On Comparison of Random, Partition, and Proportional Partition Testing

In IEEE Transactions on Software Engineering, October 2001

Presentation by Nada Hashmi

Road Map

- Introduction
- Previous Studies
- The Experiment
- Results: Proportional Partition vs Random
- Results: Partition vs. Random with other factors
- Conclusion

Introduction

- Random Testing
  - Random selection of test cases from ENTIRE input domain
  - Successfully employed in practice
  - Relatively easy to employ
    - Effort
    - Cost
  - Problems
    - No guarantee
    - Not systematic

Introduction (cont)

- Partition Testing
  - Partition into disjoint sub-domains
  - Coverage of all sub-domains
  - At least one test case from each sub-domain
  - Problems
    - Can not truly have disjoint sub-domains
    - Is one test case per sub-domain enough?

Introduction (cont)

- Proportional Partition Testing
  - Associates a probability *p* to each sub-domain
  - Probability *p* based on when that sub-domain is likely to occur.
  - *n* testcases to the *k* sub-domains according to *p*
  - Example
    - Grades
  - Problems
    - 2 sub-domains and a million test cases

Previous Studies

- Duran and Ntafos
  - July 1984
  - Performance of random and partition very close
  - Random more cost-effective than partition
- Hamlet Taylor
  - Dec 1990
  - Similar results

Previous Studies (cont)

- Weyuker and Jeng
  - July 1991
  - Partition testing 'at least' as well as random
  - "IF" sub-domains are of equal size
- Most Studies:
  - 'Proportional partition testing' as the strategy increases the chances of proper coverage of the particular sub-domain

Previous Studies (cont)

- Problems:
  - Non-realistic approach- looked at:
    - How many times random testing did better than proportional partition testing
    - $P_r > P_p$
    - What about 'did as well as'?
  - Other factors
    - Cost
    - Effectiveness

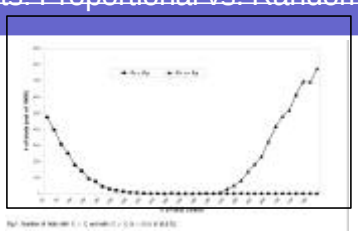
The Experiment

- "Simulations"
  - Details in Duran and Ntafos , July 1984
  - *k* = 20 sub-domains
  - 20 <= *n* <= 800 test cases
  - Probability and failure rate for (k,n) generated
  - Each experiment run 1000 times
  - P_r = Probability of random test detecting at least one failure
  - P_p = Probability of partition test detecting at least one failure

The Experiment (cont)

- Proportional partition vs. random testing
  - Three sets of experiments
  - Small number of sub-domains
  - Vary the number of sub-domains
  - Vary the number of test cases
  - Vary the failure rates
- Partition vs. random testing with cost and effectiveness as factors
  - Five sets of experiments
  - Vary the failure rate distribution
  - Vary the number of test cases

Results: Proportional vs. Random



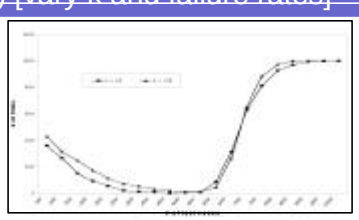
- 'U curve': realistic

Results: Proportional vs. Random (cont)

TABLE I
Random vs. Proportional Partition Testing ($k = 20$)

n	random vs. prop1	random vs. prop2	Random vs. prop3	$(P_p-P_r)/P_r$
20	475-0-525	475-0-525	475-0-525	0.005312262
40	426-0-574	404-0-596	396-0-604	0.002652833
60	337-0-663	299-0-701	306-0-694	0.001743906
80	280-0-720	255-0-745	251-0-749	0.000649685
100	218-0-782	179-0-821	179-0-821	0.000342222
120	166-0-834	145-0-855	141-0-859	0.000178710
140	138-0-862	104-0-896	93-0-907	0.000089087
160	124-0-876	72-0-928	76-0-924	0.000037538
180	78-0-922	46-0-954	45-0-955	0.000018094
200	60-0-940	31-0-969	30-0-970	0.000010344
220	36-0-964	19-0-981	21-0-979	0.000004140
240	35-0-965	14-0-986	14-0-986	0.000002511
260	27-0-973	8-0-992	7-0-993	0.000001129
280	24-0-976	4-0-996	5-0-995	0.000000646
300	14-0-986	3-0-997	2-0-998	0.000000027
320	17-0-983	4-0-996	1-0-999	0.000000014
340	9-0-991	1-0-999	1-0-999	0.000000007
360	11-0-989	2-0-998	0-0-1000	0.000000015
380	7-0-993	2-0-998	2-0-998	0.000000003
400	3-0-997	0-0-1000	0-0-1000	0.000000001
420	2-0-998	0-0-1000	0-0-1000	-
440	5-0-995	0-0-1000	0-0-1000	0.000000001
460	2-2-996	0-2-998	0-1-999	-
480	1-5-994	0-5-995	0-5-995	-

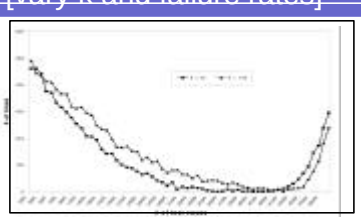
Results: Proportional vs. Random (cont) [vary k and failure rates]



failure rate: (0, 0.01]

•Similar with a minor delay

Results: Proportional vs. Random (cont) [vary k and failure rates]



failure rate: (0, 0.001]

•As failure rate becomes smaller, longer delay
•BUT same shape

Results: Proportional vs. Random

- Proportional Partition Testing is not more effective than Random
- Random Testing does 'as well as' Proportional Testing
- Proportional Testing not a worthwhile goal due to other factors

Results: Partition vs. Random

- Other factors
  - Cost
    - Hard to measure
    - Data not easy to obtain
    - E.g.
      - Cost of preparing and executing test cases
      - Training for the technique
      - Cost of failures left undetected
      - Testing Tools

Results: Partition vs. Random (cont)

- Other factors (cont)
  - Relative effectiveness
    - Selecting test cases in partition testing
      - Functional testing
      - Fault-based strategies
  - Homogeneity of faults

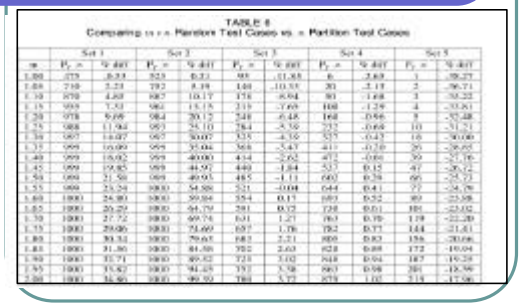
Results: Partition vs. Random (cont)

- For simulations
  - Translate cost and relative effectiveness into equivalent number of additional test cases
  - It takes *x* number of random test cases to be equivalent to the *n* number of partition testing
  - x = *m*n*

Results: Partition vs. Random (cont)

- Set 1
  - *k*=20 ,*n*=20, failure rate dist.: (0,0.1]
- Set 2
  - *k*=20 ,*n*=20, failure rate dist.: (0,0.001]
- Set 3
  - *k*=20 ,*n*=20, failure rate dist.: 95% in (0,0.1]
- Set 4
  - *k*=20 ,*n*=40, failure rate dist.: 95% in (0,0.1]
- Set 5
  - *k*=20 ,*n*=20, failure rate dist.: 95% in (0,0.1]

Results: Partition vs. Random (cont)



Results: Partition vs. Random (cont)

Moral of the Story:

- If Homogeneity, use Partition Testing
- Else, use Random Testing
- Homogeneity is not always true in practice

Conclusion

- Proportional Testing is not the 'way' to do testing
- Random Testing has advantage when no homogeneity and cost-effectiveness factors included
  - IF random is less effective and cheaper than partition
- "simulations":
  - More empirical studies necessary?