NRC7292 Evaluation Kit User Guide

(11s mesh)

Ultra-low power & Long-range Wi-Fi

Ver 1.3
Sep 16, 2021

NEWRACOM, Inc.

NRC7292 Evaluation Kit User Guide (11s mesh)
Ultra-low power & Long-range Wi-Fi

Office

Newracom, Inc.
25361 Commercentre Drive, Lake Forest, CA 92630 USA
http://www.newracom.com

Contents

List of Figures

1 Overview

This document describes the IEEE 802.11s mesh networking to create a Wi-Fi mesh network, which may be used for relatively fixed (not mobile) topologies and wireless ad hoc networks.

1.1 Terminology

As shown in **Error! Reference source not found.**, the 11s mesh network has four device classes.

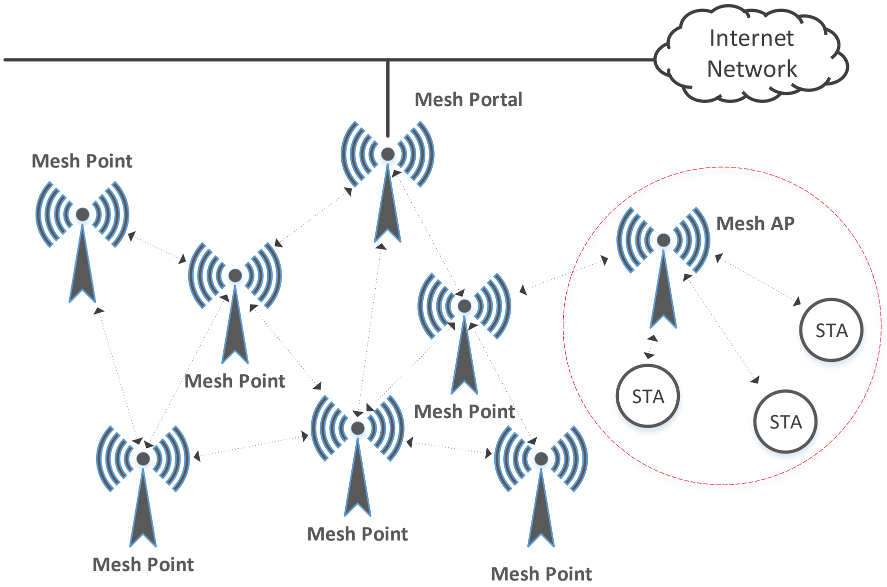


Figure 1.1 11s mesh network

Following is the brief description of each device class.

- Mesh Point (MP): establishes peer links with MP neighbors, full participant in WLAN Mesh services
- Mesh AP (MAP): functionality of a MP, collocated with AP which provides BSS services to support communication with STAs
- Mesh Portal (MPP): point at which MSDUs exit and enter a WLAN Mesh (relies on higher layer bridging functions)
- Station (STA): outside of the WLAN Mesh, connected via Mesh AP
- MBSS: Mesh Basic Service Set
- Peering: relationship between two mesh STA for direct communication

- Path Metric: performance, quality, and eligibility of a mesh path
- Hybrid Wireless Mesh Protocol (HWMP): mesh path selection protocol

1.2 Addressing

As shown in **Error! Reference source not found.** and **Error! Reference source not found.**, the 11s mesh frame has additional address fields which are used in proxy transmission from/to DS.

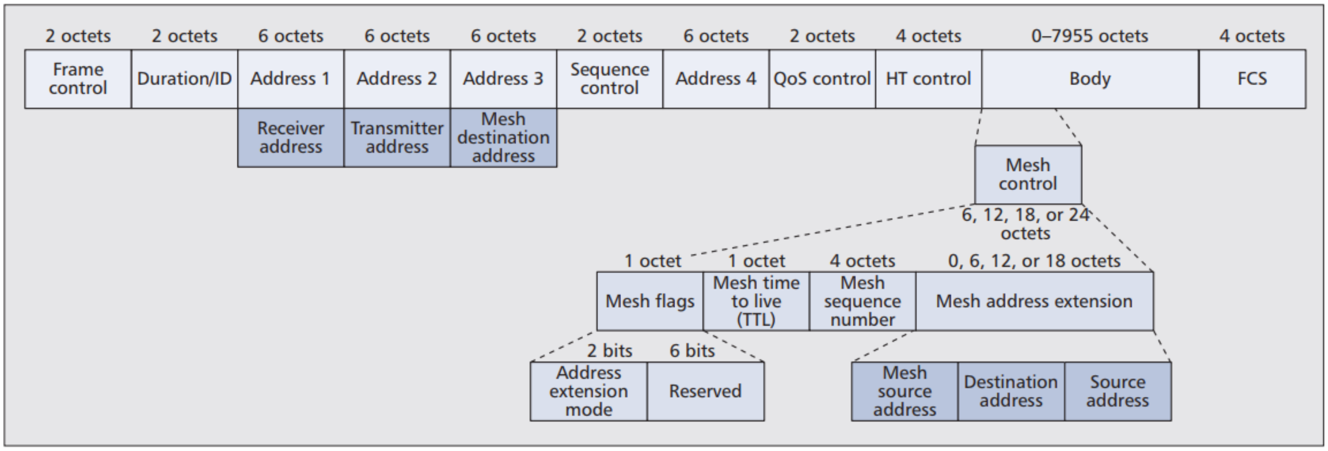
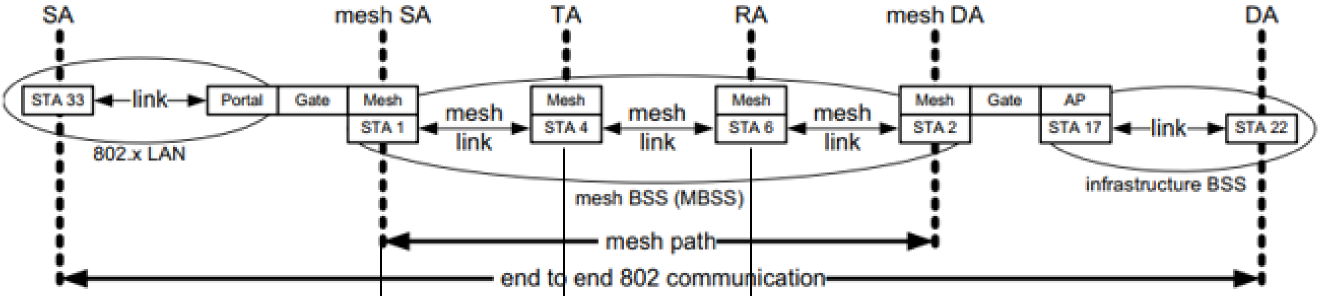


Figure 1.2 Mesh address fields



To/From	AE	Address1	Address2	Address3	Address4	Address5	Address6
11	10	STA4	STA1	STA2	STA1	STA22	STA33

To/From	AE	Address1	Address2	Address3	Address4	Address5	Address6
11	10	STA6	STA4	STA2	STA1	STA22	STA33

To/From	AE	Address1	Address2	Address3	Address4	Address5	Address6
11	10	STA2	STA6	STA2	STA1	STA22	STA33

Figure 1.3 Mesh addressing example (STA33 >> STA22)

Error! Reference source not found. shows a mesh addressing example. Address5 and Address6 include DS addresses.

1.3 Mesh Discovery

As shown in **Error! Reference source not found.**, mesh STAs broadcast beacon with own mesh profile including mesh configuration element. Each mesh STA registers neighbor mesh STAs as candidates for peering if the received mesh profile is matched with own mesh profile.

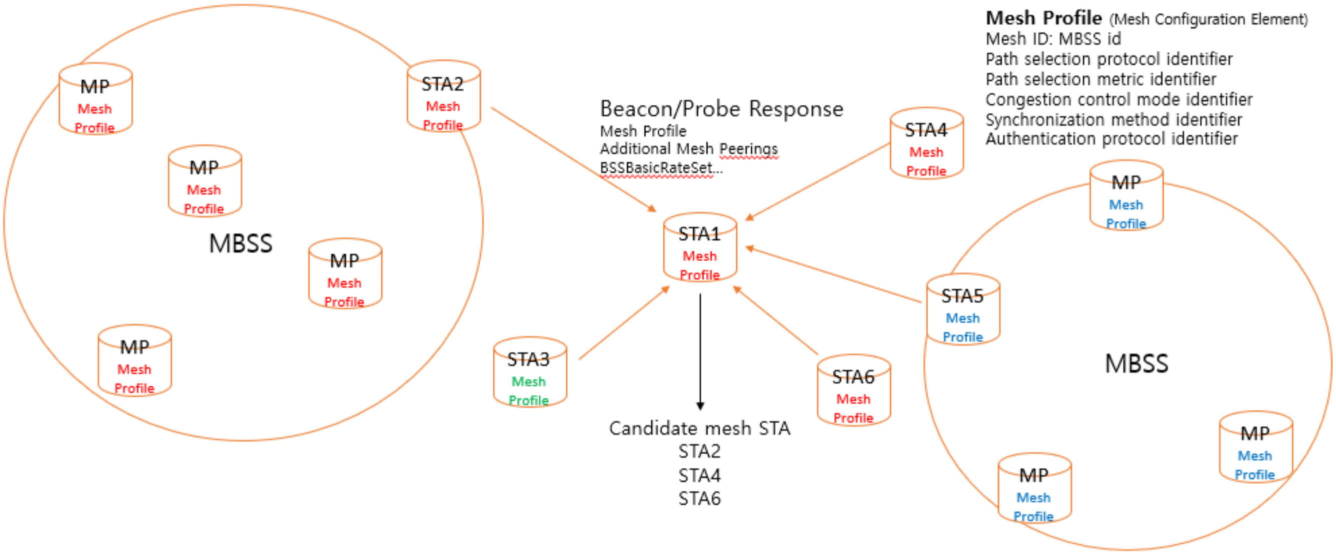

MP
Mesh Profile
MBSS
STA2
STA1
STA3
STA4
STA5
STA6
Beacon/Probe Response
Mesh Profile
Additional Mesh Peerings
BSSBasicRateSet...
Mesh Profile (Mesh Configuration Element)
Mesh ID: MBSS id
Path selection protocol identifier
Path selection metric identifier
Congestion control mode identifier
Synchronization method identifier
Authentication protocol identifier
Candidate mesh STA
STA2
STA4
STA6

Figure 1.4 Mesh Discovery

1.4 Mesh Peering

To connect mesh network, the mesh STA should succeed in peering handshake. Peering shall be completed only if a 'Mesh Peering Open' frame is sent/received and a 'Mesh Peering Confirm' is received. If the AMPE protocol is used for peering, the SAE authentication is performed before. **Error! Reference source not found.** shows this process shortly.

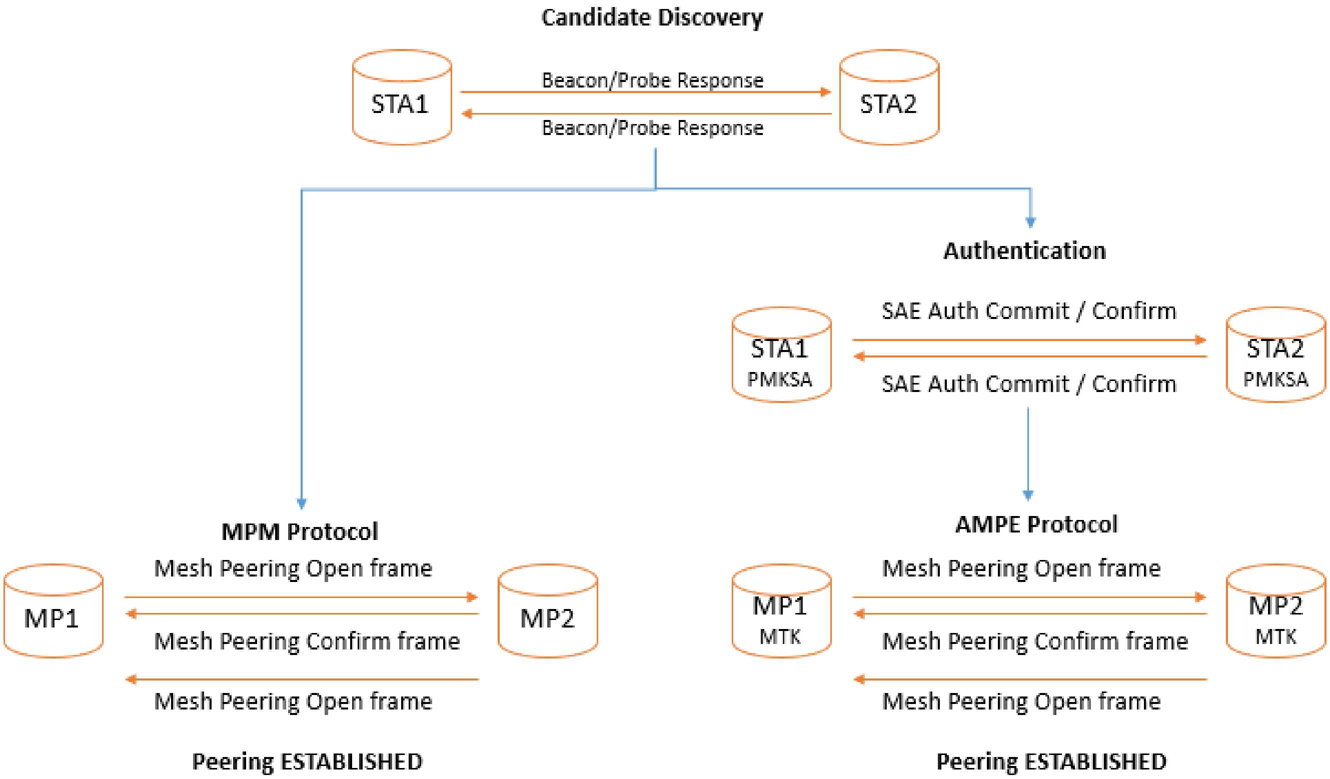


Figure 1.5 Mesh Peering

1.5 Hybrid Wireless Mesh Protocol (HWMP)

HWMP is a default mesh path selection protocol.

- Path originator: The mesh STA that triggers the path discovery.
- Path Target: The entity to which the path originator attempts to establish a path.
- HWMP Sequence Number: Each mesh HWMP path selection element contains an HWMP seuquence number.
- Forwarding information: allows the mesh STA to perform its path selection and forwarding functions.
  - Destination Address
  - Next-hop Address
  - Path Metric
  - Lifetime, HWMP SN, Hop Count

- Element TTL: an integer number that is used to limit the number of hops an HWMP element may be processed and propagated.
- Airtime link metric: default metric formula.
  - $C_a = \left[O + \frac{B_t}{r}\right]\frac{1}{1-e_f}$
  - O : PHY Header Overhead Time
  - r : Transmission Time
  - B_t : Number of bits in test frame
  - $\frac{1}{1-e_f}$: Error Adjustment
- PREQ element: discovering/maintenance/confirming a path to one or more target mesh STAs
- PREP element: establish a forward path to a target and confirm that a target is reachable
- PRRR element: announcing an unreachable destination

Error! Reference source not found. shows a simple HWMP process. The originator A broadcasts PREQ to neighbors propagating it consecutively. If PREQ is reached the target H, it updates its forwarding information for the originator A and reply the PREP to the originator A through the updated path. If the originator A receives a PREP, it updates its forwarding information for the target H. Now the originator A has the best path to the target H.

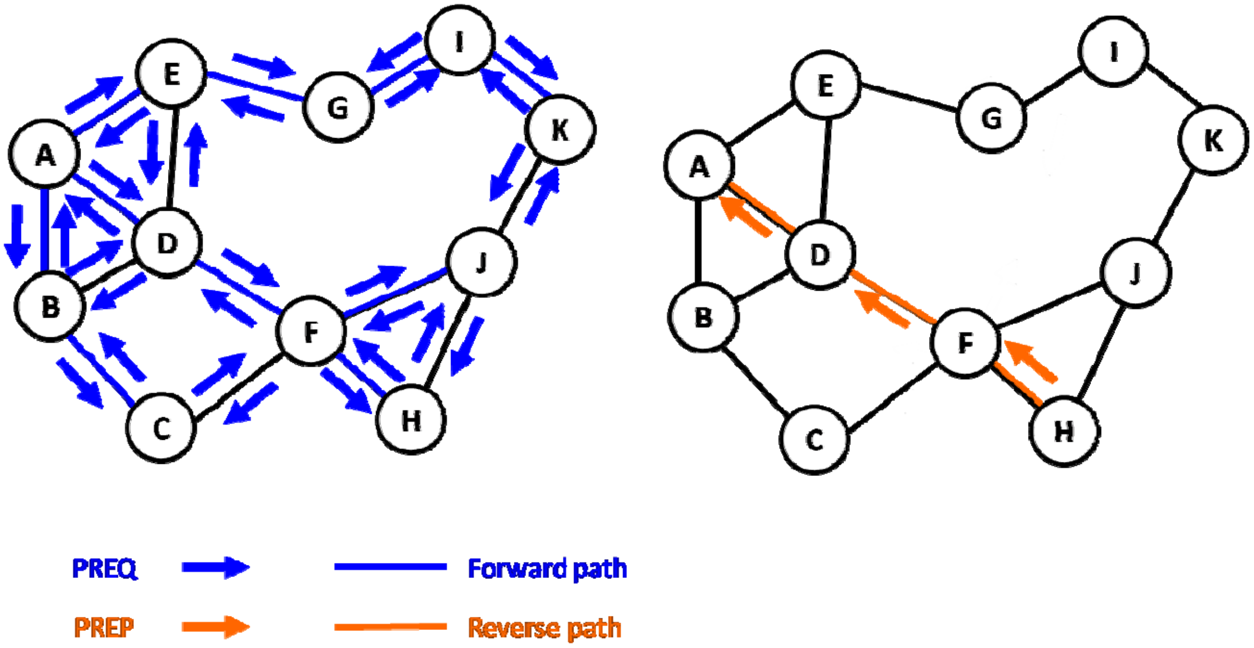
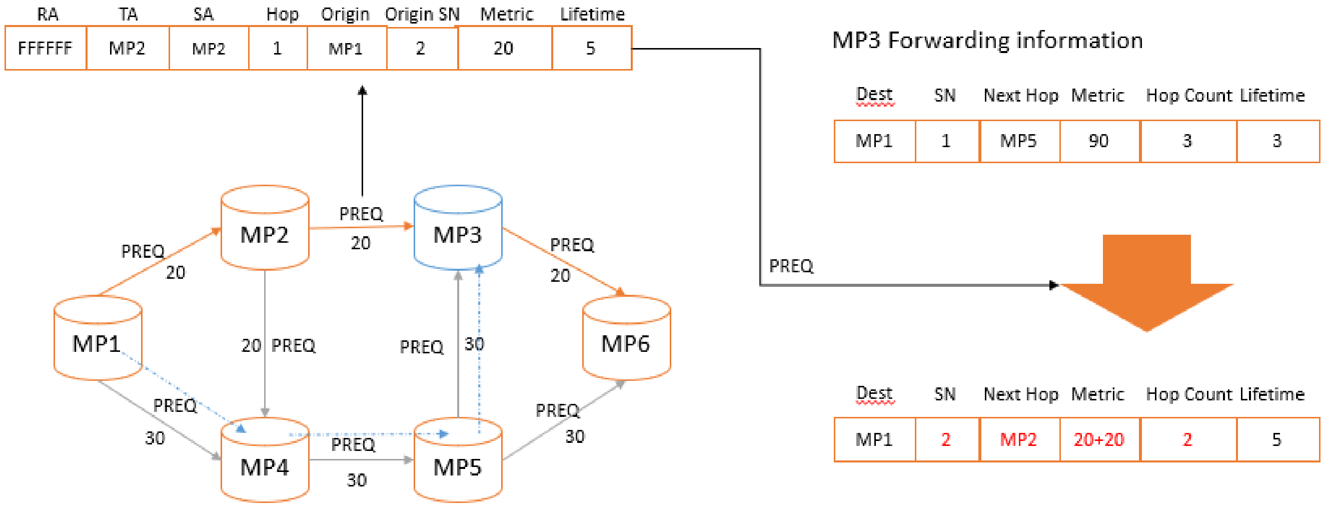


Figure 1.6 HWMP

Metric is a cumulated airtime value from the metric owner to the target. As shown examples in **Error! Reference source not found.** and **Error! Reference source not found.**, the forwarding information will be updated if the new metric is better than before.



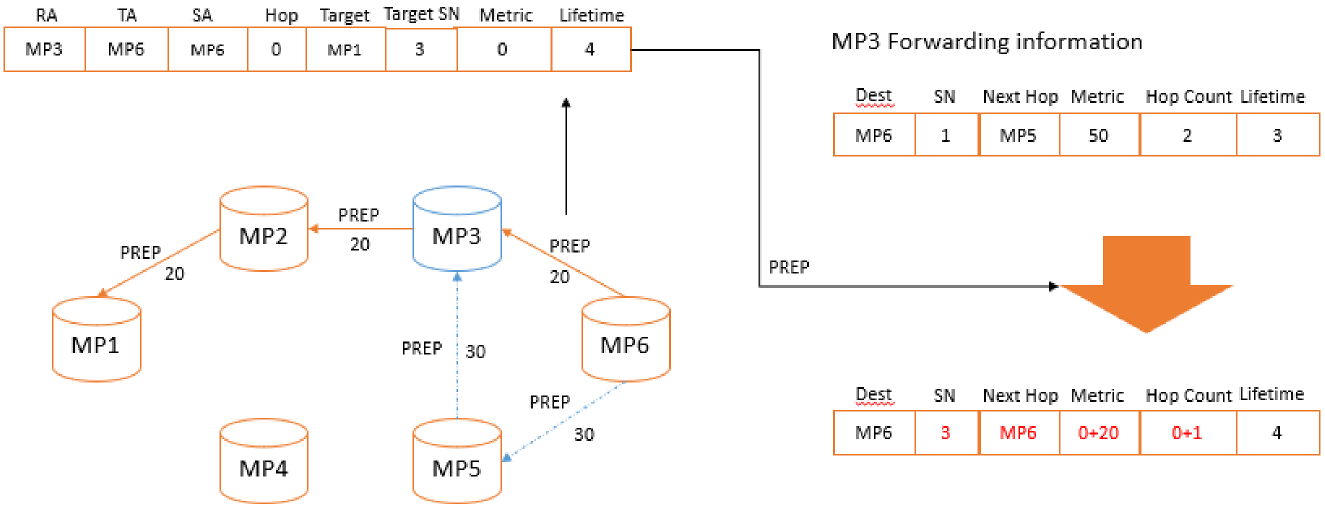
RA	TA	SA	Hop	Origin	Origin SN	Metric	Lifetime
FFFFFF	MP2	MP2	1	MP1	2	20	5

MP3 Forwarding information

Dest	SN	Next Hop	Metric	Hop Count	Lifetime
MP1	1	MP5	90	3	3

Dest	SN	Next Hop	Metric	Hop Count	Lifetime
MP1	2	MP2	20+20	2	5

Figure 1.7 Update forwarding information in PREQ



RA	TA	SA	Hop	Target	Target SN	Metric	Lifetime
MP3	MP6	MP6	0	MP1	3	0	4

MP3 Forwarding information

Dest	SN	Next Hop	Metric	Hop Count	Lifetime
MP6	1	MP5	50	2	3

Dest	SN	Next Hop	Metric	Hop Count	Lifetime
MP6	3	MP6	0+20	0+1	4

Figure 1.8 Update forwarding information in PREP

1.6 Mesh Procedure Example

Here is a general mesh procedure example.

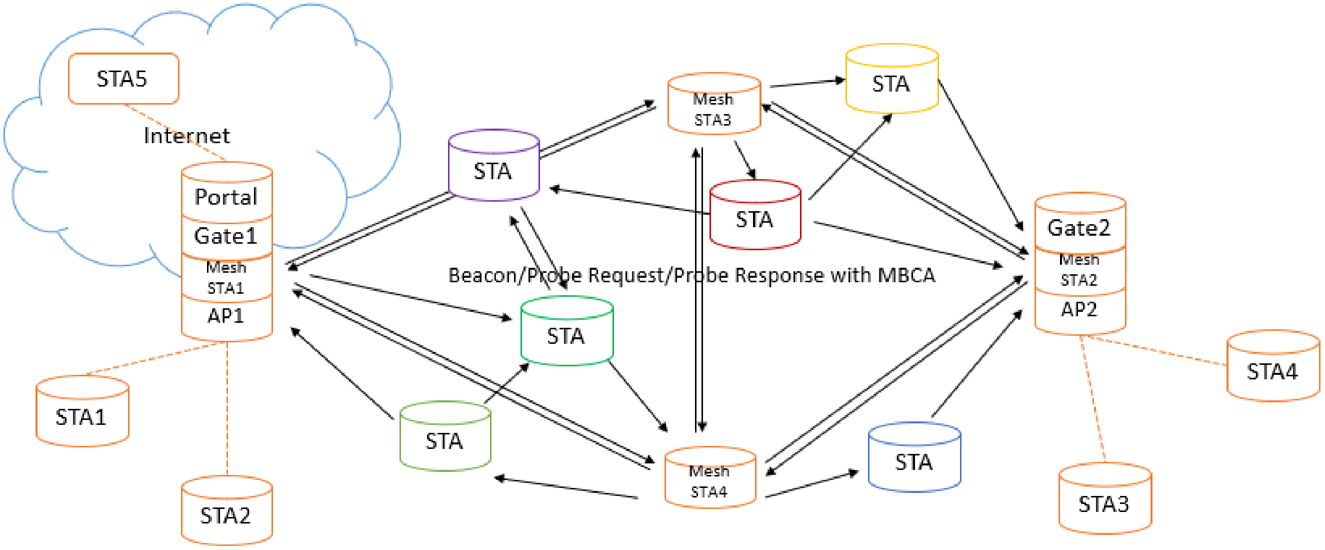


Figure 1.9 Scanning to find candidates for peering

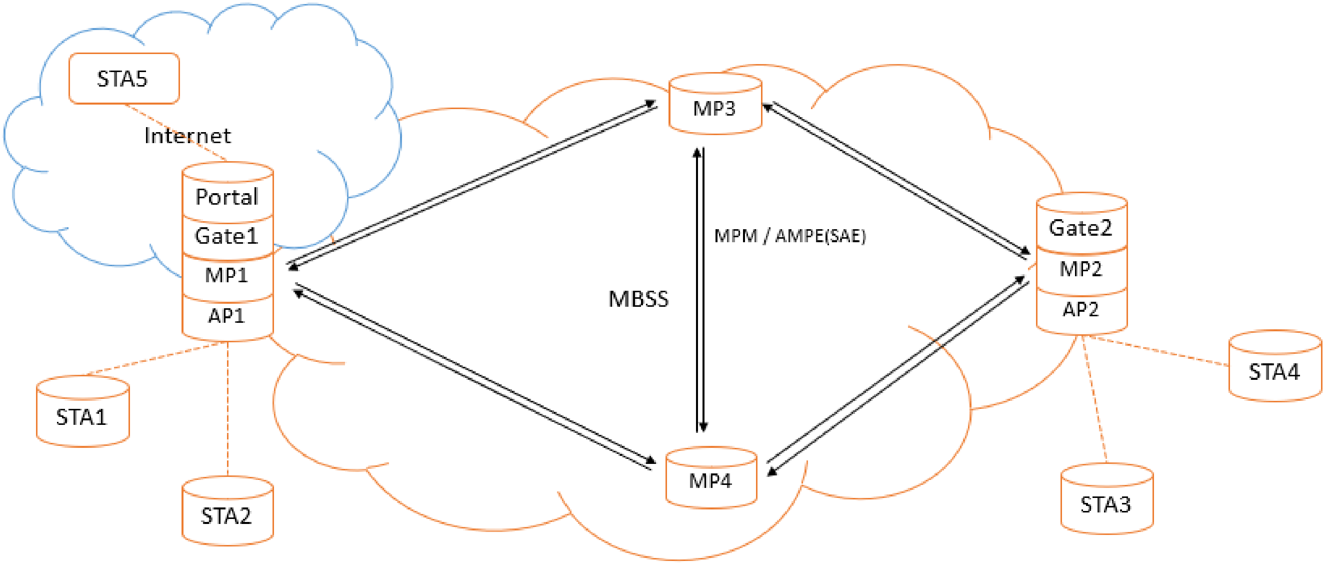


Figure 1.10 Peering with candidates

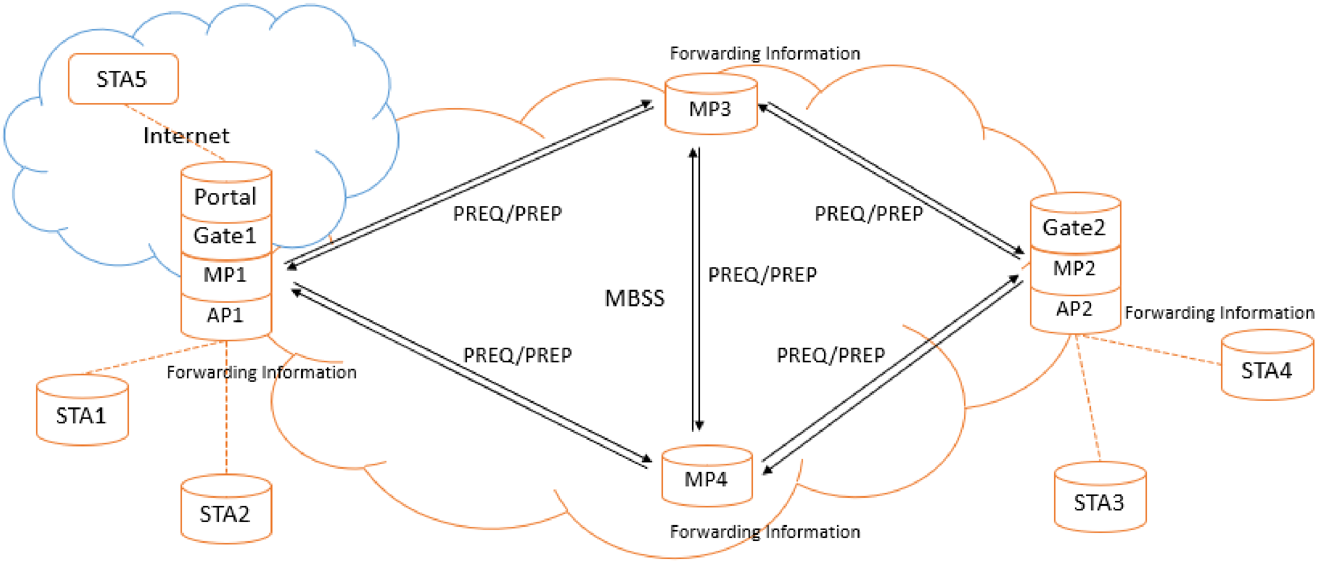


Figure 1.11 Update forwarding information via HWMP

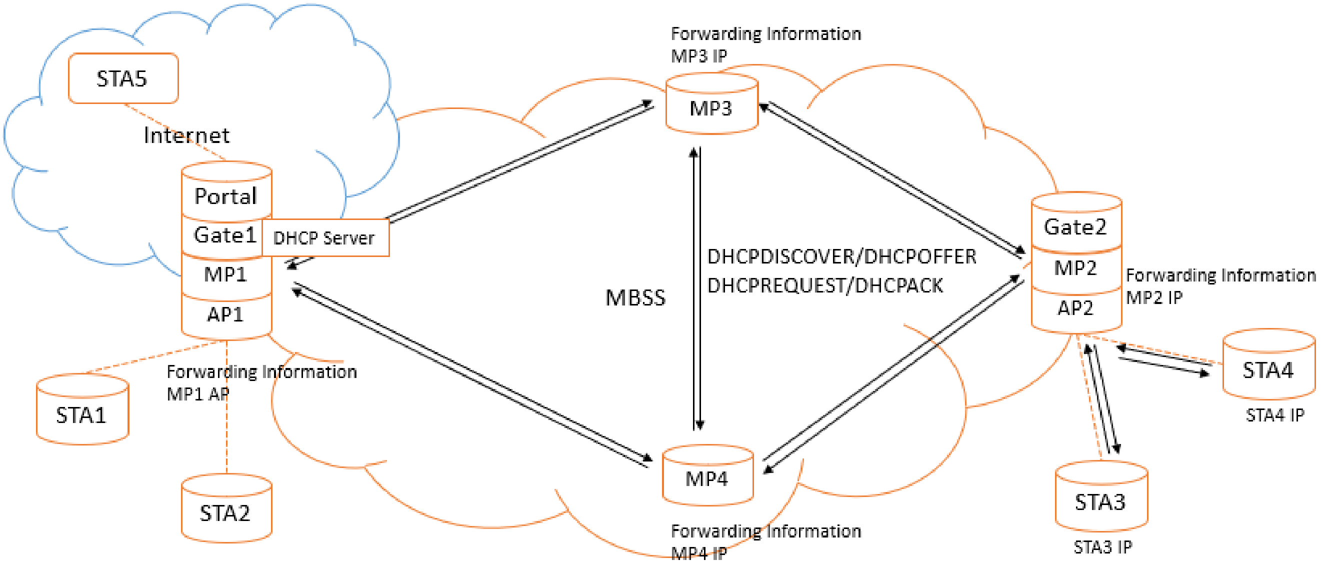


Figure 1.12 IP address allocation from DHCP server in Mesh Portal

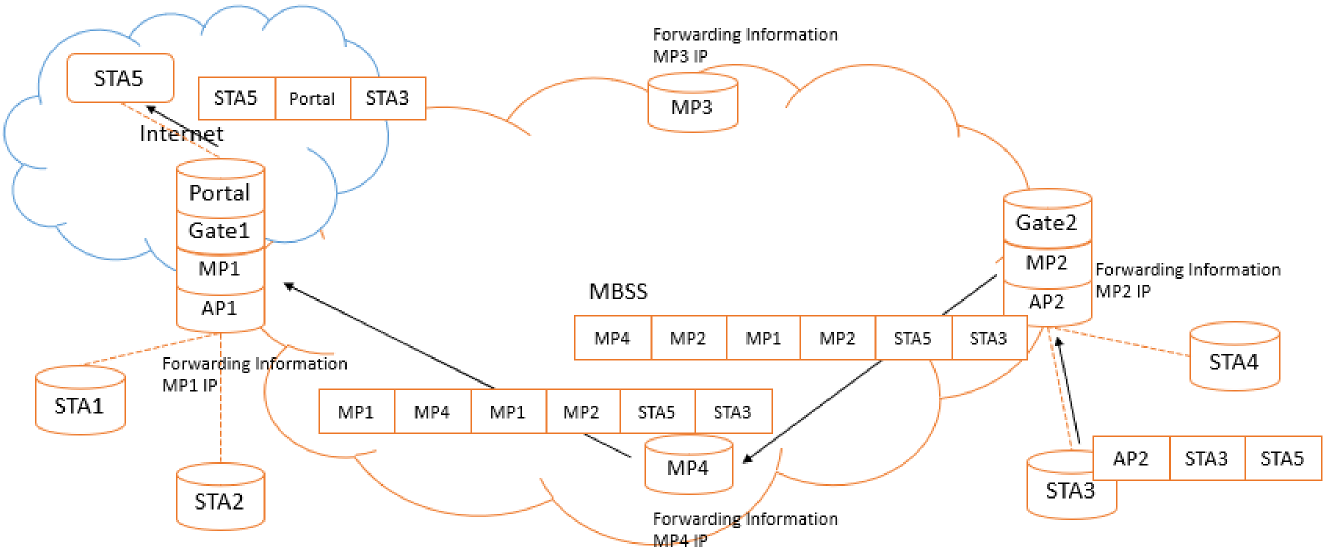


Figure 1.13 Data transmission as the forwarding information

2 NRC7292 EVK 11s mesh operation

By using different class of 11s mesh device, the user can configure the wireless ad-hoc network or relatively fixed network.

These contents in this clause contains commands to run the 11s mesh devices with NRC7292 EVKs including steps to build wpa_supplicant and establish links among MPs.

2.1 Prerequisite: enable CONFIG_MESH

There's no dependency on the wpa_supplicant version, but it's recommended to use version 2.9 to enable WFA3-Personal(SAE) and CONFIG_MESH.

```
$ wget https://w1.fi/releases/wpa_supplicant-2.9.tar.gz
$ tar zxf wpa_supplicant-2.9.tar.gz
$ sudo apt-get update
$ sudo apt-get install libnl-3-dev libnl-genl-3-dev libssl-dev
$ sudo apt-get install libdbus-1-dev libdbus-glib-1-dev
$ cd wpa_supplicant-2.9/wpa_supplicant
$ cp defconfig .config

Edit .config to include below:
CONFIG_IEEE80211N=y
CONFIG_OWE=y
CONFIG_SAE=y
CONFIG_MESH=y
CONFIG_IEEE80211W=y

$ make -j4
$ sudo make install
```

2.2 Prerequisite: install bridge tools

Bridge tools should be installed as below.

```
$ sudo apt-get install bridge-utils
```

2.3 Run script: start.py

The start.py can be found in nrc_pkg and following is the usage of start.py script.

- nrc_pkg/script/start.py
- start.py [sta_type] [security_mode] [country] [mesh_mode]
- sta_type : 4 (Mesh)
- security mode : 0 (Open), 3(WPA3-SAE)
- country : US, JP, KR, CN, EU, TW
- mesh_mode : 0 (MPP), 1(MP), 2(MAP)

2.4 MP (Mesh Point)

MP runs on wpa_supplicant with mesh mode (5).

dhcpcd allocates IP address from DHCP server (MPP).

You can modify wpa_supplicant configuration file to change the SSID & channel.

- nrc_pkg/script/conf/[country]/mp_halow_[security_mode].conf

```
$ ./start.py 4 0|3 US|JP|KR|CN|EU|TW 1
```

```
      ctrl_interface=/var/run/wpa_supplicant

      country=US
      update_config=1
      network={
         ssid="nrc_mesh"
         mode=5
         scan_ssid=1
         proto=RSN
         key_mgmt=SAE
         pairwise=CCMP
         group=CCMP
         psk="12345678"
         ieee80211w=2
         beacon_int=100
         frequency=5795
         freq_list=5795
         scan_freq=5795
         dot11MeshRetryTimeout=1000
         dot11MeshHoldingTimeout=400
         dot11MeshMaxRetries=4
         mesh_rssi_threshold=0
         mesh_basic_rates= 60 120 240
         #no_auto_peer=1
      }
      p2p_disabled=1
      ignore_old_scan_res=1
```

2.5 MPP (Mesh Portal)

MPP has DHCP server using dnsmasq.

NAT, IP forwarding is needed to connect with Internet.

You can modify wpa_supplicant configuration file to change the SSID & channel.

- nrc_pkg/script/conf/[country]/mp_halow_[security_mode].conf

```
$ ./start.py 4 0|3 US|JP|KR|CN|EU|TW 0
```

```
Use same configuration files with MP's (mp_halow_sae.conf, mp_halow_open.conf)
```

2.6 MAP (Mesh AP: MP + AP)

MAP is composed of a MP interface and an AP interface.

The key management of AP interface can be changed to SAE, WPA2 or OWE.

- 0 (Open), 1 (WPA2-PSK), 2 (WPA3-OWE), 3 (WPA3-SAE)

You can modify wpa_supplicant configuration file to change the SSID & channel.

- nrc_pkg/script/conf/[country]/mp_halow_[security_mode].conf
- nrc_pkg/script/conf/[country]/map_halow_[security_mode].conf

```
$ ./start.py 4 0|1|2|3 US|JP|KR|CN|EU|TW 2
```

```
MP interface use same files with MP's (mp_halow_sae.conf, mp_halow_open.conf)
AP interface use map_halow_sae.conf, map_halow_open.conf, map_halow_owe.conf, map_halow_wpa2.conf

      ctrl_interface=/var/run/hostapd
      country_code=US
      interface=wlan0
      ssid=nrc_mesh
      hw_mode=a
      bridge=br0
      basic_rates=60 120 240
      channel=159
      ieee80211h=1
      ieee80211d=1
      ieee80211n=1
      ieee80211w=2
      #wmm_enabled=1
      macaddr_acl=0
      driver=nl80211
      beacon_int=100
      #disable_sgi=0
```

```
       #ap_max_inactivity=3
       ap_max_inactivity=16779
       # USF=0, UI=2 : 3
       # USF=1, UI=2 : 16780
       # USF=0, UI=10: 11
       # USF=1, UI=10: 16788
       #listen_interval=2

       #WPA3-SAE
       #auth_alg=1 #1=wpa, 2=wep, 3=open
       wpa=2     # wpa2 only
       wpa_key_mgmt=SAE
       wpa_pairwise=CCMP
       rsn_pairwise=CCMP
       wpa_passphrase=12345678
```

2.7 STA

Use same SSID & channel with mesh network.

- nrc_pkg/script/conf/[country]/sta_halow_[security_mode].conf

```
$ ./start.py 0 0|1|2|3 US|JP|KR|CN|EU|TW
```

2.8 Static IP

Static IP setting is possible like below.

- start.py [sta_type] [security_mode] [country] [mesh_mode] [mesh_ip]

```
$ ./start.py 4 3 US 0 192.168.222.1       # SAE mode Mesh Portal with Static IP
$ ./start.py 4 0 US 2 192.168.222.2       # Open mode Mesh AP with Static IP
```

2.9 Manual Peering

Mesh node do peering automatically by default but you can disable it and connect manually.

- start.py [sta_type] [security_mode] [country] [mesh_mode] [peer mac address]

```
$ ./start.py 4 3 US 1 8c:0f:fa:00:29:46   # SAE mode Mesh Point trying to peer only 8c:0f:fa:00:29:46
$ ./start.py 4 0 US 2 192.168.222.2       # Open mode Mesh AP trying to peer only 8c:0f:fa:00:29:46
```

Both Static IP & Manual Peering are possible in one command like below.

- start.py [sta_type] [security_mode] [country] [mesh_mode] [peer mac address] [mesh_ip]

```
# SAE mode Mesh Point trying to peer only 8c:0f:fa:00:29:46 and set static IP 192.168.222.2
$ ./start.py 4 3 US 1 8c:0f:fa:00:29:46 192.168.222.2
```

3 NRC7292 EVK 11s mesh test procedure

The contents in this clause contains steps to compose 11s mesh network with NRC7292 EVKs.

3.1 Setup Mesh Network

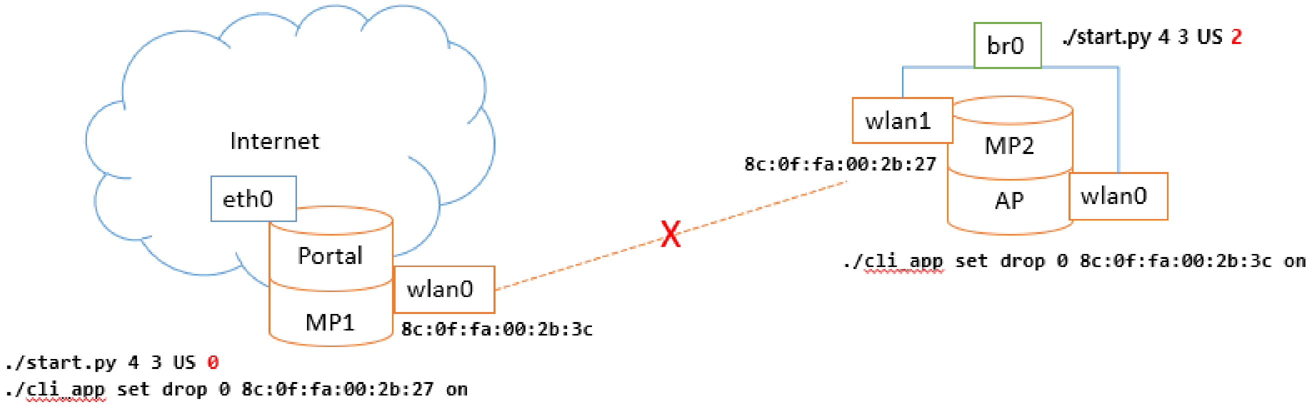
3.1.1 Mesh Distancing

NRC7292 has very wide coverage so it is too hard to setup each MP to distance physically.

Following CLI can set the MAC addresses to drop all packets from it.

```
Usage: set drop <vif_id> <mac address> <on|off>
./cli_app set drop 0 8c:0f:fa:00:2b:3c on
DROP a frame from 8c:0f:fa:00:2b:3c
DROP a frame from 8c:0f:fa:00:2b:3c
DROP a frame from 8c:0f:fa:00:2b:3c
DROP a frame from 8c:0f:fa:00:2b:3c
```

By setting it both sides, they runs like as if they are apart despite physically close distance.



```
./start.py 4 3 US 2
./cli_app set drop 0 8c:0f:fa:00:2b:3c on
./start.py 4 3 US 0
./cli_app set drop 0 8c:0f:fa:00:2b:27 on
```

Figure 3.1 MPP & MAP configuration with Mesh Distancing

Now MPP & MAP is configured but they cannot see mutual Beacon.

3.1.2 Connecting MP and STA

Establish the mesh network connecting one or more MP to MPP and MAP.

Finally connect one or more STA to MAP then a test topology is completely set up.

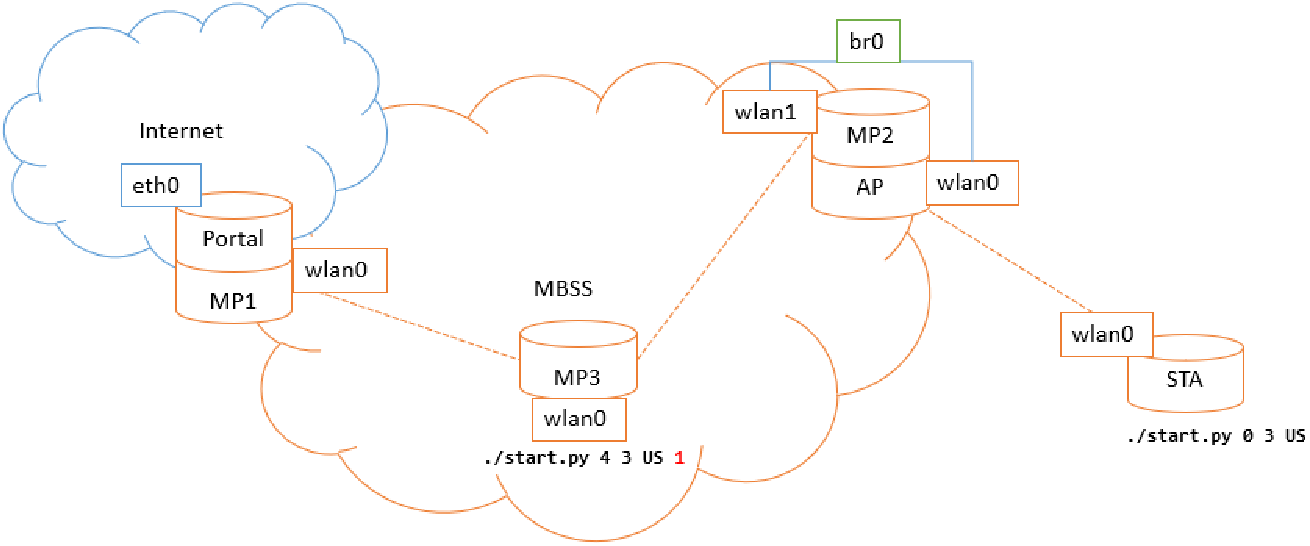


Figure 3.2 Mesh test topology

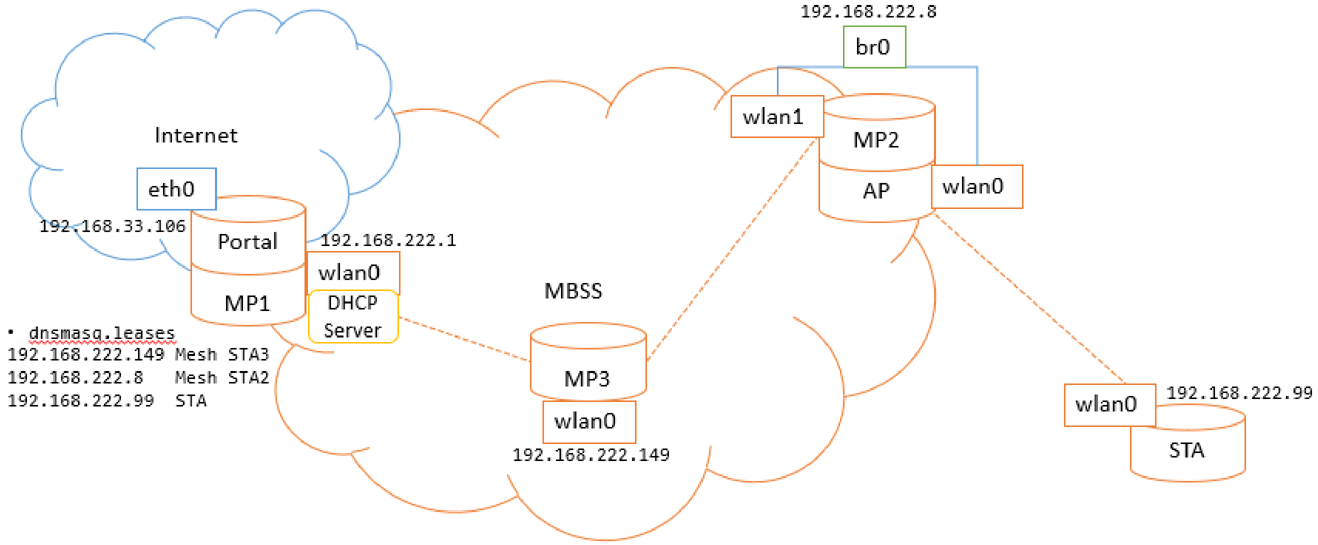
3.2 IP Allocation with DHCP

MPP is a DHCP server for the mesh network and default gateway is 192.168.222.1.

You can change this address with -a option in the run_mesh.sh script.

MAP is a DHCP server for the AP-STA network and default gateway is 192.168.200.1.

You can change this address with -b option in the run_mesh.sh script.



```
• dnsmasq.leases
192.168.222.149 Mesh STA3
192.168.222.8   Mesh STA2
192.168.222.99  STA
```

Figure 3.3 IP allocation of mesh network

3.3 Test commands

3.3.1 Ping

STA can communicate with internet server.

```
$ ping google.com
```

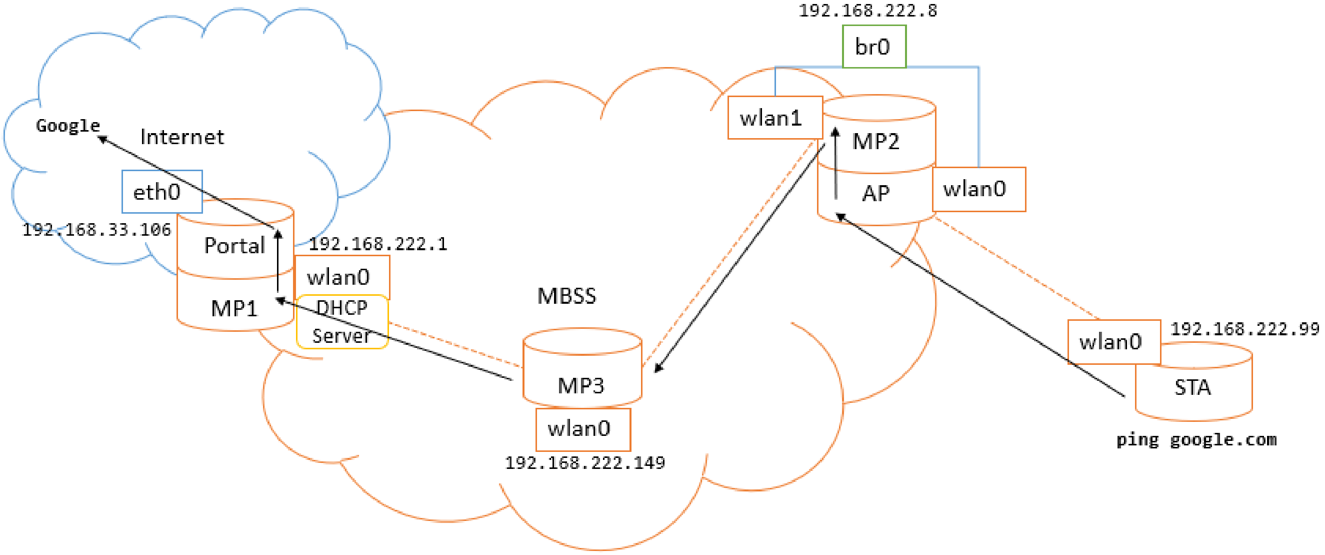


Figure 3.4 ping google.com from STA

MPP can communicate with STA.

```
$ ping 192.168.222.99
```

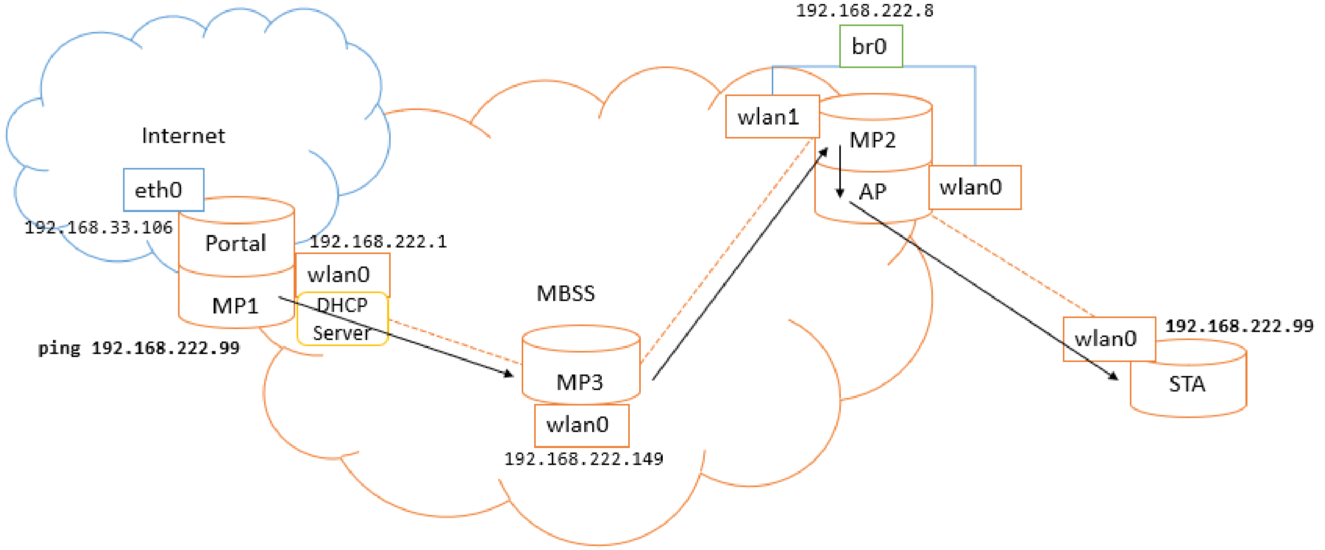


Figure 3.5 ping [STA IP] from MPP

3.3.2 Mesh status check

Following two commands can be used to check the 11s mesh connection status.

- To see a list of mesh paths: sudo iw dev wlan0 mpath dump
- To get current WPA status: sudo wpa_cli status

```
$ sudo iw dev wlan0 mpath dump
DEST ADDR         NEXT HOP          IFACE    SN    METRIC  QLEN    EXPTIME    DTIM   DRET    FLAGS
02:00:eb:cf:0e:d4 02:00:eb:cf:0e:d4 wlan0    15    1377     0      3590       0      0       0x15
02:00:eb:98:eb:25 02:00:eb:98:eb:25 wlan0    1     1377     0      3600       0      0       0x15
$ sudo wpa_cli status
Selected interface 'wlan0'
bssid=00:00:00:00:00:00
freq=5795
ssid=halow_mesh
id=0
mode=mesh
pairwise_cipher=UNKNOWN
group_cipher=UNKNOWN
key_mgmt=UNKNOWN
wpa_state=COMPLETED
ip_address=192.168.201.2
address=00:01:02:03:04:14
uuid=1aa91a70-1281-53aa-9a1d-ce76cc1f0d70
```

4 Appendix

4.1 Peering Timer

As shown in **Error! Reference source not found.** and **Error! Reference source not found.**, Too many MP neighbors cause fail to peering.

- No Confirm frame received in dot11MeshRetryTimeout x dot11MeshMaxRetries

  (default 40ms x 2 = 80ms)
- No Open frame received in dot11MeshHoldingTimeout (default 40ms)
- 2 Mesh Point → 2 Open frames
- 3 Mesh Point → 6 Open frames
- 4 Mesh Point → 12 Open frames
- n Mesh Point →nPn-1 Open frames

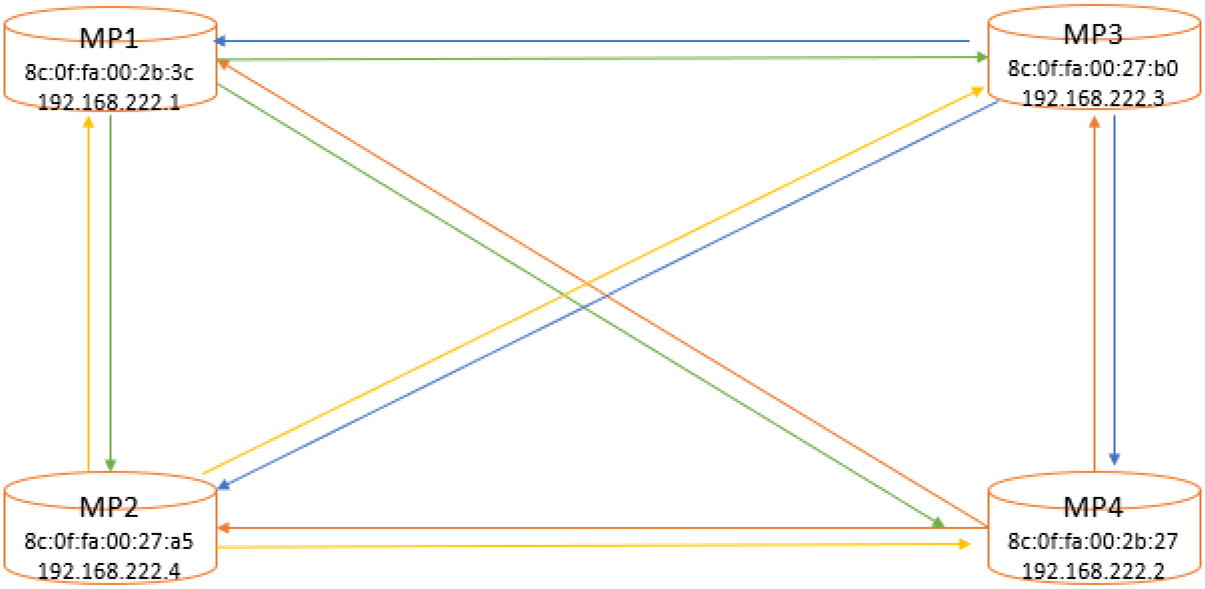


Figure 4.1 4 MP need min 12 Open frames

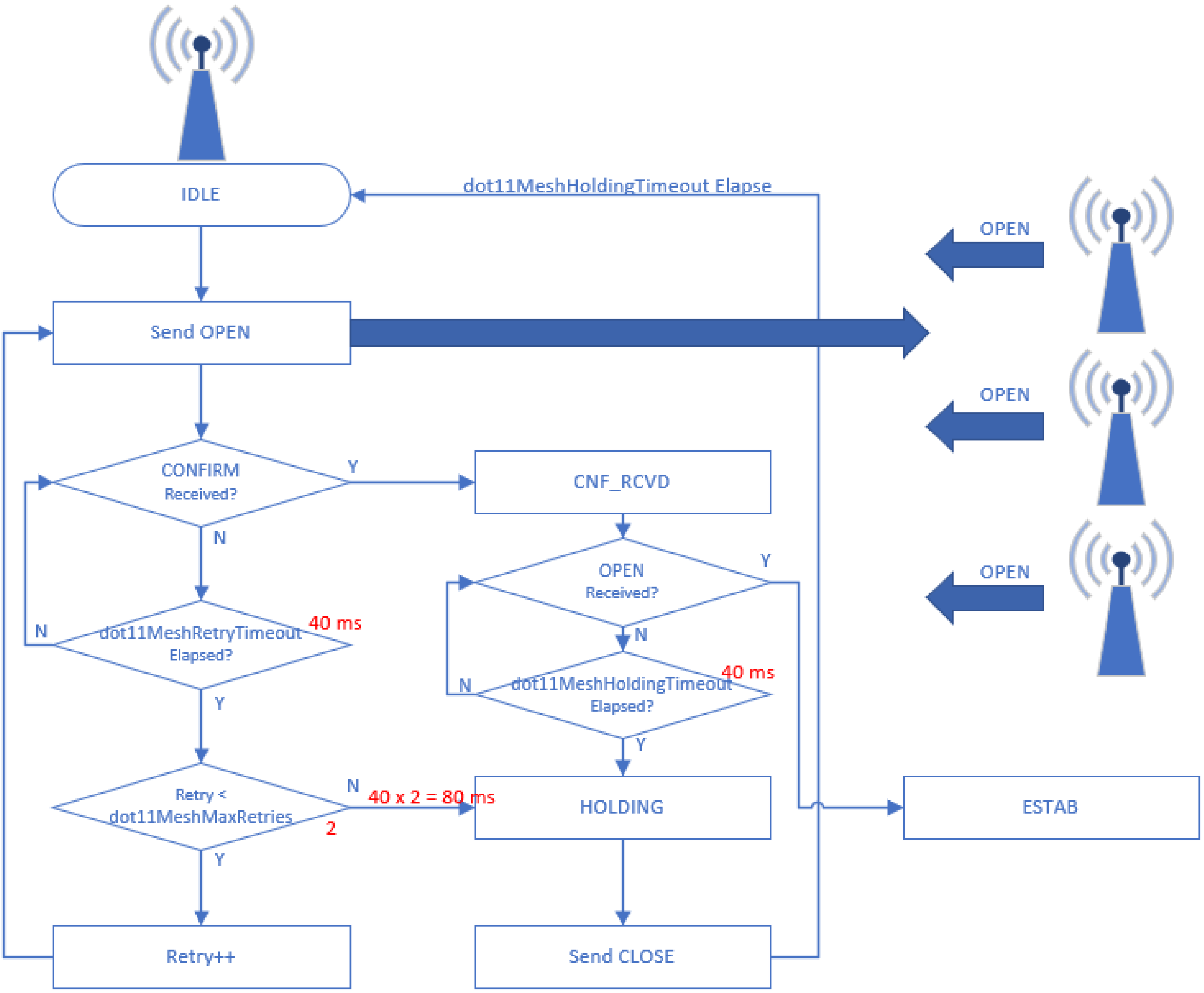


Figure 4.2 Peering timeout flow

To establish the mesh network, MPM frames must be received in the peering timeout so we need to increase this timeout value for more peers.

- dot11MeshRetryTimeout=40 (default) → 1000
- dot11MeshHoldingTimeout=40 (default) → 400
- dot11MeshMaxRetries=2 (default) → 4

4.2 RSSI Threshold

To prevent MP from peering with weak signal, you can change rssi_threshold setting in the script.

- Default: 0 (disable rssi_threshold setting)

```
@ieee80211_mesh_rx_bcn_presp (mac80211/mesh.c)
      if (mesh_matches_local(sdata, &elems)) {
             mpl_dbg(sdata, "rssi_threshold=%d rx_status->signal=%d from %pM\n",
                sdata->u.mesh.mshcfg.rssi_threshold, rx_status->signal, mgmt->sa);
             if (!sdata->u.mesh.user_mpm ||
                sdata->u.mesh.mshcfg.rssi_threshold == 0 ||
                sdata->u.mesh.mshcfg.rssi_threshold < rx_status->signal)
                mesh_neighbour_update(sdata, mgmt->sa, &elems);
      }
```

```
write_config_mp() {
          local FILENAME=$1
       cat << EOF > $FILENAME
       ctrl_interface=/var/run/wpa_supplicant
       country=US
       network={
          ssid="${SSID0}"
          mode=5
          frequency=${FREQ}
          key_mgmt=${KEY_MGMT}
          psk="${PASSWORD0}"
          #no_auto_peer=1
          dot11MeshRetryTimeout=1000
          dot11MeshHoldingTimeout=400
          dot11MeshMaxRetries=4
          mesh_rssi_threshold=0
       }
       p2p_disabled=1
       EOF
 }
```

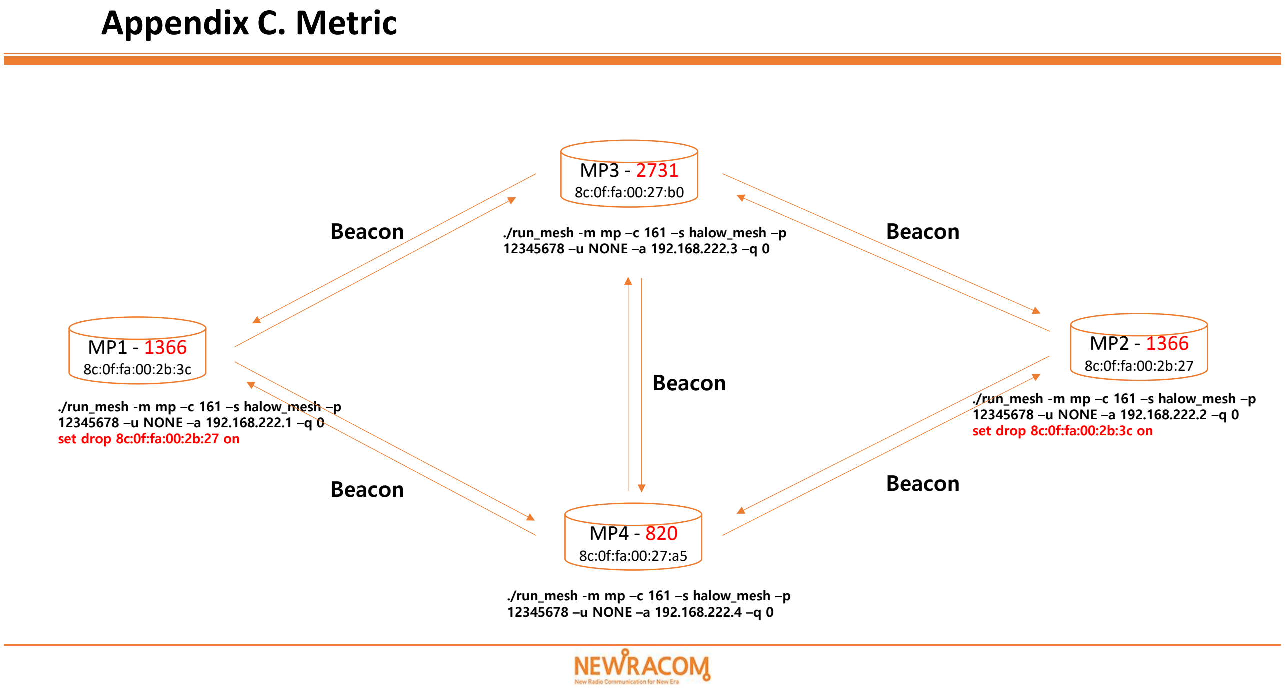
4.3 Changing Metric

Implement 'get_expected_throughput' of ieee80211_ops in NRC driver.

```
@airtime_link_metric_get (mac80211/mesh_hwmp.c)
#define TEST_FRAME_LEN  8192
#define MAX_METRIC  0xffffffff
#define ARITH_SHIFT 8
          rate = DIV_ROUND_UP(sta_get_expected_throughput(sta), 100);
          if (rate) {
              err = 0;
          } else {
              if (fail_avg > LINK_FAIL_THRESH)
                  return MAX_METRIC;

              sta_set_rate_info_tx(sta, &sta->tx_stats.last_rate, &rinfo);
              rate = cfg80211_calculate_bitrate(&rinfo);
              if (WARN_ON(!rate))
                  return MAX_METRIC;

              err = (fail_avg << ARITH_SHIFT) / 100;
          }

          /* bitrate is in units of 100 Kbps, while we need rate in units of
           * 1Mbps. This will be corrected on tx_time computation.
           */
          tx_time = (device_constant + 10 * test_frame_len / rate);
          estimated_retx = ((1 << (2 * ARITH_SHIFT)) / (s_unit - err));
          result = (tx_time * estimated_retx) >> (2 * ARITH_SHIFT);
     return (u32)result;
```

- 10000Kbps → 820 Metric
- 6000Kbps (default) → 1366 Metric
- 3000Kbps → 2731 Metric
- The return of 'sta_get_expected_throughput' must be over 200Kbps due to the overflow at tx_time*estimated retx

Refer to this PPT to see HWMP process with Metric Change.



Appendix C. Metric

```
./run_mesh -m mp –c 161 –s halow_mesh –p
12345678 –u NONE –a 192.168.222.3 –q 0
```

```
./run_mesh -m mp –c 161 –s halow_mesh –p
12345678 –u NONE –a 192.168.222.1 –q 0
set drop 8c:0f:fa:00:2b:27 on
```

```
./run_mesh -m mp –c 161 –s halow_mesh –p
12345678 –u NONE –a 192.168.222.2 –q 0
set drop 8c:0f:fa:00:2b:3c on
```

```
./run_mesh -m mp –c 161 –s halow_mesh –p
12345678 –u NONE –a 192.168.222.4 –q 0
```

Figure 4.3 HWMP example with metric change

5 Revision history

Revision No	Date	Comments
Ver 1.0	11/06/2019	Initial version
Ver 1.1	03/31/2021	Update overview and test procedure
Ver 1.2	08/18/2021	Mesh AP Concurrent mode
Ver 1.3	09/16/2021	Change start script (start.py)