Fuzzy Logic based ATM Policing

Kwan F. Cheung, Danny H.K. Tsang, Chak C. Cheng and Chiu W. Liu

*Dept Electrical and Electronic Engineering
Hong Kong U of Science and Technology, Clear Water Bay, Kowloon, Hong Kong.*

Abstract: In this paper, we will compare the performance of a fuzzy logic (FL) based policer with leaky bucket (LB) based policer. The FL based policer continuously computes the instantaneous drop-rate based on a collective evaluation of the compliance/violation level of the two parameters: (a) the ratio of the up-to-date mean burst length to the negotiated burst length, and (b) the ratio of the up-to-date mean bit rate to the negotiated mean bit rate. Compliance/violation of either of the two parameters results in an increase/decrease of the strength of the rules with consequence Pass/Drop. The throughput performance on policing mean bit rate and mean burst length of the FL based policers is compared with that based on LB based policers of different token pool sizes. Throughput performance curve of the FL based policer on policing mean bit rate is very close to the optimal performance curve of LB based policers, but significantly outperform LB based policers on policing mean burst length.

Leaky Bucket Policer

Leaky bucket policing scheme is a simple and effective policing scheme to ensure that sources do not exceed the negotiated mean bit rate. A block diagram leaky bucket policers is in Figure 1.

By using fluid flow approach, the throughput of a leaky bucket policer is found to be [1]:
where

$$P = \frac{(B_p - B_m)}{B_p} \cdot \frac{\lambda_1 - \lambda_2}{\lambda_1 e^{-(\lambda_1 - \lambda_2)M} - \lambda_2} \quad (1)$$

P = cell loss probability

$\lambda_1 = 1/h \cdot (B_p - B_m)$

$\lambda_2 = 1/(k \cdot B_m)$

M = packet size in bit

B_m = negotiated mean bit rate

B_p = peak bit rate

h = mean burst duration

k = silent burst duration

Two parameters govern the behaviour of leaky bucket policers are: B_m , the depletion rate or the leak rate, and S_m , the token pool size. The throughput performance criteria of a policer is determined by the policer's ability to bound the throughput rate at B_m . In LB scheme, ideal throughput performance is obtained by setting B_p equal to B_m . In this mode, S_m value is required to be set unbound to accommodate short bursts [1]. In this mode, LB policer is very sensitive to violation of mean rate. Large S_m , however, is not desirable due to the long response time or slow reaction to long bursts. Policing of mean rate can be

maintained with larger B_p 's and smaller S_m 's. With these settings, long bursts are contained except that the leak rate is also larger than the negotiated mean rate and thus can also degrade the policer's performance on policing mean rate as well as short bursts. LB based policers have long been recognized to be ineffective in policing bursty traffics. A number of LB based policing schemes are proposed to improve the policers' performance on reaction to bursts. One effort is to modify non-ideal LB based policers ($B_p > B_m$) by cascade a spacer after the policer to smooth traffic burst. The spacer is activated by appending space to inter-arrival time between subsequent packets only after the policer detects burst traffic. Another effort is to cascade a few LB based policers, each is capable to police different burst lengths. Please refer to [2,3] for an overview of leaky bucket variants.

Fuzzy Logic Policer

In this paper, we will describe the formulation of a fuzzy logic (FL) based policers. The formulation is of the same goal: to design a policer that simultaneously polices mean rate and reject bursts. Policing of mean rate is a straightforward decision. Cells are discarded when the mean rate is above the negotiated rate. Policing of bursts, however, is not as mechanical. This requires a collective evaluation of the mean rate as well as the burst type before decisions of discarding cells are made. Consider the scenario that the mean rate is below the negotiated rate while the burst rate is a bit above the negotiated rate. Discarding cells can slightly improve the burstiness situation of the bit stream but also worsen the throughput capacities of the mean rate. Violation of the negotiated burst rate under such scenario may need to be pardon. However, given the same scenario but the burst rate is much higher than the negotiated rate, some penalty should be imposed by discarding some fraction of incoming cells. If both the mean rate and burst rate are violated, a stronger penalty should be imposed. Using this type of reasoning, one can state the desired action of the policer at each combination of compliance/violation levels of both the mean rate and the burst rate. The overall control action of the policer is an aggregation or interpolation of the desired policing action of all the combination.

In a FL controller, the control action is based on the overall evaluation of the truthness of all the embedded control rules [4-9]. Each of the rules is stated in an propositional logic statement of "if A then B " where A is the antecedent of the statement stating the combination of conditions and B is the corresponding consequent. In this case, the antecedent in every rule is one of the combinations of the compliance/violation levels of mean rate and burst rate, and the consequent is the desired policing action. Given

some input to the controller, the truthness of the antecedent of each rule will be evaluated. The truthness of the consequent is equal to the truthness of the antecedent of the rule. The final output of the FL controller is based on an aggregation of the all the consequents weighted by the corresponding truthness values. The block diagram of a fuzzy logic controller is illustrated in Figure 2. Every fuzzy logic controller consists of four blocks. The Fuzzification block is to map each of the input parameters, which are crisp values, into a set of membership value characterizing a collection of fuzzy sets. The Rule-Base block stores the control rules stating the relationship between input and output. The Rule Execution Logic block is responsible for evaluating the truth value of the antecedent and infer the membership function of the consequent for every rule. The Defuzzification block is to aggregate the weighted consequents and convert them into a crisp output value. In effect, fuzzy logic control is that every rule is a sample point in decision space. A fuzzy logic controller interpolates the sample points into a smooth control surface [7]. Readers can refer to [8,9] for a more detail description of various form of fuzzy logic control.

The FL based policer presented in this paper is a two-input, nine-rule, one-output system. The two-input values to the policer are: (a) the ratio of the up-to-date mean burst length to the negotiated mean burst length, and (b) the ratio of the up-to-date mean bit rate to the negotiated mean bit rate. The output is a value in [0,1] represents the instantaneous drop rate. The value 0 implies 0total DROP, and 1 implies total PASS. The policer computes the instantaneous drop-rate based on a collective evaluation of the compliance/violation levels of the two input values. The output value is fed to a pass/drop switch. The switch may be a memoryless switch. The switch will switch to DROP position when the computed drop rate is above some threshold value. The policer introduced in this paper has a memory size of ten cells. If the switch has memory, discarding of cells can be flexible. Given a computed drop value, the switch will discard a portion of cells held in memory in accordance with some discarding policy.

We used the up-to-date mean, $\bar{x}(\tau)$, as the mean value of a random process $x(t)$:

$$\bar{x}(\tau) = \frac{1}{\tau} \int_0^{\tau} x(t) dt$$

Windowed mean can also be used. Three fuzzy sets: COMPLY, SORT OF COMPLY, AND VIOLATE, are defined on the two input parameters. And Three fuzzy sets: PASS, BETWEEN PASS AND DROP, AND DROP, are defined for the output space of [0,1]. The membership functions of all the nine fuzzy sets are manually tuned to optimized the throughput performance.

The control rules are stated in implicative logic form :

IF the ratio of the up-to-date mean bit rate to the negotiated mean bit rate, **AND** the ratio of the up-to-date mean burst length to the negotiated mean burst length, **THEN** Consequent.

Let A_1 denote the ratio of up-to-date-mean bit rate to the negotiated mean bit rate, and A_2 denote the ratio of up-to-date mean burst length to the negotiated mean burst length, and C denote the consequent. The nine rules listed below are obtained from policing experts and are used in our FL based policer:

Rules	A_1	A_2	C
R_1	COMPLY	COMPLY	PASS
R_2	COMPLY	SORT OF COMPLY	BETWEEN PASS AND DROP
R_3	COMPLY	VIOLATE	DROP
R_4	SORT OF COMPLY	COMPLY	BETWEEN PASS AND DROP
R_5	SORT OF COMPLY	SORT OF COMPLY	BETWEEN PASS AND DROP
R_6	SORT OF COMPLY	VIOLATE	DROP
R_7	VIOLATE	COMPLY	DROP
R_8	VIOLATE	SORT OF COMPLY	DROP
R_9	VIOLATE	VIOLATE	DROP

Violation (Compliance) of either of the two parameters will increase (decrease) the strength of the rules whose consequence are DROP (PASS) respectively. The aggregation operation used is the center-of-gravity (COG) method commonly adopted in fuzzy logic control systems. In particular, the output from the fuzzy logic controller is a crisp number in [0,1] corresponding to the drop rate. The output value 0 implies total DROP, and 1 implies total PASS. The pass/drop switch is a controller to execute the drop schedule in accordance with the drop rate calculated by the fuzzy logic controller.

Performance of FL Based Policer

The FL based policer is tested on the policing of packet voice and still images. Both traffic pattern were generated by poisson random number generators. For the packet voice generation, we set $B_m = 26$, $B_p = 75$, $L_m = 26$ (L_m is the mean burst length). For the still picture generation, we set $B_m = 205$, $B_p = 4717$, $L_m = 2358$. The FL based policer is tested in parallel with leaky bucket policers of different combinations of B_p and S_m . Five LB based policers were used for comparisons:

Leaky Bucket	Depletion Rate (B_s)	Threshold Value (S_m)
B1	67.5	69
B2	52.5	237
B3	37.5	649
B4	30	1,829
B5	27	9,004

In the policing of still pictures, six LB based controllers were used for comparisons:

Leaky Bucket	Depletion Rate (B_s)	Threshold Value (S_m)
B1	4245	4,830
B2	3302	10,074
B3	2359	27,271
B4	1698	36,625
B5	472	79,715
B6	212	339,016

Testing is performed on both violation of mean bit rate and violation of mean burst length. The statistics obtained from every test is accumulated from a population of 10^9 packets. The throughput performance of the FL based policer to LB based policers are shown in Figures 4 to 7. The dash line is the ideal curve computed by (1).

In policing of mean rate, LB based policers perform very well in the policing of mean rate as B_s approaches B_m and with large buffer sizes. The performance deteriorates when B_s deviates away B_m . In a peer-to-peer comparison, the FL based policer performs extremely well, only slightly inferior to the leaky bucket policer B5, which has the highest rate and largest token pool size. The FL based policers excel LB policers on policing bursty traffics. FL based policer outperforms all LB based policers substantially in the policing of the traffic sources. The performance curve is very close to the ideal performance curve.

Conclusion

In this paper, a method of utilizing fuzzy logic control methodology on policing is presented. Simulation results demonstrate that, in terms of throughput performance, FL based policer can outperform leaky bucket based controller by substantial margin, in particular in the policing of burst traffic.

References

- [1] M. Butto, E. Cavallero and A. Tonietti, "Effectiveness of the "Leaky Bucket" Policing Mechanism in ATM Networks", *IEEE J. Selected Areas in Comm*, vol. 9, No. 3, April 1991, pp.335-341.
- [2] D. Hoong and T. Suda, "Congestion Control and Prevention in ATM Networks," *IEEE Network Magazine*, July 1991, pp.10-16.
- [3] L. Fratta, L. Musumeci, G. Gallassi, L. Verri, "Congestion Control Strategies in ATM Networks," *European Transaction on Telecommunications and Related Technologies*, vol. 3, no. 2, March-April 1992, pp.183-193.
- [4] L. Zadeh, "Fuzzy Logic", *IEEE Computer Magazine*, 1988.
- [5] L.A. Zadeh, "The concept of a linguistic variables and its application to approximate reasoning: Part 1, 2, and 3," *Information Science*, vol.8 pp. 199-249, pp. 301-357, and vol. 9, pp. 43-80, 1975.
- [6] Yamazaki and M. Sugeno, "Fuzzy Control," *Systems and Controls*, vol. 28, pp.442-446, 1984. (in Japanese).
- [7] P. Wang, H. Lui and T. Goh, "Fuzzy Controllers Make Interpolation Using Fuzzy Samples," Collection of Papers, *Workshop on the Future Directions of Fuzzy Theory and Systems*, The Chinese University of Hong Kong, Oct, 1993.
- [8] T. Terano, K. Asai and M. Sugeno, *Fuzzy Systems Theory and Its Applications*, Academic Press, 1992, Chapter 10.
- [9] M. Sugeno and T. Yasukawa, "A Fuzzy-Logic-Based Approach to Qualitative Modelling," *IEEE Trans. Fuzzy Systems*, vol. 1, No. 1, Feb 1993, pp.7-31.

Figures

Figure 1. The block diagram of a leaky bucket policer.

Figure 2. The block diagram of a fuzzy logic control system.

Figure 3 The block diagram of the fuzzy logic based policer

Figure 4. Packet Voice with $L_m = 26$.

Figure 5. Packet Voice with $B_m = 26$.

Figure 6. Still Picture with $L_m = 2358$.

Figure 7. Still Picture with $B_m = 205$.