Temporal Property Preservation Under Z Refinement in CSP-OZ Specifications

Azman Bujang Masli, Abdul Rahman Mat, Suriati Khartini Jali and Noor Hazlini Borhan
Faculty of Computer Science and Information Technology
Universiti Malaysia Sarawak
Kota Samarahan, Sarawak, Malaysia
{bmazman, marahman, jskhartini, bnhazlini}@fit.unimas.my

***Abstract*—One way to verify the correctness of an implementation under refinement in formal specifications is by verifying the system against a set of properties we wish to have in the final implementation. This is in such a way that the relevant properties are preserved in each development step. The difference here is that we have a separate specification of system properties. These properties are those that are satisfied by the initial specification. As the development of the system progresses from one step to another, the correctness of the concrete specification is verified by checking the satisfaction of the properties. The correctness of the abstract specification is preserved in the concrete specification (or an implementation) if the concrete specification satisfies all properties the abstract specification satisfies [1]. In other words, the properties are preserved and hold in the concrete specification. This paper extends the result on LTL property preservation for Z specifications in [2] to the OZ part of CSP-OZ specifications. This is where Z refinement exists side-by-side with CSP refinement in the CSP part of a CSP-OZ specification.**

I. Introduction

Refinement in formal specifications is defined in terms of behavioural substitutivity. This is where one system can substitute another system without noticeable differences in behaviour being detected by users [3], [4]. However, for complex systems, some properties may not be related to the systems' behaviour at all, for example properties on states of a system. In most cases, checking that some *good properties* of the system's states are preserved is desired. Hence, there is a need to investigate the preservation of such properties under refinement. This includes properties that are not readily available within the system specification (as invariants), such as *temporal properties* [2].

Properties that are not part of system's invariants need to be specified in a totally different notation. With this respect, the combination of *linear temporal logic* (LTL) [5] and first-order logic has been shown to be sufficient in precisely specifying properties of reactive systems [6]. A temporal structure (*Kripke* structure) is defined to link the semantics of properties, specified in LTL, and the system specification. We can then model check the temporal structure to check for the holding of a property. We will do this in this paper and the same is also done in [2] for checking properties under Z refinement.

Our intention here is to extend the result on LTL property preservation under Z refinement for Z specification to the OZ part of CSP-OZ specification. As the first step, and due to the limited space as well, we will only consider traces refinement in the CSP part of CSP-OZ speficiations in this paper.

II. Related Work

There is very little work in the literature that study property preservation in existing refinement design framework of any formalism in formal specifications. The lack of work on the relationship between the preservation of properties specified in LTL and refinement is recognised in [7], where the authors addressed issues of property preservation under traces and failures refinements of CSP. They shown that both refinements do not preserve LTL properties on infinite traces or branching system. Limiting CSP processes to only finite state processes, however, LTL properties are preserved under failures refinement. We, therefore, will only consider finite states systems in this paper. The restriction is also assumed by Derrick and Smith in [2] on LTL property preservation for Z specifications.

In [2], Derrick and Smith investigated the preservation of properties of Z specification that may refer to the states of the specification, which are not observable. They applied the notion of refinement to states in sequences or *paths*. This is because the temporal logic used for specifying temporal properties, which is *Linear Temporal Logic* (LTL), is defined on paths. The refinement is in such a way that for every concrete path, there exists an abstract path such that every state of the concrete path is related to the corresponding state of the abstract path by the retrieve relation of the refinement. This is expressed in the following lemma.

Lemma 1: Given Z specifications, A and C, if $A \sqsubseteq C$ under retrieve relation R, then for all paths $\pi^C = t_0 t_1 t_2 \ldots$ of C there exists a path $\pi^A = s_0 s_1 s_2 \ldots$ of A such that each state t_i of π^C is related to the corresponding state s_i of π^A by R.

The general theorem of temporal property preservation between states is changed accordingly for refinement between states in sequences as stated in the following theorem.

Theorem 1: Given Z specifications, A and C, and temporal property P, if there exists an $i : \mathbb{N}$ such that for all abstract paths π^A, A, $\pi^A_i \models P$ and $A \sqsubseteq C$ under retrieve relation R then for all concrete paths π^C, C, $\pi^C_i \models \exists AState \bullet P \wedge R$.

The result above is further extended to include other temporal logics in [8]. Due to the lack of space, this paper will only extend the general results of [2] to the OZ part of CSP-OZ specifications with the co-existance of Z and CSP traces refinements.

978-1-4673-1938-6/12/$31.00 ©2012 IEEE