

基于蚁群优化算法的 Ad Hoc 网络路由协议研究

高伟峰

(五邑大学计算机学院, 广东 江门 529020)

摘要: 路由协议的研究是 Ad Hoc 网络的重点和难点。文章在分析已有 Ad Hoc 网络路由协议的基础上, 结合蚁群算法的原理, 首次提出了根据蚂蚁释放有效信息素的比率, 引入服务代理来减少蚂蚁代理开销的路由协议。在网络仿真平台 NS-2 上进行了算法仿真实验, 并与现有的典型协议在不同的实验场景下进行性能分析比较, 仿真结果表明新的基于蚁群优化算法的 Ad Hoc 网络路由协议具有比其它算法更好的性能。

关键词: Ad Hoc 网络; 路由协议; 蚁群算法; NS-2

中图分类号: TP393

文献标识码: A

文章编号: 1673-1131(2014)10-0032-03

1 蚁群算法在通信中的研究

最近几年, 许多学者将 ACO(Ant Colony Optimization)算法应用于通信领域, 特别是用它来解决网络路由问题。通信网络中的路由是通过路由表进行的, 在每个节点的路由表中, 对每个目的节点都列出了与该节点相邻的节点, 当有数据报到达时, 通过查询路由表可知下一个将要到达的节点。

Ruud Schoonderwoerd 等人最早提出的基于自适应路由算法 ABC(Ant-based Control)使用的是一个和蚂蚁网络非常类似的方法, 该算法是为电话网而特地设计的。在该算法中, 蚂蚁被放在网络的各个节点中, 这些蚂蚁通过调整所在节点的路由表信息素值去适应网络结构的动态变化, 而且该算法可以根据网络的需要来调整参数值。Schoonderwoerd 等人发现该算法比其它现存的算法更具有健壮性。在算法的实现过程中, 蚂蚁旅行只在一个方向上进行, 并且可能在阻塞的节点中被延迟。在到达某个节点的蚂蚁的作用会随着这个蚂蚁的年龄(旅行时间)的增加而减少。为打电话而使用一个路径可能导致它的阻塞, 这将会依次引起包括阻塞节点路由接入能力的降低。在蚂蚁决策移动方向时添加噪声或抖动的方法, 证明能够促进随机探测。

受路由算法 ABC 的启发, Di Cargo 和 Dorigo M 在相关的文献中将蚁群优化算法 ACO 应用于网络路由问题, 并称这种

算法为 AntNet^[5]。AntNet 是一种自适应、基于移动代理的算法, 是受蚂蚁群体行为启发而设计的。在一些分组交换通信网络中, 该算法被认为是目前该领域内存在的路由算法中表现最好的算法。

在 AntNet 算法中, 要求每一个节点都要维持一个路由表, 这个路由表给每个目的地提供选择某个邻近节点作为下一跳节点的可能性。在实际的网络中, 总是选择具有最高可能性的下一跳节点。每个节点将周期性地发起前向蚂蚁网络探测代理到每一个目的地, 在每个节点处, 前向蚂蚁将按照该节点的路由表选择他们的下一跳。当蚂蚁代理访问某个节点时, 将把到达的时间和节点标志记录进堆栈中。到达目的地的前向蚂蚁代理将被转换成一个后向蚂蚁代理, 并向源节点发送此后向蚂蚁代理。

后向蚂蚁代理将沿着前向蚂蚁代理走过的路径返回源节点, 在到达某个前向蚂蚁访问过的节点时, 后向蚂蚁代理会调用该节点堆栈中的记录, 并将后向蚂蚁的到达时间拿来同前向蚂蚁的到达时间作比较。这样就得到到达目的地一个往返旅行时间。这个往返旅行时间用于和到达那个目的地的平均来回旅行时间作比较。如果新的往返时间更小, 那么选择那个路径的可能性就增加了; 如果新的时间大一些, 那么选择那个路径的可能性就降低。

10.5 GHz, 21.2 GHz 和 31.8 GHz 拍频信号, 相应的拍频强度为 -38.497 dBm, -44.926 dBm 和 -59.785 dBm, 如图 3(b) 所示。从图 3(b) 还可以看到, 随着拍频频率的升高, 信号强度呈下降趋势, 这种现象是由光电探测器带宽受限引起的。本文所用光电探测器的带宽只有 20GHz, 若换用较高带宽的探测器, 这种现象是可以避免的。

通过调整偏振控制器, 可改变激光器的工作状态, 从而可实现拍频信号的频率转换, 图 4 为光栅激光器在不同工作状态下的拍频输出。

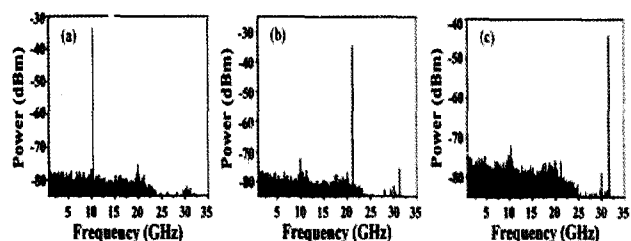


图 4 光纤光栅激光器不同工作状态下的拍频输出

2 结语

本文提出了一种基于光外差和 FBG 的微波信号产生方案, 在理论上详细分析了 FBG 的工作原理, 并在实验上验证了系统的可行性, 获得了良好的效果。该方案的核心在于利用 FBG 实现多波长光纤激光器, 通过调整其输出光的偏振状态, 可实现光生成微波信号及其微波频率的转换。本方案结构简单, 成本低, 具有独特的优势和良好的实用价值。

参考文献:

- [1] Jian Zhang, Hongwei Chen, Minghua Chen, Tianliang Wang, and Shizhong Xie. Photonic generation of a millimeter-wave signal based on sextuple-frequency multiplication. Opt. Lett., 2007, 32(9):1020-1022
- [2] M. Brunel, F. Bretenaker, S. Blanc, V. Crozatier, J. Brisset, T. Merlet, A. Poezevara. High-spectral purity RF beat note generated by a two-frequency solid-state laser in a dual thermooptic and electrooptic phase-locked loop. IEEE Photon. Technol. Lett., 2004, 16(3):870-872

随着蚂蚁算法的不断发展,在通信领域已经得到越来越多的应用,不论在应用层,接入层或是网络层都有蚂蚁算法的应用研究。

2 基于蚁群优化的 Ad Hoc 网路由算法

受到蚂蚁寻找最短路径方法的启发,我们可以在网络的各个节点上保存一个路由表,该表的元素值可以仿效蚂蚁的挥发物质 pheromone,我们把它设为选择某一相邻节点的概率值。每个人工蚂蚁(移动 agent)模拟现实蚂蚁在网络中寻找路由,它可以根据路由表上的概率值选择路径,又可以反过来修改路由表上的概率值,从而寻找到合适的路由。蚂蚁 agent 将搜索到的信息交给服务 agent,然后服务 agent 根据所搜索到的信息进行路由表的更新。

路由算法 ACOAHR 中的 agent 可以分为三类:一类是前向蚂蚁代理 F_{ant} (Forward ant agents),表示从源节点到目的节点的蚂蚁 agent,在其向目的节点的前进过程中收集信息;一类是后向蚂蚁代理 B_{ant} (Backward ant agents),表示从目的节点返回到源节点的蚂蚁 agent;另一类是服务代理 S_{ant} (Service ant agents),根据接收 B_{ant} 的信息或自己搜集的信息对路由表进行更新。

该算法的设计分为两个部分:路由的建立、路由的探索与维护。

我们用图 $G=\{V, E\}$ 来表示实现中的网络,其中 V 是网络中所有节点的集合, E 是连接两节点的边集。每个节点维护一个路由信息表 T ,每个蚂蚁可根据路由信息表选择一条合适的路由进行发送数据。假设蚂蚁在节点 i ,信息表 T_i 如下表 1 所示,其中 D 是某个节点可以选择的目的节点数, N 为该节点相邻的节点数,表 1 给出了以某个节点为目的节点时选择下一个节点的概率。

表 1 路由信息表

$P_{1,1}$	$P_{1,2}$	$P_{1,D}$
$P_{2,1}$	$P_{2,2}$	$P_{2,D}$
.....
$P_{N,1}$	$P_{N,2}$	$P_{N,D}$

假设 $\tau_{n,d}^i$ 是在节点 i 通过相邻节点 n 到达目的节点 d 的信息素的值,那么蚂蚁利用以下规则选择下一跳节点 n ,用概率 $P_{n,d}$ 来表示:

$$P_{n,d} = \frac{(\tau_{n,d}^i)^\alpha}{\sum_{j \in N_d'} (\tau_{j,d}^i)^\alpha} \quad (1)$$

$$\text{其中, } \sum_{j \in N_d'} P_{j,d} = 1 \quad (2)$$

上式中, α 表示蚂蚁代理探测路径行为的控制参数 $\alpha \geq 1$, N_d' 表示节点 i 的邻居节点数集合, i 通过这些节点到达目的节点 d 。

F_{ant} 在向目的节点寻找路由的过程中,如果节点的信息表中的值有效,它根据路由信息表中数据随机选择下一跳(概率大的选择的机率大),否则, F_{ant} 将进行广播,因此,一个中间节点或者目的节点可能收到同一个 F_{ant} 的复制品,此节点只接收到最早到达而且跳数最少的那个 F_{ant} ,将其余的丢弃。

从源节点到目的节点的过程中,每个 F_{ant} 将所访问过节点

的信息保存在 F_{ant} 本身的一个栈中:

$$P = \{(s, t, d_{s,j_1}^s, 0), (j_1, t, d_{j_1,j_2}^{j_1}, h_1), \dots, (j_k, t, d_{j_k,d}^{j_k}, h_k)\} \quad (3)$$

其中, s, j_1, j_2, \dots, d 为节点序号, t 为蚂蚁代理在节点上选择下一条所用的时间, $d_{j_1,j_2}^{j_1}$ 表示节点 j_1 到邻居节点 j_2 的时延, h_k 为跳数。若出现环路,即 F_{ant} 又返回到一个已经访问过的节点时,所有与形成环路的节点有关的信息全部去掉, F_{ant} 然后在邻居节点中按相同的概率值选择其它的一个以免再次进入环路。 F_{ant} 到达目的节点 d 或者有到达目的节点路由的中间节点后,转换成后向蚂蚁代理 B_{ant} , B_{ant} 包含了 F_{ant} 收集到的所有路径信息。 B_{ant} 根据路径信息以更高的优先级沿原路径返回,在每个中间节点 i , B_{ant} 可以读出从该节点到相邻节点的时延 d_{i,j_1}^i 以及经该节点到目的节点的跳数,并根据(4)计算出从此节点 i 经该相邻节点 j_1 到达目的节点 d 的时延。

$$d_{j_1,d}^i = d_{i,j_1}^i + d_{j_1,j_2}^{j_1} + \dots + d_{j_k,d}^{j_k} \quad (4)$$

然后把这些信息交给服务代理 S_{ant} 。据此,我们可利用 S_{ant} 进一步计算出信息素的增量:

$$\Delta \tau_{j_1}^i = \left(\frac{d_{j_1,d}^i + h T_{hop}}{2} \right)^\beta \cdot e^{-\gamma} \quad (5)$$

其中, $\Delta \tau_{j_1}^i$ 表示从节点 i 选择 j_1 作为下一跳的信息素的增量。 T_{hop} 是一个参数,代表蚂蚁代理在节点上选择下一条所用的时间。 $\beta \leq -1$, 用来调整信息素增量的速度。 e^γ 表示蚂蚁释放有效信息素的比率,在初始时刻,有效信息素量最大,随着时间的推移,释放的有效信息素越来越少,其中, t 表示时间, γ 控制有效信息素的参数。我们用下式来调整信息素的值:

$$\tau_{j_1,d}^i = (1 - \lambda) * \tau_{j_1,d}^i + \Delta \tau_{j_1}^i \quad (6)$$

其中, λ 为信息素衰减的系数,取值范围 $0 < \lambda < 1$ 。

最后,根据式(1)更新路由信息表。当 B_{ant} 到达源节点时,正向路由建成。

经过路由建立之后,网络中存在一系列保存有“有信息素计算的路由信息表”的节点,这些节点会按照一定算法对路由表进行维护更新。

在源节点发送数据时,多个数据报同时从一条路由上传送,由于 Ad Hoc 网络本身带宽受限,会导致数据包拥塞,从而使现存的路由可能不是最优的路由。因此,源节点在每发送 n 个数据报的同时,发送 F_{ant} 进行维护与探索。这些 F_{ant} 一般来说是单播,根据公式计算出来的概率选择下一跳,但也可能以一定的概率(假设为 ϵ)进行广播。采用这种方法我们有两个目的:

(1) 如果 F_{ant} 一直是单播,根据概率选择下一跳,到达目的节点,它可以搜集本条路由信息,据此用 B_{ant} 对路由信息表进行更新维护。

(2) 如果 F_{ant} 根据某个概率进行广播,它可以离开已存在的路径而探测到一条新的最优路径。

另外,我们数据传输的同时,每经过一段时间利用 S_{agent} 访问其相邻节点,搜集相关信息,来对本节点的信息素表进行更新。

3 仿真结果与协议分析

为了验证协议 ACO-AHR 的有效性,我们对此协议进行

了仿真试验,试验平台是 NS2 (Network Simulation Version 2)。NS2 是一个离散事件模拟器,适合做无线网络实验^[7]。本次仿真我们选取两个网络场景,分别使用 ACOAHR、AODV、DSR 和 DSDV 作为节点间通信时的路由协议,我们在两种不同的场景下进行仿真试验。所有节点的初始位置是随机的,而且移动的方向也是随机的,移动的速度为 0-20m/s,节点在某处的停留时间随试验场景的不同而定,仿真时间为 200s。另外,为了直接考察路由性能,本文采用 CBR(常速数据流)数据源,并将发送率设为每秒四个 512B 大小的数据包,信道带宽为 2Mbps/s。

在第一个场景中,50 个节点在 $800 \times 400\text{m}^2$ 中移动,我们主要测试节点在目的地停留时间不同(停留时间为 0-180s)时的性能。第二个场景中,我们主要测试节点个数对路由性能的影响,节点个数的变化范围为 10~100,虚拟环境为 $1000 \times 300\text{m}^2$,节点的停留时间为 30s。

3.1 平均端到端的延迟

平均端到端的延迟会随着网络节点数的增加而增加,随着节点暂停时间变化不太有规律,这是因为在寻找有效的到达目的节点的路由的时候,数据包必须在接口队列中等待一段时间。但总体上,采用 DSDV 协议网络分组平均时延相对最小,而 AODV 端到端延迟较大,DSR 更大。这是因为 DSDV 属于先应式路由协议,在需要发送数据时,通过查表直接获取路由,所以延迟时间较小。而 DSR 和 AODV 协议属于按需式路由协议,在需要发送数据的时候才寻找可用的路由,因此延迟较大。而 ACOAHR 所得到的平均端到端的延迟比 DSR 比 AODV 低,和 DSDV 相差不大。虽然 ACOAHR 虽然在路由建立时是按需建立路由,和 AODV 一样,但是我们在维护和探索时采用主动的策略,是一种混合是路由协议,这和我们采取的措施有关,总的来说有两个方面的原因:①数据包的寻路时间比较短,由于服务代理的出现,减少了蚂蚁代理大小,并且减少了在节点的停留时间,将计算任务交给了服务代理。②得到更优的路径。

3.2 数据包的传输率

图 1 和图 2 分别显示了各路由协议在两个场景中,数据包的传输率之间性能的关系比较图。传输率描述了传输层的丢失率,它表明协议的可靠程度。从下图我们可以看出,随节点的暂停时间和节点数的增加,传输率总体来说是下降的,这主要是因为在这样的情况下数据包的丢失很多。但 DSDV 随节点的暂停时间略有上升,由于路由协议 DSDV 比较适合拓扑变化不快环境,这和我们的分析一致。从图 4.4 可以看出,按需路由在节点数不多的环境下传输率是比较高的,尤其是动态源路由协议 DSR。

路由协议 ACOAHR 的数据包传输率比其它三种路由算法总体上来说好一些,在我们实验中显示出了比较全面的优良性能,这是由于在链路中断时,我们采取一定的措施,从而使数据包的丢失减少,提高了数据的传输率。这和我们预想的结果基本一致,因为这几种协议的实现机制不同。DSDV 协议中的各节点都维护到全网所有节点的路由,当节点的频繁移动,导致建立新的到全网节点的路由比较困难,使得节点没有可用路由而丢包,按需路由协议 AODV 和 DSDV 就尽量避免出现这种情况,而我们所设计的路由协议 ACOAHR 是一个混合路由协议,吸收了两方面的优点,将按需建立路由和主动对路由进行维护和探索结合起来,达到较好的性能。

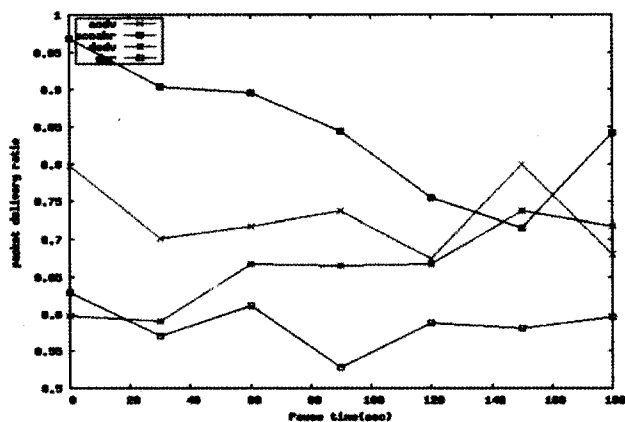


图 1 数据包的传输率与节点暂停时间的关系

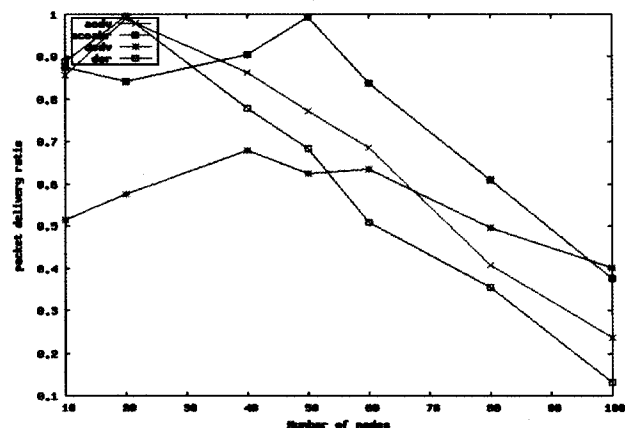


图 2 数据包的传输率与节点数的关系

4 总结与展望

从以上分析可以得出,算法 ACOAHR 的综合性能在一些场景下已经达到或超过路由协议 AODV、DSR、DSDV,虽然在网络开销和吞吐率表现还不尽如意,但是,通过对算法的进一步设计,希望能达到更好的效果。在算法本身的效率方面,ACOHR 比其它三个也要稍高一些,已经达到了该算法的部分设计目标。更重要的是 ACOHR 算法在数据包的时延和传输率明显好于其它三个,这对于今后进行网络路由的 Qos 支持将会提供很大帮助。

综上所述,改进的蚁群算法路由协议应用于无线移动自组织网的网络环境是有效和可行的,其网络性能比传统 AODV 路由协议的网络性能有较大提高,可作为一种新的 Ad Hoc 网络路由协议方案。

参考文献:

- [1] 郑相全等编著.无线自组网技术实用教程[M].北京:清华大学出版社,2004
- [2] C.E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector routing (DSDV) for Mobile Computers", Comp. Commun. Rev, Oct. 1994, 234-244
- [3] 段海滨著.蚁群算法原理及其应用[M].北京:科学出版社,2005
- [4] M. Heissenbittel and T. Braun. Ants-Based Routing in Large Scale Mobile Ad-Hoc Networks, Kommunikation in verteilten Systemen (KiVS03), March 2003

作者简介:高伟峰(1978-),男,湖北大悟人,讲师,硕士,研究方向为网络安全与网络存储。