The Last Byte

Scott Davidson
Scott.Davidson@eng.sun.com

The Scientific Method and Design and Test

A FEW YEARS AGO, during a well-attended open Design Automation Conference benchmark forum,[1] a panelist pointed out that "...reporting experimental results is a science and an art. A survey of the literature may reveal a consistent methodology.… Also, we should address the verification of reported results." In retrospect, the suggestion may have hinted at the vast body of techniques commonly known as "experimental design." However, this forum was not ready to expand on the subject.

Most any search engine currently on the Web returns tens of thousands of URLs in response to keyword searches using terms such as experimental design or design of experiments. Few, if any, of these search results point to an evaluation of CAD algorithms. In contrast, the experimental design methodology, pioneered by Fischer during the 1920s in agricultural research,[2] is now firmly established in science and manufacturing. Its application to biomedical research can save lives. Biomedical journals have strict guidelines on how to report experimental results so others may replicate experiments. A URL from a medical school points to a concise illustration of a simple experimental design flow and the terminology used (http://www.fammed.ouhsc.edu/TUTOR/expdes.htm) (Figure 1).

Adopting the accepted norms of experimental design will give us a scientific method to conduct, verify, and report comparative performance evaluations of CAD algorithms. As indicated in Figure 1, we should adopt and adapt three basic procedures: 1) formation of an equivalence class of experimental subjects, selected randomly from a population of subjects eligible for treatment; 2)

continued on p. 142

Population	All subjects to whom a treatment might possibly be applied. For example, all adult-onset diabetic patients in the world.
Sample	A subset of population
R	Random selection
E	Experimental treatment (applied to experimental group)
C	Alternative (placebo) treatment (applied to control group)
D	Observation (measurement) of the effect of the treatment

Figure 1. Experimental design in biomedicine: measuring effectiveness of two treatments.

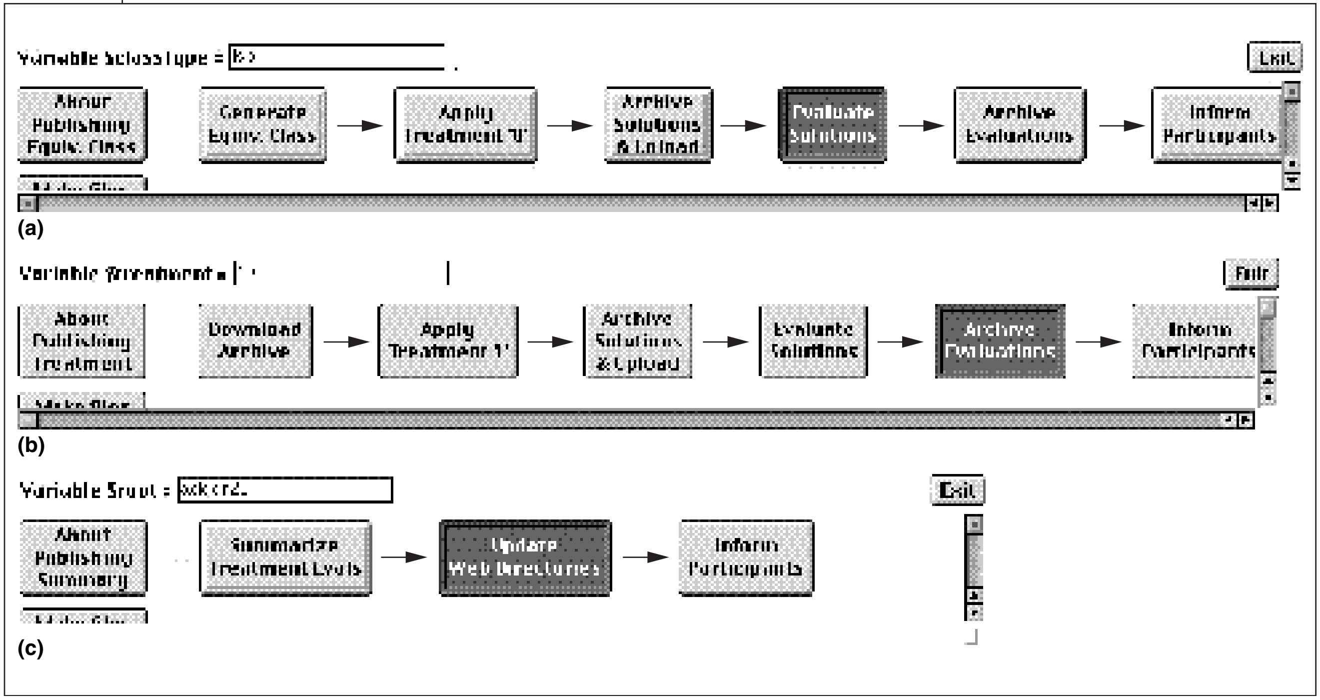

Figure 2. Task flows to support execution and publication of three phases of experimental designs: (a) task flow publish_classData, (b) task flow publish_treatment, and (c) task flow publish_summary.

continued from p. 144

application of one or more treatments to subjects in the same class; and 3) statistical evaluation of each treatment's effectiveness. Adopting the terminology, we understand that an equivalence class of experimental subjects is synonymous with a set of the same or similar netlists; and "applying treatments" is synonymous with "invoking algorithms."

The immediate benefit of adopting this methodology can give us insight into the relative merits of the current benchmarking paradigm itself. A current benchmark-set sampling reveals the typical 5-50% improvements, as reported in the experimental results sections of current publications, have little statistical significance.[3] These improvements are reported for single instances of very diverse benchmark set, which don't constitute a class for a well-defined experiment. Moreover, reapplying the same algorithms to different orderings of the same benchmark can induce variations that may be larger than such improvements. We can construct an isomorphism netlist class that may expose such variations for most heuristics by creating 32 or more nominal benchmark copies, where both names and node order in the netlist are randomized. In an experiment with an isomorphism equivalence class, cost functions such as layout area, total wire length, size of binary decision diagrams (BDD), and total backtracks during automatic test-pattern generation (ATPG) approach a normal distribution with a standard deviation that may exceed the largest improvement reported for a given benchmark. Variance of 0 is a rare event for most algorithms and benchmarks. An ATPG algorithm that reports 100% fault coverage under a specific netlist order may reach a backtrack limit under a different netlist order, and must therefore report a reduced fault coverage. Similarly, it may not even be possible to build a BDD for a given benchmark under every netlist order.

While of fundamental importance, the isomorphism class is but one of the equivalence classes that should be considered when designing experiments to compare CAD algorithms. Recent reports discuss a number of approaches to equivalence netlist class generation and include a research update on the topic.[4] To advance the case for benchmarking CAD algorithms under the discipline of experimental design, we must provide more than the new equivalence classes accessible from the Web. OpenExperiment is a prototype client/server environment for distributed experimental design.[5] The client interface captures important attributes that render three executable and user-configurable task flows in Figure 2:

- publish_classData supports a distributed team to coordinate generation, characterization, and Web-posting of netlist equivalence classes on a centralized host. This set is also characterized by 'treatment 0,' that is, evaluation of the problem-specific cost function before any treatment is applied. For example, we may report the crossing number after each placement a randomly reordered netlist from an isomorphism class. (By class definition, all such orders are random.)
- publish_treatment supports any participating experimentalist to download the data sets, execute the algorithm on the local host, and then to submit and execute solutions on the centralized server where the same evaluator looks at submissions from all participants.
- publish_summary supports the archivist periodically summarizing treatments submitted by participants and posting them on the Web. In the example shown, treatments refer to crossing number minimization in bipartite graphs. Statistical experiment summaries shows three treatments: TR00 for random placement, as reported by the netlist class contributor, followed by TR12 and TR17 results reported by two distinct algorithms.

We classify the experiments in terms of cost functions evaluated on the underlying graph representation. Each class of experiments may need a configuration of experiment-specific task flows, while also reflecting participant preferences. The special International Symposium on Circuits and Systems (ISCAS) session in 1985 that brought together not only nine research teams but also a shared data set contributed significantly to rapid dissemination and adoption of the benchmarking paradigm as we currently know it. Shifting the existing paradigm to experimental design disciplines will require effort from interdisciplinary teams, and not only data sets but an environment such as OpenExperiment will have to be distributed and supported. An international workshop may provide a suitable forum to initiate the process.

To express your views regarding this and related subjects, we invite you to browse and review the links accessible from http://www.cbl.ncsu.edu/OpenProjects.

References

1. F. Brglez, "A *D&T* Special Report on ACM/SIGDA Design Automation Benchmarks: Catalyst or Anathema?", *IEEE Design & Test of Computers*, Vol. 10, No. 3, Sept. 1993, pp. 87–91.
2. R.A. Fisher, *Statistical Methods, Experimental Design, and Scientific Inference*. Oxford University Press, Oxford, UK. Reprinted, with corrections, from earlier versions, 1925-1973, 1993.
3. F. Brglez and R. Drechsler, "Design of Experiments in CAD: Context and New Data Sets for ISCAS'99." *Proc. IEEE 1999 Intl. Symp. Circuits Systems—ISCAS'99*, May 1999, pp.424-427.
4. D. Ghosh, *Generation of Tightly Controlled Equivalence Classes for Experimental Design of Heuristics for Graph-Based NP-hard Problems*, doctoral dissertation, North Carolina State Univ., Electrical and Computer Engineering Department, Raleigh, N.C., May 2000. Also available at http://www.cbl.ncsu.edu/publications/2000-Thesis-PhD-Ghosh.
5. F. Brglez et al., *OpenExperiment: A Configurable Environment for Collaborative Experimental Design and Execution on the Internet*, Tech Report 2000-TR@CBL-02-Brglez, CBL, CS Dept., NCSU, Box 8206, Raleigh, NC, 27695, Mar. 2000. Also available at http://www.cbl.ncsu.edu/publications/#2000-TR@CBL-02-Brglez.

Franc Brglez, our author, is a research professor in the Computer Science Department of North Carolina State University, Raleigh, NC. You may contact him at brglez@cbl.ncsu.edu.