Verification of Autonomy Software

Contact: Charles Pecheur (RIACS)
pecheur@email.arc.nasa.gov
with Tony Lindsey (QSS)
Stacy Nelson (NelsonConsult)
Reid Simmons (Carnegie Mellon)
Alessandro Cimatti (IRST, Italy)
Controlled vs. Autonomous

- **Controller**
- **Tester**

  - "Valve 1 stuck"
  - "Open valve 2"

  - Short time cycle (sec..min)
  - Human deals with unexpected
  - Open-loop, easy to test
  - Tractable state space, testing is appropriate

  ![Controller][1]
  ![Tester][2]

- **Controller**
- **Planner**
- **Exec**
- **MIR**

  - "Go to Saturn"
  - "Here we are"
  - Long time cycle (day..year)
  - Machine deals with unexpected
  - Closed-loop, hard to test
  - Huge state space, testing is insufficient

  ![Controller][3]
  ![Planner][4]
  ![Exec][5]
  ![MIR][6]
  ![Tester][7]
A model-based diagnosis system, uses a discrete, qualitative model to detect and diagnose faults.
Livingstone-to-SMV Translator

Diagnosis

- Livingstone
- Livingstone Model
- Livingstone Specification (enriched)
- Livingstone Trace

Verification

- SMV Model
- SMV Specification (CTL logic)
- SMV Trace

- Allows exhaustive analysis of Livingstone models ($10^{50+}$ states)
- Uses SMV: symbolic model checker (BDD and SAT)
- Enriched spec syntax (vs. SMV's core temporal logic)
- Hide away SMV, offer a model checker for Livingstone
- Graphical interface, trace display
Livingstone PathFinder (LPF)

- Execute the Real Program in a simulated environment (testbed)
- Instrument the Code to be able to backtrack between alternate paths
- Modular architecture, allows different diagnosis, simulators, search algorithms
  - e.g. depth-first / breadth-first / random / guided / interactive / ...
**Verification of Diagnosability**

**Q:** From observations (input/output), can diagnosis always tell when plant comes to a **bad** state?

**A:** **YES unless** plant can go **good** or **bad** with the same observations (and therefore diagnosis cannot tell)

---

**Verification using model checking (SMV)**

- Two "siamese twin" copies of the plant (L/R), with coupled observations
- verify that one cannot reach: 
  
  (L in **good**) and (R in **bad**)
Verification of IVHM* for Next-Gen Space Vehicle

*IVHM = Integrated Vehicle Health Management
= Integrated prognosis/diagnosis

IVHM framework developed by Northrop Grumman Corp.
- Adopted Model-Based Diagnosis, including Livingstone Technology infusion project:
  - Survey of NASA current V&V practice, applicable formal methods, our verification tools
  See [ase.arc.nasa.gov/vvivhm](http://ase.arc.nasa.gov/vvivhm)
  - Maturation of Livingstone verification tools (translator and LPF): tool extensions, GUI, improved documentation and packaging, integration with other IVHM tools
Symbolic Model Checking

**Model Checking** = verification by exhaustive exploration
  + Full coverage (incl. non-determinism)
    - Limited by state space explosion
• **Symbolic** Model Checking =
  Processes *sets of states*,
  Represented as *boolean formulas*,
  Encoded as *binary decision diagrams* (BDDs).
• Can handle larger state spaces ($10^{50}$ and up)
  - but BDD size can explode too
• Works very well for Livingstone models
• Most widely used: SMV (Carnegie Mellon / Cadence / IRST)
• Variant: **Bounded Model Checking** using **SAT** solvers
To Probe Further

On-Line

• Livingstone to SMV Translator:  
  ase.arc.nasa.gov/mpl2smv
• Livingstone PathFinder:  
  ase.arc.nasa.gov/lpf
• Verification of IVHM:  
  ase.arc.nasa.gov/vvivhm

Publications

• Stacy Nelson, Charles Pecheur. Formal Verification of a Next-Generation Space Shuttle. FAABS II, Greenbelt, MD, October 2002. To be published in LNCS.
• Charles Pecheur, Reid Simmons. From Livingstone to SMV: Formal Verification for Autonomous Spacecrafts. FAABS I, April 2000. LNCS 1871, Springer Verlag.

Reports


Publications and Reports available on-line at:
http://ase.arc.nasa.gov/pecheur/publi.html