그리드 환경에서 사설 IP 클러스터간의 통신을 위한 고성능 중계기법

최시열° 박금례 박성용 권오영† 박형우‡
서강대학교 컴퓨터학과 분산시스템 연구실 한국기술교육대학교† 한국과학기술정보연구원‡
{adore°, namul}@sogang.ac.kr, parksy@ccs.sogang.ac.kr, oykwon@kut.ac.kr†, hwpark@kisti.re.kr‡

High Performance Communication Relay method for Clusters that use Private IPs in GRID environment

Siyoul Choi° Kumrye Park Sungyong Park Ohyoung Kwon† Hyoungwoo Park‡
Distributed Computing and Communication Lab., Dept. of Computer Science, Sogang Univ.
Korea University of Technology and Education†
Korea Institution of Science and Technology Information‡

요 약

본 논문에서는 NAT와 프락시를 조합하여 그리드 환경에 적합한 사설 IP 클러스터의 통신 연결 중계 방안을 제시한다. 기존에 서비스 되고 있는 NAT와 사용자 영역의 프락시를 조합하여 사용하면 프락시만을 사용한 것에 비해 문맥 전환 비용을 절반으로 줄일 수 있고 커널 영역에서만 구현한 것에 비해 이식성을 높일 수 있는 장점이 있다. 성능 측정 결과 25% 정도의 성능 향상이 있었으며 차후에는 이 메커니즘을 MPICH-G2에 이식하여 사설 IP를 사용한 글로버스 시스템의 구축이 가능하도록 할 것이다.

1. 서 론

그리드 컴퓨팅은 지역적으로 떨어져 있는 자원(슈퍼컴퓨터, 클러스터 등)들을 통합하는 기술에 관한 연구이다[1]. 자원을 통합하기 위해서는 주로 글로버스 툴킷[2]이 사용되며 그 위에서 수행되는 응용 프로그램의 통신을 위해서 MPICH-G2[3,4]가 사용된다. MPICH-G2를 사용하여 클러스터끼리 통신을 하려면 계산노드의 IP를 기존의 사설 IP(Private IP)에서 공인 IP(Public IP)로 변경해야 하는데, 이는 사설 IP를 사용하면 클러스터 계산노드 간의 직접 통신을 할 수 없기 때문이며, 보안 등의 여러 가지 사항에 문제를 야기 시킨다.

이 문제를 해결하기 위해서는 사설 IP로 설정된 두 클러스터의 계산 노드들이 서로 통신을 할 수 있도록 해주는 메커니즘이 필요하다. 이러한 메커니즘들은 크게 NAT[5], Gateway, Proxy[6,7] 방법으로 나눌 수 있는데 크게 구현영역, 내부통신 수단, 프론트 노드 간 연결의 수와 같은 특징을 갖는다. 앞의 두 가지는 커널 영역에서 구현되어 있기 때문에 사용자 영역에서 구현된 프락시 보다 두 배 정도 성능이 좋지만[8] 커널을 수정해야 하기 때문에 이식성이 떨어진다.

본문에서는 NAT와 프락시를 조합하여 그리드 환경에 적합한 통신 연결 중계 방안을 제시한다. 기존에 서비스 되고 있는 NAT와 사용자 영역의 프락시를 조합하여 사용하면 프락시만을 사용한 것에 비해 문맥 전환 비용을 절반으로 줄일 수 있고 커널 영역에서만 구현한 것에 비해 이식성을 높일 수 있는 장점이 있다. 차후에는 이 메커니즘을 MPICH-G2에 이식하여 사설 IP를 사용한 글로버스 시스템의 구축이 가능하도록 할 것이다.

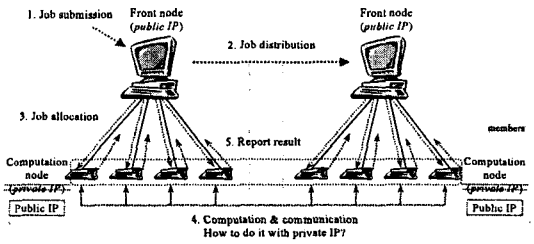
본 논문의 구성은 다음과 같다. 2장에서는 그리드 환경에서 MPI 프로그램을 수행할 때의 문제점과 이를 해결하기 위한 기존의 방안들을 소개한다. 3장에서는 그리드 환경에 적합한 통신 중계 방안을 제시하며 4장에서는 기존 통신방법과 비교 분석하고 마지막으로 5장에서는 결론 및 향후과제를 제시한다.

2. 그리드 환경에서 MPI 실행의 문제점과 기존 연구

2.1. 그리드 환경의 클러스터에서 MPI 실행의 문제점

클러스터 내에서 MPI 프로그램이 수행될 경우 프론트 노드는 각 계산 노드들에게 일을 분배하며, 각 계산노드들은 분배된 일을 수행하는 과정 중에 서로 통신을 하게 된다. 클러스터 내부에서만 통신이 이루어지기 때문에 계산노드 들의 사설 IP는 문제가 되지 않는다.

그러나 사설 IP는 그리드 환경으로 넘어가면 문제가 될 수 있다. [그림 1]은 글로버스를 설치한 두 대의 클러스터에서 MPI 프로그램을 수행시키는 과정이다. 과정은 동일하지만 여기서 사설 IP만을 가지고는 한쪽 클러스터의 계산 노드가 다른 쪽 클러스터의 계산노드와 통신을 할 수 없다. 그렇기 때문에 현재의 글로버스는 [그림 1]과 같이 모든 계산노드에도 공인 IP를 할당하여 계



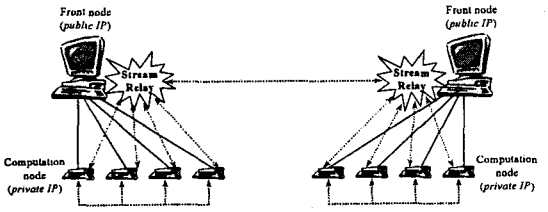
[그림 1] 글로버스를 설치한 두 클러스터에서 MPI 수행 과정

산노드끼리 직접 통신을 하고 있다.

이러한 설정은 문제를 간단히 해결할 수 있지만 계산 노드가 외부 네트워크에 노출되어 보안이 취약하고 계산 노드 간에 그물(mesh) 형태의 연결이 형성되기 때문에 계산에 참여하는 노드의 수가 증가하면 연결의 수가 급속히 증가한다는 약점이 있다. 따라서 사설 IP를 사용하면서도 각 클러스터의 계산노드들이 통신을 할 수 있는 환경이 구축되어야만 한다.

2.2. 기존 연구

사설 IP를 갖고 있는 계산 노드들이 다른 클러스터의 계산노드와 통신을 하기 위해서는 외부로 나갈 수 있는 경로가 필요하고 이를 위해서는 [그림 2]와 같이 프론트 노드에 연결을 중계시켜 줄 수 있는 데몬이 필요하다.



[그림 2] 사설 IP로 설정된 계산노드들 간의 통신을 위한 연결 중계

[표 1]은 계산노드의 통신 연결을 중계해 줄 수 있는 기존의 방법을 간략하게 정리한 것이다. 각각의 방법은 기본적으로 프론트 노드에 설치되며 사설 IP를 가진 계산노드를 외부 네트워크로 중계해 주는 역할을 한다.

프론트 노드간 연결의 수는 단일 연결일 경우 계산노드의 연결과는 상관없이 모든 데이터를 하나의 연결에 통합하여 전송하며 다중일 경우는 계산노드의 연결 당 하나의 연결을 생성하여 전송한다. 내부 통신의 경우 벤더가 제공하는 빠른 속도의 MPI를 사용하거나 TCP 연결을 사용할 수 있으며 구현 영역의 경우 커널 영역에서 구현된 것과 사용자 영역에서 구현된 것이 있다.

[1] 기존의 통신 연결 중계 방법 및 각각의 특성

구분	프론트 노드간 연결의 수	클러스터 내부 통신	구현영역
NAT	다중	TCP	커널
Gateway	단일	벤더 MPI	커널
Proxy	단일	TCP	사용자

각 방법은 이러한 특성에 따라 다른 성능을 보이는데 성능에 가장 영향을 많이 미치는 특성은 구현 영역이다. NAT와 Gateway는 커널 영역에서 구현되어 있기 때문에 Proxy보다 대역폭과 지연시간 측정에서 두 배 정도 좋은 성능을 보여주었다[8]. 이는 커널-사용자 영역을 넘으며 발생하는 문맥 전환 비용과 TCP/IP 스택을 거치는 비용이 크기 때문이다. 그럼에도 불구하고 프락시 방법은 PACX-MPI[6]에서 사설 IP문제를 해결하기 위해 사용되었고, 또한 방화벽 환경에서 MPICH-G를 사용하기 위해 사용되었다[7].

3. NAT와 Proxy를 병행한 고성능 중계 기법

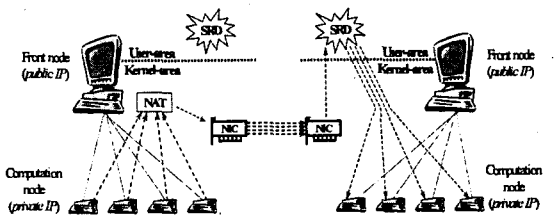
그리드 환경은 단순히 클러스터들을 연동하는 것과는 다른 환경이기 때문에 이러한 방법은 직접 적용하기 보다는 그리드 환경에 맞는 적절한 방법으로 수정해야 한다. 그리드 환경에 따른 [표 1]의 특성에서 고려할 점은 다음과 같다.

- 프론트 노드간 연결의 수
  그리드의 자원들은 지역적으로 멀리 떨어져 있다. 지역적으로 떨어져 있는 자원들 사이의 TCP 통신은 다중 스트림을 사용하는 것이 좋다. 이는 TCP의 고정된 윈도우 크기 때문이다[9].
- 클러스터 내부 통신
  외부 통신에만 TCP를 사용하고 내부 통신에는 벤더가 제공하는 빠른 통신수단을 사용하는 것이 좋다.
- 구현 영역
  커널 영역의 구현이 좋은 성능을 보이지만 글로버스 툴킷의 이식성을 고려하면 커널의 수정은 피하는 것이 좋다.

이러한 특성을 종합하면 그리드 환경에서는 성능을 위해서 중계 데몬이 커널에서 구현되어 있어야 하지만 글로버스의 이식성을 위해서는 커널이 수정되는 것은 피해야 한다는 결론을 내릴 수 있다. 따라서 본 논문에서는 커널 영역의 방법과 사용자 영역의 방법을 통합하여 NAT와 프락시를 결합한 이식성 높은 고성능 통신 중계 방안을 제시한다.

[그림 3]은 이 방법을 설명한다. 각각의 클러스터는 프론트 노드에 NAT 서비스를 활성화 시켜야 한다. 이는 루트의 ipchains 명령을 통해 이루어지며 커널 컴파일 및 재부팅 없이 활성화 될 수 있다. 또한 사용자 영역에 있는 SRD(Stream Relay Daemon)는 다른 쪽 클러스터에서 전달된 스트림을 자신의 계산노드에게 중계해 주는 역할을 한다. 한쪽 클러스터의 계산 노드는 프론트 노드의 NAT 서비스를 통하여 다른 쪽 클러스터의 프론트 노드에 연결할 수 있으며 다른 쪽 클러스터의 프락시가 이를 받아들여 자신의 계산노드에게 연결 해준다.

이는 커널 영역에서만 구현된 것에 비하면 성능 저하가 있을 수 있지만 커널이 수정되지 않는다는 장점이 있으며 사용자 영역에서만 구현된 것에 비하면 한번의 문맥 전환 비용과 TCP/IP 스택을 거치는 비용이 줄어들어 성능상의 이점이 있을 수 있다.

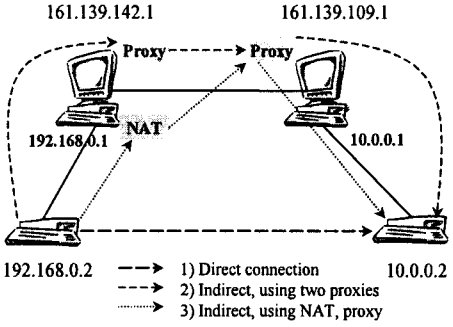


[그림 3] NAT와 Proxy를 병행한 연결 중계 방법

4. 성능 측정

4.1. 측정 환경

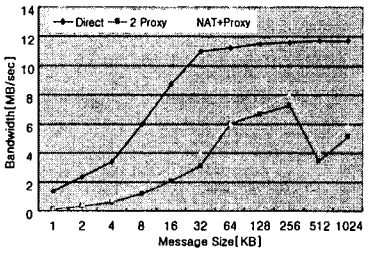
성능 측정을 위한 환경은 [그림 4]와 같다. 순수하게 방법에 따른 성능차이 만을 비교하기 위하여 사설 IP를 가진 클러스터의 환경을 하나의 프론트 노드와 하나의 계산노드로 간단히 구성하였다. 측정할 값은 각 계산노드끼리의 통신 지연시간과 대역폭이며 통신 방법은 계산노드 간의 1) 직접 통신, 2) 두 프락시를 사용한 간접통신, 3) NAT와 프락시를 사용한 간접 통신으로 하였다. 모든 NIC은 100MB 이더넷을 사용하였다.



[그림 4] 성능 측정 환경

4.2. 통신 지연시간, 대역폭 측정 결과 및 분석

측정된 지연시간과 대역폭은 [그림 5]와 같다. 지연시간의 경우 1) 두 프락시를 사용한 방법과 2) NAT · 프락시를 같이 사용한 방법은 직접 연결에 비해 2.1배, 1.5배 정도 높은 값을 보여 2)가 1)보다 28% 정도 낮았다. 대역폭의 경우 1), 2) 모두 두 번의 중계를 거치기 때문에 직접 연결보다 절반 이하의 성능을 보였지만 1~32KB의 작은 메시지의 경우, 2)가 1)보다 18~27% 정도 높은 값을 보였고, 128KB~1MB의 큰 메시지의 경우는 6~16%정도 높은 값을 보였다.



	Latency[㎲]
Direct	1550
2 Proxy	3284
NAT+Proxy	2372

[그림 5] 각 방법의 대역폭 및 지연시간 측정결과

이러한 결과는 그리드 환경의 특성을 고려하면 두 클러스터의 거리가 멀어짐에 따라 프론트 노드 간의 지연시간이 커지기 때문에 지연시간의 차이는 무시할 수 있다. 또한 대역폭의 경우, 두 번의 중계과정을 거치면서 발생하는 손실은 어쩔 수 없지만 적절한 크기의 메시지를 사용할 때 2)의 방법을 사용하면 1)의 방법에 비해 약 25% 정도의 성능 향상을 얻을 수 있다.

5. 결론 및 향후 과제

본 논문에서는 그리드 환경에서 사설 IP를 사용한 클러스터의 통신 중계 기법을 살펴보고 NAT와 프락시를 사용한 대안을 제시하였다. 성능 측정 결과 NAT와 프락시를 사용한 방법은 기존의 사용자 영역의 기법보다 25%정도의 성능 향상이 있었으며 이는 커널 영역의 기법보다는 성능이 낮지만 커널을 수정하지 않는 다는 점에서 유리하다.

차후에는 본 논문이 제시한 방법을 그리드 환경에 이식할 예정이며, 이를 위해서 글로버스 툴킷의 통신 라이브러리인 MPICH-G2를 수정하여 사설 IP를 사용한 클러스터를 연동할 수 있도록 할 것이다.

참고문헌

[1] I. Foster, C. Kesselman and S. Tuecke. The Anatomy of the grid : Enabling scalable virtual organizations. *International Journals of Supercomputing Applications*, 15(3), 2001.

[2] I. Foster and C. Kesselman. *The Grid : A Blueprint for a New Computing Infrastructure*. Morgan Kaufmann, 1998.

[3] W. Gropp, E. Lusk, and A. Skjellum. Using MPI Portable Parallel Programming with the Message-Passing Interface. *Parallel Computting.*, 22:789-828, 1996

[4] I. Foster, J. Geisler, W. Gropp, N. Karonis, E. Lusk, G. Thiruvathukal, and S. Tueche, Wide-area implementation of the message passing standard. *Parallel Computing*, 24, 1998

[5] P. Srisuresh and K. Egevang. *Traditional IP Network Address Translator(Traditional NAT), RFC 3022.* International Engineering Task Force, Jan 2001

[6] E. Gabriel, M. Resch, T. Beisel and R. Keller, Distributed computing in a heterogeneous computing environment, In *Recent Advances in Parallel Virtual Machine and Message Passing Interface*, Lecture Nodes in computer Science. Springer, 1998

[7] Y. Tanaka, M. Sato, M. Hirano, H. Nakada, and S. Sekiguchi. Performance evaluation of a firewall-complaint globus-based wide-area cluster system. In *Proceedings of the Ninth IEEE International Symposium on High Performance Distributed Computing*, pages 121-128. IEEE Computing Society, 2000

[8] M. Mller, M. Hess, E. Gabriel, Grid enabled MPI solutions for Clusters, In *3rd International Symposium on Cluster Computing and the Grid*, 2003

[9] H. Sivakumar, S. Bailey, and R. Grossman, PSockets: The Case for Application-level Network Striping for Data Intensive Applications using high Speed Wide Area Networks, *SC2000: High-Performance Network and Computing Conference*, Dallas, TX, 11/00