THE METHOD OF CONDITIONAL PROBABILITIES: DERANDOMIZING THE PROBABILISTIC METHOD

JAMES ZHOU

Abstract. We describe the probabilistic method as a nonconstructive way of proving the existence of combinatorial structures. We then introduce the method of conditional probabilities as a procedure for derandomizing these probabilistic proofs into deterministic algorithms. We develop the idea of pessimistic estimators as a tool for this end. We end with an application to a combinatorial game.

Contents

1. Introduction

A frequent interest in combinatorics is determining the existence of structures with certain given properties and what bounds suffice for such existence. The *probabilistic method* is an important nonconstructive proof approach developed in the second half of the 20th century, notably by Paul Erdős. In **Section 2**, we consider a basic example to illustrate the idea.

While the probabilistic method provides a means of determining the existence of structures with certain properties, obtaining such structures remains of theoretical and practical interest. Despite the fact that the probabilistic method is nonconstructive, we can in some cases leverage it to create deterministic algorithms for obtaining the desired structures. We introduce the procedure for doing so, known as the *method of conditional probabilities*, in **Section 3**, and we further develop it in **Section 4**. Finally, we apply it to a combinatorial game through a slight adaptation in **Section 5**.

2. The Probabilistic Method

Traditional constructive proofs in combinatorics give explicit examples of classes of structures that meet given properties. The *probabilistic method* instead considers

Date: AUGUST 26, 2018.

the entire class of structures in question as a probability space, whose points may or may not satisfy the properties. If the probability that an arbitrary point in the probability space meets the properties is positive, there must exist one such point. For illustration, consider the following example from [4], originally appearing in [5].

A *tournament* T is a complete directed graph, denoted by ordered pairs of vertices. A tournament T is *transitive* if $(x, y), (y, z) \in T \implies (x, z) \in T$.

Let $v(n)$ be the largest integer such that every tournament on $[n]$ contains a transitive subtournament on $v(n)$ vertices.

Theorem 2.1.

$$v(n) \leq 1 + \lfloor 2 \log_2 n \rfloor.$$

Proof. Consider the discrete uniform distribution whose points are members of $\mathcal{T}_n$, the set of all tournaments on $[n]$. We may view this as choosing the direction for each edge independently, where for a given $T \in \mathcal{T}_n$ and $i, j \in [n]$, the probability that $(i, j) \in T$ is $\frac{1}{2}$.

We wish to consider the probability of the event C where T contains a transitive subtournament on v players.

Consider a subtournament on $A \subseteq [n]$ with $|A| = v$. Let B be the event where this subtournament is transitive.

For a given A, each transitive subtournament on A is in bijective correspondence with an ordering on the vertices in A. Thus, the number of transitive subtournaments on A is equal to the number of permutations on $[v]$. There are a total of $2^{\binom{v}{2}}$ equally likely subtournaments for A, so $P(B) = v!2^{-\binom{v}{2}}$.

By the union bound, the probability that any subtournament over all A is transitive is less than or equal to the sum of the probabilities that a subtournament for each given A is transitive. There are $\binom{n}{v}$ possible A, so

$$P(C) \leq \binom{n}{v} P(B) = \binom{n}{v} v!2^{-\binom{v}{2}}.$$

Suppose $v > 1 + 2\log_2 n$. Then

$$\binom{n}{v} v!2^{-\binom{v}{2}} = \frac{n!}{(n-v)!} 2^{-\binom{v}{2}} = \frac{n!}{(n-v)!} \frac{1}{2^{v(v-1)/2}} < \frac{n!}{(n-v)!} \frac{1}{n^v} < 1.$$

Since the probability that there exists a transitive subtournament is less than 1, there must exist a tournament with no transitive subtournaments.

Thus, $v(n) \leq 1 + 2\log_2 n$, and $v(n)$ must be integral, which completes the proof. □

Alternatively, we may make use of random variables and expectation, where the expectation serves as a bound on the number of desired substructures.

Proof. Consider again the symmetric probability space whose points are the elements $T \in \mathcal{T}_n$. Define A and B as before. Let X be the indicator variable for B, so

$$X = \begin{cases} 1 & \text{if } B. \\ 0 & \text{if not } B. \end{cases}$$

The expected value $E(X) = P(B) = v!2^{-\binom{v}{2}}$.

Let Y be the random variable denoting the number of transitive subtournaments, so Y is the sum of X over all $\binom{n}{v}$ possible A. As before, suppose $v > 1 + 2\log_2 n$. By the linearity of expectation,

$$E(Y) = E(\sum X) = \sum E(X) = \binom{n}{v} v! 2^{-\binom{v}{2}} < 1.$$

There must exist a tournament with at most as many transitive subtournaments as the expectation. Since the expectation is less than 1, there exists a tournament with 0 transitive subtournaments, the only valid such value of Y. The rest of the proof follows as before. □

More sophisticated problems may require bounds on probabilities rather than exact values, such as **Theorem 4.5**, or other alterations we do not consider here. Many examples can be found in [1] and [4].

3. The Method of Conditional Probabilities

We have not yet considered how to obtain desired combinatorial structures. In the example from **Theorem 2.1**, randomized selection may be acceptable in some cases. For instance, suppose we are seeking a tournament of size 127 not containing a transitive subtournament of size 16. Then the probability that a randomly selected tournament meets this condition is bounded by

$$1 - \binom{127}{6} 16!\, 2^{-\binom{16}{2}} > 0.998.$$

Although our selection is not guaranteed to succeed, for practical considerations this approach suffices.

However, a deterministic way of obtaining structures is still preferred, especially in cases where the probabilities are not as favorable. To this end, we utilize the *method of conditional probabilites*, which we illustrate through an example from [1].

Theorem 3.1. *There exists an edge two-coloring of the complete graph K_n with at most $\binom{n}{a}2^{1-\binom{a}{2}}$ monochromatic K_a.*

Proof. Color the edges independently at random (in a discrete uniform distribution on all possible colorings. If we do not specify, we always refer to this case.) The probability that any K_a is monochromatic is $2^{1-\binom{a}{2}}$, so the expected number of monochromatic K_a is $\binom{n}{a}2^{1-\binom{a}{2}}$. There exists a coloring with at most as many monochromatic K_a as the expectation. □

Theorem 3.2. *There exists a deterministic algorithm polynomial in n for finding such a coloring.*

Proof. Index the $\binom{n}{2}$ edges arbitrarily. Our algorithm colors the edges sequentially.

Consider a partial coloring where edges $\epsilon_1, \ldots, \epsilon_j$ are already colored. Suppose the remaining edges are colored independently at random. Let P_j^i be the probability that a given K_a^i is monochromatic.

Note that values with a subscript j are conditional on the partial coloring on the edges $\epsilon_1, \ldots, \epsilon_j$.

If K_a^i has at least one edge of each color, then $P_j^i = 0$. If K_a^i has no edges that are colored, then $P_j^i = 2^{1-\binom{a}{2}}$. Otherwise, $P_j^i = 2^{x_j^i - \binom{a}{2}}$, where x_j^i is the the

number of edges K_a^i has of one color. The expected number E_j of monochromatic K_a is $\sum P_j$ over all K_a.

Note that the partial coloring can be completed to a coloring with a number of monochromatic K_a less than the expectation E_j. This is akin to the original existence proof, which deals with E_0.

Consider the two possible colorings for ϵ_{j+1}, which we call red and blue. Note that

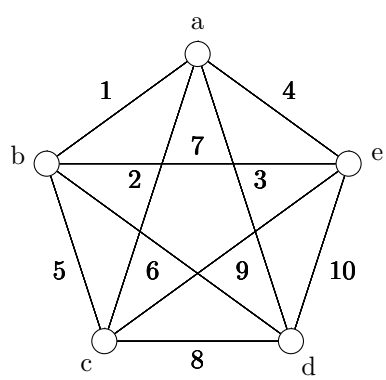
$$E_j = \frac{E_{j+1}(\text{If } \epsilon_{j+1} \text{ is red}) + E_{j+1}(\text{If } \epsilon_{j+1} \text{ is blue})}{2}.$$

Thus, there exists a choice of color for ϵ_{j+1} that decreases the expectation or keeps it the same.

Hence, if we follow the procedure where we color each edge sequentially with the color that minimizes the expectation, the expectation for each partial coloring is less than or equal to the original expectation for the uncolored graph. After all edges are colored in this manner, the expectation $E_{\binom{n}{a}}$ is then equal to the actual number of monochromatic edges, and we obtain the desired coloring.

For each K_a^i, the calculation of P_j^i has constant bound. There are $\binom{n}{a}$ such K_a, which is polynomial. We do these calculations at most 2 times for each of the $\binom{n}{2}$ edges, which is also polynomial. Thus, the algorithm is polynomial in n. □

Example 3.3. Consider the following graph and index for $a = 3$ and $n = 6$:



edge	index j
(a,b)	1
(a,c)	2
(a,d)	3
(a,e)	4
(b,c)	5
(b,d)	6
(b,e)	7
(c,d)	8
(c,e)	9
(d,e)	10

Initially, $E_0 = \binom{5}{3}2^{1-\binom{3}{2}} = 2.5$, so we seek a coloring with at most 2 monochromatic K_3 (triangles). Coloring edge-by-edge with red or blue, considering red first:

Edge 1*:*

Red gives 3 triangles with 1 red edge only and 7 triangles with no colored edges. Thus, $E_1(\text{red}) = (3+7)2^{1-\binom{3}{2}} = 2.5$. This is equal to E_0, so it is the minimum, and we select red. Noting that whenever red and blue are symmetric with equal expectations, we select red.

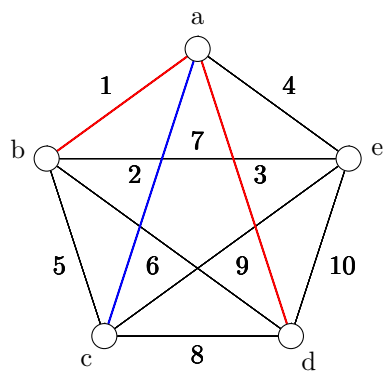
Edge 2*:*

Red gives 1 triangle with 2 red edges only, 4 triangles with 1 red edge only, and 5 triangles with no colored edges. Thus, $E_2(\text{red}) = (1)2^{2-\binom{3}{2}} + (4+5)2^{1-\binom{3}{2}} = 2.75$. This is greater than E_1, so we consider blue instead, which is guaranteed to succeed.

Blue gives 1 triangle with 1 red and 1 blue edge, 2 triangles with 1 red edge only, 2 triangles with 1 blue edge only, and 5 triangles with no colored edges. Thus, $E_2(\text{blue}) = (1)0 + (2+2+5)2^{1-\binom{3}{2}} = 2.25$. This is less than E_1, so we select blue.

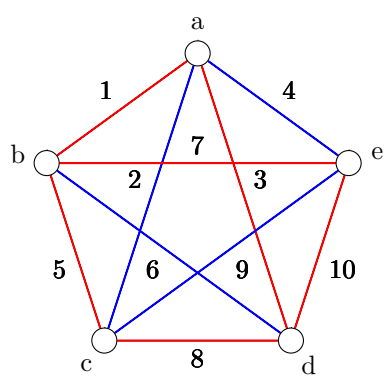
Edge 3:

Red gives 2 triangles with 1 red and 1 blue edge, 1 triangle with 2 red edges only, 2 triangles with 1 red edge only, 1 triangle with 1 blue edge only, and 4 triangles with no colored edges. Thus, $E_3(\text{red}) = (2)0 + (1)2^{2-\binom{3}{2}} + (2+1+4)2^{1-\binom{3}{2}} = 2.25$. This is equal to E_2, so we select red. After we color the first 3 edges, we have:



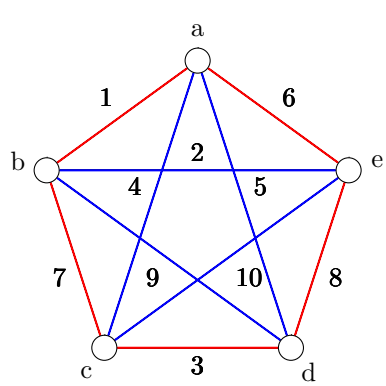
edge	j	color	E_j
(a,b)	1	red	2.5
(a,c)	2	blue	2.25
(a,d)	3	red	2.25
(a,e)	4	—	—
(b,c)	5	—	—
(b,d)	6	—	—
(b,e)	7	—	—
(c,d)	8	—	—
(c,e)	9	—	—
(d,e)	10	—	—

After running the algorithm to completion, we get:



edge	j	color	E_j
(a,b)	1	red	2.5
(a,c)	2	blue	2.25
(a,d)	3	red	2.25
(a,e)	4	blue	2
(b,c)	5	red	2
(b,d)	6	blue	1.25
(b,e)	7	red	1.25
(c,d)	8	red	1.25
(c,e)	9	blue	1
(d,e)	10	red	1

We end with 1 monochromatic triangle, which is less than our original expectation E_0 and equal to our final expectation E_{10}. Note that while the algorithm is deterministic, it depends on our index. With a different index, we get:



edge	j	color	E_j
(a,b)	1	red	2.5
(b,e)	2	blue	2.25
(c,d)	3	red	2.25
(a,c)	4	blue	1.75
(a,d)	5	blue	1.5
(a,e)	6	red	1
(b,c)	7	red	1
(d,e)	8	red	1
(b,d)	9	blue	0.5
(c,e)	10	blue	0

We now have 0 monochromatic triangles, which is optimal. There does not exist a color-preserving isomorphism between the graphs colored by each index.

There are numerous potentially interesting questions about the relationship between the indices and the colorings resulting from them. For example, we can consider if there is an index that gives $E_{10} = 2$ or 2 monochromatic triangles, the highest number potentially possible with the method of conditional probabilities. More broadly, we may consider what range of final expectations we can have in general beyond our initial guaranteed bound E_0 for this and other problems. However, the answers to these questions are likely to be specific to each problem and not to the method of as a whole, so we do not consider them here.

Just like the original existence proof, the algorithm relies on conditional expectations to construct the graph with the given bound. We now describe the procedure used in this example, known as *the method of conditional probabilities*, more generally.

Suppose we have a finite probability space. It contains $\prod_{j=1}^{r} x_j$ points, represented by selections from r sets $\mathbf{v}_j = \{v_1^j, \ldots, v_{x_j}^j\}$, where $x_j = |\mathbf{v}_j|$. A point in the space is denoted by $(\epsilon_1, \ldots, \epsilon_r)$, where each ϵ_j is an element of $\mathbf{v}_j$.

Thus, when we select a point from the space, we select an element ϵ_j from each $\mathbf{v}_j$ independently. The selection of an v_k^j from each $\mathbf{v}_j$ is made with probability p_k^j, where $\sum_{k=1}^{x_j} p_k^j = 1$.

Let $A_1, \ldots, A_s$ be events in the space such that $\sum_{i=1}^{s} P(A_i) = b$, so b is the expected number of A_i that occur. We wish to find a point with at most b occurrences of A_i by sequentially selecting ϵ_j.

Note that if $s = 1$, then $b \leq 1$ should then be the probability of an unwanted event. Additionally, we can proceed similarly if we want a point above some lower bound instead.

Suppose we have already selected $\epsilon_1, \ldots, \epsilon_j$. Then

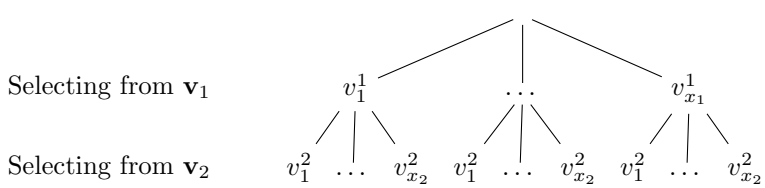
$$P(A_i \mid \epsilon_1, \ldots, \epsilon_j) = \sum_{k=1}^{x_j} [p_k^j P(A_i \mid \epsilon_1, \ldots, \epsilon_j, v_k^{j+1})].$$

Since the right hand side is a convex combination,

$$\sum_{i=1}^{s} P(A_i \mid \epsilon_1, \ldots, \epsilon_j) \geq \min \left(\sum_{i=1}^{s} [P(A_i \mid \epsilon_1, \ldots, \epsilon_j, v_k^{j+1})] \right).$$

Thus, selecting the ϵ_{j+1} that results in the minimum either decreases the expectation or keeps it the same so that it always remains less than or equal to the initial bound, b. When all ϵ are selected, the resulting point satisfies our condition.

We can visualize this as a decision tree.



Each level past the first (of which only two are shown) corresponds to a different $\mathbf{v}_j$. At each level, we select the ϵ_{j+1} that results in the lowest expectation, and our final point is the set of these selections. (The ϵ_{j+1} are ordered only due to our arbitrary index used for running the algorithm).

Note that in a practical algorithm, as long as our selections of ϵ_{j+1} keep the expectation below the initial bound, we do not necessarily have to select the ϵ_{j+1} that results in the minimum. For instance, we can instead select the first v_i^j as ϵ_{j+1} that gives an expectation less than or equal to the current conditional expectation, which could possibly be more efficient on average. If we do this, however, we may still have to calculate the resultant expectations for each v_i^j in $\mathbf{v}_j$ to select ϵ_{j+1}. We need to do this when we find the minimum, so the worst case inefficiencies for both algorithms is the same. Thus, we will continue to consider the minimum case.

In **Theorem 3.2**, the sample space is the set of colorings; the sets $\mathbf{v}_j$ are the sets of colors for each edge; the probabilities p_k^j are all symmetrically $\frac{1}{2}$, and the events A_i are whether a given K_a^i is monochromatic.

We now remark upon the example in **Theorem 2.1**. Recall that if $w(n) > 1+2\log_2 n$, then there exists a tournament on $[n]$ that does not contain a transitive subtournament on $w(n)$ vertices.

In principle, the method of conditional probabilities may also be applied. We index the edges and select one of two directions for each edge to minimize the expected number of transitive subtournaments. To do so, we calculate the conditional probabilities that each subset of $w(n)$ vertices can be completed to a transitive subtournament given the previous selections. However, these calculations are more complicated than those in **Theorem 3.2**, so we do not examine them.

4. Pessimistic Estimators

The efficiency of algorithms using this method is largely dependent on the efficiency of calculating the conditional probabilities. In **Theorem 3.1** and **Theorem 3.2**, these calculations are efficient. However, when this is not the case, using the method directly is impractical. In some examples, the precise probabilities may be difficult to calculate or unable to be calculated at all, as mentioned at the end of **Section 3**. In these cases, we may sometimes still apply the method by considering certain functions that are bounds on the probabilities.

Consider the general probability space described in **Section 3**. Let U be a function given any selections of $(\epsilon_1, \dots, \epsilon_j)$ such that

$$U(\epsilon_1, \dots, \epsilon_j) \geq \sum_{i=1}^{s} P(A_i \mid \epsilon_1, \dots, \epsilon_j).$$
$$U(\epsilon_1, \dots, \epsilon_j) \geq \min\left[U(\epsilon_1, \dots, \epsilon_j, v_k)\right].$$

Since U is an upper bound for the expectaions, we may apply the method of conditional probabilities with U in place of the expectations. We then obtain points with at most the initial value of U events occurring. U is known as a *pessimistic estimator* function [6]. A common case is when U is the sum of multiple U_i serving as bounds on individual $P(A_i \mid \dots)$.

We proceed to consider an example using pessimistic estimators developed in [6]. First, we prove a multiplicative Chernoff bound we use to construct pessimistic estimators for certain approximation problems.

Consider the random variable $\Psi = \sum_{j=1}^{r} a_j X_j$, where $a_1, \ldots, a_r \in (0,1]$ and the X_j are independent Bernoulli trials with $E(X_j) = p_j$. Let $m = E(\Psi) = \sum_{j=1}^{r} a_j p_j$.

Theorem 4.1. *If $\delta > 0$ and $m > 0$, then*

$$P(\Psi > (1+\delta)m) < \left[\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right]^m. \tag{4.2}$$

If $\gamma \in (0,1]$ and $m > 0$, then

$$P(\Psi < (1-\gamma)m) < \left[\frac{e^{\gamma}}{(1+\gamma)^{1+\gamma}}\right]^m. \tag{4.3}$$

Proof. For any $t > 0$,

$$P(\Psi < (1-\gamma)m) = P(e^{t\Psi} < e^{t(1-\gamma)m})$$

By Markov's Inequality, this is

$$\begin{aligned}
&< e^{t(1-\gamma)m} E(e^{-t\Psi}) \\
&= e^{tm(1-\gamma)} \prod_{j=1}^{r} E(e^{-ta_j X_j}) \\
&= e^{tm(1-\gamma)} \prod_{j=1}^{r} (p_j e^{-ta_j} + 1 - p_j) \\
&\leq e^{tm(1-\gamma)} \prod_{j=1}^{r} \exp[p_j(e^{-ta_j} - 1)].
\end{aligned}$$

Let $t = -\ln(1-\gamma)$. Then this is

$$= (1-\gamma)^{-(1-\gamma)m} \exp\left(\sum_{j=1}^{r} p_j[(1-\gamma)^{a_j} - 1]\right).$$

By Bernoulli's Inequality, this is

$$\begin{aligned}
&\leq (1-\gamma)^{-(1-\gamma)m} \exp\left(\sum_{j=1}^{r} -p_j a_j \gamma\right) \\
&= \left[\frac{e^{-\gamma}}{(1-\gamma)^{1-\gamma}}\right]^m.
\end{aligned}$$

Using techniques in calculus, we can show that

$$\left[\frac{e^{-\gamma}}{(1-\gamma)^{1-\gamma}}\right]^m < \left[\frac{e^{\gamma}}{(1+\gamma)^{1+\gamma}}\right]^m$$

within the $(0,1]$ interval for γ. We omit it for brevity. □

We also omit the proof the proof for (4.2) as it is similar and included in [6].

Let $B(m,\delta) = \left[\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right]^m$. $B(m,\delta)$ is a symmetric bound on the probability.

Let $D(m,x)$ be such that $B(m, D(m,x)) = x$. $D(m,x)$ is the deviation that gives a bound of x.

Remark 4.4. The following upper bounds on $D(m,x)$ are given in [6], which we may calculate in place of $D(m,x)$ where appropriate.

If $m > \ln \frac{1}{x}$, then

$$D(m,x) \leq (e-1)\left[\frac{\ln(1/x)}{m}\right]^{1/2}.$$

If $m \geq \ln \frac{1}{x}$, then

$$D(m,x) \leq \frac{e \ln 1/x}{m \ln[(e \ln 1/x)/m]}.$$

In approximation problems, we want the error in our approximations to be bounded by some factor involving $D(m,x)$. Our relevant probabilities for the probabilisitc method are then to be bounded by $B(m,\delta)$. We proceed to the following problem using pessimistic estimators, again from [6].

The Lattice Approximation Problem. Let C be an $n \times r$ matrix with $c_{ij} \in \{0,1\}$ and $\mathbf{p}$ be an r vector with $p_j \in [0,1]$. We wish to approximate $\mathbf{p}$ by some r vector $\mathbf{q}$ with $q_j \in \{0,1\}$, where C indicates which components of $\mathbf{p}$ are to be approximated and to what degree.

Let δ_i be the discrepancies

$$\delta_i = \left|\sum_{j=1}^{r} c_{ij}(p_j - q_j)\right|,$$

which we bound by the inner products

$$s_i = \sum_{j=1}^{r} c_{ij} p_j.$$

Assign each q_i with a value of 1 with probability p_i, so we can define the random variable $\Psi_i = \sum_{j=1}^{r} c_{ij} q_j$. Note that $E(\Psi_i) = s_i$ and $\delta_i = |\Psi_i - s_i|$. Define $B(m,\delta)$ and $D(m,x)$ in accordance with our previous bounds.

Theorem 4.5. *There exists a* $\mathbf{q}$ *such that*

$$\delta_i \leq s_i D(s_i, \frac{1}{2n})$$

for all i.

Proof. By our definitions and our q_i assignment above, for each i,

$$P[\Psi_i > s_i(1 + D(s_i, \frac{1}{2n}))] < \frac{1}{2n},$$
$$P[\Psi_i < s_i(1 - D(s_i, \frac{1}{2n}))] < \frac{1}{2n},$$

so $P[\delta_i > s_i D(s_i, \frac{1}{2n})] < \frac{1}{n}$.

Thus, the expected number of i such that $\delta_i > s_i D(s_i, \frac{1}{n})$ is less than the sum of these bounds, $n\frac{1}{n} = 1$, so there exists a $\mathbf{q}$ with no such i. □

Theorem 4.6. *There exists a deterministic algorithm for finding such a* $\mathbf{q}$ *polynomial in* n *and* r.

Proof. Let $L_{i+} = s_i(1 + D(s_i, \frac{1}{2}n))$ and $L_{i-} = s_i(1 - D(s_i, \frac{1}{2}n))$.

Let $t_i > 0$. Then for any fixed i, similarly to **Theorem 4.1**,

$$\begin{aligned} P(\Psi_i < L_{i-}) &= P(e^{t_i \Psi_i} < e^{t_i L_{i-}}) \\ &< e^{t_i L_{i-}} E(e^{-t_i \Psi_i}) \\ &= e^{t_i L_{i-}} \prod_{j=1}^{r} E(e^{-t_i c_{ij} q_j}) \\ &= e^{t_i L_{i-}} \prod_{j=1}^{r} (p_j e^{-t_i c_{ij}} + 1 - p_j). \end{aligned} \tag{4.7}$$

Again, we must consider how this probability changes when certain q_k are assigned values. If $q_k = 1$, then

$$\begin{aligned} P(\Psi_i < L_{i-}) &< e^{t(L_{i-} - c_{ik})} \prod_{j \neq k} E(e^{-t_i c_{ij} q_j}) \\ &= e^{t(L_{i-} - c_{ik})} \prod_{j \neq k} (p_j e^{-t_i c_{ij}} + 1 - p_j). \end{aligned} \tag{4.8}$$

If $q_k = 0$, then

$$\begin{aligned} P(\Psi_i < L_{i-}) &< e^{t L_{i-}} \prod_{j \neq k} E(e^{-t_i c_{ij} q_j}) \\ &= e^{t L_{i-}} \prod_{j \neq k} (p_j e^{-t_i c_{ij}} + 1 - p_j). \end{aligned} \tag{4.9}$$

The probability that $\Psi_i < L_{i-}$ for some i is bounded by the sum

$$\sum_{i=1}^{n} e^{t_i L_{i-}} \prod_{j=1}^{r} (p_j e^{-t_i c_{ij}} + 1 - p_j). \tag{4.10}$$

We can do similar analysis for the upper bound to get

$$P(\Psi_i > L_{i+}) < e^{-t_i L_{i+}} \prod_{j=1}^{r} (p_j e^{t_i c_{ij}} + 1 - p_j). \tag{4.11}$$

Thus, for our pessimistic estimator, we combine (4.7), (4.10), and (4.11) to get

$$\begin{aligned} U &= \sum_{i=1}^{n} U_i \\ &= \sum_{i=1}^{n} \left[e^{-t_i L_{i+}} \prod_{j=1}^{r} (p_j e^{t_i c_{ij}} + 1 - p_j) + e^{t_i L_{i-}} \prod_{j=1}^{r} (p_j e^{-t_i c_{ij}} + 1 - p_j) \right]. \end{aligned} \tag{4.12}$$

This is the value of U when no q_j are assigned values. When some are assigned, we incorporate the substitutions made in (4.8) and (4.9).

Recall that $L_{i+} = s_i(1 + D(s_i, \frac{1}{2}n))$. If $t_i = \ln[1 + D(s_i, \frac{1}{2n})]$, then the left and right hand products of the U_i are bounded by $B(s_i, D(s_i, \frac{1}{2n}))$ in a manner similar to the proof from **Theorem 4.1**. Thus, recalling **Theorem 4.5**, $U < 1$, so U is an appropriate upper bound for showing existence and applying the method of conditional probabilities.

Suppose $q_1, \ldots, q_j$ are already assigned values. Then (4.12) becomes

$$U(q_1, \ldots, q_j) = \sum_{i=1}^{n} [B_i(p_{j+1}e^{t_i c_{i(j+1)}} + 1 - p_{j+1}) + C_i(p_{j+1}e^{-t_i c_{i(j+1)}} + 1 - p_{j+1})]$$

$$= \sum_{i=1}^{n} [p_{j+1}(B_i e^{t_i c_{i(j+1)}} + C_i e^{-t_i c_{i(j+1)}}) + (1 - p_{j+1})(B_i + C_i)], \tag{4.13}$$

where B_i and C_i are constants that are the products of

(1) the e^{tL} coefficients,
(2) the substitutions made in (4.8) and (4.9) for selections of $q_{<j+1}$,
(3) and the expectations for $q_{>j+1}$.

If $q_{j+1} = 1$, (4.13) becomes

$$\sum_{i=1}^{n} (B_i e^{t_i c_{i(j+1)}} + C_i e^{-t_1 c_{i(j+1)}}). \tag{4.14}$$

If $q_{j+1} = 0$, (4.13) becomes

$$\sum_{i=1}^{n} (B_i + C_i). \tag{4.15}$$

Since (4.13) is a convex combination of (4.14) and (4.15), it is at least the minimum of the latter two, so U decreases if we assign q_{j+1} appropriately.

U satisfies both conditions for use as a pessimistic operator, so applying the method of conditional probabilities gives us the desired algorithm.

The calculations for U_i have constant bound (including calculations for s_i and $D(s_i, \frac{1}{2n})$, recalling *Remark 4.4*). In U, there are n such U_i to be calculated. We do these calculations r times. Thus, the algorithm is polynomial in n and r. □

Note that **Theorem 4.6** makes use of a non-symmetric probability space in addition to requiring a pessimistic estimator, unlike the simple examples in **Theorem 3.2** and **Theorem 2.1**. [6] has additional examples involving the bound from **Theorem 4.1**.

5. Combinatorial Games

In a sense, we may consider the process of trying to obtain a certain structure with given properties as a single player combinatorial game for which the method of conditional probabilities is a winning strategy. Accordingly, the method can be adapted for use in two player adversial combinatorial games.

Let $\mathcal{H} = ([m], \{A_1, \ldots, A_n\})$ be a hypergraph.

Theorem 5.1. *If $\sum_{i=1}^{n} 2^{1-|A_i|} < 1$, then there exists a vertex two-coloring for $\mathcal{H}$ with no monochromatic edges.*

Proof. Color the vertices independently at random. The probability that any A_i is monochromatic is $2^{1-|A_i|}$, so the expected number of monochromatic edges is $\sum_{i=1}^{n} 2^{1-|A_i|}$. Since the expectation is less than one, there exists a coloring with no monochromatic edges. □

Theorem 5.2. *There exists a deterministic algorithm polynomial in n and m for finding such a coloring.*

The proof follows **Theorem 3.2** closely.

Proof. Consider a partial coloring where the vertices $1, \ldots, j$ are already colored. Suppose the vertices are colored independently at random. Let $P_j(i)$ be the probability that A_i is monochromatic.

If A_i has at least one vertex of each color, then $P_j(i) = 0$. If A_i has no vertices that are colored, then $P_j(i) = 2^{1-|A_i|}$. Otherwise, $P_j(i) = 2^{x_i - |A_i|}$, where x_i is the number of vertices A_i has of one color. The expected number E_j of monochromatic A_i is $\sum_{i=1}^{n} P_j$.

If the two possible colorings for $j+1$ are red and blue, then

$$E_j = \frac{E_{j+1}(\text{If } j+1 \text{ is red}) + E_{j+1}(\text{If } j+1 \text{ is blue})}{2}.$$

Thus, there exists a choice of color for $j+1$ that decreases the expectation.

Following the method of conditional probabilities gives us the desired algorithm.

For each A_i, the calculation of $P_j(i)$ has constant bound. There are n such A_i. We do these calculations at most 2 times for each of the m vertices. Thus, the algorithm is polynomial in n and m. □

Now, we examine an analogous combinatorial game from [3].

Consider the positional game where two players, Maker and Breaker, construct a two-coloring for $\mathcal{H}$. They each have a color, and they alternate turns assigning vertices to their respective colors. Maker goes first, and she wins by making an edge monochromatic in her color. Breaker wins by preventing Maker from winning.

This is a generalization of tic-tac-toe. For a further generalization, where Maker chooses p vertices and Breaker chooses q vertices, see [2].

Theorem 5.3. *If $\sum_{i=1}^{n} 2^{1-|A_i|} < 1$, then Breaker has a winning strategy.*

Proof. Consider a partial coloring after an arbitrary number of moves are made. Suppose the remaining unpicked vertices are randomly assigned the players' colors with equal probability. Let P_i be the probability that any given A_i is monochromatic with Maker's color.

If Breaker has already picked any vertex from A_i, then $P_i = 0$. If he has not, then $P_i = 2^{x_i - |A_i|}$, where x_i is the number of vertices Maker has already picked from A_i. The expected number of monochromatic edges is $\sum_{i=1}^{n} P_i$.

If Maker ever makes a winning move, then there exists an A_k such that $P_k = 1$, so $\sum_{i=1}^{n} P_i \geq 1$. Thus, it suffices to show that Breaker can prevent Maker from ever making a move that makes the expectation at least one.

Consider the game state at the beginning of the game.

After Maker makes any first move, she has picked at most 1 vertex from each edge, so $\sum_{i=1}^{n} P_i < \sum_{i=1}^{n} 2^{1-|A_i|} < 1$.

If Breaker chooses some vertex v for a subsequent move, then P_i for all $A_i \ni v$ decreases to 0, and the other probabilities are unaffected. Thus, the maximum Breaker can decrease the expectation on his move is $\sum_{i \in J} P_i$ for some $J \subseteq [n]$.

Similarly, if Maker chooses some vertex w for a subsequent move, then P_i for $A_i \ni w$ doubles, and the other probabilities are unaffected. Thus, $\sum_{i \in J} P_i$ is also the maximum Maker can increase the expectation.

Therefore if Breaker always plays to minimize the expectation, Maker can never raise the expectation more than Breaker lowers it, so the expectation never rises above one, and Maker can never win. □

Just as the algorithm in **Corollary 5.2** relies on keeping the expectation of monochromatic edges of either color low, the strategy Breaker employs in **Theorem 5.3** relies on keeping the expectation of monochromatic edges in Maker's color low.

Acknowledgments

I would like to thank my mentor Dylan Quintana for guiding me throughout the process of writing the paper. I would also like to thank Peter May for directing the UChicago REU program and providing this opportunity.

References

[1] N. Alon, J.H. Spencer, *The Probabilistic Method*, 3rd. ed., John Wiley & Sons, Inc., 2008.

[2] J. Beck, "Remarks on positional games. I," *Acta Math. Acad. Sci. Hungar.*, 40 (1–2), 1982, pp.65-71. (https://link.springer.com/content/pdf/10.1007/BF01897304.pdf)

[3] P. Erdős, J.L. Selfridge, "On a combinatorial game," *Journal of Combinatorial Theory*, Series A, Volume 14, Issue 3, 1973, pp. 298-301, ISSN 0097-3165, https://doi.org/10.1016/0097-3165(73)90005-8 (http://www.sciencedirect.com/science/article/pii/0097316573900058)

[4] P. Erdős, J.H. Spencer, *Probabilistic Methods in Combinatorics*, Academic Press, 1974.

[5] P. Erdős, L. Moser, "A Problem on Tournaments," *Canadian Math Bulletin*, 7, 1964, pp. 351-356. (http://combinatorica.hu/~p_erdos/1964-03.pdf)

[6] P. Raghavan, "Probabilistic construction of deterministic algorithms: Approximating packing integer programs," 27th Annual Symposium on Foundations of Computer Science (sfcs 1986)(FOCS), vol. 00, no., 1986, pp. 10-18. doi:10.1109/SFCS.1986.45