

## PARALLEL MATCHING FOR RANKING ALL TEAMS IN A TOURNAMENT

KEI KOBAYASHI,\* *The Institute of Statistical Mathematics*

HIDEKI KAWASAKI\*\* \*\*\* AND

AKIMICHI TAKEMURA,\*\* \*\*\*\* *The University of Tokyo*

### Abstract

We propose a simple and efficient scheme for ranking all teams in a tournament where matches can be played simultaneously. We show that the distribution of the number of rounds of the proposed scheme can be derived using lattice path counting techniques used in ballot problems. We also discuss our method from the viewpoint of parallel sorting algorithms.

*Keywords:* Ballot problem; Bradley–Terry model; extreme value distribution; Hasse diagram; Kolmogorov–Smirnov statistic; lattice path counting; odd–even merge; parallel sorting; partially ordered set

2000 Mathematics Subject Classification: Primary 05A16

Secondary 60C05

### 1. Introduction

Consider a tournament-style matching of teams or players in sports such as football or tennis. The winner of the final match is naturally considered to be the strongest. Usually the loser of the final match is considered to be the second strongest. However, this might not be true, considering the possibility that the true second-strongest team might have been defeated by the strongest team at an earlier stage of the tournament. In this case, the true second-strongest team could not proceed further. In fact, any team defeated by the strongest team at some stage may be the second-strongest team. In order to determine the true second-strongest team, we have to arrange further matches between the teams defeated by the strongest team. If we want to determine the third-strongest team, the problem becomes more complicated. In this paper we propose a simple and efficient scheme for determining ranks of all teams in a tournament.

The scenario above describes a single-elimination, or knockout, tournament, where a team exits from the tournament once it is defeated by another team. The problem of a strong team being eliminated too early can be alleviated by double-elimination tournaments [6], [17], where a team exits from the tournament after two losses. However, double elimination does not directly address the problem of ranking.

The initial motivation for this work was the above simple question of how to determine the true ranking of teams or players in a tournament. However, the authors found that this problem

---

Received 6 May 2003; revision received 20 April 2006.

\* Postal address: The Institute of Statistical Mathematics, 4-6-7 Minami-Azabu, Minato-ku, Tokyo 106-8569, Japan.  
Email address: kei@ism.ac.jp

\*\* Postal address: Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan.

\*\*\* Email address: kawasaki@expm.t.u-tokyo.ac.jp

\*\*\*\* Email address: takemura@stat.t.u-tokyo.ac.jp

dates back to Lewis Carroll [4] and that it has close connections with lattice path counting in ballot problems, with the extreme value distribution, with partially ordered sets, and with the parallel sorting algorithms. The authors believe that, in addition to proposing a new scheme and deriving its properties, the present paper has merit in bringing together these rather separate fields in a problem of historical interest.

In many sports games, matches can be played simultaneously. We call each set of simultaneous matches a *round*. We use the number of rounds as a measure of the efficiency of a ranking scheme; other research has instead been on evaluating the number of matches in tournaments (see, for example, [5, p. 82]). We assume some probabilistic models to describe the result of each game and investigate the distribution of the number of rounds to determine the full ranking. It is combinatorially very difficult to obtain the optimal ranking scheme in the sense of minimizing the expected number of rounds. Although our scheme is not optimal, we will check numerically that the expected number of rounds in our scheme is close to optimal for a small number of teams.

We adopt the following notation and assumptions in this paper. ‘Match’ and ‘game’ are used synonymously (in context).

1. The set of teams is denoted by  $\mathcal{T} = \{t_1, \dots, t_N\}$ , where  $N$  is the number of teams. For example,  $\mathcal{T} = \{\text{Yankees, Blue Jays}, \dots\}$ . The ranking function is defined to be an injection  $r: \mathcal{T} \rightarrow \{1, \dots, N\}$  such that  $r(t_i) < r(t_j)$  if and only if  $t_i$  is (considered to be) stronger than  $t_j$ . The strongest team,  $t^*$ , satisfies  $r(t^*) = 1$ . Similarly, for a subset  $\mathcal{S} \subset \mathcal{T}$ , the relative rank of  $t_i \in \mathcal{S}$  in  $\mathcal{S}$  is denoted by  $r_{\mathcal{S}}(t_i)$ ,  $1 \leq r_{\mathcal{S}}(t_i) \leq \text{card } \mathcal{S}$ .
2. Each match is played between two teams. The result of each game is either win or lose: we assume that there are no ties. Each team plays at most one game in a round and each game between two teams is played at most once. We denote the match between  $t_i$  and  $t_j$  by  $[t_i \text{ vs } t_j]$ , or simply by  $[i \text{ vs } j]$ , and the set of matches by

$$\mathcal{G} = \{[t_i \text{ vs } t_j]: 1 \leq j < i \leq N\}.$$

Furthermore, let

$$w_{ij} = \begin{cases} 1 & \text{if } t_i \text{ wins against } t_j, \\ -1 & \text{if } t_j \text{ wins against } t_i. \end{cases}$$

Thus,  $w_{ij} = -w_{ji}$ .

3. Let  $\mathcal{G}_k \subset \mathcal{G}$  denote the set of matches belonging to the  $k$ th round. We do not impose any upper bound on the size of  $\mathcal{G}_k$ , and  $\lfloor N/2 \rfloor$  matches can be played simultaneously (where  $\lfloor \cdot \rfloor$  is the floor function). This corresponds to the situation where there is no restriction on the number of playing fields.
4.  $\mathcal{G}_k$  can depend on the results of  $\bigcup_{i=1}^{k-1} \mathcal{G}_i$ , the past rounds.
5. If team  $t_i$  wins against  $t_j$  in a match, the ranking of  $t_i$  must be higher than  $t_j$ . In other words,  $w_{ij} = 1 \Rightarrow r(t_i) < r(t_j)$ .
6. Two teams  $t_i$  and  $t_j$  should not play a match if their relative strength has already been determined by results of past rounds, i.e. if there exist teams  $t_{l_1}, \dots, t_{l_m}$  such that  $w_{i l_1} = w_{l_1 l_2} = \dots = w_{l_m j} = 1$  (or  $-1$ ) in past rounds.

The last three of these assumptions are special characteristics of our problem, which differ from those of the problem of parallel sorting algorithms. Since the outcome of a match may be

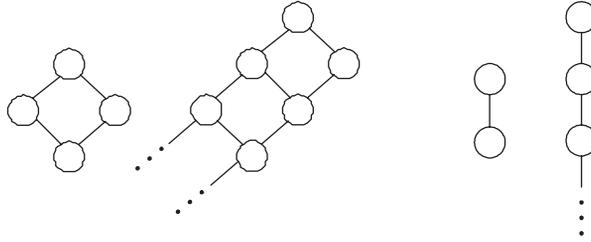


FIGURE 1: Isolated and connected diamonds (left) and line-segments (right).

random, a ranking scheme can lead to loops or contradictions. For example, if  $w_{ij} = w_{kl} = 1$  in past rounds, we should avoid simultaneous matches  $[i \text{ vs } l]$  and  $[j \text{ vs } k]$  in the current round, because if  $w_{il} = w_{kj} = -1$  then we have the contradiction

$$r(t_i) < r(t_j) < r(t_k) < r(t_l) < r(t_i).$$

Therefore, in designing a ranking scheme we have to consider not only the result of the past rounds but also all possible results of the future rounds. Our goal is to find a noncontradictory game scheduling scheme which requires only a small number of rounds to determine the full ranking.

Before proposing our ranking scheme, we consider the following subproblem.

*The parallel merge problem.* Suppose that there are two sets of teams,  $A = \{a_1, \dots, a_n\}$  and  $B = \{b_1, \dots, b_m\}$ , with  $n \leq m$ , and assume that the relative ranks within each set have been already determined to be  $r_A(a_i) = i, i = 1, \dots, n$ , and  $r_B(b_i) = i, i = 1, \dots, m$ . Determine the ranks of the teams in the union  $A \cup B$ .

We propose *equivalent-rank matching (ERM)* as a solution to this problem. For simplicity, we first assume that the sizes of the sets  $A$  and  $B$  are equal, i.e.  $n = m$ .

- *Step 1.* In the first round,  $a_i$  plays  $b_i$  simultaneously for each  $i = 1, \dots, n$ . As a result, a partial order is introduced on  $A \cup B$ . The partial order can be conveniently displayed by a diagram where we put a stronger team immediately above the teams next in the partial order. This diagram is called the Hasse diagram representing the partial order (see the next section for the precise definition of the Hasse diagram). As a result of the first round, the Hasse diagram becomes some combination of (isolated or connected) *diamonds* and *line-segments* (see Figure 1). Connected diamonds correspond to the winning streaks of teams from  $A$  or teams from  $B$ . The rankings of the parts of line-segments are already determined at this stage. Thus, if the Hasse diagram becomes a single line, the full ranking has been determined.
- *Step 2.* In the next round, only the parts of connected diamonds have to be considered. We match the teams corresponding to the horizontal nodes of each diamond in the Hasse diagram. Figure 2 shows an example of this process. In Figure 2, ‘W’ indicates the winner of the match between horizontal nodes. It is easy to see that the result of this round is also represented by a Hasse diagram consisting of line-segments and/or diamonds. The diamonds correspond to further winning streaks from the same set of teams as in the previous round.

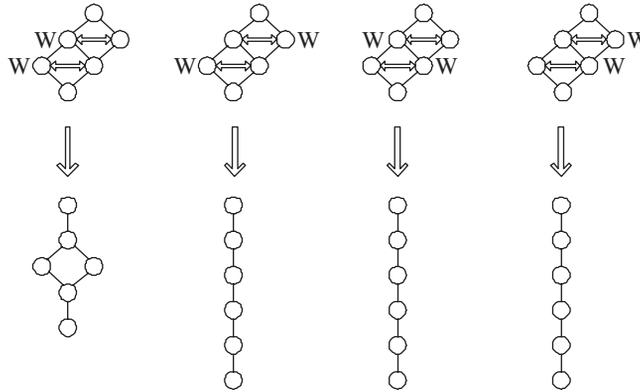


FIGURE 2: The process of step 2 of ERM.

- *Step 3.* If the Hasse diagram becomes a single line, the full ranking has been determined. Otherwise, go back to step 2.

This method terminates after at most  $n$  rounds, because the first round creates at most  $n - 1$  diamonds and the number of diamonds decreases in every round.

So far we have treated the case  $n = m$  for simplicity. If  $n < m$  then we add  $m - n$  dummy teams,  $a_{n+1}, \dots, a_m$ , which are defined to be weaker than  $b_m$ . We then apply the above merging scheme as for the case  $n = m$ .

With ERM, we can determine the full ranking of  $\mathcal{T}$  recursively, as follows. We call this method *parallel merge sort by equivalent-rank matching (PMS)*.

- *Step 1.* Randomly divide  $\mathcal{T}$  into two sets of teams such that the difference of their sizes is at most 1. Repeat these divisions until each set consists of just one team. This process is represented as a binary tree in which each leaf corresponds to a team (see the left-hand part of Figure 3). We call this binary tree the *merging tree*.
- *Step 2.* Merge the sets of the teams in the reverse order of step 1. Use ERM for each merging (see the right-hand part of Figure 3). Sets at the same horizontal level of the merging tree are merged in parallel and each merging requires a random number of rounds. We call the set of rounds needed to merge sets at the same horizontal level a *stage*. For example, the rounds at the bottom of the binary tree constitute the first stage and the rounds of the final merging of two sets constitute the last stage.

If we use PMS, it is easy to visualize and grasp the schedule and progress of the games. This is one of this method's benefits.

In this paper, we investigate the distribution of the number of rounds for merging using ERM. We assume two probabilistic models, the sure-winner model and the totally random model, defined in Section 2.2. Consider the problem of merging two groups of  $n$  teams. Let  $E_{n,n}^Q$  and  $E_{n,n}^P$  be the expected numbers of rounds using ERM in the sure-winner model and the totally random case, respectively. As one of our main results, in Sections 3 and 4 the asymptotic evaluation of  $E_{n,n}^Q$  and  $E_{n,n}^P$  will be presented: we find that

$$\lim_{n \rightarrow \infty} E_{n,n}^Q / \sqrt{n} = \sqrt{\pi} \log 2 \quad (\text{Theorem 3.3(b)})$$

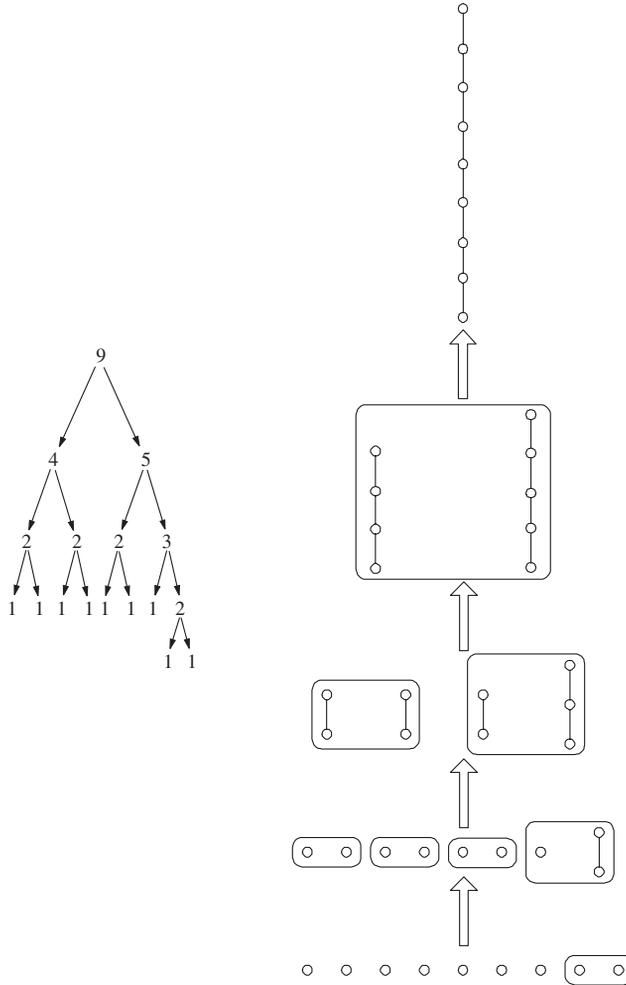


FIGURE 3: Step 1 of PMS (left) and step 2 of PMS (right), for  $N = 9$ .

and

$$\lim_{n \rightarrow \infty} E_{n,n}^P / \sqrt{\log n} = \sqrt{2/\log 2} \quad (\text{Theorem 4.3}).$$

The rest of the paper is organized as follows. In Section 2, basic notions of partially ordered sets and their Hasse diagrams are stated and a class of scheduling schemes including PMS is defined. In Section 3, the sure-winner probabilistic model is investigated. In Section 4, the totally random probabilistic model is investigated, with each match being a fair coin tossing. These two models are simple probabilistic models and the distribution of the number of rounds can be evaluated using the lattice path counting techniques used in ballot problems. We prove that there exists a stochastic order between these two models (Theorem 4.4). In Section 5, we discuss our scheme in view of the existing literature on parallel sorting algorithms. We also evaluate by simulation the distribution of the number of rounds in the case of a one-parameter Bradley–Terry model connecting the sure-winner model and the totally random model.

## 2. Preliminaries

In this section, we prepare some basic tools for investigating the behavior of PMS. First, we review basic notions of partially ordered sets and their Hasse diagrams. In a later part of this section, we propose a class of schemes for merging two sets of teams called *rectangle merge schemes*.

### 2.1. Partially ordered set and Hasse diagrams

Here we quote basic notions of partially ordered sets from Chapter 3 of [16]. A partially ordered set  $P$  is a set together with a binary relation ' $\leq$ ' satisfying the following three axioms: reflexivity ( $x \leq x$  for all  $x \in P$ ), antisymmetry ( $x \leq y, y \leq x \Rightarrow x = y$ ) and transitivity ( $x \leq y, y \leq z \Rightarrow x \leq z$ ). The notation  $x < y$  means that  $x \leq y$  and  $x \neq y$ . We say that two elements  $x$  and  $y$  of  $P$  are comparable if  $x \leq y$  or  $y \leq x$ ; otherwise,  $x$  and  $y$  are said to be incomparable. If every pair of elements of  $P$  is comparable,  $P$  is called a totally ordered set. If  $x, y \in P$  then we say that  $y$  covers  $x$  if and only if  $x < y$  and there is no element  $z \in P$  satisfying  $x < z < y$ . The Hasse diagram of a finite partially ordered set  $P$  is the graph whose vertices are the elements of  $P$ , whose edges are the cover relations, and in which  $y$  is drawn above  $x$  if  $x < y$ .

In our case,  $P = \mathcal{T}$  and a partial order is induced by matches between the teams. Note that in the Hasse diagram of the teams, we draw stronger teams higher. Since a stronger team has a smaller rank, the order in the above definition is conversely related to the ranks. Note that the teams  $t_i$  and  $t_j$  can be comparable, i.e. have known relative strength, even if they have not played each other in a match.

If team  $t_i$  covers  $t_j$  then it follows that  $t_i$  has defeated  $t_j$  in a match, but the converse is not necessarily true. Suppose that in past rounds both  $t_i$  and  $t_j$  have defeated  $t_k$  and that both cover  $t_k$  before the current round. If the match [ $i$  vs  $j$ ] is played in the current round and if  $w_{ij} = 1$ , then  $t_i$  no longer covers  $t_k$ .

As the rounds progress, the partial order becomes finer and finer until the full ranking is determined and  $\mathcal{T}$  becomes a totally ordered set.

The Hasse diagram is an effective way to display the game process. However, the Hasse diagrams appearing in most of the game scheduling schemes become much more complicated than those appearing in PMS. If PMS is used, all Hasse diagrams during the game consist of line-segments and diamonds only.

### 2.2. Probabilistic models

In the next two sections, we investigate the following two probabilistic models:

- *The sure-winner model.* Here the 'true' ranking is determined before the tournament. If team  $t_i$  is truly stronger than  $t_j$ , then  $t_i$  wins against  $t_j$  with probability 1. The randomness comes from the random assignment of  $N$  teams to the leaves of the merging tree. All permutations of the teams are equally likely.
- *The totally random model.* The result of each match is independently and identically sampled from a Bernoulli distribution with success probability  $\frac{1}{2}$ .

In the sure-winner model, the final ranking 'determined' by the results of the games is of course equal to the true ranking. In the totally random model, however, the final ranking is determined only by the random result of the matches. The Bradley–Terry model, which is a model intermediate between the sure-winner model and the totally random model, seems to be more realistic than these two models. However, theoretical investigation of the Bradley–Terry

model in our case seems to be difficult. We study the Bradley–Terry model by simulation in Section 5.

### 2.3. Rectangle merge scheme for the parallel merge problem

In this subsection, we consider a parallel merge problem of teams  $A = \{a_1, \dots, a_n\}$  and  $B = \{b_1, \dots, b_m\}$ . We assume that these two sets are already ordered, as follows:  $r_A(a_i) = i$ ,  $i = 1, \dots, n$ , and  $r_B(b_i) = i$ ,  $i = 1, \dots, m$ . Let  $\mathbb{I}_{n,m} = \{(i, j) : 1 \leq i \leq n, 1 \leq j \leq m\}$  be the index set of pairs consisting of a team in  $A$  and a team in  $B$ . We construct an  $n \times m$  matrix  $X = (x_{ij})$  whose element  $x_{ij}$  represents the round of the match between  $a_i$  and  $b_j$ . For example, if  $x_{12} = 2$  then the match between  $a_1$  and  $b_2$  is scheduled for the second round. Note that the match between  $a_i$  and  $b_j$  might or might not be played; we use  $x_{ij}$  to represent the round in the case that the match between  $a_i$  and  $b_j$  has to be played. One of our strategies is to determine all elements of  $X$  before the matches. We call a game scheduling scheme of this class a *static rectangle scheme*. We have to impose some restriction on  $X$  to cause no contradictions. If we select all matches  $[i \text{ vs } j]$  in round  $k$  such that  $x_{ij} = k$  (by definition) and  $a_i$  and  $b_j$  are incomparable, then any set of results of round  $k$  must be noncontradictory. A matrix  $X$  satisfying this requirement is called an *adequate scheduling matrix*.

The conditions for the adequacy of  $X$  are different in the sure-winner and totally random models. If  $X$  is adequate in the totally random model, then it is adequate in the sure-winner model. However, the converse does not hold. The necessary and sufficient conditions for adequacy in the two models are as follows.

$X$  is an adequate scheduling matrix in the sure-winner model



$$\begin{aligned} &\text{for all } (i_1, j), (i_2, j) \in \mathbb{I}_{n,m} \text{ such that } i_1 < i_2 \text{ and } x_{i_1 j} = x_{i_2 j}, \\ &\qquad\qquad\qquad \text{there exists an } i \text{ such that } i_1 < i < i_2 \text{ and } x_{ij} < x_{i_1 j}, \\ &\qquad\qquad\qquad \text{and} \end{aligned} \tag{2.1}$$

$$\begin{aligned} &\text{for all } (i, j_1), (i, j_2) \in \mathbb{I}_{n,m} \text{ such that } j_1 < j_2 \text{ and } x_{ij_1} = x_{ij_2}, \\ &\qquad\qquad\qquad \text{there exists a } j \text{ such that } j_1 < j < j_2 \text{ and } x_{ij} < x_{ij_1}. \end{aligned}$$

$X$  is an adequate scheduling matrix in the totally random model



$$\begin{aligned} &\text{for all } (i_1, j_1), (i_2, j_2) \in \mathbb{I}_{n,m} \text{ such that } i_1 \leq i_2, j_2 \leq j_1, (i_1, j_1) \neq (i_2, j_2), \text{ and } x_{i_1 j_1} = x_{i_2 j_2}, \\ &\qquad\qquad\qquad \text{there exists a pair } (i, j) \in \mathbb{I}_{n,m} \text{ such that } i_1 \leq i \leq i_2, j_2 \leq j \leq j_1, \text{ and } x_{ij} < x_{i_1 j_1}. \end{aligned} \tag{2.2}$$

Proofs of (2.1) and (2.2) are given in the preprint version [11] of this paper, which is available from the authors. The bitonic merge discussed in Section 5.1 satisfies (2.1) but not (2.2).

Static rectangle schemes determine all elements of  $X$  before the matches occur. A more general scheduling method is to determine the matches of the  $k$ th round depending on the result of the past rounds. We call methods of this class *dynamic rectangle schemes*. Every noncontradictory merging scheme can be specified as a dynamic rectangle scheme satisfying the above conditions in each round. However, a dynamic rectangle scheme is not represented as a single matrix, while the static rectangle scheme is. Therefore, it is very difficult to understand the process of games visually using dynamic rectangle schemes.

We define  $\check{X} = (\check{x}_{ij})$ , where  $\check{x}_{ij} = x_{j,m-i+1}$ , i.e.

$$\check{X} = \begin{bmatrix} x_{1m} & x_{2m} & \cdots & x_{nm} \\ \vdots & \vdots & \ddots & \vdots \\ x_{12} & x_{22} & \cdots & x_{n2} \\ x_{11} & x_{21} & \cdots & x_{n1} \end{bmatrix}.$$

We use this matrix for notational consistency with lattice path counting techniques.

We consider a class of scheduling methods whose matrices  $\check{X}$  have the same values for the elements in each diagonal line of 45 degrees. We arrange the diagonal lines such that between any two lines whose values are the same there is at least one line in which the elements have a smaller value, for example

$$\begin{bmatrix} 5 & 2 & 4 & 3 & 1 & 4 \\ 2 & 4 & 3 & 1 & 4 & 2 \\ 4 & 3 & 1 & 4 & 2 & 3 \\ 3 & 1 & 4 & 2 & 3 & 4 \\ 1 & 4 & 2 & 3 & 4 & 5 \end{bmatrix}.$$

It is evident that methods of this class satisfy conditions (2.1) and (2.2). They are called *45 degrees methods*.

Furthermore, the matrices  $\check{X}$  when using ERM are of the form

$$\begin{bmatrix} 5 & 4 & 3 & 2 & 1 & 2 \\ 4 & 3 & 2 & 1 & 2 & 3 \\ 3 & 2 & 1 & 2 & 3 & 4 \\ 2 & 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 & 6 \end{bmatrix}. \tag{2.3}$$

Thus, the ERM scheme is a 45 degree method and satisfies conditions (2.1) and (2.2).

#### 2.4. Notation for merging trees and stages

Finally, we set up notation for the merging tree and its stages. Let  $T$  be a merging tree. Let  $J$  denote the number of stages of  $T$ ; then  $J + 1$  is the depth of the rooted tree  $T$ . At the  $j$ th stage,  $1 \leq j \leq J$ ,  $2k_j$  sets of teams  $A_{ji}$ ,  $i = 1, \dots, 2k_j$ , are merged into  $k_j$  sets in parallel. Let  $Y_{jl}$ ,  $l = 1, \dots, k_j$ , denote the number of rounds needed to merge the sets. Then the total number of rounds,  $Y_j$ , of the  $j$ th stage is  $Y_j = \max(Y_{j1}, \dots, Y_{jk_j})$ . The total number of rounds is

$$Y = Y_1 + \cdots + Y_J. \tag{2.4}$$

Note that, in the above definition of  $Y_j$ , all teams wait until all  $k_j$  mergings are finished at the  $j$ th stage. We call this *synchronous* tree merging. In terms of reducing the total number of rounds, synchronous tree merging is clearly not optimal. For example, as soon as  $A_{j1}$  and  $A_{j2}$  are merged and  $A_{j3}$  and  $A_{j4}$  are merged at the  $j$ th stage, we could instead begin merging  $A_{j1} \cup A_{j2}$  and  $A_{j3} \cup A_{j4}$  at the  $(j + 1)$ th stage. We call this scheme *asynchronous* tree merging. In the asynchronous case, the stages lose simultaneity and it becomes harder to grasp the progress of the games. Furthermore, the total number of rounds is only defined recursively. For example, with  $N = 8$  teams the total number of rounds for the asynchronous case is written as

$$Y = Y_{31} + \max(Y_{21} + \max(Y_{11}, Y_{12}), Y_{22} + \max(Y_{13}, Y_{14})).$$

In this paper we adopt synchronous tree merging, for simplicity.

### 3. Sure-winner model

In this section, we study the distribution of the number of rounds in the sure-winner probabilistic model. First we establish the basic independence of the numbers of rounds for different mergings, in Theorem 3.1. Then, in Theorem 3.2, we derive recursion formulae for evaluating the distribution function of the number of rounds in ERM when merging two sets of teams.

Let  $Y_{j1}, \dots, Y_{jk_j}$ ,  $j = 1, \dots, J$ , be as defined in Section 2.4. Consider any merging scheme of two sets of teams,  $A$  and  $B$ . We call the merging scheme *local* if the resulting ranking of  $A \cup B$  only depends on the outcomes of matches between teams of  $A$  and teams of  $B$ . We then have the following basic independence of the numbers of rounds in the sure-winner probabilistic model.

**Theorem 3.1.** *In the sure-winner model, for any local merging scheme of two sets of teams,  $Y_{jl}$ ,  $l = 1, \dots, k_j$ ,  $j = 1, \dots, J$ , are mutually independently distributed.*

*Proof.* We argue recursively from the last stage. At the last stage we merge two sets of teams of  $A$  and  $B$  of sizes  $n$  and  $m$ ,  $n + m = N$ . Because the assignment of the teams at the leaves of the binary tree is random, the sets of ranks of teams of  $A$  are equally likely and each set has the probability  $1/\binom{N}{n}$ . Given the set of ranks of  $A$ , the merging process for  $A$  depends only on the relative ranks of teams within  $A$ . Therefore, the (conditional) joint distribution of the rounds for merging  $A$  is the same as in the original problem with  $N$  replaced by  $n$ , and the joint distribution does not depend on the ranks of teams of  $A$  within  $\mathcal{T} = A \cup B$ . The same thing holds for  $B$ . Therefore, by induction the numbers of rounds are mutually independent.

By this independence, the distribution function of  $Y_j = \max(Y_{j1}, \dots, Y_{jk_j})$  is evaluated as

$$P(Y_j \leq y) = \prod_{l=1}^{k_j} P(Y_{jl} \leq y).$$

Furthermore,  $Y_1, \dots, Y_J$  are independent. Therefore, the distribution function of  $Y = Y_1 + \dots + Y_J$  can be evaluated if the distribution of  $Y_{jl}$  is known.

Now we consider the distribution of the number of rounds  $Y_{jl}$  in a parallel merge problem. First, we give a necessary and sufficient condition for identification of the ranking from the viewpoint of rectangle schemes. We use the notation  $\mathcal{T}$ ,  $A$ ,  $B$ ,  $t_i$ ,  $a_i$ , and  $b_j$  established in the previous sections. Here  $\mathcal{T} = A \cup B$  and the size of  $\mathcal{T}$  is  $n + m$ . Consider an  $n \times m$  square lattice whose nodes are denoted by  $(i, j)$ ,  $i = 0, 1, \dots, n$ ,  $j = 0, 1, \dots, m$ . We consider a path which starts from the origin and at the  $i$ th step translates to the right by the vector  $e_1 = (1, 0)$  if  $r^{-1}(i) \in A$ , i.e. if the  $i$ th strongest team in the true ranking belongs to  $A$ , and translates upward by the vector  $e_2 = (0, 1)$  if  $r^{-1}(i) \in B$ . Thus, each true ranking order corresponds to a path of length  $n + m$  from the origin to  $(m, n)$  along the edges of the square lattice. If we compare the square lattice and the matrix  $\check{X}$ , each square is regarded as a match. It is evident that the ranking is determined uniquely if and only if the matches at every ‘inner corner’ of the corresponding path have been played. Here the inner corner is defined to be the lower-right square when the path turns to the right () and the upper-left square when the path turns to the left (). For example, the path denoted by the thick line-segments in Figure 4 is identified by the matches  $\alpha \leftrightarrow x_{12}$ ,  $\beta \leftrightarrow x_{43}$ , and  $\gamma \leftrightarrow x_{55}$ . By comparing Figure 4 and the matrix  $\check{X}$  in (2.3), we see that  $x_{12} = x_{43} = 2$  and  $x_{55} = 1$ . Therefore, this path is identified in the second round.

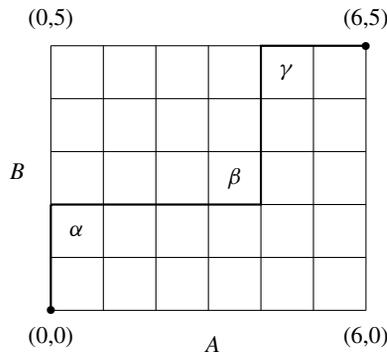


FIGURE 4: A path corresponding to a ‘true’ ranking order in the sure-winner model.

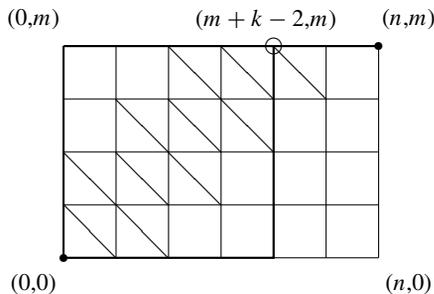


FIGURE 5: All paths must pass through  $(m + k - 2, m)$ . (In this example,  $n = 6, m = 4,$  and  $k = 2.$ )

Let  $Y(n, m)$  denote the number of rounds for the  $n \times m$  merge problem and let

$$Q_{n,m}(k) = P(Y(n, m) \leq k \mid \text{sure-winner model})$$

denote the distribution function of  $Y(n, m)$  in the sure-winner model. For  $n = 0$  or  $m = 0,$   $Q_{n,m}$  is formally defined as  $Q_{n,m}(k) = 1$  for all  $k \geq 0.$  The following theorem presents recurrence formulae for  $Q_{n,m}(k).$

**Theorem 3.2.** For all  $n, m \geq 1,$

$$\binom{n+m}{n} Q_{n,m}(k) = \begin{cases} \binom{2m+k-2}{m} Q_{m+k-2,m}(k) & \text{if } n \geq m+k, \\ \binom{2n+k-2}{n} Q_{n,n+k-2}(k) & \text{if } m \geq n+k, \\ \binom{n+m-1}{n} Q_{n,m-1}(k) + \binom{n+m-1}{m} Q_{n-1,m}(k) & \text{otherwise.} \end{cases}$$

*Proof.* We first prove the theorem for  $n \geq m + k.$  Figure 5 is an example of a square lattice in this case (with  $n = 6, m = 4,$  and  $k = 2).$  The squares with the diagonal  $(\square)$  correspond to the matches which may be played up to the  $k$ th round. Because  $n \geq m + k$  in this example,

the upper-right square of the lattice (corresponding to  $x_{nm}$ ) does not have a diagonal. The uppermost and rightmost square which has a diagonal is the one corresponding to  $x_{m+k-1,m}$ . Thus, all paths from the origin to  $(n, m)$  pass through the point  $(m + k - 2, m)$ . Moreover, all paths from the origin to  $(m + k - 2, m)$  can be extended to  $(n, m)$ . The number of such paths is  $\binom{2m+k-2}{m} Q_{m+k-2,m}(k)$ . Therefore,

$$\binom{n+m}{n} Q_{n,m}(k) = \binom{2m+k-2}{m} Q_{m+k-2,m}(k) \quad \text{if } n \geq m+k.$$

The equation for  $m \geq n+k$  follows by symmetry. In all other cases, the square corresponding to  $x_{nm}$  has a diagonal. In this situation, all paths from the origin to  $(n, m - 1)$  or  $(n - 1, m)$  can be extended to  $(n, m)$ . The number of such paths is

$$\binom{n+m}{n} Q_{n,m}(k) = \binom{n+m-1}{n} Q_{n,m-1}(k) + \binom{n+m-1}{m} Q_{n-1,m}(k).$$

Thus, we obtain Theorem 3.2.

To understand the distribution  $Q_{n,m}(k)$ , it is convenient to consider an easier problem in which paths can pass through all edges of squares with a diagonal. This problem is directly related to lattice path counting for the two-sample Kolmogorov–Smirnov statistic. Let  $\tilde{Y}(n, m)$  denote the number of rounds in this problem. This corresponds to the largest (horizontal or vertical) distance between the 45 degree line through the origin and the path. The following inequality holds between  $Y(n, m)$  and  $\tilde{Y}(n, m)$ :

$$\tilde{Y}(n, m) \leq Y(n, m) \leq \tilde{Y}(n, m) + 1. \tag{3.1}$$

Let  $\tilde{Q}_{n,m}(k)$  denote the distribution function of  $\tilde{Y}(n, m)$  and let  $E_{n,m}^{\tilde{Q}} = E[\tilde{Y}(n, m)]$  denote the expectation. The following results on the two-sample Kolmogorov–Smirnov statistic are well known (see, e.g. [10], [7], [8, pp. 342–343], and [14, pp. 101–108]).

**Lemma 3.1.** *We have*

$$\begin{aligned} \tilde{Q}_{n,m}(k) &= \binom{n+m}{n}^{-1} \sum_{j=-\infty}^{\infty} (-1)^j \binom{n+m}{n+j(k+1)}, \\ E_{n,m}^{\tilde{Q}} &= \binom{n+m}{n}^{-1} \sum_{k=0}^{\infty} \sum_{j \neq 0} (-1)^{j+1} \binom{n+m}{n+j(k+1)}. \end{aligned}$$

If  $m = n$  or  $m = n + 1$  then, for  $z > 0$ ,

$$\lim_{n \rightarrow \infty} \tilde{Q}_{n,m}(\sqrt{n}z) = 1 - 2 \sum_{j=1}^{\infty} (-1)^{j-1} \exp(-j^2 z^2).$$

From these results we can prove the following results.

**Lemma 3.2.** (a) *For any odd natural number  $J_1$  and even natural number  $J_2$ ,*

$$\binom{n+m}{n}^{-1} \sum_{j=1}^{J_2} (-1)^{j+1} \alpha_j \leq E_{n,m}^{\tilde{Q}} \leq \binom{n+m}{n}^{-1} \sum_{j=1}^{J_1} (-1)^{j+1} \alpha_j,$$

where

$$\alpha_j = j^{-1} \sum_{l=0}^{j-1} \omega_j^{-ln} (1 + \omega_j^l)^{n+m} - \binom{n+m}{n}$$

and  $\omega_j$  is a primitive  $j$ th root of unity. In particular,

$$\binom{n+m}{n}^{-1} 2^{n+m-1} \leq E_{n,m}^{\tilde{Q}} \leq \binom{n+m}{n}^{-1} 2^{n+m} - 1.$$

(b) If  $m = n$  or  $m = n + 1$  then

$$\lim_{n \rightarrow \infty} E_{n,m}^{\tilde{Q}} / \sqrt{n} = \sqrt{\pi} \log 2.$$

*Proof.* A proof of (a) is presented in the preprint [11] of this paper. Here we prove only (b). The asymptotic expectation under  $\tilde{Q}$  is evaluated as

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{E_{n,n}^{\tilde{Q}}}{\sqrt{n}} &= \lim_{n \rightarrow \infty} \sum_{k=0}^{\infty} \frac{1 - \tilde{Q}_{n,n}(k)}{\sqrt{n}} \\ &= 2 \sum_{j=1}^{\infty} (-1)^{j-1} \int_0^{\infty} \exp(-j^2 z^2) dz \\ &= \sqrt{\pi} \sum_{j=1}^{\infty} (-1)^{j-1} j^{-1} \\ &= \sqrt{\pi} \log 2. \end{aligned}$$

Here the interchanges of integrals and limits can be easily justified by dominated convergence and Fubini’s theorem. The result is the same for  $m = n + 1$ .

Recall that  $E_{n,m}^Q = E[Y(n, m)]$  denotes the expectation of  $Y(n, m)$  in the sure-winner model. By (3.1), inequalities for  $E_{n,m}^Q$ , the asymptotic distribution of the number of rounds for ERM, and the asymptotic expectations  $E_{n,n}^Q$  and  $E_{n,n+1}^Q$  are evaluated as follows.

**Theorem 3.3.** (a) We have

$$\binom{n+m}{n}^{-1} 2^{n+m-1} \leq E_{n,m}^Q \leq \binom{n+m}{n}^{-1} 2^{n+m}.$$

(b) If  $m = n$  or  $m = n + 1$  then

$$\begin{aligned} \lim_{n \rightarrow \infty} Q_{n,m}(\sqrt{n}z) &= 1 - 2 \sum_{j=1}^{\infty} (-1)^{j-1} \exp(-j^2 z^2), \quad z > 0, \\ \lim_{n \rightarrow \infty} E_{n,m}^Q / \sqrt{n} &= \sqrt{\pi} \log 2. \end{aligned}$$

### 4. Totally random model

In this section, we evaluate the number of rounds in the totally random model with ERM. The result of each match is independently and identically sampled from a Bernoulli distribution with success probability  $\frac{1}{2}$ .

First, as in the sure-winner model, we state the following basic independence of the numbers of rounds in the totally random model.

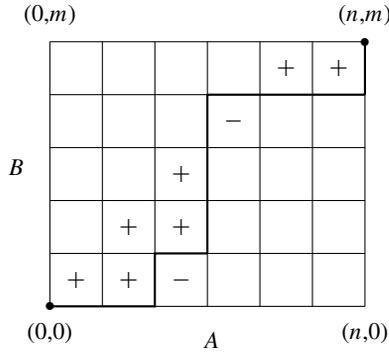


FIGURE 6: The games played before until a path is identified. ('+' indicates that A wins and '-' indicates that B wins.)

**Theorem 4.1.** *In the totally random model, for any local merging scheme of two sets of teams,  $Y_{jl}, l = 1, \dots, k_j, j = 1, \dots, J$ , are mutually independently distributed.*

*Proof.* In the totally random model, the result of each match is sampled independently from a Bernoulli distribution. Therefore, the result of the theorem is evident.

Now we concentrate on square lattices, as in the previous section. In the sure-winner model, a uniform probability is assigned to each path. However, in the totally random case, the probabilities are different. If  $k$  matches are necessary to identify the path, then the probability assigned to the path is  $2^{-k}$ , because the other matches are not played under ERM. In Figure 6, nine games are needed to identify the path denoted by the thick line-segments. According to (2.3), in that example  $k = 3$ .

Let  $P_{n,m}(k)$  denote the distribution function of the number of rounds when using ERM in the totally random model:

$$P_{n,m}(k) = P(Y(n, m) \leq k \mid \text{totally random model}).$$

The following theorem presents recurrence formulae for  $P_{n,m}(k)$ .

**Theorem 4.2.** *For all  $n, m \geq 1$ ,*

$$P_{n,m}(k) = \begin{cases} P_{m+k-1,m}(k) - 2^{-k} P_{m+k-1,m-1}(k) & \text{if } m + k \leq n, & (4.1) \\ P_{n-1,m}(k) - 2^{-n+m-1} P_{n-1,m-1}(k) + 2^{-n+m-1} P_{n,m-1}(k) & \text{if } m < n < m + k, & (4.2) \\ 2^{-1} P_{n-1,m}(k) + 2^{-1} P_{n,m-1}(k) & \text{if } n = m, & (4.3) \\ P_{n,m-1}(k) - 2^{n-m-1} P_{n-1,m-1}(k) + 2^{n-m-1} P_{n-1,m}(k) & \text{if } n < m < n + k, \\ P_{n,n+k-1}(k) - 2^{-k} P_{n-1,n+k-1}(k) & \text{if } n + k \leq m. \end{cases}$$

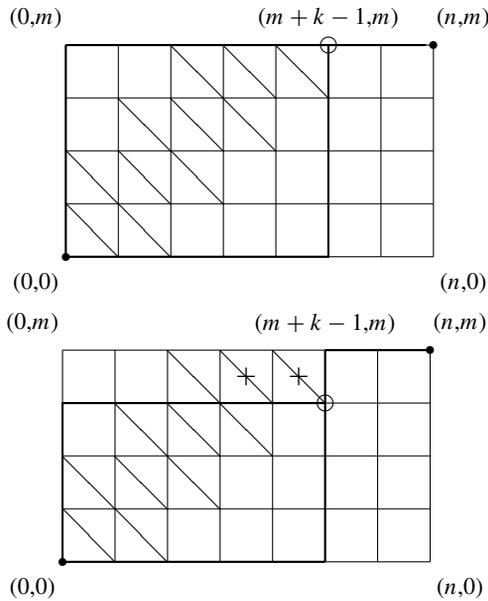


FIGURE 7: Paths through  $(m + k - 1, m)$ . (In this example,  $n = 7, m = 4,$  and  $k = 2.$ )

*Proof.* For  $n = m$ , the equation follows from the fact that the match  $[a_n \text{ vs } b_n]$  is played in the first round. Because  $P_{n,m}(k) = P_{m,n}(k)$  by symmetry, we have only to prove the case  $m < n$ . As in the sure-winner model, if  $n \geq m + k$  then all paths corresponding to the ranking orders identified up to the  $k$ th round must pass through the point  $(m + k - 2, m)$ . We compute the probability of these paths by subtracting the probability of paths through  $(m + k - 1, m - 1)$  and  $(m + k - 1, m)$  from the probability of paths through  $(m + k - 1, m)$  (see Figure 7). The probability of the former paths is  $2^{-k} P_{m+k-1,m-1}(k)$ . Here,  $2^{-k}$  is the probability that  $b_m$  loses to all of  $a_m, a_{m+1}, \dots, a_{m+k-1}$ . Thus,

$$P_{n,m}(k) = P_{m+k-1,m}(k) - 2^{-k} P_{m+k-1,m-1}(k). \tag{4.4}$$

If  $m < n < m + k$  then all paths corresponding to the ranking orders identified up to the  $k$ th round must pass through point  $(n - 1, m)$  or point  $(n, m - 1)$  (see Figure 8). By an argument similar to that leading to (4.4), the probability of the paths through  $(n - 1, m)$  is given by  $P_{n-1,m}(k) - 2^{-(n-m+1)} P_{n-1,m-1}(k)$ . On the other hand, the probability of the paths through  $(n, m - 1)$  is  $2^{-(m-n+1)} P_{n-1,m}(k)$ . Here,  $2^{-(n-m+1)}$  is the probability that  $b_m$  loses to all of  $a_m, a_{m+1}, \dots, a_n$ . Thus,

$$P_{n,m}(k) = P_{n-1,m}(k) - 2^{-(n-m+1)} P_{n-1,m-1}(k) + 2^{-(n-m+1)} P_{n,m-1}(k),$$

as required.

The following lemma gives some properties of  $P_{n,m}(k)$ . We use them later in proving the theorems in which  $P_{n,m}(k)$  and its expected value are evaluated.

**Lemma 4.1.** (a) *We have*

$$P_{n,n}(k) = P_{n,n-1}(k) = P_{n-1,n}(k). \tag{4.5}$$

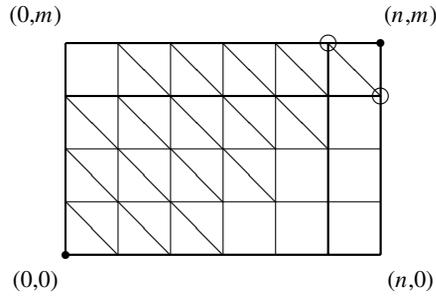


FIGURE 8: All paths must pass through  $(n - 1, m)$  or  $(n, m - 1)$ . (In this example,  $n = 6, m = 4$  and  $k = 3$ .)

(b) If  $m < n \leq m + k$  then

$$P_{n,m}(k) \leq P_{n-1,m}(k). \tag{4.6}$$

The inequality is strict if and only if  $n \geq k + 1$ .

(c) If  $m < n < m + k$  then

$$P_{n,m}(k) \geq P_{n,m-1}(k).$$

The inequality is strict if and only if  $n \geq k + 1$ .

(d) If  $m \leq n$  then

$$P_{n,m}(k) \geq 2^{-1} P_{m,m}(k). \tag{4.7}$$

*Proof.* Part (a) is an immediate consequence of (4.3) and the symmetry  $P_{n,n-1}(k) = P_{n-1,n}(k)$ .

Next we prove parts (b) and (c). If  $m < n \leq k$  then  $P_{n,m}(k) = P_{n-1,m}(k) = P_{n,m-1}(k) = 1$ . Therefore, we have only to prove the  $n \geq k + 1$  case. If  $m = n - k$  then

$$P_{n,m}(k) - P_{n-1,m}(k) = -2^{-k} P_{n-1,m-1}(k) < 0$$

by (4.1). Assume that  $P_{n,m}(k) < P_{n-1,m}(k)$  for a pair  $(m, n)$  such that  $m + 1 < n \leq m + k$ ; then

$$P_{n,m+1}(k) - P_{n-1,m+1}(k) = 2^{-n+m} (P_{n,m}(k) - P_{n-1,m}(k)) < 0,$$

by (4.2). Induction on  $m$  proves part (b). If  $m < n < m + k$  then

$$\begin{aligned} P_{n,m}(k) &= P_{m,m}(k) + \sum_{j=1}^{n-m} (P_{m+j,m}(k) - P_{m+j-1,m}(k)) \\ &\stackrel{(4.2)}{=} P_{m,m}(k) + \sum_{j=1}^{n-m} 2^{-(j+1)} (P_{m+j,m-1}(k) - P_{m+j-1,m-1}(k)) \\ &\stackrel{(a),(b)}{\geq} P_{m,m-1}(k) + \sum_{j=1}^{n-m} (P_{m+j,m-1}(k) - P_{m+j-1,m-1}(k)) \\ &= P_{n,m-1}(k). \end{aligned}$$

Thus, part (c) is proved.

A proof of part (d) is as follows. From part (b) and (4.1), if  $n < m + k < n'$  then  $P_{n,m}(k) \geq P_{m+k,m}(k) = P_{n',m}(k)$ . Therefore, it is enough to prove the  $n = m + k$  case:

$$\begin{aligned}
 P_{m+k,m}(k) &\stackrel{(4.1),(4.2)}{=} P_{m,m}(k) + \sum_{j=1}^{k-1} 2^{-(j+1)} (P_{m+j,m-1}(k) - P_{m+j-1,m-1}(k)) \\
 &\quad - 2^{-k} P_{m+k-1,m-1}(k) \\
 &\geq P_{m,m}(k) - \sum_{j=1}^{k-1} 2^{-(j+1)} P_{m+j-1,m-1}(k) - 2^{-k} P_{m+k-1,m-1}(k) \\
 &\stackrel{(b)}{\geq} P_{m,m}(k) - P_{m,m-1}(k) \left( \sum_{j=1}^{k-1} 2^{-(j+1)} + 2^{-k} \right) \\
 &\stackrel{(a)}{=} 2^{-1} P_{m,m}(k).
 \end{aligned}$$

Thus, part (d) is proved.

Note that, from (4.5) and (4.6), we have  $P_{n,n}(k) \leq P_{n-1,n-1}(k)$ , i.e.  $P_{n,n}(k)$  is decreasing in  $n$ .

Recall that  $E_{n,m}^P$  denotes the expectation of  $Y(n, m)$  in the totally random model. The following theorem gives the order of  $E_{n,m}^P$ .

**Theorem 4.3.** *If  $m = n$  or  $m = n + 1$  then*

$$\lim_{n \rightarrow \infty} \frac{E_{n,m}^P}{(\log n)^{1/2}} = \sqrt{\frac{2}{\log 2}}. \tag{4.8}$$

*Proof.* Fix an arbitrarily small  $\delta > 0$ . At the end of the proof we will let  $\delta \downarrow 0$ .

We first consider an upper bound of  $E_{n,m}^P$  for  $m = n$  or  $m = n + 1$ . Let  $W_n^M = (w_{ij}^M)_{i,j=1,\dots,n}$  be defined such that  $w_{ij}^M = 1$  if  $a_i$  wins against  $b_j$ ,  $w_{ij}^M = -1$  if  $b_j$  wins against  $a_i$ , and  $w_{ij}^M = 0$  if the match is not played. Let

$$\check{W}_n^M = (\check{w}_{ij}^M), \quad \check{w}_{ij}^M = w_{j,m-i+1}^M,$$

be as defined in Section 2.3. At least  $k + 1$  rounds are necessary to identify a ranking if and only if  $\check{W}_n^M$  contains at least one  $(k + 1) \times (k + 1)$  submatrix of the form

$$\begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 1 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 1 & \cdots & 1 & 1 \\ 1 & 1 & \cdots & 1 & \pm 1 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} \pm 1 & -1 & \cdots & -1 & -1 \\ -1 & -1 & \cdots & -1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -1 & -1 & \cdots & 0 & 0 \\ -1 & 0 & \cdots & 0 & 0 \end{bmatrix}.$$

Thus, by Bonferroni’s inequality,

$$1 - P_{n,m}(k) \leq (n - k)2^{-(k+1)(k+2)/2+2} \leq 4n2^{-k^2/2}. \tag{4.9}$$

Therefore,

$$\begin{aligned}
 E_{n,m}^P &= 1 + \sum_{k=1}^{m-1} (1 - P_{n,m}(k)) \\
 &\leq 1 + \sum_{k=1}^{\lfloor 2(\log_2 n)^{1/2} \rfloor} (1 - P_{n,m}(k)) + \sum_{k=\lfloor 2(\log_2 n)^{1/2} \rfloor + 1}^{m-1} (n-k)2^{-(k+1)(k+2)/2+2} \\
 &\leq 1 + \sum_{k=1}^{\lfloor 2(\log_2 n)^{1/2} \rfloor} (1 - P_{n,m}(k)) + 4n^2 2^{-2 \log_2 n} \\
 &= \sum_{k=1}^{\lfloor 2(\log_2 n)^{1/2} \rfloor} (1 - P_{n,m}(k)) + 5.
 \end{aligned}$$

Now we divide the sum on the right-hand side into two parts: schematically,

$$\sum_{k=1}^{\lfloor 2(\log_2 n)^{1/2} \rfloor} = \sum_{k=1}^{\lfloor (2(1+\delta) \log_2 n)^{1/2} \rfloor} + \sum_{k=\lfloor (2(1+\delta) \log_2 n)^{1/2} \rfloor + 1}^{\lfloor 2(\log_2 n)^{1/2} \rfloor}. \tag{4.10}$$

By (4.9), for  $k \geq (2(1 + \delta) \log_2 n)^{1/2}$  we have

$$1 - P_{n,m}(k) \leq 4n \times n^{-(1+\delta)} = 4n^{-\delta}.$$

Therefore,

$$\sum_{k=\lfloor (2(1+\delta) \log_2 n)^{1/2} \rfloor + 1}^{\lfloor 2(\log_2 n)^{1/2} \rfloor} (1 - P_{n,m}(k)) \leq 8(\log_2 n)^{1/2} n^{-\delta}.$$

In the first summation on the right-hand side of (4.10), we use only the fact that  $1 - P_{n,m}(k) \leq 1$ . Combining these bounds yields

$$E_{n,m}^P \leq 5 + (2(1 + \delta) \log_2 n)^{1/2} + 8(\log_2 n)^{1/2} n^{-\delta}.$$

Therefore,

$$\limsup_{n \rightarrow \infty} \frac{E_{n,m}^P}{(\log n)^{1/2}} \leq \sqrt{\frac{2(1 + \delta)}{\log 2}}. \tag{4.11}$$

We now give a lower bound of  $E_{n,m}^P$ . First, we have

$$\begin{aligned}
 P_{n,n}(k) &= P_{n,n-1}(k) \\
 &\stackrel{(4.1),(4.2)}{\leq} P_{n-1,n-1}(k) - 2^{-2}(P_{n-1,n-2}(k) - P_{n,n-2}(k)) \\
 &\leq P_{n-1,n-1}(k) - 2^{-2-3}(P_{n-1,n-3}(k) - P_{n,n-3}(k)) \\
 &\leq \dots \\
 &\leq P_{n-1,n-1}(k) - 2^{-k(k+1)/2+1}(P_{n-1,n-k}(k) - P_{n,n-k}(k)) \\
 &\stackrel{(4.1)}{=} P_{n-1,n-1}(k) - 2^{-(k+1)(k+2)/2+2} P_{n-1,n-k-1}(k) \\
 &\stackrel{(4.7)}{\leq} P_{n-k-1,n-k-1}(k) - 2^{-(k+1)(k+2)/2+1} P_{n-k-1,n-k-1}(k) \\
 &= (1 - 2^{-(k+1)(k+2)/2+1}) P_{n-k-1,n-k-1}(k).
 \end{aligned}$$

From  $P_{k,k}(k) = 1$ , we then have

$$P_{n,n}(k) \leq (1 - 2^{-(k+1)(k+2)/2+1})^{\lfloor (n-k)/(k+1) \rfloor} \leq (1 - 2^{-(k+2)^2/2})^{n/(k+2)-2}.$$

Since  $P_{n,n}(k)$  is increasing in  $k$ , for  $k \leq (2(1 - \delta) \log_2 n)^{1/2} - 2$  we obtain

$$\begin{aligned} P_{n,n}(k) &\leq (1 - n^{-(1-\delta)})^{n/(2(1-\delta) \log_2 n)^{1/2}-2} \\ &= ((1 - n^{-(1-\delta)})^{n^{1-\delta}})^{n^\delta/(2(1-\delta) \log_2 n)^{1/2}-2} \\ &\rightarrow 0 \quad \text{as } n \rightarrow \infty. \end{aligned}$$

Therefore, there exists an  $N > 0$  such that, for all  $n \geq N$ ,

$$\begin{aligned} E_{n,m}^P &\geq E_{n,n}^P \\ &\geq 1 + \sum_{k=1}^n (1 - P_{n,n}(k)) \\ &\geq 1 + \sum_{k=1}^{\lfloor (2(1-\delta) \log_2 n)^{1/2}-2 \rfloor} (1 - P_{n,n}(k)) \\ &> 1 + (1 - \delta)((2(1 - \delta) \log_2 n)^{1/2} - 3). \end{aligned}$$

Therefore,

$$\liminf_{n \rightarrow \infty} \frac{E_{n,m}^P}{(\log n)^{1/2}} \geq (1 - \delta) \sqrt{\frac{2(1 - \delta)}{\log 2}}. \tag{4.12}$$

Letting  $\delta \downarrow 0$  in (4.11) and (4.12) proves (4.8).

In Section 3, we derived the asymptotic distribution of  $Q_{n,m}$ . We now argue that the asymptotic distribution of  $P_{n,m}$  does not exist.

**Remark 4.1.** There exists no sequence  $\{c_n\}$  or distribution function  $F(z)$  such that

$$P_{n,n}((z - c_n)(\log n)^{1/2}) \rightarrow F(z) \quad \text{as } n \rightarrow \infty.$$

This fact is derived from the asymptotic theory of extreme values. Let  $n$  be large and consider matching  $[a_i \text{ vs } b_i]$ ,  $i = 1, 2, \dots$ , in sequence. As the first round of ERM progresses, the Hasse diagram grows and consists of blocks of connected diamonds and line-segments. The number of rounds for each block of connected diamonds is independently and identically distributed. By the strong law of large numbers, the asymptotics in  $n$  and the asymptotics in the number of blocks are essentially the same. We denote the distribution of the number of rounds in each block by  $P(k)$ . Then  $P_{n,n}(k)$  corresponds to the distribution of the maximum value of independent, identically distributed samples from  $P(k)$ . Because

$$1 - P(l) \leq \sum_{k=0}^{\infty} 2^{-(l+k)+1} (k + 1)(1 - P(l - 1)),$$

there exists a  $c > 0$  such that

$$1 - P(l + 1) \leq 2^{-l} c(1 - P(l))$$

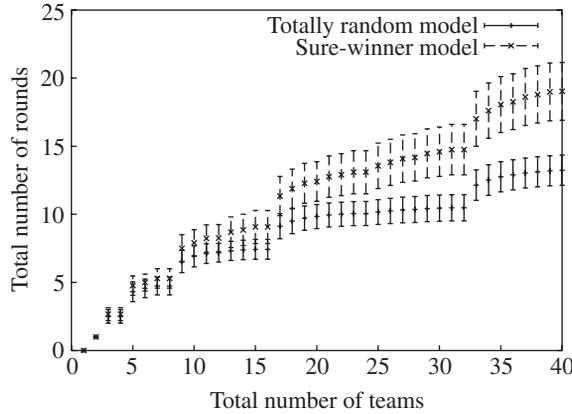


FIGURE 9: Expectation and standard deviation of the number of rounds when using PMS.

for all sufficiently large  $l$ . From Theorem 2.4.5 of [9] or Theorem 1.7.13 of [12], the asymptotic distribution of  $P_{n,n}$  does not exist.

From Theorems 3.3 and 4.3, the asymptotic order of  $E_{n,m}^P$  is smaller than that of  $E_{n,m}^Q$  when  $m = n$  or  $m = n + 1$ . Next we prove that  $E_{n,m}^P$  is smaller than  $E_{n,m}^Q$  for all  $n, m \geq 1$ . Actually we prove a stronger result, on stochastic order in the two models we have considered.

**Theorem 4.4.** *The number of rounds  $Y(n, m)$  is stochastically larger in the sure-winner model than it is in the totally random model, i.e.*

$$P_{n,m}(k) \geq Q_{n,m}(k) \quad \text{for all } n, m \geq 1 \text{ and } k \geq 1.$$

The inequality is strict if and only if  $\max(n, m) \geq k + 1$ .

The proof is presented in the preprint [11] of this paper. The next result follows from Theorem 4.4.

**Corollary 4.1.** *We have*

$$E_{m,n}^P \leq E_{m,n}^Q.$$

The inequality is strict if and only if  $\max(m, n) \geq 2$ .

The values of  $Q_{n,n}(k)$  and  $P_{n,n}(k)$  for some small  $n$  are presented in Tables 5 and 6 and Figure 12 in [11].

Since stochastic order is preserved under convolution and under taking the maximum of independent random variables, we have the following corollary on the stochastic order of the total number of rounds,  $Y$  (see (2.4)), in our two models.

**Corollary 4.2.** *Let  $Y$  be the total number of rounds required to determine the ranking of  $N$  teams. Then*

$$P(Y \leq k \mid \text{totally random model}) \geq P(Y \leq k \mid \text{sure-winner model}) \quad \text{for all } k \geq 1.$$

The inequality is strict for all  $N \geq 3$  and for all  $k$  in the support of  $Y$ .

Figure 9 presents the expected value and the standard deviation of the total number of rounds when using PMS, for  $N = 2, \dots, 40$  teams. Figure 10 shows the distribution of the number of rounds for  $N = 16$  and  $N = 64$ .

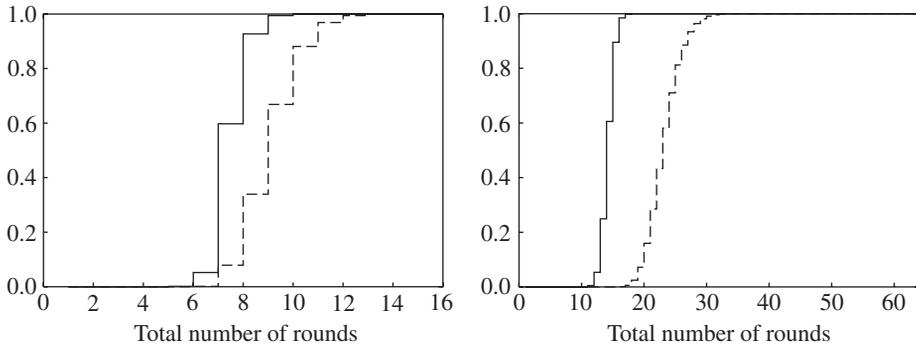


FIGURE 10: Distribution of the number of rounds in the totally random model (solid) and the sure-winner model (dashed) when using PMS, for  $N = 16$  (left) and  $N = 64$  (right).

### 5. Discussion

#### 5.1. Parallel sorting

There is an extensive literature on parallel sorting algorithms [15], [13]. The computing model EREW P-RAM (exclusive-read, exclusive-write parallel random-access machine), which allows no concurrent reads and no concurrent writes, is the closest to our setting. However, there are several differences between the parallel sorting problem using EREW P-RAM and our parallel matching problem. In the parallel sorting problem, only the sure-winner stochastic model is assumed, i.e. a ‘true’ ranking is given initially and kept fixed during the whole algorithm. No contradiction of ranking occurs. Thus, a class of algorithms (scheduling methods) wider than that of the parallel matching problem is admissible. Furthermore, the parallel sorting algorithms compare already comparable teams. Although this seems to be a waste of calculation cost (or an unnecessary increase in the number of necessary rounds), if we consider the cost of the worst-ordered case (or the worst ‘true’ ranking case), the waste does not usually affect the order of the cost. Actually, in most papers on parallel sorting algorithms the number of rounds is evaluated in the worst case. This is one of the biggest differences between those papers and the present paper, in which we evaluate the expectation of the number of rounds.

In spite of these differences, some results on the parallel sorting problem are useful for the parallel matching problem. Odd–even merge is one of the standard parallel merge algorithms [2]. If we omit matches between comparable teams, odd–even merge corresponds to a rectangle method which has the matrix  $X$  with

$$x_{ij} = \min\{k > 0 : n + 1 - i - j \equiv 0 \pmod{2^{\lceil \log_2 n \rceil - k + 1}}\};$$

an example of such a matrix is

$$\begin{bmatrix} 4 & 3 & 4 & 2 & 4 & 3 & 4 & 1 \\ 3 & 4 & 2 & 4 & 3 & 4 & 1 & 4 \\ 4 & 2 & 4 & 3 & 4 & 1 & 4 & 3 \\ 2 & 4 & 3 & 4 & 1 & 4 & 3 & 4 \\ 4 & 3 & 4 & 1 & 4 & 3 & 4 & 2 \\ 3 & 4 & 1 & 4 & 3 & 4 & 2 & 4 \\ 4 & 1 & 4 & 3 & 4 & 2 & 4 & 3 \\ 1 & 4 & 3 & 4 & 2 & 4 & 3 & 4 \end{bmatrix}, \tag{5.1}$$

TABLE 1: The expected number of rounds using optimum scheduling and PMS.

Number of teams	Optimum scheduling	PMS
2	1.00	1.00
3	2.67	2.67
4	2.67	2.67
5	4.00	4.77

with  $n = m = 8$ . It is evident that this algorithm is a 45 degree method and that the scheduling matrix is adequate. The order of the calculation cost of the worst case of this algorithm is easily proved to be  $O(\log n)$ . Thus, the expected number of rounds in the sure-winner model is at most  $O(\log n)$ . This means that the order of expectation,  $O(\sqrt{n})$ , of ERM in the sure-winner model is not optimal. Another standard parallel merge algorithm is the bitonic merge [2], whose scheduling matrix is, for example, (5.1) with its rows in reverse order. In particular, the bitonic merge matches teams in the reverse order:  $[a_1 \text{ vs } b_n], [a_2 \text{ vs } b_{n-1}], \dots, [a_n \text{ vs } b_1]$  in the first round. This is clearly contradictory in the totally random model.

Even if we use the odd–even merge or bitonic merge, the whole sorting has costs of  $O((\log n)^2)$  in the worst case. There are several optimal parallel sorting algorithms, particularly that of [1], which have  $O(\log n)$  costs. However, all of them are contradictory in the totally random case.

## 5.2. Some optimality of PMS for small-size problems

In this subsection, we present some results of the sure-winner model using PMS. When the number of teams is small, we can count all permutations of ‘true’ rankings and find the optimal algorithm in the sense of the minimum expected number of rounds. For example, the optimal algorithm for  $N = 5$  is presented in Appendix D of [11]. The results for  $N = 2, 3, 4, 5$  are given in Table 1. This shows that PMS is optimal if  $N \leq 4$ . Although the order of the expected number of rounds for  $n \times n$  PMS is not optimal, PMS is close to optimal for small  $N$ . For larger  $N$ , it seems combinatorially formidable to find the optimal algorithm in the sure-winner model.

## 5.3. Bradley–Terry model

In the general Bradley–Terry model [3], a positive parameter of strength  $\pi_i$  is assigned to each team  $t_i$ . The probability that team  $t_i$  beats team  $t_j$  is calculated as  $p_{ij} = \pi_i / (\pi_i + \pi_j)$ . One of the most remarkable properties of the model is that it satisfies  $p_{ij} p_{jk} p_{ki} = p_{ji} p_{ik} p_{kj}$ . In other words, there is no three-cornered deadlock. It is evident that, on the one hand, if  $\pi_i = \pi_j$  for all  $i$  and  $j$ , then the Bradley–Terry model corresponds to the totally random model. On the other hand, if  $\pi_i / \pi_j \rightarrow \infty$  for all  $j = i + 1$ , then the asymptotic model corresponds to the sure-winner model. Therefore, the two probabilistic models investigated in the previous sections are two extremes of the Bradley–Terry model.

In this sense, the Bradley–Terry model is much more general than the two probabilistic models and more realistic in describing the actual games. However, it is very difficult to evaluate the number of rounds of the general Bradley–Terry model because we have to consider the ranking of each team within the whole tournament even in the recursive subproblems. If we adopt the sure-winner or totally random model, only the ranking in each subset is needed.

We did some simulation studies of PMS using the Bradley–Terry model. In Figure 11, we present the expected number of rounds in the following simple one-parameter Bradley–Terry

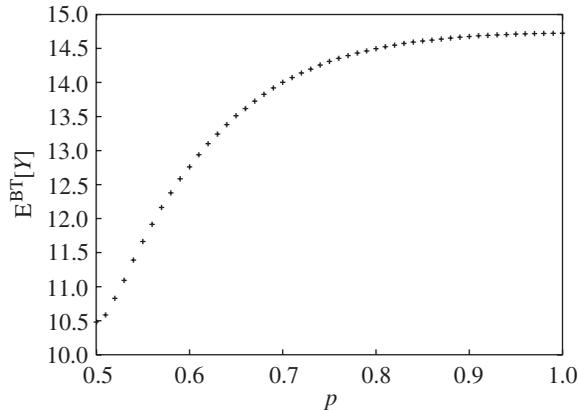


FIGURE 11: Expected number of rounds,  $E^{\text{BT}}[Y]$ , using PMS in the Bradley–Terry model, for  $N = 32$ .

model:  $\pi_i/(\pi_i + \pi_{i+1}) = p$ ,  $i = 1, \dots, N-1$ ,  $\frac{1}{2} \leq p \leq 1$ . Figure 11 shows that the expected number increases monotonically in  $p$ .

#### 5.4. Concluding remarks and future work

In this paper, we have presented a new game scheduling scheme, namely parallel merge sorting with equivalent-rank matching. Ranking using this scheme does not cause any contradiction between any possible results of simultaneous matches. Furthermore, the Hasse diagram of the partial order using this scheme is simple and easy to understand. Two probabilistic models, the sure-winner model and the totally random model, were investigated. For each model, the recurrence formula of the distribution function of the number of rounds for merging was given and the order of the expected number of rounds evaluated.

In the sure-winner model, a parallel sorting algorithm for parallel computers gives an expected number of rounds of smaller order than does the proposed scheme. Optimality in the sense of the expected number of rounds needs more research. Our investigation of the optimal ranking scheme for  $N = 5$  suggests that it is combinatorially very difficult to obtain the optimal ranking scheme for  $N \geq 6$ .

In the totally random model, almost nothing is known about this optimality.

#### Acknowledgements

The authors would like to thank the associated editor and the referees for their comments and suggestions on an earlier version of the paper. We are very grateful to Tomomi Matsui for initial suggestion of the idea of ERM. This work is supported in part by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for JSPS Fellows, 17.10018, 2005.

#### References

- [1] AJTAI, M., KOMLOS, W. AND SZEMEREDI, E. (1983). Sorting in  $c \log(n)$  parallel steps. *Combinatorica* **3**, 1–19.
- [2] BATCHER, K. (1968). Sorting networks and their applications. In *Proc. AFIPS Spring Joint Comput. Conf.*, Vol. 32, Tomson Book Company, Washington, DC, pp. 307–314.
- [3] BRADLEY, R. A. AND TERRY, M. E. (1952). Rank analysis of incomplete block designs. I. The method of paired comparisons. *Biometrika* **39**, 324–345.

- [4] CARROLL, L. (1947). Lawn tennis tournaments. In *The Complete Works of Lewis Carroll*. Modern Library, New York.
- [5] DAVID, H. A. (1988). *Method of Paired Comparisons*, 2nd edn. Oxford University Press.
- [6] EDWARDS, C. T. (1996). Double elimination tournaments: counting and calculating. *Amer. Statistician* **50**, 27–33.
- [7] FELLER, W. (1948). On the Kolmogorov–Smirnov limit theorems for empirical distributions. *Ann. Math. Statist.* **19**, 177–189.
- [8] FELLER, W. (1968). *Introduction to Probability Theory and Its Applications*, Vol. 2, 3rd edn. John Wiley, New York.
- [9] GALAMBOS, J. (1987). *The Asymptotic Theory of Extreme Order Statistics*, 2nd edn. Krieger, Melbourne, FL.
- [10] GNEDENKO, B. V. AND KOROLJUK, V. S. (1961). On the maximum discrepancy between two empirical distributions. In *Selected Translations in Mathematical Statistics and Probability*, Vol. 1, Institute of Mathematical Statistics, Providence, RI, pp. 13–16.
- [11] KOBAYASHI, K., KAWASAKI, H. AND TAKEMURA, A. (2003). Parallel matching for ranking all teams in a tournament. Tech. Rep. METR 2003-16, Department of Mathematical Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Available at <http://www.keisu.t.u-tokyo.ac.jp/Research/techrep.0.html>.
- [12] LEADBETTER, M. R. (1983). *Extremes and Related Properties of Random Sequences and Processes*. Springer, New York.
- [13] LEIGHTON, F. T. (1992). *Introduction to Parallel Algorithms and Architectures*. Morgan Kaufmann, San Mateo, CA.
- [14] MOHANTY, S. G. (1979). *Lattice Path Counting and Applications*. Academic Press, New York.
- [15] PARHAMI, B. (1999). *Introduction to Parallel Processing*. Plenum Press, New York.
- [16] STANLEY, R. P. (1997). *Enumerative Combinatorics*, Vol. I. Cambridge University Press.
- [17] THOMPSON, P., JARYSZAK, E. AND WAMIL, J. (2000). Reducing the pairing effect in 4-team double-elimination tournaments. *Math. Scientist* **24**, 110–121.