

移动Ad hoc网络中基于能量的自适应广播策略

孙丹丹, 苗建松, 丁 炜

(北京邮电大学电子工程学院 北京 海淀区 100876)

【摘要】为了在广播中避免产生广播风暴,提出了能量自适应转发的泛洪广播算法。该算法使用自适应调节转发概率的方法来减少信息不必要的重传,通过剩余能量最高的邻节点进行转播,减少了分组在网络中的重复传输,降低节点间的竞争和碰撞,有效抑制了泛洪。分析和仿真表明自适应广播策略比传统的泛洪方法具有明显的优越性,提高了传输成功率。

关键词 Ad hoc网络; 广播; 能量自适应; 抑制泛洪
中图分类号 TN929 **文献标识码** A

Energy Adaptive Flooding Broadcast Strategy for Ad hoc Network

SUN Dan-dan, MIAO Jian-song, and Ding Wei

(School of Electronic Engineering, Beijing University of Posts and Telecommunications Haidian Beijing 100876)

Abstract In order to avoid broadcast storm, the energy adaptive flooding broadcast (EAB) strategy is proposed. The algorithm is based on probability rebroadcast to reduce the number of rebroadcast. According the remain energy of the nodes, EAB decides which nodes need to rebroadcast to reduce the chance of contention and collision among neighboring nodes in the networks and restrain the flooding effectively. Analyses and simulations show that adaptive flooding broadcast strategy outperforms the ordinary flooding broadcast method.

Key words Ad hoc networks; broadcast; energy adaptive; restrain flooding

广播策略是Ad hoc网络中最基本的信息传输方式之一^[1-2],泛洪是最直观和常用的一种广播算法。其基本思想是源节点将消息以分组的形式发给其相邻的节点,相邻的节点再转发给它们的相邻节点,继续下去,直至分组到达网络中所有的节点。但是在Ad hoc网络中,简单的泛洪广播会导致所谓的“广播风暴”问题^[3]。大量的冗余转播不仅浪费了大部分带宽,还消耗了大量节点能源,同时产生的冲突和碰撞严重影响了Ad hoc网络的性能。

目前,一些新的广播算法在文献[3-7]中被陆续提出,但这些算法多数依赖于分群或者基于邻居信息,这通常要求节点间交换“Hello”消息或其他控制消息,会增加一定的通信负荷。

本文以减少信息的不必要重传为出发点,通过预测网络节点的剩余能量,提出了能量自适应转发的泛洪广播策略(EAB)。该策略计算优先指数(PI)及相关的转播延迟来禁止一些节点转播操作,与现有的普通泛洪广播策略^[8-10](ordinary flooding broadcast method, OBM)相比,EAB算法节省了转发节点的个

数,从而减少了不必要的转发。当网络规模越大时,EAB算法的优越性越明显。

1 能量自适应泛洪广播算法

自适应算法不需要任何周期性的控制信息,甚至不需维护邻居信息。该算法引入竞争机制,通过节点间的竞争转播^[4],能够使某些满足特定条件的节点优先转播广播消息以控制转播广播消息的节点数量。本文假定所有节点共享同一无线信道,MAC协议采用IEEE802.11。本文中的节点均指自组网中的移动节点,网络中可以相互直接通信的节点互称邻居。

当节点*i*接收到一个广播消息inf后,随即进入广播程序,按如下步骤竞争转播inf:

1) 节点*i*检查重复消息表,以确定 inf 是否是新的广播消息,如果是新的,转至步骤 2); 否则转至步骤 3)。

2) 新消息: 将 inf 的信息记入重复消息表,此时重复次数 $dn = 1$ 。节点*i*根据其当前状态和有关信

息计算一个优先指数 PI, 将 inf 存入消息缓存区; 根据 PI 计算竞争时间长度 CT, 节点 i 等待 CT 结束以决定是否转播 inf。转至步骤 4)。

3) 重复消息: 将 inf 删除, 并将重复消息表中该消息的 dn 增 1。此时并不删除重复消息表中与该 inf 相关的记录。转至步骤 4)。

4) 监听信道, 如果接收到一个消息, 转至步骤 1); 如果时间事件被触发, 转至步骤 5)。

5) 当 inf 的时间事件被触发, 节点检查重复消息表。

(1) 如果 $dn=1$, 说明在等待 CT 结束时尚未听到相同的 inf, 则节点 i 认为赢得了该次竞争, 节点从消息缓存中取出 inf 并传递到 MAC 层进行广播, 并退出竞争程序。

(2) 如果 $dn>1$, 说明在等待 CT 期间, 节点 i 又听到一个重复的 inf, 该节点“认为”其他具有更高优先权的邻居节点已经赢得了该次竞争并转播了该 inf, 该节点在竞争中失败, 从缓存中删除 inf, 并退出竞争程序。

本文采用文献[7]中提出的能量消耗模型, 设节点 i 的电池能量最大值是 $E_{i\max}$, E_i 表示当前节点剩余能量, $E_{i\text{consume}}$ 表示节点 i 已经消耗的总能量, 因此, 可以得到:

$$E_i = E_{i\max} - E_{i\text{consume}}$$

则节点的剩余能量归一化值为:

$$G(i) = (E_{i\max} - E_{i\text{consume}}) / E_{i\max}$$

定义 PI 为:

$$PI = \frac{E_i}{E_{i\max}}$$

节点的能量消耗主要可以分为 3 种状态, 分别是发送、接收和转发。设节点发送一个数据分组所需能量为:

$$E_{se} = P_s T_p = I_s v T_p \quad (2)$$

接收一个数据分组所需要的能量为:

$$E_{re} = P_r T_p = I_r v T_p \quad (3)$$

转发一个数据分组所需能量为:

$$E_{ifw} = (P_s + P_r) T_p \quad (4)$$

可以得到节点 i 已经消耗的总能量为:

$$E_{i\text{consume}} = E_{se} + E_{re} + (N-1)E_{ifw} \quad (5)$$

式中 P_s 为发射功率; P_r 为接收功率; T_p 表示发送和接收一个分组所需时间, 在传输速率为 1 Mb/s 的情况下 $T_p = P_{\text{size}} \times 10^{-6}$, P_{size} 为分组长度; I_s 为发送电流; I_r 为接收电流。根据文献[5], 在 Ad hoc 网络环境下, 经过多次试验所测的对于不同传输速率的无线

网卡在相同状态下的能量消耗值近似。因此, 取 $v=4.74 \sim 5.00$ V, 在发射状态下 $I_s=280$ mA, 接收状态下 $I_r=204$ mA。

对于能量自适应转发的泛洪广播算法, 竞争时间函数 CT 是节点在转播消息前需要等待的时间长度, 将 CT 定义为 PI 的函数, 同时要满足: 当 PI 接近最大值 PI_{\max} 时, CT 加速下降。这样, 可以拉开具有较大 PI 也就是具有较高的转播概率的节点间的竞争时间长度, 以降低信号碰撞概率。根据 CT 要满足的条件, 利用双曲正切函数 $\tanh(\cdot)$ 得到函数 $\tanh\left(\frac{1-x}{n}\right)$, 如图 1 所示。该函数的值域 $\in [0,1]$, 是一个有界正函数, 其中 n 是一个常数。根据图 1 可以看到, 当 $n=0.3$ 或 $n=0.4$ 时, 函数能够更好地满足 CT 的要求, 本文取 $n=0.4$ 。

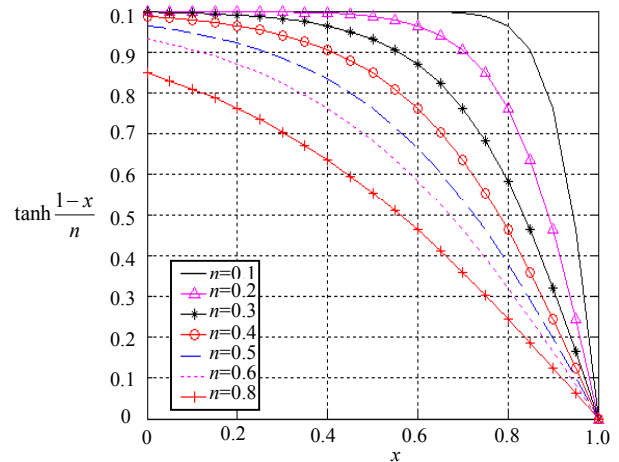


图 1 双曲正切函数 $\tanh((1-x)/n)$

令 $x = \frac{PI}{PI_{\max}}$, 则 CT 定义为:

$$CT = f(PI) = D \cdot \tanh\left(\left(1 - \frac{PI}{PI_{\max}}\right) / n\right) \quad (6)$$

式中 D 为设定的最长等待时间。

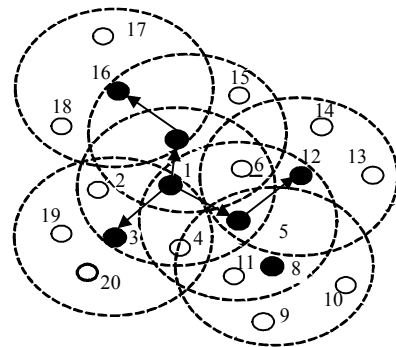


图 2 20 个节点的广播实例

图 2 中给出了一个广播实例, 图中共有 20 个节点。根据 EAB 算法, 当源节点(图中节点 1)需要广播

消息inf时,其邻居节点(图中节点2~7)可以同时接收到该广播消息,即进入竞争广播。节点2~7根据各自的剩余能量分别计算出自适应竞争时间,确定自己是否进行转发广播消息。由于节点3的 E_i 最大,其PI最大,则CT最小,将首先赢得转播权。节点3转播后,节点2和节点4将听到节点3的转播,因此节点2和节点4失去转播权。节点5比节点6有较大的 E_i ,又由于其听不到节点3的转播,在其CT结束时也将转播该消息。类似地,节点7在节点3转播后将获得转播权。

2 性能分析

为了进一步验证EAB算法的有效性,建立仿真环境进行验证分析,实验中的网络拓扑随机生成,节点总数为100,节点的传输半径设为300 m,MAC层选用IEEE802.11协议DCF模式,链路带宽为2 Mb/s,仿真通信采用大小为512 B的定长数据包,发包速率为300 kb/s,路径的端节点随机选出。设节点的电池能量最大值是 $E_{\max}=100$ J,但每个节点的初始能量不同,随机生成。仿真次数为20,取其平均作为仿真结果。

评价的性能参数^[3]为:

(1) 转播节省率(saved rebroadcast, SRB)。

$SRB = \frac{n_r - n_s}{n_r}$, 其中, n_r 是接收到广播消息的主机数;

n_s 是实际发送广播消息的主机数。EAB算法在不同节点数下的转播节省率如图3所示。

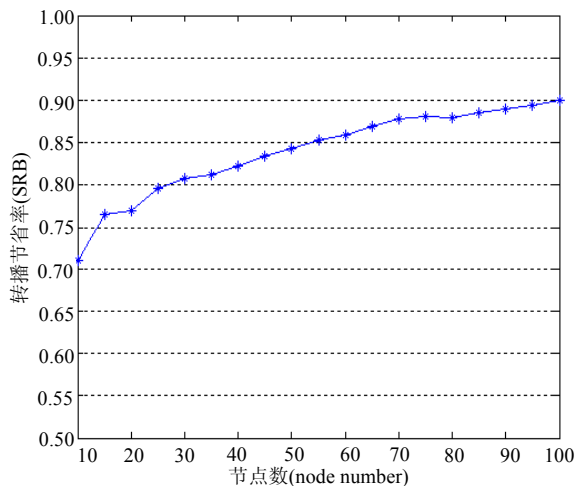


图3 EAB算法在不同节点数下的转播节省率

(2) 到达率(reach ability, RE)。 $RE = \frac{n_r}{n_m - 1}$,

其中, n_m 是网络中的主机总数。两种算法在不同节点数下的到达率比较如图4所示。

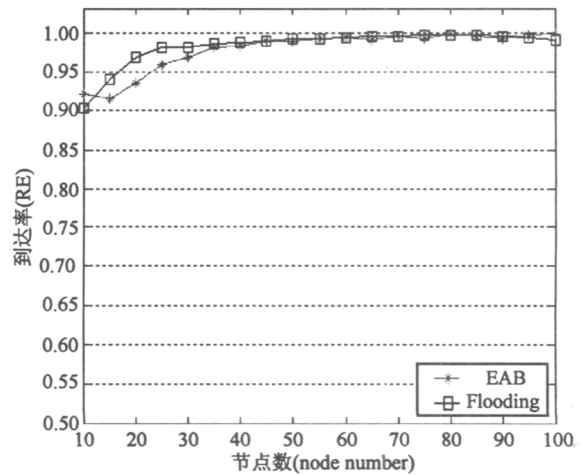


图4 两种算法在不同节点数下的到达率比较

(3) 平均广播延迟 d 是分组从广播源节点传输到每个接收节点的延迟的平均值:

$$d = \sum_{i \in R} \frac{d_i}{n_r} \quad (7)$$

式中 R 是接收到广播分组的节点集; d_i 是广播源节点到节点 i 的广播延迟。两种算法在不同节点数下的平均时延如图5所示。

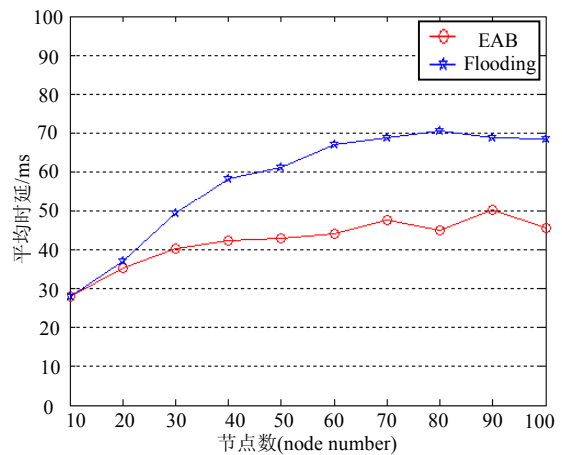


图5 两种算法在不同节点数下的平均时延

从图3中可以看出,在不同节点数下,EAB算法与泛洪算法比较转播节省率SRB都在70%以上,大部分在75%~87%之间,抑制了泛洪算法从源节点以波浪形式向外传播,减少了泛洪的盲目性,可见EAB算法具有较高的效率。

如图4所示,在所有情况下能量自适应广播算法和泛洪算法都具有较高的可达率RE,而且当节点个数越多EAB算法表现出的性能越好。与泛洪方案相比,EAB算法不仅去除了大量冗余转播,还改善了广播性能。

从图5中可以看到,平均时延随着网络规模的逐渐增大而增大,同时EPB算法的平均时延要比泛洪

算法小。由于EPB算法减少了广播消息的碰撞,降低冲突概率,使广播消息在网络中传播的更快,因此改善了广播时延的性能。

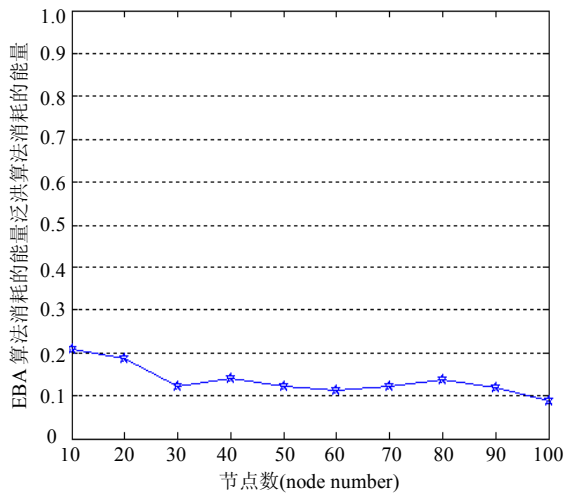


图6 两种算法在不同节点数下的能量消耗之比

图6是EAB算法与泛洪算法在能量消耗上的比较,由于在EAB算法中转发广播消息的节点是有选择性的,一些节点不进行转发,节省了大量的能量,减少了广播的开销。从图中可以看出,EAB算法在广播过程中节约了能量,同时,通过平衡各个节点的能量消耗可以延长网络的生命周期。

3 结 论

本文从网络中能量优先的角度出发,提出了一种能量自适应的广播算法,该算法不需要增加额外的通信负荷,同时能有效地使用节点能量,延长节点生命周期,因而具有更好的动态适应性,能够使某些满足特定条件的节点优先转播广播消息。通过仿真表明,该算法去除了大量的冗余转播,同时可

降低节点间的竞争和碰撞,有效地抑制了泛洪。

参 考 文 献

- [1] MACKER J P, CORSON M S. Mobile Ad hoc networking and the IETF[J]. *Mobile Computing and Communication Review*, 1998, 2(2): 9-15.
- [2] BERTSEKAS D, GALLAGER R. *Data network*[M]. USA: Prentice-Hall, 1992, 368-370.
- [3] NI S Y, TSENG Y C, CHEN Yuh-shyan, et al. The broadcast storm problem mobile Ad hoc network[C]//*Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing Networks*. Washington, USA: IEEE, 1999: 151-162.
- [4] ZHOU Bo-sheng, WU Jie-yi, FEI Xiang, et al. PCBA: A priority-based competitive broadcasting algorithm in mobile Ad hoc networks[J]. *Computer Sci and Technol*, 2003, 18(5): 598-606.
- [5] LAUER G. Address servers in hierarchical networks[C]//*Proc, ICC. Atlanta, GA, USA: IEEE*, 1988: 443-451.
- [6] WU J, LI H. On calculating connected dominating set for efficient routing in Ad hoc wireless networks[C]//*Proc DIAL M*. New York, NY, USA: ACM, 1999: 7-14.
- [7] WANG Yu, GARCIA-LUNA-ACEVES J J. Collision avoidance in multi-hop Ad hoc networks[C]//*10th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS'02)*. TX, USA: IEEE, 2002.
- [8] STOJMENOVIC I, SEDDIGH M, ZUNIC J. Internal node based broadcasting algorithms in wireless networks[C]//*Proceedings of the 34th Annual Hawaii International Conference on System Sciences*. Maui, Hawaii: [s.n.], 2001
- [9] PAGANI E, ROSSI G P. Providing reliable and fault tolerant broadcast delivery in mobile ad-hoc networks[J]. *Mobile Networks and Applications*, 1999, 5(4): 175-192.
- [10] MAHESH K M, DAS S R. On-demand multipath distance vector routing in Ad hoc network[C]//*Ninth International Conference on Network Protocols*. Washington D. C., USA: [s. n.], 2001: 14-23.

编辑 漆 蓉