Dynamic Resource Allocation via Objective Function Approximation

Guowei Sun, Dingyuan Xu

December 16, 2014

Abstract

Resource allocation deals with the problem of allocating a limited simulation budget to a number of alternatives to achieve a better probability of correct selection.For OCBA and the usual equal allocation strategy, simulation results on each alternative are treated to be independent, ie, one simulation result on one alternative will not change our belief about any other alternatives. But, we know that in general the objective function is usually a almost everywhere differentiable and convex function with respect to design parameters. Considering this fact, we propose a method which approximates the objective function and dynamically allocates the simulation budget. We show through simulation that our method works better under very limited budget for practical problems.
Keywords: Resource Allocation, Least Square,Objective Function Approximation

1 Motivation

In stochastic optimization and ranking and selection area, we are in general dealing with the problem

$$\theta^* = \underset{\theta\in\Theta}{argmin} J(\theta)$$
$$J(\theta) = E(T(\theta))$$
$$\Theta = \{\theta_1, ...\theta_K\} for\ ranking\ and\ selection$$

Ranking and selection is in general easier than optimization over a continuous space, because the search space is much smaller.

1.1 Assumptions on Objective Function for Stochastic Optimization

Stochastic approximation is the stochastic version of gradient based search in deterministic optimization problems. Where the update is performed as

$$\theta_{n+1} = \theta_n + a_n g(\theta_n)$$
$$g(\theta_n) = \frac{dJ(\theta)}{d\theta}|_{\theta_n}$$

For the SA algorithm to converge to the optimal value, the objective function $J(\theta)$ should be differentiable with a single minimum point. An example of a nice function would be shown in figure1.

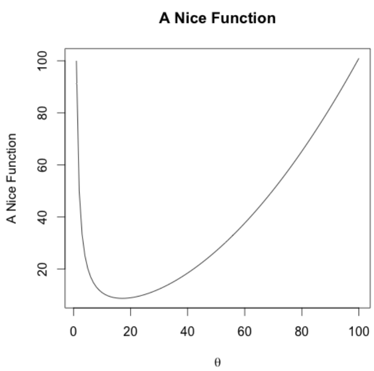

Figure 1: A Nice Function Which SA Could be Applied

1.2 Learning about all Alternatives from each simulation result

This is in fact a very strong assumption on the system under study. If we are facing with the scenario as below, where we have enough simulation results on alternative $\theta_1, \theta_2, \theta_4$ and θ_5.

The key motivation from the regular function

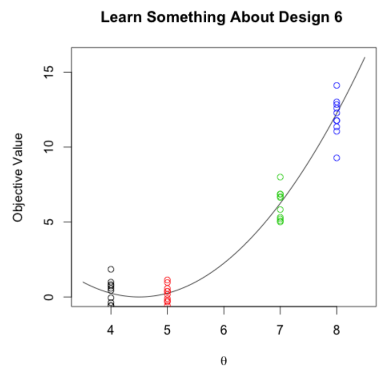

Figure 2: We Should be able to Lean Something about Design 6 though there is no simulation allocated to it

assumption is that, we should be able to learn about all alternatives with observations on any of the alternative. In optimal learning community, where Bayes approach is applied, this is done through the correlation structure designed. But here, we propose the method of estimating the objective function as the way to consider all observations on all alternatives.

1.3 The Problem with OCBA

The OCBA follows an asymptotically optimal allocation policy derived using a bayes approach. But it still requires significant amount of experiment or observations on each alternative to properly estimate the mean and variance of each design. The examples given in the original paper are designed to be difficult problems which intrinsically need large amount of information to make a good decision. While real problems are much less difficult with more expensive information. Also, practically, if you can do 400 simulation, there is usually no problem in doing 2000. The simple rule of equal allocation will perform pretty well.

In general, we want to find a way to select from finite alternatives using very small simulation budget. Like in drug design problems, experiments or observations are very very expensive. We still start from designed problems, and then we move onto optimize more realistic problems using the queueing system we have already simulated.

2 Objective Function Approximation

We design the basis for estimating objective function to be

$$\hat{J}(\theta) = a\theta^2 + b\theta + c + \frac{d}{\theta}$$

the last term is to make the function less symmetric. This would be a a.e differentiable and convex function. It only covers a single family of functions, but we show later in our simulation experiment that this basis design is sufficient for our purpose.

We estimate the parameters through the least square method using observations on all the alternatives

$$\hat{\beta} = \underset{\beta}{argmin} \sum_{i=1..K} \sum_{j=1..n_i} (y_{ij} - \hat{J}(\theta;\beta))$$

$$\beta = (a, b, c, d)$$

$$i = 1...k, alternatives$$

$$j = 1...n_i, \textit{number of observations on alternative } i$$

The computational setup are as follows:
we have observed

$$y_{ij}, i = 1, .., k, j = 1, .., n_i$$

The corresponding feature matrix is

$$\Phi = \begin{bmatrix} \theta_{11}^2 & \theta_{11} & 1 & \frac{1}{\theta_{11}} \\ \theta_{12}^2 & \theta_{12} & 1 & \frac{1}{\theta_{12}} \\ & . & & \\ & . & & \\ & . & & \\ \theta_{kn_k}^2 & \theta_{kn_k} & 1 & \frac{1}{\theta_{kn_k}} \end{bmatrix}$$

where $\theta_{ij} = \theta_i$ for all $j = 1, 2, ..n_i$

At each step, the estimated parameters are

$$\hat{\beta} = (\Phi^T\Phi)\Phi y$$

Each time, we assign one single simulation to alternative i, then we add a row to the feature matrix, and get a new observation

$$\Phi_{N+1} = (\theta_i^2, \theta_i, 1, \frac{1}{\theta_i})$$

$$y_{N+1} = y_{i,n_i+1}$$

$$N : \textit{Total Number of Observations on all Alternatives}$$

The recursive updates for least square function approximation update is then given by

$$\beta^{(N+1)} = \beta^N + \frac{y_{N+1} - \Phi_{N+1}\beta^N}{1 + (\Phi^{N+1})^T B^N (\Phi^{N+1})} B^N$$
$$B^{N+1} = B^N - \frac{B^N \Phi^{N+1} (\Phi^{N+1})^T (B^N)^T}{1 + (\Phi^{N+1})^T B^N (\Phi^{N+1})}$$

During the ranking and selection procedure, the only thing to keeping track of are these matrix and vectors. There is no matrix inverse involved, therefore the simulation cost is acceptable.

3 Resource Allocation Procedure

We consider three objectives when assigning simulation budget to each alternative

- Simulate more on the current best alternative to verify its optimality
- Achieve a more accurate estimation of the objective function
- Simulate the potentially good candidate

So, with the current observations at hand, we should select the next generation of alternatives to simulate, we propose a selection scheme as follows

- Simulate the alternative with the best estimated objective value

$$\theta^{(1)} = \underset{\theta \in \Theta}{argmin} \hat{J}(\theta)$$

- Simulate the alternative with the best current mean

$$\theta^{(2)} = \underset{\theta \in \Theta}{argmin} \bar{Y}$$

- Simulate the alternative with the second best estimated objective value

$$\theta^{(3)} = \underset{\theta \neq \theta^{(1)}}{argmin} \hat{J}(\theta)$$

- Simulate the alternative with the biggest average bias from the estimated objective

$$\theta^{(4)} = \underset{\theta \in \Theta}{argmin} \frac{1}{n_i} \sum_{j=1..n_i} Y_{ij} - \hat{J}(\theta)$$

So after the selection procedure, we at most could have 4 more alternative to simulate. But $\theta^{(1)}, \theta^{(2)}$, and $\theta^{(3)}$ have a good chance to coincide. Therefore, we would most likely assign simulation runs to 2 or 3 candidates. With our selection rule designed, our algorithm runs as follows

- *Simulate one observation on all alternative i=1,2,..k, and get an estimate of the objective function*
  - *select out the next generation of candidate to simulate*
  - *estimate the objective function*
- *Select the alternative with the best estimated objective value*

This algorithm is trying to exploit more on the current and potential good candidates while trying to get a better estimate of the objective function.

A typical selection procedure is shown on figure 3, performed on the example described in section 4. In this case, the estimated function fits the real objective function very well, and we indeed select out the best alternative.

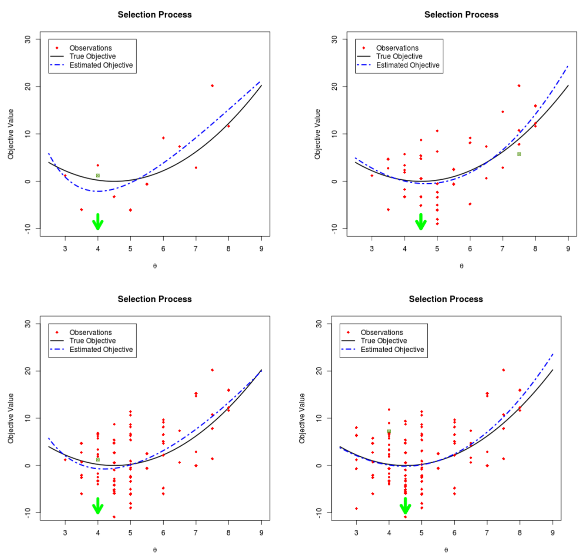

Figure 3: The Selection Process

4 An Experiment

We design a relatively difficult problem following the same idea from the original OCBA paper.

the alternatives are

$$\theta = 3, 3.5, 4, 4.5, 5, 5.5, 6, 6.5, 7, 7.5, 8$$

the objective function is

$$J(\theta) = (\theta - 4.5)^2$$

the observation on each alternative is a R.V.

$$Y = J(\theta) + 5Z, with\ Z \sim N(0, 1)$$

The best alternative would be $\theta_3 = 4.5$ with an objective value 0. But the second best is $\theta_2 = 4$ and $\theta_4 = 5$ with objective value 0.25. We are trying to distinguish the difference of 0.25 from noise with variance 25. Which is very difficult under limited simulation budget.

We compare our procedure with two other procedures

- Equal Allocation, where simulation budget is allocated to each alternative equally
- OCBA, where a initial simulation of n_0 on each alternative is required to estimate the variance and bias to perform the resource allocation

4.1 Equal Allocation

This is the easiest approach. We allocate our simulation budget equally, and then choose the alternative with the best mean. For this approach, the probability of correct selection for 10000 simulation budget is only around 0.78.

4.2 Sequential OCBA Procedure

OCBA procedure requires a relatively large simulation budget, because it needs between 5 20 initial simulations to estimate each mean and variance(reference here).And some further simulation budget.This means we would at least 100 simulation budget to perform OCBA procedure. But our proposed method could work with only 20 simulation budget.

Still, we use OCBA as a bench mark to test the strength of our method under large simulation budget.

OCBA procedure performs as follows

Step0. Perform n_0 simulation replications for all designs i;

Step1. If $\sum_{i=1}^{k} N_i > T$,stop;

Step2. Increase the computing budget by Δ and compute the new budget allocation $N_1^{l+1}, N_2^{l+1}...N_k^{l+1}$ using the formula

$$(1)\frac{N_i}{N_j} = (\frac{\sigma_i/\delta_i}{\sigma_j/\delta_j})^2, i \neq j \neq b$$

$$(2)N_b = \sigma_b \sqrt{\sum_{i=1,i\neq b}^{k} \frac{N_i^2}{\sigma_i^2}}$$

Step3. Perform additional max(0,$N_i^{l+1} - N_i^l$) simulations for design i.
Go to step 1.

We apply the rule for selecting the Δ and n_0 to be $\Delta = (N - kn_0)/3$,$n_0 = min(N/4, 5)$.

4.3 Experiment Result Comparison

We propose our experiment as follows:

- Use the above designed problem for testing
- simulate for simulation budget

  $$40, 60, 80, 100, 120, 160, 200, 240,$$
  $$280, 320, 360, 400, 650, 700, 1000, 2000,$$
  $$3000, 4000, 5000, 6000, 7000, 8000$$

- For each simulation budget, run 1000 selection procedures using all three approaches.

The simulation result is shown on figure 4.

From the figure, we can see that our algorithm dominates OCBA and EQ algorithm at all simulation budget. Our algorithm would achieve probability 1 of correct simulation using simulation budget as small as 4000. While at 8000,OCBA still have not reached 1 yet. In the designed problem, our algorithm is orders of magnitude better than OCBA. But this is only a designed example with very ideal properties

- Unbiased Objective Function: the designed objective function $J(\theta) = (\theta-4, 5)^2$ could be accurately estimated with our designed basis function. Which could almost never happen in real case.

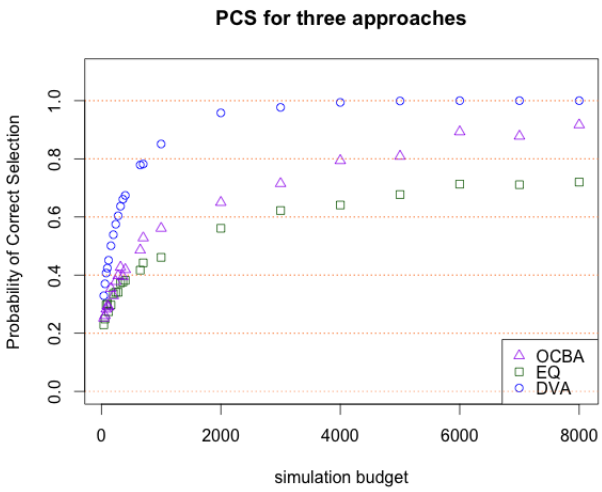

Figure 4: Probability of Correct Selection on the Designed Problem

- Equal Variance: In our algorithm, we allocate some of our simulation budget to the alternatives with observations far away from the estimated objective. This is desirable with equal variance case because biased observation could mean we are not estimating the objective well enough. But if we do not have equal variance, it could possibly be due to the larger variance on specific alternatives.

5 Application to M/M/c/c queueing system

Our method works well with the designed example. We now apply our method to one of the well studied system in during the course, a M/M/c/c queueing system.

5.1 M/M/c/c Queue

Here we study a M/M/c queue, where there is no waiting space. If a customer arrives and sees no available service agent, the customer would leave. Each served customer will bring in some profit. But, placing a trained agent would also bring some cost. This situation well describes some real systems such as a gas station, where people usually do not wait for available service stations

5.2 General Set Up of The Problem

Objective: maximize the expected eight hour profit

$$J(l) = pE(N(l)) - cl$$
$$l : number\ of\ servers$$
$$N : expected\ number\ of\ served\ customers$$
$$p : profit\ from\ one\ customer$$
$$c : cost\ of\ training\ an\ agent$$

The queueing system:

- A, inter arrival time, follows an exponential distribution with rate 5 per hour;
- X, service time, follows exponential distribution with rate 1 per hour;

Simulation Specification:

$$simulate\ for\ 8\ hours$$
$$p = 1$$
$$c = 4$$
$$choose\ between\ alternatives$$
$$\theta = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)$$

This will give us a reasonable problem to experiment on.

5.3 Analysis of the System

Before doing any selection over the alternatives, we investigate on the described problem to create a bench mark for further research.
For each alternative, we simulate 10000 independent replications.
The number of served customers in 8 hours is increasing with the number of servers, which is expected. And at 10, the expected value is around 40, meaning almost all customers are served.
The variance in the number of customers served seems to be a linear function of number of servers. It is very different between different alternatives.
The profit reach its maximum at alternative 5, meaning 5 servers would bring in optimal profit.
For this problem, we are trying to find an optimal expected profit of 10.48 with variance about 16. While the second best alternative is 10.04 with variance 13.6. This is still a very

difficult problem in terms of doing the correct selection.
The information about the designed system is

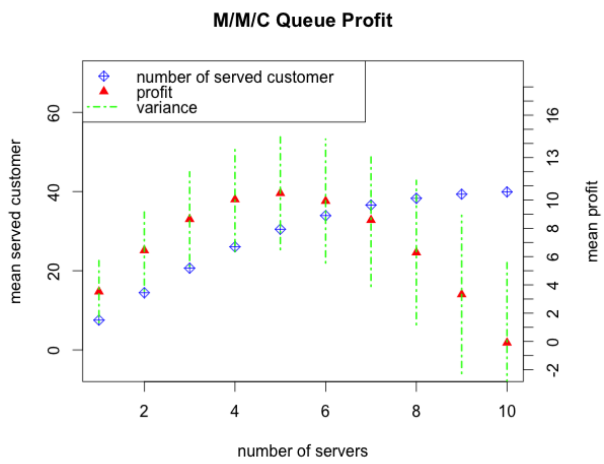

Figure 5: Information about M/M/C System

5.4 Application of OCBA,EQ and DVA Selection Procedure

This is a real problem, and we assume that the experiment or simulation cost is very very high. Therefore, we compare the probability of correct selection under the simulation budget

$$(40, 80, 120, 160, 200, 240, 280, 320, 360, 400)$$

We simulate 1000 times for each budget and compare the PCS From the figure we can see

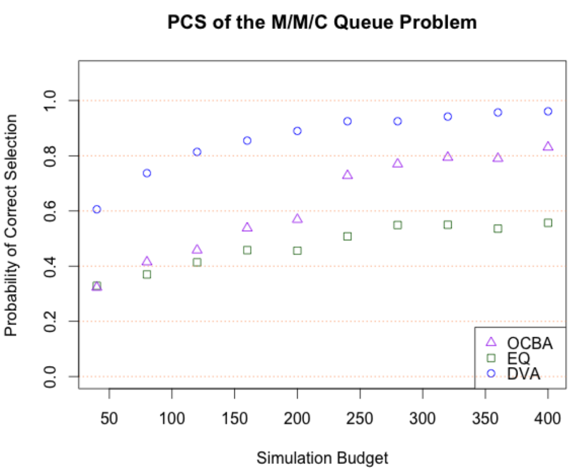

Figure 6: PCS for M/M/c/c Problem

that DVA outperforms OCBA and EQ procedures.

6 Application to (s,S) inventory control problem

Here we apply our selection method to an intentionally designed (s,S) inventory problem.

6.1 Statement of the System

We deal with a (s,S) inventory system with backlogging allowed. The important random variables for the systems are specified as

Notation	Specification	Value
d_i	inter demand time	$U[0,1]$
L_i	order lead time	$U[0,1]$
D_i	Demand	$Exp(\frac{1}{5})$
h	weekly holding cost	0.10
p	weekly storage cost	10
c	order set up cost	10
m	ordering cost	0.10

The cost would be calculated as

$$\begin{aligned} Z =& h\int I(t)^{+}dt - p\int I(t)^{-}dt \\ &+ c\sum_{reviews} I_{\{X_i>0\}} + m\sum_{reviews} X_i \\ &Z : total\ cost \\ &X_i : order\ amout \\ &I(t) : inventory\ level \\ &I(t)^{+} : max(0, I(t)) \\ &I(t)^{-} : min(0, I(t)) \end{aligned}$$

There are two parameters to optimize, s and S. Here we set $s = 20$, and choose from a set of S to minimize our cost. The intuition is that, at lower S, the ordering cost would dominate because you have to constantly make large orders. At large S, the holding cost would dominate, because you have to hold many inventory all the time.

6.2 Benchmark for the Problem

The general benchmark for the problem is shown in figure 4. We are choosing from the set

$$\{80, 140, 200, 300, 400, 500, 600, 700, 820, 950\}$$

We simulate 1000 independent replications for each alternative to create this benchmark. The best alternative would be $S = 300$ with estimated annual cost to be a little over 2000. Variance decreases with S in this case, and the true objective function is clearly not a parabola.

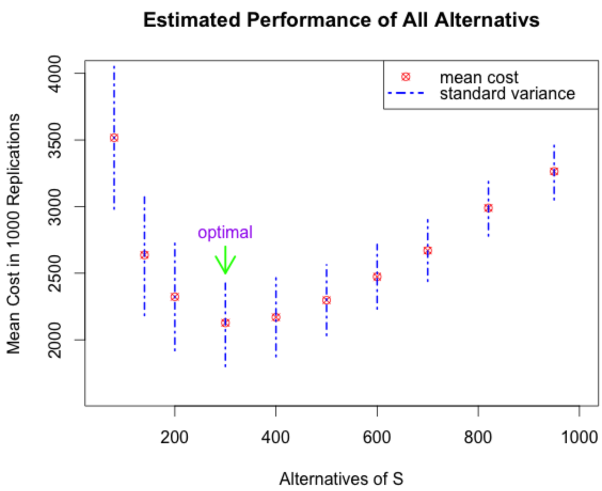

Figure 7: Information about (s,S) inventory system

6.3 Application of Our Selection Procedure

We simulate 200 selection procedures for each simulation budget

$$(40, 80, 120, 160, 200, 240, 280, 320, 360, 400)$$

The result is shown on figure 8. The plot is

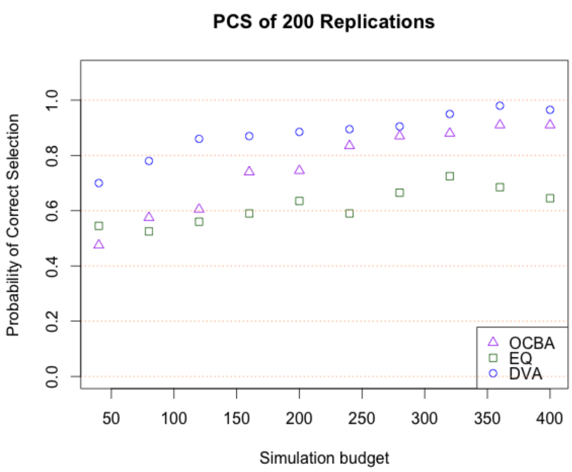

Figure 8: PCS for (s,S) Inventory Problem

not so stable as the previous ones because we only simulated 200 times for each budget. Simulation of the annual cost is quite slow, the entire simulation takes about 7 hours paralled on 7 cores.

But we can still see from the figure that DVA method dominates both OCBA and EQ procedure.

7 Summary

In this project, we designed an algorithm which uses objective function approximation to help us more efficiently assign resources. This is motivated by the fact that 1) usually the performance is a nice function with respect to the design parameter and 2) we should make full use of the information we have obtained rather than using only the ones directly related to each alternative. We showed using simulation that our algorithm works very well both for the designed ideal experiment and two realistic problems.
The validity of regular function assumption is the key to our procedure. There are ways we can get around it like designing the return to be the observed optimal rather than the estimated optimal. But in this case, the information we have on other alternatives would be misleading for our belief on the optimal alternative leading to worse performance than OCBA or EQ procedure.