

RANDOMIZED RULES FOR THE TWO-ARMED-BANDIT WITH FINITE MEMORY¹

By S. M. SAMUELS

Purdue University

1. Summary. In the “two-armed-bandit with finite memory” problem, each rule which has been proposed (see [2], [3], and [4]) can be improved by using a corresponding randomized rule. The performance of various randomized rules is computed.

2. The problem and the rules. The “two-armed-bandit with finite memory” problem was proposed by Robbins [3] and is as follows: We are given two coins with unknown probabilities, p_1 and p_2 , of heads. At each stage, *based only on the results of the previous r tosses*, we must decide which coin to toss next. Our goal is to find the rule which maximizes the limiting proportion of heads. A more precise definition of the worth of a rule makes the problem well-defined and avoids trivialities.

The rules which have been proposed may all be described as follows: First toss one of the coins until it gives r consecutive tails. This is a long block. Then test the other coin. If it passes the test, toss it until it, in turn, gives r consecutive tails. If it fails, return to the original coin and repeat the process.

The following tests have been proposed:

Robbins [3] test: Start with the other coin and toss the two coins alternately until one of them gives heads. If this coin is the other coin, it passes; if not, it fails.

Isbell [2] and Smith and Pyke [4] tests: Let $\delta = (\delta_1, \delta_2, \dots, \delta_s)$ be a binary vector. At the k th stage of the test toss the other coin if $\delta_k = 1$, the original coin if $\delta_k = 0$. The test continues until either the other coin gives tails, the original coin gives heads, or the end of the δ -vector is reached. In the latter case the other coin passes; otherwise it fails.

Isbell uses the simplest δ -vector: $\delta = (1)$ Smith and Pyke first generalize Isbell's test to δ 's consisting of s ones ($1 \leq s \leq r - 2$), then consider the longest possible δ 's: maximal-length memory wheels. (The constraints on δ are imposed not only by the finiteness of the memory, but also by the desire to avoid having the same memory state occur in both long blocks and test blocks. The latter constraint rules out, for example, a δ consisting of $r - 1$ ones.)

Isbell's test is uniformly better than Robbins'. Moreover, as s increases from 1 to $r - 2$, the test gets uniformly better. Smith and Pyke offer numerical evidence which suggests that a memory-wheel test is still better.

Received 22 January 1968.

¹ This research was supported in part by the Aerospace Research Laboratories Contract AF 33(657) 11737 at Purdue University. Reproduction in whole or in part is permitted for any purpose of the United States Government.

To each of the preceding rules we associate the following randomized rule: If a coin gives r consecutive tails, start the test with probability $1 - b$ (i.e., with probability b toss the same coin again). Otherwise follow the rule as stated.

If $b = 0$, there is no randomization; if $b = 1$ we stick with one coin from some point on, which is not allowed.

We shall show that the bigger b is, the better the rule, so that by being sufficiently reluctant to get on with the test we can bring the performance of our rule arbitrarily close to the (unattainable) maximum at $b = 1$.

3. The performance of the rules. The limiting proportion of heads is an increasing function of the limiting ratio of the number of tosses with the better coin to the number of tosses with the worse coin. Denote this ratio by n_1/n_2 , assume that p_1 is greater than p_2 and let $q_1 = 1 - p_1$, $q_2 = 1 - p_2$.

For the nonrandomized rules,

$$(3.1) \quad n_1/n_2 = [m_1(\lambda_1 + \sigma_1') + m_2\sigma_1]/[m_2(\lambda_2 + \sigma_2') + m_1\sigma_2] \\ = [t_1(\lambda_1 + \sigma_1') + t_2\sigma_1]/[t_2(\lambda_2 + \sigma_2') + t_1\sigma_2]$$

where λ_1 is the expected length of a long block with the better coin, σ_1 is the expected number of tosses of the better coin during one of its tests, σ_1' is the expected number of tosses of the better coin during a test of the worse coin ($\sigma_1' = 0$ in the "s-test" rules), m_1 is the expected number of tests of the *worse coin* until a test is passed, and $t_2 (= 1/m_1)$ is the probability that the worse coin will pass its test. The quantities λ_2 , σ_2 , σ_2' , m_2 , and t_1 are correspondingly defined.

Equation (3.1) was derived in [4], Theorem 4.1, by considering the process as an irreducible recurrent Markov chain, choosing an appropriate state, and computing the expected numbers of tosses with each coin—denoted by n_1 and n_2 respectively—between successive returns to this state. Note that the numerator and denominator in the right side of (3.1) are *equal* to n_1 and n_2 respectively. If they were merely *proportional* to n_1 and n_2 , formulas (3.2) and (3.3) below would not follow. (I thank the referee for stressing this point.)

A routine calculation yields

$$\lambda_1 = (1 - q_1^r)/p_1q_1^r, \quad \lambda_2 = (1 - q_2^r)/p_2q_2^r.$$

For the Robbins rule

$$\sigma_1 = 1/(1 - q_1q_2), \quad \sigma_1' = q_2/(1 - q_1q_2), \quad t_2 = p_2/(1 - q_1q_2)$$

with corresponding formulas for σ_2 , σ_2' , t_1 .

Hence, for the Robbins rule,

$$n_1/n_2 = (1 + p_1\lambda_1)/(1 + p_2\lambda_2) = (q_2/q_1)^r.$$

For the other rules, we let

$$\rho_0 = 0, \quad \rho_k = \sum_{i=0}^k \delta_i \quad k = 1, \dots, s.$$

Then,

$$\begin{aligned} \sigma_1 &= \sum_{k=0}^{s-1} \delta_{k+1} p_1^{\rho k} q_2^{k-\rho k}, \\ \sigma_1' &= \sum_{k=0}^{s-1} (1 - \delta_{k+1}) p_2^{\rho k} q_1^{k-\rho k}, \\ t_2 &= p_2^{\rho s} q_1^{s-\rho s}, \end{aligned}$$

as given in [4], with corresponding formulas for σ_2 , σ_2' , and $t_1 = 1/m_2$.

In the corresponding randomized rules, λ_1 is replaced by

$$\lambda_1^* = \lambda_1(1 - q_1 b)/(1 - b) + b/(1 - b),$$

and λ_2 by a corresponding λ_2^* . Hence, for the Robbins rule, the ratio n_1/n_2 is replaced by

$$\begin{aligned} (3.2) \quad n_1^*/n_2^* &= (1 + p_1 \lambda_1^*)/(1 + p_2 \lambda_2^*) \\ &= (q_2/q_1)^r [(1 - q_1 b)/(1 - q_2 b)] \end{aligned}$$

which increases with b to the (unattainable) upper bound, $(q_2/q_1)^r (p_1/p_2)$.

For the other rules, n_1/n_2 is replaced by

$$\begin{aligned} (3.3) \quad n_1^*/n_2^* &= [t_1(\lambda_1^* + \sigma_1') + t_2 \sigma_1]/[t_2(\lambda_2^* + \sigma_2') + t_1 \sigma_2] \\ &= [(1 - b)n_1 + b t_1(1 + p_1 \lambda_1)]/[(1 - b)n_2 + b t_2(1 + p_2 \lambda_2)] \end{aligned}$$

which is a monotone function of b . To show that (3.3) is an *increasing* function of b , we must establish that

$$n_1/t_1(1 + p_1 \lambda_1) \leq n_2/t_2(1 + p_2 \lambda_2).$$

Now, since $p_1 > p_2$, $q_2 > q_1$,

$$\begin{aligned} (3.4) \quad &n_1/t_1(1 + p_1 \lambda_1) \\ &= [p_1^{\rho s} q_2^{s-\rho s} [(1 - q_1^r)/p_1 q_1^r + \sum_{k=0}^{s-1} (1 - \delta_{k+1}) p_2^{\rho k} q_1^{k-\rho k}] + p_2^{\rho s} q_1^{s-\rho s} \\ &\quad \cdot \sum_{k=0}^{s-1} \delta_{k+1} p_1^{\rho k} q_2^{k-\rho k}] \{p_1^{\rho s} q_2^{s-\rho s} [1 + (1 - q_1^r)/q_1^r]\}^{-1} \\ &= (1 - q_1^r)/p_1 + q_1^r \sum_{k=0}^{s-1} (1 - \delta_{k+1}) p_2^{\rho k} q_1^{k-\rho k} \\ &\quad + p_2^{\rho s} q_1^{s+r-\rho s} \sum_{k=0}^{s-1} \delta_{k+1} p_1^{-(\rho s-\rho k)} q_2^{-(s-k-\rho s+\rho k)} \\ &\leq (1 - q_2^r)/p_2 + q_2^r \sum_{k=0}^{s-1} (1 - \delta_{k+1}) p^{\rho k} q_2^{k-\rho k} \\ &\quad + p_1^{\rho s} q_2^{s+r-\rho s} \sum_{k=0}^{s-1} \delta_{k+1} p_2^{-(\rho s-\rho k)} q_1^{-(s-k-\rho s+\rho k)} \\ &= n_2/t_2(1 + p_2 \lambda_2). \end{aligned}$$

Thus (3.3) is increasing in b , to the (unattainable) upper bound:

$$(3.5) \quad t_1(1 + p_1 \lambda_1)/t_2(1 + p_2 \lambda_2) = (q_2/q_1)^{r+s-\rho s} (p_1/p_2)^{\rho s}.$$

Note that (3.5) is very similar to (4.1) of [4], since, as b increases to 1, the contribution to (3.3) from the test blocks decreases to 0.

In the best s -test $s = \rho_s = r - 2$; in the memory-wheel test, $s = 2^r - r - 2$ and $\rho_s = 2^{r-1} - 2$.

The following table shows how much improvement can be made by randomization (the columns headed "Best s -test" and "Memory-wheel test" were computed by Smith and Pyke [4]).

TABLE I
Limiting ratio of number of tosses with better coin to number with worse coin

| | Best s -test | L.U.B. for Randomized Version | Memory wheel test | L.U.B. for Randomized Version |
|------------------------|----------------|----------------------------------|----------------------|----------------------------------|
| $p_1 = 0.5, p_2 = 0.4$ | | | | |
| $r = 4$ | 2.55 | 3.24 | 9.11 | 16.40 |
| 5 | 3.76 | 4.86 | 50.02 | 420.38 |
| 6 | 5.62 | 7.29 | 122.33 | 276121.52 |
| 7 | 8.46 | 10.94 | 250.36 | 1.19×10^{11} |
| 8 | 12.76 | 16.40 | 506.34 | 2.22×10^{22} |
| 9 | 19.23 | 24.60 | 1018.33 | $> 10^{38}$ |
| 10 | 28.98 | 36.91 | 2042.33 | $> 10^{38}$ |
| 11 | 43.64 | 55.36 | 4090.32 | $> 10^{38}$ |
| 12 | 65.67 | 83.04 | 8186.32 | $> 10^{38}$ |
| 13 | 98.76 | 124.56 | 16378.32 | $> 10^{38}$ |
| 14 | 148.45 | 186.83 | 32762.32 | $> 10^{38}$ |
| 15 | 223.03 | 280.25 | 65530.32 | $> 10^{38}$ |
| 16 | 334.96 | 420.38 | 131066.32 | $> 10^{38}$ |
| 17 | 502.92 | 630.57 | 262138.32 | $> 10^{38}$ |
| 18 | 754.94 | 945.85 | 524282.32 | $> 10^{38}$ |
| 19 | 1133.04 | 1418.78 | 1048570.33 | $> 10^{38}$ |
| 20 | 1700.29 | 2128.16 | 2097146.33 | $> 10^{38}$ |
| $p_1 = 0.5, r = 5$ | | | | |
| $p_2 = 0.499$ | 1.01 | 1.02 | 1.06 | 1.06 |
| 0.490 | 1.15 | 1.17 | 1.75 | 1.82 |
| 0.400 | 3.76 | 4.86 | 50.02 | 420.38 |
| 0.100 | 53.21 | 2361.96 | 61.33 | 7.41×10^{13} |
| $p_1 = 0.5, r = 10$ | | | | |
| $p_2 = 0.499$ | 1.03 | 1.04 | 7.68 | 7.72 |
| 0.490 | 1.40 | 1.43 | 2041.14 | 7.55×10^8 |
| 0.400 | 28.98 | 36.91 | 2042.33 | $> 10^{38}$ |
| 0.100 | 1841.32 | 1.39×10^8 | 2045.32 | $> 10^{38}$ |

4. Related problems. If the memory length is r , then there are 4^r possible "memory states" (for each toss we specify which coin was tossed and what the outcome was). For some rules, however, it is not necessary to keep track of all this information. For example, the Robbins rule can be restated as "Switch coins whenever the last r tosses all resulted in tails"; hence it "uses" only 2^r states. In an oral communication T. Cover has suggested substituting the number of states used for the memory length as a basis for comparing rules. He notes that from this point of view the Robbins rule is better than the Isbell rule.

Cover [1] has also shown that if we modify the problem by allowing ourselves a "clock" (i.e., at each stage we know how many tosses we've made so far), then there exists an optimal rule. This rule is in fact as good as one could possibly hope for: the limiting proportion of heads is $\max(p_1, p_2)$.

5. Acknowledgment. I wish to thank A. Dvoretzky who introduced me to the problem and H. Robbins who suggested looking at randomized rules.

REFERENCES

- [1] COVER, THOMAS M. (1968). A note on the two-armed bandit problem with finite memory. *Information and Control*.
- [2] ISBELL, J. R. (1959). On a problem of Robbins. *Ann. Math. Statist.* **30** 606-610.
- [3] ROBBINS, HERBERT (1956). A sequential decision problem with a finite memory. *Proc. Nat. Acad. Sci.* **42** 920-933.
- [4] SMITH, CARTER VINCENT and PYKE, RONALD (1965). The Robbins-Isbell two-armed bandit problem with finite memory. *Ann. Math. Statist.* **36** 1375-1386.