Assessing the Effect of Failure Severity, Coincident Failures and Usage-Profiles on the Reliability of Embedded Control Systems

Symposium on Applied Computing

Nicosia Cyprus
March 16, 2004

Frederick T. Sheldon, Ph.D.
Oak Ridge National Laboratory

Kshamta Jerath
Microsoft Corporation
Agenda

I  Problem Definition and Motivation
II Example Embedded System – The Anti-lock Braking System
III Modeling Strategy, SPN Models and SAN Models
IV Reliability Analysis Results and Discussion
V Conclusion and Scope of Future Work
Goals

- **Model** and analyze the Anti-lock Braking System (ABS) of a passenger vehicle.
- Model **severity of failures**, **coincident failures** and **usage-profiles**.
- Carry out the **reliability** analysis using different stochastic formalisms – Stochastic Petri Nets (SPNs) and Stochastic Activity Networks (SANs).
- Develop an approach that is generic and extensible for this application domain.
Introduction (1)

- **Model**: An abstraction of a system that includes sufficient detail to facilitate an understanding of system behavior.
- **Reliability**: Probability that a system will deliver intended functionality/quality for a specified period of time, given that the system was functioning properly at the start of this period.
- **Failure**: An observed departure of the external result of operation from requirements or user expectations.
Introduction (2)

- **Severity of failure**: The impact the failure has on the operation of the system. An example of a service impact classification is critical, major and minor.

- **Coincident failures**: All failures are not independent. Components generally interact with each other during operation and affect the probability of failure of other components.

- **Usage-Profiles**: Quantitative characterization of how a system (hardware and software) is used. (a.k.a. operational profiles, workload)
Motivation

- Reliability analysis of an ABS model to predict/estimate the likelihood and characteristic properties of failures occurring in the system.
  - Reliability function & Mean Time To Failure (MTTF).
  - The need for a realistic, scalable & extensible model
    - Important to model severity and coincident failures
    - Important to model usage-profiles
  - Comparing results from two stochastic formalisms – SPNs and SANs
    - Validation by comparison against actual data beyond the scope of this research.
Part II

- Problem Definition and Motivation
- Example Embedded System – The Anti-lock Braking System
- Modeling Strategy, SPN Models and SAN Models
- Reliability Analysis Results and Discussion
- Conclusion and Scope of Future Work
Anti-lock Braking System (1)

- An integrated part of the braking system of vehicle.
  - Prevents wheel lock up during emergency stop by modulating wheel pressure.
  - Permits the driver to maintain steering control while braking.

- Main Components
  - Wheel speed sensors.
  - Electronic control unit (controller).
  - Hydraulic control unit (hydraulic pump).
  - Valves.
Anti-lock Braking System (2)

Functioning

- Wheel speed sensors measure wheel-speed.
- The electronic control unit (ECU) “reads” signals from the wheel speed sensors.
- If a wheel’s rotation suddenly decreases, the ECU orders the hydraulic control unit (HCU) to reduce the line pressure to that wheel’s brake.
- The HCU reduces the pressure in that brake line by controlling the valves present there.
- Once the wheel resumes normal operation, the control restores pressure to that wheel’s brake.
Top Level Schematic of ABS

Top level schematic showing sensors, processing and actuators

- Disc break (4 indpt)
- Wheel speed sensor (4 indpt)
- B1-4 = Brakes (LF, RF, LR, RR)
- S1-4 = Speed sensors (LF, RF, LR, RR)
- R1-2 Turning angles (of the vehicle and the tires respectively)

- Anti-lock Breaking / Anti-skid Controller
- Brake Pressure
- Electronic brake control module (EBCM)
- Hydraulic modulator valve assembly
- Master break cylinder
- Rear

- LF B1 S1 B2 RF
- LR B3 S3 S4 RR

- Accelerometer
- 90

- 0

- 2 4

- 2 4
Detailed Schematic

Legend:
BPP  Back Pressure Pump
DC   Damping Chamber (w/ mixing)
DPS  Down Pressure Storage
DV   Drain Valve
EM   Electric Motor
FL/R Front Left/Right
FR   Floating Rotor
FV   Filling Valve
HC   Hydraulic Control
HPC  Hydraulic pressure circuit
IV   Intake Valve
L/R  Left/Right
PD   Pulsing Damper
PRV  Pressure Release Valve
PS   Pressure Sensor
RL/R Rear Left/Right
SV   Switching Valve

Tank
HPC FR
Main Brake Cylinder
Brake Actuation System

Hydraulic Control Unit
Front (HC) Subunit

Front Left
Front Right

BPP DC
PD PRV (F)
DV FV PS

Front Disc Brakes

BPP DC
DV FV PS

Rear Disc Brakes

Rear Left
Rear Right

BPP DC
DV FV PS

Rear Left
Rear Right

Legend:
IV (F) SV (F)

IV (R) SV (R)
ABS Assumptions

- Modes of operation (different levels of degraded performance $\rightarrow$ failure severity)
  - Normal operation
  - Degraded mode
  - Lost stability mode

- Lifetime of a vehicle: 300-600 hours/year for an average of 10-15 years (i.e. 3000-9000 hours)

- Four-channel four-sensor ABS scheme
## Failure Rates of Components

<table>
<thead>
<tr>
<th>Component</th>
<th>#</th>
<th>Base Failure Rate</th>
<th>Probability</th>
<th></th>
<th></th>
<th>Loss of Vehicle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Degraded Operation</td>
<td>Loss of Stability</td>
<td>Loss of Vehicle</td>
<td></td>
</tr>
<tr>
<td>Wheel Speed Sensor</td>
<td>4</td>
<td>2.00E-11</td>
<td>0.38</td>
<td>0.62</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Pressure Sensor</td>
<td>4</td>
<td>1.50E-11</td>
<td>0.64</td>
<td>0.36</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Main Brake Cylinder</td>
<td>1</td>
<td>1.00E-11</td>
<td>-</td>
<td>-</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td>Pressure Limiting Valve</td>
<td>2</td>
<td>6.00E-13</td>
<td>-</td>
<td>0.22</td>
<td>0.78</td>
<td></td>
</tr>
<tr>
<td>Inlet Valve</td>
<td>4</td>
<td>6.00E-13</td>
<td>-</td>
<td>0.18</td>
<td>0.82</td>
<td></td>
</tr>
<tr>
<td>Drain Valve</td>
<td>4</td>
<td>6.00E-13</td>
<td>-</td>
<td>0.19</td>
<td>0.81</td>
<td></td>
</tr>
<tr>
<td>Toggle Switching Valve</td>
<td>2</td>
<td>6.00E-13</td>
<td>1.0</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Hydraulic Pump</td>
<td>2</td>
<td>6.80E-11</td>
<td>-</td>
<td>-</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td>Pressure Tank</td>
<td>2</td>
<td>2.00E-12</td>
<td>-</td>
<td>-</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td>Controller</td>
<td>1</td>
<td>6.00E-12</td>
<td>0.2</td>
<td>0.4</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>Tubing</td>
<td>1</td>
<td>3.00E-12</td>
<td>0.33</td>
<td>-</td>
<td>0.67</td>
<td></td>
</tr>
<tr>
<td>Piping</td>
<td>1</td>
<td>4.00E-12</td>
<td>0.33</td>
<td>-</td>
<td>0.67</td>
<td></td>
</tr>
</tbody>
</table>

† Obtained from DaimlerChrysler. The data has been falsified for publishing as part of this research.
Part III

- Problem Definition and Motivation
- Example Embedded System – The Anti-lock Braking System
- Modeling Strategy, SPN Models and SAN Models
- Reliability Analysis Results and Discussion
- Conclusion and Scope of Future Work
Stochastic Modeling

Mathematical (numerical solution) method

- Defined over a given probability space and indexed by the parameter $t$ (time).

- Markov Processes
  - Memory-less property: Future development depends only on the current state and not how the process arrived in that state.
  - Markov Reward Models (MRM): Associate reward rates with state occupancies in Markov processes.
  - Common solution method for performability.
Challenges in Modeling

Practical Issues

- Obtaining reliability data
- Limited ability of capturing interactions b/w components
- Need to estimate fault correlation b/w components
- Incorporating usage information
- Direct validation of results

Problems in stochastic modeling

- Large state space: Size of the Markov model grows exponentially with no. of components in the model.
- Stiffness: Due to the different orders of magnitude of failure rates.
Stochastic Petri Nets (SPNs)

- Graphical and mathematical tool for describing and studying concurrent, asynchronous, distributed, parallel, non-deterministic and/or stochastic systems.
- Concise description of the system, which can be automatically converted to underlying Markov chains.
- Bipartite directed graph whose nodes are divided into two disjoint sets: *places* and *transitions*. 
**Stochastic Petri Net Symbols**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Circle" /></td>
<td><strong>Places</strong> (drawn as circles) represent conditions.</td>
</tr>
<tr>
<td><img src="image" alt="Bar" /></td>
<td><strong>Transitions</strong> (drawn as bars) represent events. <em>Timed</em> transitions and <em>Immediate</em> transitions.</td>
</tr>
<tr>
<td><img src="image" alt="Arrow" /></td>
<td><strong>Arcs</strong> (drawn as arrows) signify which combination of events must hold before/after an event. <em>Input</em> arcs and <em>Output</em> arcs.</td>
</tr>
<tr>
<td><img src="image" alt="Circle with Head" /></td>
<td><strong>Inhibitor arcs</strong> (drawn as circle-headed arcs) test for zero marking condition.</td>
</tr>
<tr>
<td><img src="image" alt="Small Circle" /></td>
<td><strong>Tokens</strong> (drawn as small filled circles) denote the conditions holding at any given time.</td>
</tr>
</tbody>
</table>
Stochastic Petri Net Package

- Stochastic Petri Net Package (SPNP) allows specification of Stochastic Reward Nets (SRNs) and the computation of steady-state, transient, cumulative, time-averaged measures.
- SRNs are specified using CSPL (C-based Stochastic Petri net Language).
- Sparse Matrix techniques are used to solve the underlying Markov Reward Model (MRM).
- Version 6
SPN Models Representing Severity and Coincident Failures (1)

- **Assumptions**
  - Exponential Failure Rates to allow Markov chain analysis
  - Levels of failure severity: degraded mode, loss of stability (LOS) and loss of vehicle (LOV)
  - Impact of failure on failure rates:
    - Degraded – two orders of magnitude
    - LOS – four orders of magnitude
  - Limited number of inter-dependencies modeled

**Diagram**

- Inter-dependencies b/w components

**Components**

- Tubing
- Controller
- Pressure Tank
- Toggle Switch
- Hydraulic Pump
- Drain Valve
- Inlet Valve

**Modeling**

- Modeled
- Not modeled
SPN Models Representing Severity and Coincident Failures (2)

- All ABS components represented in the global model.
- Components grouped according to their cardinality.
- Next slide shows controller detail…
Start → Braking

Braking → Central Operation (central_op)
Braking → Axle Operation (axle_op)

Central Operation (central_op) → Central
Central Operation (central_op) → Axle

Axle Operation (axle_op) → Central Operation (central_op)
Axle Operation (axle_op) → Axle

Central → Mbrakecyl
Central → Controller
Central → Tubing
Central → Piping
Central → AxleCentral

Axle → FRWheel
Axle → RLWheel
Axle → RRWheel

Central_op → Degraded Operation
Central_op → Loss of Stability
Central_op → Loss of Vehicle

Dashed Box:
- Mbrakecyl
- Controller
- Tubing
- Piping
- AxleCentral
- FLWheel
- FRWheel
- RLWheel
- RRWheel

- Degraded Operation
- Loss of Stability
- Loss of Vehicle
SPN Models Representing Severity and Coincident Failures (3)

- Every component either functions “normally” as shown by controllerOp or “fails” as shown by controllerFail.
- Failed component may cause degraded-operation, loss-of-stability or loss-of-vehicle.
- Degraded-operation/ loss-of-stability: component continues to operate with increased failure rate (by 2 and 4 orders of magnitude respectively).

Model of an ABS component w/ coincident failures
SPN Models Representing Severity and Coincident Failures (4)

- Each failure transition has a variable rate determined by a corresponding function.
- Failure of component B affects failure rate of component A by including the condition:

```c
if failedB then
  failureA = failureA * order
```

where order is 100 in case of degraded operation and 10000 in case of loss of stability.

```c
double controllerRate()
{
  double controller_rate = 0.0000006;

  if (mark("controllerLOS") > 0)
    return controller_rate * 10000;

  if ((mark("controllerDegraded") > 0) ||
      mark("tubingDegraded") > 0))
    return controller_rate * 100;

  return controller_rate;
}
```

Variable Rate to Model Coincident Failures
SPN Models Representing Usage-Profiles (1)

- User’s interact with the system in an intermittent fashion, resulting in operational workload profiles that alternate between periods of “active” and “passive” use.

- Assumptions
  - Exponential Failure Rates to allow Markov chain analysis.
  - Infinite repair rate → all repairs occur instantaneously.
  - Exponentially distributed workload.
  - Two usage-profiles: Low usage and High usage which are two orders of magnitude different.
When a component fails, check if it was in “active” use or not.

The parameter $1/\mu$ indicates the mean duration of active use while the parameter $1/\alpha$ indicates the mean duration of passive use.

Failure of component in “active” mode only affects reliability.
SPN Models Representing Usage-Profiles (3)

- State explosion problem due to increased number of states.
- Work-around: The model was simplified to incorporate the usage parameters while calculating the failure rate itself for each component.
- The value of $mu$ was assumed to be 2.5 for infrequent use periods and 250 for frequent use periods.

```c
double controllerRate()
{
    double controller_rate = 0.0000006;

    // usage parameter
    controller_rate += controller_rate * mu;

    if (mark("controllerLOS") > 0)
        return controller_rate * 10000;

    if ((mark("controllerDegraded") > 0) ||
        (mark("tubingDegraded") > 0))
        return controller_rate * 100;

    return controller_rate;
}
```

Variable Rate to Model usage-profiles
SPN Reliability Measure

- Reliability measure expressed in terms of expected values of reward rate functions.
- The `reliab()` function defines a single set of 0/1 rewards.
- Used as an input argument to the `void pr_expected(char* string, double (*func)())` function provided by SPNP that computes the expected value of the measure returned by `func`.

```c
#include <stdio.h>

double reliab()
{
    double reward;
    if((mark("loss_of_vehicle") >= 1) ||
        (mark("loss_of_stability") >= 3) ||
        (mark("degraded_operation") >= 5))
        reward = 0;
    else
        reward = 1;
    return reward;
}
```

Function to calculate reliability reward
**SPN Halting Condition**

*When this function evaluates to zero, the marking is considered to be absorbing.*

```c
int halt()
{
    if((mark("loss_of_vehicle") >= 1) ||
       (mark("loss_of_stability") >= 3) ||
       (mark("degraded_operation") >= 5))
        return 0;
    else
        return 1;
}
```

- Necessary to explicitly impose a halting condition because the developed SPN models recycle tokens.

- The system is assumed to fail when
  - > 5 components function in a degraded mode, or
  - > 3 components cause loss of stability, or
  - the failure of an important component causes loss of vehicle.
Stochastic Activity Networks (SANs)

A generalization of SPNs, permit the representation of concurrency, fault tolerance, and degradable performance in a single model.

- Use *graphical primitives*, are *more compact* and provide *greater insight* into the behavior of the network.

- Permit both the representation of complex interactions among concurrent activities (as can be represented in SPNs) and non-determinism in actions taken at the completion of some activity.
### Stochastic Activity Network Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="triangle-flat.png" alt="Places Icon" /></td>
<td><strong>Places</strong> (drawn as circles) represent the state of the modeled system.</td>
</tr>
<tr>
<td><img src="oval.png" alt="Activities Icon" /></td>
<td><strong>Activities</strong> (drawn as ovals) represent events. <strong>Timed</strong> and <strong>Instantaneous</strong> activities. <strong>Case probabilities</strong> (as circles on right of activity).</td>
</tr>
<tr>
<td><img src="triangle-point.png" alt="Input Gates Icon" /></td>
<td><strong>Input Gates</strong> (triangles with point connected to activity) control the enabling of activities.</td>
</tr>
<tr>
<td><img src="triangle-flat.png" alt="Output Gates Icon" /></td>
<td><strong>Output Gates</strong> (triangles with flat side connected to activity) define the marking changes that occur when activity completes.</td>
</tr>
</tbody>
</table>
UltraSAN

- An X-windows based software tool for evaluating systems represented as SANs.
- Three main tools: SAN editor, composed model editor, performance model editor.
- Analytical solvers as well as simulators available.
- Steady-state and transient solutions are possible.
- Reduced base model construction used to overcome largeness of state-space problem.
- Version 3.5
SAN Models Representing Severity and Coincident Failures (1)

- Assumptions
  - Exponential Failure Rates to allow Markov chain analysis
  - Levels of failure severity: degraded mode, loss of stability (LOS) and loss of vehicle (LOV)
  - Impact of failure on failure rates:
    - Degraded – two orders of magnitude
    - LOS – four orders of magnitude
  - Limited number of inter-dependencies modeled
SAN Models Representing Severity and Coincident Failures (2)

- Three individual SAN sub-models: Central_1, Central_2 and Wheel (replicated four times).
- The division into three sub-categories done to facilitate representation of coincident failures.
- Avoid replication of sub-nets where unnecessary.
SAN Models Representing Severity and Coincident Failures (3)

- All subnets share common places: degraded, LOS, LOV and halted.
- Presence of tokens in degraded, LOS, and LOV places indicates degraded operation, loss of stability and loss of vehicle resp.
- Output cases of an activity have different probabilities to model conflict between the outcome of failure.
SAN Models Representing Severity and Coincident Failures (4)

- Degraded-operation/ loss-of-stability: failure rate increases (by 2 and 4 orders of magnitude respectively).
- Failure of component A to degraded mode causes the failure rate of component B to increase by 2 orders.
- Failure of component A to a loss of stability mode causes the failure rate of component B to increase by 4 orders.

<table>
<thead>
<tr>
<th>Activity</th>
<th>Rate</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MARK(controllerLOS) !=0? controllerRate*10000:</td>
<td>Case1</td>
</tr>
<tr>
<td></td>
<td>(MARK(controllerDegraded) !=0</td>
<td></td>
</tr>
<tr>
<td>controllerFail</td>
<td>MARK(controllerLOS) !=0?</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hydraulicPumpRate*10000:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(MARK(controllerDegraded) !=0 &amp;&amp; MARK(tubingDegraded) !=0 ?hydraulicPumpRate*100 :hydraulicPumpRate)</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Activity Rates Model Severity and Coincident Failures
<table>
<thead>
<tr>
<th>Activity</th>
<th>Rate</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MARK(controllerLOS) !=0? controllerRate*10000:</td>
<td>Case1</td>
</tr>
<tr>
<td>controllerFail</td>
<td>(MARK(controllerDegraded) !=0</td>
<td></td>
</tr>
<tr>
<td>hydraulicPump Fail</td>
<td>MARK(controllerLOS) !=0? hydraulicPumpRate*10000:</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>(MARK(controllerDegraded) !=0 ?hydraulicPumpRate*100 :hydraulicPumpRate)</td>
<td></td>
</tr>
</tbody>
</table>

**Activity Rates Model Severity and Coincident Failures**
SAN Models Representing Usage-Profiles (1)

Assumptions

- Exponential Failure Rates to allow Markov chain analysis.
- Infinite repair rate: all repairs occur instantaneously.
- Exponentially distributed workload.
- Two usage-profiles: Low usage and High usage which are one order of magnitude different.
SAN Models Representing Usage-Profiles (2)

- When a component fails, check if it was in “active” use or not.
- Failure of component in “active” mode only affects reliability.
- Work around the state explosion problem by incorporating the usage parameters while calculating the failure rate of component (lambda+mu).
- mu same for all components

State Diagram for reliability evaluation

mu-active use rate
alpha-passive use rate
lambda-failure rate
v-repair rate
SAN Reliability Measure

- Reward rates specified using a predicate and function.
- If the system is not in an absorbing state (system failed), reliability is a function of the number of tokens in degraded, LOS and LOV.
- For normal operation, the function evaluates to 1. Reliability is 0 when the predicate evaluates to false, by default.

**Predicate:**
MARK(halted)==0

**Function:**
1.0/(1+MARK(degraded)+MARK(LOS) +MARK(LOV))

**Reward Rate to Calculate Reliability**
SAN Halting Condition

- Input condition on each activity states that it is enabled only if there is no token in \textit{halted} place (common to all subnets).

- Presence of token in \textit{halted} place indicates an absorbing state.

SAN Halting Condition Depicted
Part IV

- Problem Definition and Motivation
- Example Embedded System – The Anti-lock Braking System
- Modeling Strategy, SPN Models and SAN Models
- Reliability Analysis Results and Discussion
- Conclusion and Scope of Future Work
SPN Reliability Analysis Results

- Transient Analysis carried out using SPNP (Stochastic Petri Net Package) version 6 on a Sun Ultra 10 (400 MHz) with 500 MB memory.

- 164,209 tangible markings of which 91,880 were absorbing.

- Approximate running time of the solver was 144-168 hours.
SPN Reliability Analysis Results for Coincident Failures and Severity (1)

- The Y-axis gives the measure of interest i.e. reliability, the time range (0 to 50K hrs) is along X-axis.

- MTTF for the model with coincident failures (784,856.4 hrs) is approximately 421 hours less than the model without coincident failures (785,277.6 hrs).
Reliability of ABS

MTTF (w/o) = 785277.6 hrs.
MTTF (with) = 784856.4 hrs.

SPN Reliability Analysis Results for Coincident Failures and Severity
SPN Reliability Analysis Results for Coincident Failures and Severity (2)

- Graph shows the difference between the reliability functions.
- Start diverging around 350 hours of operation.
- The difference in reliability between the two cases becomes marked (after 13K hours) only beyond the average lifetime of the vehicle (3K-9K hours).
Difference in Reliability Functions (With and without coincident failures)
SPN Reliability Analysis Results for Usage Profiles

- MTTF for the high usage case is 771,022.9 hrs as opposed to 775,111.7 hrs for the low usage case, a difference of ~ 4089 hrs.

- Reliability of the system with heavy usage decreases *alarmingly* (!) within the first 1K hrs, while the reliability of the system with low usage decreases *perceptibly* (!!) only after 2.5K hours of operation and then steadily thereafter.
Reliability Analysis with Usage Profiles

MTTF (Low Usage) = 775111.7 hrs.
MTTF (High Usage) = 771022.9 hrs.

SPN Reliability Analysis Results for Usage Profiles
SAN Reliability Analysis Results

- Transient Analysis carried out using UltraSAN version 3.5 on a Sun Ultra 10 (400 MHz) with 500 MB memory.
- 859,958 states generated.
- Approximate running time of the solver (transient solver trs) was 120-144 hours.
The reliability functions diverge perceptibly after around 1K hours of operation, difference continues to increase with time.

After 5K hours the difference is 0.025, after 10K hours it is 0.049.

Time to failure for model with coincident failures is 25,409 hours, for model without coincident failures is 29,167 hours (diff. of 3,758 hours).
Reliability of the ABS

SAN Reliability Analysis Results for Coincident Failures and Severity
SAN Reliability Analysis Results for Usage Profiles

- Reliability of the system with heavy usage starts decreasing *alarmingly* after 100 hrs, while the reliability of the system with low usage decreases only *perceptibly* after 100 hours of operation.
- At the extreme end of average lifetime (9K hours) of the vehicle, reliability has dropped to almost 0 for heavy usage and to ~ 0.4 for low usage.
- Time to failure for model with low usage is 12,262 hrs, for model with high usage is 1,687 hrs (diff. of 10,575 hrs).
Reliability of ABS with Usage-Profiles

SAN Reliability Analysis Results for Usage-Profiles
Comparing the SPN & SAN Results (1)

- Because it is beyond the scope of this research to validate the results from the analytic experiments against *real data*, . . .
  - we compare the results from SPN & SAN analyses.

- The difference in the range of actual reliability values between the SPN and SAN models may be attributed to the different ways in which the reliability reward is defined.
  - See the plots where both curves are in the same graph

- **Severity and Coincident Failures**
  - SPNs - The curves for the two cases completely overlapped.
  - SANs - The curves diverge after 1K hours of operation.
Comparison of SPN and SAN Reliability Results for Models Representing Severity and Coincident Failures
Comparison of SPN and SAN Reliability Results for Models Representing Usage-Profiles (with failure severity and coincident failures)
Comparing the SPN & SAN Results (2)

- Usage Profiles
  - SPNs – Reliability for high usage decreases alarmingly within first 1K hrs, for low usage only after 2.5K hrs.
  - SANs - Reliability for high usage decreases alarmingly after 100 hrs, for low usage only perceptibly after 100 hours.

- Results from both models agree on the fact that failure severity, coincident failures and usage-profiles contribute significantly to predicting system reliability.

- Which of these results is more realistic?

- Comparing results does not make up for validation against real data.
### Comparing the SPN & SAN Results (3)

<table>
<thead>
<tr>
<th>Criteria</th>
<th>SPN Models</th>
<th>SAN Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assumptions</td>
<td>Same</td>
<td>Same</td>
</tr>
<tr>
<td>Reliability measure</td>
<td>Different</td>
<td>Different</td>
</tr>
<tr>
<td>Number of states</td>
<td>164,209</td>
<td>859,958</td>
</tr>
<tr>
<td>Solvers’ Running time</td>
<td>144-168 hours</td>
<td>120-144 hours</td>
</tr>
<tr>
<td>Reliability at 9Khours (severity &amp; co.failures)</td>
<td>9.5792578e-01 vs. 9.5792653e-01</td>
<td>7.3672e-01 vs. 7.8600e-01</td>
</tr>
<tr>
<td>Reliability at 9Khours (usage-profiles)</td>
<td>8.9621556e-01 vs. 7.6658329e-01</td>
<td>4.455167e-01 vs. 3.130521e-03</td>
</tr>
</tbody>
</table>
Part V

- Problem Definition and Motivation
- Example Embedded System – The Anti-lock Braking System
- Modeling Strategy, SPN Models and SAN Models
- Reliability Analysis Results and Discussion
- Conclusion and Scope of Future Work
Conclusions (1)

- **Modeling and Analysis**: The Anti-lock Braking System of a passenger vehicle was modeled (with emphasis on failure severity, coincident failures and usage profiles) and analyzed.

- **Realistic Models**: The models were built incrementally to achieve the best balance between faithfulness to the real system and keeping the model tractable at the same time.

- **Extensible Models**: The models developed can be easily extended to incorporate different levels of severity, other coincident failures and usage levels.
Conclusions (2)

- **Two stochastic formalisms**: Stochastic Petri Nets & Stochastic Activity Networks, were used to analyze the developed models for reliability measures.

- **Results** justified the modeling strategy adopted and highlighted the importance of modeling severity, coincident failures and usage-profiles while examining system reliability.

- **This research has successfully established a framework for investigating system reliability and the basis for further investigations in this application domain.**
Future Work (1)

- **Sensitivity Analysis**: The analysis of the effect of small variations in system parameters on the output measures and can be studied by computing the derivatives of the output measures with respect to the parameter.

- **Model the entire system**: The ABS is a small part of the DDR (Dynamic Driving Regulation) system which consists of other subsystems like the Electronic Steering Assistance (ESA) and the traction control (TC).
Future Work (2)

- **Simulation**: Evaluate the (complex) model numerically in order to *estimate* the desired true characteristics of the system.

- **Validation**: Results from experiments on the real system to validate analysis results to incrementally arrive at a realistic model.

- Generalization of modeling strategy for modeling both software and hardware components and the way of representing severity, coincident failures and usage profiles.
Contact Information

Frederick T. Sheldon, Ph.D. and Tom Potok, Ph.D.
Software Engineering for Dependable for Systems
Applied Software Engineering Laboratory

Rick:  865-576-1339
Tom:  865-574-0834
Fax:  865-574-6275

URL:  http://www.csm.ornl.gov/~sheldon
http://computing.ornl.gov/cse_home/acer.shtml
The End