

Analysis of the Contention Access Phase of a Reservation MAC Protocol for Wide-Area Data Intensive Sensor Networks

Arindam K. Das, Sumit Roy, Abhijit Mahalanobis

Abstract— We propose a contention based reservation MAC protocol for a collaborative sensing scenario involving a set of surveillance UAVs communicating with a hub. Data transmission rights are secured via distributed contention access between the UAV nodes for K available mini-slots. Each node chooses a slot with probability p independently of the others and is only allowed one attempt in a frame. We investigate the optimal choice of p which maximizes the one-shot probability of success (or alternately, the expected number of successes), β , over the K minislots. We show that β is prone to local maxima and develop an empirical formula based on goodness-of-fit which matches the numerically obtained values closely.

I. INTRODUCTION

Use of networked unmanned aerial vehicles (UAV's) is becoming increasingly common for various tactical airborne surveillance scenarios. For example, a swarm of geographically dispersed UAV's may be deployed to observe a terrestrial area for target detection and recognition via typical imaging (RF and/or optical backscatter) mechanisms. Each UAV node is assumed to communicate with a central node (*e.g.*, a low earth satellite or other sophisticated high-altitude aircraft) in a star topology. The fusion center, typically with significant processing power, then performs on-board processing and forwards a composite image or sequence of images along with intermediate decisions to a remote (typically ground-based) command center for final evaluation/confirmation and follow-up action (see Figure 1). We assume that data at different sensors are independent (hence equally important to the overall decision) and consequently, no prioritized channel access is necessary for any subset of the UAV's. Additionally, we assume that each node has only one data packet to transmit to the fusion center.

Some distinguishing characteristics of our targeted application scenarios can be summarized as follows: (a) the size and duration of data packets are large and therefore collisions are expensive in terms of latency and wasted bandwidth, (b) the load profiles which are triggered by an event of interest are extremely bursty, thereby invalidating usual Poisson traffic models adopted for MAC performance analysis [1], [2], [3], (c) the maximum propagation delay could be potentially very large and therefore carrier sensing based MAC protocols may not be efficient, and (d) the success of such missions depends critically

Corresponding author: A.K. Das is a Post Doctoral Research Associate with the Department of Aeronautics and Astronautics, University of Washington, Box 352500, Seattle, WA 98195. *e-mail:* arindam@u.washington.edu.

S. Roy is with Department of Electrical Engineering, University of Washington, Box 352500, Seattle, WA 98195. *e-mail:* sroy@u.washington.edu

A. Mahalanobis is with Lockheed Martin Corporation, 5600 Sand Lake Rd., Orlando, FL 32819. *e-mail:* abhijit.mahalanobis@lmco.com

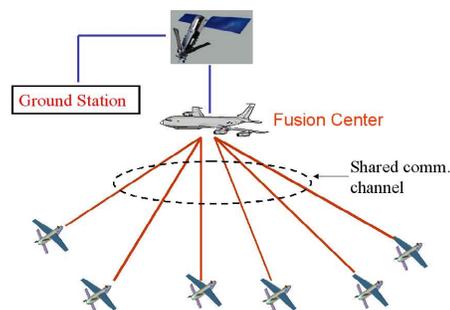


Fig. 1. Communications architecture for a swarm of UAV's on an event detection/monitoring mission.

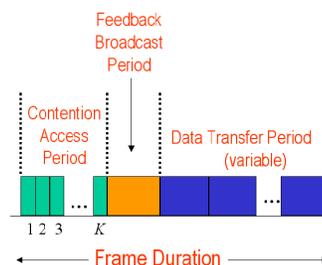


Fig. 2. Structure of a frame: K minislots for contention access (called a *contention window*), a feedback broadcast period from the fusion center (sink) and a variable length data transfer period. Nodes which are successful during the contention access period transmit their data packets in the data transfer period.

on how quickly the field images are received at the fusion center. Consequently, a key design challenge for a MAC protocol is to regulate channel contention between users to ensure that *enough* information is successfully received at the fusion center with minimal latency.

II. DESCRIPTION OF THE RESERVATION PROTOCOL

The proposed protocol is contention based variable length TDMA whose *frame* structure is shown in Figure 2. A frame comprises a sequence of K minislots for contention access (called a *contention window*), a feedback broadcast period and a variable length data transfer period. Nodes, which are assumed to be time-synchronized, use the contention window to compete for a transmission slot during the data transfer period. During the contention period, each node attempts to pick one of the K minislots with probability p , independently of the others, and transmit a reservation packet to the fusion center. Each

node may choose at most one minislot per frame. Therefore, a node which *chooses* minislot k backs out of contention till the end of the current contention window (and possibly, till the end of the current superframe if its reservation request was successful), *irrespective of whether it was successful or not*. Since carrier sensing is not assumed and the fusion center does not send a feedback after every contention minislot (to minimize latency), nodes which transmit a reservation request in slot k may not know right away (this may happen, *e.g.*, if the distance between a pair of transmitting nodes is sufficiently large) whether its transmission was successful or not. However, once feedback is received from the fusion center after a batch of K contention slots¹, all nodes which transmitted a reservation request will know if their requests were successful, and if not, they join the contention pool at the beginning of the next frame. If a node does not choose any of the K slots in the current frame, it tries again in the next frame. Any node winning a contention slot is allocated a data slot in the same frame.

We assume that each image is transmitted as a single data packet (the duration of a data transmission slot might therefore be much larger than the duration of a contention minislot) and that all data slots are of equal length. Also, necessary guard times are assumed to be built into the durations of the contention/data slots. The number of transmission slots within a data transfer period is variable and equal to the number of winners during the contention phase. Interleaved between the contention window and the data transfer period is a feedback broadcast period in which the fusion center broadcasts the following control information to the sensing nodes in a single broadcast packet: (a) the transmission schedule for the data transfer period, specifying which node transmits first, second, *etc*² and (b) information on the outcomes of the minislot contention. This may be either explicit (binary or ternary) or implicit, in which case the fusion center may report the number of contenders to be used for computing the optimal slot selection probability p^* for the next frame or it may compute p^* itself and include it in the broadcast packet.

Depending on the number of initial contenders and the value of K chosen, a sequence of several frames (which we call a *superframe*, see Figure 3) may be needed for reliable decision making at the fusion center. Successive frames are separated by a control packet carrying binary feedback broadcast by the fusion center to signal the end (or not) of the current superframe. The decision whether to terminate the current superframe is made based on the output of a suitable “decision rule” executed at the fusion center. We assume that transmitted data packets arrive without errors/losses, which obviates the need for acknowledgements from the fusion center. Acknowledgements, if required, can be provided in the control packets. This will allow a successful contender in frame m to re-contend in frame $m + 1$ if its data packet is lost or corrupted beyond recovery.

¹Alternately, feedback slots can be provided immediately after each contention slot, which would allow an unsuccessful node in contention slot k to re-contend for the subsequent slot(s) till it is unsuccessful. We however have not adopted this option in order to minimize the delay.

²Note that it is possible for the data transfer period to be empty, which will happen if there are no winners during the contention window. Since the number of data slots in a frame is variable, the feedback is also necessary for nodes to maintain time synchronization.

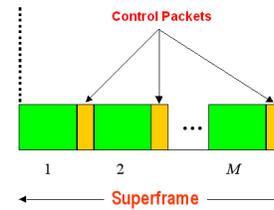


Fig. 3. Structure of a superframe comprising M frames. Interleaved between successive frames is a control packet which is broadcast by the fusion center to signal the end (or not) of the current superframe. If needed, acknowledgements on the data transmissions in the m^{th} frame can also be provided in the control packet succeeding it.

III. ANALYSIS OF THE CONTENTION ACCESS PERIOD: SINGLE CHANNEL CASE

Let N denote the number of contending nodes at the beginning of the contention access period and K ($2 \leq K \leq N$) the number of minislots in a contention window. *Throughout this paper, we will assume that $N \geq 2$.* At the beginning of each minislot, say k , each node chooses to transmit in it with probability (w.p.) p , independently of the other contending nodes. We assume that $p \leq 1/K$, the reasons for which are discussed subsequently. As mentioned before, each contending node may choose at most one minislot in every frame. Consequently, a node which chooses minislot k backs out of contention till the end of the current contention window (and possibly, till the end of the current superframe), regardless of whether it is successful or not. If it does not choose minislot k , it proceeds to minislot $k + 1$ and chooses it w.p. p . We say that minislot k is *successful* if one and only one node chooses it, *idle* if no node chooses minislot k and *colliding* if more than one node chooses minislot k .

Let $P_{succ}(k)$ denote the probability of success in any minislot k . Clearly, for $k = 1$,

$$P_{succ}(1) = Np(1-p)^{N-1} \quad (1)$$

For any other minislot $2 \leq k \leq K$, the probability of success is a function of the ‘number of nodes left to choose from minislots $k, k + 1, \dots, K$ ’, a random variable which we denote by $X_{k:K}$. The p.m.f of $X_{k:K}$ is given by:

$$\Pr(X_{k:K} = m) = \binom{N}{m} [(1-p)^{k-1}]^m \times [1 - (1-p)^{k-1}]^{N-m} \quad (2)$$

where $m = 1, 2, \dots, N$. Therefore, for $2 \leq k \leq K$, it can be shown that:

$$P_{succ}(k) = \sum_{m=1}^N \Pr(1 \text{ node chooses minislot } k \mid X_{k:K} = m) = Np(1-p)^{k-1} [1 - p(1-p)^{k-1}]^{N-1} \quad (3)$$

Typically, $P_{succ}(k)$ exhibits bimodal behavior for higher values of k when N is large. For small N , however, $P_{succ}(k)$ can exhibit bimodal behavior even for relatively smaller values of k .

Let β be the aggregate contention success rate (or alternately, the expected number of successes in a contention frame) defined as follows:

$$\beta = \sum_{k=1}^K P_{succ}(k). \quad (4)$$

The normalized version, β/K , can be interpreted as the average probability of success of any minislot. Substituting (1) and (3) in (4), we have:

$$\beta = \sum_{k=0}^{K-1} Np(1-p)^k [1-p(1-p)^k]^{N-1} \quad (5)$$

We now make a ‘large N , small $p(1-p)^k$ ’ approximation (which holds if p is small and N is large, as demonstrated subsequently):

$$[1-p(1-p)^k]^{N-1} \approx e^{-(N-1)p(1-p)^k} \approx e^{-Np(1-p)^k} \quad (6)$$

and substitute (6) in (5) to obtain,

$$\beta \approx \sum_{k=0}^{K-1} Np(1-p)^k e^{-Np(1-p)^k} \quad (7)$$

While (7) may not appear to be any more tractable than (5), we have observed that maximizing the former w.r.t. p using a line search routine (e.g. the medium-scale option of the *fmincon* function in MATLAB³) is easier. On a related note, we would like to point out that β is typically not concave over $p \in (0, 1)$ and finding its global optimum is hard. We will address this issue and its ramifications in greater detail subsequently.

Table I shows the exact (5) and approximate (7) values of β for several (N, K) combinations. As can be observed from the figures, for a given K , the approximation is extremely accurate for large N . Also, the optimal value of p , p^* , is approximately equal to $1/N$ (which, in hindsight, validates approximation (6)), which is also confirmed by numerical optimization of (7) using MATLAB’s *fmincon* function (see Table I). While it is widely known that $p^* = 1/N$ for $K = 1$, simulations suggest that the approximation $p^* \approx 1/N$ holds well for any value of $K \in [2, 1.7N]$ since the corresponding drop in β is less than 3%, even though the approximation itself could be about 35% from the optimal (see Figure 4a). For $K \geq 1.8N$, however, p^* resembles a heavy-tailed distribution and the approximation (and corresponding reduction in β) gets progressively worse (see Figure 4b). Observe also that for $K \in [2, 1.7N]$, $1/N$ always under-estimates p^* , whereas, for $K \geq 1.8N$, $1/N$ over-estimates p^* . A refined empirical model of p^* as a function of K and N is presented at the end of this section.

We now revisit our assumption $p \leq 1/K$. As illustrated in Figure 5, the plot of β vs. p can exhibit local maxima if N is allowed to increase for a given K (or, if K is allowed to decrease for a given N). This is not surprising since $P_{succ}(k)$ can be bimodal as mentioned previously. However, extensive simulations suggest that the optimum (of β) always occurs at the first

³Of course, the expression for β needs to be negated if *fmincon* is used since β is to be maximized and *fmincon* is a minimization routine.

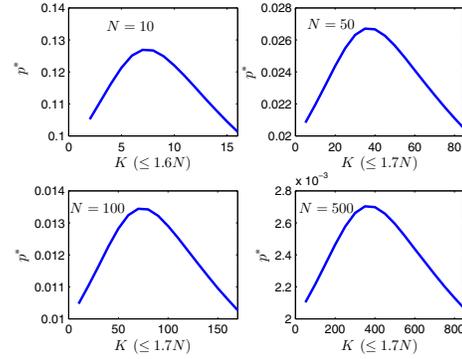


Fig. 4a. Plot of p^* as a function of $K (\leq 1.7N, \text{ except for } N = 10)$. Empirically, the maximum deviation of p^* from the approximation $p^* \approx 1/N$ occurs when K is about 70% of N , for any N . Observe also that $1/N$ always under-estimates p^* (by as much as 35% for $N = 500$) for the range of K shown. However, the drop in β due to the approximation $p^* \approx 1/N$ is less than 3% for any N , despite the fact that the deviation of the approximation $1/N$ from the optimal could be as high as 35% for large N . The plots are almost identical for the exact (5) and approximate computations (7).

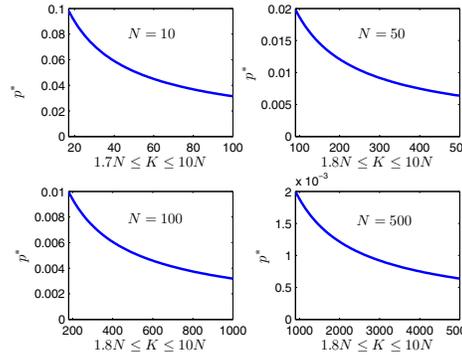


Fig. 4b. Plot of p^* as a function of $K (1.8N \leq K \leq 10N, \text{ except for } N = 10)$. Observe that the approximation $1/N$ always over-estimates p^* for this range of K . Also, the approximation ratio gets progressively worse as K increases. The plots are almost identical for the exact (5) and approximate computations (7).

peak which corresponds to $p \approx 1/N$. Consequently, the constraint $p \leq 1/K$ is adequate for capturing the first peak of the β curve when $K \leq N$. Additionally, imposing this constraint effectively shrinks the search domain leading to faster convergence of any numerical optimization routine. When K is comparable to or exceeds N , it is possible that the true optimum will be missed if the constraint $p \leq 1/K$ is imposed. Fortunately, since the β curve is much more well-behaved (tending towards unimodal) when N and K are comparable (e.g., if $N \leq 2K$), the constraint should be relaxed to, say, $p \leq 2/K$ (or dropped altogether), which we have found is adequate for capturing the true optimum. For example, the optimal p with the constraint $p \leq 1/K$ is 0.04 for $N = K = 25$, while the true optimum is 0.0506 (which is what MATLAB’s *fmincon* function returns when the constraint $p \leq 2/K$ is imposed).

In Figure 6, we have plotted $P_{succ}(k)$, evaluated at $p = p^*$, vs. k for $K = 25$. For large N and $K \ll N$, it can be seen that the probability of success in any minislot is almost constant

TABLE I

Optimal p using the exact expression of β (5) and the approximation (7). The constraint $p \leq 1/K$ was used for instances satisfying $N > 2K$. For all other instances, $p \leq 2/K$ was used. MATLAB's *fmincon* function was used for these computations.

N	K	p^* - using (5)	p^* - using (7)
25	5	0.043435	0.043444
	15	0.051838	0.051941
	25	0.050603	0.050631
50	5	0.020831	0.020832
	25	0.025458	0.025477
	50	0.025630	0.025637
100	25	0.011337	0.011338
	50	0.012815	0.012820
	100	0.012899	0.012901
200	50	0.005685	0.005685
	100	0.006429	0.006430
	200	0.006471	0.006471

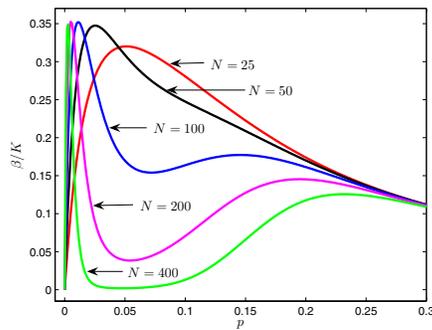


Fig. 5. Plot of β/K vs. p to illustrate the presence of local maxima. For all values of N , $K = 25$.

(strictly speaking, with a small droop). This is intuitively justified since the number of contenders is expected to remain large enough (so that the chance of a latter slot going idle is almost comparable to that of an initial slot) even towards the end of the contention window when $K \ll N$. Specifically, if Y_k denotes the number of nodes which have chosen minislot k , it can be shown that, for $k = 1, 2, \dots, K$,

$$\begin{aligned} \Pr(Y_k = n) &= \sum_{m=n}^N \binom{m}{n} p^n (1-p)^{m-n} \times \Pr(X_{k:K} = m) \\ &= \binom{N}{n} [p(1-p)^{k-1}]^n [1 - p(1-p)^{k-1}]^{N-n} \end{aligned}$$

and consequently,

$$E[Y_k] = Np(1-p)^{k-1} \approx 1, \text{ for } p = p^*, K \ll N, \text{ large } N$$

However, as $K \rightarrow N$ from below or when $K > N$, the chances of a latter minislot going idle will be significantly higher, since a substantial number of nodes would have chosen one of the previous minislots but those remaining in the contention pool (if any) would not be able to benefit from it because of the rather

small value of p^* , which, as mentioned in the previous paragraph, is approximately $1/N$. In terms of Y_k , we can therefore expect $E[Y_k]$ to be much smaller than 1 as $K \rightarrow N$ from below or when $K > N$.

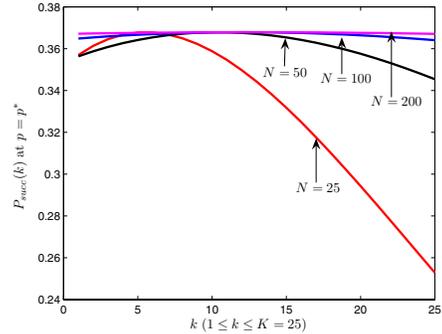


Fig. 6. Plot of $P_{succ}(k)$, evaluated at $p = p^*$, as a function of k for $K = 25$. As N increases for a fixed K , the probability of success of any minislot is almost constant. For smaller values of N , $P_{succ}(k)$ is maximum at an intermediate slot, e.g., at $k = 5$ for $N = 25$ and $k = 10$ for $N = 50$.

Previously, we have mentioned that the approximation $p^* \approx 1/N$ is reasonable if $K \in [1, 1.7N]$ (or $K \in [1, 1.6N]$ for small N) since the consequent drop in beta is less than 3%. We now propose empirical formulae for estimating p^* as a function of K and N when $K \leq 10N$. These are based on analysis of parameters from extensive goodness-of-fit tests, appear to provide excellent estimates of p^* over the entire range of K considered. Let K_1 denote the maximum value of K such that $1/N$ under-estimates p^* for all values of $K \in [1, K_1]$. Also, let K_2 be the value of K (in the range $[1, K_1]$) at which the deviation of p^* from $1/N$ is the maximum. From Figure 4a, it can be seen that:

$$K_1 = \begin{cases} 1.6N, & \text{for small } N \\ 1.7N, & \text{for large } N \end{cases} \quad \text{and} \quad K_2 = 0.7N \quad (8)$$

For large N , a good empirical approximation of p^* at $K = K_2$ is

$$p^*(K = K_2) \approx (1 + e^{-1})/N = 1.3679/N. \quad (9)$$

Assuming $p^*(K = 1) = p^*(K = K_1) = 1/N$, it can be shown that a piecewise linear approximation of p^* for $K \in [1, K_1]$ is given by:

$$p^{approx} = \begin{cases} \frac{1}{N} + \left(\frac{e^{-1}}{K_2-1}\right) \left(\frac{K-1}{N}\right), & 1 \leq K \leq K_2 \\ \frac{1}{N} - \left(\frac{e^{-1}}{K_1-K_2}\right) \left(\frac{K-K_1}{N}\right), & K_2 < K \leq K_1 \end{cases} \quad (10)$$

In Figure 7a, we have plotted p^* (solid lines) and p^{approx} (dashed lines) defined in (10) for $K \in [1, K_1]$. For large N , the approximation is extremely good, despite being slightly offset over the negative slope section of the curve. This is because we assumed p^* to be exactly equal to $1/N$ at $K = K_1$, but a close examination of Figure 7a reveals that p^* is slightly greater than $1/N$ at $K = K_1$. When N is relatively small, the error in the approximation is due to the fact that $(1/N)(1 + e^{-1})$ overestimates the optimal at $K = K_2$. For example, when $N = 10$,

$(1/N)(1 + e^{-1}) = 0.1368$ but $p^*(K = K_2) = 0.1268$. For small N , we have found that a better empirical approximation of $p^*(K = K_2)$ is (compare with (9))

$$p^*(K = K_2) \approx \left(2 - e^{-e^{-1}}\right)/N = 1.3078/N, \quad (11)$$

which evaluates to 0.1308 for $N = 10$. Similarly, for $N = 25$, $K_2 = 18$ and the r.h.s of (11) evaluates to 0.0523 compared to the optimal which is $p^*(K = K_2) = 0.0528$.

Using (11) and assuming $p^*(K = 1) = p^*(K = K_1) = 1/N$ as before, a better piecewise linear approximation of p^* for small N and $K \in [1, K_1]$ is:

$$p^{approx} = \begin{cases} \frac{1}{N} + \left(\frac{1-e^{-e^{-1}}}{K_2-1}\right) \left(\frac{K-1}{N}\right), & 1 \leq K \leq K_2 \\ \frac{1}{N} - \left(\frac{1-e^{-e^{-1}}}{K_1-K_2}\right) \left(\frac{K-K_1}{N}\right), & K_2 < K \leq K_1 \end{cases} \quad (12)$$

In Figure 7b, we have plotted p^* (solid lines) and the new p^{approx} (dashed lines) defined in (12) for $N = 10$. Clearly, the approximation is now at most 3% from the optimal compared to 8% previously.

We now consider the range $K_1 < K \leq 10N$. From Figure 4b, it appears that p^* follows a power law when $K \in [K_1, 10N]$. Indeed, a log-log transformation reveals that $\log(p^*)$ is almost linear⁴ with a slope (denoted by θ) given by:

$$\theta = \begin{cases} -0.64, & \text{for small } N \\ -0.65, & \text{for large } N \end{cases} \quad (13)$$

Using (13) and $p^*(K = K_1) = 1/N$, an approximation of p^* can easily be derived to be:

$$p^{approx} = (1/N) (K/K_1)^\theta, \quad K_1 < K \leq 10N \quad (14)$$

In Figure 7c, we have plotted p^* (solid lines) and p^{approx} (dashed lines) defined in (14) for $K \in [K_1, 10N]$. For the range of K considered, the approximation is at most 6% from the optimal, with the maximum deviation occurring at $K = 3.5N$ for $N = 10$ and $K = 3.3N$ for $N = 50/100/500$.

IV. CONCLUSION

We have proposed a contention based variable length reservation MAC protocol for fast information extraction from a wide-area sensor monitoring network such as a swarm of UAV's. Analysis was presented for numerical computation of the optimal slot selection probability p^* during the contention phase of the protocol which maximizes the expected number of successful transmissions within the contention window. Empirical formulae have also been derived for p^* which match the numerically obtained solutions closely.

REFERENCES

- [1] Y.C. Tay, K. Jamieson and H. Balakrishnan, "Collision minimizing CSMA and its Applications to Wireless Sensor Networks", *IEEE Journal on Selected Areas in Communications*, August 2004.
- [2] V. Paxson and S. Floyd, "Wide area traffic: The failure of Poisson modelling", *IEEE/ACM Trans. Networking*, vol. 3, June 1995, pp. 226-244.
- [3] K. Jamieson, H. Balakrishnan and Y.C. Tay, "Sift: A MAC protocol for event-driven wireless sensor networks", *MIT Lab. Comput. Sc., Tech. Rep. 894*, May 2003. Available online at: <http://www.lcs.mit.edu/publications/pubs/pdf/MIT-LCS-TR-894.pdf>

⁴Strictly speaking, $\log(p^*)$ is slightly curvilinear over the range of K considered, but the linear under-approximation is adequate.

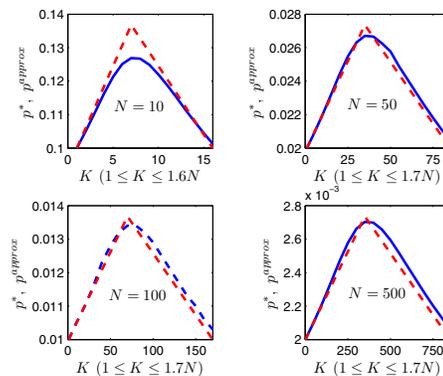


Fig. 7a. Plot of p^* (solid lines) and its piece-wise linear approximation (dashed lines) defined in (10) as a function of K and N , when $K \leq K_1$ (8). Observe that the approximation is slightly off from the optimal over the negative slope section of the curve. This is because we assumed p^* to be exactly equal to $1/N$ when $K = K_1$, but a close examination reveals that p^* is slightly greater than $1/N$ at $K = K_1$. For relatively large N , the approximation is within 4% of the optimal. For smaller values of N , it is within 8% of the optimal.

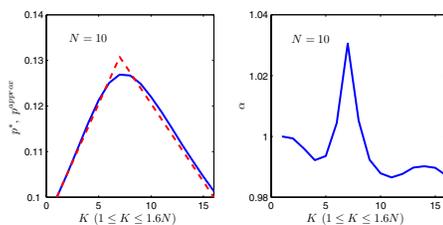


Fig. 7b. Plot of p^* (solid lines), its piece-wise linear approximation (dashed lines) defined in (12) and the approximation factor $\alpha = p^{approx}/p^*$ for $N = 10$. At its worst, the approximation is now roughly 3% from the optimal, compared to 8% in Figure 7a.

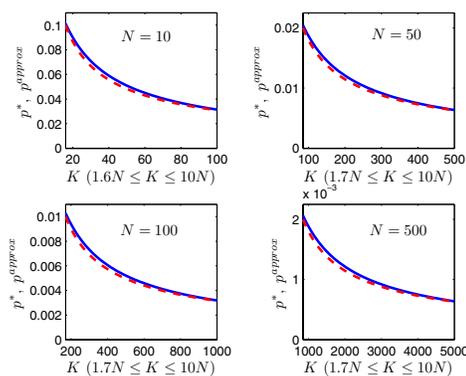


Fig. 7c. Plot of p^* (solid lines) and its power law approximation (dashed lines) defined in (14) as a function of K and N , when $K_1 \leq K \leq 10N$. For $N = 10$, we chose $\theta = -0.64$. For all other N , $\theta = -0.65$. For all N , the approximation is at most 6% from the optimal, with the maximum deviation occurring at $K = 3.5N$ for $N = 10$ and $K = 3.3N$ for $N = 50/100/500$.