Towards the Fault Tolerant Software: Fuzzy Extension of Crisp Equivalence Voters

(summary of the paper)

Milos Manic
Department of Computer Science
University of Idaho
800 Park, Blvd. Ste.#200
Boise, ID 83712, USA
milosm@cs.uidaho.edu

Deborah Frincke
Department of Computer Science
University of Idaho
P.O. Box 441010
Moscow, ID 83844-1010
frincke@cs.uidaho.edu

I. INTRODUCTION

Fault tolerance is always based on redundancy. Redundancy can be accomplished through hardware, software, information and time. This paper deals with software fault tolerance. There have been numerous examples of software fault tolerance in literature with fault tolerance achieved through different software versions that can be independently developed, compiled through independent compilers, or run on different platforms. Those redundant software versions are intended to be functionally equivalent but independent in every other aspect, in order to achieve greater system reliability and therefore fault independence.

In presence of different outputs obtained from different redundant software versions, a methodology for selecting the correct output is needed. Traditionally used methodologies are based on an output classification. Those methodologies select a correct answer depending on different conditions. This traditionally applied concept, based on numerical values, considers classification of all outputs into pair-wise disjoint subsets also known as partitions of the outputs. Elements within those subsets are, if not identical, “equal” within certain tolerance. Methods not based on output classification use a variety of Schneider’s convergence functions. Those are fault tolerant midpoint, median, average (egocentric or not), weighted average, etc. [1, 2].

SELECTED REFERENCES