Fault Models

Much work has been done on fault models. The discussion is based on the paper:


- There is an interesting follow-up paper "Verification of Hybrid Byzantine Agreement Under Link Faults" by P. Lincoln and J. Rushby that addresses a problem in the algorithm of Thambidurai and Park.
Fault Models

◆ Benign versus Malicious
  - Benign
    » error is self-evident
    » component does not undergo incorrect state transition during failure
    » examples:
      ■ crash fault
      ■ timing fault
      ■ data out-of-bound

■ what about “omissions”?
Fault Models

- Malicious
  » not self-evident to all non faulty receivers
  » can behave in two ways
  » symmetric
    ▶ received identically by all processors
  » asymmetric
    ▶ no restrictions of fault => anything goes

- Fault frequency
  » worse case every fault could behave asymmetric
  » best case all faults are benign
  » what is the best assumption for your system?
Fault Models

- Fault Taxonomy

- Relationship & Probability of Occurrence
  - note: this is not a venn diagram!
Fault Models

- Lamport Model
  - assumes that every fault is asymmetric

\[ N \geq 3t + 1 \]
\[ r' \geq t + 1 \quad \text{or} \quad r \geq t \] rebroadcasts

- Meyer + Pradhan 87
  - differentiates between malicious and benign faults

\[ N > 3m + b \]
\[ r > m \]

\[ m = \text{number of malicious faults} \]
\[ b = \text{number of benign faults} \]
Fault Models

◆ Thambidurai + Park 88
  – difference between malicious faults
    » symmetric faults
    » asymmetric faults
    » result:

\[ N > 2a + 2s + b + r \]

\[ r \geq a \]

» a = asym., s = sym., b = benign, r = rounds
» in general \( a_{\text{max}} < s_{\text{max}} < b_{\text{max}} \)
» or \( \lambda_a << \lambda_s << \lambda_b \)
» saves rounds and hardware
Fault Models

◆ Advantages of multi-fault model
  − 1) more accurate model of the system
    » less “overly conservative”
  − 2) resulting reliabilities are better
    » custom tailor recovery mechanisms
    » Example:
      ■ consider Byzantine solution using OM() algorithm
      ■ assume $N = 4, 5, 6$
      ■ still, only one fault is covered using the OM algorithm
      ■ moreover, the system reliability degrades
        − $N = 6$ results in worse reliability than $N = 4$
        − one is better off to turn the additional processors off!
    » see paper Tha88, page 98, table 1
**Fault Models**

Source: Tha88

<table>
<thead>
<tr>
<th>Model</th>
<th>N</th>
<th>P(Failure)</th>
<th>Faults</th>
</tr>
</thead>
<tbody>
<tr>
<td>BG</td>
<td>4</td>
<td>$6.0 \times 10^{-8}$</td>
<td>1 arbitrary</td>
</tr>
<tr>
<td>BG</td>
<td>5</td>
<td>$1.0 \times 10^{-7}$</td>
<td>1 arbitrary</td>
</tr>
<tr>
<td>BG</td>
<td>6</td>
<td>$1.5 \times 10^{-7}$</td>
<td>1 arbitrary</td>
</tr>
<tr>
<td>UM</td>
<td>4</td>
<td>$6.0 \times 10^{-8}$</td>
<td>1 arbitrary, $b = 0$, $s = 0$</td>
</tr>
<tr>
<td>UM</td>
<td>5</td>
<td>$1.0 \times 10^{-11}$</td>
<td>1 arbitrary, $b = 1$, $s = 0$</td>
</tr>
<tr>
<td>UM</td>
<td>6</td>
<td>$2.0 \times 10^{-11}$</td>
<td>1 arbitrary, $b = 0$, $s = 1$</td>
</tr>
<tr>
<td>UM</td>
<td>6</td>
<td>$1.1 \times 10^{-15}$</td>
<td>1 arbitrary, $b = 2$, $s = 0$</td>
</tr>
</tbody>
</table>

*Table 1: Reliability data for Example 1*
# Fault Models

Source: Tha88

<table>
<thead>
<tr>
<th>$\tau = 1$</th>
<th>$s$</th>
<th>$a = 0$</th>
<th>$a = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s$</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>$b = 0$</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>$b = 1$</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>$b = 2$</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>$b = 3$</td>
<td>5</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>$b = 4$</td>
<td>6</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>$b = 5$</td>
<td>7</td>
<td>9</td>
<td>11</td>
</tr>
<tr>
<td>$b = 6$</td>
<td>8</td>
<td>10</td>
<td>12</td>
</tr>
</tbody>
</table>

Table 2: Resiliency of a System based on the Unified Model (minimum number of processors required)
3) smarter degradation
   - we can specify the number of rounds
   - example using $N = 11$
     - let subscript $\max$ denote the maximum number of faults covered, assuming this is the only type of fault occurring.
     - if $r = 1$ then $a_{\max} = 1$ or $s_{\max} = 4$
     - if $r = 2$ then $a_{\max} = 2$ or $s_{\max} = 4$
   
   why? $s_{\max} = 4 \Rightarrow N > 2 \cdot 4 + 2 = 10$
   
   $s_{\max} = 5 \Rightarrow N \neq 2 \cdot 5 + 2 = 12$

- requirements for success
  - good estimate of fail rates $\lambda_a, \lambda_s, \lambda_b$
    - typically $\lambda_a << \lambda_s << \lambda_b$
  - good estimate of recovery rates $\rho_a, \rho_s, \rho_b$
    - typically $\rho_a < \rho_s < \rho_b$
Agreement algorithms

- Azadmanesh & Kieckhafer
  - partitions further into transmissive and omissive cases of malicious faults
Agreement algorithms

◆ Incomplete Interconnections
  - Lam82, Dol82
  - agreement only if the number of processors is less than 1/2 of the connectivity of the system’s network.

◆ Eventual vs. Immediate Byz. Agreement (EBA, IBA)
  - recall interactive consistency conditions IC1, IC2
  - an agreement is immediate if in addition to IC1 and IC2 all correct processors also agree (during the round) on the round number at which they reach agreement.
  - otherwise the agreement is called eventual
    » each processor has decided on its value, but cannot synchronize its decision with that of the others until some later phase.
    » Thus, agreement may not always need full t+1 rounds
Agreement algorithms

- Lamport OM \[ N \geq 3m + 1 \quad r = m + 1 \]
- Lamport SM \[ N \geq m + 2 \quad r \geq m + 1 \]
- Davis+Wakerly \[ N \geq 2t + 1 \quad S = t + 1 \]
- Meyer+Pradhan \[ N > 3m + b \quad r \geq m \]
- Thambidurai+Park \[ N > 2a + 2s + b + r \quad r \geq a \]
- Dol82a (EBA) \[ N > t^2 + 3t + 4 \quad r = \min(f + 2, t + 1) \]