

Theorie reaktiver Systeme

Refinement

Safety Tests

Requirement Coverage

Spezifikationsbasiertes Testen

Gegeben: Spezifikation P (abstrakt) und Implementierung Q (konkret)

Test: Feststellen von Abweichungen zwischen P und Q

Ziel: Alle Testfälle können alleine auf Grundlage der Spezifikation P konstruiert werden, um die Äquivalenz (hier: $\sim_{TE}$) zwischen P und Implementierung Q nachzuweisen.

Refinement

Eine *Verfeinerung* ist ein Transformationsschritt von einem (abstrakten) Programm (Spezifikation) *SPEC* zu einer konkreteren Implementierung *IMP*.

Dabei wird i.d.R. das Verhalten auf dem gleichen Alphabet betrachtet:

$$\alpha(SPEC) = \alpha(IMP) = I.$$

(Tatsächlich gilt häufig $IMP = X \setminus (\alpha(X) - I)$.)

Safety

Safety $\hat{=}$ "Nothing bad happens" [Lamport]

Die Implementierung darf nur solche Berechnungen ausführen, die die Spezifikation auch ausführen kann.

$$\textit{traces}(\textit{IMP}) \subseteq \textit{traces}(\textit{SPEC})$$

Trace Refinement:

$$\textit{SPEC} \sqsubseteq_T \textit{IMP} \Leftrightarrow \textit{traces}(\textit{IMP}) \subseteq \textit{traces}(\textit{SPEC})$$

Anmerkung: $\forall P : P \sqsubseteq_T \textit{STOP}$

Safety Tests

Menge von Tests, die entscheidet, ob $traces(Q) \subseteq traces(P)$.

Tests der Form $U(s, a)$ mit $s \in traces(P)$, $a \in \alpha(P)$ und $a \notin first(P/s)$.

$$U_S(s, a) = \begin{array}{l} \text{if } s = \langle \rangle \\ \text{then } \omega \rightarrow \text{STOP} \sqcap a \rightarrow \text{STOP} \\ \text{else } \omega \rightarrow \text{STOP} \sqcap head(s) \rightarrow U_S(tail(s), a) \end{array}$$

$$\mathcal{H}_{Trace}(P) = \{U_S(s, a) \mid s \in traces(P) \wedge a \in \alpha(P) \wedge a \notin first(P/s)\}.$$

P must $U_S(s, a)$ für alle $U_S(s, a) \in \mathcal{H}_{Trace}(P)$.

Falls Q must $U_S(s, a)$ für alle $U_S(s, a) \in \mathcal{H}_{Trace}(P)$,
so folgt $traces(Q) \subseteq traces(P)$.

$\mathcal{H}_{Trace}(P)$ ist minimal.

Requirement Coverage: Nach einer Berechnung mit Ereignisfolge s von IMP und $SPEC$ darf IMP keine Fortsetzung verweigern, die $SPEC$ nicht verweigern kann.

Konstruktion von Requirement Coverage Tests

Gegeben ist P in Head-Normal-Form,

$$P = \prod_{i \in I} (\Box a : B_i.a \rightarrow Q_i(a))$$

Ziel: Konstruktion einer *minimalen* Menge von Tests

Nötig:

- Abdecken der verschiedenen Möglichkeiten *nach* der internen Auswahl.
- Minimalität