Implementing “Pliris-C/R” Into the EIGER Application

Mike Davis, Cray Inc.
CUG 2015
Agenda

- EIGER application
- Cielo system
- Pliris solver library
- Pliris-C/R
- Other resiliency features for EIGER
- Results from EIGER runs
- Frequency-domain EM code
- Dense matrix factor/solve, complex-valued elements
  - Over 2M unknowns
  - Runs on 5000 Cielo (XE6) nodes, MPI everywhere
  - Factor takes ~80000 seconds
Cielo

- 96 cabinet XE6
- 8944 compute nodes
  - Dual-socket 8-core Opteron (Magny-Cours) 2.4GHz
  - 32 GB RAM
- 1.11 PF HPL
  - Number 6 on TOP 500, June 2011
Pliris

- Dense solver package, part of Trilinos
- Block data distribution with torus-wrap mapping
- Block-cyclic work distribution (LU decomposition)
- Shuffle permutation of solution
- RHS vectors known in advance
Pliris-C/R Design

- **Checkpoint/restart covers only factor()**
  - Checkpoint occurs inside loop over columns
  - Restart occurs above loop over columns
- **Process checkpoint image includes:**
  - Local block of matrix (>1 GB/process)
  - Only relevant fraction of operand matrix saved
  - Work vectors
  - Pointers
Every process does I/O (no aggregation)

I/O operations are POSIX unbuffered
  - preadv(), pwritev()

Checkpoint files spread across multiple Lustre file systems

N processes $\leftrightarrow$ M files, with turnstiling

Checkpoint operations spaced evenly across factor() column loop work space
Cielo esFS Configuration

- 4 esLogin
- 102 XE6 LNET Routers
- 40 FTA
- 11 Purge
- 1 LMN

- 324-port Director-class QDR IB SW
- 316 IB cables
- 324-port Director-class QDR IB SW
- 316 IB cables

- MDS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS
- OSS

- MDT
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900
- 7900

- 260 ports
- 320 ports
- 260 ports

- 204 IB cables
- 204 IB cables
- 204 IB cables

- 144 OST
- 288 OST
- 144 OST
Cielo /lscratch3 I/O Bandwidth (MiB/sec)

- N processes → N files using LANL fs_test
- Source: B.M. Kettering, CUG 2014 Proceedings

<table>
<thead>
<tr>
<th>Processes</th>
<th>Eff. BW</th>
<th>Raw BW</th>
</tr>
</thead>
<tbody>
<tr>
<td>1024</td>
<td>73900</td>
<td>74400</td>
</tr>
<tr>
<td>2048</td>
<td>77400</td>
<td>78500</td>
</tr>
<tr>
<td>4096</td>
<td>76200</td>
<td>75500</td>
</tr>
<tr>
<td>8192</td>
<td>72000</td>
<td>75900</td>
</tr>
<tr>
<td>16384</td>
<td>64000</td>
<td>72000</td>
</tr>
<tr>
<td>32768</td>
<td>57600</td>
<td>69400</td>
</tr>
<tr>
<td>65536</td>
<td>43600</td>
<td>60900</td>
</tr>
</tbody>
</table>

Optimum
Turnstiling Basics

Waiting for turn

Doing I/O
Turnstiling Optimizations

● Processes that share a node take turns
  ● Keeps injection demand below limit

● Processes sharing an OS image share open file descriptors
  ● Reduces metadata load
## Single-OST Checkpoint Times (sec)

<table>
<thead>
<tr>
<th>Test</th>
<th>Avg</th>
<th>Std Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>NXN</td>
<td>11640</td>
<td>367</td>
</tr>
<tr>
<td>NX1</td>
<td>7697</td>
<td>721</td>
</tr>
<tr>
<td>NX5</td>
<td>7747</td>
<td>697</td>
</tr>
<tr>
<td>TURN5</td>
<td>6918</td>
<td>800</td>
</tr>
<tr>
<td>TURN5_SFD</td>
<td>6718</td>
<td>665</td>
</tr>
</tbody>
</table>
Pliris-C/R Tuning Parameters

- PLIRIS_CR_NFS: number of file systems
- PLIRIS_CR_DIR: directory paths; 1 per FS
- PLIRIS_CR_NS: OST counts; 1 per FS
- PLIRIS_CR_NF: number of files in checkpoint set
- PLIRIS_CR_COUNT: number of checkpoint sets to write over the course of factor()
- PLIRIS_CR_SIGNUM: signal number for imminent termination due to wall time or scheduled shutdown
Pliris-C/R Settings for EIGER

- PLIRIS_CR_NFS=3
- DIR2=/lscratch2/${USER}/${PBS_JOBNAME}
- DIR3=/lscratch3/${USER}/${PBS_JOBNAME}
- DIR4=/lscratch4/${USER}/${PBS_JOBNAME}
- PLIRIS_CR_DIR="${DIR2} ${DIR3} ${DIR4}"
- PLIRIS_CR_NS="125 250 125"
- PLIRIS_CR_NF=2500
- PLIRIS_CR_COUNT=6
- PLIRIS_CR_SIGNUM=23
Coordination of Checkpoints

- Selected iterations of loop over columns
  \[ b_i = N - \frac{3}{\sqrt[k]{1 - i}} \left( \frac{N}{\sqrt[k]{1 + i}} \right) \]
  - \( i \) is the checkpoint number (1 .. \( k \))
  - \( b_i \) is the column index at which checkpoint \( i \) is written
  - \( N \) is the size of the matrix (trip count of column loop)
  - \( k \) is the number of checkpoints to write (PLIRIS_CR_COUNT)
Decrementing Checkpoint of Matrix

\[ E_i = \frac{N b_{i-1}}{p^2} \]
Decrementing Checkpoint of Matrix (2)

\[ E_i = \frac{Nb_{i-1}}{p^2} \]

- **factored**
- **eliminated**

- **N^2/p^2**
Decrementing Checkpoint of Matrix (3)

\[ E_i = \frac{N b_{i-1}}{p^2} \]

The diagram shows a matrix with an diagonal line indicating factored and eliminated elements.
Selection of Checkpoint Count

- Minimize total work time

\[ T_W(N) = M \times e^{(F+\rho)/M} \times \sum_{i=1}^{N} \left( e^{(T_S/N+\delta(i))/M} - 1 \right) \]

- \( N \) is number of segments in calculation
- \( M \) is MTBF for a 5000-node compute app (131572)
- \( F \) is matrix fill time (900)
- \( \rho \) is time to read the checkpoint sets (1440)
- \( T_S \) is total matrix factor time (81573)
- \( \delta(i) \) is time to write checkpoint set \( i \) [\( \delta(N)=0 \)]
  - \( 960 \times \frac{3}{\sqrt{(N+1-i)/N}} \)
Selection of Checkpoint Count (2)

- Values of $T_w$ for various choices of $N$

<table>
<thead>
<tr>
<th>N</th>
<th>$T_w$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>116858</td>
</tr>
<tr>
<td>2</td>
<td>99744</td>
</tr>
<tr>
<td>3</td>
<td>95334</td>
</tr>
<tr>
<td>4</td>
<td>93631</td>
</tr>
<tr>
<td>5</td>
<td>92946</td>
</tr>
<tr>
<td>6</td>
<td>92572</td>
</tr>
<tr>
<td>7</td>
<td>92832</td>
</tr>
</tbody>
</table>

Optimal
Other Pliris-C/R Resilience Features

● **EIGER job script enhancements**
  - On aprun termination, checks stdout/stderr for signs of recoverable conditions (node failures) and relaunches within the job using spare node(s)

● **Pliris_cr**
  - Tool to set up, verify, and clean up checkpoint sets
  - Saves on file open times in parallel application
  - Helps with scratch directory hygiene

● **Pliris_watch**
  - Tool to watch running EIGER job, and report/act on signs of stalls
Results from EIGER Runs with Pliris-C/R

● First successful run 4/24/2014 (Job 1474501)
  ● 6 checkpoint writes: 956 sec → 871 sec
  ● 1 checkpoint read/restart: 1435 sec
  ● Performance compares well with fs_test and other turnstiling apps

● Strange run 11/25/2014 (Job 1568851)
  ● 7 checkpoint writes: 2826 sec → 2004 sec
  ● 1 checkpoint read/restart: 2019 sec
  ● Full file system? Overlapped with file system directory tree walk?

● Latest run 2/27/2015 (Job 1627163)
  ● Assertion failed in MPI_Barrier: recv_pending (BUG 824088)
Areas of Future Work

- Port to Trinity (DataWarp + DNE)
- Skip matrix fill on restart run
- First-come, first-served queueing on turnstiles
- Improve checkpoint interval
  - Closer to optimal
  - Adjustable in restart runs
- Overlap I/O on static portion of matrix with factorization of active portion
Summary

- Adding C/R to a dense solver is viable
- Turnstiling still helps I/O
- Shared file descriptors can help I/O
- Good citizenship promotes resiliency
Acknowledgements

- Courtenay T. Vaughn (SNL), Brett M. Kettering (LANL), Dan Poznanovic (CRAY)
  - Reviewed paper and gave valuable feedback
- William W. Tucker (formerly Cray Inc.)
  - Coauthor
- Joseph D. Kotulsiki (SNL)
  - Coauthor
Q&A