

# 基于接触信息的自适应机会网络路由算法

武淑艳 韩毅刚 傅秋宇 冯飞

(南开大学电子信息与光学工程学院 天津 300350)

**摘要** 考虑到实际网络环境的不断变化以及机会网络节点总是在密集与稀疏场景间随机切换的特点,提出一种能够借助节点接触信息进行网络环境判断的自适应路由算法——CIAONR(Contact Information-based Adaptive Opportunity Network Routing)。CIAONR在节点分布式采集接触信息的过程中,利用节点接触延迟与生存期的关系判断节点所处网络环境。然后依据CIAONR约束条件指导转发路径的选择,最终按照对应交互流程完成消息交付。理论分析和仿真结果表明,CIAONR算法在不同网络环境下均保持较高投递率,网络开销和延迟也控制在一定范围内,具有普适效果。

**关键词** 机会网络 路由算法 接触信息 自适应

中图分类号 TP393 文献标识码 A DOI:10.3969/j.issn.1000-386x.2017.07.019

## ADAPTIVE OPPORTUNITY NETWORK ROUTING ALGORITHM BASED ON THE CONTACT INFORMATION

Wu Shuyan Han Yigang Fu Qiuyu Feng Fei

(College of Electronic Information and Optical Engineering, Nankai University, Tianjin 300350, China)

**Abstract** Considering the constantly changes in the actual network environment and the characteristics of opportunity networks nodes which always switch stochastically between dense and sparse scene, an adaptive routing algorithm called CIAONR which can judge the network environment with the node contact information is proposed. CIAONR uses nodes' contact delay and lifetime to determine the network environment in which the nodes are located during the process of collecting the contact information. Then, according to the CIAONR constraints, the choice of the forwarding path is guided, and finally the message delivery is completed according to the corresponding interaction flow. Theoretical analysis and simulation results show that the CIAONR maintains a high delivery rate in different network environments, and the network overhead and delay are also controlled within a certain range, which has a universal effect.

**Keywords** Opportunity network Routing algorithm Contact information Adaptive

## 0 引言

机会网络是一种基于节点接触形成通信机会进而通过逐跳转发实现数据交互的移动自组织网络 MANET(Mobile Ad-Hoc Network),是实现间歇式连通环境下节点通信的基本方法。早期机会网络主要应用于野生动物追踪、灾难环境临时组网、偏远地区网络覆盖等场景,应用局限性较大。近几年随着各种便携式设备迅速发展和普及,移动设备感知、计算、通信、存储等能

力越来越强,物联网、车联网快速兴起,智慧城市等新概念每日更新,机会网络因其对网络全连通无要求的特点,更符合实际环境的需求,再次受到高度关注。

机会网络中,节点环境常在密集与稀疏间进行反复切换<sup>[1-2]</sup>,如大量野生动物群聚生活,车联网用户在社区间穿梭等场景,而现有机会网络路由算法的研究和评估主要侧重于在一个特定场景下投递率、传输延迟和转发代价之间的权衡,对网络环境快速变化这一特点的考虑和研究相对较少。节点接触信息主要包括节点接触时长、接触延迟、相遇频率等,能够较全面地

将当前网络环境进一步量化,为机会网络路由算法提供自适应调整依据。因此,本文着眼于机会网络中节点的接触信息,提出了一种自适应路由算法——CIAONR,对密集、稀疏网络拓扑能够自行判定,进而达到提高不同网络环境下投递率,控制网络开销和延迟的目的,有效提高了路由的普适效果。

## 1 相关工作

机会网络数据传输的主要工作方式为“存储-携带-转发”,如何选择转发时机和转发中继节点是机会网络路由算法主要解决的问题<sup>[3]</sup>。现有机会转发策略基本分为两类,即零信息型和信息辅助型<sup>[4]</sup>,见图1所示。

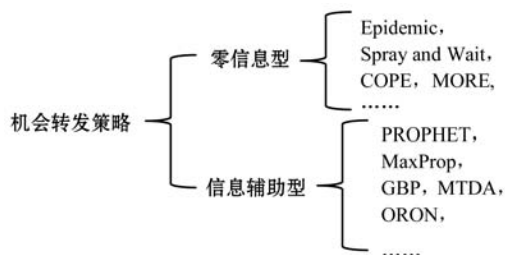


图1 机会网络转发策略分类

零信息型是最早提出的机会网络路由算法,包括Epidemic、Spray and Wait等算法。Epidemic算法是一种模仿传染病扩散的洪泛算法,相遇即转发存储,在网络资源充足的情况下能够获得最优的延时性能,但由于网络中存在大量的分组副本,网络开销也相对增大。Spray and Wait算法针对Epidemic进行了改进,通过网络中洪泛固定数目的分组副本来实现保持较低的传输延迟和网络开销的目的,其传输量远小于Epidemic,网络延迟接近最优,具有较好的扩展性。在此基础上,研究学者进一步提出了将网络编码技术应用于机会网络的路由算法,对Epidemic算法进行了一系列改进<sup>[5]</sup>。Katti等提出基于机会的网络编码方法COPE,有效提高无线网络的传输容量和吞吐量。Chachulski等对COPE进行了改进,提出MORE协议将编码信息路由到多跳邻居节点,网络吞吐量进一步提升。

由于充分利用了实际网络中反映出来的信息,近年来信息辅助型策略更受关注。PROPHET算法依据历史信息提出了基于节点接触概率的传输预测路由算法,在社区模型场景中性能较好。GBP算法在此基础上考虑了节点分组情况,认为同组节点不应保存相同副本,通过本地信息引导分组副本到达目的节点,在网络开销上表现出色,但网络延迟较大<sup>[6]</sup>。Burgess等学者提出的MaxProp算法根据链路的历史数据进行数据

分组的优先级划分,然后按照优先级调度数据分组,获得了较高的成功送达率,但网络开销、传输延迟都随之增大。Tan等学者对下一条邻居节点进行历史传输时间标记,建立了局部最小传输延迟算法MTDA<sup>[7]</sup>,减小了节点之间的传输延迟。在利用邻居节点信息指导路由转发方面,任智等学者提出了基于旁听的机会网络路由算法ORON<sup>[8]</sup>,通过旁听邻居节点信息设计博弈策略从而激励节点交易。

此外,Chen、Bulut等学者从人文社会的角度提出了根据社会关系进行节点划分的机会网络路由算法<sup>[9-11]</sup>,在社区等特定场景下获得较好的网络性能,但适用场景局限性较大。

## 2 基于接触信息的自适应路由算法

为打破路由算法适用场景局限的问题,重点考虑机会网络环境变化的特点,提出基于接触信息的自适应路由算法CIAONR。本节主要从网络模型、数据模型等方面详述了节点对接触信息的整合和应用,并进行了深入的理论分析。

### 2.1 机会网络模型

**定义1** 网络模型。机会网络用带权有向图 $G=(V,E,F)$ 表示,其中节点集合 $V=\{v_1, v_2, \dots, v_n\}$ ,网络规模为 $n, n>1$ ;网络链路集合 $E=\emptyset \cup \{e_1, e_2, \dots, e_m\}$ , $e_k=v_i v_j$ 表示节点 $v_i$ 与节点 $v_j$ 之间的链路,节点链路规模为 $m, 1 \leq m \leq n(n-1)$ ;  $F(e_k)$ 表示 $e_m$ 的权值集。

**定义2** 链路权重模型。 $F(e_k)$ 为链路 $e_k=v_i v_j$ 的权值集合, $F(e_k)=\{T_k, L_k, P_k\}$ , $T_k$ 表示节点 $v_i$ 与节点 $v_j$ 之间的链路持续时间均值, $T_k>0$ ;  $L_k$ 表示节点 $v_i$ 与节点 $v_j$ 之间的链路相邻两次建立的时间延迟均值, $L_k>0$ ;  $P_k$ 表示节点 $v_i$ 与节点 $v_j$ 之间的链路建立概率,即节点 $v_i$ 与节点 $v_j$ 的接触概率, $0 \leq P_k \leq 1$ 。

**性质1** 节点链路为有向模型,链路建立对两节点作用一致,即 $v_i v_j$ 与 $v_j v_i$ 表示的是同一条链路。但从两节点方向看链路对应权重不完全相同,链路有方向之分。

**定义3** 节点模型。在节点规模为 $n$ 的机会网络中,每个节点维护3个 $n \times n$ 数据矩阵,即接触概率矩阵 $P_{ALL}$ 、接触时间矩阵 $T_{ALL}$ 、延迟矩阵 $L_{ALL}$ 。矩阵行列交叉点即为对应节点对接触信息,如 $P_{ALL}$ 矩阵的第 $i$ 行第 $j$ 列值 $P_{ij}$ 即为从本节点角度看节点 $v_i$ 与节点 $v_j$ 的接触频率,如图2所示, $P_{ALL}$ 第一行为节点 $v_1$ 与其他 $(n-1)$ 个节点的接触概率,以此类推。各矩阵均记录

了上一次更新时间 LastUpdateTime, 节点相遇则以此为依据重新计算概率值或进行相应数据更新。

$$\begin{pmatrix} 0 & p_{1 \rightarrow 2} & \cdots & p_{1 \rightarrow n} \\ p_{2 \rightarrow 1} & 0 & \cdots & p_{2 \rightarrow n} \\ \vdots & \vdots & 0 & \vdots \\ p_{n \rightarrow 1} & p_{n \rightarrow 2} & \cdots & 0 \end{pmatrix}$$

图 2 节点接触概率矩阵  $P_{ALL}$

**性质 2** 各节点维护的接触时间矩阵  $T_{ALL}$ 、延迟矩阵  $L_{ALL}$  均为对称矩阵, 而接触频率矩阵  $P_{ALL}$  为非对称

矩阵,  $\sum_{i=1}^j p_{ij} = 1$ 。

由于链路建立对两节点作用效果是对等的, 则接触时间、接触延迟对两节点而言均一致。节点接触概率根据各节点接触其他节点的频率会发生相应变化, 因而存在差异。节点更新接触概率时会进行归一化处理, 接触概率矩阵  $P_{ALL}$  行向量逐项求和为 1。

**定义 4** 节点相遇概率。从移动模型角度看, 节点两两相遇的概率为<sup>[12]</sup>:

$$p_{遇} \approx \frac{2 \times \omega \times R \times E[V]}{A} = \frac{2 \times \omega \times R \times E[V]}{\pi r^2} \quad (1)$$

其中,  $\omega$  为移动模型常量, 对应特定移动模型为固定值。  $R$  为节点通信范围, 在本文中也设置为常量。  $A$  为节点移动区域面积, 在此以辐射区域计算。本式可简化为:

$$p_{遇} \propto \frac{1}{r^2} \quad (2)$$

**定义 5** 平均传输字节数。节点进行多次分组复制后, 对每次传输的字节数求均值记为该节点的平均传输字节数, 用符号  $B_{AVG}$  表示, 每次进行传输后自动更新。

## 2.2 数据交互模型

机会网络中节点的信息传递完全依赖节点相遇机会来完成, 网络中传递的数据分组类型主要有 3 种, 即摘要向量 SV、数据分组副本、分组到达确认信息 ACK。在本算法中, 摘要向量 SV 主要包含本节点维护的 3 个  $n \times n$  数据矩阵信息, 分组到达确认信息 ACK 则用于通知所有节点已成功交付的数据分组。具体交互流程如图 3 所示。

1) 节点  $v_i$  与节点  $v_j$  相遇首先交换本节点携带的以对方为目的节点的数据分组, 并将已成功交付信息加入到 ACK 中。

2) 交换  $v_i$  与  $v_j$  各自维护的 SV 向量表, 进行  $P_{ALL}$ 、 $T_{ALL}$ 、 $L_{ALL}$  的交换更新, 根据最新数据重新计算缓存区内数据分组副本的开销, 更新数据分组队列。

3) 交换 ACK 信息, 对应删除缓存区内已成功交付数据分组的副本。

4) 依据数据分组队列进行数据分组副本传递, 更新各副本的剩余副本数量值, 完成交互, 更新  $T_{ALL}$  中对应本节点相关数据。

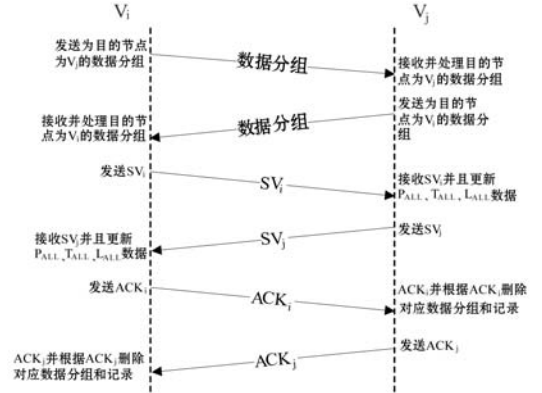


图 3 节点相遇交互流程

## 2.3 关于节点接触信息的 CIAONR 约束条件

在机会网络的信息交互过程中, 节点接触信息是对节点运动、通信最直接而有效的量化表示。大量数据分析显示, 移动网络中大部分节点相遇间隔时间较短, 呈现近似的幂律分布<sup>[13-16]</sup>, 即节点的移动和相遇有社区聚集现象。同社区节点在较短时间内可以产生大量的相遇, 而不同社区的节点相遇则需要较长的时间。

文献[16]数据显示, 在一个较长的单位时间  $T$  里, 有超过 95% 的相遇发生在远不足 10% 的时间中, 相遇发生的时间间隔短, 称为节点处于 BUSY 状态; 而在剩下的  $T$  的 90% 时间中, 仅发生了不足 5% 的相遇, 相遇时间间隔长, 节点处于 SPARE 状态。当节点处于 BUSY 状态(在本社区内运动), 应积极的与接触节点进行高效快速的信息交互; 处于 SPARE 状态下(在社区间运动), 节点则应以低开销为前提给相遇节点尽可能多的散布副本。因此根据节点接触频率、接触时长、接触时延有效判断节点社区分组和所处状态, 进而作出能够自动调整的路由算法是非常有必要的。

假设一个小型网络由 6 个节点组成, 每组包含 2 个节点, 具体分布如图 4 所示。为简化模型, 节点随机移动, 节点接触概率  $p_{接触}$  采用改进后的增量平均化方法进行估算和更新, 见式(3):

$$p_{接触} = \frac{p_{遇} \times \alpha^k + 1}{\alpha^k + 1} = \frac{p_{遇} \times \alpha^k}{\alpha^k + 1} + \frac{1}{\alpha^k + 1} \quad (3)$$

其中,  $p_{遇}$  由式(1)给出。  $\alpha$  为接触概率衰减因子,  $\alpha \in (0, 1)$ 。  $k$  是两节点相遇延迟的时间单位数。计算公式如下:

$$k = \left\lfloor \frac{\bar{L}}{T_m} \right\rfloor \quad (4)$$

其中,  $\bar{L}$  为节点相遇延迟均值,  $T_m$  是时隙长度,  $k$  取不大于二者商的最大整数。

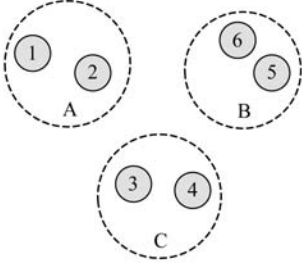


图4 小型网络节点分布图

假设有消息  $M$  要从节点 1 发送到节点 4, 消息  $M$  的生存期为  $T_{TTL}$ , 可进行一次副本复制。节点 1 首先向节点  $x$  传送 1 个副本, 节点  $x$  可在节点 2、节点 3、节点 5 和节点 6 中进行选择。节点 1 或节点  $x$  若在  $T_{TTL}$  到期前遇到节点 4 则可完成交付。

传送副本过程中, 认为同一社区内节点相遇概率高于不同社区间节点相遇概率, 简化后如式(5),  $a$  为不同社区间距离表示。

$$p_{\text{遇}} = \begin{cases} \frac{\alpha^{k_1}}{r^2} & \text{同一社区} \\ \frac{\alpha^{k_2}}{(r+a)^2} & \text{不同社区} \end{cases} \quad k_2 > k_1, \text{且均为正整数} \quad (5)$$

假设节点接触平均延迟  $\bar{L}$  相对于生存期  $T_{TTL}$  为可接受时长, 即在 2 倍平均延迟  $\bar{L}$  时间内, 生存期  $T_{TTL}$  不会到期。当  $x=2$ , 则 A 社区网络中存在 2 个  $M$  副本, 计算可得对应送达概率  $p_{1,2 \rightarrow 4}$ 。

$$p_{1,2 \rightarrow 4} = \frac{\alpha^{k_1}}{r^2} [1 - (1 - p_{1 \rightarrow 4})(1 - p_{2 \rightarrow 4})] = \frac{\alpha^{k_1}}{r^2} \left[ 1 - \left( 1 - \frac{\alpha^{k_2}}{(r+a)^2} \right)^2 \right] = \frac{2\alpha^{k_1+k_2}}{r^2(r+a)^2} + \frac{\alpha^{k_1+2k_2}}{r^2(r+a)^4} \quad (6)$$

当  $x=3$  时, 则 A 组、C 组各有 1 个副本存在, 消息  $M$  送达概率为  $p_{1,3 \rightarrow 4}$ 。

$$p_{1,3 \rightarrow 4} = \frac{\alpha^{k_2}}{(r+a)^2} [1 - (1 - p_{1 \rightarrow 4})(1 - p_{3 \rightarrow 4})] = \frac{\alpha^{k_2}}{(r+a)^2} \left[ 1 - \left( 1 - \frac{\alpha^{k_2}}{(r+a)^2} \right) \left( 1 - \frac{\alpha^{k_1}}{r^2} \right) \right] = \frac{\alpha^{2k_2}}{(r+a)^4} + \frac{\alpha^{k_1+k_2}}{r^2(r+a)^2} - \frac{\alpha^{k_1+2k_2}}{r^2(r+a)^4} \quad (7)$$

计算二者之差,

$$p_{1,3 \rightarrow 4} - p_{1,2 \rightarrow 4} = \frac{\alpha^{2k_2}}{(r+a)^4} - \frac{\alpha^{k_2+k_1}}{r^2(r+a)^2} = \frac{\alpha^{k_2+k_1}}{(r+a)^2} \left[ \frac{\alpha^{k_2-k_1}}{(r+a)^2} - \frac{1}{r^2} \right] \quad (8)$$

由于  $\alpha \in (0, 1)$ ,  $(k_2 - k_1) \geq 1$  且为正整数, 所以  $\alpha^{k_2-k_1} < 1$ ,  $(r+a)^2 > r^2$ , 易得  $p_{1,3 \rightarrow 4} - p_{1,2 \rightarrow 4} < 0$ , 即  $p_{1,3 \rightarrow 4} < p_{1,2 \rightarrow 4}$ 。

显然, 当  $x=5, x=6$  时,

$$p_{1,5 \rightarrow 4} = p_{1,6 \rightarrow 4} = \frac{\alpha^{k_2}}{(r+a)^2} [1 - (1 - p_{1 \rightarrow 4})(1 - p_{5 \rightarrow 4})] = \frac{\alpha^{k_2}}{(r+a)^2} \left[ 1 - \left( 1 - \frac{\alpha^{k_2}}{(r+a)^2} \right)^2 \right] \leq \quad (9)$$

$$p_{1,3 \rightarrow 4} \leq p_{1,2 \rightarrow 4}$$

由此可知, 当消息生存期足够长, 仅量化考虑延迟对数据分组交付的影响时, 将副本传送给本组内节点能够获得较大投递率, 对应场景为密集环境。而在文献[6]中理论分析认为, 当不考虑延迟对交付的影响, 仅考虑消息生存期限制时, 传递到不同社区节点的交付概率比传递给同社区节点交付概率大, 在此情况下传递给频繁相遇的节点会对消息的传输起到增大开销的作用, 对应稀疏场景。综上所述, 对于信息的投递时机与投递对象应根据节点所处环境进行相应调整, 从而达到投递率最大化, 网络开销最小化的目的。

提出一个以节点接触延迟均值  $\bar{L}$ 、消息生存期  $T_{TTL}$  为依据的场景判断阈值  $T_{\text{阈}}$ 。

$$T_{\text{Compare}} = \frac{\bar{L}}{T_{TTL}} \quad (10)$$

在仿真中, 设置  $T_{TTL}$  为固定值, 取值 300 min, 阈值  $T_{\text{阈}}$  主要依赖于  $\bar{L}$ 。数据显示  $\bar{L}$  分布近似服从指数分布 [13, 15], 常见路由协议平均延迟浮动范围主要在 [3 300, 7 500] (s), 采用迭代的方法求出  $\bar{L}$  的估算均值, 代入式(10) 计算得到判决阈值  $T_{\text{阈}}$ 。具体判决条件如下:

- 1) 当  $T_{\text{Compare}} \geq T_{\text{阈}}$  时, 认为节点处于密集环境, 将消息优先传送给接触概率大的节点;
- 2) 当  $T_{\text{Compare}} < T_{\text{阈}}$  时, 认为节点处于稀疏环境, 将消息优先传送给接触概率较小的节点。

## 2.4 数据分组队列与节点接触度队列维护规则

CIAONR 算法同时维护两个队列, 分别为数据分组队列和节点接触度队列, 两个队列分别存在阈值  $T_{\text{分组}}$  和  $T_{\text{节点}}$ 。记节点缓存大小为  $B$ , 当节点平均传输字节数  $B_{\text{AVG}} \leq \frac{B}{2}$  时,  $T_{\text{分组}} = B_{\text{AVG}}$ ; 当  $B_{\text{AVG}} > \frac{B}{2}$  时,  $T_{\text{分组}} = B - B_{\text{AVG}}$ 。  $T_{\text{节点}}$  则根据文献[17] 数据设定为接触度最大值  $P_{\text{MAX}}$  的 90%, 即  $T_{\text{节点}} = 0.9P_{\text{MAX}}$ 。

对于数据分组队列, 当分组跳数小于等于阈值  $T_{\text{分组}}$  时, 按照跳数进行排序, 假定跳数小的数据分组送达率高, 优先发送跳数小的数据分组; 当分组跳数大于阈值  $T_{\text{分组}}$  时, 按照路径开销排序, 单跳路径开销计算公

式如下:

$$C_{i \rightarrow j} = (1 - P_{i \rightarrow j}) \times \overline{L_{i \rightarrow j}} \quad (11)$$

使用 Dijkstra 算法求出最小开销路径,根据消息的最小开销进行排序。

节点接触度队列按照各节点与本节点接触度进行排序。节点相遇时,首先按照消息队列顺序读取数据分组,查询下一跳节点与本节点接触度  $p_{\text{下一跳}}$ 。当  $T_{\text{Compare}} \geq T_{\text{阈}}$  时,仅将满足  $p_{\text{下一跳}} \geq T_{\text{节点}}$  的消息发送给对应下一跳节点,否则跳过该消息,读取下一消息;当  $T_{\text{Compare}} < T_{\text{阈}}$  时,仅将满足  $p_{\text{下一跳}} < T_{\text{节点}}$  时,将该消息发送给对应下一跳节点,否则跳过。

### 3 仿真结果及分析

采用机会网络专用仿真平台 ONE 分别在不同网络规模和不同节点容量下进行建模和仿真,以数据分组成功投递率、网络开销、数据分组平均端到端时延三个指标对 Epidemic、Spray And Wait、GBP<sup>[5]</sup> 和 CIAONR 算法性能进行比较。

#### 3.1 仿真参数设置

仿真实验使用赫尔辛基 (Helsinki) 部分地区为背景,主要参数设置如表 1 所示。

表 1 主要仿真参数

参数	取值
仿真范围/m	4 500 × 3 400
移动模型/(m · s <sup>-1</sup> )	Shortest Path Map-Based Movement
速率区间/s	[0.5, 15]
通信半径/m	10
仿真时间/s	43 200
最大数据速率/(Mbit · s <sup>-1</sup> )	10
数据分组大小/Byte	[500 KB, 1 MB]
数据分组产生间隔/s	[25, 35]
仿真热身时间/s	1 000
随机种子	128, 130, 132, 134

在仿真时间 1 000 s 后随机选取节点,在间隔时间区间 [25, 35] 内随机产生新消息,从 1 000 s 开始,到 43 200 s 结束,共计发送 [1 206, 1 688] 个数据分组。选取随机种子集合 {128, 130, 132, 134}, 每组实验做 4 次仿真,取统计结果的平均值进行分析。

#### 3.2 固定节点缓存改变节点规模

在固定节点缓存容量为 20 MB 的情况下,对不同节点规模进行了仿真。图 5 是随着网络中节点数量的

增加各路由算法的成功投递率变化趋势。CIAONR 算法在各种节点规模下均为最佳,在节点规模为 400 时,成功投递率超过 95%,接近 100%。而在 4 500 m × 3 400 m 的范围内仅有 50 个节点时,由于 CIAONR 算法引入了自适应调整机制,在节点密度较低的情况下能够做出自行判断,成功投递率接近 60%,投递效果较为稳定。

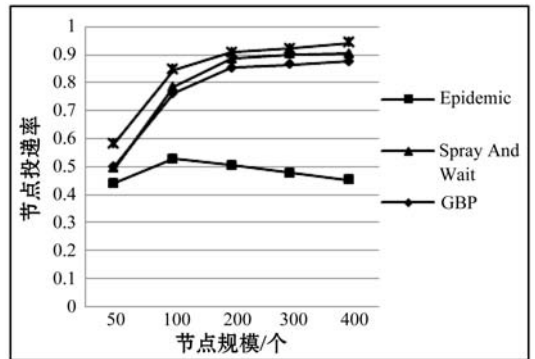
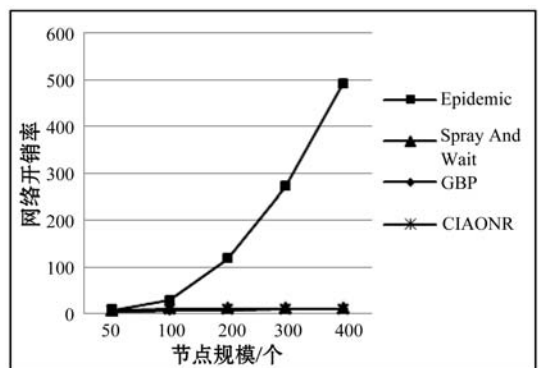
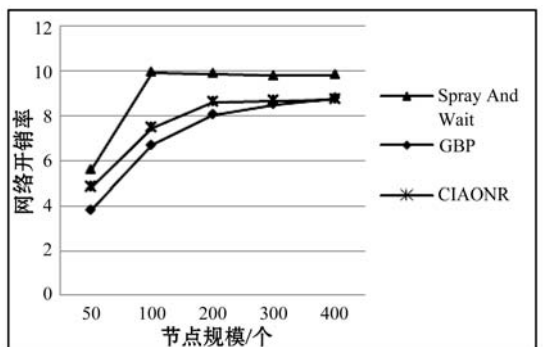


图 5 固定缓存改变节点规模-投递率

图 6(a) 为固定节点缓存容量的情况下,4 种路由算法的网络交付开销率,开销率用源节点产生的数据分组总数和所有节点转发数据分组总数之比来进行评价。由于 Epidemic 算法“随遇随传随存”的工作模式,造成开销率随节点规模的增加而大幅度增加。为方便查看具体折线,特将 Spray And Wait、GBP、CIAONR 算法折线单独在图 6(b) 中做比较。



(a) 4 种路由算法开销率对比图



(b) 3 种路由算法开销率对比图

图 6 固定缓存改变节点规模-开销率

如图6(b)所示,CIAONR算法在50~400个的节点规模中均保持低于9的开销率,在节点密度较小的环境下略高于GBP算法,其原因为在较稀疏网络环境下进行CIAONR算法进行了自适应的调整,传送较多数据分组给组外节点,造成开销率的上升,当节点规模达到300个后,则与GBP算法基本持平,到400个的节点规模时较之GBP还有所下降。

图7为4种路由算法的平均时延趋势图,CIAONR算法平均交付时延在各规模网络下均接近Spray And Wait算法的平均时延,低于Epidemic和GBP算法平均时延。在节点规模达到200个以后,其平均时延稍低于Spray And Wait算法,接近最优。

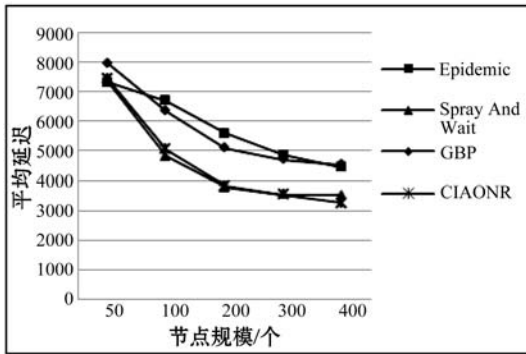


图7 固定缓存改变节点规模-平均延迟

由图5-图7可知,在固定节点缓存容量为20 MB的情况下,CIAONR算法的投递率始终高于另外3种算法,其网络开销在节点规模较小的时候虽略高于GBP算法,但在节点规模增大的过程中,也不断接近甚至低于GBP算法。CIAONR算法的平均交付时延在较稀疏网络环境中与Spray And Wait算法接近,始终低于另外2种算法。随节点规模增大其平均时延逐步降低,超过200个的节点规模后平均时延低于Spray And Wait算法,接近最优。

### 3.3 固定节点规模改变节点缓存

固定节点规模为200个,在不同节点缓存容量的情况下进行了算法仿真对比。图8-图10是随着网络中节点缓存容量的增加各路由算法的成功投递率、网络开销、平均投递延迟的变化趋势。通过对比可得,

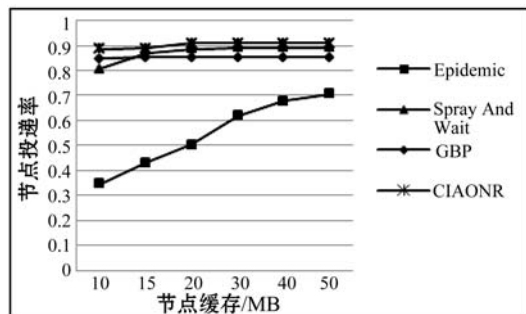
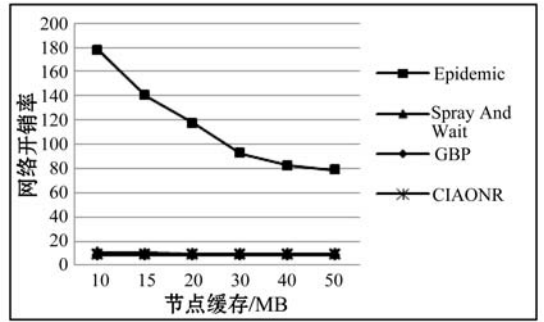
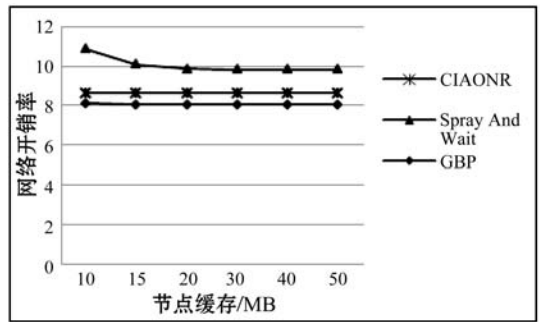


图8 固定节点规模改变缓存-投递率

CIAONR算法对节点缓存容量要求不高,除了容量为10、15 MB时稍有变化,达到20 MB后折线基本保持水平。仿真显示,Epidemic算法随节点缓存变化各参数变化较大,Spray And Wait算法在节点缓存为10 MB时也稍有不稳定,CIAONR与GBP算法对节点缓存容量要求最低。



(a) 4种路由算法开销率对比图



(b) 3种路由算法开销率对比图

图9 固定节点规模改变缓存-开销率

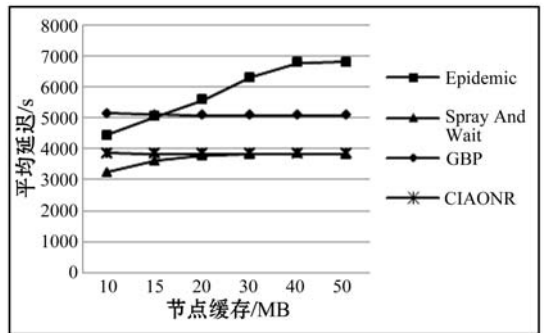


图10 固定节点规模改变缓存-平均延迟

经仿真验证,CIAONR算法较之另外3种算法拥有高且普适的成功投递率,虽然在节点密度较小时,网络开销和网络延迟略高于GBP和Spray And Wait算法,但其差值小,且随着节点规模增加,迅速得到改善,更适用于实际的网络应用。

## 4 结 语

通过对实际网络环境的观察发现,移动节点总是在不同密度网络中随机切换,因此提出一种基于节点接触信息的自适应路由算法——CIAONR算法。CIA-

ONR 算法充分利用各节点采集到的接触信息,依据 CIAONR 约束条件自行判断网络环境并以此指导路由转发,最终完成交付。仿真实验验证了 CIAONR 算法在投递率方面的显著提升,网络开销率也控制在较低范围内,未来将在降低网络延迟方面做进一步研究。

## 参 考 文 献

- [1] 吴磊, 王晓敏, 刘明, 等. 延迟容忍传感器网络中基于群组运动的事件传输[J]. 软件学报, 2012, 23(3): 629-647.
- [2] Zhang H Y, Fan W H, Wang L, et al. Real-Time and reliable greedy geographical routing for mobile wireless sensor networks[J]. Journal of Computer Research and Development, 2009, 46(5): 713-722 (in Chinese with English abstract).
- [3] 张三峰, 黄迪, 陈州. 一种面向机会网络路由的最优停止决策方法[J]. 软件学报, 2014, 25(6): 1291-1300.
- [4] 马华东, 袁培燕, 赵东. 移动机会网络路由问题研究进展[J]. 软件学报, 2015, 26(3): 600-616.
- [5] 任智, 刘智虎, 姚玉坤. 基于网络编码的机会网络高效路由算法[J]. 通信学报, 2013, 34(9): 16-23.
- [6] 杨帆, 来嘉哲. 基于分组的 DTN 路由协议[J]. 电子测量技术, 2015, 38(10): 144-148.
- [7] 谭紫逸, 陈志刚, 吴嘉, 等. 机会网络多跳节点最小传输延迟算法设计[J]. 计算机应用研究, 2016, 33(8): 2458-2461.
- [8] 姚玉坤, 张强, 杨及开. 基于社会组的高投递率机会网络路由协议[J]. 计算机应用研究, 2017, 34(2): 577-581.
- [9] Chen K, Shen H. SMART: Lightweight distributed Social Map based Routing in Delay Tolerant Networks[C]// IEEE International Conference on Network Protocols. IEEE Computer Society, 2012:1-10.
- [10] Bulut E, Szymanski B. Exploiting friendship relations for efficient routing in mobile social networks[J]. IEEE Transactions on Parallel and Distributed Systems, 2012, 23(12): 2254-2265.
- [11] 任智, 谭永银, 索建伟, 等. 一种基于旁听的机会网络路由算法[J]. 计算机应用研究, 2016, 33(11): 3396-3400.
- [12] Zhang X, Neglia G, Kurose J, et al. Performance modeling of epidemic routing[J]. Computer Networks, 2007, 51(10): 2867-2891.
- [13] Chaintreau A, Hui P, Crowcroft J, et al. Impact of Human Mobility on the Design of Opportunistic Forwarding Algorithms[C]// INFOCOM 2006. IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 23-29 A-

pril 2006, Barcelona, Catalunya, Spain. DBLP, 2006:1-13.

- [14] Karagiannis T, Boudec J Y L, Vojnovic M. Power Law and Exponential Decay of Intercontact Times between Mobile Devices[J]. IEEE Transactions on Mobile Computing, 2010, 9(10):1377-1390.
- [15] 程刚, 张云勇, 张勇. 基于人类真实场景的分时段的机会网络移动模型[J]. 通信学报, 2013, 34(S1): 182-189.
- [16] 蔡青松, 牛建伟, 刘燕. 机会网络中的消息传输路径特性研究[J]. 计算机研究与发展, 2011, 48(5): 793-801.
- [17] Tie X, Venkataramani A, Balasubramanian A. R3: robust replication routing in wireless networks with diverse connectivity characteristics [C]// International Conference on Mobile Computing and Networking, MOBICOM 2011, Las Vegas, Nevada, Usa, September. 2011:181-192.

## (上接第96页)

用该路由软件,本文在实验室环境下实际搭建了底层实现的 CCN 网络,并成功进行了文件传输实验。

## 参 考 文 献

- [1] 黄韬, 刘江, 霍如, 等. 未来网络体系架构研究综述[J]. 通信学报, 2014, 35(8): 184-197.
- [2] Lee S W, Kim D, Ko Y B, et al. Cache capacity-aware CCN: Selective caching and cache-aware routing [C]// Global Communications Conference. IEEE, 2014: 2114-2119.
- [3] Jacobson V, Smetters D K, Thornton J D, et al. Networking named content[J]. Communications of the ACM, 2012, 55(1): 117-124.
- [4] Project CCNx[EB/OL]. <http://www.ccnx.org/>.
- [5] Zhang L X, Estrin D, Burke J, et al. Named data networking (NDN) project[J]. Transportation Research Record Journal of the Transportation Research Board, 2010, 1892(1): 227-234.
- [6] 陈震, 陈硕, 马戈. 内容中心网络底层实现方法、内容中心网络以及通信方法: CN, CN 103327027 A[P]. 2013.
- [7] Jumbo frame[OL]. [http://en.wikipedia.org/wiki/Jumbo\\_frame](http://en.wikipedia.org/wiki/Jumbo_frame).
- [8] 宋敬彬. Linux 网络编程[M]. 北京:清华大学出版社, 2015:317-323.
- [9] 陈震, 曹军威. 信息中心网络[M]. 北京:清华大学出版社, 2013:103-122.
- [10] 刘斌, 汪漪. 内容中心网络中名字查找技术的研究[J]. 电信科学, 2014(9): 10-17, 44.