4. Test Generation from Finite State Models II

Model-Based Testing

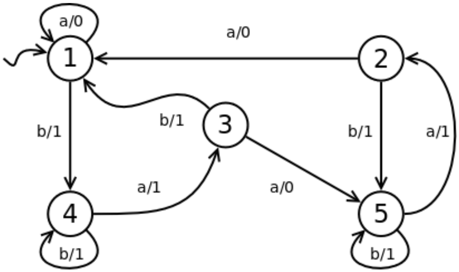
Andrea Polini

Software Engineering II – Software Testing
MSc in Computer Science
University of Camerino

December 9^{th}-11^{th}, 2014

Characterization Set

Let $M = < \mathscr{X}, \mathscr{Y}, \mathscr{Q}, q_1, \delta, \mathscr{O} >$ an FSM that is minimal and complete. A characterization set for M, denoted as $\mathscr{W}$, is a finite set of input sequences that distinguish the behaviour of any pair of states in M.



K-equivalence partitions

The notion of $\mathscr{K} - \textit{equivalence}$ leads to the notion of $\mathscr{K} - \textit{equivalence partitions}$.

Given an FSM a $\mathscr{K} - \textit{equivalence partition}$ of $\mathscr{Q}$, denoted by $\mathscr{P}_k$, is a collection of n finite sets of states denoted as $\Sigma_{k_1}, \Sigma_{k_2}, \ldots, \Sigma_{k_n}$ such that:

- $\cup_{i=1 \ldots n} \Sigma_{K_1} = \mathscr{Q}$
- States in Σ_{k_j}, for $1 \leq j \leq n$ are $\mathscr{K} - \textit{equivalent}$
- if $q_l \in \Sigma_{k_i}$ and $q_m \in \Sigma_{k_j}$, for $i \neq j$, then q_l and q_m must be $\mathscr{K} - \textit{distinguishable}$

How to derive $\mathcal{W}$ from K-equivalence partitions

1. Let M an FSM for which $P = \{P_1, P_2, ..., P_n\}$ is the set of k-equivalence partition. $\mathcal{W} = \emptyset$
2. Repeat the steps (a) through (d) given below for each pair of states (q_i, q_j), $i \neq j$, in M
   (a) Find r ($1 \leq r < n$ such that the states in pair (q_i, q_j) belong to the same group in P_r but not in P_{r+1}. If such an r is found then move to step (b) otherwise we find an $\eta \in \mathcal{X}$ such that $\mathcal{O}(q_i, \eta) \neq \mathcal{O}(q_j, \eta)$, set $\mathcal{W} = \mathcal{W} \cup \{\eta\}$ and continue with the next available pair of states. The length of the minimal distinguishing sequence for (q_i, q_j) is $r + 1$.
   (b) Initialize $z = \epsilon$. Let $p_1 = q_i$ and $p_2 = q_j$ be the current pair of states. Execute steps (i) through (iii) given below for $m = r, r-1, ..., 1$
      (i) Find an input symbol η in P_m such that $\mathcal{G}(p_1, \eta) \neq \mathcal{G}(p_2, \eta)$. In case there is more than one symbol that satisfy the condition in this step, then select one arbitrarily.
      (ii) set $z = z\eta$
      (iii) set $p_1 = \delta(p_1, \eta)$ and $p_2 = \delta(p_2, \eta)$
   (c) Find an $\eta \in \mathcal{X}$ such that $\mathcal{O}(p_1, \eta) \neq \mathcal{O}(p_2, \eta)$. Set $z = z\eta$
   (d) The distinguishing sequence for the pair (q_i, q_j) is the sequence z. Set $\mathcal{W} = \mathcal{W} \cup \{z\}$

Example

- Termination of the $\mathscr{W}$ – *procedure* guarantees the generation of distinguishing sequence for each pair.

S_i	S_i	x	$\mathscr{O}(S_i, x)$	$\mathscr{O}(S_j, x)$
1	2	baaa	1	0
1	3	aa	0	1
1	4	a	0	1
1	5	a	0	1
2	3	aa	0	1
2	4	a	0	1
2	5	a	0	1
3	4	a	0	1
3	5	a	0	1
4	5	aaa	1	0

Example

- Termination of the $\mathscr{W} - procedure$ guarantees the generation of distinguishing sequence for each pair.

S_i	S_i	x	$\mathscr{O}(S_i, x)$	$\mathscr{O}(S_j, x)$
1	2	baaa	1	0
1	3	aa	0	1
1	4	a	0	1
1	5	a	0	1
2	3	aa	0	1
2	4	a	0	1
2	5	a	0	1
3	4	a	0	1
3	5	a	0	1
4	5	aaa	1	0

Identification Set

An identification set is associated to each state $q \in \mathscr{Q}$ of an FSM.

An Identification set for state $q_i \in \mathscr{Q}$, where $|\mathscr{Q}| = n$, is denoted by $\mathscr{W}_i$ and has the following properties:

1. $\mathscr{W}_i \subseteq \mathscr{W}$ per $1 < i \leq n$
2. $\exists j, s. 1 \leq j \leq n \land s \in \mathscr{W}_i \land \mathscr{O}(q_i, s) \neq \mathscr{O}(q_j, s)$
3. No subset of $\mathscr{W}_i$ satisfies property 2.

The W-Method

The W-Method aims at deriving a test set to check the implementation (Implementation Under Test - IUT) of an FSM model

Assumptions

- M is completely specified, minimal, connected, and deterministic
- M starts in a fixed initial states
- M can be reset to the initial state. A `null` output is generated by the reset
- M and IUT have the same input alphabet

The W-Method

The W-Method aims at deriving a test set to check the implementation (Implementation Under Test - IUT) of an FSM model

Assumptions

- M is completely specified, minimal, connected, and deterministic
- M starts in a fixed initial states
- M can be reset to the initial state. A `null` output is generated by the reset
- M and IUT have the same input alphabet

W-Method steps

Given an FSM $\mathscr{M} =< \mathscr{X}, \mathscr{Y}, \mathscr{Q}, q_0, \delta, \mathscr{O} >$ the W-method consists of the following steps:

1. Estimate the maximum number of states in the correct design
2. Construct the characterization set $\mathscr{W}$ for the given machine $\mathscr{M}$
3. Construct the testing tree for $\mathscr{M}$ and determine the transition cover set $\mathscr{P}$
4. Construct set $\mathscr{Z}$
5. $\mathscr{P} \cdot \mathscr{Z}$ is the desired test set

Computation of the transition cover set

$\mathscr{P}$ - transition cover set

Let q_i and $q_j, i \neq j$ be two states of $\mathscr{M}$. $\mathscr{P}$ consists of sequences $s \cdot x$ s.t. $\delta(q_0, s) = q_i \wedge \delta(q_i, x) = q_j$ for $s \in \mathscr{X}^* \wedge x \in \mathscr{X}$. The set can be constructed using the testing tree for $\mathscr{M}$.

Testing tree

The testing tree for an FSM $\mathscr{M}$ can be constructed as follows:

1. State q_0 is the root of the tree
2. Suppose that the testing tree has been constructed till level k. The $(k+1)^{th}$ level is built as follows:
   - Select a node n at level k. If n appears at any level from 1 to $k-1$ then n is a leaf node. Otherwise expand it by adding branch from node n to a new node m if $\delta(n, x) = m$ for $x \in \mathscr{X}$. This branch is labeled as x.

Computation of the transition cover set

$\mathscr{P}$ - transition cover set

Let q_i and $q_j, i \neq j$ be two states of $\mathscr{M}$. $\mathscr{P}$ consists of sequences $s \cdot x$ s.t. $\delta(q_0, s) = q_i \wedge \delta(q_i, x) = q_j$ for $s \in \mathscr{X}^* \wedge x \in \mathscr{X}$. The set can be constructed using the testing tree for $\mathscr{M}$.

Testing tree

The testing tree for an FSM $\mathscr{M}$ can be constructed as follows:

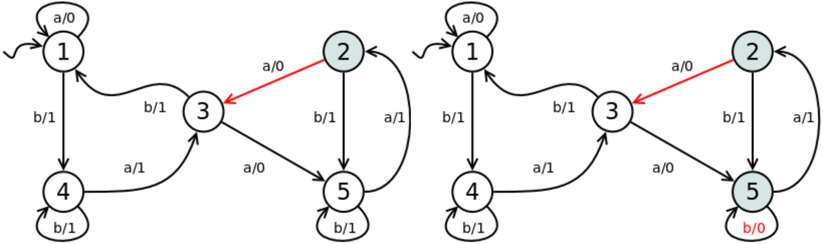
1. State q_0 is the root of the tree
2. Suppose that the testing tree has been constructed till level k. The $(k+1)^{th}$ level is built as follows:
    - Select a node n at level k. If n appears at any level from 1 to $k-1$ then n is a leaf node. Otherwise expand it by adding branch from node n to a new node m if $\delta(n, x) = m$ for $x \in \mathscr{X}$. This branch is labeled as x.

Constructing $\mathscr{Z}$

Suppose number of states estimates to be m for the IUT, and n in the specification $m > n$. We compute $\mathscr{Z}$ as:

$$\mathscr{Z} = (\mathscr{X}^0 \cdot \mathscr{W}) \cup (\mathscr{X} \cdot \mathscr{W}) \cup (\mathscr{X}^1 \cdot \mathscr{W}) \cdots \cup (\mathscr{X}^{m-1-n} \cdot \mathscr{W}) \cup (\mathscr{X}^{m-n} \cdot \mathscr{W})$$

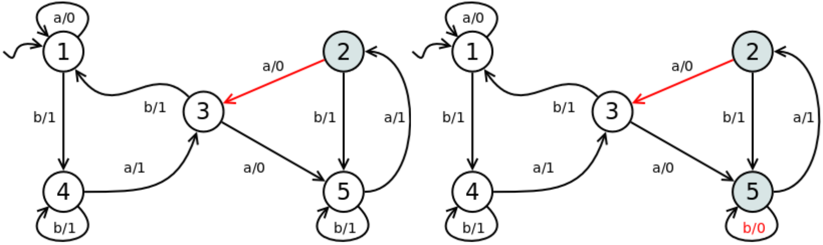
Deriving a test set – $\mathscr{P} \cdot \mathscr{Z}$



Try sequences:

- *baaaaaa*
- *baaba*

Deriving a test set – $\mathscr{P} \cdot \mathscr{Z}$



Try sequences:

- *baaaaaa*
- *baaba*