Evolutionary Resilience of Protein Interaction Networks

Jesse Hautala, Shawn Houser, Angyalka Valcsics
Network Clique Enumeration
STA 596

November 28, 2021

We tried using Python *multiprocessing* to gather network statistics, using a three-part data pipeline:

- The **Producer** process iterates over all PPINs and passes each one into the first queue (the PPIN queue).
- n **Worker** processes claim items from the PPIN queue, calculates all the network statistics and writes them to next queue (the stats queue).
- Finally, the **Consumer** process reads from the stats queue, appends each element to an internal $DataFrame$ and sends the result to the final queue (the result queue).

Thie method allows us to make more efficient use of available processing resources and vastly improves time performance of CPU-bound processing. But when we tried implementing exhaustive enumeration of all clique counts (per clique size), we found this to be impractical.

The supporting algorithm (*networkx.enumerate_all_cliques*) is memory-bound, as confirmed via execution with 64GB of RAM and 3 $Worker$ processes; execution failed with a $MemoryError$ after ~ 8 hours of processing, when one of a workers (that was already holding $\sim$ 40GB of RAM) attempted to allocate additional memory beyond available capacity (see Figure 1).

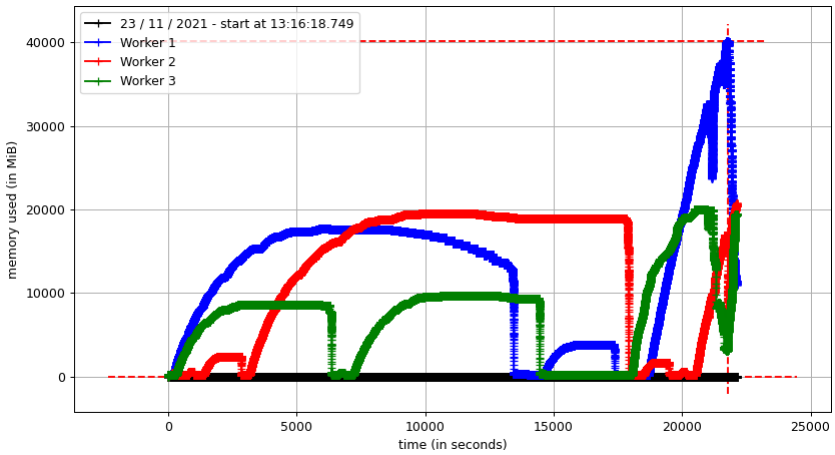

Figure 1: Process memory usage over time (until $MemoryError$)