Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational and Applied Mathematics 200 (2007) 154–167
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

Some upper and lower bounds on the coupon collector problem

S. Shioda*

Urban Environment & Systems, Faculty of Engineering, Chiba University, 1-33 Yayoi, Inage, Chiba 263-8522, Japan

Received 26 September 2005; received in revised form 12 December 2005

Abstract

The classical coupon collector problem is closely related to probabilistic-packet-marking (PPM) schemes for IP traceback problem in the Internet. In this paper, we study the classical coupon collector problem, and derive some upper and lower bounds of the complementary cumulative distribution function (ccdf) of the number of objects (coupons) that one has to check in order to detect a set of different objects. The derived bounds require much less computation than the exact formula. We numerically find that the proposed bounds are very close to the actual ccdf when detecting probabilities are set to the values common to the PPM schemes.

MSC: 60C05

Keywords: Coupon collector problem; Bounds; Distribution function; Denial of service attack; IP traceback

1. Introduction

Consider a box which contains N types of numerous objects. An object in the box is repeatedly sampled on a random basis. Let p_i (>0) denote the probability that a type-i object is sampled. The successive samplings are statistically independent and the sampling probabilities, $p_1, p_2, \ldots, p_N$, are fixed. When a type-i object is sampled for the first time, we say that a type-i object is detected. To find the number of samplings required for detecting a set of object types (say, object types indexed by $i = 1, \ldots, n$) is traditionally called *coupon collector problem*.

The study of the coupon collector problem has a long history and can be found in many texts, see e.g., Feller [2]. The coupon-collector problem has been found to be a useful mathematical model for a variety of natural phenomena and engineering applications. A concise explanation about typical applications of the coupon collector problem was found in [7]. Concerning recent studies on the coupon collector problem, please see [1,7,4,6].

To explain our motivation of this study, let us explain an application of the coupon collector problem to the field of telecommunication. It is widely recognized that the denial-of-service (DoS) attack is one of the hardest security problems in the Internet. Identifying a DoS attacker is usually called the *IP traceback* problem. Savage et al. [8] proposed a promising solution, called *probabilistic packet marking* (PPM), to the IP traceback problem. Their solution is to simply let routers probabilistically mark packets with partial information of an attack path during packet forwarding. Although each packet represents only partial information of the attack path, a victim can construct the entire path by combining the information conveyed by a modest number of marked packets.

* Tel.:/fax: +81 43 290 3237.
E-mail address: shioda@faculty.chiba-u.jp.

0377-0427/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2005.12.011

In a PPM scheme, the number of packets that the victim should receive to reconstruct the attack path is equivalent to the number of samplings required for detecting a set of object types in the coupon collector problem. (In what follows, we referred to the number of samplings required for detecting a set of object types as *detecting cost*.) Thus, analyzing the efficiency of the PPM scheme comes down to solving the coupon collector problem. In particular, the false negative ratio of a PPM scheme is given by the complementary cumulative distribution function (ccdf) of the *detecting cost* (see Section 4). Thus, it is crucial to compute the ccdf of the *detecting cost* for evaluating the efficiency of PPM schemes. As shown in Section 2, however, the ccdf of the *detecting cost* is not easy to compute exactly because of its cumbersome combinatorial formula.

The aim of this paper is to find some techniques to calculate the ccdf of the *detecting cost* with small computational time and with accuracy sufficient for practical use. To this end, in this paper, we derive some upper and lower bounds of the ccdf of the *detecting cost*. The derived bounds are much more suited to numerical computation than the exact formula. In addition to this, we find that these bounds are very close to the actual ccdf when sampling probability is set to the values typical to the PPM schemes.

The organization of the paper is as follows. In Section 2, we briefly summarize some fundamental formulas of the coupon collector problem. In Section 3, we derive the upper and lower bounds of the complementary cumulative distribution function (ccdf) of the *detecting cost*. In Section 4, we explain how the ccdf of the detecting cost is related to the attacking-path-detection efficiency of PPM schemes, and show the results of some numerical experiments to conform the tightness of the bounds.

2. Some fundamental formulas of coupon collector problem

In this section, we summarize some fundamental formulas of the coupon collector problem. Let X_i be the number of samplings required for detecting a type-i object and define

$$X \stackrel{\text{def}}{=} \max\{X_1, \ldots, X_n\}, \quad n \leqslant N.$$

Note that X corresponds to the *detecting cost* for object types indexed by $i = 1, \ldots, n$.

Let J denote a subset of set $\{1, 2, \ldots, n\}$, and $|J|$ denote cardinality of J. Furthermore, we let

$$P_J \stackrel{\text{def}}{=} \sum_{j \in J} p_j.$$

Although some representation of the distribution function of X seems to be available in some literatures, we here give a representation of the ccdf of X with its proof for completeness of the paper.

Lemma 1.

$$P[X > k] = \sum_{i=1}^{n} (-1)^{i+1} \sum_{J;|J|=i} (1 - P_J)^k. \tag{1}$$

Proof. Let $A_j^{(k)}$ denote the event that all of k sampled objects are not type j. Now define

$$S_J^{(k)} \stackrel{\text{def}}{=} \bigcap_{j \in J} A_j^{(k)}.$$

By the inclusion–exclusion principle [2],

$$P\left[\bigcup_{j=1}^{n} A_j^{(k)}\right] = \sum_{j=1}^{n} (-1)^{j-1} \sum_{J;|J|=j} P[S_J^{(k)}].$$

Since $P[S_J^{(k)}] = (1 - P_J)^k$, it follows that,

$$\begin{aligned} P[X > k] &= P\left[\bigcup_{j=1}^{n} A_j^{(k)}\right] \\ &= \sum_{j=1}^{n} (-1)^{j-1} \sum_{J;|J|=j} P[S_J^{(k)}] \\ &= \sum_{j=1}^{n} (-1)^{j+1} \sum_{J;|J|=j} (1 - P_J)^k, \end{aligned}$$

which completes the proof.

In particular, if $p_1 = \cdots = p_n = p$, then we have a simpler representation such as

$$P[X > k] = \sum_{i=1}^{n} (-1)^{i+1} \binom{n}{i} (1 - \mathrm{i}p)^k. \tag{2}$$

From representation (1), the expectation and the generating function of X are readily obtained

$$E[X] = \sum_{i=1}^{n} (-1)^{i+1} \sum_{J;|J|=i} \frac{1}{P_J}, \tag{3}$$

$$g(z) = \sum_{k=1}^{\infty} z^k P[X = k] = \sum_{i=0}^{n} (-1)^i \sum_{J;|J|=i} \frac{1 - z}{1 - z + zP_J}. \tag{4}$$

The above representation of the expectation of X is not suitable for numerical evaluation because of a combinatorial explosion. For example, the number of summing-up operations required for computing (3) exponentially increases with n because

$$\sum_{i=1}^{n} \sum_{J;|J|=i} 1 = \sum_{i=1}^{n} \binom{n}{i} = 2^n - 1.$$

To alleviate the problem, in [3] the following compact integral representation of the expectation of X was derived

$$E[X] = \int_0^{\infty} \left(1 - \prod_{i=1}^{n} (1 - \mathrm{e}^{-p_i t})\right) \mathrm{d}t.$$

Similarly, we can have the following compact integral representation of the generating function of X.

$$g(z) = 1 + (z - 1) \int_0^{\infty} \mathrm{e}^{-(1-z)t} \left(1 - \prod_{i=1}^{n} (1 - \mathrm{e}^{-zp_i t})\right) \mathrm{d}t.$$

3. Upper and lower bound of coupon collector problem

In some engineering applications, it is important to compute the ccdf of X. The exact formula of the ccdf of X (formula (1)) is computationally expensive as explained in Section 2. To alleviate this difficulty, we focus on deriving some bounds of the ccdf of X, which are easy to compute even if the number of object types is quite large.

3.1. Lower bound

Since the stochastic property of X strongly depends on sampling probability $\boldsymbol{p} = (p_1, p_2, \ldots, p_n)$, in what follows, we use notation $X(\boldsymbol{p})$ to remind us of this dependence. We define

$$\boldsymbol{p}_{\text{even}}(\boldsymbol{p}) \stackrel{\text{def}}{=} (p_{\text{ave}}, \ldots, p_{\text{ave}}), \quad p_{\text{ave}} \stackrel{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} p_i.$$

For later use, we here introduce the following definitions:

Definition 2. Vector $\boldsymbol{p} = (p_1, \ldots, p_n)$ is said to majorize vector $\boldsymbol{q} = (q_1, \ldots, q_n)$ if

$$\sum_{i=1}^{k} q_{\beta(i)} \leqslant \sum_{i=1}^{k} p_{\gamma(i)}, \quad k = 1, \ldots, n-1,$$

$$\sum_{i=1}^{n} q_{\beta(i)} = \sum_{i=1}^{n} p_{\gamma(i)},$$

where γ and β are permutations of the indices $1, \ldots, n$ that reorder $\boldsymbol{p}$ and $\boldsymbol{q}$ in ascending order, that is

$$p_{\gamma(1)} \leqslant p_{\gamma(2)} \leqslant \cdots \leqslant p_{\gamma(n)}, \quad q_{\beta(1)} \leqslant q_{\beta(2)} \leqslant \cdots \leqslant q_{\beta(n)}.$$

We write $\boldsymbol{q} \prec \boldsymbol{p}$ when $\boldsymbol{p}$ majorize $\boldsymbol{q}$.

Definition 3. A real valued function f defined on $\mathbb{R}^n$ is said to be Schur-convex (concave) if

$$\boldsymbol{q} \prec \boldsymbol{p} \Rightarrow f(\boldsymbol{q}) \leqslant (\geqslant) f(\boldsymbol{p}).$$

Definition 4. Let X_1 and X_2 be random variables in $\mathbb{R}$. Then X_1 is said to be stochastically larger than X_2 if

$$P[X_2 > x] \leqslant P[X_1 > x] \quad \text{for all } x$$

We write $X_2 \leqslant_{\text{st}} X_1$ when X_1 is stochastically larger than X_2.

Although the next lemma is suggested in [4] without proof, we give its proof in appendix for completeness of the paper.

Lemma 5. *The ccdf of X is a Schur-concave function of the sampling probability.*

Proof. Please see Appendix A.

The following result readily follows from Lemma 5.

Lemma 6. *For all sampling probabilities $\boldsymbol{p}$*

$$X(\boldsymbol{p}_{\text{even}}(\boldsymbol{p})) \leqslant_{\text{st}} X(\boldsymbol{p}).$$

Proof. Since $\boldsymbol{p} \prec \boldsymbol{p}_{\text{even}}(\boldsymbol{p})$, it follows from Lemma 5 that

$$P[X(\boldsymbol{p}_{\text{even}}(\boldsymbol{p})) > x] \leqslant P[X(\boldsymbol{p}) > x] \quad \text{for all } x,$$

which completes the proof.

Lemma 6 with (2) gives a lower bound of the ccdf of X such as

$$P[X > k] \geqslant P[X(\boldsymbol{p}_{\text{even}}(\boldsymbol{p})) > k] = \sum_{i=1}^{n} (-1)^{i+1} \binom{n}{i} (1 - \mathrm{i}p_{\text{ave}})^k. \tag{5}$$

The number of summing-up operations required for computing (5) is the second order of n, and thus it can be easily computed even when n is quite large.

We can have another lower bound of $X(\boldsymbol{p})$ by a simple probabilistic argument.

Lemma 7. *The following inequality holds*:

$$P[X(\boldsymbol{p}) > k] \geqslant \sum_{j=1}^{2} (-1)^{j+1} \sum_{J:|J|=j} (1 - P_J)^k$$

$$= \sum_{j=1}^{n} (1 - p_j)^k - \sum_{i=1}^{n} \sum_{j=i+1}^{n} (1 - p_i - p_j)^k \tag{6}$$

Proof. See Appendix B.

We numerically show the tightness of lower bounds (5) and (6) in Section 4.

Remark 8. Inequality (6) has a simple physical interpretation. The first term of the right-hand side (RHS) of (6) is the contribution by events that all object types except one have been detected until when k objects are sampled. The second term of the RHS of (6) is the correction by events that all object types except two have been detected. Inequality (6) can be further improved by taking account of the contribution by events that more than two object types have not been detected until when k objects are sampled, but the contribution of these events is expected to be negligible when k is quite large. In fact, we have confirmed through numerical examples that (6) gives very tight bounds in the tail of the cumulative distribution function of X. We note that (6) can be generalized to the following inequality (please see Appendix B):

$$P[X(\boldsymbol{p}) > k] \geqslant \sum_{j=1}^{2l} (-1)^{j+1} \sum_{J:|J|=j} (1 - P_J)^k \quad \text{for } l = 1, 2, \ldots . \tag{7}$$

3.2. Upper bound

Like lower bounds, we can have two different upper bounds of X. The first one is derived by using stochastic comparison. To show this, we first define

$$\boldsymbol{p}_{\min} \stackrel{\text{def}}{=} (p_{\min}, \ldots, p_{\min}).$$

Definition 9. A real valued function f defined on $\mathbb{R}^n$ is said to be increasing (decreasing) if

$$\boldsymbol{q} \leqslant \boldsymbol{p} \Rightarrow f(\boldsymbol{q}) \leqslant (\geqslant) f(\boldsymbol{p}),$$

where $\boldsymbol{q} \leqslant \boldsymbol{p}$ means $\boldsymbol{p}$ is larger than $\boldsymbol{q}$ in a coordinate-wise sense.

We have the following intuitive result.

Lemma 10. *The ccdf of X is a decreasing function of the sampling probability*.

Proof. Please see Appendix C.

The following result readily follows from Lemma 10.

Lemma 11. $X(\boldsymbol{p}) \leqslant_{\text{st}} X(\boldsymbol{p}_{\min})$.

Proof. Since $\boldsymbol{p}_{\min} \leqslant \boldsymbol{p}$ coordinatewise, it follows from Lemma 10 that

$$P[X(\boldsymbol{p}) > x] \leqslant P[X(\boldsymbol{p}_{\min}) > x] \quad \text{for all } x,$$

which complete the proof.

Lemma 11 with (2) gives the following upper bound of the ccdf of X.

$$P[X > k] \leqslant P[X(\boldsymbol{p}_{\min}) > k] = \sum_{i=1}^{n} (-1)^{i+1} \binom{n}{i} (1 - \mathrm{i}p_{\min})^k. \tag{8}$$

We also have another upper bound by a simple probabilistic argument.

Lemma 12. *The following inequality holds*:

$$P[X(\boldsymbol{p}) > k] \leqslant \sum_{J:|J|=1} (1 - P_J)^k = \sum_{i=1}^{n} (1 - p_i)^k. \tag{9}$$

Proof. Please see Appendix D.

We also numerically show the tightness of lower bounds (8) and (9) in Section 4.

Remark 13. The difference between the lower bound by (6) and the upper bound by (9) asymptotically becomes negligible compared with $P[X > k]$ as $k \to \infty$. To see this, observe that

$$\frac{\text{RHS of (9)} - \text{RHS of (6)}}{P[X > k]} \leqslant \frac{\sum_{i=1}^{n} \sum_{j=i+1}^{n} (1 - p_i - p_j)^k}{\sum_{i=1}^{n} (1 - p_i)^k - \sum_{i=1}^{n} \sum_{j=i+1}^{n} (1 - p_i - p_j)^k}$$

$$\sim a \left(\frac{1 - p_{\min} - p_{\min}^{(2)}}{1 - p_{\min}} \right)^k \quad \text{as } k \to \infty,$$

where a is some constant and $p_{\min}^{(2)}$ is the second smallest probability among $p_1, \ldots, p_n$. This fact implies that the pair of (6) and (9) is expected to always give very close bounds in the tail of the ccdf of X.

Remark 14. Inequality (9) can also be generalized to the following (please see Appendix B):

$$P[X(\boldsymbol{p}) > k] \leqslant \sum_{j=1}^{2l-1} (-1)^{j+1} \sum_{J:|J|=j} (1 - P_J)^k \quad \text{for } l = 1, 2, \ldots. \tag{10}$$

Remark 15. By combining some upper and lower bounds of the ccdf of X, we can easily obtain its asymptotic decay rate (see Appendix E)

$$\lim_{k \to \infty} \frac{1}{k} \log P[X > k] = \log(1 - p_{\min}), \tag{11}$$

which means that X is a light-tailed random variable.

4. Application: efficiency evaluation of PPM schemes

In this section, we explain how the ccdf of the detecting cost is related to the attacking-path-detection efficiency of PPM schemes, and show the results of some numerical experiments to conform the tightness of the bounds.

4.1. Outline of the PPM scheme

Here, we consider a single-source DoS attack where there are n routers between the victim and the attacker (the source of attack packets). We refer to the link between routers at distance $i-1$ and i from the victim as *link i*. To simplify the explanation, we assume that routers can mark packets with the *full information* of the link during packet forwarding; that is, each packet is assumed to have two 32-bit fields in its IP header to store IP addresses of both end routers of the link.

When a router decides to mark a packet, it writes its own IP address into one of the 32-bit fields, which we call *source field*. Then, the following router finds that the *source filed* has been marked but another field, which we call *sink field*, is not marked. If the following router decides not to mark the packet, then it writes its IP address into the *sink field*. Otherwise, the following router overwrites its own IP address into the *source field*. The probability that a router decides to mark a packet is p. The victim could know which link a packet traversed based on two IP addresses stored in *source and sink* fields of the packet. We say that *link i is detected* when the victim firstly receives a packet marked with the information on link i. We also say that *the entire path is detected* when all links in the entire path (besides the link between the furthest router and the attacker) have been detected.

Let p_i be the probability that the information about link i is marked in a packet received by the victim. Then, in the above mentioned scheme, p_i should be equal to $p(1-p)^{i-1}$. In this section, we refer to a vector $\boldsymbol{p} \stackrel{\text{def}}{=} (p_1, p_2, \ldots, p_n)$ as *marking probability vector*. Note that the marking probability corresponds to the sampling probability in previous sections. For later use, we define the following:

$$\boldsymbol{p}(k:l) \stackrel{\text{def}}{=} \left(\underbrace{p_1(l), \ldots, p_1(l)}_{k}, \underbrace{p_2(l), \ldots, p_2(l)}_{k}, \ldots, \underbrace{p_n(l), \ldots, p_n(l)}_{k} \right),$$

$$p_i(l) \stackrel{\text{def}}{=} p(1-p)^{i-1}/l, \quad i = 1, \ldots, n.$$

Remark 16. The above mentioned scheme needs to use the option field of IP header to store two IP addresses of both end routers of the link. However, such an implementation is not practical because appending additional data to a packet in flight is expensive [8]. We here focus on such a scheme simply because it is easy to understand how our proposal will be applied, and our proposal explained in Section 3 can also be applied to other (more practical) PPM schemes.

4.2. False negative ratio of a single-source attack

Now, define the false negative ratio $P_{\text{fn}}(N)$ by the probability that the attack path is not detected by the time when the victim receives N attack packets. It is obvious that the number of packets required for reconstructing the entire attack path is equivalent to the detecting cost of the coupon collector problem when the marking probability vector is $\boldsymbol{p}(1;1)$, and thus we obtain

$$P_{\text{fn}}(N) = P[X(\boldsymbol{p}(1;1)) > N]. \tag{12}$$

4.3. False negative ratio of a multiple-source attack

A multiple-source attack can be evaluated based on the results of a single-source attack. To see this, consider a DoS attack from L different attack paths. We assume that the numbers of routers on respective attack paths are all n, and the sets of routers on separate attack paths are mutually disjoint.

Now, let $L_{\text{detect}}(N)$ denote the expectation of the number of attack paths that have been detected by the time when the victim receives N attack packets, and define the false negative ratio by the following:

$$P_{\text{fn}}(N) \stackrel{\text{def}}{=} 1 - \frac{L_{\text{detect}}(N)}{L}.$$

Table 1
The ccdf of X when $\boldsymbol{p} = \boldsymbol{p}(1; 1)$

k	$P[X > k]$				
	Exact	Lower bound (5)	Upper bound (8)	Lower bound (6)	Upper bound (9)
50	9.991495×10^{-1}	9.985956×10^{-1}	9.999926×10^{-1}	-6.849914	5.124129
100	7.976888×10^{-1}	7.222236×10^{-1}	9.736670×10^{-1}	5.300224×10^{-1}	1.445076
200	1.331392×10^{-1}	6.777143×10^{-2}	3.934783×10^{-1}	1.328993×10^{-1}	1.409456×10^{-1}
300	1.607553×10^{-2}	4.108340×10^{-3}	7.330718×10^{-2}	1.607524×10^{-2}	1.617317×10^{-2}
400	2.024288×10^{-3}	2.428807×10^{-4}	1.174271×10^{-2}	2.024288×10^{-3}	2.025706×10^{-3}
500	2.672380×10^{-4}	1.433926×10^{-5}	1.837711×10^{-3}	2.672380×10^{-4}	2.672608×10^{-4}
600	3.647972×10^{-5}	8.465028×10^{-7}	2.865897×10^{-4}	3.647972×10^{-5}	3.648011×10^{-5}
700	5.099771×10^{-6}	4.997217×10^{-8}	4.466984×10^{-5}	5.099771×10^{-6}	5.099778×10^{-6}
800	7.256417×10^{-7}	2.950041×10^{-9}	6.961993×10^{-6}	7.256417×10^{-7}	7.256418×10^{-7}
900	1.046625×10^{-7}	1.741517×10^{-10}	1.085045×10^{-6}	1.046625×10^{-7}	1.046625×10^{-7}
1000	1.525891×10^{-8}	1.028081×10^{-11}	1.691067×10^{-7}	1.525891×10^{-8}	1.525891×10^{-8}

Now let $X^{(i)}$ denote the number of packets required for the ith attack path detection. Since $X^{(1)}, \ldots, X^{(L)}$ are identically distributed, it follows that

$$L_{\text{detect}}(N) = \sum_{i=1}^{L} E[I(X^{(i)} \leqslant N)] = LE[I(X^{(1)} \leqslant N)] = LP[X^{(1)} \leqslant N],$$

where $I(\cdot)$ is the indicator function defined by

$$I(A) \stackrel{\text{def}}{=} \begin{cases} 1 & A \text{ is true,} \\ 0 & A \text{ is false.} \end{cases}$$

Since the marking probability vector of the first attack path is equal to $\boldsymbol{p}(1; L)$, we finally have

$$P_{\text{fn}}(N) = P[X^{(1)} > N] = P[X(\boldsymbol{p}(1; L)) > N]. \tag{13}$$

Note that letting $L = 1$ in (13) yields (12).

Remark 17. In general PPM schemes, routers mark packets with one of fragments of the link information (IP addresses of both end routers of the link) because the link information should be marked into a limited-space field which is not sufficient for storing two IP addresses. If the link information is divided into K fragments in a PPM scheme, then the false negative ratio of this scheme is given by

$$P_{\text{fn}}(N) = P[X(\boldsymbol{p}(K; L)) > N].$$

4.4. Numerical results

We have conducted several numerical experiments to examine the tightness of the proposed bounds. We first numerically calculates the ccdf of X when the marking probability vector is given by

$$\boldsymbol{p} = \boldsymbol{p}(1; 1) \stackrel{\text{def}}{=} (p, p(1-p), \ldots, p(1-p)^{n-1}),$$

when $n = 20$ and $p = 0.04$. These parameter values are typical to the PPM schemes [8]. The results are listed in Table 1.

As the table indicates that the pair of (8) and (5) yields tight bounds for $P[X > k]$ when k is small, while the pair of (9) and (6) yields tight bounds when k is large. In particular, the pair of (9) and (6) are very close to the actual ccdf when the ccdf is less than 0.01.

Table 2
The ccdf of X when $\boldsymbol{p} = \boldsymbol{p}(1; 100)$

k	$P[X > k]$				
	Exact	Lower bound (5)	Upper bound (8)	Lower bound (6)	Upper bound (9)
50	9.979163×10^{-1}	9.966632×10^{-1}	9.999617×10^{-1}	-7.638817	5.216266
100	7.923407×10^{-1}	7.185807×10^{-1}	9.683892×10^{-1}	4.543995×10^{-1}	1.490598
200	1.387561×10^{-1}	7.275840×10^{-2}	3.989690×10^{-1}	1.383740×10^{-1}	1.482523×10^{-1}
300	1.717047×10^{-2}	4.618931×10^{-3}	7.673282×10^{-2}	1.716992×10^{-2}	1.729888×10^{-2}
400	2.200421×10^{-3}	2.841777×10^{-4}	1.255495×10^{-2}	2.200420×10^{-3}	2.202421×10^{-3}
500	2.953292×10^{-4}	1.745016×10^{-5}	2.000318×10^{-3}	2.953292×10^{-4}	2.953636×10^{-4}
600	4.098103×10^{-5}	1.071414×10^{-6}	3.173527×10^{-4}	4.098103×10^{-5}	4.098167×10^{-5}
700	5.823821×10^{-6}	6.578280×10^{-8}	5.031454×10^{-5}	5.823821×10^{-6}	5.823833×10^{-6}
800	8.423925×10^{-7}	4.038936×10^{-9}	7.976245×10^{-6}	8.423925×10^{-7}	8.423928×10^{-7}
900	1.235172×10^{-7}	2.479828×10^{-10}	1.264434×10^{-6}	1.235172×10^{-7}	1.235172×10^{-7}
1000	1.830677×10^{-8}	1.522566×10^{-11}	2.004438×10^{-7}	1.830677×10^{-8}	1.830677×10^{-8}

Table 3
The ccdf of X when $\boldsymbol{p} = \boldsymbol{p}(8; 1)$

k	$P[X > k]$			
	Lower bound (5)	Upper bound (8)	Lower bound (6)	Upper bound (9)
500	1.000000	1.000000	-420.1996	30.19812
1000	9.938940×10^{-1}	1.000000	-10.440447	6.529360
2000	1.376713×10^{-1}	8.010895×10^{-1}	3.203004×10^{-1}	3.967066×10^{-1}
3000	4.481859×10^{-3}	1.472110×10^{-1}	2.865208×10^{-2}	2.905747×10^{-2}
4000	1.364965×10^{-4}	1.574028×10^{-2}	2.333307×10^{-3}	2.335898×10^{-3}
5000	4.148643×10^{-6}	1.581563×10^{-3}	1.981216×10^{-4}	1.981400×10^{-4}
6000	1.260852×10^{-7}	1.579232×10^{-4}	1.740106×10^{-5}	1.740121×10^{-5}
7000	3.831966×10^{-9}	1.575925×10^{-5}	1.565741×10^{-6}	1.565742×10^{-6}
8000	1.164606×10^{-10}	1.572528×10^{-6}	1.434201×10^{-7}	1.434201×10^{-7}
9000	3.539455×10^{-12}	1.569128×10^{-7}	1.331738×10^{-8}	1.331738×10^{-8}
10000	1.075706×10^{-13}	1.565735×10^{-8}	1.249916×10^{-9}	1.249916×10^{-9}

Table 2 shows the results when the marking probability vector is given by

$$\boldsymbol{p} = \boldsymbol{p}(1; 100) \stackrel{\text{def}}{=} (p/100, p(1-p)/100, \ldots, p(1-p)^{n-1}/100),$$

when $n = 20$ and $p = 0.04$. Note that the ccdf of X when the marking probability vector is $\boldsymbol{p}(1; 100)$ gives the false negative ratio of the PPM scheme when there are one hundred attackers. The results are similar with those of Table 1.

Finally, we show the results in Table 3 when the marking probability vector is given by

$$\boldsymbol{p} = \boldsymbol{p}(8; 1) \stackrel{\text{def}}{=} \left(\underbrace{p, \ldots, p}_{8}, \underbrace{p(1-p), \ldots, p(1-p)}_{8}, \ldots, \underbrace{p(1-p)^{n-1}, \ldots, p(1-p)^{n-1}}_{8} \right),$$

when $n = 20$ and $p = 0.04$. The ccdf of X with marking probability vector $\boldsymbol{p}(8; 1)$ gives the false negative ratio of the PPM scheme when the link information is divided into eight fragments and a router marks a packet with one of the fragments when the packet traverse there (see Remark 17). In this case, we cannot compute the ccdf of X exactly because of the huge computational time and thus, in the table, we show only the upper and lower bounds. The table indicates that, except for the case when k is small, the difference between upper bound (9) and lower bound (6) is very small, which enables us to accurately evaluate the ccdf based on (9) and (6) without computing the exact formula of the ccdf.

Table 4
The ccdf of X when the sampling probability has the power-law-type distribution

k	$P[X > k]$				
	Exact	Lower bound (5)	Upper bound (8)	Lower bound (6)	Upper bound (9)
50	9.654626×10^{-1}	1.454898×10^{-10}	9.999958×10^{-1}	-5.892639	2.754208
100	6.122455×10^{-1}	1.058365×10^{-21}	9.910044×10^{-1}	5.549317×10^{-1}	8.766862×10^{-1}
200	1.165242×10^{-1}	5.600678×10^{-44}	5.899394×10^{-1}	1.163998×10^{-1}	1.220404×10^{-1}
300	1.984819×10^{-2}	2.963779×10^{-66}	1.667368×10^{-1}	1.984780×10^{-2}	1.998422×10^{-2}
400	3.526034×10^{-3}	1.568379×10^{-88}	3.718076×10^{-2}	3.526033×10^{-3}	3.529853×10^{-3}
500	6.515372×10^{-4}	$8.299586 \times 10^{-111}$	7.864564×10^{-3}	6.515372×10^{-4}	6.516535×10^{-4}
600	1.238222×10^{-4}	$4.391994 \times 10^{-133}$	1.645123×10^{-3}	1.238222×10^{-4}	1.238259×10^{-4}
700	2.400885×10^{-5}	$2.324165 \times 10^{-155}$	3.433316×10^{-4}	2.400885×10^{-5}	2.400898×10^{-5}
800	4.724491×10^{-6}	$1.229907 \times 10^{-177}$	7.161753×10^{-5}	4.724491×10^{-6}	4.724496×10^{-6}
900	9.401408×10^{-7}	$6.508450 \times 10^{-200}$	1.493762×10^{-5}	9.401408×10^{-7}	9.401410×10^{-7}
1000	1.887089×10^{-7}	$3.444157 \times 10^{-222}$	3.115546×10^{-6}	1.887089×10^{-7}	1.887089×10^{-7}

Remark 18. In the numerical examples explained above, we considered the case that the sampling probability (marking probability) $(p_1, \ldots, p_n)$ has a geometric-type distribution. As explained in Remarks 8 and 13, the pair of (6) and (9) are expected to give tight bounds even in more general cases. Table 4 shows the ccdf of X when the sampling probability has the following power-law-type distribution:

$$p_i = ci^{-2}, \quad c = 1 \Big/ \sum_{i=1}^{20} i^{-2}, \quad i = 1, \ldots, 20.$$

Table 4 confirms that the pair of (6) and (9) gives tight bounds even when the sampling probability has power-law-type distribution, while the pair of (5) and (8) (in particular, (5)) fails to give tight bounds.

5. Conclusion

In this paper, we derived upper and lower bounds of the ccdf of the detecting cost in the coupon collector problem. All of the derived bounds require much less computation compared with the exact formula. Through numerical experiments, we found that proposed bounds are very close to the actual ccdf when sampling probability is set to the value common to the PPM schemes for IP traceback problem. The derived bounds will be very helpful not only for evaluating the efficiency of various PPM schemes, but also for a wide variety of applications of the coupon collector problem.

Acknowledgements

We would like to thank Professor Naoto Miyoshi, Tokyo Institute of Technology, for useful discussions and suggestions. We also thank the anonymous reviewer for his comments and constructive advice.

Appendix A. Proof of Lemma 5

To prove the lemma, we need the following result.

Lemma 19 (*Marshall and Olkin [5]*). *A function f defined on $\mathbb{R}^n$ is Schur-convex* (*concave*) *iff f is symmetric and $f(\lambda q, (1-\lambda)q, p_3, \ldots, p_n)$ is a non-decreasing* (*non-increasing*) *function of λ for $\lambda \in (0, 1/2]$.*

To use this lemma, observe that

$$P[X(\boldsymbol{p}) > k] = \sum_{i=0}^{n-2}(-1)^i \sum_{J;|J|=I,1,2\notin J} (1 - p_1 - P_J)^k + \sum_{i=0}^{n-2}(-1)^i \sum_{J;|J|=I,1,2\notin J} (1 - p_2 - P_J)^k$$
$$+ \sum_{i=0}^{n-2}(-1)^{i+1} \sum_{J;|J|=I,1,2\notin J} (1 - p_1 - p_2 - P_J)^k + g(p_3, \ldots, p_n),$$

where $g(p_3, \ldots, p_n)$ is some function of $p_3, \ldots, p_n$. Now define

$$f(\lambda) \stackrel{\text{def}}{=} P[X(\lambda q, (1-\lambda)q, p_3, \ldots, p_n) > k].$$

Then

$$f(\lambda) = \sum_{i=0}^{n-2}(-1)^i \sum_{J;|J|=I,1,2\notin J} (1 - \lambda q - P_J)^k + \sum_{i=0}^{n-2}(-1)^i \sum_{J;|J|=I,1,2\notin J} (1 - (1-\lambda)q - P_J)^k$$
$$+ \sum_{i=0}^{n-2}(-1)^i \sum_{J;|J|=I,1,2\notin J} (1 - q - P_J)^k + g(p_3, \ldots, p_n),$$

from which

$$\frac{\mathrm{d}f}{\mathrm{d}\lambda} = -kq \sum_{i=0}^{n-2}(-1)^i \sum_{J;|J|=I,1,2\notin J} \{(1 - \lambda q - P_J)^{k-1} - (1 - (1-\lambda)q - P_J)^{k-1}\}. \tag{A.1}$$

Now let

$$h(a) \stackrel{\text{def}}{=} \sum_{i=0}^{n}(-1)^i \sum_{J;|J|=i} (1 - a - P_J)^{k-1}.$$

As shown in Appendix C, $h(a)$ is a decreasing function of a. Combining this fact with (A.1) gives $\mathrm{d}f/\mathrm{d}\lambda \leqslant 0$, which proves the assertion.

Appendix B. Proof of Lemma 7

Define

$$R_i^{(k)} = \bigcup_{J:|J|=i} S_J^{(k)} \Big/ \bigcup_{J:|J|=i+1} S_J^{(k)}.$$

Note that $R_i^{(k)}$ denote the event that i types of objects are not still detected by the time when k objects are sampled. Since $R_1^{(k)}, \ldots, R_n^{(k)}$ are mutually disjoint events, it follows that

$$P\left[\bigcup_{j=1}^{n} A_j^{(k)}\right] = \sum_{i=1}^{n} P[R_i^{(k)}].$$

By a similar argument used for proving the inclusion–exclusion principle [2],

$$\sum_{J:|J|=m} P[S_J^{(k)}] = \sum_{i=1}^{n} \binom{i}{m} P[R_i^{(k)}],$$

from which we obtain

$$\sum_{J:|J|=1} P[S_J^{(k)}] - \sum_{J:|J|=2} P[S_J^{(k)}] = \sum_{i=1}^{n} \left(\binom{i}{1} - \binom{i}{2} \right) P[R_i^{(k)}]$$
$$= \sum_{i=1}^{n} \left(1 - \binom{i-1}{2} \right) P[R_i^{(k)}]$$
$$\leqslant \sum_{i=1}^{n} P[R_i^{(k)}].$$

Thus, the assertion follows by the observation that

$$P[X > k] = P\left[\bigcup_{j=1}^{n} A_j^{(k)} \right] = \sum_{i=1}^{n} P[R_i^{(k)}]$$
$$\geqslant \sum_{J:|J|=1} P[S_J^{(k)}] - \sum_{J:|J|=2} P[S_J^{(k)}]$$
$$= \sum_{J:|J|=1} (1 - P_J)^k - \sum_{J:|J|=2} (1 - P_J)^k.$$

Note that, in general, the following relationship holds:

$$\sum_{m=1}^{l} (-1)^{m+1} \sum_{J:|J|=m} P[S_J^{(k)}] = \sum_{m=1}^{l} (-1)^{m+1} \sum_{i=1}^{n} \binom{i}{m} P[R_i^{(k)}]$$
$$= \sum_{i=1}^{n} \sum_{m=1}^{l} (-1)^{m+1} \binom{i}{m} P[R_i^{(k)}]$$
$$= \sum_{i=1}^{n} \left(1 + (-1)^{l+1} \binom{i-1}{l} \right) P[R_i^{(k)}],$$

where we use the fact

$$\sum_{m=1}^{l} (-1)^{m+1} \binom{i}{m} = \sum_{m=1}^{l} (-1)^{m+1} \left\{ \binom{i-1}{m-1} + \binom{i-1}{m} \right\}$$
$$= \left\{ 1 + \binom{i-1}{1} \right\} - \left\{ \binom{i-1}{1} + \binom{i-1}{2} \right\} + \cdots + (-1)^{l+1} \left\{ \binom{i-1}{l-1} + \binom{i-1}{l} \right\}$$
$$= 1 + (-1)^{l+1} \binom{i-1}{l}.$$

Thus,

$$\sum_{m=1}^{l} (-1)^{m+1} \sum_{J:|J|=m} P[S_J^{(k)}] \begin{cases} \geqslant \sum_{i=1}^{n} P[R_i^{(k)}] & l \text{ is odd}, \\ \leqslant \sum_{i=1}^{n} P[R_i^{(k)}] & l \text{ is even}. \end{cases}$$

This observation yields

$$P[X > k] = \sum_{i=1}^{n} P[R_i^{(k)}] \begin{cases} \leqslant \sum_{m=1}^{l} (-1)^{m+1} \sum_{J:|J|=m} P[S_J^{(k)}] & l \text{ is odd}, \\ \geqslant \sum_{m=1}^{l} (-1)^{m+1} \sum_{J:|J|=m} P[S_J^{(k)}] & l \text{ is even}, \end{cases}$$

which gives the proof of (5) and (10).

Appendix C. Proof of Lemma 10

First note that there is no coordinate-wise ordering relationship between sampling probability vectors, the sums of whose elements are respectively equal to 1. So, we can focus on the case where $\sum_{i=1}^{n} p_i < 1$. Since the ccdf of $X(\boldsymbol{p})$ is symmetric, to complete the proof it is sufficient to prove that the ccdf of $X(\boldsymbol{p})$ is a decreasing function of p_1. To this end, let $f(p_1)$ denote $P[X(\boldsymbol{p}) > k]$. Observe that

$$f(p_1) = \sum_{i=0}^{n-1} (-1)^i \sum_{J;|J|=I,1\notin J} (1 - p_1 - P_J)^k + c, \tag{C.1}$$

where c is some constant. In this case, using representation (C.1), we obtain

$$\begin{aligned}
\frac{\mathrm{d}f}{\mathrm{d}p_1} &= -k \sum_{i=0}^{n-1} (-1)^i \sum_{J;|J|=I,1\notin J} (1 - p_1 - P_J)^{k-1} \\
&= -k \left\{ (1-p_1)^{k-1} - \sum_{i=1}^{n-1} (-1)^{i-1} \sum_{J;|J|=I,1\notin J} (1 - p_1 - P_J)^{k-1} \right\} \\
&= -k \left\{ P[S_1^{(k-1)}] - \sum_{i=1}^{n} (-1)^{i-1} \sum_{J;|J|=i,1\notin J} P[S_J^{(k-1)} \cap S_1^{(k-1)}] \right\} \\
&= -k \left\{ P[S_1^{(k-1)}] - P\left[S_1^{(k-1)} \cap \bigcup_{i=2}^{n} A_j^{(k-1)} \right] \right\} \leqslant 0,
\end{aligned}$$

which completes the proof.

Appendix D. Proof of Lemma 12

Observe that

$$P[X > k] = P\left[\bigcup_{j=1}^{n} A_j^{(k)} \right] \leqslant \sum_{j=1}^{n} P[A_j^{(k)}] = \sum_{j=1}^{n} (1 - p_j)^k,$$

which completes the proof.

Appendix E. Derivation of (11)

We first assume that $p_1 = p_{\min}$ and that $p_1 < p_j$ for all $j = 2, \ldots, N$. It follows from (9) that

$$\begin{aligned}
\limsup_{k\to\infty} \frac{1}{k} \log P[X > k] &\leqslant \limsup_{k\to\infty} \frac{1}{k} \log \left\{ \sum_{i=1}^{n} (1 - p_i)^k \right\} \\
&= \limsup_{k\to\infty} \frac{1}{k} \log \left\{ (1-p_1)^k \sum_{i=1}^{n} \left(\frac{1-p_i}{1-p_1} \right)^k \right\} \\
&= \log(1-p_1) + \limsup_{k\to\infty} \frac{1}{k} \log \left\{ \sum_{i=1}^{n} \left(\frac{1-p_i}{1-p_1} \right)^k \right\} \\
&= \log(1-p_1).
\end{aligned} \tag{E.1}$$

Next observe that

$$P[X > k] = P\left[\bigcup_{j=1}^{n} A_j^{(k)}\right] \geqslant P[A_1^{(k)}] = (1 - p_1)^k,$$

Thus, we obtain

$$\begin{aligned}\liminf_{k\to\infty} \frac{1}{k} \log P[X > k] &\geqslant \liminf_{k\to\infty} \frac{1}{k} \log(1 - p_1)^k \\ &= \log(1 - p_1). \end{aligned} \tag{E.2}$$

Combining (E.1) and (E.2) proves the assertion. The similar arguments give the proof for the case that some $j \in (2, \ldots, N)$ satisfies $p_1 = p_j$.

References

[1] S. Boneh, V. Papanicolaou, General asymptotics estimates for the coupon collector problem, J. Comput. Appl. Math. 67 (1996) 277–289.
[2] W. Feller, An Introduction to Probability Theory and Its Applications, second ed., vol. 1, Wiley, New York, 1966.
[3] P. Flajolet, D. Gardy, L. Thimonier, Birthday paradox, coupon collectors, caching algorithms and self-organizing search, Discrete Appl. Math. 39 (1992) 207–229.
[4] L. Holst, Extreme value distributions for random coupon collector and birthday problems, Extremes 4 (2) (2001) 129–145.
[5] A. Marshall, I. Olkin, Inequalities: Theore of Majorization and its Applications, Academic Press, New York, 1979.
[6] S. Martinez, Some bounds on the coupon collector problem, Random Struct. Algorithms 25 (2004) 208–226.
[7] V. Papanicolaou, G. Kokolakis, S. Boneh, Asymptotics for the random coupon collector problem, J. Comput. Appl. Math. 93 (1998) 95–105.
[8] A. Savage, D. Wetherall, A. Karlin, T. Anderson, Network support for IP traceback, IEEE/ACM Trans. Networking 9 (3) (2001) 226–237.