

一种高效的无线 Ad-Hoc 网络自适应资源管理技术研究

杜建清

(丽水职业技术学院 机电信息分院, 浙江 丽水 323000)

摘 要: 本文针对无线 Ad Hoc 网络的高度动态性、网络拓扑动态变化、链路不稳定等问题, 提出了一种无线网络资源管理技术. 该技术通过探测邻居节点, 然后无线资源进行动态划分设置请求优先级从而建立动态划分资源优先级机制. 实验表明, 提出的方法能够快速进行查询, 有效提高了 Ad-Hoc 网络中资源发现的性能和系统的稳定性, 在时延方面也有一定的提高.

关键词: 无线 Ad Hoc 网络; 资源管理; 转发率; 时延

中图分类号: TP393

文献标识码: A

文章编号: 1000-7180(2012)07-0090-04

Research of an Efficient and Adaptive Resource Discovery and Management Technology for Wireless Ad Hoc Networks

DU Jian-qing

(Lishui Vocational Technology Institute, Lishui 323000, China)

Abstract: For wireless Ad Hoc networks are highly dynamic, dynamic topology changes, the link is not stable wait for a problem, put forward a wireless cyber source management technology. The technique by detecting the neighbor node, then the wireless resource dynamic partition setup request priority so as to establish the dynamic partition resources priority mechanism. Simulation results show that, the technology is capable of fast inquiry, not only improves the Ad Hoc network resource discovery performance and stability of the system, and compared with the prior art in the forward rate, delay and other aspects of good performance.

Key words: wireless Ad Hoc networks; resource discovery; flooding; adaptively

1 引言

随着无线移动网络动态拓扑技术等发展, 同时无线 Ad Hoc 网络自身的特点使其需要合适的资源发现算法与之相适应, 这关系到 Ad Hoc 网络中资源搜索的性能和效率, 因此无线 Ad Hoc 网络对于移动节点资源的发现和管理成为国内外学者研究热点之一.

在本文的文献[1]提出了一种泛洪算法. 在这种算法中, 一个节点发送一个消息给它所有的邻居节点, 这些邻居节点分别将该消息转发给它们自己的邻居节点, 直到该消息发布给整个网络. 该算法具有实现简单、健壮性高的优点, 成为资源发现的主要手

段. 然而, 由于该算法未能有效抑制冗余泛洪包和有效减少信道冲突概率, 使得每次泛洪过程都要消耗很大的能量, 而且随着网络规模扩大、节点密度增加, 这种情况将更加严重, 这在 Ad Hoc 网络中是不可行的. Ouri Wolfson 提出了一种 RBB 算法^[2], 每个运动着的节点周期性地广播该节点所知道的最相关的资源和查询信息到它的邻近节点. 该算法由于不依赖已确定的路由结构, 故更适应 Ad Hoc 网络高度动态变化的环境; 但却增加了每个节点的负荷. 文献[3]研究了一种无线资源发现、监测和管理的基于智能空间架构的低容量节点. 文献[4]分析了适用于海上水下作业的水下无线 Ad Hoc 网络资源管理机制. 文献[5]提出了一种无线网络资源基于共同发

收稿日期: 2011-12-10; **修回日期:** 2012-01-10

基金项目: 国家自然科学基金(50875245); 浙江省自然科学基金(Y107558)

现机制的同步管理算法. 文献[6]在具有能量监测功能的入侵检测移动无线 Ad Hoc 网络分析了资源管理的特性. 但是, 文献[3-5]均未对无线 Ad Hoc 网络中相邻节点对于资源管理的影响. 文献[6]忽略了动态特性对于资源管理的影响.

本文提出了一种无线网络资源管理技术. 该技术通过探测邻居节点, 然后无线资源进行动态划分设置请求优先级从而建立动态划分资源优先级机制. 实验表明, 提出的方法能够快速进行查询, 有效提高了 Ad-Hoc 网络中资源发现的性能和系统的稳定性, 在时延方面也有一定的提高.

2 ERDA 算法

2.1 探测邻居节点

Ad Hoc 网络中的每个节点周期性的广播 HELLO 信息给它的邻居节点. HELLO 信息包括节点的标识符和一个确认收到信息的确认列表. 当收到邻居节点的 HELLO 信息后, 节点扫描接收到的确认列表[7], 并更新自己的确认列表. 然后, 节点根据更新后的确认列表丢弃已经执行过的请求. 因为, 请求已经执行过了, 没有必要再传播这些请求, 确认列表是减少在网络中传播冗余信息的方法之一. 本文提出的算法在传输之前, 节点首先核实哪个节点是其邻居节点, 并把现在的邻居节点和以前传输信息的邻居列表进行比较, 从而决定是否传输信息. 如果和陈旧的邻居节点相比, 新邻居节点太少, 须等待足够的新邻居节点进入其传输范围, 这样避免了传输冗余的请求信息和资源. 相反, 如果新邻居节点的比例高于陈旧邻居节点的比率, 开始准备传输.

为了便于分析, 无线 Ad Hoc 网络节点不按照固定的时间间隔传输信息, 因为没有必要消耗带宽来为相同节点传输之前传输过的相同数据.

2.2 动态划分请求优先级

如果是肯定的, 节点要根据可用的带宽来决定将哪些数据列入传输列表, 同时, 节点以前没有执行成功的请求还需要在网络中传播^[1]. 由于带宽的限制, 算法需要选择在某个时间点在网络中非常重要的请求来执行. 因此, 有必要对请求信息进行动态的划分等级, 并选择优先级高的请求进行传播.

在该算法中, 通过以下的排序函数来划分请求的优先级.

$$f(t) = \frac{1}{\delta \sqrt{2\pi}} e^{-\frac{(t-\mu)^2}{2\sigma^2}} * n \quad (1)$$

其中, $f(t)$ 是请求的排名函数; t 是请求时间, 表示请求在网络中已经有效的的时间; n 是请求, 表示请求已经在网络中经过的跳数. δ 和 μ 代表最长时间和门槛时间.

从公式(1)可以看出, 决定请求的优先级主要有请求时间 t 和跳数 n 两个参数. 如果 t 值越大, 则 $f(t)$ 值越小, 表示请求的重要性越小, 这说明请求已经非常陈旧; 反之, t 值越小, 表示请求越新, 应该在网络中传播. 如果 n 值越大, 则 $f(t)$ 值越小, 表示请求的重要性越小, 因为该请求经过了很多跳数, 也就说明很多节点可能已经接收到了这个请求, 它的重要性降低了; 反之, n 值越小, 表示请求越新, 应该在网络中传播.

最后, 将划分好优先级的请求保存到节点自身的数据库中. 优先级高的前几个资源进入列表准备传播给邻居节点.

2.3 动态划分资源优先级

资源是一个节点在网络中请求的实际的数据. 在传输的时候, 每个节点有一系列要传输的资源. 在网络中传输一个节点的所有资源是不可能的. 资源需要进入节点自身数据库列表方可发送. 因此, 有必要对资源划分优先级. 划分资源优先级, 要考虑网络中请求资源的时间, 这些划分等级的请求表明了网络中对资源的重要性. 对划分资源优先级来说, 另一个重要的因素是网络中有多少节点具有特殊的资源, 这可以利用资源标志来计算, 资源标志类似于请求的标志. 可以利用下面的函数来划分资源优先级.

$$f(d_i) = \frac{\#n(d_i)}{\#(d_i)} \quad (2)$$

其中, d_i 代表资源或数据. $\#n(d_i)$ 代表请求 d_i 的节点数量. $\#(d_i)$ 代表拥有 d_i 的节点数量. $\#n(d_i)$ 可以由前一阶段划分优先级的请求来决定. $\#(d_i)$ 可以根据资源的标识来决定, 每个资源具有一个标识, 该标识表明有多少节点拥有这个资源. 比如, 对于一个资源 $r[A, B, C]$, $\#(d_i)$ 的值是 3, 因为 r 存在于节点 A, B, C . 选择 $\#n(d_i)$ 和 $\#(d_i)$ 这两个参数的作用在于知道在某个时间资源的重要性. 从公式(2)可以看出, 资源的优先级随着 $\#n(d_i)$ 值的增大而增大, 随着 $\#(d_i)$ 值的增大而减小.

在该阶段最后, 将划分好优先级的资源保存到节点自身的数据库中. 优先级高的前几个资源进入列表准备传播给邻居节点.

2.4 传播请求和资源

算法的目标是把优先级高的请求和资源传送给

它的邻居节点. 在前两个阶段, 确定了节点请求和资源的优先级. 传送请求和资源主要担心带宽的不足, 由于带宽的限制, 算法需要确定有多少请求和资源需要被传送. 文中算法动态的为请求和资源的传送分配带宽, 通常, 有固定数目的请求和资源需要传输, 但是, 也可动态的变化, 这需要为传送请求和资源分配的带宽相应的动态变化. 预先把请求和资源的带宽分配比例定为 2:3, 假设, 为一个节点的传输分配 100 kb 的带宽, 每个请求和资源的数据量大小是 8 kb. 则根据请求和资源的带宽比例 2:3, 在分配的带宽里可以传输 5 个请求和 7 个资源.

如果按照请求和资源固定的比例利用带宽, 结果将是低效的. 假设为请求分配的带宽一次可以传输 5 个请求, 但是, 只有一个新的请求需要传输, 其余的请求在过去经常传输. 那么, 其余的带宽被不太重要的请求占用, 也就是带宽被浪费了. 因此, 文中采用动态传输机制, 即动态的调整请求和资源的带宽比例来进行传输. 假设一个节点存在 9 个请求和 12 个资源, 如表 1 所示. 对于这里讨论的问题, 存在如下四种情况:

表 1 节点请求和资源

请求	$f(t)$	资源	$f(d_i)$
q_1	0.092	r_1	2.2
q_2	0.083	r_2	1.85
q_3	0.080	r_3	1.71
q_4	0.076	r_4	1.64
q_5	0.066	r_5	1.52
q_6	0.052	r_6	1.50
q_7	0.048	r_7	1.49
q_8	0.044	r_8	1.45
q_9	0.036	r_9	1.30
		r_{10}	1.28
		r_{11}	1.19
		r_{12}	1.13

第一种情况, 如果 $f(t) > 0.05$, 同时 $f(d_i) > 1.40$, 则有 6 个请求 $f(t)$ 值高于设定值, 8 个资源 $f(d_i)$ 值高于设定值, 因为最多有 5 个请求和 7 个资源被传送, 所以优先级高的前 5 个请求和前 7 个资源被传输. 请求 q_6 和资源 r_8 没有被传播出去, 虽然它们的值分别高于请求设定值和资源设定值.

第二种情况, 如果 $f(t) > 0.07$, 同时 $f(d_i) > 1.29$, 则有 4 个请求 $f(t)$ 值高于设定值, 9 个资源 $f(d_i)$ 值高于设定值, 而被分配的带宽只有 100kb, 所以优先级高的前 4 个请求和前 8 个资源被传输. 资源 r_9 没有被传播出去, 虽然它的值高于资源设定值.

第三种情况, 如果 $f(t) > 0.04$, 同时 $f(d_i) > 1.51$, 则有 8 个请求 $f(t)$ 值高于设定值, 5 个资源 $f(d_i)$ 值高于设定值, 需要传输的请求数量大于预先设置的请求数量, 对于 8 个请求, 只有 7 个请求考虑被传播, 因为它们达到了预先规定的带宽. 所以优先级高的前 7 个请求和前 5 个资源被传输. 请求 q_8 没有被传播出去, 虽然它的值高于资源设定值.

第四种情况, 如果 $f(t) > 0.07$, 同时 $f(d_i) > 1.49$, 则有 4 个请求 $f(t)$ 值高于设定值, 6 个资源 $f(d_i)$ 值高于设定值, 而被分配的带宽可以传输更多的数据, 所以优先级高的前 5 个请求和前 7 个资源被传输.

3 算法分析验证

ERDA 算法的主要特点表现在:

(1) 高效. ERDA 算法的高效主要体现在 2 个方面, 一是利用确认列表机制, 能大幅度减少冗余泛洪包; 二是采用动态传输机制, 使得物理信道冲突概率大大降低, 减少了冗余包的重传机率.

(2) 短时延. 由于 ERDA 算法采用了动态传输机制, 使得时延变低而且增加比较平缓.

(3) 便于在无线移动节点上实现而且控制开销低, 具有较好的技术兼容性. ERDA 算法仅需要在每个节点内部维护一张邻居信息列表以及在每个泛洪包中包含发送节点的 ID, 具有较低的控制开销, 易于与其它路由协议结合.

4 实验分析

本文为了验证 ERDA 算法的优越性, 本文采用 NS-2 工具仿真实验对该算法进行性能分析. 基于算法比较的公平性, 与泛洪算法和 RBB 算法进行性能对比.

在仿真实验中, 网络拓扑范围为 $800\text{ m} \times 800\text{ m}$, 节点的传输半径固定为 100m, 网络中不同节点的移动速度在 $3 \sim 15\text{ m/s}$ 范围内随机给定, 而对于每个节点其移动速度在测试过程中不再改变; 节点数量以 50 个节点为增量在 $[50, 300]$ 范围内变化.

图 1 给出了三种算法的转发率与节点数量的变

化趋势.由图1可知,在节点数量增加(节点密度增大)的情况下,BFA算法因为采用盲目转发机制,使得网络中所有节点都要转发一次泛洪包,因此,转发率为100%,RBB算法虽能抑制泛洪包,可抑制45%左右,但效果不是很好.而ERDA算法则能抑制率高达80%以上.

时延与节点数量的变化趋势如图2所示.发现,在节点数量增加(节点密度增大)的情况下,BFA算法的时延随之增加,这是因为影响BFA算法时延的主要因素是泛洪包冲突与竞争信道,因此在这种情况下,时延会随着增大;RBB算法由于采用了排序策略而对网络泛洪延时不敏感;而ERDA算法由于采用了动态传输机制,使得时延变低而且增加比较平缓.

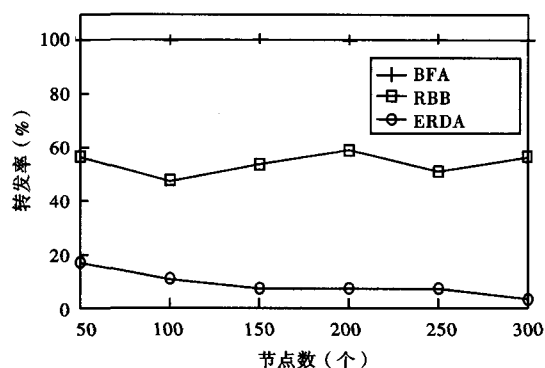


图1 转发率与节点数量的关系

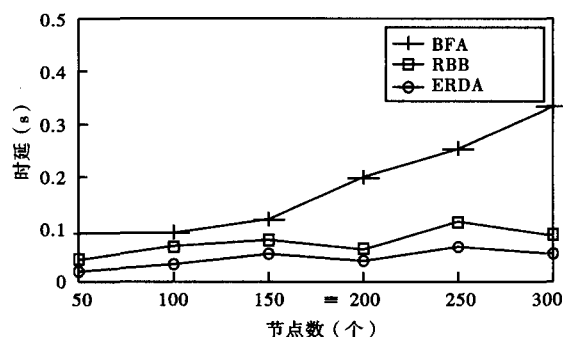


图2 时延与节点数量的关系

5 结束语

本文提出一种高效的自适应资源发现和管理控制技术.能有效抑制泛洪包,缩短时延,适用于大规模的无线 Ad Hoc 网络,同时验证了该技术的有效性和可行性.

参考文献:

- [1] Cardoso M, Chandra A. Resource bundles: using aggregation for statistical large-scale resource discovery and management[J]. IEEE Transactions on Parallel and Distributed Systems, 2010, 21(8): 1089-1102.
- [2] Naseer A, Stergioulas L K. Web-services-based resource discovery model and service deployment on healthgrids [J]. IEEE Transactions on Information Technology in Biomedicine, 2010, 14(3): 838-845.
- [3] Ozcelebi T, Lukkien J, Bosman R, et al. Discovery, monitoring and management in smart spaces composed of low capacity nodes[J]. IEEE Transactions on Consumer Electronics, 2010, 56(2): 570-578.
- [4] Foo K Y, Atkins P R, Collins T, et al. Sea trials of an underwater, ad hoc, acoustic network with stationary assets[J]. IET Radar, Sonar & Navigation, 2010, 4(1): 2-16.
- [5] Vigato A, Vangelista L, Measson C, et al. Joint discovery in synchronous wireless networks [J]. IEEE Transactions on Communications, 2011, 59(8): 2296-2305.
- [6] Boppana R V, Xu Su. On the effectiveness of monitoring for intrusion detection in mobile Ad Hoc networks [J]. IEEE Transactions on Mobile Computing, 2011, 10(8): 1162-1174.
- [7] Al-Oqily I, Karmouch A. SORD: a fault-resilient service overlay for mediaport resource discovery[J]. IEEE Transactions on Parallel and Distributed Systems, 2009, 20(8): 1112-1125.

作者简介:

杜建清 男,(1970—),讲师.研究方向为计算机网络.