Specifying and synthesizing Shield logically using DCSYNTH

Amol Wakankar, Paritosh K. Pandya and Raj Mohan Matteplackel

1 Formalization

Different notions of shield synthesis can be specified using hard and soft goals in DCSYNTH. We give several such definitions below. In the following, given a set of variables

- Conservative burst error shield.
  - Input: $I \cup O$. Output: $O1$
  - Hard requirement: `REQ[O/O1]`
  - Soft requirement: $\wedge_{o \in O}$`(true ^ <o=o1>)`
- Conservative k-shield-V1
  - Input: $I \cup O$. Output: $O1$
  - Hard requirement: `REQ[O/O1]&&[]([[`$\vee_{o \in O}$`(o`$\neq$`o1)]]=>slen<k)`.
- Conservative k-shield-V2.
  - Input: $I \cup O$. Output: $O1$
  - Hard requirement: `REQ[O/O1]&&[]([[ind && `$\vee_{o \in O}$`(o`$\neq$`o1)]]=>slen<k)`.
  - Indicator `ind : REQ(I,O)`
- Conservative k-shield-V3.
  - Input: $I \cup O$. Output: $O1$
  - Hard requirement: `REQ[O/O1]&&[]([[ind && `$\vee_{o \in O}$`(o`$\neq$`o1)]]=>slen<k)`.
  - Indicator `ind : exists O2. ((([O2=O1] || pt)^<O2=O>)  && REQ(I,O2)`

2 Experiments

In this section we give the time and the states for the following formulas for each sheild.

- $\varphi_0 = \mathcal{G}\ \neg q \vee \mathcal{F}_{\ <=n}(q\ \wedge \mathcal{F}_{\ <=n} p)$. The corresponding QDDC formula is given by `[[!q]] || (slen <= n) ^ <q> ^ (slen <= n) ^ <p> ^ true`.
- $\varphi_1 = \mathcal{G}\ \neg q \vee \mathcal{F}_{\ <=n}(q\ \wedge \mathcal{F}_{\ <=n} p)$. The corresponding QDDC formula is given by `[[!q]] || pref((slen <= n) ^ <q> ^ (slen <= n) ^ <p> ^ true)`.

- $\varphi_2 = \mathcal{G}\,((q \wedge \neg r) => \neg r \mathcal{U}_{<=n}(p \wedge \neg r))$. The corresponding QDDC formula is `[]((<q && !r> ^ true) => ((([!r]||pt) && slen <= n) ^ <p&&!r> ^ true) || ([[!r]] && slen<n))`.

- $\varphi_3 = \mathcal{G}\,((x => y \vee z) \wedge ((z \wedge \mathcal{X}\ {}^{n}true) => \mathcal{X}\ \mathcal{G}_{=n}!z)$. The corresponding QDDC formula is `[]((<x> => <y||z>) && ({{z}} ^ (slen==n) => (slen==1) ^ [[!z]]))`.

- $\varphi_4 = G_1 \wedge G_2 \wedge G_3$ where
  - $G_1 = \mathcal{G}\ (HREADY => \mathcal{X}\ START)$.
    QDDC equivalent of G_1 is `[]({{!HREADY}} => (slen == 1) ^ <!START>)`.
  - $G_2 = \mathcal{G}\ ((HMASTLOCK \wedge (HBURST = INCR) \wedge START \wedge HMASTERi) => \mathcal{X}\ (\neg START\ Unless\ \neg HBUSREQi))$.
    QDDC formula of G_2 is roughly $\forall i \in \{0, 1\}$: `[]({{HMASTLOCK && INCR && START && HMASTERi}} ^ true  => (slen == 1) ^ ([[!START]] || [[!START]] ^ <!HBUSREQi> ^ true))`.
  - $G_3 = \mathcal{G}\ ((HMASTLOCK \wedge HBURST = BURST4 \wedge START) => ((HREADY \wedge \mathcal{X}\ (\neg START\ \mathcal{U}_{3}\ HREADY)) \bigwedge (\neg HREADY \wedge \mathcal{X}\ (\neg START\ \mathcal{U}_{4}\ HREADY))))$.
    QDDC formula is `[](<HMASTLOCK && BURST4 && START> ^ true => ( {{HREADY}} ^ ([[!HREADY]] || (([[!START]] && (scount HREADY == 3)) ^ true))) || ({{!HREADY}} ^([[!HREADY]] || (([[!START]] && (scount HREADY == 4)) ^ true)))`.

Shield	Formula	n	k	States	Time
burst error	φ_0	16	-	36	1.404
	φ_1	16	-	4	1.372
	φ_2	12	-	14	0.064
	φ_3	4	-	7	0.012
	φ_4	-	-	6	0.056
K-Shield V1	φ_0	16	1	19	1.428
	φ_1	16	1	3	1.424
	φ_2	12	1	14	0.068
	φ_3	4	4	Unrealizable	0.016
	φ_4	-	1	Unrealizable	0.016
K-Shield V2	φ_0	16	1	174	1.516
	φ_1	16	1	6	1.512
	φ_2	12	1	27	0.160
	φ_3	4	4	16	0.052
	φ_4	-	1	24	0.460
K-Shield V3	φ_0	16	1	20	2.736
	φ_1	16	1	4	2.636
	φ_2	12	1	14	0.108
	φ_3	4	4	7	0.036
	φ_4	-	1	12	0.132

Table 1: Experiments for various shield synthesis notions.