

Read-Modify-Write Performance in NUMA Architectures

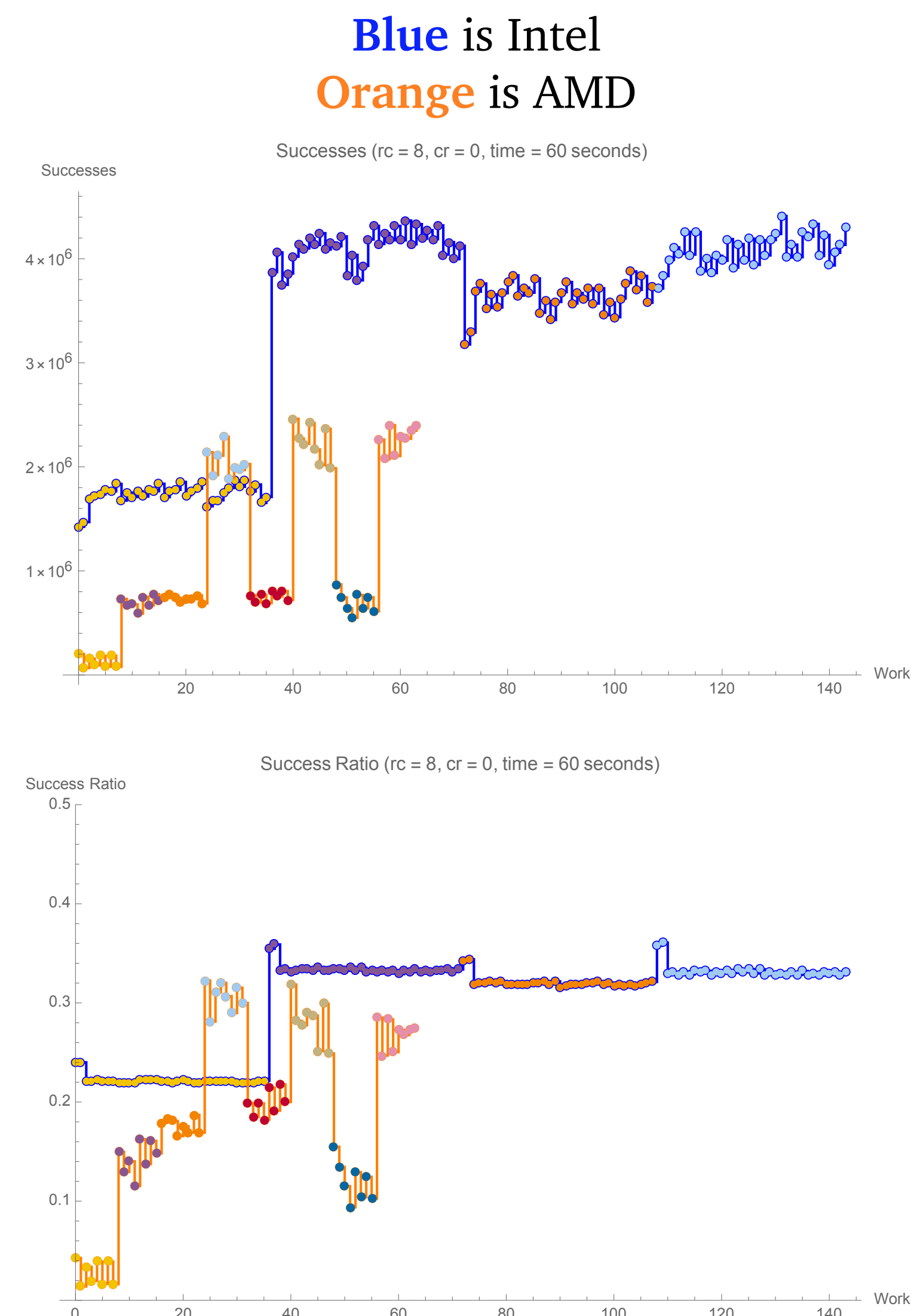
Naama Ben-David and Ziv Scully

Benchmark

```
do {
    wait(cr)
    val = Read(x)
    new_val = (myID, iter++)
    wait(rc)
} while(!CAS(x, val, new_val))
```

rc (read-to-CAS) delay represents “real work”
cr (CAS-to-read) delay is backoff

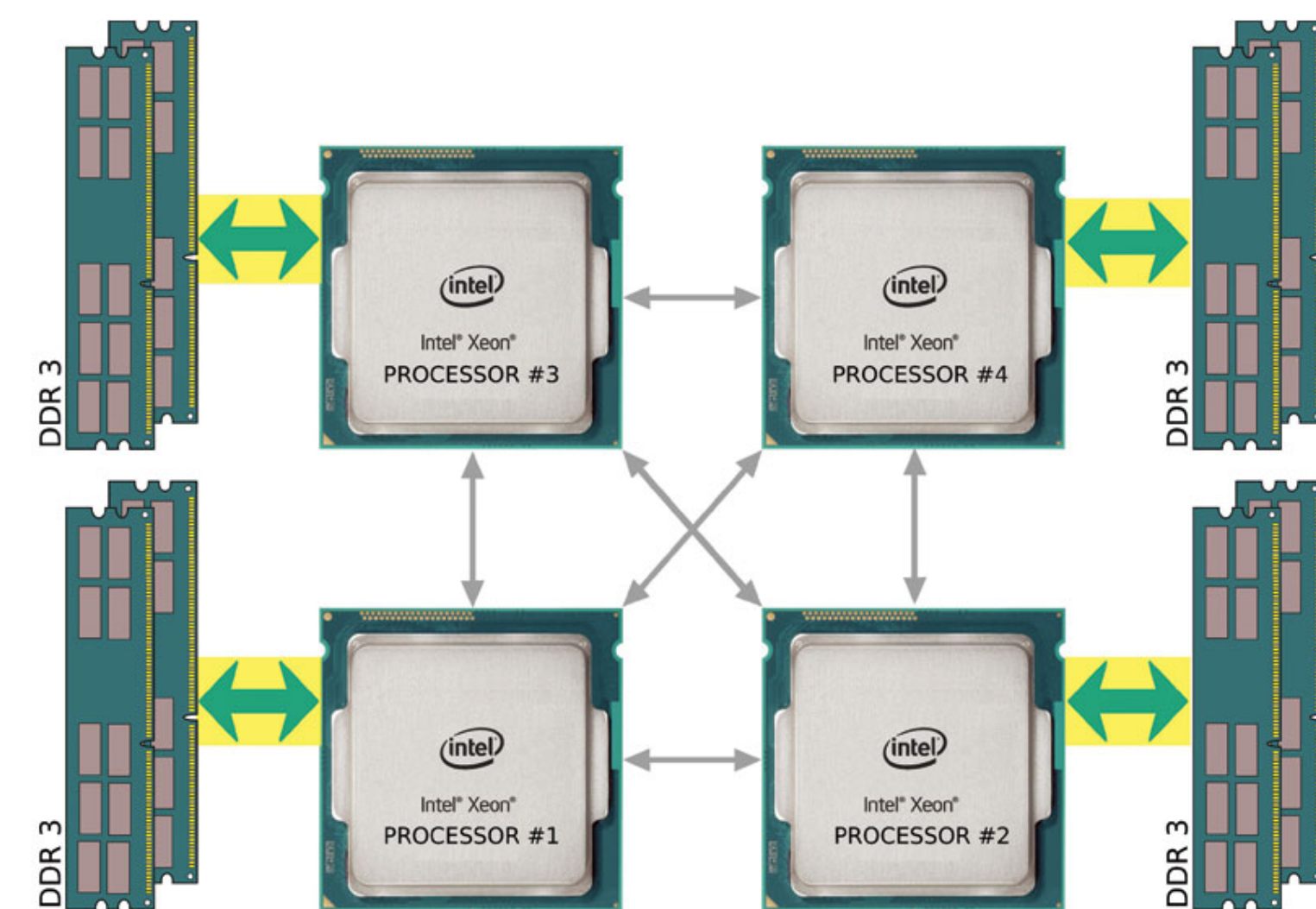
Machine Comparison



Execution Traces



NUMA Architecture



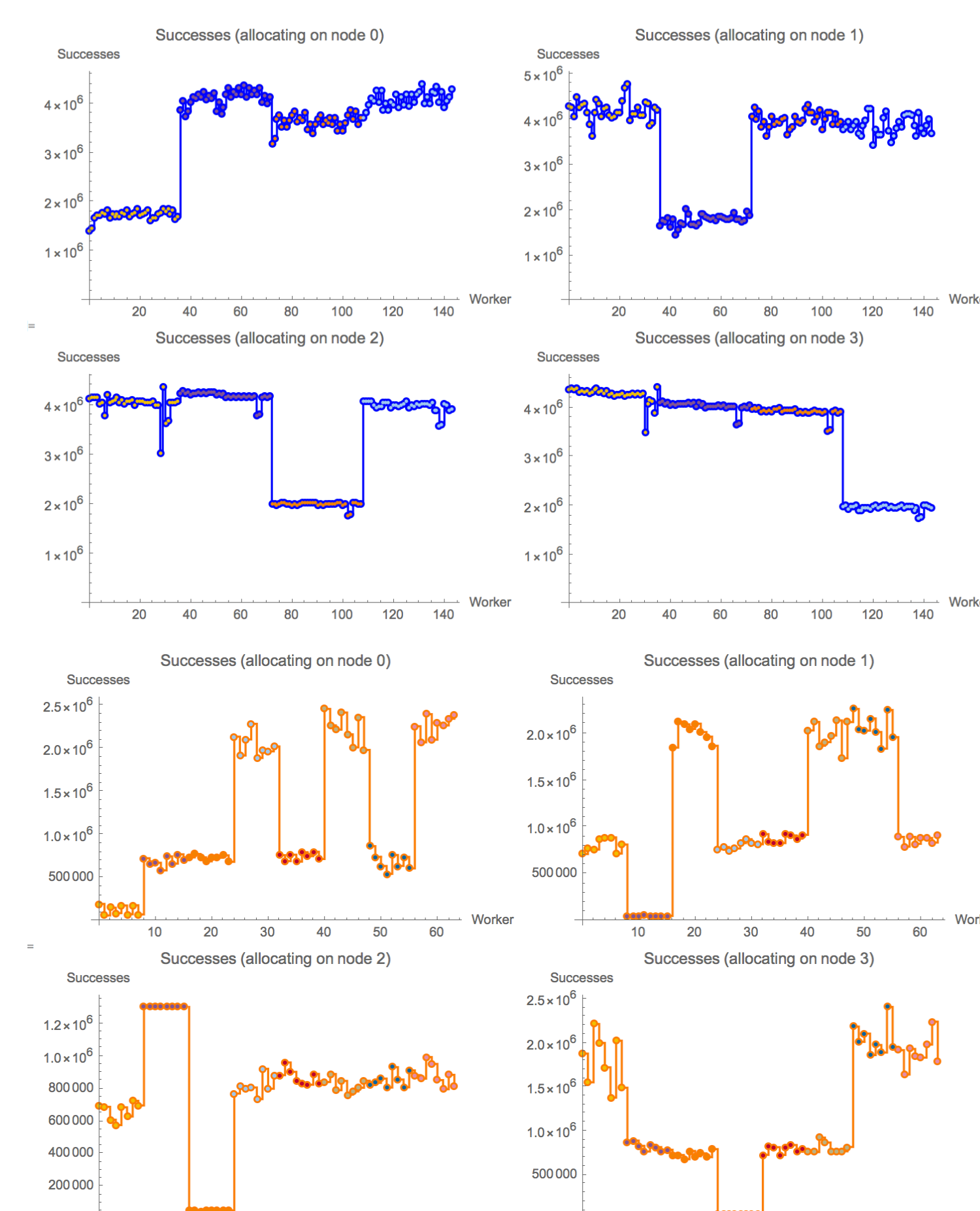
Intel Xeon Phi

- 4 nodes
- 18 cores per node
- 2 threads per core

AMD Bulldozer

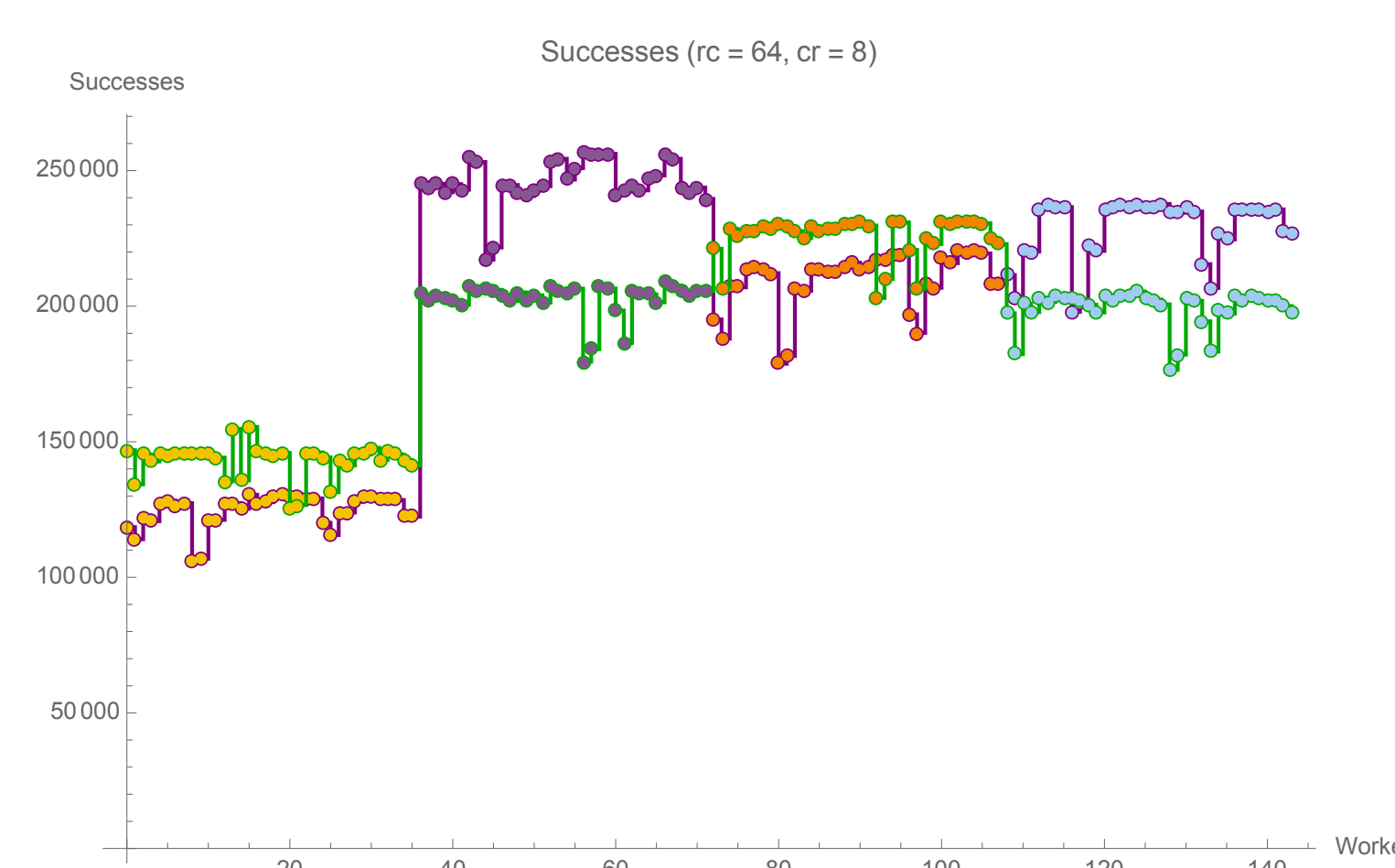
- 8 nodes (2 per socket)
- 4 cores per node
- 2 threads per core

Node Heterogeneity



Backoff Strategies

Purple is constant backoff
Green is “smart” backoff



Workers using “smart” backoff skip CAS attempt if another worker from their node had the most recent success

Just CAS

Purple reads before each CAS
Green reads only before first CAS

