1概述

最佳链路状态路由协议（Optimized Link State Routing Protocol，OLSR），是专门为无线移动Ad Hoc网络提出来的一种标准化的先验式的优化链路状态路由协议。

1.1设计原理

OLSR由传统LSR路由协议改进而来。传统链路状态协议每个节点通过周期性的交换链路状态信息维护整个网络的拓扑信息。在OLSR协议中，网络中的每个节点只选择对称邻居节点的一个子集作为多点中继集MPR（Multipoint Relay），只有被选为MPR的节点才产生并转发TC（Topology Control）分组，同时OLSR只利用MPR节点到MS（MPR Selector）节点之间的链路状态信息来建立最短路由，这样很大程度上减少了转发的信息，减少了网络中洪泛的控制信息。

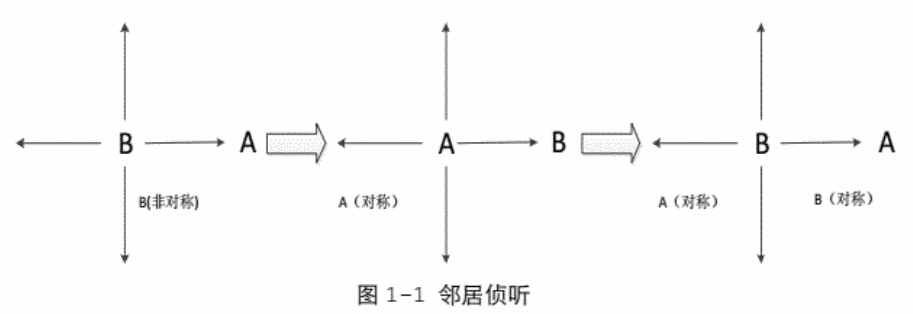
1.2协议的主要流程（未完待续）

1.2.1链路感知

因为无线电传播的不确定性可能会导致一些链路是单向的，所以每个节点必须检测自己和邻居节点之间的链接是否是双向的，也就是在两个方向上都能传输数据。本地链路信息表存储了到邻居的链接的信息，完善这一集合的过程称为“链路感知”。链路感知通过HELLO分组的周期性交互实现，当一个节点收到HELLO分组时，更新自身的本地链路信息表。

1.2.2邻居侦听

每个节点必须检测与哪些邻居节点具有双向链路，节点周期性地广播HELLO分组。HELLO分组用来侦听邻居节点的状态，分布在一跳范围内转播，不能被转发。如图1-1所示，在初始化阶段，节点B广播HELLO分组，当节点A收到这个分组之后，将B放入到自己的邻居节点集中，并标记A到B链路状态为非对称的；然后A广播HELLO分组，其中包含B是A的邻居节点且B到A的链路状态为非对称这一信息，当B收到这一分组时，会在自己的邻居集中将到A的链路状态更新为对称；当B再次广播HELLO分组，A收到这一分组就会将邻居集中的A到B的链路状态更新为对称的。



1.2.3MPR选择

OLSR采用MPR机制对路由信息进行选择性的洪泛。网络中的每个节点选择自己的一跳对称邻居节点的子集作为中继节点，即MPR，而该节点本身作为MS节点。

计算MPR集需要知道自身一跳和两跳邻居的信息。一跳邻居信息通过HELLO分组来获取。对于节点i的一跳邻居节点j来说，i的一跳邻居节点（除j）就是j的二跳邻居节点，所以节点i在发送HELLO分组时附上自己的一跳邻居节点列表，当节点j收到来自i的分组时，就获得了节点j的两跳邻居节点信息。MPR集应满足以下两点要求：节点与MPR之间必须是一跳双向链路；节点能通过MPR集到达所有的严格两跳邻居节点。

1.2.4TC分组处理

TC分组由MPR广播和转发。MPR节点每隔一段时间就向全网洪泛TC消息来维护网络拓扑信息。

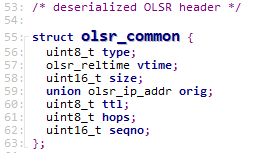
1.2.5路由表的计算与维护

网络中的每一个节点都维护一个路由表，路由表的计算是基于节点维护的本地链路信息表和拓扑集。因此，如果本地链路信息表和拓扑表发生变动时，路由表会重新计算并更新表项。在OLSR标准协议中，协议根据最小跳数建立每个节点的路由表。任意一个节点的路由表添加过程如下：首先，添加对称链路的一跳邻居节点；然后，添加两跳邻居节点；最后，循环添加跳数等于h+1（h=2开始）的节点进入路由表。

2数据结构部分

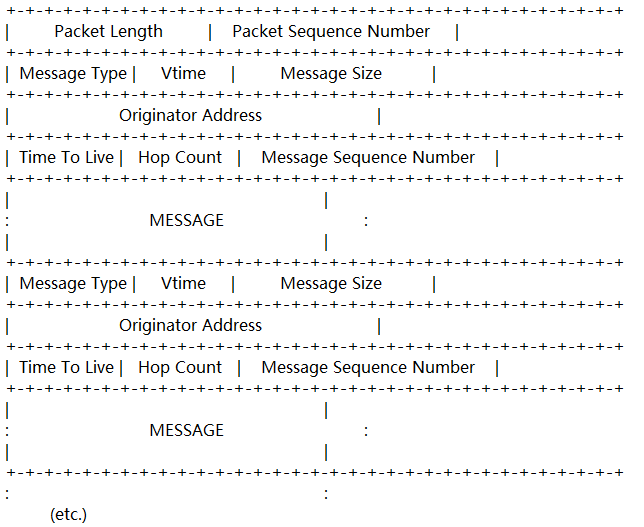
2.1部分消息包的报文格式

2.1.1OLSR首部（省略IP和UDP报头）



olsr\_common是OLSR协议的基本数据包首部。

对于与协议相关的所有数据，OLSR使用统一的数据包格式进行通信，这样做的目的是在不破坏向后兼容性的情况下促进协议的可扩展性。这也提供了一种简单的方法，将不同“类型”的信息汇集到一个单一的传输中。这些数据包嵌入在UDP数据报中，使用UDP通信，IANA将端口698分配给OLSR协议专用。每个分组封装一个或多个消息，这些消息共享一个通用的报头格式，使节点能够接受和重传未知类型的消息。OLSR协议分组的基本格式如图所示：



a.数据包首部

*Packet Length：*数据包的长度（单位：字节）

*Packet Sequence Number：*数据包序列号，每当传送一个新的OLSR分组时，序列号加一。为每个接口维护一个单独的分组序列号，以便对通过接口发送的分组进行顺序枚举。

b.消息首部

*Message* *Type* ：消息类型，在0-127范围内的消息类型保留给本协议中的所有消息和以后可能的扩展。

*Vtime* ：该字段表明接收节点后多长时间必须将消息中包含的信息视为有效，除非接收到对信息的最新更新。有效时间由它的尾数a（Vtime字段的四个最高位）和它的指数b（Vtime字段的四个最低位）表示。有效期=C\*（1a/16）\*2^b[秒]，C为一系数。

*Message Size*：消息的大小，以字节为单位计算。并从“消息类型”字段的开头到下一个“消息类型”字段的开头（如果没有下一个“消息类型”，则到该消息的结尾为止)。

*Originator Address*：此字段包含最初生成此消息的节点的主地址，重传过程中地址不发生改变。此字段不应与IP标头中的源地址混淆，后者每次都更改为重传该消息的中间接口的地址。

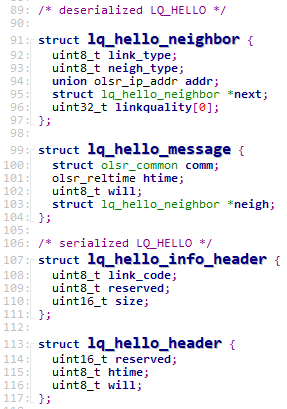
*Time To Live*：此字段包含将发送消息的最大跳数。消息每次重传前TTL减1，当TTL为0或1时，则不再重传。通过此方式可以限制一个消息的洪泛范围。

*Hop Count*：一个消息获得的跳数的数量。每次重传消息前，跳数加一。初始值为0。

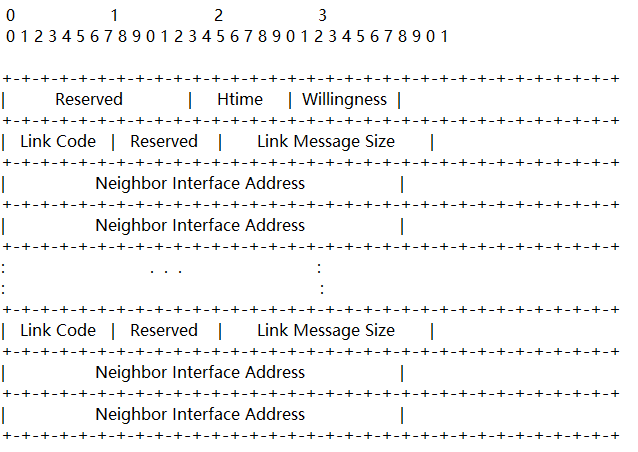
*Message Sequence Number*：由源节点产生的一个消息的唯一标识，用来保证一个消息不会被任何节点重传一次以上。

2.1.2HELLO消息

OLSR协议采用一种通用的机制来填充本地链路信息库和邻居信息库，即周期性地交换HELLO消息。



106-117：结构体lq\_hello\_info\_header和lq\_hello\_header共同组成HELLO消息数据包的首部。格式如下图：



这一部分是将OLSR基本数据包首部的"Message Type"设为HELLO\_MESSAGE，TTL设为1，Vtime设为NEIGHB\_HOLD\_TIME。

Reserved:保留字段,必须设置设为“0000000000000

Htime:指定节点在特定接口上的HELLO发射间隔，即下一个HELLO传输前的时间。Hello发射间隔用尾数(Htime的四个最高位)和指数(Htime的四个最低位)来表示。即HELLO发射间隔=C\*（1+a/16）\*2^b[秒]。其中a是Htime字段的四个最高位表示的整数，b是Htime字段的四个最低位表示的整数。

willingness:此字段指定节点为其他节点承载和转发流量的意愿。

Willingness有三种取值：

1. WILL\_NEVER：永远不会选择意愿为WILL\_NEVER的节点作为MPR。
2. WILL\_ALWAYS：具有WILL\_ALWAYS意愿的节点将始终被选择为MPR。
3. WILL\_DEFAULT：默认情况下，节点意愿为WILL\_DEFAULT。

Link Code:

此字段指定发送方的接口和邻居列表中邻居接口之间的链路类型。它还指定有关邻居状态的信息。链路类型不为节点所知的邻居信息被静默丢弃:

链路类型有以下三种：

(1) ASYM\_LINK:发送HELLO分组的节点与邻居列表中的节点间的链路是非对称的。表示可以收到邻居节点的消息,但不确定邻居节点能否收到本节点的消息。

(2) SYM\_LINK:发送HELLO分组的节点与列表中的邻节点间的链路是对称的。表示链路已经被验证为双向的。

(3)MPR\_LINK:表示列表中的节点已被发送该HELLO分组的节点选择为MPR。

Link Message Size：本链路消息的大小，以字节为单位，从Link Code字段开始到下一个Link Code字段之前（如果没有下一个Link Code，则到该消息尾部）。

Neighbor Interface Address：邻居节点的接口地址。每一种链路类型之后都紧跟一组邻居节点接口地址，表明该节点与这组邻居节点中的每一个节点链路类型都相同，都为前面给出的链路类型。

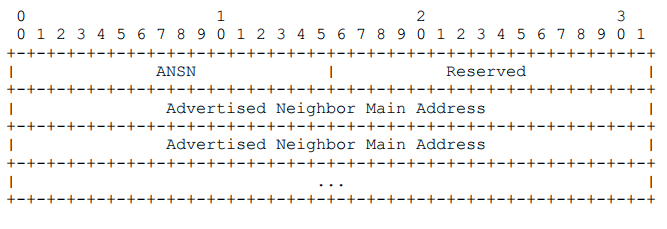


49-57：结构体hello\_neighbor是HELLO消息邻居节点集。Status，记录邻居的状态；link指明链路类型；main\_address是邻居的主地址；address是邻居的其他地址；cost，链路代价；linkquality，链路质量。

59-69：hello\_message是消息数据包。vtime，消息有效时间；htime，HELLO消息的发射间隔；source\_addr，发送消息的原地址；packet\_seq\_number，数据包的序列号；hop\_count，消息已经经历的跳数；ttl，数据包生命周期；willingness，节点进行转发的意愿；neighbors，下一个传递的邻居节点。

2.1.3.TC消息

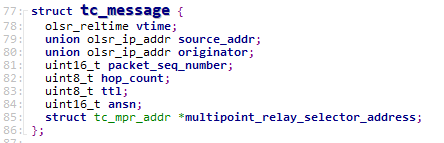
该部分是将OLSR首部中“MessageType”设置为TC\_Message。TTL设置为255(最大值)，以便将消息传播到网络中Vtime相应地设置为Top\_hold\_time的值。



ANSN：序列号与公布的邻居集相关。每当节点检测到其邻居集中的更改时，它都会递增这个序列号。当节点接收到TC消息时，它可以根据该消息来判定所接收到的信息是否比现有消息更新。

Advertised Neighbor Main Address：此字段包含邻居节点（此处的邻居节点仅为该节点的MPR Selector集中的节点）的主地址。

Reserved：此字段是保留的，必须设置为“0000000000000000”。

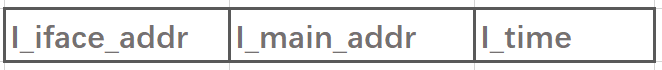


77-86：tc\_message是TC消息数据包格式。OLSR协议利用TC拓扑表记录接收到的TC消息内容。TC拓扑表的介绍见“节点存储的表项”。

2.2节点存储的表项

2.2.1. 接口关联集合

网络中的每一个目的地节点存储了接口关联多元组。



i\_iface\_addr是节点的接口地址；

i\_main\_addr是该节点的主地址；

i\_time指定此元组过期的时间，以及必须删除的时间。

2.2.2.本地链路信息表



L\_local\_iface\_addr是本地节点(即链路的一个端点)的接口地址；

L\_neighbor\_iface\_addr是相邻节点的接口地址；

L\_SYM\_time是链路被视为对称的时间；

L\_ASYM\_time是被认为听到邻居接口的时间；

L\_Time指定了此记录过期的时间，并且必须被删除。当L\_SYM\_TIME和L\_ASYM\_TIME过期时，该链接被视为丢失。

2.2.3.一跳邻居表

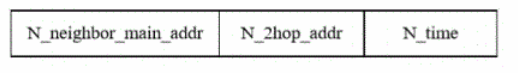


N\_neighbor\_main\_addr：邻居的主地址；

N\_status：指定节点是NOT\_SYM还是SYM；

N\_willingness：指定节点的携带意愿，取值为0到7之间的整数。

2.2.4.二跳邻居表



N\_neighbor\_main\_addr是邻居的主地址；

N\_2hop\_addr是具有与N\_neighbor\_main\_addr对称链接的2跳邻居的主地址，也就是说节点通过邻居节点N\_neighbor\_main\_addr到达其二跳邻居节点N\_2hop\_addr；

N\_time指定元组过期和必须删除的时间。

2.2.5.MPR表

一个节点维护着其被选为MPR的邻居节点的集合，这些被选为MPR的邻居节点的主地址存放在MPR集中

2.2.6.MPR Selector表

描述选择本节点为MPR的邻居节点的地址。



MS\_main\_addr：是节点的主地址，该节点选择本节点为MPR；

MS\_time指定元组过期的时间和必须删除的时间。

2.2.7.TC拓扑表

网络中的每个节点维护有关网络的拓扑信息。此信息从TC消息中获取，并用于路由表计算。



T\_dest\_addr：目的地节点的主地址，它可以从地址为T\_last\_addr的节点经过一跳到达；

T\_last\_addr：是T\_dest\_addr的MPR；

T\_seq是一个序列号；

T\_time指定此元组过期的时间，以及必须删除的时间。

2.2.8.路由表



R\_dest\_addr：目的地节点地址

R\_next\_addr：下一跳节点地址

R\_dist：本节点到目的节点的距离

R\_iface\_addr：该节点转发路由信息的接口

**3.代码分析部分**

3.1