在 Intel 手机上实现 aodv 协议

一、工程代码说明

aodv-Android

Based on the FBAODV, I have transplanted it to the Android 4 with linux kernel 3.4
Change the makefile varible 'KDIR', redirect it to your kernel file.
Then, export ARCH & CROSS_COMPILE.
with x86, this can be:

export ARCH=i386

export CROSS_COMPILE=i686-android-linux-

with arm, this can be:

export ARCH=arm

export CROSS_COMPILE=arm-none-linux-gnueabi-

After environment variable was set, 'make' can start the compile.
wait and *.ko is what you want.
transport the .ko adhoc-config.sh file to your device, run the adhoc-config.sh to start.
like:
sh adhoc-config.sh 192.168.1.1
the IP address is your device's IP.
PLUS: Android has block the linux kernel package forwarding, you can use
iptables –P FORWARD ACCEPT
to open that function.(untest)
OR, you can change linux kernel source code:
ip_forward， /net/ipv4/ip_forward.c:55

change

```
return NF_HOOK(NFPROTO_IPV4, NF_INET_FORWARD, skb, skb->dev,
               rt->dst.dev, ip_forward_finish);
```

to

```
return ip_forward_finish(skb);
```

二、Intel 内核编译

1. 清空已有的编译中间文件
   make mrproper
2. 将原有的 config 文件（kernel_config）复制到已解压的内核文件目录下，重命名为.config。
3. 设置编译变量

export ARCH=i386
export CROSS_COMPILE=i686-android-linux-
make ANDROID_TOOLCHAIN_FLAGS=-mno-android

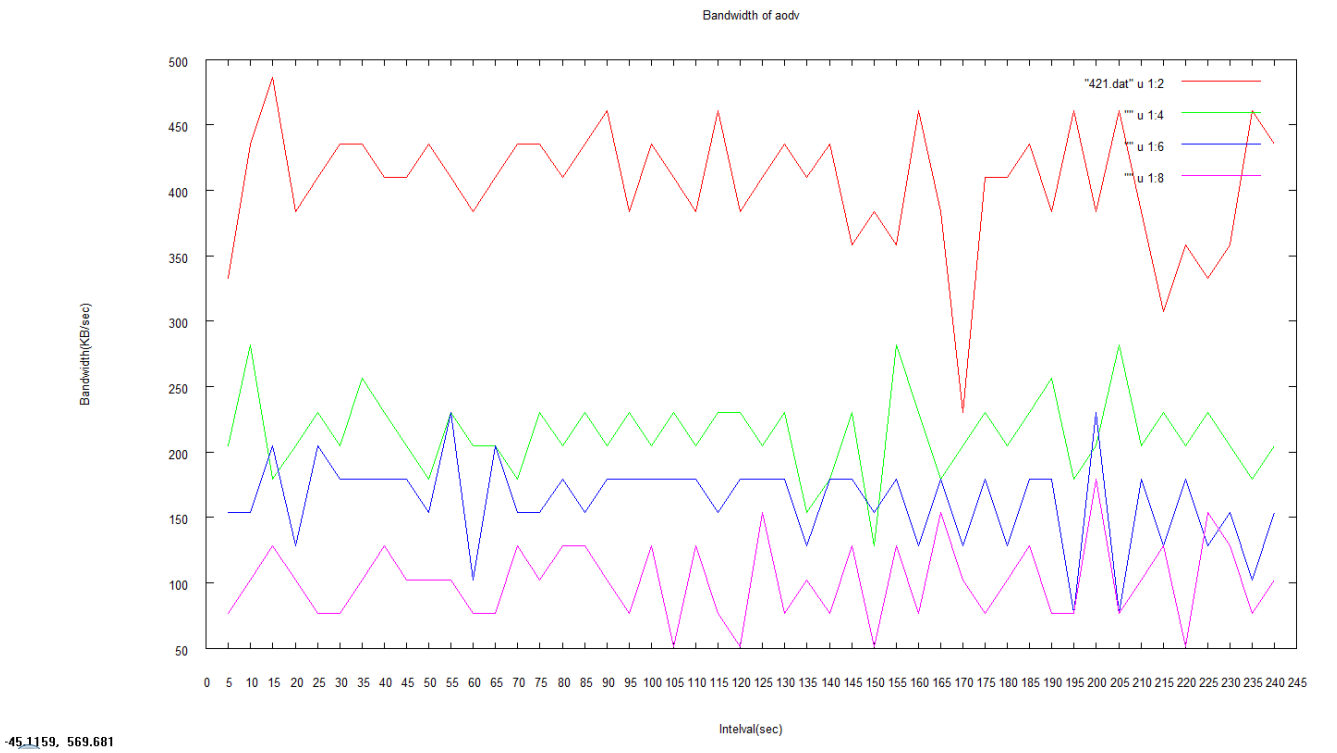
（备注：第三步为编译，编译 intel 内核需要加入变量 ANDROID_TOOLCHAIN_FLAGS=-mno-android）

4. 若编译成功后，进入 FBAODV 所在目录。在 Makefile 中将 KDIR 变量替换为上一步编译的 intel 内核所在目录。
5. 执行以下命令：
   export ARCH=i386
   export CROSS_COMPILE=i686-android-linux-
   make
6. 运行结束后看到目录下生成 fabodv.ko 文件，这就是所需要的内核模块。

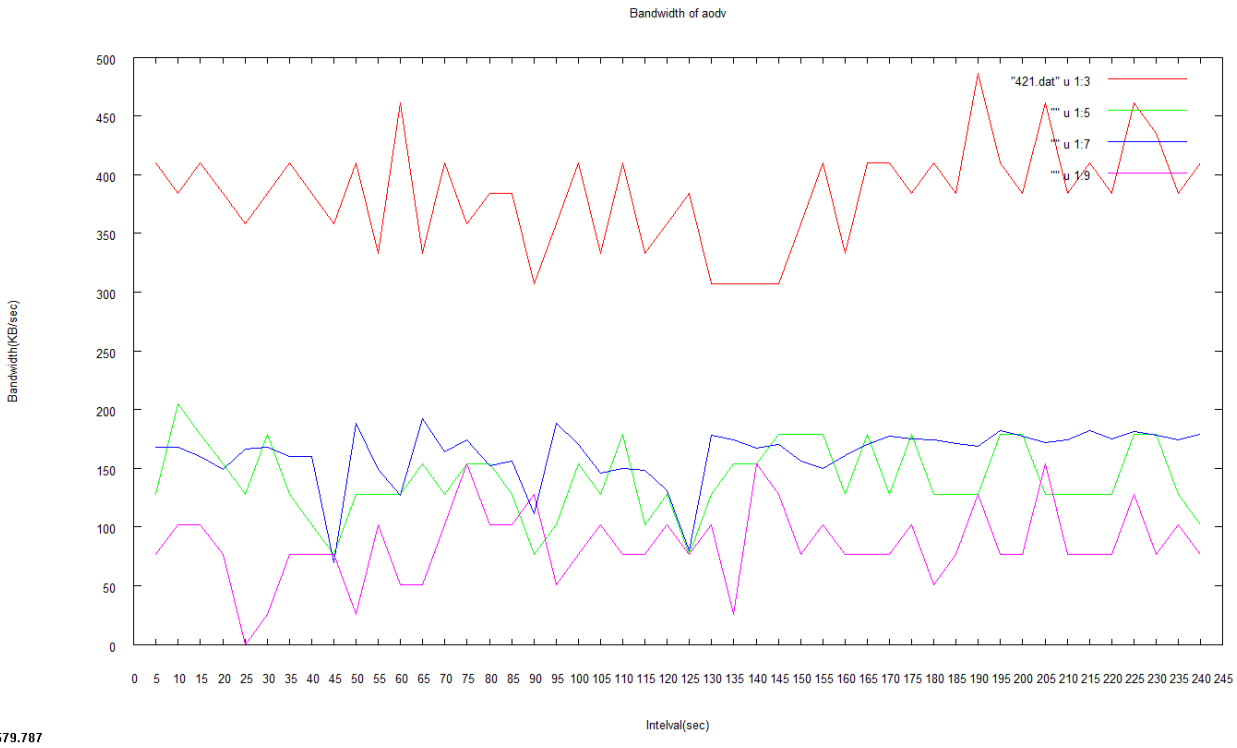
三、实验结果：

1. 不同条数的带宽比较

红色、绿色、蓝色、紫色的线分别表示一跳、两跳、三跳和四跳的数据。横坐标为时间轴，纵坐标为传输速度。



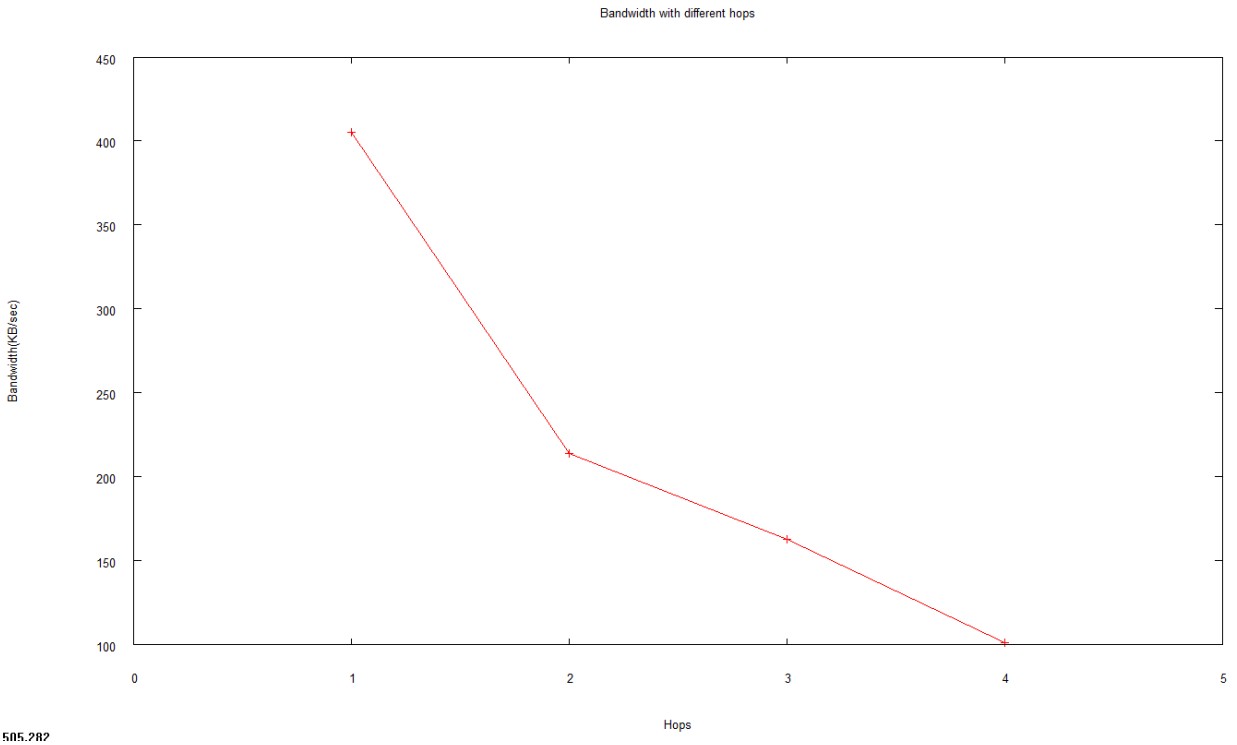
-45.1159, 569.681



-45.3150, 579.787

2. 不同跳数的数据平均值分布。

横坐标为跳数，纵坐标为速度。



-0.924797, 505.282