An IEEE 802.11g simulation model with extended debug capabilities

Sorin Cocorada

University Transilvania of Brasov
Romania

sorin.cocorada@unitbv.ro

March 7, 2008 Marseille

Summary

1. Introduction
2. Omnet++ simulation environment description
    1. Physical layer modeling
    2. Data link layer modeling
3. Debugging the model using Wireshark
4. Conclusion

1. Intoduction – why a 802.11g model

Communication protocols running at the data link layer are mostly implemented in hardware/firmware because of the severe timing requirements (amounts of micro-seconds). Under these circumstances, modifying or extending existing protocols may be difficult or even impossible.

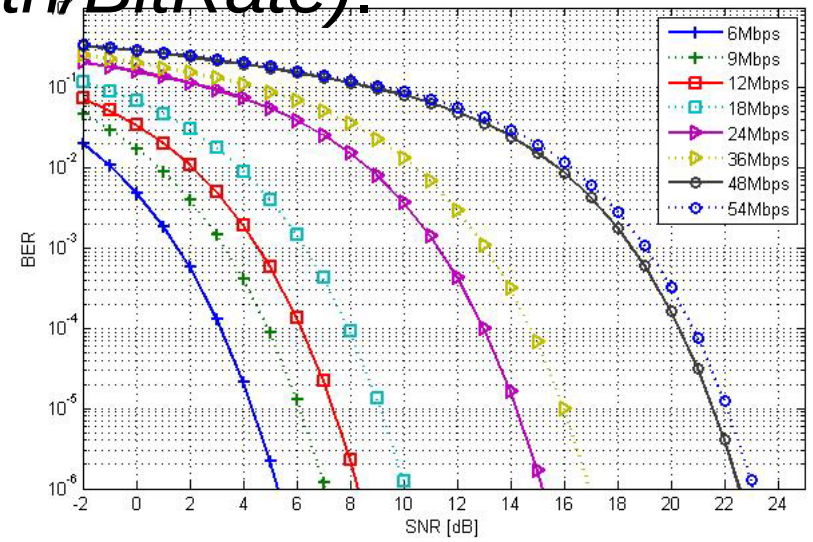
2.1 Physical layer modeling (AWGN channel)

$$P_b^{(2^k)}(\gamma) = \frac{2^{2-\frac{k}{2}}}{k}\left(Q\left(\sqrt{\gamma}\right) + Q\left(3\sqrt{\gamma}\right)\right) + \frac{k-2}{k} \cdot P_b^{2^{k-2}}(\gamma)$$

$$P_b^{(4)} = Q(\sqrt{\gamma})$$

2^k-QAM constellation,

$\gamma = 3k/(2^k-1) \cdot SNIR \cdot (Bandwidth/BitRate)$.

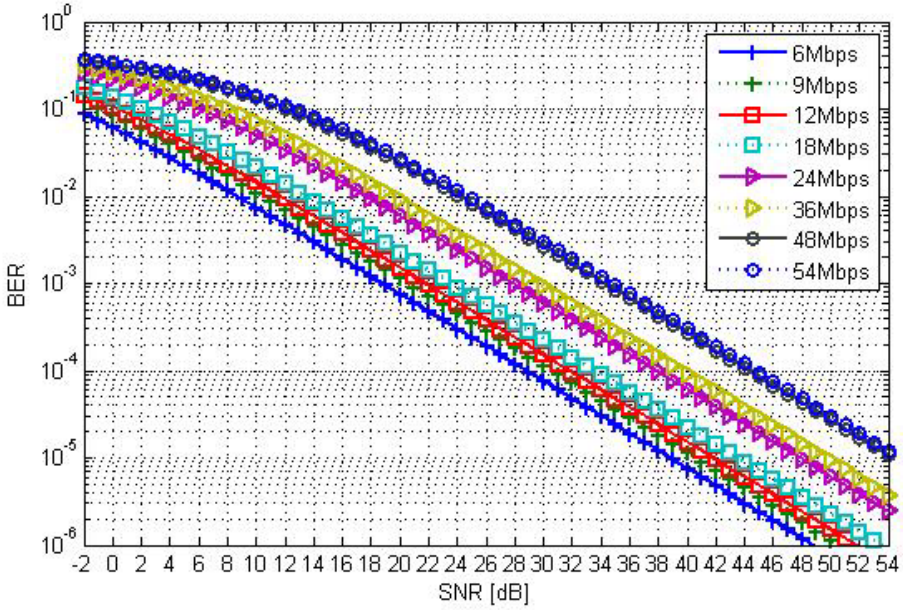


2.1 Physical layer modeling (Rayleigh channel)

$$P_b^{(4)} = \frac{1}{2}\left(1 - \sqrt{\frac{\gamma}{1+\gamma}}\right)$$

$$P_b^{(16)} = \frac{5}{8} - \frac{3}{8}\sqrt{\frac{2\gamma}{5+2\gamma}} - \frac{1}{4}\sqrt{\frac{18\gamma}{5+18\gamma}}$$

$$P_b^{(64)} = \frac{13}{24} - \frac{7}{24}\sqrt{\frac{\gamma}{7+\gamma}} - \frac{1}{4}\sqrt{\frac{9\gamma}{7+9\gamma}}$$



Viterbi decoding

We have used the upper bond probability of error under the assumption of binary convolutional coding and hard decision Viterbi decoding.

$$P_d = \sum_{i=(d+1)/2}^{d} C_i^d p^i (1-p)^{d-i}$$

$$P_d = 0.5 C_{d/2}^d p^{d/2} (1-p)^{d/2} + \sum_{i=d/2+1}^{d} C_i^d p^i (1-p)^{d-i}$$

$$P_b \le \sum_{d=d_{free}}^{\infty} a_d P_d$$

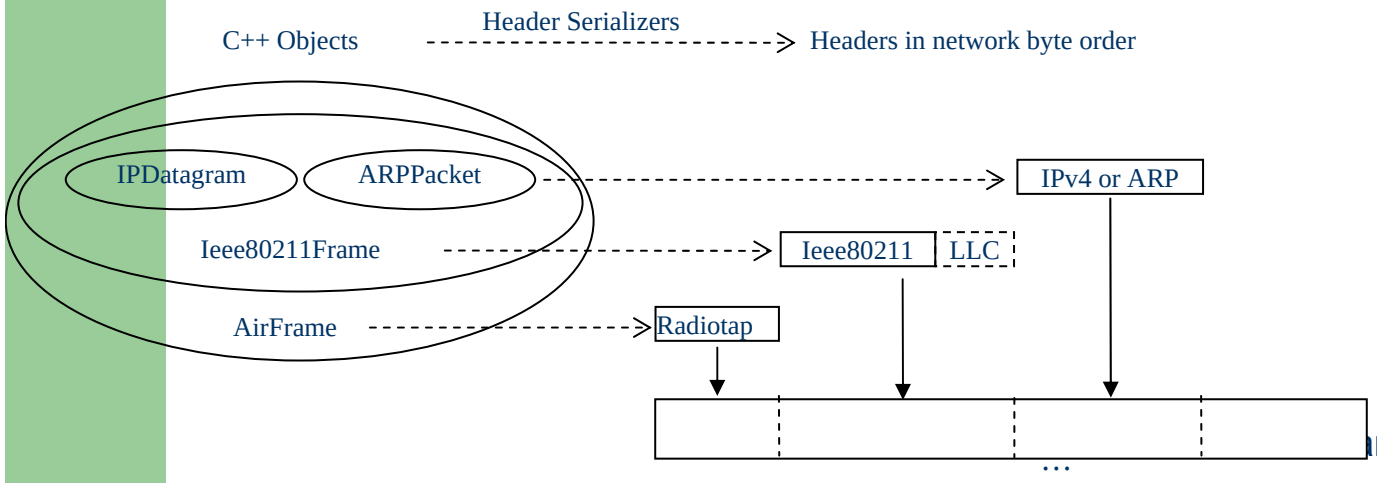
2.2 Data link layer modeling

The existing 802.11b FSM has been reused with few modifications to reflect the 802.11g parameters:

- slot time has been reduced from 20µs to 9µs but it can take any other value
- the duration field of the frames transmitted at 802.11g rates has been calculated using the formula provided in the standard

3. Debugging the model using Wireshark

Radiotap header	AirFrame
-MAC timestamp	-*simulation time*
-rate	-bitrate
-channel	-channelNumber
-transmit power (dBm)	-pSend



4. Conclusion

A complete IEEE 802.11g simulation model was presented along with a debugging solution using network protocol analyzers

In future works, we are going to investigate the possibility of using Omnet++ for transmitting IEEE 802.11 frames over the air