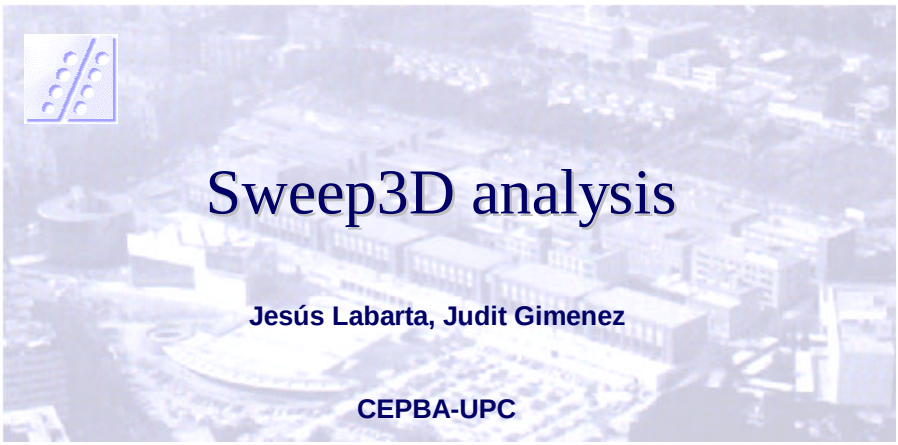


Sweep3D analysis

Jesús Labarta, Judit Gimenez

CEPBA-UPC

Technology Transfer | Research | Training | Mobility of Researchers
User Support | Education | HPC Facilities | Parallel Expertise

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

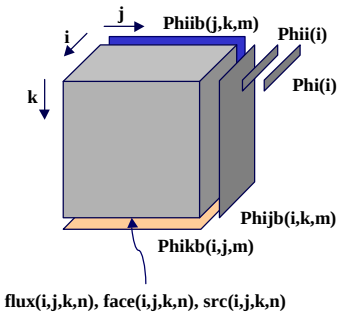
Not much

Sweep3D

- **Outer computational structure**

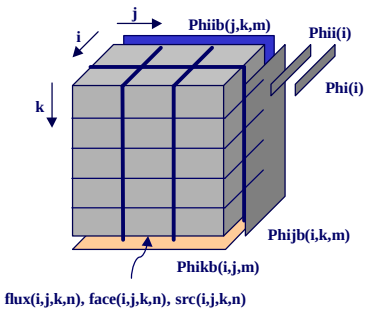
```
Iterate till convergence
   :
   Compute sweep from each corner
   to its opposite.
   Reduction on each direction
   :
```

- **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



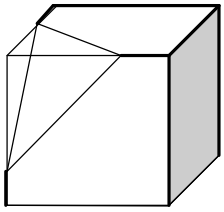
- **Input file parameters**
  - Blocks on `i` and `j` directions
  - Pipelining block

MPI data parallelization

Flux, face,src:	**DISTRIBUTED**
phii, phi:	**PRIVATIZED**
phikb:	**DISTRIBUTED**
phijb,phiib:	**DISTRIBUTED&REPLICATED**
	=> Communication

OpenMP Parallelization

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



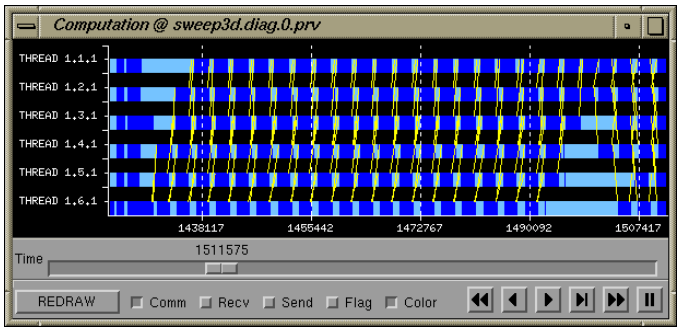
```
DO idiag
   DO jkm =1, #points in wavefront
      j,k,m=f(idiag,jkm)
      DO n,i     ! phi, src
      DO i       ! phijb, phikb, phii, phi
      DO n,i     ! flux, phi
      DO i       ! face, phii, phijb, phikb
```

MPI vs. OpenMP: some numbers

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

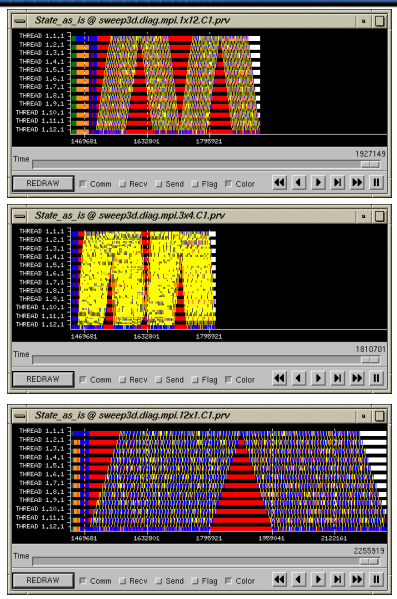
NB Domains	OpenMP time	Decomposition	MPI time
6	7.78	1x6	3.97
		2x3	3.61
		3x2	3.71
		6x1	4.47
12	6.55	1x12	3.50
		2x6	2.74
		3x4	2.21
		4x3	2.25
		6x2	2.97
		12x1	3.98

MPI



MPI : decomposition effect

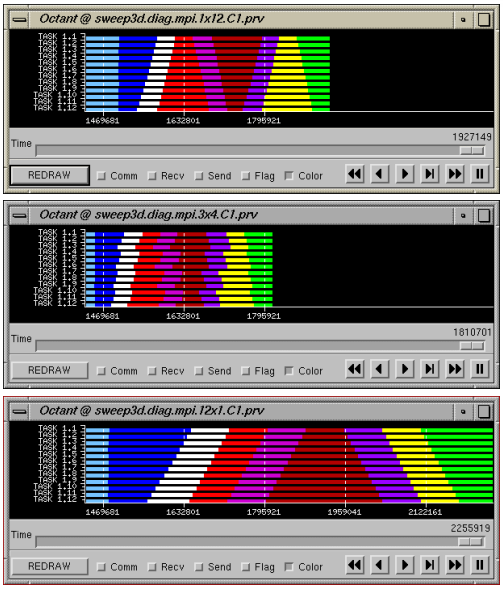
- 1 x 12
- 3 x 4
- 12 x 1



Jesús Labarta, Judit Gimenez, SCICOMP3, May 2001

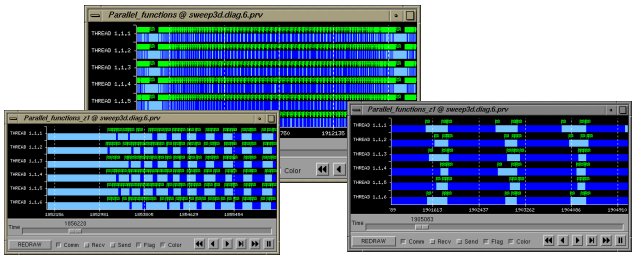
MPI : decomposition effect

- 1 x 12
- 3 x 4
- 12 x 1



Jesús Labarta, Judit Gimenez, SCICOMP3, May 2001

Pure OpenMP



```
DO idiag
   :
   DO jkm =1, #points in wavefront
      j,k,m=f(idiag,jkm)
      DO n,i      ! phi, src
      DO i        ! phijb, phikb, phii, phi
      DO n,i      ! flux, phi
      DO i        ! face, phii, phijb, phikb
```

Parallel

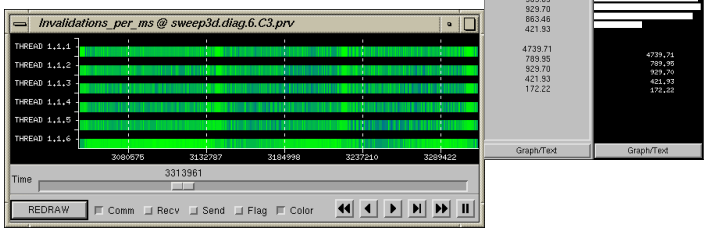
Computation:
- **Complex**
- **Overhead**

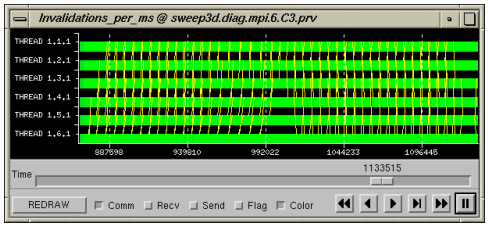
Triangular trip count
- **OpenMP RTL overhead**
- **Invalidation traffic**

Computation:
- **Complex**
- **Overhead**

Pure OpenMP

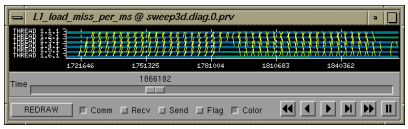
- **Invalidations**
  - OpenMP
  - MPI

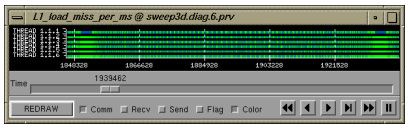




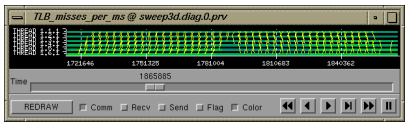
MPI vs. OpenMP

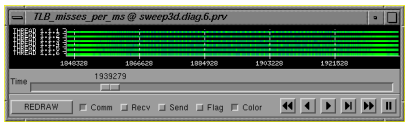
- MPI
- OpenMP



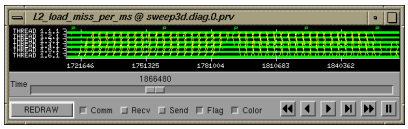


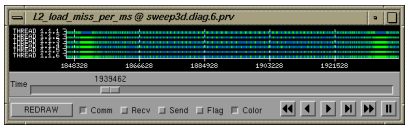
L1





TLB



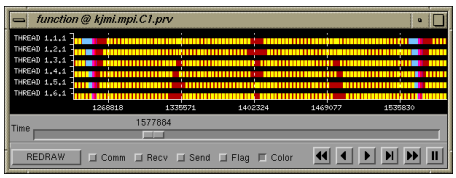


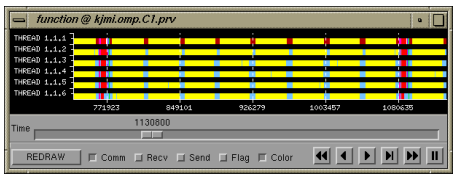
L2

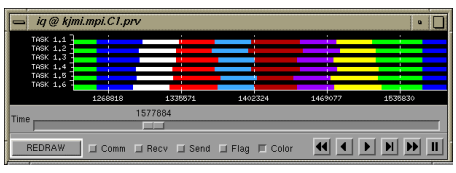
Jesús Labarta, Judit Gimenez, SCICOMP3, May 2001

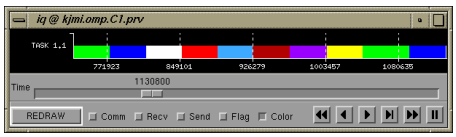
MPI vs. OpenMP

- MPI
- OpenMP









Jesús Labarta, Judit Gimenez, SCICOMP3, May 2001

Alternative structures

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

mkji form

in the distributed source

```
DO m
   DO k
      DO j
         DO n,i      ! phi, src
         DO i        ! phijb, phikb, phii, phi
         DO n,i      ! flux, phi
         DO i        ! face, phii, phijb, phikb
```

Alternative forms

- mkji

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

- Kjkmi

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

- kjmi

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

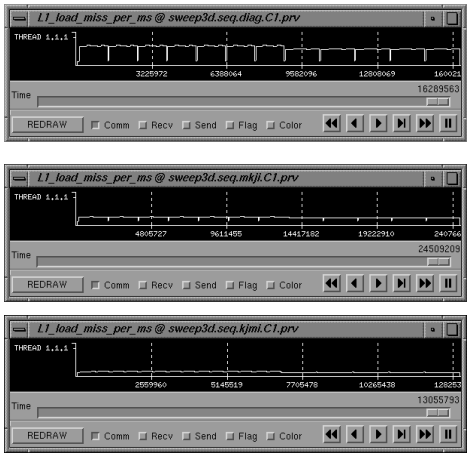
- jkmi

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

- . . .

Sequential

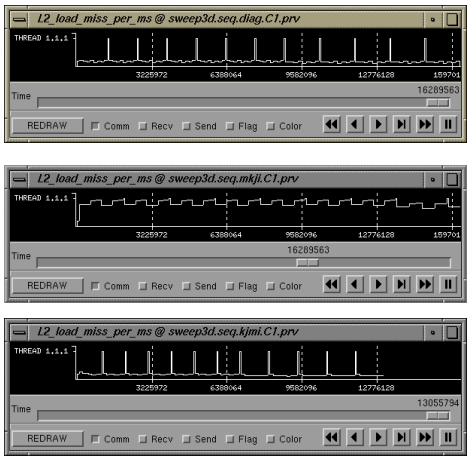
- **L1 misses**



- **average**
  - diag: 6195
  - mkji: 2723
  - kjmi: 1473
- **Program phases in "diag"**
  - Conditional

Sequential

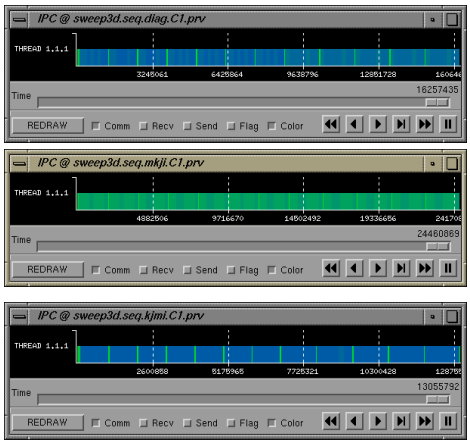
- **L2 misses**



- **average**
  - diag: 485
  - mkji: 2062
  - kjmi: 479
- **Better L1 ¹ better L2**
- **Influence of direction in mkji**
  - same source loop

Sequential

- **IPC**



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

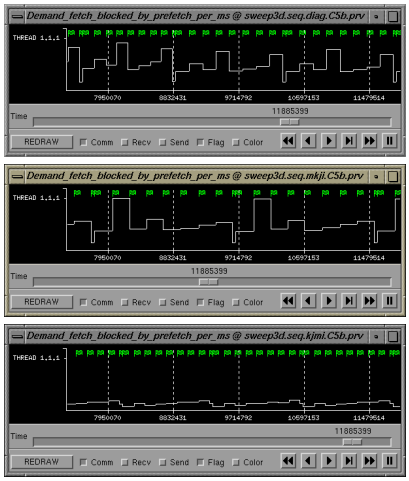
Sequential

- **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

Insufficient parallelism

Locality

Contention on locks

Invalidation, RTL overhead

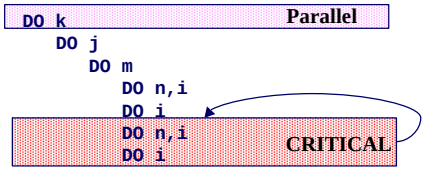
Elapsed Time	OMP_NUM_THREADS																
Version	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
ccrit	24.40	28.26	24.41	26.84	26.47	29.28	30.34						30.43				
cpipe	24.60	25.63	18.45	13.01	12.53	10.06	7.67						7.76				
diag	16.21	17.28	13.09	11.40	9.64	8.50	7.78						6.55				
kjmi	12.95	14.86	10.01	7.35	5.82	4.89	4.34	3.80	3.62	3.38	3.09	3.04	2.88	2.69	2.69	2.64	2.53
Kjkmi	12.94	14.91	8.47	6.35	4.91	4.24	3.58	3.46	2.90	2.81	2.78	2.65	2.29	2.22	2.16	2.19	2.15

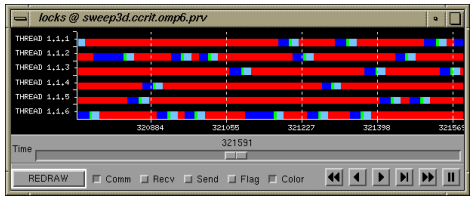
Invalidations

Instruction count overhead?

OpenMP : contention on locks

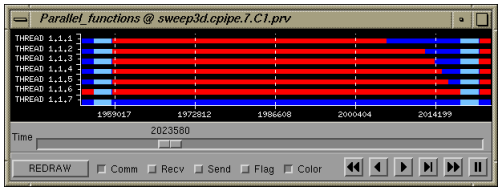
- **Version ccrit, 6 Threads**



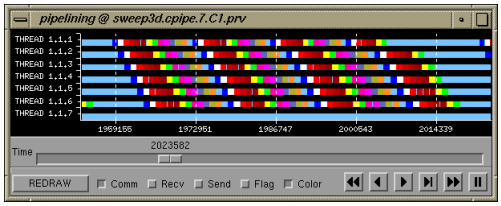


OpenMP: Insufficient parallelism

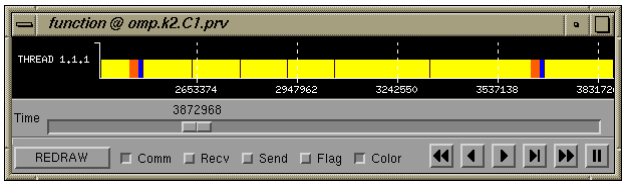
- **Version cpipe, 7 threads**
  - outer iteration count: 6
  - parallel + worksharing



  - Internal pipelined iteration



OpenMP: instruction count overhead ?



	Sequential version	OMP_NUM_THREADS=1	%
Instructions	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

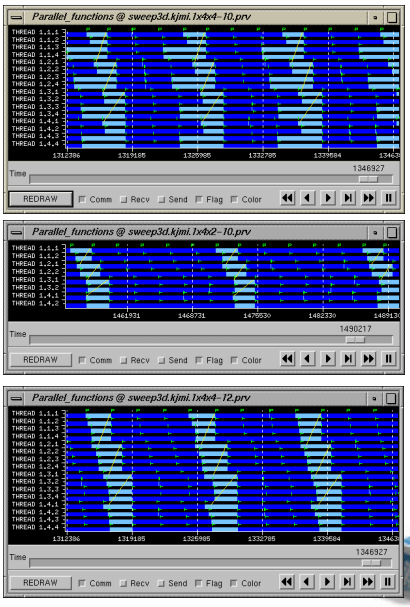
		Seq	OpenMP				
MPI	partition		1	2	4	8	16
0	-				66,35		
1	1x1				66,99	38,21	23,55
2	1x2			79,34	50,90	36,58	
	2x1			89,60	57,52	42,36	
4	1x4		71,67	41,92	28,24		
	2x2	79,72	86,71	52,01	34,60		
	4x1	76,04	86,04	54,51	37,05		
8	1x8	37,83	40,27	24,90			
	2x4	42,93	46,04	28,84			
	4x2	46,39	50,74	33,79			
	8x1	44,71	50,91	36,34			
16	1x16	23,12	24,82				
	4x4	24,75	26,86				
	16x1	28,40	32,60				

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
- **OMPItrace + 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