



西北工业大学

本科毕业设计

英文翻译

专业名称 计算机科学与技术

学生姓名 裘莹

指导教师 李士宁

完成时间 2009.06

本科毕业设计英文翻译

指导教师评阅意见

学生姓名：_____ 班级：_____ 得分：_____

请指导教师用红笔在译文中直接进行批改，并就以下几方面填写评阅意见，给出综合得分（满分按 15 分计）。

1. 专业术语、词汇翻译的准确性；
2. 翻译材料是否与原文的内容一致；
3. 翻译材料字数是否符合要求；
4. 语句是否通顺，是否符合中文表达习惯。

指导教师（签名）：_____

年 月 日

四位无线链路估计

Rodrigo Fonseca, Omprakash Gnawali, Kyle Jamieson, Philip Levis

译：裘莹

摘要

我们考虑 ad-hoc 无线网络中的链路质量估计问题。我们认为要作好链路质量估计必须综合网络层、链路层和物理层的信息。我们为各层设计了窄的、精确的、与协议无关的链路估计接口，它总共提供 4 位信息：物理层 1 位，链路层 1 位，网络层 2 位。在我们大规模的多跳测试中，使用了这些接口的设计和实现，发现它比现有的方法减少了 44% 的包传输开销，获得了 99% 的传输率。

1 简介

精准的链路质量估计是高效的无线网络路由所必须的：拙劣的链路估计会使网络吞吐率降低 200% 以上^[5]。除了它的重要性之外，链路估计还有一个问题，这是各种因素共同作用使它具有挑战性，比如介质链路估计的流行^[19]，无线信道的时变特性^[15]，多路码间干扰^[2]，非对称连接^[10]，以及平台的变化^[20]。此外，物理层、链路层和网络层都拥有有价值的信息以改进链路估计值，比如信道质量，包传输率，路由设施和确认帧。特定芯片和协议可以提供的丰富的信息，这使设计更具复杂性。许多协议都使用跨层设计，每个层都可以使用与协议相关的信息以改进性能。

在本文中，我们提供了一种不同的方法。我们将物理层、链路层和网络层回馈的信息提炼成窄接口用于精确的链路估计。这些接口提供了 4 位信息：1 位来自物理层，1 位来自链路层，2 位来自网络层。这些信息位是协议无关的，这使各层分离并且避免造成不可预见的依赖从而阻碍网络的演进。

为了检验 4 位法的有效性，我们考虑将它用于棘手的无线网络——无线传感器网络。与具有较高能量供应的无线网络不同，无线传感器网络中的 RAM 限制意味着它不能存储所有邻居的状态。这个限制要求路由 IP（例如加上 6lowpan^[1]）需要良好的路由信息。因此，并不能只关心链路估计的精确性，还要能选择合适的邻居用于估计。

协议栈的每层都可以提供信息以达成这个目标。从物理层我们可以在一个包的传输中测量信道的质量。并不是所有的包都是一样的：错误位比较少的包很可能比错误多的包链路质量好。物理层的测量是快速并且廉价的，可以避免估

计数器在网络边缘或劣质的连接上浪费精力。我们可以把它提炼为一个 white 位，它表示在接收包时信道链路质量的好坏。

从链路层中，我们可以测量包是否被传输并确认。基于广播探测的估计器面临着一个问题，比如 ETX^[5]和 MintRoute^[18]，它们将链路估计和数据通信分开实现：如果链路质量变坏导致包丢失，链路估计器要直到下一个路由信标被丢弃时才能反映这种变化。我们可以把它也提炼为一个 ack 位，表示节点在一次传输中是否收到 2 层的确认。

在网络层，我们可以了解到哪个连接是高层性能上最有价值的。如果没有 3 层的信息，估计器可能选择一条路由环路，或者在最坏的情况下会与网络失去连接。不止一个无线传感器网络的部署曾因 2 层和 3 层链路表的不一致而导致故障^[9]。我们可把这些信息提炼为 2 个位：pin 位，用于告诉估计器不要剔除正在使用的连接；compare 位，用于告诉网络层这个连接看起来颇有希望。

本文有三个研究贡献。第一，在第 2 节，我们指出每层可以为链路估计提供的有用信息并从实验上分析每层不能识别而导致失败的情况。第二，在第 3 节，我们定义了一个窄接口的集合，用于为链路估计器提供每层可以获取的信息。第三，在第 4 节中，我们描述了一个使用这些接口的估计器原型，并评价了它与现有方法的改进程度。尽管这些接口只提供了 4 位信息，但是实验显示我们的估计器好于现有的跨层方法。我们把我们的估计器与 MultiHopLQI 作了比较，后者是许多传感器协议和系统所使用的估计器^[3,6,11,14,17]。在 Intel mirage 测试平台^[4]上，我们的链路估计器减少了 29% 的包传输开销，并且保持了 99.9% 的传输率，而 MultiHopLQI 只有 93%。在 USC Tutenet 测试平台上，我们的链路估计器减少了 44% 的包传输开销并保持 99% 的传输率，而 MultiHopLQI 只有 85%。此外，由于这个估计器独立于三个层，因此它可以很容易地与大多数协议协同工作。

这些结果力证了链路估计器可以从特定的层实现中分离出来并保持高效性和精确性。以这种方式分离链路估计器简化了网络协议栈并促进了协议的演进并且改进了互操作性。

2 层的限制

链路估计器必须是精确和有效的。它应当提供良好的链路质量估计，以及对变化的灵敏检测，同时要最小化内存需求和通信开销。每一层都可以为链路估计提供有价值的信息（图 1）。我们认为链路估计器应当使用所有三层的信息来达到这些目标，这不仅是因为每层可以提供的信息是唯一的或者是能更廉价地获得，还因为有的链路状态某些层可以检测到而其它的则不可以。根据物理层中每个包的质量所计算出的信道质量估计并不总能检测到临时的变化。链路层可以精确地计算 ETX，但不能廉价地决定估计哪一个连接。网络层知道哪个连接是路由中最有用的，但网络层的链路估计是效率低下的，并且适应得比较慢。

为了给我们的讨论打下基础，我们先看一种名为汇聚的多跳通信，多个节点用任意播的方式将数据发送到若干基站中的一个。这是在无线传感器网络中最

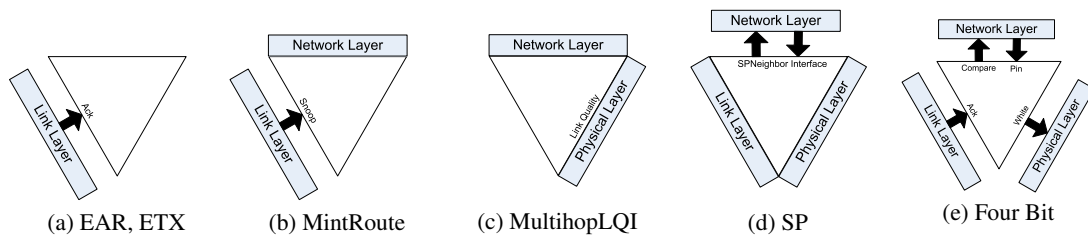


图 1 每个图中中间的三角形代表链路估计器与三个层间的相互联系。边上附着的框表示相结合的实现。指出的箭头表示估计器向接收到的包请求信息。指入的箭头表示某一层主动提供的信息。

常见的通信形式。然而我们的结论也同样适用与多数 any-to-any 多跳通信。

作为我们如何利用信息来作链路估计例子，我们先看 TinyOS 2 中的两种汇聚协议，汇聚树协议 (CTP)^[16]和 MultiHopLQI^[13]。CTP 使用基于检测的链路估计器，而 MultiHopLQI 仅依赖于 CC2420 无线芯片提供的链路质量指示器 (LQI)。我们在有 85 个节点的测试平台上用较低的传输率作汇聚实验，每个节点隔 10 秒种产生一个包。图2展示了 CTP (a)，MultiHopLQI (b)，没有路由表大小限制的 CTP (c) 产生的路由树。同时它也标出了平均代价，即每个投递到的包的传输次数。较低的代价意味着较短的路径和较好的质量。

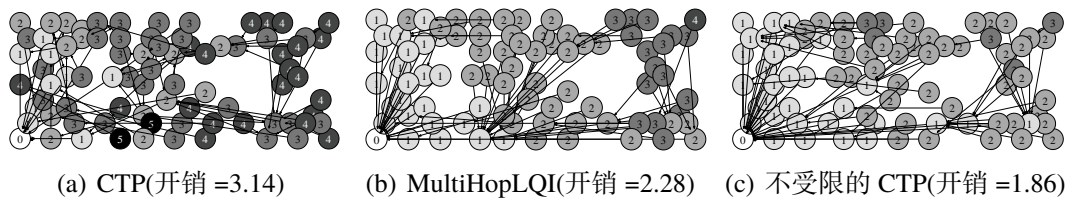


图 2 由具有 10 个节点的连接表的 CTP、MultiHopLQI 和连接表不受限的 CTP 在 85 个节点上形成的路由树。在括号中指出了投递每个消息的平均开销。左下角的是根节点，颜色越暗的节点意味着到根的路径越长。

CTP 的开销比 MultiHopLqi 高，尽管后者只用到物理层的信息。这有两个原因：一是由于 CTP 使用双向的基于探测的链路估计器，它的连接表大小限制了节点的入度。第二，仍然是因为连接表的大小限制，它可能使最好的连接并不在连接表中。图2展示了当连接表没有限制时，CTP 胜过了 MultiHopLQI。在第 4 节中，我们展示了如何使用物理层、链路层和网络层的信息来缓和这些问题。下面的小节详尽地讲述各层的利用价值和限制。

2.1 物理层

物理层可以通过解包来提供介质的质量信息。这个信息提供了一种避免边界或边缘问题的快速且廉价的方法。它可以增加估计器的灵敏度，并且为连接表项提供第一层过滤。在图2中，MultiHopLQI(c) 和 SP(d) 在链路估计中使用了物理层信息。

由于物理层信息属于单个包，并且它只能对收到的包作估计，故信道的变化会使它不准确。比如，在低功耗无线个人网络中，大多数连接是双向模式的^[15]，

在高质量（100%PRR，packet reception ratio）和低质量（0%PRR）间变化。在这种连接中，接收者如果只使用物理层的信息的话，那它将会发现许多高信道质量的包，从而认为该连接是好的，即使中间丢了不少包。

图3展示了物理层信息的限制，我们观察了在 94 个节点上使用 MultiHopLQI 协议的 12 个小时低速率汇聚实验。如图1所示，MultiHopLQI 不使用链路层的信息。尽管总体上来看这个协议表现的不错，但是仍然有突发的包丢失现象。

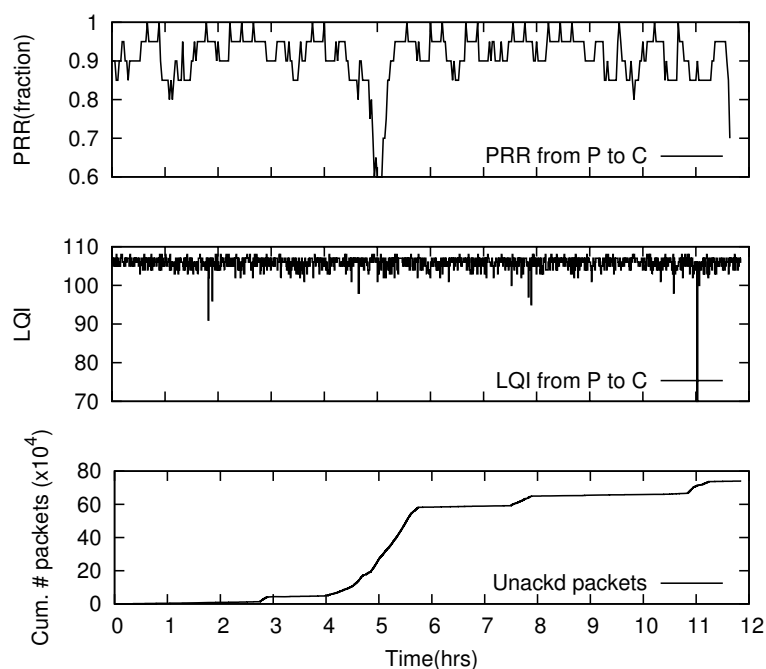


图3 由于没有意识到 PRR 的下降，MultiHopLQI 在第 4 和第 6 个小时尝试在同一条链路上发包从而导致由于重传而未被确认的包增加。

如图3所示，在一段时间中，节点 C 和 P 间的 PRR 从 0.9 降到将近 0.6。这个连接质量的下降并不伴随着 LQI 解码质量的下降。C 收到的包的质量都很好，但它并没有收到全部的包。

2.2 链路层

链路估计器如 ETX 和 MintRoute 使用周期性的广播来估计包的接收速率。这些估计器计算双向链路质量——包被传输并收到 ACK 的概率——即两个方向链路质量的乘积。虽然简单，这种方法适应地比较慢，并且需要假设周期性的广播和数据通信的行为相似。

通过使用 2 层的 ACK 统计收到 ACK 或没有收到 ACK 的包个数，链路估计器可以在同等的数据通信量上计算出更准确的估计值。这些估计值也是双向的。在图1(a)中，EAR 和 ETX 使用链路层的反馈来作链路估计。通过数据通信的链路层信息可以计算出比取控制包接收速率的乘积这种方式更精确的 ETX。然而，尽管这很精确、很相关、很快速，但是发送数据包需要路由信息，而生成路由信息又需要链路质量估计值。这种自举最好在低层完成。同时，特别是在密集

的网络中，选择合适的连接的集合来估计与如何计算估计值是同样重要的，如果仅在链路层做的话代价会比较昂贵。

2.3 网络层

物理层可以提供连接质量的粗略估计，让估计器避免在边缘或坏的连接上白花力气。一旦估计器估计了一个连接，网络层就可以决定哪些连接是有价值的。这在连接表有限的情况下是很重要的。比如，地域路由^[7]可以从邻居是向各个方向平等的蔓延这一特性中获取有用信息，S4 路由^[12]能从到信标的距离最小这一特性中获取有用信息。最近有一个不知名的无线传感器网络只能投递收集到的数据中的 2%，部分原因是由于网络层和链路层使用连接的不一致所造成。由于一些原因，MintRoute 协议（图 1(b)）在网络层中整合了链路估计器。SP（图 1(d)）为网络层提供了丰富的接口，用于检查和更改估计器的邻居表。网络层发现邻居并作链路估计，但并不使用重传、ACK 或是包解码质量这些信息，这样的估计是适应得较慢并且是代价昂贵的。

在下面的部分中我们描述了链路估计器如何有效地使用三层的信息，它只用 4 位的信息，而且有一些定义地很明确的接口。我们在第 4 节中演示了通过层间有效信息的交换使我们的接口的性能得到了很大的提升。

3 设计

第 2 部分展示了每层可以得到的有助于链路估计的信息。物理层可以为到达的包估计传输介质的质量。链路层可以知道包是否被传输并确认。网络层可以确定估计哪个连接是最值得的。本节提出了三层间的接口并描述了一个使用这些接口的链路估计器。

3.1 估计接口

图 3.1 展示了各层为链路估计器提供的接口。三层总共提供 4 位的信息：两位用于到达的包，一位已传输的单播包，一位用于链路表项。

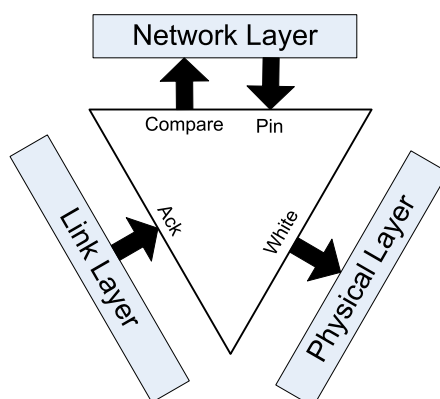


图 4 中间的三角形代表链路估计器，使用三个层的四位信息。指出的箭头表示估计器向接收到的包发出请求。指入的箭头表示某一层主动提供的信息。

物理层提供一位的信息。white 位如果设为 1，则表示接收到的包解码出错

的概率很低。white 位说明在接收中介质质量是很好。反过来说则不一定正确。如果 white 位是 0, 则介质质量可能是不好的, 也可能是好的。

链路层为每个已传输的包提供了一位的信息: ACK 位。链路层在收到 2 层确认的时候设置 ACK 位。如果 ACK 位是 0, 则说明包可能没有成功地传输, 也可能已经成功的传输了。

网络层提供了两位的信息, pin 位和 compare 位。pin 位用于连接表项。如果网络层在表项上设置了 pin 位, 则链路估计器在 pin 位被清 0 前不能把它从表中移除。链路估计器可以为一个包向网络层获取 compare 位信息。compare 位表示发送者提供的路由是否比连接表中的一项或多项要好。我们在 3.3 节中描述了链路估计器是如何使用 compare 位的。

3.2 接口考虑

我们认为 4 位的信息是链路估计器所需的信息的最小集合。此外, 我们相信接口也足够简单, 可以在大多数系统上实现。比如, 无线的物理层提供的信号强度和噪声信号可以用于计算信噪比, 根据信噪比错误率曲线, 使用阈值的方法计算 white 位。物理层还可以使用比特错误恢复报告或芯片纠错率。在最坏的情况下, 可以永不设置 white 位。

接口对链路层有一个限制: 要有 2 层的同步确认。虽然这看上去要求很高, 但其实大多数常用的链路层比如 802.11 和 802.15.4 都有这一功能。在这个模型中, 新的或特定程序的链路层必须有 2 层确认。

compare 位需要网络层能分辨包发送者的路由是否比连接表中的当前表项的路由要好。compare 位并不要求网络层对所有的包都作比较, 而只选择它们的一个子集。这意味着诸如路由信标中会含有路由质量信息。

3.3 一个混合的估计器

我们将描述一个混合的估计器, 它使用了周期性的信标帧, 结合了三层的信息, 提供精确、时效和有用的链路估计值。该估计器维护一个较小的连接表 (比如大小为 10), 内含候选连接的 ETX 值。它周期性地广播包含这些连接信息的信标帧。网络层协议也通过估计器广播包, 从而使它看起来像一个 2.5 层的协议, 它在 3 层到 2 层的包中加头部加尾部。

估计器使用由 Woo et al.^[18]描述的表管理算法, 但有一个例外: 它并不假设最小传输率, 因为它可以根据出站数据传输率来检测坏的连接。链路估计广播的包中含有顺序号, 接收者使用它来计算信标接收率。

估计器使用 white 位和 compare 位对标准的表替换策略作补充。当它接收到一个 white 位为 1 的网络层的路由包并且发送者并不在连接表中时, 估计器询问网络层 compare 位是否已设置。如果设置了, 则估计器随机地剔除一个没有 pin 住的表项, 并用当前包的发送者代替。

估计器使用 ACK 位使链路估计值更准确。结合广播和单播 ETX 估计为一个混合的值, 这是一种在连接通信量不断变化的网络中使用的方法^[8]。我们使用 Woo et al.^[18]描述的链路估计方法, 分别为单播包和广播包计算 k_u 或 k_b 个包

的 ETX 值。如果 k_u 个发出的包中有 a 个被接收者确认，则单播包的 ETX 估计值为 $\frac{k_u}{a}$ 。如果 $a = 0$ ，则估计值是从上次成功的接收算起的失败传输次数。对于广播包的估计也是类似的，但是有一个额外的步骤。我们在计算出的接收率上使用窗口化的指数移动平均值（EWMA），并转化连续的采样值为 ETX 值。从两种估计方法中得到的 ETX 值通过第二次 EWMA 结合起来，如图 3.3 所示。结果是数据和信标窗口化 EWMA 估计值的混合。当数据流较大时，单播包的估计值占主导地位。当网络比较安静时，广播估计占主导地位。

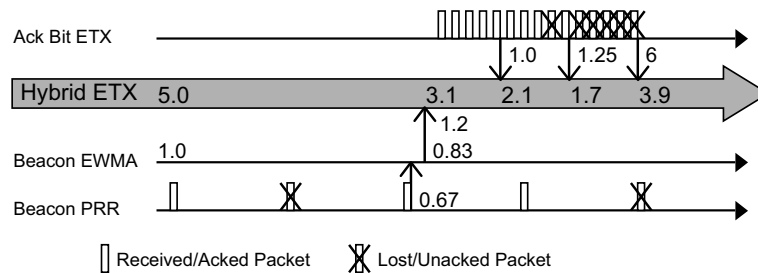


图 5 我们的链路估计器结合了单播包和广播通信的估计生成 ETX 值，使用的窗口大小分别为 $k_u = 5$ 和 $k_b = 2$ 。后者在结合之前要对自身作一次平均。我们用小方框表示到来的包，给丢失的包标上“X”。估计器在垂直箭头处计算估计值。

与大多数纯广播的估计器相比，我们的估计器并不通过信标帧积极地交换和维护双向估计值。因为 ack 位也可以测量双向连接特性，因而我们的估计器可以避免只使用信标帧估计作为链路估计的自举值。这是一个很重要的特性，它把网络拓扑中的节点入度从连接表中分离出来。

4 评价

我们在 TinyOS 2 中实现了 3.3 节所描述的链路估计器原型，通过替换 CTP 中标准的链路估计器来测试它。我们的估计器对物理层、链路层、网络层使用 4 位接口，我们修改了 CTP 中的这些层以使用该接口。在本节中，我们通过实验和比较详细地评测了我们的原型、原来的 CTP 和 MultiHopLQI。在下面的讨论中，我们的原型称为“4B”。MultiHopLQI 如上面所述用了链路质量指示器（LQI），这是 CC2420 无线模块具有的特性，对具有该模块的系统来说，MultiHopLQI 是目前 TinyOS 中性能最好的汇聚协议实现。

在比较中，我们在 Mirage 测试平台上运行了这三个协议，用 85 个 MicaZ 节点，其中一个节点作为基站。同时我们也在另一个测试平台 TutorNet 上作了测试，使用了 94 个 TelosB 节点。传输功率若非特别指出都设为 0dBm。在每个实验中，我们使用统一分布错开启动时间在 30 秒的范围内。每个节点发送一个有些许抖动的汇聚包以避免与其它包产生包同步。每个节点提供的工作负载是常数速率发送到根节点的包流。这会使网络中产生并发的传输流汇聚到根节点。在 Mirage 上实验时间在 40 到 60 分钟之间。在 TutorNet 上的实验时间则要长的多，在 3 到 12 小时左右。测试平台是静态的并且我们所有的实验结果在各个实验平台上一致，这给了我们信心确信较短的运行时间得出的结果也将是一致的。

我们评价性能的主要标准是开销：每个单播包传输的总次数。开销的重要性在于它直接联系到网络的生存时间。它考虑了路径的跳数、每个连接的重传数以及由于包丢失而造成的浪费。为了考虑开销，我们也注意到了拓扑树的平均深度。如果所有的连接是完好的，平均深度将是开销的下界。两者间的差异意味着选择连接的质量，这可能是由于重传或丢包所造成的。最后我们还考查了投递率，即根节点收到唯一消息的百分率。

我们首先看 3.1 节中附加的每个位是如何影响开销和路由长度的。我们比较原来的 CTP 和 4B，以及两种中间的实现。在图4中左边最上面的点展示了运行在 Mirage 测试平台上的 CTP 的开销和深度。为 CTP 增加单向链路估计和 ack 位能减少 93% 的平均树深度，并减少 31% 的开销。单向估计把入度从连接表大小中分离出来，因此深度大大减小。

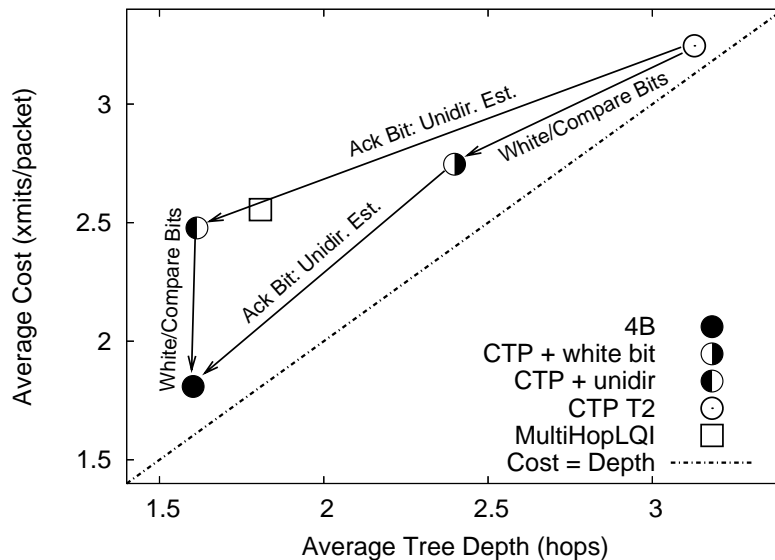


图 6 在 CTP 中增加 ack 位、white 位和 compare 位以降低开销（越低越好）和节点在路由树中平均深度。

与原来的 CTP 相比，增加 white 和 compare 位能减少 55% 的开销，这可能是由于父节点的选择得到了改进。为 CTP 仅增加 white 和 compare 位使开销减少 15%，平均深度减少 23%。该图还列出了在相同测试平台上的 MultiHopLQI 的开销和深度以作比较。仅当我们使用全部三层的信息时，4B 才比 MultiHopLQI 好。4B 的开销要低 29%，路径要短 11%。在 TutorNet 测度平台上，4B 的开销和平均深度相应的比 MultiHopLQI 低 44% 和 9.7%。4B 产生的树和平均树深度与不限制路由表大小的 CTP（图 2）非常类似。

图4比较了在 Mirage 测试平台上的 4B 和 MultihopLQI 的开销和平均节点深度。发射功率在 -20dBm 到 0dBm 间变化。在每个协议中，我们看到平均节点深度和开销随着传输功率的减小而增大，因为节点需要经过更多跳的路由。4B 相对于 MultihopLQI 在开销方面的改进在 11% 到 29% 之间，而在平均深度上的改进在 11% 到 3.5% 之间。在 0 到 -10dBm 间，4B 的开销最多也就是下界的 13%，而 MutiHopLQI 的是 43.4%。在多跳网络中，这两个协议都效果不好，相应开销

的增加（平均深度 4B 增加 62% 以上，MultiHopLQI 为 95%）意味着重传和包丢失增加。即使树深度相似，4B 能选择更好的连接。

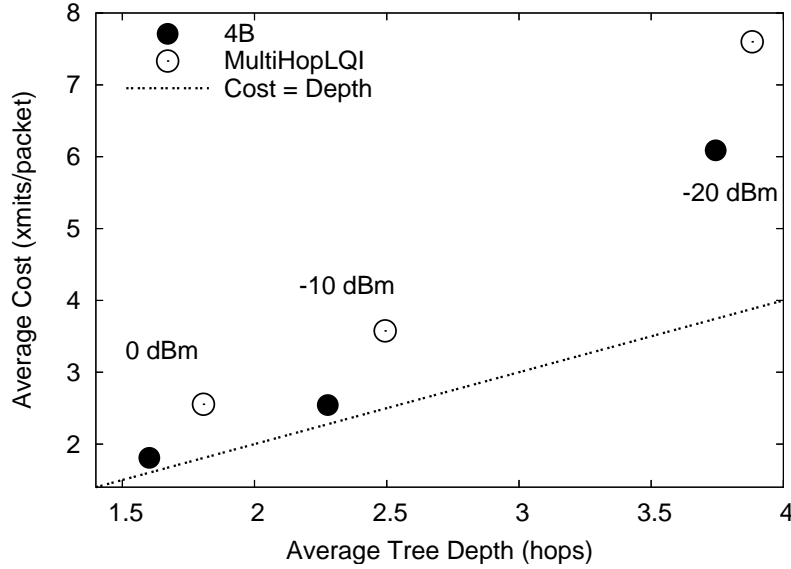


图 7 在 Mirage 测试平台上降低传输功率时 MultiHopLQI 和 4B 的平均节点深度和开销。4B 可以降低 19%-28% 的开销。

图4展示了每个节点的投递率，并解释了图3.1中开销比平均树深度增长地快的原因。在 0 到 10dBm 间，4B 的平均投递率在 99.9% 以上，最低是 99.3%。在 0dBm 时，MultihopLQI 的平均投递率是 95.9%，最坏的节点是 64%。当发射功率下降时，相应的 RF 噪声的影响加重，使网络中局部不对称。如第 2 节中的例子，MultihopLQI 的性能下降是由于这种连接质量的变化不能被物理层指示器捕捉到。我们打算观察网络的动态行为，但是即使在 -20dBm 下，4B 的包丢失数也很少，这意味着开销的增大主要由于重传，而不是包丢失。这说明估计器足够灵敏，可以检测到包丢失并切换到新的路由。

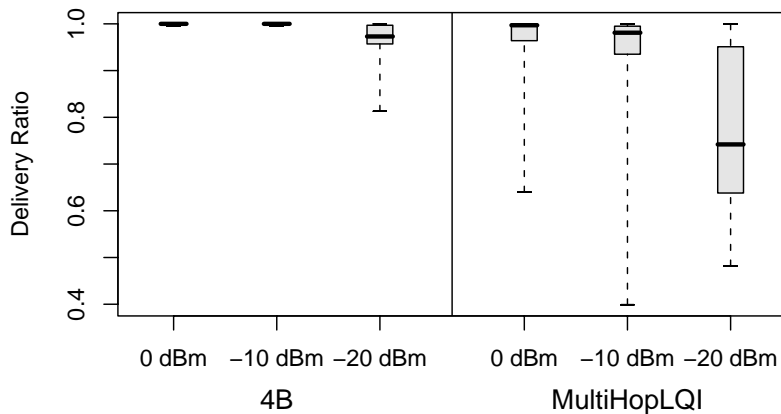


图 8 在发送功率逐渐降低的情况下，4B 和 MultihopLQI 中每个节点的投递率分布框图。端点表示最大和最小值。框表示四分之一和四分之三的位置。粗线是中位线。4B 维持了较高和较一致的投递率。

5 结论

在本文中我们提出了一个窄的、完善定义的接口，它允许链路估计器使用来自物理层、链路层和网络层的信息。我们的原型展示了在开销和投递率上很大的改进，同时也保持了网络的分层抽象结构。这是令人鼓舞的，因为我们还没有完全掌握使用 4 位接口的所有特性。一个可移植、精确和有效的链路估计器在规模与日俱增的网络、低功耗的 IP 扩展和像 IETF 6lowpan 这类嵌入式网络中越来越重要。比如，TCP 的性能是对包丢失非常敏感的，通过使用我们的链路估计器达到的改进可能会对端到端的吞吐量产生很大的影响。

参考文献

- [1] 6lowpan charter. <http://www.ietf.org/html.charters/6lowpan-charter.html>.
- [2] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris. Link-level measurements from an 802.11 b mesh network. In *Applications, Technologies, Architectures, and Protocols for Computer Communication: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications: Portland, Oregon, USA*. Association for Computing Machinery, Inc, One Astor Plaza, 1515 Broadway, New York, NY, 10036-5701, USA., 2004.
- [3] Moteiv Corp.: Boomerang. <http://www.moteiv.com/software>.
- [4] BN Chun, P. Buonadonna, A. AuYoung, C. Ng, DC Parkes, J. Shneidman, AC Snoeren, and A. Vahdat. Mirage: A microeconomic resource allocation system for sensornet testbeds. In *Embedded Networked Sensors, 2005. EmNetS-II. The Second IEEE Workshop on*, pages 19--28, 2005.
- [5] DD Couto, D. Aguayo, J. Bicket, and R. Morris. A highthroughput path metric for multi-hop wireless networks. *Proceedings of the 9th Annual MOBICOM*, 2003.
- [6] O. Gnawali, K.Y. Jang, J. Paek, M. Vieira, R. Govindan, B. Greenstein, A. Joki, D. Estrin, and E. Kohler. The tenet architecture for tiered sensor networks. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 153--166. ACM New York, NY, USA, 2006.
- [7] B. Karp and HT Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 243--254. ACM New York, NY, USA, 2000.
- [8] K.H. Kim and K.G. Shin. On accurate measurement of link quality in multi-hop wireless mesh networks. In *Proceedings of the 12th annual international conference on Mobile computing and networking*, pages 38--49. ACM New York, NY, USA, 2006.
- [9] K. Langendoen, A. Baggio, and O. Visser. Murphy loves potatoes: Experiences from a pilot sensor network deployment in precision agriculture. In *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, page 8, 2006.
- [10] R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan. Analyzing the MAC-level Behavior of Wireless Networks in the Wild. In *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 75--86. ACM New York, NY, USA, 2006.

- [11] G. Mainland, M. Welsh, and G. Morrisett. Flask: A language for data-driven sensor network programs. *Harvard Univ., Cambridge, MA, Tech. Rep. TR-13-06*, 2006.
- [12] Y. Mao, F. Wang, L. Qiu, S.S. Lam, and J.M. Smith. S4: Small state and small stretch routing protocol for large wireless sensor networks. In *Proc. of NSDI*, 2007.
- [13] MultiHopLQI. <http://www.tinyos.net/tinyos-1.x/tos/lib/multihoplqi>, 2004.
- [14] S. Rangwala, R. Gummadi, R. Govindan, and K. Psounis. Interference-aware fair rate control in wireless sensor networks. In *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 63--74. ACM New York, NY, USA, 2006.
- [15] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis. Understanding the causes of packet delivery success and failure in dense wireless sensor networks. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 419--420. ACM New York, NY, USA, 2006.
- [16] Tinyos 2 tep 123: The collection tree protocol. <http://www.tinyos.net/tinyos-2.x/doc/txt/tep123.txt>.
- [17] G. Werner-Allen, G. Tewari, A. Patel, M. Welsh, and R. Nagpal. Firefly-inspired sensor network synchronicity with realistic radio effects. In *Proceedings of the 3rd international conference on Embedded networked sensor systems*, pages 142--153. ACM New York, NY, USA, 2005.
- [18] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 14--27. ACM New York, NY, USA, 2003.
- [19] J. Zhao and R. Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 1--13. ACM New York, NY, USA, 2003.
- [20] M. Zuniga and B. Krishnamachari. Analyzing the transitional region in low power wireless links. In *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, pages 517--526, 2004.