

# Parimutuel Betting on Permutations

Shipra Agrawal \*

Zizhuo Wang <sup>†</sup>

Yinyu Ye <sup>‡§</sup>

## Abstract

We focus on a permutation betting market under parimutuel call auction model where traders bet on the final ranking of  $n$  candidates. We present a *Proportional Betting* mechanism for this market. Our mechanism allows the traders to bet on any subset of the  $n^2$  ‘candidate-rank’ pairs, and rewards them proportionally to the number of pairs that appear in the final outcome. We show that market organizer’s decision problem for this mechanism can be formulated as a convex program of polynomial size. More importantly, the formulation yields a set of  $n^2$  *unique* marginal prices that are sufficient to price the bets in this mechanism, and are computable in polynomial-time. The marginal prices reflect the traders’ beliefs about the marginal distributions over outcomes. We also propose techniques to compute the joint distribution over  $n!$  permutations from these marginal distributions. We show that using a maximum entropy criterion, we can obtain a concise parametric form (with only  $n^2$  parameters) for the joint distribution which is defined over an exponentially large state space. We then present an approximation algorithm for computing the parameters of this distribution. In fact, the algorithm addresses the generic problem of finding the maximum entropy distribution over permutations that has a given mean, and may be of independent interest.

## 1 Introduction

Prediction markets are increasingly used as an information aggregation device in academic research and public policy discussions. The fact

that traders must “put their money where their mouth is” when they say things via markets helps to collect information. To take full advantage of this feature, however, we should ask markets the questions that would most inform our decisions, and encourage traders to say as many kinds of things as possible, so that a big picture can emerge from many pieces. Combinatorial betting markets hold great promise on this front. Here, the prices of contracts tied to the events have been shown to reflect the traders’ belief about the probability of events. Thus, the pricing or ranking of possible outcomes in a combinatorial market is an important research topic.

We consider a permutation betting scenario where traders submit bids on final ranking of  $n$  candidates, for example, an election or a horse race. The possible outcomes are the  $n!$  possible orderings among the candidates, and hence there are  $2^{n!}$  subset of events to bid on. In order to aggregate information about the probability distribution over the entire outcome space, one would like to allow bets on all these event combinations. However, such betting mechanisms are not only intractable, but also exacerbate the thin market problems by dividing participants attention among an exponential number of outcomes [6, 11]. Thus, there is a need for betting languages or mechanisms that could restrict the possible bid types to a tractable subset and at the same time provide substantial information about the traders’ beliefs.

### 1.1 Previous Work

Previous work on parimutuel combinatorial markets can be categorized under two types of mechanisms: a) posted price mechanisms including the Logarithmic Market Scoring Rule (LMSR) of Hanson [11, 10] and the Dynamic Pari-mutuel Market-Maker (DPM) of Pennock [16] b) call auction models developed by Lange and Economides [13], Peters et al. [17], in which all the

\*Department of Computer Science, Stanford University. Email: shipra@stanford.edu

<sup>†</sup>Department of Management Science and Engineering, Stanford University. Email: zzwang@stanford.edu

<sup>‡</sup>Department of Management Science and Engineering, and, Institute of Computational and Mathematical Engineering (ICME), Stanford University. Email: yinyu-ye@stanford.edu

<sup>§</sup>Research supported in part by NSF DMS-0604513.

orders are collected and processed together at once. An extension of the call auction mechanism to a dynamic setting similar to the posted price mechanisms, and a comparison between these models can be found in Peters et al. [18].

Chen et al. (2008) [4] analyze the computational complexity of market maker pricing algorithms for combinatorial prediction markets under LMSR model. They examine both permutation combinatorics, where outcomes are permutations of objects, and Boolean combinatorics, where outcomes are combinations of binary events. Even with severely limited languages, they find that LMSR pricing is #P-hard, even when the same language admits polynomial-time matching without the market maker. Chen, Goel, and Pennock [3] study a special case of Boolean combinatorics and provide a polynomial-time algorithm for LMSR pricing in this setting based on a Bayesian network representation of prices. They also show that LMSR pricing is NP-hard for a more general bidding language.

More closely related to our work are the studies by Fortnow et al. [8] and Chen et al. (2006) [5] on *call auction* combinatorial betting markets. Fortnow et al. [8] study the computational complexity of finding acceptable trades among a set of bids in a Boolean combinatorial market. Chen et al. (2006) [5] analyze the auctioneer’s matching problem for betting on permutations, examining two bidding languages. Subset bets are bets of the form candidate  $i$  finishes in positions  $x, y$ , or  $z$  or candidate  $i, j$ , or  $k$  finishes in position  $x$ . Pair bets are of the form candidate  $i$  beats candidate  $j$ . They give a polynomial-time algorithm for matching divisible subset bets, but show that matching pair bets is NP-hard.

## 1.2 Our Contribution

This paper extends the above-mentioned work in a variety of ways. We propose a more generalized betting language called *Proportional Betting* that encompasses *Subset Betting* [5] as a special case. Further, we believe that ours is the first result on *pricing* a parimutuel call auction under permutation betting scenario.

In our proportional betting mechanism, the traders bet on one or more of the  $n^2$  ‘candidate-position’ pairs, and receive rewards proportional

to the number of pairs that appear in the final outcome. For example, a trader may place an order of the form “Horse A will finish in position 2 OR Horse B will finish in position 4”. He <sup>1</sup> will receive a reward of \$2 if both Horse A & Horse B finish at the specified positions 2 & 4 respectively; and a reward of \$1 if only one horse finishes at the position specified. The market organizer collects all the orders and then decides which orders to accept in order to maximize his worst case profit.

In particular, we will present the following results:

- We show that the market organizer’s decision problem for this mechanism can be formulated as a convex program with only  $O(n^2 + m)$  variables and constraints, where  $m$  is the number of bidders.
- We show that we can obtain, in polynomial-time, a small set ( $n^2$ ) of ‘marginal prices’ that satisfy the desired price consistency constraints, and are sufficient to price the bets in this mechanism.
- We show that by introducing non-zero starting orders, our mechanism will produce unique marginal prices.
- We suggest a maximum entropy criteria to obtain a maximum-entropy joint distribution over the  $n!$  outcomes from the marginal prices. Although defined over an exponential space, this distribution has a concise parametric form involving only  $n^2$  parameters. Moreover, it is shown to agree with the maximum-likelihood distribution when prices are interpreted as observed statistics from the traders’ beliefs.
- We present an approximation algorithm to compute the parameters of the maximum entropy joint distribution to any given accuracy in (pseudo)-polynomial time <sup>2</sup>. In fact, this algorithm can be directly applied to the generic problem of finding the maximum entropy distribution over permutations that has a given expected value, and may be of independent interest.

<sup>1</sup>‘he’ shall stand for ‘he or she’

<sup>2</sup>The approximation factors and running time will be established precisely in the text.

## 2 Background

In this section, we briefly describe the Convex Parimutuel Call Auction Model (CPCAM) developed by Peters et al. [17] that will form the basis of our betting mechanism.

Consider a market with one organizer and  $m$  traders or bidders. There are  $S$  states of the world in the future on which the traders are submitting bids. For each bid that is accepted by the organizer and contains the realized future state, the organizer will pay the bidder some fixed amount of money, which is assumed to be \$1 without loss of generality. The organizer collects all the bids and decides which bids to accept in order to maximize his worst case profit.

Let  $a_{ik} \in \{0, 1\}$  denote the trader  $k$ 's bid for state  $i$ . Let  $q_k$  and  $\pi_k$  denote the limit quantity and limit price for trader  $k$ , i.e., trader  $k$ 's maximum number of orders requested and maximum price for the bid, respectively. The number of orders accepted for trader  $k$  is denoted by  $x_k$ , and  $p_i$  denotes the price computed for outcome state  $i$ .  $x_k$  is allowed to take fractional values, that is, the orders are 'divisible' in the terminology of [5]. Below is the convex formulation of the market organizer's problem given by [17]:

$$\begin{aligned} \max_{x, s, r} \quad & \pi^T x - r + \mu \sum_{i=1}^S \theta_i \log(s_i) \\ \text{s. t.} \quad & \sum_k a_{ik} x_k + s_i = r \quad 1 \leq i \leq S \\ & 0 \leq x \leq q \\ & s \geq 0 \end{aligned} \tag{1}$$

The 'parimutuel' price vector  $\{p_i\}_{i=1}^S$  is given by the dual variables associated with the first set of constraints. The parimutuel property implies that when the bidders are charged a price of  $\{\sum_i a_{ik} p_i\}$ , instead of their limit price, the payouts made to the bidders are exactly funded by the money collected from the accepted orders in the worst-case outcome.  $\theta > 0$  represents starting orders needed to guarantee uniqueness of these state prices in the solution.  $\mu > 0$  is the weight given to the starting order term.

The significance of starting orders needs a special mention here. Without the starting orders, (1) would be a linear program with multiple dual solutions. Introducing the convex barrier term involving  $\theta$  makes the dual strictly convex resulting in a unique optimal price vector. To understand its effect on the computed prices, consider

the dual problem for (1):

$$\begin{aligned} \min_{y, p} \quad & q^T y - \mu \sum_{i=1}^S \theta_i \log(p_i) \\ \text{s. t.} \quad & \sum_i p_i = 1 \\ & \sum_i a_{ik} p_i + y_k \geq \pi_k \quad \forall k \\ & y \geq 0 \end{aligned}$$

Observe that if  $\theta$  is normalized, the second term in the objective gives the K-L distance<sup>3</sup> of  $\theta$  from  $p$  (less a constant term  $\sum \theta_i \log \theta_i$ ). Thus, when  $\mu$  is small, the above program optimizes the first term  $q^T y$ , and among all these optimal price vectors picks the one that minimizes the K-L distance of  $p$  from  $\theta$ . As discussed in the introduction, the price vectors are of special interest due to their interpretation as outcome distributions. Thus, the starting orders enable us to choose the unique distribution  $p$  that is closest (minimum K-L distance) to a prior specified through  $\theta$ .

The CPCAM model shares many desirable properties with the limit order parimutuel call auction model originally developed by Lange and Economides [13]. Some of its important properties from information aggregation perspective are 1) it produces a self-funded auction, 2) it creates more liquidity by allowing multi-lateral order matching, 3) the prices generated satisfy "price consistency constraints", that is, the market organizer agrees to accept the orders with a limit price greater than the calculated price of the order while rejecting any order with a lower limit price. The price consistency constraints ensure the traders that their orders are being duly considered by the market organizer, and provide incentive for informed traders to trade whenever their information would change the price. Furthermore, it is valuable that the model has a unique optimum and produces a unique price vector.

Although the above model has many powerful properties, its call auction setting suffers from the drawback of a delayed decision. The traders are not sure about the acceptance of their orders until after the market is closed. Also, it is difficult to determine the optimal bidding strategy for the traders and ensure truthfulness. In

<sup>3</sup> The Kullback Leibler distance (KL-distance) is a measure of the difference between two probability distributions. The K-L distance of a distribution  $p$  from  $\theta$  is given by  $\sum_i \theta_i \log \frac{\theta_i}{p_i}$ .

a consecutive work, Peters et al. [18] introduced a “Sequential Convex Parimutuel Mechanism (SCPM)” which is an extension of the CPCAM model to a dynamic setting, and has additional properties of immediate decision and truthfulness in a myopic sense. The techniques discussed in this paper assume a call auction setting, but can be directly applied to this sequential extension.

### 3 Permutation Betting Mechanisms

In this section, we propose new mechanisms for betting on permutations under the parimutuel call auction model described above. Consider a permutation betting scenario with  $n$  candidates. Traders bet on rankings of the candidates in the final outcome. The final outcome is represented by an  $n \times n$  permutation matrix, where  $ij^{th}$  entry of the matrix is 1 if the candidate  $i$  takes position  $j$  in the final outcome and 0 otherwise. We propose betting mechanisms that restrict the admissible bet types to ‘set of candidate-position pairs’. Thus, the trader  $k$ ’s bet will be specified by an  $n \times n$   $(0, 1)$  matrix  $A_k$ , with 1 in the entries corresponding to the candidate-position pairs he is bidding on. We will refer to this matrix as the ‘bidding matrix’ of the trader. If the trader’s bid is accepted, he will receive some payout in the event that his bid is a “winning bid”.

Depending on how this payout is determined, two variations of this mechanism are examined: a) Fixed Reward Betting and b) Proportional Betting. The intractability of fixed reward betting will provide motivation to examine proportional betting more closely, which is the focus of this paper.

**Fixed reward betting** In this mechanism, a trader receives a fixed payout (assume \$1 w.l.o.g.) if *any* entry in his bidding matrix matches with the corresponding entry in the outcome permutation matrix. That is, if  $M$  is the outcome permutation matrix, then the payout made to trader  $k$  is given by  $I(A_k \bullet M > 0)$ . Here, the operator ‘ $\bullet$ ’ denotes the Frobenius

inner product<sup>4</sup>, and  $I(\cdot)$  denotes an indicator function. The market organizer must decide which bids to accept in order to maximize the worst case profit. Using the same notations as in the CPCAM model described in Section 2 for limit price, limit quantities, and accepted orders, the problem for the market organizer in this mechanism can be formulated as follows:

$$\begin{aligned} \max \quad & \pi^T x - r \\ \text{s. t.} \quad & r \geq \sum_{k=1}^m I(A_k \bullet M_\sigma > 0) x_k \quad \forall \sigma \in \mathcal{S}_n \\ & 0 \leq x \leq q \end{aligned} \tag{2}$$

Here,  $\mathcal{S}_n$  represents the set of  $n$  dimensional permutations,  $M_\sigma$  represents the permutation matrix corresponding to permutation  $\sigma$ . Note that this formulation encodes the problem of maximizing the worst-case profit of the organizer with no starting orders.

Above is a linear program with exponential number of constraints. We prove the following theorem regarding the complexity of solving this linear program.

**Theorem 3.1.** *The optimization problem in (2) is NP-hard even for the case when there are only two non-zero entries in each bidding matrix.*

*Proof.* The separation problem for the linear program in (2) corresponds to finding the permutation that “satisfies” maximum number of bidders. Here, an outcome permutation is said to “satisfy” a bidder, if his bidding matrix has at least one coincident entry with the permutation matrix. We show that the separation problem is NP-hard using a reduction from maximum satisfiability (MAX-SAT) problem. In this reduction, the clauses in the MAX-SAT instance will be mapped to bidders in the bidding problem. And, the number of non-zero entries in a bidding matrix will be equal to the number of variables in the corresponding clause. Since, MAX-2-SAT is NP-hard, this reduction will prove the NP-hardness even for the case when each bidding matrix is restricted to have only two non-zero

<sup>4</sup> The Frobenius inner product, denoted as  $A \bullet B$  in this paper, is the component-wise inner product of two matrices as though they are vectors. That is,

$$A \bullet B = \sum_{i,j} A_{ij} B_{ij}$$

entries. See the appendix for the complete reduction.

Using the result on equivalence of separation and optimization problem from [9], the theorem follows.  $\square$

This result motivates us to examine the following variation of this mechanism which makes payouts proportional to the number of winning entries in the bidding matrix.

**Proportional betting** In this mechanism, the trader receives a fixed payout (assume \$1 w.l.o.g.) *for each coincident entry* between the bidding matrix  $A_k$  and the outcome permutation matrix. Thus, the payoff of a trader is given by the Frobenius inner product of his bidding matrix and the outcome permutation matrix. The problem for the market organizer in this mechanism can be formulated as follows:

$$\begin{aligned} \max \quad & \pi^T x - r \\ \text{s. t.} \quad & r \geq \sum_{k=1}^m (A_k \bullet M_\sigma) x_k \quad \forall \sigma \in \mathcal{S}_n \\ & 0 \leq x \leq q \end{aligned} \quad (3)$$

The above linear program involves exponential number of constraints. However, the separation problem for this program is polynomial-time solvable, since it corresponds to finding the maximum weight matching in a complete bipartite graph, where weights of the edges are given by elements of the matrix  $(\sum_k A_k x_k)$ . Thus, the ellipsoid method with this separating oracle would give a polynomial-time algorithm for solving this problem. This approach is similar to the algorithm proposed in [5] for *Subset Betting*. Indeed, for the case of subset betting [5], the two mechanisms proposed here are equivalent. This is because subset betting can be equivalently formulated under our framework, as a mechanism that allows non-zero entries only on a single row or column of the bidding matrix  $A_k$ . Hence, the number of entries that are coincident with the outcome permutation matrix can be either 0 or 1, resulting in  $I(A_k \bullet M_\sigma > 0) = A_k \bullet M_\sigma$ , for all permutations  $\sigma$ . Thus, subset betting forms a special case of the proportional betting mechanism proposed here, and all the techniques derived in the sequel for proportional betting will directly apply to it.

## 4 Pricing in Proportional Betting

In this section, we reformulate the market organizer's problem for Proportional Betting into a compact linear program involving only  $O(n^2 + m)$  constraints. Not only the new formulation is faster to solve in practice (using interior point methods), but also it will generate a compact dual price vector of size  $n^2$ . These 'marginal prices' will be sufficient to price the bets in Proportional Betting, and are shown to satisfy some useful properties. The reformulation will also allow introducing  $n^2$  starting orders in order to obtain unique prices.

Observe that the first constraint in (3) implicitly sets  $r$  as the worst case payoff over all possible permutations (or matchings). Since the matching polytope is integral [9],  $r$  can be equivalently set as the result of following linear program that computes maximum weight matching:

$$\begin{aligned} r = \max_M \quad & (\sum_{k=1}^m x_k A_k) \bullet M \\ \text{s.t.} \quad & M^T e = e \\ & M e = e \\ & M_{ij} \geq 0 \quad 1 \leq i, j \leq n \end{aligned}$$

Here  $e$  denotes the vector of all 1s (column vector). Taking dual, equivalently,

$$\begin{aligned} r = \min_{v, w} \quad & e^T v + e^T w \\ \text{s.t.} \quad & v_i + w_j \geq \sum_{k=1}^m (x_k A_k)_{ij} \quad \forall i, j \end{aligned}$$

Here,  $(x_k A_k)_{ij}$  denotes the  $ij^{th}$  element of the matrix  $(x_k A_k)$ . The market organizer's problem in (3) can now be formulated as:

$$\begin{aligned} \max_{x, v, w} \quad & \pi^T x - e^T v - e^T w \\ \text{s.t.} \quad & v_i + w_j \geq \sum_{k=1}^m (x_k A_k)_{ij} \quad \forall i, j \\ & 0 \leq x \leq q \end{aligned} \quad (4)$$

Observe that this problem involves only  $n^2 + 2m$  constraints. As we show later, the  $n^2$  dual variables for the first set of constraints can be well interpreted as marginal prices. However, the dual solutions for this problem are not guaranteed to be unique. To ensure uniqueness, we can use starting orders as discussed for the CPCAM model in Section 2. After introducing one starting order  $\theta_{ij} > 0$  for each candidate-position

pair, and slack variables  $s_{ij}$  for each of the  $n^2$  constraints, we get the following problem:

$$\begin{aligned} \max_{x,v,w,s} \quad & \pi^T x - e^T v - e^T w + \sum_{i,j} \theta_{ij} \log(s_{ij}) \\ \text{s.t.} \quad & v_i + w_j - s_{ij} = \sum_{k=1}^m (x_k A_k)_{ij} \quad \forall i, j \\ & s_{ij} \geq 0 \quad \forall i, j \\ & 0 \leq x \leq q \end{aligned} \quad (5)$$

and its dual:

$$\begin{aligned} \min_{y,Q} \quad & q^T y - \sum_{i,j} \theta_{ij} \log(Q_{ij}) \\ \text{s.t.} \quad & Qe = e \\ & Q^T e = e \\ & A_k \bullet Q + y_k \geq \pi_k \quad \forall k \\ & y \geq 0 \end{aligned} \quad (6)$$

Next, we will show that model (5) and (6) possess many desirable characteristics.

**Lemma 4.1.** *Model (5) and (6) are convex programs. And if  $\theta_{ij} > 0, \forall i, j$ , the solution to (6) is unique in  $Q$ .*

*Proof.* Since logarithmic function is concave and the constraints are linear, we can easily verify that (5) and (6) are convex programs. Also, according to our assumption on  $\theta$ , the objective function in (6) is strictly convex in  $Q$ . Thus, the optimal solution of (6) must be unique in  $Q$ .  $\square$

Therefore, we know that this program can be solved up to any given accuracy in polynomial time using convex programming methods and produces unique dual solution  $Q$ .

We show that the dual matrix  $Q$  generated from (6) is well interpreted as a “parimutuel price”. That is,  $Q \geq 0$ ; and, if we charge each trader  $k$  a price of  $A_k \bullet Q$  instead of their limit price ( $\pi_k$ ), then the optimal decision remains unchanged and the total premium paid by the accepted orders will be equal to the total payout made in the worst case. Further, we will show that  $Q$  satisfies the following extended definition of “price consistency condition” introduced in [13].

**Definition 4.2.** *The price matrix  $Q$  satisfies the price consistency constraints if and only if for all  $j$ :*

$$\begin{aligned} x_j = 0 & \Rightarrow Q \bullet A_j = c_j \geq \pi_j \\ 0 < x_j < q_j & \Rightarrow Q \bullet A_j = c_j = \pi_j \\ x_j = q_j & \Rightarrow Q \bullet A_j = c_j \leq \pi_j \end{aligned}$$

*That is, a trader’s bid is accepted only if his limit price is greater than the calculated price for the order.*

To see this, we construct the Lagrangian function for program (5):

$$\begin{aligned} L(x, Q, s, v, w, y) &= \pi^T x - e^T v - e^T w + \sum_{i,j} \theta_{ij} \log s_{ij} \\ &- \sum_{i,j} Q_{ij} (s_{ij} + \sum_k (x_k A_k)_{ij} - v_i - w_j) \\ &+ \sum_i y_i (q_i - x_i) \end{aligned}$$

Now, we can derive the KKT conditions:

$$\begin{aligned} \pi_k - Q \bullet A_k - y_k &\leq 0 & 1 \leq k \leq m \\ x_k \cdot (\pi_k - Q \bullet A_k - y_k) &= 0 & 1 \leq k \leq m \\ Qe &= e \\ Q^T e &= e \\ \frac{\theta_{ij}}{s_{ij}} - Q_{ij} &\leq 0 & 1 \leq i, j \leq n \\ s_{ij} \cdot (\frac{\theta_{ij}}{s_{ij}} - Q_{ij}) &= 0 & 1 \leq i, j \leq n \\ y_k \cdot (x_k - q_k) &= 0 & 1 \leq k \leq m \\ y &\geq 0 \end{aligned}$$

Since  $s_{ij} > 0$  for any optimal solution, the above conditions imply that  $Q_{ij} = \frac{\theta_{ij}}{s_{ij}}$ , or  $s_{ij} = \frac{\theta_{ij}}{Q_{ij}}$  for all  $ij$ . Since,  $\theta_{ij} > 0$ , this implies  $Q_{ij} > 0$ , for all  $ij$ . Also, the first constraint in the primal problem (5) now gives:  $v_i + w_j = \sum_k (x_k A_k)_{ij} + \frac{\theta_{ij}}{Q_{ij}}$ . Multiplying with  $Q_{ij}$ , and summing over all  $i, j$ :

$$r = e^T v + e^T w = \sum_k x_k (A_k \bullet Q) + \sum_{i,j} \theta_{ij}$$

Since,  $r$  gives the worst case payoff, charging the bidders according to price matrix  $Q$  results in a parimutuel market (except for the amount invested in the starting orders, an issue that we address later). Also, if we replace  $\pi_k$  with  $A_k \bullet Q$  in the above KKT conditions and set  $y_k = 0$ , the solution  $x, s, Q$  will still satisfy all the KKT conditions. Thus, the optimal solution remains unchanged. Further, observe that the first two conditions along with the penultimate one are exactly the price consistency constraints. Hence,  $Q$  must satisfy the price consistency constraints.

In the above model, market organizer needs to seed the market with the starting orders  $\theta_{ij}$  in order to ensure uniqueness of the optimum state price matrix. The market organizer could actually lose this seed money in some outcomes. In practice, we can set the starting orders to be very

small so that this is not an issue. On the other hand, it is natural to ask whether the starting orders can be removed altogether from the model to make the market absolutely parimutuel. The following lemma shows that this is indeed possible.

**Lemma 4.3.** *For any given starting orders  $\theta$ , as we reduce  $\theta$  uniformly to 0, the price matrix converges to a unique limit  $Q$ , and this limit is an optimal dual price for the model without the starting orders given in (4).*

*Proof.* The proof of this lemma follows directly from the discussion in Section 3.1 of [17].  $\square$

Moreover, as discussed in [17], such a limit  $Q$  can be computed efficiently using the path-following algorithm developed in [21].

To summarize, we have shown that:

**Theorem 4.4.** *One can compute in polynomial-time, an  $n \times n$  marginal price matrix  $Q$  which is sufficient to price the bets in the Proportional Betting mechanism. Further, the price matrix is unique, parimutuel, and satisfies the desired price-consistency constraints.*

## 5 Pricing the Outcome Permutations

There is analytical as well as empirical evidence that prediction market prices provide useful estimates of average beliefs about the probability that an event occurs [1, 14, 15, 20]. Therefore, prices associated with contracts are typically treated as predictions of the probability of future events. The marginal price matrix  $Q$  derived in the previous section associates a price to each candidate-position pair. Also, observe that  $Q$  is a doubly-stochastic matrix (refer to the constraints in dual problem (6)). Thus, the distributions given by a row (column) of  $Q$  could be interpreted as marginal distribution over positions for a given candidate (candidates for a given position). One would like to compute the complete price vector that assigns a price to each of the  $n!$  outcome permutations. This price vector would provide information regarding the joint probability distribution over the entire outcome space. In this section, we discuss methods for computing this complete price vector from the marginal prices given by  $Q$ .

Let  $p_\sigma$  denote the price for permutation  $\sigma$ . Then, the constraints on the price vector  $p$  are represented as:

$$\begin{aligned} \sum_{\sigma \in \mathcal{S}_n} p_\sigma M_\sigma &= Q \\ p_\sigma &\geq 0 \quad \forall \sigma \in \mathcal{S}_n \end{aligned} \quad (7)$$

Note that the above constraints implicitly impose the constraint  $\sum_{\sigma} p_\sigma = 1$ . Thus,  $\{p_\sigma\}$  is a valid distribution. Also, it is easy to establish that if  $Q$  is an optimal marginal price matrix, then any such  $\{p_\sigma\}$  is an optimal joint price vector over permutations. That is,

**Lemma 5.1.** *If  $Q$  is an optimal dual solution for (4), then any price vector  $\{p_\sigma\}$  that satisfies the constraints in (7) is an optimal dual solution for (3).*

*Proof.* The result follows directly from the structure of the two dual problems. See appendix for a detailed proof.  $\square$

Finding a feasible solution under these constraints is equivalent to finding a decomposition of doubly-stochastic matrix  $Q$  into a convex combination of  $n \times n$  permutation matrices. There are multiple such decompositions possible. For example, one such solution can be obtained using Birkhoff-von Neumann decomposition [2, 7]. Next, we propose a criterion to choose a meaningful distribution  $p$  from the set of distributions satisfying constraints in (7).

### 5.1 Maximum entropy criterion

Intuitively, we would like to use all the information about the marginal distributions that we have, but avoid including any information that we do not have. This intuition is captured by the ‘Principle of Maximum Entropy’. It states that the least biased distribution that encodes certain given information is that which maximizes the information entropy.

Therefore, we consider the problem of finding the maximum entropy distribution over the space of  $n$  dimensional permutations, satisfying the above constraints on the marginal distributions. The problem can be represented as follows:

$$\begin{aligned} \min \quad & \sum_{\sigma \in \mathcal{S}_n} p_\sigma \log p_\sigma \\ \text{s.t.} \quad & \sum_{\sigma \in \mathcal{S}_n} p_\sigma M_\sigma = Q \\ & p_\sigma \geq 0 \end{aligned} \quad (8)$$

The maximum entropy distribution obtained from above has many nice properties. Firstly,

as we show next, the distribution has a concise representation in terms of only  $n^2$  parameters. This property is crucial for combinatorial betting due to the exponential state space over which the distribution is defined. Let  $Y \in R^{n \times n}$  be the Lagrangian dual variable corresponding to the marginal distribution constraints in (8), and  $s_\sigma$  be the dual variables corresponding to non-negativity constraints on  $p_\sigma$ . Then, the KKT conditions for (8) are given by:

$$\begin{aligned} \log(p_\sigma) + 1 - s_\sigma &= Y \bullet M_\sigma \\ \sum_\sigma p_\sigma M_\sigma &= Q \\ s_\sigma, p_\sigma &\geq 0 \quad \forall \sigma \\ p_\sigma s_\sigma &= 0 \quad \forall \sigma \end{aligned} \quad (9)$$

Assuming  $p_\sigma > 0$  for all  $\sigma$ , this gives  $p_\sigma = e^{Y \bullet M_\sigma - 1}$ . Thus, the distribution is completely specified by the  $n^2$  parameters given by  $Y$ . Once  $Y$  is known, it is possible to perform operations like computing the probability for a given set of outcomes, or sampling the highly probable outcomes.

Further, we show that the dual solution  $Y$  is a maximum likelihood estimator of distribution parameters under suitable interpretation of  $Q$ .

**Maximum likelihood interpretation** For a fixed set of data and an *assumed* underlying probability model, maximum likelihood estimation method picks the values of the model parameters that make the data “more likely” than any other values of the parameters would make them. Let us assume in our model that the traders’ beliefs about the outcome come from an exponential family of distributions  $D_\eta$ , with probability density function of the form  $f_\eta \propto e^{\eta \bullet M_\sigma}$  for some parameter  $\eta \in R^{n \times n}$ . Suppose  $Q$  gives a summary statistics of  $s$  sample observations  $\{M^1, M^2, \dots, M^s\}$  from the traders’ beliefs, i.e.,  $Q = \frac{1}{s} \sum_k M^k$ . This assumption is inline with the interpretation of the prices in prediction markets as mean belief of the traders.

Then, the maximum likelihood estimator  $\hat{\eta}$  of  $\eta$  is the value that maximizes the likelihood of these observations, that is:

$$\begin{aligned} \hat{\eta} &= \arg \max_\eta \log f_\eta(M^1, M^2, \dots, M^s) \\ &= \arg \max_\eta \log \left( \prod_k \frac{e^{\eta \bullet M^k}}{\sum_\sigma e^{\eta \bullet M_\sigma}} \right) \end{aligned}$$

The optimality conditions for the above unconstrained convex program are:

$$\frac{1}{Z} \sum_\sigma e^{\eta \bullet M_\sigma} M_\sigma = \frac{1}{s} \sum_k M^k$$

where  $Z$  is the normalizing constant,  $Z = \sum_\sigma e^{\eta \bullet M_\sigma}$ . Since  $\frac{1}{s} \sum_k M^k = Q$ , observe from the KKT conditions for the maximum entropy model given in (9) that  $\eta = Y$  satisfies the above optimality conditions. Hence, the parameter  $Y$  computed from the maximum entropy model is also the maximum likelihood estimator for the model parameters  $\eta$ .

## 5.2 Complexity of the Maximum Entropy Model

In this section, we analyze the complexity of solving the maximum entropy model in (8). As shown in the previous section, the solution to this model is given by the parametric distribution  $p_\sigma = e^{Y \bullet M_\sigma - 1}$ . The parameters  $Y$  are the dual variables given by the optimal solution to the following dual problem of (8)

$$\max_Y Q \bullet Y - \sum_\sigma e^{Y \bullet M_\sigma - 1} \quad (10)$$

We prove the following result regarding the complexity of computing the parameters  $Y$ :

**Theorem 5.2.** *It is #P-hard to compute the parameters of the maximum entropy distribution  $\{p_\sigma\}$  over  $n$  dimensional permutations  $\sigma \in \mathcal{S}_n$ , that has a given marginal distribution.*

*Proof.* We make a reduction from the following problem:

**Permanent of a (0, 1) matrix** The permanent of an  $n \times n$  matrix  $B$  is defined as  $\text{perm}(B) = \sum_{\sigma \in \mathcal{S}_n} \prod_{i=1}^n B_{i, \sigma(i)}$ . Computing permanent of a (0, 1) matrix is #P-hard [19].

We use the observation that  $\sum_\sigma e^{Y \bullet M_\sigma} = \text{perm}(e^Y)$ , where the notation  $e^Y$  is used to mean component-wise exponentiation:  $(e^Y)_{ij} = e^{Y_{ij}}$ . For complete proof, see the appendix.  $\square$

Interestingly, there exists an FPTAS based on MCMC methods for computing the permanent of any non-negative matrix [12]. Next, we derive a polynomial-time algorithm for approximately computing the parameter  $Y$  that uses this FPTAS along with the ellipsoid method for optimization.

## 5.3 An Approximation Algorithm

In this section, we develop an approximation algorithm to compute the parameters  $Y$ . We first relax the formulation in (8) to get an equivalent problem that will lead to a better bounded dual.



Consider the problem below:

$$\begin{aligned} \min \quad & \sum p_\sigma (\log p_\sigma - 1) \\ \text{s.t.} \quad & \sum p_\sigma M_\sigma \leq Q \\ & p_\sigma \geq 0 \end{aligned} \quad (11)$$

We prove the following lemma:

**Lemma 5.3.** *The problem in (11) has the same optimal solution as (8).*

*Proof.* See the appendix.  $\square$

The Lagrangian dual of this problem is given by:

$$\begin{aligned} \max \quad & Q \bullet Y - \sum_\sigma e^{Y \bullet M_\sigma} \\ \text{s.t.} \quad & Y \leq 0 \end{aligned} \quad (12)$$

Note that  $Y$  is bounded from above. Next, we establish lower bounds on the variable  $Y$ . These bounds will be useful in proving polynomial running time for ellipsoid method.

**Lemma 5.4.** *The optimal value  $OPT$  and the optimal solution  $Y$  to (12) satisfy the following bounds:*

$$\begin{aligned} 0 &\geq OPT \geq -n \log n - 1 \\ 0 &\geq Y_{ij} \geq -n \log n / q_{\min} \quad \forall i, j \end{aligned}$$

Here,  $q_{\min} = \min\{Q_{ij}\}$ .

*Proof.* See the appendix.  $\square$

*Remark:* Note that if  $Q_{ij} = 0$  for any  $i, j$ , in a pre-processing step we could set the corresponding  $Y_{ij}$  to  $-\infty$  and remove it from the problem. So, w.l.o.g. we can assume  $q_{\min} > 0$ . However, some  $Q_{ij}$  could be very small, making the above bounds very large. One way to handle this would be to set very small  $Q_{ij}$ s (say less than  $\delta$  for some small  $\delta > 0$ ) to 0, and remove the corresponding  $Y_{ij}$ s from the problem. This would introduce a small additive approximation of  $\delta$  in the constraints of the problem, but ensure that  $q_{\min} > \delta$ .

From KKT conditions for the above problem, we obtain that  $p_\sigma = e^{Y \bullet M_\sigma}$  at optimality. Substituting  $p_\sigma$  into the primal constraints  $\sum p_\sigma = 1$  and  $\sum p_\sigma M_\sigma \leq Q$ , we can obtain the following equivalent dual problem with additional constraints:

$$\begin{aligned} \max \quad & Q \bullet Y - 1 \\ \text{s.t.} \quad & \sum e^{Y \bullet M_\sigma} M_\sigma \leq Q \\ & Y_{ij} \geq (-n \log n) / q_{\min} \quad \forall i, j \\ & Y_{ij} \leq 0 \quad \forall i, j \end{aligned} \quad (13)$$

The problem can be equivalently formulated as that of finding a feasible point in the convex body  $\mathbf{K}$  defined as:

$$\begin{aligned} \mathbf{K}: \quad & Q \bullet Y - 1 \geq t \\ & \sum e^{Y \bullet M_\sigma} M_\sigma \leq Q \\ & Y_{ij} \geq (-n \log n) / q_{\min} \quad \forall i, j \\ & Y_{ij} \leq 0 \quad \forall i, j \end{aligned}$$

Here,  $t$  is a fixed parameter. An optimal solution to (13) can be found by binary search on  $t \in [-n \log n - 1, 0]$ . We define an approximate set  $\mathbf{K}_\epsilon$  by modifying the RHS of second constraint in  $\mathbf{K}$  defined above to  $Q(1 + \epsilon)$ . Here,  $\epsilon$  is a fixed parameter.

Next, we show that the ellipsoid method can be used to generate  $(1 + \epsilon)$ -approximate solution  $Y$ . We will make use of the following lemma that bounds the gradient of the convex function  $f(Y) = \sum e^{Y \bullet M_\sigma} M_\sigma$  appearing in the constraints of the problem.

**Lemma 5.5.** *For any  $ij$ , the gradient of the function  $g(Y) = f_{ij}(Y) = \sum_\sigma e^{Y \bullet M_\sigma} (M_\sigma)_{ij}$  satisfies the following bounds:*

$$\|\nabla g(Y)\|_2 \leq n g(Y) \leq \sqrt{n} \|\nabla f(Y)\|_2$$

*Proof.* See the appendix.  $\square$

Now, we can obtain an approximate separating oracle for the ellipsoid method.

**Lemma 5.6.** *Given any  $Y \in R^{n \times n}$ , and any parameter  $\epsilon > 0$ , there exists an algorithm with running time polynomial in  $n$ ,  $1/\epsilon$  and  $1/q_{\min}$  that does one of the following:*

- asserts that  $Y \in \mathbf{K}_\epsilon$
- or, finds  $C \in R^{n \times n}$  such that  $C \bullet X \leq C \bullet Y$  for every  $X \in \mathbf{K}$ .

### Algorithm

1. If  $Y$  violates any constraints other than the constraint on  $f(Y)$ , report  $Y \notin K$ . The violated inequality gives the separating hyperplane.
2. Otherwise, compute a  $(1 \pm \delta)$ -approximation  $\hat{f}(Y)$  of  $f(Y)$ , where  $\delta = \min\{\frac{\epsilon}{12}, 1\}$ .
  - (a) If  $\hat{f}(Y) \leq (1 + 3\delta)Q$ , then report  $Y \in K_\epsilon$ .

- (b) Otherwise, say  $ij^{th}$  constraint is violated. Compute a  $(1 \pm \gamma)$ -approximation of the gradient of the function  $g(Y) = f_{ij}(Y)$ , where  $\gamma = \delta q_{min}/2n^4$ . The approximate gradient  $C = \hat{\nabla}g(Y)$  gives the desired separating hyperplane.

**Running time** Observe that  $f_{ij}(Y) = \text{perm}(e^{Y'_{ij}})$ , where  $Y'_{ij}$  denotes the matrix obtained from  $Y$  after removing the row  $i$  and column  $j$ . Thus,  $(1 \pm \delta)$  approximation to  $f(Y)$  can be obtained in time polynomial in  $n, 1/\delta$  using the FPTAS given in [12] for computing permanent of a non-negative matrix. Since,  $1/\delta$  is polynomial in  $n, 1/\epsilon, 1/q_{min}$ , this gives polynomial running time for estimating  $f(Y)$ . Similar observations hold for estimating the gradient  $\nabla f_{ij}(Y)$  in above.

**Correctness** The correctness of the above algorithm is established by the following two lemmas:

**Lemma 5.7.** *If  $\hat{f}(Y) \leq (1 + 3\delta)Q$  and all the other constraints are satisfied, then  $Y \in K_\epsilon$ .*

*Proof.* See the appendix.  $\square$

**Lemma 5.8.** *Suppose the  $ij^{th}$  constraint is violated, i.e.,  $\hat{f}_{ij}(Y) > (1 + 3\delta)Q_{ij}$ . Then,  $C = \hat{\nabla}f_{ij}(Y)$  gives a separating hyperplane for  $K$ , that is,  $C \bullet (X - Y) \leq 0, \forall X \in K$ .*

*Proof.* See the appendix. The proof uses the bounds on  $X, Y$  and  $\nabla f_{ij}(Y)$  established in Lemma 5.4 and Lemma 5.5, respectively.  $\square$

**Theorem 5.9.** *Using the separating oracle given by Lemma 5.6 with the ellipsoid method, a distribution  $\{p_\sigma\}$  over permutations can be constructed in time  $\text{poly}(n, \frac{1}{\epsilon}, \frac{1}{q_{min}})$ , such that*

- $(1 - \epsilon)Q \leq \sum_\sigma p_\sigma M_\sigma \leq Q$
- $p$  has close to maximum entropy, i.e.,  $\sum_\sigma p_\sigma \log p_\sigma \leq (1 - \epsilon)OPT_E$ , where  $OPT_E(\leq 0)$  is the optimal value of (8).

*Proof.* Using the above separating oracle with the ellipsoid method [9], after polynomial number of iterations we will either get a solution  $Y \in K_\epsilon(t)$ , or declare that there is no feasible solution. Thus, by binary search over the

$t$ , we can get a solution  $\bar{Y}$  such that  $\bar{Y} \in K_\epsilon$  and  $Q \bullet \bar{Y} - 1 \geq OPT$ . The dual solution thus obtained will have an objective value equal to or better than optimal but may be infeasible. We reduce each of the  $\bar{Y}_{ij}$ s by a small amount  $(\frac{1}{n} \log(1 + \epsilon))$  to construct a new feasible but sub-optimal solution  $\hat{Y}$ . Some simple algebraic manipulations show that the new solution  $\hat{Y}$  satisfies:  $(1 - \epsilon)Q \leq \sum_\sigma e^{\hat{Y} \bullet M_\sigma} M_\sigma \leq Q$ . Thus,  $\hat{Y}$  is a feasible solution to the dual, and,  $Q \bullet \hat{Y} - 1 \leq OPT$ . We can now construct the distribution  $p_\sigma$  as  $p_\sigma = e^{\hat{Y} \bullet M_\sigma}$ . Then from above,  $(1 - \epsilon)Q \leq \sum_\sigma p_\sigma M_\sigma \leq Q$ . Also,

$$\begin{aligned} \sum_\sigma p_\sigma \log p_\sigma &= \sum_\sigma e^{\hat{Y} \bullet M_\sigma} M_\sigma \bullet \hat{Y} \\ &\leq (1 - \epsilon)Q \bullet \hat{Y} \\ &\leq (1 - \epsilon)(OPT + 1) \\ &= (1 - \epsilon)OPT_E \end{aligned}$$

$\square$

## 6 Conclusion

We introduced a Proportional Betting mechanism for permutation betting which can be readily implemented by solving a convex program of polynomial size. More importantly, the mechanism was shown to admit an efficient parimutuel pricing scheme, wherein only  $n^2$  marginal prices were needed to price the bets. Further, we demonstrated that these marginal prices can be used to construct meaningful joint distributions over the exponential outcome space.

The proposed proportional betting mechanism was developed by relaxing a ‘fixed reward betting mechanism’. An interesting question raised by this work is whether the fixed betting mechanism could provide further information about the outcome distribution. Or, in general, how does the complexity of the betting language relates to the information collected from the market? A positive answer to this question would justify exploring approximation algorithms for the more complex fixed reward betting mechanism.

**Acknowledgements** We thank Arash Asadpour and Erick Delage for valuable insights and discussions.

## References

- [1] Christopher Adams. Does learning in prediction markets lead to information aggregation. *mimeo, Federal Trade Commission*, 2006.
- [2] G. Birkhoff. Three observations on linear algebra. *Univ. Nac. Tucuman Rev. A* 5, 1946.
- [3] Y. Chen, S. Goel, and D. M. Pennock. Pricing combinatorial markets for tournaments. *ACM Symposium on Theory of Computing*, 2008.
- [4] Yiling Chen, Lance Fortnow, Nicolas Lambert, David M. Pennock, and Jennifer Wortman. Complexity of combinatorial market makers. *CoRR*, abs/0802.1362, 2008.
- [5] Yiling Chen, Lance Fortnow, Evdokia Nikolova, and David M. Pennock. Betting on permutations. *EC '07: Proceedings of the 8th ACM conference on Electronic commerce*, 2007.
- [6] Yiling Chen, Lance Fortnow, Evdokia Nikolova, and David M. Pennock. Combinatorial betting. *SIGecom Exch.*, 7(1):61–64, 2007.
- [7] L. Dulmage and I. Halperin. On a theorem of Frobenius-König and J. von Neumann’s game of hide and seek. *Trans. Roy. Soc. Canada Sect. III* 49, 1955.
- [8] Lance Fortnow, Joe Kilian, David M. Pennock, and Michael P. Wellman. Betting boolean-style: a framework for trading in securities based on logical formulas. *EC '03: Proceedings of the 4th ACM conference on Electronic commerce*, 2003.
- [9] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, 1988.
- [10] R. D. Hanson. Logarithmic market scoring rules for modular combinatorial information aggregation. *Journal of Prediction Markets*, 2007.
- [11] Robin Hanson. Combinatorial information market design. *Information Systems Frontiers*, 5(1), 2003.
- [12] Mark Jerrum, Alistair Sinclair, and Eric Vigoda. A polynomial-time approximation algorithm for the permanent of a matrix with non-negative entries. *STOC '01: Proceedings of the thirty-third annual ACM symposium on Theory of computing*, 2001.
- [13] J. Lange and N. Economides. A parimutuel market microstructure for contingent claims. *European Financial Management*, 11(1), 2005.
- [14] Charles Manski. Interpreting the predictions of prediction markets. *Economic Letters*, 91(3), 2006.
- [15] Marco Ottaviani and Peter Norman Srensen. Aggregation of information and beliefs in prediction markets. *mimeo, London Business School*, 2006.
- [16] David M. Pennock. A dynamic pari-mutuel market for hedging, wagering, and information aggregation. *EC '04: Proceedings of the 5th ACM conference on Electronic commerce*, 2004.
- [17] Mark Peters, Anthony Man-Cho So, and Yinyu Ye. A convex parimutuel formulation for contingent claim markets. *Working Paper*, 2005. <http://www.stanford.edu/~yyye/cpcam-ec.pdf>.
- [18] Mark Peters, Anthony Man-Cho So, and Yinyu Ye. Pari-mutuel markets: Mechanisms and performance. *WINE*, 2007.
- [19] L.G. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 1979.
- [20] Justin Wolfers and Eric Zitzewitz. Interpreting prediction market prices as probabilities. Working Paper 12200, National Bureau of Economic Research, May 2006. <http://www.nber.org/papers/w12200>.
- [21] Y. Ye. A path to the Arrow-Debreu competitive market equilibrium. *Mathematical Programming*, 2005.

## APPENDIX

**Proof of Theorem 3.1** Consider the complete bipartite graph with the  $n$  candidates in one set and the  $n$  positions in the other set. In our betting mechanism, each bidder  $k$  bids on a subset of edges in this graph which is given by the non-zero entries in his bidding matrix  $A_k$ . A bidder is “satisfied” by a matching (or permutation) in this graph if at least one of the edges he bid on occurs in the matching. The separation problem for the linear program in (2) corresponds to finding the matching that satisfies the maximum number of bidders. Thus, it can be equivalently stated as the following matching problem:

*Matching problem:* Given a complete bipartite graph  $K_{n,n} = (V_1, V_2, E)$ , and a collection  $\mathcal{C} = \{E_1, E_2, \dots, E_m\}$  of  $m$  subsets of  $E$ . Find the perfect matching  $M \subset E$  that intersects with maximum number of subsets in  $\mathcal{C}$ .

*MAX-SAT problem:* Given a boolean formula in CNF form, determine an assignment of  $\{0, 1\}$  to the variables in the formula that satisfies the maximum number of clauses.

*Reduction from MAX-SAT to our matching problem:* Given the boolean formula in MAX-SAT problem with  $n$  variables  $x, y, z, \dots$ . Construct a complete bipartite graph  $K_{2n, 2n}$  as follows. For each variable  $x$ , add two nodes  $x$  and  $x'$  to the graph. And, for the possible values 0 and 1 of  $x$ , construct two nodes  $x_0$  and  $x_1$ . Connect by edges all the nodes corresponding to the variables to all the nodes corresponding to the values. Now, create the collection  $\mathcal{C}$  as follows. For  $k^{th}$  clause in the boolean formula, create a set  $E_k$  in  $\mathcal{C}$ . For each negated variable  $x$  in this clause, add edge  $(x, x_0)$  to  $E_k$ ; and for each non-negated variable  $x$  in the clause, add an edge  $(x, x_1)$  to  $E_k$ .

We show that every solution of size  $l$  for the MAX-SAT instance corresponds to a solution of size  $l$  for the constructed matching problem instance and vice-versa. Let there is an assignment that satisfies  $l$  clauses of MAX-SAT instance. Output a matching  $M$  in the graph  $K$  as follows. For each variable  $x$ , consider the nodes  $x, x', x_0, x_1$ . Let the variable  $x$  is assigned value

0 in the MAX-SAT solution. Then, add edges  $(x, x_0), (x', x_1)$  to  $M$ . Otherwise, add edges  $(x, x_1), (x', x_0)$  to  $M$ . It is easy to see that the resulting set  $M$  is a matching. Also, if a clause  $k$  satisfied in the MAX-SAT problem, then the matching  $M$  will have an edge common with  $E_k$ . Therefore  $M$  intersects with at least  $l$  subsets in  $\mathcal{C}$ .

Similarly, consider a solution  $M$  to the matching problem. Form a solution to the MAX-SAT problem as follows. Let the set  $E_k$  is satisfied (intersects with  $M$ ). Then, one of the edges in  $E_k$  must be present in  $M$ . Let  $(x, x_0)$  ( $(x, x_1)$ ) is such an edge. Then, assign 0 (1) to  $x$ . Because the  $M$  is a matching, any node  $x$  will have at the most one edge in  $M$  incident on it, and both  $(x, x_0)$  and  $(x, x_1)$  cannot be present in  $M$ . This ensures that takes  $x$  will take at the most one value 0 or 1 in the constructed assignment. For the remaining variables, assign values randomly. By construction, if a set  $E_k$  is satisfied in the matching solution, the corresponding  $k^{th}$  clause must be satisfied in the MAX-SAT problem - resulting in a solution of size at least  $l$  to MAX-SAT. This completes the reduction.

Note that in above, if we reduced from MAX-2-SAT, then each subset  $E_k$  would contain exactly two edges, that is, we would get an instance in which each bidder bids on exactly two candidate-position pairs. Because MAX-2-SAT is NP-hard, this proves that this special case is also NP-hard.

**Proof of Lemma 5.1** The dual for (4) is:

$$\begin{aligned} \min_{y, Q} \quad & q^T y \\ \text{s.t.} \quad & A_k \bullet Q + y_k \geq \pi_k \quad \forall k \\ & Qe = e \\ & Q^T e = e \\ & y \geq 0 \\ & Q \geq 0 \end{aligned} \tag{14}$$

The dual for (3) is:

$$\begin{aligned} \min_{y, p} \quad & q^T y \\ \text{s.t.} \quad & \sum_{\sigma} (A_k \bullet M_{\sigma}) p_{\sigma} + y_k \geq \pi_k \quad \forall k \\ & \sum_{\sigma} p_{\sigma} = 1 \\ & y \geq 0 \end{aligned} \tag{15}$$

Suppose  $p'_{\sigma}$  is a solution to (15), and  $\sum_{\sigma} p'_{\sigma} M_{\sigma} = Q'$ , then the first constraint in (15)

is equivalent to  $A_k \bullet Q' + y_k \geq \pi_k$ . Hence, for any solution  $p'_\sigma$  to (15), there is a corresponding solution  $Q' = \sum_\sigma p'_\sigma M_\sigma$  to (14) with the same objective value. Thus, if  $Q$  is an optimal solution to (14), then all  $\{p_\sigma\}$  satisfying  $\sum_\sigma p_\sigma M_\sigma = Q$  have the same objective value and are optimal.

**Proof of Lemma 5.2** The optimality condition for the dual problem in (10) is specified as (setting derivative to 0):

$$Q = \sum_\sigma e^{Y \bullet M_\sigma - 1} M_\sigma \quad (16)$$

Thus, given a certificate  $Y$ , verifying its optimality requires computing the function  $f(Y) = \sum e^{Y \bullet M_\sigma} M_\sigma$  for a given  $Y$ . Note that the  $ij^{th}$  component of this function is given by

$$e^{Y_{ij}} \sum_{\sigma: j=\sigma(i)} e^{Y \bullet M_\sigma} = e^{Y_{ij}} \text{perm}(e^{Y'})$$

where  $Y'$  is the matrix obtained from  $Y$  after removing row  $i$  and column  $j$ . We show that computing the permanent of  $e^{Y'}$  is #P-hard by reducing it to the problem of computing permanent of a  $(0, 1)$  matrix. The reduction uses the technique from the proof of Theorem 1 in [4]. We repeat the construction below for completeness. Suppose  $A$  is a  $(n-1) \times (n-1)$   $(0, 1)$  matrix whose permanent we wish to find. Then, construct a matrix  $Y'$  as follows:

$$Y'_{kl} = \begin{cases} \log(n! + 2) & A_{kl} = 1 \\ \log(n! + 1) & A_{kl} = 0 \end{cases}$$

Then,  $\text{perm}(e^{Y'}) \bmod (n! + 1) = \text{perm}(A) \bmod (n! + 1) = \text{perm}(A)$ , since  $\text{perm}(A) \leq n!$ . Hence, even the verification problem for this optimization problem is at least as hard as computing the permanent of a  $(0, 1)$ -matrix.

**Proof of Lemma 5.3** Observe that the problem in (8) involves implicit constraints  $\sum p_\sigma = 1$ . Further, we show that the equality constraints can be relaxed to inequality. We will show that it is impossible that  $\sum p_\sigma M_\sigma < Q$  for some elements in the optimal solution. Observe that the matrix  $Q - \sum p_\sigma M_\sigma$  has the property that each row and each column sums up to  $1 - \sum p_\sigma$ . That is,  $(Q - \sum p_\sigma M_\sigma)/(1 - \sum p_\sigma)$  is a doubly-stochastic matrix. *Birkhoff-von Neumann theorem* [2] proves that any doubly stochastic matrix can be represented as a convex combination of

permutation matrices. Since  $Q - \sum p_\sigma M_\sigma > 0$ , there must be at least one strictly positive coefficient in the Birkhoff-von Neumann decomposition of this matrix. This means that we can increase at least one  $p_\sigma$  a little bit without violating the inequality constraint. However, the derivative of the objective w.r.t one variable is  $\log p_\sigma$ . Therefore, when  $p_\sigma < 1$ , increasing  $p_\sigma$  will always decrease the objective value, which contradicts with the assumption that we have already reached the optimal. Thus we have shown that the problem in (11) shares the same optimal solution as (8).

**Proof of Lemma 5.4** Note that  $Y = -\log n \times \text{ones}(n, n)$  forms a feasible solution to (12). Hence, the optimal value to the dual must be greater than  $-n \log n - 1$ , that is,  $0 \geq OPT \geq -n \log n - 1$ . Also, from KKT conditions, the optimal solutions to the primal and dual are related as  $p_\sigma = e^{Y \bullet M_\sigma}$ . Hence, as discussed in proof of Lemma 5.3 for the primal solution, the optimal dual solution must satisfy  $\sum e^{Y \bullet M_\sigma} M_\sigma = Q$ , implicitly leading to  $\sum e^{Y \bullet M_\sigma} = 1$  at optimality. Along with the lower bound on  $OPT$ , this gives  $Q \bullet Y \geq -n \log n$ , which implies  $Y_{ij} \geq -n \log n / q_{min}$ .

**Proof of Lemma 5.5** The gradient of  $g(Y)$  is  $\nabla g(Y) = \sum_\sigma (e^{Y \bullet M_\sigma} M_{\sigma, ij}) M_\sigma$ . That is,  $\nabla g(Y)$  is an  $n \times n$  matrix defined as:

$$\nabla g(Y)_{k,l} = \begin{cases} \sum_{\sigma: j=\sigma(i)} e^{Y \bullet M_\sigma} & \text{if } (k,l)=(i,j) \\ \sum_{\sigma: j=\sigma(i), l=\sigma(k)} e^{Y \bullet M_\sigma} & \text{if } \{k,l\} \cap \{i,j\} = \emptyset \\ 0 & \text{o.w., if } \{k,l\} \cap \{i,j\} \neq \emptyset \end{cases}$$

We will use the notation  $e^Y$ , where  $Y$  is a matrix, to mean component-wise exponentiation:  $(e^Y)_{ij} = e^{Y_{ij}}$ . Let  $\chi$  denote the permanent of the non-negative matrix  $e^Y$ . Denote by  $\chi_{ij}$ , the “permanent” of the submatrix obtained after removing row  $i$  and column  $j$  from  $e^Y$ . Then, observe that  $g(Y) = e^{Y_{ij}} \cdot \chi_{ij}$ . Also, the gradient of  $g(Y)$  can be written as:

$$\nabla g(Y)_{k,l} = \begin{cases} e^{Y_{ij}} \cdot \chi_{ij} & \text{if } (k,l)=(i,j) \\ e^{Y_{ij}} e^{Y_{kl}} \cdot \chi_{ij,kl} & \text{if } \{k,l\} \cap \{i,j\} = \emptyset \\ 0 & \text{o.w., if } \{k,l\} \cap \{i,j\} \neq \emptyset \end{cases}$$

where  $\chi_{ij,kl}$  denotes the permanent of the matrix obtained after removing rows  $i, k$  and columns  $j, l$  from  $e^Y$ .

Using the relation between permanent of a matrix and its submatrices, observe that:

$$\begin{aligned} \|\nabla g(Y)\|_1 &= e^{Y_{ij}} \cdot \chi_{ij} + e^{Y_{ij}} \sum_{kl: ij \neq kl} e^{Y_{kl}} \cdot \chi_{ij,kl} \\ &= ne^{Y_{ij}} \chi_{ij} \\ &= ng(Y) \end{aligned}$$

Hence,

$$\|\nabla g(Y)\|_2 \leq ng(Y) \leq \sqrt{n} \|\nabla g(Y)\|_1$$

**Proof of Lemma 5.7** For any such  $Y$ ,

$$f(Y) \leq \frac{\hat{f}(Y)}{(1-\delta)} \leq \frac{(1+3\delta)}{(1-\delta)} Q \leq (1+12\delta)Q \leq (1+\epsilon)Q$$

**Proof of Lemma 5.8** Suppose the  $ij^{th}$  constraint is violated. That is,  $f_{ij}(Y) > (1+3\delta)Q_{ij}$ . This implies that  $f_{ij}(Y) > (1+\delta)Q_{ij}$ . This is because if  $f_{ij}(Y) \leq (1+\delta)Q_{ij}$ , then  $\hat{f}_{ij}(Y) \leq (1+\delta)f(Y) \leq (1+3\delta)Q_{ij}$ .

In below we denote the function  $f_{ij}(Y)$  by  $g(Y)$  and  $Q_{ij}$  by  $b$ . Given any  $X \in K$ , since  $g(\cdot)$  is a convex function,

$$\nabla g(Y)^T(X - Y) \leq g(X) - g(Y) \leq b - g(Y)$$

Therefore, using the bounds on  $X$  and  $Y$ ,

$$\begin{aligned} \hat{\nabla} g(Y)^T(X - Y) &\leq \nabla g(Y)^T(X - Y) + \|\nabla g(Y) - \hat{\nabla} g(Y)\| \cdot \|X - Y\| \\ &\leq b - g(Y) + \gamma \|\nabla g(Y)\| \frac{n^2 \log n}{q_{min}} \\ &\leq b - g(Y) + \gamma \cdot ng(Y) \cdot \frac{n^2 \log n}{q_{min}} \\ &\leq b - b(1+\delta)(1 - \gamma \frac{n^3 \log n}{q_{min}}) \end{aligned}$$

where the second last inequality follows from the bound on gradient given by Lemma 5.5. The last inequality follows from the observation made earlier that  $g(Y) = f_{ij}(Y) > 1 + \delta$ . Now,

$$\gamma = \frac{\delta q_{min}}{2n^4} \leq \frac{\delta}{1+\delta} \cdot \frac{q_{min}}{n^3 \log n}$$

Hence, from above,

$$\hat{\nabla} g(Y)^T(X - Y) \leq 0$$