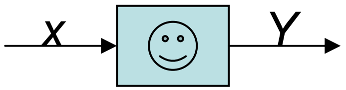


Experimental Design for Simulation

Russell R. Barton, Penn State
W. David Kelton, University of Cincinnati

Introductory Tutorials Track
2003 Winter Simulation Conference
New Orleans

Abstract

- You have some simulation models – how should you use, experiment with them?
- Introduce ideas, issues, challenges solutions, opportunities
- Careful up-front planning of experiments saves time, effort in the end
  - And gets you better, more results
  - Efficient estimates of effects of inputs on outputs
- Discuss traditional experimental design in simulation context, and broader issues of planning simulation experiments

Introduction

- Real meat of simulation project – running model(s), understanding results
- Need to plan ahead before doing runs
  - Just trying different models, model configurations haphazardly is inefficient way to learn
  - Careful planning of runs
    - Improves efficiency
      - Both computational and statistical
        - Really just two sides of the same coin
    - Suggests further experimentation

Introduction (cont'd.)

- *Experimental design* traditionally refers to physical experiments
  - Origins in agriculture, laboratory experiments
- Can recycle most such traditional methods into simulation experiments
  - Will discuss some of this
- Also discuss different situation in simulation, both broader and more specific
  - Overall purpose, what the outputs are, random-number use, effects of input changes on output, optimum-seeking

Introduction (cont'd.)

- Example questions in simulation experiment
  - What model configurations, versions to run?
    - What are the input factors?
    - How should they be varied?
    - Use the same or different random numbers across configurations?
  - Run length?
  - Number of runs?
  - Interpretation, analysis of output?
  - How to make runs most efficiently?

Introduction (cont'd.)

- Purpose here is to call attention to issues, and how to deal with them
  - Not a lot of technical details
- See WSC *Proceedings* paper for this talk for many references to books, papers with complete "do-it-yourself" operational details

Purpose of the Project?

- Maybe obvious, but be clear, specific about ultimate purpose of project
  - Answer can point different ways for design
  - Failure to ask/answer will leave you adrift – unlikely that you'll reach solid conclusions, recommendations
- Even if there's just one model in one configuration, or a very few fixed cases
  - Still questions on run length, number of runs, random-number allocation, output analysis

Purpose of the Project? (cont'd.)

- But if there's more general interest in how changes in inputs affect outputs
  - Clearly, questions on which configurations to run
  - Plus all the single/few scenario questions above
  - Especially in optimum-seeking, need to take care in deciding which configurations to try, ignore
- Goals, strategies often evolve or become more ambitious (or less ...) during project
  - In designed experiments, can use results from early experiments to help choose later ones

Types of Goals

Cycle	Goal
1. Early	Validation
2. Next	Screening
3. Middle	Sensitivity Analysis, Understanding
4. Middle	Predictive Models
5. Later	Optimization, Robust Design

Output Performance Measures?

- Think ahead about what you want out of your simulations
- Most simulation software produces lots of default output
  - Time-based measures, counts
  - Economic-based measures (cost, value added)
  - You can specify or create more
  - Often get averages, minima, maxima
- Easier to ignore things you have than to get things you don't have (to state the obvious ...)
  - But extraneous output can significantly slow runs

Output Performance Measures? (cont'd.)

- One fundamental question for output measures – time frame of simulation/system
  - *Terminating* (a.k.a. *transient*, *short-run*, *finite-horizon*)
    - There's a natural way to start and stop a run
    - Start/stop rules set by system and model, not by you
    - Need to get these right – part of building a valid model
  - *Steady-state* (a.k.a. *long-run*, *infinite-horizon*)
    - Outputs defined as a limit as simulation run length $\rightarrow \infty$
    - No natural way to start – system has already been running forever
    - In theory, never stop run – but you must decide how to

Output Performance Measures? (cont'd.)

- Regardless of time frame, need to decide what aspects of output you want
  - In stochastic simulation, outputs are observations from (unknown) probability distributions
    - Ideally, estimate the whole distribution – ambitious goal
  - Usually get summary measures of output distributions
    - Means (maybe too much focus on these)
    - Extrema
    - Variance, standard deviation
    - Quantiles of output distribution
  - Output desired can affect model, data structure

How to Use Random Numbers?

- Most simulation models are *stochastic*
  - Random inputs from probability distributions
- Simulation software has ways to *generate* observations from input distributions
  - Rely on *random-number generator*
    - Algorithm to produce a sequence of values that appear independent, uniformly distributed on [0, 1]
  - RNGs are actually fixed, recursive formulae generating the same sequence
  - Will eventually *cycle*, and repeat same sequence

How to Use Random Numbers? (cont'd.)

- Obviously, want "good" RNG
  - LONG cycle length
    - An issue with old RNGs on new machines ...
  - Good statistical properties
  - Broken into streams, substreams within streams
  - RNG design is complicated, delicate
- With a good RNG, can ignore *randomization* of treatments (model configurations) to cases (runs) – a concern in physical experiments

How to Use Random Numbers? (cont'd.)

- RNG is controllable, so randomness in simulation experiment is controllable – useful?
  - Controlling carefully is one way to reduce variance of output, without simulating more
- Part of designing simulation experiments is to decide how to allocate random numbers
  - First thought – independent (no reuse) throughout
    - Certainly valid and simple statistically
    - But gives up variance-reduction possibility
    - Usually takes active intervention in simulation software
      - New run always starts with same random numbers – override

How to Use Random Numbers? (cont'd.)

- Better idea when comparing configurations
  - Re-use random numbers across configurations – *common random numbers*
  - Differences in output more likely due to differences in configurations, not because the random numbers bounced differently (they didn't)
  - Probabilistic rationale:

$$\mathrm{Var}\,(A - B) = \mathrm{Var}(A) + \mathrm{Var}(B) - 2\,\mathrm{Cov}(A, B)$$

  - Hopefully, $\mathrm{Cov}(A, B) > 0$ under CRN
    - Usually true, though (pathological) exceptions exist
  - Must ***synchronize*** RN use across configurations
    - Use same RNs *for same purposes*
    - Use of RNG streams, substreams helpful

… 15 1 12 3

… arrival service arrival service

… 15 12

… 1 3 …

Separate ‘arrival’ and ‘service’ streams

… 15 1 12 3

… arrival arrival service service

Sensitivity of Outputs to Inputs?

- Simulation models involve *input factors*
  - Quantitative – arrival rate, number of servers, pass/fail probabilities, job-type percentages, ...
  - Qualitative – queue discipline, topology of part flow, shape of process-time distribution, ...
- *Controllable* vs. *uncontrollable* input factors
  - In **real** system, usually have both
    - Number of servers, queue discipline – controllable
    - Arrival rate, process-time-distribution – uncontrollable
  - In **simulation**, *everything* is controllable
    - Facilitates easy "what-if" experimentation
    - Advantage of simulation vs. real-world experimentation

Sensitivity of Outputs to Inputs? (cont'd.)

- Input factors presumably have *some* effect on output – what kind of effect?
  - Sign, magnitude, significance, linearity, ...
- Mathematical model of a simulation model:

$$\text{Output}_1 = f_1(\text{Input}_1, \text{Input}_2, \ldots)$$
$$\text{Output}_2 = f_2(\text{Input}_1, \text{Input}_2, \ldots)$$
$$\vdots$$

f_1, f_2, ... represent simulation model itself

- Common goal – estimate change in an output given a change in an input
  - Partial derivative
  - But we don't know f_1, f_2, ... (why we're simulating)
  - Now discuss different estimation strategies

Classical Experimental Design

- Has been around for ~80 years
  - Roots in agricultural experiments
- Terminology
  - Inputs = Factors
  - Outputs = Responses
- Estimate how changes in factors affect responses
- Can be used in simulation as well as physical experiments
  - In simulation, have some extra opportunities

Classical Experimental Design (cont’d.)

- Two-level factorial designs
  - Each input factor has two levels (“–”, “+” levels)
  - No general prescription for setting numerical levels
    - Should be “opposite” but not extreme or unrealistic
  - If there are k input factors, get 2^k different combinations of them ... *2^k factorial design*
  - Run simulation at each combination
    - Replicate it? Replicate whole design?
  - Get responses R_1, R_2, ..., R_{2^k}
  - Use to learn about effects of input factors

Classical Experimental Design (cont'd.)

- Design matrix for $k = 3$ (with responses):

Run (i)	Factor 1	Factor 2	Factor 3	Response
1	–	–	–	R_1
2	+	–	–	R_2
3	–	+	–	R_3
4	+	+	–	R_4
5	–	–	+	R_5
6	+	–	+	R_6
7	–	+	+	R_7
8	+	+	+	R_8

- *Main effect* of a factor: average change in response when factor moves from "–" to "+"
  - Main effect of factor 2:
    $(-R_1 - R_2 + R_3 + R_4 - R_5 - R_6 + R_7 + R_8)/4$

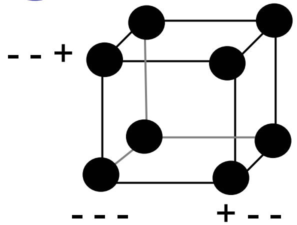
Classical Experimental Design (cont'd.)

- Two-way interaction: does the effect of one factor depend on the level of another?
  - "Multiply" sign columns of the two factors, apply to response column, add, divide by 2^{k-1}
  - Interaction between factors 1 and 3:
    $(+R_1 - R_2 + R_3 - R_4 - R_5 + R_6 - R_7 + R_8)/4$
  - If an interaction is present, cannot interpret main effects of involved factors in isolation

Classical Experimental Design (cont'd.)

- Example: car maintenance/repair shop
  - Kelton, Sadowski, Sturrock, *Simulation With Arena*, 3rd ed., 2004, McGraw-Hill, Chapt. 6
  - Outputs:
    - Daily profit
    - Daily Late Wait Jobs = Cars/day that are "late" for customers waiting
  - Inputs:
    - Max Load = max hours/day that can be booked
    - Max Wait = max number of customer-waiting cars/day that can be booked
    - Wait allowance = hours padded to predicted time in system for waiting customers

Classical Experimental Design (cont'd.)



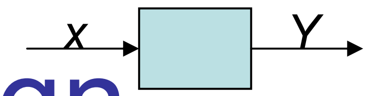
- 2^3 factorial design
  - 100 replications per design point
  - Used Arena Process Analyzer to manage runs:

Scenario Properties				Controls			Responses	
S	Name	Program File	Reps	Max Load	Max Wait	Wait Allowance	Daily Profit	Daily Late Wait Jobs
	Base Case	1 : Model 06-04.p	100	24.0000	5.0000	1.0000	492.628	0.699
	- - -	1 : Model 06-04.p	100	20.0000	1.0000	0.5000	326.596	0.257
	+ - -	1 : Model 06-04.p	100	40.0000	1.0000	0.5000	489.135	0.259
	- + -	1 : Model 06-04.p	100	20.0000	7.0000	0.5000	328.400	0.892
	+ + -	1 : Model 06-04.p	100	40.0000	7.0000	0.5000	480.321	1.771
	- - +	1 : Model 06-04.p	100	20.0000	1.0000	2.0000	326.596	0.083
	+ - +	1 : Model 06-04.p	100	40.0000	1.0000	2.0000	489.135	0.076
	- + +	1 : Model 06-04.p	100	20.0000	7.0000	2.0000	328.400	0.284
	+ + +	1 : Model 06-04.p	100	40.0000	7.0000	2.0000	480.321	0.590

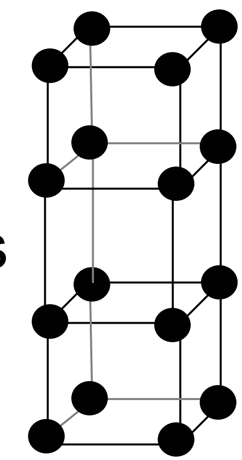
  - Main effects on Daily Profit: +157, –4, 0
    - Implication: should set Max Load to its "+" value
    - Other two factors don't matter
  - Interactions on Daily Profit: –5 (1x2), others 0

Link to spreadsheet

Classical Experimental Design (cont'd.)



- Other limitations of 2^k factorial designs:
  - Implicitly assumes a particular underlying regression model
    - Linear in main effects, product-form interactions
    - Can generalize to more complex designs
  - What if *k* is large (coming soon ...)?
  - Responses are random variables, so what about statistical significance of effects estimates?
    - Can replicate *whole design*, say, *n* times
    - Get *n* i.i.d. estimates of effects
    - Form confidence intervals, tests for expected effects
      - If confidence interval misses 0, effect is statistically significant



Which Inputs Are Important?

- With many factors, probably just a few are important ... *screen out* the others
  - Could theoretically do via main effects in 2^k factorial designs, but, we have:
- *Barton's theorem*:

  If k is big, then 2^k is *REALLY* big
  - Too many factor combinations (and runs)
- Remedies:
  - Fractional factorial designs – run just a fraction (1/2, 1/4, 1/8, etc.) of the full 2^k
  - Specialized factor-screening designs
- Drop some (most?) factors, focus on the rest

Response Surfaces

- Most experimental designs are based on an algebraic regression model
  - Output = dependent (*Y*) variable
  - Inputs = independent (*x*) variables
  - For example, with *k* = 2 inputs, full quadratic form:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + \beta_4 x_1^2 + \beta_5 x_2^2 + \varepsilon$$

- A *regression* model of the *simulation* model – a *metamodel*
  - In *k* = 2 example, also called a *response surface*

Response Surfaces (cont'd.)

- Estimate the model (β coefficients) by making runs, do a regression of Y on x's
  - Which runs to make? Many methods in literature
- Uses of response surfaces in simulation
  - Literally take partial derivatives to estimate effects
    - Any interactions would be naturally represented
  - Proxy for the simulation
    - Explore a wide range of inputs quickly, then simulate intensively in regions of interest
    - Optimize response surface as approximation for model
- Limitations, cautions
  - Regression-model form
  - Variation in response-surface estimates

Optimum Seeking

- May have one output performance measure that's by far the most important
  - Bigger is better – throughput, profit
  - Smaller is better – queueing delays, cost
- Look for a combination of input factors that optimizes (maximizes or minimizes) this
- Like a math-programming formulation
  - Max or min output response over inputs
  - Subject to constraints on inputs, requirements on other outputs
  - Search through the input-factor space

Optimum Seeking (cont'd.)

- Example: car maintenance/repair shop

Maximize **Daily Profit**

Objective function is the simulation model

Subject to

$20 \le$ **Max Load** ≤ 40

$1 \le$ **Max Wait** ≤ 7

$0.5 \le$ **Wait Allowance** ≤ 2.0

Constraints on the input control (decision) variables

Daily Late Wait Jobs < 0.75

An output *requirement*, not an input *constraint*

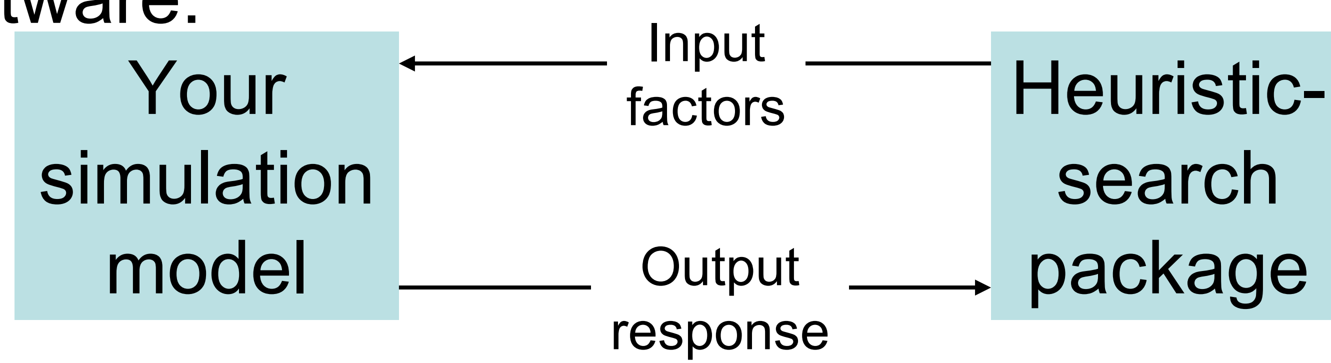
Could also have constraints on linear combinations of input control variables (but we don't in this problem)

Optimum Seeking (cont'd.)

- This is a difficult problem
  - Many input factors – high-dimensional search space
  - Cannot "see" objective function clearly – it's an output from a stochastic simulation
  - May be time-consuming to "evaluate" the objective function – have to run the whole simulation each time
- So, cannot absolutely guarantee to "optimize your simulation"
- Still, it may well be worth trying to get close

Optimum Seeking (cont'd.)

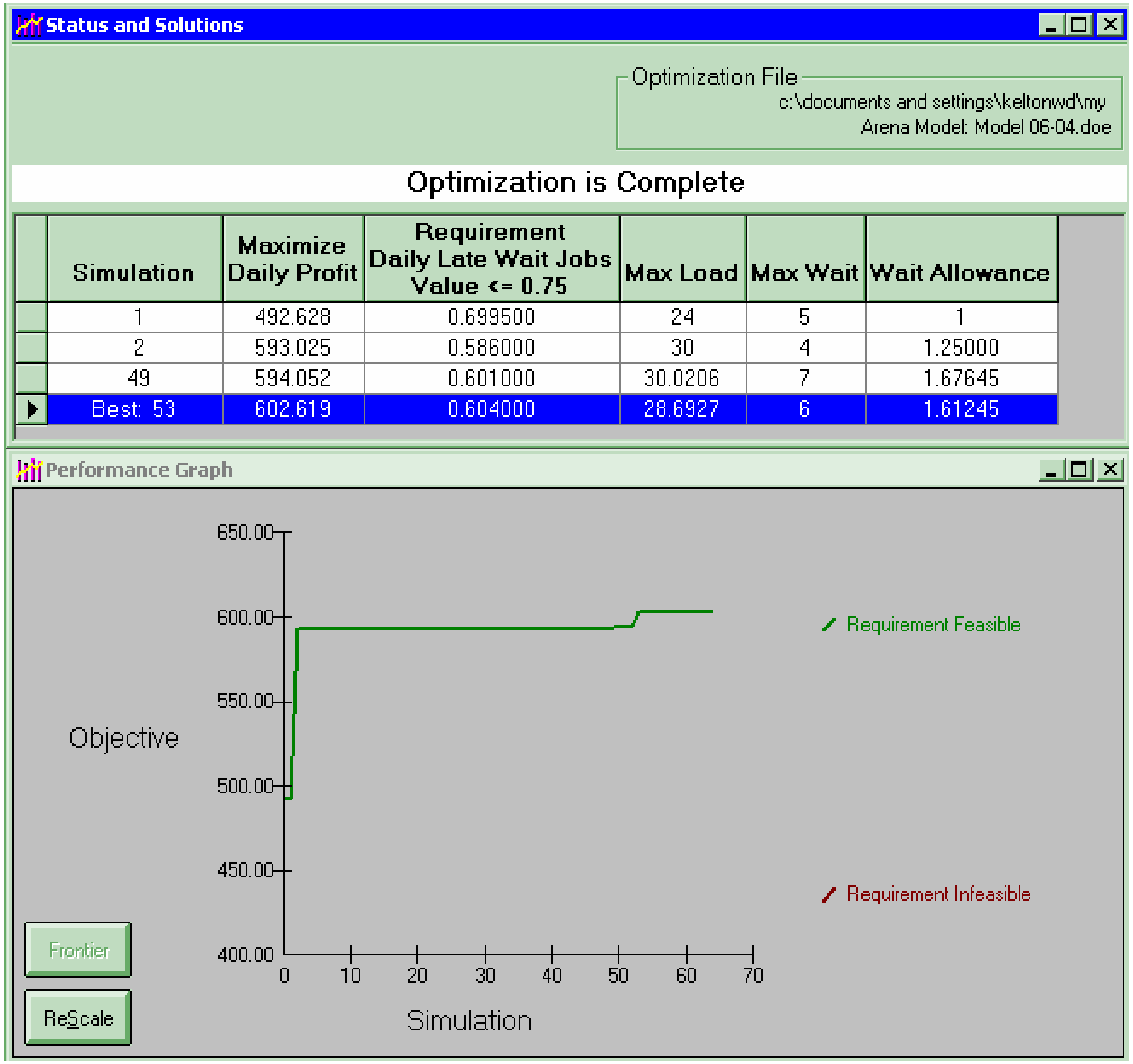
- Heuristic search methods (TABU, Genetic, Pattern) can "move" the model from one input-factor point to another, use response data to decide on future moves
- Several have been linked to simulation-modeling software:



- User must also specify starting point, stopping conditions (can be problematic)

Optimum Seeking (cont'd.)

- Example: car maintenance/ repair shop
- OptQuest optimum-seeker with Arena modeling software
- Ran for 20 minutes



Simulation	Maximize Daily Profit	Requirement Daily Late Wait Jobs Value <= 0.75	Max Load	Max Wait	Wait Allowance
1	492.628	0.699500	24	5	1
2	593.025	0.586000	30	4	1.25000
49	594.052	0.601000	30.0206	7	1.67645
Best: 53	602.619	0.604000	28.6927	6	1.61245

Conclusions

- Designing simulation experiments deserves your attention
  - Capitalize on your (substantial) modeling effort
  - Unplanned, hit-or-miss course of experiments unlikely to yield much solid insight
- There are several formal experimental-design procedures that are quite amenable to simulation experiments
  - Simulation experiments present unique opportunities not present in physical experiments
- Uses computer time – cheaper than your time