Experimental Design and Analysis in Software Engineering
Part 4: Choosing an Experimental Design

Shari Lawrence Pfleeger
Centre for Software Reliability
City University
Northampton Square
London EC1V 0HB England
phone: +44-71-477-8426 – fax: +44-71-477-8585
shari@csr.city.ac.uk

In the previous tutorials, we began to explore the reasons for choosing one experimental design over another. Now, we continue that discussion.

ACTORS VS. BLOCKS

Once you decide on the number of factors appropriate for your experiment, you must determine how to use blocking to improve the experiment's precision. However, it is not always easy to tell when something should be a block instead of a factor. To see how to decide, we use the example of staff experience with software design.

In many experiments, we suspect that the experience of the subjects will affect the outcome. One option in the experimental design is to treat experience as a blocking factor, as described in part 2 of this tutorial. To do this, we can assess the experience of the designers in terms of the number of years each has had experience with design. We can match staff with similar experience backgrounds and then assign staff randomly to the different treatments. Thus, if we are investigating design methods A and B, each block will have at least two subjects of approximately equal experience; within each block, the subjects are assigned randomly to methods A and B.

On the other hand, if we treat experience as a factor, we must define levels of experience and assign subjects in each level randomly to the alternative levels of the other factor. In the design example, we can classify designers as having high and low experience (the two levels of experience); then, within each group, subjects are assigned at random to design method A or B.

To determine which approach (factor or block) is best, consider the basic hypothesis. If we are interested in whether design A is better than design B, then experience should be treated as a blocking variable. However, if we are interested in whether the results of using design methods A and B are influenced by staff experience, then experience should be treated as a factor. Thus, if we are not interested in interactions, then blocking will suffice; if interactions are important, then multiple factors are needed.

In general, then, we offer the following guidelines about blocking:

- If you are deciding between two methods or tools, then you should identify state variables that are likely to affect the results and sample over those variables using blocks to ensure an unbiased assignment of experimental units to the alternative methods or tools.
- If you are deciding among methods or tools in a variety of circumstances, then you should identify state variables that define the different circumstances and treat each variable as a factor.

In other words, use blocks to eliminate bias; use factors to distinguish cases or circumstances.

CHOOSING BETWEEN NESTED AND CROSSED DESIGNS

When you have decided on the appropriate number of factors for your experiment, you must select a structure that supports the investigation and answers the questions you have. As we shall see, this decision is more complicated in software engineering than in other disciplines, because assigning a group not to use a factor may not be sensible or even possible. That is, there are hidden effects that must be made explicit, and there are built-in biases that must be addressed by the structure of the experiment. In addition, other issues can complicate this choice.

Suppose that a company wants to test the effectiveness of two design methods, A and B, on the quality of the resulting design, with and without tool support. The company identifies twelve projects to participate in the experiment. For this experiment, we have two factors: design method and tool usage. The first factor has two levels, A and B, and the second factor also has two levels, use of the tool and lack of use. A crossed design makes use of every possible treatment combination, and it would appear that a crossed design could be used for this experiment.

Crossed		Design Method	
		Method A	Method B
Tool Usage	Not used	Projects 1, 2 and 3	Projects 7, 8 and 9
	Used	Projects 4, 5 and 6	Projects 10, 11 and 12

Figure 5. Crossed design for design methods and tool usage

As shown in Figure 5, the twelve projects are organized so that three projects are assigned at random to each treatment in the design. Consider the implications of the design as shown. Any project has been assigned to any treatment. However, unless the tools used to support method A are exactly the same as the tools used to support method B, the factor levels for tool usage are not comparable within the two methods. In other words, with a crossed design such as this, we must be able to make sense of the analysis in terms of interaction effects. We should be able to investigate down columns (in this example, does tool usage make a difference for a given method?) as well

as across rows (in this example, does method make a difference with the use of a given tool?). With the design in Figure 5, the interaction between method and tool usage (across rows) is not really meaningful. The crossed design yields four different treatments based on method and tool usage that allow us to identify which treatment produces the best result. But the design does not allow us to make statements about the interaction between tool usage and method type.

We can remedy this situation by using a nested design, as shown in Figure 6. The nested design is analyzed differently from the crossed design (a one-way analysis of variance, as opposed to a two-way analysis of variance), so there is no risk of meaningless interaction effects, as there was with the crossed design.

Design Method			
Method A		Method B	
Tool Usage		Tool Usage	
Not used	Used	Not used	Used
Projs. 1,2,3	Projs. 4,5,6	Projs. 7,8,9	Projs. 10,11,12

Figure 6. Nested design for design methods and tool usage

Thus, a nested design is useful for investigating one factor with two or more conditions, while a crossed design is useful for looking at two factors, each with two or more conditions. This rule of thumb can be extended to situations with more than two factors. However, the more factors, the more complex the resulting analysis. For the remainder of the tutorial, we focus on at most two factors, as most situations in software engineering research will involve only one or two factors, with blocking and randomization used to ameliorate the effects of other state variables.

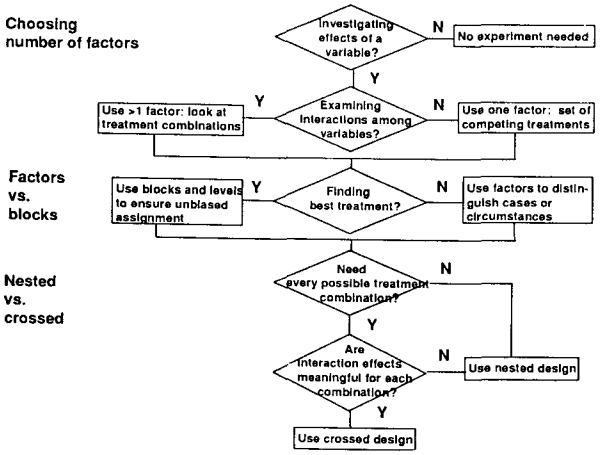

Figure 7. Flow chart for choosing design

Figure 7 summarizes some of the considerations explained so far. Its flow chart helps you to decide on the number of factors, whether to use blocks, and whether to consider a crossed or nested design.

However, there are other, more subtle issues to consider when selecting a design. Let us examine two more examples to see what kinds of problems may be hidden in an experimental design. Consider first the crossed design described by Figure 8. The design shows an experiment to investigate two factors: staff experience and design method type. There are two levels of experience, high and low, and two types of design method. The staff can be assigned to a project after the project's status is determined by a randomization procedure. Then, the project can be assigned to a treatment combination. This example illustrates the need to randomize in several ways, as well as the importance of assigning subjects and treatments in an order that makes sense to the design and the goals of the experiment.

Crossed		Design Method	
		Method A	Method B
Staff Experience	Low	Projects 1, 2 and 3	Projects 7, 8 and 9
	High	Projects 4, 5 and 6	Projects 10, 11 and 12

Figure 8. Crossed design for method types and staff experience

Figure 9 is similar to Figure 8, except that it is examining method usage, as opposed to method type. In this case, it is important to define exactly what is meant by "not used." Unlike medicine and agriculture, where "not used" means the use of a placebo or the lack of treatment with a chemical, "not used" in software engineering may be difficult or impossible to control. If we tell designers not to use a particular method, they are likely to use an alternative method, rather than no method at all. The alternative method may be hidden, based on how they were trained or what experience they have, rather than an explicitly-defined and well-documented other method. In this case, the design is inappropriate for the goals of the experiment. However, if the goal of the experiment is to assess the benefit of a tool to support the given method, then the design is sufficient.

Crossed		Design Method	
		Used	Not used
Staff Experience	Low	Projects 1, 2 and 3	Projects 7, 8 and 9
	High	Projects 4, 5 and 6	Projects 10, 11 and 12

Figure 9. Crossed design for method usage and staff experience

FIXED AND RANDOM EFFECTS

Some factors allow us to have complete control over them. For example, we may be able to control what language is used to develop a system, or what processor the system is developed on. But other factors are not easy to control, or are pre-

determined; staff experience is an example of this type of factor. The degree of control over factor levels is an important consideration in choosing an experimental design. A fixed-effects model has factor levels or blocks that are controlled. A random-effects model has factor levels or blocks that are random samples from a population of values. If staff experience is used as a blocking factor to match subjects of similar experience prior to assigning them to a treatment, then the actual blocks are a sample of all possible blocks, and we have a random-effects model. However, if staff experience is defined as two levels, low and high, the model is a fixed-effects model.

The difference between fixed- and random-effects models affects the way the resulting data is analyzed. For completely randomized experiments, there is no difference in analysis. But for more complex designs, the difference affects the statistical methods needed to assess the results. If you are not using a completely randomized experiment, you should consult a statistician to verify that you are planning to use techniques appropriate to the type of effects in your model.

The degree of randomization also affects the type of design that is used in your experiment. You can choose a crossed design when subjects can be assigned to all levels (for each factor) at random. For example, you may be comparing the use of a tool (in two levels: using the tool and not using the tool) with the use of a hardware platform (using a Sun, a PC or a Macintosh, for instance). Since you can assign developers to each level at random, your crossed design allows you to look for interaction between the tool and the platform. On the other hand, if you are comparing tool usage and experience (low level of experience versus high level of experience), then you cannot assign people at random to the experience category; a nested design is more appropriate here.

MATCHED OR SAME SUBJECT DESIGNS

Sometimes, economy or reality prevents us from using different subjects for each type of treatment in our experimental design. For instance, we may not find enough programmers to participate in an experiment, or we do not have enough funds to pay for a very large experiment. We can use the same subjects for different treatments, or we can try to match subjects according to their characteristics in order to reduce the scale and cost of the experiments. For example, we can ask the same programmer to use tool A in one situation and then tool B in another situation. The design of matched- or same-subject experiments allows variation among staff to be assessed and accounts for the effects of staff differences in analysis. This type of design usually increases the precision of an experiment, but it complicates the analysis.

Thus, when designing your experiment, you should decide how many and what type of subjects you want to use. For experiments with one factor, you can consider testing the levels of the factor with the same subjects or with different subjects. For two or more variables, you can consider the question of same-or-different separately for each variable. To see how, suppose you have an experimental design with four different treatments, generated by a crossed design with two factors. If different subjects are used for each treatment (that is, for each of both variables), then you have a completely unrelated between-subjects design. Alternatively, you could use the same subjects (or subjects matched for similar values of each level) and subject them to all four treatments; this is a completely related within-subjects design. Finally, you can use the same subjects for one factor but different subjects for the other factor to yield a mixed between- and within-subjects design.

REPEATED MEASUREMENTS

In many experiments, one measurement is made for each item of interest. However, it can be useful to repeat measurements in certain situations. Repeating a measurement can be useful in validating it, by assessing the error associated with the measurement process. We explain the added value of repeated measurements by describing an example.

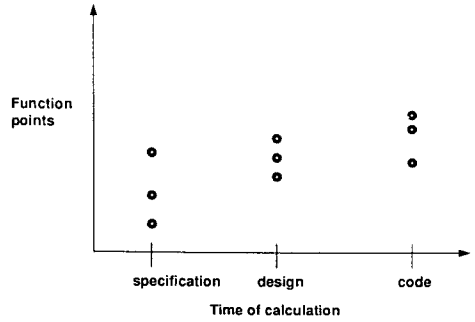

Figure 10. Repeated measurements on function point calculations

Figure 10 depicts the results of an experiment involving one product and three developers. Each developer was asked to calculate the number of function points in the product at each of three different times during development: after the specification was completed, after the design was finished, and after the code was done. Thus, in the figure, there are three points marked at each of the three estimation times. For example, at specification, each of the three developers produced a slightly different function point estimate, so there are three distinct points indicated above "specification" on the x-axis. The figure shows that there were two kinds of variation in the data that resulted. The horizontal variation indicates the variation over time, while the vertical differences at each measurement time indicates the variation due to the differences among the developers. Clearly, these repeated measurements add value to the results of the experiment, but at the cost of the more complex analysis required. The horizontal variation helps us to understand the error about the line connecting the means at each measurement time, and the vertical error helps us to understand observational error.

As you can see, there are many issues to consider when choosing a design for your experiment. Once it is chosen and the experiment is run, the resulting data can be analyzed. The next issue's tutorial will explain how to select appropriate analysis techniques.