

自组网中基于自适应波束天线的拓扑控制算法

王东¹ 陈文斌¹ 李晓鸿¹ 胡婷¹ 张大方^{1,2}

¹(湖南大学计算机与通信学院 长沙 410082)

²(湖南大学软件学院 长沙 410082)

(wangd@hnu.cn)

Distributed Topology Control Algorithm for Ad Hoc Networks Using Steered Beam Directional Antennas

Wang Dong¹, Chen Wenbin¹, Li Xiaohong¹, Hu Ting¹, and Zhang Dafang^{1,2}

¹(College of Computer and Communication, Hunan University, Changsha 410082)

²(College of Software, Hunan University, Changsha 410082)

Abstract The topology of Ad Hoc networks using directional antennas is more complex than that with traditional omni-directional antennas. Based on a given steered beam directional antennas and using a method to obtain local neighbor location information, the authors propose a distributed topology control algorithm—SDTC (the steered beam directional antenna based topology control algorithm). The topology is controlled not only by adjusting the transmission powers of nodes but also by changing the direction, beam width and gain of directional antenna. Each node in the networks needs to take the responsibility of collecting information of its neighbor nodes, choosing the optimum adjacent nodes by using the strategy of power control scheduling, and then selecting the minimum transmission power as its transmission power, which can cover all of the optimum adjacent nodes it has chosen. The algorithm preserves the connectivity of the resulting topology, which can be transformed into the one equipped with bi-directional links as well. At the same time, the resulting network topology can reduce the energy consumption, decrease the traffic interference, and improve the network throughput as the results of less node transmission power and lower node degrees. Simulation results show that the proposed algorithm significantly improves the network performance. And the steered beam directional antenna based topology control algorithm for mobile Ad Hoc network remains to be studied in the future.

Key words Ad Hoc networks; topology control; directional antennas; network simulation; throughput

摘要 定向天线自组网拓扑的构建问题比全向天线网络复杂。基于自适应波束定向天线模型提出一种分布式拓扑控制算法,通过调整节点发射功率,改变天线波束的朝向、宽度和增益来构建拓扑。网络中每个节点收集其邻居节点信息,采用功率控制调度策略选择最优相邻节点,并选取覆盖所有最优相邻节点的最小发射功率为此节点的发射功率。算法在保证网络连通性与无向性的同时,降低了节点的发射功率,减小了节点的平均度数,从而降低节点能耗,减少了节点间干扰,提高了网络吞吐量。仿真结果表明,算法显著提高了网络性能。

收稿日期:2008-06-12;修回日期:2009-06-18

基金项目:国家“九七三”重点基础研究发展计划基金项目(2007CB310702);国家自然科学基金项目(60703097);湖南省自然科学基金项目(06JJ4078)

关键词 自组网; 拓扑控制; 定向天线; 网络仿真; 吞吐量

中图法分类号 TP393.03

近年来, 由于定向天线能显著提高系统的性能和容量, 定向天线技术开始在无线通信中广泛应用. 定向天线信号只需发向某个特定的方向, 其本身就具有很好的节能特性. 文献[1-2]表明, 在相同场景下, 与使用全向天线相比, 节点使用定向天线后网络传输能力提高, 使得端到端延迟减小.

自组网的拓扑结构是基于节点物理位置和信号传输范围自治形成的. 与全向天线信号覆盖区域被模型化为一个圆形碟子相比, 定向天线模型使得网络拓扑构建问题更加复杂. 首先, 由于信号主要覆盖一定角度的区域, 这就有可能减少节点的邻居数, 从而影响网络连通性; 其次, 由于定向天线增益的方向性, 节点使用相同的发射功率, 其覆盖距离比全向天线更远, 这又使得节点在不增加能耗的条件下可能与更多的节点直接通信, 提高网络的连通性. 拓扑控制主要研究如何通过调整节点发射功率和建立合适相邻关系的方法构建自组网拓扑; 在维持拓扑连通性的前提下, 达到节约节点能量、延长网络生命周期、降低网络干扰、提高网络吞吐量的目的[3].

许多学者提出了一些有影响的拓扑控制算法[3-5], 但是这些拓扑控制算法均假设自组网中节点使用全向天线. 目前, 已有一些学者针对多波束转换天线的自组网进行了拓扑控制方法的研究[6-7]. 基本思想都是利用已有的全向天线拓扑控制算法求得每个节点的逻辑邻居, 根据逻辑邻居的方位, 利用多波束转换天线的特性打开合适的天线扇区, 实现网络连通. 文献[8]的研究表明自适应波束天线可以增加节点间多跳路径的连通概率, 但是没有深入讨论定向天线对整个网络拓扑图连通性的影响.

针对自适应波束定向天线, 采用获取局部邻居位置信息的方法, 提出一种基于自适应波束定向天线的分布式拓扑控制算法——SDTC 算法(steered beam directional antenna based topology control algorithm). 算法主要思想是, 每个节点收集其邻居节点信息, 采用功率控制调度策略选择最优相邻节点, 并选取覆盖所有最优相邻节点的最小发射功率为此节点的发射功率, 构建一个节点度很小的无向连通图. 仿真结果表明, SDTC 算法显著降低了节点的发射功率, 提高了网络的吞吐量.

1 自适应波束定向天线模型

随着数字信号处理技术和天线阵列技术的发展, 定向天线的成本和设备大小已经可以应用于体积较小的无线设备[9]. 按照波束形成方法, 定向天线分为多波束转换天线和自适应波束天线 2 大类[10]. 为了便于讨论, 本文采用如图 1 所示的自适应波束天线模型.

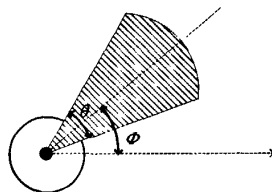


Fig. 1 Steered beam directional antennas model.

图 1 自适应波束天线模型

其中 Φ 表示波束的朝向, 节点可以控制波束的朝向; θ 表示波束的宽度, 根据天线阵列技术, θ 可调; 阴影部分为主瓣, 其余区域为旁瓣, 各自区域内的增益为定值, 且主瓣不小于旁瓣的增益(当主瓣增益和旁瓣增益相等时, 可认为是全向天线). 本文假定所有节点初始的最大发射功率相同, 节点采用自适应波束天线进行通信, 能够定向侦听、接收信号.

假设节点均匀分布在一个二维平面中, 每个节点具有唯一的标识 id 号. 任意节点 u 都配有一个自适应波束定向天线, 其发射功率 $p(u)$ 可以在最小和最大之间连续调节, 即 $0 \leq p(u) \leq p(u)_{\max}$, $p(u)_{\max}$ 表示节点 u 的最大发射功率; 其旁瓣增益 G_s 为 1(0 db), 主瓣增益 G_m 不小于旁瓣增益. 节点具有相同的最大发射功率 P_{\max} , 相同的接收功率阈值 P_{th} , 相同的旁瓣增益 $G_s = 1$, 且配备 GPS 接收器. 采用自由空间路径损耗模型[11], 发射功率 P_t 与接收功率 P_r 的关系为:

$$P_r = \frac{P_t G_t G_r \lambda^2}{(4\pi d)^2 L}, \quad (1)$$

其中, G_t, G_r 分别是发射和接收天线的增益, λ 为波长, d 是发射天线与接收天线之间的距离, L 为系统损耗因子($L \geq 1$).

节点通过侦听信道中的能量强度来判断是否有

信号在信道中传输.当侦听到的能量强度高于节点的接收阈值 P_{th} 时,则表明信道中有信号传递,接收节点能够正确接收信道内的数据.由此可得出节点所需的最小发射功率 P_{t-min} 为:

$$P_{t-min} = \frac{P_{th} (4\pi)^2 d^2 L}{G_t G_r \lambda^2}. \quad (2)$$

令 $c = (4\pi)^2 \cdot L \cdot P_{th} / \lambda^2$ (c 为常量),由式(2),可得

$$P_{t-min} = c \cdot d^2 / (G_t \cdot G_r). \quad (3)$$

假设节点采用自适应波束天线模型后,有 $G_s \leq G_t, G_r \leq G_m$, 因为 $G_m = kG_s$ ($k \geq 1$), 由式(3),可得

$$c \cdot d^2 / G_s^2 \geq P_{t-min} \geq c \cdot d^2 / (k^2 \cdot G_s^2). \quad (4)$$

假设节点采用全向天线收发数据时,有 $G_t, G_r = G_s$. 由式(3),可得

$$P_{t-min} = c \cdot d^2 / G_s^2. \quad (5)$$

由式(3)可得,节点的最小发射功率与距离的平方成正比,与发射增益、接收增益成反比.由于自适应波束天线的主瓣增益不小于旁瓣增益,且全向天线的增益等于旁瓣增益,可得式(4)和(5).由式(4)和(5)可知,与采用全向天线相比,节点采用自适应波束天线通信时,当距离一定时,节点发射功率的下限更小.因此,采用自适应波束天线进行通信使得节点的发射功率得以优化.

为了方便讨论基于自适应波束定向天线的自组网模型,下面给出相关概念和术语:

1) $P(u, v)$ 表示节点 u 将数据传输到 v 所需的最小功率.

2) $G(u, v)$ 表示节点 u 和 v 间的增益.

3) $u \rightarrow v$ 表示从节点 u 到节点 v 的一条有向边, $u \vdash v$ 表示节点 u, v 间无链接, $u \leftrightarrow v$ 表示节点 u, v 有双向边.

4) 图 G_0 表示所有节点都使用最大发射功率 P_{max} , 整个网络形成的拓扑结构. $V(G_0)$ 为 G_0 中所有节点的集合, $E(G_0)$ 为 G_0 中所有有向边的集合.对于任意一条有向边 $(u, v) \in E(G_0)$, 当且仅当 $u \neq v$ 且 $P(u, v) \leq P_{max}$. G_0 表示未进行拓扑控制时的网络拓扑结构.

5) $N(u)$ 表示未进行拓扑控制时节点 u 的一跳邻节点集,即 u 在图 G_0 中的邻节点.

6) 节点 u 的邻节点个数称为节点 u 的度数,记作 $deg(u)$.

7) G 表示拓扑控制后的拓扑图.对于任意一条有向边 $(u, v) \in E(G)$, 当且仅当 $u \neq v$ 且 $P(u, v) \leq P(u)$, 其中 $P(u)$ 表示经过拓扑控制后节点 u 选择

的发射功率, $0 \leq P(u) \leq P_{max}$. 显然, G 为 G_0 的生成子图.

8) $Neigh(u)$ 表示节点 u 在图 G 中的一跳邻节点集,即进行拓扑控制后节点 u 的一跳邻节点. $\forall v \in Neigh(u)$ 有 $P(u, v) \leq P(u)$, 显然, $Neigh(u) \subseteq N(u)$.

2 基于自适应波束定向天线的拓扑控制算法

一个有效的拓扑控制算法应满足下列性质^[12]:

1) 算法在设置节点最小发射功率的同时必须保证网络连通; 2) 算法能分布式运行; 3) 算法只需要每个节点收集自己局部的信息, 从而减少信息的交互开销, 提高算法的收敛速度; 4) 算法生成的拓扑图是对称、边稀疏和节点度有界的.

基于以上性质, 本文提出了一种基于自适应波束天线的分布式拓扑控制算法(SDTC). 算法主要由3阶段组成: 信息收集、拓扑构建和功率设置阶段. 在信息收集阶段, 为了确定发射功率, 节点 u 首先需要收集一跳邻节点的信息, 确定 $N(u)$. 节点 u 以功率 P_{max} 向外发送“Hello”消息, “Hello”消息中包含了节点 u 的 id 号. 节点 v 收到 u 发送的“Hello”消息后(当且仅当节点 v 的接收功率 P_r 大于接收功率阈值 P_{th}), 将向节点 u 回应“ACK”消息, “ACK”消息中包含了节点 v 的 id 号、位置信息以及自适应波束天线的朝向. 当收到节点 v 的响应消息后, 节点 u 把节点 v 加入到集合 $N(u)$ 中. 网络中所有节点收集完毕它们的邻节点信息就可以构造出拓扑控制之后的网络拓扑图 G .

2.1 SDTC 算法描述

算法描述如下:

1) 节点 u 设置其初始时发射功率 $P(u) = P_{max}$, 设置天线随机选择一个朝向 Φ .

2) 节点 u 以 P_{max} 发送“Hello”消息, 并收集来自其他节点的“ACK”, 如果节点 u 发现了一个新节点 v , 则节点 u 将 v 存入其邻节点集 $N(u)$ 中.

3) 节点 u 根据位置信息和天线朝向计算出其与 $N(u)$ 中所有节点间所需的最小发射功率, 采用功率控制调度策略选取最优相邻节点, 将其存入节点 u 的邻节点集 $Neigh(u)$ 中.

4) 节点 u 将其与 $Neigh(u)$ 内所有节点所需最小发射功率的最大值设置为其发射功率, 算法结束.

SDTC 算法伪代码如下:

① for each node $u \in V(G_0)$ do

- ② $current_pow = P_{\max}$;
- ③ $N(u) = \Phi$;
- ④ $Broadcast(u)$;
- ⑤ $GetACK(u, v, pos, \Phi)$;
- ⑥ Calculate $G(i, j)$ from each node i to all other node j using $N(u), pos(u)$ and its Φ while $i, j \in u \cup N(u)$;
- ⑦ Initialize set $Neigh(u)$ equals NULL;
- ⑧ Initialize set M equals NULL and add u to M ;
- ⑨ while $M \neq N(u) \cup u$ do
- ⑩ next node $v = findnext()$;
- ⑪ Add v to M ;
- ⑫ $P(u) = \max\{\forall_{v \in Neigh(u)} P(u, v)\}$.

/* 功率控制调度策略 */

Procedure $findnext()$

- ① for each node $m \in M$ do
- ② for each node $n \in N(u) \cup u - M$ do
- ③ $power-of-m-when-add-n = min-Power(m, n)$;
- ④ $n = \min\{\forall_{u,v} power-of-m-when-add-n\}$;
- ⑤ if (m equals u)
- ⑥ Add n to $Neigh(u)$;
- ⑦ return n .

/* 计算最小发射功率 */

Procedure $min-Power(u, v)$

- ① $upower = \frac{(4\pi d(u, v))^2}{P_{th} G(u, v) \lambda^2}$;
- ② return $upower$.

算法使得每个节点收集其邻居节点信息,并以最小发送功率为权值构建最小生成树 T_u ,选取其一跳邻居最小发送功率的最大值为发送功率.算法伪代码的第①行保证此算法为分布式算法,第④⑤行说明每个节点收集自己局部的信息,功率控制调度策略则得到节点较小的发射功率和较低的节点度.

2.2 SDTC 算法特性

2.2.1 网络拓扑的连通性

定义 1. 对于给定 2 条边 (u_1, v_1) 和 (u_2, v_2) ,权值函数 P' 有如下定义:

$$P'(u_1, v_1) > P'(u_2, v_2) \Leftrightarrow$$

$$P(u_1, v_1) > P(u_2, v_2) \text{ 或 } \Leftrightarrow$$

$$P(u_1, v_1) = P(u_2, v_2) \ \&\& \ \max\{id(u_1), id(v_1)\} > \max\{id(u_2), id(v_2)\} \text{ 或 } \Leftrightarrow$$

$$P(u_1, v_1) = P(u_2, v_2) \ \&\& \ \max\{id(u_1),$$

$$id(v_1)\} = \max\{id(u_2), id(v_2)\} \ \&\& \ \min\{id(u_1), id(v_1)\} > \min\{id(u_2), id(v_2)\}.$$

定义 1 确保任意节点 u 以最小发送功率为权值构建最小生成树 T_u 的唯一性.

当初始拓扑图 G_0 连通时,本文证明由 SDTC 算法构建的生成子图 G 也连通.先给出如下定义:

定义 2. 对于 $\forall u, v (u, v \in V(G))$,若 u, v 存在一条路径 $(w_0 = u, w_1, \dots, w_{m-1}, w_m = v) (w_j \leftrightarrow w_{j+1}, (w_j, w_{j+1}) \in V(G) \text{ 且 } (w_{j+1}, w_j) \in V(G), j = 0, 1, \dots, m-1, w_k \in V(G), k = 0, 1, \dots, m)$,记作 $u \Leftrightarrow v$.

定理 1. 若初始拓扑图 G_0 连通,则 G 连通.

证明. 把图 G 中所有节点对 $[u, v]$,分为 $P(u, v) \leq P_{\max}$ 和 $P(u, v) > P_{\max}$ 2 种情况.

1) 将图 G 中所有满足 $P(u, v) \leq P_{\max}$ 的节点对 $[u, v]$,按其权值升序排列: $P'(u_1, v_1) < P'(u_2, v_2) < \dots < P'(u_m, v_m) < \dots < P'(u_l, v_l)$.

下面用数学归纳法证明 $u \Leftrightarrow v$:

① 当 $m = 1$ 时,即对于权值最小节点对 $[u_1, v_1]$,SDTC 算法保证 $u_1 \leftrightarrow v_1$,所以 $u_1 \Leftrightarrow v_1$.

② 假设对于所有 $m < k (2 \leq k \leq l)$ 的节点对都有 $u \Leftrightarrow v$,那么当 $m = k$ 时,因为 $P(u_k, v_k) \leq P_{\max}$,所以当运行 SDTC 算法后,有以下 2 种情况:

(i) 若 $u_k \leftrightarrow v_k$,显然 $u_k \Leftrightarrow v_k$.

(ii) 若 $u_k \mapsto v_k$ 或 $v_k \mapsto u_k$ 或 $u_k \mapsto v_k$ 且 $v_k \mapsto u_k$.不妨设 $u_k \mapsto v_k$.当 u_k 运行功率控制调度策略后,必能在 G_0 中找到一条从节点 u_k 到节点 v_k 的路径: $path = (w_0 = u_k, w_1, \dots, w_{p-1}, w_p = v_k)$.又由节点 u_k 构建最小生成树的唯一性,可知, $P'(w_i, w_{i+1}) < P'(u_k, v_k)$,其中 $i = 0, 1, \dots, k-1$.所以,由归纳假设可得 $w_i \Leftrightarrow w_{i+1}$,即对于 $path$ 中的任意相邻节点都有路径可达,所以 $u_k \Leftrightarrow v_k$.

由①②得,对于图 G 中节点对 $[u, v]$,若 $P(u, v) \leq P_{\max}$,则 $u \Leftrightarrow v$.

2) 对于图 G 中 $P(u, v) > P_{\max}$ 的节点对 $[u, v]$,因为 G_0 连通,所以在 G_0 中,存在路径 $path = (l_0 = u, l_1, \dots, l_{p-1}, l_p = v)$.且 $P'(l_i, l_{i+1}) \leq P_{\max}$,其中 $i = 0, 1, \dots, p-1$.由 1) 的证明知 $l_i \Leftrightarrow l_{i+1}$.所以,对于 $P(u, v) > P_{\max}$ 的节点,对 $[u, v]$ 同样有 $u \Leftrightarrow v$.

由 1)2) 可得,当 G_0 连通,对于 $\forall u, v \in G, u \Leftrightarrow v$,即 G 连通. 证毕.

2.2.2 网络拓扑的无向性

记 G^+ 为将 G 中的所有有向边加上反向边所形成的图; G^- 为去掉 G 中所有单向边形成的图.此时, G^+, G^- 均相当于无向图.

定理 2. G 连通 $\Rightarrow G^+, G^-$ 连通.

证明. 1) 因为 G 是 G^+ 的子图, 所以, 若 G 连通, 则 G^+ 一定连通.

2) 由定理 1 的证明过程可知, 对于所有的单向边关联的节点, 都存在双向可达的路径, 故去掉单向边不影响图的连通性, 即 G^- 也是连通的. 证毕.

因此, 初始拓扑图 G_0 连通时, 由 SDTC 算法生成的子图能够保证网络的连通性与无向性.

3 性能评估

为了验证 SDTC 算法的有效性, 本文仿真研究了其导出拓扑图的性质, 在此基础上, 采用 DSR 路由协议, 分析了不同网络负载、不同的自适应波束天线波束宽度和增益下生成拓扑结构的网络吞吐量、平均端到端延时和平均路由跳数.

3.1 拓扑结构的性质

拓扑结构的好坏直接反映了拓扑控制算法性能的优劣, 而评价拓扑性能好坏的一个很重要的衡量

标准就是节点度 $deg(u)$, 本文从节点平均度、节点平均发射功率 2 个与节点度相关的指标进行比较分析.

假设节点随机分布在 $1000\text{ m} \times 1000\text{ m}$ 的二维区域中, 节点数为 $50 \sim 250$ 个. 仿真中, 可设置不同的自适应波束天线的波束宽度和主瓣增益.

图 2 示例了网络节点数为 160、不同算法所形成的拓扑结构. 图 2(a) 是未经过拓扑控制的网络拓扑. 图 2(b)~2(d) 是经过拓扑控制后的网络拓扑图, 从图 2(b)~2(d) 中可以很明显地发现, 拓扑控制后所导出的网络拓扑结构简单、节点平均度小、节点功率的有效性高. 图 3(a) 显示了不同算法所导出的拓扑的节点平均度. 节点平均度小能增强信道空间复用度, 提高网络容量, 降低分组碰撞概率. 从图 3(a) 中可以看出, 当未采用拓扑控制时, 随着节点密度的增加, 节点度直线增加, 而 SDTC 算法所形成的网络拓扑的节点平均度随着网络节点数的增加几乎保持不变, 节点平均度基本控制在 5 以内.

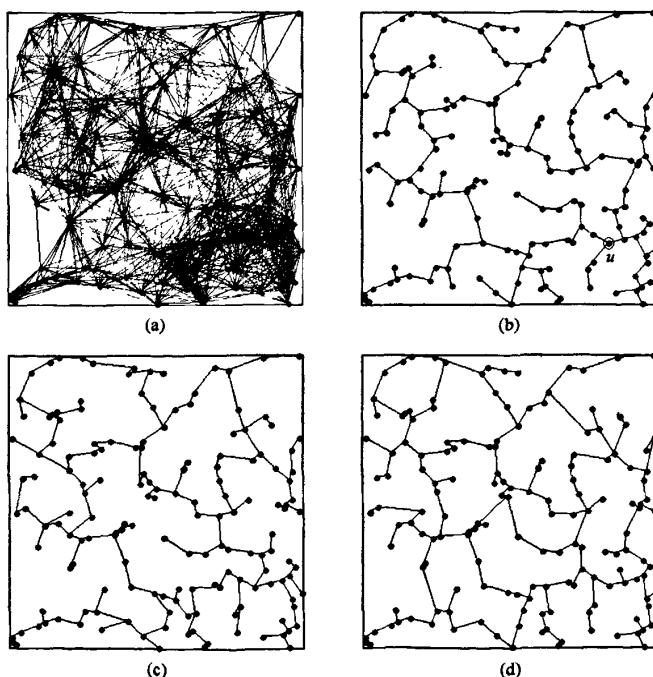


Fig. 2 The topologies by different algorithms. (a) UDG (All nodes using omni-directional antennas); (b) SDTC_1 (The gain of main lobe is 1 using directional antennas); (c) SDTC_30_2 (The beamwidth is 30° and the gain of main lobe is 2 using directional antennas); and (d) SDTC_90_2 (The beamwidth is 90° and the gain of main lobe is 2 using directional antennas).

图 2 不同拓扑控制算法生成的拓扑图. (a) UDG (节点都采用全向天线); (b) SDTC_1 (定向天线主瓣增益为 1); (c) SDTC_30_2 (定向天线的波束宽度为 30° , 主瓣增益为 2); (d) SDTC_90_2 (定向天线的波束宽度为 90° , 主瓣增益为 2)

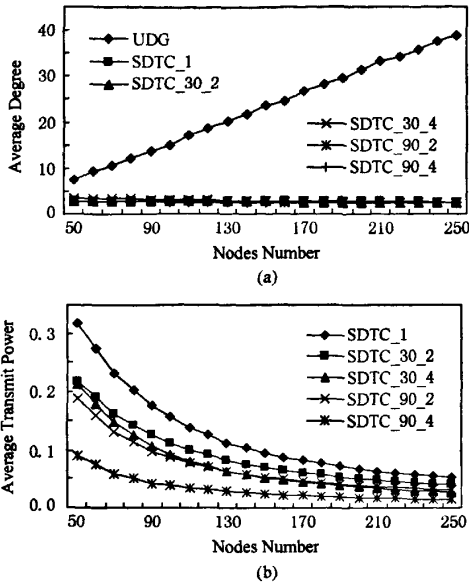


Fig. 3 The average performance of various structures. (a) Average degree of nodes and (b) Average transmission power of nodes (transmission power of UDG is 1).

图3 拓扑控制算法生成拓扑图的性能比较。(a)节点平均度;(b)节点平均发射功率(令UDG的发射功率为1)

在自组网中,能耗是人们最为关心的问题之一。一个算法的优劣在很大程度上取决于其能量节省的能力,以保证能最大限度地延长网络的生命周期。本文评估了不同拓扑控制算法下网络中节点平均发射功率,如图3(b)所示。从图3(b)可以看出,经过拓扑控制算法后确定的节点发射功率(小于最大功率的35%)要远远优于未经拓扑控制的节点发射功率。UDG未采用拓扑控制,所有节点均以最大发射功率进行通信。SDTC采用了拓扑控制机制,随着节点密度的增加,节点发射功率呈现下降的趋势,且当自适应波束天线的波束宽度和主瓣增益越大时,SDTC算法所确定的节点平均发射功率更小。因此,SDTC算法在维护网络连通性的同时保证了节点最小能量特性,在提高网络能量有效性方面效果显著。

3.2 网络性能仿真

为了分析不同拓扑结构网络的性能,本文采用OPNET10.5网络仿真工具,仿真场景如下:取 n 个节点随机分布在 $1000\text{ m}\times 1000\text{ m}$ 的二维区域中,选择20个源节点,每个节点每次将1 KB的数据发送给随机选择的的节点,发送间隔时间呈Poisson分布,网络层使用DSR路由协议,传播损耗模型使用自由空间模型,节点在链路层采用基于802.11标

准的MAC协议。每个节点采用自适应波束天线,天线高度为1.5 m,接收阈值为 -80 dBm ,节点最大发射功率为 0.0064 W (此时每个节点旁瓣传输半径为250 m)。仿真时间为500 s。当节点变化时,仿真结果类似,本文取 $n=160$ 。仿真结果如图4~6所示:

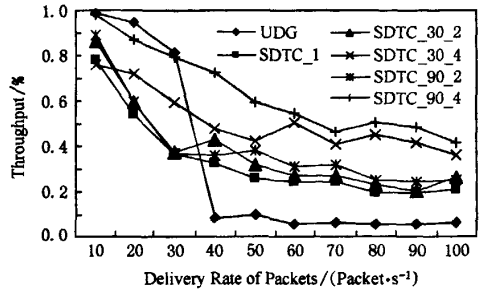


Fig. 4 Throughput of various algorithms.

图4 不同算法吞吐量的比较

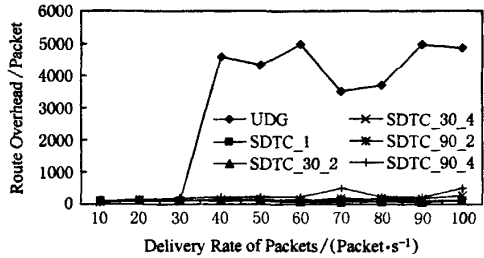


Fig. 5 Overhead of networks.

图5 整个网络的路由开销

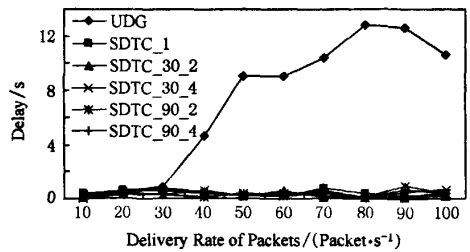


Fig. 6 Delay of networks.

图6 网络延时的比较

图4为不同拓扑控制算法下,网络吞吐量(接收数据量和总发送数据量的比值)的变化趋势。以UDG对应的曲线为例,网络的发包速率为 v (单位:Packet/s),当 $10\leq v\leq 30$ 时,吞吐量大于80%;但当 $30<v<40$ 时,吞吐量骤然下降,不到10%,并且随着发包速率增加,吞吐量逐渐降低。由图4不难看出,当 $40\leq v\leq 100$ 时,SDTC明显提高了网络吞吐量,但在 $10\leq v\leq 30$ 时,却不如未经过拓扑控制的

UDG 的吞吐量,且当自适应波束天线的主瓣增益和波束宽度增大时,SDTC 所对应的网络性能越好.这种变化趋势跟通信冲突概率密切相关.

由文献[13]可知,自组网中节点通信的冲突期望为:

$$E_c = \begin{cases} \frac{7}{32}\pi^2\rho T_x R^4, & r \leq \frac{R}{2}; \\ \pi\rho T_x \left(\frac{\sqrt{3}}{2} + \frac{17}{48}\pi\right) R^4, & \frac{R}{2} \leq r \leq R. \end{cases} \quad (6)$$

由等式(3)(6),得

$$E_c = \begin{cases} \frac{7}{32}\pi^2\rho T_x \left(\frac{P_{t,\min} G_t G_r}{c}\right)^2, & P_{t,\min} \leq \frac{P_{\max}}{4}; \\ \pi\rho T_x \left(\frac{\sqrt{3}}{2} + \frac{17}{48}\pi\right) \left(\frac{P_{t,\min} G_t G_r}{c}\right)^2, & \frac{P_{\max}}{4} \leq P_{t,\min} \leq P_{\max}. \end{cases} \quad (7)$$

其中 ρ 为节点的分布密度, T_x 为源节点发送数据的长度, r, R 与发射功率成正比.

由式(7)得,在节点的分布密度和发送数据长度相同时,节点通信冲突的期望与发射功率成正比.故在拓扑控制中,影响网络性能的主要因素为节点发射功率.当节点发射功率一定,即节点间冲突期望一定时,随着发包速率增大,节点间干扰概率增大,所有的拓扑对应的吞吐量呈下降趋势(见图 4),即拓扑控制只是延缓网络性能的下降.当网络发包速率不变时,节点平均发射功率(如表 1 所示)越大,节点间的干扰概率越大,链路失败的次数增多,DSR 路由发现频繁,路由开销增大(见图 5),导致所对应的网络性能越差.

Table 1 Average Power of Nodes and Routing Hop Numbers of Networks of Various Algorithms

表 1 不同算法中节点平均功率和网络平均路由跳数的比较

Topology Control Algorithm	Average Transmit Power/W	Average Number of Hops per Route
UDG	0.0064	3.15
SDTC_1	0.00053	7.23
SDTC_30_2	0.000469	8.92
SDTC_30_4	0.000456	7.98
SDTC_90_2	0.000346	8.34
SDTC_90_4	0.000151	10.58

然而在 $10 \leq v \leq 30$ 时,SDTC 算法却比 UDG 算法所对应的吞吐量小.这是由于在自组网中,影响网络性能的不只是节点间干扰这一因素,瓶颈节点也是影响网络性能的重要因素^[14].在 IEEE 802.11 协议中,节点发生冲突后将执行退避算法,随机退避一段时间.以图 2(b)中节点 u 为例,当网络的发包速率较低时,若瓶颈节点 u 受到周围节点的干扰,由于

退避机制通过节点 u 的路由会断开,DSR 重新启动路由发现机制.在图 2(a)中,图的连通度为 6,即任意节点间至少有 6 条不同的路径,有大量的路径可供源节点选择,重新找一条路即可,不影响节点间的通信.但在图 2(b)中,以牺牲连通度(连通度仅为 1)为代价降低节点的发射功率,导致瓶颈节点的分组碰撞概率增大,且可供源节点选择的路径很少,因此节点 u 失效即意味着经由 u 转发数据的源节点和目的节点间的通信结束.因此在网络发包速率较小时 SDTC 算法比 UDG 算法所对应的吞吐量小.

从图 4 中可以看出,UDG 出现吞吐量剧减的现象.当网络发包速率不大时($v \leq 30$),节点间干扰不大,路由开销主要受瓶颈节点影响;但当网络发包速率增大($40 \leq v \leq 100$),节点间的干扰起主导作用,网络的路由开销急剧上升(见图 5).还是以图 2(b)中的节点 u 为例.当网络的发包速率增大到一定值(此定值与节点分布,源节点、目的节点的选择,节点的发射功率和路由缓存的大小等因素有关.本文对应的环境下,此定值在 30~40 之间),节点 u 受到的干扰概率增大,路由缓存溢出(如图 7).由于 DSR 的

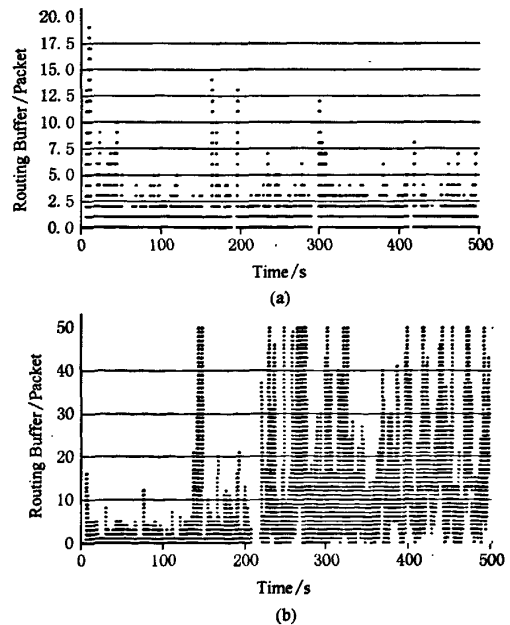


Fig. 7 Routing buffer of node u (the maximum size of routing buffer is 50). (a) The routing buffers of node u when the speed is 30 Packet/s; (b) The routing buffers of node u when the speed is 40 Packet/s.

图 7 节点 u 的路由缓存(网络中每个节点的路由缓存最大为 50). (a) 发包速率为 30 时 UDG 图所对应的节点 u 的路由缓存; (b) 发包速率为 40 时 UDG 图所对应的节点 u 的路由缓存

路由发现机制, 节点 u 告知源节点重新进行路由发现, 然而路由变更后, 新建路由上的转发节点依然受其他节点干扰, 从而导致路由发现过程反复进行. 最后, 由于“累积效应”, 使得整个网络的链路失败概率增加, 端到端延时增大 (见图 6), 路由开销剧增, 节点的有限缓存耗尽, 吞吐量下降.

综上所述, 当网络发包速率大于一定值时, 节点间的干扰是影响网络吞吐量的主要因素, 采用 SDTC 算法使得节点的发射功率变小, 从而达到减少网络节点间干扰, 增强空间复用能力, 显著提高网络吞吐量的目的. 但当网络每秒的发包个数小于此定值时, SDTC 算法导致网络拓扑中瓶颈节点数增多, 平均路由跳数增加 (见表 1), 链路失效概率增大, 反而降低了网络吞吐量.

4 结论和进一步的研究

本文针对自适应波束定向天线的特点, 通过调整网络中节点的发射功率, 改变天线的增益和波束宽度进行自组网拓扑控制. 由于每个节点使用了较低的发射功率以及该算法形成了具有较小的平均节点度的网络拓扑, 算法降低节点能耗, 优化网络性能. 仿真发现, 网络性能和网络负载、瓶颈节点、天线属性密切相关.

在实际应用中, 定向天线实现的复杂程度与天线增益、波束宽度有关. 因此, 下一步工作中, 将结合实际情况, 在网络负载、节点发射功率和定向天线的实现代价之间加以折中, 进一步优化网络性能; 节点移动和失效等异动情况也会改变网络的拓扑结构, 基于 SDTC 算法设计拓扑维护策略也将是我们下一步的工作.

参 考 文 献

- [1] Ramanathan R. On the performance of Ad Hoc networks using beamforming antennas [C] //Proc of the ACM MobiHoc. New York: ACM, 2001: 95-105
- [2] Yi S, Pei Y, Kalyanaraman S. On the capacity improvement of ad hoc wireless networks using directional antennas [C] //Proc of the 4th ACM Int Symp on Mobile Ad Hoc Networking and Computing. New York: ACM, 2003: 108-116
- [3] Santi P. Topology control in wireless ad hoc and sensor networks [J]. ACM Computer Surveys, 2005, 37(2): 164-194
- [4] Lu Gang, Zhou Mingtian, Niu Xinzhen, et al. A survey of proximity graphs in wireless networks [J]. Journal of Software, 2008, 19(4): 888-911 (in Chinese)
(路纲, 周明天, 牛新征, 等. 无线网络临近图综述 [J]. 软件学报, 2008, 19(4): 888-911)
- [5] Wu Zhendong, Li Shanping. A topology control algorithm based on link reliability and multi-path for sensor networks [J]. Journal of Computer Research and Development, 2007, 44(2): 216-222 (in Chinese)
(吴震东, 李善平. 传感器网络中一种基于链路稳定性和多路径的拓扑控制算法 [J]. 计算机研究与发展, 2007, 44(2): 216-222)
- [6] Huang Zhuochuan, Shen Chien-Chung, Srisathapornphat C, et al. Topology control for Ad Hoc networks with directional antennas [C] //Proc of the 11th Int Conf on Computer Communications and Networks. Miami: IEEE IC3N, 2002: 16-21
- [7] Kumar U, Gupta H, Das S R. A topology control approach to using directional antennas in wireless mesh networks [C] //Proc of the IEEE ICC. Los Alamitos, CA: IEEE Computer Society, 2006: 4083-4088
- [8] Bettstetter C, Hartmann C, Moser C. How does randomized beamforming improve the connectivity of Ad Hoc networks [C] //Proc of the IEEE Int Conf on Communications (ICC). Los Alamitos, CA: IEEE Computer Society, 2005: 3380-3385
- [9] Jakllari G, Luo W, Krishnamurthy S V. An integrated neighbor discovery and MAC protocol for Ad Hoc networks using directional antennas [C] //Proc of the IEEE WoW-MoM. Los Alamitos, CA: IEEE Computer Society, 2005: 11-21
- [10] Lehne P H, Pettersen M. An overview of smart antenna technology for mobile communication systems [J]. IEEE Communication Surveys, 1999, 2(4): 2-13
- [11] El-Hoiydi A, Decotignie J D. WiseMAC: An ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks [C] //Proc of the 9th Int Symp on Computers and Communications. Piscataway, NJ: IEEE, 2004: 244-251
- [12] Wang Yu, Li Xiangyang. Localized construction of bounded degree planar spanner for wireless networks [C] //Proc of the DIALM-POMC Joint Workshop on Foundations of Mobile Computing. New York: ACM, 2003: 59-68
- [13] Li Fangmin, Xu Wenjun, Gao Chao. A power control MAC protocol for wireless sensor networks [J]. Journal of Software, 2007, 18(5): 1080-1091 (in Chinese)
(李方敏, 徐文君, 高超. 一种适用于无线传感网络的功率控制 MAC 协议 [J]. 软件学报, 2007, 18(5): 1080-1091)
- [14] Tian Le, Xie Dongliang, Han Bing, et al. Study on bottleneck nodes in wireless sensor networks [J]. Journal of Software, 2006, 17(4): 830-837 (in Chinese)
(田乐, 谢东亮, 韩冰, 等. 无线传感器网络中瓶颈节点的研究 [J]. 软件学报, 2006, 17(4): 830-837)

如何学习天线设计

天线设计理论晦涩高深, 让许多工程师望而却步, 然而实际工程或实际工作中在设计天线时却很少用到这些高深晦涩的理论。实际上, 我们只需要懂得最基本的天线和射频基础知识, 借助于 HFSS、CST 软件或者测试仪器就可以设计出工作性能良好的各类天线。

易迪拓培训(www.edatop.com)专注于微波射频和天线设计人才的培养, 推出了一系列天线设计培训视频课程。我们的视频培训课程, 化繁为简, 直观易学, 可以帮助您快速学习掌握天线设计的真谛, 让天线设计不再难...



HFSS 天线设计培训课程套装

套装包含 6 门视频课程和 1 本图书, 课程从基础讲起, 内容由浅入深, 理论介绍和实际操作讲解相结合, 全面系统的讲解了 HFSS 天线设计的全过程。是国内最全面、最专业的 HFSS 天线设计课程, 可以帮助你快速学习掌握如何使用 HFSS 软件进行天线设计, 让天线设计不再难...

课程网址: <http://www.edatop.com/peixun/hfss/122.html>

CST 天线设计视频培训课程套装

套装包含 5 门视频培训课程, 由经验丰富的专家授课, 旨在帮助您从零开始, 全面系统地学习掌握 CST 微波工作室的功能应用和使用 CST 微波工作室进行天线设计实际过程和具体操作。视频课程, 边操作边讲解, 直观易学; 购买套装同时赠送 3 个月在线答疑, 帮您解答学习中遇到的问题, 让您学习无忧。

详情浏览: <http://www.edatop.com/peixun/cst/127.html>



13.56MHz NFC/RFID 线圈天线设计培训课程套装

套装包含 4 门视频培训课程, 培训将 13.56MHz 线圈天线设计原理和仿真设计实践相结合, 全面系统地讲解了 13.56MHz 线圈天线的工作原理、设计方法、设计考量以及使用 HFSS 和 CST 仿真分析线圈天线的具体操作, 同时还介绍了 13.56MHz 线圈天线匹配电路的设计和调试。通过该套课程的学习, 可以帮助您快速学习掌握 13.56MHz 线圈天线及其匹配电路的原理、设计和调试...

详情浏览: <http://www.edatop.com/peixun/antenna/116.html>



关于易迪拓培训:

易迪拓培训(www.edatop.com)由数名来自于研发第一线的资深工程师发起成立,一直致力和专注于微波、射频、天线设计研发人才的培养;后于 2006 年整合合并微波 EDA 网(www.mweda.com),现已发展成为国内最大的微波射频和天线设计人才培养基地,成功推出多套微波射频以及天线设计经典培训课程和 ADS、HFSS 等专业软件使用培训课程,广受客户好评;并先后与人民邮电出版社、电子工业出版社合作出版了多本专业图书,帮助数万名工程师提升了专业技术能力。客户遍布中兴通讯、研通高频、埃威航电、国人通信等多家国内知名公司,以及台湾工业技术研究院、永业科技、全一电子等多家台湾地区企业。

我们的课程优势:

- ※ 成立于 2004 年, 10 多年丰富的行业经验
- ※ 一直专注于微波射频和天线设计工程师的培养,更了解该行业对人才的要求
- ※ 视频课程、既能达到了现场培训的效果,又能免除您舟车劳顿的辛苦,学习工作两不误
- ※ 经验丰富的一线资深工程师主讲,结合实际工程案例,直观、实用、易学

联系我们:

- ※ 易迪拓培训官网: <http://www.edatop.com>
- ※ 微波 EDA 网: <http://www.mweda.com>
- ※ 官方淘宝店: <http://shop36920890.taobao.com>