

文章编号: 2095-4980(2020)04-0723-06

一种基于负载均衡的高能效 LLN 路由算法

秦晓琴, 李 永, 李 巨

(重庆广播电视大学 教务处, 重庆 400052)

摘 要: 针对低功耗有损网络(LLN)中由于节点部署不均匀易导致负载不均衡的问题, 提出一种基于负载均衡的高能效LLN路由算法(EELB-RPL)。通过将链路质量、传输时延、节点剩余能量、吞吐量以及拥塞检测因子等因素相结合选出最优父节点; 再通过调整溪流计时器, 使节点根据网络拓扑密度调整自身抑制机制, 避免了抑制不公平性导致负载不均衡。仿真结果表明, 所提算法与现有算法相比较, 能够有效实现负载均衡。其中, 数据包投递率提升了14.6%, 根节点吞吐量提升了28.5%, 网络平均寿命提升了8.96%。

关键词: 低功耗有损网络; 路由算法; 负载均衡; 溪流计时器

中图分类号: TN915; TP393

文献标志码: A

doi: 10.11805/TKYDA2019105

An Energy-Efficient LLN routing algorithm based on Load Balancing

QIN Xiaoqin, LI Yong, LI Ju

(Academic Affairs Office, Chongqing Radio & TV University, Chongqing 400052, China)

Abstract: Aiming for the load unbalance problem in Low Power and Lossy Networks(LLN) due to uneven node deployment, an Energy-Efficient Routing Protocol for LLN based on Load Balancing(EELB-RPL) is proposed. This protocol has made the following improvements: the optimal parent node is selected by comprehensive examination of the link quality, transmission delay, node residual energy, throughput and congestion detection factor. By adjusting the Trickle timer, the nodes adjust their suppression mechanism according to the network topology density, so as to avoid the suppression unfairness and load unbalance. The simulation results show that compared with the existing algorithms, the EELB-RPL can effectively achieve load balancing where the average packet delivery rate is increased by 14.6%, the throughput of sink node is increased by 28.5% and the network lifetime is increased by 8.96%.

Keywords: low power and lossy network; routing protocol; load balancing; Trickle timer

低功耗有损网络(LLN)^[1-2]通常指由众多能量、存储能力以及处理能力受限的无线传感器节点组成的网络。LLN 网络在物联网领域^[3-4]起着至关重要的作用。因此, LLN 具有广阔的发展前景。国际互联网任务工程组在 2010 年制定了一种基于 IPv6 的低功耗有损网络路由算法(Routing Protocol for LLN, RPL)^[5-6]。基于 RPL 算法构建的拓扑中传感器节点均需要定期广播面向目的地有向无环图(Destination Oriented Directed Acyclic Graph, DODAG)信息对象消息(DODAG Information Object, DIO), 该消息主要用于网络拓扑的构建以及维护结构的稳定性, 且该消息的发送周期间隔由溪流计时器^[6]控制, 溪流计时器的基本原理是为受限制的网络工作节点提供一个本地的、健壮的、高效的、简单的以及可伸缩的信息交换算法。近年来, 学者们对 RPL 进行了深入的研究。文献[6]在网络构建时仅将传输跳数作为路由度量, 但是却没有考虑无线链路的质量, 导致网络丢包率高。文献[7]通过使用期望传输次数来进行组网, 对无线链路的可靠性进行了考量, 但却忽略了网络传输的实时性以及节点的能耗速率, 可能会导致能耗较高的节点被选做父节点, 造成节点能耗持续增大, 降低节点生命周期。文献[8]在 RPL 路由算法的基础上结合模糊函数的思想, 对节点的能耗、节点通信距离进行综合考量, 但是该方案的计算量较大而不适用于大规模场景。文献[9]提出了一种均衡能量瓶颈节点能耗速率的 RPL 路由算法。在该算法中, 新节点需要计

收稿日期: 2019-03-28; 修回日期: 2019-05-05

基金项目: 重庆市教育委员会科学技术研究计划资助项目(KJQN201804006)

作者简介: 秦晓琴(1989-), 女, 硕士, 讲师, 研究方向为无线传感网、RPL 路由协议。email:805634967@qq.com

算对每条路径中能量瓶颈节点的期望寿命影响大小,该算法能够在一定程度上均衡各条路径中能量瓶颈节点的能耗速率,但效果不是很显著。针对文献[7]和文献[9]存在的问题,文献[10]采用多维度量构建网络的拓扑,但是未考虑随着网络深度的增加,上游节点需要传输的数据量增大,导致能耗过高。文献[11]首先考虑了冗余常数 K 对网络性能的影响,其结果表明,如果 K 值配置不当,溪流计时器抑制机制可能导致次优路由,作者通过对长时间没有广播的节点进行优先级排序来消除这种问题。文献[12]对冗余常数 K 和 I_{\min} 对 RPL 算法性能的影响进行了广泛的仿真研究,他们观察到,影响拓扑构造最大的参数之一是冗余常数值 K 。

通过上述研究发现,在现有 RPL 算法中存在着以下问题: a) 在拓扑的构建过程中,节点在选择最优父节点时采用的路由度量判据过于单一从而导致节点所选择的父节点并非最优以及未对处于上游的中继节点路由度量权重进行自适应调整; b) 当网络发生结构较大变化时,溪流计时器维护网络拓扑稳定的时间过长; c) 尽管溪流计时器机制已被证明是有效的,但最近的工作表明^[11-12],就负载分布而言该机制本质上是不公平的,其性能严重依赖于网络拓扑,同时,冗余常数 K 是其最重要的参数之一,该值的设置不应该独立于网络密度。因此,为了解决上述问题,提出一种基于负载均衡的高能效 LLN 路由算法,并对该算法进行了详细的阐述及仿真对比。

1 RPL 拓扑构建原理

在基于 RPL 路由算法的 DODAG 构建中,主要运用了 RPL 定义的 3 种控制消息: a) DIO 控制消息,其主要用于上行路由的构建; b) DODAG 目的广播对象(DODAG Destination Advertisement Object, DAO),该消息主要用于下行路由的构建; c) 面向 DODAG 请求消息(DODAG Information Solicitation, DIS),该消息主要用于消息请求,即新节点可以广播该消息助其加入拓扑中,图 1 为 DODAG 的构建过程。DODAG 拓扑构建步骤具体描述如下:

- 1) DODAG 根节点广播 DIO 控制消息(DIO 消息中包含一个网络深度 Rank 值,根节点的 Rank 为 0)。
 - 2) 邻居节点 a 收到 DIO 控制消息后,将根节点选作父节点,且将包含自身路由信息的 DAO 消息单播至根节点,随后更新自身信息后广播 DIO 消息。
 - 3) 同理,节点 b 收到节点 a 广播的 DIO 消息后,重复节点 a 的操作。
 - 4) 节点 c 在等待一段时间(等待时间可根据网络实际特征进行设置,本文 DIS 消息的等待时间为 2^{12} ms)后未收到邻居节点广播的 DIO 消息,则节点 c 向周围广播 DIS 控制消息,请求加入 DODAG 中。
 - 5) 邻居节点 b 收到 DIS 消息后,向节点 c 单播一个 DIO 消息,节点 c 收到 DIO 控制消息后根据该消息中所携带的路由度量信息来选择是否选择发送端作为父节点,若加入,则回复一个 DAO 控制消息;反之,不予处理。
 - 6) 节点 b 收到节点 c 的 DAO 消息后,将该消息转发至根节点,至此整个 DODAG 的拓扑构建完成。
- 通常,节点会同时收到来自多个邻居节点广播的 DIO 消息。在收到多个 DIO 消息后,将消息的发送节点添加到备选父节点集中。随后解析出每个 DIO 消息所携带的信息计算出路由度量,选择路由度量值最小的备选父节点作为最优父节点。

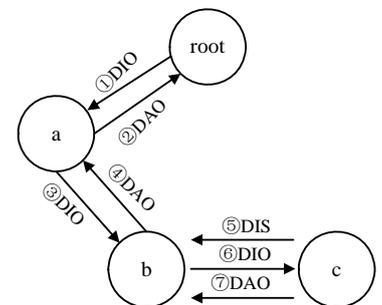


Fig.1 Build process of DODAG
图 1 DODAG 构建过程

2 EELB-RPL 算法

针对现有算法在选择最优父节点时采用的路由度量判据过于单一,从而导致节点所选择的最优父节点并非最优,且溪流计时器的不公平抑制性导致无法有效实现负载均衡和能效低等问题,EELB-RPL 主要作了以下优化: a) 在选择父节点时,结合多种路由因素来避免单一路由因素对网络性能带来影响。b) 当网络发生不一致时,对溪流计时器的时间参数进行调整,使网络能快速收敛稳定从而降低时延。c) 对溪流计时器中的参数 K 进行自适应调整,使节点根据网络拓扑调整各自的抑制机制来适应网络密度。

2.1 最优父节点选择

在网络拓扑构建时,当节点接收到邻居节点广播的 DIO 消息后,结合多种路由因素计算其与邻居节点的度量值。下面以节点 d 为例,综合多种因素的具体步骤如下:

步骤 1: 根据节点 d 的当前剩余能量计算期望寿命^[9],该值有利于子节点避免选择剩余能量过低的邻居节点作为最优父节点。期望寿命计算过程如下:

1) 计算节点 d 在单位时间内的需要传输的数据流量总和 T_d ，见式(1)， $T_{\text{gen}}(d)$ 为节点 d 产生数据包的速率； T_i 为其子节点 i 的发包速率。

$$T_d = T_{\text{gen}}(d) + \sum_{i \in \text{children}} T_i \quad (1)$$

2) 计算节点 d 在单位时间内，成功将数据包传输至其最优父节点 a 的大致传输次数，见式(2)， $ETX(d,a)$ 为节点 d 成功发送单个数据包到其父节点 a 大致所需的期望传输次数， ETX 能够大致反映出无线链路的质量。

$$M_d = T_d \times ETX(d,a) \quad (2)$$

期望传输次数 ETX 由前向转发成功率 d_f 与反向 ACK 确认成功率 d_r 计算得到^[13]。这 2 个值可根据节点广播固定大小，设定发送周期为 τ 的探测包计算。期望传输次数计算公式为：

$$ETX = \frac{1}{d_f \times d_r} \quad (3)$$

3) 计算节点 d 将数据包传输至其最优父节点 a 所需的时间，见式(4)：

$$t = \frac{M_d}{r_{\text{data}}} \quad (4)$$

式中 r_{data} 为节点 d 的数据包传输速率。该公式将同时考虑链路质量、吞吐量、数据传输速率进行汇聚。

4) 计算节点 d 在单位时间内将数据包成功发送至其最优父节点 a 的能量消耗，见式(5)：

$$E_r(d) = t \times P_{\text{tx}}(d) \quad (5)$$

式中 $P_{\text{tx}}(d)$ 为节点 d 的数据包发送功率。

5) 根据节点 d 的剩余能量以及能耗速率计算出节点的期望寿命 ELT ，见式(6)， $E_{\text{res}}(d)$ 为节点 d 剩余能量。

$$ELT(d) = \frac{E_{\text{res}}(d)}{E_r(d)} \quad (6)$$

步骤 2 计算节点 d 的拥塞因子，节点的队列利用率大致能反映出节点的负载程度，因此本文以队列利用率来反映节点的拥塞因素，见式(7)：

$$CF = \frac{\gamma_{\text{now}}}{\gamma_{\text{max}}} \quad (7)$$

式中： γ_{now} 为节点 d 当前队列缓存占用的大小； γ_{max} 为节点 d 的最大队列缓存空间。

步骤 3：节点 d 将其期望寿命、拥塞因子等相关信息添加到 DIO 控制消息的选项字段中，并且向邻居节点广播该消息。

步骤 4：邻居节点收到节点 d 发送的 DIO 控制消息后，获取消息中携带的相关信息。计算自身与节点 d 的路由度量值(metric)，度量值越大，说明两节点直接的传输代价越大；反之，越小。见式(8)。

$$\text{metric} = \psi [h(d)+1] + (1-\psi) \frac{CF}{ELT} \quad (8)$$

式中： $h(d)$ 为节点 d 到根节点的传输跳数，该值能大致反映传输时延；阈值 ψ 则根据节点的当前传输跳数进行设定，其意义在于：若节点距离根节点相对越近，需要转发的数据流量就越大，则需要对该节点的缓存利用率以及剩余能量进行加权处理。 ψ 的计算见式(9)， h_{now} 为节点自身的传输跳数， h_{max} 为网络允许的最大网络深度。

$$\psi = \frac{h_{\text{now}}}{h_{\text{max}}} \quad (9)$$

2.2 溪流计时器公平抑制策略

溪流计时器是 RPL 路由算法的重要组成部分，主要目的是调节网络中 DIO 控制消息的发送。当检测到网络出现不一致(网络结构发生较大变化)时，溪流计时器通常会被重置，加快 DIO 控制消息的发送，使网络尽快稳定；而当网络拓扑结构较为稳定时，DIO 控制消息的发送时间间隔将会逐渐增大，从而减小节点能量的消耗。其工作原理如下：

步骤 1：初始化网络，当前发送间隔 I 取 $[I_{\text{min}}, I_{\text{max}}]$ 间任意值， I_{min} 为最小发送间隔， I_{max} 为最大发送间隔，由第 1 个时隙开始。

步骤 2：设置冗余计时器 $C=0$ ，时间值 t 取 $[I/2, I]$ 间任意值。

步骤 3：判断接收到的消息是否一致，Trickle 定时器每收到一个表示网络一致的 DIO 控制消息，则将计数 C 加 1，进行步骤 4；接到的消息不一致时，直接跳到步骤 6。

步骤 4: 当时间到达 t 时, 比较冗余常量 K 与计数值 C 的大小。如果 $C > K$, 则禁止节点向邻居节点发送 DIO 控制消息; 反之, 则允许发送。

步骤 5: 当前时间间隔 I 结束后, 令 $I = 2 \times I$, 若 $I_{\max} < 2 \times I$, 则将 I 值设置为 I_{\max} 。

步骤 6: 一旦接收到表示网络不一致的 DIO 控制消息, 重置 Trickle 定时器, 令 $I = I_{\min}$, 重复步骤 2 直至网络时间运行结束。

但是, 经过研究分析发现, 溪流计时器也存在着不足: a) 当检测网络发生不一致时, 若时间 t 在 $[I/2, I]$ 区间内取值, 可能会导致维护网络稳定的时间较长, 造成网络时延高; b) 在稀疏网络中, 节点的邻居节点数量较少, 为了使抑制机制发挥作用, K 值应该设置为一个较低的值, 而在密集的网络中, 较高的 K 值更具意义。然而 LLN 网络拓扑的节点部署不均匀, 部分区域节点密集, 部分区域节点稀疏, 网络的异构性会导致 K 值的设置困难。

因此针对上述问题, 提出一种溪流计时器公平抑制策略, 图 2 为所提策略的流程图, 具体操作步骤描述如下:

步骤 1: 初始化网络, 发送间隔 I 取 $[I_{\min}, I_{\max}]$ 间任意值, 由第 1 个时隙开始。

步骤 2: 设置计数值 $C = 0$, 时间值 t 取 $[I/2, I]$ 间任意值。

步骤 3: 判断接收到的消息是否一致, Trickle 定时器每收到一个表示网络一致的 DIO 控制消息, 则将计数值 C 加 1, 进行步骤 4; 接收到不一致消息时, 直接跳到步骤 6。

步骤 4: 当时间到达 t 时, 比较冗余常量 K 与计数值 C 的大小。如果 $C > K$, 则禁止节点向邻居节点发送 DIO 控制消息; 反之, 则允许发送。

步骤 5: 通过对溪流计时器的分析可知, 节点通过计数值 C 记录在一个时间间隔内接收 DIO 消息的数量, 该值可以大致反映出邻居节点的数量。所以当节点的计时器到达时间 I 时, 引入一个关于 C 的函数 $F(C)$, $F(C)$ 的表达式如下:

$$F(C) = \begin{cases} K_{\min} & aC < K_{\min} \\ \lfloor aC \rfloor & K_{\min} \leq aC \leq K_{\max} \\ K_{\max} & aC > K_{\max} \end{cases} \quad (10)$$

式中: $a \in [0, 1]$, $K_{\min} \in \mathbb{N}$, $K_{\max} \in \mathbb{N}$ 。因此, 当前时间间隔 I 结束后, 令 $I = 2 \times I$, 若 $I_{\max} < 2 \times I$, 则将 I 值设置为 I_{\max} , 节点根据间隔内 C 值的大小, 按照式(10)对 K 值进行自适应设置。

步骤 6: 一旦接收到表示网络不一致的 DIO 控制消息, 重置 Trickle 定时器, 令 $I = I_{\min}$, 令 $C = 0$, $t = [0, I]$ 后重复步骤 3 直至网络时间运行结束。将时间值 t 的取值区间进行更改, 原因在于当网络发生不一致时, 计时器可以更快地解决不一致, 同时不会带来额外的成本开销。

3 仿真及结果分析

采用 Contiki2.7^[14-15] 仿真软件, 对 EELB-RPL 各方面网络性能进行模拟仿真验证, 在相同模拟仿真场景下, 选取文献[7]和文献[9]进行对比分析。

3.1 仿真环境及参数

在 $200 \text{ m} \times 200 \text{ m}$ 的仿真区域内分别构建不同网络规模的仿真场景, 网络规模大小分别为 10, 30, 50, 70, 90 和 110, 且每种场景中根节点数量均为 1, 节点通信半径为 40 m, 仿真周期为 3 600 s。此外, 为了确保网络的稳定性和可靠性, 网络中所有节点均采用静态或准静态模型。仿真中用到的其他主要参数设置见表 1。

3.2 性能指标

1) 数据包投递率(Packet Delivery Ratio, PDR): 节点成功传输至根节点的数据包与节点发送的数据包总数的比值为:

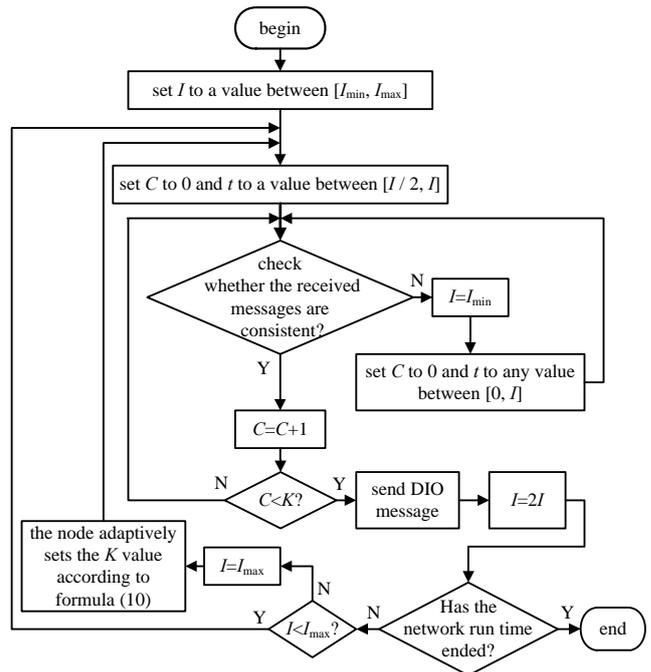


Fig.2 Flow chart of Trickle timer adjustment

图 2 溪流计时器公平抑制策略流程图

$$H_{rate} = \frac{\sum_{i=1}^n G_i}{\sum_{i=1}^n R_i} \tag{11}$$

式中： G_i 为成功到达根节点的数据包数； R_i 为节点发送的数据包总数。

表 1 主要参数
Table 1 Main parameters

RF model	data link	physical	node send packet rate(pkt/s)	packet size /bytes	node initial energy	channel shadow attenuation factor	a	K_{min}	K_{max}
UDGM(Unit Disk Graph Medium)	CSMA+ ContikiMAC	IEEE 802.15.4	2	127	10J	2	0.65	1	15

2) 根节点平均吞吐量：表示在单位时间内网络中成功传输的数据包的数量，是衡量 LLN 网络性能的一项重要指标，公式如下：

$$\bar{P} = \frac{K_s}{T} \tag{12}$$

式中： K_s 为成功到达根节点的数据包比特数； T 为网络运行时间。

3) 网络平均寿命：以网络中最先死亡的节点的生存时间作为网络平均寿命，其中死亡节点是指当前剩余能量仅为初始能量的 5% 的节点，公式如下：

$$D_t = d_{end} - d_{start} \tag{13}$$

式中： d_{start} 为网络开始运行的时间； d_{end} 为网络出现第一个死亡节点的时间。

3.3 仿真结果分析

3.3.1 数据包投递率

从图 3 可以看出，EELB-RPL 路由算法的 PDR 高于其他 2 种路由算法，经过分析主要原因有：a) 在网络拓扑构建过程中，EELB-RPL 路由算法综合考虑了期望传输次数、吞吐量以及负载因子等，能够减少由于无线链路的不稳定性以及缓存占用率过高导致的数据包丢失。b) EELB-RPL 算法通过对溪流计时器调整，使节点能够自适应地根据网络拓扑密度设置 K 值，从而使网络获得更优的路径，降低了由于次优路径引起的丢包率。

3.3.2 根节点吞吐量

图 4 表明，根节点的吞吐量随着节点数量的增加逐渐加大。但是，EELB-RPL 路由算法相较于其他两种路由算法，根节点的吞吐量明显要高。其原因主要在于：a) EELB-RPL 路由算法在构建拓扑结构时考虑了节点的吞吐量以及拥塞因子，能够最大化地实现网络负载，减少了数据包由于高负载的丢失；b) EELB-RPL 算法根据节点所处的网络深度对阈值进行自适应设置，延长了上游节点的生命周期，使其能够最大化地向上传输数据流量。最后，EELB-RPL 算法对溪流计时器进行了改进，节点可根据周围邻居节点的密度来设置 K 值以达到公平抑制 DIO 控制消息的目的，从而网络能够使节点之间均衡地分配消息负载，提高吞吐量。

3.3.3 网络平均寿命

从图 5 可知，网络平均寿命随着节点数量的增加逐渐缩短，这是由于网络中节点数量增加导致上游节点需要转发大量数据从而加剧能耗。但 EELB-RPL 路由算法相较于其他两种路由算法，网络的生存时间明显要长。其原因主要在于：首先，EELB-RPL 路由算法将节点剩余能量以及拥塞因子相结合，能够将剩余能量高以及未拥塞的节点选为最优

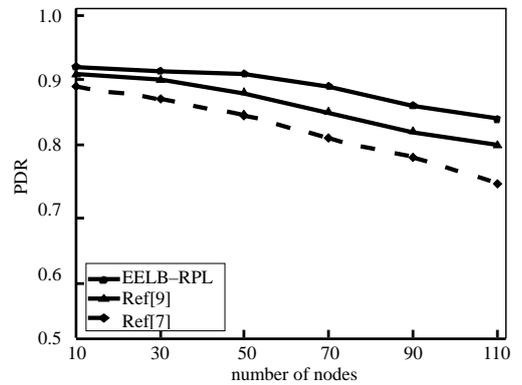


Fig.3 Packet delivery rate
图 3 数据包投递率

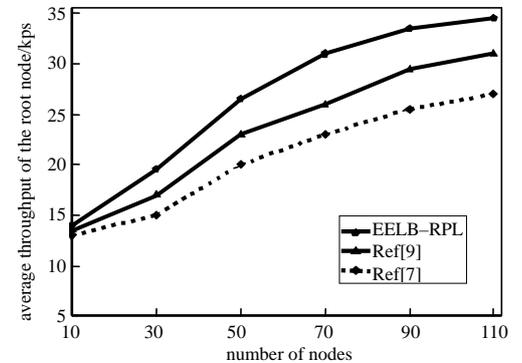


Fig.4 Average throughput of the root node
图 4 根节点平均吞吐量

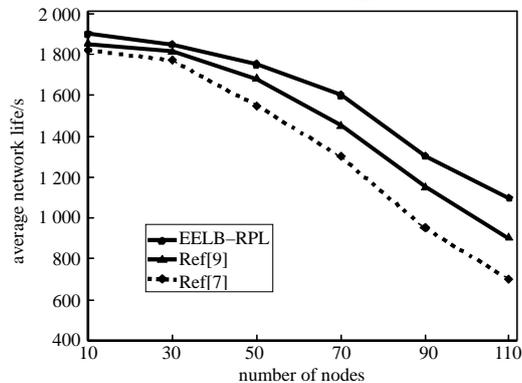


Fig.5 Average network life
图 5 网络平均寿命

父节点,从而延长网络寿命;其次,EELB-RPL 算法根据节点所处的网络深度进行阈值的设置,在距离 sink 节点越近时,节点的剩余能量和拥塞因子阈值越高,从而延长了网络寿命;最后,对溪流计时器进行了改进,自适应 K 值在保持较低的控制流量开销的同时,仍能发现较好的路由,避免了节点选择的路由质量过差导致高能耗。

4 结论

针对低功耗有损网络中现有路由算法在网络构建时不能较好地负载均衡,提出一种基于负载均衡的高能效 LLN 路由算法。在选择父节点时,通过将多种路由度量相结合,避免了单一的路由判据造成次优节点被选作最优父节点。为了避免抑制不公平性导致负载不均衡,通过对溪流计时器调整,使节点根据网络拓扑密度调整抑制机制。此外,虽然本文所提算法考虑的输入参数相较于现有算法有所增多,但所增加的计算参数为节点本身固有的参数,不需复杂计算得到。因此,本文所提算法不会明显增加复杂度。仿真结果表明,相较于现有算法,EELB-RPL 在网络平均寿命、数据包投递率以及根节点吞吐量几个方面均有所提升。未来将在移动的场景下进行 RPL 路由算法的负载均衡机制研究。

参考文献:

- [1] PAEK J. Fast and adaptive mesh access control in Low-Power and Lossy Networks[J]. IEEE Internet of Things Journal, 2015,2(5):435-444.
- [2] LIU X, SHENG Z, YIN C, et al. Performance analysis of Routing Protocol for Low power and Lossy Networks(RPL) in large scale networks[J]. IEEE Internet of Things Journal, 2017,4(6):2172-2185.
- [3] 邱继云,陈翔. 通过低功耗蓝牙串口启动的智能物联网网关[J]. 太赫兹科学与电子信息学报, 2018,16(4):719-723. (QIU Jiyun, CHEN Xiang. Bluetooth low energy UART booting based smart gateway for Internet of Things[J]. Journal of Terahertz Science and Electronic Information Technology, 2018,16(4):719-723.)
- [4] GHALEB B, AI-DUBAI A, ROMDHA I, et al. Drizzle: adaptive and fair route maintenance algorithm for Low-Power and Lossy Networks in IoT[C]// IEEE International Conference on Communications(ICC). Paris, France: IEEE, 2017:1-6.
- [5] RFC 6550. RPL: IPv6 routing protocol for Low-Power and Lossy Networks[S]. 2012.
- [6] RFC 6552. THUBERT P. Objective function zero for the routing protocol for low-power and lossy networks[S]. 2012.
- [7] RFC 6719. GNAWALI O, LEVIS P. The minimum rank with hysteresis objective function[S]. 2012.
- [8] 冯馨于,仇英辉. 基于剩余级别负载均衡的 RPL 路由协议[J]. 传感技术学报, 2018,31(4):607-612. (FENG Xinyu, QIU Yinghui. The RPL routing protocol based on load balance of remaining level[J]. Chinese Journal of Sensors and Actuators, 2018,31(4):607-612.)
- [9] 宋海龙,张书真. 基于期望寿命与均衡能量消耗的 RPL 路由协议[J]. 计算机工程, 2016,42(1):77-82. (SONG Hailong, ZHANG Shuzhen. RPL routing protocol based on expected lifetime and balance energy consumption[J]. Computer Engineering, 2016,42(1):77-82.)
- [10] 姚玉坤,刘江兵,任智,等. 集中式网络拥塞控制的高效 RPL 路由协议[J]. 系统工程与电子技术, 2017,39(12):2810-2816. (YAO Yukun, LIU Jiangbing, REN Zhi, et al. High-efficiency centralized network congestion control for RPL routing protocol[J]. Systems Engineering and Electronics, 2017,39(12):2810-2816.)
- [11] VALLATI C, MINGOZZI E. Trickle-F: fair broadcast suppression to improve energy-efficient route formation with the RPL routing protocol[C]// 2013 Sustainable Internet and ICT for Sustainability(SustainIT). Palermo: IEEE Press, 2013:1-9.
- [12] KERMAJANI H, GOMEZ C. On the network convergence process in RPL over IEEE 802.15.4 multihop networks: improvement and trade-offs[J]. Sensors, 2014,14(7):11993-12022.
- [13] AHMED A M, PAULUS R. Congestion detection technique for multipath routing and load balancing in WSN[J]. Wireless Networks, 2016,23(8):1-8.
- [14] KOVATSCH M, DUQUENNOY S, DUNKELS A. A low-power CoAP for contiki[C]// 2011 IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems. Valencia: IEEE Computer Society, 2011:855-860.
- [15] SHELAMI T R, SHAHRA E Q, SHAKSHUKI E M. Performance comparison of three localization protocols in WSN using Cooja[J]. Journal of Ambient Intelligence and Humanized Computing, 2017,8(3):1-10.