GENERALIZED GROUP TESTING FOR RETRIEVING DISTRIBUTED INFORMATION

Yao-Win Hong and Anna Scaglione

School of Electrical and Computer Engineering, Cornell University
E-mail: yh84@cornell.edu; anna@ece.cornell.edu

ABSTRACT

The goal of group testing is to efficiently classify the state of a set of distributed agents through a sequence of tests by imposing each test simultaneously upon groups of agents. In this work, we describe the concept of group testing in a generalized framework and propose to apply this concept to solve the scheduling and multiple access problem in a large scale wireless sensor network. Since the standard approach is to dedicate a single channel to each sensor, we discuss the efficiency of group testing by comparing it to the case where each sensor is tested individually. Through the sequence of tests, the group testing strategy successively refines the observation space of the set of sensors and eventually identifies the status of each sensor when the space is refined to only one element. We show that the successive refinement property of group testing (similar to that of arithmetic coding) plays an important role in its performance. Based on this concept, we provide insight into choosing optimal group testing strategies for general applications.

1. PREMISE

Since [1] reported the negative scaling laws in large scale wireless networks, several papers have tried to consider assumptions and models that overcome the vanishing throughput of multi-hop transmission in wireless networks. The problem has been approached from two main directions. Some have studied models for the aggregate data rate that scale favorably and have proposed either distributed compression techniques [2, 3] or combined routing and compression methods [4]. Some have contended that this problem has no solution, in spite of the fact that the vanishing aggregate data rate [5]. Others have considered ways of ideally cooperating among nodes [6] using space-time codes across multiple network nodes acting cooperatively as a MIMO system that would provide greater capacity than the sum of the individual point-to-point links could provide.

Even if compression or cooperative transmission can, in principle, provide scalable solutions, the complexity of the algorithms and their optimal design justifies some reasonable skepticism around the fact that large sensor networks of unmanned agents are going to be designed following the ideas contained in these papers.

This work supported in part by the National Science Foundation under grant CCR-0227676.

With this premise, it is clear that transmission and compression have to be viewed in completely different terms to derive designs that scale in performance as well as in complexity. Due to the fact that the receivers used for group testing can be implemented with simple energy detectors and because of the simple feedback that it requires, we think that the paradigm of group testing provides a compelling example of a scalable family of solutions for sensor networks.

2. CLASSICAL GROUP TESTING

Group testing (GT) was first proposed by Dorfman [7] during World War II to efficiently identify syphilitic men that were called upon to serve in the US army. Since the event of having this disease is relatively rare, Dorfman realized that it was extremely inefficient to test the blood samples of each individual separately and propose to reduce the average number of tests necessary by pooling a number of blood samples together into one test. Since then, the concept of group testing has arise in many industrial applications [8] such as testing the leakage of devices or identifying the defective light bulbs by arranging a group of them in series.

Group testing has also been studied in the context of random access scheduling [9]. In contrast to TDMA where each node in the network is assigned a unique transmission channel, group testing allocates the same time slot for the transmission of multiple nodes. If more then one node within the group has a packet to transmit, a collision will occur and a subgroup of these nodes will then be chosen to transmit in the future time slots. If there is at most one node transmitting within a time slot, the group of nodes that are allocated to this time slot is then completely resolved.

Interestingly, the effect of group testing is equivalent to classifying the network of agents into classes that correspond to their respective state. For example, in the blood testing case, group testing classifies the blood samples into those that are contaminated with the disease and those that are not; in the random access example, nodes in the network are classified into those that have a packet to transmit and those that are idle. From this point of view, group testing can be used, in general, to classify the status of agents in a large population. By accurately partitioning the agents into classes, the central agent equivalently obtains complete knowledge of the information contained in the agents, *i.e.* it

effectively retrieves the data from the distributed agents.

Although group testing was proven to be advantageous in the case of detecting a rare event under the *i.i.d.* Bernoulli model, we have shown, recently, that these methods may also be used to efficiently classify correlated information [10] or to retrieve data from a quantized and correlated sensor field [11]. *The major contribution of this work is to provide a generalized formulation of the group testing problem.* This generalization is particularly important and useful to determine optimized strategies to retrieve the information from a set of distributed agents and derive a completely novel class of multiple access techniques that are combined with compression. Furthermore, we show that the essence of group testing lies in the fact that it successively refines the set of possible events through the sequence of tests which shows similarities with that of arithmetic coding (c.f. Section 4). This concept justifies the strategies chosen in classical group testing problems and provides insight into choosing group testing strategies for general applications.

3. GENERALIZED GROUP TESTING

Consider a set of N agents $\mathcal{S} = \{s_1, \cdots, s_N\}$ and let X_i be the state of agent s_i. The set of states $\mathbf{X} = [X_1, \cdots, X_N]$ are modelled as a sequence of random variables with the joint probability distribution $p_{X_1,\cdots,X_N}(x_1, \cdots, x_N)$. In general, the state value of the agent i belongs to the symbol alphabet $\mathcal{A}_i$, where $\mathcal{A}_i$ may not be binary, and the distribution of the states may be correlated. In classical group testing, *e.g.* the blood testing problem, each item in the population can be either "defective" or "non-defective". Therefore, it is common to model the set of states $\{X_1, \cdots, X_N\}$ as a sequence of *i.i.d.* Bernoulli random variables with parameter $p = \Pr\{X_i = 1\} = \Pr\{s_i \text{ is defective}\}$ and

$$p_{X_1,\cdots,X_N}(x_1, \cdots, x_N) = p^{\sum_i x_i}(1-p)^{(N-\sum_i x_i)}.$$

In order to distinguish between the defective and non-defective items, classical strategies typically choose to impose a question $T \equiv \{$"Are you defective?"$\}$ upon all agents within a group U. If at least one item in the group is defective, the outcome of the test Z is positive (*i.e.* $Z = 1$) regardless of the number of agents that are defective within the group, while the outcome of the test is negative (*i.e.* $Z = 0$) if and only if all agents are non-defective. When a positive feedback is observed, a subgroup of agents that belong to the previous group must be assigned again to a future (T', U') test in order to identify specifically which agent or agents are actually defective. The goal of group testing is to minimize the expected number of tests L necessary to completely resolve the sequence of states.

Definition 1 *A group testing strategy is defined by the group testing tree $(\mathcal{T}, \mathcal{U}, \mathcal{F})$, where $\mathcal{T}$ is the set of questions asked on each corresponding group determined in $\mathcal{U}$, and $\mathcal{F}$ is the set of possible outcomes (or feedback) of the tests.*

To resolve a particular sequence $\mathbf{X}$ through group testing, a central agent must impose a sequence of tests $T_0, T_1, \cdots, T_{L-1}$ to the groups $U_0, U_1, \cdots, U_{L-1}$ where the sequence of (T, U)-pairs represents a path along the group testing tree $(\mathcal{T}, \mathcal{U}, \mathcal{F})$ and L is the length of the path, *i.e.* the random variable representing the number of tests necessary to reconstruct $\mathbf{X}$. If $\mathcal{F}$ is the set of possible feedbacks, then the node representing the test (T, U) will have $|\mathcal{F}|$ branches extending from itself which leads, respectively, to another test (T', U') or terminates at a completely resolved sequence $\mathbf{x}$. Each different path corresponds to a different realization of the agents' information $\mathbf{X}$. However, if there exists a path in the tree that terminates at more than one sequence, the proposed strategy would not be able to distinguish between these sequences. In order to uniquely resolve the contents of each agent, the sequence of outcomes resulting from the sequence of tests must unambiguously determine the state of all agents. Therefore, we define a class of unambiguous group testing strategies as follows:

Definition 2 *A group testing strategy $(\mathcal{T}, \mathcal{U}, \mathcal{F})$ is considered as unambiguous if it uniquely resolves the sequence of states $\mathbf{X} = [X_0, \cdots, X_{N-1}]$.*

In general, the question T_l that is imposed upon the group $U_l = \{s_{i_1}, \cdots, s_{i_{|U_l|}}\}$ can be seen as asking the question "Is $[X_{i_1}, \cdots, X_{i_{|U_l|}}] = [a_{i_1}, \cdots, a_{i_{|U_l|}}]$?", where $a_i \in \mathcal{A}_i$. By imposing the test (T_l, U_l), the outcome Z_l must belong to the set of possible feedbacks F_l which is equivalent to the output alphabet of the channel between the central agent and the distributed agents within the group U_l. For example, in the classical problem, the question imposed upon a group of size n is typically equal to "Is $[X_{i_1}, \cdots, X_{i_n}] = [0, \cdots, 0]$?". In this case, the response Z_l is equal to 1 if there exists $j \in U_l$ such that $X_j = 1$, and $Z_l = 0$ otherwise. Therefore, the channel between the distributed agent and the central node is modelled as the "noiseless OR channel", where the answer beared in the received signal after each test is

$$Z_l = \vee_{\{j:s_j \in U_l\}}\{X_j \neq a_j\} = \vee_{\{j:s_j \in U_l\}}\{X_j \neq 0\}. \quad (1)$$

Therefore, the feedback signal contains the binary information: **r.1)** all nodes are of the bit 0, *i.e.* $Z_l = 0$; and **r.2)** there exists a node with bit 1, *i.e.* $Z_l = 1$. Although the group testing problem addressed in most classical group testing problems refer to this particular type of multiple access channel, one could certainly choose another kind of channel corresponding to their physical layer implementation and derive the optimal group testing strategy that would utilize most efficiently the feedback obtained from that channel. In fact, different variations of the group testing strategy was derived for different sets of feedback in [9].

Since each path in an unambiguous group testing strategy leads to a unique sequence of states, the sequence of outcomes $\mathbf{Z} = [Z_1, \cdots, Z_L]$ that routes through this path

uniquely encodes the information from the distributed agents. If the group testing strategy is performed optimally, the average number of tests should not exceed that of testing each agent individually, *i.e.* $\mathbf{E}\{L\} \leq N$. Therefore, this sequence of outcomes contains the lossless information of the agent's states, thus, allowing group testing to serve as a form of source compression technique. Furthermore, since the type of feedback indicated in (r.1) and (r.2) could be obtained using the simple transmission of a pulse and an energy detector at the receiver, this method of scheduling transmission provides a practical solution for the combined distributed compression and multiple access problem.

3.1. Complexity

In solving the information retrieval problem in a large sensor networks, the methods mentioned in Section 1 have a great amount of complexity that makes them difficult to implement. However, in group testing, once the set of tests and the possible groups are fixed, there are more or less centralized architectures that can implement the procedures. The algorithm could be completely distributed if the feedback was received by all nodes and all nodes knew how to identify the next (T, U) pair, *i.e.* by assuming that each node has knowledge of the $(\mathcal{T}, \mathcal{U}, \mathcal{F})$ group testing tree.

Leaving aside for a moment the issue of reliability and optimization of the physical test, as we argued before, a simple pulse transmission strategy and energy detection at the receiver could be the physical implementation to convey the desired feedback. In a distributed implementation, the nodes would know what question to ask in which time slot by synchronously following the path down the tree that is routed by the received feedback. Hence, the overhead involved in this operation could be reduced to that of synchronizing the nodes to a common time frame. This is not trivial, but there are effective methods to attain synchronization and it is certainly less complex than routing a cooperative MIMO transmission, which also have very demanding synchronization needs. Although, the optimization of the strategy over all possible choices within the unambiguous class is NP-Hard [13], the complexity of group testing lies in the construction of a good sequence of tests, i.e. it is in the design not in the implementation.

4. GROUP TESTING AND ARITHMETIC CODING

In group testing, the purpose of each test (T, U) is to provide the central agent with more information about the states of each agent. After each test, the central agent is able to eliminate or lower the probability of certain sequences $\mathbf{X}$. Therefore, the set of probable sequences are refined successively after each test, thus, allowing the central node to progressively resolve the state of the agents. Interestingly, the successive refinement property of group testing coincides with that of arithmetic coding [12]. In arithmetic coding, each uncoded sequence is assigned a non-overlapping region within the interval $(0, 1)$ that is equal to the probability of that particular sequence. Since the sum of the probability of all sequences that have the same length is equal to 1, the non-overlapping regions of these sequences cover entirely the $(0, 1)$ interval. The regions are constructed such that, for all m, the regions representing the m-length subsequence $[x_0, \cdots, x_{m-1}]$ is nested within the region representing the $(m-1)$-length subsequence $[x_0, \cdots, x_{m-2}]$. This can be done since the probability of $[x_0, \cdots, x_{m-2}]$ is equal to the probability of $[x_0, \cdots, x_{m-1}]$ saturated over all values of x_{m-1}. This standard construction of arithmetic coding allows the encoder to successively refine the region after each additional symbol is known within the entire sequence. For a sequence of fixed length N, the encoder will eventually assign a non-overlapping region to represent uniquely each particular sequence. The central agent that imposes the tests serves as an encoder that successively refines the possible set of sequences until all information is resolved.

Following the concept of arithmetic coding, we construct, in the group testing case, a mapping of each sequence $\mathbf{x}$ onto an interval of length $r_{\mathbf{x}} = \Pr(\mathbf{X} = \mathbf{x})$ within the $(0, 1)$ interval. Similarly, the interval of each sequence is nested within the interval of the subsequence that is a prefix of the original sequence. After each test, the central agent observes a feedback that allows it to refine the set of possible sequences by eliminating the interval that corresponds to the events with zero measure given the information obtained from previous tests. The refinement process continues until the path of (T, U)-tests leads to only one sequence within the refined interval. If a test (T, U) does not contribute in refining the set (or the interval), this test is considered to be redundant since no knowledge can be gained through this test. In view of its successive refinement property, the goal of group testing is equivalent to finding the fastest way to eliminate the impossible events (given the increased knowledge from each test) and efficiently identify the exact realization of the sequence. Therefore, it is desirable to design the tests in our strategy to eliminate at each stage the largest possible region within the $(0, 1)$ interval in order to reduce further tests. Based on this concept, we introduce a group testing algorithm, called the *Maximum Refinement Algorithm (MRA)*, with the purpose of providing insight into the successive refinement property of group testing.

Suppose that the channel to be used between the central and distributed agents is the noiseless OR channel, as in most group testing scenarios. In the MRA, the test is designed such that the question T_i imposed upon the group U_i^* in the i-th test is equal to "Is $[X_{i_1}, \cdots, X_{i_{|U_i|}}] = x_i^*$?", where the group U_i^* and the sequence x_i^* is defined by:

$$(U_i^*, x_i^*) = \arg\max_{U_i, \mathbf{x}} |U_i| \cdot \Pr\{[X_{i_1}, \cdots, X_{i_{|U_i|}}] = \mathbf{x} | T_0^{i-1}\} \quad (2)$$

where the conditioning on $(T, U)_0^{i-1}$ represents the information obtained from tests (T_0, U_0) to (T_{i-1}, U_{i-1}). The

optimal group and the optimal sequence, as defined in (2), is chosen to maximize the expected number of agents that are resolved at the present stage.

In general, the best (T, U) test, at each stage, is chosen over all possible groups and sequences. Therefore, although we construct the intervals to be nested with respect to their prefix sequences, we note that the refinement may not be nested unless the distribution among the states of agents are spatially homogeneous (since agents that are chosen may not be consecutive to each other). If a test is negative, meaning that one of the bits in the guessed sequence is wrong, then we eliminate the interval or intervals representing this sequence. If the test is positive, then the complement of the intervals representing this sequence is eliminated which refines the region to the intervals that contain the sequences of resolved agents. The choice of the test in (2) refines the interval by a large amount if the answer is positive and reduces the number of future tests if the answer is negative. We note that this method is suboptimal, in general, since it considers only the step-by-step optimization instead of considering all possible events that may occur in future tests.

4.1. Independent Bernoulli Case

In the classical setting, the state of agents are modelled as *i.i.d.* Bernoulli random variables with parameter p. The typical question that is asked for a group of length m is equal to "Is $[X_{i_1}, \cdots, X_{i_m}] = [0, \cdots, 0]$?". From the point of view of maximizing the refinement of each test, the choice of asking the all 0 sequence is optimal for $p < 1/2$ since the probability of this sequence has the highest probability among all sequences of the same length. However, this is not true when the probability $p > 1/2$. In this case, the all 1 sequence has the highest probability of occurring over all other sequences. This property was remarked by Berger in [9] where the reversing technique was applied to choose the all 1 sequences as questions when $p > 1/2$. However, without applying the reversing technique, the best strategy for p larger than a certain cutoff point $p^* = \frac{1}{2}(3 - \sqrt{5})$ would be no better than to test each node individually (see *e.g.* [9]).

In Fig. 1, we compare the performance of the MRA to that of the Recursive Algorithm shown in [9] without applying the reversing technique. We notice that even with the inferior property of not considering completely the future events in our algorithm, the MRA still achieve an average number of tests that is close to the optimal recursive method. This shows that the successive refinement property is truely the essence of group testing since it contributes the most to its performance. (In fact, for $p > 1/2$, the MRA performs optimally in choosing only one agent in each test.) This provides us with great insight in designing group testing strategies for other generalized applications.

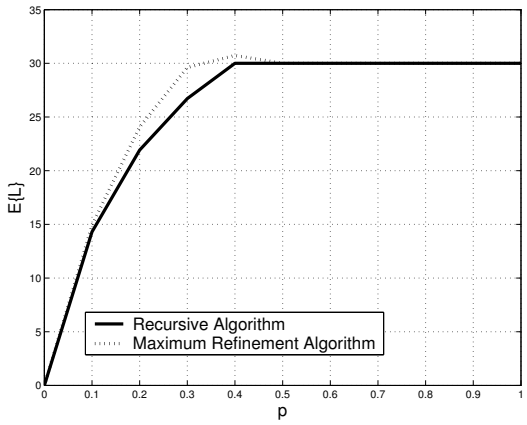

Fig. 1. For $N = 30$, we show the average number of tests $\mathbf{E}\{L\}$ vs p for the MRA and the Recursive Algorithm.

5. REFERENCES

[1] P. Gupta and P.R. Kumar, "The capacity of wireless networks," *IEEE Trans. on IT*, vol. 46, no. 2, Mar. 2000.

[2] S.S. Pradhan, J. Kusuma, and K. Ramchandran, "Distributed compression in a dense microsensor network," *IEEE Signal Processing Mag.*, vol. 19, no. 2, pp. 51–60, Mar. 2002.

[3] Qian Zhao and M. Effros, "Lossless and near-lossless source coding for multiple access networks," *IEEE Trans. Inform. Theory*, vol. 49, no. 1, pp. 112–128, Jan. 2003.

[4] A. Scaglione and S. Servetto, "On the interdependence of routing and data compression in multi-hop sensor networks," *ACM/Kluwer Mobile Networks and Appl. (MONET)*, 2003.

[5] D. Marco, E. Duarte-Melo, M. Liu, and D. L. Neuhoff, "On the many-to-one transport capacity of a dense wireless sensor network and the compressibility of its data," in *Proc. of IPSN*, Palo Alto, CA, Apr. 2003.

[6] A. Jovicic, P. Viswanath, and S. R. Kulkarni, "Upper bounds to transport capacity of wireless networks," in *to appear in IEEE Trans. on Information Theory*, Nov. 2004.

[7] Robert Dorfman, "The detection of defective members of large population," *The Annals of Mathematical Statistics*, vol. 14, no. 4, pp. 436–440, Dec. 1943.

[8] Milton Sobel and Phyllis A. Groll, "Group testing to eliminate efficiently all defectives in a binomial sample," *The Bell System Tech. Journal*, vol. 38, pp. 1179–1253, Sept. 1959.

[9] T. Berger, N. Mehravari, D. Towsley, and J. Wolf, "Random multiple-access communication and group testing," *IEEE Trans. Commun.*, vol. 32, no. 7, pp. 769–779, July 1984.

[10] Yao-Win Hong and Anna Scaglione, "On multiple access for correlated sources: A content-based group testing approach," in *to be published in IEEE IT Workshop*, Oct. 2004.

[11] Yao-Win Hong and Anna Scaglione, "Content-based multiple access: Combining source and multiple access coding for sensor networks," in *to be published in IEEE International Workshop on Multimedia Signal Processing*, Sept. 2004.

[12] J. Rissanen and G.G. Langdon, "Arithmetic coding," *IBM J. of Res. and Develop.*, vol. 23, no. 2, pp. 149–162, Mar. 1979.

[13] Ding-Zhu Du and Frank K. Hwang, *Combinatorial Group Testing and Applications*, vol. 12 of *Series on Applied Mathematics*, World Scientific Pub Co Inc., 2 edition, Oct. 1993.