [9, 10], particle swarm algorithm [11, 12], coevolutionary algorithms [13], and artificial immune algorithm [14, 15].

The WMN usually provides the last mile access service to various nodes. Different nodes have different functions, which affect the efficiency and reliability of network diversely. Considering the feature of WMN, we propose a multi-objective optimization model based on immune algorithm (MOM-IA). Our algorithm defines the network architecture and node model at first. Then, the nodes adopt a different analysis model according to their level; in addition, the nodes belong to the same level has different weights. Based on that, we analyze the restrict conditions, create the MOM, and design the efficient immunization algorithm to solve the problem.

This paper is structured as follows: Section 2 introduces the related work of efficient and reliable model in WMN. The network model and assumptions are presented in Section 3. Section 4 proposes the MOM and designs the efficient immunization algorithm to solve the problem. The simulation shows the restrict relationship of each factor and the feature of Pareto front distribution in Section 5. Section 6 concludes the paper.

2. RELATED WORK

T. Vanhatupa et al. propose a performance analysis model based on IEEE 802.11s in WMN [16]. The model adopts several metrics for evaluating such as network volume, throughput, equitableness of access point, coverage range, and the average rate of service flow. Utilizing the network topology as input, it optimizes the network performance by some different algorithms. The advantage of the model is that it describes the evaluating metrics definitely, and the disadvantage of the model is that it does not analyze the relationship of these metrics.

Z. Jun et al. formulate multiple QoS parameters as a multi-objective model and propose an effective discrete particle swarm optimization algorithm to approximate the Pareto front by generating a set of nondominated solutions [17]. However, the algorithm lacks scalability, and the particle swarm optimization algorithm has a limitation on long evolution.

J. Rezgui et al. propose a MOM for channel assignment (CA), which is performed during the multi-radio WMNs planning process [18]. Given the expected traffic, its goal includes minimizing user handoff overhead, minimizing traffic load variances to achieve load balancing, and maximizing overall throughput. The CA is an important aspect in WMN, but we still need a scalable MOM from the global network perspective.

A. Barolli et al. propose the integer linear programming model for the deployment of the WMN [19]. This model chooses the number of routers and distribution of access point as the design destination. It considers the influence in the aspects of the flow rate, interference, self-adaption problem, the CA, and so on, and it also solves the puzzle by integer linear programming to achieve the minimum deployment cost. As we know, it is possible that the multiple objects have contradictions, which lead to the failure of linear programming.

L. Fang et al. propose a novel immune algorithm on the basis of V-detector to solve the multi-objective optimization problem (V-detector) in wireless network [20]. The advantage of the algorithm is that it designs the crowding factor in update operation to optimize the Pareto-front. However, the algorithm adopts the nonuniform mutation, which is only suitable for continuous variation decision space. It has good global searching ability in the earlier stage and local searching ability in the later stage. Because the decision space is discrete in WMN, the evolution may stop in the later stage [21].

Copyright © 2014 John Wiley & Sons, Ltd.

DOI: 10.1002/dac
3. PRELIMINARY

3.1. The model and assumption of network

(1) In this paper, the architecture of WMN is two layers as in Figure 1. The upper layer is composed of mesh routers, which can connect directly by routing protocol. The under layer consists of mesh clients, which can only connect to each other by mesh routing relaying.

(2) Each client node can access multiple routers and switches the router according to the network situation.

(3) If no router covers a client node, we can increase the node power level or utilize store-carry-forward method to recover the connection. In our model, the weight of this node is small.

3.2. The definition of symbols

For ease of understanding, we define the main symbols in Table I. Note that $x_j$ is the sum of column in array $(A_{ij})_{m \times n}$. Each member in array $(A_{ij})_{m \times n}$ is an integer; hence, $x_j$ is also an integer. Similarly, most values and the weights of multiple objects referring connectivity, redundancy, and throughput are integers. Furthermore, the range of weights is between 1 and 5, where level 1 is the lowest significance level, and level 5 is the highest significance level. Particularly, $y_p$ is a boolean value, and the average throughput $P$ is real. In addition, the limitations of multiple objects, such as connectivity, redundancy, throughput, and the sizes of immune algorithm parameters, such as initial population and memory unit, are all positive integers. And the ratio of antibody $R_c (0 < R_c < 1)$ is a real.

4. MULTI-OBJECTIVE OPTIMIZATION MODEL BASED ON IMMUNE ALGORITHM

In this paper, our solution has two components. The first one is the mathematical model of multi-objective relationship and restriction conditions. The second one is the immune algorithm to achieve the optimum solution.

![Figure 1. Two-level architecture.](image)
4.1. The multi-objective optimization model

The aim of our paper is to propose a paradigm to optimize multiple objects even if these objects are incompatible. For better explanation, we choose connectivity, redundancy, and throughput as examples. From the restriction conditions and mathematic model, it is easy to find out that our model has good scalability.

(1) The connectivity of network

Defining the relationship array \((A_{ij})_{m \times n}\) while \(m (1 \leq i \leq m)\) denotes the number of routers in upper layer and \(n (1 \leq j \leq n)\) denotes the number of client nodes in under layer. If router \(R_i\) and client node \(C_j\) are connected, then \(A_{ij} = 1\), or \(A_{ij} = 0\).

Defining the connected routers number of client node \(j\) as \(x_j\) and the corresponding weight is \(w_j\); hence, \(x_j = \sum_{i=1}^{m} A_{ij}\) and the connectivity of client node in under layer is

\[
\sum_{j=1}^{n} w_j x_j
\]  

(1)

In the WMN, the routers of upper layer are connected. We assume that the connectivity weight of routers is \(\beta\) and the connectivity value of routers is \(B_m\), which is a function related to \(m\).

The connectivity of network is

\[
\sum_{j=1}^{n} w_j x_j + \beta B_m
\]  

(2)

(2) The redundancy of network

\(y_p (p = 1, 2, \ldots, n)\) indicates whether the client node \(C_p\) connects any router. If there is no router connected to \(C_p\), then \(y_p = 1\), or \(y_p = 0\).

\(u_p\) denotes the weight of extra links. The cost of extra links for covering these client nodes is

\[
\sum_{p=1}^{n} u_p y_p
\]  

(3)

As previously mentioned, \(x_p > 1\) means that \(C_p\) connects to multiple routers at the same time, and there are redundant links existing in WMN.

Table I. The definition of symbols.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Explain</th>
</tr>
</thead>
<tbody>
<tr>
<td>((A_{ij})_{m \times n})</td>
<td>The relationship array between routers and client nodes. And (m (1 \leq i \leq m)) denotes the number of routers in upper layer, and (n (1 \leq j \leq n)) denotes the number of client nodes in under layer</td>
</tr>
<tr>
<td>(x_j)</td>
<td>The number of connected routers of client node (j). (x_j = \sum_{i=1}^{m} A_{ij})</td>
</tr>
<tr>
<td>(w_j)</td>
<td>The weight of (x_j).</td>
</tr>
<tr>
<td>(\beta)</td>
<td>The connectivity weight of routers.</td>
</tr>
<tr>
<td>(B_m)</td>
<td>The connectivity value of routers which is related to (m).</td>
</tr>
<tr>
<td>(y_p)</td>
<td>The judgement whether the client node (C_p) connects any router. (1 \leq p \leq n)</td>
</tr>
<tr>
<td>(u_p)</td>
<td>The weight of redundancy.</td>
</tr>
<tr>
<td>(\beta)</td>
<td>The average throughput of node.</td>
</tr>
<tr>
<td>(R_{min}, R_{max})</td>
<td>The lower and upper limitation of connectivity.</td>
</tr>
<tr>
<td>(Thr_{min})</td>
<td>The lower limitation of throughput.</td>
</tr>
<tr>
<td>(M)</td>
<td>The size of the initial population.</td>
</tr>
<tr>
<td>(M_0)</td>
<td>The size of the memory unit.</td>
</tr>
<tr>
<td>(R_c)</td>
<td>The ratio of antibody.</td>
</tr>
</tbody>
</table>

\( v_p \) presents the weight of redundancy, and the sum of redundancy is

\[
\sum_{p=1}^{n} v_p (x_p - 1)
\]  

(4)

Putting off the cost of extra links, the total redundancy of the WMN is

\[
\sum_{p=1}^{n} v_p (x_p - 1) - \sum_{p=1}^{n} u_p y_p
\]  

(5)

In addition, the extra links induce the enhancement of connectivity. It means that the connectivity will be improved when \( y_p = 1 \). \( w_0 \) indicates the weight of connectivity in this case. Hence, the increased connectivity is

\[
w_0 \sum_{j=1}^{n} y_j
\]  

(6)

As a result, the connectivity of the WMN is as follows.

\[
\sum_{j=1}^{n} w_j x_j + \beta B_m + w_0 \sum_{j=1}^{n} y_j
\]  

(7)

(3) The throughput of network

For ease of algorithm design, we simplify the computing method of throughput. \( P_1 \) denotes the average throughput of node \( C_j \) with the weight \( h_1 \) when \( x_j > 0 \). \( P_2 \) denotes the average throughput of node \( C_j \) with the weight \( h_2 \) when \( x_j = 0 \) or \( y_j = 1 \). Hence, the total throughput of client node is as follows.

\[
h_1 P_1 \sum_{j=1}^{n} x_j + h_2 P_2 \sum_{j=1}^{n} y_j
\]  

(8)

\( P_m \), the function of \( m \), presents the throughput between routers with the weight \( h_m \). Hence, the total throughput of WMN is

\[
h_1 P_1 \sum_{j=1}^{n} x_j + h_2 P_2 \sum_{j=1}^{n} y_j + h_m P_m
\]  

(9)

(4) Restriction condition

In order to guarantee the normal operation of WMN, MOM-IA must satisfy some restriction conditions as follows.

The total number of links between client nodes and routers should not be less than the number of client nodes.

\[
\sum_{j=1}^{n} x_j \geq n
\]  

(10)

For communication requirement, the connectivity should not be less than the connectivity threshold \( Con_{\text{min}} \).

\[
\sum_{j=1}^{n} w_j x_j + \beta B_m + w_0 \sum_{j=1}^{n} y_j \geq Con_{\text{min}}
\]  

(11)
For fault tolerant requirement, the redundancy has the lower limitation $R_{\text{min}}$.

$$\sum_{p=1}^{n} v_p(x_p - 1) - \sum_{p=1}^{n} u_p y_p \geq R_{\text{min}}$$  \hspace{1cm} (12)

In order to guarantee that the redundancy is limited in acceptable range, it also has the upper limitation $R_{\text{max}}$.

$$\sum_{p=1}^{n} v_p(x_p - 1) - \sum_{p=1}^{n} u_p y_p \leq R_{\text{max}}$$  \hspace{1cm} (13)

Toward more and more services in WMN, the requirement of throughput becomes larger and larger. $\text{Thr}_{\text{min}}$ means the threshold to provide service normally.

$$h_1 P_1 \sum_{j=1}^{n} x_j + h_2 P_2 \sum_{j=1}^{n} y_j + h_m P_m \geq \text{Thr}_{\text{min}}$$  \hspace{1cm} (14)

We construct the restriction functions as follows.

$$g_1(x) = n - \sum_{j=1}^{n} x_j$$  \hspace{1cm} (15)

$$g_2(x) = C_0 - \left( \sum_{j=1}^{n} w_j x_j + \beta B_m + w_0 \sum_{j=1}^{n} y_j \right)$$  \hspace{1cm} (16)

$$g_3(x) = R_{\text{min}} - \left( \sum_{p=1}^{n} v_p(x_p - 1) - \sum_{p=1}^{n} u_p y_p \right)$$  \hspace{1cm} (17)

$$g_4(x) = \left( \sum_{p=1}^{n} v_p(x_p - 1) - \sum_{p=1}^{n} u_p y_p \right) - R_{\text{max}}$$  \hspace{1cm} (18)

$$g_5(x) = \text{Thr}_{\text{min}} - \left( h_1 P_1 \sum_{j=1}^{n} x_j + h_2 P_2 \sum_{j=1}^{n} y_j + h_m P_m \right)$$  \hspace{1cm} (19)

(5) Mathematic model

In order to satisfy the requirement of multi-objective optimization problem, we transfer equations (7), (9), and (5) to acquire the equations (20), (21), and (22) as follows.

$$f_1(x) = \left( - \left( \sum_{j=1}^{n} w_j x_j + \beta B_m + w_0 \sum_{j=1}^{n} y_j \right) \right)_{\text{min}}$$  \hspace{1cm} (20)

$$f_2(x) = \left( - \left( h_1 P_1 \sum_{j=1}^{n} x_j + h_2 P_2 \sum_{j=1}^{n} y_j + h_m P_m \right) \right)_{\text{min}}$$  \hspace{1cm} (21)

$$f_3(x) = \sum_{p=1}^{n} v_p(x_p - 1) - \sum_{p=1}^{n} u_p y_p$$  \hspace{1cm} (22)

Considering restriction conditions, we set

$$G_j(x) = \max \{0, g_j(x)\}, 1 \leq j \leq 5$$  \hspace{1cm} (23)
\[ f_4(x) = G(x) = \sum_{j=1}^{s} G_j(x) \]  

As a result, the MOM is expressed as follows.

\[
\begin{align*}
\min y &= f(x) = (f_1(x), f_2(x), \ldots, f_{k+1}(x)) \\
x &= (x_1, x_2, \ldots, x_n) \in X \\
X &= \{ (x_1, x_2, \ldots, x_n) | 0 \leq x \leq n \} \\
y &= (y_1, y_2, \ldots, y_n) \in Y
\end{align*}
\]

where \( k = 3 \), which means that there are four destination functions denoting the connectivity, redundancy, throughput and restriction. If \( f_{k+1}(x) = 0 \), it means that \( x \) satisfies the restriction conditions and it is a feasible solution, or it is an infeasible solution.

From the aforementioned equations, our model is easy to scale and is suitable for more objects no matter whether they are conflicting. The transformations are only presented in the aspect of parameters and equations.

4.2. The solving scheme based on immune algorithm

To clearly express our point of view, we present the process of the immune algorithm in Figure 2.

The detail of process is described as follows.

(1) Initialization

In this stage, we set some parameters for algorithm. \( count \) indicates the number of iterations. The initial value is 0, and the maximum is \( count_{\text{max}} \). With accumulation of \( count \), our algorithm stops when \( count = count_{\text{max}} \).

![Figure 2. The process of multi-objective optimization model based on immune algorithm.](image-url)
$M$ denotes the size of random antibody population, and $M_0$ represents the size of memory unit. $N_{\text{max}}$ indicates the maximum of antibody in each generation. If the number of antibody exceeds $N_{\text{max}}$, the update operation is executed to remove part of antibody. The initial random set of antibody population is $X^{(\text{count})} = \{X_1^{(\text{count})}, X_2^{(\text{count})}, \ldots, X_m^{(\text{count})}\}$, whereas the member of set corresponds $x_p$. We select $M_0$ antibodies to compose the set of memory unit

$$M^{(\text{count})} = \{m_1^{(\text{count})}, m_2^{(\text{count})}, \ldots, m_{M_0}^{(\text{count})}\}$$

(2) The clone operation of antibody

In this stage, we clone the initial antibodies according to certain proportion $C_m$. It means that each antibody has $C_m$ individuals after cloning. This operation is prepared for mutation operation. The antibody population after cloned is

$$X'^{(\text{count})} = \{X_1'^{(\text{count})}, X_2'^{(\text{count})}, \ldots, X_m'^{(\text{count})}\}$$

while $X_m^{(\text{count})} = \{X_1^m^{(\text{count})}, X_2^m^{(\text{count})}, \ldots, X_m^m^{(\text{count})}\}$.

(3) The mutation operation

This operation is to mutate the set of cloned and memory unit. Assuming $X$ is an antibody. $X = (x_1, x_2, \ldots, x_i, \ldots, x_{m-1}, x_m)$. After mutation, we can get $\bar{X} = (x_1, x_2, \ldots, j, \ldots, x_{m-1}, x_m)$ . The process is shown in Figure 3.

The mutation rule is as follows.

Choosing random number $r_1$ and computing $i = (r_1 \mod n) + 1$, we select the $x_i$ as the mutation object.

Choosing random number $r_2$ and computing $j = r_2 \mod m$, we set $x_i = j$ which means $j$ replaces $x_i$.

The set of antibody population after mutation is marked as $X'^{(\text{count})}$.

(4) The clone selection operation

Computing the $(k+1)$th function value of restriction conditions in this generation, we classify the antibody according to these values. For each antibody after mutation operation, its $(k+1)$th function value is compared with zero. If the value is equal to zero, the antibody is classified into the set of feasible solution $X_f^{(\text{count})}$ or the antibody is classified into the set of infeasible solution $\bar{X}_f^{(\text{count})}$ . Furthermore, the set of feasible solution $X_f^{(\text{count})}$ is divided into Pareto-dominance solution set $P^{(\text{count})}$ and non-Pareto-dominance solution set $\bar{P}^{(\text{count})}$. And the set of infeasible solution $\bar{X}_f^{(\text{count})}$ is divided into beneficial set $Q_b^{(\text{count})}$ and nonbeneficial set $\bar{Q}_b^{(\text{count})}$, which is determined by the violation level to restriction. The antibody population after clone is denoted as $X''^{(\text{count})}$.

(5) The update operation of antibody population

After some generations, the number of the antibodies, which satisfy the condition of Pareto-dominance grows. It means that the computation load becomes larger, and the speed turns slower. Hence, the antibody number of Pareto-dominance must be checked after clone operation whether it reaches the maximum threshold $N_{\text{max}}$.

![Figure 3. The mutation operation.](image-url)
The update operation is as follows.

(a) Input the dimension of destination vector $m$ and the scale of antibody $\text{N}_{\text{non}}(\text{count})$, which presents the number of antibody in population. Let $i = j = 1$.

(b) Sort antibody population ascending by the $i$th destination function.

\[
[F(\cdot)]: i = [f(X'(\text{count})): i]
= ([f_1(X'(\text{count})), f_2(X'(\text{count})), \ldots, f_m(X'(\text{count}))]: i)
= [f_i(X'(\text{count}))]
\]

Assign an infinite value of fitness to antibody in boundary solution.

\[
c_{ij} = n_{\text{max}}, c_{im} = n_{\text{max}}
\]

Assign the fitness value $c_{ij}$ to other antibody.

\[
c_{ij} = \frac{(F(X'(\text{count}))(j + 1, i) - (F(X'(\text{count}))(j - 1, i))}{\delta + \max(F(X'(\text{count}))(\cdot, i) - \min(F(X'(\text{count}))(\cdot, i))}, 1 < j < m
\]

where $F(X'(\text{count}))(j + 1, i)$ indicates the $i$t destination function value of antibody $X'_{j+1}(\text{count})$.

(c) Compute the fitness value of $j$th antibody while $j(= 1, 2, \ldots, \text{N}_{\text{non}}(\text{count}))$

\[
f(c_j) = c_{ij} + c_{2j} + \ldots + c_{mj}
\]

(d) If $\text{N}_{\text{non}}(\text{count}) - \text{N}_{\text{max}}$ then stop, or go to step 5.

(e) Delete the antibody with the smallest fitness value. Get the new antibody population $X'''(\text{count})$ and destination function array $F_1(X'''(\text{count}))$. Let $\text{N}_{\text{non}}(\text{count}) := \text{N}_{\text{non}}(\text{count}) - 1, X''(\text{count}) = X'''(\text{count})$, $i := 1, j := 1$, then return to step 2.

Note $F(X'''(\text{count}))$ denotes the array composed by the destination function of antibody population $X'''(\text{count})$.

\[
\max(F(X'''(\text{count}))(\cdot, i) \text{ and } \min(F(X'''(\text{count}))(\cdot, i))\text{ indicate the maximum and minimum of the } i\text{th destination value in antibody population, respectively. } \delta \text{ is a small positive number, which is mainly to protect the denominator of } c_{ij} \text{ not to zero while } \max(F(X'''(\text{count}))(\cdot, i) = \min(F(X'''(\text{count}))(\cdot, i)). \text{ The antibody population after update operation is denoted by } X'''(\text{count}).
\]

(6) The learning operation of memory unit

Compute the antibody of beneficial set $Q_b(\text{count})$ if the violation level of an antibody is less than that in some member of $M(\text{count})$, then we replace the older one with this antibody in $M(\text{count})$.

(7) The preparation for next generation

After the computation of this generation, we check whether the scale of Pareto-dominance population achieves the initial scale. If it exceeds the scale of initial population, we completely replace initial population with antibodies from Pareto-dominance population while making sure the scale of population is stable. Otherwise, we adapt the Pareto-dominance population and pad the vacancy from initial population randomly.

5. THE ANALYSIS AND EXPERIMENT

5.1. The analysis of algorithm

(1) The highlight of algorithm

(a) Because some targets are incompatible, most of existing works usually focus on one aspect of the WMN. Even though some literatures consider multiple objects, they only provide an
approximate solution. Our paper proposes a paradigm to consider the incompatible targets at the same time. For better explanation, we choose three targets to create MOM. Furthermore, we design an efficient immune algorithm, which can evolve for a long time to compute optimum solution.

(b) MOM-IA classifies the solution set according to their characteristic. Even in infeasible solution set, it also remains some members with the low violation level. As a result, MOM-IA has the evolving ability from infeasible domain to Pareto-dominance domain. In addition, because the parameters and solutions in WMN are discrete integers, we adopt uniform random mutation. As a result, MOM-IA has incessancy evolving ability after many generations.

(c) MOM-IA includes two searching spaces which are independent. One is the antibody population after cloning initial population. The other is the population in memory unit. The first one based on random mutation has good performance in global searching aspect. The second one based on learning can locally search the boundary of infeasible solutions to improve the uniformity and universality of solution set. Hence, MOM-IA approaches the optimum solution from both sides of feasible and infeasible solution at the same time.

(2) The time complexity of algorithm

In this section, we analyze the time complexity of MOM-IA. \( m \) denotes the number of objects, and \( N_{max} \) indicates the reserved number of nondominance antibody in each generation.

In initialization stage, MOM-IA randomly creates the initial population with the scale of \( M \) and the memory unit with the scale of \( M_0 \). Hence, the time complexity is \( O(M + M_0) \). In clone stage, MOM-IA clones the antibody according to the ratio \( R_c \) in both initial population and memory unit. The time complexity is \( O(R_c(M + M_0)) \). Because the mutation operation is executed both in two populations, the time complexity of mutation is also \( O(R_c(M + M_0)) \). In the choosing process, because the solutions are divided into several types, we must consider the different situations. At first, the destination functions should be computed to determine whether the solution is feasible or infeasible. The time complexity is \( O(mR_c(M + M_0)) \). Then, we should subdivide the feasible solution set and infeasible solution set. The time complexity for classifying the feasible solutions is \( O(R_c^2(M^2 + M_0^2)) \), and the time complexity for estimating beneficial solutions is \( O(R_c(M + M_0)) \). The complexity for cloning Pareto-domiance solutions is \( O(R_c(M + M_0)) \). The complexity of update operation is \( O(R_c(M + M_0)) \). In the learning stage and the preparing stage, the complexity is \( O(R_cMM_0) \) and \( O(M) \), respectively.

Specially, from the clone stage, each step must loop \( count_{max} \) times. Hence, the total time complexity is

\[
O(M + M_0) + count_{max} \left[ O\left( (m + 3)R_c(M + M_0) + O(R_cMM_0) + O(M) + O(R_c(M^2 + M_0^2)) \right) \right]
\]

Because \( 0 < R_c < 1 \), the time complexity is \( O(M^2 + M_0^2) \).

We compare the time complexity of V-detector [20] and MOM-IA in Table II. From Table II, we can find that our algorithm has similar performance with V-detector in initialization, clone, and choosing stage. But the improvement in clone stage is obvious. As a result, the time complexity of MOM-IA in worst situation is \( O(M^2 + M_0^2) \), which is better than that of V-detector.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Initialization</th>
<th>Clone</th>
<th>Mutation</th>
<th>Choosing</th>
<th>Total (worst situation)</th>
</tr>
</thead>
<tbody>
<tr>
<td>V-detector</td>
<td>( O(M) )</td>
<td>( O(M_0^3) )</td>
<td>( O(M_0^4) )</td>
<td>( O(M + M_0) )</td>
<td>( O((M + M_0)^3) )</td>
</tr>
<tr>
<td>MOM-IA</td>
<td>( O(M + M_0) )</td>
<td>( O(R_c(M + M_0)) )</td>
<td>( O(R_c(M + M_0)) )</td>
<td>( O(mR_c(M + M_0)) )</td>
<td>( O(M^2 + M_0^2) )</td>
</tr>
</tbody>
</table>

MOM-IA, multi-objective optimization model based on immune algorithm.
5.2. Experiment

In our opinion, the aim of our experiment is to compare the efficiency and evolution ability of our schemes with that of other algorithms. Hence, we focus on the improvement of immune algorithm and choose the parameters of multiple objects randomly.

(1) The setting of parameters

In this paper, we collect our experiment data from a mesh network test-bed. The specific configurations of the test-bed are listed in Table III. In our experiment, we set 10 client nodes and 10 routers in WMN.

The weights of connectivity, redundancy, and throughput in client nodes are randomly set as Table IV.

The other parameters are set as follows. In aspect of connectivity, the connectivity factor is $\beta = 2$. The connectivity function of router is $B_m = m^2 + m$, and the connectivity weight of extra links is 1. The minimum threshold of network connectivity is $Con_{min} = 300$.

In redundancy aspect, the redundancy weight of extra links is $u_p = 1$. The minimum redundancy requirement for network fault-tolerant is $R_{min} = 60$, and the maximum redundancy requirement for network efficiency is $R_{max} = 90$.

In throughput aspect, the throughput weight of client node and extra links are $h_1 = 2$ and $h_2 = 1$, respectively. The throughput of router is $h_m = 3$. The throughput function of router is $P_m = m^2 + m$, and the minimum weight sum of network throughput is $Thr_{min} = 400$.

In population aspect, the initial scale of population is $M = 50$ and the scale of memory unit is $M_0 = 50$. The number of generation is $count_{max} = 100$ and the max reserved scale of Pareto-dominance population in each generation is $N_{max} = 50$.

(2) The analysis of result

For verifying the performance of proposed model and solving algorithm, we analyze the connectivity, redundancy, and throughput of the WMN and compared the results between MOM-IA and V-detector, which are both multi-objects optimization based on immune algorithm. In the experiment, we focus on the three aspects as follows.

### Table III. The configurations of test-bed.

<table>
<thead>
<tr>
<th>Device</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mesh nodes</td>
<td>Five notebook computers and five wireless cards in different desktop computers (ALFA AWPCI085H: 802.11a/b/g 1000mW miniPCI card)</td>
</tr>
<tr>
<td>Routers</td>
<td>10 mesh tool boxes which are embedded system. Each of them includes one soekris motherland, two miniPCI radio receivers, ethernet interface, antenna and outdoor stent equipment.</td>
</tr>
<tr>
<td>Interface</td>
<td>U.FL.to RP-TNC</td>
</tr>
<tr>
<td>RF cable</td>
<td>Extend the distance between receivers.</td>
</tr>
<tr>
<td>Adapter</td>
<td>The PCI interfaces of wireless cards to desktop computers.</td>
</tr>
<tr>
<td>Battery</td>
<td>Interface number 27 DC-2, group scale 27, CA @ 32’ F-715, CCA @ 0’ F-575, cruising ability 175 min @ 90 AH</td>
</tr>
<tr>
<td>Converter</td>
<td>Power express 400 Watt mains power inverter</td>
</tr>
<tr>
<td>GPS</td>
<td>Garmin 18× USB GPS</td>
</tr>
</tbody>
</table>

PCI, peripheral component interconnect.

### Table IV. The weights of client nodes.

<table>
<thead>
<tr>
<th>ID</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connectivity</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Redundancy</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Throughput</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
(a) The relationship between efficiency and reliability

In WSN, throughput is the key factor of efficiency while the redundancy presenting the fault-tolerant ability and connectivity denoting the service providing ability are the key factors of reliability.

With the random selection method, 50 solutions are chosen from Pareto-dominance set to compute the weight of connectivity, redundancy, and throughput. The relationship between redundancy and throughput is shown in Figure 4.

From the Figure 4, we can find that the redundancy becomes larger with the throughput increasing. When the throughput is small, the requirement of fault-tolerant is low, and the redundant data is little. With the throughput rising, the requirement of redundancy is increased. Hence, the expectations of low redundancy and high throughput are conflicting.

Figure 5 shows the relationship between connectivity and redundancy. Obviously, the redundancy is increased with the requirement of connectivity enhancing. Therefore, the expectations of good connectivity and small redundancy are contradictory.

As a result, the destination of MOM-IA is to balance these aspects in WMN.

(b) The Pareto-front distribution of algorithm

Because of the similarity between MOM-IA and V-detector, we compare them to analyze the relative merits of our algorithm. Randomly selecting instances from Pareto-dominance solution set and computing the weight of connectivity, redundancy and throughput, we create a three-dimensional reference system and draw the Figure 6 as follows.

Figure 4. The relationship between redundancy and throughput.

Figure 5. The relationship between connectivity and redundancy.
Figure 6(a) and (b) denote the Pareto-front distribution of MOM-IA and V-detector. Because of the memory unit, MOM-IA has the probability to increase the feasible solutions from the infeasible solution set near the boundary. This design produces some advantages. From the figures, we can find that the solution set diversity and Pareto-front distribution uniformity of MOM-IA are better than that of V-detector in the same initial conditions.

On the one side, if the number of Pareto-dominance solution set is small in the clone selection stage of next generation, MOM-IA can randomly replenish the set from the boundary feasible solutions. As a result, this mechanism enhances the solution set diversity of MOM-IA.

On the other side, we randomly select 50 Pareto-dominance solutions by MOM-IA and V-detector, respectively. Computing the weight of connectivity, redundancy, and throughput, which are coordinate components of three-dimensional reference system, we get the results as in Table V.

Table V. The coordinate components statistics.

<table>
<thead>
<tr>
<th>Component</th>
<th>MOM-IA</th>
<th>V-detector</th>
<th>MOM-IA</th>
<th>V-detector</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Max</td>
<td>Min</td>
<td>Expect</td>
<td>Standard deviation</td>
</tr>
<tr>
<td>Connectivity</td>
<td>339</td>
<td>300</td>
<td>316.28</td>
<td>9.58</td>
</tr>
<tr>
<td>Redundancy</td>
<td>90</td>
<td>61</td>
<td>75.84</td>
<td>7.86</td>
</tr>
<tr>
<td>Throughput</td>
<td>450</td>
<td>414</td>
<td>429.28</td>
<td>8.61</td>
</tr>
</tbody>
</table>

MOM-IA, multi-objective optimization model based on immune algorithm.
The statistics results show that the maximum of each component in MOM-IA is larger than that in V-detector, and the minimum of each component in MOM-IA is smaller than that in V-detector. It means that MOM-IA has wider Pareto-front distribution than V-detector. In addition, MOM-IA has bigger standard deviation, which means that it has better distribution uniformity.

As a result, MOM-IA has bigger probability to achieve optimal solution than V-detector in the environment of WMN.

(c) The evolution ability of algorithm

Because of the dynamic feature of the WMN, the proposed algorithm needs good evolution ability and reliability to make sure that it is suitable for the environment of WMN. The ratio of new antibody in different generation is presented in Figure 7.

Figure 7 shows that MOM-IA has the high generation ratio of new antibody, which means the high evolution ability and reliability. However, the antibody generation ratio of V-detector decreases observably after 50 generations. When generation reaches 80, there is almost no new antibody created.

V-detector adopts nonspecific mutation, which has good global searching effect in early stage and good local searching effect when the solutions are continuous variables. However, most parameters in WMN are discrete integers. MOM-IA is suitable for dealing with discrete integers and has good evolution ability. Hence, MOM-IA is more compatible with WMN.

6. CONCLUSION

Wireless mesh network can extend the scale of existing network conveniently, so it has wide application prospection. With the growth of network scale, some metrics may decrease sharply and become the bottleneck to improve the performance of network. Hence, analyzing the relationship of multiple metrics is the basis of network optimization. Considering the restriction between the metrics of WMN, this paper proposes an MOM-IA. MOM-IA provides a paradigm to create optimization model with multiple conflicting objects. Because of the dynamic feature of WMN, we design the scheme on the basis of immune algorithm, which has good evolution ability. The experiment results show that MOM-IA is qualified to optimize WMN performance.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (Grant numbers 61272451, 61272405, 61173154, 61373169). The authors would like to thank the anonymous reviewers of the paper for their valuable comments. The corresponding author is Fajiang Yu.
REFERENCES


