Objective & index

Objective:
- Describe the analysis and improvements in the Sweep3D code using Paraver
- Compare MPI and OpenMP versions

Index
- The algorithm
- Analysis of the original versions
- Alternative structures
- Analysis:
  - Sequential
  - OpenMP
  - Mixed mode
- Conclusions
Sweep3D

- All we know about it
  - US DOE ASCI Benchmark
  - 3D deterministic particle transport simulation
  - Problem size: 50^3, 125^3, XXX^3

Not much

Sweep3D

- Outer computational structure
  Iterate till convergence
  : Compute sweep from each corner to its opposite.
  Reduction on each direction
  :

- Iteration space
  - 3D space : i, j, k
  - plus : m, n

- Data structures
  - 3D space
  - 2D projections
  - 1D temporal reductions
### MPI Parallelization

- **Grid Partitioning of $ij$ plane**
- **One column per MPI task**
- **Reduction on each direction:**
  - Communicate on $i$ and $j$
  - Pipeline $k$ dimension
  - + overlap consecutive sweeps if possible

- **Input file parameters**
  - Blocks on $i$ and $j$ directions
  - Pipelining block

---

### OpenMP Parallelization

- **Thought for mixed mode (MPI + OpenMP)**
  - Just core computational loop
- **Reduction on each direction**
  - Compute along diagonal wavefronts

```fortran
DO idia
  DO jkm =1, #points in wavefront
    j,k,m = f(idia,jkm)
    DO n,i ! phi, src
    DO i ! phijb, phiib, phi, phi
    DO n,i ! flux, phi
    DO i ! face, phiib, phiib, phiib
```

---

Jesús Labarta, Judit Gimenez, SCICOMP3, May 2001
**MPI vs. OpenMP: some numbers**

- **Problem**
  - size: $50^3$
  - k plane pipelining: 10

<table>
<thead>
<tr>
<th>NB Domains</th>
<th>OpenMP time</th>
<th>Decomposition</th>
<th>MPI time</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>7.78</td>
<td>1x6</td>
<td>3.97</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2x3</td>
<td>3.81</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3x2</td>
<td>3.71</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6x1</td>
<td>4.47</td>
</tr>
<tr>
<td>12</td>
<td>6.55</td>
<td>1x12</td>
<td>3.50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2x6</td>
<td>2.74</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3x4</td>
<td>2.21</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4x3</td>
<td>2.25</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6x2</td>
<td>2.97</td>
</tr>
<tr>
<td></td>
<td></td>
<td>12x1</td>
<td>3.98</td>
</tr>
</tbody>
</table>
MPI: decomposition effect

- 1 x 12
- 3 x 4
- 12 x 1
DO diag
  DO jkm = 1, #points in wavefront
    j, k, m = f(diag, jkm)
  DO n, i
    ! phi, src
    DO i
      ! phi, phijb, phikb, phi, phi
    DO n, i
      ! flux, phi
    DO i
      ! face, phi, phijb, phikb

Computation:
- Complex Overhead
- Triangular trip count
- OpenMP RTL overhead
- Invalidation traffic

Invalidations
- OpenMP
- MPI
Alternative structures

DO idia
DO jkm =1, #points in wavefront
j,k,m=f(idia,jkm)
DO n,i  ! phi, src
  DO i    ! phi, src
  DO n,i  ! flux, phi
  DO i    ! face, phi, phi, phi

MKji form
in the distributed source

Alternative forms

- mkji
  DO k
  DO j
  DO m
  DO n,i
  DO i

- kjmi
  DO k
  DO j
  DO m
  DO n,i

- Kjmi
  DO K
  DO j
  DO m
  DO n,i

- jkmi
  DO j
  DO k
  DO m
  DO n,i

- ...

Jesús Labarta, Judit Gimenez, SCICOMP3, May 2001
Sequential

- **L1 misses**
  - average
    - diag: 6195
    - mkji: 2723
    - kjmi: 1473
  - Program phases in “diag”
    - Conditional

Sequential

- **L2 misses**
  - average
    - diag: 485
    - mkji: 2062
    - kjmi: 479
  - Better L1 ≠ better L2
  - Influence of direction in mkji
    - same source loop
Sequential

- IPC

  - average
    - diag: 0.95
    - mkji: 0.62
    - kjmi: 1.16

Sequential

- Prefetching:
  - 4 Prefetch streams
    - ????
    - + info
  - Big difference between versions
  - Big difference between directions
Sequential

- Prefetching:
  - Demand fetch blocked by prefetch
    - ???
    - + info
  - Difference between versions
  - Difference between directions

OpenMP: some numbers

- Problem
  - size: 50^3
  - k plane pipelining: 10

<table>
<thead>
<tr>
<th>Elapsed Time</th>
<th>OMP</th>
<th>OLM_THREADS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>ccrit</td>
<td>24.40</td>
<td>28.26</td>
</tr>
<tr>
<td>cpipe</td>
<td>24.60</td>
<td>25.63</td>
</tr>
<tr>
<td>diag</td>
<td>16.21</td>
<td>17.28</td>
</tr>
<tr>
<td>kjmi</td>
<td>12.95</td>
<td>14.86</td>
</tr>
<tr>
<td>Kjmii</td>
<td>12.94</td>
<td>14.91</td>
</tr>
</tbody>
</table>

Invalidations

Instruction count overhead?
OpenMP: contention on locks

- Version ccrit, 6 Threads

```
DO k
  DO j
    DO m
      DO n, i
        DO i
          DO n, i
          DO i
```

OpenMP: Insufficient parallelism

- Version cpipe, 7 threads
  - outer iteration count: 6
  - parallel + worksharing
  - Internal pipelined iteration
OpenMP: instruction count overhead?

Sequential version OMP_NUM_THREADS=1  %
Intructions 54.65 60.84 11.3
Loads 21.08 21.4 1.5
Stores 11.37 11.41 0.4
Branches 3.36 7.97 137.2
Intops 2.257 3.45 52.9
Flops 16.583 16.61 0.2

Comparing modes

Single mode better

Decomposition effect  Scheduling Interference

Comparing modes

Decomposition effect  Scheduling Interference

Comparing modes

Decomposition effect  Scheduling Interference
Mixed mode: scheduling interference

- MPI (4 tasks) + kjmi
  - 4 threads, k pipeline=10

- 2 threads, k pipeline=10

- 4 threads, k pipeline=12
  - Less K iterations

Conclusions

- Performance analysis
  - An art / full of surprises
  - Never ending task
  - Chatting is nice / seeing is believing / measuring is better

- OMPIttrace + Paraver
  - Useful / Flexible

- Mixed model MPI+OpenMP
  - Take care with interferences

- Sweep3D
  - Performance depends on many input parameters in a coupled way
  - Interesting / great fun