ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

# Correlation Analysis Between Page Faults And Free Frames

[1] C.Mallikarjuna, [2]DR.P.Chitti Babu
[1]Asst. Professor, Dept. of MCA, APGCCS, [2] Professor and Principal, APGCCS

**Abstract: Now a day's operating system manages the memory through Memory management schemes. Paging is one among them. If process required pages are not currently available in main memory, then through paging scheme, the necessary pages are copied from the secondary storage to main memory through page replacement. This phenomena is called as page fault. If process required pages are not available in main memory due to lack of free frames, while making execution over processor, automatically many page faults are occur. As a consequence the process is executed over the processor for a lengthy period. If sufficient number of free frames are available in main memory, then page faults will never occur. So here, in this paper we study about the correlation between Page faults and free frames.**

**Index Terms- Correlation; Free frames Page fault; Page replacement;**

## I. INTRODUCTION

Earlier unequal fixed size and variable size partition memory management schemes are inefficient to manage memory effectively. Because unequal fixed size partition results in internal fragmentation, whereas variable size partition memory management scheme results in external fragmentation. However, that main memory is partitioned into equal fixed size chunks of the same size. Then the chunks of the process is also divided into small fixed size of the chunks of the same size. Then the chunks of the process known as pages could be assigned to available chunks of the main memory, is known as frames. At a given point in time, some of the frames in main memory are in use and some are free.

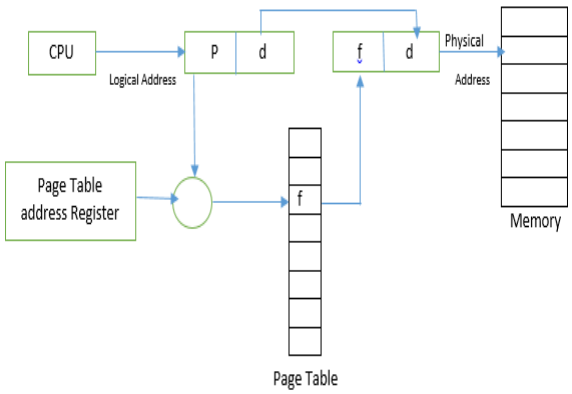


***Fig 1: Paging Memory Management Scheme***

A list of free frames is maintained by operating system for further usage. However the blocks which allocate by operating system need not be contiguous, they can be anywhere in main memory because OS asks the MMU to dynamically reallocate each block separately [12]. This technique is known as paging [1]. This is illustrated in above figure.

## II. PROBLEM DESCRIPTION

The main idea of the paging memory management scheme is, that when a process is swapped in, the pager only loads into memory those pages that it expects the process to need Pages that are not loaded into memory are marked as invalid in the page table, using the invalid bit. If the pages which are ever required by process, those willonly loaded into memory. If all the pages were loaded in to memory then the process would runsuccessfully without interrupting any page faults. On the other side, if a page is required and that was not originally loaded up, then a page fault is encounter.The page fault is serviced with the help of followingprocedure [6], [8], [9].

•The memory address requested is first checked, to make sure it was a valid memory request.
•If the reference was invalid, the process is terminated. Otherwise, the page must be paged in.
•From a list of free frames, one of the frame address is identified.
•An I/O operation is initiated to bring in the required page from the disk. When the I/O operation is complete, the process's page table is updated with the new frame number, and the invalid bit is changed to valid, to indicate that this is now a valid page reference.
•The instruction which caused the page fault must now be restarted from the same instruction to complete remaining execution.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

## III. LITERATURE SURVEY

The page replacement will be occurred, when a page needs to be brought into main memory and no free frame is exists [3]. A currently used page in main memory must be selected as a victim page to replace with the required page. There are number of approaches for selecting the victim page from the main memory [5]. When a page is brought out into main memory for the first time, a time stamp is recorded for the page. When a page is required to bring into main memory, a page table will be searched for selecting the victim page. For instance if operating system use FIFO Page replacement approach, to select the victim page as the oldest page in the page table will be selected to replace required page in main memory. When a page is frequently needed as soon as it is loaded in main memory, it will be assigned with latest time stamp value.

The page frame data table is used for page replacement [8]. Several pointers are used to create lists within this table. All of the available frames are linked together in a list of free frames available for bringing in pages. When the number of free frames drops below certain threshold, the kernel will steal a number of frames to compensate.

When a page fault occurs, the operating system has to select a victim page to remove from primary memory to create some free space for the page that has to be brought in[8][9]. The page replacement is done by swapping the required pages from backup storage to main memory and vice-versa. The dirty pages will be rewritten to the disk to maintain data consistency. If, however, the page has not been changed the disk copy is already up to date, so no rewrite is needed. The performance of the page replacement algorithms depend upon number of page faults are generated while a process is executed over the processor. Operating system uses different page replacement algorithms [10]. While making selection it will give the priority to the algorithm, with least page fault rate is considered. Optimal page replacement algorithm generates minimum page faults.

## IV. EXPERIMENTAL EVOLUATION

The principle of the optimal page algorithm is that the page with the highest label should be removed. If one page will not be used for 8 million instructions and another page will not be used for 6 million instructions, removing the former pushes the page fault that will fetch it back as far into the future as possible[11],[12]. Computers, like people, try to put off unpleasant events for as long as they can.The only problem with this algorithm is that it is unrealizable. At the time of the page fault, it is very difficult to forecast when each of the pages will referenced next. Still, by running a program on a simulator and keeping track of all page references, it is possible to implement optimal page replacement on the second time by using the page reference information collected during the first time.A mathematical method for measuring the intensity or magnitude of linear relationship between two variable series was suggested by Karl Pearson who is great British Bio-metrician and statistician and is by far the most widely used method in practice [14]. Karl Pearson's measure, known as Pearson Correlation co efficient between two variables X and Y, usually denoted by r(X, Y) or rxy or simply r, which is a numerical measure of linear relationship between them and is define as the ratio of the co variance between X and Y, written as Cov(X, Y) to the product of the standard deviation of X and Y. this is symbolically represented as R=COV(X, Y)/ x. y [14]. This is also written as follows.

$$r = \frac{\sum dx\, dy}{\sqrt{dx^2 dy^2}}$$

Where dx anddy denote the deviations of x and y values from their arithmetic means of X and Y respectively.
Interpretation of r Value
The r value is interpreted as follows.
•If r=1 then, there exist perfect positive correlation between the variables.
•If r=-1 then, there exist a perfect negative Correlation between the variables.
•If r=0 then, there exist no relationship between the two variables.
•If r is between 0.75 and 1 there exist high positive correlation between the two variables.
•If r is between -0.75 and -1 there exist high negative correlation between the two variables.
•If r is less than 0.5 then, there exists low positive correlation between the two variables.
•If r is less than -0.5 then, there exist low negative correlation between the two variables.
•If r is between 0.5 and 0.75, there exist moderate positive correlation between the two variables.

**Assumptions**
There is linear relationship between the two variables. It means that if we plotted the values of both variables on a scatter diagram, plotted points will forms a straight line.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

• There is a cause and effect relationship between the forces affecting the distribution of the items of the both the series. Correlation is meaningless if there is no such relationship.

## V. RESULTS AND DISCUSSION

Consider the following reference string and initially there are 3 free frames in main memory. Now we illustrate the optimal page replacement algorithm in the following table by assuming the reference string as "7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1" and the number of free frames in main memory are "3".

| Ref.string | 7 | 0 | 1 | 2 | 0 | 3 | 0 | 4 | 2 | 3 | 0 | 3 | 2 | 1 | 2 | 0 | 1 | 7 | 0 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 7 | 7 | 7 | 2 | | 2 | | 2 | | | 2 | | | 2 | | | 7 | | | |
| | | 0 | 0 | 0 | | 0 | | 4 | | | 0 | | | 0 | | | 0 | | | |
| | | | 1 | 1 | | 3 | | 3 | | | 3 | | | 1 | | | 1 | | | |
| Page faults | 1 | 2 | 3 | 4 | | 5 | | 6 | | | 7 | | | 8 | | | 9 | | | |

*Fig 2: Optimal Page Replace Algorithm.*

In this case the optimal algorithm generates 9 page faults for having 3 free frames. Similarly we study about number of page faults by having different number of free frames in main memory. Following table illustrates about different page faults for different number of free frames in main memory for the above mentioned reference string.

| FreeFrames(X) | Page Faults(Y) | dx | dy | $dx^2$ | $dy^2$ | dx.dy |
|---|---|---|---|---|---|---|
| 1 | 20 | -5 | 10.9 | 20.25 | 118.81 | -49.05 |
| 2 | 16 | -4 | 6.9 | 12.25 | 47.61 | -24.15 |
| 3 | 9 | -3 | -0.1 | 6.25 | 0.01 | 0.25 |
| 4 | 9 | -2 | -0.1 | 2.25 | 0.01 | 0.15 |
| 5 | 7 | -1 | -2.1 | 0.25 | 4.41 | 1.05 |
| 6 | 6 | 0.5 | -3.1 | 0.25 | 9.61 | -1.55 |
| 7 | 6 | 1.5 | -3.1 | 2.25 | 9.61 | -4.65 |
| 8 | 6 | 2.5 | -3.1 | 6.25 | 9.61 | -7.75 |
| 9 | 6 | 3.5 | -3.1 | 12.25 | 9.61 | -10.85 |
| 10 | 6 | 4.5 | -3.1 | 20.25 | 9.61 | -13.95 |

*Fig 3: Calculation of Correlation Coefficient*

Based above table, Correlation Coefficient(r) [13] is calculated as follows.

$$r = \frac{\sum dx\, dy}{\sqrt{dx^2 dy^2}}$$

By substituting related values from the above table we get r values as "-0.82".The following Graph Represents the relationship between page faults and free frames.

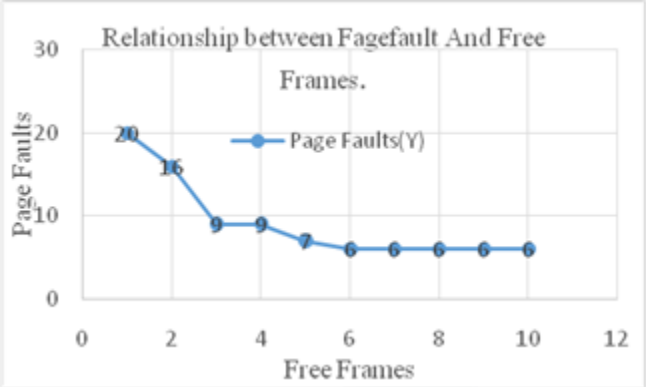


*Fig 4: Graph for correlation between Page faults and free Frames.*

## CONCLUSION

Finally we conclude that, based on the value of r we can analyze that there is a "high negative correlation is exist between page faults and number of free frames in main memory". It means that if the number of free frames increase gradually, the number of page faults decrease accordingly.

## REFERENCES

[1]. Manisha Koranga, Nisha Koranga "Analysis on Page Replacement Algorithms with Variable Number of Frames", IJARCSSE, Volume 4, Issue 7, July 2014.
[2]. HeikkiPaajanen, "Page replacement in operating system memory management", Master's Thesis, University of Jyväskylä, October 23, 2007.
[3].S.M. ShamsheerDaula, Dr. K.E Sreenivasa Murthy, G Amjad Khan. ", A Throghput Analysis on page replacement algorithm", International Journal of Engineering Research and Applications (IJERA), ISSN: 2248- 9622, Vol. 2, Issue 2, pp.126-13, Mar-Apr 2012
[4]. S. Jiang and X. Zhang. "LIRS: an efficient low inter-reference regency set replacement policy to improve buffer cache performance". SIGMETRICS Performance Evaluation Review, 30(1), 2002.
[5]. S. Jiang, and X. Zhang, "LIRS: An Efficient Policy to improve Buffer Cache Performance", IEEE Transactions on Computers, pp. 939-952, 2005.
[6]. Mukesh kumari, Anish kumar,Neha"Simulation of LRU Page Replacement Algorithm for improving system performance", Int.J.Computer Technology & Applications, Vol 4 (4), 683-685.

[7]. Abharm Silberschtaz "Operating system Principles", Willey Edition, and Pages (278-292,312,315-327).
[8]. Stuart E.Madnick, John. Donovan, "Operating Systems" TGMH Publishers, Pages (13-134,151).
[9]. William Stallings "Operating System: Internals and Design" Pearson Publishers, Pages (326-330,380).
[10]. Elamsri, Carrick & Levine "Operating System –A spiral approach", TGMH Edition, Pages (226-233,239,240).
[11]. D.M.Dhamdhere "Operating System-A Concept Based Approach" by, TGMH Publishers, Pages (215-219,266-274).
[12].H.M.Dietel, "Operating System" Pearson Publishers, Pages (220-236,253,255).
[13]. Dr.P.C.Tulsian, Jhunjhunu Wala"Business Statics" by, Chand Publishers.
[14]. Richard Levin "Statics for Management" By, Pearson Publishers.
[15]. C.Mallikarjuna,Dr.P.Chitti babu, "Performance Analysis of disk scheduling Algorithms", IJCSE, E-ISSN: 2347-2693, IJCSE-01650, IJCSE Volume- 4, and Issue-5.