CHAPTER 5

SIMULATION RESULTS AND DISCUSSION

In this chapter, the analysis of the simulation results is discussed. Simulation results are presented in the form of table for various errors control algorithm examined under given network scenario. The definition of various error control strategies and channel conditions have been described in the previous chapter. All data used in experimentation was analyzed based on their performance respectively.

Through the observation of the overall results, this will guide us to develop an adaptive protocol in terms of what choices should be made if given a certain set of environmental conditions. Simulation results are grouped according to the type of data sources used to collect them, and discussions are made with regard to varying both error control strategy (FEC, ARQ) and the channel conditions. In our discussion below, we use the same parameter for the two types of test. As mentioned earlier, experiments were conducted with regards to two types of data; *data transmission* and *speech transmission.*

The rest of the chapter is organized as follows. In Section 5.1, we describe the simulation scenario, the types of test to be simulated and a little bit explanation about the test output parameters. In the following section (Section 5.2), we make an overview of the three types of error control algorithms to be tested. In Section 5.3, we analyze and discuss the result of the different types of test in Section 5.2, and finally the conclusion in the last section (Section 5.4).

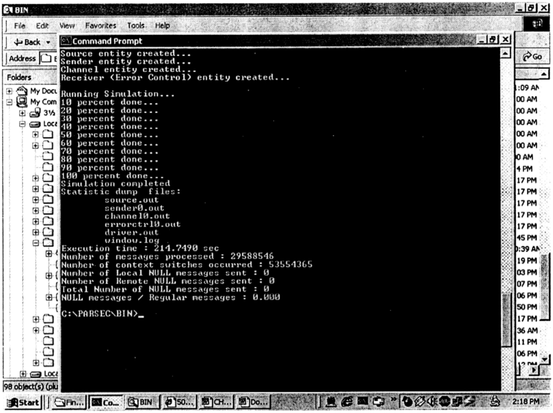

Figure 5.2: A Sample of Simulation System Interface

5.2 Overview of the Algorithms

The main objective of this project is to develops the three algorithms named non-adaptive error control with SACK (non-adaptive SACK), non-adaptive error control with FEC (non-adaptive FEC) and adaptive error control (adaptive tuning SACK/FEC) algorithms and compare their performance respectively. We also investigate the influence of mobility speed to these three algorithms. In this case, we used a pedestrian speed to represent slow fading and a car speed to represent fast fading for channel transition. For these experiments, we only measure the algorithms in term of packet loss percentage for each data and speech transmission.

As the analysis of simulation results are largely based on comparison of performance among the algorithms, all the algorithms is referred to by their number to avoid unnecessary repetition of the lengthy algorithm names, as well as for the sack of simplicity. The algorithm names and their respective algorithm number are listed as follows:

1. Non-adaptive error control algorithm (SACK alone)
2. Non-adaptive error control algorithm (FEC alone)
3. Adaptive error control algorithm (SACK + FEC code)

Algorithm 1 and 2 are the non-adaptive error control algorithms, which is exactly the same algorithm as proposed in (Lettieri, 97). Algorithm 3 is an extended algorithm which is proposed in this dissertation.

The algorithms basically perform the same computation and measurement, but differ in the way they estimate the error control scheme to be used in the next data transmission. The basic concept of the three algorithms are reviewed as follows:

Algorithm 1 : <u>**Non-Adaptive Error Control (SACK)**</u>

The algorithm will find the optimum output such as throughput and packet loss based on error control scheme (SACK) for certain channel condition. In term of wireless communication, SACK was chosen because this scheme tends to require the fewest transmissions for a set of lost packets.

Here, the optimum output can be defined as minimum packet loss and we can also investigate the simulation time depending on the data source type with different call arrival pattern and different mobility speed. Real-time traffic such as speech transmission will prefer minimum delay, while most traditional data application such as FTP application will prefer a maximum throughput solution.

Algorithm 2 : <u>**Non-Adaptive Error Control (FEC)**</u>

This algorithm perform job like the Algorithm 1 but differ in the way they choose the error control scheme to be used. In this case, FEC scheme such as Reed Solomon code will be used to test the simulation. Instead of SACK, this scheme was chosen in wireless data transmission because they have certain burst-error correcting capability. It is able to correct an exact amount of symbol errors depending on the number of redundancy symbols.

Algorithm 3: **Adaptive Error Control (tune between SACK and FEC)**

Basically, this algorithm requires no priori knowledge of the channel BER or channel conditions. The only knowledge necessary is a good estimate of the packet error probability to interpret the current channel BER. The algorithm learns and adapts its decision based on previous execution of data transmission. In another word, it will update their current information to be used as a new estimation error control for next job transmission. Here, it will use the estimated packet error rate not only to predict the channel condition, but also to determine the optimum error control should be used in the next transmission.

It needs a two-way communication between sender and receiver to request and indicate a new parameter setting which includes measurement of packet error rate and calculation of channel BER. If the algorithm has detected a change of the channel BER, then it will react to this change by modifying certain error control parameters and will used this error control scheme estimation for next data transmission. This can be simulated by using both forward and feedback channel.

In order to achieve this objective, the four experiments were carried out to investigate the effectiveness of the algorithms for four different simulation scenarios.

5.3 Performance Comparison between Adaptive and Non-Adaptive Error Control Algorithms

Here we discuss the results obtained from simulations that have been carried out. As mentioned previously, all tests were divided into two parts; one part with data transmission source type and another part with speech transmission source type. Our main focus is to investigate the performance of the three error control algorithms into four different network scenario.

For four types of test below, we examined these sources of transmissions at two different mobile speeds. The pedestrian and car speeds were representative of the mobile speed for channel transition. The values for pedestrian speed and car speed for the channel transition are 0.0000167 km/h and 0.0000530 km/h respectively. The values for pedestrian speed were choose to represent people who are moving in their office, while the car speed rate to represent people who are moving by car. These figures are also based on the works of Lettieri (1997). The same value is preserved in our experiment in order to validate our model against theirs. The rate of data transmission and speech transmission source types has been discussed in Chapter 4 (Section 4.2.3)

There are two call arrival patterns tested for both traffic source types, which are referred as normal call arrival pattern and fluctuating call arrival pattern. In case of normal call pattern, the interval between call arriving at sender seeking for admission from source, is an independently distributed exponential variable with mean 0.1 second. However, in case of the fluctuating call pattern, the call arrival rate is switched every 100 seconds on average, between 10 sources per second with probability 2/3 and 0.01 sources per second with probability 1/3. The value of rate switching interval is an independent identically distributed random variable with mean 100 second.

For each test, we discuss the number of packets lost and simulation time for all of the three algorithms. We compare the best performance out of these three algorithms and determine which approach is suitable for a given network scenario. In addition, the number of packets generated at the source entity depends on the source generator and call arrival pattern.

5.3.1 Experiment A: Data Transmission at Pedestrian Speed

The first set of simulation results is shown in Table 5-1(a) and Table 5-1(b). Table 5-1(a) shows the performance of the algorithms under *data transmission* sources and *pedestrian mobile speed* wireless network scenario, with normal call arrival pattern. This is followed by Table 5-1(b) which shows the performance of the algorithms under the same data source type and mobile speed as previous, but with fluctuating call arrival pattern.

The algorithm number for the error control algorithm tested is indicated in the first column of the tables. The second and third column of both tables indicates the number of packet dropped by simulation and number of packet safely arrives at the receiver entity. The percentage of packet errors is shown in the fourth column. Values in the fifth column indicate the simulation time. Finally, the number of message passing among the entities is shown in last column of both tables.

Algo.	Number of packet dropped	Number of packet receipt	Percentage of packet loss (%)	Simulation time (sec)	Number of messages
1	7135	29009	19.74	207.81	29588546
2	7183	28852	19.93	211.67	29544504
3	7086	29148	19.55	222.78	29694628

Table 5-1(a): Simulation result with normal call pattern (data, ped)

Algo.	Number of packet dropped	Number of packet receipt	Percentage of packet loss (%)	Simulation time (sec)	Number of messages
1	3193	7703	29.30	109.75	12537805
2	6039	18373	24.73	180.84	24727368
3	2876	9297	23.62	120.37	12108585

Table 5-1(b): Simulation result with fluctuating call pattern (data, ped)

From the Table 5-1(a), we can see that both non-adaptive algorithm (Algorithm 1 and Algorithm 2) is outperformed by adaptive algorithm (Algorithm 3), which adaptively tune their error control scheme, as the current bit error rate becomes higher due to channel degradation. In case of adaptive algorithm is used, the percentage of packet loss can be reduced by its decision to used FEC code when worst channel BER arises. In other hand, if better channel was estimated from previous windows transmission, then the adaptive algorithm will adapt its decision to use the SACK scheme. For example, the simulation result in Table 5-1(a) shows that the value of packet error percentage for Algorithm 3 is 19.55% while 19.74% and 19.93% for Algorithm 1 and 2 respectively.

According to the Figure 5-1(b) above, we can see that adaptive algorithm perform well and also can predict either the channel is in good condition or not. Algorithm 3 is outperform the others algorithm in term of packet loss percentage. This is because this algorithm will tune to FEC when higher channel bit error rate appear. However the simulation time needs to run this algorithm is more time consuming compare to Algorithm 1 as shown in both of the above table.

Conclusion

From the results discussed above, we see that the best performance in term of packet throughput is clearly shown by adaptive algorithm (Algorithm 3). While, the number of packet loss is relatively highest when we use Algorithm 1 since it's implement SACK scheme alone in the case of channel quality is in a bad condition.

5.3.2 Experiment B: Data Transmission at Car Speed

In Experiment A, the simulation parameters resulted in a network scenario with low mobile speed (at pedestrian speed). In this experiment, the algorithms' responses and sensitivity to higher mobile speed such as car speed is investigated. Data transmission sources were still used here.

Algo.	Number of packet dropped	Number of packet receipt	Percentage of packet loss (%)	Simulation time (sec)	Number of messages
1	7383	28530	20.55	210.65	29752928
2	7197	28938	19.91	254.52	29544512
3	7186	28915	19.90	268.15	29913645

Table 5-2(a): Simulation result with normal call pattern (data, car)

Algo.	Number of packet dropped	Number of packet receipt	Percentage of packet loss (%)	Simulation time (sec)	Number of messages
1	3458	8932	27.90	159.99	13607260
2	5388	15103	26.29	169.40	22820195
3	2864	9068	24.00	177.90	11976654

Table 5-2(b): Simulation result with fluctuating call pattern (data, car)

In this experiment, the higher mobility speed rate was expected to be influence the performance of the packet error percentage. With reference to the Table 5-2(a), for all algorithms, we see that the result is quite similar to Experiment A (pedestrian speed) but it has a little bit increase in packet loss due to the faster mobility speed appeared. As stated earlier in literature, the SACK scheme performs best at low bit error rate as no retransmission is necessary, while FEC scheme waste channel bandwidth, thus resulting in more late frames and higher overall packet loss rate. The algorithm 3 (adaptive) performance approaches that of SACK alone at low channel BER and is better than other schemes for high channel BER evaluated.

With reference to both Table 5-2(a) and Table 5-2(b), the simulation time needs to run the experiment for Algorithm 2 and 3 is higher compared to experiment for Algorithm 1 which is use SACK scheme. Generally, the behavior of each algorithm is not much different from previous cases where normal call arrival pattern is used.

Conclusion

The results above are expected and the discussion is similar to the previous section. The important thing here is that the percentage of packets dropped is higher compared to the testing done under pedestrian speed. Meanwhile, other parameters are not too much different compared with the previous test.

5.3.3 Experiment C: Speech Transmission at Pedestrian Speed

In the previous two simulation experiments, *data transmission* sources pattern have been used to generate network traffic from source. Peak rates of the data transmission sources are exponentially distributed with a mean of 32kbps. In this section, *speech transmission* sources are used to generate network traffic instead. These sources have a fixed peak rate of 32kbps. The main purpose here is to examine how the algorithms react to multimedia data type when compared to data transmission source in the previous two experiments. The others parameter is still remain as in previous experiment.

As in the previous section, two types of call arrival pattern have been used in the simulations. Table 5-3(a) shows the performance of the algorithms under *speech transmission* sources and *pedestrian speed* wireless network scenario, with a normal call arrival. This is followed by Table 5-3(b) which shows the performance of algorithms under the same parameter setting as Table 5-3(a), but with a fluctuating call arrival pattern. The algorithm number for the error control algorithm tested is indicated in the first column of the tables. The second and third column of both tables indicates the number of packet dropped by simulation and number of packet receipt at the receiver. The percentage of packet loss is shown in the fourth column. Values in the fifth column indicate the simulation time. Finally, the number of message passing among the entities is shown in last column of both tables.

Algo.	Number of packet dropped	Number of packet receipt	Percentage of packet loss (%)	Simulation time (sec)	Number of messages
1	9355	26682	25.95	270.76	29568261
2	9330	26806	25.81	276.80	29574512
3	9217	26961	25.47	283.69	29870859

Table 5-3(a): Simulation result with normal call pattern (speech, ped)

Algo.	Number of packet dropped	Number of packet receipt	Percentage of packet loss (%)	Simulation time (sec)	Number of messages
1	6253	12767	32.87	194.00	21115784
2	3845	8471	31.21	192.17	13391969
3	6282	13995	30.98	200.84	21384305

Table 5-3(b): Simulation result with fluctuating call pattern (speech, ped)

From the Figure 5-3(a) above, we see that both non-adaptive algorithms outperformed by adaptive algorithm in term of the packet throughput. The percentage of packet loss for this kind of transmission is increased compared to the data transmission because the speech transmission has a delay constraint that will influence the number of packet can reached at destination.

However, by using the adaptive algorithm the percentage of packet loss can be reduced compared to both non-adaptive algorithms. This is shown by the simulation result in both Table 5-3(a) and Table 5-3(b).

Conclusion

From the results above, we see that the best performance is again achieved by adaptive error control algorithm. The algorithm performs well in term of packet dropped and BER. When using Algorithm 3, the amount of packet lost is the lowest compared to the other non-adaptive algorithms. The value of packet lost percentage is low as 25.47% when tested under an adaptive error control algorithm (Algorithm 3). Meanwhile, the amount of packet lost is relatively higher when we use Algorithm 1. It's show that the channel quality is not in good condition for this experiment.

5.3.4 Experiment D: Speech Transmission at Car Speed

In this case, the higher mobility speed is assumed in order to investigate whether it will give any impact on the throughput performance. Logically, the more mobility speed is increased, and then more probability of transmission error would occur. This definitely will give advantage to adaptive algorithm because it can predict what the future channel BER will appear and decide to change their error control scheme in order to reduce the number of packet loss.

In Experiment C, the simulation parameters use a network scenario with low mobile speed (at pedestrian speed). However, in this section, the algorithms' responses and sensitivity to higher mobile speed such as car speed is investigated. Speech data transmission sources were used to represent multimedia data.

Algo.	Number of packet dropped	Number of packet receipt	Percentage of packet loss (%)	Simulation time (sec)	Number of messages
1	9716	26594	26.75	266.26	29440734
2	9384	26479	26.16	262.46	29710202
3	9193	26869	25.49	266.79	29929441

Table 5-4(a): Simulation result with normal call pattern (speech, car)

Algo.	Number of packet dropped	Number of packet receipt	Percentage of packet loss (%)	Simulation time (sec)	Number of messages
1	4820	8857	35.24	174.48	16906712
2	4005	8755	31.38	130.27	12872348
3	5567	12208	31.31	168.57	18494518

Table 5-4(b): Simulation result with fluctuating call pattern (speech, car)

With reference to the both Table 5-4(a) and Table 5-4(b), it's clearly show that Algorithm 3 (adaptive) outperform Algorithm 1 and Algorithm 2 in term of packet throughput. The percentage of packet error for Algorithm 3 is relatively low than both non-adaptive algorithms. For example, the simulation result in Table 5-4(a) shows that the value of packet error percentage for Algorithm 3 is 25.49% while 26.75% and 26.16% for Algorithm 1 and 2 respectively. The simulation time needs to run the experiments in Table 5-4(a) is slightly increased compared to experiment conducted in Table 5-4(b). This is because of in the first experiment; we need to wait for data to admit at 0.1 sec interval time instead of 0.01 sec interval time in the first experiment. The number of messages processed in Algorithm 3 also is higher than in Algorithm 1 and 2. In order to handle the tuning of error control scheme in adaptive algorithm, more messages passing occurred among the entities.

Conclusion

The results above are expected and the discussion is similar to the previous section. The important thing here is that the numbers of packets dropped in percentage is more higher compared to the testing done under pedestrian speed. Since, a speed rate for car is faster than pedestrian, the number of packet dropped is increased for the same speech transmission and algorithms used. As we can see from both tables in this experiment, we can expect that Algorithm 3 again will gain benefit from it's adaptation behavior if channel bit error rate is changed from good to bad and vice versa by implementing a suitable error control scheme.

5.4 Overall Conclusion

As mentioned earlier, we only summarize all the discussions at the end of the test because the results and discussion are similar for the tests under different source types. There are no graphs for this section due to the complexity of this simulation.

By using adaptive algorithm higher error rate percentage can be reduced. If there is low bit error rate resulted, then SACK is inserted to reduce power consumption cost. Otherwise, the FEC code will used to recover form higher channel error occurred. From all the above figures we can conclude that, the average performance of adaptive is much better than both non-adaptive for the case of higher channel bit error rate appeared. An adaptive algorithm will always react to change their error control scheme reflect to the changes of channel bit error rate after each data transmission done. In another word, it is capable of adapting its behavior over time and of improving its estimates.

In summary, all the algorithms have been tested in four simulation scenarios to evaluate their performance in different network environments. The simulation results have proved that adaptive tuning of error control scheme yields improvement in reducing the packet error percentage. We discuss the conclusion and some future enhancement ideas in the next chapter.