트레이딩 플랫폼의 네트워크 지연 비교 연구

박지영, 손석원
호서대학교 컴퓨터공학과
e-mail:silberin@naver.com

Network latency comparison of the trading platform

Jiyoung Park, Surgwon Sohn
Dept of Computer Engineering, Hoseo University

요 약

Windows 환경에서 상용 저지연 NIC를 이용하여 컴퓨터 네트워크 통신 지연을 감소시킬 수 있다. 일반적으로 시스템의 커널에서 네트워크 처리를 담당하지만 본 논문은 커널을 우회하여 NIC에서 처리하여 운영체제에서 발생하는 지연을 최소화한다. 상용 NIC과 광섬유 케이블을 사용하여 네트워크 지연에 대한 비교결과를 보이며 네트워크 저지연 시스템의 구성을 제시한다.

1. 서론

트레이딩 시스템에서 시세 및 주문 정보의 전송 지연(Latency)은 시세정보(Market data)가 거래소부터 투자가까지, 그리고 주문정보가 투자가로부터 거래소까지 도달하는데 걸리는 시간을 말한다. 이것은 크게 세 가지 요인에 의해 발생하는데 CPU에서 시세정보를 처리하고 거래 알고리즘을 처리하는데 발생하는 처리지연 (Processing latency), 네트워크에서 발생하는 통신지연 (Transmission latency), 그리고 물리적 전송선에서 발생하는 전파지연 (Propagation latency)이다.

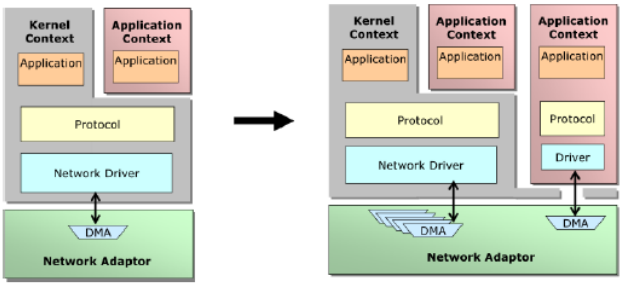
본 논문은 네트워크 지연에 관한 것으로, 네트워크 지연을 줄이는 방법에는 일반적으로 운영체제의 TCP/UDP 기능을 NIC(Network Interface Controller)의 하드웨어로 대체하여 연산 부하를 덜어내는(Offload) 방법을 많이 사용한다. 네트워크 지연은 상용 NIC에 따라 커다란 차이를 보인다. ASIC 또는 FPGA를 사용하는 NIC과 그렇지 않은 NIC에 따라서 성능차이를 보인다. 따라서 적절한 시스템을 구성하면 저지연 트레이딩 플랫폼을 구축할 수 있으며 본 연구에서 이를 제안한다. 하드웨어로는 ASIC 또는 FPGA를 이용해서 TOE(TCP Offload Engine) 기능을 사용한다. TOE에는 Checksum Offload, Large Receive Offload, Large Send Offload 등이 있다.

2. 관련 연구

일반적으로 고성능 상용 NIC에는 Solarflare, Chelsio사의 NIC가 사용된다[1]. 이러한 하드웨어에는 네트워크 지연을 개선하기 위한 TCP 오프로드 엔진 기능이 존재한다. 일반적으로 커널에서 TCP/IP 스택처리를 하지만 TOE는 네트워크 컨트롤러에서 스택처리를 하여 CPU의 부하를 오프로드하기 위해 사용된다[2-4].

3. 네트워크 지연

Linux에 적용되는 저지연 기술로서 Solarflare사의 오픈소스 OpenOnload 네트워크 스택이 있다[5]. OpenOnload는 이 처리를 하드웨어에서 처리하는 프로토콜 스택을 가지고 있다. 그림 1을 보면 일반적으로 네트워크 어댑터에서 애플리케이션과 연결되려면 커널을 통해서 접근이 가능하지만 OpenOnload의 경우에는 네트워크 어댑터와 애플리케이션이 직접 연결을 하여 OpenOnload는 주로 시장 데이터 및 고주파거래 애플리케이션에 사용된다[5,8].



(그림 1) OpenOnload 스택 구조

Windows환경에서 사용하는 NIC의 기능은 리눅스 운영체제에서보다 제한적인 기능과 성능을 보여주며 다음과 같은 기능이 있다.

- TOE(TCP Offload Engnie) 는 NIC에서 전체 TCP/IP 스택처리를 네트워크 컨트롤러로 오프로드 하는데 사용되는 기술이다[4]. 기가비트 이더넷에서 네트워크 스택의 오버헤드 처리에 사용된다. 오프로드는 시스템의 메인 CPU가 다른 작업을 수행할 수 있게 한다. Windows에서 사용되어지는 기능은 일부만 있다. 그 중 본문에서는 CSO(CheckSum Offload), LSO(Large Send Offload)가 있

다. CSO는 체크섬 계산을 시스템 CPU에서 처리하는 것이 아닌 네트워크 어댑터에서 한다. CPU의 작업량을 줄여서 애플리케이션 처리 작업을 집중적으로 할 수 있게 한다. LSO는 TCP데이터를 작은 패킷으로 분할하여 어댑터에서 처리한다. CPU의 처리 성능에 부하를 주지 않고 대기시간에 영향을 미치지 않게 한다.

- 인터럽트 병합은 수신된 다수의 패킷 이벤트나 완료된 이벤트를 단일 인터럽트로 결합한다. 네트워크 어댑터에서 인터럽트의 수를 줄여서 CPU로 전달되는 작업의 수를 줄인다.

- NUMA(Non-Uniform Memory Access)는 프로세서와 메모리가 하나의 그룹으로 이루어진 시스템 구조이다[6,7]. 메모리의 위치에 따라 프로세서가 접근하는 시간이 다르다. 상대적인 위치가 가까울수록 접근 시간은 빨라진다.

- RSS(Receive Side Scaling)은 CPU의 데이터 처리를 동적으로 분배하여 작업의 부하를 분산시키기 위해 사용한다. NIC는 해시 기능을 사용하여 수신 큐에 들어오는 트래픽을 분산시킨다. 각 수신 큐의 대기열은 인터럽트에 할당된다. 인터럽트의 처리를 다른 CPU에서 처리하도록 분산시켜주어 대기시간이 감소하여 CPU의 성능을 향상 시킬 수 있다.

네트워크 저지연 기능이 있는 NIC의 사용유무에 따라 데이터의 속도 차이를 비교하고 그에 따라 상용 NIC를 사용하여 네트워크의 지연을 개선한다. 응용 프로그램은 트레이딩 플랫폼이며 국내 증권사에서 제공하는 OpenAPI+를 사용한다. C# 언어를 사용하여 .NET Framework 4.0에서 실행되는 트레이딩 소프트웨어를 개발하고 주식 주문 및 조회기능의 처리속도를 확인한다. 호스트 응용 프로그램으로서 Windows 서버의 상용 NIC(Network Interface Controller)을 이용하여 네트워크의 지연을 줄이는 것을 목표로 한다.

4. 실험 및 결과

운영체제는 Windows Server 2016과 Windows10을 사용하고 지연시간측정은 hrping 소프트웨어를 사용하였다. NIC는 Solarflare사의 XtremeScale X2522-10GbE와 Realtek사의 PCIe GBE(1GbE)을 사용하였다. 전송매체로서 CAT6 케이블과 광케이블을 사용하였다.

hrping프로그램을 이용하여 각 네트워크 지연을 측정했으며 TCP ping test를 50번 실행하여 산술평균을 했다.

hrPing을 통해 측정한 값은 표1과 같다.

<표 1> 각 NIC TCP hrPing Test (단위: μs)

NIC / Cable	Solarflare	Realtek
CAT6	175μs	393μs

측정시간은 왕복시간(RTT, Round-Trip Time)을 의미하며 ping client로부터 데이터가 출력되어 ping server를 거쳐 다시 클라이언트로 되돌아 온 시간을 말한다. 따라서 보다 정확한 지연 시간은 RTT/2가 된다.

또한, Solarflare NIC은 CAT6 뿐만 아니라 멀티모드 광섬유도 같이 측정을 하였다. 결과는 지연시간은 131μs가 측정되어 Solarflare NIC과 광섬유를 사용했을 경우 지연시간이 최소가 됨을 확인하였다.

5. 결론

Windows 환경에서 네트워크 지연에 영향을 주는 요소들을 분리하여 테스트하였는데 사용자가 개선할 수 있는 부분은 크게 CPU 지연(처리지연)과 네트워크 지연임을 확인하였다. CPU 지연이 네트워크 지연보다 상대적으로 크며 트레이딩 플랫폼의 경우에 클라이언트보다 서버의 영향이 더 크다. 상용 NIC과 구리선 및 광섬유의 전송선을 사용하여 실험하였고 이에 대한 지연시간의 비교를 표로 나타내었다.

향후 네트워크 지연을 보다 정밀하게 세분화해서 측정할 필요가 있다. 또한 Onload기능을 이용한 Linux에서의 네트워크 지연과 트레이딩 플랫폼의 CPU 처리지연에 대한 연구가 필요하다.

사사:이 논문은 2018년도 한국연구재단의 지원을 받아 수행된 연구임(NRF-2016R1D1A1B03930435)

참고문헌

[1] Kuperman, Y, Moscovici, E., Nider, J., Ladelsky, R., Gordon, A., &Tsafrir, D. “Paravirtual remote i/o”. In ACM SIGARCHComputer Architecture News (Vol. 44, No.2, pp. 49-65). 2016.

[2] 한윤정, 이권용, 박성용, "TOE 기술을 적용한 Ceph 분산 스토리지의 성능 평가 및 분석," 한국정보과학회 학술발표논문집, , pp. 1515-1517, 2016.

[3] T. Lukaseder, L. Bradatsch, B. Erb, R. W. van der Heijden, and F. Kargl, “A Comparison of TCP Congestion Control Algorithms in 10G Networks,” 2016 IEEE 41st Conference on Local Computer Networks (LCN 2016), 2016.

[4] David Sidler, Gustavo Alonso, Michaela Blott, Kimon Karras, Kees Vissers and Raymond Carley, “Scalable 10Gbps TCP/IP Stack Architecture for Reconfigurable Hardware”, in Field-Programmable Custom Computing Machines (FCCM), 2015 IEEE 23rd Annual International Symposium on, pp. 36-43, 2015.

[5] Solarflare Communications, Inc [WebSite], https://www.openonload.org/,2018

[6] Sergey Blagodurov, Sergey Zhuravlev, Alexandra Fedorova, and Ali Kamali. “A case for numa-aware contention management on multicore systems. In

International Conference on Parallel Architectures and Compilation Techniques, 2010.
[7] David Ott, Optimizing Applications for NUMA | Intel® Software[WebSite], https://software.intel.com/en-us/articles/optimizing-applications-for-numa, 2011
[8] Noa Zilberman, Matthew Grosvenor, Diana-Andreea Popescu, Neelakandan Manihatty-Bojan, Gianni Antichi, Marcin Wojcik, and Andrew W. Moore. “Where Has My Time Gone?”. In Passive and Active Measurement (PAM). Springer. 2017