Software Model Checking
by Program Specialization

We present a method for performing the model checking of imperative programs by using techniques based on the specialization of constraint logic programs (CLP). We have considered a simple imperative language, called SIMP, extended with a nondeterministic choice operator, and we have used CLP for defining its operational semantics via an interpreter. Thus, we have defined our software model checking method which consists in: (i) translating a given SIMP program, annotated with a desired safety property and the initial configurations, into a CLP term, (ii) specializing the CLP interpreter with respect to the above translation, and (iii) computing the least model of the specialized program. By simply inspecting the obtained least model we are able to verify whether or not the given SIMP program satisfies the safety property. The method is fully automatic and has been implemented using the MAP transformation system. We have shown the effectiveness of our method by applying it to some examples taken from the literature and we have compared its performance with that other state-of-the-art software model checkers.