

For: Mary Lou Sutherland
CSMISS IT Symposium



**Automated Statechart Model Checking
with Promela/SPIN**

Paula J. Pingree, Task Lead
Ed Benowitz, Developer
Autonomy and Control Section 345
Jet Propulsion Laboratory



Task Overview

- Extended capabilities for flight software verification by introducing formal method model checking
- Evaluated and implement software tools that will help to automate the process
- Applied method and tools to Fault Protection (FP) flight software (FSW) implemented in StateFlow® statecharts as a prototype
- *Infusing this verification technology in future projects*

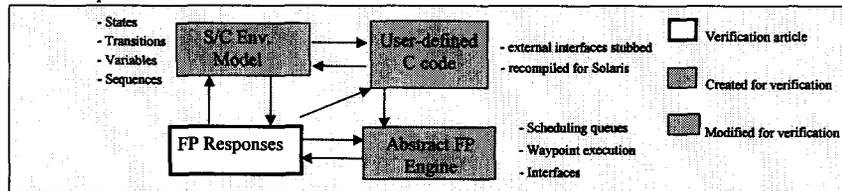
Team:

- Paula J. Pingree (JPL), Lead
Micah Clark, Eddie Benowitz
Software Engineering & Technology Infusion Group Autonomy & Control Section (345)
- Erich Mikk (Erlangen, Germany), Independent Consultant
Developer of Extended Hierarchical Automata (EHA)
- Gerard Holzmann, Margaret Smith, Dennis Dams (Bell Labs, Murray Hill, NJ), Co-Investigators
Computer Principles Research Department

2

The Task in General

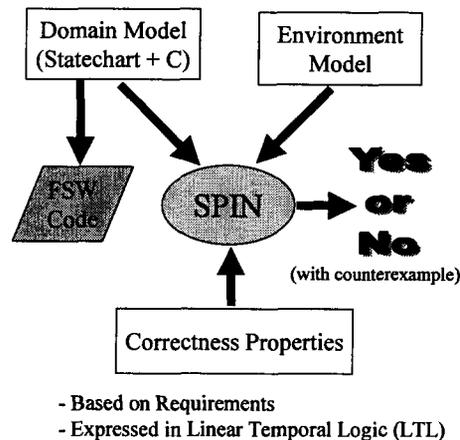
- Translation Programs
Getting the statecharts of interest into Promela for SPIN
- **The Environment - "Closing-the-loop"**
Creating a meaningful system for SPIN verification
- SPIN Model Checking
Evaluating the integrated system model against user-specified correctness properties
- Maintaining the Vision
Application to Flight Projects and Technology Infusion
Presentations to gain interest & support
Proposals to continue the work



3

Applying Model Checking to FSW

- We provide automated translation of the statechart model from Stateflow to Promela, the input modeling language of SPIN
- Key Benefits:
 - SPIN validation model and FSW code, now both auto-generated, have the same source (the Stateflow statechart)
 - Design validation can occur earlier in development cycle and without the use of valuable testbed resources



4



Products

1. Our Toolset: HiVy

- Translates Stateflow statecharts into Promela for SPIN
- Includes the following programs:
 - SfParse : implemented in Perl
 - sf2hsa : implemented in C
 - hsa2pr : implemented in C
 - HSA merge facility : implemented in C
- Produces syntactically and semantically correct Promela models
- Toolset is under CVS control and runs on UNIX workstations
- C programs developed with the VDM compiler

5



Products

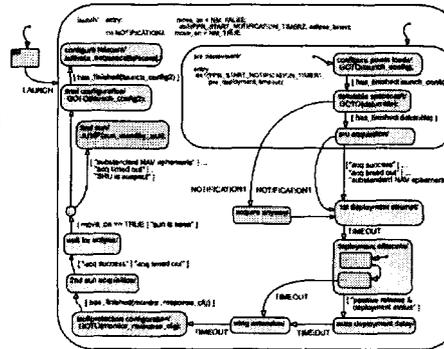
2. Documentation (see References)

- HiVy Toolset User's Guide v1.0 (Draft)
- HSA Format - to be submitted to the 2003 SPIN Workshop
- The Abstract State Machine, (a formal semantics definition for Stateflow)
- Conference paper for the 21st Digital Avionics Systems Conference titled "Validation of Mission Critical Software Design and Implementation using Model Checking"

6

The HiVy Toolset Work Package

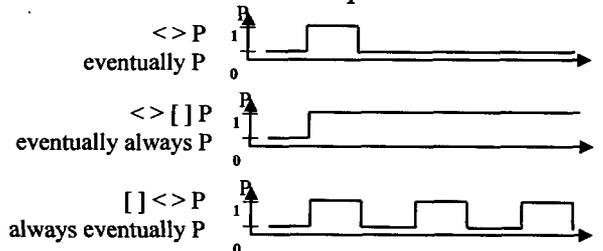
1. 'SfParse' recognizes all relevant elements from Stateflow *.mdl files
2. 'sf2hsa' transforms parsed elements into HSA for further processing
3. 'hsa2pr' program supports:
 - Sequential automata with states, transitions and default transitions
 - Transition labels with conditions and actions varying over boolean and integer variables
 - Hierarchy
 - AND-states
 - Event handling
 - Inter-level transitions
 - Junctions
 - Condition actions
 - User-defined C code
4. HSA Merge facility allows integration of multiple HSA files into a single Promela model upon translation



A section of the launch statechart, showing sun acquisition and pre-deployment of the DS1 solar array panels.

Correctness Properties

- Verification: Correctness Properties



- Correctness Properties (CP) are formal statements of the expected behavior of a system¹
- The accuracy of verification results depends on the accuracy and completeness of the CPs¹
- CP events and states must be linked to concrete events and states in the model¹

➤ **hsa2pr produces prop_list for generating CPs!**

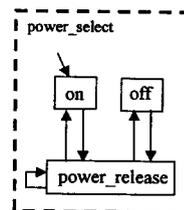
¹Credit to Margaret Smith, Bell Labs

Alternate Environment Model

- Can be time consuming to enter diagrams
- Specify a state machine within an Excel spreadsheet
 - States
 - Transitions
 - Hierarchy

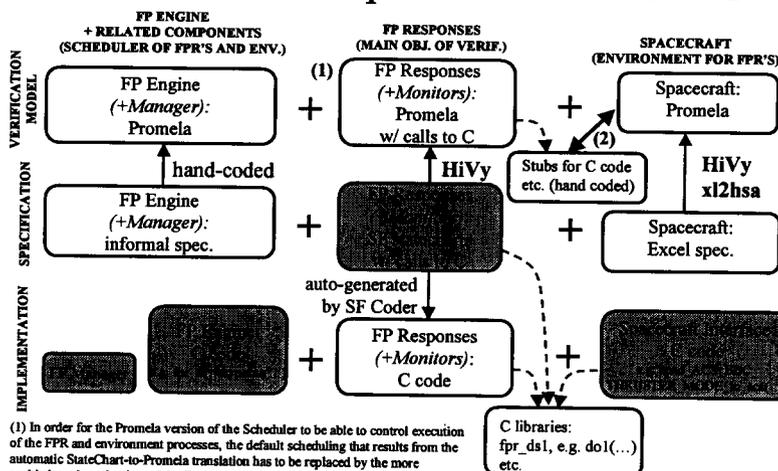
State Name	Parent State	Is Initial State	Has Parallel Substates
power_select	remote control frame	TRUE	
on	power_select	TRUE	
off	power_select		
power_release	power_select		

Transition Name	From State	To State
TRUE	on	power_release
TRUE	power_release	on
TRUE	off	power_release
TRUE	power_release	off
TRUE	power_release	power_release



9

The DS1 FP Response Verification - 1



(1) In order for the Promela version of the Scheduler to be able to control execution of the FPR and environment processes, the default scheduling that results from the automatic StateChart-to-Promela translation has to be replaced by the more sophisticated mechanism of the Engine. This is currently done manually.
 (2) The C code stubs in the libraries need to be able to manipulate the Promela variables in the S/C environment model.

10

The DS1 FP Response Verification - 2

- Current capabilities
 - Translation of FP Response statecharts to HSA via **Hivy**
 - Translation of S/C environment spec. to HSA via 'xl2hsa'
 - Merge of Response and S/C HSA into integrated Promela model via **Hivy**
 - Integration of user-defined C-code; mechanism defined, interface tested
- Work in progress
 - Integration with Deep Impact

11

What's Next

- FY03 R&TD Proposal funded
 - “Rapid Adoption of Model-Based Validation for Mission-Critical Flight Software Architectures & Domains”
 - Under the Advanced Software Technology and Methods Initiative (ASTMI)
 - Using HiVy translation to perform model checking on various domains
 - Deep Impact FP Responses
 - MDS Architecture example
 - MER Surface Ops Behavior example
- Continued support in FY03 from SQI
 - Toolset maintenance and improvement
 - Technology infusion for future projects

12



References

- P. Pingree, E. Mikk. HiVy User's Guide v1.0 (Draft), 2002
- E. Mikk. HSA Format, 2002
- E. Mikk, P. Pingree, M. Clark. The Abstract State Machine (Draft), 2002
- E. Benowitz. User Level Documentation for XL2HSA (Draft), 2002
- P. Pingree, E. Mikk, G. Holzmann, M. Smith, D. Dams, Validation of Mission Critical Software Design And Implementation Using Model Checking. *Accepted for the 21st Digital Avionics Systems Conference*, October 2002 (<http://eis/~ppingree/pubs.html>)
- E. Mikk, Semantics and Verification of Statecharts. PhD Thesis. *Technical Report of Christian-Albrechts-University in Kiel*, October 2000
- E. Mikk, Y. Lakhnech, M. Siegel and G. Holzmann, Implementing Statecharts in PROMELA/SPIN. In *Proceedings of the 2nd IEEE Workshop on Industrial-Strength Formal Specification Techniques*. pages 90-101. IEEE Computer Society 1999
- G.J. Holzmann, The model checker Spin, *IEEE Trans. on Software Eng.*, 5(23):279-295, 1997