计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.200800132

蚁群算法在动态网络持续性路径预测中的运用及仿真

杨 林 王永杰
国防科技大学电子对抗学院 合肥 230037
安徽省网络空间安全态势感知与评估重点实验室 合肥 230037
(yanglin0815@nudt.edu.cn)

摘 要 随着主动防御手段的广泛运用,动态多变性成为了网络系统的显著特征,在讨论了网络系统安全性时不可避免地需要以动态网络环境为基础,路径预测作为网络安全评估的常用方法,也需要适应动态网络环境以具备持续高效的特性。为了解决这个问题,提出将蚁群优化算法运用到网络持续性路径预测中,并设计仿真实验,在寻优精度和寻优速度两个方面,将所提方法与完全随机算法和贪婪算法进行比较。仿真实验结果表明,原始蚁群算法的寻优精度不如完全随机算法,但由于启发式信息的引导,其寻优速度远优于完全随机算法。为了均衡原始蚁群算法和完全随机算法各自的优势,提出新的蚁群信息素更新策略,并再次设计仿真实验验证算法的寻优效率。最终的实验结果显示,改进后的蚁群优化算法能够较好地综合原始蚁群算法和完全随机算法的优点,达到寻优精度和寻优速度的均衡。然而,在下一步的研究中还需要继续进行算法优化,使其能够更好、更完全地继承两者的优点,实现精度和速度兼优。

关键词:蚁群优化算法;动态网络;路径预测;仿真实验
中图法分类号 TP393.08

Application and Simulation of Ant Colony Algorithm in Continuous Path Prediction of Dynamic Network

YANG Lin and WANG Yong-jie
College of Electromagnetic Countermeasure, National University of Defense Technology, Hefei 230037, China
Anhui Province Key Laboratory of Cyberspace Security Situation Awareness and Evaluation, Hefei 230037, China

Abstract With the widespread use of active defense methods, dynamic variability has become a prominent feature of network systems. When discussing network system security, it is inevitable to base on dynamic network environment. Path prediction, as a common method of network security assessment, also needs to adapt to dynamic network environment and have the characteristics of continuous and efficient. In order to solve this problem, it is proposed to apply the ant colony optimization algorithm to the continuous path prediction of the network, and to design a simulation experiment to compare it with the completely random algorithm and the greedy algorithm in terms of optimization accuracy and optimization speed. The simulation experiment results show that the optimization accuracy of the original ant colony algorithm is not as good as the completely random algorithm, but due to the guidance of heuristic information, its optimization speed is much better than the completely random algorithm. In order to balance the advantages of the original ant colony algorithm and the completely random algorithm, a new ant colony pheromone update strategy is proposed, and a simulation experiment is designed to verify the efficiency of the algorithm. The final experimental results show that the improved ant colony optimization algorithm can better integrate the advantages of the original ant colony algorithm and the completely random algorithm, and achieve a balance between optimization accuracy and optimization speed. However, it is necessary to continue to optimize the algorithm in the next research, so that it can better and more completely inherit the advantages of the original ant colony algorithm and the completely random algorithm, and achieve a high level both in accuracy and speed.

Keywords Ant colony optimization algorithm, Dynamic network, Path prediction, Simulation experiment

1 引言

计算机网络攻防是防御方和攻击方之间的不对称策略博弈[1]。系统架构的确定性和静态性质使得攻击者有足够的时间发起检测和攻击[2],增大了攻防博弈中攻击方的获胜概率。与此相反,动态的系统架构能够形成有效的防御体系,延缓攻击过程,从而降低攻击成功率。

随着网络攻击手段日益复杂化、智能化和多样化[3],为了形成动态的网络环境,许多针对性的主动防御技术被提出。移动目标防御(MTD)是主动防御技术的一种,最初由美国科学技术委员会提出[4],指采用IP地址和端口跳变、IP安全协议随机化用户角色随机化、地址空间随机化、指令集合随机化以及动态路由等技术手段动态调整系统的网络状态,使得攻击面呈现不可预测的效果。

通信作者:王永杰(w_yong_j@189.cn)

相比被动防御，以 Wu 首创的拟态防御[5]为代表的主动防御技术能够提前发现系统漏洞和潜在的安全威胁[6]并采取应对策略，引起了网络环境的更迭改变，如图 1 所示。

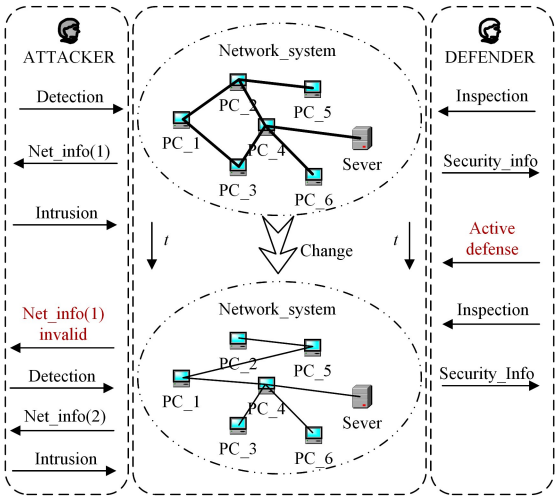


图 1 主动防御导致网络环境改变

Fig. 1 Active defense leads to network environment change

如图 1 所示，当防御方采取主动防御后，网络系统的拓扑环境发生改变，使得攻击方第一次探测得到的网络信息 *Net_info*(1)无效，因此需要继续进行网络信息探测，这种持续性的探测过程贯穿网络攻防博弈始终。与此同时，防御方需要经常对网络系统状态进行安全检查与评估。网络安全评估手段丰富多样，攻击图是一种基于图论的网络脆弱性评估方法，使用攻击图模型能够找出网络系统中各结点脆弱性间的联系，发现潜在的威胁及攻击因果关系[7]，并以攻击路径[8]或者攻击树[9]的形式展现出来。

使用攻击图进行网络安全评估需做入侵路径预测，动态环境下持续性路径预测相比静态环境更加复杂。随机和贪婪算法在持续性路径预测中效率低下，基于启发式信息优化的群体智能算法能够在多次迭代过程中不断累积经验，学习迭代过程中的优化结果，适用于持续性路径预测。

基于以上分析，为了提高防御方持续性路径预测效率，本文利用蚁群在路径分泌信息素的特性来提高历史信息学习能力，设计了一种基于蚁群算法的持续性路径预测算法，并通过仿真环境验证了该算法的效率。

2 相关工作

在动态网络环境下，攻击路径的预测依赖于动态路径规划算法。在机器人的路径规划中，Agent 路径规划的目标是在未知或者已知的环境中寻找从起点至终点的无碰撞路径，且满足最优的原则[10]。许多学者对动态或者未知环境下的路径规划方法进行了研究。文献[11]通过维护一个扩展树(expansion tree)并制定修剪规则来解决空间网络动态最短路径监视问题(Dynamic Shortest Path Monitoring，DSPM)，这种方法能够有效加快动态网络空间中最短路径持续计算速度。

最短路径需要动态监视的原因有两个：1)路径成本的改变；2)路径搜索偏离了预先计划的路径[11]。可以总结为需要持续维持最优路径的导向性，引导最优路径寻优以提高寻优效率。

图 2 给出了某一时刻 t 的网络结构及该时刻下的预测路径，i 为路径上的一条链路。假设在 $t+\Delta t$ 时刻，链路 i 由于防御方采取某种主动防御措施而断开，而其他节点和链路状态不变，那么将出现两种可选的预测路径，如图 3 所示。

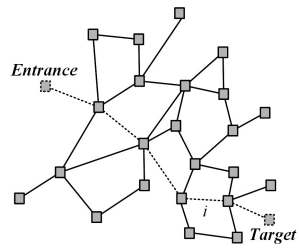


图 2 t 时刻的网络结构和预测路径

Fig. 2 Network structure and prediction path at time t

此时若采用随机算法重新规划路径，那么由于随机算法不对历史经验进行学习而笼统地追求随机化，可能会出现对整个搜索空间的遍历，导致寻优效率慢且精度低。

若采用贪婪算法重新进行路径规划，由于贪婪算法盲目地追求局部最优化，可能会为寻优添加错误引导，全局寻优能力弱。

图 3 中，如果盲目地追求局部最优，由于链路 a 在 t 时刻被认为是一条局部最优链路，贪婪算法不舍得放弃该链路而紧接着采用链路 b，倘若链路 a 和 b 的总成本远大于路径 c 和 d 或者其他链路，那么此时的路径预测就会错过路径②或其他更需要被选中的路径，引起错误判断，从而导致攻击方采取错误的防御策略。

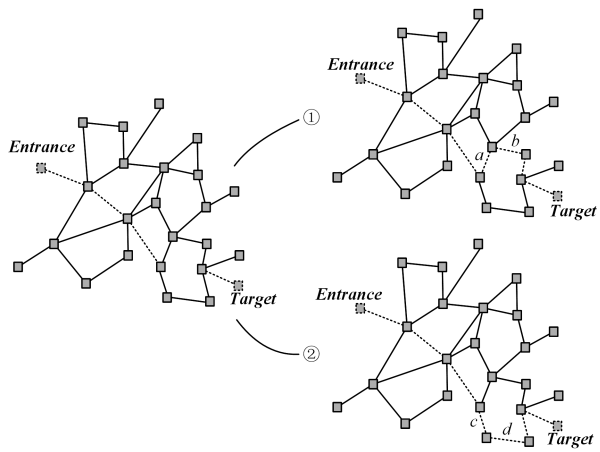


图 3 $t+\Delta t$ 时刻可选的预测路径

Fig. 3 Optional prediction path at time $t+\Delta t$

因此，在动态网络环境中进行路径规划时，需要在追求局部最优信息的基础上保留部分随机因素，同时不断弱化前一时刻的判断对当前时刻的影响，这恰好符合蚁群优化算法的思想。蚁群算法是一种优化算法，其利用经验启发式信息，能够优化一些复杂问题的求解过程，如常用于优化解决复杂网络中的社区发现问题[12-13]和复杂公路网中无人驾驶车辆的路径规划问题[14]，显然运用在动态路径规划中能够提高多次迭代中的路径导向能力，提升优化效率。

3 优化方法与优化原则

蚁群算法最初由 Dorigo 于 1991 年首次应用于解决同样是路径规划问题的 TSP 问题[15]。受蚂蚁群体觅食行为启发，学者提出了信息素的概念，蚂蚁觅食过程中借助信息素进

行信息的交流和传递，能够根据所走路径的长度及信息素浓度自主选择下一跳方向，并表现出正反馈行为，这种正反馈机制能够帮助蚂蚁更快地找到最优觅食路径，形成优化迭代。

作为一类启发式仿生进化算法，蚁群优化算法已被广泛应用于多个领域，并取得了较好的效果。动态网络环境下的路径规划问题与无人驾驶车辆在复杂公路网中的路径规划问题具有相似性，蚁群优化算法早已应用于优化求解该问题模型。

蚁群优化算法根据信息素更新策略的不同被分为 3 类，分别是蚁周模型、蚁量模型和蚁密模型，设蚁群数量为 m，网络节点规模为 n，典型蚁周模型的算法流程如图 4 所示。

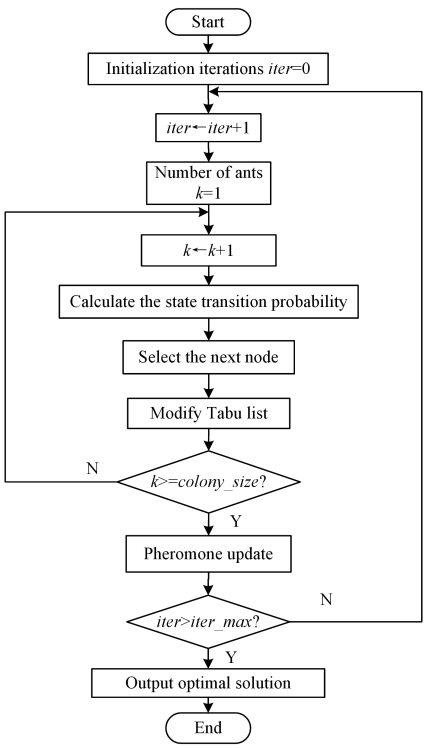


图 4　典型蚁周模型的算法流程

Fig. 4　Algorithm flow of typical ant-cycle model

上述算法流程包含两个过程，分别称为状态转移过程和信息素更新过程。

(1)状态转移。在 t 时刻蚂蚁随机选择下一个节点，位于 i 节点的蚂蚁 k 选择 j 节点作为转移目标的概率计算公式为：

$$\rho_{ij}^{k}(t)=\begin{cases}\dfrac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum\limits_{s\in allow_k}\tau_{is}^{\alpha}(t)\eta_{is}^{\beta}(t)}, & j\in allow_k\\ 0, & j\notin allow_k\end{cases}$$

其中，τ_{ij} 为在 t 时刻路径 (i,j) 上的信息素浓度；η_{ij} 为路径 (i,j) 距离的启发式信息，蚁周模型通常定义 $\eta_{ij}=1/d_{ij}$；$allow_k$ 是蚂蚁 k 在 t 时刻未访问的节点集，常用禁忌表进行访问限制。

(2)信息素更新。为防止信息素堆积导致信息启发作用增强而淹没期望启发的作用，每只蚂蚁在完成一步(蚁量模型和蚁密模型)或者一次迭代(蚁周模型)之后需要对路径上的信息素进行积累和挥发处理。$t+n$ 时刻在路径 (i,j) 上的信息素更新公式为：

$$\tau_{ij}(t+n)=(1-\rho)\tau_{ij}(t)+\Delta\tau_{ij}(t)$$

$$\Delta\tau_{ij}(t)=\sum_{k=1}^{m}\Delta\tau_{ij}^{k}(t)$$

其中，ρ 为信息素挥发因子，$(1-\rho)\tau_{ij}(t)$ 为挥发剩余信息素，$\Delta\tau_{ij}(t)$ 为 t 时刻的信息素增量，在蚁周模型中信息素的增量为 Q/L_k，其中 Q 称为信息素增强系数，L_k 为整个路径的长度。在网络攻防博弈中，“路径长度”常常体现为攻击方发起该链路所对应攻击动作产生的攻击成本，在网络攻防博弈模型[16]中这种攻击成本可以通过通用基于漏洞评分系统 CVSS 评估或者通过攻防策略分析来量化。

在图 3 所示的情况下，采用蚁周模型时，蚁群在迭代过程中积累经验并发现链路 b 成本较高，逐渐停止继续沿着 a 和 b 前进，而改用路径②或者其他更优路径，如图 5 所示。蚁群优化算法这种“学习总结”的能力使其在迭代过程中具有更高的寻优速度。

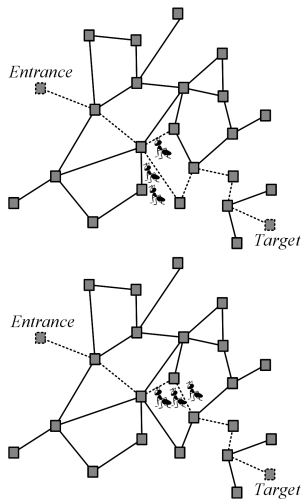


图 5　采用蚁群优化的路径预测

Fig. 5　Path prediction of ACO is adopted

蚁群优化算法应用于网络持续路径预测的伪代码如算法 1 所示。

算法 1　基于蚁群算法的持续路径规划算法

输入：G，m，ρ，α，β，Q，iter_max

输出：Path，c

```
1. algorithm NetACO
2. 初始化：iter←1；/ * iter 为当前迭代数 * /
3. η←1/G；/ * 启示因子设为路径成本的倒数 * /
4. Phe←(1，sizeof(G)，sizeof(G))；/ * 信息素表 * /
5. Tabu←(0，sizeof(G))；/ * 禁忌表 * /
6. Path←Φ；/ * 最优路径 * /
7. c←0；/ * 最优路径成本 * /
8. while iter<iter_max do
9.     Tabu(1：m，1)←1；/ * 由 Entrance 出发 * /
10.    for j←1 to sizeof(G)
11.      for i←1 to m do
12.        SelectNextHop(Tabu，α，β)；/ * 依概率函数选择下一跳节点 * /
13.      end for
14.    end for
15.    c←Min(length(Path(1：m)))；/ * 当前迭代最优路径长度 * /
16.    Path←Path(Path(1：m)==c)；/ * 最优路径 * /
17.    Phe←Updata(Phe，ρ，Q)；/ * 更新信息素 * /
18.    Tabu←(0，sizeof(G))；/ * 禁忌表清零 * /
19.    if Check(NetStatus)==‘changed’ do / * 持续检查网络状态 * /
20.      NetACO(GetNewPara())；/ * 若网络发生改变则重新开始算法 * /
21.    else / * 否则返回迄今得到的最优路径 * /
```

```
22.     return (Path,c)
23.     iter←iter+1;/*继续迭代*/
24. end while
25. end algorithm /*算法结束*/
```

为了验证蚁群优化算法对网络持续路径预测带来的影响，结合动态网络的实际情况，设计仿真环境对蚁群优化算法与随机和贪婪算法的持续性路径预测效果进行比较。

4 原始蚁群算法的仿真实验

考虑网络拓扑结构而不讨论网络的具体状态、脆弱性及双方的攻防细节，观察算法对路径预测的效果。

在抽象网络拓扑中，点或者边的连通度最早被用来描述网络的抗毁性[17]，其被定义为为了使图变成不连通或者平凡图所需要去掉的最少节点或者边的数目[18]。点(边)连通度采用逆向排除的方法定义网络的抗毁性，在稀疏网络下具有一定的优势，但是当网络复杂性增加时，逆向排除达到目标条件的时间开销实际上远大于正向搜索。

对于一个全连通网络，假设共有 n 个节点，且将其考虑为无向图，那么这个网络共存在 $n(n-1)/2$ 条边。设置连通完整度 φ，将其定义为网络中边的数目与网络全连通状态下边数的比值。

$$\varphi=\frac{e_f}{e_c}$$

其中，e_f 为节点数为 n 的全连通网络边的数目，e_c 为实际网络边的数目。显然，若当前网络为全连通网络，则 $e_f=e_c$，此时 $\varphi=1$。又因为 $e_f=n(n-1)/2$，所以 φ 的计算式如下：

$$\varphi=\frac{n(n-1)}{2e_c}$$

连通完整度 φ 可以简单地映射出节点 a 和节点 b 之间边 (a,b) 出现的概率。

设计两种规模的动态网络拓扑环境，其节点个数和连通完整度分别设置为 $n=10$，$\varphi=0.25$ 和 $n=48$，$\varphi=0.05$，并在动态变化过程中尽量保证 φ 值不变，如图 6 和图 7 所示，图中各链路权重不做详细标注。接下来以这两种动态网络拓扑环境为基础，进行算法的仿真实验。

为了使完全随机算法和蚁群优化算法仅在路径寻优方法上体现出差异，而不受寻优群体规模的影响，固定蚁群规模 $m=1$，其余参数设置如表 1 所列。

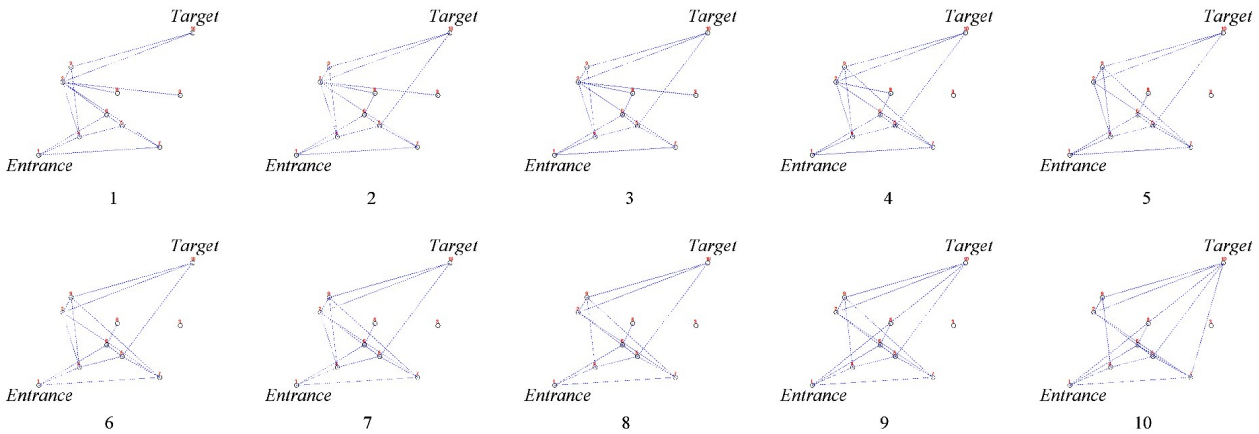


图 6 10 节点动态网络拓扑

Fig. 6 Dynamic network topology with 10 nodes

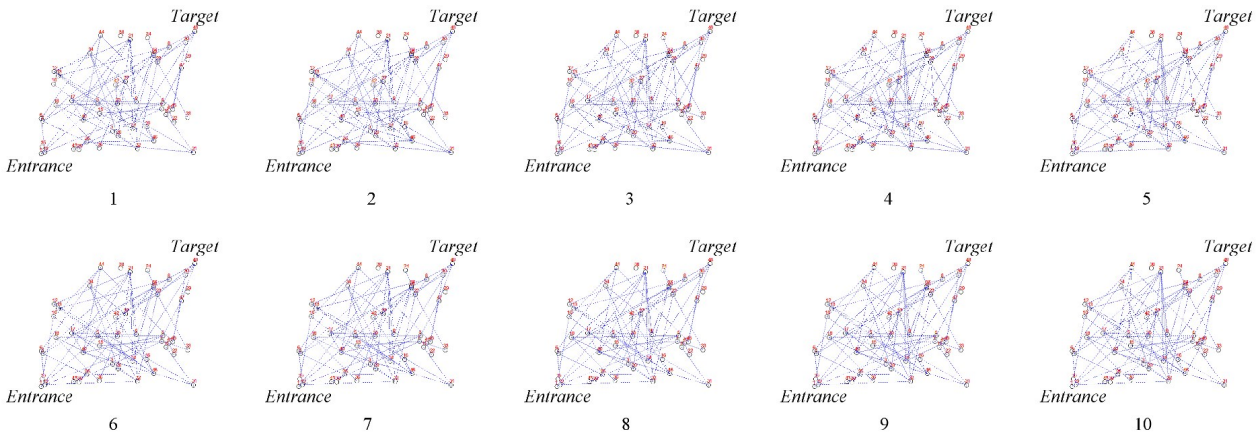


图 7 48 节点动态网络拓扑

Fig. 7 Dynamic network topology with 48 nodes

表 1 实验参数设置

Table 1 Experimental parameter setting

Parameter	*phe*	α	β	ρ	Q	*iter_max*
Value	100	1	2	0.5	1	100

表 1 中，*phe* 为信息素初始含量，*iter_max* 为当前网络状态下的最大迭代次数。

记录 3 种算法在每一种网络状态下的寻优结果 *opt* 以及收敛到该最优结果所需迭代的次数 *co*，10 节点和 48 节点动

态网络下的实验结果如表 2 和表 3 所列。

表 2　10 节点动态网络的实验结果

Table 2　Experimental results of 10-nodes dynamic network

Status	*Random*		*Greedy*		*ACO*	
	opt	*co*	*opt*	*co*	*opt*	*co*
1	99	17	∞	∞	99	7
2	99	22	∞	∞	99	1
3	164	18	∞	∞	189	1
4	164	4	∞	∞	189	1
5	164	24	∞	∞	189	1
6	186	84	∞	∞	210	1
7	186	67	∞	∞	210	1
8	190	41	∞	∞	201	1
9	38	5	38	1	201	1
10	38	2	38	1	201	1

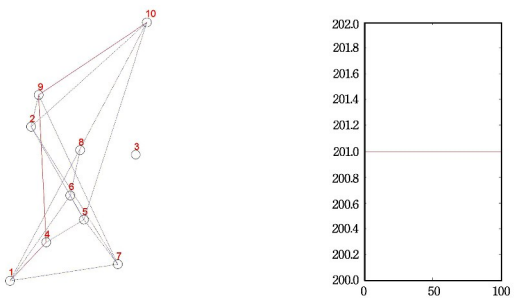
表 3　48 节点动态网络的实验结果

Table 3　Experimental results of 48-nodes dynamic network

Status	*Random*		*Greedy*		*ACO*	
	opt	*co*	*opt*	*co*	*opt*	*co*
1	217	2	∞	∞	386	3
2	244	6	∞	∞	358	1
3	312	14	∞	∞	332	1
4	312	18	∞	∞	332	1
5	312	3	∞	∞	332	1
6	312	6	∞	∞	332	1
7	312	6	∞	∞	332	1
8	312	10	∞	∞	332	1
9	312	5	∞	∞	332	1
10	305	100	∞	∞	332	1

5　实验分析及改进

从表 2 和表 3 的实验结果可以发现，尽管蚁群算法的优化速度较快，但是优化结果的精度太低。通过分析发现，由状态 8 变化到状态 9 时，网络中添加了一条由 Entrance 节点到 Target 节点的链路，且该链路的攻击成本低，然而蚁群无法及时清除路径上残留的信息素，导致该链路的期望启发式因子作用被淹没，从而忽略该链路，此时所记录的蚁群算法优化过程如图 8 所示。



(a)当前优化结果　　(b)平均权值和最短权值

图 8　状态 9 下蚁群算法的优化过程

Fig. 8　Optimization process of ACO under status 9

经过上述实验可以发现，在路径寻优初期，增大算法的随机性能够改善寻优精度，虽然这样会降低寻优速度，但以初期的寻优结果引导蚁群分泌信息素可以优化后期蚁群寻优，因此改变蚁群信息素更新策略如下。

$$\tau_{ij}(t+n)=(1-\rho)\lceil \tau_{ij}(t)+\Delta\tau_{ij}(t)\rceil$$

为了让蚁群在初始状态下达到完全随机寻优的效果，以提高寻优精度，设置初始挥发率 $\rho=1$，并根据时间来动态调整挥发率，挥发率的调整公式随时间变化的公式如下。

$$\rho=1-time/(time\_max+1)$$

其中，$time$ 为当前状态的时间编号，可以理解为算法对网络的监视时间，$time\_max$ 为最大状态数，当无法估计最大状态数目时，可以依据经验将其设置为一个合理值，但不应该使信息素挥发率降至 0，因此这个值应该足够大。

在上述信息素更新策略下，算法初期链路上信息素挥发能力强，倾向于随机寻优，以建立更好的前期引导作用。为了验证改进后的蚁群优化算法的改进效果，设置新的动态网络环境：设置更大的网络连通完整度 $\varphi=0.5$，以增加寻优难度，使各算法的寻优效果能够明显差异化；其次延长状态数目为 1000，即 $time\_max=1000$，这使得对网络监视时间跨度更长，凸显了寻优后期蚁群优化算法的缺点，以清晰地显示出改进算法的优化效果；其余参数设置同表 1。比较原始蚁群算法、改进蚁群算法和随机算法的优化精度和优化速度，实验结果如图 9 和图 10 所示。

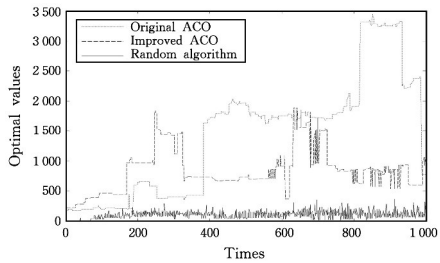


图 9　优化精度对比

Fig. 9　Optimization precision comparison

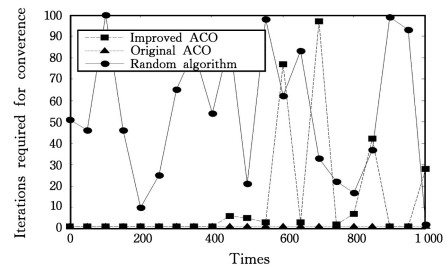


图 10　优化速度对比

Fig. 10　Optimized speed comparison

在上述所有实验中始终取 $m=1$，这意味着在所使用的蚁群算法中只是运用了蚁群分泌信息素的思想，而并未采取“群体优化”方式，相对完全随机算法来说是公平公正的。

可以发现，原始蚁群算法总能够在变化的网络环境下快速地寻得一条路径，在实际实验结果中，原始蚁群算法基本上只需花费一个迭代周期就可以获得路径。改进的蚁群算法在迭代后期损失了寻优速度，但是寻优结果远好于原始蚁群算法，且在寻优中后期能较准确地跳出局部最优。相比完全随机算法，其寻优结果仍然不是很好，但是寻优速度较完全随机算法有大幅度的提升。

结束语　本文主要讨论了蚁群算法在动态网络环境下的持续路径预测问题上的应用，通过仿真实验分析了贪婪算法、完全随机算法和传统蚁群算法的路径规划效果，其次综合完全随机算法和传统蚁群算法的优缺点，改进了蚁群优化算法信息素更新策略，并进行了更充实的仿真实验分析。实验结果表明，改进的蚁群优化算法作为传统蚁群算法与完全随机算法的平衡，能够较好地兼顾寻优精度和寻优速度，然而绝对

最优秀的算法需要既具备随机算法的精度又有拥有蚁群算法的寻优速度，这还需要在后续实验中进行研究和探讨。

参 考 文 献

[1] DURKOTA K, LISY V, KIEKINTVELD C, et al. Case Studies of Network Defense with Attack Graph Games[J]. IEEE Intelligent Systems, 2016, 31(5): 24-30.

[2] XIONG X L, YANG L, ZHAO G S. Effectiveness Evaluation Model of Moving Target Defense Based on System Attack Surface [J]. IEEE Access, 2019, 7: 9998-10014.

[3] GORDON L, LOEB M, LUCYSHYN W, et al. 2016 CSI/FBI computer crime and security survey [C]// Proceedings of the 2016 Computer Security Institute, 2016. San Francisco: IEEE.

[4] NITRD. Cybersecurity game-change research & development recommendations [OL]. http://www.nitrd.gov/pubs/CSIAIWG%20Cyberscurty%20GameChange_RD%20Recommendations 20100513.pdf, 2014-03-21.

[5] WU J X. Research on Cyber Mimic Defense[J]. Journal of Cyber Security, 2016, 1: 1-10.

[6] NI Z, LI Q M, LIU G. Game-Model-Based Network Security Risk Control [J]. Computer, 2018, 51(4): 28-38.

[7] HU Q. Attack Prediction Method Based on Multi - step Attack Scenario[J]. Computer Science, 2019(46): 365-369.

[8] ZHANG Y C, ZHOU T Y, GE X Y, et al. An improved attack path discovery algorithm through compact graph planning [J]. IEEE Access, 2019, 7: 1.

[9] MUÑOZ-GONZALEZ L, SGANDURRA D, PAUDICE A, et al. Efficient Attack Graph Analysis through Approximate Inference [J]. ACM Transactions on Privacy and Security, 2017, 20(3): 1-30.

[10] BENNET D J, MCINNES C R. Distributed control of multi-robot systems using bifurcating potential fields [J]. Robotics and Autonomous Systems, 2010, 58(3): 256-264.

[11] SHOU S, CHEN L S, GUO D H, et al. Dynamic Shortest Path Monitoring in Spatial Networks [J]. Journal of Computer Science and Technology, 2016, 31(4): 637-648.

[12] LIU R C, LIU J D, HE M M. A multi-objective ant colony optimization with decomposition for community detection in complex networks [J]. Transactions of the Institute of Measurement and Control, 2018, 41(9): 2521-2534.

[13] SHAHABI SANI N, MANTHOURI M, FARIVAR F. A multi-objective ant colony optimization algorithm for community detection in complex networks [J]. Journal of Ambient Intelligence and Humanized Computing, 2020, 11(1): 5-21.

[14] SONG Q, ZHAO Q L, WANG S X, et al. Dynamic Path Planning for Unmanned Vehicles Based on Fuzzy Logic and Improved Ant Colony Optimization [J]. IEEE Access, 2020, 8: 62107-62115.

[15] DORIGO M, GAMBARDELLA L M. Ant colony system: a cooperative learning approach to the traveling salesman problem [J]. IEEE Transactions on Evolutionary Computation, 1997, 1(1): 53-66.

[16] LIU J W, LIU J J, LU Y L, et al. Optimal Defense Strategy Selection Method Based on Network Attack-Defense Game Model [J]. Computer Science, 2018(45): 117-123.

[17] FRANK H, FRISCH I. Analysis and Design of Survivable Networks [J]. IEEE Transactions on Communication Technology, 1970(5): 501-519.

[18] WU J, TAN S Y, TAN Y J, et al. Analysis of Invulnerability in Complex Networks Based on Natural Connectivity[J]. Complex Systems and Complexity Science, 2014(11): 77-86.

YANG Lin, born in 1995, postgraduate, is a member of China Computer Federation. His main research interests include cyberspace security, network security situational awareness and artificial intelligence.

WANG Yong-jie, born in 1974, Ph.D, associate professor. His main research interests include cyberspace security, risk assessment and information system modeling and simulation.