Chapter Three: Theory of Program Testing

Sem. II - 2017

Department of Software Engineering
ITSC-AAIT

Natnael A.

Basic Concepts in Testing Theory

- The idea of program testing is as old as computer programming.
- In 1970, a new field of research called testing theory emerged.
- Testing theory puts emphasis on
  - Detecting defects through program execution
  - Designing test cases from different sources: requirement specification, source code, and input and output domains of programs
  - Selecting a subset of tests cases from the entire input domain
  - Effectiveness of test selection strategies
  - Selection of Test oracles used during testing
  - Prioritizing the execution of test cases
  - Adequacy analysis of test cases

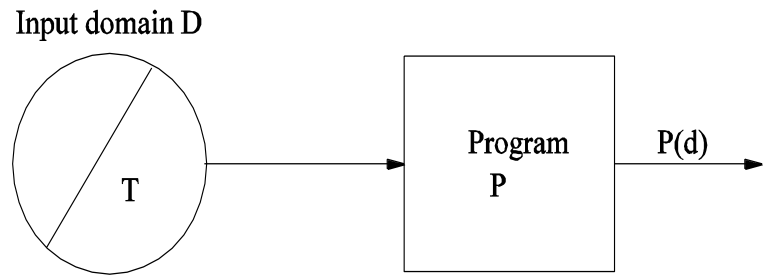
Basic Concepts in Testing Theory

- A theoretical foundation of testing gives testers and developers valuable insight into software systems and development processes.
- As a consequence, testers design more effective test cases at a lower cost.
- Any testing theory must inherit the fundamental limitation of testing. The limitation of testing has been best articulated by Dijkstra: Testing can only reveal the presence of errors, never their absence.
- There are three well known testing theories:
  - Theory of Goodenough and Gerhart (ideal test)
  - Theory of Weyuker and Ostrand (uniformly ideal test)
  - Theory of Gourlay (specifications first)

Theory of Goodenough and Gerhart

Fundamental Concepts

- Let P be a program, and D be its input domain. Let T $\subseteq$ D. P(d) is the result of executing P with input d.



- **OK(d)**: Represents the acceptability of **P(d).** OK(d) = true iff P(d) is acceptable.
- **SUCCESSFUL(T)**: T is a successful test iff $\forall t \in T$, OK(t).
- **Ideal Test:** T is an ideal test if **OK(t)**, $\forall t \in T \Rightarrow$ OK(d), $\forall d \in D$.
- An Ideal test is interpreted as follows. If from the successful execution of a sample of the input domain we can conclude that the program contains no errors , then the sample constitutes an ideal test.

Theory of Goodenough and Gerhart

- **Test Selection**
  - **Reliable Criterion**: A test selection criterion C is reliable iff either every test selected by C is successful, or no test selected is successful.
  - **Valid Criterion**: A test selection criterion C is valid iff whenever P is incorrect, C selects at least one test set T which is not successful for P.
- Fundamental Theorem

$(\exists T \subseteq D)\ \underline{(\text{COMPLETE}(T,C) \wedge \text{RELIABLE}(C) \wedge \text{VALID}(C) \wedge \text{SUCCESSFUL}(T))} \Rightarrow (\forall d \in D)\ \text{OK}(d)$

- Thorough test T is defined with exhaustive or complete test in which case T = D
- C defines the properties of a program that must be exercised to constitute a thorough test.

Theory of Goodenough and Gerhart

❑ Theory of Testing

- Let C denote a set of test predicates. If $d \in D$ satisfies test predicate $c \in C$, then c(d) is said to be true.

$$\text{COMPLETE}(T, C) \equiv (\forall c \in C)(\exists t \in T)\, c(t) \wedge (\forall t \in T)(\exists c \in C)\, c(t)$$

- For every test predicate, we select **a test** such that the **test predicate is satisfied**. Also **for every test** selected there exists **a test predicate** w/c is satisfied by the selected test.
- Let's assume *P* fails on input *d*. In other words, the actual outcome of executing *P* with input *d* is not the same as the expected outcome, **¬OK(*d*)** is true.
- **VALID(*C*)** implies that there exists a complete set of test data *T* such that **¬SUCCESSFUL(*T*).**
- **RELIABLE(*C*)** implies that if one complete test fails, all tests fail. However, this leads to a contradiction that there exists a complete test that is successfully executed.

Theory of Goodenough and Gerhart

- Finding a reliable and valid criterion enables to detect all faults with small set of test cases. However, this is impossible because of the following reasons.
  - Faults in a program are unknown. A criterion is guaranteed to be both reliable and valid if it selects the entire input domain D.
  - Neither reliability nor validity is preserved during the debugging process, where faults keep disappearing.
  - If P is correct, Test selection criteria are reliable and valid. But if P is incorrect in general no way of knowing whether a criterion is ideal w/o knowing the errors in *P*.

Theory of Goodenough and Gerhart

- **Program Errors**
  - Any approach to testing is based on **assumptions about the way program faults occur**. Faults are due to two main reasons:
    - inadequate understanding of all **conditions** that a program must deal with.
    - failure to realize that certain combinations of conditions require special **care**.
- **Goodenough and Gerhart classify program faults as follows:**
  - **Logic fault** (fault present in the program not because of the lack of the resource)
    - **Requirement fault** - Fault of capture the real requirement of the customer
    - **Design fault** - Failure to satisfy and understood requirements
    - **Construction fault** - Failure to satisfy the design
  - **Performance fault** - leads to failure of the program to produce expected result within specified or desired resources limitation.

Theory of Goodenough and Gerhart

- Sources of Faults

- **Missing control-flow paths** - A path may be missing from a program if we fail to identify a condition and specify a path to handle that condition (division by zero)
- **Inappropriate path selection**
  - A program execute an Inappropriate path if a condition is expressed incorrectly

Desired behavior	Implemented behavior
if (A) proc1(); else proc2();	if (A && B) proc1(); else proc2();

Example of inappropriate path selection.

- **Inappropriate or missing action**
  - Calculate a value using a method that does not necessarily give the correct result (Ex. Desired → X=X*W , The Actual → X=X+W , X= 1.5 , W=3) , Failing to assign a value to a variable or Calling a function with the wrong argument.

Theory of Goodenough and Gerhart

- **Conditions for Reliability of a set of test predicates C**

- A set of test predicates must at least satisfy the following conditions to have any chance of being reliable
  - Every branching condition must be represented by a condition in C.
  - Every potential termination condition must be represented in C.
  - Every condition relevant to the correct operation of the program must be represented in C.
- **Drawbacks of the Theory**
  - Difficulty in assessing the **reliability and validity of a criterion.**
  - The concepts of reliability and validity are defined **w.r.t.** to a program. The goodness of a test should be independent of individual programs.
  - Neither reliability nor validity is preserved throughout the debugging process.

Theory of Weyuker and Ostrand

- They proposed the concept of a uniformly ideal test selection criterion for a given out put specification.
- d $\in$ D is the input domain of program P and T $\subseteq$ D.
- **OK(P, d)** = true iff P(d) is acceptable.
- **SUCC(P, T):** T is a successful test for P iff $\forall$t $\in$ T, OK(P, t).
- **Uniformly valid criterion**: Criterion C is uniformly valid iff

  $$(\forall P)\ [\ (\exists d \in D)(\neg OK(P, d)) \Rightarrow (\exists T \subseteq D)\ (C(T) \wedge \neg SUCC(P, T))\ ].$$

- **Uniformly reliable criterion**: Criterion C is uniformly reliable iff

  $$(\forall P)\ (\forall T1, \forall T2 \subseteq D)\ [\ (C(T1) \wedge C(T2)) \Rightarrow (SUCC(P, T1) \longleftrightarrow SUCC(P,T2))\ ]$$

- **Uniformly Ideal Test Selection**
  - A uniformly ideal test selection criterion for a given specification is both uniformly valid and uniformly reliable.

Theory of Weyuker and Ostrand

- A subdomain S is a subset of D
  - Criterion C is revealing for a subdomain S if whenever S contains an input which is processed incorrectly, then every test set which satisfies C is unsuccessful. In other word any test selected by C is successfully executed, then every test in S produces correct output. A predicate called REVEALING (C,S)

REVEALING(C, S) iff $(\exists d \in S)(\neg OK(d)) \Rightarrow (\forall T \subseteq S)(C(T) \Rightarrow \neg SUCC(T))$

Theory of Gourlay

- The theory assumes that the program specifications are correct, and the specification is the sole arbiter of the correctness of a program. The program is said to be correct if it satisfies its specification.
- Gourlay's theory aims to establish a relationship between three sets of entities namely **specifications S, programs P, and Tests T**.
- We can then extend the OK predicate as follows:
  - OK (p, t, s) : The predicate is true if the result of testing **p with t** is judged to be successful with respect to specification **s**. We aim to make the predicate **OK(p, t, s)** true for every t in T, where t is a subset of T
- In this context a program is correct with respect to its specifications denoted by **CORR(p,s), iff OK(p, s, t) for every t in T.**

Testing Systems

- A testing system is defined as a collection of **<P, S, T, CORR, OK>** for which for every **p, s, t** in P, S, T (where P, S, T are subsets of $\mathcal{P}, \mathcal{S}, \mathcal{T}$) **CORR(p, s)** implies **OK(p, s, t).**
- **1. Set Construction System**
  - The set construction corresponds to a test that consists of **a set of trials**, and **success of the test** as a whole depends on the success of all trials.
  - Failure of any one run is enough to invalidate the test.
- **2. Choice Construction System**
  - The choice construction models the situation in which a test engineer is given a number of alternative ways of testing the program, all of which is assumed to be equivalent.

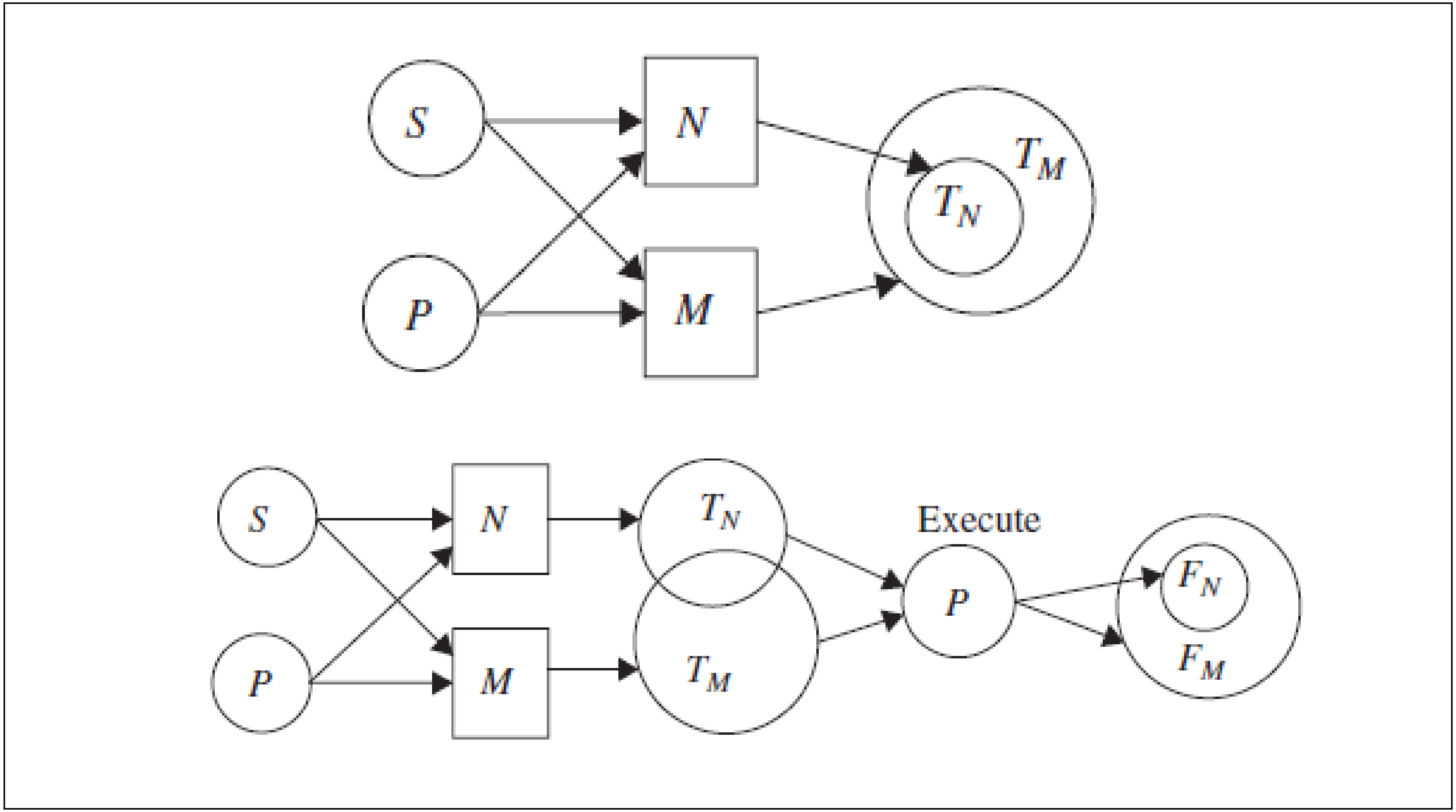
Test Methods

- A test method can be considered as a function M: P X S → T
- That is, in the general case, a test method takes a program P, and a specification S, and produces test cases T (where P, S, T are subsets of P , S, T respectively).
- Test methods can be:
  - **Program Dependent T** = M(P) → (White Box Testing)
  - **Specification Dependent T** = M(S) → (Black Box Testing)
  - **Expectation Dependent T** = M(S’), where S’ are the expectations of the customers, or the view the customers have on the specifications → (Acceptance Testing)

Power of Test Methods

- A fundamental problem in testing is to assess whether one test method is better than another (in recovering faults).
- Let M and N be two testing methods, and let F_M, F_N be the faults that can be discovered by M and N respectively.
- For M to be *at least as good as* N, we must have the situation that whenever N finds a fault, so does M. **In other words F_N is a subset of F_M.**
- Let **T_N and T_M**, be the test cases produced by methods **N and M** respectively. The **"investigative"** power of methods N and M can be classified in two cases
  - **Case 1: $T_N \subseteq T_M$**. In this case, method M is at least as good as method N
  - **Case 2:** TN and TM overlap, but TN $\underline{\subset}$ TM. This case suggests that TM does not totally contain TN and in order to compare their fault detection ability we execute program P under both test sets **TN, TM**. Let FN and FM be the sets of faults discovered by TN and TM. If FN $\subseteq$ FM then we say that method M is at least as good as N.

Power of Test Methods



Reading

Kshirasagar Naik and Priyadarshi Tripathy, “Software Testing and Quality Assurance - Theory and Practice”, University of Waterloo, 2008.

Page [32-46]

And read other online references