

2020 年第十三届“认证杯”数学中国 数学建模网络挑战赛第一阶段论文

题目: 基于分布式无线广播网络的整体通信效率
关键词: 通信冲突; 整体通信效率; 回归模型; 动态模拟;
仿真

摘 要

目前,网络技术得到迅速发展,但在无线网络通信系统的分布式环境下节点之间信息的传输行为较为复杂。因此,对无线网通信效率的研究具有一定的意义。本文针对在给定的无线网环境,提取能够反映网络的整体通信效率的指标,对指标进行定量分析后建立回归模型来描述网络的整体通信效率。

针对问题一:分析无线网中的环境,由通信节点是低速连续运动,其运动可以看作布朗运动的形式;节点是否接收到信息与通信距离有关;对某节点发送、接收、产生冲突信息的计数采用叠加原理。把无线网通信的节点放在三维空间里面研究,根据动态节点间的距离,对某时段各节点发送信息、接收信息、发生通信冲突进行研究,得到冲突率、时延率、吞吐率和承载率四个指标。分别将四个指标与通信效率进行相关性分析,再通过回归分析,建立通信效率的回归模型。即得到由四个指标生成的线性组合来表示网络的整体通信效率。

通过仿真模拟,得到多组关于通信效率与冲突率、时延率、吞吐率和承载率的数据,运用线性回归模型的程序求解出回归模型的回归系数,即得到网络的整体通信效率模型。在对模型进行实例验证,进一步说明模型的可行性和准确性。

针对问题二:当发送信息时发现存在冲突,在相同条件下去研究如何调整重发时间段来使网络通信效率尽可能高。由于网络的承载能力和冲突之间有着显著性的特征,在研究问题二的模型时分轻载和重载进行讨论。以轻载为例,通过建立有通信冲突的最小连通支配集来确定节点通信冲突的点集,通过分离冲突域、建立独立子集来尽可能和减小通信冲突。结合问题一中的通信效率模型,对指标进一步刻画,找到通信效率的最大值,从而解出对应的的时间段,由完成这一过程所需要的的时间与时间段做处理,可得到重发的时间段。即通过相应算法,结合问题一的模型,网络系统可自动调整重发时间段。

用 MATLAB2018a 软件,对节点数为 10 的通信网络进行仿真动态模拟,可得到存在冲突时,高通信效率下所对应重发的时间段。

本文最后,对问题一和问题二建立的模型及求解进行了优缺点分析,以及对模型的展望进行了说明。

参赛队号: 34768

参赛密码

(由组委会填写)

所选题目: B 题

Abstract

At present, network technology has developed rapidly, but in the distributed environment of wireless network communication system, information transmission between nodes is more complex. Therefore, the study of wireless network communication efficiency has certain significance. In this paper, in a given wireless network environment, the indicators that can reflect the overall communication efficiency of the network are extracted, and a regression model is established to describe the overall communication efficiency of the network after the quantitative analysis of the indicators.

For problem 1: By analyzing the environment in wireless network, the motion of communication node can be regarded as the form of Brownian motion because the communication node moves continuously at low speed. Whether the node receives information is related to the communication distance. The superposition principle is used to count the conflicting information sent, received and generated by a node. The wireless network communication nodes are studied in a three-dimensional space. According to the distance between the dynamic nodes, the information sent, received and communication conflicts are studied for each node in a certain period of time, and the four indexes of conflict rate, delay rate, throughput rate and bearing rate are obtained. The correlation analysis between the four indicators and the communication efficiency is carried out respectively, and then the regression model of communication efficiency is established through the regression analysis. In other words, a linear combination of four indexes is obtained to represent the overall communication efficiency of the network.

Through simulation, we can obtain multiple sets of data on communication efficiency and conflict rate, delay rate, throughput rate and load-bearing rate, and the regression coefficient of the regression model is solved by the program of linear regression model, that is, the overall communication efficiency model of the network is obtained. The feasibility and accuracy of the model are further illustrated by an example.

For problem 2: When the message is sent, there is a conflict. Under the same condition, we study how to adjust the retransmission period to make the network communication efficiency as high as possible. Because of the significant characteristics between the carrying capacity of the network and the conflict, the model of problem two is discussed in light load and heavy load. Taking light load as an example, the point set of node communication conflicts is determined by establishing the minimum connected dominating set with communication conflicts, and the communication conflicts are minimized and minimized by separating the conflict domains and establishing independent subsets. In combination with the communication efficiency model in problem 1, the index is further described to find the maximum of the communication efficiency, so as to solve the corresponding time period. The time period of retransmission can be obtained from the time and time period required to complete this process. That is, through the corresponding algorithm, combined with the model of problem one, the network system can automatically adjust the retransmission period.

We used the software MATLAB2018a to carry out the simulation and dynamic simulation on the communication network with the number of nodes of 10, and the corresponding time period of retransmission under high communication efficiency can be obtained when there is conflict.

At the end of this paper, we analyzed the advantages and disadvantages of the model and its solution of problem 1 and problem 2, and explained the prospect of the model.

Key words: Communication conflict ; Overall communication efficiency ; The regression model; Dynamic simulation; The simulation;

目录

1. 问题重述和背景	1
1.1 问题重述	1
1.2 背景分析	1
2. 问题分析	2
2.1 问题一的分析	2
2.2 问题二的分析	2
3. 模型假设	2
4. 定义与符号说明	3
5. 模型的建立与求解	4
5.1 问题一: 网络的整体通信效率模型	4
5.1.1 模型的基础及节点的运动分析	4
5.1.2 各指标的合理设立	5
5.1.3 构建通信效率的回归模型	6
5.1.4 模型的求解	7
5.1.5 基于通信效率的回归模型的实例验证	9
5.2 问题二: 自动调整时段的效率模型	11
5.2.1 模型的网络环境基础	11
5.2.2 模型的建立	11
5.2.3 自动调整时段的效率模型的仿真模拟	13
6. 模型评价	14
6.1 模型的优点	14
6.2 模型的优点	14
7. 模型的改进与推广	15
7.1 模型的改进	15
7.2 模型的推广	15
8. 参考文献	16
9. 附录	17

1 问题重述和背景

1.1 问题重述

在使用广播的方式进行无线网通信时，解决节点的广播之间发生冲突是一个关键的问题。

对于本题，主要考虑这样的一个无线网：它的每个通信节点都是低功率的发射器，并且每个节点在整个空间上做的低速连续运动，因此，对于每一个节点而言，只有和它距离在一定范围之内节点才能收到它的信号，在建模过程中，我们把它叫作是通信距离，当任意两个节点之间的距离在通信距离之内时，我们就说两个节点之间可以互相通信，反之则不能互相通信。我们需要注意的是，节点会相互接近或远离。另外，在每个节点发送信号时，它只需要把自己的信息广播出去，需要此条信息的节点只需要被动收听即可，并不需要点对点地持续交换信息。题目中还规定所有信息都使用同一个频率进行发送，当有两个或多个节点的广播发生冲突，能同时收听到它们的节点就都能监听到冲突。

本题要求我们发送每条信息所占用的时间都是等长的并且所有的节点都拥有同样的内置算法，且拥有一个校准过的高精度时钟的条件下，我们需要解决如下两个问题：问题 1 根据通信网的特点构造一些合理的指标来评价网络的通信效率；问题 2 是设计一个方案来选择发送的时间段，并且考虑在发送信息时发现存在冲突时，能够自动调整重发的时间段，使得网络的整体通信效率尽可能高。

1.2 背景分析

自从信息理论创建以来，人们对无线信道容量的估计就成为了人们关注的聚焦话题之一。近年来，随着无线通信技术、网络技术、分布式、自组织网络等网络技术的不断发展，人们研究的重点也开始由无线信道容量转向网络容量估计，使得发展低成本、低功耗、小体积的多功能传感器成为可能。在这种无线传感器网络中，各个节点协同工作，实时准确地监测、感知网络覆盖范围中的环境，并且能够发送和处理用户需要的信息，最终将这些信息传送到接受者 [1]。

广播是无线传感器网络中最基础的通信方式之一，其目的是将信息从网络中的需要发送信息的节点发送给网络中其余的每一个节点。虽然无线传感器网络节点位置相对稳定，但是由于节点数量较多、密集度高，这样网络中节点在发送和接收消息时就会产生冲突，引起网络拥堵，增加节点的能量消耗，从而会降低网络的整体通信效率，因此，设计优化无线传感器网络的广播算法去提高网络的整体通信效率具有很强的现实意义，这不仅能够解决网络拥堵问题，而且能够减少节点能量的消耗，延长网络的寿命。因此，容量估计理论对于提高网络效率、增强网络业务保障能力都具有重要的理论价值。

2 问题分析

本题主要的工作就是建立一些合理的指标并且找到指标之间的关系建立一个能够描述网络的整体通信效率的模型，为了尽量避免冲突，需设计一个方案来提高网络的整体通信效率。

2.1 问题一的分析

需要构造一个合理的指标来描述网络的整体通信效率。首先，把网络上的节点放到空间直角坐标系上考虑，节点的运动遵循布朗运动的形式，由此我们根据节点的动态坐标得到节点间的距离 d 并给定一个通信距离 D ，当 $d \leq D$ 时，则两个节点之间可以互相通信，反之则不能进行互相通信；其次，我们再找出描述网络的整体通信效率的相关指标，有冲突率、时延率、吞吐率和承载率；然后，我们根据建立的四个指标与通信效率之间建立多元线性回归分析模型来描述网络的整体通信效率；最后，为了验证模型的准确性，我们把模型带到具体通信网络的实例中进行计算和分析。

2.2 问题二的分析

节点发送信息时可能发现存在冲突，这种冲突尽可能少就更有利于通信效能的提高。通过建立有通信冲突的最小连通支配集来确定节点冲突的点集；通过分离冲突域，建立独立子集来尽可能减小通信冲突，均通过相应的算法来实现。网络在运作的过程中，所承载的能力对通信冲突存在不一样的影响，此时，对网络的承载能力分为轻载和重载进行分类讨论。结合第一问建立的模型，在指标明确的情况下，找出尽可能高的通信效率，从而找到对应的时间，与初始时间一处理即可得到调整后的时间段。相应的可以做动态仿真模拟，在轻载或是重载下，冲突域的改变，可以调整出不同的时间段，使通信效率尽可能高。

3 模型假设

- (1) 假设每个节点都有相同的内置算法；
- (2) 假设每个节点的通信距离始终相等，即为常数；
- (3) 假设在没有广播冲突的前提下，节点一旦接收到消息就一定是接收到完整的信息；
- (4) 假设不考虑某节点未完成接收已知的所需信息时不会发送其他信息；
- (5) 假设发送每条信息这一操作所占用的时间是等长的且忽略不计；
- (6) 假设计算网络中信息传播的最大时间与实际网络中信息传播的最大时间之间存在的误差几乎为 0。

4 定义与符号说明

表 1 符号说明

符号	符号说明
$H(X)$	每个节点发出消息的信源熵
$(S_x(t), S_y(t), S_z(t))$	节点在某时间段内时刻的空间坐标
(x_1, x_2, x_3, x_4)	网络通信中的冲突率、时延率、吞吐率、承载率
N	某时间内有通信节点的总数
n	某时间内时间段内发送信息的节点数
m_i	i 节点所对应的接受信息的节点数
D	能够感知相邻节点接受到信号的临界值距离
i, j	分别为发送信息和接收信息的节点
$d_{ij}(t_i)$	i 节点与 j 节点在信息的发送与接受过程中的距离
v	在网络节点间信息传递的速度
w	在某时间内通信冲突的节点数
t_i	在某时间内 i 节点从信息发出到需要接收此信息的节点完成接收的时间
T	在某时间内完成信息传输的总时间
y	网络的通信效率
V	网络中所有节点的集合
E	网络中所有边的集合
$N(a)$	与节点 α 相邻节点的集合（包括节点 α ）
$N(N(a))$	节点 α 及相邻节点产生通信冲突时的时刻的集合
κ	网络的通畅度
$(x_{1e}, x_{4e}), e = 1, 2, \dots, l$	关于冲突率和承载率的离散点列
λ	节点 v 的邻节点子集
α	产生通信冲突的点集
$[T_2 + T', 2T_2 - T_1 + T']$	最高通信效率时节点发送消息的时间段

5 模型的建立与求解

5.1 问题一：网络的整体通信效率模型

5.1.1 模型的基础及节点的运动分析

对于无线网络而言，因为每个节点发送消息的频率相同，我们假设消息的广播频率为 H ；而由每个节点都有相同的内置算法，我们清楚的知道所有节点在该通信网中发送一条消息的数据是相同的，这里我们对消息进行量化，即得出每个节点发出的消息信源熵是相等的，把信源熵记为 $H(X)$ ，为了方便计算，进一步设定，在没有广播冲突的前提下，如果节点一旦接收到消息，那么一定是正确的接收到了全部的信息，即广播过程中的噪声熵 $H(Y|X) = 0$ 。

在无线网络中，不妨设所有通信节点所处的通信网是一个立体的网络空间。每个通信节点在空间上进行着无规则的低速连续运动，其与热效应中的分子运动状态极为相似，所以，我们可以将节点的运动状态抽象理解为布朗运动，将用经典的随机过程对其进行数学描述。

把空间上的通信节点放到空间直角坐标系上，某节点在时间段 $[T_1, T_2]$ 内某时刻 t 的空间坐标为： $(S_x(t), S_y(t), S_z(t))$ ，

由于其为低速连续运动，则满足公式 1：

$$\begin{cases} \lim_{t \rightarrow 0} \frac{dS_x(t)}{dt} = 0 \\ \lim_{t \rightarrow 0} \frac{dS_y(t)}{dt} = 0 \\ \lim_{t \rightarrow 0} \frac{dS_z(t)}{dt} = 0 \end{cases} \quad (1)$$

即

$$\begin{cases} v_x = \frac{dS_x(t)}{dt} \\ v_y = \frac{dS_y(t)}{dt} \\ v_z = \frac{dS_z(t)}{dt} \end{cases} \quad (2)$$

此节点经时间后的空间坐标为： $(S_x(t) + v_x \Delta t, S_y(t) + v_y \Delta t, S_z(t) + v_z \Delta t)$

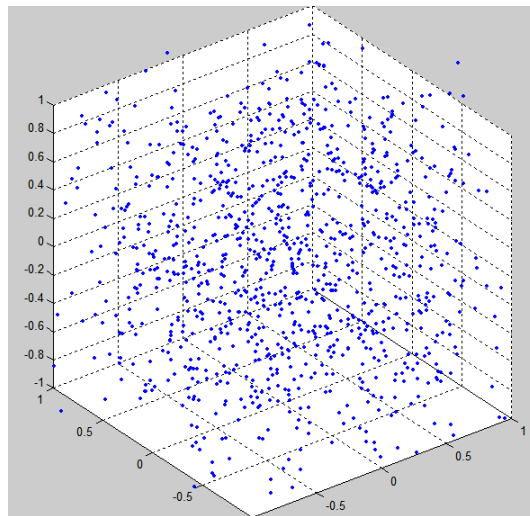


图 1 MATLAB 仿真的布朗运动

5.1.2 各指标的合理设立

(1) 冲突率

冲突率是某时间段内遭受通信冲突的节点数与整个网络节点数的占比 [5]，若同一节点在规定时间内多次遭受通信冲突则采用叠加原理计算节点数。如某一个节点在某时段遭受 4 次冲突，则此时段的冲突数为 4。

对于某一通信网络，在时间 $[T_1, T_2]$ 内有通信节点的总数为 N ，此时间段内发送信息的节点数为 n 。由于一个节点把信息广播出去，可能会有多个节点需要收听此条信息，所以 i 节点所对应的接受节点数为 m_i ，其中 $i = 1, 2, \dots, n$ 。在整个网络中，节点间信息传递的速度是相同的。

取 $t_i \in [T_1, T_2]$ ， t_i 为 i 节点发送信息的时刻， T_{ij} 为 j 节点收到 i 节点所发出的信息的时刻， $j = 1, 2, \dots, m_i, m_j = 1, 2, \dots$ 。由于节点在网络空间中是运动着的，则在不同时刻两节点之间的距离也是变化的，记为 i 节点到 j 节点的距离，由前期分析知， t_i 时刻 i 节点的空间坐标为 $(S_{x_i}(t_i), S_{y_i}(t_i), S_{z_i}(t_i))$ ， j 节点的空间坐标为 $(S_{x_j}(t_j), S_{y_j}(t_j), S_{z_j}(t_j))$ ，则 j 点接受信息时的空间坐标为 $(S_{x_j}(t_i) + v_x \Delta t, S_{y_j}(t_i) + v_y \Delta t, S_{z_j}(t_i) + v_z \Delta t)$ ，其中 $\Delta t = T_{ij} - t_i$ ，则 i 节点与 j 节点在信息的发送与接受过程中的距离

$$d = \sqrt{[S_{x_j}(t_i) + v_x \Delta t - s_{x_i}(t_i)]^2 + [S_{y_j}(t_i) + v_y \Delta t - s_{y_i}(t_i)]^2 + [S_{z_j}(t_i) + v_z \Delta t - s_{z_i}(t_i)]^2} \quad (3)$$

满足 $d_{ij}(t_i) < D$ 时，有

$$t_i + \frac{d_{ij}(t_i)}{v} = T_{ij} \quad (4)$$

对节点 $k(k = 1, 2, \dots, j)$ 收到信息的时刻有 $T_{1K}, T_{2K}, \dots, T_{GK}, g = 0, 1, \dots, n$ ，对这些时刻点按大小进行分类，分成 $p(p = 1, 2, \dots, q, q \leq n)$ 个类，当分类 p 构中的元素个数大于 1，记：当分类构中的元素个数等于 1，记 $G_p = 1$ ；则节点 p 的通信冲突数为 1，记 $G_p = 0$ ；则节点 k 的通信冲突数为

$$Q_k = \sum_{p=0}^q G_p \quad (5)$$

则整个时段通信冲突数

$$w = \sum_{k=0}^j Q_k \quad (6)$$

每一节点发出信息都需要对节点总数 N 进行一次叠加，则通信冲突率

$$x_1 = \frac{w}{nN} \quad (7)$$

(2) 时延率

时延率是某时间段内从信息发出传输请求到被成功地传输到目的节点所需要的时间与总时段的占比。

每条信息发送的时间等长，由于节点完成发送这一操作的时间与传输时间相比非常小，可忽略不计。[3] 那么，在时间 $[T_1, T_2]$ 内， i 节点从信息发出到需要接收此信息的节点完成接收的时间：

$$t_i' = \max\left(\frac{d_{ij}(t_i)}{v}\right) \quad (8)$$

在时间 $[T_1, T_2]$ 内完成信息传输的总时间

$$T' = \sum_{i=0}^n t_i' \quad (9)$$

于是，可得时延率为

$$Tx_2 = \frac{T'}{n(T_2 - T_1)} \quad (10)$$

(3) 吞吐率

吞吐率是某时间段内发送信息的的节点数与整个网络节点数的占比由上述分析可知，在时间 $[T_1, T_2]$ 内，吞吐率：

$$x_3 = \frac{n}{N} \quad (11)$$

(4) 承载率

承载率是某时间段内接收信息和产生冲突的节点数与整个网络节点数的占比，若同一节点在规定时间内发送信息则采用叠加原理计算整个网络的节点数。

在时间内 $[T_1, T_2]$ ， i 节点所对应的接受节点数为 m_i ，而总的节点数为

$$m = \sum_{i=0}^n m_i \quad (12)$$

由冲突率的分析可知， m 包含了产生冲突的节点数，故承载率：

$$x_4 = \frac{m}{nN} \quad (13)$$

5.1.3 构建通信效率的回归模型

(1) 回归模型的建立

由上述描述可知，冲突率是某时间段内遭受通信冲突的节点数与整个网络节点数的占比；时延率是某时间段内从信息发出传输请求到被成功地传输到目的节点所需要的时间与总时段的占比；吞吐率是某时间段内发送信息的节点数与整个网络节点数的占比；承载率是某时间段内接收信息和产生冲突的节点数与整个网络节点数的占比。因此，在一定范围内，根据定性分析的结果，我们可知冲突率、时延率、吞吐率和承载率和网络整体通信效率之间是存在负相关关系的，为了更加直观地看出它们与网络整体通信效率之间的关系，我们将其图像表示分别如下：

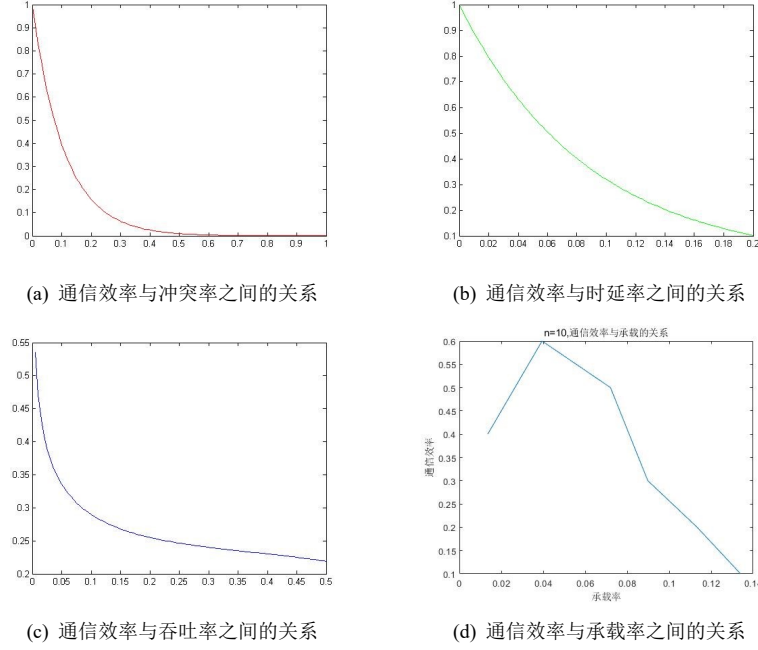


图 2 通信效率与 4 个指标之间的关系

由上述图像可以看出，除了个别时间段之外，冲突率、时延率、吞吐率和承载率和网络整体通信效率之间是存在相关关系的，于是，我们用来描述网络的整体通信效率，结合冲突率、时延率、吞吐率和承载率四个指标建立多元线性回归模型，有：

$$\begin{cases} y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \varepsilon \\ \varepsilon \sim N(0, \sigma^2) \end{cases} \quad (14)$$

其中：回归系数 $\beta_0, \beta_1, \beta_2, \beta_3, \beta_4$ 和都与指标 x_1, x_2, x_3, x_4 无关。

通过相关数据的收集，得到多组关于某通信网络的效率、冲突率、时延率、吞吐率和承载率的数据，作相应处理，即给定 n 组独立数据 $[b_i, a_{i1}, \dots, a_{i4}], i = 1, \dots, n, n > 4$ ，其中，指

$$\text{标矩阵 } X = \begin{bmatrix} 1 & a_{11} & \cdots & a_{14} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & a_{n1} & \cdots & a_{n4} \end{bmatrix}, \text{ 效率矩阵 } Y = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$$

$\varepsilon = [\varepsilon_1 \cdots \varepsilon_n]^T, \beta = [\beta_0 \ \beta_1 \ \beta_2 \ \beta_3 \ \beta_4]^T$ ，则由式 14 可得

$$\begin{cases} Y = X\beta + \varepsilon \\ \varepsilon \sim N(0, \sigma^2 E_n) \end{cases} \quad (15)$$

其中， E_N 为 n 阶单位矩阵。

5.1.4 模型的求解

结合数据，我们通过利用 MATLAB 程序进行仿真，得到了通信网络在整个仿真时间过程中发生的冲突数、冲突率与网络中的节点数、每个节点发送消息的频度之间的关系，如下图所示：

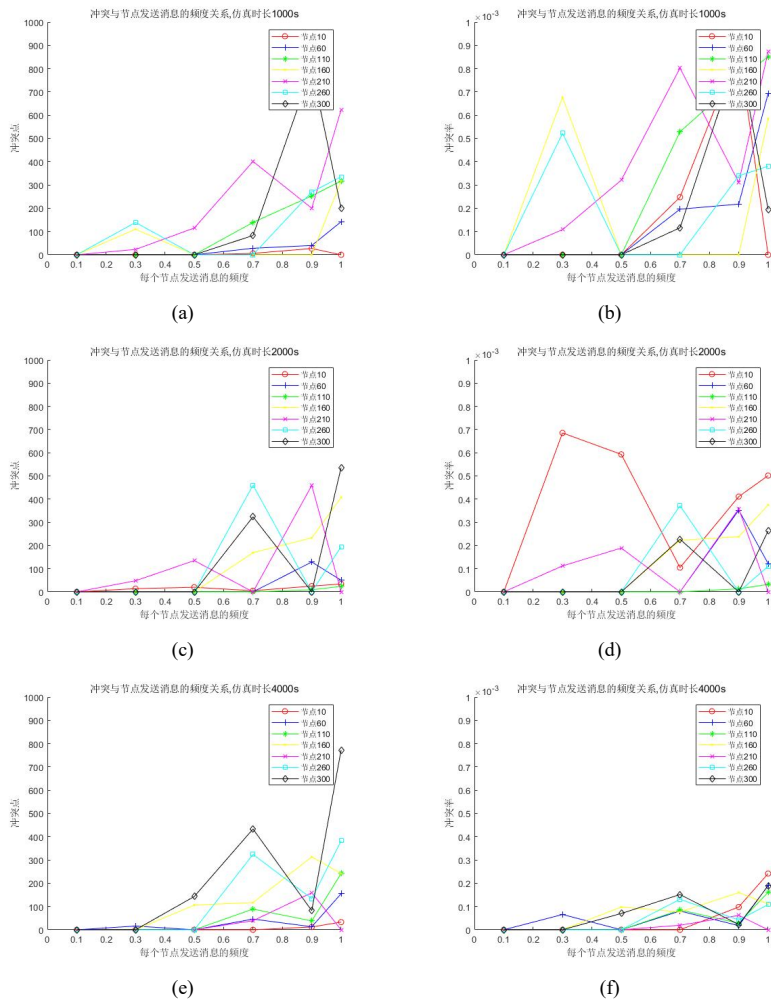


图 3 关于节点产生不同冲突数的仿真

联系实际可知，在任何一种通信网络中，都存在某个时间段内不会出现一个冲突节点的合理情况，这个在我们仿真的过程中有所体现。整体上，观察上图，我们发现每个节点发送消息的频度越大，通信网中方产生的冲突点的数量趋于越多；通信网中的节点数量越多，通信网中方产生的冲突点的数量趋于越多。

选取一个节点数为 10、仿真时长为 1000s 的网络，并对其通信效率做出评估，通过模拟仿真，得到如下统计数据：

表 2 关于 10 个节点的仿真数据

节点数	发消息频度	冲突率	时延率	吞吐率	承载率	通信效率
10	0.1000	0.0000	2.1751	0.3766	0.0135	0.4000
10	0.3000	0.0000	2.2512	0.3418	0.0394	0.6000
10	0.5000	0.0000	0.2079	0.3875	0.0720	0.5000
10	0.7000	0.0006	0.9276	0.3464	0.0898	0.3000
10	0.9000	0.0000	1.9265	0.3370	0.1133	0.2000
10	1.0000	0.0002	1.9484	0.3510	0.1340	0.1000

运用上述的通信效率的回归模型，并运用计算机数据处理技术，用 MATLAB2018a 软件，得到则网络的整体通信效率回归模型

再用通信效率去验证模拟出的数据是否合理以及模型的假定是否合理，于是我们作出以下残差图进行验证，残差图如下图所示：

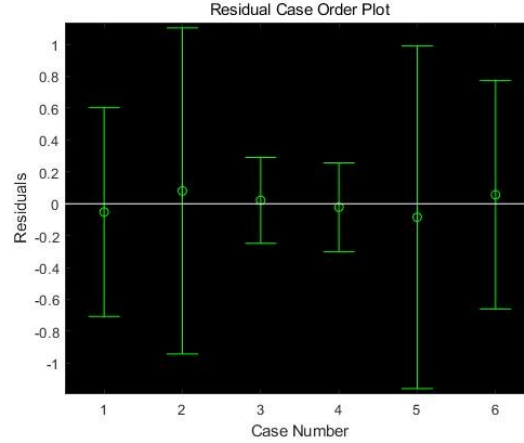


图 4 残差分析图

由上图可以看出，非线性回归模型拟合的还是非常号的，偏差的取值稳定在 $[-1,1]$ 之间。因此，说明仿真出的数据是比较合理的以及模型的假定也是比较合理的，这样也说明了我们所建立通信效率模型的合理性。

5.1.5 基于通信效率的回归模型的实例验证

假设现在某通信网络中总共有 10 个节点，分别给它们编号为 $1, 2, \dots, 10$ ，某节点在时间段 $[8:00, 9:00]$ 内， $t = 8:00$ 时刻的空间坐标为 $(1, 1, 1)$ ，由问题一模型可知，它满足如下规律：

$$\begin{cases} \lim_{t \rightarrow 0} \frac{dS_x(t)}{dt} = 0 \\ \lim_{t \rightarrow 0} \frac{dS_y(t)}{dt} = 0 \\ \lim_{t \rightarrow 0} \frac{dS_z(t)}{dt} = 0 \end{cases} \quad (16)$$

于是，有 $v_x = 1, v_y = 1, v_z = 1$

当此节点经过 $\Delta t = 0.1s$ 时间后的空间坐标为： $(1.1, 1.2, 1.3)$

下面，建立一些合理的指标去描述网络的整体通信效率。

(1) 各指标的设置

- 冲突率

对上述所给的通信网络，在时间 $[8:00, 9:00]$ 内有通信节点的总数为 10，此时间段内发送信息的节点数为 5，通信距离为 $D = 5$ 。假设第 1 个节点所对应的接受节点数为 5。取 $t_1 = 8:30, T_{12} = 8:30$ 为第 2 个节点收到第 1 个节点所发出的信息的时刻， $j = 1, 2, \dots, m_i, m_i = 1, 2, \dots$ 。因此， $t_1 = 8:00$ 时刻 1 节点的空间坐标为 $(1, 2, 3)$ ， j 节点的空间坐标为 $(3, 3, 3)$ ，则 2 点接受信息时的空间坐标为 $3.1, 3.2, 3.3$ ，则第 2 个节点与第 1 个节点在信息的发送与接受过程中的距离为：

$$d_{12}(t_i) = \sqrt{(1.1 - 3.1)^2 + (1.2 - 3.2)^2 + (1.3 - 3.3)^2} = 2\sqrt{3} \quad (17)$$

于是满足 $d_{12} < 5$ 时，则有 $T_{12} = 8:02$ ，同理可算出其他节点接收消息的时刻，当两个时刻点相同时，我们就认为这两节点发生冲突。则整个时段通信冲突数

$$w = \sum_{k=0}^j Q_k = 2 \quad (18)$$

每一节点发出信息都需要对节点总数 10 进行一次叠加，则通信冲突率

$$x_1 = \frac{w}{nN} = 0.0026 \quad (19)$$

• 时延率；

由于每条信息发送的时间等长，且节点完成发送这一操作的时间与传输时间相比非常小，可忽略不计。那么，在时间 $[8:00, 9:00]$ 内，每一个节点从信息发出到需要接收此信息的节点完成接收的时间分别为 $0.01s, 0.30s, 0.25s, 0.20s, 0.25s, 0.30s, 0.112s, 0.35s, 0.312s, 0.50s$ ，在时间 $[8:00, 9:00]$ 内完成信息传输的总时间

$$T' = \sum_{i=0}^{80} t_i' = 22.593 \quad (20)$$

于是，可得时延率为

$$x_2 = \frac{T'}{n(T_2 - T_1)} = 0.0054 \quad (21)$$

• 吞吐率

吞吐率是某时间段内发送信息的的节点数与整个网络节点数的占比，于是，由上述分析可知，在时间 $[8:00, 9:00]$ 内，吞吐率为

$$x_3 = \frac{n}{N} = 0.3788 \quad (22)$$

• 承载率

在时间 $[8:00, 9:00]$ 内，节点所对应的接受节点数为 m_i ，而总的节点数为

$$m = \sum_{i=0}^n m_i (i = 1, 2, \dots, 10) \quad (23)$$

由冲突率的分析可知， m 包含了产生冲突的节点数，故承载率为

$$x_4 = \frac{m}{nN} = 0.0989 \quad (24)$$

(2) 整体通信效率求解

把上述计算得到的冲突率、时延率、吞吐率和承载率的值代入 (18) 式，得到整体网络的整体通信效率

$$y = 0.3494 \quad (25)$$

5.2 问题二: 自动调整时段的效率模型

以广播的形式进行无线网通信, 可能会产生网络的拥塞、信道的竞争和数据的冲突, 从而降低了网络的性能, 增加了节点的能量消耗, 即整个网络的通信效率降低。[9] 那么, 如何提高网络的通信效率就有着很大的现实意义。

在给定的无线网环境中, 信息的发送只使用同一频率, 这里即研究节点间以单一信道的方式传输信息。节点发送信息时可能发现存在冲突, 那我们可以对网络环境进行分析, 在不同的网络承载能力下去讨论发生冲突时, 怎么的时间段, 能够使整个网络的通信效率尽可能高, 即分类讨论问题。

5.2.1 模型的网络环境基础

无线网络可用二元组 $G = (V, E)$ 来表示, 其中 V 为网络中所有节点的集合, E 为网络中所有边的集合, G 中任意两个点之间至多只有一条边, 且各节点是无自圈的, 并且满足 $E \subseteq V^2$ 。如果边 $(\mu, v) \in E$, 则节点 μ 与 v 可以互相通信。 R 是节点 v 的最大通信距离, $d(\mu, v)$ 表示两节点之间的距离, 则 $E = \{(\mu, v) | (\mu, v) \in V^2 \text{ 且 } d(\mu, v) \leq R\}$ 。

若节点 x, y 是 G 中的节点, 节点 x, y 分别在对方有效的通信范围之内且可以相互通信, 则称 x, y 互为邻节点, 由于发送和接收工作可以同时进行, 即可认为链路通信是双向的。[8] 各节点可以在某一时间段来实现各自节点和邻节点的信息传递。 $N(a)$ 表示与节点 a 相邻节点的集合, $N(a) = \{v | (a, v) \in E, v \in V\}$, 包括节点 a , $N(N(a))$ 表示节点 a 及相邻节点产生通信冲突时的时刻的集合, $N(N(a)) = \{n | n \notin N(a), n \in \bigcup_{t \in N(a)} N(t), \square(n, t) \in E\}$ 。不妨设定网络中各节点相互连通, 初始时, 整个网络是连通的, 各节点具有相同的能量等级。

对于有 m 个节点的通信网络来说, 我们将其看作 m 个节点的有向图。那么, 在最初的状态下, 节点之间是两两强连通的图。信誉度则是边上的权值 [4]。

网络的通畅度 κ , 可以表示为连通图当前度数与完全强连通时度数的比值:

$$\kappa = \frac{\sum_{i=1}^m d(\mu_i, v_i)}{2 \cdot (m-1)^2} \quad (26)$$

5.2.2 模型的建立

网络在运作的过程中, 所承载的能力与通信冲突之间是否存在显著性特征, 若有, 在分析发生冲突时, 如何调整时段来提高效率就需要根据特征来讨论, 在此, 以冲突率 x_1 和承载率 x_4 之间建立联系, 找到关于 (x_1, x_4) 的 1 组互不相同的离散点列 $(x_{1e}, x_{4e}), e = 1, 2, \dots, l$ 做插值并画出关系曲线

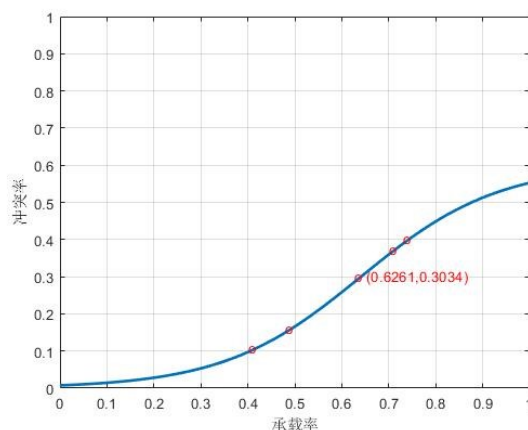


图 5 承载率与冲突率的关系

从图中，可以看到存在突变值 $\mu = 0.6261$ ，即为对应的承载率 $x_4 = 0.6261$ 这时，我们可以对网络的承载能力做一个划分：

当承载率在 0–75% 之间为轻载；当承载率在 75%–100% 之间为重载。

现以轻载为例，建立自动调整时段的效率模型，关于重载的讨论类似分析即可得。

(1) 建立有通信冲突的最小连通支配集

在无线网络 $G = (V, E)$ 中，通过构建最小连通支配集来描述网络效率，从而在效率尽可能高的情况下找到最佳的时间段。设节点 μ 传输信息到节点 v ，节点 v 的邻节点子集 $\lambda = N(v) - N(\mu)$ ，产生通信冲突的点集 $\alpha \{ \alpha | \alpha \subseteq C_\lambda \subseteq N(N(\mu)) \}$ ， C_λ 表示集合 λ 的补集，[6] 具体算法如下：

- Step1 对于网络中任意节点 μ 发送信息，节点 v 收到所需信息，令 $\lambda = N(v) - N(\mu)$ ， $\varphi = N(N(v)) - N(N(\mu))$
- Step2 While ($\varphi = 0$) 此时把子集中的邻节点记为 τ ;
- Step3 $\lambda = \lambda - \tau$;
End While
- Step4 Return(α)

(2) 分离冲突域，建立独立子集

在信息的传递过程中，当两个及两个以上节点同时向一邻居节点发送信息时将导致通信冲突的发生。为了避免这种情况的发生，首先将集合 α 中的节点分成若干个独立的子集，同一子集内的节点无共同的邻节点，因此他们在同时发送信息时不影响各自邻节点接收信息。[6] 具体算法如下：

- Step1 初始化独立子集数 $NUM - IS = 1$
- Step2 While(集合 α 非空) 从集合 α 选取任意一节点，设为 τ ；For $i = 0$ to $NUM - IS$ 如果节点 τ 和独立子集 $IS(i)$ 中所有节点没有公共的邻节点，则将节点 τ 加入独立子集 $IS(i)$ 中并且转 step4；否则转 step3；
- Step3 $NUM - IS = NUM - IS + 1$ ；创建新的独立子集 $NUM - IS$ ，将节点 τ 加入该独立子集 $NUM - IS$ ；
- Step4 从集合中删除节点，并将各独立子集按照节点的总度数的降序排列；；

也就是说，通过分离冲突域达到了减少冲突数的目的。

(3) 通信效率对应的最佳时间段

要评判产生冲突时，哪个时段的通信效率最高，那么，均为时间点且，把时间段看成是自变量，整个无线网络的通信效率看成因变量。某时间段内发送信息的节点数都是相同的，则吞吐率相对于通信效率而言是一个常数。由 (18) 式，使网络的整体通信效率尽可能高，即

$$\max y = -306.0038x_1 - 0.1720x_2 - 5.5231x_3 - 4.2903x_4 + 2.9636 \quad (27)$$

亦即

$$\max y = -306.0038 \frac{w}{nN} - 0.1720 \frac{T'}{n(T_2 - T_1)} - 5.5231x_3 - 4.2903x_4 + 2.9636 \quad (28)$$

其中， w 为通信冲突点集 α 的冲突数，吞吐率 x_3 为常数， $0 \leq x_4 \leq 72\%$ 。

通过仿真模拟可以知道 $\max y$ ，以及对应的时间段 $[T_1, T_2]$ ，在判定 $\max y$ 时，所需要的时间为 T' ，经调整后的时间段为 $[T_2 + T', 2T_2 - T_1 + T']$ ，即在发送信息发现存在冲突时，根据通信效率，能自动调整重发的时间段。

5.2.3 自动调整时段的效率模型的仿真模拟

取节点数 $n=20$ ，通过 MATLAB2018a 软件，运用自动调整时段的效率模型进行仿真动态模拟，下面选取某时刻点的图进行说明：

(1) 轻载情况下，取 $x_4 = 0.3221$

经仿真动态模拟，得到此次过程完成信息传递的时间 $T = 00 : 02 : 12$ ，效率最高为 $\max y = 0.78$ ，其对应的时间段 $[T_1, T_2] = [8 : 05 : 32, 8 : 08 : 41]$ ，则经调整后重新发送的时间段 $[T_2 + T', 2T_2 - T_1 + T'] = [8 : 10 : 53, 8 : 14 : 02]$ ，为在 $[T_1, T_2]$ 中取两个时刻 $t_1, t_2 \in [T_1, T_2]$ 。

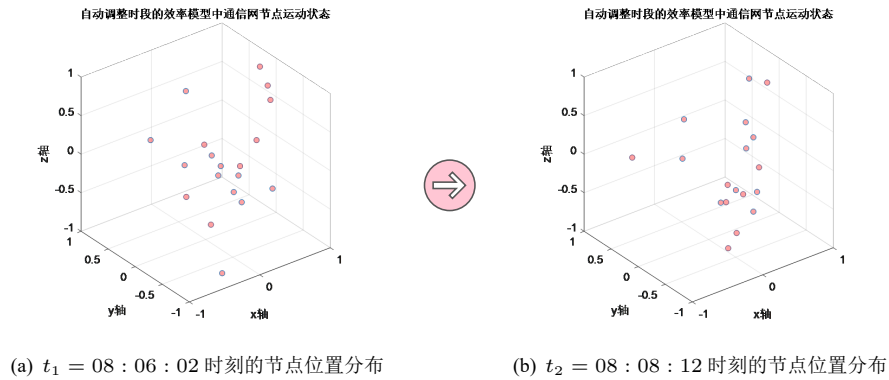


图 6 轻载时动态模拟

(2) 重载情况下，取 $x_4 = 0.8231$

经仿真动态模拟，得到此次过程完成信息传递的时间 $T = 00 : 05 : 17$ ，效率最高为 $\max y = 0.53$ ，其对应的时间段 $[T_1, T_2] = [8 : 02 : 32, 8 : 07 : 14]$ ，则经调整后重新发送的时间段 $[T_2 + T', 2T_2 - T_1 + T'] = [8 : 12 : 31, 8 : 17 : 02]$ ，为在 $[T_1, T_2]$ 中取两个时刻 $t_1, t_2 \in [T_1, T_2]$ 。

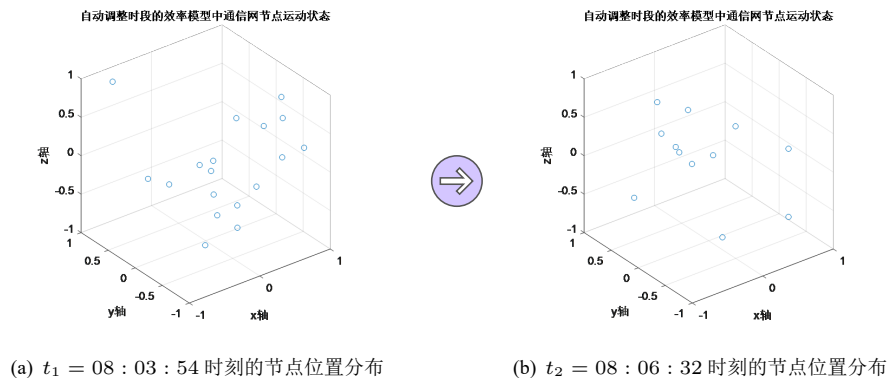


图 7 重载时动态模拟

6 模型评价

6.1 模型的优点

- 问题一中把节点的空间上的低速连续运动看作是布朗运动，这样有助于我们对节点运动的追踪；
- 所建立的每个指标都是通过计算之后得到的，这样有利于提高选取指标的准确性与独立性；
- 把所有指标通过使用线性多元回归分析给它们建立起联系，把它当作一个大指标去描述网络的整体通信效率，这样有利于简化我们的模型；
- 在指标构造完成之后，我们将其带入到已有的通信网络进行实证分析，这样做不仅省去了灵敏度的检验，还可以提高解题效率；
- 问题二中我们利用计算机编程多次进行模拟且循环结果以数列的形式展现出来，这样有利于根据最佳的效率指标反馈的结果选择合适的时间段。

6.2 模型的缺点

- 布朗运动并不能完全地描述节点的运动规律，它只能近似地描述每一个节点的运动规律；
- 在计算每两个节点接收和发送信息时，没有考虑接收和发送信息的实际时间，因为计算出的理论时间与实际时间可能存在一定的偏差。

7 模型的改进与推广

7.1 模型的改进

在问题一中，我们构造出了冲突率、时延率、吞吐率和承载率四个指标，然后运用线性多元回归模型建立起这四个指标之间的联系，来描述网络的整体通信效率。但是，在计算最大时间的时候，我们根据问题一模型中给出的公式计算出的最大时间与实际网络中信息传播的最大时间可能存在一定的误差 [7]。我们在建模时，为了简化建模过程，认为它们存在的误差几乎为 0，于是在此基础上做了补充，设理论算出的最大时间为 T_{IJ} ，实际的最大时间为 T_{ij}^l ，那么实际时间与理论时间之间的绝对误差为 $T = |T_{ij}^l - T_{ij}|$ ，于是当 T 的值越小且实际的最大时间也越小时，网络的通信效率越好。

指标之间我们建立的是线性回归模型，但是在实际生活中，它们之间可能非线性关系的，于是我们在此基础进行改动，于是 (14式) 写为 $y = \beta_0^1 + \beta_1^1 e^{x_1} + \beta_2^1 e^{x_2} + \beta_3^1 e^{x_3} + \beta_4^1 e^{x_4} + \varepsilon$ ，再用运用计算机进行仿真得到相关的参数。[2] 这样做更有利于将我们的模型运用在实际问题中，也有利于模型的推广。

7.2 模型的推广

本文主要研究的是分布式无线网络的整体通信效率的问题，并且讨论了在网络处于轻载与重载的情况下，当节点之间存在冲突时，我们运用计算机进行仿真，经过多次模拟，根据最佳的通信效率选择出合适的重发时间段。我们可以推广到电磁波的传播问题，如何设计网络才能使网络通信效率更高的问题以及如何解决通信网络卡顿的等问题，并加以改进，具有很强的现实意义。

8 参考文献

- [1] 赵宇航, 马修军. 一种高效率的多智能体协作学习通信机制 [J]. 信息安全研究, 2020, 6(04): 345-349.
- [2] 霍天. 开关柜内局部放电电磁波特性分析 [J]. 电工电气, 2019(12): 22-27+58.
- [3] 刘金魁. 基于大数据的网络通信传输效率控制 [J]. 信息通信, 2019(11): 39-40.
- [4] 孙晓玮, 陈青松. 通信效率的确定性与可靠性分析 [J]. 产业与科技论坛, 2014, 13(13): 81-82.
- [5] 曹剑馨, 梁庚. CAN 总线通信控制协议的仿真与性能分析 [J]. 中兴通讯技术, 2014, 20(03): 48-51.
- [6] 张晓玲, 梁炜, 于海斌, 封锡盛. 无线传感器网络传输调度方法综述 [J]. 通信学报, 2012, 33(05): 143-157.
- [7] 杨盘隆, 陈贵海. 无线网状网容量分析与优化理论研究 [J]. 软件学报, 2008(03): 687-701.
- [8] 卢先领, 于继明, 孙亚民. 一种能量高效的 Adhoc 无线网络广播算法 [J]. 计算机工程与应用, 2007(31): 119-122.
- [9] 刘欣. 通信网络性能评估研究 [D]. 西安电子科技大学, 2007.

9 附录

下面是我们小组做模型仿真时用到的代码。

Input MATLAB source:

```
%2020 Team #34768
clc,clear all
hang=1;
ss=zeros(7*5,4);
%length(number)
rrate=zeros(7*5,6);
for number=[10:50:300,300]
    x=2*rand(number,1)-1;
    y=2*rand(number,1)-1;
    z=2*rand(number,1)-1;
    for rate=[0.1:0.2:1.0,1.0]
        [m,sumc,d,ssum] =jdfz(number,rate,x,y,z);
        ss(hang,:)= [number,rate,m,sumc];
        %±£
        rrate(hang,:)= [number,rate,m/sumc,d,ssum/sumc,ssum/(9*number*1000)];
        hang=hang+1;
    end
end
disp("ð²¿ •      ;")
```

Input MATLAB source:

```
clc,clear all
n=10;
s=0.002;
D=0.1;
x=2*rand(n,1)-1;
y=2*rand(n,1)-1;
z=2*rand(n,1)-1;
h=plot3(x,y,z,'.');
xmin=-1;xmax=1;ymin=-1;ymax=1;zmin=-1;zmax=1;
axis([xmin xmax ymin ymax zmin zmax]);
axis square
grid on
set(h, 'MarkerSize',5)% òs ĝ

e={};
l=1;
for i=linspace(1,10,5000)
    %addres=randperm(n,1);
    alphabet = [0 1]; prob = [0.8 0.2];
    zhuangtai=randsrc(n,1,[alphabet; prob]);
```

```

for j=1:n
    if zhuangtai([j])==1
        alphabet1 = [1 2 3 4 5 6 7 8 9]; prob1 = [0.3 0.2 0.1 0.1 0.1 0.05 0.05 0.05 0.05];
        shuliang=randsrc(1,1,[alphabet1; prob1]);
        addres=randperm(n,shuliang);
        for k=1:shuliang
            addres1=addres([k]);
            d=sqrt((x([j])-x([addres1]))^2+(y([j])-y([addres1]))^2+(z([j])-z([addres1]))^2);
            if d<=D
                e{1}=d;
                l=l+1;
            end
            ee=cell2mat(e);
            uniq=length(ee)-length(unique(ee));
            if uniq>0

                end
                l=1;
                e={};
            end
        end
    end
end

x=x+s*randn(n,1);

y=y+s*randn(n,1);

z=z+s*randn(n,1);

set(h,'XData',x,'YData',y,'ZData',z);

end

```

解决本题第二问所用到的仿真代码。

Input MATLAB source:

```

clc,clear all
n=20;% ½
D=1;
T=5000;
s=0.005;% ¶

%² n, p(x,y,z)~[-0.5~0.5]
x=2*rand(n,1)-1;
y=2*rand(n,1)-1;
z=2*rand(n,1)-1;
h=plot3(x,y,z,'o');

```

```

xmin=-1;xmax=1;ymin=-1;ymax=1;zmin=-1;zmax=1;
axis([xmin xmax ymin ymax zmin zmax]);
axis square
grid on
set(h, 'MarkerSize',5);% 设置

title(" 图 1: 模型拟合结果 "); % 标题
xlabel("x "); % X 轴
ylabel("y "); % Y 轴
zlabel("z "); % Z 轴

% 生成 5000 个随机点
e={};
ae={};
aee=1;
m=0;% 记录命中次数
l=1;% 记录当前点坐标
% 生成 5000 个随机点
for i=linspace(1,10,5000)
for i=1:1:T
    alphabet = [0 1]; prob = [0 1.0];
    zhuangtai=randsrc(n,1,[alphabet; prob]);
    for j=1:n% 遍历所有点
        if zhuangtai([j])==1% 命中
            alphabet = [1 2 3 4 5 6 7 8 9]; prob1 = [0.3 0.2 0.1 0.1 0.1 0.05 0.05 0.05 0.05];
            shuliang=randsrc(1,1,[alphabet1; prob1]);% 生成随机数
            address=randperm(n,shuliang); % 生成随机地址, [1,shuliang] 为地址
            address2=find(address==j);
            % 记录命中次数
            while ~isempty(address2)% 地址命中
                address([address2])=randperm(n,1);
                address2=find(address==j);
            end

            for k=1:shuliang
                address1=address([k]);
                d=sqrt((x([j])-x([address1]))^2+(y([j])-y([address1]))^2+(z([j])-z([address1]))^2);
                if d<=D
                    e{1}=d;
                    l=l+1;
                    %ae{aee}=d;
                    %aee=aee+1;
                end
            end
            ee=cell2mat(e);
            uniq=length(ee)-length(unique(ee));
            if uniq>0
                m=m+1;% 记录命中次数
            end
        end
    end
end

```



```

end

if i<T
    l=1;
    clc,clear all
n=20;% %
D=1;
T=5000;
s=0.005;% %

%2 n, p(x,y,z) <math>\in [-0.5,0.5]</math>
x=2*rand(n,1)-1;
y=2*rand(n,1)-1;
z=2*rand(n,1)-1;
h=plot3(x,y,z,'o');

xmin=-1;xmax=1;ymin=-1;ymax=1;zmin=-1;zmax=1;
axis([xmin xmax ymin ymax zmin zmax]);
axis square
grid on
set(h, 'MarkerSize',5);% 设置

title(" 点 云 分布 "); % 标题
xlabel("x "); % X 轴
ylabel("y "); % Y 轴
zlabel("z "); % Z 轴

%  $\sigma = 5000 \cdot \Sigma^{-2}$  计算
e={};
ae={};
aee=1;
m=0;% 计算 平均值
l=1;% 计算 方差
%for i=linspace(1,10,5000)
for i=1:1:T
    alphabet = [0 1]; prob = [0 1.0];
    zhuangtai=randsrc(n,1,[alphabet; prob]);
    for j=1:n% 计算 平均值
        if zhuangtai([j])==1% 计算 方差
            alphabet1 = [1 2 3 4 5 6 7 8 9]; prob1 = [0.3 0.2 0.1 0.1 0.1 0.05 0.05 0.05 0.05];
            shuliang=randsrc(1,1,[alphabet1; prob1]);% 计算 平均值
            addres=randperm(n,shuliang); % 计算 方差, [1,shuliang] p
            addres2=find(addres==j);
            % 计算 平均值
            while ~isempty(addres2)%addres j <math>\in</math> 0% while
                addres([addres2])=randperm(n,1);
                addres2=find(addres==j);
            end
        end
    end
end

```

```

end

for k=1:shuliang
    addres1=addres([k]);
    d=sqrt((x([j])-x([addres1]))^2+(y([j])-y([addres1]))^2+(y([j])-y([addres1]))^2);
    if d<=D
        e{1}=d;
        l=l+1;
        %ae{aee}=d;
        %aee=aee+1;
    end
    ee=cell2mat(e);
    uniq=length(ee)-length(unique(ee));
    if uniq>0
        m=m+1;%± ġ ĩkuj Ĩ° £- ° ±
    end

    if i<T
        l=1;
        e={};
    end
end
end

drawnow

x=x+s*randn(n,1);

y=y+s*randn(n,1);

z=z+s*randn(n,1);

set(h,'XData',x,'YData',y,'ZData',z);
p=sprintf(' ĩµ ¶ ġ ĩ - ',i);
%£-t=%d
title(p);
end
disp(m) % ĩ ĩ ĩ 4µ µL ,»¹ ĩ £-disp<--->print/printf
e={};
end
end
end
end

drawnow

x=x+s*randn(n,1);

```

```

y=y+s*randn(n,1);

z=z+s*randn(n,1);

set(h,'XData',x,'YData',y,'ZData',z);
p=sprintf('  μ  ¶  ¢  ¤  - ',i);
%£~t=%d
title(p);
end
disp(m) %  ¤  ¤  ¤ 4μ  pL  ,»¹  休  £~disp<-->print/printf

```

我们小组在做节点通信仿真的过程得到的模拟数据。

表 3 仿真时长为 1000s, 节点数为 60 的模拟数据

节点数	发消息频度	冲突率	时延率	吞吐率	承载率
60	0.1	0	1.527135728	0.367036964	0.013827778
60	0.3	0	0.972965825	0.377111002	0.042592593
60	0.5	0	2.075187278	0.381591582	0.072122222
60	0.7	0	0.84892162	0.380326488	0.100311111
60	0.9	0	0.882804935	0.380881251	0.128909259
60	1	0.000279129	0.969348098	0.37209792	0.140712963

表 4 仿真时长为 1000s, 节点数为 110 的模拟数据

节点数	发消息频度	冲突率	时延率	吞吐率	承载率
110	0.1	0	1.078679882	0.379719525	0.014222222
110	0.3	0	0.724574408	0.378831247	0.043047475
110	0.5	0.000869927	1.439692915	0.37909613	0.071749495
110	0.7	0	0.848922598	0.378353642	0.09979899
110	0.9	0.000736893	0.456306619	0.380571077	0.128852525
110	1	0.000931426	0.630048011	0.381758543	0.143659596

表 5 仿真时长为 1000s, 节点数为 160 的模拟数据

节点数	发消息频度	冲突率	时延率	吞吐率	承载率
160	0.1	0	0.698144222	0.351591536	0.013270139
160	0.3	0	1.131931482	0.355134392	0.040170139
160	0.5	0	0.595880504	0.35310472	0.066765972
160	0.7	0.000739643	0.557012704	0.353367745	0.093228472
160	0.9	0.000621401	1.76325103	0.35519512	0.12106875
160	1	0.000705272	2.639349453	0.355794002	0.134177083

表 6 仿真时长为 1000s, 节点数为 210 的模拟数据

节点数	发消息频度	冲突率	时延率	吞吐率	承载率
210	0.1	0	2.19902122	0.35092071	0.013249206
210	0.3	0	0.911162555	0.349439511	0.039534921
210	0.5	0	1.843725072	0.350296346	0.06648254
210	0.7	0	0.217971327	0.349703841	0.092620635
210	0.9	1.39926E-05	0.870204348	0.350464478	0.119268254
210	1	0.000715366	2.803121263	0.351625322	0.132895767

表 7 仿真时长为 1000s, 节点数为 260 的模拟数据

节点数	发消息频度	冲突率	时延率	吞吐率	承载率
260	0.1	0	1.57116132	0.368208845	0.01389359
260	0.3	0.000257962	1.218781716	0.369404981	0.041614103
260	0.5	0.000452375	1.0781204	0.371275786	0.070498291
260	0.7	0.000389244	2.615649557	0.373255701	0.098351282
260	0.9	0	1.413706261	0.37154412	0.126342308
260	1	0	1.502370876	0.370791524	0.140103419

表 8 仿真时长为 1000s, 节点数为 300 的模拟数据

节点数	发消息频度	冲突率	时延率	吞吐率	承载率
300	0.1	0	1.654786595	0.363894355	0.01363
300	0.3	0	0.53351015	0.363963432	0.041065185
300	0.5	0	1.584902457	0.364364686	0.068927407
300	0.7	0.000378458	2.279867218	0.364997155	0.096443333
300	0.9	0.000141994	2.512373153	0.367554201	0.124632593
300	1	0.00064076	1.616284367	0.363799805	0.137524815

说明: 我们在仿真的过程中还模拟了仿真时长分别为 2000s,4000s 的随机过程, 仿真数据不在此处展示。