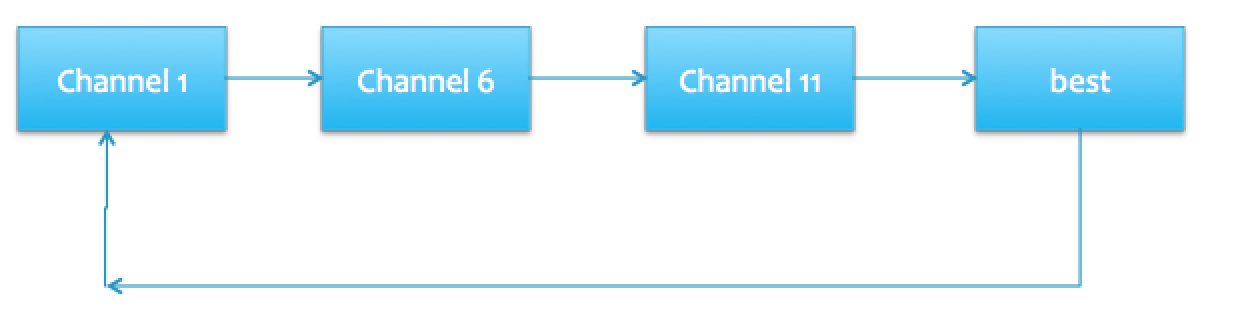
综述

第四，利用我们在前面提到的延迟测量方法，我们对于当前最为常用的最小拥塞信道选择（Least Congestion Channel Search）算法进行了优化，我们直接将我们测量的延迟结果作为最小拥塞信道选择算法的输入参数，而不是其常用的信道利用率。最终在我们针对每个独立无线路由器选择算法中，我们将无线路由器的最后一跳的延迟降低了5倍。并且考虑到切换信达可能对于用户造成的影响，我们采用了802.11n中提出的信道调整方法，能够在对于用户影响最小的情况下，调整信道。

优化的信道选择算法

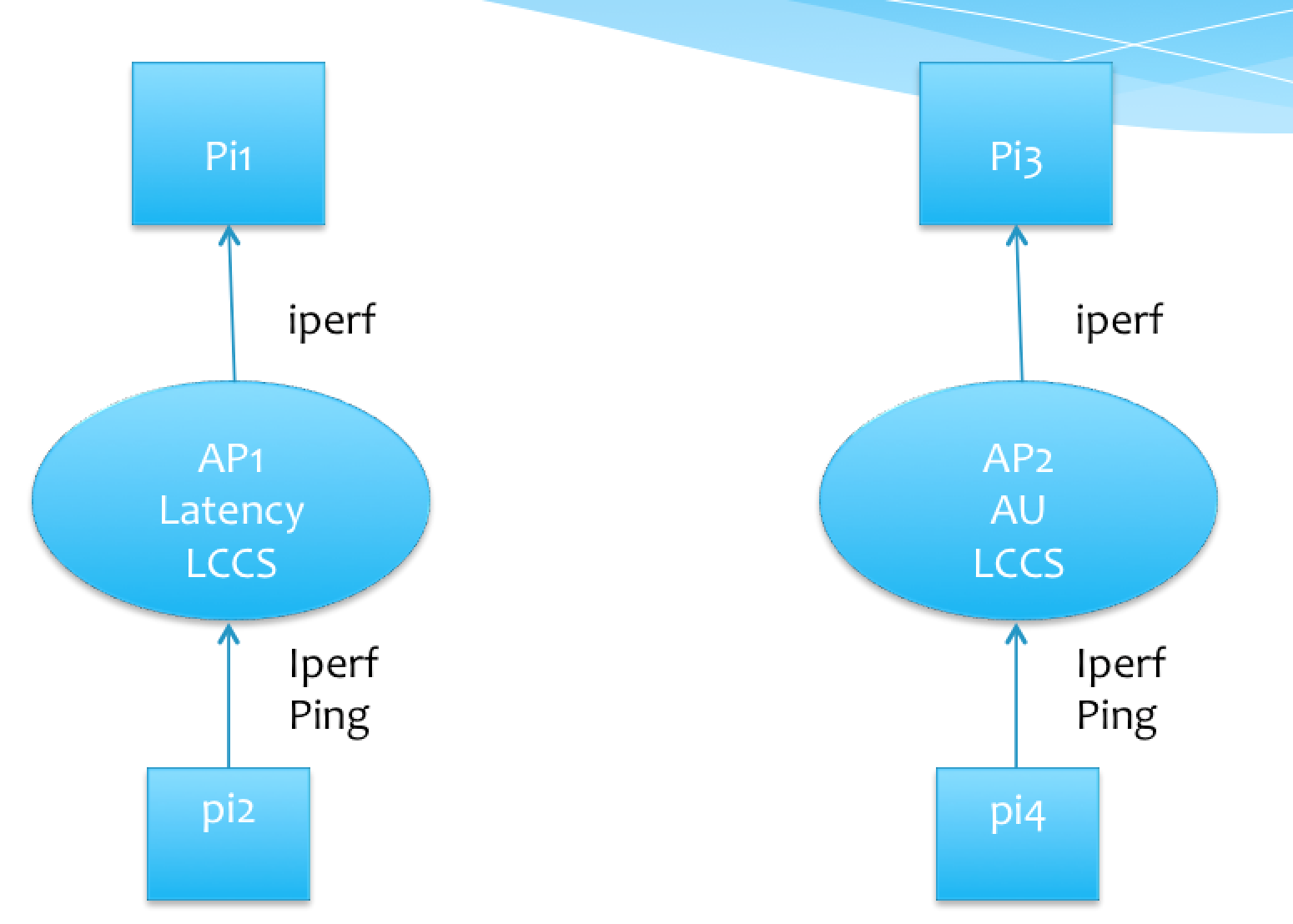
1最小拥塞信道选择算法

在受到干扰的环境下，信道调整是最有效也是最简单的优化无线路由器性能的方法。在实际的操作中，最小拥塞信道选择算法（Least Congestion Channel Search）是当前基于单个无线路由器，最为常用的无线信道选择算法。其算法执行的核心如图？？所示。由于在mac 802.11协议所规定的2.4GHz的无线信道中，只有1、6、11三个信道互不干扰，因此最小信道拥塞选择算法，就是将无线路由器的无线信道依次调整到1、6、11三个信道，并且同时采集它的无线参数，并根据这些无线参数，选择一个性能最好的信道，然后将无线信道调整到这个信道上。经过一定的时间间隔后，再次重复之前的过程。总的来说最小信道拥塞选择算法就是一个轮训的算法。基于这个算法最初的目的——减少无线信道之间的拥塞，因此这个算法最常用的选择最优无线信道的无线参数是信道利用率。而基于之前的结果，我们可以看到在信道利用率小于0.5时，无线延迟基本没有变化，也就是说，最小的无线信道利用率并不代表着最小的无线延迟。而很显然的是，对于用户来说，他所关注的重点并不在于信道是否拥塞，而是当前的无线延迟是否是最低的。因此我们想要采用我们所测量的无线延迟结果直接作为最小拥塞信道选择算法的输入参数。以下我们将说明，我们使用wing的无线延迟测量结果能够更好的优化无线路由器的无线延迟。



2实验场景介绍

为了实现对于这两种不同参数的公平评价，我们采用了背靠背，同本地流量，同干扰流量的实验方法。我们在我们部署的无线路由器中，选择了7个最为活跃的无线路由器（分布在不同的学生宿舍中），并且记录了他们一周的流量行为。另外我们利用树莓派及两台无线路由器，实现了对于这个无线流量的重放。实验拓扑如图？？所示。我们将利用在树莓派上的iperf工具，来对我们的无线流量进行重放，并且同时在无线路由器1上我们将执行基于延迟的最小拥塞信道调整算法，在无线路由器2上我们将执行基于信道利用率的最小拥塞信道调整算法。在我们的实验中，我们检测每个信道的时间间隔是5分钟，我们保持在最好信道的时间是30分钟。通过这种的拓扑及实验方法，我们可以保证每个无线路由器所受到周围无线路由器的无线干扰是相同的。并且由于他们之间的无线流量相同，因此，他们互相之间所产生的无线干扰也是相同的。并且为了公平起见，我们将在这个过程中，针对每种算法，收集我们由ping获取的无线延迟结果，并将其作为我们评价两种算法的参数。



3实验结果及分析

图？？展示了对于不同的算法（也就是对于最小拥塞信道选择算法使用不同的输入参数）我们所测量的无线延迟。我们可以看到对于所有的测试路由器，我们利用无线延迟作为信道选择参数最终的无线延迟，都要小于利用信道利用率最终的无线延迟。其中最大的延迟降低出现在无线路由器6，一共有着5倍的延迟降低。

由以上的结果可以看出，信道利用率只是对于当前无线信道占空比的一个反映，它可能对于无线延迟有着一定的影响，但是他并不能直接的反映无线延迟。并且信道利用率的采样是在一段时间内平均的结果，却不能直接的说明，高的信道利用率是否对于当前本机的数据包产生了影响。这里我们可以举一个极端的例子，如果在某无线路由器周围有一个高吞吐率的无线路由器正在发包，而其流量正好是和我们的无线路由器的流量错开，那么在这个一段时间内的信道利用率一定是一个很高的值，但是却对当前我们的无线流量并没有任何的影响。而我们在本文章中使用的无线延迟，是对每一个数据包的采样，由此可以规避对于大的延迟包的平均效应，而更加的容易识别出这其中可能存在的高延迟，从而选择一个更好的信道。

4关于信道调整的补充说明

鉴于以上我们提供的优化方法，这里我们需要对于用户的无线信道进行相对频繁的调整。首先为了不影响用户服务的使用，我们在用户的流量高峰期是不会对无线信道进行调整的。其次我们采用了mac 802.11n协议中所规定的，使用户与无线路由器协同调整信道的方法，而规避了由信道调整可能带来的用户断连。

在之前的方法中，我们采用openwrt 内部的uci wifi命令进行信道切换，这将会重启无线模块，这就会使用户收到来自无线路由器的取消认证的包，也就是告诉用户：此路由器在这段时间内无法连接。然后用户就会重新的对周围的无线路由器进行扫描，选择一个最好的无线路由器进行连接，并且需要向新的路由器发送认证连接的请求。这个过程不但可能会导致用户的断连，而且可能会使用户在这个时候就直接连接了一个新的无线路由器，而放弃了原来的无线路由器。因此整个这个过程是得不偿失的。

而在mac 802.11n中所规定的无线路由器调整方法中，在无线路由器调整信道之前，它会向用户发送探测包，来告诉用户它要调整信道了，然后在约0.9秒的倒计时后，无线路由器开始调整信道，同样的用户也会调整信道，这一切的行为都是发生在mac层的，而不会对上层的应用产生影响。经过我们的抓包实验后，也证实了这种方式，可以保证TCP正常的通信过程。

目前的OpenWrt内部的hostapd模块已经集成了这个功能，我们只需要使用hostapd\_cli就可以完成以上的信道切换过程。