

클러스터 파일 시스템을 위한 LDPC 코드 복제 기법 최적화

김세희*, 이원주°, 전창호*

*한양대학교 컴퓨터공학과,

°인하공업전문대학 컴퓨터정보과

e-mail: kaiser38@hanyang.ac.kr, wonjoo2@inhac.ac.kr, chj5193@hanyang.ac.kr

Optimization of LDPC Code Replication Scheme for Cluster File System

Se-Hoe Kim*, Won Joo Lee°, Chang-Ho Jeon*

*Dept. of Computer Science & Engineering, Hanyang University,

°Dept. of Computer Science, Inha Technical College

● 요약 ●

최근 이슈가 되고 있는 클라우드 컴퓨팅은 대용량의 데이터를 분산 저장하고 제공할 수 있는 클러스터 파일 시스템을 필요로 한다. 이러한 클러스터 파일 시스템은 높은 신뢰성과 고가용성을 보장하기 위해서 파일 복제 기법을 사용하고 있다. 가장 많이 쓰이고 있는 복제 기법은 전체-파일 복제 기법으로 높은 파일 가용성을 제공하지만 그만큼 스토리지 오버헤드가 크다는 단점이 있다. 또 다른 복제 기법으로는 LDPC 코드를 이용한 것으로 비교적 적은 스토리지 오버헤드를 가지면서 동시에 비슷한 수준의 파일 가용성을 제공한다. 따라서 본 논문에서는 클러스터 파일 시스템을 위한 LDPC 코드 복제 기법의 최적화 방법을 제안한다.

키워드: 클러스터 파일 시스템(Cluster file system), 복제 기법(Replication scheme), LDPC 코드

I. 서론

최근 인터넷과 통신 기기의 발달로 사용자들에게 언제 어디서든지 원하는 서비스를 사용할 수 있도록 제공하는 클라우드 컴퓨팅이 대두되었다. 이러한 클라우드 컴퓨팅은 대용량의 데이터를 분산 저장하고 제공할 수 있는 클러스터 파일 시스템을 필요로 한다. 또한 클러스터 파일 시스템에서는 시스템의 신뢰성과 가용성을 보장하기 위해 파일 복제 기법을 사용하고 있다.

본 논문에서는 기존 클러스터 파일 시스템에서 많이 사용하고 있는 전체-파일 복제 기법(Whole-file replication)과 최근에 연구되고 있는 LDPC(Low density parity check) 코드를 이용한 복제 기법에 대해 알아본다. 또한 클러스터 파일 시스템을 위한 LDPC 코드 복제 기법의 최적화 방법을 제안한다.

가장 많이 쓰이는 복제 기법으로, 원본 파일에 대해서 정해진 수만큼의 복제본을 추가로 저장한다. 또한 기본적으로 2~3개의 복제본을 저장하므로 고가용성을 보장하지만 동시에 스토리지 오버헤드가 비교적 크다는 단점이 있다.

2.2 LDPC 코드

LDPC 코드[2]는 오류정정코드로 사용되는 erasure 코드의 하나이다. 그림 1과 같이 (n, m) LDPC 코드는 n 개의 왼쪽 노드와 m 개의 오른쪽 노드를 가지는 Tanner 그래프로 정의한다. 이때 각 m 개의 오른쪽 노드는 연결된 왼쪽 노드들의 XOR 연산으로 생성한다.

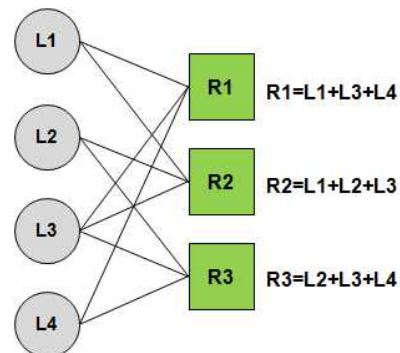


그림 1. (4, 3) LDPC 코드의 Tanner 그래프

II. 관련 연구

2.1 클러스터 파일 시스템

클러스터 파일 시스템은 LAN 기반의 분산 파일 시스템으로 Google의 GFS[1], Hadoop의 HDFS, Ceph 등이 있다. 이러한 클러스터 파일 시스템에서는 데이터가 여러 노드에 분산 저장된다. 또한 노드 실패와 같은 장애에 대비하여 복제 기법을 사용한다. 그 중에서 전체-파일 복제 기법은 클러스터 파일 시스템에서

II. LDPC 코드 복제 기법

LDPC 코드 복제 기법에 대한 연구는 주로 P2P 파일 시스템에서 연구되었으며^[3] 클러스터 파일 시스템에서는 아직 활발히 연구되지 않았다^[4]. LDPC 코드 복제 기법에 사용되는 용어는 표 1과 같다.

표 1. 용어 정리

용어	심볼
데이터 블록 수(number of data blocks)	n
코딩 블록 수(number of coding blocks)	m
부호화율(Code rate)	$R = n/(n+m)$
스토리지 오버헤드(Storage overhead)	$S = (n+m)/n$
노드 간선 수에 대한 확률분포 (Edge degree distribution)	\mathcal{A}
	\mathcal{P}
평균 다운로드 블록 수(Overhead)	o
오버헤드 요인(Overhead factor)	$f = o/n$
노드 가용성(Peer availability)	μ
파일 가용성(File availability)	A

LDPC 코드 복제 기법은 입력 파일을 총 $(n+m)$ 개 블록으로 분산 저장하고, 이 중에서 원본 파일 복구를 위해 평균 $(f * n)$ 개 블록을 필요로 한다. 따라서 오버헤드 요인 f 가 1에 가까울수록 LDPC 코드 복제 기법의 성능은 좋으며, 이러한 성능은 n 과 m , 사용할 코드의 종류 그리고 그래프 정규성 등에 의해 달라진다.

먼저 n 과 m 을 결정하기 위해 파일 가용성(A)와 스토리지 오버헤드(S)에 대해서 분석한다. 전체-파일 복제 기법에서는 $f = 1.1$, $\mu = 0.95$ 일 때 $S = 3$, $A \approx 0.9999875$ 이다^[3]. LDPC 코드에서는 n 과 m 에 따라 A 와 S 가 변한다. 따라서 다양한 n 과 m 에 대해서 A 와 S 를 분석한 결과, 전체-파일 복제 기법보다 더 높은 파일 가용성($S \leq 1.8$)을 가지면서 동시에 적은 스토리지 오버헤드($A \geq 0.99994$)를 가지는 (n, m) 쌍들을 발견하였다.

대표적인 LDPC 코드 종류에는 Systematic 코드, Gallager 코드, IRA 코드 등이 있다^[5]. 이 중에서 Systematic 코드는 $n < 100$ 일 때 가장 낮은 f 를 가지며, 반대로 $n \geq 100$ 일 때는 가장 높은 f 를 가진다^[5]. 따라서 LDPC 코드의 n 에 따라 코드 종류를 선택해야 한다.

그래프 정규성에는 좌-정규(left-regular)와 우-정규(right-regular)가 있다. 좌-정규란 그래프의 왼쪽 노드들의 간선 수가 모두 동일한

것을 의미하며, 반대로 우-정규란 그래프의 오른쪽 노드들의 간선 수가 모두 동일한 것을 의미한다. 이러한 그래프의 정규성은 \mathcal{A} , \mathcal{P} 에 의해 결정되며, LDPC 코드 복제 기법의 성능을 나타내는 f 에 미치는 영향에 대해서는 아직 연구가 미흡하다.

IV. 결론 및 향후 연구 방향

본 논문에서는 클라우드 컴퓨팅에서 사용되는 클러스터 파일 시스템을 위한 LDPC 코드 복제 기법의 최적화 방법을 제안하였다. 기존의 전체-파일 복제 기법보다 높은 파일 가용성을 가지면서 동시에 낮은 스토리지 오버헤드를 가지는 LDPC 코드의 n 과 m 값을 찾고, 이러한 n 과 m 값에 대해서 어떤 종류의 LDPC 코드가 적합한지 알아보았다. 이러한 연구 결과를 이용하여 향후 다양한 그래프를 생성하고, 최적의 오버헤드 요인 f 를 가지는 LDPC 코드를 찾는다. 또한 그래프 정규성이 LDPC 코드 복제 기법의 성능에 미치는 영향에 대해 연구할 계획이다.

참고문헌

- [1] S. Ghemawat, H. Gobioff, S.T. Leung, "The Google file system," ACM SIGOPS Operating Systems Review, Vol. 37, No. 5, pp. 29-43, Dec. 2003.
- [2] R. G. Gallager, "Low Density Parity-Check Codes," MIT Press, Cambridge, MA, 1963.
- [3] B. Gaidioz, B. Koblitz, N. Santos, "Exploring high performance distributed file storage using LDPC codes," Parallel Computing, Vol. 33, No. 4-5, pp. 264-274, Feb. 2007.
- [4] X. Li, C. Xie, Q. Wei, and Q. Cao, "A Reliable Scheme for Cluster Storage System," In Proc. of 3th International Conference on Semantics, Knowledge and Grid, IEEE, pp. 394-397, Oct. 2007.
- [5] J. S. Plank and M. G. Thomason, "A practical analysis of low-density parity-check erasure codes for wide-area storage applications," In International Conference on Dependable Systems and Networks, IEEE, pp. 115-124, June 2004.