

# 6LoWPAN 中 route-over 分片转发策略的改进<sup>①</sup>

刘日仙<sup>1</sup>, 袁利永<sup>2</sup>

<sup>1</sup>(金华职业技术学院 信息工程学院, 金华 321017)

<sup>2</sup>(浙江师范大学 数理与信息工程学院, 金华 321004)

**摘要:** 6LoWPAN 使得在无线传感网络上进行基于 IPv6 的数据传输成为可能, 它定义了 route-over 和 mesh-under 两种路由策略. 针对传统的 route-over 策略在分片转发中存在的问题, 提出了基于虚电路的 route-over 分片流水转发策略, 详细介绍了该改进策略的详细设计, 即根据第 1 个分片的 IP 压缩头部信息建立后续分片的转发映射. 仿真实验结果表明, 改进的策略较原始策略在网络传输和时延上有更好的表现.

**关键词:** 6LoWPAN; route-over; 虚电路; 分片转发

## Improvement on Route-over Forwarding for IP Fragmented Packets in 6LoWPAN

LIU Ri-Xian<sup>1</sup>, YUAN Li-Yong<sup>2</sup>

<sup>1</sup>(College of Information Engineering, Jinhua Polytechnic College, Jinhua 321017, China)

<sup>2</sup>(College of Mathematics Physics and Information Engineering, Zhejiang Normal University, Jinhua 321004, China)

**Abstract:** 6LoWPAN enables the IPv6 communication over Wireless Sensor Networks, which defines two routing strategies: mesh-under if the layer concerned is adaptation layer and route-over if the network layer. To address problems in route-over forwarding for IP fragmented packets in 6LoWPAN, this paper proposes an improved route-over strategy based on virtual circuit which enables fragments forwarding in pipelined manner, and presents the design of the improved strategy which create a forwarding mapping for subsequent fragments based on IP compression header of the first fragment. The experiments show that the improved strategy proposed in this paper perform better than original one in transmission and time delay in dealing with IP fragmented packets.

**Key words:** 6LoWPAN; route-over; virtual circuit; fragments forwarding

个域网(Personal Area Network, PAN)是指个人范围(数米之内)的计算设备组成的通信网络, IEEE 802.15.4<sup>[1]</sup>标准是一种用于低速率、易丢包个域网(Low Rate and Lossy Wireless Personal Area Network, LoWPAN)上的无线链路. 它使 2.4GHz 频段, 最高数据传输速率只有 250Kb/s, 最大传输单元(MTU)为 127Byte, 且通讯距离较短. 由于需满足大量部署、易携带、嵌入部署、无人值守等要求, LoWPAN 设备必须小尺寸和低成本, 这也往往导致 LoWPAN 设备的计算能力、存储空间和能量等资源十分有限. IPv6 作为下一代互联网的网络层协议<sup>[2]</sup>, 其地址长度由原来 IPv4 的 32 位增加到 128 位, 可用地址数大大增加, 这使得

为嵌入式应用环境中数量众多的每个设备分配一个全局唯一的 IPv6 地址成为可能, 而 IP 协议的使用又可以大大提高了嵌入式应用环境与外部网络(如互联网)的交互性. 然而要让 LoWPAN 支持 IP 协议, 人们要面对巨大的挑战. 最主要的原因是 IPv6 规定其链路层的 MTU 不能小于 1280Byte, 而 IEEE802.15.4 的链路层 MTU 只有 127Byte.

为了解决上述问题, IETF 的 6LoWPAN(IPv6 over LoWPAN)任务组发布了用于在无线个域网上进行 IPv6 数据传输的 6LoWPAN 标准 RFC4944<sup>[3]</sup>. 6LoWPAN 在 TCP/IP 协议栈的网络层与链路层之间定义了适配层(Adaptation Layer), 适配层主要实现了 3 种

<sup>①</sup> 收稿时间:2012-12-19;收到修改稿时间:2013-01-28

功能: 头部压缩、分片重组以及第二层的数据转发。头部压缩可对较长的 IPv6 头部进行压缩以节约空间, 分片重组功能将较长的 IPv6 数据包分片以适应 802.15.4 链路层的 MTU, 数据转发功能则是根据链路层地址进行路由。目前, 人们对 6LoWPAN 的研究主要集中在头部压缩<sup>[4]</sup>、数据包的分片重组、地址配置和路由技术<sup>[5,6]</sup>等方面, 其中针对 6LoWPAN 的路由研究较少。6LoWPAN 的路由根据负责路由决策程序所属层的不同, 分别为 route-over 和 mesh-under 两种策略。一些文献[7,8]对 route-over 和 mesh-under 分别进行了研究, 并对其不同条件的性能进行了比较。文献[9]的研究认为 route-over 比 mesh-under 更具可行性, 因为要实现 mesh-under 路由方案, 需要在链路层模拟出一个完全广播域, 然而在 LoWPAN 上模拟一个完全广播域要比在传统的基于基础设施的 802.11 上要复杂的多。

首先简要介绍了 6LoWPAN 的分片机制, 分析了传统的 route-over 策略在分片转发过程中存在的不足, 然后提出了基于虚电路方法的 route-over 分片转发策略及其设计实现, 最后通过实验仿真对新策略的有效性进行了验证。

## 1 相关工作

### 1.1 6LoWPAN 的分片转发

6LoWPAN 中的分片通过 Fragment Header 实现, 分片头格式如图 1 所示, 图 1(a)表示第 1 个分片头(前 5 位是 11000), 图 1(b)表示后续分片头(前 5 位是 11100)。datagram\_size 表示未分片时 IP 数据包的大小, 同一个 IP 数据包所有分片的 datagram\_size 都相同。datagram\_tag 用于标识该分片属于哪一个 IP 数据包, 同一个 IP 数据包所有分片的 datagram\_tag 都应该相同, 该值由进行分片的结点维护与设置。后续分片头中的 datagram\_offset 表示当前分片数据在原始 IP 数据包中的偏移量, 单位为 8 字节。

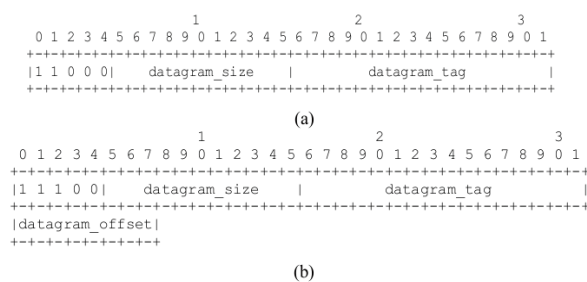


图 1 Fragment Header 格式

route-over 和 mesh-under 在传输 IP 数据包分片时存在比较大的区别: mesh-under 是基于点对点的路由方式, 分片可通过不同路径从源节点到达目的节点; route-over 是基于逐跳的路由方式, 分片在转发到下一跳后需要将收到的分片重组以检查收到的分片是否完整, 然后再将分片转发给下一跳。

### 1.2 route-over 存在的不足

如 1.1 所述, route-over 在处理 IP 数据包分片转发过程中, 接收分片的结点需要在适配层将收到的分片进行重组以检查是否有分片丢失, 然后再把重组后的 IP 数据包交付给网络层。若网络层判断当前结点就是该 IP 数据包的目的地, 则将该 IP 数据包中的有效载荷交付给上层; 否则, 网络层通过路由决策获得 IP 数据包的下一跳, 然后将数据重新交付给适配层, 适配层需要重新分片, 并转发给下一跳。由此可见, 针对 IP 数据包需要分片的情况, 在多跳的 LoWPAN 中, IP 数据包经过的每个中间结点都需要进行重组与分片, 这些工作需要消耗较多的计算资源。

另外, 由于中间结点需要等到所有分片都到齐后才进行路由和转发操作, 因此传统的 route-over 策略会消耗结点较多的存储资源, 尤其在同时收到多个 IP 数据包分片的情况下, 结点在等待重组转发时, 甚至会遭遇缓存溢出的问题。

### 1.3 相关研究进展

文献[6]对 route-over 和 mesh-under 这两种路由策略进行了分析, 认为 route-over 比 mesh-under 更适合 LoWPAN。文献[7]对 6LoWPAN 中的两种路由策略进行了比较, 在无需进行 IP 分片的情况下, mesh-under 在时延方面比 route-over 更好, 该研究反映了这样的事实, route-over 比 mesh-under 多了这么一些处理环节, 即适配层需要先将数据包交付给网络层, 再由网络层进行路由决策, 然后再将数据包交付给适配层等。文献[8]使用相同的概率模型对多跳网络且需要 IP 包分片情况下的 mesh-under 和 route-over 进行了比较, 研究表明 route-over 拥有更高的分片到达概率, 但 route-over 也可能会遭受缓存溢出问题。

## 2 基于虚电路的 route-over 分片转发改进

针对 1.2 所描述的传统 route-over 策略在分片转发中存在的问题, 提出了基于虚电路的 route-over 分片转发策略。由于一个 IP 数据包的所有分片中只有

第 1 个分片中携带 HC1 编码, 该 HC1 编码实现了对 IPv6 头部的压缩. 基于虚电路的 route-over 分片转发策略的基本过程如下: 中间结点使用第 1 个分片中携带的 IP 头部信息进行路由以确定下一跳, 然后为该 IP 数据包建立一个转发映射. 当中间结点收到后续分片时, 只需根据前面建立的转发映射将后续分片直接转发给下一跳. 下面重点介绍基于虚电路的 route-over 分片转发策略设计中用到的转发映射和分片状态的表示, 以及使用它们实现的分片流水转发机制.

2.1 转发映射的建立与使用

转发映射表是实现基于虚电路的 route-over 分片转发策略的关键, 其中的转发映射项可表示为一个四元组  $M=\{A_P, T_P, A_N, T_N\}$ , 其中  $A_P$  表示当前分片前一跳结点的 MAC 地址,  $T_P$  表示当前分片的 TAG 标识,  $A_N$  表示根据网络层路由决策后确定的下一跳 MAC 地址,  $T_N$  表示新的 TAG 标识, 当前分片转发给下一跳时需要分片的 TAG 标识更新为此值.

当中间结点收到某个 IP 数据包的第 1 个分片时, 建立针对该 IP 数据包的转发映射项, 其处理过程如图 2 所示. 首先获取该分片所在帧的源 MAC 地址和该分片的 TAG 标识; 然后, 对 HC1 进行解压, 将解压获得的 IP 头部交付给网络层; 网络层通过收到的 IP 头部信息完成路由决策, 确定下一跳 IP 地址; 适配层根据下一跳 IP 地址获得下一跳的 MAC 地址, 并为当前 IP 数据包分配一个新的 TAG 标识, 同时将源 MAC 地址、老的 TAG 标识、下一跳 MAC 地址和新分配的 TAG 标识建立一个转发映射, 并把第 1 个分片根据新建的转发映射转发给下一跳.

|     |                          |                         |
|-----|--------------------------|-------------------------|
| 网络层 | ④路由选择, 确定下一跳 IP 地址       |                         |
| 适应层 | ③解压 HC1, 交网络层            | ⑤获得下一跳地址                |
|     | ②获取源 MAC 地址<br>分片 TAG 标识 | 分配新的 TAG 标识<br>⑥建立转发映射项 |
| 链路层 | ①接收帧, 交给适应层              | ⑦将帧转发给下一跳               |

图 2 中间结点收到第 1 个分片的处理过程

中间结点收到后续分片时的处理过程如图 3 所示. 首先, 记录该分片所在帧的源 MAC 地址和该分片的 TAG 标识; 然后从转发映射表中查找相关的转发映射; 最后, 根据对应的转发映射更新该分片的 TAG 标识, 并将其发送给下一跳 MAC 地址.

|     |  |           |
|-----|--|-----------|
| 网络层 | 无需操作   |           |
| 适应层 | ②获取源 MAC 地址、分片头 TAG 标志<br>③根据转发映射更新 TAG 标识<br>同时确定下一跳 MAC 地址 |           |
| 链路层 | ①接收帧, 交给适应层  | ④将帧转发给下一跳 |

图 3 中间结点收到后续分片的处理过程

在本文提出的改进策略中, 同一 IP 数据包所有分片在每个 IP 跳中都需要更新 TAG 标识. 这样的设计主要考虑了如图 4 所示的情况: 结点 X 向结点 C 发送数据, 结点 Y 向结点 D 发送数据. 如果每一跳转发不更新 TAG 标识, 则 A 和 B 上建立的转发映射如表格 1 所示. 由于结点 X 和结点 Y 独立生成分片的 TAG 标识, 结点 X 生成的 TAG<sub>X</sub> 和结点 Y 生成的 TAG<sub>Y</sub> 有可能存在相同. 由于 TAG<sub>X</sub>=TAG<sub>Y</sub>, 结点 B 的转发就会有问题, 因为结点 B 无法确定来自结点 A 的分片究竟应该转发给结点 C 还是结点 D. 所以, 我们对同一 IP 数据名所有分片的 TAG 标识在每一 IP 跳中都进行了更新.

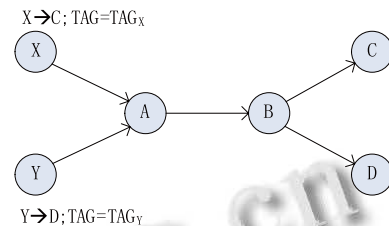


图 4 网络通信例 1

表 1 网络通信例 1 中各结点的分片转发映射

| 当前结点 | 上一跳 | 分片 TAG           | 下一跳 |
|------|-----|------------------|-----|
| A    | X   | TAG <sub>X</sub> | B   |
| A    | Y   | TAG <sub>Y</sub> | B   |
| B    | A   | TAG <sub>X</sub> | C   |
| B    | A   | TAG <sub>Y</sub> | D   |

2.2 分片状态信息的表示

除转发映射表外, 每个中间结点还需要建立并维护一个分片状态信息表, 其中的分片状态信息项表示某个 IP 数据包的分片状态信息, 可表示为一个五元组  $FS=\{A_P, T_P, L_{total}, Timeout, ListFS\}$ . 其中  $A_P, T_P$  的含义同上所述,  $L_{total}$  表示当前 IP 数据包总长度, 从第 1 个分片的 HC1 经压缩后获得.  $Timeout$  表示接收当前 IP 数据包分片的过期时间, 如果该结点未能在  $Timeout$  指定的时间前收到所有分片, 则当前结点应该丢弃所有收到

的分片,并释放对应的分片状态信息项和转发映射项。 $ListFS$  是一个列表,表示已收到分片的相关状态,其中每一个列表项  $ItemFS$  都可以表示为一个四元组  $\{Offset, Length, SS, Bufpoint\}$ 。其中,  $Offset$  表示当前分片数据在整个 IP 数据包中的偏移量,  $Length$  表示当前分片数据的长度,  $SS$  表示当前分片是否已经成功发送给下一跳,  $Bufpoint$  表示指向缓存当前分片数据的指针。

当结点收到某个 IP 数据包  $P_k$  的第 1 分片时,除了按 3.1 描述的方法建立对应的转发映射项  $M_k$  外,还需要建立一个相应的分片状态信息项  $FS_k$ 。然后,结点将  $T_p, L_{total}$  分别赋值为分片头中 TAG 和 datagram\_size 字段的值,将  $A_p$  设置为该分片所属数据帧的源 MAC 地址;将  $Timeout$  设置为当前系统时间加上最长分片收集时间(系统参数);然后在  $FS_k$  的  $ListFS$  中添加相应的  $ItemFS$  项,将  $ItemFS$  项中的  $Offset$  设置为该分片头中 datagram\_offset 字段的值,将  $Length$  设置为该分片所承载数据的长度,而  $SS$  设置为 0 以表示该分片还没有成功发送给下一结点,  $Bufpoint$  指向保存当前分片数据的缓存。当结点收到某个 IP 数据包  $P_k$  的后续分片时,需要向  $FS_k$  的  $ListFS$  中添加对应的  $ItemFS$  项。通过对  $FS_k$  的  $ListFS$  列表进行检索与统计,可以方便地获得当前 IP 数据包分片的到达情况,并生成相关的确认信息。确认信息由以下格式的若干个二元组  $\{Noffset, Nlength\}$  构成,  $Noffset$  表示空缺数据块的偏移量,  $Nlength$  表示空缺数据块的长度。发送结点可根据确认信息获前一

批数据的发送成功情况,也可获得接收结点期待哪些分片的信息。

### 2.3 基于虚电路的分片流水转发机制

基于转发映射表和分片状态信息表,我们设计了基于虚电路的 IP 数据包分片的流水转发机制。在正常情况下,当某一结点(起始发送方)需要发送某个 IP 数据包时,首先对该 IP 数据包进行分片,并建立与之对应的分片状态信息项,然后全部或分批将分片按顺序发送给下一结点,发送前,需要为每一批分片中的最后分片设置“需要确认”标识。因此,我们在 6LoWPAN 中新增了“需要确认”的分片头,其格式如表格 2 所示。下一结点在收到第 1 个分片时,根据 3.1 所描述的方法建立一个转发映射项和一个分片状态信息项,同时保存收到的分片数据。当收到需要确认的分片时,该结点对该分片之前的分片的收到情况进行统计,并向上一结点发送确认。上一结点根据收到的确认信息,将下一结点已成功发送分片所对应的  $ItemFS$  项中的  $SS$  设置为 1,并释放  $ItemFS$  项中  $Bufpoint$  指向的缓存。若当前 IP 数据包还有部分分片没有成功发送,则继续向下一结点发送分片;否则,表明当前 IP 数据包的所有分片都已成功发送到下一结点,当前结点要将该 IP 数据包所对应的转发映射项和分片状态信息项删除。中间结点在等待后续分片时,也可以将已经收到的分片发送给后一跳,从而实现了基于虚电路的 route-over 分片流水转发机制。

表 2 支持分片接收确认的 Fragment Header

| 0-1 位 | 2-7 位  | 类型     | 描述   |
|-------|--------|--------|--|
| 11    | 000xxx | FRAG1  | Fragmentation Header (first)                             |
| 11    | 001xxx | FRAG1A | Fragmentation Header (first,acknowledgement needed)      |
| 11    | 100xxx | FRAGN  | Fragmentation Header (subsequent)                        |
| 11    | 101xxx | FRAGNA | Fragmentation Header (subsequent,acknowledgement needed) |

## 3 实验分析

为了验证本文提出改进方案的有效性,对传统的 route-over 分片转发策略和本文提出的分片转发策略分别在 NS2 上进行了实验仿真。由于本次实验主要目的是为了比较这两种策略的性能,与采用何种路由算法没有关系,因此,我们在实验中采用类似  $S \rightarrow A \rightarrow B \rightarrow D$  的线性网络结构,并在实验中采用了固定路由方案,即: S 单跳到 A, A 单跳到 B, B 单跳到 D。测试数据选用 UDP 数据包,其载荷大小以 100 字节为间隔从 100 个字节递增到 1200 个字节,每种载

荷分别进行 10 次实验得到相关的端到端的时延,并求其平均值。得到两种策略的端到端平均时延如图 5 所示。

从实验结果中可以看出,本文提出的 route-over 分片转发改进策略能够显著减少数据包端到端的传输时延。这主要归功于以下两点:(1)与传统 route-over 策略相比,本文提出的策略使得每个中间结点节省了对 IP 数据包进行重组和分片的操作;(2)本文提出的策略在一定程度上实现了分片在路由路径上的流水传输。

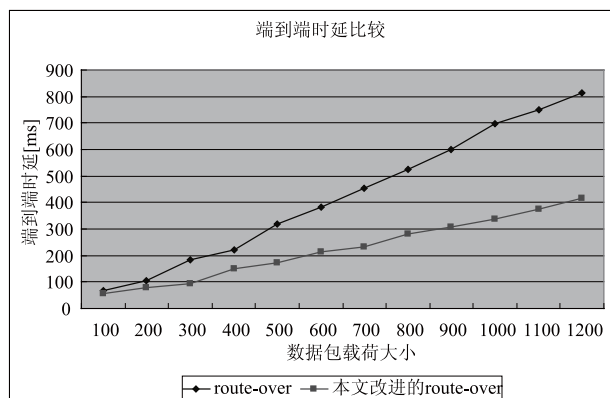


图5 端到端时延比较

#### 4 结束语

本文分析了传统 route-over 策略在分片转发上存在的问题,提出了基于虚电路的 route-over 分片转发策略和相关设计,实现了分片在路由路径上的流水传输.实验结果表明改进的 route-over 分片转发策略比传统的 route-over 在端到端时延指标上有显著提高.另外从理论上讲,本文提出的策略能够在一定程度上缓解中间结点在缓存空间方面的压力,这也是我们下一步工作要重点研究的内容.

#### 参考文献

- 1 IEEE Computer Society. IEEE Std. 802.15.4-2003. IEEE Std. 802.15.4-2003, October 2003.
- 2 Deering S, Hinden R. Internet Protocol, Version 6 (IPv6)

Specification. RFC: 2460, December 1998.

- 3 Montenegro G, et al. Transmission of IPv6 Packets over IEEE 802.15.4 Networks. RFC: 4944, September 2007.
- 4 Tripathi J, de Oliveira J, Vasseur J. A performance evaluation study of RPL: routing protocol for low power and lossy networks. 44th Annual Conference on Information Sciences and Systems (CISS), May 2010.
- 5 Hui J, Thubert P. Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks. RFC: 6282, September 2011.
- 6 Kim K, Yoo S, Lee J. Hierarchical Routing over 6LoWPAN (HiLOW)[Internet-Draft]. 2007.
- 7 Ludovici A, Calveras A. Implementation and Evaluation of Multi-Hop Routing in 6LoWPAN. Proc. of the 9th Conference of Telematic Engineering, 2010.
- 8 Chowdhury AH, Ikram M, Cha HS, Redwan H, Saif Shams SM, Kim KH, Yoo SW. Route-Over vs Mesh-Under Routing in 6LoWPAN. Proc. of IWCMC'09, the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly, June 2009.
- 9 Hui JW, Culler DE. Extending IP to Low-Power, Wireless Personal Area Networks. IEEE, Internet Computing, 2008, 12(4).
- 10 Hui JW. An Extended Internet Architecture for Low-Power Wireless Networks-Design and Implementation [Ph.D. Thesis]. University of California, Berkeley, CA, USA, 2008: 85-87.

(上接第 211 页)

Graph. Appl, 1986, 6(9): 40-53.

- 7 Benjamin W, Neff W, Hodges LF. Supra-threshold control of peripheral LOD. ACM Trans. on Graphics, 2004, 23(4): 750-759.
- 8 隋树元, 王树山. 终点效应学. 北京: 国防工业出版社, 2000.
- 9 王乘, 李利军, 周均清, 等. Vega 实时三维视景仿真技术. 武汉: 华中科技大学出版社, 2005.

- 10 Zhou YH, Wen D, Cao SK, Lv MY. The algorithm of cloth fast collision detection based on the characteristics triangle. IJACT, 4(7): 231-238.
- 11 王晓, 鲁娜, 杨占营. Vega 中的碰撞检测技术研究. 信息技术, 2007(11): 78-80.
- 12 顾文彬, 郑向平, 唐勇. Vega 下带碰撞检测响应视点方式的开发. 系统仿真学报, 2008, 20(22): 6197-6200