

用遗传算法寻找 OLSR 协议的最小 MPR 集^{*}

张信明^{1,2,3+}, 曾依灵^{1,2}, 干国政^{1,2}, 陈国良^{1,2}

¹(中国科学技术大学 计算机科学技术系,安徽 合肥 230027)

²(国家高性能计算中心(合肥),安徽 合肥 230027)

³(Department of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology, Daejeon 305701, Korea)

Finding the Minimum MPR Set in OLSR Protocol with Genetic Algorithms

ZHANG Xin-Ming^{1,2,3+}, ZENG Yi-Ling^{1,2}, GAN Guo-Zheng^{1,2}, CHEN Guo-Liang^{1,2}

¹(Department of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China)

²(National High Performance Computing Center at Hefei, Hefei 230027, China)

³(Department of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology, Daejeon 305701, Korea)

+ Corresponding author: Phn: +86-551-3603145, Fax: +86-551-3601013, E-mail: xinning@ustc.edu.cn, <http://cs11.ustc.edu.cn>

Zhang XM, Zeng YL, Gan GZ, Chen GL. Finding the minimum MPR set in OLSR protocol with genetic algorithms. *Journal of Software*, 2006,17(4):932-938. <http://www.jos.org.cn/1000-9825/17/932.htm>

Abstract: The characteristic that nodes can enlist into the network topology freely and independently makes mobile Ad hoc networks (MANET) widely used in various environments such as disaster rescue, battlefield and so on. In MANET, the routing mechanism should adapt rapidly to the frequently changed network topology and in the mean time economize valuable network resources with its best. The Optimized Link State Routing Protocol (OLSR) is an important MANET routing protocol in which the key technique is MultiPoint Relays (MPR). After introducing the OLSR protocol and its MPR technique, the shortcoming of presently used heuristic algorithm in finding the minimum MPR sets is revealed. Then the new algorithm based on genetic algorithm (GA) is presented, and the convergence of the algorithm is proved. A series of 4 genetic algorithms are further developed by adopting different GA strategies and simulated in many topologies that are created randomly. Analysis on simulating results shows that the genetic algorithms are feasible and applicable and the choice of heuristic strategies is advisable and appropriate.

Key words: OLSR (optimized link state routing protocol); MPR (multipoint relays); heuristic algorithm; genetic algorithm; mobile ad hoc network

* Supported by the Key Project of the National China Next Generation Internet(CNGI) 2005 (国家“CNGI(下一代互联网示范工程)”专项重点支持项目); the Key Postdoctoral Foundation of the Ningbo City of China under Grant No.2003A61003 (宁波市重点博士科学基金); the Foundation of Science and Technology of Huawei of China under Grant No.YJCB2004036WL (华为科技基金); the International Scholar Exchange Fellowship (ISEF) of the Korea Foundation for Advanced Studies 2005-2006 (韩国高等教育财团国际交换学者奖)

摘要: 节点可以自由、自主地进入网络拓扑的特性,使得移动 Ad hoc 网络(mobile ad hoc network,简称 MANET)被广泛应用于诸如灾难救援、战场等多种环境中.MANET 中的路由要能迅速地适应频繁的网络拓扑结构的变化,同时最大限度地节约网络资源.OLSR(optimized link state routing protocol)协议是一个重要的 MANET 路由协议,而支撑此协议的一个关键技术是 MPR(multipoint relays).在介绍了 OLSR 协议及 MPR 技术之后,揭示了目前启发式算法在寻找最小 MPR 上的弱点,提出了一种基于遗传算法(genetic algorithm,简称 GA)的新算法,并证明了该算法的收敛性.通过采用不同遗传策略将此遗传算法衍生成了 4 个系列算法,并在随机生成的拓扑上对其进行模拟.模拟结果分析显示:提出的遗传算法是可行和适用的,选择的启发式策略也是恰当和正确的.

关键词: OLSR;MPR;启发式算法;遗传算法;移动 Ad hoc 网络

中图法分类号: TP393 **文献标识码:** A

在无线通信技术飞速发展的今天,无线网络因其灵活性、扩展性、方便性而受到人们的喜爱.MANET (mobile ad hoc network)具有自创造、自组织和自管理的特点.网络不依赖于任何固定的网络设施,而是通过移动节点间的相互协作来进行网络互联.针对 MANET 的研究是从军事应用开始的,因此军事应用仍是 MANET 的主要应用领域.但在民用方面,MANET 也有非常广泛的应用前景.在 MANET 中,每个节点兼备路由器和主机两种功能,远距离节点之间的网络互连是通过多跳技术实现的.互连拓扑动态变化的特性,给该领域带来了许多难题.因此,路由问题是 MANET 能够有效应用而必须解决的问题.MANET 的路由算法主要分为两类^[1]:表驱动路由(table-driven)和按需路由(on-demand).这两类路由算法都有各自的优缺点.目前,在 MANET 中主要采用基于需求的路由协议.然而,表驱动类路由协议更适合于 MANET 拓扑动态变化特性及 QoS(quality of service)路由的需求.

目前,国际上关于 MANET 的研究正逐步深入,研究者已经在路由、组播路由、传输调度、功率控制、QoS、传输层等方面发布了相关的研究成果.在众多的难题中,MANET 的路由问题尤其关键,路由协议的研究仍然是 MANET 研究成果最集中的部分.现在面临的挑战是,路由算法如何适应网络拓扑的动态变化以及物理、链路、网络层之间相互影响的问题^[2].国内目前的研究成果主要还是集中在对这些协议的改进^[3-5]上,其他方面的研究正逐步得到重视.而本文的研究主要集中在 OLSR(optimized link state routing protocol)协议的 MPR(multipoint relays)技术上.

1 OLSR 协议中的 MPR

1.1 OLSR协议

OLSR^[6]是一种基于 LS(link state)的表驱动路由协议.MANET 因其节点的移动性使得拓扑结构不断变化,需要对信息及时更新.因此,广播成为一种本质上的需求.每个节点需要不断地向四周的节点发送广播来声明自己在网络拓扑中的位置.信息更新越是及时,就越能有效而精确地处理网络中的移动节点.然而,无线网络的无线特质决定了可用带宽的稀有,而过多的广播信息势必浪费相当宝贵的带宽资源.因此,最优化 MANET 中的广播成为一个极具挑战性的难题,这必须在尽量小的广播发送和广播的可靠性之间作出折衷.为了限制或最优化无线网络中广播的洪泛,研究者已提出了不少方法和技术^[5,7-11].文献[7,8]对洪泛算法的性能进行了评测;文献[9]针对特定物理层考虑了洪泛问题;文献[10]研究了异构 MANET 的洪泛问题;文献[11]将 MPR 技术应用到了组播协议中.

为了限制 MANET 中的广播,OLSR 协议引入了关键技术 MPR^[12].网络中的每个节点都选择自己邻居的一个子集,称为多点中继集(MPR);只有 MPR 中的节点才产生 LS 信息.与纯 LS 协议相比,OLSR 有以下不同(如图 1 所示):只有 A 的 MPR 节点才发送由 A 发起的传播更新;节点 A 的 LS 更新包减小了,因为更新信息中只包括那些选择 A 作为它们 MPR 的节点信息,因此,传播的只是局部的网络拓扑信息,并且 A 只可能通过它的 MPR 到达.在 OLSR 协议中,一个节点通过它的邻居节点的信息以及其他所有节点的 MPR 信息来计算到某个目

的节点的最短路径.OSLR 对节点密度大的网络是非常适用的.此时,MPR 技术显得非常有效;当网络十分稀疏时,OLSR 就退化为一般的 LS 算法.

1.2 MPR的计算

节点 A 的 MPR 集合是满足下面条件的节点集合: MPR 中的每个节点都是 A 的一跳邻居节点; 通过 MPR 中的节点,A 可以将信息传播到所有的二跳节点.为了计算一个节点的 MPR,必须知道它两跳以内的 LS.一个邻居在发送 HELLO 信息时,同时附上自己的邻居信息.这样,一个节点接收到自己邻居传来的 HELLO 信息之后,就可以计算出自己两跳以内的 LS,每个节点就可以独立地计算自己的 MPR 集合.

显然,为了有效地抑制网络中控制信息的泛滥,应该计算出尽可能小的 MPR 集合.图 1 给出了节点 A 在两跳范围内的 LS,并可由此计算出它的最小 MPR 集合{E,F,G}.在通常的泛洪方法下,A 的 7 个邻居节点都需要转发 A 的信息;但有了最小 MPR 之后,只需要 3 个节点的转发就可以达到同样的效果,并且有效地节约了网络资源.

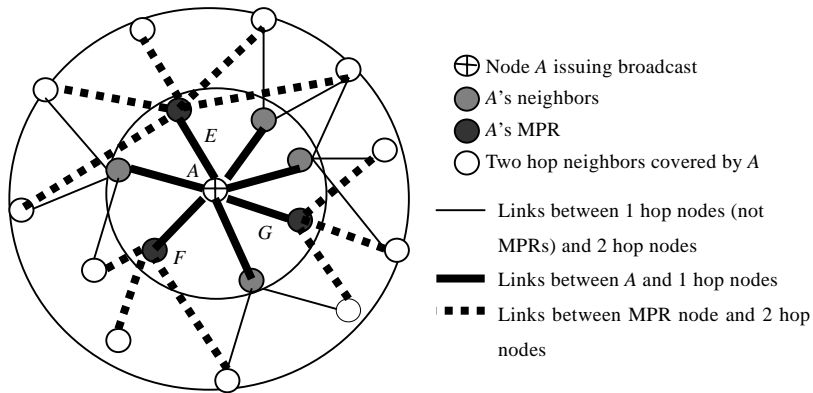


Fig.1 The link state and its MPRs of Node A within 2 hops

图 1 A 节点 2 跳范围内的链路信息及其 MPR

但是,文献[12]中证明了最小 MPR 问题是一个 NP 完全问题.因此,要计算出一个节点的 MPR 相当困难,只能利用一些启发式算法去寻找它的近优解.假定 A 节点的一跳邻居集为 $N(A)$,两跳邻居集为 $N_2(A)$ ($N_2(A)$ 中不包括 $N(A)$ 的节点),文献[12]中提出一种基于贪心策略的启发式算法: 置 A 的 MPR 集合为空 $MPR(A)=\emptyset$;

$N(A)$ 中的一个节点是 $N_2(A)$ 中某个节点的唯一一跳邻居,将此节点放入 $MPR(A)$; 当 $N_2(A)$ 中仍有未被 $MPR(A)$ 覆盖的节点时:(a) 对于每个 $N(A)$ 中未被选入 $MPR(A)$ 的节点 J ,计算其在 $N_2(A)$ 覆盖的未被 $MPR(A)$ 所覆盖的节点数 $C(j)$;(b) 将 $C(j)$ 值最大的节点加入 $MPR(A)$.

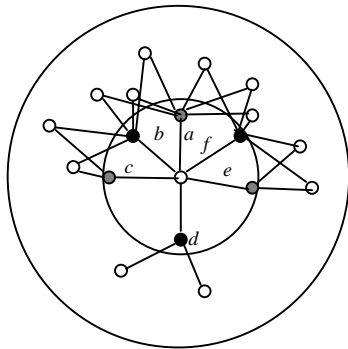


Fig.2 Heuristic algorithms may not find optimal solution for MPR

图 2 启发式算法未必能找到 MPR 最优解

以上算法可在较短的时间($O(n \times \log(n))^{[12]}$)内找到 MPR 问题的一个不错的解,但由于贪心算法自身的局限性,所找到的解却未必是最优解.比如以上算法在处理如图 2 所示的拓扑结构时,首先通过 Step2 选择节点 d,再通过 Step3 依次选择 a,b,f,于是得到 $MPR=\{a,b,d,f\}$,但此图的最小 MPR 却是 $\{b,d,f\}$.

本文将提出的是一个计算 MPR 的 GA(genetic algorithm),并在第 3 节中证明它能够最终收敛到最优解.由于 MPR 问题的 NP 完全性,使其非常适合于用 GA 进行求解.同时,由于计算 MPR 是每个节点独立地进行,只需考虑两跳以内的 LS,即使在网络相当庞大、稠密的情况下,两跳以内的拓扑信息也不会复杂到让遗传算法耗费大量运行时间的地步.因此,GA 能在较短的时间内得到一个相当令人满意的解,使其在动态的拓

扑结构中也能及时计算出 MPR,以反映最新的拓扑信息.

2 MPR 问题的遗传算法

GA^[13]本质上是一个通过群体的迭代来不断优化的过程.它简单、通用,鲁棒性强,应用广泛,很适合解决 NP 完全问题.为方便算法的描述,将涉及到的节点分成 3 类: 节点 A ; 节点 A 的一跳邻居节点.这些节点放入一跳邻居表 NB_table1 并用 $N_1^1, N_2^1, \dots, N_n^1$ 表示,其中 n 为一跳邻居节点的个数; 节点 A 的二跳邻居节点,即节点 A 需要通过两跳才能达到的邻居节点(不包括一跳邻居表中的节点).把这些节点放入二跳邻居表 NB_table2 并用 $N_1^2, N_2^2, \dots, N_m^2$ 表示,其中 m 为二跳邻居节点的个数.通常情况下, m 为 n 的倍数.

2.1 编 码

对于 MPR 问题,我们采用的编码方式是对一跳邻居表进行编码.如果节点 N_i^1 被选入 MPR,则对应位置 1, 否则置 0.此种编码方式使得染色体长度较短(长度为 n),而且每一位只有 0 或 1 两种取值,使得 GA 的搜索空间较小.但是,此种编码也会带来一些不足: 随机生成的初始个体不能保证是一个 MPR,即将染色体每一位随机置为 0 或 1 所得到的初始个体,其所代表的节点集合未必能覆盖所有的二跳邻居; 交叉和变异之后也可能破坏一个个体的 MPR 性质,即代表两个不同 MPR 的个体在交叉和变异之后所得到的后代,很可能并非一个 MPR.但这些问题都可以通过维护操作得到解决.

2.2 初始群体的选择

如何产生初始个体.为了使生成的每个个体都能代表一个 MPR 集合,我们用如下的方法随机产生初始个体:二跳邻居表中的每一个节点都随机地选择一个覆盖自己的一跳邻居节点,被选中的一跳邻居节点对应的染色体位置 1.由于二跳邻居都有覆盖自己的一跳邻居,很显然,这样选出来的初始个体一定是一个 MPR.而且,这样的初始算法还有启发式的效果:希望选择那些覆盖二跳邻居多的一跳邻居,因为这样的节点很可能就在最小的 MPR 之中.事实上,节点覆盖的二跳邻居越多,有机会选到它的二跳邻居就越多,被选中的概率也就越大.

初始个体的规模.若群体规模太小,算法的优化性能不会很好,而且容易陷入局部最优解;若群体规模太大,又将使得算法的计算复杂度太高,导致耗时太长.经过不断的模拟实验发现:当群体规模取和一跳邻居表同一个数量级时,算法有较好的性能,同时也能很快地逼近最优解.

2.3 适应度函数

在 GA 中,适应度函数需要进行比较排序,并在此基础上计算选择概率,因此,适应度函数必须为正值.通过以下方式直接将 MPR 中节点的个数映射成适应度函数,使得所选的节点数越少,适应度函数越高: $Fitness(i) = NB_table1.size - gen[i].number$.其中 $NB_table1.size$ 为一跳邻居表中的节点数; $gen[i].number$ 为群体中第 i 个个体所选择的 MPR 集的大小.为了让算法有更好的性能,我们还将进行适应度函数的定标.利用 Gillies 提出的乘幂标,将适应度函数定标为 $Fitness'(i) = Fitness(i)^K$.实验表明:当 K 取 1.05~1.005 之间时,所取得的性能最好.

2.4 选择方法

本遗传算法采用适应度比例方法(fitness proportional model),也称为赌轮法或蒙特卡罗(Monte Carlo)选择方法.其中,当代群体中第 i 个个体被选中的概率 sp_i 为 $sp_i = Fitness(i) / \left(\sum_{j=1}^{gen_size} Fitness(j) \right)$.算法还在新一代个体产生之后采用最佳个体保存方法(elitist model),即如果新一代群体中适应度最高的个体适应度小于老一代最佳个体,而把老一代最佳个体保存入新一代.

2.5 交叉策略

在本文涉及的遗传算法中,考虑以下两种交叉策略: 一点交叉:这是一种基本的交叉策略,又称为简单交叉.它在个体串中随机选择一个交叉点,实行交叉时,该点前或后的两个个体部分结构进行交换,并生成两个新的个体; And_OR 交叉:由于所有个体的染色体位长相同,而且每一位只有 0 或 1 两个取值,所以可以进行这样

的交叉操作:将两个染色体的每位相与,得到一个新的个体;将两个染色体每位相或,得到另一个新个体.由遗传算法的选择特性可知,适应度大的个体被选择的概率也大,而适应度大的个体中的相同节点很可能就是它们优秀性能的保证,也很可能在最终的最小 MPR 当中.而无论是 AND 操作还是 OR 操作,无疑都将保留这些节点.这样的交叉操作,由于保留了个体中的优良因子,其性能也很可能比第 1 种交叉更优(后面的模拟实验也证明了这一点).

2.6 变异操作

因编码方式选择了二进制位串,当采取位变异(即每一位以一定的概率被选中进行变异)时,该位的值取反.变异概率的选择也是值得考虑的因素之一:若变异概率太大,会导致搜索的不稳定;若变异概率太小,又很难突破局部解.本算法在变异概率的选择上经过了不断的实验,实验表明,变异概率 $p_m \in [0.005, 0.05]$ 时效果较好.

2.7 维护操作

维护操作不属于基本的 GA,但在本文的算法中,因编码方式这一原因而显得必不可少.无论是交叉还是变异,都很有可能破坏个体的 MPR 性质.经过分析发现问题出在两个方面:新个体中的节点无法覆盖所有的二跳节点;新个体中有冗余节点,它所覆盖的所有节点已被其余节点覆盖.本算法指定两种维护策略,其操作都从二跳节点出发:随机策略:对二跳表中每一个节点,若该节点未被新个体的任何一个节点覆盖,随机为它指定一个覆盖它的节点;若该节点有多个覆盖它的一跳节点,随机在其中选择一个,抛弃其他;启发式策略:对二跳表中每一个节点,若该节点未被新个体的任何一个节点覆盖,指定覆盖它的一跳邻居中覆盖能力最强的一个(其覆盖的二跳节点最多);若该节点有多个覆盖它的一跳节点,选择覆盖能力最强的一个,抛弃其他.

3 收敛性分析

本文提出的遗传算法能够收敛到全局最优解.

证明:在本文提出的遗传算法中:交叉概率为 1;变异概率为 $p_m \in (0, 1)$;交叉个体的选择采用适应度比例方法,并在新群体产生后采用最佳个体保存方法.根据文献[13]的定理 2.6,具有变异概率 $p_m \in (0, 1)$ 、交叉概率 $p_c \in [0, 1]$,同时采用比例选择法,且在选择后保留当前最优解的遗传算法,最终能收敛到全局最优解.故本文提出的算法可以收敛到全局最优解.

4 模拟实验

在本文的模拟实验中,采取与文献[12]相似的拓扑结构:在 50×50 的网格上随机地生成 300 个节点,每一个节点都落在网格坐标上.为这些节点定义一个覆盖半径,在节点 A 的覆盖半径内的所有节点都将视为 A 的一跳邻居.通过调节覆盖半径,能够有效地控制节点一跳邻居的多少,也就相应地模拟了网络的疏密状况.为了考虑最复杂最稠密的状况,实验中总是选取网格正中的节点作为计算对象.通过调节覆盖半径,我们使算法最多能支持 50 个一跳节点.通过策略的选择,生成以下 4 个遗传算法以比较其性能的优劣(见表 1).

Table 1 The selection of strategies for genetic algorithms

表 1 各遗传算法的策略选择

	GAMPR1	GAMPR2	GAMPR3	GAMPR4
Cross over strategy	One point	AND_OR	One point	AND_OR
Fitness function	Fitness(*)	Fitness'(*)	Fitness(*)	Fitness'(*)
Maintenance strategy	Random	Random	Heuristic	Heuristic
Mutation probability	0.01	0.03	0.01	0.05

算法在 Intel Celeron 2.00GHz(512M DDR)的微机运行.对于每个随机生成的拓扑和指定的覆盖半径,GAMPR1,GAMPR2,GAMPR3 和 GAMPR4 都将执行 20 次,以获取其平均性能.图 3 中给出在不同拓扑情况下,4 个遗传算法的收敛情况.其中横坐标为发送节点的一跳邻居数;纵坐标为各算法找到最优解所需的平均代数.图 4 中则给出各算法找到最优解所需的平均时间,时间单位为 ms.

由实验结果可以看出:未用启发式策略的 GAMRP1 无论在收敛代数还是收敛时间上,都不如后面的几个遗传算法好,算法的性能随一跳邻居数的变化和网络拓扑的变化而波动较大;而拥有最多启发式策略的 GAMRP4,具有最好的性能,并且显得相当稳定,收敛代数和收敛时间随网络拓扑变化得并不明显,甚至在一跳邻居数接近 50 的情况下,它仍然能在短短的几代以内找到最优解,耗费的时间也仅为数毫秒.当然,如果把图 4 放大数倍,可以看到:在一跳邻居相当稀疏的情况下,GAMRP4 的执行时间反而略高于其余 3 个算法.这是因为当网络拓扑简单时,无须启发式策略就能迅速找到最优解,而启发式策略的应用又导致了每一代执行时间的略微增长.在这种情况下,GAMRP4 执行时间反而显得略慢.GAMRP2 和 GAMRP3 的性能则处于 GAMRP1 和 GAMRP4 之间,其中 GAMRP2 显得更为稳定.

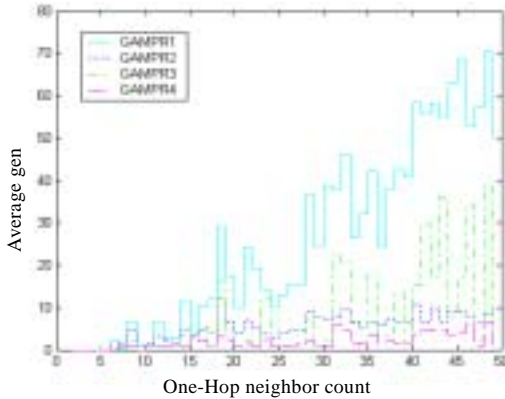


Fig.3 Average generation of genetic algorithm convergence

图 3 遗传算法收敛的平均代数

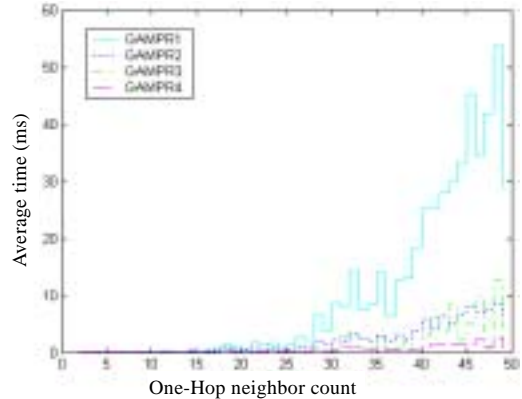


Fig.4 Average time of genetic algorithm convergence

图 4 遗传算法收敛的平均时间

5 总结

MANET 中的路由要能迅速适应频繁的网络拓扑结构的变化,同时最大限度地节约网络资源.包括 OLSR 在内的表驱动类路由协议更适合于 MANET 动态变化特性及 QoS 路由的需求.针对 OLSR 无线路由协议中的 MPR 问题,本文给出了遗传算法的解决思路,并细化成了 4 个相互比较的特定遗传算法.这些算法都做到: 采用恰当的编码方式,以有效地减少编码空间和搜索空间; 通过遗传策略的选择,以使算法一定能收敛到最优解; 实验决定变异参数,以使每个算法的性能最优; 合理选择启发式策略,以进一步提高算法性能.模拟实验表明我们提出的遗传算法有着不错的性能.即使在稠密的网络环境下,即使未使用任何启发式策略的遗传算法,也能在令人满意的时间内收敛到最优解;实验也表明了我们选择的启发式策略的合理性与正确性,因为采用最多启发式策略的 GAMRP4 在最小的代数和最短的时间内找到了最优解.根据我们研究范围所及,并未发现以前有研究者试图用遗传算法解决过 MPR 问题.在下一步的研究中,我们将对遗传算法在无线路由领域的应用作更深入的研究和探索.

References:

- [1] Abolhasan M, Wysocki T, Dutkiewicz E, Abolhasan M. A review of routing protocols for mobile ad hoc networks. *Ad Hoc Networks*, 2004,2:1-22.
- [2] Setton B, Yoo T, Zhu XQ, Goldsmith A, Girod B. Cross-Layer design of ad hoc networks for real-time video streaming. *IEEE Wireless Communications*, 2005,(8):59-65.
- [3] Xie F, Zhang XM, Guo JF, Chen GL. Delay oriented adaptive routing protocol for mobile ad hoc networks. *Journal of Software*, 2005,16(9):1661-1667 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/16/1661.htm>

- [4] Guo JF, Zhang XM, Xie F, Chen GL. Leisure degree adaptive routing protocol for mobile ad hoc network. Journal of Software, 2005,16(5):960-969 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/16/960.htm>
- [5] Zou SH, Wu HT, Cheng SD. A simple and efficient broadcasting scheme for mobile ad hoc networks. Journal of Software, 2005, 16(8):1104-1111 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/16/1104.htm>
- [6] Clausen T, Jacquet P. Optimized link state routing protocol (OLSR). IETF RFC3626, 2003.
- [7] Mans B, Shrestha N. Performance evaluation of approximation algorithms for multipoint relay selection. In: Proc. of the 3rd Annual Mediterranean Ad Hoc Networking Workshop. 2004. <http://www.ece.osu.edu/medhoc04/>
- [8] Guo H, Ingelrest F, Simplot-Ryl D, Stojmenovic I. Performance evaluation of broadcasting protocols for ad hoc and sensor networks. In: Proc. of the 4th Annual Mediterranean Ad Hoc Networking Workshop. France, 2005. <http://www.site.uottawa.ca/~ivan/performance.pdf>
- [9] Ingelrest F, Simplot-Ryl D. Maximizing the probability of delivery of multipoint relay broadcast protocol in wireless ad hoc networks with a realistic physical layer. Technical Report, INRIA, 2005. <http://www.lifl.fr/RD2P/uploads/Papers/ingelrest-tr-05-2.pdf>
- [10] Lipman J, Boustead P, Judge J. Utility-Based multipoint relay flooding in heterogeneous mobile ad hoc networks. In: Proc. of the Workshop on the Internet Telecommunications and Signal Processing (WITSP 2002). Australia, 2002. <http://citeseer.ist.psu.edu/672512.html>
- [11] Zhao Y, Xu LM, Shi ML. On-Demand multicast routing protocol with multipoint relay (ODMRP-MPR) in mobile ad-hoc network. In: Proc. of the ICCT 2003. http://www.stanford.edu/class/ee360/lec12_sp3.pdf
- [12] Qayyum A, Viennot L, Laouiti A. Multipoint relaying: An efficient technique for flooding in mobile wireless networks. In: Proc. of the 35th Hawaii Int'l Conf. on System Sciences (HICSS 2002). 2002. <http://ieeexplore.ieee.org/iel5/7798/21442/00994521.pdf?arnumber=994521>
- [13] Chen GL, Wang XF, Zhuang ZQ. Genetic Algorithm and Its Application. Beijing: People's Post and Telecommunications Press, 1996 (in Chinese).

附中文参考文献:

- [3] 谢飞,张信明,郭嘉丰,陈国良.延迟主导的自适应移动 Ad hoc 网络路由协议.软件学报,2005,16(9):1661-1667. <http://www.jos.org.cn/1000-9825/16/1661.htm>
- [4] 郭嘉丰,张信明,谢飞,陈国良.基于节点空闲度的自适应移动 Ad hoc 网络路由协议.软件学报,2005,16(5):960-969. <http://www.jos.org.cn/1000-9825/16/960.htm>
- [5] 邹仕洪,邬海涛,程时端.一种移动自组网中简单高效的广播算法.软件学报,2005,16(6):1104-1111. <http://www.jos.org.cn/1000-9825/16/1104.htm>
- [13] 陈国良,王煦法,庄镇泉.遗传算法及其应用.北京:人民邮电出版社,1996.



张信明(1964 -),男,安徽天长人,博士,副教授,KAIST 访问教授,主要研究领域为无线网络,IP 网络 QoS 控制,网络性能分析.



于国政(1976 -),男,硕士生,主要研究领域为计算机网络.



曾依灵(1981 -),男,硕士生,主要研究领域为计算机网络.



陈国良(1938 -),男,教授,博士生导师,中国科学院院士,CCF 高级会员,主要研究领域为并行算法及其应用,并行计算机体系结构.