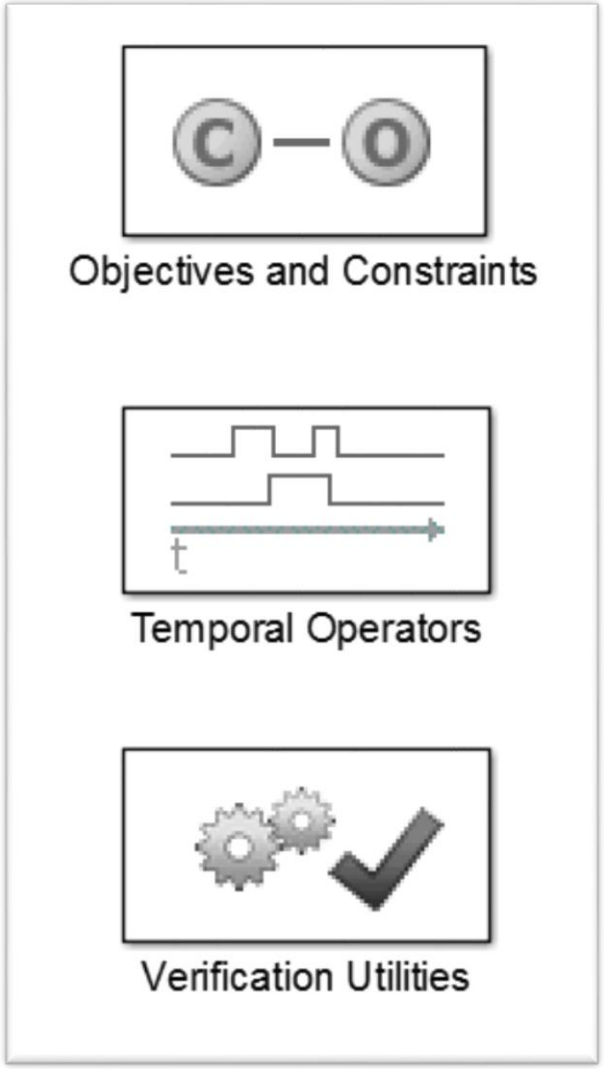


Exploring SLDV – 05

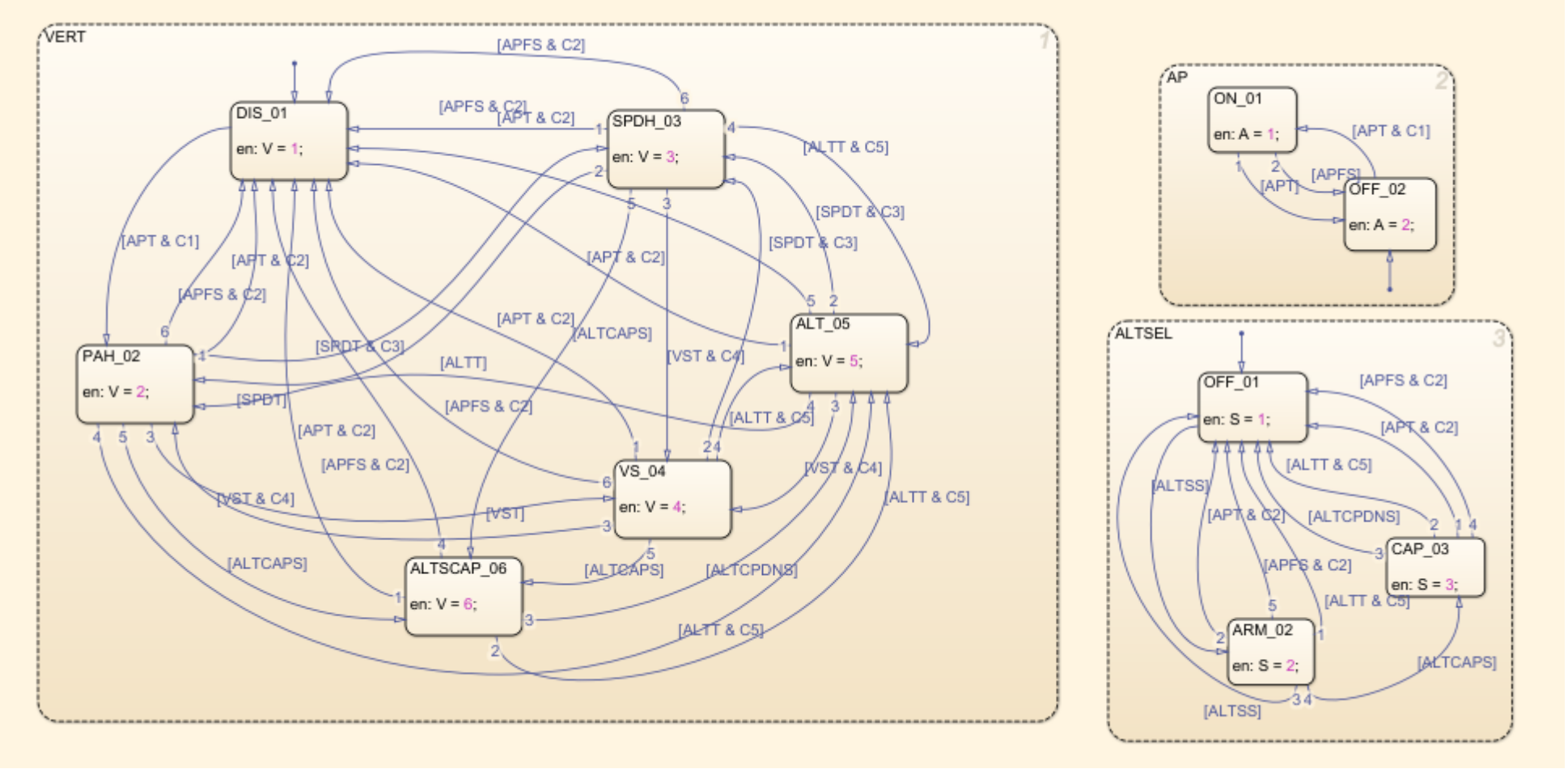
Natasha Y Jeppu

1. Stateflow mutant generation
2. Test case generation from assertions

Autopilot Example

- The Autopilot mode transition logic (see https://in.mathworks.com/matlabcentral/fileexchange/51567-exploring-simulink-design-verifier-2) has been converted into a Stateflow model.
- The three modes (Vertical, AP and ALTSEL) work independently and hence are represented as three separate state charts. The state blocks within the charts represent the states in every mode. The condition table defines the transitions between states in the Stateflow model.

Stateflow Autopilot Example



Mutating Stateflow

- Mutants are generated by modifying the transition conditions between states.
- A transition is chosen at random and altered. The model is saved as a mutant model and SLDV is used to check if the mutant is detected.
- Types of mutants:
  1. Change in variable
  2. Change in operation
  3. Negation of variable
  4. Change in execution order

Generating Mutants

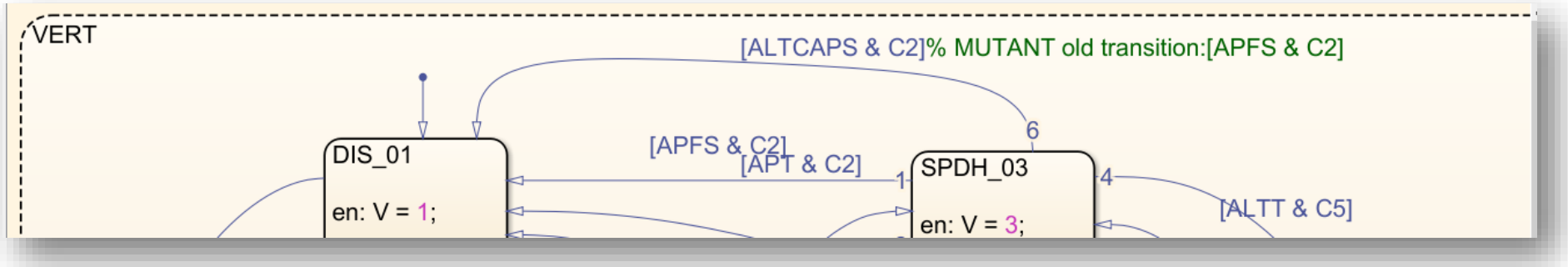
- The variables used in every transition condition is extracted and stored.
- For change of variable mutant, a transition is selected at random and one of the condition variables are replaced by another variable from the above set.
- Similarly, variables are replaced by their negations to generate type 3 mutants
- For change of operation mutant, relational (>,<), arithmetic (+,-,*,/) and logical (&,|) operators are identified in a given transition and replaced with another operator from the corresponding set.
- Execution order of transitions with execution order (EO) > 1 are replaced by value ‘x’ where $1 < x < EO$.

Mutate Script

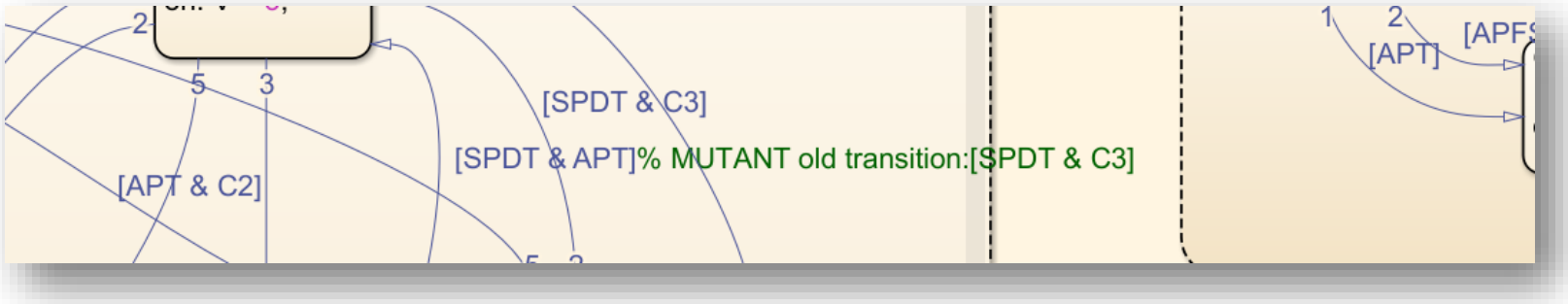
- The mutant generation MATLAB script produced around 650 odd unique mutants from the autopilot stateflow for testing.
- The script generates 1000 mutants of types 1,2 and 3 and all possible change in execution order type mutants and filters them to produce a set of unique non repeating mutants.
- Each mutant model is saved separately and a file containing details of type of mutant and the model name corresponding to that mutant is created for reference.

Mutant examples

- Model: mutant1.slx, Mutant: [ALTCAPS & C2]% MUTANT old transition:[APFS & C2]



- Model: mutant43.slx, Mutant: [SPDT & APT]% MUTANT old transition:[SPDT & C3]



Results

- All the mutants are killed by the SLDV. The maximum time taken by SLDV to capture the mutant is 59.1177 seconds and the minimum time is 21.2129 seconds
- The mean is 29.8047 with a standard deviation of 6.1007 seconds
- The batch script file run_batch.m runs all the mutants with the with sldv set to prove that no violations exist. In this mode SLDV proves that there is a violation of an assertion in the mutant file thus killing the mutant successfully.

Test case generation from assertions

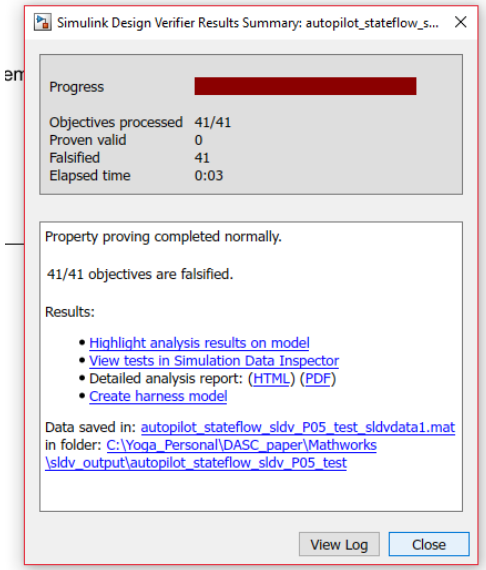
- Reference: https://in.mathworks.com/matlabcentral/fileexchange/54945-exploring-simulink-design-verifier-03
- The script make_mtl_code_M_gen.m was used to easily convert mode transitions defined as tables to Matlab code for use in SDV.
- Test generation has been added as an addition to the script. This is done by inserting sldv.prove statements with negation of the assertion that will generate a test case for the assertion by proving it “falsified” and generating a counter example.

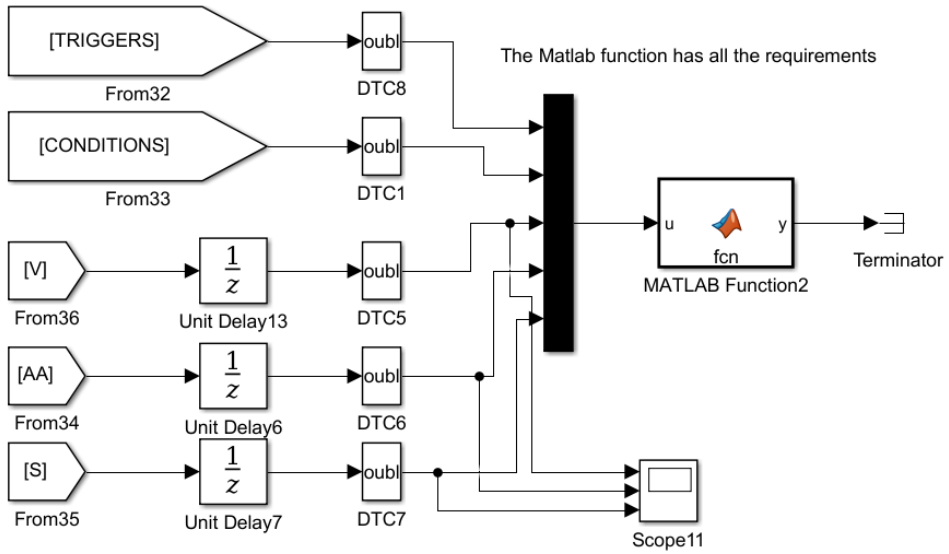
Example

- The file autopilot_stateflow_sldv_P05_test.mdl has the matlab function file for the assertions.

```
% 1) While in State DIS AND Trigger AP occurs THEN transit to PAH if condition C1 is True
if  (V == 1) && (TRIG == 2) && (C(1) == 1) ; V = 2;
sldv.prove(V ~= 2,' 1) While in State DIS AND Trigger AP occurs THEN transit to PAH if condition C1 is True');
```

41 test cases are created. Each test case is a requirements based test which is indicated in the html report. This is a good way to test any mode transition defined as tables.





HTML report

Summary

Model Item: MATLAB Function2

Property: 4) While in State PAH AND Trigger ALTCAP occurs THEN transit to ALTS_CAP if condition [No Condition required]
sldv.prove(V ~= 6,' 4) While in State PAH AND Trigger ALTCAP occurs THEN transit to ALTS_CAP if condition [No Condition required] ')

Status: Falsified

Counterexample

Time	**0**	**0.1**	**0.2**	**0.3**
Step	**1**	**2**	**3**	**4**
In3	1	0	0	-
In4	1	0	0	-
In5	1	0	0	-
APFAIL	0	0	0	0
AP	1	1	0	0
ALTCPDN	0	0	0	0
ALTCAP	0	0	0	1
ALT	0	0	0	0
ALTS	0	0	1	0
SPD	0	0	0	0
VS	0	0	0	0
Pitch	-10	-10	0	0
Speed	100	100	100	100
Altitude	100	100	100	100

This is a requirement that needs to be tested.

Thank you

- Natasha.Jeppu@gmail.com