Chapter Seven

Conclusions and Future Work

7.1 Conclusions

In this thesis, an AQM method based on RED called Dynamic RED (REDD) has been proposed, that identifies congestion at router buffers at an early stage as in RED. The proposed REDD method like RED in using an average queue length (aql) as a congestion measure. REDD method aims to achieve performance measure results better than those of RED or at least similar to RED. Also, the REDD method aims to 1) Obtain more satisfactory performance measure results with reference to mean queue length (mql) and average queueing delay (D) than either RED or ARED method when congestion is exists, 2) lose fewer packets due to overflow router buffers than RED or ARED when congestion is given, and 3) the proposed REDD algorithm decrease the RED dependency on $\max threshold$ parameter as a constant instead it is used an adaptive $\max threshold$ parameter. The proposed REDD method probabilistically dropping the arriving packets when the aql is between $\min threshold$ and $\max threshold$ positions at its router buffer. Further, REDD uses an adaptive $\max threshold$ position aiming to keep the aql between the $\min threshold$ and the $\max threshold$ at a particular level (T_{aql}), which may lead to losing fewer packets. The REDD technique has been compared with the RED and ARED algorithms with regard to the performance measures (mql, throughput (T), D, overflow packet loss probability (P_L), packet dropping probability (D_p), the total of overflow loss and dropping probabilities for packets (P_{Loss})) to identify which method offers better performance measure results regarding the values of the packet arrival probability parameter.

The calculations of the performance measures of the RED, ARED and REDD methods were carried out based on data collected only after the simulation system had reached a steady state. Each compared AQM method was run ten times by changing the seed in order to provide a statistical analysis of every performance measure including confidence intervals. This indicates by how much the ten run results are dispersed from the mean. The (mql, T, D, P_L, D_p, P_{Loss}) represent the performance measure mean results for the ten runs.

This comparison mainly aims to determine which method offers better performance measure results with regard to the packet arrival probability. All the compared methods in this thesis are simulated using a Java environment on a computer machine with 1.66 Centrino and 1024 MB RAM.

The proposed REDD and both RED and ARED methods offer similar performance results (mql, T, D, P_L, D_p, P_{Loss}) when the packet arrival probability value is smaller than the packet departure probability value. On other hand, if the value of probability of packet arrival is larger than the packet departure probability value, the proposed REDD method slightly outperforms both the RED and ARED methods in terms of mql and D results. In addition, at these values for the packet arrival probability: 1) all compared AQM methods provide similar T and P_{Loss} results, 2) the REDD method slightly better than either RED or ARED method according to P_L results, and 3) the proposed REDD method drops a higher number of packets than RED and ARED since its router buffer overflows less than the RED and ARED methods.

Whatever the packet arrival probability value is, the statistical analysis measurement results (variation (σ^2), standard deviation (σ) and 95% confidence interval (CI), upper limit, lower limit) of the compared AQM methods are extremely small, and these results reflect a small dispersion for every performance measure ten run results from their mean. Therefore, the performance measure mean results can be considered accurate. When the probability of packet arrival was set to 0.48, this gave the largest fluctuation in the results. This was clearly due to statistical fluctuations in the input changing the system to drift between a congestion and non congestion situation.

Also in this thesis, new AQM methods based on integrating the RED and fuzzy logic have been developed in order to utilise them as congestion control polices in wired networks. These hybrid methods identify congestion and control it incipiently using the fuzzy inference process (FIP) rather than the traditional congestion metrics (aql, queue length (ql)). The developed methods aim to: 1) offer more satisfactory performance measure results than the RED algorithm regarding to mql and D when a high congestion is present, 2) lose fewer packets due to overflow than the RED algorithm when a high congestion exists, and 3) decrease the reliance upon the parameter settings, i.e.

$\min threshold$, $\max threshold$ for the RED algorithm. Therefore the proposed fuzzy logic algorithms do not depend on the $\min threshold$ and $\max threshold$ as congestion boundaries and instead they use a FIP.

The proposed AQM algorithms that use both RED and fuzzy logic have been called REDD1, REDD2 and REDD3.

The proposed fuzzy logic controller algorithms are compared with RED according to the $(mql, T, D, P_L, D_p, P_{Loss})$ to identify which algorithm can offer more enhanced performance measure results. The performance measure results have been accomplished by setting the packet arrival probability parameter to different values. So a decision about which algorithm gives better performance measure results is issued only based on the parameter values for the packet arrival probability. Furthermore, the performance measures of the RED and proposed REDD1, REDD2 and REDD3 algorithms were calculated after the system reached a steady state.

The RED and the proposed REDD1, REDD2 and REDD3 algorithms were run ten times through changing the seed with aim to generate a statistical analysis for each performance measure results as before.

This comparison mainly aims to obtain which method offers better performance measure results with regard to the packet arrival probability parameter.

The proposed REDD2 and REDD3 algorithms give similar performance measure results $(mql, T, D, P_L, D_p, P_{Loss})$ whether the packet arrival probability value was high or not. The proposed algorithms and RED provide similar mql and D results when the packet arrival probability value was set to either 0.18 or 0.33. However, when the packet arrival probability becomes very close to the packet departure probability value, i.e. 0.48, all compared algorithms excluding the proposed REDD1 algorithm give similar mql and D results and these results slightly outperform their corresponding results in the REDD1 algorithms. In case of the packet arrival probability value was set to 0.63. The RED, REDD2 and REDD3 algorithms continue providing better results with reference to mql and D than the REDD1 algorithm and the REDD2 and REDD3 algorithms became better than the RED algorithm in terms of these results. The proposed REDD1, REDD2 and REDD3 algorithms outperform the RED algorithm according to the results of mql and D when a high congestion is exists, i.e. the packet arrival probability value = 0.78 or 0.93. Moreover, at these values for the packet arrival probability, the proposed REDD2 and REDD3 algorithms outperform the proposed REDD1 algorithm with respect to mql and D results. The RED and proposed REDD1, REDD2 and REDD3 algorithms obtain similar T and P_{Loss} results whether the packet arrival probability is high or not.

The RED and the developed REDD1, REDD2 and REDD3 algorithms lose and drop similar amounts of packets at their router buffers when the packet arrival probability value was set to either a smaller value than the packet departure probability value or 0.63, where the packet loses is due to router buffer overflows. When the packet arrival probability value was given to either 0.78 or 0.93, the proposed REDD1, REDD2 and REDD3 algorithms lose fewer packets than the RED algorithm and the REDD1 outperforms both REDD2 and REDD3 with reference to P_L. The RED algorithm drops less number of packets than the proposed REDD1, REDD2 and REDD3 algorithms when the packet arrival probability is equal to 0.78 or 0.93. Also, at these values for the packet arrival probability, the REDD1 algorithm drops larger numbers of packets than either REDD2 or REDD3.

The statistical analysis results (σ^2, σ, 95% CI, upper limit, lower limit) of the RED, REDD1, REDD2 and REDD3 algorithms for every performance measure were extremely small. Consequently, the ten run results for every performance measure are not widely dispersed from their mean, and this gives confidence of a more accurate mean. In addition, the worst statistical analysis results of the compared algorithms were obtained when the packet arrival probability value was equal to 0.48. This was for the same reasons as previously discussed.

Also in this thesis, some discrete-time queue analytical models based on different AQM methods (DRED, BLUE, RED and GRED) within wired networks were proposed. These proposals are made in order to exploit them as congestion control methods in wired networks and also to achieve satisfactory performance measure results when a heavy congestion is present.

The developed analytical models and both the DRED and BLUE algorithms depend on the instantaneous ql as a congestion metric, whereas the RED and GRED algorithms rely upon the aql as a congestion metric. The proposed analytical models have been called: DRED-Alpha, DRED-Linear, BLUE-Alpha, BLUE-Linear, RED-Alpha, RED-Linear, GRED-Alpha and GRED-Linear.

Every proposed analytical model has been compared with the corresponding original AQM method with reference to the performance measures mentioned above in order to identify which one gave more convinced performance measure results.

The comparison of the DRED analytical models and the original DRED algorithm revealed the following: When the packet arrival probability value is lower than the packet departure probability value, the DRED-Alpha and Linear models and DRED method give similar mql and D results. On the other hand, the proposed DRED-Alpha outperforms either DRED-Linear or DRED in terms of mql and D results when the packet arrival probability value is larger than the packet departure probability value. Also, at the same values of packet arrival probability, the mql and D results of DRED are better than the mql and D results of DRED-Linear. Whether the value of packet arrival probability is high or not, the DRED and analytical models produce similar T and D_p results. The DRED-Alpha and Linear models give similar P_{Loss} results whatever the packet arrival probability value is. The DRED method achieves similar P_{Loss} and P_L results as the DRED models provided the packet arrival probability value was equal to 0.18 or 0.33. If the value of packet arrival probability increased to 0.48 or larger than the packet departure probability value, then the DRED-Alpha and Linear marginally outperform the DRED method with reference to P_{Loss} and P_L due to the models losing fewer packets than the DRED method.

From the comparison of the BLUE analytical models and the original BLUE algorithm, the following were concluded: The BLUE-Alpha and BLUE provide similar mql and D results whatever the packet arrival probability value is. In addition, the BLUE-Linear gives mql and D results similar to those of BLUE and BLUE-Alpha only when the value of packet arrival probability is smaller than the value of packet departure probability, whereas if the packet arrival probability value is larger than the packet departure probability value, then both BLUE and BLUE-Alpha model outperform BLUE-Linear model with reference to mql and D results. The BLUE and BLUE models offer similar T and D_p results whether the packet arrival probability value is high or not. The BLUE and BLUE models produce similar P_{Loss} and P_L results when either the packet arrival probability value is larger than the packet departure probability value or equal to 0.63. On the other hand, if the packet arrival probability value was set to either 0.78 or 0.93, the BLUE outperforms BLUE-Linear according to P_{Loss} since the BLUE loses marginally fewer packets due to overflow than the BLUE-Linear. Also, at these values of packet arrival probability, the P_{Loss} results of BLUE-Alpha are better than the P_{Loss} results for BLUE because the BLUE loses more packets than the BLUE-Alpha. The optimal th position at routers buffers of BLUE and BLUE models was set to a value as small as possible in order to obtain the most satisfactory results for mql and D. When the th was set to any tested value [4-14], the most satisfactory T, P_{Loss}, P_L and D_p results were accomplished.

From the comparison of the RED analytical models and the RED method, the conclusions are given as: Similar mql and D results for the RED and the two RED models (Alpha, Linear) were obtained when the packet arrival probability value was set to 0.18 or 0.33. However, if the packet arrival probability was given to a value very close to the packet departure probability value, i.e. 0.48 or a value larger than the packet departure probability value, then the RED models outperform the RED method according to the mql and D results. The RED and RED models were offered similar T results when the packet arrival probability value was set to 0.18, 0.33, 0.78 or 0.93. On the other hand, if the packet arrival probability value was tuned to a value near the packet departure probability such as 0.48 or 0.63, the T results of RED are marginally better than the T results for the RED models. Additionally, the RED-Linear is outperformed the RED-Alpha with respect to T results.

Both RED and the two models based on RED give similar P_{Loss} results when the packet arrival probability was equal to 0.18. Moreover, slightly better P_{Loss} results for RED than the RED models were produced when the packet arrival probability value was set to 0.33, 0.48 or 0.63. Also, at these values, the RED-Linear outperforms the RED-Alpha according to P_{Loss} results. The RED models and RED provide similar P_{Loss} results when the packet arrival probability value was configured to 0.78 or 0.93.

The RED and RED-Alpha and Linear models lose similar numbers of packets when the packet arrival probability value is lower than the packet departure probability value. On the other hand, if packet arrival probability value > packet departure probability value, the RED models lose fewer packets than the RED method. The two models based on RED lose similar amounts of packets whether the value of packet arrival probability value is high or not. When a packet arrival probability value was set to 0.18, then the RED and its models drop similar number of packets at their routers buffers. However, the RED outperforms the models according to D_p results when the packet arrival probability value is larger than 0.18. Furthermore, the RED-Linear drops fewer packets than the RED-Alpha when the packet arrival probability value = 0.33, 0.48 or 0.63. Finally, both RED models were gave similar D_p results when the packet arrival probability value is high, such as 0.78 or 0.93.

The most satisfactory mql, T, D and P_{Loss} results of RED were accomplished when the $\min threshold$ was configured to any tested value [4-9]. Moreover, in RED, by setting the $\min threshold$ to a value furtherest from the $\max threshold$, the fewer packets were lost due to overflow. On the contrary, if the $\min threshold$ was set to a value as near as possible to the $\max threshold$, then the fewest number of dropped packets at RED router buffer were obtained.

Comparing the proposed GRED analytical models and the original GRED method, the following was observed: The GRED and the analytical models similar mql, T and D results once the packet arrival probability value was set to 0.18 or 0.33. Whereas, if the packet arrival probability value was given to a value equal to or larger than 0.48, the GRED-Alpha and Linear outperform the GRED concerning to mql and D results. Moreover, at these values for the packet arrival probability, the GRED-Alpha provides better mql and D results than the GRED-Linear. The GRED somewhat is better than the GRED analytical models in terms of T result when the packet arrival probability was tuned to 0.48, and also the T result of GRED-Linear is slightly better than the T result for the GRED-Alpha. The GRED and GRED-Linear give similar T results when the packet arrival probability value was set to 0.63, and these results were better than the T result of the GRED-Alpha. In accordance with a high congestion situation packet arrival probability values given by 0.78 or 0.93, the GRED and the models achieve similar T results.

The results of P_{Loss} for the GRED and GRED-Linear are better than the P_{loss} results of the GRED-Alpha when the packet arrival probability was set to one of the following values: 0.33, 0.48 or 0.63. On the other hand, if the packet arrival probability value was set to 0.18, 0.78 or 0.93, the GRED-Alpha and both GRED and GRED-Linear provide similar P_{Loss} results. The GRED-Alpha and Linear lose similar numbers of packets at their routers buffers due to overflow whether the packet arrival probability value is large or not. Also, the GRED was lost similar number of packets when packet arrival probability value < packet departure probability. While, if the packet arrival probability value is larger than the packet departure probability value, the GRED loses more packets due to overflow than the GRED models.

The GRED and the two models drop similar amounts of packets when the packet arrival probability value = 0.18. When the packet arrival probability value is equal to 0.33, the GRED and GRED-Linear drop similar numbers of packets and their dropped packets are fewer than dropped packets by the GRED-Alpha. GRED outperforms the GRED models by dropping fewer packets when the packet arrival probability value ≥ 0.48. GRED-Linear drops fewer packets than the GRED-Alpha when the packet arrival probability value was set to 0.48 or 0.63. Finally, if a high congestion is set up with the packet arrival probability value to either 0.78 or 0.93, the GRED models drop similar amounts of packets.

The proposed AQM algorithms (REDD, REDD1, REDD2, REDD3) and the developed analytical models based on the AQM methods could be applied as congestion control approaches in the internet. The transport protocol could be either TCP or User Datagram Protocol (UDP), since the UDP is a suitable for delivery of real time multimedia traffic such as voice and video due to the UDP being sensitive for a delay and has a low sensitivity for packet loss. The internet traffic relies upon requirements of QoS.

Parameter settings of the proposed algorithms and models could be tailored to which type of service is appropriate. For instance, in the proposed algorithms and models, if a low delay, i.e. average queueing delay (D), is necessary with using a high packet arrival probability value, i.e. 0.78. These requirements are requested by some real time applications, such as telephony, Voice over IP (VoIP), conferencing, video conferencing, voice and live video. In addition, the parameters of th and $\min threshold$ in the BLUE and RED models, respectively can be adjusted to a real time traffic service to be used in the BLUE and RED models, therefore real time applications such above can apply in the proposed algorithms and models. For example, if a low D is required and the probability of packet arrival is high, a real time traffic service can be utilised in the proposed BLUE and RED models by setting the th parameter to a value as small as possible and the parameter of $\min threshold$ to a value as far as possible from the $\max threshold$. Moreover, if the following is requested: a low packet loss probability (P_L), a low packet dropping probability (D_p) and a low total of lost and dropped packets (P_{Loss}), this might be characteristic of such applications as file transfer or electronic mail, since these applications concern in the status of packets and do not rely on delay. In the proposed BLUE and RED models, the most satisfactory P_L, D_p and P_{Loss} were obtained by setting th and $\min threshold$ to any considered value, i.e. for th and $\min threshold$ the considered values are in [4-14] and [4-9], respectively since these values were not affected by the positions of th and $\min threshold$.

7.2 Future Work

In terms of the future work, the following are suggested:

1. Apply the AQM methods examined in Chapter 3 to cellular wireless networks to control the congestion at the base station queues.
2. Examine the performance of the proposed REDD algorithm that was given in Chapter 4 within the environment of wireless networks.
3. Apply the proposed REDD algorithm using TCP as the transport protocol to evaluate the performance of the algorithm with TCP connections.
4. Extend the REDD algorithm to make it depends on updating both $\max threshold$ and $D_{\max}$ parameters to see how well it can stabilise aql between $\min threshold$ and $\max threshold$ positions.
5. Use the proposed REDD1, REDD2 and REDD3 approaches in TCP networks.
6. Apply the REDD1, REDD2 and REDD3 approaches as congestion control methods in wireless networks.
7. Develop new fuzzy logic control methods that rely on average queueing delay as an input linguistic variable to control the congestion in either wired or wireless networks.
8. Develop other fuzzy logic algorithms based on RED, GRED, DRED and BLUE algorithms that depend on the traffic load level as an input linguistic variable.
9. Employ the proposed discrete-time queue analytical models within networks of queues that consists of N queue nodes.

10. Use batch arrivals and/or batch departures rather than single arrivals and/or single departures in each slot in the proposed discrete-time queue analytical models.
11. Exploit other arrival processes, for instance Markov-modulated Bernoulli process (MMBP) and Pareto in the developed analytical models in order to determine performance with bursty traffic and long range dependence.
12. Revise the proposed analytical models so they depend on the average queue length rather than the instantaneous queue length, perhaps using equilibrium point analysis (EPA) [112].
13. Applying different classes of services, i.e. expedited, assured and best effort in the proposed analytical models by giving different threshold positions for each class of service.
14. Investigate the use of a hybrid model that switches from using aql to instantaneous ql depending on specific criteria.