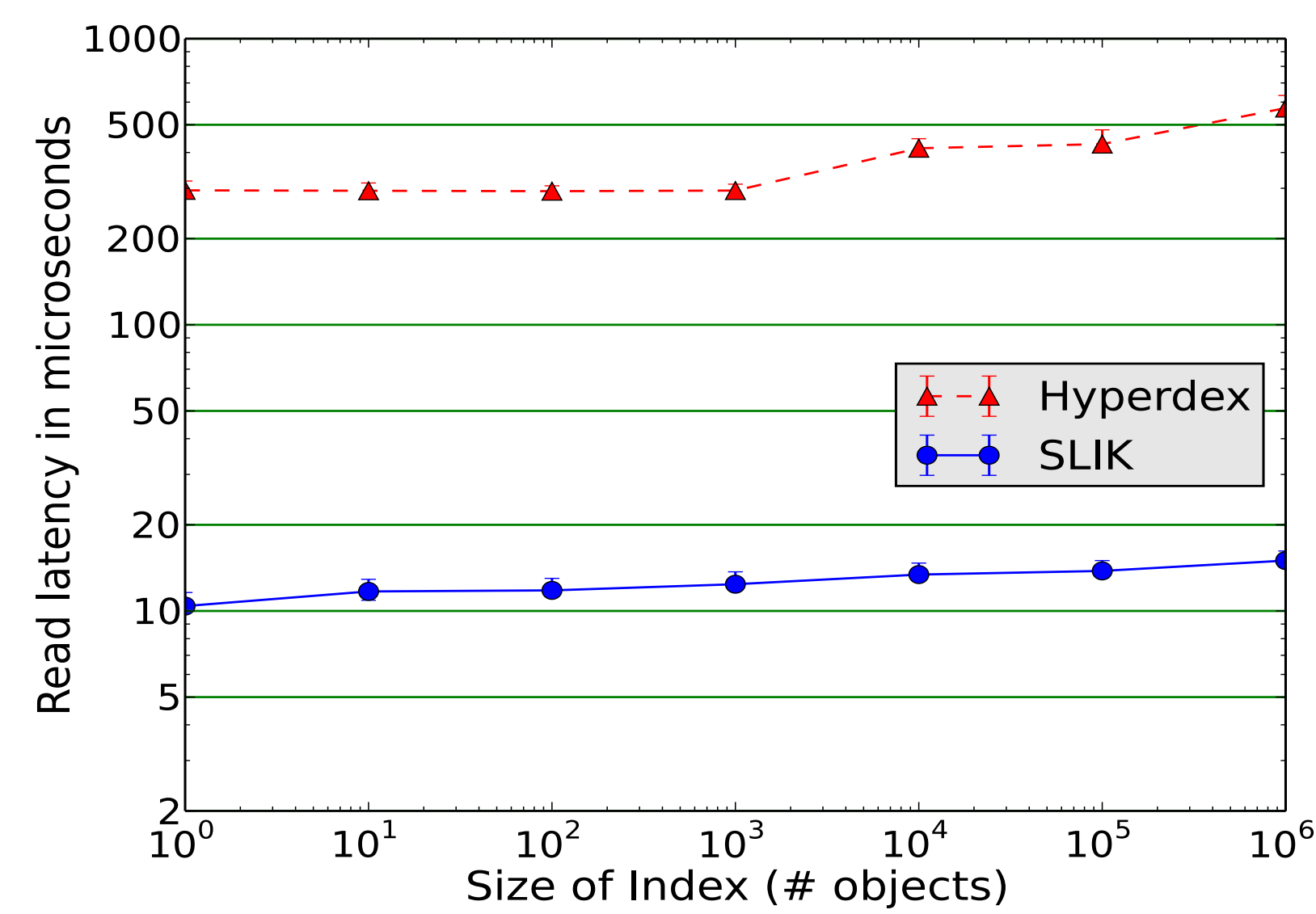


Performance Measurements For SLIK

Ankita Kejriwal, Arjun Gopalan, Ashish Gupta, Zhihao Jao, John Ousterhout

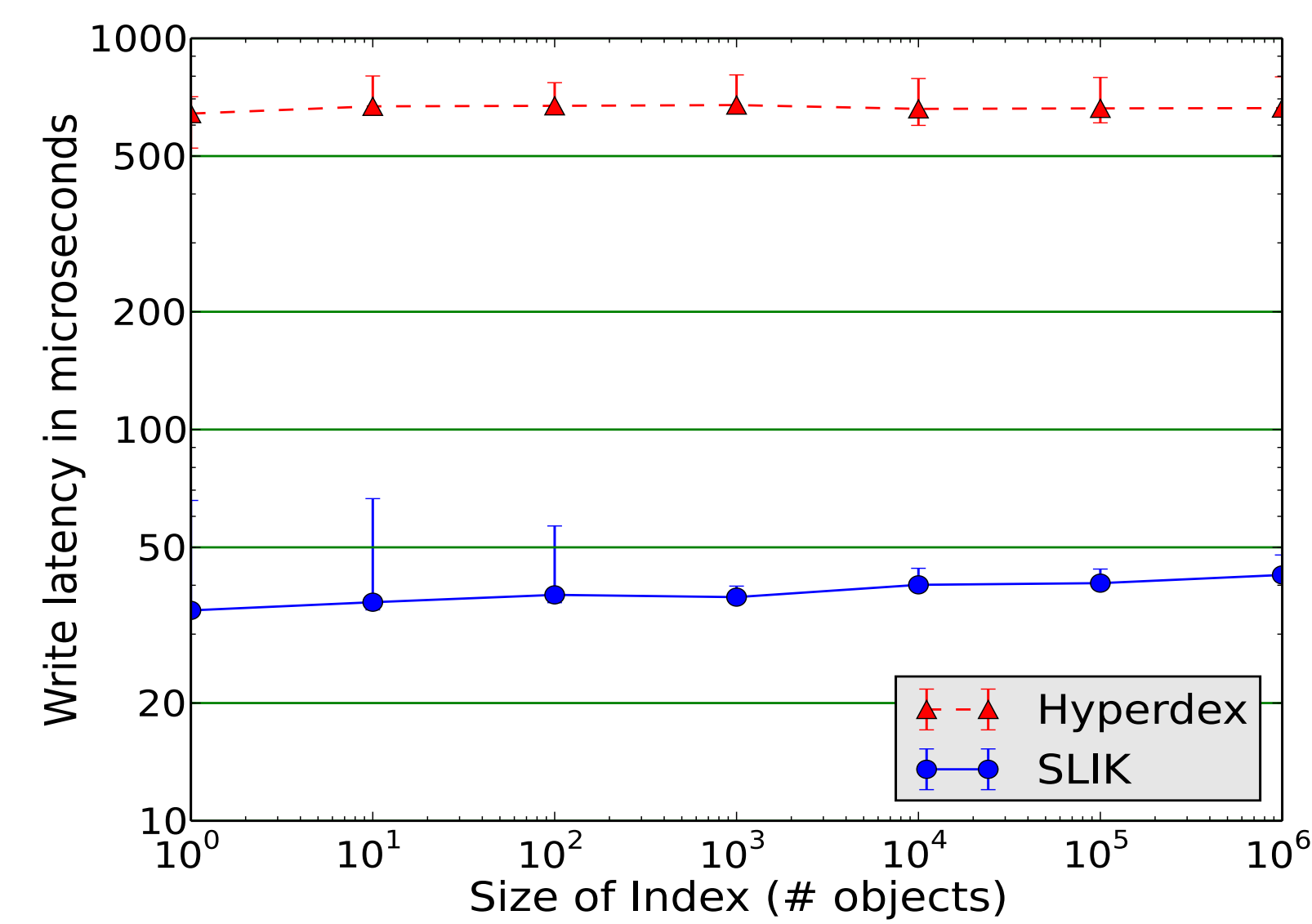
Indexed Read



Latency of basic operations as a function of index size: reading a single object using a secondary key.

Each data point displays the 10th percentile, median, and 90th percentile latencies across 1000 measurements. A single table was used, with one secondary index; all object values were 100 bytes and all index keys were 30-byte strings.

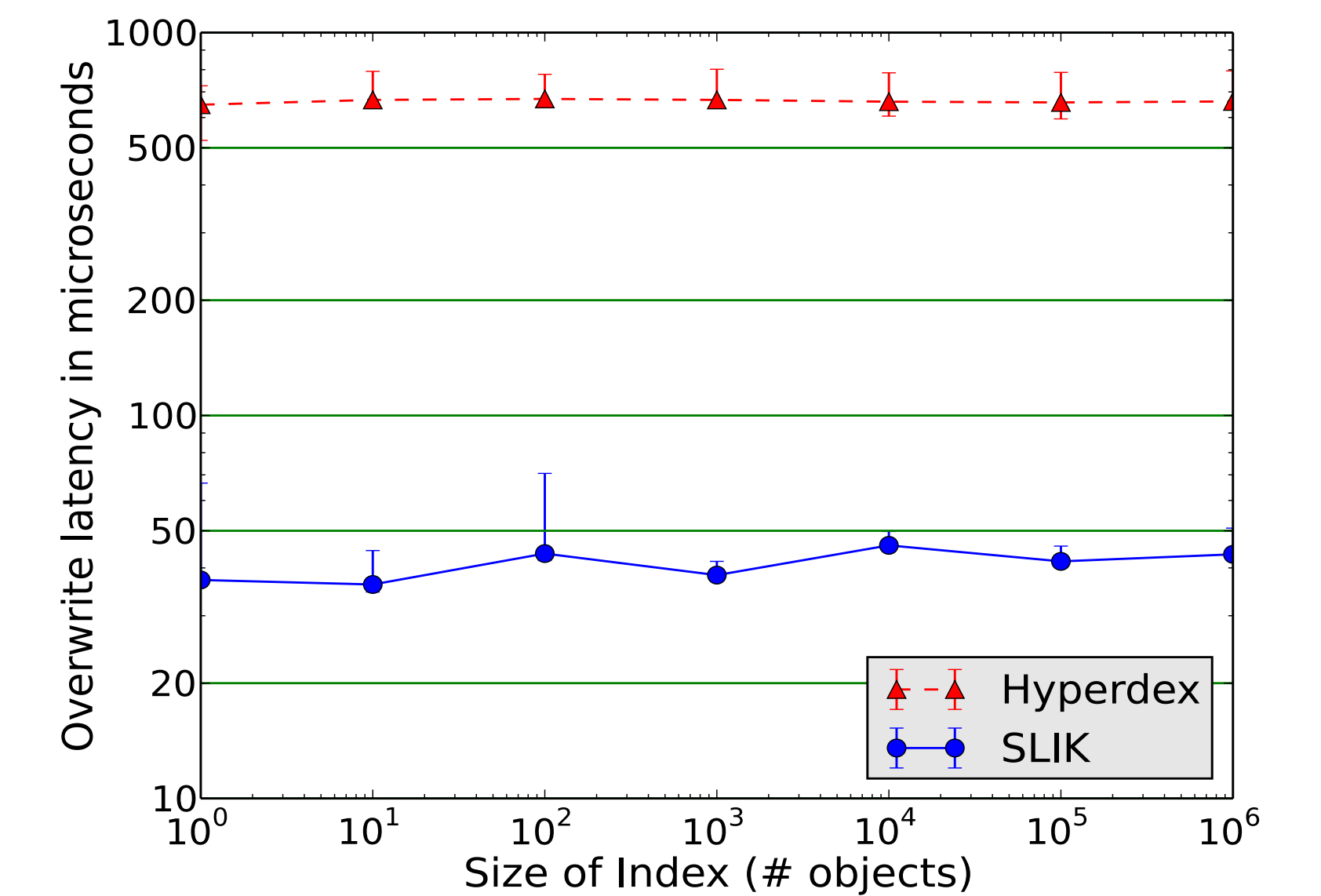
Index Write



Latency of basic operations as a function of index size: creating a new object (no old index entries need to be removed).

Each data point displays the 10th percentile, median, and 90th percentile latencies across 1000 measurements. A single table was used, with one secondary index; all object values were 100 bytes and all index keys were 30-byte strings.

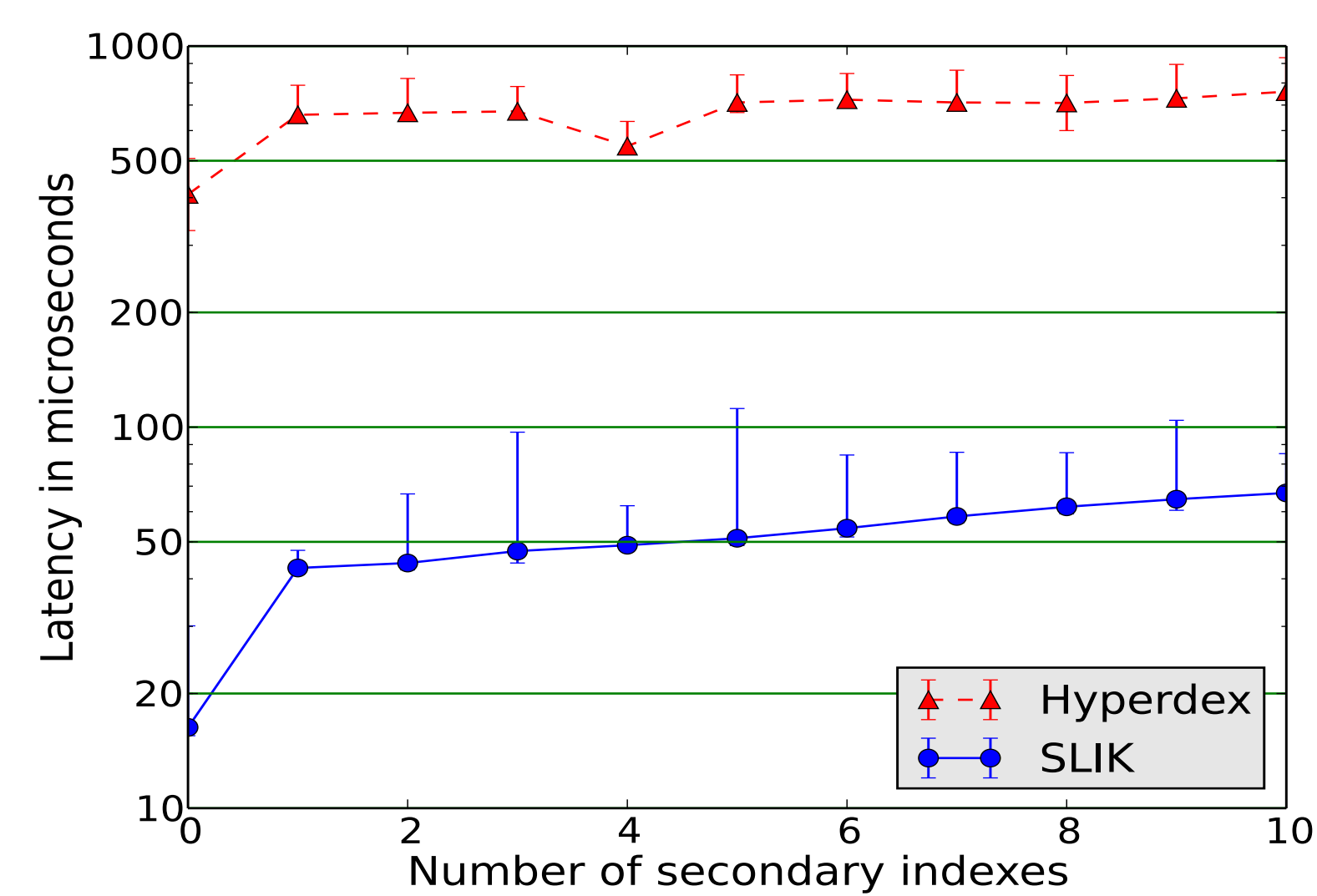
Index Overwrite



Latency of basic operations as a function of index size: overwriting an existing object (new index entries must be created, old ones removed).

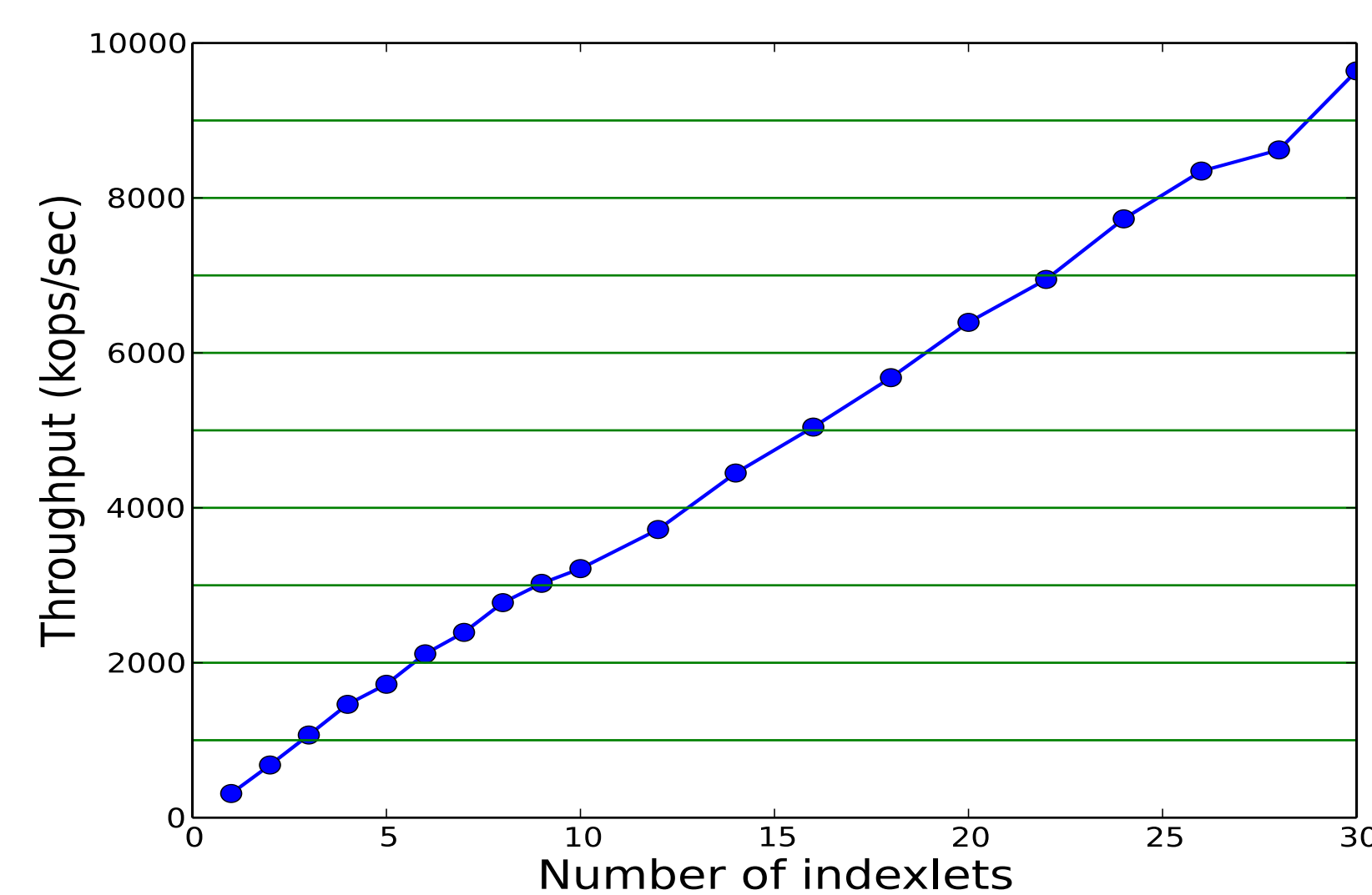
Each data point displays the 10th percentile, median, and 90th percentile latencies across 1000 measurements. A single table was used, with one secondary index; all object values were 100 bytes and all index keys were 30-byte strings.

Multi-index Overwrite



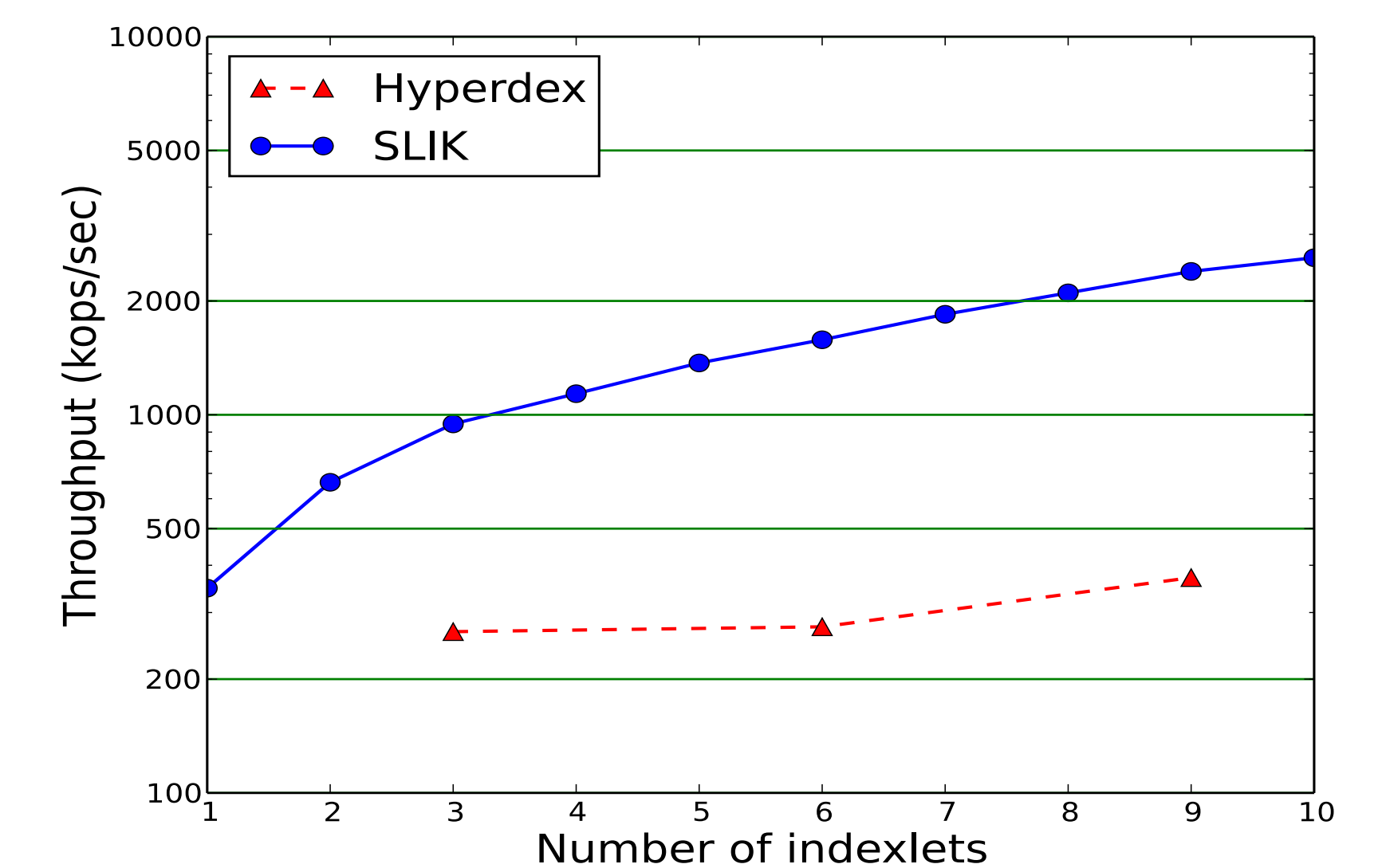
Latency of overwrites as a function of the number of secondary indexes. A single table was used, with 100-byte object values and 30-byte string keys. Each secondary index was located on a different server. Each datapoint displays the 10th percentile, median, and 90th percentile latencies across 1000 measurements.

Index Lookup Scalability



Total lookup throughput of a single index divided into multiple indexlets on different servers. For each data point the number of clients varied from 1-30 to find the maximum total throughput across all configurations. Each client issued multiple concurrent readHashes requests, using randomly chosen keys uniformly distributed across the indexlets. Each request returned a single matching key hash.

Index Read Scalability



Total readRange throughput of a single index divided into multiple indexlets on different servers. For data point the number of clients varied from 1-30 to find the maximum total throughput across all configurations. Each client issued 20 concurrent readRange requests, using randomly chosen keys uniformly distributed across the indexlets. Each request returned a single matching object. The objects of the table were split into multiple tablets stored on different servers; the number of tablets in each experiment was the same as the number of indexlets.