

Adaptive Deterministic Packet Marking

Lachlan L. H. Andrew, Stephen V. Hanly, Sammy Chan, and Tony Cui

Abstract—An efficient method is presented for signaling link price information using single-bit marks. It exploits side information in the `IPid` field of the IP header to allow the maximum price on a flow's path to be estimated. It automatically adapts the resolution with which the price is quantized, depending on how quickly the price changes. The algorithm does not depend on the number of hops in a link. A marking scheme with improved compatibility with current ECN (RFC 3168) is also proposed.

Index Terms—Congestion control, ECN, packet marking.

I. INTRODUCTION

MANY congestion control algorithms have been proposed which require explicit feedback of congestion (“price”) information from routers [1], [2], [3], [4]. The IP header contains two Explicit Congestion Notification (“ECN”) bits for this purpose, defined in RFC 3168. Pricing information can be transmitted by randomly marking packets with these bits [2], [5]. It has recently been proposed [6] that the process of setting these bits take into account “side information” contained in the IP header. This idea was extended by Thommes and Coates [7] to provide an efficient, deterministic marking algorithm, using the IP packet’s identifier, the 16 bit `IPid` field in the header, to assist in conveying the binary representation of the price. This 16-bit field is already in the IP header for fragment reassembly; that use is unaffected, as the algorithm [7] reads the value but does not alter it.

The present paper proposes adaptive deterministic packet marking (ADPM)¹. ADPM applies the idea of using `IPid` to the task of transmitting the unary representation of the price. This approach has many benefits, such as automatically adapting the quantisation resolution of the price to the rate at which the price changes, so that static values can be estimated precisely, while rapidly changing values can be tracked quickly. Like [8], it conveys the *maximum* link price, as used in [4], rather than the sum of the prices, as used in [1], [2], [3]. This yields maxmin fairness [4], [8] rather than maximum utility [1], [2], [3]. Both approaches are capable of providing high utilisation of a network, with minimal queuing and without fluctuating rates.

Manuscript received June 9, 2006. The associate editor coordinating the review of this letter and approving it for publication was Costas Georghiades. This work was supported by the Australian Research Council (DP0557611), the NSF under the FAST project, and the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU 110705).

L. Andrew is with the CS Dept. at Caltech, MC 256-80, 1200 E. California Blvd., Pasadena, CA 91125 (email: lachlan@caltech.edu).

S. Hanly and T. Cui are with the ARC Special Research Centre for Ultra-Broadband Information Networks (CUBIN), University of Melbourne.

S. Chan is with the Dept. of Electronic Engineering, City University of Hong Kong.

Digital Object Identifier 10.1109/LCOMM.2006.060898.

¹This is named after [7], [8], and is unrelated to the similarly named scheme [9], which uses the `IPid` field for IP traceback.

II. USING `IPid` FOR PACKET MARKING

Thommes and Coates [7] proposed a deterministic algorithm for communicating congestion prices, which uses side information in IP packets. The `IPid` field is set by the sender and used to identify IP fragments which belong to the same original IP packet; it will differ for all IP packets in close proximity. The key proposal of [7] was to use this value to specify how the ECN mark in a packet should be determined.

In the algorithm of [7], a router quantizes its link price to n levels, yielding a $\lceil \log_2 n \rceil = b$ -bit binary number. A hash function of the `IPid` field determines the *probe type* of a packet. When a packet of probe type i is transmitted, the router marks the packet if bit i of the quantized price is 1.

In order to communicate the sum of prices along a path of at most h hops, the algorithm introduces h probe types for each bit position. Denote the probe types by the pair (h_i, b_i) , where h_i is a hop number and b_i the number of a bit position. Each router guesses its hop number from the time to live (TTL) value [6]. For probes of type (h_i, b_i) only router h_i will mark the packet, if bit b_i in its price is set. From this, the receiver can determine the price of each hop on the path. It turns out that with a b -bit quantizer, and paths of at most h hops, then at least $2b\lceil h/6 \rceil$ packets corresponding to a given price must be received before the price can be estimated reliably. We will apply Thommes and Coates’ concept of probe types to a simpler form of marking, which does not limit the number of hops in the path.

III. SINGLE BIT MARKING FOR MAXIMUM PRICES

The original random marking schemes of [1], [2], [3], [5] essentially used unary encoding of signals; the value calculated from the number of bits received, requiring at least $n - 1$ packets to signal n different values. Adding prices was performed implicitly by the independent marking by the routers. Deterministic marking [7] allows more efficient binary encoding to be used, but this requires explicit adding of the link prices, and requires the number of bits to be fixed in advance.

In this paper, we attempt to encode the *maximum* price seen along the flow’s path, and this allows an *adaptive* quantization, as we will show. In what follows, it is convenient to assume that prices have been mapped to lie in the unit interval $[0, 1]$; from now on, we will use the term “price” to refer to the mapped value. Similarly, a mapping F is assumed, that maps `IPid` values to threshold values in $[0, 1]$. Following the terminology of [7], $i \equiv F(v)$ will be called the probe type of the packet. Implementation details behind the above assumptions are explained in Section V.

When a router with link price p forwards a packet of probe type i , it marks the packet if $p > i$, and leaves the mark

unchanged otherwise. At the receiver, the mark of a packet of probe type i will be set if any router on the path had a price exceeding i . Decoding is simple. The receiver maintains a current estimate of the price, \hat{p} . If it sees a marked packet of probe type i with $i > \hat{p}$ or an unmarked packet of probe type i with $i < \hat{p}$, then it sets \hat{p} to i .

This scheme is similar to the independently derived scheme of [8]. In that scheme, prices are quantized to $n \approx 100$ discrete values. Packets are only marked if the router's price exactly matches a price of the probe type, and an estimate is only available after approximately n packets have been received. Thus the primary difference between ADPM and the scheme of [8] is that packets are marked based on *inequality* with thresholds, rather than equality to quantised values. That makes the "effective resolution" of the quantizing process adaptive to the number of packets seen, as will be shown in the following section.

In this algorithm, the interpretation of each mark is independent of the values of other marks. In contrast, with binary signalling, a price change from 3 (011) to 4 (100) could yield any price estimate from 000 to 111, depending on the order in which bits are signalled.

IV. PERFORMANCE ANALYSIS

Consider the encoding of a fixed price, p . Each packet that arrives provides a bound on p ; packets of probe type i tell us whether or not $p \geq i$. After k packets have arrived, there is an interval in which p is known to lie, given by $[i^-, i^+)$, where i^- is the largest probe type, i , which has been seen such that $i \leq p$ and i^+ is the smallest probe type which has been seen such that $i > p$. The actual estimate, \hat{p} , is either i^- or i^+ , as determined by the initial value of the estimator. The error in the estimated price \hat{p} , can thus be bounded above by

$$|\hat{p} - p| < i^+ - i^-. \quad (1)$$

The statistics of the right hand side of (1) can be approximated by noticing that, for large k , the points i approximate a Poisson process of rate k on the interval $[0, 1]$. Let i^- and i^+ denote the endpoints of the inter-arrival interval of the Poisson process that contains p . Then \hat{p} is either i^- or i^+ , depending on the initial estimate. The well-known *waiting time paradox* for Poisson processes [10, Section 4.1] implies that the distance from a specific number, p , to the immediately preceding point of the Poisson process has mean $1/k$, and similarly, the distance from p to the next point of the Poisson process has mean $1/k$. Averaging over p , we obtain:

$$E[|\hat{p} - p|] \approx \frac{1}{k}, \quad (2)$$

although the error can vary significantly about this mean.

Note that (2) is within a factor of four of the mean absolute quantisation error of a k -level quantizer, $1/(4k)$. However, this resolution is adaptive to the number of samples that have been seen, and need not be set *a priori*. If the price changes slowly, then a large number of samples are received, and a high resolution estimate is obtained. However, if the price changes rapidly, then only a small number of packets will still yield a good estimate \hat{p} . To achieve this, random marking [1], [2], [3], [5] would need an adaptive averaging interval, and binary

marking [7] or the approach of [8] would need an adaptive quantizer resolution.

Now consider a more sophisticated probe sequence. Many operating systems assign sequential IPid values to packets. This allows the sequence of probes types to form a sequence of "bit-reversed counting" (BRC) values, that is, 100...010...0110...0...0, preceded by a "binary point". That is, using the sequence $R(1), R(2), R(3), \dots$, where $R(\sum_{i=0}^{\infty} a_i 2^i) = \sum_{i=0}^{\infty} a_i 2^{-i}$ for any sequence of bits $\{a_i\}$.

If the probe type sequence is bit-reversed counting, ($R(1), R(2), R(3), \dots$) then after $k = 2^j - 1$ probe packets, probes $2^{-j}, 2 \times 2^{-j}, \dots, 1 - 2^{-j}$ will have been received. This divides the possible values for p into uniformly spaced "uncertainty intervals" of width 2^{-j} . ADPM knows which interval the price is in, but not where within that interval. Thus the estimation error after k packets is bounded above by

$$|\hat{p} - p| < 2^{-\lfloor \log_2 k \rfloor} \in [1/k, 2/k]. \quad (3)$$

Thus, BRC bounds the error without increasing its mean.

In practice, prices change. It can be shown [11] that if the price increases by δ per packet, the MSE is 2δ .

V. IMPLEMENTATION DETAILS

IP has two ENC bits (RFC 3168), which a source sets to 00 to indicate that it does not understand ECN, or 01 or 10 to indicate an unmarked packet. Routers mark packets with 11 to indicate congestion, which the source must treat like a packet loss. Sending 11 frequently [5], [6], [7], [8] will cripple standards-compliant flows. To prevent this, ADPM uses 01 to indicate no mark, and 10 to indicate a mark due to pricing. As such, ADPM is "ECN-friendly".

The function F that maps IPid values into probe types can be implemented either at the source or in the routers. Many operating systems assign sequential IPid values to packets. The F corresponding to bit reverse counting can therefore be implemented in the routers in a simple and scalable way by simply reversing the order of the IPid bits and comparing this with the price. Other operating systems use uniformly distributed pseudo-random values (*e.g.* NetBSD). The routers can reverse the bits before the comparison with the price, without affecting the uniform distribution for the resulting probe type. If necessary, IPid values can be replaced at the source without changing their primary function, providing nearby packets have unique identifiers.

Note that the mapping of prices and IPid to $[0, 1]$ is purely a mathematical convenience, and does not limit the actual range of prices. Either mapping may be nonlinear. In particular, the mapping of prices can be selected to get better quantization and faster estimation of the true price at the receiver.

VI. NUMERICAL RESULTS AND CONCLUSION

In this section, we first numerically evaluate the estimation error in ADPM after k probe packets have been received by the receiver, and then compare it with related schemes. We consider three kinds of probe type sequence: pure random, "pure BRC" (with bit-reversed thresholds $R(1), R(2), \dots$) and "random BRC" (with thresholds $R(d), R(d+1), \dots$ and d uniformly distributed on $[0, 65535]$). Random BRC reflects the

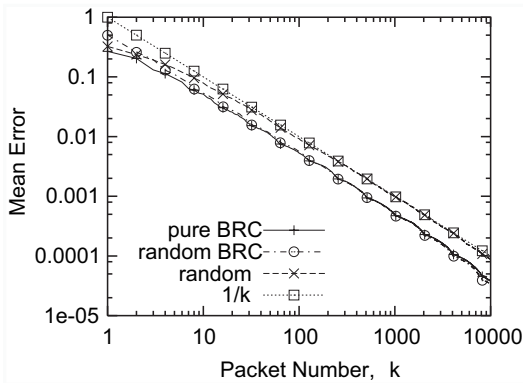


Fig. 1. Mean estimation error of ADPM after k packets have been received.

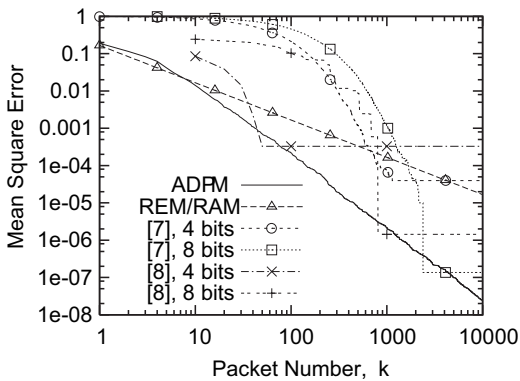


Fig. 2. MSE of different marking schemes after receiving k packets.

need for connections to start with unique IPid values. The maximum link price is uniformly distributed on $[0,1]$.

Figure 1 plots the mean estimation error against k . For pure BRC, results are averaged over 1000 prices; for the random schemes, they are averaged over 256 different random probe sequences, each using 100 different prices. For all of the probe sequences considered, the estimation error of ADPM decreases as the reciprocal of the number of probe packets received. This is because marking in ADPM is based on inequality of thresholds and thus each probe packet can potentially provide useful information for price estimation. Even after a small number of probe packets are received, a useful estimation on the price can already be obtained, and there is no arbitrary floor to the precision. Figure 1 also shows that BRC outperforms random probing. This is because BRC generates a systematic probe sequence which does not leave large ranges unprobed. Random BRC performs like random probing initially. However, as more packets are received, its performance approaches that of pure BRC because it too is systematic. The accuracy of (2) is also verified by the simulation results which closely match the curve $1/k$.

The mean square error (MSE) performance of ADPM with purely random probes is compared with alternative marking schemes in Figure 2. The simulations use 10 routers, with independent prices. Our scheme attempts to estimate the same maximum price as [8]. The other schemes REM [5], RAM [6], and [7] estimate the sum of prices. For all schemes, the actual price was mapped to a "price", p , in $[0,1]$; for [5] this is provided by the exponential mapping $p = \exp(-\phi \text{ price})$, and for [6] and [7], it is the sum of prices scaled by $1/10$. To ensure

a fair comparison of the different estimators, all schemes (except [7]) have link prices distributed such that p is uniform on $[0,1]$. Since [7] estimates each link price separately, and then adds them up, all link prices were uniformly distributed on $[0,1]$ for this scheme.

The scheme of [8] estimates the maximum price as 0 until a mark is set by a router. The curve "[8], n -bit" used 2^n quantization levels. For [7], each bit was initially taken as 0.5, until it is probed. The curve for "[7], n -bit" used $30n$ probe types allowing paths of up to 30 routers with n bit quantisation.

The results for REM [5] and RAM [6] are the closed form expression, $1/(6k)$, which is the average of $p(1-p)/k$ for p uniform in $[0,1]$. The results for REM/RAM are optimistic, since the averaging interval is always set to k ; with fixed interval L , a floor of $1/(6L)$ is hit for all $k > L$.

The results for ADPM and REM/RAM show power law behaviour, with ADPM yielding significantly lower error after a moderate number of packets. The results of [7] and [8] are more complex. When only a small fraction of the probe types have been received, it is likely that some high-order bits have not been received [7] or the price has not been "hit" [8], yielding a large MSE. The error then drops rapidly after sufficient probe types have been received. However, because a fixed quantiser is used, there is a square-error floor. This shows the trade-off inherent in these two schemes between responsiveness and steady state accuracy. The floor is about 10 times lower for [7] than [8], due to the rescaling to the range $[0,1]$. The scheme of [8] outperforms ADPM when k exactly equals the number of quantization levels. This is because the quantisation of [8] rounds prices to the nearest quantization level, while ADPM's implicit quantisation always rounds in one direction, toward the value closest to the current estimate.

ADPM outperforms the alternative schemes in most circumstances, and more importantly removes the need for tuning the number of quantization levels, or a measurement interval.

REFERENCES

- [1] F. Kelly, "Charging and rate control for elastic traffic," *European Trans. Telecommun.*, vol. 8, pp. 33–37, 1997.
- [2] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *J. Op. Res. Soc.*, vol. 49, pp. 237–378, 1998.
- [3] S. H. Low and D. E. Lapsley, "Optimization flow control I: basic algorithm and convergence," *IEEE/ACM Trans. Networking*, vol. 7, pp. 861–875, Dec. 1999.
- [4] B. Wyrowski, L. L. H. Andrew, and M. Zukerman, "MaxNet: a congestion control architecture for scalable networks," *IEEE Commun. Lett.*, vol. 7, pp. 511–513, Oct. 2003.
- [5] S. Athuraliya, V. H. Li, S. H. Low, and Q. Yin, "REM: active queue management," *IEEE Network*, vol. 15, pp. 48–53, May/June 2001.
- [6] M. Adler, J.-Y. Cai, J. K. Shapiro, and D. Towsley, "Estimation of congestion price using probabilistic packet marking," in *Proc. IEEE INFOCOM 2003*, vol. 3, pp. 2068–2078.
- [7] R. W. Thommes and M. J. Coates, "Deterministic packet marking for congestion price estimation," in *Proc. IEEE INFOCOM 2004*.
- [8] H.-K. Ryu and S. Chong, "Deterministic packet marking for max-min flow control," *IEEE Commun. Lett.*, vol. 9, pp. 856–858, Sept. 2005.
- [9] A. Belenky and N. Ansari, "IP traceback with deterministic packet marking," *IEEE Commun. Lett.*, vol. 7, pp. 162–164, Apr. 2003.
- [10] J. F. C. Kingman, *Poisson Processes*. Oxford Science Publications, 1993.
- [11] L. L. H. Andrew, S. V. Hanly, S. Chan, and T. Cui, "Efficient deterministic packet marking," available at <http://netlab.caltech.edu/~lachlan/dmtm.pdf>.