Erlangen Regional Computing Center

UNIVERSITÄT GREIFSWALD
Wissen lockt. Seit 1456

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG

Winter term 2020/2021

Parallel Programming with OpenMP and MPI

Dr. Georg Hager
Erlangen Regional Computing Center (RRZE) at Friedrich-Alexander-Universität Erlangen-Nürnberg
Institute of Physics, Universität Greifswald

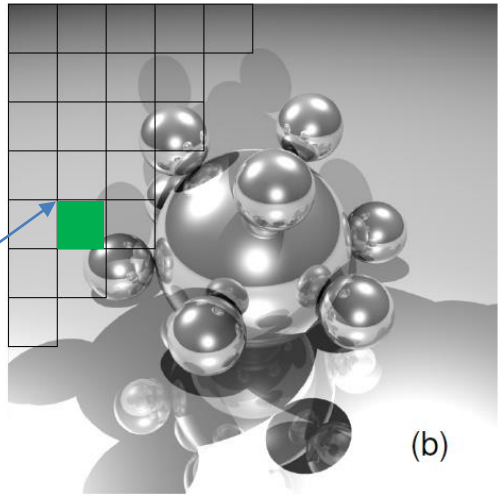
Assignment 9 discussion

High Performance Computing

Assignment 9, Task 1 – ray tracer w/o explicit copy

- Goal: get rid of this code on the master:

```
for(i=0; i<tilesize; i++){
  tilebase=ts2*tilesize*xtiles+ts1;
  memcpy((void*)(picture+tilebase+i*tilesize*xtiles),
                 (void*)(tile+i*tilesize),
                  tilesize*sizeof(unsigned char));
}
```



- … but rather receive the data directly:

```
MPI_Recv(picture + y*tilesize*xtiles + x, // target address
         1,                               // 1 element
         tiletype,                        // of this type
         source,                          // from source
         1,                               // tag
         MPI_COMM_WORLD, &status);
```

Assignment 9, Task 1 – ray tracer w/o explicit copy

- Setting up the data type:

```
ar_subsizes[0] = ar_subsizes[1] = tilesize;
ar_starts[0] = ar_starts[1] = 0;
if(0==my_rank) {
  ar_sizes[0] = ar_sizes[1] = size;
} else {
  ar_sizes[0] = ar_sizes[1] = tilesize;
}
MPI_Type_create_subarray(2, ar_sizes,
                         ar_subsizes,
                         ar_starts,
                         MPI_ORDER_C,
                         MPI_CHAR,
                         &tiletype);
MPI_Type_commit(&tiletype);
```

Assignment 9, Task 2 – dMVM analysis with ITAC

- Link with tracing library:

```
$ module load intel64 itac
$ mpiicc -Ofast -xHost -trace dmvm.cc timing.o
```

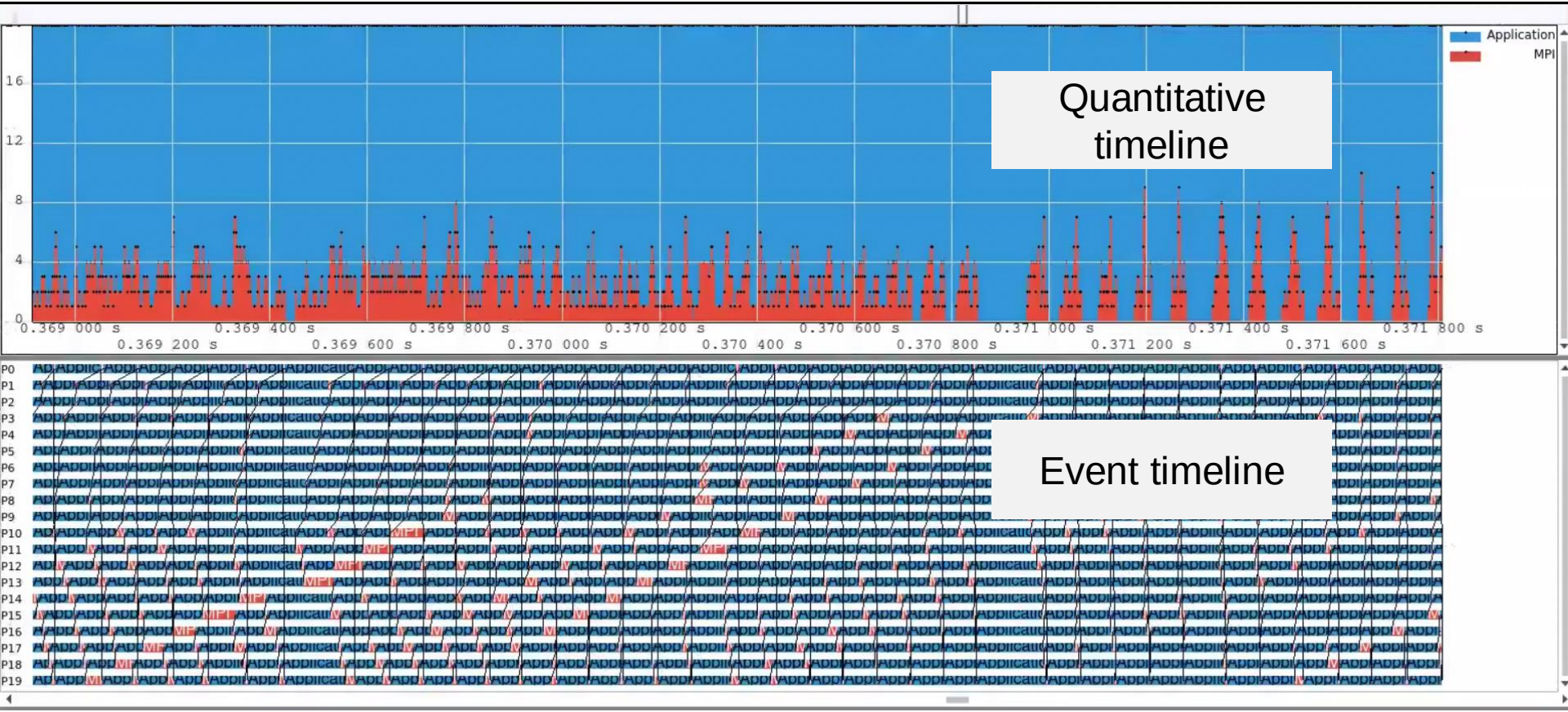
- Run as usual:

```
$ mpirun_rrze -np 20 -pin 0_1_...._19 ./a.out 10000 10000
```

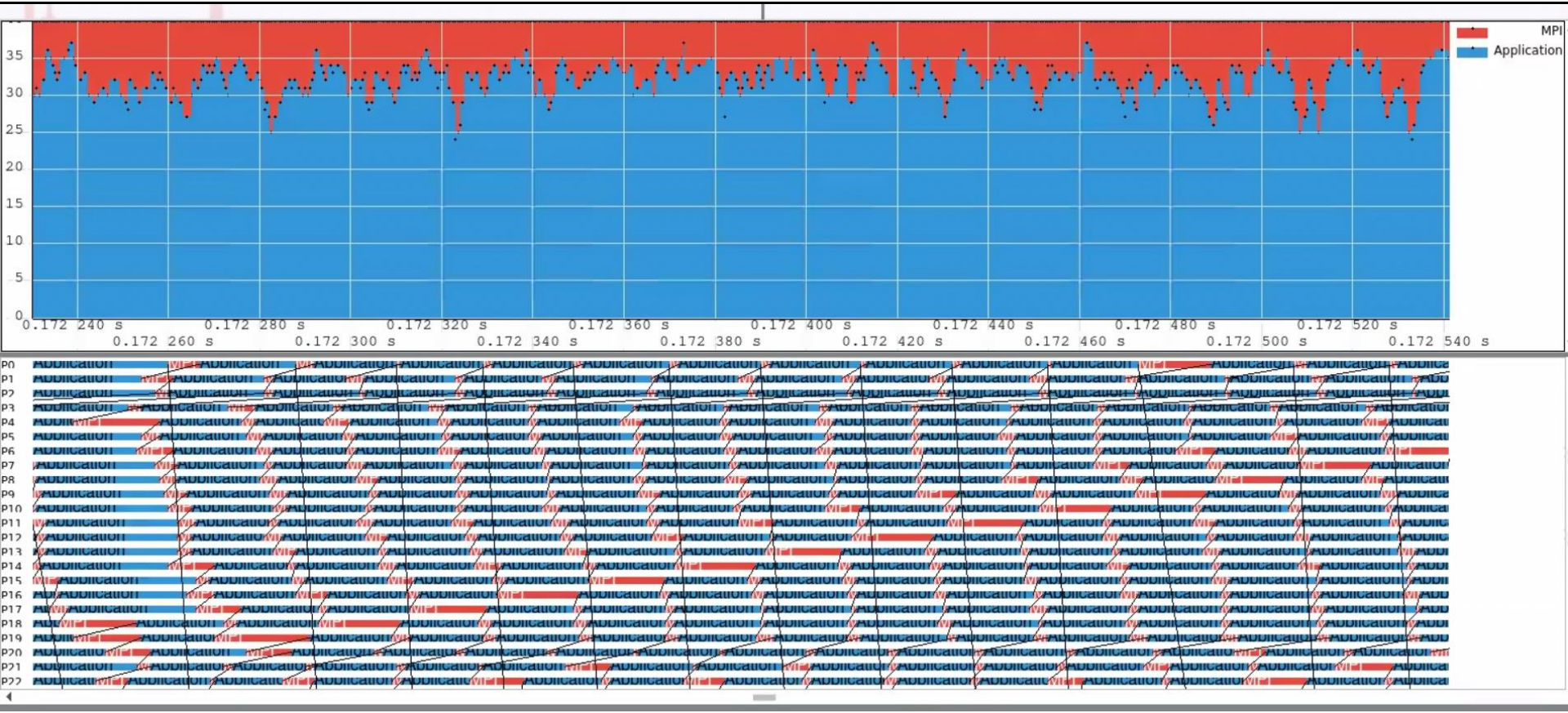
- Then:

```
$ traceanalyzer a.out.stf
```

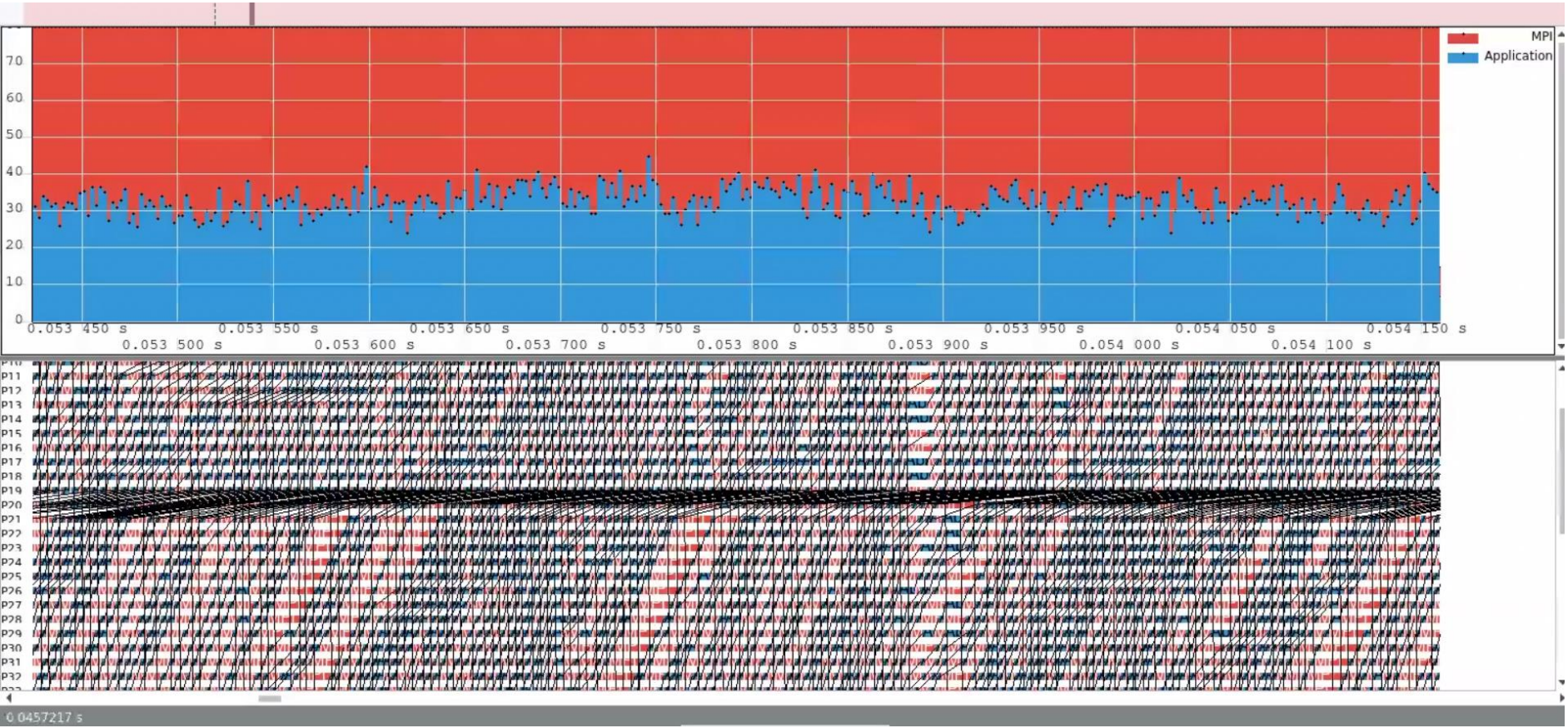
Size=4000, np=20



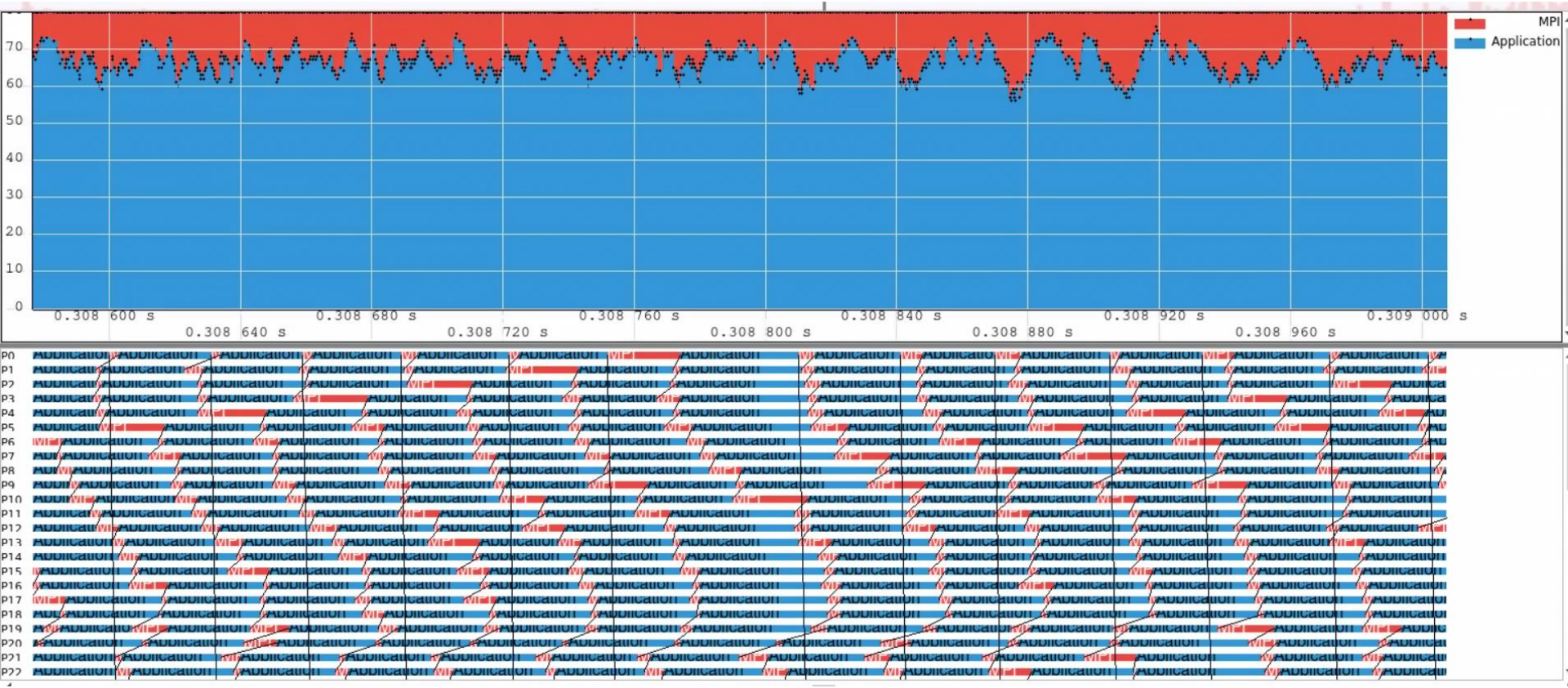
Size=4000, np=40



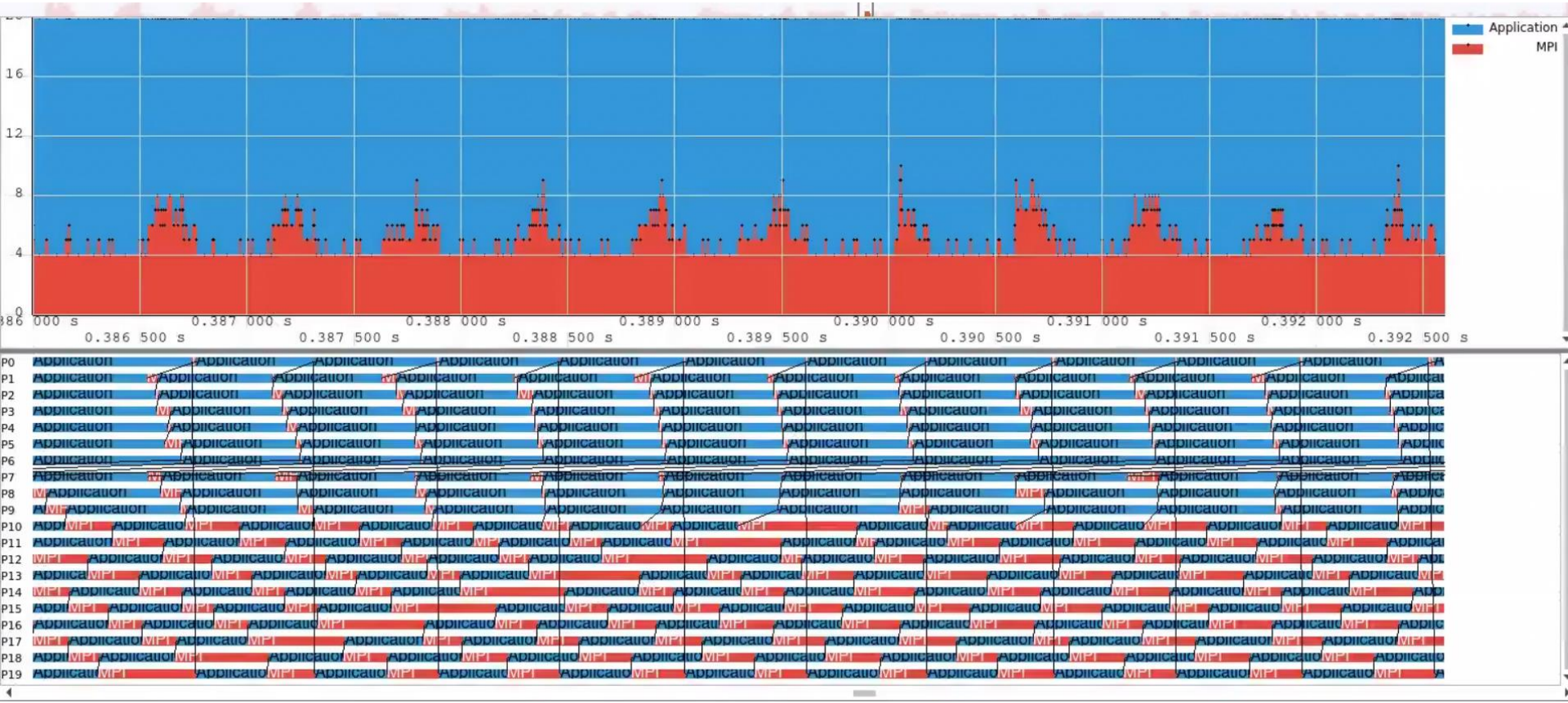
Size=4000, np=80



Size=10000, np=80



Size=10000, np=20



Assignment 9, Task 3: a bus network

- Back-of-the-envelope overhead estimate for 80 processes and n=10,000
  - Time for actual MVM execution:

$$T_{exec} = \frac{8 \times 10^8 \text{ B}}{8 \times 40 \times 10^9 \text{ B}/s} = 2.5 \times 10^{-3} \text{s}$$

  - Time for communication:
    Need 80 ring shifts with 10000/80 = 125 elements,
    i.e. 1000 byte
    → $\approx 3\ \mu s$ per shift on 4 nodes (bus network) → $\approx 4 \times 240\ \mu s$ overhead
    → 40% communication overhead in this case