

# 第十三届中国数学建模网络挑战赛

地址：数学中国数学建模网络挑战赛组委会  
电话：0471-4969085

邮编：010021

网址：[www.tzmcm.cn](http://www.tzmcm.cn)  
Email: service@tzmcm.cn

## 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 基于蒙特卡洛模拟仿真的广播通信模型

关 键 词 无线网通信模型 典型指标法 蒙特卡洛模拟 MATLAB 仿真

### 摘 要：

广播是无线传感器网络最基础的通信方式之一，它的目的是将信息从网络中源节点发送给网络中其余的每一个节点。本文针对无线网节点间广播通信的问题，建立指标选取模型构造一个衡量通信效率的指标，并以仿真模拟的方法建立完整的网络通信模型。

针对问题一，要求选取 1 个合理的指标来描述网络的通信效率，是指标构造问题。首先，建立影响网络通信效率的整体指标体系，包括传输时延、冲突率、传输速率、传输距离、利用率、吞吐量以及带宽 7 个指标；然后，利用单相关系数选取典型指标的方法建立指标选取模型，计算得到指标间的相关系数矩阵  $R$ ；之后，通过比较每一指标与其余指标的决定系数的平均值  $\bar{r}_i^2$  的大小，得到  $\bar{r}_i^2$  最大的 2 个指标：冲突率和利用率，其结果分别为 0.5423, 0.6917。最后，综合 2 个指标构造出 1 个评价通信效率的总指标  $E$ 。

针对问题二，将通信过程分为三个阶段分别设计用于通信的内置算法，分别为：发送前，发送中，碰撞处理。对于阶段一，采取载波监听的策略；对于阶段二，采用碰撞避免的方法；对于阶段三，分轻载和重载两种情况提出 2 种方案解决冲突问题，轻载时，采用 1 坚持退避算法，重载时，采用截断二进制指数算法；然后基于此内置算法，利用蒙特卡洛模拟的方法，建立完整的通信模型，利用 MATLAB 软件实现仿真；最后，在问题一的基础上，对仿真结果进行分析，得到上述方案能使网络的通信效率尽可能高的结论。

最后，将模型应用于无线网广播通信。对模型进行客观评价，对模型的实用性进行评估，并给出针对模型缺陷的改进方案。

参赛队号： 32507

所选题目： B 题

参赛密码 \_\_\_\_\_  
(由组委会填写)

# 第十三届中国数学建模网络挑战赛

地址：数学中国数学建模网络挑战赛组委会  
电话：0471-4969085

邮编：010021

网址：[www.tzmcm.cn](http://www.tzmcm.cn)  
Email: [service@tzmcm.cn](mailto:service@tzmcm.cn)

---

## Abstract

Broadcast is one of the most basic means of communication in wireless sensor networks. Its purpose is to send information from the source node in the network to every other node in the network. In this paper, aiming at the problem of wireless network broadcast communication between nodes, an index selection model is established to construct an index to measure the communication efficiency, and a complete network communication model is established by means of simulation.

In the first question, it is a problem of index construction to select one reasonable index to describe the communication efficiency of network. Firstly, the overall index system that affects the network communication efficiency is established, including 7 indicators of transmission delay, conflict rate, transmission rate, transmission distance, utilization rate, throughput and bandwidth. Secondly, the model of index selection is established by using the method of selecting typical index by single phase relation number, and the correlation coefficient matrix  $R$  is calculated. After that, by comparing the average value of the determination coefficient of each indicator with that of other indicators, the two largest indicators are obtained: conflict rate and utilization rate. The results were 0.5423 and 0.6917 respectively. Finally, a total index  $E$  is constructed to evaluate the communication efficiency.

In the second question, the communication process is divided into three stages to design the built-in algorithm for communication. The three stages include: before sending, during sending and collision handling. For the first stage, carrier monitoring strategy is adopted. For the second stage, collision avoidance method is adopted. For the third stage, two solutions are proposed to solve the conflict problem in light load and heavy load cases. For light load cases, 1 stick to retreat algorithm is adopted, while for heavy load cases, truncation binary exponential algorithm is adopted. Then, based on the built-in algorithm, the Monte Carlo simulation method is used to establish a complete communication model, achieving simulation by MATLAB. Finally, on the basis of problem 1, the simulation results are analyzed, and the conclusion is drawn that the above scheme can make the network communication efficiency as high as possible.

Furthermore, the model is applied to wireless network broadcast communication. Besides, it is evaluated objectively, its practicability is evaluated, and the improvement plan is given.

## 目录

一、 问题重述 .....	1
1.1 问题的背景 .....	1
1.2 问题的提出 .....	1
二、 问题分析 .....	1
2.1 问题一的分析 .....	1
2.2 问题二的分析 .....	1
三、 问题假设 .....	2
四、 符号说明 .....	3
五、 模型的建立、求解与分析 .....	4
5.1 问题一模型的建立、求解与分析 .....	4
5.1.1 模型准备 .....	4
5.1.2 基于典型指标法建立的指标筛选模型 .....	5
5.1.3 模型的求解 .....	6
5.2 问题二模型的建立、求解与分析 .....	7
5.2.1 内置算法的设计 .....	7
5.2.2 蒙特卡洛仿真模拟实现通信模型 .....	9
5.2.3 模型的求解与验证 .....	10
六、 模型的评价与推广 .....	16
6.1 模型的优点 .....	16
6.2 模型的缺点 .....	16
6.3 模型的推广与改进 .....	17
七、 参考文献 .....	18
八、 附录 .....	19
附录 1: 476~1000 $\mu s$ 无线网信息传递拓扑图 .....	19
附录 2: MATLAB 仿真模拟程序代码 .....	19

## 一、 问题重述

### 1.1 问题的背景

以广播的方式来进行无线网通信，必须解决发送互相冲突的问题。我们考虑这样的一个无线网：每个通信节点都是低功率的发射器，并且在进行着空间上的低速连续运动，每个节点只有和它距离在一定范围之内的节点才能收到它的信号，而且节点会相互接近或远离。此外，每个节点需要不定期地、断续地发送信息，但会时刻保持收听信息。发送和收听工作可以同时进行。在这个通信网络中，完全没有网络的基础设施，而且每个节点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。所有信息只能使用同一个频率发送，一旦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。

### 1.2 问题的提出

根据以上背景，需要建立合理的数学模型，解决以下问题：

问题 1：请构造一个合理的指标来描述网络的整体通信效率。

问题 2：在每个节点需要发送信息时，请设计一个方案来选择发送的时间段。如果发送信息时发现存在冲突，如何自动调整重发的时间段，才能使网络的整体通信效率尽可能高？可能需要分情况（例如分轻载/重载等不同情形，或其他分法）来考虑问题。

## 二、 问题分析

### 2.1 问题一的分析

对于问题一，题目要求构造一个合理的指标描述整体的通信效率。由参考文献分析可知<sup>[1]</sup>，评价一个网络的整体通行效率的影响因素有很多，主要可以聚类为两大类：基本通信性能指标和网络性能指标。其中基本通信性能指标包括：传播时延、冲突率、传输速率、传播距离；网络性能指标包括：利用率、吞吐量、带宽。由于考虑的指标过多，对于上述指标我们需要选取最具代表性的作为评价指标，可以从这两大类中分别选取最具代表性的一个指标，最后将其形成综合指标用来评价网络通信效率。

其中已知的指标选取方法有条件广义方差极小法、极大不相关法、典型指标法等，典型指标法相对于前面两种方法计算量较小且计算简单，所以我们采用单相关系数选取典型指标的方法，基于此，需要分析各个指标之间的相关性，从而计算得到相关系数矩阵，通过计算，得到可用于描述通信效率的典型指标。

### 2.2 问题二的分析

对于问题二，题目要求设计一个方案来选择发送的时间段，并且当存在信息冲突时，设置调整重发的时间段，使网络的整体通信效率尽可能高。首先，从问题分析中可知所有节点都拥有同样的内置算法，所以我们针对节点发送信息的整体过程设计一个内置算法：从信息发送前、信息发送中、碰撞处理三个阶段进行讨论，确定每个节点的发送时间段。

其中，对信息发送前这一阶段利用载波监听的方法，即发送前先监听；对信息发送中这一阶段采用碰撞检测的方法，即边发送边监听；最后针对碰撞处理这一阶段，当监听到发生冲突时，为了使网络的整体通信效率尽可能高，我们使用截断二进制指数退避算法来解决碰撞问题。

此外，在碰撞处理的过程中还需要分两种情况进行讨论：情况一，当网络属于轻载的情形时；情况二，当网络属于重载的情形时。最后，针对这两种不同的情况设计出完善合理的解决方案，选择信息发送的时间段，再基于问题一求出的评价网络通信效率的指标验证所求方案是否高效。



图 1 思维导图

### 三、 问题假设

假设 1：网络中信息的发送不考虑处理时延与排队时延，即总时延等于传播时延与发送时延之和；

假设 2：设发送每条信息所占用的时间都是等长的；

假设 3：所有的节点都拥有同样的内置算法，并拥有一个校准过的高精度时钟；

假设 4：在信道不发生冲突的情况下，每次都可以成功发送，忽略丢包率和误码率；

假设 5：每个节点收到信息后都可以立即广播出去，忽略转播时延。

#### 四、 符号说明

符号	符号意义
$T_{delay}$	传输时延
$\sigma$	发送时延
$\tau$	传播时延
$\zeta$	冲突率
$U$	利用率
$count'$	单位时间碰撞次数
$\lambda$	单位时间发送信息次数
$t_{busy}$	信道忙碌的时间
$t_{sum}$	任务完成的总时间
$r_{ij}$	指标 $x_i$ 与指标 $x_j$ 的相关系数
$E$	描述网络通信效率的指标
$2\tau$	争用期
$N$	网络中节点个数
$L$	轻载状态
$W$	重载状态
$X$	仿真次数
$T$	节点随机发送信息的时间范围
$m$	随机生成发送信息的节点个数
$P_{area}$	生成属于范围内的节点的概率

## 五、模型的建立、求解与分析

### 5.1 问题一模型的建立、求解与分析

本问结合评价网络通信效率的指标需求，我们从基本通信性能指标和网络性能指标 2 个方面构建评估指标体系。其中基本通信性能指标包括：传播时延、冲突率、传输速率、传播距离；网络性能指标包括：利用率、吞吐量、带宽。然后，利用单相关系数选取典型指标的方法选择用来描述网络的整体通信效率的一个指标。

#### 5.1.1 模型准备

##### (1) 通信效率的指标体系构建：

随着网络信息技术的发展，采用无线网络通信技术进行数据传输和信号通信成为未来保密通信传输的重点发展趋势<sup>[2]</sup>。通信系统中的通信环境复杂，影响通信性能的因素众多，找到用于描述网络整体通信效率的指标至关重要。因此，通信网络的建设不仅需要考虑传播时延、冲突率、传输速率、传播距离等基本通信性能指标，而且需将利用率、吞吐量、带宽等网络性能指标纳入综合评估范围。指标体系如图 2 所示。

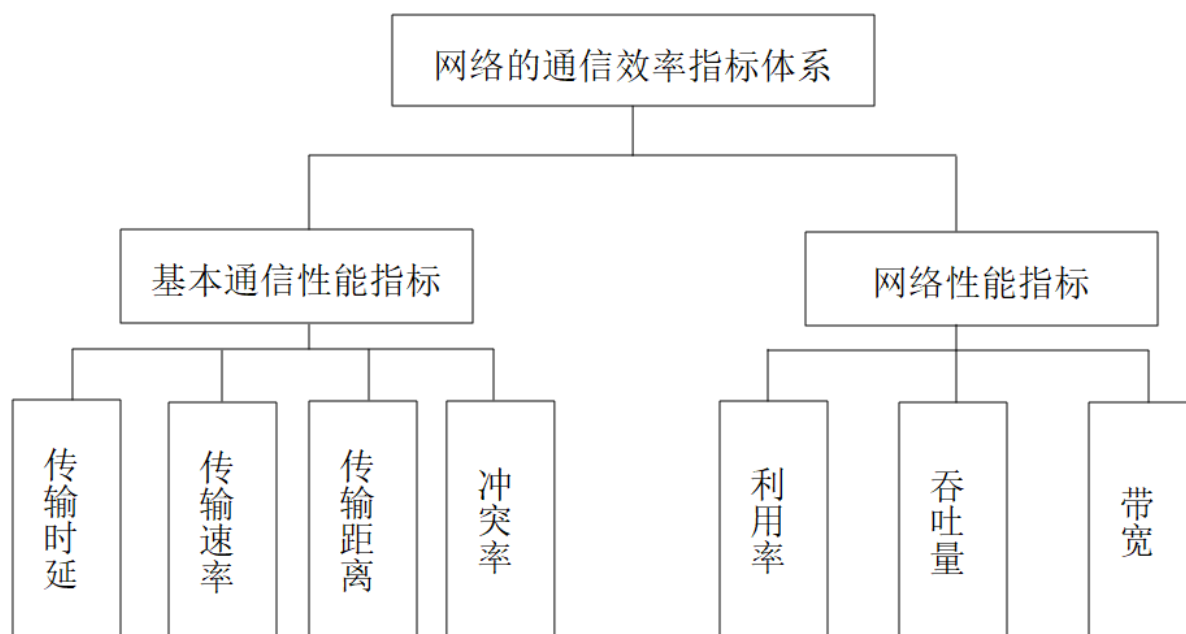


图 2 网络通信效率的指标体系

##### (2) 对于各个指标的解释：

**传输时延：**由假设 1 可知，该传输时延表示传播时延与发送时延的总和。发送时延是每个节点发送数据帧所需要的时间，也就是从发送数据帧的第一个比特算起，到该帧的最后一个比特发送完毕所需的时间；传播时延是信息在信道中传播一定的距离需要花费的时间，又由假设 2 可知，该无线网的发送时延和传播时延是固定不变的，即总时延

不变。时延的计算公式见式 (1)

$$\begin{cases} T_{delay} = \sigma + \tau \\ \sigma = \frac{b}{\omega} \\ \tau = \frac{m}{v} \end{cases} \quad (1)$$

式 (1) 中  $T_{delay}$  代表总的传输时延,  $\sigma$  表示发送时延,  $\tau$  表示传播时延,  $b$  代表数据帧长度,  $\omega$  表示信道带宽 ( $b/s$ ),  $m$  表示信道长度,  $v$  表示信息在信道上的传播速率 ( $m/s$ ).

**冲突率:** 在这里冲突率表示单位时间碰撞次数  $count'$  与单位时间发送次数  $\lambda$  的比值, 冲突率的计算公式见式 (2), 冲突率用  $\zeta$  表示。

$$\zeta = \frac{count'}{\lambda} \quad (2)$$

**传输速率:** 表示节点在信道上传送数据的速率, 用  $v$  表示。

**传输距离:** 表示一个节点信息发送到目标节点通过的距离。

**利用率:** 表示信道忙的时间  $t_{busy}$  与总任务完成时间  $t_{sum}$  的比值, 利用率的计算公式见式 (3), 利用率用  $U$  进行表示。

$$U = \frac{t_{busy}}{t_{sum}} \quad (3)$$

**吞吐量:** 表示在单位时间内通过网络的数据量。吞吐量会受到网络的带宽或网络的额定速率的限制。

**带宽:** 用于表示网络的通信线路所能传送数据的能力, 因此网络带宽表示在单位时间内从网络中的某一点到另一点所能通过的“最高数据率”。

### 5.1.2 基于典型指标法建立的指标筛选模型

若评价指标过多, 则可将相近的指标聚成类, 本问主要分为两大类: 基本通信性能和网络性能, 然后, 在每一类中选取若干个典型指标, 其中可用的方法有条件广义方差极小或极大不相关的方法, 但是计算量太大。用单相关系数选取典型指标法, 该方法更加简单<sup>[3]</sup>。具体步骤如下:

**Step1:** 假设为同一类的指标的个数为  $n$ , 分别为  $a_1, a_2, a_3, \dots, a_n$ . 这里一共有两类, 第一类的指标个数为4, 第二类的指标个数为3. 首先针对每一类分别计算各类中  $n$  个指标之间的相关系数矩阵  $R$ , 为对称矩阵。

$$R_1 = \begin{pmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ r_{41} & r_{42} & r_{43} & r_{44} \end{pmatrix} \quad R_2 = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$$

其中  $R_1$  为基本通信性能指标的关系矩阵,  $R_2$  为网络性能指标的关系矩阵,  $r_{ij}$  表示指



标  $x_i$  与指标  $x_j$  的相关系数。相关公式见式 (4)， $s_{ij}$  是指标  $x_i$  与指标  $x_j$  的协方差， $x_{k,i}$  表示样本均值。

$$\begin{cases} r_{ij} = \frac{S_{ij}}{\sqrt{S_{ii} \cdot S_{jj}}} \\ S_{ij} = \frac{1}{m} \sum_{k=1}^m (x_{ki} - \bar{x}_i)(x_{kj} - \bar{x}_j) \\ \bar{x}_i = \frac{1}{n} \sum_{k=1}^m x_{k,i} \end{cases} \quad (4)$$

Step2: 计算每一指标与其它  $n-1$  个指标的决定系数(相关系数的平方)的平均值  $\bar{r}_i^2$ ，即

$$\bar{r}_i^2 = \frac{1}{n-1} \left( \sum_{j=1}^n r_{ij}^2 - 1 \right), i=1, 2, \dots, n. \quad (5)$$

$\bar{r}_i^2$  反映了  $x_i$  与其它  $n-1$  个指标的相关程度。之所以用相关系数的平方，是为了防止相关系数为负，无法直接相加求平均。若相关系数均为正，则可以直接用相关系数。

Step3: 比较  $\bar{r}_i^2$  的大小，令  $\bar{r}_i^2 = \max_{1 \leq i \leq n} \bar{r}_i^2$ ，则选  $x_k$  作为  $a_1, a_2, a_3, \dots, a_n$  的典型指标。如果还需继续选择指标，可在余下的  $n-1$  个指标里继续选取。(此时，相关系数的矩阵为原来的矩阵去掉第  $k$  行，第  $k$  列后剩下的  $n-1$  阶矩阵。)

### 5.1.3 模型的求解

通过上述模型的建立，首先对第一类反映基本通信性能指标：传输时延，冲突率，传输速率以及传输距离( $a_1, a_2, a_3, a_4$ )，写出它的相关系数矩阵  $R_1$

$$R_1 = \begin{pmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ r_{41} & r_{42} & r_{43} & r_{44} \end{pmatrix} = \begin{pmatrix} 1 & 0.5598 & -0.6545 & 0.5587 \\ & 1 & 0.8655 & 0.7512 \\ & & 1 & 0.5433 \\ & & & 1 \end{pmatrix}$$

求出  $a_i$  与其余 3 个指标的相关系数的平方  $\bar{r}_i^2, i=1, 2, 3, 4$ .

$$\bar{r}_1^2 = \frac{1}{4-1} (1^2 + 0.5598^2 + (-0.6545)^2 + 0.5587^2 - 1) = 0.3513$$

$$\bar{r}_2^2 = \frac{1}{4-1} (1^2 + 0.5598^2 + 0.8655^2 + 0.7512^2 - 1) = 0.5423$$

$$\bar{r}_3^2 = \frac{1}{4-1} (1^2 + (-0.6545)^2 + 0.8655^2 + 0.5433^2 - 1) = 0.4909$$

$$\bar{r}_4^2 = \frac{1}{4-1} (1^2 + 0.5587^2 + 0.7512^2 + 0.5433^2 - 1) = 0.3905$$

比较  $\bar{r}_i^2$  的大小，发现  $\bar{r}_2^2$  最大，故选  $a_2$  (冲突率) 作为  $a_1, a_3, a_4$  的典型指标。

然后，对第二类反映网络性能的指标：利用率，吞吐量以及带宽 $(b_1, b_2, b_3)$ ，写出它的相关系数矩阵 $R_2$

$$R_2 = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} = \begin{pmatrix} 1 & 0.8854 & 0.7742 \\ & 1 & 0.7623 \\ & & 1 \end{pmatrix}$$

求出 $b_i$ 与其余 2 个指标的相关系数的平方 $\bar{r}_i^2, i=1,2,3$ .

$$\bar{r}_1^2 = \frac{1}{3-1}(1^2 + 0.8854^2 + 0.7742^2 - 1) = 0.6917$$

$$\bar{r}_2^2 = \frac{1}{3-1}(1^2 + 0.8854^2 + 0.7623^2 - 1) = 0.6825$$

$$\bar{r}_3^2 = \frac{1}{3-1}(1^2 + 0.7742^2 + 0.7623^2 - 1) = 0.5902$$

比较 $\bar{r}_i^2$ 的大小，发现 $\bar{r}_1^2$ 最大，故选 $b_1$ （利用率）作为 $b_2, b_3$ 的典型指标。

表 1 指标的相关系数平方

基本通信性能指标		网络性能指标	
传输时延	0.3513	利用率	0.6917
冲突率	0.5423	吞吐量	0.6825
传输速率	0.4909	带宽	0.5902
传输距离	0.3905		

综上所述，观察表 1 可知，利用单相关系数选取典型指标的方法，我们选取冲突率 $\zeta$ ，即单位时间碰撞次数 $count'$ 与单位时间发送次数 $\lambda$ 的比值，以及利用率 $U$ ，即信道忙的时间 $t_{busy}$ 与总任务完成时间 $t_{sum}$ 的比值，这 2 个指标作为评价指标。

为了综合 2 个指标对网络的整体通信效率的影响，我们将把冲突率与利用率结合在一起，综合考虑得出用于描述通信效率的 1 个总指标 $E$ ，如式（6）：

$$E = \max(U - \zeta) \quad (6)$$

由此，构造出一个合理的指标 $E$ 来描述网络的整体通信效率。

## 5.2 问题二模型的建立、求解与分析

问题二要建立完整的网络通信模型，整个模型包括发送前、发送中、碰撞处理三部分，对这三部分分别采用载波监听、碰撞检测以及截断二进制指数的方法，设计一个节点发送信息的内置算法，得到信息发送的时间段。最后，在问题一的基础上利用蒙特卡洛仿真模拟的方法，验证这一设计方案是否能使网络的整体通信效率较高。

### 5.2.1 内置算法的设计

通过问题分析可知，每个通信节点需要不定期地、断续地发送信息，并时刻保持收听信息，所有信息只能使用同一个频率发送，当有两个或多个节点进行广播时，便发生冲突，基于此，我们需要设计一个算法来解决碰撞问题，确定信息重发的时间段。以下

是实现内置算法的具体步骤：

*Step1*:信息发送前，进行载波监听，每一个节点在数据发送之前先要检测一下总线上是否有其他节点在发送数据，如果信道忙，则一直监测信道，直至空闲；

*Step2*:信息发送中，进行碰撞避免，由于数据帧在信道中传输时有传播时延 $\tau$ ，可能存在两个及以上的用户同时发送数据，从而造成信道冲突，因此，节点每次发送信息时需要边发送边进行监测，若监测到碰撞出现，就立即停止发送；

*Step3*:随机回退，当节点监测到碰撞后，需要分轻载和重载两种情况进行讨论，所以设计两种预选方案：

方案一：采用 1 坚持退避算法，即信道忙则持续监听，一旦信道由忙变为空闲则立即发送数据，若有两个及两个以上的节点时，则随机选取一个节点进行数据传送的操作。

方案二：为降低再次发生碰撞的概率，不是等待信道变为空闲后就立即再发送数据，而是利用截断二进制指数算法计算出一个推迟时间，一般为 $r$ 倍的争用期。其中具体的退避算法如下：

(1) 确定基本退避时间，通常采用端到端的往返传播时延为基准单位，即争用期 $2\tau$ ；

(2) 从离散的整数集合 $[0,1,...,(2^k-1)]$ 中随机取出一个数，记为 $r$ 。重传的推迟时间就为 $r$ 倍的争用期；

(3) 定义参数 $k$ 按公式(7)计算：

$$k = \text{Min}[\text{重传次数}, 10] \quad (7)$$

可见当重传次数不超过 10 时，参数 $k$ 等于重传次数；当重传次数超过 10 时， $k$ 就不再增大而一直等于 10。内置算法见图 3 所示：

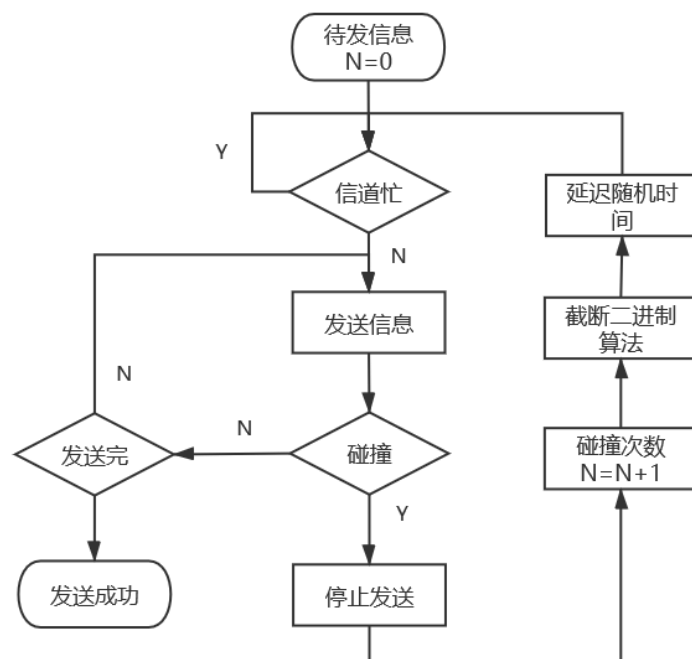


图 3 节点信息发送的内置算法

### 5.2.2 蒙特卡洛仿真模拟实现通信模型

上文讨论的基于截断二进制指数策略，采用这种技术，可以动态地适应试图发送的节点数的变化，使随机等待时间随着冲突的产生按指数递增，既可保证轻载时的响应时间，又可保证重载时的系统稳定性<sup>[4]</sup>。

利用 MATLAB 编程对上述通信模型的仿真实现中，其主要包括：（1）信道状态的判断；（2）截断二进制指数退避算法处理。信道有三种状态：空闲、成功传输及冲突。对未发送数据的节点而言，其只需要判断信道是否空闲；发送数据的节点需要在送数据后  $r$  倍的争用期才能判断信道的状态<sup>[5]</sup>。在仿真程序设计中，可以通过监测同一时刻信道中进行数据传送的节点数  $m$ ，有以下三种情况：

$$\begin{cases} m \geq 2, \text{信道发生冲突} \\ m = 1, \text{信道成功传输} \\ m = 0, \text{信道空闲} \end{cases}$$

利用蒙特卡洛仿真实实现的具体过程如下：假设节点个数为  $N$ 。蒙特卡罗方法的实质是按一定概率分布产生随机数的方法来模拟项目中实际可能出现的随机现象，并统计出它们的数据特征，从而得到实际问题的数值解<sup>[6]</sup>。

*Step1*:由题可知，为实现每个通信节点在空间上进行着低速连续运动，并且每个节点只和它距离在一定范围之内节点才能收到它的信号这一条件。以概率  $P_{area}$  随机生成属于范围内的节点  $N_{in} = \{n_{x1}, n_{x2}, n_{x3}, \dots, n_{xj}\}, j \leq N$ ,

$$Y = \begin{cases} 0, \text{不在范围内} \\ 1, \text{在范围内} \end{cases}$$

*Step2*:随机生成某一时刻，有  $m(0 \leq m \leq n)$  个节点想要发送信息，针对轻载和重载两种不同的情况，按照以下概率生成  $m$ ：

$$\begin{cases} m = 0, p_0 \\ m = 1, p_1 \\ m \geq 2, p_2 \\ p_0 = p_1 > p_2, N = L \end{cases}$$

*Step3*:根据  $m$  的值，若  $m = 0$  则不选取节点，若  $m \geq 1$  则随机选择  $m$  个节点序号；

*Step4*:随机生成每个节点发送的目标节点序号；

*Step5*:根据目标序号，判断该目标序号是否属于范围内，若  $Y=0$  表示不在范围内则放弃发送并使  $m$  减 1；若  $Y=1$  表示在范围内则可以发送，然后再判断  $m$  的值，如果  $m = 1$ ，则直接发送， $m > 1$ ，则根据轻载和重载的不同情况执行其内置算法。

碰撞次数的判定：

（1）设置初始碰撞次数  $count = 0$

（2） $m = 1$ ，在当前节点发送信息的时间段内，若有其他节点也想要发送，则碰撞次数  $count + 1$ 。

(3)  $m > 1$ ，即在同一时刻有两个或两个以上节点想要发送信息，则碰撞次数  $count + 1$ ，再执行内置算法。

整体的仿真流程见图 4 所示：

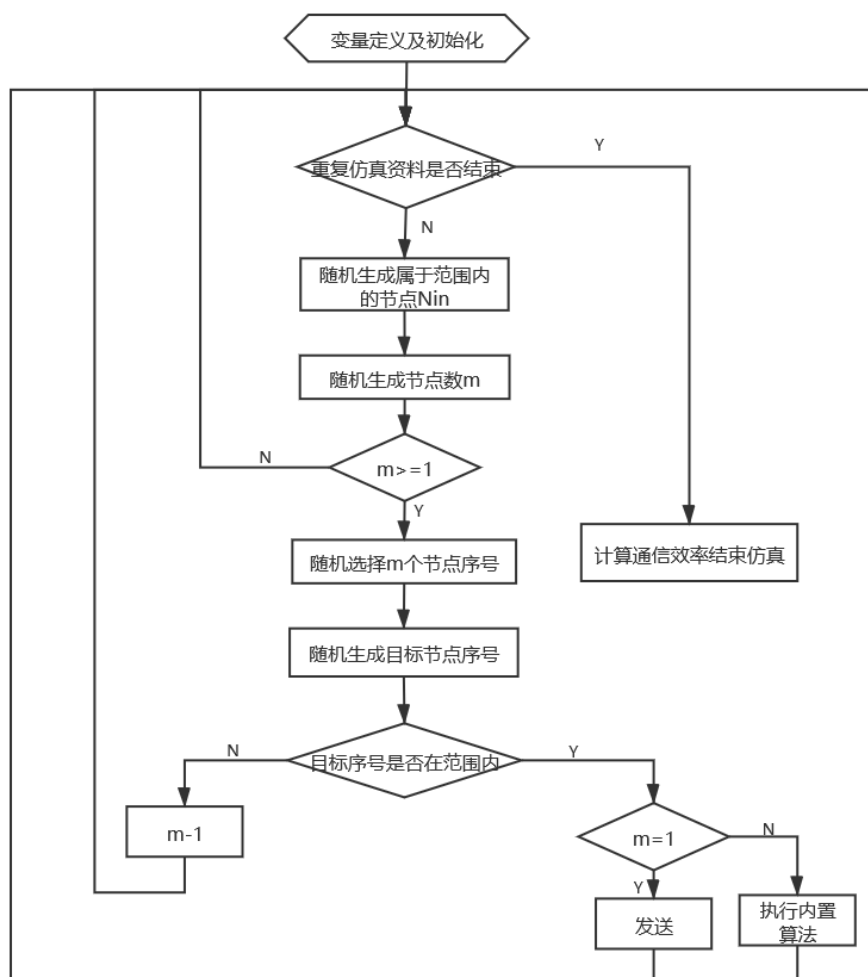


图 4 仿真流程

### 5.2.3 模型的求解与验证

#### (1) 方案的确定：

在问题一得到的用于评价通信效率的指标的基础上，采用蒙特卡洛仿真模拟的方法，利用 MATLAB 对整个通信过程进行仿真，并且根据节点数目的不同，我们将网络状态分为：轻载状态  $L$  和重载状态  $W$ ；根据信息在信道发送的状态不同，我们将信道状态分为：空闲状态  $I$ ，发送成功状态  $S$ ，冲突状态  $Q$ 。

MATLAB 软件已经广泛应用于通信仿真，所以在通信领域，它具有极其重要的地位。MATLAB 仿真的优势是函数库庞大、程序开发简单、调试方便、作图功能强大，容易实现离散时间及时间驱动的处理，对调整算法参数提供很大的便利，所以用它实现上述的仿真流程十分适用。

定义仿真的具体参数，见表 2：

表 2 仿真参数

节点数目	争用期	传播时延	发送时延	仿真次数	固定段时间
$N$	$2\tau$	$\tau$	$\sigma$	$X$	$T$

注：固定段时间，代表程序仿真的节点随机发送信息的时间范围

利用上述设计的内置算法与仿真流程，结合定义的仿真参数，通过 MATLAB 编写仿真程序模拟通信模型，计算得到该网络冲突率与利用率随节点数不同的数值，作出冲突率-节点数变化曲线（图 5）和利用率-节点数变化曲线（图 6），用于评价通信效率。具体仿真结果如下：

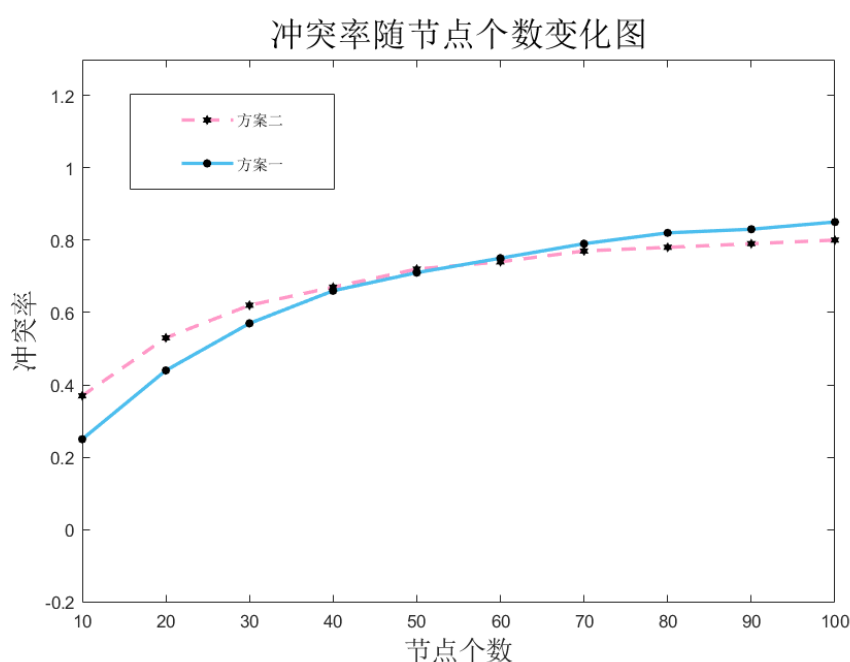


图 5 冲突率-节点数变化曲线图

观察图 5 可知，图中两条不同颜色的曲线，代表的是上述内置算法针对轻载和重载状态所设计的两种碰撞处理方案：

方案一：采用 1 坚持退避算法，即信道忙则持续监听，一旦信道由忙变为空闲则立即发送数据，若有两个及两个以上的节点时，则随机选取一个节点进行数据传送的操作。

方案二：为降低再次发生碰撞的概率，不是等待信道变为空闲后就立即再发送数据，而是利用截断二进制指数算法计算出一个推迟时间，一般为  $r$  倍的争用期。

从图上可以看出采用不同的碰撞处理方案，它的冲突率都会随着节点数的增加而增加，并逐渐趋于平稳，但对应的网络冲突率不同，当节点数  $N < 50$  的时候，采用方案一计算得出的冲突率小于方案二的冲突率，即说明 1 坚持退避算法的冲突率小于截断二进制指数退避算法的冲突率；而当节点数  $N \geq 50$  的时候，方案二的冲突率小于方案一的冲

突率，说明这种情况下，方案二优于方案一。

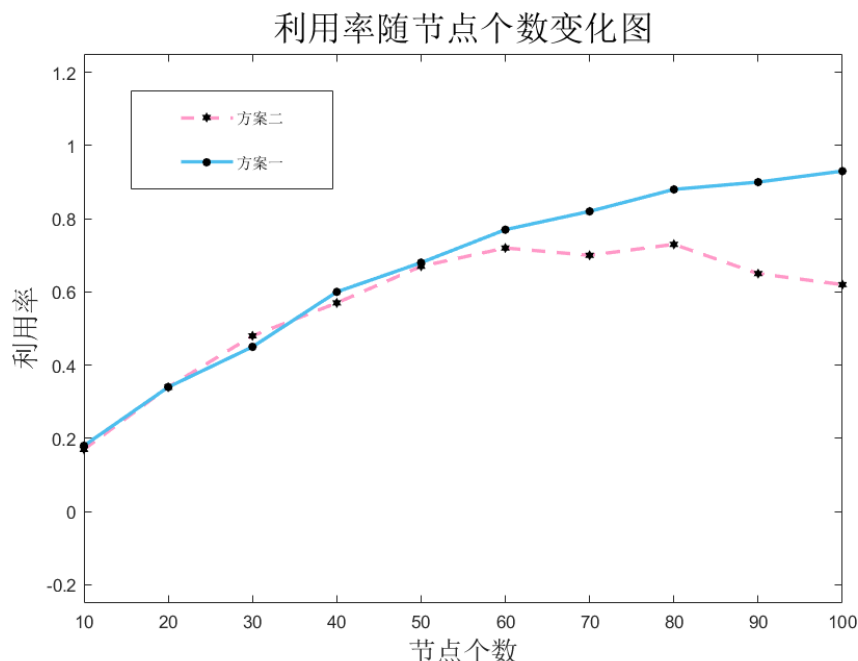


图 6 利用率-节点数变化曲线图

由图 6 可以看出，采用不同的方案，方案一的利用率会随着节点数的增加而无限接近于 1；而方案二的利用率随着节点数的增加，在  $N \geq 80$  时，随节点数的增加网络利用率逐渐呈下降趋势。此外，我们还可以看出当  $N < 50$  时，采用方案一和方案二时网络的利用率几乎相同；当  $N \geq 50$  时，明显看出方案一的利用率要高于采用方案二时网络的利用率。

综合图 5 和图 6 可知，就利用率而言，方案一明显优于方案二，但由于冲突率对网络通信的影响十分重要，无法忽略，即使利用率很高，但冲突率也很高的情况下，网络的通信效率就很低。所以就冲突率而言，当节点数目较少时，采用方案一更优，但实际的网络通信情况中节点数都较多，则采用方案二更优。所以综合两种情况而言，当网络处于轻载状态时采用方案一进行碰撞处理；当网络处于重在状态时采用方案二进行碰撞处理。

表 3 N 不同时方案的选取

轻载	方案一
重载	方案二

## (2) 通信模型的仿真模拟：

通过分析冲突率与利用率随节点数的变化，得出选择哪种方案的结论。下面通过表 3 得到的结论，再次进行多次仿真，计算网络通信效率的指标  $E$ （同冲突率和利用率均有关系）的值，在仿真过程中，我们将节点发送信息的时间  $T$  固定为一段时间，通过作图观察不同仿真次数下， $E$  值的变化，见图 7，图 8。

其中图 7 是当节点数  $N=50$  时，此时网络属于轻载的状态，根据表 3 可知采用方案一进行仿真得到的结果；图 8 是当节点数  $N=100$  时，此时网络属于重载状态，应采用方案二进行仿真模拟。

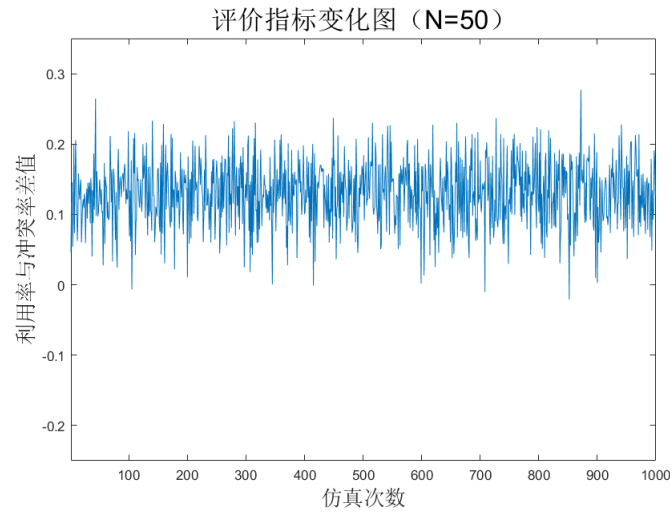


图 7  $N=50$  时  $E$  值的变化图

表 4  $N=50$  时仿真结果

仿真次数	100	200	300	400	500	600	700	800	900	1000
平均 $E$ 值	13.33%	13.21%	13.69%	13.30%	13.19%	13.16%	13.45%	13.02%	13.28%	13.12%

由于论文篇幅限制，我们仅选取部分仿真结果进行展示（表 4）。表 4 是当仿真次数  $X=100, 200, 300, \dots, 1000$  时，所计算的平均  $E$  值，仿真次数不同，其值略微有变化，但总体平均  $E$  值大约为 13%，因为在仿真过程中我们所设置的固定段时间比较小，导致求出的  $E$  值比较小，但不影响总体的仿真过程。

从图 7 可以看出，评价指标  $E$  值随仿真次数呈震荡变化，在不同仿真次数下，其评价指标的值也会变化。

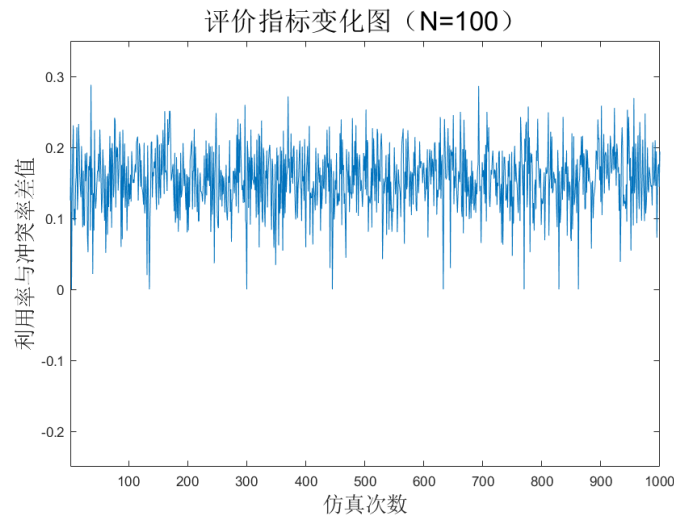


图 8  $N=100$  时  $E$  值的变化图



表 5 N=100 时仿真结果

仿真次数	100	200	300	400	500	600	700	800	900	1000
平均E值	15.66%	15.03%	15.26%	15.31%	15.59%	15.12%	15.27%	15.64%	15.15%	15.59%

表 5 是当  $N = 100$  时，采用方案二得到的平均  $E$  值的仿真结果，结合图 8 可以看出，与  $N = 50$  的结论类似，平均  $E$  值会随着仿真次数的变化做震荡运动，大约为 15%.

(3) 节点间信息传递的具体过程展示：

下面我们将无线网中节点传递信息的仿真模拟过程进行展示，见图 8 所示：

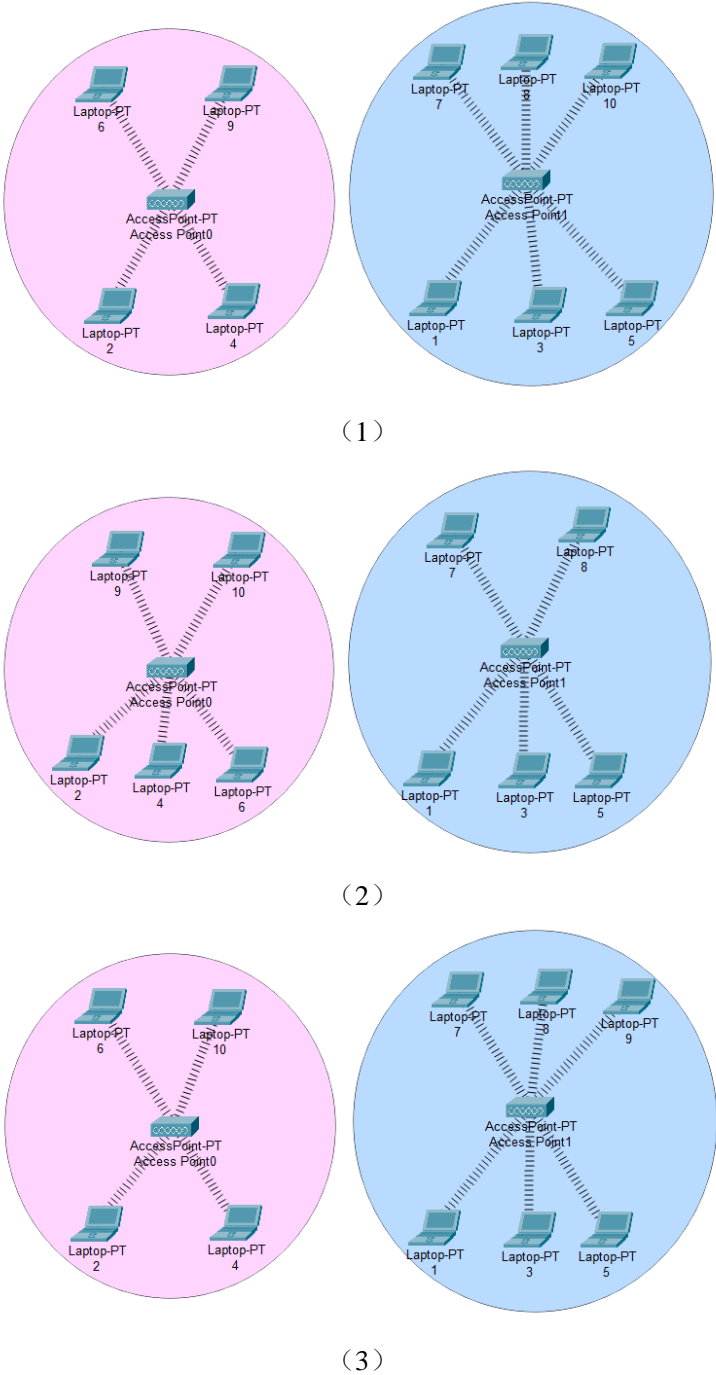


图 9 无线网信息传递拓扑图

图 9 是无线网的节点信息传递图，其中在同一圆圈内的节点属于同一范围，它们之间可以进行通信，而不在同一范围内的节点无法进行通信。假设在这个网络中每个节点至多只发送 1 次信息，在这里仅以 10 个节点为例，并且是在固定段时间  $T=1000\mu s$  的条件下进行模拟的，其中图（1）是  $0\sim 206\mu s$  内节点进行信息传递的拓扑图，图（2）是  $207\sim 223\mu s$  时间内信息传递的拓扑图，图（3）是  $224\sim 475\mu s$  时间内信息传递的拓扑图，剩下  $476\sim 1000\mu s$  时间内的拓扑图见附录 1.节点信息的具体发送过程见表 6 所示：

表 6 无线网中节点传递信息的具体过程及信道状态

	源主机	目的主机	初始欲发送时刻	最终发送时刻	原因
信息1	3	5	60	60	正常发送
信息2	10	2	177	207	初始时超出通信范围，后因为节点运动，在同一范围，可以进行通信
信息3	1	9	194	224	
信息4	6	4	410	410	
信息5	5	1	427	427	正常发送
信息6	8	5	476	476	正常发送
信息7	2	8	500	506	产生冲突
信息8	9	7	578	578	正常发送
信息9	7	4	818	818	正常发送
信息10	4	7	845	848	产生冲突

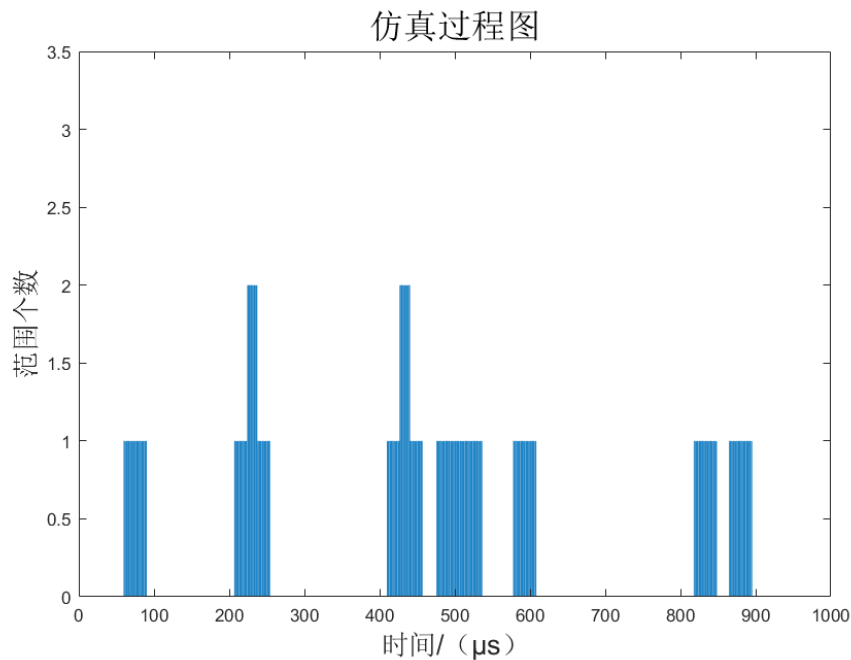


图 10 仿真过程的范围变化图

表 6 所示的是不同节点发送给不同目的节点的初始欲发送时刻和最终发送时刻，可以发现有两种情况：第一种是初始发送时刻=最终发送时刻；第二种是初始发送时刻 $\neq$ 最终发送时刻。造成这两种情况的原因包括：

(1) 若某一节点向目标节点能够正常发送信息，则满足第一种情况；

(2) 造成第二种情况出现的情形有两种：一种是由于信道内产生了冲突；另一种是某一节点向目标节点发送信息时，发现目标节点不在可发送范围内，后面因节点低速运动与目标节点处于同一范围内时即可发送；

分析表 6 我们还能得出以下结论：不在同一范围内的节点可以并发的与处于自己发送范围内的节点发送信息，例如，节点 1 与 9 同属于一个范围，节点 2 和 10 同属于另一个范围，这种情况下它们发送信息的过程是独立互不影响的，这也是这个无线网络的特点。

观察图 10 可知，随着时间的变化节点也在进行运动，各个节点可能相互远离或接近，这就有可能造成不同节点属于不同的范围，如图 10 所示，在 200~300  $\mu s$ 、400~500  $\mu s$  的时间间隔内，都可能出现网络被划分为两个范围的情况，而处于同一范围的节点间能够进行相互通信。

## 六、 模型的评价与推广

### 6.1 模型的优点

1.在问题一中，我们采用单相关系数选取典型指标法，相较于其他选取典型指标的方法，例如条件广义方差极小或极大不相关的方法，典型指标法的计算量相对较小，计算简单，过程清晰易懂。

2.对于问题二的内置算法，我们从 3 个阶段进行讨论设计，并且针对信道出现的碰撞状态，考虑网络轻载与重载两种情况，提出了不同的碰撞处理解决方案，使该算法全面完善，既可保证轻载时的响应时间，又可保证重载时的系统稳定性。

3.蒙特卡洛仿真模拟是建立一个随机过程，使其参数为问题所要求的解，然后通过模型或过程的观察或抽样试验来计算所求参数的统计特征<sup>[7]</sup>。对于本问中网络通信的各个参数无法确定的情况，利用蒙特卡洛仿真模拟的方法其可靠性高，并且十分适用。

4.在问题二的仿真模拟过程中，我们还考虑了由于节点不断运动而导致这些节点所属范围不同的情况，利用给定概率随机生成范围的方法进行模拟，易于程序的实现。并且，在模型的最后我们还展示了无线网节点信息传送的具体过程，使整个模型更清晰易懂，便于理解。

### 6.2 模型的缺点

1.典型指标法的计算量虽然较小且简单，但这个方法较为粗略。

2.在实际的无线传感器网络中，最简单的广播策略是泛洪每个节点将其收到的广播

分组转播给所有的邻居节点<sup>[8~11]</sup>。转播过程中可能存在转播延迟，但为了简化问题，本文没有考虑它的转播时延，所以还需在此基础上对模型进行改进，具体改进方法见模型的推广与改进部分。

### 6.3 模型的推广与改进

#### (1) 无冲突广播算法 CFBA 的推广<sup>[12]</sup>

该无冲突广播算法 CFBA 主要由两个步骤组成：

*Step1*: 构建最小连通树。它的主要思想是：对于任意网络  $G=(V,E)$ ，从节点集  $V=(v_1, v_2, v_3, \dots, v_n)$  选取一个最小  $m$  元子集  $P=(p_1, p_2, p_3, \dots, p_n)$ ，其中  $P \subset V$ ，使得  $V$  中的所有节点在  $P$  中的节点的一跳转传输范围之内。

*Step2*: 进行转播调度。就是将最小连通树中每层节点分成若干个最大冲突域，不同的冲突域的节点可以并发的进行广播，同一冲突域中的节点轮流进行转播，从而避免转播冲突的发生，增加广播的可达性，也加快广播数据覆盖整个网络，减少广播延迟。

#### (2) MLDB 算法的推广<sup>[13]</sup>

MLDB 算法是一种利用“转播时延”的机制来减小传统广播算法（OBM）中的转播冗余和能量消耗的算法。算法的具体步骤如下：

*Step1*: 等待接收广播的分组。如果收到广播分组，执行第 2 步，若没有收到，执行第（7）步；

*Step2*: 收到广播分组后，确定它是否为一个新分组。是，则执行第（3）步，若该节点已收到过该分组，则执行第（4）步；

*Step3*: 检查节点的剩余电量。若节点电量处于危险状态，则执行第（9）步；

*Step4*: 确定上游转播节点，并更新上游转播节点集；

*Step5*: 对节点的度进行更新。见图 11 所示：

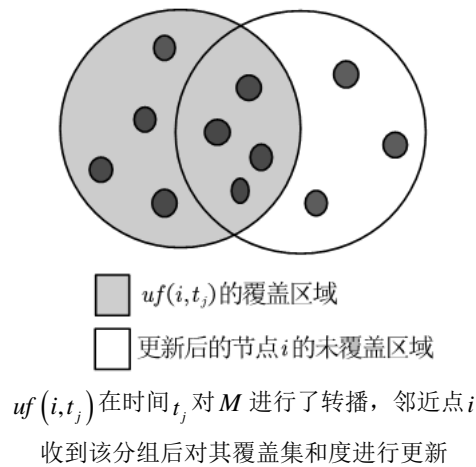


图 11 节点度的更新

*Step6*: 计算转播时延；

*Step7*:判断转播时延是否超时;

*Step8*:决定是否转播;

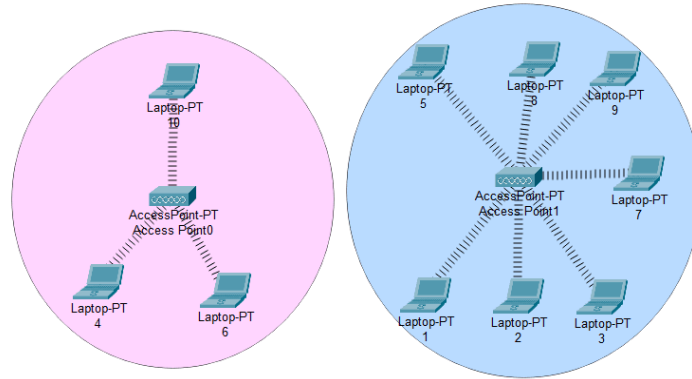
*Step9*:结束。

## 七、 参考文献

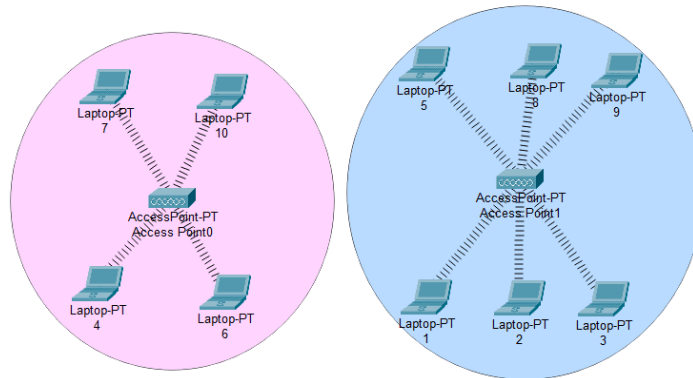
- [1] 谢希仁.计算机网络[M].5 版.北京:电子工业出版社, 2008.1
- [2] 耿丽丽.失效网络中节点可通信性能评估方法研究[J].现代电子技术,2016,39(22):27-31
- [3] 刘丽莉.评价指标选取方法研究[J].河北建筑工程学院学报,2004(01):134-136.
- [4] 金顺福.基于 1 坚持指数退避算法的时隙 CSMA/CD 协议的排队模型的建立与分析[J].计算机工程与应用,2002(05):51-52+73.
- [5] 方飞,谢丽春.CSMA/CD 性能分析及 MATLAB 仿真[J].内江师范学院学报,2015,30(12):14-20.
- [6]杨莉,戴育雷,李南.基于蒙特卡洛仿真的项目完工成本概率预测[J].计算机仿真, 2014, 31(09):30-304+309.
- [7]朱健. 四川电网可靠性评估的蒙特卡洛仿真[D].重庆大学,2007.]
- [8] SharmaV,HellstrandEdF.FrameworkforMPLS-basedrecovery [S].IETF RFC3469,2003.
- [9] Makam S, Sharma V, Owens K, et al. Protection/restoration of MPLS networks[S].1999.
- [10] Haskin D,Krishnan R.A method for setting an alternative label switched paths to handle fast reroute[S].2000.
- [11] Ping Pan,Der-Hwa Gan,George Swallow,et al.Fast reroute extensions to RSVP-TE for LSP[S].
- [12] 卢先领.无冲突的无线传感器网络广播算法[J].计算机工程与设计,2008(16):4191-4193.
- [13] 赵瑞琴,刘增基,文爱军.有效延长无线传感器网络寿命的分布式广播算法[J].高技术通讯,2008,18(05):469-474.

## 八、附录

附录 1：476~1000  $\mu s$  无线网信息传递拓扑图



(1) 476-817  $\mu s$



(2) 818~1000  $\mu s$

附录 2：MATLAB 仿真模拟程序代码

方案一：轻载

```

1.  clc
2.  clear
3.  n=10;% 10 个通信节点
4.  m=1;% 每个节点一个分组
5.  time=fix(rand(n,m)*5000);% 设定初始发送时间
6.  goal=1+fix(rand(n,m)*n);% 设定目的节点
7.
8.  for i=1:n
9.      time(i,:)=sort(time(i,:));% 排序时间
10. end
11. oldtime=time;
12.
13.
14. %排除自身发送给自身的情况
15. for i=1:n

```

```

16.     for j=1:m
17.         if goal(i,j)==i
18.             goal(i,j)=1+mod(i+1,n);
19.         end
20.     end
21. end
22.
23. tt=10
24. tt1=30;
25. t=0;%时间流
26. m1=0;%同一时刻想发送数据节点个数
27. sumtime=0;%用来判断是否可以结束循环
28. L=1;
29. count=0;
30. N=100000;
31. LY=zeros(N,1);
32.
33. %% 开始测试
34. while 1
35.     sumtime=0;
36.     for i=1:n
37.         for j=1:m
38.             if t==time(i,j)
39.                 m1=m1+1;
40.                 match_i=i;%找到匹配的发送节点
41.                 match_j=j;
42.                 match_goal=goal(i,j);%对应的目的节点
43.                 block(m1,1)=match_i;
44.                 block(m1,2)=match_goal;
45.                 block(m1,3)=match_j;
46.             end
47.             sumtime=sumtime+time(i,j);
48.         end
49.     end
50.     if sumtime==0
51.         break;
52.     end
53.
54.     if m1==1
55.         if randperm(4,1)==1 || randperm(4,1)==2 || randperm(4,1)==3
56.             T=time(match_i,match_j);%该节点想发送的时间
57.             num=1;
58.             for q=1:n
59.                 for p=1:m
60.                     if time(q,p)-T>0 && time(q,p)-T<tt1
61.                         if randperm(2,1)==1
62.                             old=time(q,p);
63.                             time(q,p)=T+num*tt1;
64.                             num=num+1;
65.                             count=count+1;
66.                             blocktime(count,1)=T;%拥堵时对应的时间
67.                             blocktime(count,2)=old;
68.                         end
69.                     end
70.                 end

```

```

71.         end
72.         list(L,1)=match_i;%录入发送节点
73.         list(L,2)=match_goal;%录入目的节点
74.         list(L,3)=T;%录入发送时间
75.         list(L,4)=T+tt1;%录入结束时间
76.         L=L+1;
77.         for ly=T:T+tt1
78.             if ly~=0
79.                 LY(ly)=LY(ly)+1;
80.             end
81.         end
82.         time(match_i,match_j)=0;
83.     else
84.         time(match_i,match_j)=time(match_i,match_j)+tt;%若不在范围，则将该点发送时间延
后
85.     end
86.     elseif m1>1
87.         for ii=1:m1
88.             if randperm(4,1)==1 ||randperm(4,1)==2 ||randperm(4,1)==3
89.                 if ii==1
90.                     T=time(block(ii,1),block(ii,3));
91.                     list(L,1)=block(ii,1);%录入发送节点
92.                     list(L,2)=block(ii,2);%录入目的节点
93.                     list(L,3)=T;%录入发送时间
94.                     list(L,4)=T+tt1;%录入结束时间
95.                     L=L+1;
96.                     for ly=T:T+tt1
97.                         if ly~=0
98.                             LY(ly)=LY(ly)+1;
99.                         end
100.                    end
101.                    time(block(ii,1),block(ii,3))=0;
102.                else
103.                    T=time(block(ii,1),block(ii,3));
104.                    time(block(ii,1),block(ii,3))=time(block(ii,1),block(ii,3))+tt1;
105.                    count=count+1;%拥堵次数+1
106.                    blocktime(count,1)=T;
107.                end
108.
109.            else
110.                T=time(block(ii,1),block(ii,3));
111.                list(L,1)=block(ii,1);%录入发送节点
112.                list(L,2)=block(ii,2);%录入目的节点
113.                list(L,3)=T;%录入发送时间
114.                list(L,4)=T+tt1;%录入结束时间
115.                L=L+1;
116.                for ly=T:T+tt1
117.                    if ly~=0
118.                        LY(ly)=LY(ly)+1;
119.                    end
120.                end
121.                time(block(ii,1),block(ii,3))=0;
122.            end
123.        end

```



```

124.     end
125.
126.     m1=0;
127.     t=t+1;
128. end
129. %% 数据处理
130. LY1=LY;
131. for i=1:N
132.     if LY(i)~=0
133.         LY(i)=1;
134.         maxi=i;
135.     end
136. end
137. LY1=LY1(1:maxi);
138. bar(LY1);
139. S=sum(LY);
140. t=maxi;
141. CTL=count/(count+n*m)
142. LYL=S/t

```

## 方案二：重载

```

1.     clc
2.     clear
3.     n=10;% 10 个通信节点
4.     m=1;% 每个节点一个分组
5.     time=fix(rand(n,m)*5000);% 设定初始发送时间
6.     goal=1+fix(rand(n,m)*n);% 设定目的节点
7.
8.     for i=1:n
9.         time(i,:)=sort(time(i,:));% 排序时间
10.    end
11.    oldtime=time;
12.
13.    for i=1:n
14.        for j=1:m
15.            if goal(i,j)==i
16.                goal(i,j)=1+mod(i+1,n);
17.            end
18.        end
19.    end
20.
21.    tt=20;
22.    tt1=30;
23.    t=0;% 时间流
24.    m1=0;% 同一时刻想发送数据节点个数
25.    sumtime=0;% 用来判断是否可以结束循环
26.    L=1;
27.    count=0;
28.    N=100000;
29.    LY=zeros(N,1);
30.
31.    K=ones(n,m);
32.
33.    %% 开始测试

```

```

34. while 1
35.     sumtime=0;
36.     for i=1:n
37.         for j=1:m
38.             if t==time(i,j)
39.                 m1=m1+1;
40.                 match_i=i;%找到匹配的发送节点
41.                 match_j=j;
42.                 match_goal=goal(i,j);%对应的目的节点
43.                 block(m1,1)=match_i;
44.                 block(m1,2)=match_goal;
45.                 block(m1,3)=match_j;
46.             end
47.             sumtime=sumtime+time(i,j);
48.         end
49.     end
50.     if sumtime==0
51.         break;
52.     end
53.
54.     if m1==1
55.         if randperm(2,1)==1
56.             T=time(match_i,match_j);
57.             for q=1:n
58.                 for p=1:m
59.                     if time(q,p)-T>0 && time(q,p)-T<tt1
60.                         if randperm(2,1)==1
61.                             old=time(q,p);
62.                             k=K(q,p);
63.                             if k>10
64.                                 k=10;
65.                             end
66.                             K(q,p)=K(q,p)+1;
67.                             r=randperm(2^k-1,1);
68.                             time(q,p)=time(q,p)+r*tt;
69.                             count=count+1;
70.                             blocktime(count,1)=T;%拥堵时对应的时间
71.                             blocktime(count,2)=old;
72.                         end
73.                     end
74.                 end
75.             end
76.             list(L,1)=match_i;%录入发送节点
77.             list(L,2)=match_goal;%录入目的节点
78.             list(L,3)=T;%录入发送时间
79.             list(L,4)=T+tt1;%录入结束时间
80.             L=L+1;
81.             for ly=T:T+tt1
82.                 if ly~=0
83.                     LY(ly)=LY(ly)+1;
84.                 end
85.             end
86.             time(match_i,match_j)=0;
87.         else
88.             time(match_i,match_j)=time(match_i,match_j)+tt;

```

```

89.     end
90.     elseif m1>1
91.         iii=0;
92.         for ii=1:m1
93.             if randperm(2,1)==1
94.                 if iii==0
95.                     T=time(block(ii,1),block(ii,3));
96.                     list(L,1)=block(ii,1);%录入发送节点
97.                     list(L,2)=block(ii,2);%录入目的节点
98.                     list(L,3)=T;%录入发送时间
99.                     list(L,4)=T+tt1;%录入结束时间
100.                    L=L+1;
101.                    for ly=T:T+tt1
102.                        if ly~=0
103.                            LY(ly)=LY(ly)+1;
104.                        end
105.                    end
106.                    time(block(ii,1),block(ii,3))=0;
107.                    iii=iii+1;
108.                else
109.                    T=time(block(ii,1),block(ii,3));
110.                    k=K(block(ii,1),block(ii,3));
111.                    if k>10
112.                        k=10;
113.                    end
114.                    K(block(ii,1),block(ii,3))=K(block(ii,1),block(ii,3))+1;
115.                    r=randperm(2^k-1,1);
116.                    time(block(ii,1),block(ii,3))=time(block(ii,1),block(ii,3))+r*tt1;
117.                    count=count+1;%拥堵次数+1
118.                    blocktime(count,1)=T;
119.                end
120.            else
121.                T=time(block(ii,1),block(ii,3));
122.                list(L,1)=block(ii,1);%录入发送节点
123.                list(L,2)=block(ii,2);%录入目的节点
124.                list(L,3)=T;%录入发送时间
125.                list(L,4)=T+tt1;%录入结束时间
126.                L=L+1;
127.                for ly=T:T+tt1
128.                    if ly~=0
129.                        LY(ly)=LY(ly)+1;
130.                    end
131.                end
132.                time(block(ii,1),block(ii,3))=0;
133.            end
134.        end
135.    end
136.
137.    m1=0;
138.    t=t+1;
139.
140. end
141. %% 数据处理
142. LY1=LY;
143. for i=1:N

```

```

144.     if LY(i)~=0
145.         LY(i)=1;
146.         maxi=i;
147.     end
148. end
149. LY1=LY1(1:maxi);
150. bar(LY1);
151. S=sum(LY);
152. t=maxi;
153. CTL=count/(count+n*m)
154. LYL=S/t

```