

Media-independent Error Correction Scheme에 관한 연구

박덕근⁰ 박원배
경북대학교 정보통신학과
papil7@knuicsvr2.knu.ac.kr wbpark@ee.knu.ac.kr

Research of Media-independent Error Correction Scheme

Durk Gun Park⁰ Won Bae Park
Dept. of Information Communication, Kyungpook National University

요 약

실시간의 특성을 가지는 데이터의 경우 네트워크상에서 분실된 패킷을 복구시키기 위해서 FEC 방법을 사용한다. FEC는 최소한의 지연만으로 손실 패킷의 복구를 효율적으로 할 수 있는 장점을 가지고 있으나 네트워크상에서의 패킷 손실 특성에 많이 의존되는 경향이 있다. ITU-T의 Study Group 16에서는 Real-Time Transport Protocol (RTP)를 사용하여 네트워크에서 분실된 패킷을 복원시키는 방법으로 Media-independent error-correction scheme을 정하였다. 이 Scheme에 의해 만들어진 error-correction을 위한 신호와 media bitstream은 UDP에 의해 encapsulation될 RTP에 실리게 된다. Scheme은 real-time이라는 환경에 유리하도록 bandwidth와 latency 그리고 cost를 최소화하려고 했으며 이에 따라 네 가지 scheme을 정하였다. 네 가지의 Scheme은 오버헤드와 지연시간의 크기가 차별화를 두어 네트워크 환경의 변화에 적응하도록 하였다. 그러나 네트워크 환경에 보다 더 탄력적이며 효율적으로 적응하기 위해서 또 하나의 scheme을 제안한다. 새로 고안한 이 다섯번째 scheme은 scheme 3에 비해 작은 latency를 가지는 장점이 있는 반면 연속적으로 분실된 패킷에 대한 복원확률이 다소 떨어진다. 하지만 scheme 1과 2에 비해서는 연속적인 패킷 분실의 복원확률이 높아 네트워크환경에 따라 scheme 4를 사용하면 네 개의 scheme을 사용하여 분실패킷의 복원을 하는 경우보다 보다 효율적인 전송과 복원이 이루어질 것이다.

1. 서론

네트워크상의 정체현상은 패킷 손실의 문제를 발생시킨다. 하지만 이러한 정체현상으로 인한 패킷 손실문제를 짧은 시간 내에 근본적으로 해결하기는 어렵다. 따라서 패킷이 분실되었을 때 손실 패킷을 복구하는 방법에 대해 활발한 연구가 진행 중이며 현재 대표적으로 사용되고 있는 방법 중 가장 확실한 것이 재전송(ARQ) 방법이다. 하지만 재전송의 경우에 비교적 많은 시간이 필요하며 실시간 처리가 필요한 데이터의 경우 이러한 시간을 허용할 수가 없다. 따라서 이러한 경우를 위해 사용되는 방법 중에 대표적인 것이 FEC(Forward Error Correction)이다. FEC는 원래의 신호인 미디어데이터가 분실하더라도 수신측에서 분실되지 않은 나머지 패킷으로 분실된 패킷을 복원시키는 방법이다. 특히 FEC는 최소한의 지연만으로 손실된 패킷을 복원할 수 있는 장점이 있으나 복원가능의 유무는 네트워크상에서의 패킷 손실 특성에 크게 따른다. 이런 특성의 FEC 방법으로 만들어진 FEC패킷과 원래의 미디어 패킷은 RTP(Real-time Transfer Protocol)에 실리게 된다. RTP는 실시간 데이터 전송에 적합한 전송 프로토콜이며 따라서 주로 오디오, 비디오 등의 실시간데이터를 멀티캐스트나 유니캐스트로 전송하는데 사용된다. RTP는 다시 실시간 전송에 알맞은 UDP 패킷 안으로 encapsulation된다. 그림 1에서는 Sender-based repair 기술의 분류를 보여준다.

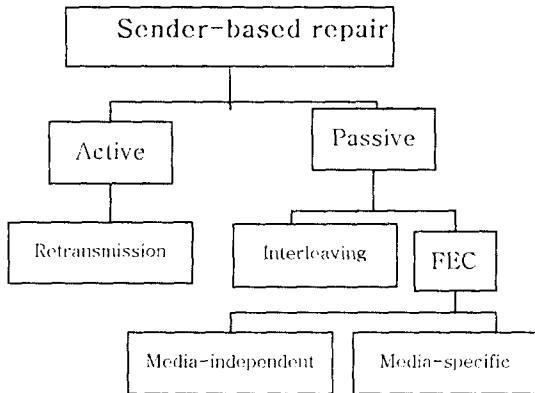


그림1. Sender중심의 복구 기술

2. 관련연구

ITU-T Study Group 16에서는 실시간 전송에 적합한 RTP(Real-time Transfer Protocol)를 사용할 수 있는 Media-independent Error Correction에 대한 Proposal을 제출하였다. Proposal에서는 네트워크 상태에 따라 XOR을 사용한 Media-independent Error Correction scheme 네가지를 적용시키고 있다.

FEC Scheme에 따른 특징을 (n:k:d)로 나타내는데 n은 원시패킷수, k는 FEC 전송패킷수, d는 패킷지연을 나타내게 된다.

2.1 Error-Correction Scheme 0 (1:1:0)

Scheme 0은 별도의 FEC패킷을 생성하지 않고 그대로 전송하는 방법이다. 따라서 네트워크상에서 분실된 패킷을 복구할 수 없다. 네트워크의 손실율이 아주 낮은 경우에 사용하며 지연문제와 오버헤드문제가 없다는 것이 장점이다.

2.2 Error-Correction Scheme1 (2:1:1)

Scheme 1은 두 개의 미디어 패킷 사이에 XOR한 FEC를 첨가해서 보낸다.

$$a, b, c, d, \dots \Rightarrow a, ab, b, bc, c, cd;$$

2.3 Error-Correction Scheme2 (3:2:1)

원래의 미디어 패킷은 전송하지 않고 FEC패킷만을 전송하는 방법이다.

$$a, b, c, d, e, f, g \Rightarrow$$

$$ab, ac, abc, cd, ce, cde, ef, eg, efg, \dots$$

Scheme2는 다른 Scheme과 달리 Carry-over의 유무에 따라 그 복원방법이 차이를 가지게 된다. 여기서 Carry-over는 그룹으로 나뉘어지 전송할 때 한 그룹의 제일 끝의 미디어 패킷이 된다. 따라서 특정 미디어 패킷이 복원되느냐 아니냐는 다음 패킷의 복원에 영향을 미친다.

Carry-over 값이 있을 경우

$$b = (a)(ab) = (ac)(abc)$$

$$c = (a)(ac) = (ab)(abc)$$

Carry-over 값이 없을 경우

$$a = (ab)(ac)(abc)$$

$$b = (ac)(abc)$$

$$c = (ab)(abc)$$

Scheme 2의 경우 연속이 아닌 부분적으로 분실된 모든 패킷을 복원할 수 있으며 2개의 연속적인 패킷 손실에도 70~80% 정도의 복구율을 보인다. 이러한 복구율이 Scheme1에 비해 다소 떨어지더라도 오버헤드를 낮출 수 있는 장점이 있다.

2.4 Error-Correction Scheme3 (2:1:4)

Scheme3은 네 가지 Scheme 중에 연속적인 패킷의 손실 복구에 가장 강점을 가진다. 이러한 장점은 지연시간 차원에서 성능이 떨어지는 결과를 가져 올 수 밖에 없다.

$$a, b, c, d \dots \Rightarrow a, b, abc, c, acd, abd, d, bcd$$

$$a = (a) = (b)(c)(abc) = (b)(d)(abd) = (c)(d)(acd) = (d)(abc)(bcd) = (c)(abd)(bcd) = (b)(acd)(bcd) = (abc)(abd)(acd)$$

$$b = (b) = (a)(c)(abc) = (a)(d)(abd) = (c)(d)(bcd) = (d)(abc)(acd) = (c)(abd)(acd) = (a)(bcd)(acd) = (abc)(abd)(bcd)$$

$$c = (c) = (a)(b)(abc) = (a)(d)(acd) = (b)(d)(bcd) = (d)(abc)(abd) = (b)(acd)(abd) = (a)(bcd)(abd) = (abc)(acd)(bcd)$$

$d = (d) = (a)(b)(abd) = (a)(c)(acd) = (b)(c)(bcd)$
 $= (c)(abd)(abc) = (b)(acd)(abc) = (a)(bcd)(abc)$
 $= (abd)(acd)(bcd)$

$a, b, c; d, e, \dots \Rightarrow a, ab, b, ac, bc, abc; d, de, \dots$

이 새로운 Scheme은 패킷 그룹끼리는 독립적이어서 다른 그룹에 영향을 받지 않는다.

복원 방법은 5가지가 있다

$a = (a) = (b)(ab) = (bc)(abc) = (b)(ac)(bc)$
 $= (ab)(bc)(abc)$

$b = (b) = (a)(ab) = (ac)(abc) = (a)(ac)(bc)$
 $= (ab)(bc)(abc)$

$c = (a)(ac) = (b)(bc) = (ab)(abc) = (a)(b)(abc)$
 $= (ac)(bc)(abc)$

Scheme 4는 (2:1:3)으로 규정된다.

3. 결론 및 향후 연구 방향

지금까지 ITU-T의 Study Group 16에서 제시한 RTP를 사용하는 Media-independent Error Correction을 살펴보고 보다 효율을 올릴 수 있는 새로운 Scheme을 제안하였다. 새롭게 제시한 Scheme4는 (2:1:3)의 특성을 가지며 Scheme3에서만 가능한 연속적인 4개의 패킷 분실에도 복원이 일부가능하며 Scheme 3에 비하여 지연특성보다 더 좋은 특성을 가진다. 따라서 향후 연구에서는 Scheme 0~4가 가장 효과적인 기능을 하는 네트워크 환경을 분석해 내고 이를 최적화로 제어하는 프로토콜을 구현할 것이다.

4. 참고 문헌

[1]. Bolot, J.-C. and Vega-Garcia, A., "The case for FEC-based error control for packet audio in the Internet," *Multimedia Systems*, 1997

[2]. H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A transport Protocol for Real-Time Applications," RFC 1889.

[3]. D. Budge, R. McKenzie, W. Mills, W. Diss, P. Long "Media-independent Error Correction using RTP" 1997

[4]. 박준석, 고대식 "인터넷폰의 설계 및 개발"

[5]. Colin Perkins, Orion Hodson, and Vicky Hardman "A Survey of Packet Loss Recovery Techniques for Streaming Audio"

[6]. Jean-Chrysostome Bolot, Sacha Fosse-Parisis, Don Towsley "Adaptive FEC-Based Error Control for Internet Telephony"

[7]. Jean-Chrysostome Bolot, Andres Vega-Garcia "Control Mechanisms for Packet Audio in the Internet"