

Livermore loops optimization with OpenMP

Kernel 21:

- optimization manner:

- ♦ changing loops order (bigger loop be outer)
- ♦ loop unrolling (5 times)
- ♦ `omp_set_num_threads(THREADS);`
- ♦ `#pragma omp parallel shared(px, vy, cx) private(k, i, j)`

- compiling:

```
gcc -fopenmp ker21.c -m64 -lrt -lc -lm -o ker21
```

- analyzing best number of threads:

average speedup per 20 time of run

| Threads | 2 | 3 | 4 | 5 |
|----------------------|-------|-------|-------|-------|
| N = 101 (array size) | 1.780 | 2.908 | 3.217 | 2.264 |

| Threads | 6 | 8 | 10 | 12 |
|-----------------------|-------|-------|-------|-------|
| N = 1001 (array size) | 7.373 | 9.508 | 0.862 | 0.782 |

- results:

```
saleh@saleh-afzoon:~/Desktop/Link to Master,
list size n = 101
number of threads= 4
times of run = 20.
-----
average serial time: 1631.800 microsecond
average paralel time: 507.250 microsecond
-----
speed up: 3.217 microsecond
```

```
saleh@saleh-afzoon:~/Desktop/Link to Master,
list size n = 1001
number of threads= 8
times of run = 20.
-----
average serial time: 39311.150 microsecond
average paralel time: 4134.400 microsecond
-----
speed up: 9.508 microsecond
```

Kernel 22:

- optimization manner:

- ♦ loop unrolling (4 times)
- ♦ `omp_set_num_threads(THREADS);`
- ♦ `#pragma omp parallel shared(y, u, v, w, x) private(k)`

- compiling:

```
gcc -fopenmp ker22.c -m64 -lrt -lc -lm -o ker22
```

- results:

```
saleh@saleh-afzoon:~/Desktop/Link to Master/  
list size n = 101  
number of threads= 3  
times of run = 20.  
-----  
average serial time: 24.050 microsecond  
average paralel time: 20.550 microsecond  
-----  
speed up: 1.170 microsecond
```

```
saleh@saleh-afzoon:~/Desktop/Link to Master/  
list size n = 1001  
number of threads= 5  
times of run = 20.  
-----  
average serial time: 53.650 microsecond  
average paralel time: 40.350 microsecond  
-----  
speed up: 1.330 microsecond
```

Kernel 23:

- optimization manner:

- ♦ changing loops order (bigger loop be outer)
- ♦ loop unrolling (2 times)
- ♦ `omp_set_num_threads(THREADS);`
- ♦ `#pragma omp parallel private(j)`
- ♦ collapsing nested loops with `#pragma omp collapse(2)`

- compiling:

```
gcc -fopenmp ker23.c -m64 -lrt -lc -lm -o ker23
```

- results:

```
saleh@saleh-afzoon:~/Desktop/Link to Master/  
list size n = 101  
number of threads= 4  
times of run = 40.  
-----  
average serial time: 112.500 microsecond  
average paralel time: 32.950 microsecond  
-----  
speed up: 3.414 microsecond
```

```
saleh@saleh-afzoon:~/Desktop/Link to Master/  
list size n = 1001  
number of threads= 5  
times of run = 40.  
-----  
average serial time: 418.975 microsecond  
average paralel time: 63.675 microsecond  
-----  
speed up: 6.580 microsecond
```

Kernel 24:

- optimization manner:
 - ♦ loop unrolling (4 times)
 - ♦ `omp_set_num_threads(THREADS);`
 - ♦ `#pragma omp parallel private(k)`

- compiling:

```
gcc -fopenmp ker24.c -m64 -lrt -lc -lm -o ker24
```

- results:

```
saleh@saleh-afzoon:~/Desktop/Link to Master
list size n = 1001
number of threads= 4
times of run = 20.
-----
average serial time: 27.350 microsecond
average paralel time: 21.450 microsecond
-----
speed up: 1.275 microsecond
```