

无源感知网络中采用动态分片的 IPv6 数据包传递方案

邱树伟¹, 李英龙², 袁利永³

(1. 汕头职业技术学院 计算机系, 广东 汕头 515078; 2. 浙江工业大学 计算机科学与技术学院, 杭州 310014;
3. 浙江师范大学 行知学院, 浙江 金华 321004)

摘 要: 在无源感知网络(PSN)中, 基于网络编码的 IPv6 包传递方案大多采用固定的分片个数, 传输时延较高。为此, 考虑到无源传感器节点的特性, 提出一种适用于 PSN 的 IPv6 包传递方案 IPDS-DF。基于偏移正交相移键控调制方式的误码率模型, 推导出无线链路的成功传输概率, 进而得出每跳的平均传输时延、平均能耗和端到端数据包递交率。以最小化每跳的传输时延为目标, 对传输路径上每个节点所采用的分片个数进行动态优化。实验结果表明, IPDS-DF 的平均传输时延和能耗均低于现有 IPv6 包传递方案。

关键词: 无源感知网络; 网络编码; 网际协议; 物联网; 传输时延

中文引用格式: 邱树伟, 李英龙, 袁利永. 无源感知网络中采用动态分片的 IPv6 数据包传递方案[J]. 计算机工程, 2017, 43(7): 100-109.

英文引用格式: Qiu Shuwei, Li Yinglong, Yuan Liyong. IPv6 Packet Delivery Scheme Using Dynamic Fragment in Passive Sensing Network[J]. Computer Engineering, 2017, 43(7): 100-109.

IPv6 Packet Delivery Scheme Using Dynamic Fragment in Passive Sensing Network

QIU Shuwei¹, LI Yinglong², YUAN Liyong³

(1. Department of Computer Science, Shantou Polytechnic, Shantou, Guangdong 515078, China;
2. School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, 310014, China;
3. Xingzhi College, Zhejiang Normal University, Jinhua, Zhejiang 321004, China)

[Abstract] In Passive Sensing Network(PSN), Network Coding(NC) based IPv6 packets delivery schemes mostly use fixed number of fragments, which leads to long transmission delay. In order to reduce the transmission delay, an IPv6 Packet Delivery Scheme called IPDS-DF which can be applied to PSN is proposed considering the characteristics of PSN nodes. The Successful Transmission Probability(STP) is derived based on the BER model of Offset Quadrature Phase Shift Keying(OQPSK). Furthermore, the average transmission delay, the average energy consumption and the end-to-end Packet Delivery Ratio(PDR) are also derived. For minimizing the transmission delay, the number of fragments in each hop of the route are dynamically optimized. Experimental results indicate that the IPDS-DF outperforms the existing Reliable IPv6 Packet Delivery Scheme(RIPDS) in terms of average transmission delay and average energy consumption.

[Key words] Passive Sensor Network(PSN); network coding; Internet protocol; Internet of Things(IoT); transmission delay

DOI: 10.3969/j.issn.1000-3428.2017.07.017

0 概述

无源感知网络(Passive Sensing Network, PSN)是物联网感知层中一种新型的无线传感器网络(Wireless Sensor Network, WSN)。PSN 由没有配备电池的无源传感器节点组成, 节点通过从环境中捕获能量来维持网络的运行。例如, PSN 节点可以从阳光、风力、温度、振动、压力以及射频信号等周围环

境捕获能量^[1]。PSN 节点的能量不依赖于电池, 有效地解决了 WSN 节点能量受限的问题。因此, PSN 在人体健康监测、车载网络以及工业控制等方面有着广泛的应用前景^[2]。

在物联网中, 位于 IPv6 网络中的用户常常需要向 PSN 中特定的节点发送控制/查询命令, 这些命令被封装在 IPv6 数据包(下文简称“IPv6 包”)中。例如, 战场环境监测系统可通过 IPv6 网络向部署在战

基金项目: 国家自然科学基金(61502421); 汕头职业技术学院创新强校项目(STP-ZZ-011)。

作者简介: 邱树伟(1979—), 男, 讲师、硕士, 主研方向为分布式计算; 李英龙, 讲师、博士; 袁利永, 讲师、硕士。

收稿日期: 2016-11-08 **修回日期:** 2016-12-16 **E-mail:** city-qsw@163.com

场上的 PSN 节点发送 IPv6 包,以控制相应的节点进行特定的操作。这类应用要求在可靠传递 IPv6 包的前提下,尽可能地降低传输时延^[3]。

由于 IPv6 包的最大传输单元(Maximum Transmission Unit, MTU)至少为 1 280 Byte,而基于 IEEE 802.15.4 标准的 PSN 节点所能支持的数据包的最大长度仅为 127 Byte,除去头部、尾部和安全选项的开销,一个 PSN 数据包的最大有效载荷仅为 81 Byte。因此,向 PSN 节点传递 IPv6 包时,需借助 6LoWPAN(IPv6 over Low-Rate Wireless Personal Area Network)协议,6LoWPAN 协议先将 IPv6 包进行分片,然后将每个分片逐一传递到 PSN 中的目的节点,最后,目的节点将所接收到的所有分片重组回原始 IPv6 包。

在 IPv6 包传递的过程中,任何一个分片丢失,目的节点就无法对 IPv6 包进行重组。为了解决这个问题,文献[4-7]利用 NC 来提高 IPv6 包传递的可靠性。假定源节点 A_0 位于 IPv6 网络与 PSN 的边界(即网关),它通过路径 $A_0 \rightarrow A_1 \rightarrow \dots \rightarrow A_n$ (假定 A_0 至 A_n 的路径已事先由网络层的路由算法所确定)将 IPv6 包传递给位于 PSN 中的目的节点 A_n , n 表示路径的总跳数, $n \in \mathbb{Z}^+$ 且 $n \geq 1$ 。基于 NC 的 IPv6 包传递方案如下: A_0 先将 IPv6 包分为 N 个分片并编码为 M 个编码包($M > N$),再将编码包逐一发送给 A_1 。当 A_1 正确接收到 N 个编码包时(M 个编码包中的任意 N 个),便向 A_0 回复一个 ACK 确认包(下文简称“ACK 包”),以告知 A_0 不用再继续发送。接着, A_1 将接收到的编码包进行解码并重组 IPv6 包。然后, A_1 再对 IPv6 包进行“分片-编码-发送”等操作。后续节点的操作以此类推,直到 A_n 成功重组 IPv6 包为止。

显然,上述方案能明显提高 IPv6 包传递的可靠性,但是该方案存在 2 个问题:1) 每个节点均采用固定的分片个数(均为 N),且分片个数的选取并未考虑到当前无线链路的质量,传输时延较高;2) 仅适用于传统 WSN,并不能直接应用于 PSN(PSN 节点是无源传感器节点,这种节点时常因能量短缺而暂停工作)。

为解决上述 2 个问题,本文提出一种采用动态分片的 IPv6 包传递方案。该方案考虑了 PSN 节点的特性,结合无线链路的质量,通过优化方程对传输路径上每个节点所采用的 IPv6 包分片个数进行动态优化。

1 相关工作

在 PSN 节点能量捕获方面,文献[1]给出了在能量捕获材料面积为 10 cm^2 的情况下,节点在捕获

热能、太阳能、室内光能以及振动能等能量源时的平均能量捕获速率(单位为 J/s)。文献[8]提出了节点预测太阳能捕获规律的指数加权移动平均算法;文献[9]提出了基于天气条件的移动平均算法,这种方法不受能量源的限制,能大体上反映 PSN 节点能量捕获的一般规律。文献[10]认为能量源的出现是一个泊松过程,通过泊松过程的特点和相关的概率论知识,可以求解出节点处于稳定状态下的平均能量捕获速率。以上成果大多通过预测模型或随机过程模型来求解节点在某个时间段之内的平均能量捕获速率,很好地刻画了节点的能量捕获特性。

在基于 NC 的可靠数据传递方面,文献[4]采用 6LoWPAN 协议将 IPv6 包可靠传递给无线个域网(Wireless Personal Area Network, WPAN)中的目标节点。为了提高数据包的成功递交率,文献[4]将 IPv6 数据包分片编码成一组编码包,然后按组逐跳传递给目的节点,当目的节点收到这组编码包时,可以还原出 IPv6 包。文献[5]介绍了 Reed Solomon 编码技术及其应用,采用 Reed Solomon 编码时,接收方一旦收到固定个数的编码包,就能解码并还原出原始数据包,而且,Reed Solomon 编码的传输开销较小,易于在资源受限的传感器节点上实现。文献[6]在 WSN 中采用基于擦除码的数据传递方案,该方案在传输路径上的每个节点均运行擦除码,并且通过逐跳的“解码-编码-发送”操作来减少源节点的编码开销,同时减少了冗余的数据包;文献[7]为了解决 WSN 中端到端数据包传输的低效率问题,采用了喷泉码编码方案,提高了数据传递的可靠性,该方法以较低的开销获得较高的传输成功率。

从以上的研究成果可以看出,利用 NC 技术能有效提高数据传递的可靠性。但是,由于以上方案都是采用固定的数据包长度,当固定分片个数时,传输时延较长,而且这些研究工作仅针对传统 WSN 中的应用,并不适用于 PSN。

2 IPDS-DF 方案

2.1 等待时延的定义

由于 PSN 节点工作时所需的能量来自环境,节点在发送/接收数据包之前,如果其剩余能量不足以发送/接收,那么需要耗费一定的时间进行能量捕获,此时,节点处于等待发送/接收状态,直到节点所捕获的能量足以发送/接收时,才开始发送/接收。由此可见,PSN 节点的传输时延应包含“等待时延”和“发送时延”2 个部分,即传输时延 = 等待时延 + 发送时延。

(A_i, A_{i+1}) 表示 A_i 和 A_{i+1} 之间的无线链路, $i = 0, 1, \dots, n-1$, A_i 和 A_{i+1} 是相邻节点。 $W_{i,i+1}$ 和

$T_{i,i+1}$ 分别表示 A_i 在链路 (A_i, A_{i+1}) 上的等待时延和发送时延。

等待时延 $W_{i,i+1}$ 的定义如下:

$$W_{i,i+1} = \max\{C_i, C_{i+1}\} \quad (1)$$

其中, C_i 表示节点 A_i 从 t_i 时刻开始为了捕获足以成功发送 IPv6 包所需的能量所持续的时间(单位为 s), $t_i \in [0, \infty)$; C_{i+1} 表示节点 A_{i+1} 从 t_i 时刻开始为了捕获足以成功接收 IPv6 包所需的能量所持续的时间。 $W_{i,i+1}$ 这样定义是为了保证在 A_i 开始发送时, A_{i+1} 总是有足够的能量用于接收。

发送时延 $T_{i,i+1}$ 是指 A_i 在链路 (A_i, A_{i+1}) 上成功发送 IPv6 包所需的时间。 $C_i, C_{i+1}, W_{i,i+1}$ 和 $T_{i,i+1}$ 的关系如图 1 所示。

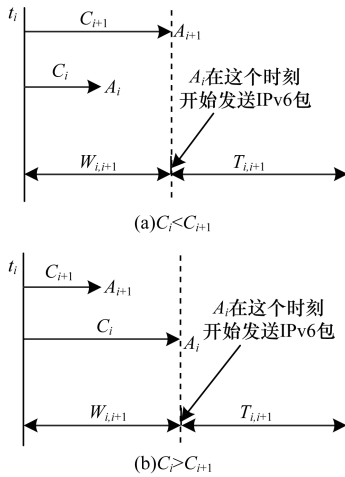


图 1 等待时延与发送时延示意图

在图 1(a) 中, 从 t_i 时刻开始, A_i 为了成功发送 IPv6 包需要事先充电 C_i s, 而 A_{i+1} 为了成功接收 IPv6 包需要事先充电 C_{i+1} s, 且 $C_i < C_{i+1}$ 。在这种情况下, A_i 不能在 $t_i + C_i$ 时刻就开始发送, 因为此时 A_{i+1} 的能量还不足以成功接收 IPv6 包, A_i 只能等到 $t_i + C_{i+1}$ 时刻(即从 t_i 时刻起等待 C_{i+1} s)才开始发送, 可得 $W_{i,i+1} = C_{i+1}$ 。同理, 图 1(b) 中, 当 $C_i > C_{i+1}$ 时, 有 $W_{i,i+1} = C_i$ 。易知, 当 $C_i = C_{i+1}$ 时, 有 $W_{i,i+1} = C_i$ 或 C_{i+1} 。当 A_i 和 A_{i+1} 在时刻 t_i 的剩余能量分别足够用于发送和接收时, 有 $C_i = C_{i+1} = 0$, 则 $W_{i,i+1} = 0$ 。

2.2 IPDS-DF 的操作流程

IPDS-DF 方案由“ A_0 的操作流程”“ A_i ($1 \leq i \leq n-1$) 的操作流程”和“ A_n 的操作流程”3 个部分组成。为了提高 IPv6 包传递的可靠性, IPDS-DF 采用了 NC 技术。在编码包传递的过程中, 每个节点要设置一个编码包计数器, 用于记录当前正确接收到的编码包个数, 不妨以 $Count$ 表示之, 初始值为 0。

以 $N_{i,i+1}$ 和 $M_{i,i+1}$ 分别表示在链路 (A_i, A_{i+1}) 中 A_i 所采用的分片个数和编码包个数。

A_0 的操作流程如下:

步骤 1 根据算法 Finding_Optimal_N 得到链路 (A_0, A_1) 的最优分片个数 $N_{0,1}$ 和等待时延 $W_{0,1}$, 将 IPv6 包分为 $N_{0,1}$ 片并编码为 $M_{0,1}$ 个编码包, 形成发送队列。

步骤 2 判断 $W_{0,1}$ 是否为 0, 若是, 则转向步骤 3; 否则等待 $W_{0,1}$ s 后转向步骤 3。

步骤 3 将发送队列中位于队首的编码包发送给 A_1 。若收到 A_1 回复的 ACK 包, 则停止发送并清空缓冲区中剩余的编码包; 否则返回步骤 3。

步骤 4 结束。

A_i ($1 \leq i \leq n-1$) 的操作流程如下:

步骤 1 接收来自 A_{i-1} 的编码包并判断是否出现误码, 若出现误码, 则弃之并返回步骤 1; 否则置 $Count = Count + 1$ 。

步骤 2 判断 $Count$ 的值是否等于 $N_{i-1,i}$, 若是, 则向 A_{i-1} 回复 ACK 包并转向步骤 3; 否则返回步骤 1。

步骤 3 对接收到的 $N_{i-1,i}$ 个编码包进行解码并重组 IPv6 包。

步骤 4 根据算法 Finding_Optimal_N 得到链路 (A_i, A_{i+1}) 的最优分片个数 $N_{i,i+1}$ 和等待时延 $W_{i,i+1}$, 将 IPv6 包分为 $N_{i,i+1}$ 片并编码为 $M_{i,i+1}$ 个编码包, 形成发送队列。

步骤 5 判断 $W_{i,i+1}$ 是否为 0, 若是, 则转向步骤 6; 否则等待 $W_{i,i+1}$ s 后转向步骤 6。

步骤 6 将发送队列中位于队首的编码包发送给 A_{i+1} 。若收到 A_{i+1} 回复的 ACK 包, 则停止发送并清空缓冲区中剩余的编码包; 否则返回步骤 6。

步骤 7 结束。

A_n 的操作流程与 A_i ($1 \leq i \leq n-1$) 的操作流程中的步骤 1 ~ 步骤 3 相同。以上流程的主要步骤如图 2 所示。

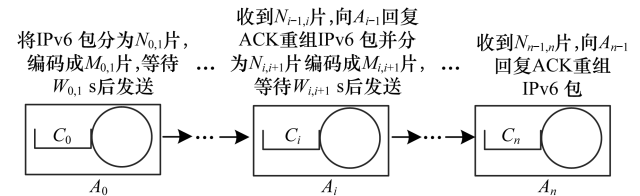


图 2 采用动态分片的 IPv6 包传递方案

从图 2 可以看出: 1) 发送节点总是要等到自身的剩余能量足以发送且接收节点的剩余能量足以接收时才开始发送; 2) 传输路径上每个节点所采用的

分片个数不是固定的。

3 数学模型

3.1 链路的成功传输概率

在 IEEE 802.15.4 标准中,对于采用直序扩频技术(Direct Sequence Spread Spectrum, DSSS)和偏移正交相移键控(Offset Quadrature Phase Shift Keying, OQPSK)调制方式的信道来说,链路(A_i , A_{i+1})的误码率 $b_{i,i+1}$ 可表示为^[11]:

$$b_{i,i+1} = Q(\sqrt{2\psi(d_{i,i+1})B_N/R}) \quad (2)$$

其中, B_N 和 R 分别表示噪声带宽和数据速率; $d_{i,i+1}$ 表示 A_i 和 A_{i+1} 之间的距离(单位为 m)。 $Q(\cdot)$ 是高斯尾部积分函数,有:

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-t^2/2} dt, x \geq 0 \quad (3)$$

$$\psi(d_{i,i+1}) = P_t - P_{\text{lost}}(d_0) - 10\eta \lg(d_{i,i+1}/d_0) - P_{\text{th}} \quad (4)$$

其中, $\psi(\cdot)$ 表示接收信噪比; P_t 表示发射功率; $P_{\text{lost}}(d_0)$ 表示单位距离 d_0 (1 m) 下的功率损耗; η 表示功率衰减系数,取值范围为 2 ~ 4; P_{th} 表示接收功率阈值(即节点接收功率大于或等于 P_{th} 时才能正常接收数据包)。将式(3)、式(4)代入式(2)中,可求得误码率 $b_{i,i+1}$ 。

以 L 表示 IPv6 包的长度,以 $L_{i,i+1}$ 表示链路(A_i , A_{i+1})中 A_i 所采用的数据包(即 IPv6 包分片)的有效载荷长度,则:

$$L_{i,i+1} = \lceil \frac{L}{N_{i,i+1}} \rceil \quad (5)$$

其中,符号“ $\lceil \cdot \rceil$ ”表示向上取整。以 H 和 F 分别表示数据包的头部长度和尾部长度。以 $L_{i,i+1}$ 表示数据包的总长度,则:

$$\begin{aligned} L_{i,i+1} &= H + L_{i,i+1} + F \\ &= H + \lceil L/N_{i,i+1} \rceil + F \end{aligned} \quad (6)$$

A_i 通过链路(A_i , A_{i+1})成功传输一个数据包给 A_{i+1} 的概率为:

$$\begin{aligned} p_{i,i+1} &= (1 - b_{i,i+1})^{L_{i,i+1}} \\ &= (1 - b_{i,i+1})^{H + \lceil L/N_{i,i+1} \rceil + F} \end{aligned} \quad (7)$$

3.2 编码包个数的选取

在传输成功概率为 $p_{i,i+1}$ 时,成功传输 1 个编码包所需传输的编码包的平均个数为 $\lceil \frac{1}{p_{i,i+1}} \rceil$ 。例如,当传输成功概率为 1 时,成功传输 1 个编码包所需传输的平均编码包个数为 $\lceil 1/1 \rceil = 1$ 个;而当传输成功概率为 0.5 时,成功传输 1 个编码包所需传输的平均编码包个数为 $\lceil 1/0.5 \rceil = 2$ 个。

根据图 2 中所描述的方案,为了使 A_{i+1} 能够正确接收到 $N_{i,i+1}$ 个编码包,可取编码包的个数为:

$$\begin{aligned} M_{i,i+1} &= \lceil \alpha \frac{N_{i,i+1}}{p_{i,i+1}} \rceil \\ &= \lceil \alpha N_{i,i+1} / (1 - b_{i,i+1})^{H + \lceil L/N_{i,i+1} \rceil + F} \rceil \end{aligned} \quad (8)$$

其中, α 是编码冗余度系数,取大于 1 的正数。在式(8)中加入系数 α 是为了进一步增加编码包的个数,从而增加 A_{i+1} 正确接收到 $N_{i,i+1}$ 个编码包的可能性。

3.3 平均传输能耗、平均发送时延和 PDR

首先,分析 A_i 发送单个编码包/ACK 包的平均发送能耗和平均发送时延。以 $E_{i,i+1}^{\text{Tx}}(l)$ 表示 A_i 向 A_{i+1} 发送长度为 l 的编码包的能耗,根据文献[12],有:

$$E_{i,i+1}^{\text{Tx}}(l) = l(\varepsilon_0 + \varepsilon_1 D^2) \quad (9)$$

其中, ε_0 表示节点发送或接收 1 bit 电路上的能耗; $\varepsilon_1 D^2$ 表示自由空间中节点发送 1 bit 时放大器的能耗; D 表示节点的通信半径。

以 $E_{i,i+1}^{\text{Rx}}(l)$ 表示 A_i 接收来自 A_{i+1} 的长度为 l 的编码包的能耗,有^[12]:

$$E_{i,i+1}^{\text{Rx}}(l) = l\varepsilon_0 \quad (10)$$

由于 A_i 在发送每个编码包时,不管发送成功与否,其发送能耗都是一样的,因此结合式(7)和式(9),可得到 A_i 向 A_{i+1} 发送一个长度为 $L_{i,i+1}$ 的编码包的期望能耗为:

$$\begin{aligned} \mu_{i,i+1}^{\text{Tx-data}} &= p_{i,i+1} E_{i,i+1}^{\text{Tx}}(L_{i,i+1}) \\ &\quad + (1 - p_{i,i+1}) E_{i,i+1}^{\text{Tx}}(L_{i,i+1}) = E_{i,i+1}^{\text{Tx}}(L_{i,i+1}) \end{aligned} \quad (11)$$

同理,结合式(7)和式(10), A_i 接收 A_{i+1} 回复的长度为 L_{ACK} 的 ACK 包的期望能耗为:

$$\mu_{i,i+1}^{\text{Rx-ack}} = E_{i,i+1}^{\text{Rx}}(L_{\text{ACK}}) \quad (12)$$

另外, A_i 向 A_{i+1} 发送一个长度为 $L_{i,i+1}$ 的编码包的发送时延为:

$$\tau_{i,i+1}^{\text{Tx-data}} = \sigma_c + \frac{L_{i,i+1}}{R} + T_{\text{IFS}} \quad (13)$$

其中, σ_c 表示节点发送编码包之前竞争信道的平均时间; R 表示数据速率; $L_{i,i+1}/R$ 表示编码包的发送时长; T_{IFS} 表示帧间间隔。帧间间隔可分为长帧间间隔(Long Interframe Spacing, LIFS)和短帧间间隔(Short Interframe Spacing, SIFS)。当数据包的长度小于或等于 18 Byte 时, $T_{\text{IFS}} = \text{SIFS}$, 否则, $T_{\text{IFS}} = \text{LIFS}$ 。

易知,不管编码包发送成功与否,其发送时延均一致,与式(11)的推导类似,可得到 A_i 向 A_{i+1} 发送一

个编码包的期望发送时延为:

$$\eta_{i,i+1} = p_{i,i+1} \times \tau_{i,i+1}^{Tx-data} + (1 - p_{i,i+1}) \tau_{i,i+1}^{Tx-ack} = \tau_{i,i+1}^{Tx-data} \quad (14)$$

由于节点发送 ACK 包时不需要竞争信道,因此 A_{i+1} 向 A_i 回复 ACK 包的发送时延为:

$$\tau_{i,i+1}^{Tx-ack} = \frac{L_{ACK}}{R} + T_{IFS} \quad (15)$$

其中, L_{ACK} , R 和 T_{IFS} 等参数均为常数,那么, $\tau_{i,i+1}^{Tx-ack}$ 也是一个常数。以 η_{ACK} 表示路径上任意节点发送 ACK 包的期望发送时延,显然, $\eta_{ACK} = \tau_{i,i+1}^{Tx-ack}$ 。

其次,分析 A_i 成功发送 1 个 IPv6 包给 A_{i+1} 的平均发送时延和平均能耗。从图 2 所示的方案可知, A_{i+1} 只有正确接收到 $N_{i,i+1}$ 个编码包时才能解码并重组 IPv6 包,为了得到 A_i 成功发送一个 IPv6 的平均发送时延和平均能耗,需事先求得 A_{i+1} 正确接收到

$N_{i,i+1}$ 个编码包的概率。随机事件“ A_i 发送了 $N_{i,i+1} + x$ ($x=0,1,\dots,M_{i,i+1}-N_{i,i+1}$) 个编码包而 A_{i+1} 正确接收到 $N_{i,i+1}$ 个”等价于随机事件“ A_{i+1} 成功接收到第 $N_{i,i+1} + x$ 个编码包且在此前的 $N_{i,i+1} + x - 1$ 个编码包中 A_{i+1} 刚好成功接收了 $N_{i,i+1} - 1$ 个”。以 $P_{i,i+1}(x)$ 表示上述的随机事件的概率,则:

$$P_{i,i+1}(x) = \binom{N_{i,i+1}+x-1}{N_{i,i+1}-1} (p_{i,i+1})^{N_{i,i+1}-1} (1-p_{i,i+1})^x p_{i,i+1} \\ = \binom{N_{i,i+1}+x-1}{N_{i,i+1}-1} (p_{i,i+1})^{N_{i,i+1}} (1-p_{i,i+1})^x \quad (16)$$

其中, $x=0,1,\dots,M_{i,i+1}-N_{i,i+1}$ 。在 A_{i+1} 正确接收到 $N_{i,i+1}$ 个编码包的条件下, A_i 传输的编码包的个数(至少为 $N_{i,i+1}$ 个)和对应的概率、发送时延和能耗如表 1 所示。

表 1 编码包发送情况

发送的编码包个数	概率	发送时延/s	能耗/J
$N_{i,i+1}$	$P_{i,i+1}(0)$	$N_{i,i+1}\eta_{i,i+1} + \eta_{ACK}$	$N_{i,i+1}\mu_{i,i+1}^{Tx-data} + \mu_{i,i+1}^{Rx-ack}$
$N_{i,i+1} + 1$	$P_{i,i+1}(1)$	$(N_{i,i+1} + 1)\eta_{i,i+1} + \eta_{ACK}$	$(N_{i,i+1} + 1)\mu_{i,i+1}^{Tx-data} + \mu_{i,i+1}^{Rx-ack}$
$N_{i,i+1} + 2$	$P_{i,i+1}(2)$	$(N_{i,i+1} + 2)\eta_{i,i+1} + \eta_{ACK}$	$(N_{i,i+1} + 2)\mu_{i,i+1}^{Tx-data} + \mu_{i,i+1}^{Rx-ack}$
...	...	...	...
$M_{i,i+1}$	$P_{i,i+1}(M_{i,i+1} - N_{i,i+1})$	$M_{i,i+1}\eta_{i,i+1} + \eta_{ACK}$	$M_{i,i+1}\mu_{i,i+1}^{Tx-data} + \mu_{i,i+1}^{Rx-ack}$

表 1 中的第 2 行表示 A_i 发送的编码包个数为 $N_{i,i+1}$ 的概率 $P_{i,i+1}(0)$, 对应的发送时延和能耗分别为:

$$N_{i,i+1}\eta_{i,i+1} + \eta_{ACK} \text{ 和 } N_{i,i+1}\mu_{i,i+1}^{Tx-data} + \mu_{i,i+1}^{Rx-ack}$$

第 3 行表示 A_i 发送的编码包个数为 $N_{i,i+1} + 1$ 的概率 $P_{i,i+1}(1)$, 对应的发送时延和能耗分别为:

$$(N_{i,i+1} + 1)\eta_{i,i+1} + \eta_{ACK} \\ (N_{i,i+1} + 1)\mu_{i,i+1}^{Tx-data} + \mu_{i,i+1}^{Rx-ack}$$

表 1 中从第 4 行起至最后一行的含义以此类推。

根据表 1, 可得到在 A_{i+1} 正确接收到 IPv6 包的条件下, A_i 的期望发送时延为:

$$E[T_{i,i+1}] = \frac{\sum_{x=0}^{M_{i,i+1}-N_{i,i+1}} P_{i,i+1}(x) [(N_{i,i+1} + x)\eta_{i,i+1} + \eta_{ACK}]}{\sum_{x=0}^{M_{i,i+1}-N_{i,i+1}} P_{i,i+1}(x)} \quad (17)$$

类似地, 以 $\theta_{i,i+1}^{Tx}$ 表示 A_i 成功发送 IPv6 包的能耗, 则其期望值为:

$$E[\theta_{i,i+1}^{Tx}] = \frac{\sum_{x=0}^{M_{i,i+1}-N_{i,i+1}} P_{i,i+1}(x) [(N_{i,i+1} + x)\mu_{i,i+1}^{Tx-data} + \mu_{i,i+1}^{Rx-ack}]}{\sum_{x=0}^{M_{i,i+1}-N_{i,i+1}} P_{i,i+1}(x)} \quad (18)$$

同理, 参考式(11)和式(12)中 $\mu_{i,i+1}^{Tx-data}$ 和 $\mu_{i,i+1}^{Rx-ack}$

的含义, 以 $\mu_{i,i+1}^{Rx-data}$ 和 $\mu_{i,i+1}^{Tx-ack}$ 分别表示 A_{i+1} 接收一个来自 A_i 的编码包的期望能耗和 A_{i+1} 向 A_i 回复 ACK 包的期望能耗。以 $\theta_{i,i+1}^{Rx}$ 表示 A_{i+1} 成功接收到 IPv6 包的能耗, 则其期望值为:

$$E[\theta_{i,i+1}^{Rx}] = \frac{\sum_{x=0}^{M_{i,i+1}-N_{i,i+1}} P_{i,i+1}(x) [(N_{i,i+1} + x)\mu_{i,i+1}^{Rx-data} + \mu_{i,i+1}^{Tx-ack}]}{\sum_{x=0}^{M_{i,i+1}-N_{i,i+1}} P_{i,i+1}(x)} \quad (19)$$

此外, 由于路径 $A_0 \rightarrow A_1 \rightarrow \dots \rightarrow A_n$ 的 PDR 等于路径上每一跳 IPv6 包的成功传输概率的累积, 可知:

$$PDR = \prod_{i=0}^{n-1} \sum_{x=0}^{M_{i,i+1}-N_{i,i+1}} P_{i,i+1}(x) \\ = \prod_{i=0}^{n-1} \sum_{x=0}^{M_{i,i+1}-N_{i,i+1}} \binom{N_{i,i+1}+x-1}{N_{i,i+1}-1} (p_{i,i+1})^{N_{i,i+1}} (1-p_{i,i+1})^x \quad (20)$$

路径的端到端数据包递交率 PDR 是衡量数据传递方案优劣的重要性能指标之一, PDR 越高, 说明 IPv6 包从信源成功传递到信宿的概率越大, 那么, IPv6 包传递的可靠性也就越高。在本文的 IPv6 包传递方案中, 通过设置合适的编码冗余度系数来提高 PDR, 以确保 IPv6 包传递的可靠性。

3.4 路径的端到端传输时延

以 t_i 表示从时刻 0 开始直到 A_i 成功接收到 IPv6

包所持续的时间, $i = 1, 2, \dots, n$ 。可知, t_1 等于链路 (A_0, A_1) 的传输时延, 即 $t_1 = W_{0,1} + T_{0,1}$; t_2 等于链路 (A_0, A_1) 和链路 (A_1, A_2) 的传输时延之和, 即 $t_2 = W_{0,1} + T_{0,1} + W_{1,2} + T_{1,2}$, 以此类推, 可得到:

$$t_i = \sum_{j=0}^{i-1} (W_{j,j+1} + T_{j,j+1}), i = 1, 2, \dots, n \quad (21)$$

进一步, 可以得到路径 $A_0 \rightarrow A_1 \rightarrow \dots \rightarrow A_n$ 的端到端传输时延的期望值为:

$$E[t_n] = \sum_{j=0}^{n-1} (E[W_{j,j+1}] + E[T_{j,j+1}]) \quad (22)$$

路径端到端传输时延如图 3 所示。

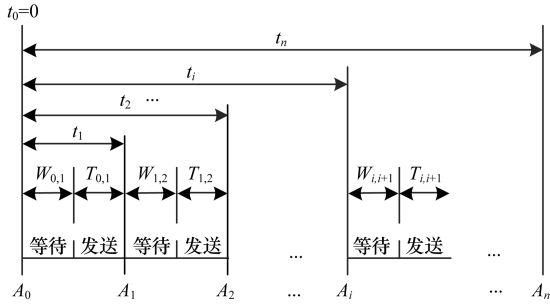


图 3 路径的端到端传输时延

3.5 平均等待时延

以 $e_i(t_i)$ 表示 A_i 在时间点 t_i 的剩余能量, $t_i \in [0, \infty)$ 。假设初始状态下所有节点的能量均为 e_0 , 即 $e_i(t_0) = e_0, i = 0, 1, \dots, n$ 。以 δ 表示节点的计算能耗, 以 $\beta (0 < \beta < 1)$ 表示节点电容器的充电效率系数, 以 ρ 表示电容器的平均漏电速率 (单位为 J/s), 以 λ_i 表示 A_i 的平均能量捕获速率。根据图 3, 在时间点 t_i, A_i 的剩余能量为:

$$e_i(t_i) = e_0 + \beta \lambda_i t_i - \rho t_i - \delta - E[\theta_{i-1,i}^{Rx}] \quad (23)$$

其中, $\beta \lambda_i t_i$ 和 ρt_i 分别表示 A_i 经历时间段 t_i 之后所捕获到的总能量和电容漏电的总量; $E[\theta_{i-1,i}^{Rx}]$ 表示 A_i 的接收能耗 (见式 (19))。

由于 A_{i+1} 在时间点 t_i 尚未接收到 IPv6 包, 不存在接收能耗, 因此 A_{i+1} 的剩余能量为:

$$e_{i+1}(t_i) = e_0 + \beta \lambda_{i+1} t_i - \rho t_i - \delta \quad (24)$$

现考虑如何求解 A_i 的平均等待时延 $E[W_{i,i+1}]$ 。根据式 (1), $W_{i,i+1}$ 的值由 C_i 和 C_{i+1} 共同决定。由式 (18) 可知, A_i 成功发送 IPv6 包的期望能耗为 $E[\theta_{i,i+1}^{Tx}]$, 若 A_i 在时间点 t_i 的剩余能量 $e_i(t_i) \geq E[\theta_{i,i+1}^{Tx}]$, 则说明此时 A_i 的剩余能量足以发送 1 个 IPv6 包, 则它不需要耗费时间来捕获能量, 即 $C_i = 0$; 若 $e_i(t_i) < E[\theta_{i,i+1}^{Tx}]$, 则说明此时 A_i 的剩余能量不足以发送 1 个 IPv6 包, 需要耗费一定的时间来捕获能量, 即 $C_i > 0$ 且 C_i 需满足以下方程:

$$e_i(t_i + C_i) = e_i(t_i) + \beta \lambda_i C_i - \rho C_i - \delta = E[\theta_{i,i+1}^{Tx}] \quad (25)$$

式 (25) 表示 A_i 从时间点 t_i 开始经历 C_i s 之后的剩余能量为 $E[\theta_{i,i+1}^{Tx}]$, 也即 A_i 需从时间点 t_i 开始至少等待 C_i s 之后才能开始发送 IPv6 包。

解方程式 (25), 得到:

$$C_i = \frac{E[\theta_{i,i+1}^{Tx}] - e_i(t_i) + \delta}{\beta \lambda_i - \rho} \quad (26)$$

类似地, 由式 (19) 可知, A_{i+1} 成功接收 IPv6 包的期望能耗为 $E[\theta_{i,i+1}^{Rx}]$, 若 A_{i+1} 在时间点 t_i 的剩余能量 $e_{i+1}(t_i) \geq E[\theta_{i,i+1}^{Rx}]$, 说明此时 A_{i+1} 的能量足以接收 1 个 IPv6 包, 则 $C_{i+1} = 0$; 若 $e_{i+1}(t_i) < E[\theta_{i,i+1}^{Rx}]$, 说明此时 A_{i+1} 的剩余能量不足以接收 1 个 IPv6 包, 则 $C_{i+1} > 0$, 且 C_{i+1} 需满足以下方程:

$$e_{i+1}(t_i + C_{i+1}) = e_{i+1}(t_i) + \beta \lambda_{i+1} C_{i+1} - \rho C_{i+1} - \delta = E[\theta_{i,i+1}^{Rx}] \quad (27)$$

这说明了 A_{i+1} 需从时间点 t_i 开始至少等待 C_{i+1} s 之后才能开始接收 IPv6 包。解方程式 (27), 得到:

$$C_{i+1} = \frac{E[\theta_{i,i+1}^{Rx}] - e_{i+1}(t_i) + \delta}{\beta \lambda_{i+1} - \rho} \quad (28)$$

综合以上分析, 可总结出:

$$E[W_{i,i+1}] = \begin{cases} C_i = C_{i+1} = 0, & e_i(t_i) \geq E[\theta_{i,i+1}^{Tx}] \wedge e_{i+1}(t_i) \geq E[\theta_{i,i+1}^{Rx}] \\ C_i = \frac{E[\theta_{i,i+1}^{Tx}] - e_i(t_i) + \delta}{\beta \lambda_i - \rho}, & e_i(t_i) < E[\theta_{i,i+1}^{Tx}] \wedge e_{i+1}(t_i) \geq E[\theta_{i,i+1}^{Rx}] \\ C_{i+1} = \frac{E[\theta_{i,i+1}^{Rx}] - e_{i+1}(t_i) + \delta}{\beta \lambda_{i+1} - \rho}, & e_i(t_i) \geq E[\theta_{i,i+1}^{Tx}] \wedge e_{i+1}(t_i) < E[\theta_{i,i+1}^{Rx}] \\ \max\{C_i, C_{i+1}\} = \max\left\{\frac{E[\theta_{i,i+1}^{Tx}] - e_i(t_i) + \delta}{\beta \lambda_i - \rho}, \frac{E[\theta_{i,i+1}^{Rx}] - e_{i+1}(t_i) + \delta}{\beta \lambda_{i+1} - \rho}\right\}, & \text{其他} \end{cases} \quad (29)$$

其中, $i = 0, 1, \dots, n-1$ 。将式 (29) 和式 (17) 代入式 (22), 可以求得路径的端到端期望传输时延 $E[t_n]$ 。

4 IPv6 分片个数的优化

为了最小化端到端传输时延 $E[t_n]$, 可以通过最小化每一跳的传输时延 $E[W_{i,i+1}] + E[T_{i,i+1}]$ 来实现。从式 (7) 中可以看到, 无线链路的成功传输概率 $p_{i,i+1}$ 与 IPv6 包分片个数 $N_{i,i+1}$ 的取值有关, 而 $p_{i,i+1}$ 又与每一跳的平均发送时延 $E[T_{i,i+1}]$ 和平均能耗 (包含 $E[\theta_{i,i+1}^{Tx}]$ 和 $E[\theta_{i,i+1}^{Rx}]$) 相关联。此外, 从式 (29) 也可以看出, 等待时延 $E[W_{i,i+1}]$ 的大小也

与 $E[\theta_{i,i+1}^{Tx}]$ 和 $E[\theta_{i,i+1}^{Rx}]$ 有关。由此可见, $N_{i,i+1}$ 的不同取值, 直接影响了无线链路中 IPv6 包分片的成功传输概率, 进而影响到每一跳的平均传输时延 $E[W_{i,i+1}] + E[T_{i,i+1}]$ 。直观上, 可以通过优化 $N_{i,i+1}$ 这个参数, 使得 $E[W_{i,i+1}] + E[T_{i,i+1}]$ 取最小值。于是, 可得到以下优化方程:

$$\begin{aligned} & \min E[W_{i,i+1}] + E[T_{i,i+1}] \\ & \text{w. r. t. } N_{i,i+1} \\ & \text{s. t. } \begin{cases} N_{i,i+1} \in \{m_0, m_0 + 1, m_0 + 2, \dots, m_1\} \\ i = 0, 1, \dots, n-1 \end{cases} \end{aligned} \quad (30)$$

式(30)中, $N_{i,i+1}$ 的取值范围可由以下分析得出。在 IEEE 802.15.4 标准中, 数据包头部和尾部占了 25 Byte, 加上安全选项 21 Byte, 共占用 46 Byte (称为“基本开销”)。另外, 6LoWPAN 协议头部和编码信息需占用 4 Byte。因此, 数据包的有效载荷长度的最大值为 $127 - 46 - 4 = 77$ Byte, 即 $l_{i,i+1}$ 的上界为 77 Byte, 可得到 $N_{i,i+1}$ 的下界为 $m_0 = \lceil L/77 \rceil$; 另外, 为了提高数据传输的效率, 不妨令数据包的有效载荷不少于数据包的“基本开销”, 于是, 可得到 $N_{i,i+1}$ 的上界为 $m_1 = \lceil L/46 \rceil$ 。由此可知, $N_{i,i+1} \in \{m_0, m_0 + 1, m_0 + 2, \dots, m_1\}$ 。

考虑到 46~77 之间只有 32 个整数, 那么, 在 m_0 和 m_1 之间最多也只有 32 个整数, 采用穷举法解式(30)即可求出使目标函数 $E[W_{i,i+1}] + E[T_{i,i+1}]$ 取最小值的最优分片个数 $N_{i,i+1}$ 。求解算法如下:

算法 1 Finding_Optimal_N

```
1 Initialize parameters D, Land LACK, etc. ;//初始化通信
//半径 D, L 和 LACK 等参数值
2 SetTime = ∞; u = 0; Q = {m0, m0 + 1, m0 + 2, ..., m1};
3 for (y = 1; y ≤ |Q|; y++) do// |Q| 表示集合
//Q 的元素个数
4 Ni,i+1 = Q(y); //Q(y) 表示返回集合 Q 中第 y 个元素
5 Compute E[Wi,i+1] according to (29); //计算等
//待时延
6 Compute E[Ti,i+1] according to (17); //计算传
//输时延
7 u = E[Wi,i+1] + E[Ti,i+1];
8 if u < Time
9 Time = u;
10 Save Ni,i+1 and E[Wi,i+1];
11 end if
12 end for
13 Return Ni,i+1 and E[Wi,i+1]; //返回最优分片个数
//和等待时延
```

算法的时间复杂性主要由循环语句(第 3 行~第 12 行)决定, 共有 $|Q|$ 次循环。因此, 算法的时间复杂度为 $O(|Q|)$ 。这个时间复杂度对于传感器节点来说, 是可以接受的。循环语句(第 3 行~第 12 行)实现了对 IPv6 包最优分片个数 $N_{i,i+1}$ 的穷举搜索。

算法 Finding_Optimal_N 在“A_i的操作流程”步骤 4 中运行, $i = 1, 2, \dots, n-1$ (A_n 是目的节点, 不用再往下一跳发送 IPv6 包, 因此, 无需运行该算法), 运行结束后所返回的 $N_{i,i+1}$ 和 $E[W_{i,i+1}]$ 保存在节点内存中, 供 IPDS-DF 方案执行的过程中调用。在 $i=0$ 时, 算法则在“A₀的操作流程”步骤 1 执行。

5 仿真实验与性能分析

为了验证 IPDS-DF 的性能, 采用 Matlab R2012b 进行仿真实验。实验基本参数设置如下: 节点的通信半径 $D = 50$ m, 每一跳所采用的数据包长度由式(30)和式(6)计算得出, 式(6)中的头部和尾部的总长度为 $H + F = 59$ Byte。IPv6 包的长度 $L = 1\,300$ Byte, ACK 包的长度 $L_{ACK} = 5$ Byte。

物理层参数设置如下^[13]: $P_t = 0$ dBm, $B_N = 30$ kHz, $P_{\text{lost}}(d_0) = 55$ dB, $\eta = 2$, $P_{\text{th}} = -95$ dBm。采用 O-QPSK-A 调制技术, 数据速率 $R = 50$ kb/s, $SIFS = 0.003\,84$ s, $LIFS = 0.012\,8$ s。

另外, 节点在发送数据包之前必须先竞争信道(发送 ACK 包则不需要竞争信道)。在竞争信道时, 节点需在集合 $\{0, 1, \dots, 2^{BE} - 1\}$ 中随机选择一个退避计数器(Backoff Counter, BC), 其中, BE 是退避指数, 然后, 节点退避 BC 个时隙, 每个时隙长度为 UB (单位退避时隙长度)个符号周期。当信道空闲时, 节点的平均退避时间为 $(2^{BE} - 1)/2$ 个时隙。文献[14]指出, 每个节点的平均退避次数接近于 1, 因此, 节点竞争信道的平均时间 $\sigma_c = (2^{BE} - 1)/2 \times UB + CCA$, 其中, CCA 是评估信道是否空闲的时间。由于 O-QPSK-A 调制方式的符号周期为 $320\,\mu\text{s}$, CCA 为 4 个符号周期, UB 为 20 个符号周期, $BE = 3$, 因此可得到 $\sigma_c = (2^3 - 1)/2 \times 20 \times 320 + 4 \times 320 = 0.023\,68$ s。

设置节点的能量捕获速率为 $\lambda_i = 0.005$ J/s^[1], 节点初始能量 $e_0 = 0$ 。相比于通信能耗, 节点的计算能耗可忽略不计, 可设置 $\delta = 0$ 。此外, 设置 $\varepsilon_0 = 50$ nJ/bit, $\varepsilon_1 = 10$ pJ/(bit · m²), $\beta = 0.9$ 和 $\rho = 0.000\,01$ J/s。

本文采用与文献[4, 15]相同的编码方法, 即 Reed-Solomon 编码。实验的思路如下: 在 $1\,000\,\text{m} \times 1\,000\,\text{m}$ 的方形区域中, 分别随机生成路径长度为 2 跳, 4 跳, 6 跳, ..., 20 跳的传输路径, 相邻节点之间的物理距离(单位为 m)在区间 $[1, 50]$ 随机选取。路径生成之后, 从源节点发送 100 个 IPv6 包到目的节点, 并逐跳统计传输时延、传输能耗和传输成功概率, 当第 100 个 IPv6 包在路径上传输结束(第 100 个 IPv6 包被目的节点接收到的时刻或者第 100 个 IPv6 包在传输路径中出现丢包的时刻, 即为传输结束的时刻)时, 统计出路径的 PDR。对以上实验重复做 100 次, 并对所得到的结果取平均值。

本文首先取编码冗余度系数 $\alpha = 1.5$, 对本文所

推导的传输时延公式(即式(17)和式(29))、能耗公式(即式(18)和式(19))和 PDR 公式(即式(20))进行验证。实验结果如图 4~图 6 所示。

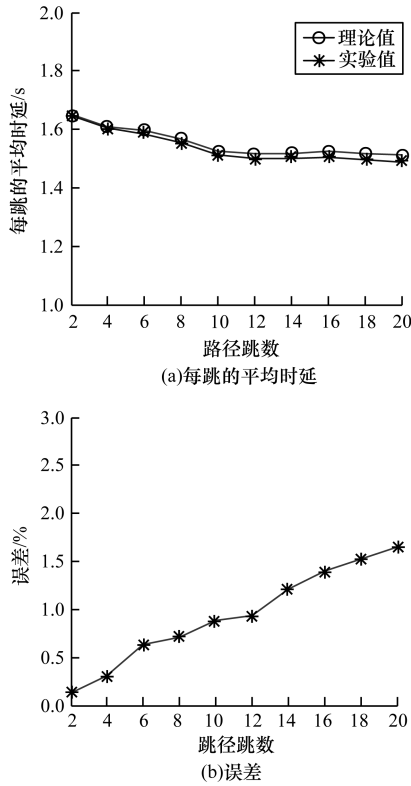


图 4 每跳的平均传输时延及误差

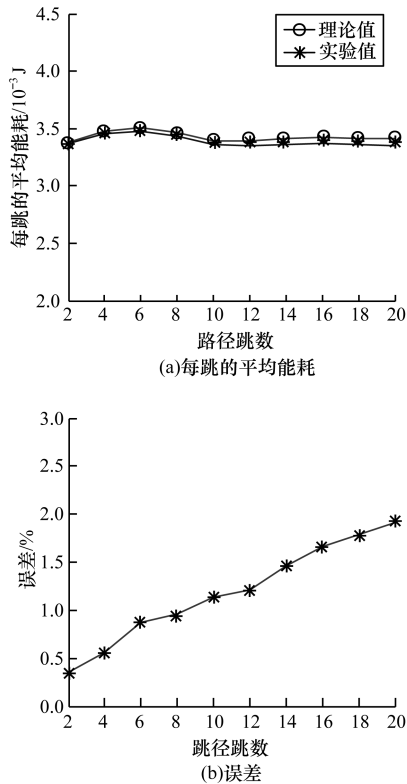


图 5 每跳的平均能耗及误差

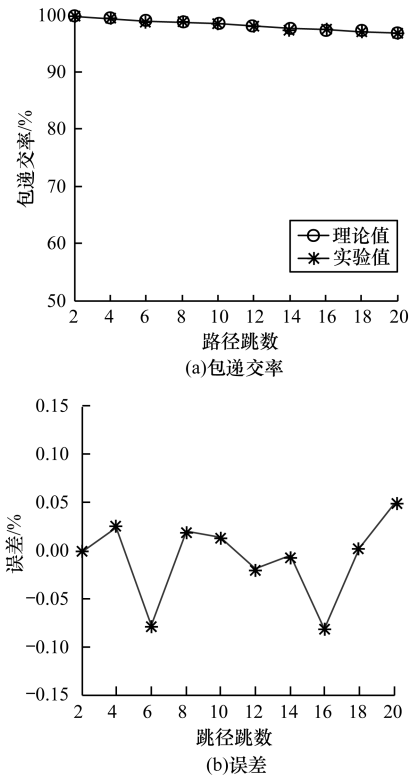


图 6 路径的 PDR 及误差

从图 4~图 6 可以看出,每一跳的平均传输时延、平均能耗和路径的 PDR 的实验结果和理论计算结果非常接近,误差均小于 2%。于是,本文所推导的公式得到了验证。

其次,保持编码冗余度系数 α 不变,对 IPDS-DF 方案与文献[4]所提出的 RIPDS (Reliable IPv6 Packet Delivery Scheme) 方案进行性能对比与分析。RIPDS 是仅适用于传统 WSN 的 IPv6 包传递方案,不能直接应用于 PSN。在 RIPDS 方案中引入本文所定义的“等待时延”,使其适用于 PSN。RIPDS 方案最主要的特征是它在发送 IPv6 包时采用固定的分片个数,根据文献[4],RIPDS 方案所采用的 IPv6 包的分片个数在 26 以内,限于篇幅,不妨对 RIPDS 方案分别取分片个数为 21 和 26 (分别记为 RIPDS ($N=21$) 和 RIPDS ($N=26$)) 与本文方案 IPDS-DF 进行性能对比。实验结果如图 7~图 9 所示。

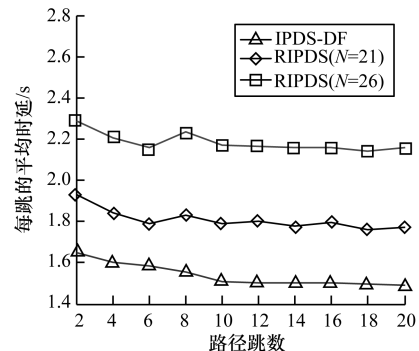


图 7 每跳的平均时延对比

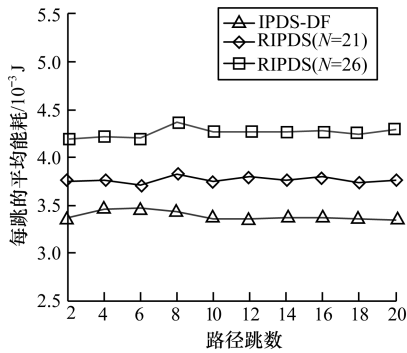


图 8 每跳的平均能耗对比

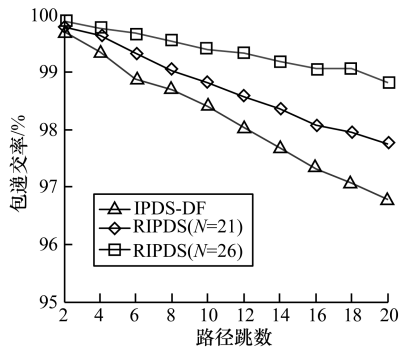


图 9 路径的 PDR 对比

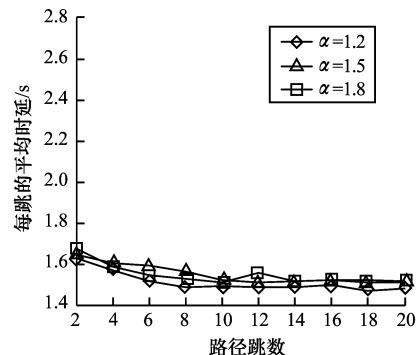
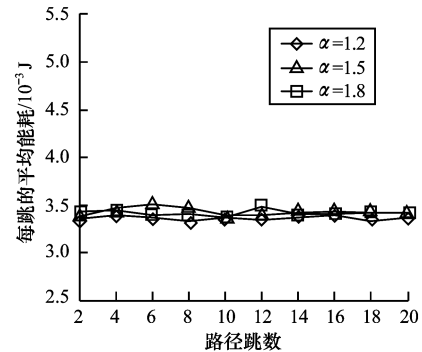
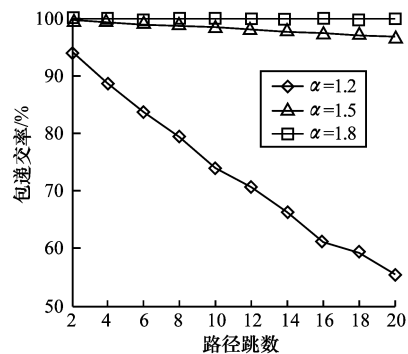
由图 7~图 8 的数据统计得出,IPDS-DF, RIPDS ($N=21$) 和 RIPDS ($N=26$) 每跳的平均传输时延分别为 1.56 s, 1.84 s 和 2.23 s; 每跳的平均能耗分别为 3.39×10^{-3} J, 3.79×10^{-3} J 和 4.31×10^{-3} J。IPDS-DF 平均每跳的传输时延分别比 RIPDS ($N=21$) 和 RIPDS ($N=26$) 降低了 15.45% 和 30.20%; 在每跳的平均能耗方面, 则分别降低了 10.79% 和 21.38%。

通过图 7 和图 8 可以看到,IPDS-DF 降低了传输时延 (见图 7) 的同时, 也降低了传输能耗 (见图 8)。这是由于 IPDS-DF 是以最小化每一跳的传输时延为目标的, 优化方程式 (30) 总是趋向于最小化发送次数, 即分片个数尽可能少, 而当分片个数较少的情况下, 编码包的个数也较少, 发送能耗自然就少了。值得注意的是, 如果分片个数太少, 每个分片的长度会较长, 根据式 (7) 和式 (8), 每个编码包的成功传输概率会下降且编码包的个数反而增多, 这势必会造成时延和能耗增多。

从图 9 的数据统计得出, IPDS-DF, RIPDS ($N=21$) 和 RIPDS ($N=26$) 的平均 PDR 分别为 98.90%, 99.17% 和 99.59%。IPDS-DF 分别比 RIPDS ($N=21$) 和 RIPDS ($N=26$) 降低 0.27% 和 0.69%。根据上文分析, 优化方程式 (30) 总是趋向于使分片的长度尽可能长, 根据式 (7)、式 (8) 和

式 (20), PDR 会随之减少。这与直观感觉是相吻合的。

最后, 考察编码冗余度系数 α 的不同取值对 IPDS-DF 性能的影响。当 α 分别取 1.2, 1.5 和 1.8 时的实验结果如图 10 所示。

(a) α 对传输时延的影响(b) α 对传输能耗的影响(c) α 对 PDR 的影响图 10 α 对 IPDS-DF 性能的影响

从图 10(a)、图 10(b) 中可以看出, α 以步长 0.3 从 1.2 变化到 1.8 时对每一跳的平均传输时延和平均传输能耗的影响是不大的 (3 条数据曲线非常接近), 原因如下: 根据式 (8), α 的取值只影响到编码包的数量, α 取值越大, 编码包的数量越多 (占用节点的内存也越多), 接收节点能够成功接收到 IPv6 包的概率也越大, 反之亦然, 也即 α 的取值影响了路径的 PDR, 但对时延及能耗的影响甚微; 从图 10(c) 可以看出, PDR 随 α 增加而增加, 当 $\alpha = 1.8$ 时,

IPDS-DF 的 PDR 接近于 1,此时,IPv6 包传输的可靠性得到了保证。在实际工程项目的实施过程中,在传感器节点内存空间较大的情况下,可考虑设置编码冗余度系数 $\alpha \geq 1.8$ 。

6 结束语

在物联网中,IPv6 网络与 PSN 的数据交换具有广泛的应用,低时延且可靠地传递 IPv6 包至 PSN 中的特定节点,是 IPv6 网络与 PSN 数据交换的重要形式。本文提出一种采用动态分片的 IPv6 包传递方案 IPDS-DF,从仿真实验的结果和性能分析可以看出,IPDS-DF 方案是可行和有效的。在 PSN 中,节点从环境中所捕获到的能量会随着能量捕获条件的变化而出现波动,如何根据能量捕获条件的变化对节点的数据传输策略进行自适应调整,将是下一步研究的方向。

参考文献

- [1] Seah W K G, Zhi Ang, Tan H. Wireless Sensor Networks Powered by Ambient Energy Harvesting (WSN-HEAP)-Survey and Challenges [C]//Proceedings of the 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology. Aalborg, Denmark: [s. n.], 2009.
- [2] 应 岚,林一民,池凯凯. 能量捕获无线传感器网络能耗感知的路由方案[J]. 计算机工程, 2016, 42(2): 108-112.
- [3] 黄 媛. 无线传感器网络中一种基于可靠性的数据收集算法[J]. 计算机工程, 2015, 41(2): 85-90, 95.
- [4] Zhu Yihua, Chi Kaikai, Tian Xianzhong, et al. Network Coding Based Reliable IPv6 Packet Delivery over IEEE 802.15.4 Wireless Personal Area Networks[J]. IEEE Transactions on Vehicular Technology, 2016, 65(4): 2219-2230.
- [5] Wicker S B, Bhargava V K. Reed-Solomon Codes and Their Applications; An Introduction to Reed-Solomon Codes[M]. New Jersey, USA: Willy Inc., 1999.
- [6] Srouji M S, Wang Zhonglei, Henkel J. RDTS: A Reliable Erasure-coding Based Data Transfer Scheme for Wireless Sensor Networks [C]//Proceedings of the 17th IEEE International Conference on Parallel and Distributed Systems. Washington D. C., USA: IEEE Press, 2011.
- [7] James A, Madhukumar A S, Kurniawan E. Performance Analysis of Fountain Codes in Multihop Relay Networks[J]. IEEE Transactions on Vehicular Technology, 2013, 62(9): 4379-4391.
- [8] Kansal A, Hsu J, Zahedi S, et al. Power Management in Energy Harvesting Sensor Networks [J]. ACM Transactions on Embedded Computing Systems, 2007, 6(4): 1-35.
- [9] Pioro J R, Bergonzini C, Atienza D, et al. Prediction and Management in Energy Harvested Wireless Sensor Nodes [C]//Proceedings of the 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology. Aalborg, Denmark: [s. n.], 2009.
- [10] Ventura J, Chowdhury K. Markov Modeling of Energy Harvesting Body Sensor Networks [C]//Proceedings of the 22nd International Symposium on Personal Indoor and Mobile Radio Communications. Toronto, Canada: [s. n.], 2011.
- [11] Landolsi M A, Stark W E. On the Accuracy of Gaussian Approximations in the Error Analysis of DS-CDMA with OQPSK Modulation [J]. IEEE Transactions on Communications, 2002, 50(12): 2064-2071.
- [12] Heinzelman W B, Chandrakasan A P, Balakrishnan H. An Application-specific Protocol Architecture for Wireless Microsensor Networks [J]. IEEE Transactions on Wireless Communications, 2002, 1(4): 660-670.
- [13] 李燕君,邵剑集,雷艳静. 传感器网络利用效用模型的跨层优化路由策略[J]. 软件学报, 2013, 24(S1): 117-124.
- [14] Timmons N F, Scanlon W G. Analysis of the Performance of IEEE 802.15.4 for Medical Sensor Body Area Networking [C]//Proceedings of the 1st Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks. Washington D. C., USA: IEEE Press, 2004: 16-24.
- [15] 朱艺华,徐 骥,田贤忠,等. 无线传感器网络应用简单 Reed-Solomon 编码的低能耗和低时延可靠数据收集方案[J]. 计算机学报, 2015, 38(10): 2106-2124.

编辑 刘 冰 陆燕菲