An improve SJF Scheduling algorithm to Reduce Starvation under Multiprocessor Environment

Robin Verma
Computer Science & engineering, MTECH, ITS, Bhiwani, Haryana, India

Kiran Siwach
HOD, Computer Science & engineering, ITS, Bhiwani, Haryana, India

Sonia Rana
Assistant Professor, Computer Science & engineering, ITS, Bhiwani, Haryana, India

Abstract— multiple processes are get executed by different users in a computer system. It is the responsibility of the processor to avail the CPU time and resources to these processes effectively and to execute these processes effectively. The order of execution of these processes is known as the scheduling approach. To achieve the effective throughput and maximum utilization of processes there are number of scheduling approaches. Also, Sometimes the problem of starvation is occur in such scheduling schemes. An improved priority shift mechanism has defined by the presented work to minimize the average waiting time and to avoid starvation problem. The presented work includes the priority change mechanism as well as dynamic prioritization of the processes. The work is compared with SJF (shortest job first) scheduling approach and the results shows that the presented work has improved the work under all the parameters.

Keywords- Resource Allocation, Dynamic Prioritization, SJF, Maximizing Throughput.

I. INTRODUCTION

In today's world most of the available computers are able to provide the multi processing architecture to enhance the efficiency and throughput provided by the system.
The multi processor system is capable to manage the system resources and to provide the efficient turnaround time over the processor system. Such kind of computer systems are divided in two main categories, the first one is called the tightly coupled multiprocessor system and second one is called the loosely coupled multiprocessor system.

In loosely coupled, each processor have the memory to manage the processes and a separate queue. Where as in case of tightly coupled multi processor system having the queue and single memory and for all the processors.
A tightly coupled multiprocessor system is focused by the presented work. The multiprocessor system is work on the single job queue to accept all the processes executed by the users.
As a user, executes a process, the process enter to the job queue. It is scheduler's work to select a process from the job queue and avail it to the processor for its execution as well as to load it in the memory. There are so many existing scheduling approaches to take out the processes from the queue. These approaches includes the SJF (Shortest Job First), FIFO (First In First Out), LIFO (Last In First Out).
The framework provided by Processor Computing is basically to improve the probabilistic efficiency of the processing device with better resource usage achieve the better utilization of resources as well as to better processing throughput or result.
The easiest use of Processor computing would be to run an existing application on several computer systems. The work is defined for the multiprocessor system in which the effective queue management is done so that if the devices are busy, the execution of the task would be delayed.
Scheduling mechanism's related aspects deals by the processor computing. The scheduling mechanism aspects includes the processor utilization, memory utilization, processor contribution and the time state analysis of different processes. The processor computing becomes more complex in case of heterogeneous systems, where the work is

divided among the processors respective to the complexity of the task or type of the process. The processor computing is also having the responsibility to manage the resources to different processes in such way effective utilization of resources will be done and the situation of deadlock would never occur.

The scheduling mechanism adapted by a processing system could be preemptive or non-preemptive. In case of preemptive scheduling approach, the process under execution can be interrupted by the other processes of higher priority. Whereas the non-preemptive processes are never interrupted by the other processes. The processor computing is effective enough to take the decision regarding the allocation of the resources respective to scheduling mechanism or based on the requirement of the resources. In case of multi processor and multi processes systems, each resource also having a separate job queue to keep the processes in wait if the resource is not sharable.

The scheduling approach has defined along with each resource provider under three main perspectives that called the job queue, capability of the resource and the cost of the execution of a process. The capability is here been defined in terms of execution speed of the process either by the IO controller or by the processor. It also depends on the sharable property of resource. Another aspect included here is the utilization ratio or the throughput ratio respective to the speed. The execution time of the process can be estimated, based on these parameters. Another important perspective is the job queue. An important role has played by the size of the job queue and its characteristics while deciding the sequence of the scheduling. The queue can be a double ended queue, priority queue etc. The final aspect is about the cost of the process execution. The cost of the process depends on the number of resources required for a process as well as number of CPU clocks required.

A) Processor Computing

A computer system can easily solve complex problems and even the scientific problems as well. These systems are capable to solve the problems in different real time application are as such as security system, engineering, medicine etc. Now the computer system is more capable than a human brain to solve these problems in adaptive time and to provide the more effective computational solution over these problems. The processor system is basically defined to achieve the high throughput over the computing devices. The main objective of work is to achieve the essential computing over the system with shared resources mechanism. These systems are basically the mainframe systems that provide the power of unity in the computing system. The computing resources along with the system include the resource utilization, enhance the capabilities of the computing system. The computer system itself having the several components including the network system, CPU, memories etc.

B) Scheduling in Processor

In the traditional systems, a static model is been presented to manage the resources over the system. To manage these resources, a centralized controller is defined to manage the resources and job respectively. There are number of static and dynamic approaches to manage these jobs as well as the resources. One of the main objective of the scheduling is to define the inter-relation between these resources and the processes so that effective resource allotment and the process execution will be done. The processor system is defined to handle the challenges in terms of resource management such as scalability, adaptability fault tolerance and the reliability and to compute the cost estimation on these devices.

II. LITERATURE REVIEW

Lot of work is done in the area of resource allocation and the process scheduling already. Some of the earlier work done in same area is presented here. Vikki Tang has defined a work to reduce the instruction scheduling under the dynamic compilers, in year 2006. Author defined a scheduling approach under the feedback analysis so that effective allocation will be done. The presented framework is defined to benefit the instruction scheduling under multi threaded server applications. In year 2013, Lichaen Weng defined a work on multithreaded processor system to perform the dynamic modelling. The paper describe the design under three simple steps. At first, author conveyed a scheduling policy to dynamic to evaluate the runtime of pattern mapping. Another step is to define the regression model to achieve the scheduling policy to identify the changing behavior of the threading system. The main objective of author was to define a scalable heuristic approach for estimating the growth of the system count [2]. Hsiang-Yun Cheng is defined as an analytical model to achieve the task scheduling under the analytical modelling. Author estimated the potential aspects under the bandwidth analysis and memory to restrict the number of task.

Author implemented the scheduling under the real hardware [3]. In year 2013, Vishakha Gupta has performed the performance analysis for the functionality analysis under asymmetric platforms. Author has performed the analysis under the heterogeneity under the utility and applicability analysis. Author has defined the work under the workload analysis and defined it under different processes and different configuration for the resource analysis [4].

For multi programmed environments, Morris A. Jette defined the characteristics analysis under the scheduling process. Author defined a space and time slicing mechanism for the parallel programming and defined the concurrent job execution under the single processor environment. Author has defined a performance analysis system under the utilization and responsiveness under different computing platforms [5]. Another work for the heterogeneous scheduling policies for real time multi processor system is considered for the multimedia mapping for the space designing. Author has defined a suitable scheduling policy so that system energy can be minimized. The presented framework includes the analysis on energy reduction approaches for dynamic power management [6]. Another work on power management for multi-core architecture for the process scheduling is defined for the process estimation under platform evaluation. Author defined the effectiveness and scalability of the system. Author highlighted the scalability limitations for the thread scheduling algorithm for small scale multi processor system. Author has defined the scheduling overhead without loss of accuracy [7]. In Year 2005, Rony Ghattas examines the practicality of using these advanced techniques to save power and energy for commodity 8-bit microcontrollers while leveraging their built-in low-power modes. The benefits of the techniques are weighed against their complexity and cost. First, Author mathematically model the power dissipation characteristics of 11 popular 8-bit microcontrollers [8].

In Year 2003, Andrei Terechko defined the scheduling under the high level language with some variable definition with global values. Author defined the long range and large impact schedule for the compiler optimization for local values under the scheduling units. The paper has defined three main algorithms for assigning the values to different cluster under the multi pass scheduling approach under the variable definition. Author also defined the performance measures for optimizing the algorithm [9]. In Year 2004, Andrew Riffel also defined a multi pass partitioning problem with recursive denominator split along with heuristic algorithm so that the robustness over the approach will be achieved. This paper redefines the MPP as a scheduling problem and uses scheduling algorithms that allow incremental resource estimation and pass computation in effective time [10]. Another work on improvement over the energy efficiency was presented by Hiroshi Sasaki. The proposed method groups several instructions as a single issue unit and reduces the required number of ports and the size of the structure for dispatch, wakeup, select, and issue. The present paper describes the micro architecture mechanisms and shows evaluation results for energy savings and performance [11]. Flavius Gruian presented an addresses scheduling approach for reduced energy of hard real-time tasks with fixed priorities assigned in a rate monotonic or deadline monotonic manner. The approach Author describe can be exclusively implemented in the RTOS. It targets energy consumption reduction by using both on-line and off-line decisions, taken both at task level and at task-set level [12].

Martin Schoeber performed the investigation on the overhead analysis on object oriented operations. Investigated the overheads of object-oriented operations, such as virtual method dispatch and field access, in the context of an embedded processor for real-time systems. Author also evaluate the hardware cost of this optimization with respect to the application speedup [13]. In Year 2000, Jared Stark presented work on instruction scheduling for pipelined processing. This paper offers a third, acceptable, alternative: pipelined scheduling with speculative wakeup. This technique pipelines the scheduling logic without eliminating its ability to execute dependent instructions in consecutive cycles. With this technique, you sacrifice little IPC, and no clock frequency [14].

III. PROPOSED WORK

To achieve the interrelated resources and optimize utilization of processor, it is required to provide a better scheduling mechanism. The main objective of the work is to achieve the maximum throughput from the processor and integrated components and to reduce the wait time. The presented work is focused on a multi processor system in which a process can require input output devices for the execution and more than one processor. Once a process is executed, it is loaded in a job queue. From this job queue, the process is taken out for the execution. If the process is an input output process, then the process is also loaded in the queue maintained by input output device. The work is about to reduce the waiting time in terms of processor execution either it is in process queue or it is in input output queue.

In this presented work, the main focus is to reduce the starvation as well as resource allocation wait time. The work is about to analyze the wait time of each process to get the access on the required input/output device. Each process defined by the user is analyzed under the capabilities of the processor as well as the capabilities of input output devices. Number of access units or clocks are identified and analyzed to estimate the throughput and wait time of the processors. The objective of the work is to reduce the number of wait cycles so that overall reduction over the wait time will be done.

Another concept here is the prioritization. The initial priority assignment is here defined under the resource requirement. This requirement is defined based on the number of input output devices and the number of CPU required. Once the priority is decided, a limited threshold is applied to change the priority of the processes. Another aspect to change priority is defined based on the waiting threshold. If the wait time of a low priority process is increased by its threshold, we would like to modify the existing algorithms.

IV. CONCLUSION

The presented work is about to improve the effectiveness of the Processor system along with reliability. In this system we will keep the most frequent data items in cache by estimating the data frequency. As the most required data items are kept in the cache itself, it will improve the hit ratio and improve the reliability of data access. The system will give the better service allotment such way that the starvation will not occur over the system.

REFERENCES

[1] Vikki Tang," A Framework for Reducing Instruction Scheduling Overhead in Dynamic Compilers".
[2] Lichen Weng," Scheduling Optimization in Multicore Multithreaded Microprocessors through Dynamic Modeling", CF'13, May 14–16, 2013, Ischia, Italy. ACM 978-1-4503-2053-5
[3] Hsiang-Yun Cheng," An Analytical Model to Exploit Memory Task Scheduling", INTERACT-14, March 13, 2010, Pittsburgh, PA, USA ACM 978-1-60558-921-3/10/03
[4] Vishakha Gupta," Kinship: Efficient Resource Management for Performance and Functionally Asymmetric Platforms", CF'13, May 14–16, 2013, Ischia, Italy. ACM 978-1-4503-2053-5
[5] Morris A. Jette," Performance Characteristics of Gang Scheduling in Multiprogrammed Environments", 1997 ACM 0-89791-985-8/97/0011
[6] Minyoung Kim," Design Space Exploration of Real-time Multi-media MPSoCs with Heterogeneous Scheduling Policies", CODES+ISSS'06, October 22–25, 2006, Seoul, Korea. ACM 1-59593-370-0/06/0010
[7] Jonathan A. Winter," Scalable Thread Scheduling and Global Power Management for Heterogeneous Many-Core Architectures", PACT'10, September 11–15, 2010, Vienna, Austria. ACM 978-1-4503-0178-7/10/09
[8] Rony Ghattas," Energy Management for Commodity Short-Bit-Width Microcontrollers", CASES'05, September 24–27, 2005, San Francisco, California, USA. ACM 1-59593-149-X/05/0009
[9] Andrei Terechko," Cluster Assignment of Global Values for Clustered VLIW Processors", CASES'03, Oct. 30 – Nov. 1, 2003, San Jose, California, USA. ACM 1-58113-676-5/03/0010
[10] Andrew Riffel," Mio: Fast Multipass Partitioning via Priority-Based Instruction Scheduling".
[11] Hiroshi Sasaki,"Energy-Efficient Dynamic Instruction Scheduling Logic through Instruction Grouping", ISLPED'06, October 4–6, 2006, Tegernsee, Germany. ACM 1-59593-462-6/06/0010
[12] Flavius Gruian," Hard Real-Time Scheduling for Low-Energy Using Stochastic Data and DVS Processors", ISLPED'01, August 6-7, 2001, Huntington Beach, California, USA. ACM 1-58113-371-5/01/0008
[13] Martin Schoeberl," Architecture for Object-Oriented Programming Languages", JTRES '07 September 26-28, 2007 Vienna, Austria ACM 978-59593-813-8/07/09
[14] Jared Stark," On Pipelining Dynamic Instruction Scheduling Logic", 0-7695-0924-X/2000© 2000 IEEE