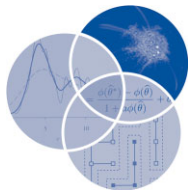
Focus Article



Algorithms for response adaptive sampling designs

Janis Hardwick[1] and Quentin F. Stout[1]*

An experimental design is a formula or algorithm that specifies how resources are to be utilized throughout a study. The design is considered to be good or even optimal if it allows for sufficiently precise and accurate data analysis with the least output of resources such as time, money and experimental units. Most experimental designs use fixed sampling procedures in which the sample sizes and order of allocations to different study groups are known in advance. © 2009 John Wiley & Sons, Inc. *WIREs Comp Stat* 2009 1 118–122

INTRODUCTION

Adaptive designs are those that allow investigators to adjust resource expenditures while the experiment is being carried out. In some experiments, adaptation depends merely on knowledge of ongoing study group sample sizes. This may occur, for example, when attempting to balance allocations across covariates or when randomizing to achieve predetermined sampling proportions. However, more flexibility and efficiency can be gained when adaptation also incorporates knowledge of the responses observed to date. Such designs are referred to as *response* adaptive designs. Although response adaptive designs can offer significant ethical and cost advantages, their adaptive nature greatly complicates the explicit determination of their statistical properties. This precludes classic analytic approaches in the evaluation of such designs. As an example, even something as basic as the proportion assigned to a given study group is a random variable whose distribution cannot be obtained analytically.

Nevertheless, with the advent of increased computational power and improved algorithms, various classes of response adaptive designs can be assessed in exact terms. The easiest members of this class to analyze are the fully sequential designs in which, at each stage, only one observation is allocated to a study group. Group or 'staged' designs are more complex to address.

*Correspondence to: Qstout@umich.edu

[1]CSE Department, University of Michigan, Ann Arbor, MI 48109, USA

DOI: 10.1002/wics.025

FULLY SEQUENTIAL EXPERIMENTS

We begin with fully sequential designs modeled after classic *multi-armed bandit* problems. In its simplest form, the multi-armed bandit arises from an analogy to slot machines. Suppose the machine has k arms and you have n tokens. Assume that the outcomes from pulling an arm are dichotomous, so that when you deposit a token and pull (observe) arm i, you receive \$1 with an unknown probability p_i and \$0 with probability $(1 - p_i)$, $i = 1, \ldots, k$. The 'arms' are independent as are the outcomes, and your goal is to sample from them in such a way as to maximize your winnings after n pulls. This can also be viewed as trying to minimize your losses. While we stick to the example of independent Bernoulli outcomes for each group or arm, other bandit designs have more general distributions.

The two-armed Bernoulli bandit has long been used to model clinical trials in which there are ethical concerns regarding the well being of patients both during and after the trial.[1] One reason for this is that the strategy for solving this problem results in excellent information gathering which is critical to making good decisions after the experiment. Also important, however, is the fact that the optimal solution to the problem minimizes patient losses during the trial. In this way two important trial goals are addressed.

To formalize the problem, suppose there are two arms available, A_1 and A_2. At any point in the experiment, a sufficient statistic is the number of successes s_1, s_2 and failures f_1, f_2 on each arm. Note that $s_1 + s_2 + f_1 + f_2$ is the number of observations made so far. The vector (s_1, f_1, s_2, f_2) is called a *state*. The general design problem is to decide which arm should be pulled next at any given state.

There is a well-known approach to finding an optimal sampling scheme for this and similar problems. It is known as *dynamic programming* (DP), which was initially developed by Bellman and is delineated in Ref 2. It can be described as follows: given a state σ, let $p_1(\sigma)$ and $p_2(\sigma)$ denote the probabilities of success for the two arms. Given some objective function $\mathcal{O}$, such as maximizing successes, let $V(\sigma)$ denote the maximal expected value of $\mathcal{O}$ among all possible sequences starting at σ.

The DP equations that allow one to compute V are

$$V(\sigma) = \max\{V_1(\sigma), V_2(\sigma)\}, \quad (1)$$

where

$$V_i(\sigma) = p_i(\sigma)\cdot V(\sigma+s^i) + (1-p_i(\sigma))\cdot V(\sigma+f^i), \quad i=1,2 \quad (2)$$

where s^i (f^i) denotes one success (failure) observed on arm i. $V_i(\sigma)$ represents the maximum expected value possible if the next pull is on arm i, and Eq. (1) says that the optimal decision is to pull the arm with greatest maximum. Extending to more arms is straightforward.

The DP equations are based on a Bayesian framework, from which one can determine $p_i(\sigma)$ as the mean of a posterior distribution. For Bernoulli variables the beta distributions, $\beta(a,b)$, are conjugate priors with simply expressed means, $a/(a+b)$. At any stage, if s successes and f failures have been observed, the posterior mean is $(s+a)/(s+f+a+b)$. This simplicity helps explain their nearly universal usage as priors for Bernoulli bandits.

Note that Eqs (1) and (3) can be used to optimize quite general objective functions, not just bandit problems that accumulate rewards as the experiment proceeds. DP is a very powerful technique that has been applied to a wide range of problems. Numerous books and courses on the subject exist and most computer science students are exposed to DP as undergraduates.

Writing a program based on DP equations is typically straightforward, starting at the end of the experiment and progressing toward the beginning. For example, for an experiment with fixed sample size n, one must first determine the value of V for all states with n observations. These equations are then used to determine the value of V for all states with

```
{Initialize V at terminal states}
for f2=0 to n do
   for s2=0 to n-f2 do
      for f1=0 to n-f2-s2 do
         s1=n-f2-s2-f1
         V(s1,f1,s2,f2)= O(s1,f1,s2,f2)
      end for f1
   end for s2
end for f2

{evaluate V at all nonterminal states}
for m=n-1 downto 0 do
   for f2=0 to m do
      for s2=0 to m-f2 do
         for f1=0 to m-f2-s2 do
            s1=m-f2-s2-f1
            V1=p1(s1,f1)*V(s1+1,f1,s2,f2)+(1-p1(s1,f1))*V(s1,f1+1,s2,f2)
            V2=p2(s2,f2)*V(s1,f1,s2+1,f2)+(1-p2(s2,f2))*V(s1,f1,s2,f2+1)
            V(s1,f1,s2,f2)= max{v1,v2}
         end for f1
      end for s2
   end for f2
end for m
```

FIGURE 1 | Dynamic programming to optimize two-armed bandit.

	Number of Arms		
n	2	3	4
10	1001	8008	43,758
50	316,251	32,468,436	$1.9\ 10^9$
100	4,598,126	$1.7\ 10^9$	$3.5\ 10^{11}$
1000	$4.2\ 10^{10}$	$1.4\ 10^{15}$	$2.6\ 10^{19}$

FIGURE 2 | Number of states as a function of number of arms and experiment size n.

$n-1$ observations, then $n-2$, etc. This is illustrated in Figure 1. Unfortunately, the computations can be quite lengthy and require significant storage space. An experiment with sample size n and k arms has $\binom{n+2k}{2k} \approx n^{2k}/2k!$ states, which grows polynomially with n and exponentially with k (the 'curse of dimensionality'). Figure 2 shows some sample values of the number of states.

Storing this in a straightforward manner would use a $2k$-dimensional array where each coordinate has extent $[0{:}n]$ (the coordinates represent the components of the states), which has $(n+1)^{2k}$ entries. This can be reduced by a factor of $2k!$ by noting that the condition $s_1 + f_1 + \ldots f_k \leq n$ restricts the states to a corner of the array. By linearizing the index scheme the memory required can be reduced to the number of states. See Ohemke et al.[3] for this and other time and space optimizations for serial and parallel programs. Using these optimizations, it is currently possible to optimize the two-armed Bernoulli bandit problem for experimental sizes of many hundreds on a laptop computer, and the three-armed bandit for many hundreds on a parallel computer. We are unaware of any work in which the optimal solution is determined for more arms and a non-trivial experiment size, despite interest in multiarm problems such as in Ref. 4–6.

One common trait of DP approaches is that they produce the optimal value of the objective function. However, more work is required to determine the experimental decisions that would achieve this value. To accomplish this, when the optimal i is determined in Eq. (1) it is stored in a separate array, say D. From this, the design can be recreated top-down. For example, if $D(0,0,0,0) = 1$ then arm 1 is the first arm pulled in the experiment. If it is a success then the next pull is on $D(1,0,0,0)$, while if it is a failure the next pull is on arm $D(0,1,0,0)$, and so on.

Given an adaptive sampling design, researchers often wish to evaluate its characteristics on multiple criteria. For example, the design may optimize successes, but the expected number of pulls on arm 1 is also be of interest. It may also be that the design is *adhoc* and both of these criteria need to be determined even though the design optimizes nothing. Another possibility is that for various fixed success probabilities p_1 and p_2, certain operating characteristics need to be known; e.g., the robustness of its frequentist properties.

It is possible to carry out such evaluations by proceeding as in the calculation of the D matrix above: first evaluate the criterion at the terminal states, and then at each state, use the choice the design would make to calculate the expected value of the criterion at that state. If there are many such evaluations, as with pointwise evaluations, then a more efficient approach is useful: one determines, for each terminal state σ, the number of different ways that the design could end at σ, and then evaluates each criterion using these counts. See Hardwick and Stout[7] for details of this approach, which is known as *path induction*.

An important variation of the bandit model described earlier is the case in which rewards and costs are associated with each pull as part of an infinite process. Typically one utilizes a geometric discount, so that a reward r after p pulls is worth $r\delta^p$ for some $\delta < 1$. From the perspective of the clinical trials problem, this model represents a greater emphasis on continuing to gather information to treat both trial and future patients. For example, in the bandit problem originally described, the last arm pulled will always be the myopic choice, i.e., the arm with highest expected value. In the geometric design, information is always being gathered for future decisions, so the arm pulled at the last step might have a smaller mean but higher variance. The geometrically discounted bandit is also popular due to an elegant theorem by Gittins and Jones[8] that reduces the dimensionality of the problem. It states that one can compute an index for each arm, independent of all other arms, and then make the optimal choice at each stage by taking the arm with the highest index. Unfortunately, while the dimension of the problem is greatly reduced, the computation of the indices is very difficult, so, in practice, approximate solutions are used.

STAGED EXPERIMENTS

While the fully sequential designs described optimize a specified objective function, they also introduce the requirement that the result of each pull be observed before the next pull. In many experimental settings this is not a feasible scenario and the sampling must be done in stages. Most staged designs considered use one of two basic scenarios that have dominated the literature for over 50 years.

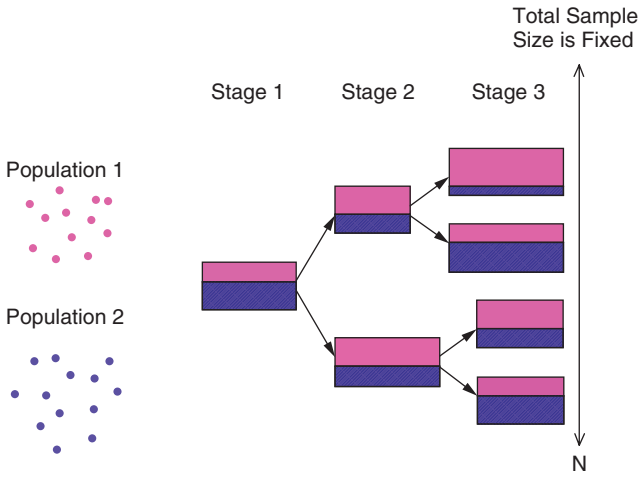


FIGURE 3 | Representation of an optimal three-stage design.

In one scenario the total sample size is a random variable and the designs have either two or three stages. The problem is to find asymptotic approximations for the stage sizes when seeking fixed precision confidence intervals or for minimizing risk functions when observational costs are incurred. See for example Stein[9] and Ghurye and Robbins[10] for early versions and Ghosh et al.[11] for a survey.

The other prominent staged design uses a fixed total sample size, n, and the goal is to maximize expected reward under the following conditions. There are two stages. In the first stage, allocation is in pairs. In the second stage, all observations are made from the apparently better population after stage one. The only variable in the problem is the first stage size which depends only on n and, in the Bayesian case, on the priors. This problem was proposed in 1965 by Colton[12] and later addressed by Canner.[13] Using a Bayesian model with uniform priors, Canner conjectured that the optimal first stage length was approximated by $\sqrt{2n+4}-2$. Despite the fact that algorithms to solve this problem exactly were available in 1995 in Hardwick and Stout,[14] approximate first stage sizes were still being suggested as recently as 2003 in Cheng et al.[15] An application of the algorithms in Ref 14 to this two-stage problem appears in Ref 16.

More general and flexible multi-stage designs allow for many more variables than a single stage size. First, assuming a fixed sample size, n, the allocations within each stage (except the first) may vary as a result of all observations obtained as of the last completed stage. Second, no stages have predetermined length. Finally, it is even allowable that the number of stages is unknown in advance. Flexible designs of this nature are shown in Figure 3.

In Refs 14,16 algorithms were developed to fully optimize flexible staged designs with diverse objective functions. For most objective functions flexible two-stage designs are relatively easy to optimize computationally. Optimizing staged allocation with more than two stages is complicated because of the large number of options, and their outcomes, at each state. For example, in an study of fixed size n, for a given stage and given state, if there are m observations remaining in the experiment, then the number of observations o_1, o_2 assigned to arm 1 and 2, respectively, need only satisfy $0 \le o_1, o_2$ and $1 \le o_1 + o_2 \le m$, i.e., there are $\binom{m+2}{2} \approx m^2/2$ sampling options, as opposed to the two options for the fully sequential case. Further, given o_1, o_2, the number of possible outcomes is $(o_1+1)\cdot(o_2+1)$, so a straightforward evaluation of all of the sampling options, and their outcomes, involves $\binom{m+4}{4} \approx m^4/4!$ values at each state. There are $\binom{n+4}{4}$ states, and the total number of values would be $\binom{n+8}{8} \approx n^8/8!$. If there are t stages then the total time is further multiplied by t. This is rather unmanageable for sample sizes of interest. Fortunately, by reusing intermediate results this can be reduced to time proportional to the number of distinct states and options, i.e., $\Theta(tn^6)$ time—see Ref 16. Using this, multi-stage designs with sample sizes in the hundreds have been optimized. Significant, simple, time reductions are possible for $t \le 2$. It is interesting that imposing fixed stage sizes apparently increases the computing time required, growing exponentially in the number of stages.[16] We are unaware of optimal designs for three or more arms with more than two stages.

FINAL REMARKS

This paper has concentrated on the basic approach of using DP to solve a variety of response adaptive

sampling problems for populations with outcomes that are Bernoulli random variables. The approach can also be used to evaluate the worth of suboptimal designs for the same problems. There are many other desiderata, such as randomization or reducing the number of times a fully sequential design switches treatments, that can also be optimized through DP. However, for populations with more complex distributions, e.g., normal, the computational approaches are only approximations, although they can be made as accurate as desired. In some settings, such as optimizing a minimax objective, DP is apparently not applicable because it is typically based on a linear objective, and in yet other settings, such as finding the maximum of a unimodal function, it is not applicable because there is no natural probability distribution over the space of such functions.

REFERENCES

1. Robbins H. Some aspects of the sequential design of experiments. *Bull Am Math Soc* 1952, 55:527–535.
2. Bellman R. *Dynamic Programming*. Dover paperback edition. Princeton, NJ: Princeton University Press; 1957.
3. Ohemke R, Hardwick J, Stout QF. Scalable algorithms for adaptive statistical designs. *Sci Prog* 2000, 8:183–193.
4. Coad DS. Sequential allocation involving several treatments. In: Flournoy N, Rosenberger WF eds. *Adaptive Designs*. Institute of Mathematical Statistics, Hayward, CA, Lecture Notes 25; 1995, 95–109.
5. Palmer C. Sequential elimination procedures in clinical trials of three Bernoulli response treatments. In: Flournoy N, Rosenberger WF eds. *Adaptive Designs*. Institute of Mathematical Statistics Lecture Notes 25; 1995, 110–123.
6. Thall PF, Simon R, Ellenberg SS. A two-stage design for choosing among several experimental treatments and a control in clinical trials. *Biometrics* 1989, 45:537–547.
7. Hardwick J, Stout QF. Path induction for evaluating sequential allocation procedures. *SIAM J Sci Comput* 1999, 21:67–87.
8. Gittins JC, Jones DM. A dynamic allocation index for the sequential design of experiments. In: Gani J et al. eds. *Progress in Statistics*. North Holland, The Netherlands, 241–266.
9. Stein C. A two-sample test for a linear hypothesis whose power is independent of the variance. *Ann Math Stat* 1945, 16:243–258.
10. Ghurye SG, Robbins H. Two-stage procedures for estimating the difference between means. *Biometrika* 1954, 41:146–152.
11. Ghosh M, Mukhopadhyay N, Sen PK. *Sequential Estimation*. New York: Wiley; 1997.
12. Colton T. A two-stage model for selecting one of two treatments. *Biometrics* 1965, 21:169–180.
13. Canner PL. Selecting one of two treatments when the responses are dichotomous. *J Am Stat Assoc* 1970, 65:293–306.
14. Hardwick J, Stout QF. Determining optimal few stage allocation rules. *Proc 18th Symp Interface* 1995, 27:342–346.
15. Cheng Y, Su F, Berry DA. Choosing sample size for a clinical trial using decision analysis. *Biometrika* 2003, 90(4):923–936.
16. Hardwick J, Stout QF. Optimal few-stage designs. *J Stat Plan Inference* 2002, 104:121–145.

