Checking for Feasibility

Before we discuss model checking response properties, we discuss the problem of checking whether a given FDS is feasible. For every state \( s \exists s \text{-path leading from } s \text{ to some } S \text{-state.}

Each state \( s \exists s \subseteq S \text{ has a } d \text{-successor in } S.\)

All states in \( S \text{ are reachable.}\)

A set of states \( S \text{ is defined to be an } F \text{-set if it satisfies the following requirements:} F_1. \text{ All states in } S \text{ are reachable.} F_2. \text{ Each state } s \exists S \text{ has a } d \text{-successor in } S. F_3. \text{ For every state } s \subseteq S \text{ and every fairness requirement } J \exists \exists J, \text{ there exists an } S \text{-path leading from } s \text{ to some } J \text{-state.} F_4. \text{ For every state } s \subseteq S \text{ and every comparison requirement } (b, c) \exists \exists (b, d) \text{, either } s \text{ satisfies } b \text{ or } s \text{ satisfies } c.

Analysis of Reactive Systems, NYU, Fall, 2009

66
F-sets imply feasibility

Claim 5. [F-sets]

A reachable state $s$ is feasible if and only if there exists a path leading from $s$ to some $F$-set.
It is not difficult to see that the infinite sequence constructed in this way is a computation.

Continued.

Proof

Consider in turn each of the justice requirements \( J \). We append to the \( S \)-path \( \gamma \). We append to the \( S \)-set \( \gamma \). We know that \( \gamma \) has a successor \( s \). Append to the end of \( \gamma \). We consider in turn each of the justice requirements \( J \). We consider in turn each of the compassion requirements \((p,q)\) of \( C \). If there exists an \( S \)-path from \( \gamma \) to a \( q \)-state, we append \( q \) to the end of \( \gamma \). Otherwise, we do not modify \( \gamma \). We observe that if there does not exist an \( S \)-path leading from \( \gamma \) to a \( q \)-state, \( \gamma \) must satisfy (by \( \gamma \) state \( a \) and all of its progeny within must satisfy the state \( b \) state, then \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state). If there exists an \( S \)-path leading from \( \gamma \) to a \( f \)-state, we append \( f \) to the \( \gamma \) state. We consider in turn each of the compassion requirements \((p,q)\) of \( C \). If there exists an \( S \)-path from \( \gamma \) to a \( q \)-state, we append \( q \) to the end of \( \gamma \). Otherwise, we do not modify \( \gamma \). We observe that if there does not exist an \( S \)-path leading from \( \gamma \) to a \( q \)-state, then \( \gamma \) state \( a \) and all of its progeny within state \( b \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state, \( \gamma \) state). If there exists an \( S \)-path leading from \( \gamma \) to a \( f \)-state, we append \( f \) to the \( \gamma \) state. We consider in turn each of the justice requirements \( J \). We consider in turn each of the compassion requirements \((p,q)\) of \( C \).
Assume an assertion \( \phi \) which characterizes an \( F \)-set. Translating the requirements 1–4 into formulas, we obtain the following requirements:

\[
\begin{cases}
(b \lor \phi) \land (d \lor \phi) \land d \rightarrow \\
\forall \phi \land (f \lor \phi) \land (d \lor \phi) \lor \\
\exists (b,d) \\
\end{cases}
\]

as:

\[
\begin{cases}
(b \lor \phi) \land (d \lor \phi) \land d \rightarrow \\
\forall \phi \land (f \lor \phi) \land (d \lor \phi) \lor \\
\exists (b,d) \\
\end{cases}
\]

This can be summarized as:

For every \( \phi \in \mathcal{C} \):

\[
(b \lor \phi) \land (d \lor \phi) \land d \rightarrow 
\]

For every \( f \in \mathcal{F} \):

\[
(f \lor \phi) \land (d \lor \phi) \land 
\]

Every \( \phi \)-state has a \( d \)-successor:

\[
\phi \land d 
\]

Reachable:

\[
\rightarrow 
\]

Assume an assertion \( \phi \) which characterizes an \( F \)-set.
Algorithmic Interpretation

Computing the maximal $x$-point as a sequence of iterations, we can describe the computational process as follows:

Start by letting $\text{reachable}^x := \phi$. Then repeat the following steps:

1. Remove from $\phi$ all states which do not have a $\phi$-successor.
2. For each $f \in \phi$, remove from all states which do not have a $\phi$-path leading to a $f$-state.
3. For each $b \in \phi$, remove from all states which do not have a $\phi$-path leading to a $b$-state.

until no further change.

To check whether an FDS is feasible, we compute for it the maximal $F$-set and check whether it is empty. $\phi$ is feasible if and only if the maximal $F$-set is not-empty.

To check whether it is empty.

A.Pnueli

Analysis of Reactive Systems, NYU, Fall 2009
We conclude that the above FDS is feasible.

As an example, consider the following FDS:

\[
\begin{align*}
\forall & : x \\
\exists & : x \\
\exists & : x \\
\exists & : x \\
\exists & : x \\
\end{align*}
\]

Example

We set \( \{ \exists \} : \exists \) and then proceed as follows:

- Removing from \( \{ \exists \} \) all states which do not have a \( \exists \)-path leading to a \( \exists \)-state, we are left with \( \{ \exists \} : \exists \).
- Succesively removing from \( \{ \exists \} \) all states without successors, leaves \( \{ \exists \} : \exists \).
- Removing from \( \{ \exists \} \) all states which do not have a \( \exists \)-path leading to an \( \exists \)-state, we are left with \( \{ \exists \} : \exists \).
- Removing from \( \{ \exists \} : \exists \) all states which do not have an \( \exists \)-path leading to a \( \exists \)-state, we are left with \( \{ \exists \} : \exists \).

No reasons to remove any further states from \( \{ \exists \} : \exists \), so this is our final set.

We end up with \( \{ \exists \} : \exists \).
Let $D$ be a computation of which contains a $d$-state at position $k$, and has no following $\cdots, s_{k-1}, s_k, s_{k+1}, \cdots$, such that $s_0$ is $\Theta$-initial. The infinite sequence $s_0, s_1, s_2, \cdots$ exists, a finite sequence $b^d \Diamond \subseteq d$ states. This sequence violates a $d$-state of which contains a $d$-state at position $k$, and has no following $\cdots, s_{k-1}, s_k, s_{k+1}, \cdots$.

Thus, the infinite sequence $s_0, s_1, s_2, \cdots$ exists. By the definition of $\Theta$, $b^d \Diamond \subseteq d$. Indeed, let $\langle c, d, a \rangle : b^d \Theta$ be a computation of which contains a $d$-state which violates the response property $\Theta$, can be extended to a computation of which contains a $d$-state which violates the response property $\Theta$.

We define an auxiliary FDS $\langle c, d, \Theta \rangle : b^d \Theta$, where $\langle c, \Theta \rangle$ is the set of all reachable states in $\Theta$. Let $\langle c, d, \Theta \rangle : b^d \Theta$, be the assertion characterizing all the reachable states.

Thus, there exists a finite sequence $b^d \Diamond \subseteq d$, allows any $d$-step as long as the successor does not satisfy $b^d \Theta$. Thus, $b^d \Theta$ characterizes all the reachable states which do not satisfy $b^d \Theta$. We wish $b^d \Theta$, let $\langle c, d, \Theta \rangle : b^d \Theta$, be a response property we wish $b^d \Diamond \subseteq d$ and be an FDS.
Assume we wish to verify the property \( C_2 \). We start by forming MUX-SEM, whose set of reachable states is given by:

\[
N_1; T_2; 1; T_1; T_2; 1; C_1; T_2; 0; T_1; C_2; 0; N_1; C_2; 0; N_2; 1; T_1; N_2; 1; C_1; N_2; 0; T_1; C_2; 0; N_1; N_2; 1
\]

Following is the set of all reachable states of program MUX-SEM.

Example: MUX-SEM

\( C_2 \)-state. This leaves us with:

First, we eliminate all \((T_2 \wedge T_L) = 0\)-states which do not have a path leading to a

\( C_2 \) state.
Next, we eliminate all states which do not have a path leading to a $C_1$-state. This leaves us with nothing. We conclude that $\text{MUXSEM}(M) \leq \mu \iff C_2$. 

A. Pnueli

Analysis of Reactive Systems, NYU, Fall 2009

Lecture 4: Feasibility and Response