

## Implementation of Databases ( Homework 3 )

kartik vishwakarma  
2017csm1001

In Emp. x Dept. loop join where Dept. as inner table, Initially MRU having very higher Miss compare to LRU.

Reason behind may be since Dept. table has 100 blocks, in **MRU it keep B-1 records remain same, (where B buffer size) while throw away each new items in last buffers.**

Nature of MRU cause to much miss.

In Databases, for large dataset, MRU, suppose to perform better than LRU with buffer size lesser than data.

*Red line: MRU*

*Blue line: LRU*

For LRU hot point: 102 buffer (size of inner table +2 )

For MRU hot point: 1101 buffer (size of inner table + size of outer table + 1)

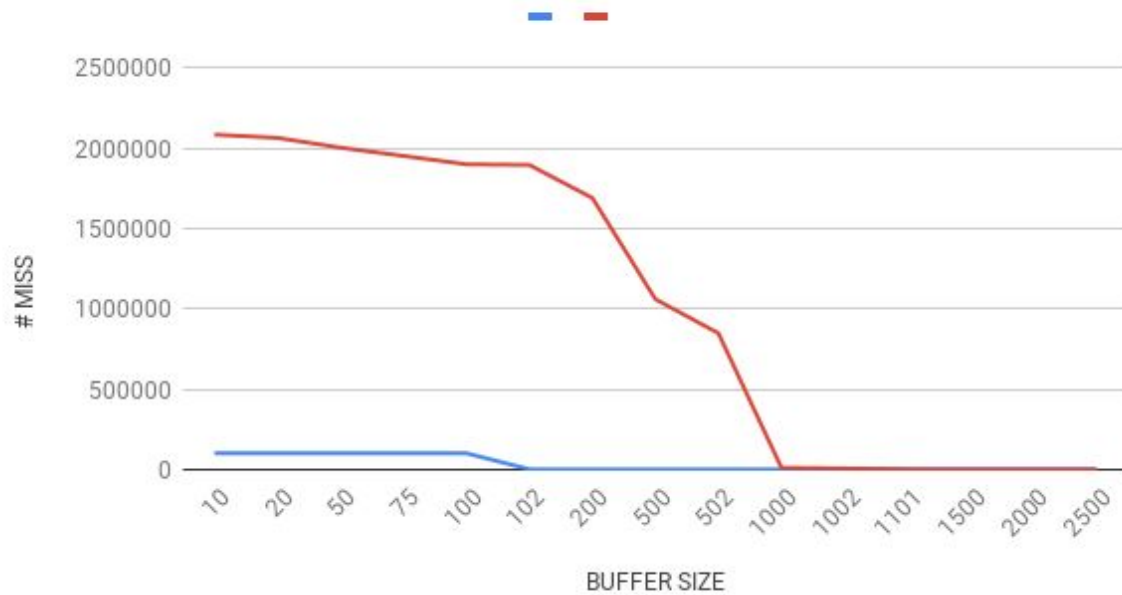
In Databases, for large dataset, MRU, suppose to perform better than LRU with buffer size lesser than data.

With increase in buffer size, MRU #miss decrease accordingly but for LRU, it does not affects #miss in LRU #miss suddenly decreases at "hot point" only.

After hot point, #hit remain same both LRU and MRU, it does not affected by providing more buffer .

In both LRU and MRU , #miss proportional to size of outer table, higher size of outer table more #miss,

# MISS in Emp. x Dept.



Similar to above observation, In Dept. x Emp. nested join with Emp as inner table, having of block size 1000, MRU having higher page fault compare to LRU, but less than Emp. x Dept. nested join.

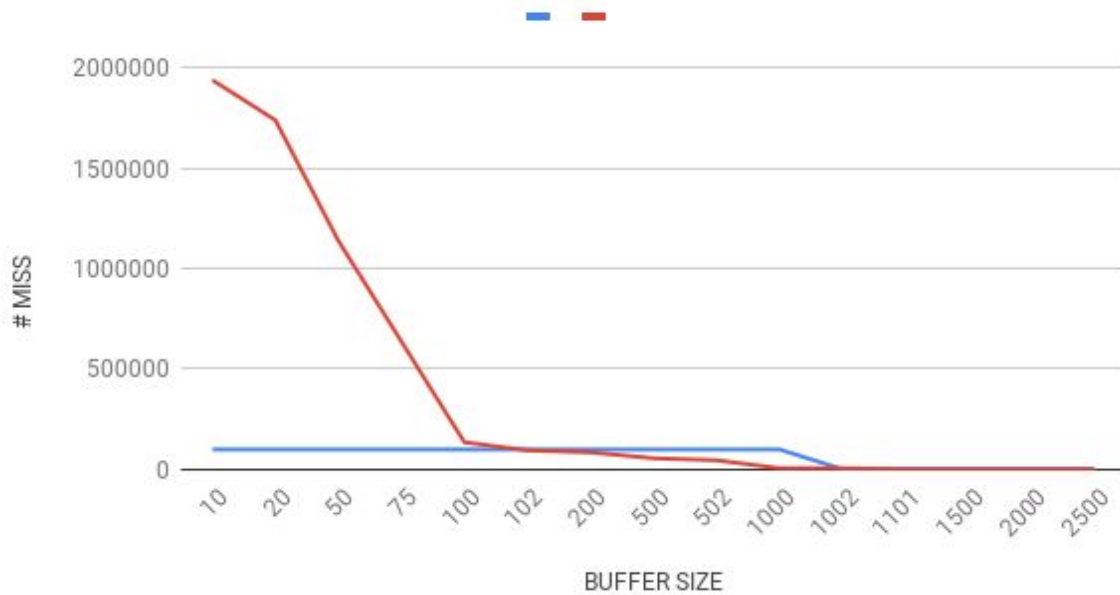
While in LRU, it #miss remain same irrespective of size of inner table.

Here also, MRU #miss decreases with increase in buffer size, until hot point.

For LRU : with Emp. as inner table hot point will be 1002

For MRU: hot point remain 1101.

# MISS in Dept. x Emp.



In Proj. x Dept. join relation, with Dept as inner join Since size of outer table is less compare to Emp table, #miss is less in both LRU and MRU.

LRU: hot point is size of inner table + 2 i.e.  $(100 + 2) = 102$

MRU: hot point is size of outer table + inner table + 1, i.e.  $(500 + 100 + 1) = 601$

# MISS in Proj. x Dept.



Following trend of MRU and LRU describe above,

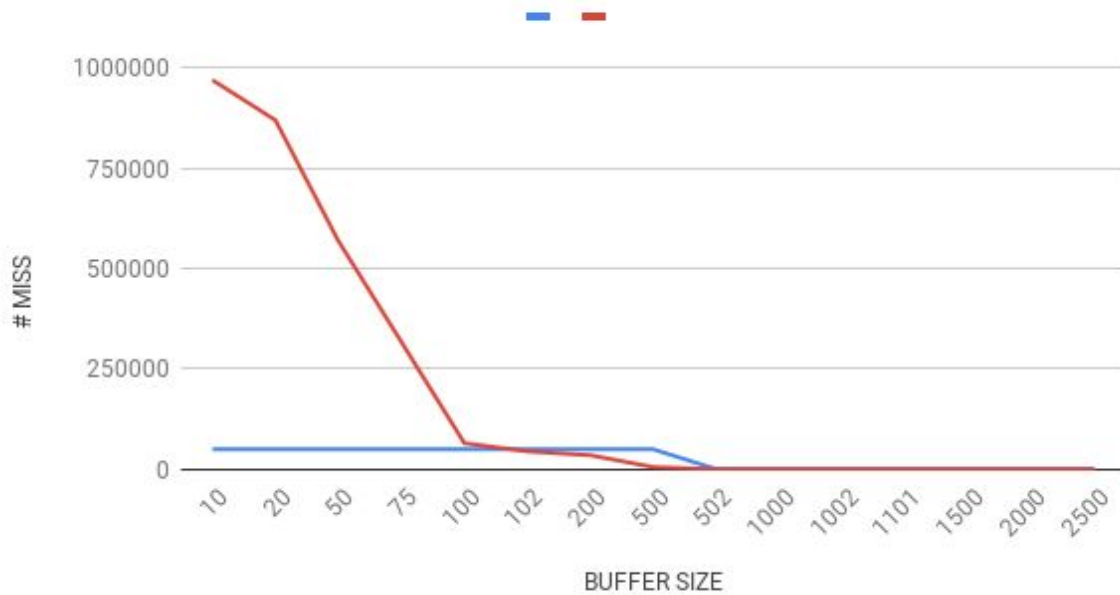
For nested join between Dept. x Proj. with Proj. as inner of block size 500 while outer table of block size 100, both LRU and MRU have less #miss compare to Proj. x Dept.

But MRU have much higher #miss as observed other nested join operation.

LRU: hot point size of inner table +2, i.e.  $(500 + 2) = 502$

MRU: hot point remain same, i.e. size of inner + outer table + 1, i.e.  $(100 + 500 + 1) = 601$

# MISS in Dept. x Proj.



Conclusion:

- In case of #block greater than #available buffer, in all case LRU perform better than MRU.
- With increase in buffer size, less than hot point, for MRU, #miss decreases linear, while in LRU it does not affect #miss.
- #miss depend upon size of outer table, higher size of outer table high number of miss will be.
- To avoid thrashing in MRU, we maintain some lookup on block with being thrown out, if a block came very recently, we avoid such block to go out of buffer.



