Improvement of The Level Performance For Interrupt-Driven Kernel in Gigabit Networks

Fadi Hani Kduher
Wisam Abd Al-Hussein
Raad K. Hassan
Samar A. Mahmood
Yasameen H. Shamkhi
Ministry of Science and Technology
Physics Researches and Science Directorate

Abstract

In this work presents models and analytical techniques for studying the nature of the doing system of an interrupt-driven kernel due to high packet arrival rate appeared in Gigabit networks. An analytical study is presented describing the effect of high interrupt rate on system performance. The performance is studying in requisites of throughput, latency, and system power. The derived equations of system throughput, latency, system power, and stability condition. The effect of interrupts on system performance had never been denoted to it yet analytically in the past. and this analytical work is the first of its kind. In this work is also using simulation considered both Poisson and bursty traffic with empirical packet size distribution.

1. Introduction

Interrupt overhead of Gigabit network devices can have important negative effect on system performance. Traditional operating systems were designed to handle network devices that interrupt on a rate of arround 1000 bundles per second, as is the case for 10Mbps Ethernet. The cost of handling interrupts in these traditional system was low that any normal system would spend only a part of the its CPU time handling interrupts. For 100 Mbps Ethernet, the interrupt rate increases to about 8000 interrupts per second using the maximum 1500 byte bundle. For Gigabit Ethernet , the interrupt rate for the maximum sized bundle of 1500 bytes increases to 80000 interrupts per second.

In Gigabit network, the bundle arrival rate more than the system bundle processing rate with includes network protocol cumulative processing and

interrupt handling. Interrupt driven ststems tend to perform very badly under such heavy load. Interrupt level handling by definition, has absolute priority over all other tasks. If interrupt rate is high, the system will spend all of its time responding to interrupts, and nothing else will be performed, and therefore the system throughput will drop to zero. This method is called receive livelock[1]. In this method is not stop, but it makes no process in tasks, at low bundle arrival rate, the cost of interrupt overhead and latency for handling incoming bundle are low. Therefore, interrupt overhead cost increases with an increasing of bundle arrival rates, causing livelock.

The receive livelock condition was shown by experiments and measurements in real system [2,3]. In this work we present a model for the livelock method and show its analytical solution. These models can be utilized to understand predict the performance and behavior of interrupt driven system and can be served as a reference model for comparing performance of these proposed solution to resolve the receive livelock method. And the paper presents an analytical study of system performance in terms of throughput, latency, and system power due of high rate of interrupts found in Gigabit networks.

A number of solutions have been proposed to minimize the interrupt overhead and resolve receive livelock method. Such solutions include interrupt coalescing, OS-bypass protocol pushing some or all protocol processing to hardware, etc. Some of these solutions are listed in [4,5].

The rest of the work is organized as follows. Section two presents analysis for two models: an ideal system that ignores the effect of interrupts on system performance, and a second model that captures the system behavior under low and high network traffic intensity. Section thrrr presents the numerical examples. Finally, section four has the conclusion and identifies future work.

2. Theoretical Aspect

in this section the analytical study to examine the effect of interrupts on system performance was done. At first the system performance were defined. Let λ be the average incoming bundle arrival rate, and μ be the average protocol processing rate by the kernel. Therefore $1/\mu$ is the time it takes the system to process the incoming bundle and deliver it to the application program. This time includes primarily the network protocol stack processing by the kernel, excluding any interrupt handling. However, the interrupt handling time will be denoted as T_{ISR} , which is basically the interrupt service routine time for handling incoming bundle. ρ as a measure of the traffic intensity or system load and was defined as λ/μ .

It was studied the system performance in term of three used performance metrics. These metrics include throughput, latency, and system power. System

throughput γ is the rate at which bundle are delivered by the kernel to the application program. Latency or the mean response time R which is the time duration between a bundle arrival at the NIC and the delivary to the application program. Since improvement in system throughput would a have a negative effect on latency, system power p was proposed in [6] which resolves this contradiction. System power gives the correct operating point that maximizes throughput and minimize latency.

In the ideal system: the analysis for the ideal situation in which the overhead involved in generating interrupts to totally ignored. Assuming bundle are all of fixed size, we can simply model such a system as an an $M/M/1/B$ queue with a Poisson distribution bundle arrival rate λ and a mean protocol processing time of $1/\mu$ that has an exponential distribution. B is the maximum size the system buffer can hold. $M/M/1/B$ queueing model is chosen as opposed to $M/M/1$ since we can have arrival rate go beyond the service rate, i.e. $\rho > 1$. This assumption is true in Gigabit environment where under heavy load λ can be very high compared to μ .

It is worth mentioning that in our analysis we assume a Poisson arrival for network traffic. It is has to be stated that network traffic is not always Poisson distribution nature.

In $M/M/1/B$ model, the system throughput can be expressed as

$$\gamma = \mu(1 - p_o), \qquad \ldots\ldots \quad (2-1)$$

where p_o is the probability that the system is ideal and give by

$$p_o = \begin{cases} \dfrac{1-\rho}{1-\rho^{B+1}} & (\rho \neq 1), \\ \dfrac{1}{B+1} & (\rho = 1). \end{cases}$$

System bundle latency R can be given by

$$R = \frac{E(n)}{\lambda(1 - p_B)}, \qquad \ldots\ldots \quad (2-2)$$

where $E(n) = \dfrac{\rho}{1-\rho} - \dfrac{B+1}{1-\rho^{B+1}}\rho^{B+1}$ and p_B is the probability of bundle dropped due to buffer being full. And system power is expressed by [7] as

$$P = \gamma^{\alpha} / R. \qquad \ldots\ldots \quad (2-3)$$

Where α is a positive real number, Where increasing throughput and decreasing latency are given equal weight. For the study it will set $\alpha = 1$.

Effect of Gigabit network interrupt: modeling an interrupt driven system is a challenging task especially when it considered the Gigabit networking environment where $\rho > 1$. For every incoming bundle , an interrupt is initialed. The system processes the bundle by first executing the ISR and then handling it to the protocol stack where it gets processed. Hence, the system protocol processing time per bundle is simply equal to $T_{ISR} + 1/\mu$. However the value of this processing time is not true all the time and it depends on the arrival time of the next bundle. If the next bundle arrives while handling the interrupt of a previous bundle, i.e. while the system execution has not finished the current ISR , the value of this process time will be $T_{ISR} + 2/\mu$. This is true since the new interrupt is being masked off because another interrupt of the same interrupt priority level is being serviced. So a new T_{ISR} is not incurred. However, kernel time to process two bundle by the protocol stack will be $2/\mu$.

As aggod design practice, that would like to minimize the execution time of the ISR as much as possible. We assume the primary job of the ISR is to notify the kernel of the arrival of a new bundle. The notification only happens after the bundle is copied by the direct memory access (DMA) to the system host memory. This assumption is valid since in Gigabit networkink environment, the use of DMA becomes necessary in order to elimination any CPU overhead involved in copying bundles from the NIC to kernel memory.

After the notification of the arrival of a new bundle, the kernel will process the bundle by first examining the type of frame being received and then invoking immediately the proper handling stack functionor protocol, e.g. ARP, IP, TCP. The bundle will remain in the kernel or system host memory until it is discarded or delivered to the user program or application.

We also assume that the protocol processing for bundle by the kernel will continue as long as there are bundles available in the system memory buffer. This protocol processing of bundles can be interrupted by ISR executions as a result of new bundle arrivales. This is so because bundle processing by the kernel runs at a lower priority than the ISR.

One may think that such on interrupt driven system can be simply modeled as a priority queueing system with preemption in which there are two arrivals of defferent priorites. The first arrival constitutes that for ISR_S and has the higher priority. The second arrival is the arrival for incoming bundle, and has the lower priority. As noted the ISR execution preemptsprotocol processing. This is an invalid model because ISR handling is ignored if the system as shown in figure (1).

Mean Effective Service Time: In this method, it found the mean effective service time for processing bundle in the kernel protocol stack. We first find the formula for the mean effective service time. The system can be modeled as an M/G/1 queue with a Poisson distribution bundle arrival rate of λ and a mean effective service rate of μ' that takes a general distribution.

As shown in figure (1), the effective serivce time is the actual time available for servicing a bundle, exclusive of T_{ISR} distribution. The available service time is the available time between successive T_{ISR}. If bundle or multiple bundles arrive during T_{ISR}. We will have batched or masked-off interrupts and the bundle will be queued in to the system with effectively one T_{ISR} disrupting the service time. The disruption of the srvice time is mainly influenced by the arrival rate of the bundle λ and $T_{ISR.}$

Let us assume that T_{ISR} is exponentially distributed with mean T_{ISR}=1/r. One can express the mean effective service rate as:

$\mu' \rightarrow$ Rate at which bundles are processed by the kernels network protocol with no interrupt disruption.

$\mu' = \mu$ (% CPU availability for protocol processing) ….. (2-4)

In order to determine the CPU availability percentage for protocol processing and interrupt handling, we use a Markov process to model the CPU usage, as illustrated in figure (2). The process has state (0,0) and state (1,n). State (0,0) represents the state where the CPU is a vailable for protocol processing. State (1,n) with $0<n<\infty$ represent the state where the CPU is busy handling interrupts. n denotes the number of bundle arrivals that are being batched or masked off during T_{ISR}. Note that when process is in state (1,0), this means there are no interrupts being masked off and the CPU is handling a single interrupt.

The steady state difference equations can be derived from $0 = pQ$ where $p = \{p_{0,0}, p_{1,0}, p_{1,2}, ...\}$ and Q is the rate transition matrix and is defined as follows

$$Q = \begin{Bmatrix} -\lambda & \lambda & 0 & 0 & 0 & ... \\ r & -(\lambda + r) & \lambda & 0 & 0 & ... \\ r & 0 & -(\lambda + r) & \lambda & 0 & ... \\ r & 0 & 0 & -(\lambda + r) & \lambda & ... \\ r & 0 & 0 & 0 & -(\lambda + r) & ... \end{Bmatrix}$$

This will yield

$$-\lambda\, p_{0,0} + r(p_{1,0} + p_{1,1} + p_{1,2} + ...) = 0.$$

Since you know that $p_{0,0} + \sum_{i=0}^{\infty} p_{i,j} = 1,$ than $-\lambda\, p_{0,0} + r(1 - p_{0,0}) = 0.$

Solving for $p_{0,0}$ you thus have

$$p_{0,0} = \frac{r}{\lambda + r}\cdot \text{ and } 1 - p_{0,0} = \frac{\lambda}{\lambda + r}.$$

Therefore the percentage of the CPU available for protocol processing bundle and handling interrupts are $r/(\lambda + r)$ and $\lambda/(\lambda + r)$, respectively. And thus, the mean effective service rate can be expressed as:

$$\mu' = \mu \frac{r}{\lambda + r}. \qquad \ldots\ldots \quad (2-5)$$

It is to noted from equation (2.4) that the mean effective service rate μ' is exponential. the system can be modeled as M/M/1/B queue as is the case for the ideal system. The mean service rate μ will be replaced by the mean effective service rate μ'. Hence, the system throughput γ, latency R, and power P are expressed by equations (2-1),(2-2) and (2-3), respectively.

A particular point of interest is finding the stability condition for the system. The stability condition is the situation where ρ<1, or is defined as the cliff point for system throughput. It is where the throughput starts falling to zero as the system load increases. The stability condition for the system can be expressed as:

$$\rho < 1 \quad or \quad \lambda < \mu \frac{r}{\lambda + r}$$

Solving for λ get.

$$\lambda(\lambda + r) < \mu r \quad \Rightarrow \quad \lambda^2 + r\lambda - \mu r < 0$$

The roots of the quadratic equation $\lambda^2 + r\lambda - \mu r = 0$ are

$$\lambda = \frac{-r \pm \sqrt{r^2 + 4\mu r}}{2} = \frac{-r \mp \sqrt{1 + 4\frac{\mu}{r}}}{2}.$$

Since the term under the square root is always greater than one then the negative sign is neglected. The system will be stable whenever

$$\lambda < \frac{r}{2}\left(\sqrt{1 + 4\frac{\mu}{r}} - 1\right). \qquad \ldots\ldots \quad (2-6)$$

Another important point is finding the maximum system power point. This point is also the system correct operating point which gives maximum throughput and the minimum latency. In order to accomplish this, we take the derivative of the power function with respect to λ, and solving the derivative after making it equal to zero. From[8], the maximum power point occurs when $\rho < 1$. It is suitable to model the system in this case only as M/M/1 , since there

is no need to consider the case when $\rho > 1$ as we all along assumed. For this case, the throughput and latency as a function of λ are denoted by $\gamma(\lambda)$ and $R(\lambda)$.

$$\gamma(\lambda) = \mu'(1 - p_o) = \mu'\left(1 - \left(1 - \frac{\lambda}{\mu'}\right)\right) = \mu'\left(\frac{\lambda}{\mu'}\right) = \lambda.$$

$$R(\lambda) = \frac{E(n)}{\lambda} = \frac{\frac{\lambda}{\mu' - \lambda}}{\lambda} = \frac{1}{\mu' - \lambda}.$$

$$\therefore P(\lambda) = \frac{\lambda(\gamma)}{R(\lambda)} = \frac{\lambda}{1/(\mu' - \lambda)} = \lambda(\mu' - \lambda).$$

Taking the derivative of $P(\lambda)$,

$$\frac{dP(\lambda)}{d(\lambda)} = \mu' - 2\lambda$$

Setting $dP/d\lambda = 0.$ we get $\lambda = \frac{1}{2}\mu'$.

The maximum power point occurs when

$$\lambda = \frac{r}{2}\left(\sqrt{1 + 2\frac{\mu}{r}} - 1\right). \qquad \text{.....} \quad (2-7)$$

3. Numerical Example

In this method, it reported some numerical result of our analytical model to study the behavior of the system and the effect of interrupt on system performance. The system performance is studied as a function of traffic intensity ρ. Numerical result are also given for the ideal system when ignoring interrupts. For all of these result, we fix μ to 1 and B to a size of 1000.

Examine the system throughput as a function of traffic intensity ρ was examined firstly. This study relate with three T_{ISR} time unit 0.2,0.3, and 0.5. A T_{ISR} time unit of 0.2 means that the interrupt service duration is 20% of the duration of the bundle protocol processing time $1/\mu$.

Figure 3 shown the effect of high and low traffic intensity of system throughput. We note for the ideal system, the throughput is the expected one and matches very closely to the behavior of receive livelock. The throughput is different when considering interrupt effect, i.e., the receive livelock phenomenon. We note that the throughput doesn't fall rapidly to zero due to interrupt batching as illustrated in section mean effective service time. Figure 3

shows the system throughput for three cases of T_{ISR} 0.2,0.3, and 0.5. it is noted that as the interrupt overhead increases.

Figure 3 also shows the cliff points for the system throughput. As previously defined, the cliff points are those points where system throughput starts falling to zero as the system load increases. As shown, the cliff points in term of traffic intensity ρ for T_{ISR} of 0.2,0.3 and 0.5 are 0.85, 0.81, and 0.73, respectively. Since we are fixing μ to 1, the cliff points are the same for the system throughput, traffic intensity, and bundle arrival rate. These points match exactly the points derived by equation (2-6) for finding the stability.

Figure 4 shown the relation between bundle latency and traffic intensity for the same system parameter values considered for system throughput. The effect of low and high traffic intensity on system power is shown in figure 5. In the ideal system, the maximum system power is shown $\rho = 0.5$. However , the maximum system power decreases with different values of T_{ISR} , giving the least value for T_{ISR}=0.5. In addition the figure shows that the maximum power point for the system for T_{ISR} of 0.2, 0.3, and 0.5, are for λ of 0.46, 0.45, and 0.41. These points match also exactly the points derived by equation (2-7) for finding λ that give the maximum power point.

4. Conclusion

it was presented a valid analytical model that captures the behavior of interrupt driven system when subjected to high interrupt rates. We proposed and studied two models. In ideal system that ignores the effect of interrupts on system performance, and second model which capture the system behavior under low and high traffic intensity. Simulation and report experimental results show that our analytical model is valid and give a good approximation. It was concluded that the system performance under bustry traffic was similar to that of Poisson distribution traffic. this analysis effort provided equation that can be used to easily and predict the system performance

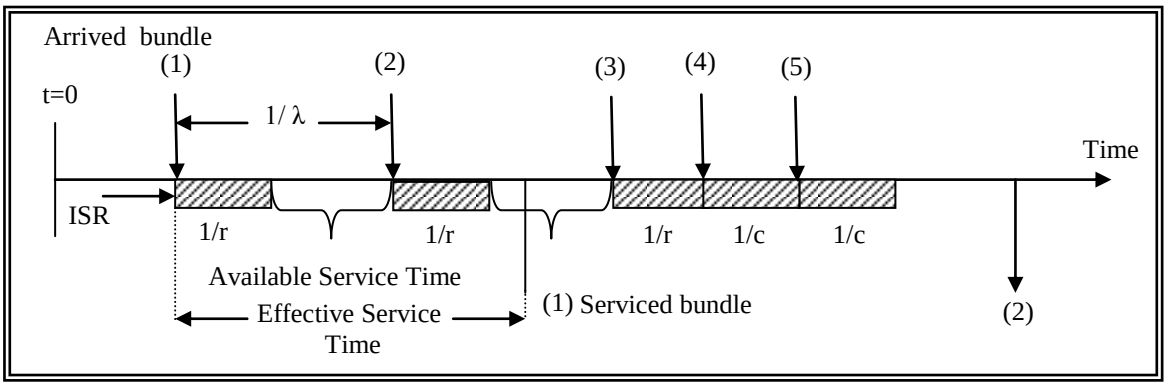

Figure (1): Effective Service Time

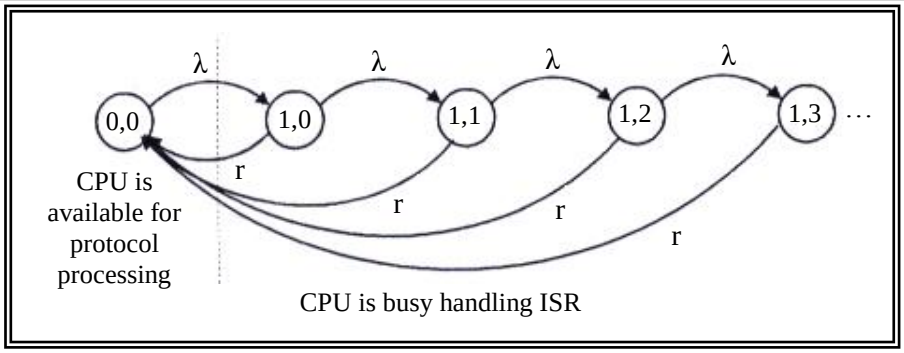

Figure (2): Markov state transition diagram to modeling CPU usage

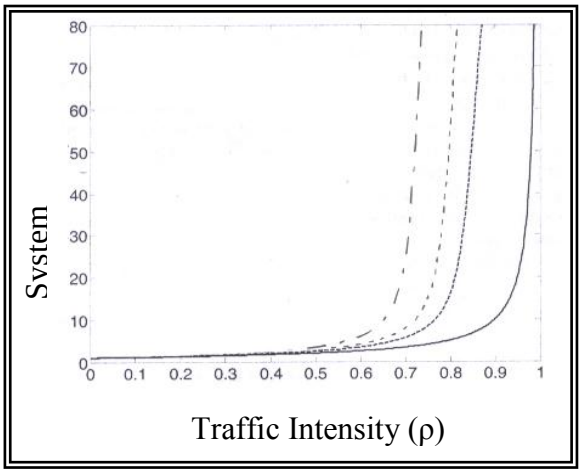

Figure (4): System Latency Traffic Intensity

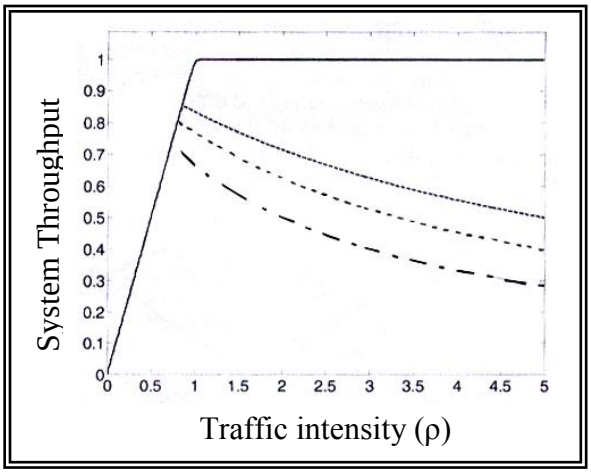

Figure (3): System Throughput Traffic Intensity

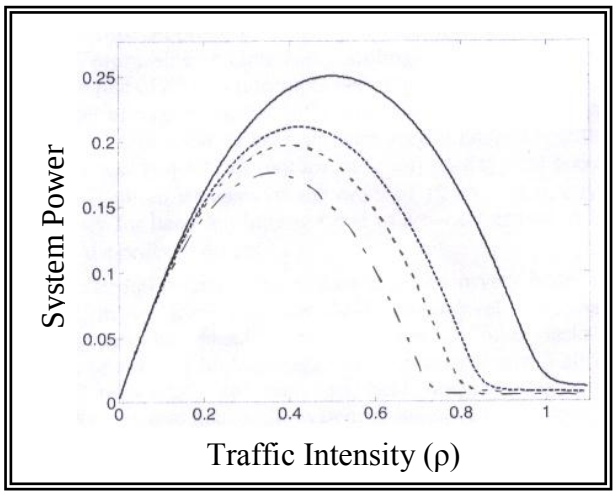

Figure (5): System Power Traffic Intensity

References

1- K. Ramakrishnan, "Performance Consideration in Designing Network Interface" IEEE Journal on Selected Areas in Communications, vol. 11, no. 2, February 1993, pp. 203-219.
2- J. Mogul, and K. Ramakrishna, "Eliminating Receive Livelock in An Interrupt Driven Kernel" ACM Ttrans. Computer Systems, vol. 15, no. 3, August 1997, pp. 217-252.
3- A. Indiresan, A. Mehra, and K. G. Shin, "Receive Livelock Elimination via Intelligent Interface Backoff" TCL Technical Report, University of Michigan, 1998.
4- P. Druschel, and G. Banga, "Lazy Receive Processing (LRP): a Network Subsystem Architecture For Sserver System" Proc. Second USENIX Symp. On operating system Design and Implementation, October 1996, pp. 261-276.
5- P. Shivan, P. Wyckoff, and D. Panda, "EMP: Zero-Copy OS- bypass NIC-Driven Gigabit Ethernet Message Passing" Proceedings of SC2001, Denver, Colorado, USA, November 2001.
6- C. Dovrolis, B. Thayer, and P. Ramanathan, "HIP: Hybrid Interrupt Polling For the Network Interface" ACM Operating System Reviews, vol. 35, October 2001, pp. 50-60.
7- A. Giessler, J. Haanle, A. Konig, and E. Pade, "Free Buffer Allocation an Investigation By Simulation" Computer Networks, vol. 1, no. 3, july 1978, pp. 191-204.
8- L. Kleinrock, "On The Modeling and Analysis of Computer Network" Proceedings of The IEEE, vol. 81, no.8, 1993.

تحسين مستوى الأداء لـ Interrupt-Driven kernel في شبكات الجيجابت

***فادي هاني خضر، وسام عبد الحسين موسى، رعد كاطع حسن، سمر عزيز محمود،**

ياسمين حميد شمخي

***وزارة العلوم والتكنولوجيا / دائرة علوم وبحوث الفيزياء**

الملخص

في هذا البحث نقدم نماذج وتقنيات تحليلية لدراسة طبيعة عمل نظام تقاطع النواة بسبب ظهور حزمة عالية ظهرت في شبكات Gigabit . نقدم دراسة تحليلية التي تصف تأثير مستوى التقاطع العالي على أداء نسبة النظام . يدرس الأداء في متطلبات الطاقة الإنتاجية ، الاختفاء ، وقوة النظام . المعادلات مشتقة من طاقة النظام الإنتاجية ، الاختفاء ، قوة النظام ، وشرط الاستقرار. تأثير التقاطع على أداء النظام لم يسبق وان عرف لحد ألان بشكل تحليلي في الماضي. وهذا العمل التحليلي الأول من نوعه. في هذا البحث استعملت المحاكاة أيضاً واعتبرت كل من توزيع بواسون والمرور المتقطع بتوزيع حجم الحزمة التجريبي.