Week 5 results

Tried different way to reduce the overhead and using mixed metrics to get the rank of each file.

Found a big bug in my code, upon fixing this bug, I found that the "accumulated access frequency" can predict the file access more accurately than any other combinations with very low overhead.

Although, finally I found a way to greatly reduce the overhead of graph based algorithm, but the comparison with use accumulated access frequency shows that the latter one has both better performance and less overhead.

Parameter settings:

Trace file: 120 days file access trace

Edge adding window: 10s

Edge expire window: 1 day

Update period: 1day

Total file count: 90345

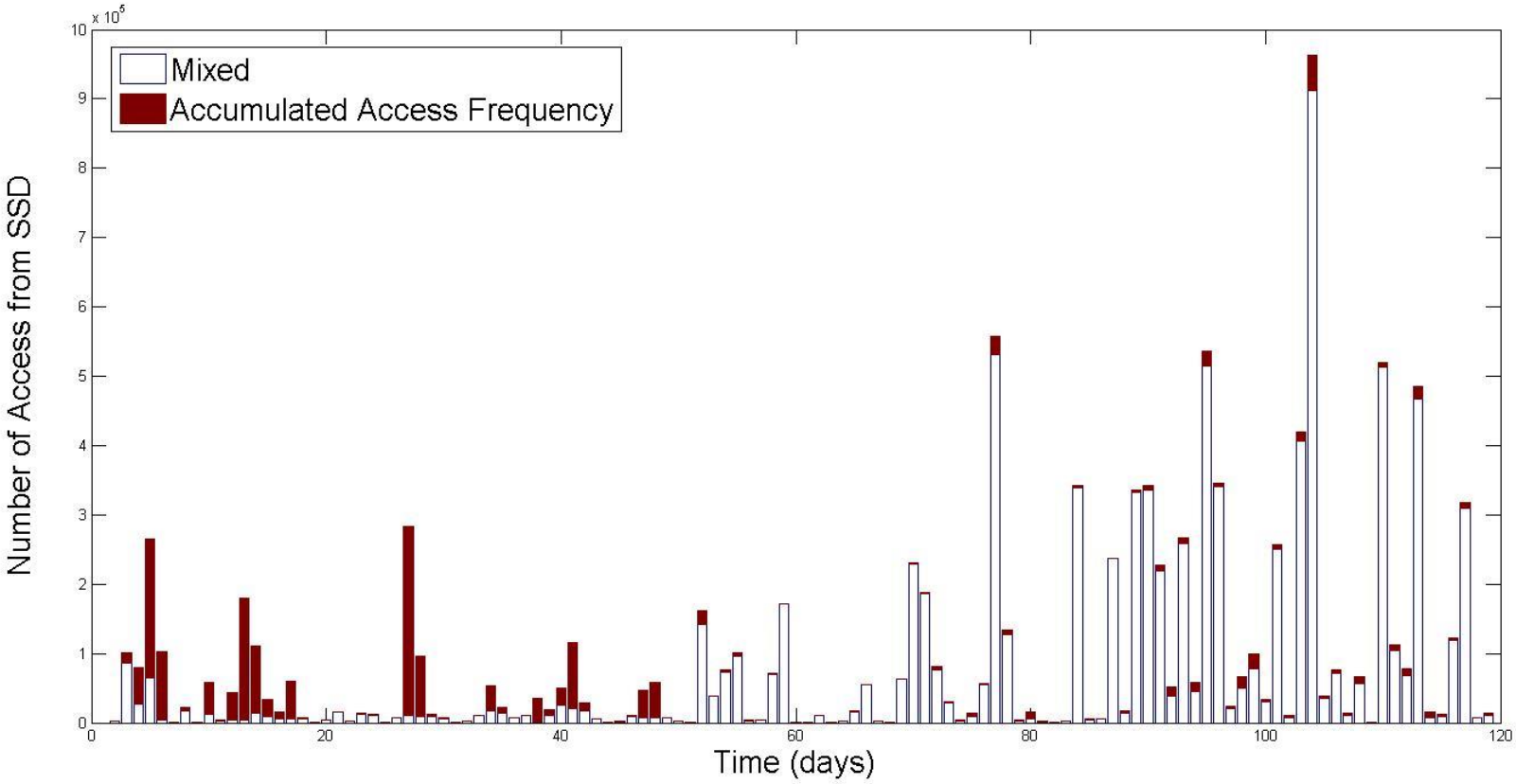
Using fixed ssd capacity ratio to control the data placement instead of using threshold.

Three metrics for placement:

1. Number of connections:
   number of connections with all neighbors, a connection means two files have been accessed together for once. An edge may contain several connections.
2. Ratio of connections from SSD (Main metric):
   number of connections with neighbors in SSD / number of connections with all neighbors
3. Access frequency:
   number of access, default as non-accumulated.

Results:

1. Performance comparison



2. Overhead comparison