Regression Test Suite Reduction Using Extended Dependence Analysis

SOQUA '07 Fourth international workshop on Software quality assurance: in conjunction with the 6th ESEC/FSE joint meeting

201273275 박두호

PROGRAM Language

KU KONKUK UNIVERSITY

Index

- Background
- Model-Based Regression Testing
- Regression Test Suit Reduction
- Conclusion

Background [1/5]

- Regression test suite(RTS) may not need to target the same coverage as an original test suite.
  - Only part of the SUT will be tested by the RTS. RTSs are selected to test the SUT to verify that modifications have not caused unintended effects and that the SUT complies with the changes in the requirements.
  - Frequent regression testing during software maintenance suggests minimizing the size of the RTS

Background [2/5]

EFSM (Extended Finite State Machine)

An **EFSM** is a 5-tuple <*S, I, O, V, T*> where

- *S* is a nonempty finite set of states with two states designated as *Start* and *Exit* states of the EFSM
- *I* is a nonempty finite set of input interactions, each with a (possibly empty) set of input interaction parameters
- *O* is a nonempty finite set of output interactions, each with a (possibly empty) set of output interaction parameters
- *V* is the nonempty finite set of all variables which is the union of set of all local variables and set of all interaction parameters
- *T* is a nonempty finite set of transitions

Each transition $t \in T$ is a 6-tuple $<s_s, s_t, i, c, o, a>$ where

- s_s , $s_t \in S$ are the *starting* and *terminating* states of *t*
- $i \in I$ is the input interaction of *t*
- *c* is the *enabling condition* of *t* which is a Boolean expression defined over the set of all local variables and set of all input interaction parameters
- $o \in O$ is the output interaction of *t*
- *a* is a sequence of actions of *t* expressed as functions $f: V \rightarrow V$

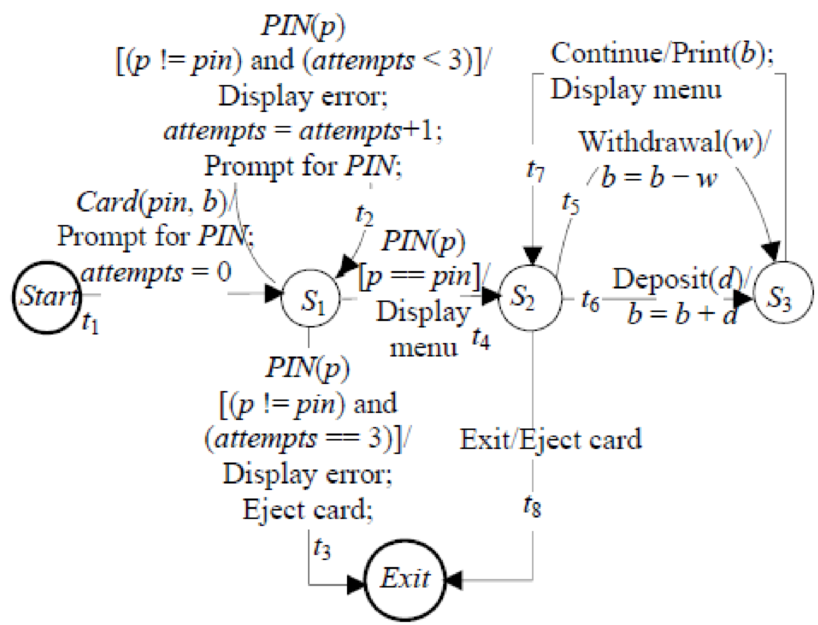

Figure1. EFSM model of a simplified ATM system

- For each transition, we assume there is at most one Elementary Modification(EM) in the given set of elementary modifications. Multiple modifications on the same transition are therefore assumed to be combined into one EM. … **data and control dependences may exist between transitions.**

Background [3/5]

❖ Data Dependence

- captures the notion that one transition defines a value for a variable and the same or some other transition may potentially use this value.

A definition (def) of $v \in V$ is an occurrence of v in a transition by which v takes a value (e.g., an occurrence of v on the left hand side of an action or in an input interaction). A use of $v \in V$ is an occurrence of v in a transition which directly affects the computation being performed (e.g., an occurrence of v on the right hand side of an action), or allows one to see the result of some earlier definitions (e.g., an occurrence of v in an output interaction), or directly affects the control flow in the EFSM (e.g., an occurrence of v in the enabling condition). A *path* (t_1 t_2 … t_{m-1} t_m) is a sequence of adjacent transitions. A path (t_1 t_2 … t_{m-1} t_m) is said to be from the *starting state* of t_1 to the *terminating state* of t_m. A path (t_1 t_2 … t_{m-1} t_m) is a *def-clear path from t_1 to t_m with respect to (w.r.t.)* $v \in V$ if either $m = 2$ or $m > 2$ and v is not defined at t_2 … t_{m-1}. A pair (def of v at t_1 and use of v in t_m) is a *def-use pair* (*du-pair*) w.r.t. v if def of v at t_1 is the last def of v at t_1, use of v at t_m is a use of v in t_m (before v is (possibly) defined at t_m), and there is a def-clear path from t_1 to t_m w.r.t. v [9].

Suppose that t_j and t_k are transitions, and $v \in V$ in an EFSM. There is a *data dependence* (***DD***) *from t_j to t_k* w.r.t. v, denoted (t_j, t_k, v), iff there is a du-pair (def of v in t_j, use of v in t_k) w.r.t. v. In the example EFSM, t_1 defines b, t_5 uses b, and along t_1 (t_2) t_4 t_5, b is not redefined. Thus, there exists a DD from t_1 to t_5 w.r.t. b. Also, t_5 defines b, t_5 uses b, and along t_5 t_7 t_5, b is not redefined. Thus, a DD exists from t_5 to t_5 w.r.t. b.

Data Dependence Terminology

Background [4/5]

❖ Control Dependence

- captures the notion that one transition may "influence" the traversal of another transition.

Suppose that S_1 and S_2 are two distinct states, and t is an outgoing transition from S_1 in an EFSM. Then, S_2 post-dominates S_1 iff S_2 is on every path from S_1 to *Exit* and S_2 post-dominates t iff S_2 is on every path from S_1 to *Exit* through t. In our example EFSM, S_2 does not post-dominate S_1, while S_2 post-dominates t_4.

Suppose that t_j and t_k are outgoing transitions from S_1 and S_2, respectively. There is a *control dependence* (**CD**) from t_j to t_k, denoted (t_j, t_k), iff S_2 does not post-dominate S_1 and S_2 post-dominates t_j. In our example EFSM, S_2 does not post-dominate S_1 but S_2 post-dominates t_4. Thus, there is a CD from t_4 to t_5.

Concept of post-dominance

Background [5/5]

- Static Dependence Graph (*SDG*)
  - graphically represents DDs and CDs in an EFSM

Static Dependence Graph (*SDG*) graphically represents DDs and CDs in an EFSM. In SDG, nodes represent EFSM transitions and directed edges represent DDs and CDs. Let D and C be the set of all DDs and CDs in an EFSM, respectively. That is, $D = \{(t_j, t_k, v) | (t_j, t_k, v)$ is a DD from t_j to t_k w.r.t. $v\}$ and $C = \{(t_j, t_k) | (t_j, t_k)$ is a CD from t_j to $t_k\}$. The SDG of a given EFSM is constructed as a directed graph $G(N, E)$ as follows:

Let $t_j, t_k \in T$ and $v \in V$ of the EFSM.

$E \leftarrow \varnothing$; $N \leftarrow \{n_i | n_i$ for each $t_i \in T\}$

For each $(t_j, t_k) \in C$, $E \leftarrow E \cup \{$a dashed edge from t_j to $t_k\}$.

For each $(t_j, t_k, v) \in D$, $E \leftarrow E \cup \{$a solid edge from t_j to $t_k\}$.

Figure 2 shows SDG of the example ATM system.

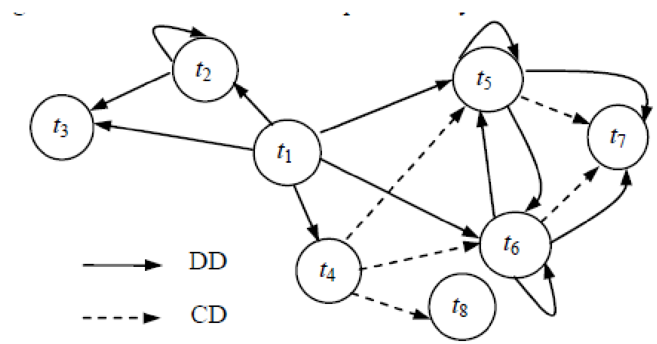

Figure 2. SDG for the EFSM example

Model-Based Regression Testing

- Changes in the requirements lead to modifications in the EFSM model representing the SUT.
  - Testing the effects of the model on the modification
  - Testing the effects of the modification on the model
  - Testing the side-effects of the modification on the unmodified parts of the model.

Effects of the Addition of a Transition

In a modified EFSM model, addition of a transition t_i may

- introduce new DDs and/or new CDs representing the effects of the model on t_i which are called *Affecting DD* and *Affecting CD*, respectively [20].
- introduce new DDs and/or new CDs representing the effects of t_i on the model which are called *Affected DD* and *Affected CD*, respectively [20].

We observe that since t_i did not exist in the original EFSM model, there were neither existing CDs nor existing DDs involving the added transition that could be eliminated. Besides being directly involved in forming both Affecting DDs/Affecting CDs and Affected DDs/Affected CDs in a modified EFSM model, an added transition t_i may also have indirect effects on a modified EFSM model. That is, in a modified EFSM model, addition of a transition t_i may

- introduce new DDs between other transitions which are called *Activation DD* [11],
- introduce new CDs between other transitions which we call *Activation CD*,
- eliminate existing CDs between other transitions which we call *Activation GCD*.

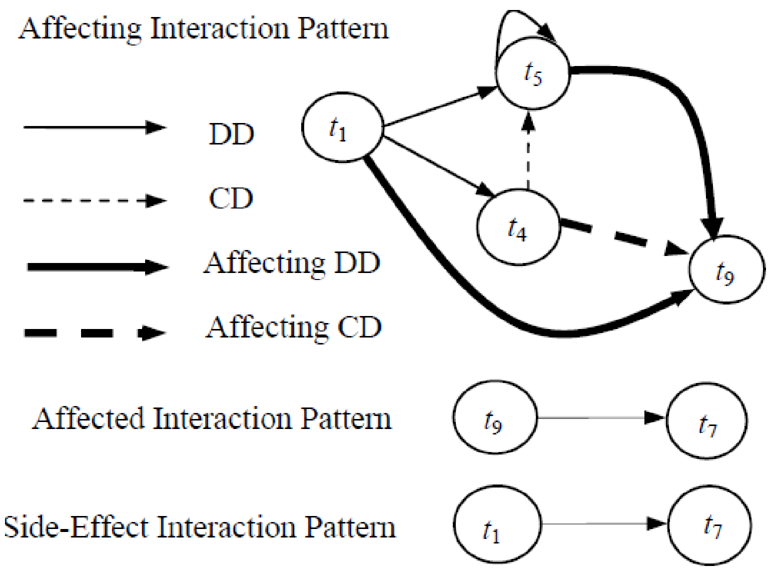

Figure 3. Interaction patterns for added transition t_9

❖ Effects of the Deletion of a Transition

Since a deleted transition does not exist in a modified EFSM, we adopt the scheme used by Korel et al. in [11]: for each deleted transition t_i, a new transition td_{new}, with an empty sequence of actions, is added to the modified EFSM model at the starting state of t_i to represent $i(t_i)$ and $c(t_i)$.

In a modified EFSM model, deletion of a transition t_i may

- eliminate existing DDs associated with t_i where t_i was dependent on another transition which are called *Affecting GDD* [11],
- eliminate existing CDs associated with t_i where t_i was dependent on another transition which we call *Affecting GCD*,
- eliminate existing DDs associated with t_i where some transitions were dependent on t_i which are called *Affected GDD* [20],
- eliminate existing CDs associated with t_i where some transitions were dependent on t_i which we call *Affected GCD*,
- introduce new CDs between other transitions which we call *Activation CD*,
- eliminate existing DDs between other transitions which is called *Activation GDD* [20],
- eliminate existing CDs between other transitions which we call *Activation GCD*.

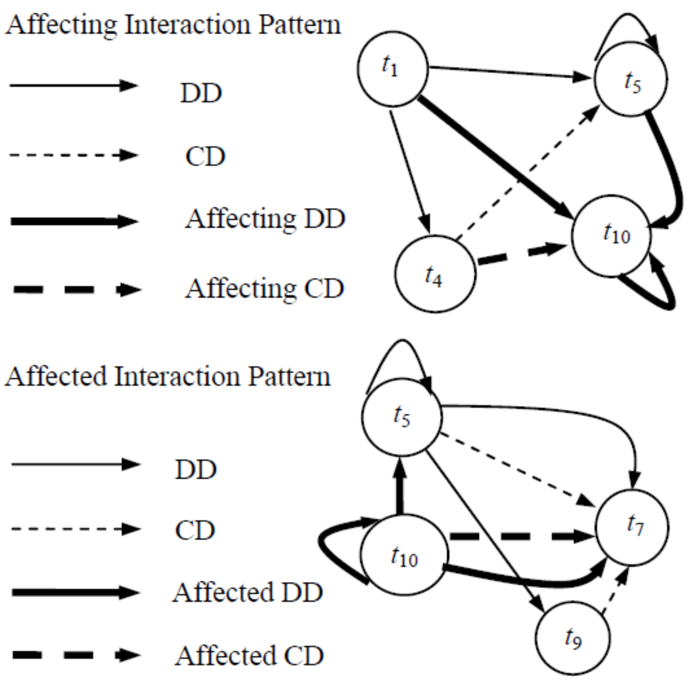

Figure 4. Interaction patterns for deleting t_6

Effects of a Changed Transition

In a modified EFSM model, changes in a transition t_i may

- introduce new DDs representing the effects of the model on t_i which we call *Affecting DD*,
- eliminate existing DDs associated with t_i where t_i was dependent on another transition which we call *Affecting GDD*,
- introduce new DDs representing the effects of t_i on the model which we call *Affected DD*,
- eliminate existing DDs associated with t_i where some transitions were dependent on t_i which we call *Affected GDD*,
- introduce new DDs between other transitions which we call *Activation DD*,
- eliminate existing DDs between other transitions which we call *Activation GDD*.

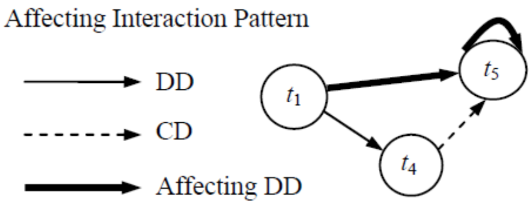

Figure 5. Interaction patterns for changing t_5

RTS REDUCTION

Given an original EFSM model R_o, its static dependence graph SDG_o, a regression test suite RTS, and a set M of EMs, the proposed RTS reduction method has the following steps:

- Step (1) applies M to R_o to get the modified EFSM model R_M, and then generates SDG_M for R_M using R_o, R_M, M and the definitions of NDPMs given in Section 3,
- Step (2)
  * for each EM m in M, identifies a subset RTS_m of RTS consisting of test cases in RTS containing the transition corresponding to m (note that a test case in RTS will then be in p subsets of RTS where $p << |M|$, is the number of EMs in M related to the transitions in the test case),
  * for each EM m in M and for each test case ts in RTS_m,
    + constructs up to 3 interaction patterns for ts by using SDG_o and SDG_M according to the definitions of interaction patterns given in Section 3,
    + ts is included in the reduced RTS if at least one of its interaction patterns has not been produced for any of the test cases in the reduced RTS (If the same interaction pattern of a certain type is produced for two different test cases for m, then these test cases are considered equivalent w.r.t. m and thus one of them is eliminated).

Step (1) of the proposed method is a straight-forward process, and Step (2) implements our discussions in Section 3 and compares the identified interaction patterns to determine which test cases yield at least one unique interaction pattern.

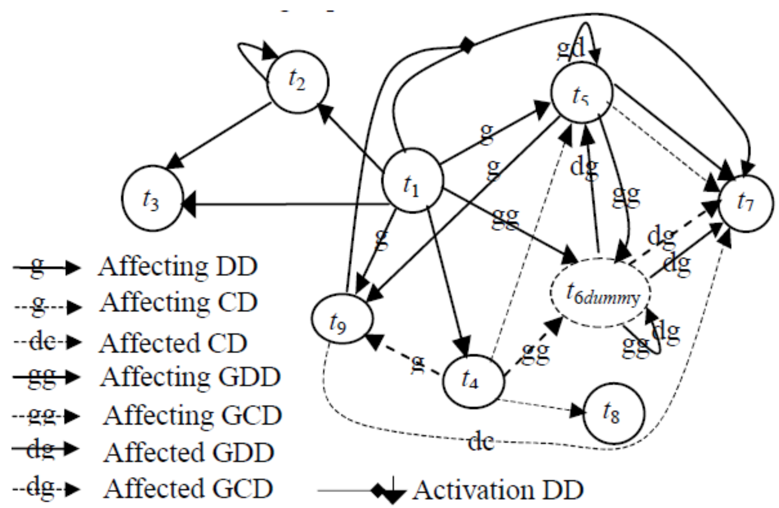

Figure 6. Modified SDG for the Simplified ATM Model

Conclusion

- Presented a regression test suite reduction method for a give set of EMs.

- The reduced RTS still facilitates testing both direct effects of the EMs on the changed parts of the SUT and indirect effects of the EMs on the unchanged parts of the SUT.