**Generating System Test cases from UML Interaction Diagrams**

Introduction:

System testing is crucial to any project and a system is guaranteed to be successful if it passes every test case. Any project starts with analysis and then proceeds towards design in which the UML diagrams play a major role. The scope of this paper is to derive system test cases from UML Interaction/Sequence diagrams. This paper describes how to handle the time, branching and looping when generating the system test cases from the Interaction diagrams.

Background and Related Work:

There has been notable work done on generating test cases from sequence diagrams. This paper is an extension to a paper titled “Automatic Test Case Generation from UML Sequence Diagrams” [reference] in which the Sequence Diagram is first transformed into a graph called sequence diagram graph and test cases are derived from traversing this graph.

Approach:

Although the paper gives an approach to transform a sequence diagram to a graph and eventually tracing the graph to get the test cases, it does not propose handling of the following cases within the sequence diagram:

1. Negative Path – Delay & Loss
2. Branching/Alternate path
3. Looping

Our approach to each of the above is as following:

Negative Path: Time outs are introduced by adding a node between two nodes. This new node acts as condition check for time out.

Branching: Same as above

Looping: To Do.

Results: To Do.

References:

1. *“Automatic Test Case Generation from UML Sequence Diagrams”* Monalisa Sarma, Debasish Kundu, Rajib Mall.