ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

Modification of Linux Kernel with Jumping Virtual Clock Round Robin Scheduling

Laxmi M. Gurnani[1], Anurag Sinha[2], Gurmeet Singh Gujral[3], Abhijeet Patil[4], Ashish T. Bhole[5]

UG Student, Department of Computer Engineering, SSBT's COET, Jalgaon, Maharashtra, India[1,2,3,4]

Associate Professor, Department of Computer Engineering, SSBT's COET, Jalgaon, Maharashtra, India[5]

ABSTRACT: The open-source software and free operating system has attracted the attention of many users. People have failed to notice that while installing regular distributions, user is often forced to install many programs which are probably never used. These programs waste resources. Additionally in Linux, the performed scheduling algorithms are with high complexity such as Weighted Fair Queuing. Unless one can change a Linux OS, one can't avoid presence of components which will never be required. Linux From Scratch (LFS) system allows to create compact Linux systems. LFS guides for building Linux system, compiling and installing all packages one by one, setting up boot scripts and installing the kernel. Additionally process scheduling can be implemented to meet growing needs such as faster execution. Over LFS system, one can build a hybrid scheduling algorithm called Jumping Virtual Clock Round Robin algorithm which combines Dynamic JVC and Static RR to solve the problems present in existing Weighted Fair Queueing and Leaf Forward Virtual Clock algorithms.

KEYWORDS: Linux based distribution, Fair Scheduling, Leaf Forward Virtual Clock, Weighted Fair Queuing, Self Clock Fair Queuing, Linux from Scratch (LFS), Jumping Virtual Clock Round Robin (JVCRR).

I. INTRODUCTION

Linux based distribution consist of Linux kernel and set of packages which constitute the Linux based distribution. Linux kernel is modular in nature, thus consists of a scheduler module. Scheduler module is build using scheduling algorithm. The scheduling algorithm in the Linux kernel version 2.6 is Fair Scheduling. The Fair scheduling doesn't guarantee to execute all the tasks at O(1) rate[1]. Hence the proposed work is implemented by using a new hybrid scheduling algorithm Jumping Virtual Clock Round Robin (JVCRR) which is a combination of three scheduling algorithm namely virtual clock, jumping queue and Round Robin (RR).

The scheduler developed cannot be directly incorporated in the Linux kernel as it may lead to several malfunctions which can be very difficult to identify and understand at development level hence a Linuxsche tool will be used first to measure the functionality of the scheduler is correct in all aspect or not. The Linuxsche is a simulator which will make the same environment available for the testing the correctness of the simulator and this Linuxsche tool is again open source and hence are available easily and free.

The other part of the work which deals with the packages to build the operating system on top Linux kernel and hence these packages are included in Linux distribution with the help of an open source tool known as LFS. LFS is basically a collection of steps one need to follow in order to develop a Linux distribution. LFS support each and every aspect of Linux distribution development. Thus work aims at development of a new Linux scheduler for Linux kernel version 2.6 and developing a basic Linux based distribution for basic computation needs.

LFS is project under which all the guideline for the developers is provided so that one can develop their own personalized distribution. The modular nature of Linux makes it most versatile operating system and an operating system which can be ported on simple personal computer to super computer. LFS mainly deals with the compilation of various packages available for the Linux in repositories which are distributed all over the world and are brought together with the help of internet.

The most obvious reason which makes it most easy to share and develop is its open source tag which allows any developer to use the 3 step away from a new Linux distribution which is copy, paste and edit. This work aims at

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

making an efficient Linux distribution. Efficient Linux distribution is based on the effective utilization of system resources. A new scheduling algorithm JVCRR is implemented at kernel level and basic packages meeting all the requirement of the user will be included in Linux distribution. Linux distribution will be developed with the help LFS.

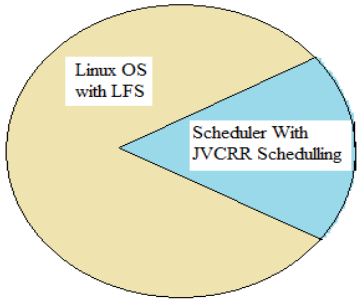

Figure 1: Linux Operating System

As shown in Figure 1, Scheduler is part of kernel, which is modified with new scheduling algorithm JVCRR. The remaining part that is non-kernel part and kernel part (other than scheduler) of operating system is implemented with the help of LFS. By the use of Jumping Virtual Clock, the complexity of calculating Virtual Time is reduced to the O(1). If the selection of next queue can also be lowered to O(1), the scheduling algorithm is with the lowest complexity. Thus, the Jumping Virtual Clock is incorporated with Round Robin to become the JVCRR [3].

II. RELATED WORK

The efficiency of any operating system depends mainly on how fast it executes the task. In an OS, scheduler module schedules the task to kernel according to priority or shortest job. The decision of selecting next task to execute is depends on the scheduling algorithm used in an OS. There are various scheduling algorithms are available and hence to choose an effective algorithm is very crucial.

The scheduling algorithms like Weighted Fair Queuing, Virtual Clock, Round Robin and Jumping Virtual Clock are available but not so efficient. As all of them require, so-much time to execute any instruction. The Complexity of all above algorithms is approximately O(log N) which is not so much effective. Any OS which is said to be better and efficient should have at least O (1) complexity [3].

The algorithms like Virtual Clock and Leaf Forward Virtual Clock (LFVC) involves O(1) complexity for calculating virtual time of process. But for selecting next queue these algorithms require complexity O(log N) and O(log log N) respectively which is again not enough for any faster operating system[3].

In current Linux kernel, Fair scheduling is used which is efficient, but does not guarantee to execute programs at O(1). Hence new scheduling algorithm called JVCRR which guarantees O(1) rate is used to improve the efficiency of overall Linux system.

This algorithm JVCRR is the combination of Jumping Virtual Clock and Round Robin scheduling algorithm. This algorithm involves calculating virtual time and selecting next queue. Both calculating virtual time and selecting next queue involves O(1) time complexity making its overall complexity as O(1). Hence, JVCRR is better as compared to other mentioned scheduling algorithms.

III. PROPOSED WORK

The proposed work is the system which is a basic Linux distribution according to LFS and hence will include only packages which are needed for primitive and basic operation of operating system. Hence exclusion of remaining packages makes Linux distribution very compact. The proposed system also includes a kernel module which is a scheduler working on JVCRR scheduling algorithm which has a time complexity of O(1) and provide better efficiency than other scheduling algorithms. The involvement of both the things make Linux distribution very faster for executing the tasks present in system or tasks generated by the user.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

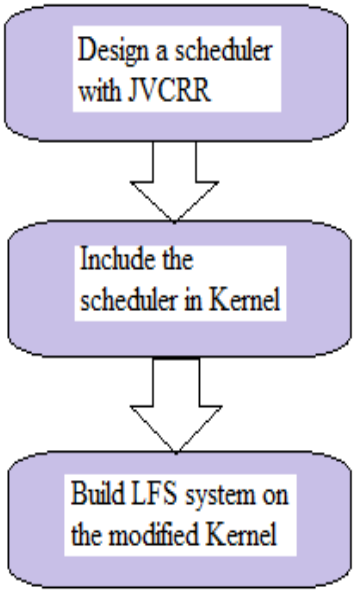

Figure 2: Flow of System.

The Figure 2, shows the flow of overall system i.e. the basic steps one need to follow while modifying Linux kernel with JVCRR scheduling. The first step is to design a scheduler by modifying the scheduling algorithm in current Linux system. Next step is to include the modified scheduler in the Linux system i.e. replace the previous scheduler with the new scheduler. Third and the final step is to build the LFS system on the modified Kernel.

The proposed work is developed using Linux distribution, as Linux is open source and someone can easily modify the code. Also it is freely available operating system so, it is economically feasible. Someone can easily download the source code and make the changes. As compare to other operating system like Windows , Linux is more feasible in various aspects and hence it includes the Linux.

IV. RESULTS AND DISCUSSION

The resultant Linux is very efficient because of the involvement of scheduling algorithm JVCRR which schedules the tasks at O (1) rate. The Complexity of JVCRR is compared with other frequently used scheduling algorithms which are shown in Table 1.

Table 1: Complexity of Algorithms

Algorithm	Overall Complexity
WFQ	O(N)
LFVC	O(log log N)
SCFQ	O(log N)
VC	O(log N)
JVCRR	O(1)

As shown in Table 1, the JVCRR involves the least complexity, which is O(1). The other algorithms like Weighted Fair Queuing(WFQ) have complexity as O(N), Leaf Forward Virtual Clock(LFVC) and Self Clock Fair Queuing(SCFQ) involve O(log log N) and O(log N) complexities respectively and Virtual Clock Algorithm have O(log N)[3].

The other thing which makes the resultant Linux distribution as efficient is the selection of only required packages for developing Linux distribution using LFS and integrating it with the modified scheduler. The selection of only required packages, saves the memory and time to load and boot the OS. As the memory is saved, because of compact Linux, the

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

more tasks can reside in memory at a time for their execution. Hence the overall execution will become faster, increasing the performance rate of the Linux system.

V. CONCLUSION

The Linux distribution developed as a product is for the very need of user who wants a small, complete and an operating system which will utilize system resources efficiently. The proposed work will assure that the distribution takes least amount of memory and still be able to provide full functionality of an operating system. It will be upto the user whether to incorporate any other components or not. If yes, the user can add packages with the help of package management tool YUM. The scheduler developed with JVCRR included in Linux kernel 2.6, has improved the performance of the system in accordance with the current scheduler present in kernel.

REFERENCES

1. Jonathan Corbet, Amanda McPherson, 'Linux Kernel Development', The Linux Foundation, http://www.linuxfoundation.org, Dec 2010.
2. Masrurkhah, Alireza Abed, Amir Seyed Danesh, and Seyyedeh Narjes Ghiami Taklimi, 'A survey on implementation of a Linux-based operating system using LFS method' International Journal of Computer Science Issues (IJCSI) 9, no. 2 (2012).
3. Hwang, Lain-Chyr, Steen J. Hsu, San-Yuan Wang, and Yong-Hua Huang. 'A hybrid scheduling algorithm with low complexity: Jumping Virtual Clock Round Robin', In Distributed Computing Systems Workshops, 25th IEEE International Conference on, pp. 698-703. IEEE, 2005.
4. Sukumar B. B, Neelima P. N, and Sunil. Kumar.B, 'Efficient Round Robin CPU Scheduling Algorithm', volume 4, Issue 9, pp. 36-42, November 2012.
5. Robert Love 'Linux Kernel Development', Third Edition, ISBN-13: 978-0-672-32946-3, June 2010.
6. Gerard Beekmans, 'Linux From Scratch', Version 7.4, 2013.
7. Silberschatz, 2.P. Galvin, and 3.G. O. Gagne, "Operating System Concepts", 7th Edition, John Wiley and Sons. INC, 2005.