Supporting Materials for
Unrealizable Cores for Reactive Systems Specifications

Abstract—This document provides supporting materials for the ICSE’21 paper submission titled “Unrealizable Cores for Reactive Systems Specifications”.

I. A REMARK ABOUT USING MEMOIZATION CORRECTLY

Recall the QuickCore algorithm (Alg. 1 below).

Algorithm 1 QuickCore Given an unrealizable specification find a locally minimal subset of guarantees that keeps it unrealizable

Require: An unrealizable specification $S = (V_e, V_s, D, M_e, M_s)$, where $M_e = (I_e, T_e, J_e)$ and $M_s = (I_s, T_s, J_s)$

Ensure: An unrealizability core of $S$

1. if Realizable($V_e, V_s, D, M_e, (I_s, T_s, \emptyset)$) then
   2. $J_c \leftarrow \text{MinWBase}(\text{DDMin}, B_{M_s}, I_s \cup T_s, J_s, \neg \text{Realizable}(S))$
   3. else
   4. $J_c \leftarrow \emptyset$
   5. $M_e \leftarrow (I_e, T_e, \emptyset)$
   6. end if

2. $T_c \leftarrow \text{MinWBase}(\text{DDMin}, B_{M_s}, I_s \cup J_c \cup T_s, J_s, \neg \text{Realizable}(S))$

3. $w \leftarrow \text{ComputeWinRegion}(V_e, V_s, D, M_e, (I_s, T_c, J_c))$

4. $I_c \leftarrow I_s$

      5. envIni $\leftarrow \land_{d \in I_c} d$

5. for $i \in I_c$ do
   6. $I_c \leftarrow I_c \setminus \{i\}$
   7. if SysWin(envIni, $\land_{d \in I_c} d, w$) then
   8. $I_c \leftarrow I_c \cup \{i\}$
   9. end if
10. end for
11. return $I_c \cup T_c \cup J_c$

Remark 1. Note that realizability results in QuickCore may correspond to specifications with an altered set of assumptions, because from line 7 of QuickCore onward, if no justice guarantees are required for a core, all environment justices are removed. These realizability check results are stored and used by the memoization of QuickCore and PQC. However, considerations detailed in the correctness of QuickCore ensure that in this case realizability checking results are the same for specifications where the justice assumptions are kept.

II. A REMARK WITH ADDITIONAL RESULTS

Some additional results concerning AM+GN which we use in the paper only for scaling, because they are automatically generated.

Remark 2. AM+GN specifications are parametric and are automatically generated. Therefore the nature of the cores of these specifications and finding all their cores, do not seem to be of practical interest. Accordingly, We did not report results for them in Tables 3 and 4. We did observe, however, that multiple cores abound for these specifications and that some of them have over 100 cores. Moreover QuickCore finds slightly smaller cores than DDMin for AMBA and GENBUF specifications, and both algorithms find cores which are about 15 times smaller than the total number of all guarantees on average.