우수한 네트워크 부하 분배 특성을 가지는 이중 연결 CC-NUMA 시스템

서효중°
가톨릭대학교 컴퓨터정보공학부
hjsuh°@catholic.ac.kr

An Effective Load Balanced Dual-link CC-NUMA System

Hyo-Joong Suh°
School of Computer Science and Information Engineering
The Catholic University of Korea

요 약

CC-NUMA 시스템은 메모리를 분산시켜 트랜잭션을 지역화함으로써 고성능 및 확장성을 꾀하는 구조이다. 그러나 CC-NUMA 시스템에서 여러 병렬 프로그램들이 수행될 경우, 각 프로그램의 부하 차이에 의하여 균등한 네트워크 활용을을 나타내지 못하며, 이중 링 CC-NUMA 시스템에서 이러한 불균등한 네트워크 부하로 인한 성능 감소가 발생한다. 본 논문은 이중 연결 구조중 하나를 건너뛰 연결을 갖도록 배치하여 균등한 네트워크 부하를 나타내도록 하며, 이중 링에 비하여 균등한 네트워크 부하를 나타냄을 시뮬레이션을 통하여 검증한다.

1. 서 론

CC-NUMA(Cache-Coherent Non Uniform Memory Access) 시스템은 공유 메모리를 분할하여 각 프로세서에 가깝게 배치하고, 프로세서로부터 발생하는 메모리 접근을 가능한 가까운 지역에서 제공함으로써 보다 높은 확장성과 고성능을 얻을 수 있는 구조이다. 따라서 CC-NUMA 시스템의 성능은 가능한 높은 지역성을 얻을수록 고성능을 나타내며, 높은 지역성을 얻기 위해 원격 캐시 등을 채용하고 있다[1].

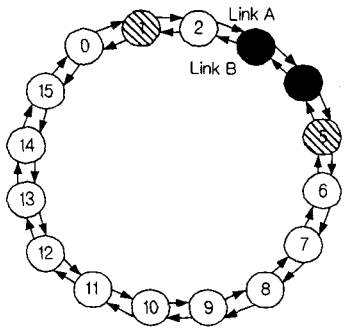
SCI(Scalable Coherent Interface)[2]등 점대 점 연결 구조는 높은 대역폭과 적은 지연으로 IBM 의 NUMAQ[3] 등의 시스템에 적용되었으며, Data General의 AViiON[4], 서울대학교의 PANDA-2 시스템[5]에서 이중 링 구조로 확장된 바 있다. 그러나, 노드간 트랜잭션 전달 경로가 점대 점 연결을 경유하므로, 연결 경로상 균등하지 못한 높은 경쟁을 보이는 링크가 발생할 경우, 이 링크에 대한 경쟁으로 인하여 전체 시스템 성능이 저하된다.

본 논문은 이중 링 CC-NUMA 구조가 이러한 불균등한 부하로 인한 성능 저하가 발생함을 보이고, 이중 링 구조와 동일한 네트워크 비용을 소모하면서도 균일한 부하 분배를 보이는 형태로 연결을 구성한 네트워크 구조를 제시하며, 프로그램 구동형 시뮬레이션을 통하여 제시한 구조의 네트워크 부하 분배가 균등하게 발생하고, 이중 링 구조에 비하여 낮은 네트워크 경쟁과 높은 성능을 보임을 제시한다.

논문의 구성은 2 장에서 이중 링 구조와 건너뜀 구조를 갖는 CC-NUMA 시스템을 비교하고, 3 장에서 시뮬레이션 환경 및 부하를 설명하며, 4 장에서 실험 결과를 제시하고, 5 장에서 결론을 맺는다.

2. 이중 연결 CC-NUMA 시스템

다음 그림 1은 Data General의 AViiON, 서울대학교 PANDA-2 시스템과 같은 이중 링 구조의 CC-NUMA 시스템 이다. 점대 점 링크는 SCI연결을 사용하였으며, 각 노드간에 반대방향의 링크를 이용하여 대역폭과 트랜잭션 전송 경로 길이를 단축한 것이다.

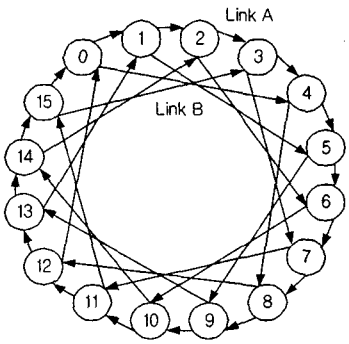


<그림 1> 이중 링 구조의 CC-NUMA 시스템

그러나 이와 같은 구조에서 3 번 과 4 번 노드간에 많은 트랜잭션이 집중되고, 1 번과 5 번 노드간에 트랜잭션의 전송이 일어날 경우, 1 번과 5 번 노드간의 트랜잭션 전송 경로는 경쟁이 심한 3 번과 4 번 사이의 링크를 통과하여

야 하기 때문에 경쟁에 의한 지연이 발생한다.

그림 2 은 본 논문에서 제시하는 건너뜀 링크를 갖는 구조로 방송 트랜잭션과 일대 일 트랜잭션의 적절한 전송 경로 길이에 따라 적당한 건너뜀 링크로 연결된 것이다.



<그림 2> 4 건너뜀을 갖는 이중 연결 구조의 시스템

이중 링 연결에 대하여 건너뜀 링크를 갖는 구조는 방송 트랜잭션과 일대 일 트랜잭션에서 다음과 같은 전송 경로 단계 개선을 나타낸다.

N : 총 노드의 개수
l : 건너뜀 수
d : 이중 링 구조의 방송 트랜잭션 전송 경로
d' : 건너뜀 구조의 방송 트랜잭션 전송 경로
D : 이중 링 구조의 일대 일 트랜잭션 전송 경로
D' : 건너뜀 구조의 일대 일 트랜잭션 전송 경로

$d - d' = N - N/l - l + 1 = N - (N/l + l) + 1$
$D - D' = N(N - 4N/l + 6)/8$

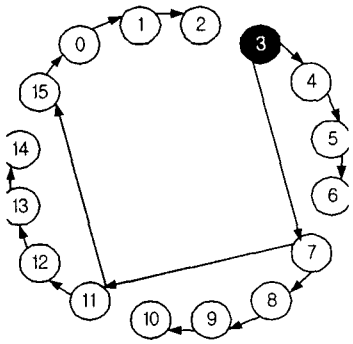
이 수식에 의하여 적절한 건너뜀 길이를 설정하고 방송 트랜잭션과 일대 일 트랜잭션은 정적인 경로에 따라 최단 경로로 트랜잭션을 전송하게 된다. 그림 3은 방송 트랜잭션과 일대 일 트랜잭션이 전송되는 경로를 나타낸 것이다.

3. 시뮬레이션 환경 및 부하

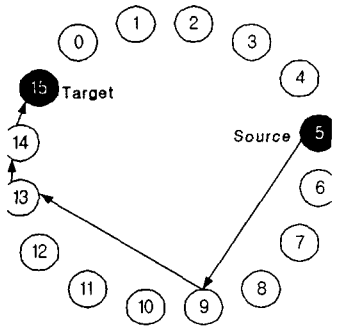
앞서 제시한 이중 링 구조의 CC-NUMA 시스템과 건너뜀 연결을 갖는 CC-NUMA 시스템을 프로그램 구동형 시뮬레이터인 Augmint를 이용하여 시험하였다[6]. Augmint는 Mint를 기반으로 x86 기계의 프로그램을 사용할 수 있도록 확장한 다중 프로세서 시뮬레이터로써 시험 대상의 프로그램에 삽입되어 메모리 접근을 추적할 수 있다.

성능 평가에 사용한 프로그램은 다중 프로세서 성능측정에 다수 사용되는 SPLASH-2 벤치마크[7] 프로그램중 세 개를 사용하였으며, 표 1와 같은 부하를 가지도록 하였다.

표 2 는 실험 대상 시스템 인수들로, 총 32 프로세서 /32 노드에서 각 연결은 SCI 링크로 구성하였다. 건너뜀 연결 구조에서 건너뜀 수는 앞서 제시한 수식에 따라서



(a) 방송 트랜잭션 전송 경로



(b) 일대 일 트랜잭션 전송 경로

<그림 3> 트랜잭션 전송 경로

<표 1>SPLASH-2 프로그램 부하

프로그램	부하
FFT	-m14 -p8 -n2048 -l5
LU	-n128 -p8 -b16
RADIX	-p8 -n9000 -r1024 -m2097152

방송 트랜잭션과 일대 일 트랜잭션에서 가장 짧은 경로를 나타내는 8 건너뜀을 적용하였다.

<표 2> 시스템 인수

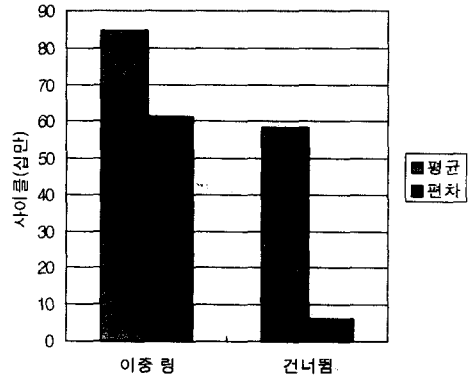
항목	값
프로세서 클럭 속도	1 GHz
시스템의 프로세서 수	32 개
프로세서 당 캐시 크기	128 Kbyte
프로세서 캐시 연관	4 way
노드 개수	32 개
노드 내 버스 속도	266 MHz
노드 당 원격 캐시 크기	512 Kbyte
원격 캐시 연관	8 way
링크 전송 대역폭	1Gbyte/s
캐시 교체 정책	최근 최소 사용(LRU)

4. 시뮬레이션 결과

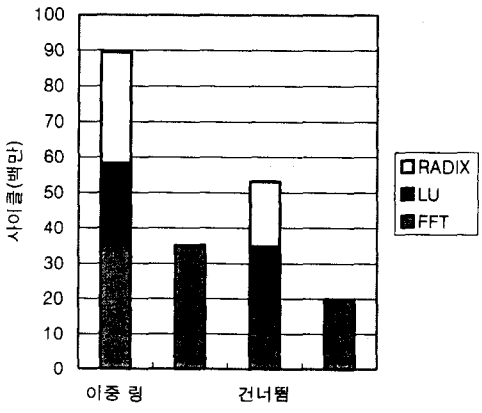
그림 4는 세 개의 프로그램을 동시에 수행시키고, 각 프로그램의 프로세스를 난수적으로 배치시켰을 때, 이중 링 구조와 건너뜀 구조에서의 각 링크에 나타난 링크 점유 시간의 평균값과 편차이다.

그림에서 나타난 것처럼 이중 링 구조에 대비하여 건너뜀 연결을 갖는 구조는 적어진 링크 점유 시간과 적은 편차를 나타냈다. 이는 건너뜀 링크에 의한 보다 짧아진 트랜잭션 전송 경로 길이에 따른 효율적인 링크 사용과 더불어 적절한 링크 부하 분배로 모든 링크가 골고루 사용되었음을 의미한다.

그림 5은 이중 링 구조와 건너뜀 연결 구조에서의 프로그램 수행시간을 나타낸 것이다. 세 개의 프로그램이 동시에 수행되므로, 왼쪽 막대는 각 프로그램이 수행한 시간을 누적하여 표시한 것이며, 오른쪽 막대는 세 개의 프로그램이 모두 종료한 시간을 나타낸 것이다.



<그림 4> 링크의 평균 점유 시간 및 편차



<그림 5> 프로그램 수행 시간

앞서 제시한 링크에 대한 점유 및 부하분배에서 건너뜀 링크를 갖는 구조가 짧은 트랜잭션 경로 길이 및 고른 부하 분배를 나타내므로, 프로그램 수행 시간도 1.5 배 이상의 성능 개선을 보여주고 있다.

5. 결론

이중 링 CC-NUMA 시스템은 고속의 SCI 점대 점 링크와 NUMA 시스템의 지역성에 따른 고성능을 얻고자 하는 구조이다. 그러나 연결 경로상의 불균일한 부하는 시스템 성능의 저하를 일으키며 특정 링크에 대한 병목 현상을 발생시킨다. 본 논문에서 제시한 건너뜀 구조는 방송 트랜잭션과 일대 일 트랜잭션에 대하여 효율적인 경로 설정을 제시하였으며, 링크에 대한 고른 사용이 이루어지도록 함으로써 성능 향상을 보였다.

참고문헌

[1] D. Lenoski, etal., " The Stanford Dash multiprocessor" , Computer, Vol. 25 No.3, pp.63-79, Mar. 1992.

[2] IEEE Computer Society, IEEE Standard for Scalable Coherent Interface(SCI), Institute of Electrical and Electronics Engineers, Aug. 1993.

[3] T. Lovett, R. Clapp, "STiNG : A CC-NUMA Computer System for the Commercial Marketplace", Proc. of the 23th International Symp. on Computer Architecture, pp. 308-317, May 1996.

[4] http://www.dg.com/

[5] http://panda.snu.ac.kr/nrl/

[6] A-T. Nguyen, etal.," The Augmint multiprocessor simulation toolkit for Intel x86 architecture" , Proc. of the IEEE International Conf. on Computer Design, Oct. 1996.

[7] S.C.Woo, etal., " Methodological considerations and characterization of the SPLASH-2 parallel application suite" , Proc. of the 22th Annual International Symp. on Computer Architecture, pp24-36, 1995.

서효중

1991년 서울대학교 학사
1994년 서울대학교 컴퓨터공학석사
2000년 서울대학교 컴퓨터공학박사
2002년(주)지씨티리서치 선임연구원
2003년 현재 서울대학교 컴퓨터연구소 객원연구원
2003년 현재 가톨릭대학교 컴퓨터정보공학부 전임강사
관심분야 : 컴퓨터 구조, 병렬처리 시스템, 내장형 시스템, 클러스터 시스템