Letter to the Editor

It is with some regret that I have to write to you again to protest about the unrealistic assumptions contained in a paper published in RESS. I refer to the paper by Yamachi et al [Multi-objective genetic algorithm for solving N-version program design problem, *Reliability Engineering and System Safety* 91 (2006) 1083-1094]. The false assumption is contained in Section 4.1: "Failures of versions for each module (subtask or subsystem) are s-independent."

Some months ago you were kind enough to publish my complaints about a similar assumption in an earlier paper ('Comments on "Reliability and performance analysis for fault-tolerant programs consisting of versions with different characteristics," by Gregory Levitin [*Reliability Engineering and System Safety* 86 (2004) 75-81], *Reliability Engineering and System Safety* 91 (2006) 119-120). I shall not repeat here the detailed reasons why such an assumption is wrong - readers should consult this earlier note.

I have no reason to believe that the mathematical modeling reported by Yamachi et al is incorrect - but their results have no value if the assumptions upon which the modeling is founded are wrong. I think it would be regrettable if *Reliability Engineering and System Safety* became known as a journal that published worthless mathematical exercises.

I take no comfort in noting that Dr Levitin, the author of the paper that was the subject of my first complaint, should have been the editor of the Special Issue in which the second paper was published.

Bev Littlewood
Professor of Software Engineering
Centre for Software Reliability
City University
London EC1V 0HB, UK
b.littlewood@csr.city.ac.uk

Reply

As the commentator pointed out, our assumption of independence of failures in software components is the most simplified model in the software reliability models and not realistic. It is obvious that when software components or versions are developed based on the same requirement specification and are functionary equivalent, discussing about their reliabilities without considering the common caused failures (CCF) does not provide meaningful results.

The purpose of our paper, however, is proposing a method that solves the NVP design problem, and is not proposing a software reliability model. We employed the most simplified reliability model in order to illustrate the characteristics of our solving method clearly. Because this problem is one of the NVP complete problems for the number of

versions or subtasks, Ashrafi and Levitin proposed the genetic algorithms that solve the problem. They employed the simplest model as the basis for their discussion, and succeeded to obtain the optimal approximations. We are trying to obtain the exact solutions by using the proposed method. Therefore, we need to evaluate our method on the same basis, which is the simplest reliability model, because that is the model used in the preceded researches.

Even if we employ reliability model that take the CCF into consideration, the only place we need to revise in our algorithm is calculating the system reliability. This revision may add some complexity in the reliability calculation, but we expect no more effects. Thus, if we had employed a reliability model that takes the CCF into account, it would not affect the conclusion that our method of solving NVP problem is efficient and useful. Therefore, we consider that our discussion in the paper holds without discussing the validity of s-independent reliability model.

On the other hand, the validity of the reliability model is an extremely important issue, and the research for methods that solve the NVP problem based on more practical models is indispensable. As a future direction, we will construct a reliability model that takes the CCF into account in order to obtain solutions in more realistic environment.

Hidemi Yamachi
Department of Computer and Information Engineering
Nippon Institute of Technology
Miyashiro, Saitama 345-8501, Japan
yamachi@nit.ac.jp