使用自然启发式元启发算法进行物联网应用中的高效雾节点部署

Abdenacer Naouri、Nabil Abdelkader Nouri、Sahraoui Dhelim、Amar Khelloufi、Abdelkarim Ben Sada 和 Huansheng Ning

arXiv:2302.05948v1 [cs.NI] 12 Feb 2023

摘要——**管理从边缘到云端的数据爆炸式增长，需要采用智能化监管，例如雾节点部署，这是评估网络可操作性的一项关键任务。为确保网络可操作性，部署过程必须在两个主要因素方面有效开展：连通性和覆盖范围。网络连通性基于雾节点部署，该部署决定网络的物理拓扑；而覆盖范围则决定网络可访问性。二者都会对网络性能产生显著影响，并保障网络服务质量。确定一种能够降低成本、减少计算和通信开销，同时提供高水平网络连接覆盖率的最优雾节点部署方法极其困难。因此，在保持网络连通性的同时最大化覆盖范围，是一个并非易于解决的优化问题。本文提出了一种雾节点部署算法，能够有效连接雾节点并覆盖所有边缘设备。首先，我们将雾节点部署建模为具有大规模搜索空间的多目标优化问题实例。随后，利用海洋捕食者算法（MPA）解决部署问题，并证明与其他基于种群的算法相比，MPA具有更快的收敛速度和更低的计算复杂度，因此非常适用于雾节点部署。最后，我们在由不同雾场景配置生成的多种基准测试实例上评估所提出的算法。实验结果表明，与用于确定雾节点最优部署的当前先进方法相比，所提出的算法能够取得非常有前景的结果。**

I. 引言

随着物联网（IoT）近期的发展，我们见证了医疗保健、监控和智慧城市等不同领域中许多新应用的出现。物联网系统可以采用不同的计算与通信架构来实现，各种架构能够根据可用资源提供数据处理、存储和分析方案 [1]。雾计算和边缘计算被视为物联网系统不可或缺的组成部分，旨在解决云的局限性问题。这些问题源于大量物联网传感器卸载的请求，例如延迟响应时间过长，可能需要很长时间才能响应用户的请求 [2]。雾计算和边缘计算被视为云计算的扩展，可在边缘网络中靠近物联网传感器的位置提供分布式计算、存储和网络功能，以满足物联网应用的需求，并支持低延迟、实时响应、位置感知、设备移动性、可扩展性、异构性、瞬态存储、高速数据传播、去中心化计算和安全性 [3]。物联网传感器和边缘设备产生的大量请求可能导致网络中的数据激增，因此需要高效的网络部署。在大多数自组织网络中，计算节点的部署采用预先规划方式或自组织方式。预先规划部署方式适用于部署区域访问受限且部署成本不高的情况；而当部署区域广阔、成本高昂且包含大量计算节点时，则采用自组织方式 [4]。

部署策略基于多项因素制定，例如物联网传感器和边缘设备的位置、系统功能以及环境。此外，部署策略可能因区域性质不同而异：在开放区域，部署的目标是覆盖比封闭区域（如建筑物内部）更广的范围。为满足用户覆盖率和连通性需求，部署方式有很多种，例如可用于自然灾害、石油钻井平台、矿井、战场监视、高速移动视频游戏或公共交通等关键场景 [5]。在广阔区域内寻找最优部署方案是一项挑战，因为与小区域相比，这类区域可能面临用户密度较高且部署区域不受限制的问题。这是因为计算节点的部署可能会影响整个系统的性能，因此被视为一项关键问题 [6]。因此，在本研究中，我们专注于在边缘网络内实现雾节点的最优部署，以服务特定区域中的边界边缘设备和物联网传感器。

为了有效部署雾计算节点，我们需要充分了解网络拓扑与节点密度之间的关系，并考虑位置、密度等不同因素 [7]。因此，应解决一些问题以提升网络性能，包括连通性、覆盖范围、可靠性、可访问性等。如果安装雾节点时没有考虑感兴趣区域的实际限制因素和底层网络拓扑，可能会导致网络覆盖范围和连通性较差。

Abdenacer Naouri、Huansheng Ning、Amar Khelloufi 和 Abdelkarim Ben Sada 任职于中国北京的北京科技大学（邮编：100083）。Sahraoui Dhelim 任职于爱尔兰都柏林大学计算机科学学院。Nabil Abdelkader Nouri 任职于阿尔及利亚杰勒法的杰勒法大学数学与计算机科学系。通讯作者：SahraouiDhelim（sahraoui.dhelim@ucd.ie）。

节点部署问题已在不同网络和不同环境中得到研究。事实证明，这是一个极具挑战性的问题，属于 NP 难问题 [8]。为了解决这些问题，人们开发了元启发式技术，但这些技术通常只能提供局部最优解 [9]。

与其他方法相比，元启发式方法在解决各种现实世界的优化问题方面取得了最大的成功和发展。本研究考虑的计算场景由位于网络边缘的物联网传感器和边缘设备，以及位于雾层内的雾节点组成。由于该问题属于NP难问题，我们提出一种基于种群的 MPO 元启发式算法，以高效地解决边缘网络设备中的雾节点部署问题，并评估网络覆盖和连通性对网络性能的影响。本研究的贡献总结如下：

- 为了有效连接雾节点并覆盖所有边缘节点，我们提出一种多目标聚合目标函数，旨在同时最大化边缘设备覆盖范围和雾节点连通性，确保所有设备和雾节点均实现连接和覆盖。这将有效提升网络性能并保障网络服务质量。
- 我们提出一种受自然启发的 MPA 元启发式优化算法，并将目标函数设定为实现网络覆盖范围和连通性的最大化。我们分析各种参数对网络性能的影响，并确定最有效的优化方法。
- 为实现无缝计算，我们在各种网络环境下实现并测试了所提出的算法。结果表明，与其他方法相比，该算法在覆盖范围和连通性方面均有显著提升。

本文其余部分组织如下：第 II节回顾了雾节点部署的现有研究。第III节介绍系统建模和问题构建。第IV节详细介绍了用于雾节点部署的元启发式算法。第 V节展示评估详情和实验结果。第 VI节总结全文，并概述未来的研究方向。

II. 相关工作

如今，网络需要在网络架构中精确部署计算节点，以实现可靠测量和高效数据传输。为此，节点部署流程常被视为最关键的方法之一。尽管已有多位研究人员致力于解决这一问题，但由于现有方案存在局限性，仍需定期改进或以新方案取代。为此，可以采用优化算法，寻找符合指定准则的最优节点位置。

启发式方法和元启发式方法已被用于解决雾节点部署问题（FND），其目标各不相同。 [10] 中的作者最初研究了无线传感器网络中的覆盖问题，并证明该问题是 NP 难问题。他们专注于在传感器数量给定的情况下，提升计算节点的覆盖范围。 [11]中的作者提出了一种改进的人工蜂群算法，通过优化中继节点部署来提升双层无线传感器网络的寿命。该方法首先将问题维度整合到候选解发现公式中，并根据问题适配性和迭代次数调整局部搜索，从而帮助平衡算法的探索与开发能力。 [12], 中的作者提出了一种粒子群优化（PSO）算法，用于优化节点网络部署并提升网络的自适应能力，但该算法存在易陷入局部最优的缺点。 [13],中的作者假设网络拓扑呈六边形形式，并提出一种改进的虚拟弹簧力算法（VSFA），用于部署网络节点。这一方法实质上缩小了脆弱区域，但所提策略尚未在复杂环境中进行评估，仅在理想条件下经过测试。与 [13], 中的工作类似， [14] 中的作者在理想部署环境下进行评估，并假设节点同质；他们采用改进的鱼群算法（AIFS），以覆盖率为目标优化节点部署，从而大幅扩大网络覆盖面积并降低能耗。 [15] [16] 中的作者专注于提升能源利用率、能量均衡程度和网络覆盖；而 [15]中的作者则以线性加权方式将三个优化目标合并为单一目标，再使用鲸群算法（WGA）优化该单一目标函数。尽管这种方法简单，但计算耗时过长。 [16] 中的作者仅考虑同质节点的部署以及障碍物的存在，并采用两种多目标进化算法：一种基于分解，另一种用于联合优化模型。

[17] 的作者研究了无线传感器网络中的容错和连通性问题。他们将重点放在连通性恢复上，并提出了一种简单、快速的基于移动的 k 连通性重构算法。该算法根据节点失效是否会导致 k 值降低，将节点划分为关键簇和非关键簇。当关键节点失效时，该算法会选取并移动非关键节点。 [18]的作者提出了用于估算物联网无线传感器网络连通性的方法。他们提出了一种简单而高效的基于移动的 k 连通性恢复方法（PINC），将节点划分为关键组和非关键组。当关键节点失效时，PINC 算法会接管并移动非关键节点。该方法会将移动成本最低的非关键节点移至失效传感器节点所在的位置。

与以往将节点部署在离散网格区域中的研究不同，这种做法限制了节点的位置。我们

的方法使我们能够灵活地在连续的感兴趣区域内部署雾节点。此外，以往的方法以分层方式优化网络连通性和用户覆盖率，而在我们的研究中，我们采用多目标聚合函数，同时优化这两项指标，因为在某些实际服务中，用户覆盖率也可能至关重要。我们将提供一些实验结果，以展示同步优化的优势。我们的方法使我们能够灵活地在连续的感兴趣区域内部署雾节点。此外，诸如 [19, 20] 的方法以分层方式优化网络连通性和用户覆盖率，而在我们的研究中，我们采用多目标聚合函数，同时优化这两项指标，因为在某些实际服务中，用户覆盖率也可能至关重要。

III. 系统假设与问题模型构建

本节介绍物联网-雾-云基础设施中的雾节点部署模型。所研究的系统包括物联网传感器、设备和雾节点，如图1所示。为了满足终端物联网边缘节点对高吞吐量和高服务质量的要求，同时保持较低成本，图1展示了基于物联网边缘的雾基础设施。该图呈现了模型的核心概念，包括系统架构、覆盖范围和网络连通性，以及多目标雾节点部署所面临的挑战。

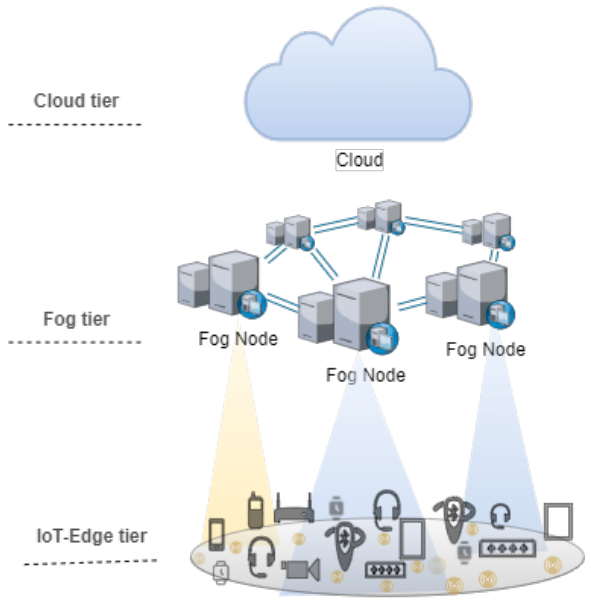


图1：基于物联网边缘的雾基础设施

A. 网络模型描述

图1 展示了基于物联网边缘的雾基础设施，该基础设施包含三个层级。第一层是物联网边缘层，

表I：模型构建中使用的数学符号

符号	含义
n	雾节点数量
m	物联网边缘客户端数量
l	雾节点位置向量
$l(x_i,\ y_i)$	节点位置
W	感兴趣区域宽度
H	感兴趣区域高度
FG	雾节点集合
EN	物联网边缘节点集合
EN_i	第 i个边缘节点客户端
E	雾节点之间的边缘链路集合
G_i	第i个子图分量
$\|G_i\|$	第 i个子图分量的大小
h	子图分量数量
$\zeta(G)$	网络连通性
$\Phi(G)$	终端物联网边缘节点覆盖
γ_i	布尔覆盖函数
X_i	第 i 个粒子
$gBest$	全局最优解
Max_{Itr}	最大步长移动

其中包括许多物联网边缘终端节点。这些节点连接到第二层，即雾层，其中包含许多雾服务器节点。第三层是云层。计算节点可以对其获取的数据进行简单分析。设基础设施中的计算节点集合表示为$U = FG \cup EN$，其中雾层由 n 个雾节点组成，表示为 $FG = \{FG_1, FG_2, FG_3..., FG_n\}$，相应的传输范围集合表示为$R(FG_i) = \{R_1, R_2, R_3..., R_n\}$，而物联网边缘层具有$m$ 个边缘节点，表示为 $EN = \{E_1, E_2, E_3..., E_m\}$。为便于问题定义，系统模型中使用的标记和符号列于表I中。

B. 系统模型假设

我们研究一种部署场景，其中雾节点和物联网边缘节点均被视为静态节点。为了响应实际网络中的部署场景，每个雾节点的通信半径长度各不相同，并且雾节点可以通过其无线电覆盖范围彼此通信 [21]。另一方面，边缘节点仅具备网络连通性所必需的功能，但不具备网关或桥接器的功能 [22]。因此，边缘节点必须通过雾节点与其他节点进行通信。此外，由于雾节点的位置是根据边缘节点的位置确定的，因此有必要预先了解边缘节点的位置 [23]。简而言之，为确保网络连通性和覆盖范围，我们制定了以下假设：

- 雾节点和物联网边缘节点均匀且随机地部署在整个网络中，并被视为在感兴趣区域内静止不动。

- 感兴趣区域内的每个雾节点都可通过蜂窝网络访问云中心；终端节点的位置被视为固定不变。
- 为适应实际中终端节点的异构性，假设每个雾节点 fg_i 具有不同的传输范围 R_{fg_i}。
- 一个雾节点只能连接一定数量的边缘节点。

注意，欧氏距离是确定网络可访问性和连通性的重要度量。因此，必须满足以下两个条件：

1) 物联网边缘节点 EN_i 当且仅当其被至少一个雾节点覆盖时，才被视为已连接 fg_i，如公式（1所示：

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \le R_i \tag{1}$$

.

2) 两个雾节点 fg_i和 fg_j当且仅当处于同一传输范围内时，才被视为已连接，如公式（2所示：

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \le \min(R_i, R_j) \tag{2}$$

图2 展示了所提出问题的一个实例。在该示例中，节点由两种类型组成：雾节点和物联网边缘节点，其中包括 10 个雾节点 n = 10 和 80 个物联网-边缘设备 m = 80，它们位于部署区域内，每个雾节点具有不同的通信范围。如果两个雾节点处于彼此的通信范围内，则它们将通过深色连线相连（例如，参见雾节点 1 和 2 之间的深色连线）。如果一个物联网边缘节点位于某个雾节点的通信范围内，则它将通过一条细黑线连接到距离最近的雾节点。此外，拓扑图包含三个子图分量，其中最大的分量大小为 5（即 $\zeta(G) = 5$），并且覆盖了 68 个物联网-边缘设备（即 $\Phi(G) == 68$）。如果将某些雾节点向物联网-边缘设备最拥挤的区域移动，如图 2a所示，则实际上几乎所有子图都将合并为一个大小为 10 的单一巨型图（即 $\Phi(G) = 10$），并且将覆盖 76 个物联网-边缘设备（即 $\zeta(G) = 76$）。因此，通过改变部分雾节点的位置，可以同时改善边缘覆盖范围和网络连通性这两个度量指标。

C. 问题构建

我们着重于确定近乎最优的雾节点部署方案：在感兴趣区域内的适当位置放置雾节点，同时确保连通性并最大化覆盖范围。网络覆盖是指被覆盖的边缘节点数量，而连通性是指网络中的雾节点彼此连通。为解决这一问题，分析整个网络并不实际；由于网络不连通性，我们很难甚至无法构建一个覆盖所有边缘节点的连通网络，因此可以将网络不连通性单独处理。因此，在我们的研究中，我们以最大的子网络（即连通程度最高的子网络）为目标。假设 L 是雾节点的初始给定位置集合 $L = \{L(x_1, y_1), L(x_2, y_2), \ldots\ldots, L(x_n, y_n)\}$。

我们的目标是更新雾节点的位置，使所覆盖的边缘节点数量和最大连通子网络的规模达到最大。我们注意到，这两个目标彼此存在冲突，这意味着较大的子组件网络并不一定代表边缘节点具有较广的覆盖范围。我们针对由部署在二维区域中的 n 个雾节点和 m 个边缘设备组成的网络，分析这一场景问题。我们表示一个无向拓扑图$G = (V, E)$，其中 $V = FG \cup EN$。V指代网络节点，包括一组记为 FG 的雾节点和一组记为 EN 的边缘节点，而 E 定义了边缘节点的连通关系。如果存在一条边（fg_i, fg_j）$\in E$且 $R_{fg_i} \cap R_{fg_j} \neq \emptyset$，则认为两个雾节点相互连通；当且仅当雾节点 $fg_i \in F$、$L_{EN_j} \in R_j$，并且存在边缘节点与雾节点之间的边 $edge(en_j, fg_i) \in E$时，才认为边缘节点 $EN_j \in E$ 被覆盖。

如前所述，由于网络不连通性，通过分析整个网络来寻找一个完全连通且覆盖完整的图颇具挑战。因此，针对一个较大的子网络或许可以提升网络连通性。我们指出，目标网络对应的图 G 可能并不连通，即 G 可能由多个子图组件构成。不过需要注意的是，最大化雾节点的网络连通性未必能够覆盖所有边缘节点。在这种情况下，我们旨在尽可能扩大最大子图组件的规模，从而最大化网络连通性。

假设图 G 包含 h 个子图 $G_1, \ldots\ldots, G_h$，位于 G 中，即 $G = G_1 \cup G_2 \cup G_h$ 和 $G_i \cap G_j = \emptyset$；对于 $i, j \in 1, \ldots\ldots, h$。为了分析和评估拓扑图 G 的性能，以下网络覆盖和连通性指标被视为优化目标。

网络连通性定义为最大连通雾节点子图的大小。其中，G 是一个图，G_h是 G 的子集。网络连通性计算如下：

$$\zeta(G) = \max_{i \in \{1,\ldots,h\}} |G_i| \tag{3}$$

相对于雾节点，边缘节点 i 的网络覆盖函数 γ_i 由如下二进制值定义：

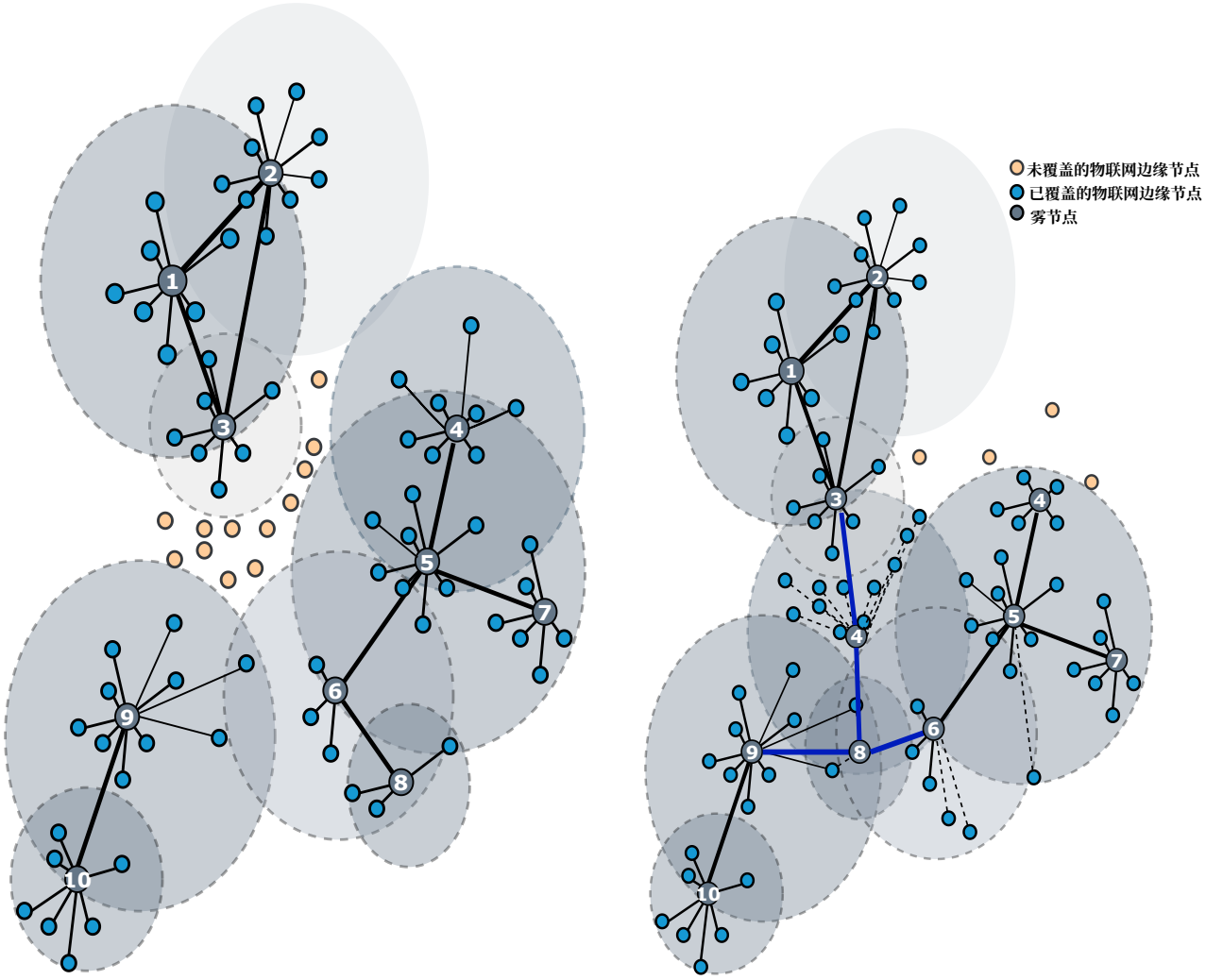
$$\gamma_i = \begin{cases} 1 & \text{if the client } i \text{ is covered by at least one fog node} \\ 0 & \text{Otherwise} \end{cases}$$

其中

$$\Phi(G) = \sum_{i=0}^{m} \gamma_i \tag{4}$$

D. 目标函数

为评估所提出拓扑图的性能，我们考虑优化两个目标：网络



（a）碎片化网络拓扑　　（b）近优网络拓扑

图 2：雾节点连通性和物联网边缘节点覆盖范围

连通性，即通过最大化最大子图分量的规模 $\zeta(G)$ 以及节点边覆盖率 $\Phi(G)$，其定义分别见公式 3 和 4。因此，我们采用加权和法，将每个目标乘以用户提供的权重后求和，从而将多目标问题转化为标量问题。聚合后的适应度函数 $f(X)$ 定义如下：

$$f(X) = \omega . \frac{\zeta(G)}{n} + (1 - \omega) . \frac{\Phi(G)}{m} \tag{5}$$

其中，ω 是一个介于零和一之间的加权系数，用于描述各目标的优先级比例。

IV. 海洋捕食者算法

MPA算法 [24] 最近提出，旨在模拟海洋捕食者追逐食物时的行为。捕食者根据捕食者与猎物之间的速度比，在布朗运动与莱维飞行策略之间进行权衡。其主要动机在于海洋捕食者广泛采用的觅食策略，即莱维运动和布朗运动 [25], , 以及捕食者与猎物之间生物相互作用中的最优遭遇率策略。MPA遵循自然调节海洋栖息地中最优觅食策略和遭遇率策略的准则 [26]。根据MPA的定义，基于MPA的优化算法如下所示：

在初始化阶段，一些猎物最初均匀分散在优化过程的搜索空间周围。公式6 展示了其数学模型。

$$X_0 = X_{\min} + \text{rand} \times (X_{\max} - X_{\min}) \tag{6}$$

X_{max} 和 X_{min}分别定义搜索空间的边界，其中 X_{max} 是上界，X_{min} 是下界，rand 是一个在 0 到 1 之间随机生成的变量。

在优化阶段，使用目标函数评估每个分布式解，并将目标值

最佳的解选为顶级捕食者。随后利用该解创建一个称为精英个体（也记作 E）的矩阵，该矩阵基于适者生存的理念。精英矩阵如下所示：

$$\text{Elite} = \begin{bmatrix} X_{1.1}^{1} & X_{1.2}^{1} & \cdots & X_{1.d}^{1} \\ X_{2.1}^{1} & X_{2.2}^{1} & \cdots & X_{2.d}^{1} \\ \vdots & \vdots & \vdots & \vdots \\ X_{n.1}^{1} & X_{n.2}^{1} & \cdots & X_{n.d}^{1} \end{bmatrix}_{\text{nxd}} \tag{7}$$

变量 X1、n 和 d 分别表示顶级捕食者向量（重复 N 次以构造精英矩阵）、种群中的搜索智能体数量以及维度数量。搜索智能体中的捕食者和猎物都通过空间探索来寻找食物。精英矩阵在每次迭代中都会根据较优捕食者进行更新，并与上一轮迭代中的顶级捕食者进行比较。通常，在初始化阶段，当捕食者改变位置时，会构建猎物。因此，定义一个与精英矩阵维度相同的矩阵，称为猎物矩阵。利用原始猎物中适应度最高的个体来构建精英矩阵。该矩阵的公式如下：

$$\text{Prey} = \begin{bmatrix} X_{1.1} & X_{1.2} & \cdots & X_{1.d} \\ X_{2.1} & X_{2.2} & \cdots & X_{2.d} \\ \vdots & \vdots & \vdots & \vdots \\ X_{n.1} & X_{n.2} & \cdots & X_{n.d} \end{bmatrix}_{n \times xd} \tag{8}$$

$X_{i.j}$ 表示 i_{th} 猎物的 j_{th} 维度。前文所述的两个矩阵在整个优化问题中发挥着重要作用。MPA 优化过程基于捕食者与猎物的移动模式，分为以下几个阶段：

1) 高速度比（阶段 *1*）：在此阶段，当探索至关重要时，猎物的移动速度快于捕食者。该阶段通常出现在迭代过程初期，因此被定义为探索阶段；猎物开始采用布朗策略探索搜索空间，以确定可能包含理想解的区域。因此，可将其数学形式表示为：

$$\text{While } t < \frac{1}{3} T_{\max} i = 1, 2, \ldots, n \tag{9}$$

$$\begin{cases} \text{stepsize}_{\ i} = \left(R_B \otimes \left(\text{ Elite}_{\ i} - R_B \otimes \text{ pre}_{\ i}\right)\right) \\ \text{prey}_{\ i} = \text{ prey}_{\ i} + P \times R \otimes \text{ stepsize}_{\ i} \end{cases} \tag{10}$$

其中，t 表示当前迭代次数，T_{max}表示算法的最大循环次数，stepsize 表示步长尺度因子，R_B 表示服从正态分布的布朗游走随机向量，$Elite_i$ 表示由顶级捕食者构成的精英矩阵，$prey_i$ 表示猎物矩阵，$\otimes$ 表示逐元素乘法，R 是取值范围为 [0,1], 的随机变量，而 P 是设定为 0.5 的固定值。

2）相对单位速度（阶段 *2*）：该阶段基于探索与开发 [27]。完成前三分之一的探索过程后，捕食者和猎物以相同速度探索搜索空间，以寻找各自的食物。这是因为可能包含理想解的位置彼此接近，因此该阶段被视为优化的中间阶段。总而言之，在此阶段，探索过程将逐步转变为开发过程。具体而言，捕食者负责探索，而猎物负责开发。为了对这一阶段进行数学建模，种群被分为两部分：第一部分由猎物通过莱维游走进行开发，第二部分由捕食者通过布朗运动进行探索。

$$\text{While } \frac{1}{3} T_{\max} < t < \frac{2}{3} T_{\max} i = 1, 2, \ldots, n/2 \tag{11}$$

$$\begin{cases} \text{stepsize}_{\ i} = \left(R_L \otimes \left(\text{ Elite}_{\ i} - R_L \otimes \text{ prey}_{\ i}\right)\right) \\ \text{prey}_{\ i} = \text{ prey}_{\ i} + P \times R \otimes \text{ stepsize}_{\ i} \end{cases} \tag{12}$$

$$\text{While } \frac{1}{3} T_{\max} < t < \frac{2}{3} T_{\max} i = n/2, \ldots, n \tag{13}$$

$$\begin{cases} \text{step size}_{\ i} = \left(R_B \otimes \left(R_B \otimes \text{ Elite}_{\ i} - \text{ prey}_{\ i}\right)\right) \\ \text{prey}_{\ i} = \text{ Elite}_{\ i} + P \times \text{CF} \otimes \text{ step size}_{\ i} \end{cases} \tag{14}$$

其中，R_L 是促进莱维运动的随机向量，CF 是由方程15确定的自适应变量；它用于控制捕食者的路径。

$$CF = \left(1 - \frac{t}{T_{\max}}\right)^{\frac{2t}{T_{\max}}} \tag{15}$$

3）低速比（阶段*3*）：在优化过程最大迭代次数的最后三分之一阶段，捕食者开始沿着莱维运动接近猎物。该阶段可建模为：

$$\text{While } t > \frac{2}{3} T_{\max}, i = 1, 2, \ldots, n \tag{16}$$

$$\begin{cases} \text{stepsize}_{\ i} = \left(R_L \otimes \left(R_L \otimes \text{ Elite}_{\ i} - \text{ prey}_{\ i}\right)\right) \\ \text{prey}_{\ i} = \text{ Elite}_{\ i} + P \times CF \otimes \text{ stepsize}_{\ i} \end{cases} \tag{17}$$

MPA 模拟捕食者习性以提高逃离局部最优的概率，这种灵活性是其另一项优势 [28]。这种效应是指捕食者的移动会受到环境因素的影响，包括涡流形成和鱼类聚集装置（FAD）。其数学模型如方程18所示：

$$\text{prey}_{\ i} = \begin{cases} \text{if } r \le P_f \\ \text{prey}_{\ i} + CF\left[X_{\min} + R \otimes \left(X_{\max} + X_{\min}\right)\right] \otimes U \\ \text{if } r > P_f \\ \text{prey}_{\ i} + \left[P_f\left(1 - r\right) + r\right] \times \left(\text{ prey}_{\ r1} - \text{ prey}_{\ r2}\right) \end{cases} \tag{18}$$

P_f 表示 FAD 影响优化过程的可能性； U 表示一个包含 0 和 1 的二进制向量； r 表示介于 [0, 1] 之间的一个不同值；而数值 $r1$ 和 $r2$ 从整个种群范围中随机生成，表示猎物矩阵的随机索引。

MPA可以通过保留猎物之前的位置来存储数据。更新现有解后，会比较每个当前解和每个过时解的适应度值；如果前一个解的适应度高于后一个解，就交换两者。所采用的海洋捕食者优化方法（海洋捕食者优化算法）的主要步骤如算法1所示。

V. 实验与结果

本节讨论使用所提出架构所取得的结果。该架构为三层物联网-边缘-雾计算架构：第一层由物联网-边缘设备构成，主要用于从各类物联网-边缘设备采集数据；第二层由雾服务器组成；最后一层则包含资源丰富的云。该架构在明确各层职责的同时，促进了资源的部署与使用。为评估所提出的MPA算法的性能，我们首先介绍实验数据和环境，然后在不同设置下开展多种场景测试，并将结果与不同算法进行比较。

A. 实验设置

在实验设置中，我们使用 Matlab 2019Ra 实现了一项实际应用，以尽可能快速地完成计算。根据算法1中给出的伪代码，我们在一个 1000m × 1000m 的矩形区域内进行了实验，雾节点采用正态/均匀分布。实验使用搭载 8 核 AMD Ryzen 7 5700U（时钟频率为 4.3 GHz、内存为 8GB）的设备，并运行于 Windows 10 环境。该实验设置使我们能够在各种条件和限制下重复实验，从而在受控环境中按照所需设置开展实验。表 II 列出了各项参数及其对应值，这些参数和值是通过多次初步实验推导并确定的。

表II：网络参数

参数	值	初始值
雾节点密度	[10, 120]	45
边缘节点密度	[30, 200]	120
通信范围	[90, 200]	100m
区域宽度	1000m	1000m
区域高度	1000米	1000米

B. 实验结果

我们开展了多项实验，以评估所提出算法相较于以下基准算法在连通性和覆盖范围方面的表现：哈里斯鹰优化算法（HHO） [29],粒子群优化（PSO） [30]和正弦余弦算法（SCA） [31]。由于目标函数涉及两个相互竞争的优先目标，且网络可能出现连通性和覆盖范围不足的情况，如图2a、2b所示，因此我们在不同的网络参数下进行了测试，以确定所定义目标函数的最佳权重 ω ，从而将所提出算法的性能与其他方法进行比较。

算法1： 海洋捕食者优化算法伪代码

输入： 搜索智能体（猎物）种群 X_i（$i = 1, 2, 3..... n$），

输出： 雾节点部署的最优适应度值 部署 fit_{best}

```
当 iter < Max_Itr 时执行
    计算每个对象的适应度值 py_i | i = 0, 1, 2, 3, .....N
    构建精英矩阵 E 并完成内存保存
    如果 iter < Max_Itr/3 则
        根据公式 (2) 更新猎物
    end
    如果 Max_Itr/3 < iter < 2.Max_Itr/3 则
        对于种群的前半部分 (i = 1, . . . , n/2)
        根据式 (3) 更新猎物
        对于另一半种群 (i = n/2, . . . , n)
        根据式 (4) 更新猎物
    end
    如果 iter > 2.Max_Itr/3 则
        根据公式（6）更新猎物
    end
    完成内存保存并更新精英个体
    应用鱼类聚集装置效应并根据公式更新 (7)
    迭代=迭代 +1
结束
```

C. 不同 ω 值下的适应度函数研究

为确定合适的权重系数 ω，我们同时分析不同 ω 值下的用户覆盖率和雾节点连通性，并采用收敛曲线——这是理解算法行为最关键的分析工具——在使用元启发式算法开发最优解的过程中进行分析。因此，为了确定最优权重系数 ω，我们进行了多次迭代，直到满足终止条件

方程中的准则5 。我们将实验执行 10 次，每次进行 1000 次迭代。每次实验均采用不同的ω值。在每次实验中，我们使用 1000 个实例来确定权重系数 ω取值变化的影响。我们观察到，使用较大的ω值时，用户覆盖率较低，这意味着雾节点倾向于聚集在较小的区域内，导致许多物联网边缘节点未被覆盖；而使用较小的值时，用户覆盖率较高，这意味着雾节点分散在整个区域，能够各自覆盖大多数物联网边缘节点，但无法形成较大的连通分量（即高连通性）。因此，我们认为，在一定程度上，权重系数 ω 决定了我们关注的两个目标（ζ 和 Φ）之间的权衡。因此，我们确定了权重系数 ω，使其最有可能同时满足两个相关准则。经过大量测试，图3显示，采用 $\omega = 0.5$ 所得到的结果实现了出色的权衡。

图4 展示了所研究系统在不同网络配置值下取得的结果。正如实验开始时所观察到的，所有算法都经历了显著变化；然而，随着迭代的推进，这种变化逐渐减小。另一方面，HHO、PSO 和 SCA 算法容易早熟收敛，因此比起全局最优解，它们更容易陷入局部最优。算法整体性能的一项关键优势在于其进展速度。如图4所示，经过一定次数的迭代后，所提出的算法展现出出色的潜力，收敛速度快于其他算法，同时也达到了最优适应度函数值。因此，结果表明，在不同的聚合值下，所提出的算法在解决雾节点部署问题方面优于其他算法 ω。

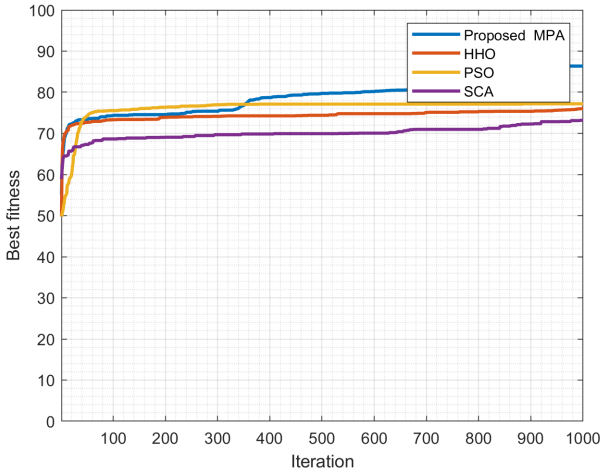


图 4：所研究算法的收敛曲线

1）算法收敛情况比较：为了评估所提出算法相较于 PSO、HHO 和 SCA 算法在所提出的目标函数 f （如公式5所示），对网络覆盖范围和用户覆盖率进行了考察。图5 展示了所提出的算法如何以更少的雾节点覆盖更多物联网边缘节点；而图5a 显示，40% 的已连接雾节点可覆盖超过 90% 的目标种群，如图5b所示。相比之下，PSO、HHO 和 SCA 分别仅连接了 (45,45,41)% 的雾服务器，覆盖了 (67,67,79)% 的终端边缘设备。因此，结果表明，我们提出的 MPA 算法优于其他方法。

D. 评估准则

为了更好地了解针对优化目标所做的调整，我们根据以下评估指标，将所提出的海洋捕食者优化算法与先前的方法进行了性能评估，即网络连通性（Φ）、用户覆盖率（ζ）和目标函数值 f。我们通过改变边缘节点数量、雾节点数量以及雾节点的传输范围来评估实验。

1）网络密度的影响：为了分析网络覆盖和连通性，我们通过部署不同规模（N）的雾网络进行实验，其中雾节点数量从 10 个变化至 120 个，如图6、7 和8所示。如图6a所示，随着更多雾节点加入网络，已连接的雾节点数量也随之增加，这反映了网络连通性。正如预期，增加雾节点数量会提高网络连通性。这是因为向网络中添加雾节点后，孤立网络便会与网络的其余部分相连。因此，网络覆盖范围会直接扩大。至于覆盖范围，图6b展示了雾节点数量与被覆盖的物联网边缘节点总数之间的关系。结果表明，我们提出的算法能够实现最优覆盖，正如预期，覆盖的物联网边缘节点数量随之增加。

2）边缘节点密度的影响：图7 展示了将物联网边缘节点总数从 30 个变化至 200 个时的网络覆盖和连通性结果。从图7a可以清楚看出，已连接的雾节点比例始终在 70% 到 90% 之间。这是因为我们的算法会尽力连接所有可连接的雾节点，以覆盖更多边缘节点。此外，图7b显示了网络的整体覆盖率，其保持在 70% 到 93% 之间。我们观察到，随着物联网边缘节点数量增加，覆盖率和连通性比例始终保持稳定。这是因为物联网边缘节点均匀分布，因此物联网节点落入已覆盖区域或未覆盖区域的概率相同。

3）通信范围的影响：图8展示了通信范围调整对多个网络覆盖范围和连通性性能的影响

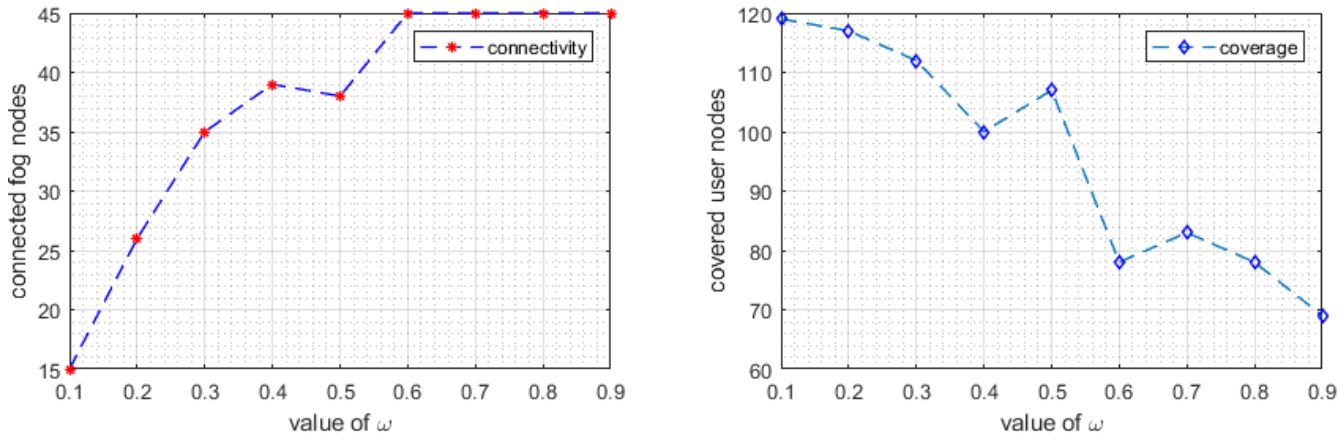
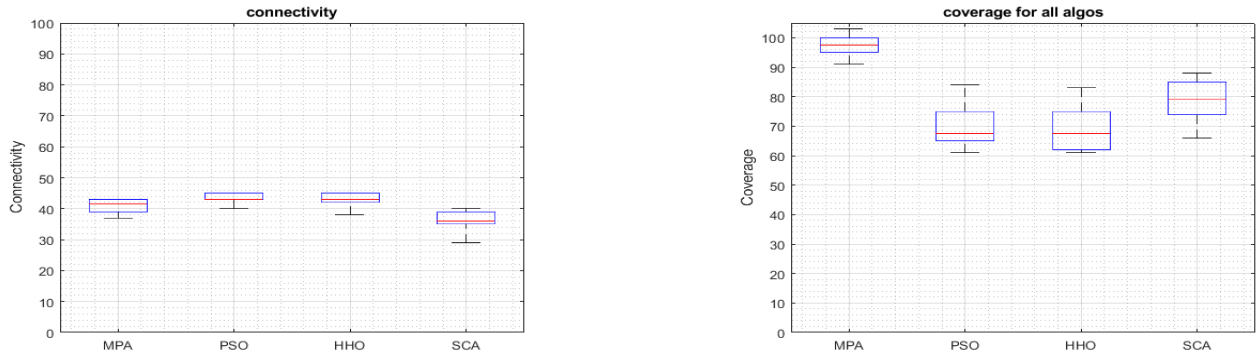
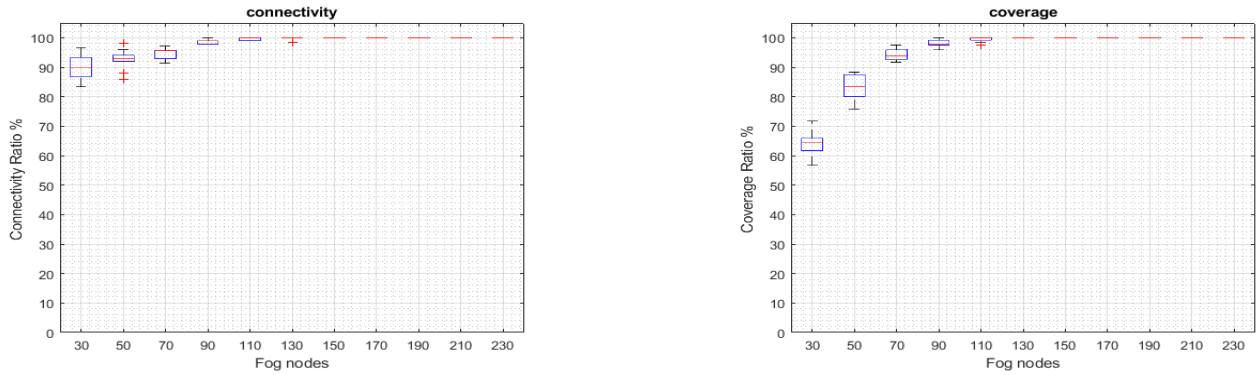


图 3：不同 ω 值下的适应度函数研究



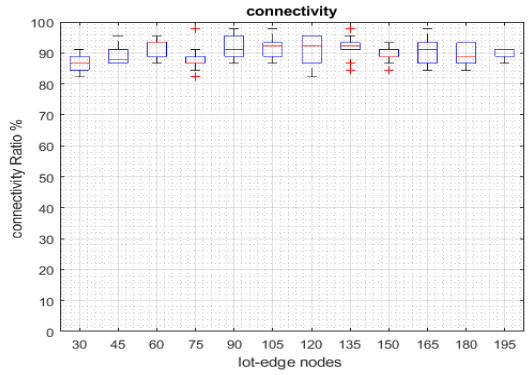
(a) 网络连通性　　(b) 物联网边缘节点覆盖率

图 5：MPA、PSO、HHO 和 SCA 的比较

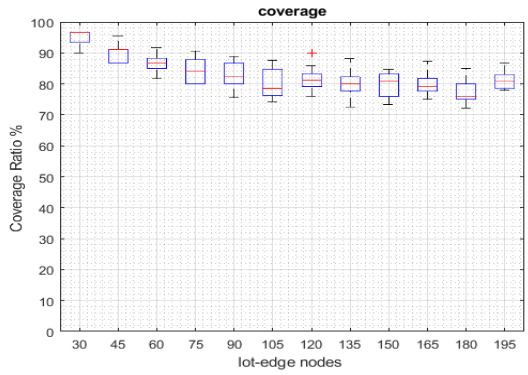


(a) 网络连通性　　(b) 物联网边缘节点覆盖率

图6：雾节点密度对覆盖范围和连通性的影响

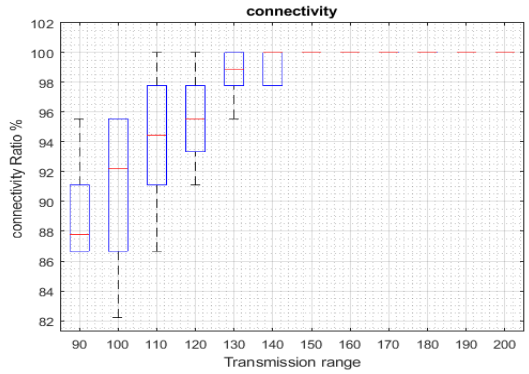


(a) 网络连通性

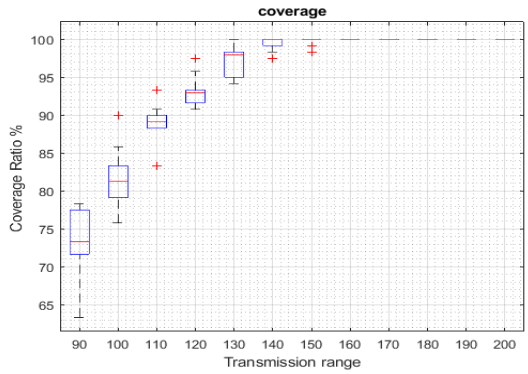


(b) 物联网边缘节点覆盖率

图7：物联网边缘节点密度对覆盖范围和连通性的影响



(a) 网络连通性



(b) 物联网边缘节点覆盖率

图8：传输范围对覆盖范围和连通性的影响

实验运行。通过将通信范围分别从 90 米逐步调整到 200 米。图8a的结果表明，当通信范围超过 140 米时，所有雾节点都开始尝试彼此通信。这是因为几乎孤立的子网络与网络的其余部分连接起来，形成一个单一的巨型网络。图8b 展示了通信范围变化对物联网边缘节点覆盖率的影响。结果证明，覆盖的边缘节点数量随传输范围增大而成比例增加。根据图8b，当通信范围超过 140 米时，网络中几乎所有终端节点都被覆盖。因此，140 米被视为一个关键通信范围。

4）实验目标函数影响：本部分展示了目标函数5中提出的用户覆盖率和网络连通性对所提算法性能的影响。采用与之前相同的实验参数，表III 和表IV 列出了所提算法得到的目标函数的精确数值结果。表III中的结果表明，当雾节点数量从 30 增加到 230 时，覆盖的物联网边缘节点百分比和目标函数值也相应增加。此外，近

表III：覆盖范围、连通性和适应度值与雾节点密度的关系

Fog	连通性%)	覆盖范围 (%)	函数
30	90,00	64,25	77,13
50	92,60	83,50	88,05
70	94,86	94,00	94,43
90	98,67	98,00	98,33
110	99,73	99,50	99,61
130	99,85	100,00	99,92
150	100,00	100,00	100,00
170	100,00	100,00	100,00
190	100,00	100,00	100,00
210	100,00	100,00	100,00
230	100,00	100,00	100,00

我们始终获得了连通的网络拓扑。然而，随着向网络中添加更多雾节点，我们提出的算法能够实现更高的物联网边缘节点覆盖率和更优的适应度值 f。根据表V中提供的统计数据，当雾节点数量从 30 个增加到 230 个时，边缘节点覆盖率和目标函数值都成比例增加

表 IV：覆盖范围、连通性和适应度值与物联网边缘节点密度的关系

物联网边缘	连通性(%)	覆盖范围 (%)	函数
30	86,7	95,0	90,83
45	89,1	90,4	89,78
60	91,8	86,8	89,31
75	88,0	84,5	86,27
90	91,8	82,7	87,22
105	92,2	80,3	86,25
120	90,9	81,8	86,36
135	91,8	80,4	86,07
150	89,6	79,8	84,68
165	90,9	79,7	85,29
180	89,3	77,2	83,25
195	90,0	81,1	85,54

表 V：覆盖范围、连通性和适应度值与通信范围的关系

范围	连通性 %)	覆盖范围 (%)	函数
90	88,89	73,42	79,61
100	90,89	81,83	85,46
110	94,00	88,50	90,70
120	95,56	93,17	94,12
130	98,44	97,00	97,58
140	99,33	99,50	99,43
150	100,00	99,75	99,85
160	100,00	100,00	100,00
170	100,00	100,00	100,00
180	100,00	100,00	100,00
190	100,00	100,00	100,00
200	100,00	100,00	100,00

节点数量和目标函数值都成比例增加。值得注意的是，我们的网络结构几乎始终保持连通。然而，获得的结果证实，每当向网络中添加更多雾节点时，所提出的方法都能提供更好的网络覆盖和更优的适应度函数值。

将终端节点数量从 30 个增加至 195 个，表IV中的结果表明，用户覆盖率、网络连通性以及目标函数值 f 几乎保持不变。这是因为节点部署分布受到完全随机因素的影响。如图所示，对于所有物联网边缘节点值，整个网络几乎都保持连通。值得注意的是，即使增加终端节点，覆盖范围仍将保持稳定，因此所有边缘节点所占比例也将保持不变。同样，统计分析表明，所提出的算法优于其他算法，并实现了更好的覆盖效果。表V中的样本说明了雾节点的通信范围如何影响覆盖范围和网络连通性。简而言之，通信范围的增加会扩大覆盖范围，进而提高适应度值。此外，使用所提出的算法能够获得的适应度值高于使用其他方法所能获得的值。

VI. 结论

为提升网络性能并在网络边缘提供高水平的用户服务质量，鉴于物联网边缘设备的资源限制以及用户需求，必须高效地部署数据与资源。因此，本文提出了一种三层物联网边缘—雾—云集成架构，并采用元启发式算法实现无缝的资源部署流程。结果表明，优化资源部署能够提升网络性能。我们提出了一个聚合双目标函数，同时最大化网络连通性（以已连接雾节点的数量衡量）和覆盖的物联网边缘节点数量，并提出了一种有效的优化算法。结果表明，与当前最先进的基准方法相比，所提出的算法取得了颇具前景的结果，且收敛速度快、计算复杂度低。此外，实验结果证明了该算法在查找网络连接方面的有效性。算法始终能够检测到完整的网络连接。不过，网络覆盖范围会受到运行区域内物联网边缘节点的数量和分布的显著影响。

致谢

本研究得到了国家自然科学基金委员会（项目编号：61872038）的资助。

参考文献

[1] S. Cui、F. Farha、H. Ning、Z. Zhou、F. Shi 和 M. Daneshmand，“关于物联网应用爆发与用户需求之间瓶颈的综述”，*IEEE*物联网学报，第 9 卷，第 1 期，第 261–273 页，2021年。[2] S. Dhelim、H. Ning、F. Farha、L. Chen、L. Atzori 和 M. Daneshmand，“物联网赋能的社会关系与人工社会智能相遇”，*IEEE*物联网学报，第 8 卷，17 828，2021年 24，17 817 期，第 – 页。[3] A. Naouri、H. Wu、N. A. Nouri、S. Dhelim 和 H. Ning，“通过优化任务卸载实现移动边缘计算的新框架”，*IEEE*物联网学报，第 8 卷，13 076，2021年 16，13 065 期，第 – 页。[4] S. Dhelim、N. Aung、M. T. Kechadi、H. Ning、L. Chen 和 A. Lakas，“Trust2vec：基于有符号网络嵌入的大规模物联网信任管理系统”，*IEEE*物联网学报，第 10 卷，第 1 期，第 553–562 页，2022年。[5] A. Khelloufi、H. Ning、S. Dhelim、T. Qiu、J. Ma、R. Huang 和 L. Atzori，“基于社会关系的社交物联网设备服务推荐系统”，*IEEE*物联网学报，第 8 卷，第 3 期，第 1859–1870 页，2020年。[6] J. Wang、D. Li 和 Y. Hu，“智能工厂中基于时空特征的雾节点部署”，*IEEE*工业信息学汇刊，第 17 卷，第 5 期，第 3534–3543 页，2020年。[7] P. Maiti、H. K. Apat、B. Sahoo 和 A. K. Turuk，“面向物联网服务的时延感知型雾智能网关部署的有效方法”，《物联网》，第 8 卷，文章编号 100091，2019年。[8] D. Kumar、G. Baranwal、Y. Shankar 和 D. P. Vidyarthi，“新兴边缘技术中计算卸载与服务部署的自然启发技术综述”，《万维网》，第 25 卷，第 5 期，第 2049–2107 页，2022年。

[9] B. Natesha 和 R. M. R. Guddeti，“面向两级雾计算架构的基于元启发式的混合服务放置策略”，《网络与系统管理杂志》，第 30 卷，47，2022年3，第 页。[10] Y. Yoon 和 Y.-H. Kim，“一种用于无线传感器网络最大覆盖部署的高效遗传算法”，《IEEE控制论汇刊》，第 43 卷，第 1473–1483 页，2013年。[11] W. Yu、X. Li、Z. Zeng 和 M. Luo，“问题特征与动态搜索平衡型人工蜂群算法：通过中继节点部署优化双层无线传感器网络寿命”，《传感器》，第 22 卷，第 22 期，2022年。 [在线]。可访问：https://www.mdpi.com/ 1424 8220 22 22 8916 - / / / [12] C. Cong，“ 一种基于改进粒子群优化算法的无线传感器网络覆盖算法 ”，载于《2015年智能交通、大数据与智慧城市国际会议》，2015年，第 12–15 页。[13] X. Deng、Z. Yu、R. Tang、X. Qian、K. Yuan 和 S. Liu，“一种面向无线传感器网络应用的基于虚拟弹簧力算法的优化节点部署方案”，《传感器》，第 19 卷，第 8 期，2019年。 [ ] 1424 8220 19 8 1817 在线。可访问：https://www.mdpi.com/ - / / / [14] N. Qin 和 J. Xu，“无线传感器网络中基于自适应鱼群的移动覆盖”，《无线通信与移动计算》，第 2018 卷，第 1–8 页，2018年 11。[15] L. Wang、W. Wu、J. Qi 和 Z. pu Jia，“基于鲸群算法的无线传感器网络覆盖优化”，《计算机科学与信息系统》，第 15 卷，第 569–583 页，2018年。[16] Y. Xu、O. Ding、R. Qu 和 K. Li，“基于分解的混合多目标进化算法用于无线传感器网络覆盖优化”，《应用软计算》，第 68 卷，第 268–282 页，2018年。 [在线]。可访问：https://www.sciencedirect.com/science/article/pii/S1568494618301868 [17] Z. Akusta Dagdeviren 和 Akram，“物联网赋能的无线传感器网络连通性估计方法”，《物联网新兴趋势及其与数据科学、云计算和大数据分析的集成》，第 22 卷，2022年。[18] V. Khalilpour Akram、Z. Akusta Dagdeviren、O. Dagdeviren 和 M. Challenger，“PINC：基于拾取非关键节点的无线传感器网络 k-连通性恢复”，《传感器》，第 21， 19, 卷。在线。2021 [ ] 1424 8220 21 19 可访问：https://www.mdpi.com/ - / / / 6418[19] F. Zeng、Y. Ren、X. Deng 和 W. Li，“无线城域网中的经济高效边缘服务器部署”，《传感器》，第 19 卷，[27] K. Hussain 、 M. N. M. Salleh 、 S. Cheng 和 Y. Shi，“ 论热门基于群体的元启发式算法中的探索与开发”，《神经计算与应用》，第 31 卷，第 7665–7683 页，2019年。

第1号，2019年。 [在线]。可访问：https://www.mdpi.com/1424-8220/ 19/1/32[20] ——，“无线城域网中的经济高效边缘服务器部署”，《传感器》（瑞士巴塞尔），第19卷，2018年。[21] M. Peng、T. Q. Quek、G. Mao、Z. Ding和C. Wang，“人工智能驱动的雾无线接入网络：最新进展与未来趋势”，IEEE无线通信，第27卷，第2期，第12–13页，2020年。[22] A. Ghosh、A. Mukherjee和S. Misra，“Sega：面向基于工业物联网的机器监控的安全边缘网关微服务架构”，IEEE工业信息学汇刊，第18卷，1956，2021年3月，1949，第—页。[23] G. M. Gilbert、N. Shililiandumi和H. Kimaro，“低压配电网络中的雾节点部署进化方法”，《国际智能电网期刊》（ijSmartGrid），第5卷，14，2021年1月，第1，第—页。[24] A. Faramarzi、M. Heidarinejad、S. Mirjalili和A. H. Gandomi，“海洋捕食者算法：一种自然启发式元启发算法”，《专家系统与应用》，第152卷，文章113377，2020年。[在线]。可访问：https://www.sciencedirect.com/science/article/ pii/S 0957417420302025 [25] N. Humphries、N. Queiroz、J. Dyer、N. Pade、M. Musyl、K. Schaefer、D. Fuller、J. Brunnschweiler、T. Doyle、J. Houghton、G. Hays、C. Jones、L. Noble、V. Wearmouth、E. Southall和D. Sims，“环境背景解释了海洋捕食者的莱维运动模式和布朗运动模式”，《自然》，第465卷，第1066–9页，2010年6月。[26] F. Bartumeus、J. Catalan、U. Fulco、M. Lyra和G. Viswanathan，“优化生物相互作用中的遭遇率：莱维策略与布朗策略”，《物理评论快报》，第88卷，097901，2002年4月，第页。[28] K. Zhong、G. Zhou、W. Deng、Y. Zhou和Q. Luo，“多目标海洋捕食者算法”，《计算机方法在应用力学与工程的应用》，第385卷，文章114029，2021年。[29] A. A. Heidari、S. Mirjalili、H. Faris、I. Aljarah、M. Mafarja和H. Chen，“哈里斯鹰优化算法：算法与应用”，《未来一代计算机系统》，第97卷，第849–872页，2019年。[在线]。可访问：https://www.sciencedirect.com/science/article/pii/S0167739X18313530[30] M. R. Bonyadi和Z. Michalewicz，“用于单目标连续空间问题的粒子群优化：综述”，《进化计算》，第25卷，第1期，第1–54页，2017年。[31] S. Mirjalili，“SCA：用于求解优化问题的正弦余弦算法”，《基于知识的系统》，第96卷，第120–133页，2016年。[在线]。可访问：https://www.sciencedirect.com/science/article/pii/S0950705115005043