

# 오류정정부호를 이용한 실용적 분산 비디오 부호화 기술

김종환, 김상효  
성균관대학교

## 요약

오류정정부호를 이용한 분산 비디오 부호화 기술에 대하여 소개한다. 먼저 분산 소스 부호화 기술을 소개하고, 이것이 어떻게 비디오 부호화에 실용적인 터보부호와 LDPC부호를 이용하여 구현되었는지 설명한다.

## I. 서론

비디오 부호화 기술은 전통적으로 매우 복잡한 부호화 과정과 비교적 간단한 복호 과정으로 특징지어져 왔다. 그런데 2000년대 초반에 분산 비디오 부호화라 불리는 새로운 비디오 부호화 기술이 소개 되었다[1][2][3].

분산 비디오 부호화 기술은 MPEG과 같은 기존의 비디오 부호화 기술과 달리 아주 간단한 부호화와 상대적으로 복잡한 복호기를 특징으로 주목을 받았다. 특히 사용자 기반 콘텐츠의 유행, 센서 네트워크에서의 비디오 검출의 중요성과 같이 낮은 계산 능력을 가진 객체가 비디오 부호화의 요구를 받는 상황에서 좋은 대안으로 떠올랐다[4].

분산 비디오 부호화 기술은 이외에도 독특한 특징을 갖는데, 이론적으로 네트워크 정보이론의 Slepian-Wolf(SW)[5] 및 Wyner-Ziv(WZ)[6]의 분산 소스 부호화에 기반을 두고 있으며, 실용적인 구현에서는 터보 부호나 LDPC(low density parity check) 부호와 같은 강력한 오류정정부호 기술을 이용하여 구현되어 왔다는 점이다.

본 고에서는 오류정정부호의 응용 관점에서 분산 비디오 부호화 기술을 소개한다. 우선 가장 신뢰도가 높은 좋은 구조로 평가 받는 [7]의 변환영역 분산 비디오 부호화를 소개하고, 특히 터보부호나 LDPC부호가 어떻게 분산 소스 부호화 및 비디오 부호화 기술에 응용되었는지를 소개하고자 한다.

## II. 분산 소스 부호화 및 비디오 압축

분산 비디오 부호화 기술은 분산 소스 부호화 기술에 기반을 두고 있다. 이 장에서는 SW 무손실 분산 소스 부호화[5], WZ 손실 분산 소스 부호화[6] 기술을 소개하고, 이를 이용한 분산 비디오 부호화 기술을 소개한다.

### 1. SW 분산 소스 부호화

1973년 Slepian과 Wolf는 상호 의존적인 2개의 의사불규칙 소스  $X, Y$ 에 대해서, 서로 협력 없이 부호화를 하더라도, 최적의 압축률로 부호화가 가능하다는 것을 보였다[5].

〈그림 1〉은 SW 부호화의 기본적인 구조를 나타낸다. 우선  $X, Y$ 라는 데이터를 생성시키는 두 개의 랜덤 소스가 존재하고, 그들은 서로 상관관계가 있다고 가정한다. 부호화는 서로 독립적으로 이루어져서 부호어  $C_X, C_Y$ 를 생성하는데 이 부호어의 데이터율을 각각  $R_X, R_Y$ 라 한다. 부호화기에서  $X, Y$ 의 상관도에 대한 사전 지식은 가지고 있다고 가정한다. 복호기는 두 부호어에 대하여 결합 복호를 수행한다.

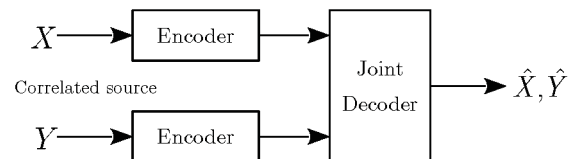


그림 1. Slepian과 Wolf의 분산 소스 부호화

〈그림 2〉는 SW 부호화를 통하여 얻어낼 수 있는 압축 영역을 보여준다. 회색으로 표현된 영역이 바로 압축 가능한 영역이다. 양 부호화기의 압축 후 데이터율의 합은 결합 부호화 한 계인  $H(X, Y)$ 를 능가할 수 없다. 점 A부터 B사이의 점들은  $R_X + R_Y = H(X, Y)$ 인 점들로, 부호율의 합 관점에서 최적인 점들이다. 이러한 최적이 되는 선의 양 끝점 A, B를 코너 지

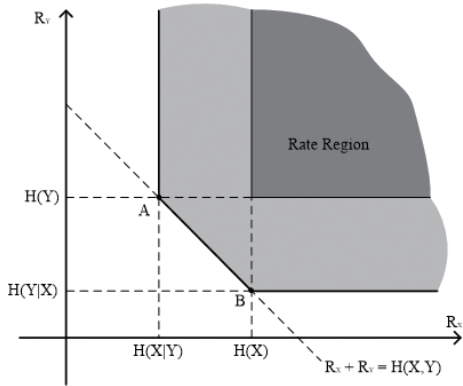


그림 2. SW 부호화의 압축 영역

점(corner point)이라 부른다.

이러한 SW 부호는 분산 비디오 부호화의 핵심 요소 기술이 된다. 특히 분산 비디오 부호 시스템 구현에 중요한 부분은 앞서 소개한 코너 지점의 부호화를 실현하는 것이다. 코너 지점의 부호화는 먼저  $Y$ 에 대해서는 독립적인 압축 한계인  $R_Y = H(Y)$ 의 데이터율로 압축을 수행한다. 반면,  $X$ 는  $H(X|Y)$  데이터율로 압축된다. 복호기에서는  $H(Y)$ 로 압축된  $Y$ 는 온전하게 복원이 가능하고,  $Y$ 를 부가정보로 이용하여,  $C_X$ 로부터  $X$ 를 추정 및 복원한다. 여기서  $X, Y$ 중 한쪽에만 양자간 상관성에 의한 추가압축을 하였기 때문에 비대칭 SW 부호화라 불린다. 분산 비디오 부호화에서는 이러한 코너 지점에서 동작하는 비대칭 SW 부호를 핵심 요소로 사용한다.

## 2. Wyner-Ziv 손실 부호화 기술

Wyner와 Ziv는 SW 분산 소스 부호화의 개념을 손실 압축으로 확장시켰다[6]. 손실 압축에서는 허용된 왜곡  $D$  한도 내에서 데이터를 최대 압축하는 것을 목적으로 한다. 앞서 언급한 코너 지점에서의 분산 소스 부호화 문제로 WZ 손실 압축을 논해 보기로 하자.

$Y$ 는 따로 부복호가 진행되므로, 복호기에서 부가정보로 제공된다고 볼 수 있다.  $X$ 에는  $X, Y$ 간의 상관도가 고려된 부호화가 적용된다.  $\hat{X}$ 를 복원 후의 데이터라 하자. 복원 후 왜곡은  $D = E[d(X, \hat{X})]$ 와 같이 정의할 수 있는데, 여기서,  $d(\cdot, \cdot)$ 은 거리 측정방법이며, 쉬운 예로 유클리드 거리를 고려할 수 있다. WZ 분산 손실 압축에서  $D$ 에 대한 최적의 압축률을  $R(D)$ 라 하자. 이 압축률이 결합 압축에 의하여 얻어질 수 있는 한계와 일치한다는 것이 증명되었다[6].

후에 실용적인 WZ 부호화의 구현이 연구되었는데, 특히 WZ 부호화가 양자화기(Quantizer)와 무손실 SW 부호의 연접으로 최적에 가까운 성능을 보일 수 있음이 보여졌다[8][9][10]. 실제

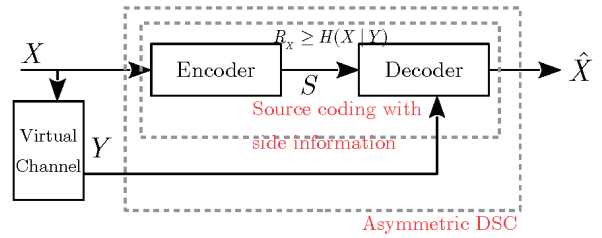


그림 3. 비대칭 SW 부호화

로 좀 더 간단한 기능을 하는 요소기술들의 조합으로 최적에 가까운 성능을 보였고, 분산 비디오 부호화에서 WZ 부호화도 이러한 요소기술 연접의 형태로 나타난다.

## 3. 채널 부호화와 비대칭 SW 부호화의 등가성

SW 분산 소스 부호화가 채널 부호화와 유사성을 갖는다는 점은 매우 일찍이 알려졌다[11]. 이러한 비대칭 문제와 채널 부호화 문제와의 유사성을 소개하여 분산 비디오 부호화가 어떻게 오류정정부호를 이용하여 구현되게 되었는지에 대한 이해를 돕고자 한다.

<그림 3>은 비대칭 SW 부호화의 구조이다. 소스  $Y$ 는 독립적으로  $H(Y)$ 의 데이터율로 부호화 되고, 복호기에서는 독립적으로  $Y$ 로 복호된다.  $X$ 는  $H(X|Y)$ 의 데이터율로  $S$ 로 압축되며, 결합 복호기에서는 소스 부호어  $S$ 와 부가정보  $Y$ 를 이용하여  $X$ 를 복원한다.

여기에서 간단하게 이 구조를 채널 부 복호화에 투영시켜 보자. 소스 벡터  $X$ 를 메시지라 하자. 이로부터 얻어지는  $S$ 는 조직적 부호의 패리티로 가정한다. 부호어는 메시지  $X$ 와 패리티  $S$ 로 이루어진 조직적인 부호의 채널 부호어가 된다.  $Y$ 는  $X$ 와 상관성을 가진 벡터로 채널을 통과한 후 잡음에 의해서 왜곡된 혹은 잡음이 섞인  $X$ 로 간주될 수 있다. 예를 들어  $X$ 와  $Y$ 가 이진 소스인 경우는  $X$ 를 입력으로 하고,  $Y$ 를 출력으로 하는 이진대칭채널(Binary Symmetric Channel)을 정의할 수 있다.

복호기에서는 압축된 부호어인 신드롬  $S$ 로부터 부가정보(side information)  $Y$ 를 이용하여 소스 데이터  $X$ 를 복원한다. 여기서 소스 부호어는  $S$ 이지만 이에 대응되는 채널 부호에서는  $S$ 는 메시지에서부터 생성된 노이즈가 없는 패리티이고,  $Y$ 는 잡음이 섞인 메시지로 이해될 수 있으며,  $Y$ 와  $S$ 의 연접된 벡터를 잡음이 섞인 조직적 부호(systematic code)의 부호어로 간주될 수 있으며, 여기서  $X$ 에 대한 최적의 추정값을 얻는 것을 일반적인 채널 복호 과정으로 이해할 수 있다. 이와 같이 비대칭 SW 부호화 문제는 구조적 채널 부호화 문제로 변환된다. 이러한 등가의 관계

에 의하여 패리티 S를 생성시킨 부호가 좋은 성능을 갖는 채널 부호라면, 이로 구현된 SW 부호의 성능도 보장될 것이다.

이러한 채널 부호화와 SW 부호 간의 유사성을 이용하여, SW 부호를 실용적으로 구현하려는 시도가 있어 왔다[12]. 초창기에는 트렐리스 부호를 이용하여 비대칭인 경우[12], 대칭인 경우 [13][14][15]에 대하여 구현이 시도되었다.

그 후 새년의 용량에 근접하는 성능을 보이는 터보 부호[16] 기술이 성숙되면서, 분산 소스 부호는 터보 부호를 이용하여 구현되기 시작했다[17][18][19]. 터보 부호는 또한 분산 소스 부호화가 아니라 단일 소스의 부호화에도 사용되었다[20]. 또한 LDPC 부호를 적용하여 SW 부호를 구현하는 시도도 뒤를 이었다[21][22][23].

#### 4. 분산 소스 부호화를 이용한 비디오 부호화

〈그림 4〉는 분산 비디오 부호화의 개념도이다. 상관성을 갖는 연속된 비디오 프레임이 존재한다. 여기서 분산 비디오 부호화의 두 소스  $X$ ,  $Y$ 와 부호어라 할 수 있는  $S$ 는 다음과 같이 얻어진다. 위 3개의 연속된 비디오 프레임에서 첫 번째와 3번째 프레임은 키 프레임으로 독립적으로 압축 부호화(intra-coding) 되는 프레임들이다. 키 프레임들과 두 번째 프레임과의 상관성을 이용하여 두 번째 프레임을 압축하려 한다. 물론 분산 소스 부호화에서는 하나의 키프레임을 사용할 수 있지만, 이 예제에서는 앞뒤로 인접한 두 개의 키프레임을 사용하는 것을 가정한다.

압축하고자 하는 데이터  $X$ 는 압축 함수인  $g(\cdot)$ 를 이용하여 압축하여 패리티 혹은 소스 부호어  $S$ 를 얻어낸다. 이 패리티 데이터는 복호기로 전송된다. 분산 비디오 복호기에서는 우선 독립적으로 부호화된  $X_1$ ,  $X_2$ 를 각자에 대한 복호과정을 통하여 획득할 수 있다. 남은 것은 부호어  $S$ 로부터 부가정보  $Y$ 를 이용하여  $X$ 를 복원하는 것이다. 여기서  $Y$ 는 복호에 필요한 부가정보인데, 온전하게 주어지는  $X_1$ ,  $X_2$ 로부터  $X$ 에 대한 최선의 추

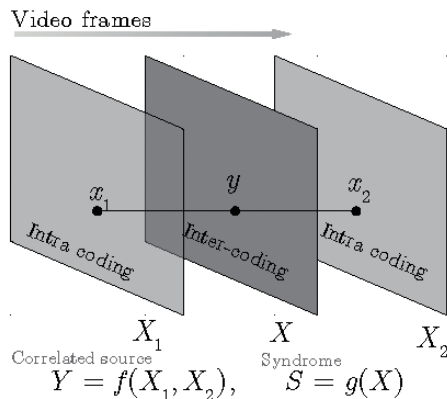


그림 4. 분산 비디오 부호화 개념도

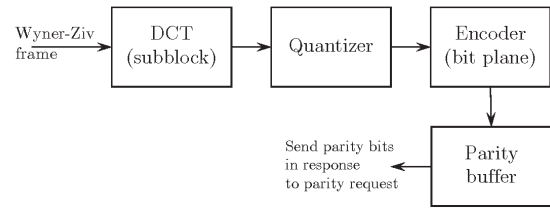


그림 5. 변환영역 WZ 비디오 부호화기

정치인  $Y = f(X_1, X_2)$ 를 얻어서 사용하게 된다. 키 프레임들로부터 최적의  $Y$ 를 얻는 일이 매우 중요한 문제이며, 간단하게는 앞뒤 프레임의 측정치의 평균을 사용할 수도 있고, 프레임 내의 상관성을 고려한 한 방법을 사용하면 더 좋은 성능을 제공하는  $Y$ 를 얻어낼 수도 있다.

앞서 II-3에서 설명한 바와 같이 비대칭 분산 소스 부호화가 채널 부호화와 등가이므로, 위에 설명한 분산 비디오 부호화는 좋은 오류정정부호를 이용하여 구현될 수 있다. 다음 장에서는 실용적인 분산 비디오 부호화의 구현을 위하여 어떤 구조와 어떤 오류정정부호가 채용되고 있는지에 대하여 논한다.

### III. 실용적 분산 비디오 부호화: 변환영역 WZ 부호 및 부호을 적응적 채널 부호

실용적인 분산 비디오 부호화를 실현하는데 고려된 사항과 해결 방법들을 소개한다.

#### 1. 분산 비디오 부호화의 구현

실제 분산 소스 부호화를 이용하여 구현되는 분산 비디오 부호화는 WZ 부호화의 형태로 구현이 된다. 그리고 이러한 WZ 부호화는 실용적인 SW 부호와 양자화기가 연결된 형태[8][9][10]로 구현된다.

한 가지 중요한 점은 일반적으로 비디오 압축은 비디오 프레임 전체에 대하여 적용하기 힘들다. 그러므로 비슷한 통계적인 상관성을 갖는 변수들을 모아 묶어서 블록 부호화를 구현하는 것이 중요하다. 인접한 비디오 프레임들 내에서 최대한 동질의 상관성을 갖는 변수들로부터 이진 블록 부호화의 매핑을 얻는 것은 분산 비디오 부호화의 큰 구조를 결정하게 된다.

이러한 관점에서 분산 비디오 부호화는 크게 픽셀영역 부호화 [2],[19]와 변환 영역 부호화[1],[4],[7]의 두 가지 방법으로 나뉜다. 픽셀 영역 부호화는 비디오 프레임의 픽셀들이 SW 압축을 위한 데이터가 되어서, 블록별로 부호화 되는 것이고, 변환 영

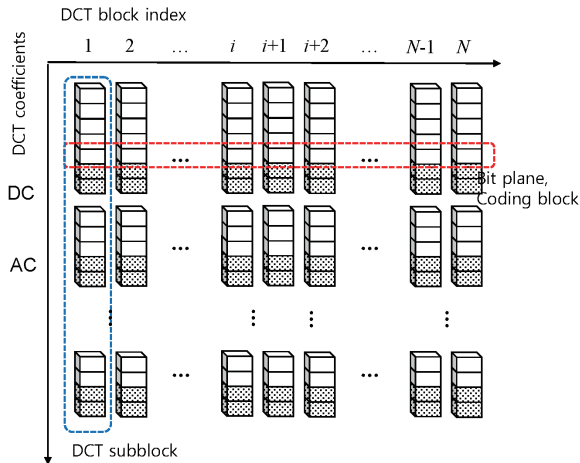


그림 6. 변환영역 WZ 비디오 부호화: 양자화 및 블록 부호화

역 부호화는 먼저 DCT(discrete cosine transform)와 같은 변환을 적용한 후에 소스 부호화를 진행하는 것이다. 본 논문에서는 성능상의 이점을 갖는 변환 영역 부호화 기법을 기반으로 설명하려 한다.

〈그림 5〉는 변환영역 WZ 비디오 부호화기의 일반적인 블록도이다. 먼저 이 부호화기는 키프레임이 아니고 분산 부호화에 의해 압축되는 WZ 프레임에 대한 작업만을 설명하고 있다. 우선 WZ 프레임은 작은 서브블록들로 분할된다. 그 후, DCT가 적용되어 주파수 영역의 데이터로 변환된다. 이 때, 서브블록별로 DCT 상수들의 집합이 주어진다. 이 DCT 계수들을 데이터의 중요도들을 고려하여 양자화를 수행한다.

그리고 양자화 후에는 SW 부호화가 진행되는데, 여기서 DCT 상수의 데이터가 어떻게 이진 블록 부호에 대응이 되는지가 중요하다. DCT 계수 데이터의 양자화된 버전으로부터 서브블록별 동일 위치에 있는 비트를 모두 모아서 부호화 혹은 패리티를 생성시킨다.

〈그림 6〉은 앞서 언급한 양자화와 SW 이진 블록 부호화의 구조를 보여준다. 〈그림 6〉에서는 위에서부터 DCT의 직류성분과 교류성분을 주파수가 높아지는 순서대로 세로축으로 나열한 것이고, 가로축은 한 프레임을 이루고 있는 DCT 서브블록의 인덱스를 나타낸다. 한 서브블록 내의 한 DCT 계수를 이루는 동일 비트를 모아서 블록 부호화를 수행한다.

이러한 변환 영역 기반의 WZ 비디오 부호화는 곧바로 개별 부호화보다는 훨씬 좋고, 모션 추정까지 고려된 MPEG에 상당히 근접하는 성능을 보였다[3]. 본 고에서는 효율적인 변환 영역 WZ 부호화를 기본으로 하고 이 구조 내에 어떤 기술들이 접목되어 있는지에 초점을 맞춘다.

## 2. 피드백을 이용한 효율적인 압축률 제어

실제 WZ 비디오 부호화의 실용적인 구현에 있어 장애는 WZ 프레임과 키프레임의 데이터들의 상관도를 미리 정확하게 알 수 없다는 것이다.

원래 최적의 비대칭 SW 부호화는 적어도 한 부호화기에서 소스 간 상관도의 크기는 미리 알고 있다는 가정을 한다. 실제 변환영역 분산 비디오 부호화를 수행하는 경우에 우선 양자화가 적절하게 되었다면, 그 후 SW 부호어 별로 최적의 상관도에 맞게 압축을 하는 것이 최선일 것이다. 하지만, 실제 비디오 부호화가 프레임 별로 독립적으로 이루어질 때에는 분석 없이 프레임 픽셀 간, DCT 계수 간 및 같은 비트평면 상의 비트 간 상관도를 정확하게 알기 힘들다. 게다가 프레임 별로 상관도가 크게 달라질 수 있기 때문에 높은 정확도로 상관도를 예측하는 것은 매우 어렵다.

소스 간 상관도에 대한 추정이 부정확한 경우에는 해당되는 압축률-왜곡 성능이 현저하게 저하될 수 있다. 분산 비디오 부호화에서 소스 부호어 단위별로 최적의 압축을 얻기 위하여 피드백을 이용한 패리티 반복 전송 방법이 제안되었다[3][24].

이러한 패리티 반복 전송 구조가 〈그림 7〉에 도시되어 있다. 각 SW 부호어에는 CRC(Cyclic Redundancy Check) 부호와 같은 오류검출부호로 외부화 되어 있다. SW 부호화기에서는 충분한 패리티를 생성하여 패리티 버퍼에 저장된다. 버퍼 내에서 패리티는 서브그룹화 되고, 일정 개수의 서브그룹이 최초로 전송된다. SW 복호기에서는 복호를 시도하고, 복호가 성공했다면, 최초의 패리티 량으로 압축 데이터 율이 결정된다. 복호가 성공하지 않은 경우는 패리티 추가 전송 메시지를 송신기에 보낸다. 부호화기는 복호가 성공할 때까지 패리티 요구에 대하여 일정량의 패리티 그룹을 복호기에 보낸다.

이러한 과정을 통하여 매 SW 부호어마다 복호 성공 가능한 최적/최소의 패리티 량에 가깝게 부호화가 가능하다. 이 기법은 물론 피드백 채널의 사용, 복호 지연 등의 단점도 존재하지만, 압축 성능 관점에서 탁월한 성능을 보여서 그 후 많은 연구에서 이 과정을 기본적으로 채용하고 있다[27].

압축 데이터율 제어 기법에 필수적으로 필요한 것은 부호율

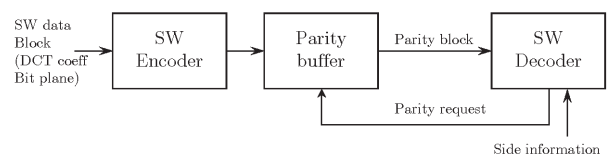


그림 7. 피드백을 이용한 압축률 제어 기법



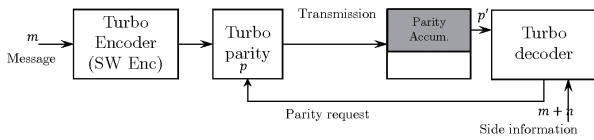


그림 8. 터보 부호를 이용한 Slepain-Wolf 부호의 구현:  
m은 메시지이고, 부가정보는 m+n임.

적응성을 갖는 채널 부호이다. 다양한 부호율에 대한 채널 부호가 존재해야함은 물론이고, 복호기에서 사용가능한 패리티의 총 그룹이 확장될 때, 즉 매번 복호기가 복호를 시도할 때마다, 그 경우에 해당되는 채널 부호가 최적화되어 있는 것이 좋다. 다음 장에서는 실용적 SW 및 실용적 분산 비디오 부호화에 사용되는 채널 부호에 대하여 소개 한다.

### 3. 분산 비디오 부호화를 위한 부호율 적응성을 갖는 오류정정부호

실용적인 SW 부호화에 사용 가능한 우수한 채널 부호로 터보 부호와 LDPC부호를 들 수 있다. 당연하게도 특정 부호율에 최적화 되어 있는 이론적 한계에 근접하는 성능을 갖는 터보 부호와 LDPC 부호는 매우 좋은 SW 압축 성능을 보일 것이다. 그런데, 부호율 적응 성질을 가지면서 좋은 성능을 갖게 하기 위해서는 설계 방법 혹은 구조의 변경이 필요하다. 실용적인 SW 부호의 구현을 위하여 천공된 터보 부호[19]와 LDPCA(low density parity check accumulated) 부호[25]가 제안되었고, 우수한 성능을 보이는 것이 확인 되었다. 이 절에서는 이 두 부호의 구조를 간단하게 소개하기로 한다.

먼저 터보 부호는 [19]에서 분산 소스 부호화에 처음으로 사용되었고, 분산 비디오 부호화에 적용되었다[7]. 기본적으로는 매우 단순한 천공된 터보 부호의 형태를 갖고 있으며, 부호율 적응에 용이한 터보 부호의 특성상 미리 생성된 패리티를 특정 순서대로 나열하고 그 순서에 맞추어서 패리티 요구에 따라 전송하는 구조를 가지고 있다. 터보 부호는 전반적으로 매우 우수한 복호 성능을 보이지만, 소스 간 상관성이 낮아서 높은 SW 압축 데이터율이 필요한 경우에 압축 성능이 급격하게 나빠지는 경향을 보인다.

반면 LDPC 부호는 비슷한 방법으로 이분 그래프로 설명된 <그림 9>와 같이 적용 가능한데, 패리티를 증가시키면서 그에 관련된 에지들도 생겨나므로, 결과적으로 패리티 검사 행렬의 밀도가 매우 높아지게 된다. 패리티검사 행렬의 밀도가 높으면 높은 확률로 4사이클이 생성되어 부호의 성능을 저하시킨다. 혹은 반대로 패리티를 많이 생성시킨 후에 천공을 통하여 부호

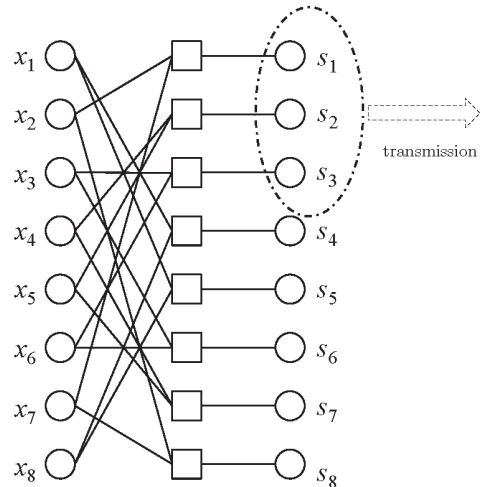


그림 9. LDPC 부호 구조를 이용한 SW부호 구현:  
x는 데이터, s는 패리티 (혹은 신드롬)이다.

를 얻으려고 하더라도 마지막에는 연결된 그래프를 유지시키기 어려워진다.

이런 이유로 Varodayan은 LDPCA (LDPC accumulate) 부호라는 매우 독특한 부호를 제안하여, LDPC부호를 분산 비디오 부호화에 성공적으로 적용하게 된다[25]. <그림 10>은 LDPCA부호의 구조 및 부호율 적응성 부여를 위한 등가 그래프를 설명한다. LDPCA부호에서는 왼쪽 그래프와 같이 신드롬 대신에 누적된 신드롬  $a$ 를 저장하고 전송한다. 이 그래프를 기저 그래프라 부르자. 만약 누적 신드롬 중  $a_1, a_3, a_5, a_7$ 을 알고 있다고 가정하면, 이 신드롬들과 연결된 체크노드를 병합할 수 있게 되고, 오른쪽과 같은 그래프를 얻게 된다. 이 때, 알고 있는 신드롬은 그래프에서 사라지고, 체크노드 병합에 의해서 중복되는 에지도 상쇄되어 제거된다.

위의 그래프 간 등가성에 의하여 증가성 패리티 혹은 신드롬의 전송을 하더라도 LDPC부호의 성능이 파괴되지 않도록 유지할 수 있게 된다. <그림 10>의 그래프로 2단계에 걸쳐서 신드롬이 전송되는 경우를 가정한다. 먼저  $a_2, a_4, a_6, a_8$ 가 전송되었다면, 오른쪽의 그래프를 이용하여 복호하면 된다. 왼쪽 그래프에서 노트 병합에 의하여 오른쪽 그래프가 얻어질 때, 기저 그래프에서 노트 병합이 이루어질 때 중복되는 에지가 사라지므로, 그래프의 에지 밀도도 너무 높아지지 않아서 좋은 성능을 보일 수 있게 설계할 수 있다.

만약 이 때, 소스 상관성이 낮아서 복호 오류가 발생했다면, 나머지  $a_1, a_3, a_5, a_7$ 를 전송한다. 이 때는 8개의 신드롬을 갖는 원그래프에 누적된 신드롬 8개 모두를 대입하여 복호할 수 있게 된다.

누적 신드롬의 전송을 여러 단계에 걸쳐서 수행해도 되며, 매

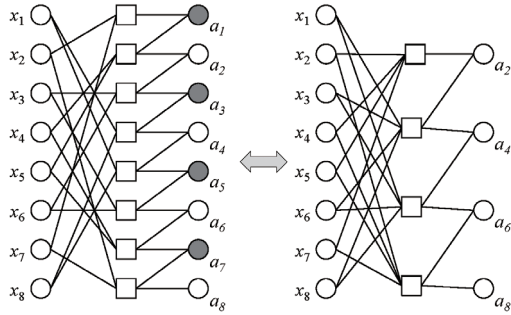


그림 10. LDPCA 부호: 누락된 신드롬을 사용한다. 위 두 부호는  $a_1, a_3, a_5, a_7$ 이 알려져 있다면 등가이다.

단계에서 원 그래프에서 노트 병합을 통하여 만들어진 그래프를 이용하여 복호가 가능하다. 그러므로, 이 부호는 좋은 부호율 적응성을 갖는다. 이러한 LDPCA 부호는 터보 부호에 비해 소스 간 상관성이 매우 낮은 영역에서도 좋은 성능을 보이는 것이 확인되었다[25][26].

#### IV. 결론

분산 비디오 부호화는 네트워크 정보이론의 대표적인 분산 소스 부호화 문제를 실용적으로 특히 가장 실용적이고, 영향력 있는 연구 분야인 비디오 부호화 분야에 적용을 한 예이다. 피드백 채널이나, 높은 복호 복잡도 등의 문제점을 갖지만, 매우 낮은 복잡도를 갖는 부호화 부분 덕분에 기존의 부호화와 분명한 차별화가 된다고 볼 수 있다. 터보 부호나 LDPC 부호와 같은 강력한 오류정정부호 기술은 분산 비디오 부호화의 실용화에 핵심적인 역할을 담당해 왔다. 이 분산 비디오 기술은 그 후 많은 발전을 거듭했고, 근래에는 분산 압축 센싱 기술과도 연관되어 관련 연구가 지속되고 있다[28][29].

#### 참고 문헌

[1] Puri and K. Ramchandran, "PRISM: a new robust video coding architecture based on distributed compression principles," in Proc. Allerton Conf. 2002.

[2] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video," in Proc. Asilomar Conf., Pacific Grove, CA, 2002.

[3] B. Girod et al., "Distributed video coding," Proc. IEEE, vol. 93, no. 1, pp. 71-83.

[4] R. Puri and K. Ramchandran, "PRISM: A 'reversed' multimedia coding paradigm," in Proc. IEEE Int. Conf. Image Processing, Barcelona, Spain, 2003.

[5] J. D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," IEEE Trans. Inf. Theory, vol. IT-19, pp. 471-480, July 1973.

[6] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," IEEE Trans. Inf. Theory, vol. IT-22, no. 1, pp. 1-10, Jan. 1976.

[7] A. Aaron, et al., "Transform-domain Wyner-Ziv codec for video," in Proc. SPIE Visual Commun. Image Proc. Conf., San Jose, CA, 2004.

[8] Z. Xiong, et al., "Nested quantization and Slepian-Wolf coding: a Wyner-Ziv coding paradigm for i.i.d. sources," in Proc. IEEE Workshop Stat. Sig. Proc. (SSP), St. Louis, MO, 2003.

[9] Z. Liu, et al., "Slepian-Wolf coded nested quantization for Wyner-Ziv coding: performance analysis and code design," in Proc. IEEE Data Compression Conf. 2004.

[10] M. Fleming, Q. Zhao, and M. Effros, "Network vector quantization," IEEE Trans. Inf. Theory, vol. 50, no. 8, pp. 1584-1604, Aug. 2004.

[11] A. Wyner, "On source coding with side information at the decoder," IEEE Trans. Inf. Theory, vol. 21, no. 3, pp. 294-300, May 1975.

[12] S.S. Pradhan, K.R. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," in Proc. IEEE Data Compression Conf., 1999.

[13] S.S. Pradhan, K.R. Ramchandran, "Distributed source coding: symmetric rates and applications to sensor networks," in Proc. IEEE Data Compression Conf., 2000, pp. 363-372.

[14] S. S. Pradhan, J. Kusuma, and K. Ramchandran, "Distributed compression in a dense microsensor network," IEEE Signal Process. Mag., vol. 19, no. 2, Mar. 2002.

[15] X. Wang and M. Orchard, "Design of trellis codes for

- source coding with side information at the decoder," in Proc. IEEE Data Compression Conf., 2001.
- [16] Claude Berrou and Alain Glavieux. "Turbo codes," Encyclopedia of Telecommunications (2003).
- [17] J. García-Frías, "Compression of correlated binary sources using turbo codes," IEEE Commun. Lett., vol. 5, no. 10, pp. 417–419, Oct. 2001.
- [18] J. Bajcsy and P. Mitran, "Coding for the Slepian–Wolf problem with turbo codes," in Proc. IEEE Globa Commun. Conf., vol. 2, 2001, pp. 1400–1404.
- [19] A. Aaron and B. Girod, "Compression with side information using turbo codes," in Proc. IEEE Data Compression Conf., 2002, pp. 252–261.
- [20] P. Mitran and J. Bajcsy, "Turbo source coding: a noise–robust approach to data compression," in Proc. IEEE Data Compression Conf., 2002, p. 465.
- [21] A. Liveris, Z. Xiong, and C.N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," IEEE Commun. Lett. vol. 6, no 10, 2002.
- [22] D. Schonberg, K. Ramchandran, and S.S. Pradhan. "LDPC codes can approach the Slepian Wolf bound for general binary sources," in Proc. Allerton Conf., 2002.
- [23] C.F. Lan, et al., "Slepian–Wolf coding of multiple M–ary sources using LDPC codes," in Proc. IEEE Data Compression Conf., 2004, p. 549.
- [24] A. Aaron, E. Setton, and B. Girod, "Toward practical Wyner–Ziv coding of video," in Proc. IEEE Int. Conf. Image Process., Barcelona, Spain, 2003.
- [25] D. Varodayan, A. Aaron and B. Girod, "Rate–Adaptive Distributed Source Coding Using Low–Density Parity–Check Codes," in Proc. Asilomar Conf., Pacific Grove, CA, November 2005.
- [26] S.–Y. Shin, et al, "Design of Binary LDPCA Codes with Source Revealing rate–Adaptation." IEEE communications letters 16.11 (2012): 1836–1839.
- [27] X. Artigas, et al., "The Discover Codec: Architecture, Techniques and Evaluation", in Proc. PCS, Lisbon, Portuga, Nov. 2007.
- [28] J. Prades–Nebot, Y. Ma, and T. Huang. "Distributed video coding using compressive sampling." in Proc. PCS 2009.
- [29] T.T. Do, et al. "Distributed compressed video sensing." in Proc. ICIP 2009.

## 약 력



김 종 환

2010년 성균관대학교 정보통신대학 학사,  
2010년~현재 성균관대학교 전기전자컴퓨터공학과 석박사과정  
관심분야: 극부호, LDPC부호, 차세대 방송 시스템



김 상 효

2004년 서울대학교 전기공학부 박사  
2004년~2006년 삼성전자 통신연구소  
2007년~2011년 성균관대학교 정보통신대학 조교수  
2012년~현재 성균관대학교 정보통신대학 부교수  
관심분야: 부호이론, 무선통신