SIG on Formal Methods

NEWS LETTER VOLUME 6 , JUNE 2015

SIG on Formal Methods:

Formal Methods (FM) has been in existence since 1940's, when Alan Turing proved the logical analysis of sequential programs using the properties of program states; and Floyd, Hoare and Naur used axiomatic techniques to prove program correctness against the specifications in 1960's.

These initial successes helped inculcate an interest in applying FM to the field of computer science.

Academia has been instrumental in bringing this field to the forefront, through continued research and development. The use of Formal Methods requires an expert skill-set expertise and therefore, its use is limited to those trained in the field.

Mission Statement:

Computer Society of India wants the field of Formal Methods to have a wider audience and more people to benefit from the application of these methods to all spheres of life. There is a need to use effective, correct and reliable approaches to design, develop and qualify complex, high assurance system software with the rigid schedules and budget. For this we need advanced tools, techniques and methods. Industry standards like RTCA DO-178C (Civil Aerospace), ISO 26262 (Automotive), IEC 61513(Nuclear), EN50126 (Railways) have recommended the usage of formal –method based approach to be used in the various phases of engineering process to achieve the required levels of safety and security.

Today there are proven techniques and tools that can be used in specification, design and verification & validation phases to assure correct requirement-capture, implementation, software functionality and security. This helps in developing high assurance software for applications such as cyber-physical systems, net-centric warfare systems, autonomous robots and Next Generation Air Transportation.

- **One day workshop on FM was conducted during April 2013**

Objectives of the Special Interest Group (SIG) are:

- To bring together scientists, academicians active in the field of formal methods and willing to exchange their experience in the industrial usage of formal methods
- To coordinate efforts in the transfer of formal methods technology and knowledge to industry
- To promote research and development for the improvement of formal methods and tools with respect to their usage in industry.
- To bring out practical engineering methods where formal methods will be integrated with current engineering methods

Some of the known applications of formal methods are:

- Formal verification, including theorem proving, model checking, and static analysis
- Techniques and algorithms for scaling formal methods
- Use of formal methods in automated software engineering and testing
- Model-based formal development
- Formal program synthesis
- Formal approaches to fault tolerance
- Use of formal methods in safety cases
- Use of formal methods in human-machine interaction analysis
- Use of formal methods in compiler validation and object code verification

3. Committee Members

1. Ms. Bhanumathi K S, Convener
2. Mr.Chander Mannar
3. Prof.Anirban basu
4. Mr. Suman Kumar
5. Prof. Shyam Sundar
6. Ms. Manju Nanda
7. Ms. J. Jayanthi
8. Ms. Saroja Devi
9. Prof. Meenakshi D'Souza
10. Dr. Yoganand Jeepu
11. Dr. Swatnalatha Rao
12. Dr. Aditya Kanade
13. Dr. A. Indira
14. Mr. Dhinakaran Pillai

Convener:

Bhanumathi K S
“Ganadhakshya” #406, 8 C Main,
H R B R First Block
Kalyan Nagar
Bangalore 560043
Email:bhanushekar@gmail.com
Mobile:+91 95350 92589

Formal Methods in Concurrent and Distributed Systems

Compiled by Bhanumathi KS

Since software systems are becoming increasingly more concurrent and distributed, modeling and analysis of interactions among their components is a crucial problem. In several application domains, message-based communication is used as the interaction mechanism, and the communication contract among the components of the system is specified semantically as a state machine.

The model-checking has been applied fairly successfully for verification of quite a few real-life systems, its applicability to a wider class of practical systems has been hampered by the state explosion problem (i.e. the enormous increase in the size of the state space).
It becomes important to develop various methods for ensuring the quality of concurrent software systems.

Drivers for Safety Related Standards

- RTCA DO-178B (USA Civil Avionics)
- Def Stan 00-55 (UK MoD)

- IEC 61508 (Generic "Programmable Systems")
  - ✓ IEC 601 (Medical Equipment)
  - ✓ (Pr)EN 50128 (Railway Industry)
- IEC 880 (Nuclear Power Control)
- MISRA (Automotive Industry)
- FDA (Medical Equipment)

Health Warning:

- There are no absolute guarantees.
- When applied correctly, formal methods have been demonstrated to result in systems of the highest integrity.

- Correctness is only guaranteed with respect to a specification — you need to validate the assumptions which under-pin the specification.
- Formal methods complement rather than replace conventional approaches, e.g. testing, simulation and prototyping.
- But Formal Methods are applied by humans who are error prone — so tools are crucial.

When Formal Method should be used?

1. Complex: Abstraction is an important technique for managing the complexity of large systems and is central to the notion of a formal method.
2. Concurrent: Distributed systems give rise to concurrency. While we find it hard to reason about concurrency, certain formal methods have been developed which ease this task.
3. Quality-critical:Applications where failure is not dangerous but economically expensive, e.g. financial applications and telecommunications.
4. Safety-critical: Applications where failure may endanger human life, e.g. fly-by-wire control systems and railway signalling systems.
5. Security-critical: Applications where failure means unauthorized access to sensitive information, e.g. medical records and security databases.
6. Standardized: where systems are designed to meet specific, internationally recognized, standards then it is important that the standards can be interpreted uniformly, e.g. language specifications and protocol standards.

What Do Formal Methods Cost?

1. The cost of applying formal methods is high, i.e. labour intensive coupled with a skills bottle-neck.
2. Need for support tools which are integrated within the conventional software development environments.
3. The potential for "re-use" within formal methods is high — At the 4th NASA Langley Formal Methods Workshop (1997), work by Rockwell Avionics Research on the formal verification of the AAMP family of microprocessors (designed for embedded real-time applications used on Boeing 737, 747, 757 & 767 aircraft) demonstrated a 6 fold speed up in the formal verification effort when the work under-taken on the AAMP-5 was reused with the AAMP-FV.

Which Formal Method is Best?

- Distributed concurrent systems:

  - ✓ Process algebras provide formalisms for modelling distributed current systems:

    - CCS: Communication and Concurrency.
    - CSP: Communicating Sequential Processes.
    - LOTOS: Language Of Temporal Ordering Specification

  - ✓ Description languages, less formal but greater industrial up-take:

    - SDL: Specification and Description Language.
    - Promela: PROcess MEta LAnguage.

Reference:
http://www.macs.hw.ac.uk/~air/dsp-spin/lectures/lec-1-introduction.pdf

****