Outlook:
Fault Tolerance in MPI Programs

Janko Strassburg

With material from W. Gropp, E. Lusk, Argonne National Laboratory

PATC Parallel Programming Workshop October 2013
Contents

Declaration

Existing FT MPI

FT & MPI standard

Write (non-transparent) FT in MPI

Summary & discussion
Fault tolerance is a property of a program, not of an API specification or an implementation.

Within certain constraints, MPI can provide a useful context for writing application programs that exhibit significant degrees of fault tolerance.
Current FT MPI

<table>
<thead>
<tr>
<th>Framework</th>
<th>API</th>
<th>Comm. Lib.</th>
<th>Automatic</th>
<th>Non Automatic</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optimistic log</td>
<td>Coordinated checkpoint</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Log based</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Pessimistic log</td>
<td></td>
</tr>
</tbody>
</table>

**Optimistic recovery in distributed systems**
- Cocheck: Independent of MPI Ste96
- Starfish: Enrichment of MPI AF 99
- Clip: Semi-transparent Checkpoint CLP97

**Sender based Mess. Log.**
- MPICH-CL: N faults ???
- Pruitt 98: 2 faults sender based Pru98
- Optimistic recovery in distributed systems
- Optimistic log
- Centralized server
- MPI-FT
- MPI-FT

**Log based**
- Manetho: FTA faults EZ92
- Manetho: Coordinated checkpoint
- Egida: RAV99

**Pessimistic log**
- Coordinated checkpoint

**FT-MPI**
- Modification of MPI routines
- User Fault Treatment

**MPI/FT**
- Redundance of tasks

**API**
- Co-check
- Independent of MPI Ste96
- Starfish
- Enrichment of MPI AF 99
- Clip
- Semi-transparent Checkpoint CLP97

**Comm. Lib.**
- MPICH-CL: N faults ???
- Pruitt 98: 2 faults sender based Pru98
- Sender based Mess. Log.
- Optimistic recovery in distributed systems
- Optimistic log
- Centralized server
- MPI-FT
- MPI-FT

**Framework**
- Optimistic recovery in distributed systems
- Coordinated checkpoint
- Coordinated checkpoint
- MPI-FT
- MPI-FT
FT is a property of an MPI program coupled with the MPI implementation.

Four lever of “survive”
- Automatically recovers (MPICH)
- Error notification (FT-MPI)
- Failure can be ignore (Manager/worker)
- Restart from checkpoint (CoCheck etc)

Ease of use
MPI Standard does mention about the FT.
- Require to implement reliable communication
- Built in or user defined error handlers
- Predefined error
Basic approach
  – Checkpointing & roll back
    • System directed
    • User directed
  – Redundancy & vote

Approach technique
  – MPI
  – Modify / Extend MPI
\[ E_T = T(1 + \frac{k_0}{t_0} + a(k_1 + \frac{t_0}{2})) \]
\[ 0 = \frac{dE_T}{dt_0} = -\frac{k_0}{t_0^2} + a/2 \]

\[ t_0 = \sqrt{\frac{2k_0}{\alpha}} \]

\[ E_T = T(1 + \alpha k_1 + \sqrt{2\alpha k_0}) \]

Additional cost
Use intercommunicators

Manager/Worker Model

Manager(s)
Centralized/Distributed work pool
intercommunicator

Worker processors

The intermediate status of the computing is stored at the manager party.
Modify/Extend MPI

Modify MPI Semantics
- Break the constrain of the MPI semantics
- Provide the programmer more error information and error handling methods

Extending MPI
- Define extensions to MPI (MPE_XXX)
- Encapsulate the MPI procedures
Summary

- MPI Standard provides in the way of support for writing fault-tolerant programs.

- Many approach could be used to write the “nontransparent” FT MPI program.