EDGE SEARCH IN GRAPHS AND HYPERGRAPHS OF BOUNDED RANK: AVERAGE SEARCH LENGTHS IN THE WEIGHTED CASE

Ingo Althöfer
Fakultät für Mathematik
Universität Bielefeld
Postfach 100 131
33501 Bielefeld
Germany

Abstract

Consider a finite undirected graph $G = (X, E)$ with a probability distribution $P = (p_1, p_2, \ldots, p_m)$ on the set E of its edges. What is the average case complexity $\overline{L}(G, P)$ of finding an unknown edge $e^* \in E$ which is distributed according to P if the following tests are admitted: For any $Y \subset X$ we may test whether $e^* \subset Y$ or not.

We prove the existence of a constant c_2 such that $H(p_1, \ldots, p_m) \leq \overline{L}(G, P) \leq H(p_1, \ldots, p_m) + c_2$ for all pairs (G, P) . Here $H(P)$ denotes the entropy of the distribution P . The upper bound stems from a construction which is presented in this note.

The result is generalized to hypergraphs of bounded rank r . A hypergraph is said to have rank r if each hyperedge in it contains at most r vertices.

1. Introduction

Suppose $G = (X, E)$ is a finite undirected graph without loops or multiple edges. In [Aig 86], Aigner introduced the following search problem on graphs which generalizes group testing problems with two defectives:

The search domain is the edge set E of G . Admissible tests are the following: For each subset $Y \subset X$ we may test whether the unknown edge e^* is in the induced subgraph $G[Y]$ or not. Let $L(G)$ denote the worst case complexity of this search problem.

The information theoretic lower bound is $L(G) \geq \lceil \log_2 |E| \rceil$, and there are graphs G with $L(G) = \lceil \log_2 |E| \rceil + 1$.

Recently Damaschke proved [Dam 93] that for all graphs G

$$L(G) \leq \lceil \log_2 |E| \rceil + 1 \ .$$

In this note we investigate the *average* search costs in case of a probability distribution $P = (p_1, \ldots, p_m)$ on the set E of edges. For a fixed search scheme and every edge $e \in E$ let $\mathrm{cost}(e)$ denote the number of questions required in case of $e^* = e$. We want to find a scheme that minimizes $\sum_{e \in E} p(e)\mathrm{cost}(e)$.

For given G and P , let $\overline{L}(G, P)$ denote the minimum possible value for this sum. The information theoretic lower bound is

$$\overline{L}(G, P) \geq H(p_1, \ldots, p_m),$$

where $H(P) = H(p_1, \ldots, p_m) = -\sum_{i=1}^{m} p_i \log_2 p_i$ is the entropy of P .

Theorem 1: *For every graph G with a probability distribution P on its edge set we give a search scheme which yields $cost(e) \leq -\log_2 p(e) + c_2$ for every edge $e \in E$. The absolute constant c_2 does not depend on G or P .*

Corollary 2: $\overline{L}(G, P) \leq H(p_1, \ldots, p_m) + c_2$ *for every graph G and every probability distribution P on its edge set.*

This paper is organized as follows. In Section 2 we present the search scheme on which Theorem 1 is based. Section 3 contains the proof of Theorem 1. In Section 4 we sketch a straightforward generalisation of our search scheme to hypergraphs of bounded rank. A hypergraph is said to have rank r if each hyperedge in it contains at most r vertices.

Theorem 3: *For every hypergraph H of bounded rank r with a probability distribution P on its set E of hyperedges there is a search scheme which yields $cost(e) \leq -\log_2 p(e) + c_r$ for every $e \in E$. The absolute constant $c_r > 0$ depends on the rank bound r , but not on the hypergraph H and P .*

Taking the equal distribution on the set of hyperedges, our arguments give a new and simpler proof of the Hypergraph Theorem 3.4 in [AlT 93, p. 8].

Corollary 4: $\overline{L}(H, P) \leq H(P) + c_r$ *for every hypergraph H of bounded rank r and every probability distribution on its set of hyperedges.*

Section 5 contains a short discussion and several open problems.

We refer the reader to [Aig 88] for an introduction to combinatorial search theory. In [AW 87] search problems are discussed in a more general setting.

2. The Algorithm in the Graph Case

Our algorithm is based on a specific version of the classical Shannon–Fano algorithm in which the set of candidates is approximately halved in each step.

Let $G_t = (X_t, E_t, P_t)$ be the resulting graph after the first t tests, where P_t is a probability distribution on E_t, resulting from the first t tests. So $G_0 = G = (X, E, P)$.

If $|E_t| = 1$ then $E_t = \{e^*\}$, and we are done. Otherwise determine for every node $x \in X_t$ the sum

$$q_t(x) = \sum_{\substack{e \in E_t: \\ x \in e}} p_t(x) \ .$$

Sort the nodes in decreasing order according to their q–values, hence

$$q_t(x_1) \geq q_t(x_2) \geq \cdots \geq q_t(x_{n_t}), \quad \text{with} \quad n_t = |X_t| \ .$$

Find the smallest $k \leq n_t$ such that

$$\sum_{e \in \widetilde{E}_t(k)} p_t(e) \geq \frac{1}{2}, \quad \text{where} \quad \widetilde{E}_t(k) = \big\{ e \in E_t \mid e \subset \{x_1, \ldots, x_k\} \big\} \ . \tag{2.1}$$

We distinguish two cases.

Case 1: $k = n_t$.

Then $Y = \{x_1, \ldots, x_{k-1}\}$ will be the $(t+1)$–th test set.

Case 2: $k < n_t$.

Then $Y = \{x_1, \ldots, x_k\}$ will be the $(t+1)$–th test set.

In both cases continue as follows.

Let $\sum_{\substack{e \in E_t: \\ e \subset Y}} p_t(e) = s_t$.

If $e^* \subset Y$, then $X_{t+1} = Y$, $E_{t+1} = \{e \in E_t \mid e \subset Y\}$, and $p_{t+1}(e) = \frac{p_t(e)}{s_t}$ for all $e \in E_{t+1}$.

If $e^* \not\subset Y$, then $X_{t+1} = X_t$, $E_{t+1} = \{e \in E_t \mid e \not\subset Y\}$, and $p_{t+1}(e) = \frac{p_t(e)}{1 - s_t}$ for all $e \in E_{t+1}$.

So, after the $(t+1)$–th test we know that $e^* \in E_{t+1}$ and that, depending on the original distribution $P = P_0$ and the first $(t+1)$ tests, e^* is now distributed according to the distribution P_{t+1} of conditional probabilities.

3. The Proof in the Graph Case

Our proof is based on the following ideas: A single test with its result is "okay" if the remaining candidates together have at most half of the original probability. So by condition (2.1) at least "all" tests with $e^* \not\subset Y$ will be okay. (Let us for a moment forget the exceptional Case 1. This situation can happen at most once at the end of the search process.)

"Bad" may be the tests with $e^* \subset Y$. By Lemma 3.1 below, however, there can not be too many tests with this answer as after every such test we get $|X_{t+1}| < \frac{3}{4}|X_t| + 1$. Moreover, by Lemma 3.2 below we have $\sum_{e \in Y} p_t(e) \leq \frac{1}{2} + \frac{2}{|Y|}$. Hence these tests fail to be okay only by a small margin if Y is large.

Let $\{k_0, k_1, \ldots, k_T\}$ with $|X| = k_0 > k_1 > \cdots > k_T > 0$ be the set of all cardinalities $|X_t|$ that occur during the search process. Combining all the bad steps, we have to estimate the following product

$$\left.\begin{aligned} &\prod_{s=1}^{T} \left[2\left(\frac{1}{2} + \frac{2}{k_s}\right)\right] \\ &= \prod_{s=1}^{T} \left(1 + \frac{4}{k_s}\right) \\ &= \left(1 + \frac{4}{k_T}\right) \cdot \left(1 + \frac{4}{k_{T-1}}\right) \cdot \ldots \cdot \left(1 + \frac{4}{k_1}\right) \\ &\approx (1+4) \cdot \left(1 + 4 \cdot \frac{3}{4}\right) \cdot \left(1 + 4 \cdot \left(\frac{3}{4}\right)^2\right) \cdot \ldots \cdot \left(1 + 4 \cdot \left(\frac{3}{4}\right)^{T-1}\right) \\ &< \prod_{s=0}^{\infty} \left(1 + 4 \cdot \left(\frac{3}{4}\right)^s\right) \ . \end{aligned}\right\} \qquad (3.1)$$

The constant c_2 in Theorem 1 is given by the binary logarithm of the product in (3.1) plus some extra terms for rounding errors ($k_s \approx \frac{3}{4}k_{s-1}$ instead of $k_s < \frac{3}{4}k_{s-1} + 1$)and the tests with small sets X_t .

Let $G = (X, E)$ be an undirected graph with n nodes and a probability distribution P on its set E of edges.

Like in Section 2 we define node weights

$$q(x) = \sum_{e : x \in e} p(e)$$

and sort the nodes so that $q(x_1) \geq \cdots \geq q(x_n)$. By definition $\sum_{i=1}^{n} q(x_i) = 2$. Let $\widetilde{E}(d) = \{e \in E \mid e \subset \{x_1, \ldots, x_d\}\}$. Then there is a unique $k \in \{1, \ldots, n\}$ such that

$$\sum_{e \in \widetilde{E}(k-1)} p(e) < \frac{1}{2} \leq \sum_{e \in \widetilde{E}(k)} p(e) \ . \tag{3.2}$$

Lemma 3.1: $k < \frac{3}{4}n + 1$.

Proof: $\displaystyle\sum_{e : e \notin \widetilde{E}(k-1)} p(e) > \frac{1}{2}$,

hence

$$\sum_{i=k}^{n} q(x_i) > \frac{1}{2} \ . \tag{3.3}$$

By the monotonicity of the q–weights we have

$$\sum_{i=k}^{n} q(x_i) \leq 2 \cdot \frac{n-(k-1)}{n} \ . \tag{3.4}$$

(3.3) and (3.4) together give

$$\frac{1}{4} < 1 - \frac{k-1}{n},$$

which is equivalent to the statement of the Lemma.

■

Lemma 3.2: *With the* k *of (3.2) we have*

$$\sum_{e \in \widetilde{E}(k)} p(e) < \frac{1}{2} + \frac{2}{k} \ .$$

Proof: $\displaystyle\sum_{i=1}^{k} q(x_i) \leq 2$,

hence $q(x_k) \leq \frac{2}{k}$ by the monotonicity.

Thus

$$\sum_{e \in \widetilde{E}(k)} p(e) \leq \sum_{e \in \widetilde{E}(k-1)} p(e) + q(x_k)$$
$$< \frac{1}{2} + \frac{2}{k} \ .$$

■

4. Generalisation to Hypergraphs

In the case of a hypergraph (X, E, P) with upper rank bound r we define our node weights $q(\,)$ in a slightly different way, namely

$$q(x) = \sum_{e : x \in e} \frac{p(e)}{|e|} .$$

This guarantees $\sum\limits_{x \in X} q(x) = 1$.

As in the graph case our algorithm is based on Shannon–Fano's classical halving procedure.

Let $q(x_1) \geq \cdots \geq q(x_n)$ and $\widetilde{E}(d) = \big\{ e \in E \mid e \subset \{x_1, \ldots, x_d\} \big\}$. Then there is a unique $k \in \{1, \ldots, n\}$ such that

$$\sum_{e \in \widetilde{E}(k-1)} p(e) < \frac{1}{2} \leq \sum_{e \in \widetilde{E}(k)} p(e) . \tag{4.1}$$

Lemma 4.1: $k < \left(1 - \frac{1}{2r}\right) \cdot n + 1$.

Setting $r = 2$ shows that Lemma 3.1 is covered by this result.

Proof of Lemma 4.1:

$$\begin{aligned} \sum_{i=k}^{n} q(x_i) &\geq \frac{1}{r} \cdot \sum_{e \notin \widetilde{E}(k-1)} p(e) \\ &= \frac{1}{r} \cdot \left[1 - \sum_{e \in \widetilde{E}(k-1)} p(e) \right] \\ &> \frac{1}{2r} \qquad \text{by condition (4.1)} . \end{aligned} \tag{4.2}$$

By the monotonicity of the q–weights we have (analogously) to (3.4))

$$\sum_{i=k}^{n} q(x_i) \leq 1 - \frac{k-1}{n} . \tag{4.3}$$

(4.2) and (4.3) together give

$$\frac{1}{2r} < 1 - \frac{k-1}{n},$$

which is the desired inequality.

■

Lemma 4.2: *With the* k *of (4.1) we have*

$$\sum_{e \in \widetilde{E}(k)} p(e) < \frac{1}{2} + \frac{r}{k} \; .$$

Proof:

$$\sum_{i=1}^{k} q(x_i) \leq 1 \; ,$$

hence $q(x_k) \leq \frac{1}{k}$ by the monotonicity.

Thus

$$\begin{aligned} \sum_{e \in \widetilde{E}(k)} p(e) &\leq \sum_{e \in \widetilde{E}(k-1)} p(e) + r \cdot q(x_k) \\ &< \frac{1}{2} + \frac{r}{k} \; . \end{aligned}$$

■

The search scheme for hypergraphs is designed analogously to the one of Section 2 for graphs:

Let $H_t = (X_t, E_t, P_t)$ be the resulting hypergraph after the first t tests with $|X_t| = n_t$, $q_t(x_1) \geq \cdots \geq q_t(x_{n_t})$, and

$$\sum_{e \in \widetilde{E}_t(k-1)} p_t(e) < \frac{1}{2} \leq \sum_{e \in \widetilde{E}_t(k)} p_t(e) \; .$$

For $k < n_t$ $Y = \{x_1, \ldots, x_k\}$ will be the next test set, for $k = n_t$ we choose $y = \{x_1, \ldots, x_{k-1}\}$.

By Lemma 4.1 we may have $k = |X_t|$ only for $n < 2r$, hence at most a limited number of times. In all other steps the answer $e^* \notin E_t(k)$ will be a "good" answer, and by a combination of Lemmas 4.1 and 4.2 we can bound the number c_r of extra questions like in the proof in Section 3.

5. Discussion and Open Questions

(i) As shown in [AiT 93] the problem of halving an edge set is NP–complete. We are firmly convinced that also the task to find an optimal test scheme for a weighted graph (X, E, P) is NP–complete. A proof of such a statement, however, might be technically extraordinarily complicated.

(ii) The Theorems 1 and 3 give good bounds for every edge of the graph and hypergraph, respectively. Here "good" means that the natural candidate $-\log p_i$ from the information–theoretic lower bound is missed by at most a constant.

Hence our scheme gives a mixture of average and worst case result. Such statements for every edge do not hold for several traditional codes, including the optimal Huffman codes: Assume the distribution $(p_1, \dots, p_m)$ with $\frac{p_{i+1}}{p_i} = \frac{\sqrt{5}-1}{2}$ for all i. In the Huffman code the i–th codeword gets length $\ell_i = i$, but $p_i \approx \left(\frac{\sqrt{5}-1}{2}\right)^i$. Hence

$$\ell_i \approx \frac{1}{\log_2 \frac{2}{\sqrt{5}-1}} \cdot \log_2 \frac{1}{p_i} \approx 1.44 \cdot \log_2 \frac{1}{p_i} .$$

(iii) Neither in [AlT 93] nor in this note we have presented explicite values for the constants $c_2, c_3, \dots$. But at least in the graph case it should be possible to prove some concrete (and small) constant c_2 and even show its optimality.

Open Question: Is there a graph with a probability distribution P on its edge set such that $-\log_2(p(e)) + 1$ tests do not suffice for every edge e in E?

(iv) Combining the statements of Theorem 1 with constant c_2 for all edges gives "only" the same constant c_2 in Corollary 2.

Open Question: What is the best possible constant for the average complexity? How much smaller than the best constant for Theorem 1 is it?

(v) Algorithms with randomization may help to improve the worst case constant $c_2', c_3', \dots$ for unweighted (hyper)graphs and the "average case" constants $c_2, c_3, \dots$ for weighted hypergraphs. We formulate the key idea for unweighted graphs $G = (X, E)$:

Sometimes the most balanced way to split the edge set $E = \{e_1, \dots, e_m\}$ leads to a set $Y \subset X$ with $\left||E[Y]| - \frac{1}{2}|E|\right| = 0(\sqrt{m})$, [Dam 93]. Here $E[Y] = \{e \in E | e \subset Y\}$. Now consider a set $\{Y_1, \dots, Y_t\}$ of any t subsets of X. For each edge $e \in E$ and each Y_i we define

$$g_i(e) = \begin{cases} |E[Y_i]|, & \text{if } e \subset Y_i, \\ |E| - |E[Y_i]|, & \text{if } e \not\subset Y_i, \end{cases}$$

and $g(e) = \frac{1}{t}\sum_{i=1}^{t} g_i(e)$. Finally we define $g_{\max} = \max_{e\in E} g(e)$.

By the worst case result of Damaschke we can obviously get $g_{\max} \le \frac{m}{2} + 0(\sqrt{m})$, even with $t = 1$.

Open Question: Is there some $\varepsilon < \frac{1}{2}$ such that $g_{\max} \le \frac{m}{2} + 0(m^{\varepsilon})$ is realizable for sufficiently large $t = t(G)$ in every graph G with m edges?

For hypergraphs and sets $\{Y_1, \dots, Y_t\}$ of subsets we may define $g_{\max}$ analogously. From [AlT 93] we know that $g_{\max} \le \frac{m}{2} + 0(m^{1-\frac{1}{r}})$ is realizable with $t = 1$ for every hypergraph of bounded rank r .

"Optimistic" Hope: Perhaps $g_{\max} = \frac{m}{2} + 0(m^{1-\frac{2}{r}})$ is realizable for every hypergraph of bounded rank r .

In the case of weighted hypergraphs $H = (X, E, P)$ we define

$$g_i(\tilde{e}) = \begin{cases} \sum_{e: e\subset Y_i} p(e), & \text{if } \tilde{e} \subset Y_i, \\ \sum_{e: e\not\subset Y_i} p(e), & \text{if } \tilde{e} \not\subset Y_i \end{cases}$$

and hope for systems $Y_1, \dots, Y_t$ of subsets of X , for which $\max_{e\in E} \left[\frac{1}{t}\sum_{i=1}^{t} g_i(e) - \frac{1}{2} \right]$ is "very small".

(vi) The results of this note are proved only for hypergraphs of bounded rank. Given a hypergraph with arbitrary edge sizes and a probability distribution on its edge set the average edge size is defined in a natural way.

Open Question: Is there a sequence $d_2', d_3', \dots$ of constants such that the average test number is not larger than $H(P) + d_i'$ in every hypergraph with average edge size $\le i$?

Answers to this question would be interesting also for the group testing situation with independently and identically distributed defects and (very small) defect probabilities p , see [AW 87].

6. References

[Aig 86] M. Aigner. Search problems on graphs. Discrete Applied Mathematics 14 (1986), 215–230.

[Aig 88] M. Aigner. Combinatorial Search. Wiley–Teubner, New York, 1988.

[AiT 93] M. Aigner and E. Triesch. Realizability and Uniqueness in Graphs. To appear in Discrete Mathematics.

[AlT 93] I. Althöfer and E. Triesch. Edge search in graphs and hypergraphs of bounded rank. Discrete Mathematics 115 (1993), 1–9.

[AW 87] R. Ahlswede and I. Wegener. Search Problems. Wiley–Teubner, New York, 1987. (Translation of the original German version "Suchprobleme", Teubner, 1979.)

[Dam 93] P. Damaschke. A tight upper bound for group testing in graphs. To appear in Discrete Applied Mathematics.

[Tri 90] E. Triesch. Suchprobleme in Graphen und Hypergraphen. Habilitation Thesis, RWTH Aachen, Department of Mathematics and Computer Science.