

This appears in *Journal of Sequential Planning and Inference* 132 (2005), pp. 149–162.

Optimal Screening Designs with Flexible Cost and Constraint Structures

Quentin F. Stout¹ and Janis Hardwick¹

EECS Dept., University of Michigan, Ann Arbor, Michigan 48109, USA

Abstract

We describe a cost- and constraint-based decision-theoretic approach to the design of screening trials, where the goal is to identify promising candidates for future study or to decide whether to accept or reject a product. An algorithmic method for optimizing this approach is presented. This method utilizes a highly flexible structure to reflect a variety of decision and experimental costs and constraints. The designs produced can range from being a single stage up to being fully sequential, depending on the sampling cost functions and constraints. These designs generalize and extend those previously available, often achieving meaningful improvements. This approach can be used for a variety of other problems. Operating characteristics of the designs are also described.

Keywords and phrases: Bayesian, design of experiments, response adaptive sampling, optimal designs, sequential trials, decision theory, dynamic programming, few-stage, stopping rule

1 Introduction

In drug development, it is often necessary to screen through large numbers of compounds prior to locating one that deserves further attention. This situation also arises in acceptance-rejection sampling in which sample products must be tested before an entire batch of product is deemed acceptable. Since there may be virtually limitless compounds, agents, products or “candidates” available for testing, one would like the average screening trial to be quick, inexpensive, and accurate. Most screening designs are developed to satisfy requirements on the false positive (F^+) and false negative (F^-) error rates. However, we believe that it is better to model the goals of a screening process in terms of costs and constraints on obtaining the observations, and costs of making false positive and false negative decisions. Observational costs may include a cost per observation (with perhaps different costs for positive and negative outcomes), set-up costs per stage and so forth. Costs of improper decisions can include opportunities lost or costs of replacing products for false negatives, and costs of conducting more extensive follow-up tests or accepting inferior products for false positives. Experimental constraints may include, among other things, a maximum possible number of observations per stage and a maximum number of stages.

An economic model that incorporates such factors can more accurately reflect the relevant aspects of the entire decision process. Most screening studies are used organizationally to make decisions as to whether to proceed to a later stage. This might be the next stage in a drug evaluation process (clinical or preclinical), or a manufacturing stage utilizing the inspected batch. Later stages require a commitment of resources, just as the study itself requires resources, so it is natural for the organization to attempt to model the economic tradeoffs explicitly. This approach is to be contrasted with, say, those used in later stages of the drug

¹Research supported in part by National Science Foundation grant DMS-0072910.

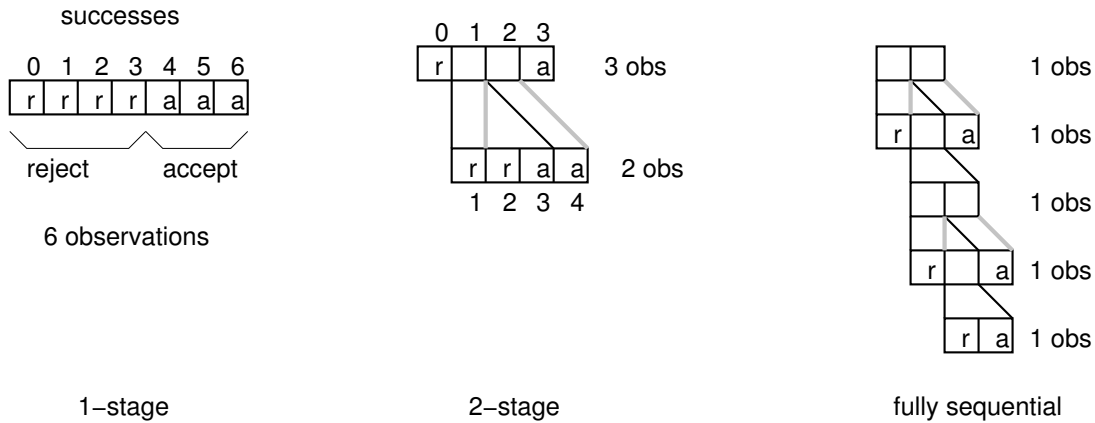


Figure 1: Design Variations in Prior Work

development process in which external approval is required. External approval, such as acceptance by the FDA in the United States, usually does not depend on internal costs but rather on the failure rates. Hence designs for later stages must be crafted to meet requirements on these rates. In Section 5 we illustrate how an economic approach can also be used to optimally achieve required error rates, although this is not the primary goal of this approach.

In the next section we review a sequence of papers addressing screening designs from a Bayesian perspective. Sections 3 and 4 contain descriptions of a more general cost/constraint model along with an approach to obtaining optimal designs for this model. Note that in some settings, such cost models are referred to as loss or risk models, utility models, or decision-theoretic models. In Section 5 we discuss the impact of some specific cost structures and provide comparisons with earlier work. In Section 6 we examine operating characteristics of the designs, and in Section 7 we offer some closing remarks.

Throughout, it is assumed that the response of each candidate is a Bernoulli random variable with an unknown success parameter p_i , $i = 1, 2, \dots$. A Bayesian approach is used, assuming a prior distribution on p_i with mean p . No assumptions are made about the prior distribution, which may be weak or quite peaked. Sometimes researchers are reluctant to utilize a Bayesian approach due to fears that misspecification of the prior will invalidate the design. However, one of the strengths of adaptive designs is that weak priors mostly influence early decisions and the observed data quickly dominate the decision-making process. Further, weak priors will likely yield an adaptive design with very good frequentist properties. In fact, one way for a frequentist to obtain good designs is to use a Bayesian design and then to carry out the analyses of the design from a frequentist perspective. These points are addressed in Section 6.

2 Prior Work

In 1996, Yao, Begg, Livingston [7] proposed 1-stage Bayesian screening designs in which fixed error rates are set. They specify a cut point λ and define a positive candidate to be one with mean success rate $\geq \lambda$. Their false positive rate is the same as our F^+ , namely the probability of making a false positive decision. However, their false negative rate \mathcal{F}^- is the probability that all candidates declared negative until the first positive is declared are indeed negative. Subject to these constraints, they minimize the expected sample size until the first promising candidate is identified.

In 1998, Yao and Venkatraman [8] examined the same problem, suggesting 2-stage designs with fixed stage sizes. On the basis of the observations from Stage 1, one either stops or conducts a second stage whose size is independent of the responses in Stage 1. Since 1-stage designs are a special case of 2-stage designs, the expected total sample size in [8] can be no worse than that in [7], though the worst case size may be. For the examples considered, the expected sample sizes for the 2-stage designs were quite a bit smaller than those for the 1-stage designs.

Also in 1998, Wang and Leung [6] continued the goal of minimizing the expected total sample size until the first promising candidate is identified. However, these authors used fully sequential designs in which one can either stop or continue after each observation and decisions are based on all observations to date.

Figure 1 illustrates the different designs considered in [7], [8] and [6] (from left to right). In each case, the top row of $i + 1$ boxes indicates a first stage of i observations. The numbers above or below the boxes are the number of successes obtained during that stage. The decision rule is described by the “r” (reject) or “a” (accept) in each box. In the 1-stage example, one rejects the candidate if there are fewer than 4 successes and accept otherwise. If a box is blank, then another stage is necessary for that observation. In the 2-stage design, one rejects with zero successes and accepts with 3, but need further information to decide with 1 and 2 successes. There will be 2 more observations taken in each case. The vertical and diagonal lines attached to the lower left and right sides of a box track the possible outcomes and decisions for that result at the next stage. For the 2-stage example with 1 in 3 successes in the first stage, the next two observations can result in a total of 1, 2 or 3 successes in 5 observations. Similarly with 2 in 3 original successes, there can only be 2, 3 or 4 total successes. After this, each option results in a terminal decision.

For a given prior and error rates, fully sequential designs minimize the expected sample size, while time is maximized since the observations cannot be obtained concurrently. To coalesce these goals into a single additive objective function, Wang and Leung introduced a Lagrangian-like control variable for making a false positive identification and another for false negative, and then optimized total cost, where the total cost is the cost of errors plus the sample size. See the comments at the end of Section 3 for a comparison of their approach and ours.

The above papers, and ours, address the decision problem from a Bayesian viewpoint, using a loss function determined by posterior distributions. In contrast, in the far more common frequentist framework one is interested in testing the null hypothesis $H_0: p \leq p_0$ versus the alternative hypothesis $H_1: p \geq p_1$, for $p_0 < p_1$. Frequentist designs have been extensively studied (see [3] and the numerous references therein). However, designs optimal for frequentist maxi-min criteria are not in general optimal for Bayesian criteria, and vice versa, though approaches used for one can sometimes be modified for the other.

3 Model

Since candidates are evaluated one at a time, screening studies can be viewed as a form of stopping rule problem. This is an area in which decision-theoretic approaches are common. We use an integrated decision-theoretic model that incorporates sampling costs as well as decision costs. The sampling procedure can be purely serial, staged, or a mixture, while decision costs are assumed merely to be a function of the prior, observations, and decision made.

While the approach used herein has similarities to some of the prior work, especially that in [6], there are many points of difference, both philosophically and in terms of the designs generated. Some of the significant ones are:

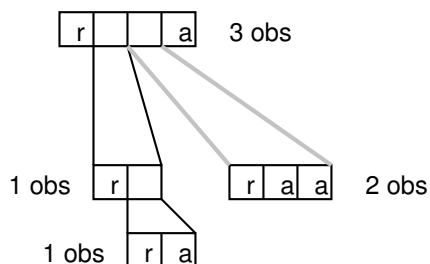


Figure 2: A multistage design with variable stage sizes and variable number of stages

-
- i. Our goal is to minimize the expected cost of a single candidate, $E(C)$, rather than the error rate in finding the first positive candidate. This reduces the problem to classical Bayesian criteria.
 - ii. In the prior work, and in Sections 4 and 5, false decision costs are constants. The costs of false positive or negative decisions are denoted by C^+ and C^- , respectively. However, by considering posterior distributions, rather than error rates, one can model more complex criteria. There may, for example, be a more serious penalty for rejecting positive candidates far above the cut point. This is examined in Section 6.
 - iii. Sampling costs can include a fixed amount per observation, as in the earlier work, and can include a cost per stage or nonlinear costs per observation within a stage. This allows one to address issues such as time or setup costs. For clinical or animal trials one can also impose ethical considerations by weighting failed tests more than successful ones. Similarly, a variety of constraints, such as a maximum number of observations per stage, can be easily accommodated.
 - iv. If a multistage design is generated, then the size of each stage encountered can depend on the observations obtained so far, unlike the fixed second stage size in [8]. Thus the design is more adaptive, permitting further savings as is shown in Table 2.

Figure 2 illustrates a sample design that might be generated by this process. Note that its structure is determined by the costs and that each step is determined by the prior and outcomes observed so far.

Some of these differences could be accommodated by the programs used by previous researchers. For example, changing the cost structure for terminal decisions (point ii) would be an easy matter for the program created by Wang and Leung. However, creating multistage designs with variable stage sizes would be extremely time-consuming using techniques developed previously since they basically search through every possibility (though the process is a bit more sophisticated). A dynamic programming approach does this as well, but organizes the computations so as to be exponentially more efficient.

More generally, a flexible cost-based approach seems to be quite natural, in that it more directly reflects the relevant factors. For example, to specify the desired false positive and negative rates, (F^+ and F^-), one tries, in theory, to take into account the costs of such mistakes and weighs these versus the costs of the screening tests. In practice, this is rarely attempted as an explicit process and often seems to involve rough guesses as to what seems reasonable. Similarly, it is difficult in to know in advance how many stages to use since the data can affect this decision and the stages may have different costs (point iii). Trying to make all of these tradeoffs more explicit, and putting them directly into the optimization phase, should improve the overall decision-making process and the quality of the results. As part of this overall process, the investigator may wish to see simulated results for different cost structures in order to gain some familiarity with the give

and take among design attributes. A common example of this is the tradeoff between the maximum and expected sample sizes of the design.

Note that a design that optimizes the expected sample size for a single candidate, for a given prior and fixed F^+ and F^- , is also a design that optimizes the expected total sample size until the first positive is produced. To see this, first note that fixing the error rates fixes the probability that the decision for a candidate will be positive. Let α be the probability that a candidate will be declared positive, let N^+ be the expected sample size for a candidate declared positive, and let N^- be the expected sample size for a candidate declared negative. Minimizing $E(N)$ per candidate is thus the problem of minimizing

$$\alpha N^+ + (1-\alpha)N^-. \quad (1)$$

To minimize the total sample until a positive candidate is identified, the expected number of candidates examined is $1/\alpha$, and thus the goal is to minimize

$$\left(\frac{1}{\alpha} - 1\right) N^- + N^+ = \frac{1}{\alpha} [(1-\alpha)N^- + \alpha N^+]. \quad (2)$$

Since α is fixed, minimizing (1) is equivalent to minimizing (2).

Converting the optimization goals into those for a single design is critical to the work presented here, simplifying the algorithm and clarifying the role of the costs. This equivalence was noted in 1999 [5], where the algorithm in the appendix first appeared, and independently in 2001 by Wang and Leung [4]. There are many techniques for optimizing a single experiment, but for a sequence of experiments. An example of this is the contrast between the dynamic programming used here and the algorithm in [6]. The complexity of optimizing a sequence of experiments requires them to iteratively compute a fixed point similar to the methods used to determine Gittins indices. Further, their costs are merely control variables and do not correspond to true costs, making it difficult to extend to concerns such as a cost per stage.

4 Optimization

In the appendix there is an algorithm that optimizes a basic cost/constraint model. The algorithm utilizes constraints on the sampling, namely a maximal sample size (N_{max}), a maximal number of stages (R), and a maximal number of observations in a single stage (N_r). These are used to bound the design space that is being searched. In many settings these constraints are meaningful, but in others they are just program bounds and the user needs to pick sufficiently large values to insure that the optimal design is generated. When the constraints are not naturally generated one could use bounding techniques to automate the process, but that has not been necessary because liberal bounds can be utilized and the computations will still complete in a minute or so. Within a single stage, `samplecost(i)` gives the cost of making i observations. This need not be linear in i .

The algorithm determines the optimal result at each possible point in an experiment. `cost(s, n, r)` records the optimal cost, and `dec(s, n, r)` records the optimal action, when one has already conducted r stages, having obtained a total of s successes among n observations. Thus the expected total cost of the experiment is `cost(0, 0, 0)`. To recover the design created, one uses `dec(0, 0, 0)` to determine the first action. If it indicates that one should make i observations in the first stage, then `dec(s, i, 1)`, $0 \leq s \leq i$, will indicate the appropriate actions in the subsequent stage, and so on.

Note that while the number of observations and the number of successes form a sufficient statistic, the state space has been expanded to include the number of stages. This is needed because the same sufficient

statistic could be obtained by different paths through the design which took a different number of stages. Since the continuation possibilities and costs may depend on the number of stages, the additional stage information is needed.

Many multistage designs, such as those in [2, 8], are constrained to have a fixed number of stages. Instead we specify a maximum number of stages and let the final number be determined by the optimization process. The algorithm will produce 1-stage designs as in [7] if one sets $R = 1$, $N_r = N_{max}$ and makes the sampling cost linear. It will produce fully sequential designs as in [6] if one sets $N_r = 1$ and $R = N_{max}$. However, it cannot produce the 2-stage designs of [8], because of their constraint that the 2nd stage, if reached, always be the same length. Straightforward modifications would permit the optimization of such designs, but if one wanted to produce k -stage designs with fixed stage sizes, $k > 2$, then the computations, while straightforward, would rapidly increase with k . This point is discussed in [2].

5 Results

The program was run for a variety of different cost structures. In some cases, we set parameters to match those in [8] and compare to previous results. In others we examined tradeoffs among cost factors. Results labeled as representing an infinite number of stages are the fully sequential solution. At some point sampling costs insure that no further observations are worthwhile, so for the calculations we merely used large values of N_{max} and verified that the maximum sample size utilized was far smaller.

Table 1 contains results for models in which there is a unit cost for each item tested and no cost per added stage. The error costs, C^+ and C^- , are equal and take on values 500, 1000, 2000, 4000. In Figure 3, the number of stages is plotted against $E(C)$ for the four sets of $C^+ = C^-$ (indicated under “FNC=FPC” to the right of each curve). As one would expect, total design costs diminish as stages are added. The final column in Table 1 shows the proportional increase in $E(C)$ using an i -stage design rather than the optimal fully sequential one, $i = 1, 2, 3$. Naturally, adding costs per stage can reorder such results.

While we expect an overall trend of reduction in $E(N)$, F^+ and F^- as the number of stages increases, there is no guarantee that all of the values will decrease monotonically. It may be more beneficial overall to increase one cost slightly to obtain a major reduction in another. Finally, note that, while the sample error costs appear to be quite large, they generate only moderate sample sizes. Note also that as error costs increase, F^+ and F^- decrease while $E(C)$ and $E(N)$ increase. Since the optimal design for one set of costs is considered among the designs for optimizing using another set of costs, it is easy to prove that $E(C)$ is strictly increasing in either error cost, and $E(N)$ is nondecreasing.

Table 2 provides $E(N)$ when using the optimal 2-stage design vs. the fixed version in [8]. The parameters in Table 2 are a subset of those appearing in Table 1 of [8]. Two decision cut points are used, 0.3 and 0.6, and the prior is specified via the mean and variance of a beta distribution. The variance is fixed at 0.08 for the entire table. The “Opt $E(N)$ ” and “Fixed $E(N)$ ” columns contain the sample sizes expected when using our and their 2-stage programs respectively. The last column shows the proportion increase in $E(N)$ required when using the design constraints in [8]. There is one unusual configuration in which the fixed design $E(N)$ is only 4% larger than the optimal one, apparently due to an extremely accurate match of their fixed design to the target error rates.

Table 2 is not a natural comparison because our approach minimizes total costs (including error rates and sample size) rather than minimizing sample size with fixed error rates and stage sizes. To obtain these results, we adjusted costs to achieve the error rates used in [8] ($F^+ = F^- = 0.1$). Thus costs can be viewed as control variables whose manipulation can yield optimal designs with more classical objectives. Further, the approach allows one to produce designs far more complex than could reasonably be produced

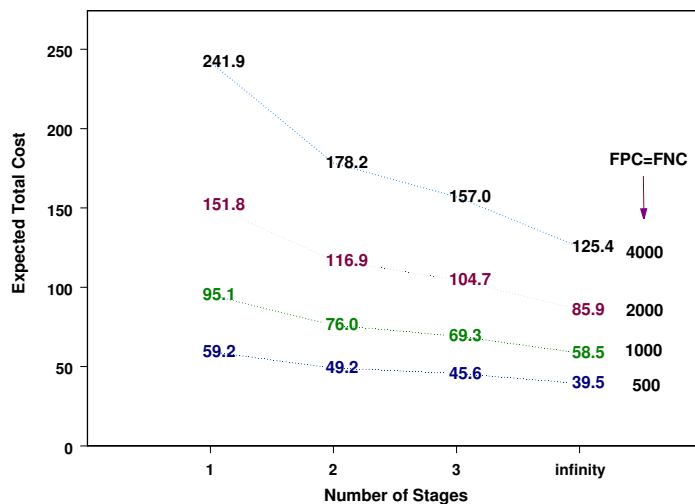


Figure 3: Total Cost versus Number of Stages; Prior = $\text{Be}(1,1)$, Cut point = 0.7

by standard techniques. For example, the designs of [8] require the determination of only 5 parameters: the sizes of each stage, the minimal number and maximal number of successes with which one can stop at the end of stage 1, and the minimal number of successes needed in stage 2 to decide positively. If one allows the second stage size to depend upon the outcome of the first stage, then the number of possible designs increases exponentially. This makes it infeasible to optimize by exhaustive search, even when combined with branch-and-bound pruning mechanisms. Reducing to a single objective (cost) enables the use of dynamic programming, which constantly prunes useless options, dramatically reducing the computations required.

6 Operating Characteristics

The designs are optimized with respect to given priors, and operating characteristics such as $E(N)$ are determined with respect to these priors. Additional evaluations that one may desire include

- Bayesian: robustness against misspecification of priors.
- Frequentist: pointwise determination of costs and F^+ , F^- rates.

The following evaluations are all exact and were obtained using path induction [1].

To illustrate robustness, suppose we specify a unit cost per observation, $R = 2$ stages, $C^+ = C^- = 1000$, and cut point $\lambda = 0.7$. In Table 3, a design was created assuming a prior of $\text{Be}(1,1)$, and then its properties are evaluated for a prior of $\text{Be}(3,3)$. A second design was created reversing the roles of the priors. Note that each design, when evaluated via the other prior, is quite similar to the optimal design for that prior. That is, the designs appear quite robust. One of the reasons for this is that the adaptive nature of 2-stage designs allows the second stage to incorporate information collected during the first.

Pointwise operating characteristics of the two designs in Table 3 are shown in Figure 4. Despite the robustness of the designs, they also clearly differ.

All of the previous designs, and the designs in prior work, were optimized for cost models in which the transition from false negative to false positive costs is a step function. While this is natural if the primary

Error Cost	Number		$E(C)$	$E(N)$	F^+	F^-	Increase over Opt
	Stages						
500	1		59.2	19.0	0.13	0.06	50%
	2		49.2	18.3	0.10	0.05	25%
	3		45.6	17.0	0.09	0.04	14%
	∞		39.5	17.1	0.07	0.04	0
1000	1		95.1	29.0	0.11	0.05	62%
	2		76.0	28.7	0.08	0.04	30%
	3		69.3	28.1	0.06	0.03	19%
	∞		58.5	25.9	0.05	0.02	0
2000	1		151.8	49.0	0.09	0.04	77%
	2		116.9	45.5	0.06	0.03	36%
	3		104.7	43.1	0.05	0.02	22%
	∞		85.9	38.6	0.04	0.02	0
4000	1		241.9	79.0	0.07	0.03	93%
	2		178.2	68.7	0.04	0.02	42%
	3		157.0	65.6	0.04	0.02	25%
	∞		125.4	56.1	0.03	0.01	0

Table 1: Prior = Be(1,1), Cut point = 0.7, $C^+ = C^- =$ Error Cost

Cut point	Mean	Opt $E(N)$	Fixed $E(N)$	Inc over Opt
0.3	0.2	28.70	35.00	22%
0.3	0.3	24.20	32.80	36%
0.3	0.4	13.46	18.30	36%
0.3	0.5	7.10	7.90	11%
0.6	0.2	65.93	77.10	17%
0.6	0.3	96.35	100.60	4%
0.6	0.4	70.62	81.50	15%
0.6	0.5	37.74	44.50	18%

Table 2: 2-Stage: Prior is beta with given Mean and Variance = 0.08, $F^+ = F^- \leq 0.1$

measures are false positive and false negative rates, it is far less natural in a real cost model. Presumably the expected value of a candidate with success probability just barely over the cut point is less than that of one with high success rates. Economically, the cut point should be determined by determining where the expected benefit of advancing to the next round in the process equals the expected loss of rejecting the candidate.

With this viewpoint, it is natural to have the costs of erroneous decisions converge to zero at the cut point. Figure 5 illustrates such a cost structure, modifying that used in the previous examples, and the resulting changes in properties of the design. These designs assume uniform priors. Note that the revised decision cost model significantly reduces the sample size near the cut point.

7 Final Comments

We have described a unified approach for generating optimal multistage designs that can incorporate a variety of costs and constraints associated with conducting experiments. Stopping or decision costs are generalized and need not correspond merely to rates of false positives and false negatives; e.g., distance

Design Prior Be(1,1)				Design Prior Be(3,3)			
cost	sample size	F^+	F^-	cost	sample size	F^+	F^-
76	28.7	0.076	0.036	90	30.4	0.145	0.045

Evaluation Prior Be(3,3)				Evaluation Prior Be(1,1)			
cost	sample size	F^+	F^-	cost	sample size	F^+	F^-
94	31.9	0.199	0.034	79	33.2	0.051	0.049

Table 3: Bayesian Robustness

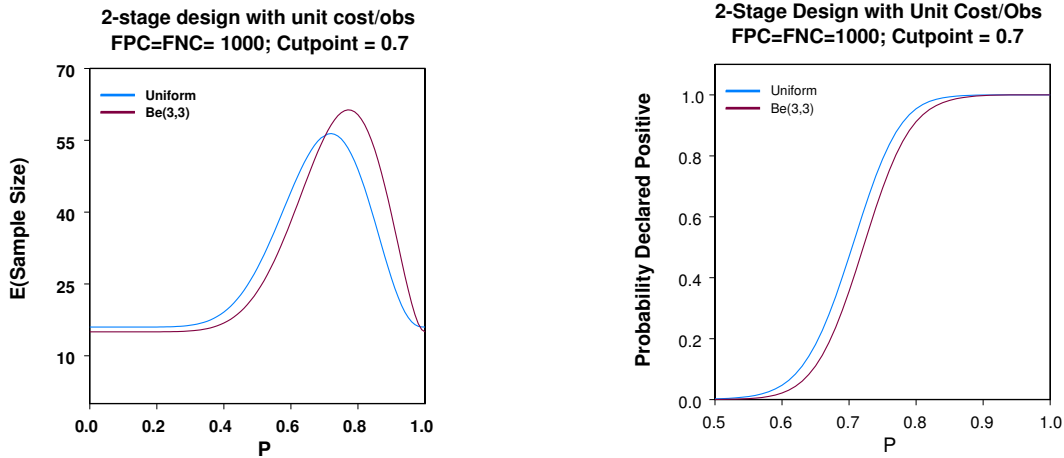


Figure 4: Pointwise Operating Characteristics

from cut point can be incorporated. Such flexible options permit designs that are not artificially constrained to fit asymptotic analysis or an overly simplified structure.

One important step in this work was the replacement of a criterion for a sequence of designs with an equivalent one for a single design. A second was the use of a single objective, minimizing cost, rather than trying to achieve multiple requirements concerning false decision rates. These two factors, coupled with the Bayesian framework, enabled the use of dynamic programming to search through the exponential number of possible designs in only polynomial time. For many applications the most important use of this may be the optimization of 2- or 3-stage designs where the size of each stage is allowed to depend on the outcomes observed previously. Such optimizations appeared in [2] in a more complex setting, though there was no incorporation of sampling costs nor early stopping. Even with just 2 stages, response adaptive stage sizes can provide useful improvements (Table 2). While these designs may be more complex than standard few-stage designs, one can use the simplest of hand-held computers to store the design and guide the experiment.

The approach used here has wide applicability. Since it focuses on a single experiment, the approach produces optimal designs for situations where there is but a single candidate to be tested, as in some group sequential testing problems. Further, since there is no constraint on the structure of the decision costs the approach can be used for quite different Bayesian problems such as parameter estimation. Also, as Table 2 shows, one can use costs as control variables to optimize designs for objectives that do not have a cost basis. In forthcoming work, we will show that this approach can also be applied to hypothesis testing problems to

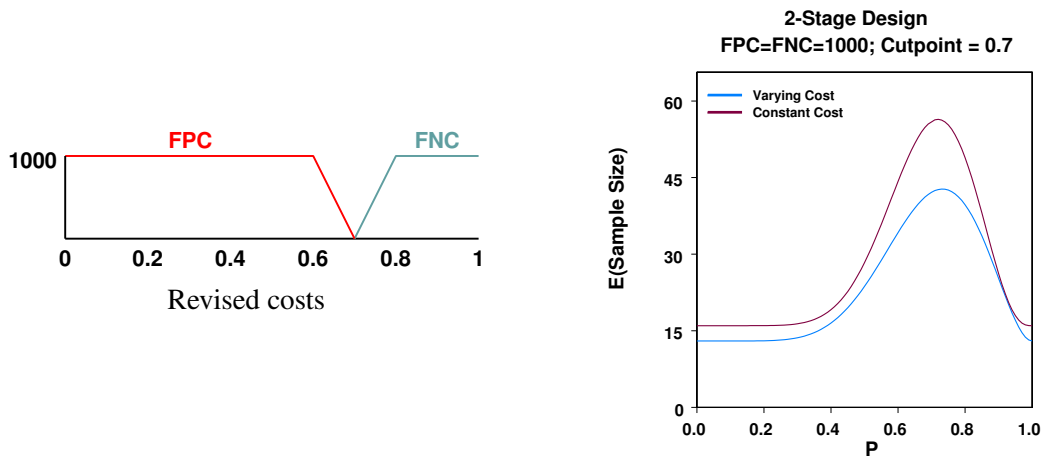


Figure 5: Effect of a more realistic cost structure

attain classical frequentist objectives with optimal operating characteristics.

There are numerous changes in cost and constraint structures that can easily be accommodated by the algorithm in the Appendix, or by small changes thereto. For example, if there is a cost for setup in a stage, which could include costs due to the time for the delayed decision, then the `samplecost` array may include an initial cost plus costs linear in the sample size. With small modifications, this could be extended to permit increasingly severe penalties for later stages, without the complete prohibition that constraints impose. There may also be differential costs for successes versus failures, which can be accommodated either in the terminal costs or by adjusting the calculation of `costnr`.

The approach here assumes a prior distribution on the success rates of different candidates, and then optimizes with respect to the prior. In a setting where this is a long-term process, one might periodically use results from previous candidates to adjust the prior. Determining a new design for this new prior is quite straightforward since the computational time is so small.

We believe that decision-theoretic approaches, with flexible cost and constraint structures, allow one to address important considerations in the design of experiments. With an ever increasing occurrence of new experimental scenarios, each with its unique characteristics, the more modeling flexibility an approach permits the more useful the designs will be.

References

- [1] Hardwick, J. and Stout, Q.F., “Using path induction to evaluate sequential allocation procedures”, *SIAM J. Scientific Computing* **21** (1999), pp. 67–87.
- [2] Hardwick, J. and Stout, Q.F., “Optimal few-stage designs”, *Journal of Statistical Planning and Inference* **104** (2002), pp. 121–145.
- [3] Jennison, C. and Turnbull, B.W., *Group Sequential Methods with Applications to Clinical Trials*, Chapman and Hall/CRC (1999).

- [4] D.H.-Y. Leung and Y.-G. Wang, “Optimal designs for evaluating a series of treatments”, *Biometrics* **57** (2001), pp. 168–171.
- [5] Stout, Q.F. and Hardwick, J., “Minimizing the costs of screening trials”, *Computing Science and Statistics* **31** (1999), pp. 440–444.
- [6] Y.-G. Wang and D.H.-Y. Leung, “An optimal design for screening trials”, *Biometrics* **54** (1998), pp. 243–250.
- [7] T.-J. Yao, C.B. Begg, and P.O. Livingston, “Optimal sample size for a series of pilot trials of new agents”, *Biometrics* **52** (1996), pp. 992–1001.
- [8] T.-J. Yao and E.S. Venkatraman, “Optimal two-stage design for a series of pilot trials of new agents”, *Biometrics* **54** (1998), pp. 1183–1189.

Appendix

The algorithm in Figure 6 first appeared in [5]. While the objective is to minimize costs, the algorithm uses constraints on the sampling, namely N_{max} , R , and N_r , to restrict the design space that is being searched.

The `dec` array records the optimal action at each situation, and `cost` gives the associated cost for performing this action and proceeding optimally. These arrays are calculated via dynamic programming, working from the end of the trial towards the beginning. To compute `dec(s, n, r)` and `cost(s, n, r)` there are 3 options: stop and declare the outcome positive, stop and declare it negative, or continue sampling following an optimal strategy. The optimal strategy is to take the option of minimal cost, i.e.,

$$\text{cost}(s, n, r) = \min \begin{cases} \text{term cost if decide positive} \\ \text{term cost if decide negative} \\ \min\{\text{cost if sample } i \mid i \text{ permissible sample size at } (s, n, r)\} \end{cases}$$

When the trial is stopped, the terminal decision depends only on the number of successes and observations, not the number of stages, and thus the `termcost` array does not need to be indexed by the number of stages. This allows it to be computed only once before the stages are gone through, and then it is utilized in each stage. The optimal terminal decision, and associated decision cost, is computed by comparing the expected cost of each of two decisions and choosing the least costly one. Note that this is not necessarily the outcome most likely. That is, it may be that at a terminal state it is slightly more likely that the candidate is positive than negative, but because the costs of false positives are very high one should reject the candidate.

If additional sampling is permissible, the costs of sampling versus stopping are determined by considering all sampling options (i.e., number of observations possible at this stage) and choosing the one with minimal expected cost assuming one proceeds optimally. Note that the expected cost depends on the probability of reaching each state at the next stage, and the expected cost of optimally continuing from that state. Thus the expected costs of sampling at a point (s, n, r) depend on costs of the form `cost(*, *, r+1)`. Since the dynamic programming approach has already determined these costs, it is possible to determine the expected sampling cost at the current state.

The algorithm is quite general and makes no assumptions about the priors. All aspects of the priors are encapsulated in the `prob` array. For the examples in this paper all priors were beta distributions, which

greatly simplifies the calculations. For a prior $\text{Be}(a, b)$, the probability of getting t successes out of m observations, given that s successes have already been observed out of n observations, is

$$\text{prob}(t, m; s, n) = \binom{m}{t} \frac{(s+a)^t (b+n-s)^{m-t}}{(a+b+n)^m}, \quad \text{where } x^{\bar{y}} = \prod_{i=0}^{y-1} (x+i)$$

N_{\max} : maximum sample size

R : maximum number of stages

N_r : maximum observations in single stage

samplecost(i): cost of i observations in one stage

termdec(s,n): decision if stop with s suc and n obs

termcost(s,n): cost of stopping with s suc and n obs

prob(sr,nr; s,n): probability of sr suc from nr obs, given that s suc have occurred from n obs

cost(s,n,r): optimal cost of completing trial, if have s suc, n obs, r stages

dec(s,n,r): optimal decision

costnr: given current state, cost of sampling nr

{determine costs and decisions for final stage}

do $n = 1, N_{\max}$

do $s = 0, n$

 cost(s,n,R) = termcost(s,n)

 dec(s,n,R) = termdec(s,n)

{determine costs and decisions for remaining stages}

do $r = R-1, 0, -1$

do $n = 1, N_{\max}$

do $s = 0, n$

 tempcost = termcost(s,n)

 tempdec = termdec(s,n)

 do $nr = 1, \min(N_r, N_{\max} - N_r)$ {for all possible sample sizes}

 costnr = samplecost(nr) + $\sum_{sr=0}^{nr}$ prob(sr,nr; s,n) · cost(s+sr,n+nr,r+1)

 if costnr < tempcost

 tempcost = costnr

 tempdec = nr

 cost(s,n,r) = tempcost

 dec(s,n,r) = tempdec

Figure 6: Optimization Algorithm
