1 Replacing Specifications by Assertions?

The previous version of Quartz made use of specification modules. In this section, it is discussed whether these specification modules can be replaced with the new assertion statements. To this end, assume that the Quartz module has been translated to a Kripke structure $\mathcal{K}$ with initial states $\mathcal{I}$, transition relation $\mathcal{R}$, and label function $\mathcal{L}$. In this case, a specification Φ will hold iff $\mathcal{I} \subseteq [\![\Phi]\!]_{\mathcal{K}}$ holds.

Next, let us consider what is checked when assertions and assumptions are used. Assume, the Quartz program makes use of the assumptions $(\varphi_1, \texttt{assume}\ \Phi_1)$, $\ldots$, $(\varphi_m, \texttt{assume}\ \Phi_m)$ and the assertions $(\psi_1, \texttt{assume}\ \Psi_1), \ldots, (\psi_n, \texttt{assume}\ \Psi_n)$. According to the semantics of assumptions and assertions, the following specification is generated:

$$\mathsf{AG}\left(\bigwedge_{i=1}^{m}(\varphi_i \rightarrow \Phi_i) \rightarrow \bigwedge_{i=1}^{n}(\psi_i \rightarrow \Psi_i)\right)$$

Obviously, all specifications that can be generated this way are of the form $\mathsf{AG}(\eta \rightarrow \xi)$, so that existential formulas can not be obtained this way. For this reason, it seems that formulas like $\mathsf{EF}\varphi$ cannot be checked with assertions.

However, note that specifications Φ must be state formulas. If we additionally assume that all initial states have at least one infinite path, then we can use the following lemma to see that in Quartz 2.0, we have not lost expressiveness:

Lemma 1 (Assuming Initial States). *Given a Kripke structure $\mathcal{K} = (\mathcal{I}, \mathcal{S}, \mathcal{R}, \mathcal{L})$, a state formula Φ, and an atomic formula* `InitStates` *such that $\mathcal{I} = [\![\texttt{InitStates}]\!]_{\mathcal{K}}$ holds. If all initial states $\mathcal{I}$ have at least one infinite path, then the following facts are equivalent:*

- $\mathcal{K} \models \mathsf{AG}\,(\texttt{InitStates} \rightarrow \Phi)$*, i.e.,* $\mathcal{I} \subseteq [\![\mathsf{AG}\,(\texttt{InitStates} \rightarrow \Phi)]\!]_{\mathcal{K}}$
- $\mathcal{K} \models \Phi$*, i.e.,* $\mathcal{I} \subseteq [\![\Phi]\!]_{\mathcal{K}}$

Proof. We prove the following two implications:

$\boxed{\Rightarrow:}$ Assume (1) $\mathcal{I} \subseteq [\![\mathsf{AG}\,(\texttt{InitStates} \rightarrow \Phi)]\!]_{\mathcal{K}}$ holds. Then, we conclude (note that $\texttt{InitStates} \rightarrow \Phi$ is a state formula):

$$\begin{aligned}
&\mathcal{I} \subseteq [\![\mathsf{AG}\,(\texttt{InitStates} \rightarrow \Phi)]\!]_{\mathcal{K}} \\
&\quad \Leftrightarrow \forall s \in \mathcal{I}.(\mathcal{K}, s') \models \mathsf{AG}\,(\texttt{InitStates} \rightarrow \Phi) \\
&\quad \Leftrightarrow \forall s \in \mathcal{I}.\forall \pi \in \mathsf{Paths}_{\mathcal{K}}(s).\ (\mathcal{K}, \pi, 0) \models \mathsf{G}\,(\texttt{InitStates} \rightarrow \Phi) \\
&\quad \Leftrightarrow \forall s \in \mathcal{I}.\forall \pi \in \mathsf{Paths}_{\mathcal{K}}(s).\forall t \in \mathbb{N}.\ (\mathcal{K}, \pi, t) \models (\texttt{InitStates} \rightarrow \Phi) \\
&\quad \Leftrightarrow \forall s \in \mathcal{I}.\forall \pi \in \mathsf{Paths}_{\mathcal{K}}(s).\forall t \in \mathbb{N}.\ (\mathcal{K}, \pi^{(t)}) \models (\texttt{InitStates} \rightarrow \Phi)
\end{aligned}$$

Now, assume s is an arbitrary initial state. According to our assumption, s has at least one infinite path π. Then, instantiating the above formula with s, one of the infinite paths $\pi \in \mathsf{Paths}_{\mathcal{K}}(s)$, and $t = 0$ yields (2) $\forall s \in \mathcal{I}.\ (\mathcal{K}, s) \models \Phi$, i.e., $\mathcal{K} \models \Phi$.

$\boxed{\Leftarrow:}$ Assume (2) $\forall s \in \mathcal{I}.\ (\mathcal{K}, s) \models \Phi$ holds. Consider an arbitrary initial state $s \in \mathcal{I}$ and an infinite path $\pi \in \mathsf{Paths}_{\mathcal{K}}(s)$. It clearly follows from (2) that $(\mathcal{K}, \pi^{(t)}) \models (\mathtt{InitStates} \rightarrow \Phi)$ holds (note that all initial states are reachable by definition). Hence, we also have (1) $\mathcal{I} \subseteq [\![\mathsf{AG}\,(\mathtt{InitStates} \rightarrow \Phi)]\!]_{\mathcal{K}}$. □

It is clear that the above lemma does not hold for states s that have no infinite paths. For example, consider $\Phi :\equiv \nu.\Diamond x$ which holds exactly in those states that have an infinite path. Clearly, Φ does not hold in s, but s satisfies every formula starting with a universal path quantifier A.

Hence, according to the above lemma, we can replace specifications with assertions that are executed in the initial macro step of the Quartz program with the restriction that all initial states must have at least one infinite path. This is always the case when the program has no runtime errors like write conflicts etc. Therefore, the entire verification has to proceed in two steps:

1. Check that there are no initial states with finite paths. Finite paths correspond with program errors and can be found by causality analysis.
2. Having checked the absence of program errors, we can replace the specifications with assertions that are executed in the initial macro step. Note further that we can use the simple translation from CTL to μ-calculus (without taking further care of finite paths).

Having seen that the replacement of specifications with (initial) assertions is possible, it might be argued that this will not be as efficient. Using global model checking, we clearly would check $\mathcal{I} \subseteq [\![\Phi]\!]_{\mathcal{K}}$ once having computed $[\![\Phi]\!]_{\mathcal{K}}$. Considering instead the specification $\mathsf{AG}\,(\mathtt{InitStates} \rightarrow \Phi)$, we would go on with the following fixpoint iteration note that ($\mathsf{AG}\psi$ is equivalent to $\nu x.\psi \wedge \Box x$): $Q_0 := [\![\mathsf{true}]\!]_{\mathcal{K}}$, $Q_1 := [\![(\mathtt{InitStates} \rightarrow \Phi) \wedge \Box\mathsf{true}]\!]_{\mathcal{K}}$. Hence, $Q_1 = ((S \setminus \mathcal{I}) \cup [\![\Phi]\!]_{\mathcal{K}}) \cap [\![\Box\mathsf{true}]\!]_{\mathcal{K}}$, and assuming that $\mathcal{I} \subseteq [\![\Phi]\!]_{\mathcal{K}}$ holds, we conclude further that $Q_1 = S \cap [\![\Box\mathsf{true}]\!]_{\mathcal{K}} = [\![\Box\mathsf{true}]\!]_{\mathcal{K}}$. Finally, note that $[\![\mathsf{true}]\!]_{\mathcal{K}} = [\![\Box\mathsf{true}]\!]_{\mathcal{K}}$, so that $Q_0 = Q_1$ and the fixpoint is already found.

However, if $\mathcal{I} \not\subseteq [\![\Phi]\!]_{\mathcal{K}}$ holds, then $Q_1 = (S \setminus \mathcal{I}) \cup [\![\Phi]\!]_{\mathcal{K}}$. Then, $\mathcal{I} \not\subseteq Q_1$, so that checking during the fixpoint computation whether the initial states are still contained in the approximation will also stop after the first iteration.

Hence, there is slightly more effort, since we have to perform one step of the fixpoint iteration of the the AG operator.