SUMMARY OF RESUTLTS FOR THE "UNFAIR" FAIRGROUND GAME

P. HUGGINS AND R. YOSHIDA

1. Setup

We have ten bins, $Bin[1], \ldots, Bin[10]$, with corresponding score values $(1, 1, 2, 3, 3, 4, 4, 5, 6, 6)$. We let $v[i]$ denote the score value of the bin $Bin[i]$.

We have eight numbered balls $Ball[1], \ldots, Ball[8]$, which are rolled one at a time into the bins, as described in Model D. We recall that winning scores are those which are either less than 16 or greater than 40.

We let X denote the set of all possible valid outcomes of rolling the eight balls into the bins. For any $x \in X$, we let $s(x)$ denote the score of outcome x (as described in the article).

2. A Useful Symmetry

Observation 1. *We have* $P(s < 16) = P(s > 40)$

Proof. We can interchange $Bin[j]$ with $Bin[11 - j]$ for all $j = 1, ..., 5$, and thereby obtain a bijection $f : X \to X$. Clearly, since f is induced by merely permuting the bins, we have that $P(f(x)) = P(x)$ for all outcomes $x \in X$. Furthermore, since $v[j] = 7 - v[11 - j]$ for all j, we have that $s(f(x)) = 56 - s(x)$ for all $x \in X$. Thus, $s(f(x)) < 16$ if and only if $s(x) > 40$. Similarly, $s(f(x)) > 40$ if and only if $s(x) < 16$. □

Thus, our desired winning probability for the game is simply $2 * P(s < 16)$.

3. Computing $P(s < 16)$

We consider disjoint cases, according to how many balls are contained in each bin. We sum together the probabilities of those cases which result in a score less than 16, and clearly this yields $P(s < 16)$.

As it turns out, there are only about 15000 possible cases altogether (and only a couple hundred of these cases actually yield scores less than 16.) Thus, from a computational perspective, this approach is very feasible, even for a home computer.

Date: January 23, 2005.

We represent each case by a 10-tuple of integers (where each entry is between 0 and 3), where the jth entry equals the number of balls contained in $Bin[j]$.

We generate all the 15000 or so cases by using a recursive method which lists the cases in "lexicographical" order as follows:

Case 1: $(3, 3, 2, 0, 0, 0, 0, 0, 0, 0)$
Case 2: $(3, 3, 1, 1, 0, 0, 0, 0, 0, 0)$
Case 3: $(3, 3, 1, 0, 1, 0, 0, 0, 0, 0)$

...

Last Case: $(0, 0, 0, 0, 0, 0, 0, 2, 3, 3)$

Then, for each case $(a_1, \ldots, a_{10})$ which yields a score of 16 or less, we compute the probability of the case according to how many bins have three balls (i.e., how many a_i equal 3):

If no bin has three balls:

$$P(case) = \binom{8}{a_1, \ldots, a_{10}} * (10^{-8}) \tag{1}$$

If exactly one bin, $Bin[r]$, has three balls:

Split into subcases according to which ball is the third (i.e. highest numbered) ball to roll into $Bin[r]$. Let $Subcase[k]$ denote the subcase that $Ball[k]$ is the third ball to roll into $Bin[r]$.

$$P(Subcase[k]) = \binom{k-1}{2} \left(\frac{5!}{\prod_{i \neq r} (a_i!)} \right) * (10^{-k} 9^{-8+k}) \tag{2}$$

$$P(case) = \sum_{k=3}^{8} P(Subcase[k]) \tag{3}$$

If exactly two bins, $Bin[r_1]$ and $Bin[r_2]$, have three balls apiece:

Split into subcases according to which ball is the third (i.e. highest numbered) ball to roll into $Bin[r_1]$ and which ball is the third to roll into $Bin[r_2]$. Let $Subcase[k_1][k_2]$ denote the subcase that $Ball[k_1]$ is the third ball to roll into $Bin[r_1]$ and $Ball[k_2]$ is the third ball to roll into $Bin[r_2]$.

By symmetry, we may suppose $k_1 < k_2$ so long as we remember to multiply our probabilities by 2 when we sum them up.

(4)

$$P(Subcase[k_1][k_2]) = \binom{k_1-1}{2}\binom{k_2-4}{2}\left(\frac{2!}{\prod_{i\notin\{r_1,r_2\}}(a_i!)}\right)*(10^{-k_1}9^{-k_2+k_1}8^{-8+k_2})$$

(5)
$$P(case) = \sum_{k_1<k_2,\ 3\le k_1\le 7,\ 6\le k_2\le 8} 2*P(Subcase[k_1][k_2])$$

4. Computational Results

We wrote a C++ program which calculated the winning probability for Model D, using the above formulas. The program was run on a Linux PC, taking advantage of native 64-bit integer arithmetic to handle large integers.

The winning probability was outputted as an exact fraction:

(6)
$$P(win) = \frac{2572423315200}{377913600000000} = 0.0068069\ldots$$

University of California, Berkeley

E-mail address: phuggins@math.berkeley.edu

Duke University

E-mail address: ruriko@math.duke.edu