

Pairwise Testing

- Necessary condition
 - For each pair of input parameters, every combination of valid values of these two parameters be covered
- Example
 - Parameter A has values A1 and A2
 - Parameter B has values B1 and B2
 - Parameter C has values C1 and C2
- Discussion

Some Test Cases

- {(A1, B1, C1), (A1, B2, C2), (A2, B1, C3), (A2, B2, C1), (A2, B1, C2), (A1, B2, C3)}
- {(A1, B1, C1), (A1, B2, C1), (A2, B1, C2), (A2, B2, C3), (A2, B1, C1), (A1, B2, C2), (A1, B1, C3)}
- {(A1, B1, C1), (A1, B2, C1), (A2, B1, C2), (A2, B2, C2), (A2, B1, C1), (A1, B1, C2), (A1, B1, C3), (A2, B2, C3)}

Growth Terms

- Horizontal
 - Let T be a pairwise test set for parameters p_1, p_2, \dots, p_{n-1}
 - Horizontal growth of T for parameter p_i is to extend each test in T by adding the value of p_i .
- Vertical
 - After applying horizontal growth
 - Let T be a test set for p_1, p_2, \dots, p_i
 - Let π be the set of tests not covered by T
 - The vertical growth of T according to π is to construct new tests for pairs in π and add them to T

Horizontal Growth

Algorithm *IPO.H*(\mathcal{T}, p_i)
 // \mathcal{T} is a test set. But \mathcal{T} is also treated as a list with elements in arbitrary order
 { assume that the domain of p_i contains values v_1, v_2, \dots , and v_q ;
 $\pi = \{ \text{pairs between values of } p_i \text{ and values of } p_1, p_2, \dots, \text{ and } p_{i-1} \}$;
 if ($|\mathcal{T}| \leq q$)
 { for $1 \leq j \leq |\mathcal{T}|$, extend the j th test in \mathcal{T} by adding value v_j and
 remove from π pairs covered by the extended test;
 }
 else
 { for $1 \leq j \leq q$, extend the j th test in \mathcal{T} by adding value v_j and
 remove from π pairs covered by the extended test;
 for $q < j \leq |\mathcal{T}|$, extend the j th test in \mathcal{T} by adding one value of p_i
 such that the resulting test covers the most number of pairs in π , and
 remove from π pairs covered by the extended test;
 }

Vertical Growth

Algorithm *IPO.V*(\mathcal{T}, π)
 { let \mathcal{T}' be an empty set;
 for each pair in π
 { assume that the pair contains value w of p_k , $1 \leq k < i$, and value u of p_i ;
 if (\mathcal{T}' contains a test with “—” as the value of p_k and u as the value of p_i)
 modify this test by replacing the “—” with w ;
 else
 add a new test to \mathcal{T}' that has w as the value of p_k , u as the value of p_i ,
 and “—” as the value of every other parameter;
 };
 $\mathcal{T} = \mathcal{T} \cup \mathcal{T}'$;
 };
