Sensitivity Analysis

Summary

A sensitivity analysis with 3 datasets and 3 applications has been performed to have an Increased understanding of the relationships between input and output variables in a system. Input variables are the following:

Graph Size
Association
Prefetcher
Replacement policy
Number of write buffers

Output variable is the miss rate of LLC.

Datasets

Slashdot Dataset

Slashdot is a technology-related news website know for its specific user community. The website features user-submitted and editor-evaluated current primarily technology oriented news.

Number of Nodes	77360
Number of Edges	905468
Diameter (Longest Shortest Path)	10
90-percentile effective diameter	4.7

Epinions Dataset

This is a who-trust-whom online social network of a general consumer review site Epinions.com. Members of the site can decide whether to "trust" each other. All the trust relationships interact and form the Web of Trust.

Number of Nodes	75888
Number of Edges	508837
Diameter (Longest Shortest Path)	14
90-percentile effective diameter	5

Facebook Dataset

This dataset represents blue verified Facebook page network of artists. Nodes represent the pages and edges are mutual likes among them.

Number of Nodes	50515
Number of Edges	819090

Applications

Breadth-First Search (BFS)

This BFS implementation makes use of the Direction-Optimizing approach [1].

Connected Components (CC)

This CC implementation makes use of the Afforest subgraph sampling algorithm [2].

Connected Components (CC_SV)

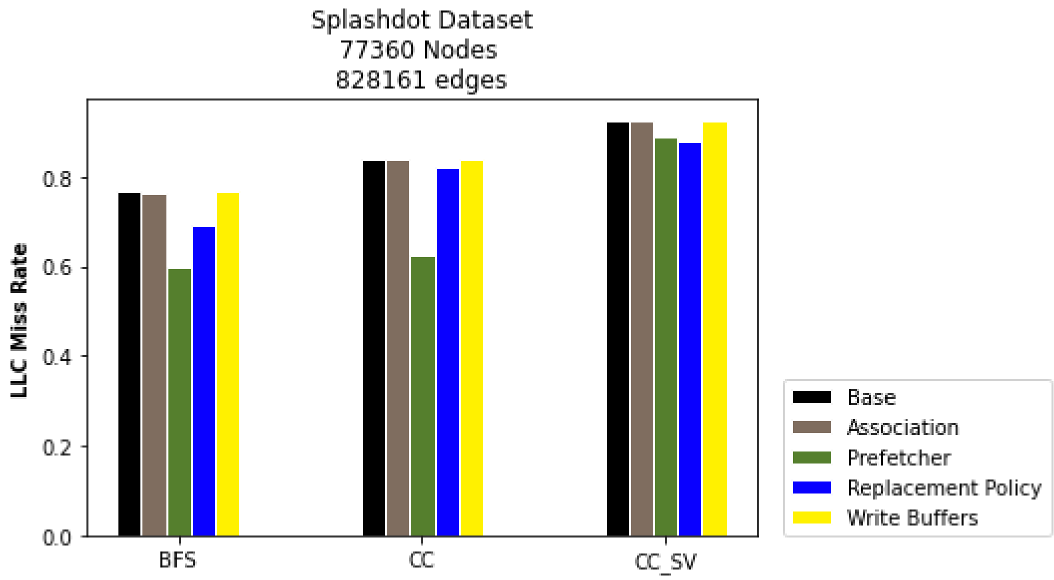
This CC implementation makes use of the Shiloach-Vishkin [3] algorithm with implementation optimizations from Bader et al. [4].

Page Rank (PR)

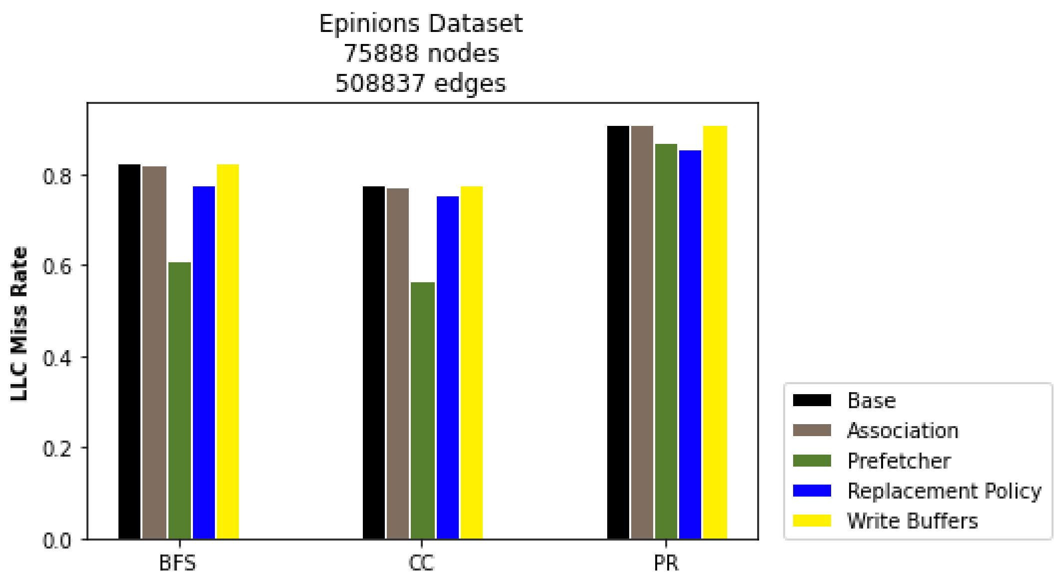
This PR implementation uses the traditional iterative approach.

Results

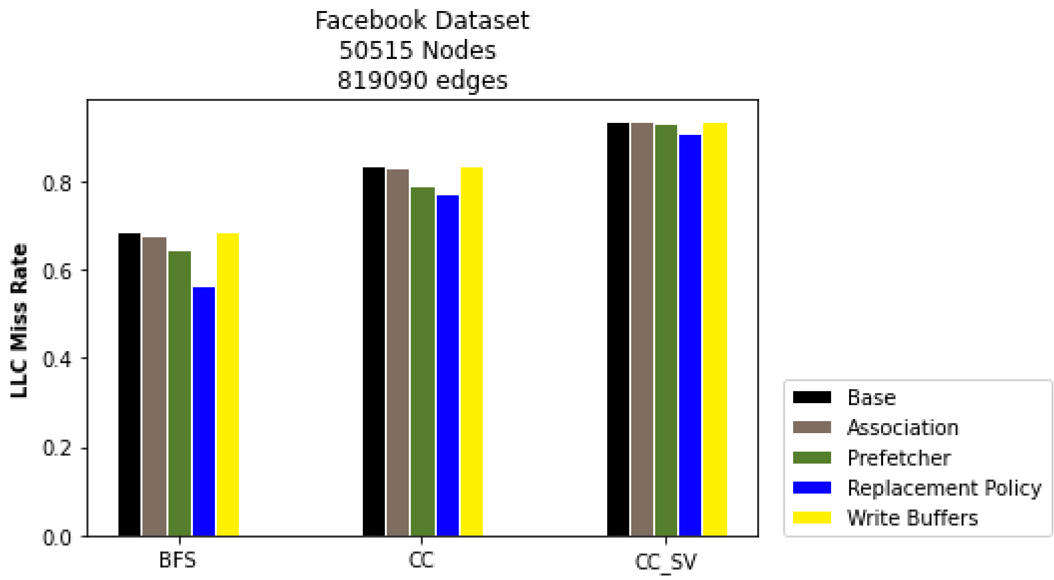
Best of Each Variable



	Base	Association	Prefetcher	Replacement Policy	Write Buffer
BFS	0.769096	0.76232	0.597223	0.690488	0.769096
CC	0.840078	0.838243	0.627644	0.821157	0.840078
CC_SV	0.926038	0.92322	0.888799	0.882393	0.926038

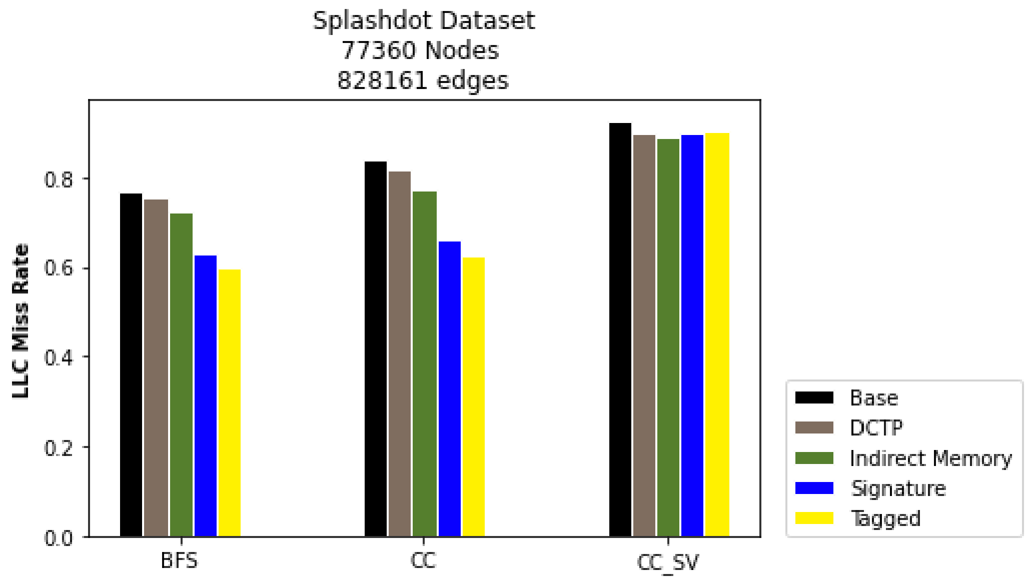


	Base	Association	Prefetcher	Replacement Policy	Write Buffer
BFS	0.823534	0.819624	0.607519	0.776174	0.823534
CC	0.775584	0.771223	0.564125	0.752731	0.775584
PR	0.909676	0.908157	0.869523	0.85679	0.909555

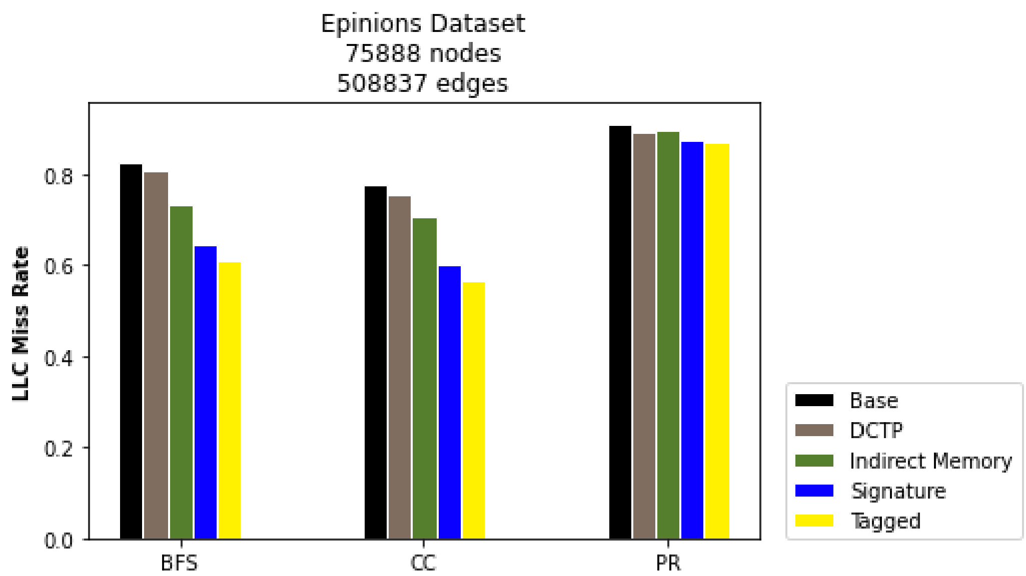


	Base	Association	Prefetcher	Replacement Policy	Write Buffer
BFS	0.687626	0.678413	0.644682	0.563701	0.687614
CC	0.835823	0.831738	0.792155	0.774904	0.835823
CC_SV	0.936802	0.935781	0.929584	0.907152	0.936802

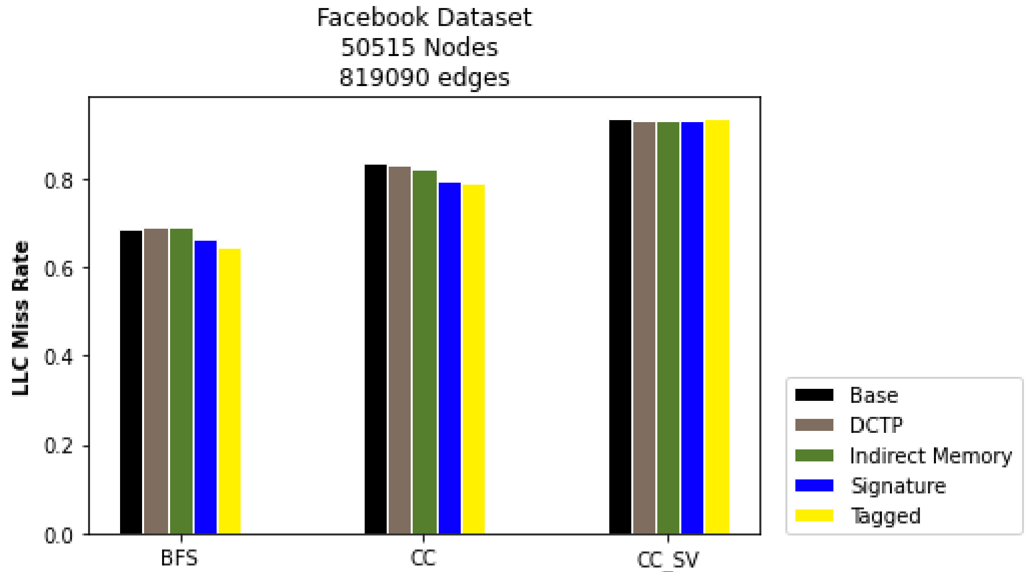
Prefetching



	Base	DCTP	Indirect Memory	Signature	Tagged
BFS	0.769096	0.755823	0.723735	0.631361	0.597223
CC	0.840078	0.817721	0.771612	0.660387	0.627644
CC_SV	0.926038	0.89721	0.888799	0.899363	0.90207

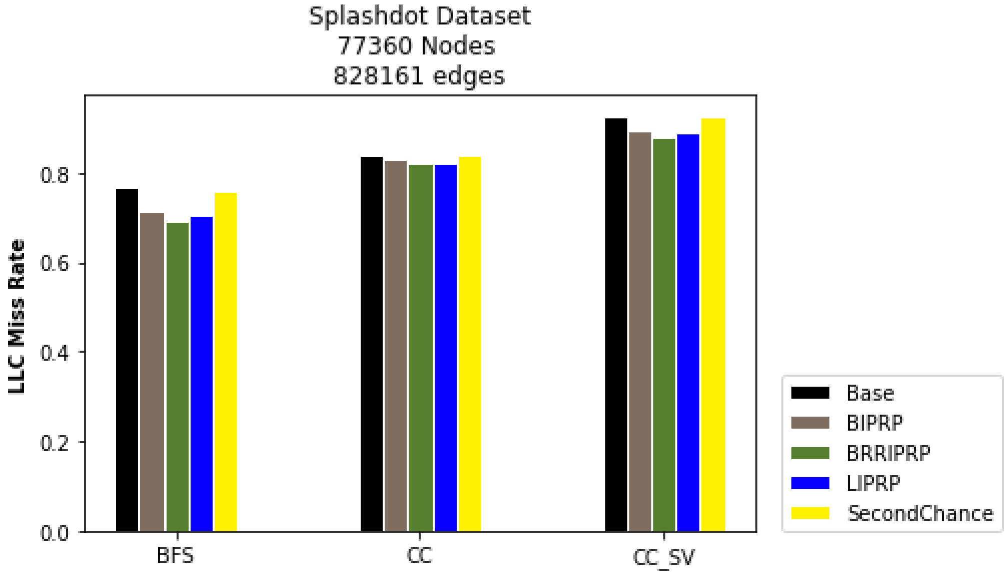


	Base	DCTP	Indirect Memory	Signature	Tagged
BFS	0.823534	0.807999	0.732892	0.64482	0.607519
CC	0.775584	0.753652	0.706054	0.602909	0.564125
PR	0.909676	0.89275	0.896745	0.874238	0.869523

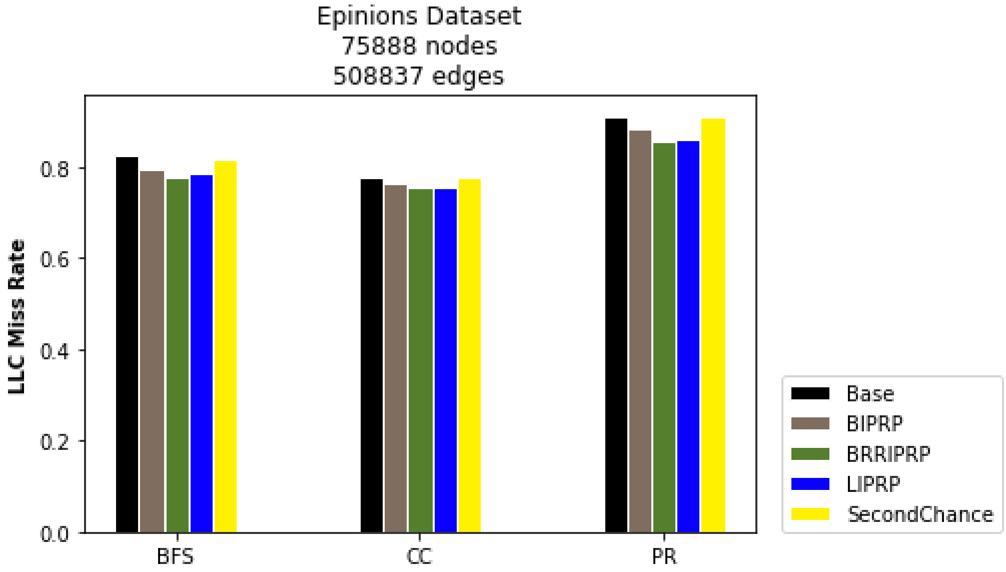


	Base	DCTP	Indirect Memory	Signature	Tagged
BFS	0.687626	0.69047	0.691413	0.665276	0.644682
CC	0.835823	0.833573	0.823552	0.795309	0.792155
CC_SV	0.936802	0.931284	0.929584	0.931847	0.934798

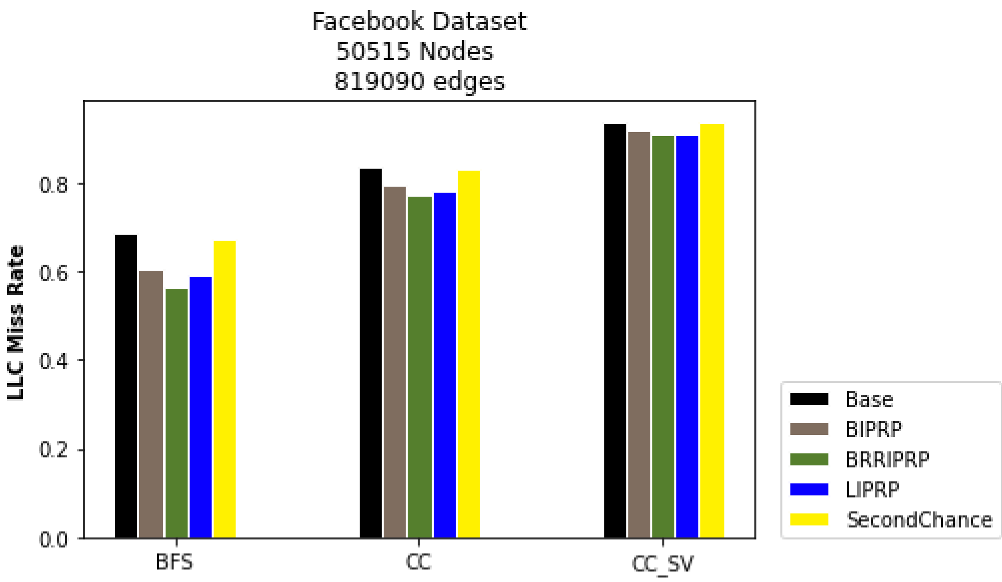
Replacement Policy



	Base	BIPRP	BRRIPRP	LIPRP	SecondChance
BFS	0.769096	0.716398	0.690488	0.705924	0.760808
CC	0.840078	0.830369	0.821157	0.821737	0.83927
CC_SV	0.926038	0.894706	0.882393	0.887863	0.924479



	Base	BIPRP	BRRIPRP	LIPRP	SecondChance
BFS	0.823534	0.795519	0.776174	0.785686	0.81814
CC	0.775584	0.763022	0.752731	0.755014	0.77602
PR	0.909676	0.880093	0.85679	0.858044	0.908522



	Base	**BIPRP**	**BRRIPRP**	**LIPRP**	**SecondChance**
BFS	0.687626	0.606441	0.563701	0.592257	0.674325
CC	0.835823	0.795997	0.774904	0.782499	0.832071
PR	0.936802	0.916996	0.907152	0.908647	0.935764

Conclusions

- Changing number of MSHRs, number of write buffers or associativity of the LLC does not change the LLC miss rate significantly.
- Using a prefetcher for LLC decreases LLC miss rate.
- Using a more suitable replacement policy (instead of default LRU policy) decreases LLC miss rate.

References

1. Scott Beamer, Krste Asanović, and David Patterson. "Direction-Optimizing Breadth-First Search." International Conference on High Performance Computing, Networking, Storage and Analysis (SC), Salt Lake City, Utah, November 2012.
2. Michael Sutton, Tal Ben-Nun, and Amnon Barak. "Optimizing Parallel Graph Connectivity Computation via Subgraph Sampling" Symposium on Parallel and Distributed Processing, IPDPS 2018.
3. Yossi Shiloach and Uzi Vishkin. "An o(logn) parallel connectivity algorithm" Journal of Algorithms, 3(1):57–67, 1982.
4. David A Bader, Guojing Cong, and John Feo. "On the architectural requirements for efficient execution of graph algorithms." International Conference on Parallel Processing, Jul 2005.