An Approach Towards the Verification of Component-Based Systems via Model Checking

Pham Ngoc Hung (410205)

School of Information Science,
Japan Advanced Institute of Science and Technology

August 11, 2006

Keywords: model checking, modular verification, component refinement.

Verification of software has received a lot of attentions of the software engineering community, specially modular verification of component-based software. However, to realize such an ideal component-based software paradigm, one of the key issues is to ensure that those separately specified and implemented components do not conflict to each other when composed - the *component consistency* issue. A potential solution to the above issue is modular verification of component-based software via model checking. The main goal in this thesis is to combine the best advantages of model checking and component-based development.

Currently there are many approaches have been proposed in modular verification of component-based software [2, 4, 7, 8, 10, 11, 22]. In [10, 11, 22], modular verification is rather closed. It is not prepared for future changes. If a component is added to the system, the whole system of many existing components and the new component must be re-checked altogether. For this reason, the *"state space explosion problem"* will occur when it checks complex software. The approach in [2, 4, 7, 8] focuses on checking a system composed of two components; M_1 and M_2 which satisfies the property p *without composing* M_1 with M_2. For this goal, this technique finds an assumption A such that it is strong enough for M_1 to satisfy p and weak enough to be discharged by M_2. From these, the composition

system $M_1 \| M_2$ satisfies p. However, this approach is viewed from a static perspective to re-generate new assumption. If the component changes after adapting some *refinements*, the assumption-generating approach is re-run on the whole component from beginning, i.e., the component model has to be re-constructed; and the assumption about the environment is then regenerated from that model. Therefore, this approach is not efficient to change the system. This thesis proposes a faster assume-guarantee verification approach for component-based software verification in the context of component refinement. In this approach, if a component is refined into a new component, the whole system of many existing components and the new component is not required to be re-checked altogether. It only checks the new component satisfying the assumption of the old system. If so, the new system also satisfies the property. Otherwise, the proposed technique performs some analysis to determine whether the property is indeed violated in the new system or whether the assumption of the old system is too strong for the new component to satisfy. If the assumption is too strong, a new assumption is re-generated. The technique in this thesis tries to reuse the results of the previous verification in order to have an incremental manner to re-generate the new assumption. It doesn't re-generate the new assumption from beginning. A case study is presented to illustrate the proposed approach. The LTSA [12] tool also is used to check correctness of the technique by some concrete examples.