



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2015년11월23일
(11) 등록번호 10-1571145
(24) 등록일자 2015년11월17일

- (51) 국제특허분류(Int. Cl.)
H04L 1/18 (2006.01) H04L 12/16 (2006.01)
H04W 28/04 (2009.01) H04W 84/12 (2009.01)
- (21) 출원번호 10-2014-7020431(분할)
- (22) 출원일자(국제) 2007년10월23일
심사청구일자 2014년07월21일
- (85) 번역문제출일자 2014년07월21일
- (65) 공개번호 10-2014-0098259
- (43) 공개일자 2014년08월07일
- (62) 원출원 특허 10-2010-7008641
원출원일자(국제) 2007년10월23일
심사청구일자 2012년09월12일
- (86) 국제출원번호 PCT/US2007/022453
- (87) 국제공개번호 WO 2009/054822
국제공개일자 2009년04월30일
- (56) 선행기술조사문헌
Philip A. Chou 외 2인, "Error Control for Receiver-Driven Layered Multicast of Audio and Video", IEEE Transactions on Multimedia, 3권, 1호, 2001.03.*
JP2004201111 A
*는 심사관에 의하여 인용된 문헌

- (73) 특허권자
틈슨 라이센싱
프랑스 92130 이씨레폴리노 루 잔다르크 1-5
- (72) 발명자
위우 명추안
미국, 뉴저지주 08550, 프린스턴 정선, 스톤레아 드라이브 10
리우 항
미국, 펜실베이니아주 19067, 야드리, 키팅 드라이브 486
- (74) 대리인
문경진, 김학수

전체 청구항 수 : 총 14 항

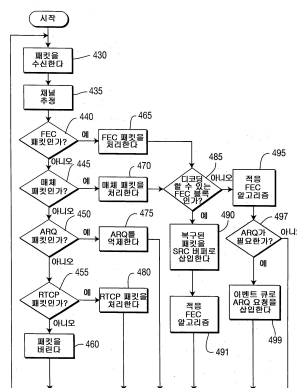
심사관 : 성경아

(54) 발명의 명칭 무선 근거리 네트워크들에서의 신뢰 가능한 멀티캐스트를 위해 병합된 자동 반복 요청으로 적응 순방향 에러 정정을 하기 위한 방법 및 장치

(57) 요약

멀티캐스팅 신뢰도를 증가시키기 위한 방법 및 장치가 설명되고, 이러한 방법은 제 1 멀티캐스트 그룹으로부터 복수의 순방향 에러 정정 코딩된 패킷들의 제 1 층과 콘텐츠를 수신하는 단계와, 복수의 순방향 에러 정정 코딩된 패킷들의 또 다른 추가 층과 함께 콘텐츠와 순방향 에러 정정 코딩된 패킷들의 추가 층들 중 하나를 수신하기 위해, 추가 멀티캐스트 그룹에 가입하는 단계를 포함한다.

대표도 - 도4b



명세서

청구범위

청구항 1

무선 근거리 네트워크들에서 멀티캐스팅(multicasting) 신뢰도를 증가시키는 방법으로서,
스트리밍 서버에서, 순방향 에러 정정 코딩된 패킷들의 복수의 층을 생성하는 단계,

상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 1 층과 콘텐츠를 수신하기 위해, 제 1 멀티캐스트 그룹에 가입(join)하라는 모바일 디바이스 또는 클라이언트 디바이스로부터의 요청을, 상기 스트리밍 서버에 의해, 수신하는 단계,

상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 1 층과 상기 콘텐츠를, 상기 스트리밍 서버에 의해, 송신하는 단계로서, 상기 콘텐츠가 송신되는 직후에 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 1 층이 송신되는, 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 1 층과 상기 콘텐츠를 송신하는 단계,

상기 콘텐츠가 상기 제 1 멀티캐스트 그룹으로부터의 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 1 층과 상기 콘텐츠로부터 복구 가능하지 않은 경우, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 2 층을 수신하기 위해, 제 2 멀티캐스트 그룹에 가입하라는 상기 클라이언트 또는 상기 모바일 디바이스로부터의 요청을, 상기 스트리밍 서버에 의해, 수신하는 단계, 및

요청의 수신시, 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 2 층을, 상기 스트리밍 서버에 의해, 송신하는 단계를

포함하고, 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 2 층의 송신은 지연되며, 어떤 모바일 디바이스 또는 클라이언트 디바이스도 제 2 멀티캐스트 그룹에 가입하지 않는 경우, 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 지연된 제 2 층은, 스트리밍 서버와 모바일 디바이스 또는 클라이언트 디바이스 사이에 위치되며 이들을 서로 간에 연결시키는 AP들 또는 라우터들에 의해 전송되지 않는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 방법.

청구항 2

제 1항에 있어서,

제 3 요청의 수신시, 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 제 3 층을, 상기 스트리밍 서버에 의해, 송신하는 단계와,

제 4 요청의 수신시, 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 제 4 층과 상기 콘텐츠를, 자동 반복 요청 서버(automatic repeat request server)에 의해, 재송신하는 단계를

더 포함하는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 방법.

청구항 3

제 2항에 있어서,

상기 콘텐츠가 상기 수신된 콘텐츠, 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 1 층, 및 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 2 층으로부터 복구 가능하지 않은 경우, 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 3 층을 수신하기 위해, 제 3 멀티캐스트 그룹에 가입하라는 상기 모바일 디바이스 또는 상기 클라이언트 디바이스로부터의 요청을, 상기 스트리밍 서버에 의해, 수신하는 단계를 더 포함하고,

상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 제 3 층의 송신은 지연되는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 방법.

청구항 4

제 2항에 있어서,

상기 콘텐츠가 상기 수신된 콘텐츠, 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 1 층, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 상기 제 2 층, 및 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 3 층으로부터 복구 가능하지 않은 경우, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 4 층과 잃어버린 콘텐츠의 재송신을 수신하기 위해, 자동 반복 요청 서버에서 자동 반복 요청 부정 통지(negative acknowledgement)를 수신하는 단계와, 제 4 멀티캐스트 그룹에 가입하라는 상기 클라이언트 또는 상기 모바일 디바이스로부터의 요청을, 상기 자동 반복 요청 서버에 의해, 수신하는 단계를 더 포함하고,

상기 잃어버린 콘텐츠와 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 4 층과 상기 잃어버린 콘텐츠의 재송신은 지연되는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 방법.

청구항 5

제 1항에 있어서,

상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 2 층은 시간의 조정 가능한 기간만큼 시간상 지연되는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 방법.

청구항 6

제 1항에 있어서,

상기 복수의 층들 각각에 적용된 순방향 에러 정정 코딩의 양은 조정 가능한, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 방법.

청구항 7

제 1항에 있어서,

상기 콘텐츠는 다수의 매체 트랙(track)을 포함하는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 방법.

청구항 8

무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 시스템으로서,

스트리밍 서버, 클라이언트 또는 모바일 디바이스, 및 스트리밍 서버와 모바일 디바이스 또는 클라이언트 디바이스 사이에 위치되며 이들을 서로 간에 연결시키는 AP들 또는 라우터들을 포함하되:

순방향 에러 정정 코딩된 패킷들의 복수의 층들을 생성하기 위한 스트리밍 서버의 생성 수단,

상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 1 층과 콘텐츠를 수신하기 위해, 제 1 멀티캐스트 그룹에 가입하라는 상기 클라이언트 또는 상기 모바일 디바이스로부터의 요청을 수신하기 위한 스트리밍 서버의 제 1 수신 수단,

상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 1 층과 상기 콘텐츠를 송신하기 위한 스트리밍 서버의 제 1 송신 수단으로서, 상기 콘텐츠가 송신되는 직후에 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 1 층이 송신되는, 스트리밍 서버의 제 1 송신 수단,

상기 콘텐츠가 상기 제 1 멀티캐스트 그룹으로부터의 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 1 층과 상기 콘텐츠로부터 복구 가능하지 않은 경우, 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 2 층을 수신하기 위해, 제 2 멀티캐스트 그룹에 가입하라는 상기 클라이언트 또는 상기 모바일 디바이스로부터의 요청을 수신하기 위한 스트리밍 서버의 제 2 수신 수단, 및

제 1 요청의 수신시, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 2 층을 송신하기 위한 스트리밍 서버의 제 2 송신 수단을

포함하고, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 2 층의 송신은 지연되며, 어떤 모바일 디바이스 또는 클라이언트 디바이스도 제 2 멀티캐스트 그룹에 가입하지 않는 경우, 상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 지연된 제 2 층은 AP들 또는 라우터들에 의해 전송되지 않는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 시스템.

청구항 9

제 8항에 있어서,

자동 반복 요청 메시지의 수신시, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 3 층을 송신하기 위한 스트리밍 서버의 제 3 송신 수단과,

자동 반복 요청 서버로서, 자동 반복 요청 콘텐츠 요청의 수신시, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 4 층과 상기 콘텐츠를 재송신하기 위한 재송신 수단을 포함하는, 자동 반복 요청 서버를

더 포함하는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 시스템.

청구항 10

제 9항에 있어서,

상기 콘텐츠가 상기 수신된 콘텐츠, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 1 층, 및 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 2 층으로부터 복구 가능하지 않은 경우, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 3 층을 수신하기 위해, 제 3 멀티캐스트 그룹에 가입하라는 상기 클라이언트 또는 상기 모바일 디바이스로부터의 요청을 수신하기 위한 스트리밍 서버의 제 3 수신 수단을 더 포함하고,

상기 순방향 에러 정정 코딩된 패킷들의 상기 복수의 층들 중 상기 제 3 층의 송신은 지연되는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 시스템.

청구항 11

제 9항에 있어서,

상기 콘텐츠가 상기 수신된 콘텐츠, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 1 층, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 2 층, 및 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 3 층으로부터 복구 가능하지 않은 경우, 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 4 층과 상기 콘텐츠를 수신하기 위해, 자동 반복 요청 부정을 수신하기 위한 자동 반복 요청 서버의 수신 수단과, 제 4 멀티캐스트 그룹에 가입하라는 상기 클라이언트 또는 상기 모바일 디바이스로부터의 요청을 수신하기 위한 자동 반복 요청 서버의 제 4 수신 수단을 더 포함하고,

상기 콘텐츠와 상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 4 층의 재송신은 지연되는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 시스템.

청구항 12

제 8항에 있어서,

상기 순방향 에러 정정 코딩된 패킷들의 복수의 층들 중 제 2 층은 시간의 조정 가능한 기간만큼 시간상 지연되는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 시스템.

청구항 13

제 8항에 있어서,

상기 복수의 층들 각각에 적용된 순방향 에러 정정 코딩의 양은 조정 가능한, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 시스템.

청구항 14

제 8항에 있어서,

상기 콘텐츠는 다수의 매체 트랙을 포함하는, 무선 근거리 네트워크들에서 멀티캐스팅 신뢰도를 증가시키는 시스템.

발명의 설명

기술 분야

[0001] 본 발명은 무선 통신 네트워크들에 관한 것으로, 특히 순방향 에러 정정 및 자동 반복 요청의 조합한 것을 적용함으로써, 멀티캐스트 애플리케이션의 신뢰도를 증가시키는 것에 관한 것이다.

배경 기술

[0002] 본 명세서에서 사용된 "콘텐츠"는, 오디오, 비디오, 및 임의의 멀티미디어 데이터를 포함하는 임의의 다른 형태의 데이터를 포함하기 위해 사용된다. 비디오와 콘텐츠라는 용어들은 본 명세서 상호 교환 가능하게 사용된다. 본 명세서에서 사용된 "/"는 동일하거나 유사한 구성 성분들 또는 구조물에 대한 대안적인 이름들을 표시한다. 즉, "/"는 본 명세서 사용된 "또는(or)"의 의미로 취해질 수 있다.

[0003] *무선 근거리 네트워크들(WLANs: Wireless local area networks)은, 그것들의 유연성 및 낮은 비용으로 인해, 가정, 호텔, 캠퍼스, 및 공항과 기차역과 같은 다른 핫 스팟(hot spot)에서 광범위하게 사용되어 왔다. 대부분의 경우, 사용자들은 웹사이트를 검색하거나 전자 우편을 확인하기 위해 WLAN에 연결하고, 실시간 멀티미디어 스트리밍을 지원하기 위해 WLAN들에 대한 수요가 증가하고 있다. 하지만, 무선 채널은 다중 경로 페이딩(fading)과 간섭들을 겪을 수 있고, 이는 랜덤한 버스트 패킷 분실을 야기할 수 있고, 스트리밍 애플리케이션의 콘텐츠 재생 품질에 영향을 미칠 수 있다. 신뢰도를 개선하기 위해, 순방향 에러 정정(FEC: forward error correction) 및/또는 자동 반복 요청(ARQ: automatic repeat request)과 같은 에러 정정 구조들이 사용될 수 있다. FEC에서, 패리티 패킷들은 최초의/소스 매체 패킷들/데이터와 함께 보내진다. 하지만, 각각의 모바일 디바이스가 상이한 채널 상태를 경험할 수 있기 때문에, 얼마나 많이 FEC가 보내질지를 결정하는 것은 어렵다. 더 낮은 FEC는 약한 보호를 야기할 수 있고, 잃어버린 패킷들/데이터가 복구되지 않을 수 있다. 더 높은 FEC는 더 많은 오버헤드(overhead)를 야기하고 네트워크 대역폭을 낭비할 수 있다. 본 발명은 대역폭 자원을 효율적으로 사용하면서 클라이언트/모바일 디바이스에 적절한 보호를 제공하는 적응 방법을 설명한다.

[0004] 에러 정정을 위해 ARQ를 사용함으로써, 클라이언트/모바일 디바이스의 데이터/패킷 복구가 긴 왕복 시간 지연을 겪을 수 있다. 멀티캐스트 애플리케이션에서는 ARQ가 또한 피드백의 급격한 증가 문제를 초래할 수 있다. 하지만, 왕복 시간 지연이 짧고, 적절한 피드백 억제 알고리즘이 사용될 때에는, ARQ가 여전히 실시간 콘텐츠 스트리밍을 위한 실행할 수 있는 에러 정정 구조이다.

[0005] FEC는 멀티캐스트 애플리케이션들에 대한 신뢰도를 개선하는 효율적인 방법이다. 다양한 FEC 구조가 패킷 레벨 에러 정정을 위해 애플리케이션 층에서 이용될 수 있다. 이들 후보에는 랜덤한 인터리빙 및 리드-솔로몬(RS: Reed-Solomon)을 구비한/구비하지 않은 프로(Pro)-MPEG이 포함된다. 모든 FEC 구조는 장점과 단점이 있다. 프로-MPEG FEC는 그것이 XOR 연산만을 이용하지만 그것의 에러 정정 능력이 그에 따라 제한되기 때문에 매우 가벼운 구조이다. 프로-MPEG은 심지어 패킷 분실이 높지 않을 때에도 일부 에러 패킷들을 정정할 수 없다. RS는 그것이 에러 패킷들과는 독립적으로 작용하기 때문에, 대부분의 경우 XOR-기반의 FEC보다 나은 에러 정정 능력을 가진다. RS를 이용하는 비용은 증가된 계산상 자원들이다. 그렇지만, 일부 경우에는 RS가 XOR-기반의 FEC 구조들보다 낮은 성능을 가지는데, 이는 FEC 코딩 블록에서 n개의 인코딩된 패킷 중에서 n-k개보다 많은 패킷 분실이 있을 경우 RS(n,k) 코드가 완전히 실패하기 때문이다. XOR 기반의 FEC는 n-k개보다 많은 패킷 분실이 존재할 때 잃어버리는 패킷들/데이터의 부분을 아마도 여전히 정정할 수 있다.

[0006] FEC가 지닌 문제점은 처음에 얼마나 많은 FEC를 보내야할지를 결정하는 것이다. 멀티캐스트 애플리케이션에서는, 상이한 모바일 디바이스가 상이한 채널 상태와 분실 속도를 가질 수 있다. 단일 정적 FEC는 모든 모바일 디바이스의 요구 사항을 충족시킬 수는 없다. 유선 네트워크에서의 서비스의 품질과 대역폭 사용 효율을 개선하기 위해 크기 조정 가능한 소스 코딩과 결합되는 층을 이룬 FEC가 종래 기술에 도입된다. 하지만 종래 기술의 구조는 ARQ 재송신을 고려하지 않았다. 수신기에 대한 패킷 분실이 FEC 코드의 능력보다 많다면, 잃어버린 패킷들은 복구될 수 없다. 게다가, 수신기는 그것이 필요로 하는 정확한 양의 FEC 패킷들을 얻을 수 없는데, 이는 수신기가 여러 층들에서 불연속적인 개수의 FEC 패킷들로 FEC를 얻기 때문이다. 또한, 중

래 기술의 층을 이룬 구조는 멀티미디어 작업 시간(session)의 다수의 트랙(track) 사이의 동기화를 고려하지 않았다. 예컨대, 멀티미디어 작업 시간에서는, 오디오 트랙과 비디오 트랙이 종종 존재하고, 비디오 트랙은 오디오 트랙보다 훨씬 더 높은 비트 속도를 가지며, 동일한 FEC 블록 크기가 사용된다면, FEC 버퍼를 채우기 위해서는 오디오 트랙에 대해 훨씬 더 긴 시간을 소비할 것이다.

[0007] 종래 기술의 해결책에서는, FEC 블록에서의 FEC 인코딩을 위해 사용될 패킷들의 최대 개수(maxK)가 설정된다. 버퍼에서의 트랙에 대한 비디오(오디오) 패킷들의 개수가 maxK에 도달할 때, 이들 비디오(오디오) 패킷들에 기초한 FEC 인코딩이 수행된다. 동시에, 나머지 오디오(비디오) 트랙 버퍼에 얼마나 많은 패킷이 있는지에 관계없이, 오디오(비디오) 매체 패킷들의 다른 트랙들에 기초한 FEC 인코딩이 또한 수행된다. N과 K가 모두 상이한 FEC 블록들에 대해 고정되지 않기 때문에, N과 K는 FEC 헤더에 포함되어야 하고, 이를 통해 정보가 클라이언트에 전달된다.

[0008] 종래 기술에서는 클라이언트들로부터 서버로의 평균 왕복 시간이 작고, 멀티캐스트 작업 시간에서의 클라이언트의 수가 적을 때, ARQ가 멀티캐스트에서 사용되었다. 피드백의 급격한 증가 문제를 회피하기 위해 종종 피드백 억제제가 구현된다. 종래 기술의 하이브리드 ARQ 구조들에서는, 최초의 매체 패킷들의 시퀀스 번호 대신 FEC 블록을 디코딩하기 위해, 클라이언트/모바일 디바이스가 그것이 필요로 하는 패리티(parity) 패킷들의 개수에 대한 요청을 보낸다. 재송신된 패리티 패킷들은 멀티캐스트 작업 시간에서 모든 클라이언트에 멀티캐스팅된다. 패리티 패킷들은 상이한 분실을 복구하기 위해 상이한 클라이언트/모바일 디바이스에 의해 사용될 수 있다.

[0009] 상이한 시나리오들은 상이한 에러 정정 구조들을 사용할 수 있다. 일부 경우, FEC가 더 효율적일 수 있지만, 다른 경우들에서는 ARQ가 더 나은 선택일 수 있다. 모든 애플리케이션 시나리오에 대한 단일 해결책을 제공하는 것이 유리할 것이다.

발명의 내용

해결하려는 과제

[0010] 본 발명은 순방향 에러 정정과 자동 반복 요청의 조합을 적용함으로써, WLAN들에서의 멀티캐스트 애플리케이션들의 신뢰도를 증가시키기 위한 적응 방법을 설명한다. 무선 방송/멀티캐스트 시스템에서는, 패킷들/데이터가 랜덤한 버스트 분실을 겪을 수 있는데, 이는 다중-경로 페이딩, 간섭, 핸드오프(handoff) 상황 등 때문이다. 신뢰도를 개선하기 위해, FEC와 ARQ와 같은 에러 정정 구조들이 채택될 수 있다. FEC와 ARQ를 조합하는 장점이 존재한다. FEC와 ARQ를 조합하는 것은 시스템 구현 유연성을 증가시키고 특정 시나리오에서 시스템의 성능을 개선할 수 있다. 본 발명은 병합된 하이브리드(merged hybrid) 구조, 즉 mhARQ로 본 명세서에서 표시된 조합된 구조를 설명한다. 본 발명의 mhARQ 시스템에서는, FEC 패킷들이 코딩되고, 하나 이상의 적응 FEC 멀티캐스트 그룹들과 ARQ 멀티캐스트 그룹으로 나누어진다. 그러므로 WLAN 방송/멀티캐스트 시스템은, 멀티캐스트 애플리케이션에서 ARQ 또는 적응 FEC 또는 이들 둘 다를 사용하도록 구성될 수 있다.

과제의 해결 수단

[0011] 멀티캐스팅 신뢰도를 증가시키기 위한 방법 및 장치가 설명되고, 이러한 방법은 제 1 멀티캐스트 그룹으로부터 복수의 순방향 에러 정정 코딩된 패킷들의 제 1 층과 콘텐츠를 수신하는 단계와, 복수의 순방향 에러 정정 코딩된 패킷들의 또 다른 추가 층과 함께 콘텐츠와 순방향 에러 정정 코딩된 패킷들의 추가 층들 중 하나를 수신하기 위해, 추가 멀티캐스트 그룹에 가입하는 단계를 포함한다. 또한 멀티캐스팅 신뢰도를 증가시키는 방법과 시스템이 설명되는데, 그러한 방법은 순방향 에러 정정 코딩된 패킷들의 복수의 층을 생성하는 단계, 상기 복수의 상기 순방향 에러 정정 코딩된 패킷들의 제 1 층과 콘텐츠를 송신하는 단계, 요청시, 상기 복수의 상기 순방향 에러 정정 코딩된 패킷들의 제 2 층을 송신하는 단계, 자동 반복 요청 메시지를 수신시, 상기 복수의 상기 순방향 에러 정정 코딩된 패킷들의 제 3 층을 송신하는 단계, 및 자동 반복 요청 콘텐츠 요청을 수신시, 상기 복수의 상기 순방향 에러 정정 코딩된 패킷들의 제 4 층과 상기 콘텐츠를 재송신하는 단계를 포함한다.

[0012] FEC 패킷들의 각 층은 다수의 멀티캐스트 그룹일 수 있다는 점이 주목되어야 한다. 즉, 전송한 제 2 층은 실제로 다수의 멀티캐스트 그룹에 의해 송신된 수 개의 하위층(sublayer)일 수 있다. 이러한 제 2 층의 경우, 수 개의 멀티캐스트 그룹이 존재하거나 존재할 수 있다. 각각의 멀티캐스트 그룹의 경우, 그 지연 시간은 상이하다(그리고 구성 가능하다). 클라이언트/모바일 디바이스는 이러한 층에서의 멀티캐스트 그룹들 모두에 가입할 필요는 없지만, 콘텐츠를 복구하기에 충분한 FEC/패리티 패킷들을 수신하기 위해 필요로 하는 멀티캐스트 그룹들에만 가입할 필요가 있다.

[0013] 본 발명은 첨부 도면과 관련하여 읽을 때 후속하는 상세한 설명으로부터 가장 잘 이해된다. 도면에는 아래에 간략히 설명된 후속하는 도면들이 포함된다.

발명의 효과

[0014] 본 발명의 순방향 에러 정정 및 자동 반복 요청의 조합한 것의 적용을 통해, 무선 통신 네트워크들에서 멀티캐스트 애플리케이션들의 신뢰도를 증가시킬 수 있다.

도면의 간단한 설명

[0015] 도 1은 핫스팟 멀티미디어 콘텐츠 WLAN캐스팅(WLAN을 통한 멀티캐스팅) 애플리케이션들에 대한 통상적인 무선 LAN 비디오 배분 시스템을 도시하는 도면.

도 2는 소스 스트림에 대한 FEC 그룹들의 3가지 유형을 도시하는 도면.

도 3은 본 발명의 병합된 하이브리드 ARQ 방법의 클라이언트/모바일 디바이스의 개략/블록도.

도 4a, 도 4b, 도 4c, 및 도 4d는 클라이언트 프록시 모듈에서의 처리의 3가지 줄거리의 흐름도.

도 5는 본 발명의 클라이언트 사이드(side) 프록시의 데이터 구조를 도시하는 도면.

도 6은 본 발명의 병합된 하이브리드 ARQ 방법의 서버 측의 개략/블록도.

발명을 실시하기 위한 구체적인 내용

[0016] 상이한 시나리오는 상이한 에러 정정 구조들을 사용할 수 있다. 일부 경우, FEC가 더 효율적일 수 있는데 반해, 다른 경우에는 ARQ가 더 나은 선택일 수 있다. 일부 시나리오에서는 이들 2가지 구조가 함께 조합되어 성능을 개선할 수 있다. 모든 애플리케이션 시나리오에서 단일 해결책을 제공하는 것이 유리할 수 있다.

[0017] 본 발명은 순방향 에러 정정과 자동 반복 요청의 조합을 적용함으로써 WLAN들에서 멀티캐스트 애플리케이션들의 신뢰도를 증가시키기 위한 적응 방법을 설명한다. 무선 방송/멀티캐스트 시스템에서, 패킷들/데이터는 랜덤한 버스트 분실을 겪을 수 있는데, 이는 다중-경로 페이딩, 간섭, 핸드오프 상황 등 때문이다. FEC와 ARQ를 조합하는 장점들이 존재한다. FEC와 ARQ를 결합시키는 것은 시스템 구현예의 유연성을 증가시키고 특정 시나리오에서 시스템의 성능을 개선할 수 있다. 본 발명의 mhARQ 시스템에서는, FEC 패킷들이 코딩되고 하나 이상의 적응 FEC 멀티캐스트 그룹과 ARQ 멀티캐스트 그룹으로 나누어진다. 그러므로 WLAN 방송/멀티캐스트 시스템은, 멀티캐스트 애플리케이션에서 ARQ 또는 적응 FEC 또는 둘 다를 사용하도록 구성될 수 있다. 즉, 본 발명은 멀티미디어 콘텐츠 애플리케이션들을 멀티캐스팅하기 위한 단일의 통합된 데이터/패킷 복구 해결책이다. 본 명세서에서 사용된 것처럼, "그룹"이라는 용어는 달리 표시되지 않는 한 멀티캐스트 그룹을 나타낸다.

[0018] 도 1은 핫스팟 멀티미디어 콘텐츠 WLAN캐스팅(WLAN을 통한 멀티캐스팅) 애플리케이션들에 대한 통상적인 무선 LAN 비디오 배분 시스템을 보여준다. 콘텐츠 서버들{주문형 콘텐츠(105)와 라이브 콘텐츠(110)}은 고속 이더넷(Ethernet) LAN(125)을 통해 무선 액세스 포인트(120a, 120b)에 연결된다. 콘텐츠 서버{주문형 콘텐츠(COD), 라이브 콘텐츠}(105, 110)는 고속 유선 네트워크를 통해, 무선 액세스 포인트(120a, 120b)에 하나 이상의 프로그램(오디오 및/또는 비디오)을 멀티캐스팅한다. 애플리케이션 층 FEC가 수행되고, FEC 패킷들은 지연되지 않은 및 지연된 FEC 멀티캐스트 그룹들 형태로 스트리밍 서버(115)에 의해 보내져, 에러 복구 능력을 제공한다. 액세스 포인트(120a, 120b)는 콘텐츠를 무선 링크들(135)을 통해 멀티캐스트로 무선 디바이스들(130a, 130b, 130c, 130d)에 콘텐츠를 배분한다. 무선 디바이스들(130a, 130b, 130c, 130d)의 사용자들은 하나 이상의 프로그램을 볼 수 있고 동시에 인터넷(140)에 액세스할 수 있다. 인터넷(140)에의 액세스는 무선 링크들(135), 적절한 WLAN 액세스 포인트(120a, 120b), 고속 이더넷 LAN(125), 및 이더넷 스위치(145), 통신 링크(150), 및 아마도 방화벽(155)을 통해 이루어진다. ARQ 서버(160)는 클라이언트/모바일 디바이스로부터의 NACK들을 수신하고, ARQ FEC 패킷들을 보낸다. 일부 경우, ARQ 서버와 스트리밍 서버는 하나의 장치/기계/디바이스에 공존할 수 있다. 클라이언트/모바일 디바이스는 지연된 FEC 패킷들이 잃어버린/손상된 패킷들/데이터를 복구하기에 충분하지 않다면, IGMP를 통해 ARQ FEC 그룹에 가입한다.

[0019] 콘텐츠/소스/비디오/오디오 스트림에 대한 FEC 패킷들은 상이한 층들로 나누어진다. 각각의 층은 상이한 멀티캐스트 그룹에서 송신된다. 도 2에 도시된 것처럼, 소스 스트림에 대한 3가지 유형의 FEC 그룹이 존재한다.

[0020] · 지연되지 않은 FEC 그룹들: 이들 그룹에서의 FEC 패킷들은 콘텐츠/소스로부터 지연되지 않는데, 즉 이들 FEC

패킷들은 그것들이 생성되자마자 보내진다. 랜덤한 분실 복구의 경우, 클라이언트/모바일 디바이스가 그것의 "장기간의(long-term)" 채널 상태에 기초하여 하나 이상의 지연되지 않은 FEC 그룹에 가입한다. 그 멤버십(membership)이 정적이거나 "반-정적(semi-static)"(분 단위로)이라면, 클라이언트/모바일 디바이스는 지연된 FEC 그룹들에 가입할 필요가 없거나 ARQ를 보낼 필요가 없다. 지연되지 않은 FEC는 클라이언트들/모바일 디바이스들로부터의 피드백 트래픽을 감소시킨다.

[0021] · 지연된 FEC 그룹들: 이들 그룹에서의 FEC 패킷들의 송신은 구성 가능한 시간만큼 매체 패킷들로부터 지연된다. 인터넷 그룹 관리 프로토콜(IGMP: Internet Group Management Protocol)을 사용하여, 클라이언트/모바일 디바이스는 지연되지 않은 FEC 패킷들이 데이터/콘텐츠 분실을 복구하기에 충분하지 않다면 지연된 FEC 그룹들을 동적으로 가입/탈퇴한다(FEC 블록마다). 가입 시간이 콘텐츠와 지연된 FEC 사이의 시간 이동(time shift)보다 크다면, 대응하는 패리티 패킷들이 수신되어 클라이언트/모바일 디바이스에 의해 잃어버린 콘텐츠/데이터 패킷들을 복구한다. 무선 대역폭을 절약하기 위해, 멀티캐스트 그룹에 대한 지연된 FEC 데이터는, AP/라우터와 연관된 어떤 클라이언트/모바일 디바이스도 이러한 특별한 그룹에 가입하지 않는다면, 무선 네트워크에서 AP/라우터(120a, 120b)에 의해 송신되지 않게 된다.

[0022] · ARQ FEC 그룹들: ARQ FEC 그룹들에서의 FEC 패킷들은 클라이언트들/모바일 디바이스들로부터의 NACK에 따라 보내진다. 클라이언트/모바일 디바이스는 그러한 NACK를 ARQ 서버에 보내고, 지연된 FEC 패킷들이 잃어버린 패킷들/데이터를 복구하기에 충분하지 않는다면, IGMP를 통해 대응하는 ARQ FEC 그룹들에 가입한다.

[0023] · ARQ 소스/콘텐츠 그룹들: 잃어버린 패킷들의 개수가 임계값을 초과하면, ARQ 서버가 멀티캐스트 그룹에서 매체 패킷들의 최초 블록을 재송신할 수 있다. 그 이유는 FEC가 사용되어, 잃어버린/손상된 패킷들/데이터 전부를 복구하거나 잃어버린/손상된 패킷들/데이터 중 그 어느 것도 복구하지 못하기 때문이다. 많은 잃어버린/손상된 매체 패킷들/데이터가 존재할 때, 수신기는 적어도 재송신된 매체 패킷들의 일부를 수신할 수 있다.

[0024] 유연성을 위해, 지연되지 않은 FEC 그룹과 지연된 FEC 그룹의 개수, 콘텐츠 스트림과 지연된 FEC 스트림 사이의 시간 이동, 및 지연되지 않은 FEC 그룹과 지연된 FEC 그룹, 및 ARQ FEC 그룹에서의 양이 구성될 수 있다.

[0025] 예컨대, 퀵타임(Quicktime), 톱슨 멀티미디어 애플리케이션 프레임워크(MMAF: Multimedia Application Framework), 및 비디오랜(VideoLAN) 클라이언트(VLC) 재생기와 같은, 이용 가능한 상용 및 프리웨어(freeware) 콘텐츠/비디오 재생기들은 FEC를 지원하지 않는다. 소스 코드는 일반적으로 상용 재생기들에 대해서는 이용 가능하지 않다. 프리웨어의 소스 코드가 이용 가능하지 않은 경우에도 소스 코드를 유지 및 갱신하는 것뿐만 아니라 FEC를 모든 프리웨어 재생기에 통합하는 것을 어렵다. 본 발명은 도 3에 도시된 클라이언트 프록시 아키텍처(architecture)를 포함한다. 클라이언트 프록시는 재생기 코드를 변경할 어떠한 필요성 없이, 임의의 상용 및 프리웨어 비디오 재생기들과 함께 작용할 수 있다. 클라이언트 프록시는 매체와 FEC 패킷들을 상이한 멀티캐스트 그룹들로부터 수신하고, FEC 멀티캐스트 그룹에 가입 및 탈퇴하며, ARQ 서버에 ARQ NACK들을 보내고, 재송신된 FEC 및 매체 패킷들을 수신하며, 잃어버린 매체 패킷들을 복구하고, 복구된 매체 패킷들을 통신 소켓(socket)을 통해 재생기에 보낸다.

[0026] 도 3을 참조하면, 클라이언트/모바일 디바이스가 WLAN 인터페이스(305)를 통해{WLAN 액세스 포인트(120a, 130b)를 통해}, 콘텐츠 서버들(105, 110)로부터의 콘텐츠를 수신한다. 클라이언트/모바일 디바이스는 동시에 공통의 사용자 데이터그램 프로토콜/인터넷 프로토콜(UDP/IP: User datagram protocol/internet protocol) 프로토콜 스택(315)을 통해, 분리된 흐름(310a, 310b)에 의해 각각 표시된 다수의 멀티캐스트 그룹들(콘텐츠, FEC, 지연된 FEC, ARG FEC, ARQ 소스/콘텐츠)의 멤버(member)일 수 있다. 공통의 UDP/IP 프로토콜 스택(315), 콘텐츠(320), 및 FEC 패킷들(325)은 분리되고, 본 발명의 FEC 클라이언트 프록시 모듈(330)에 들어간다. 복구된 콘텐츠는 저장 시스템(335)에 저장될 수 있고, 이러한 저장 시스템(335)은 나중의 재생을 위한 디스크, 테이프, CD들, 메모리, DVD들 등과 같은 임의의 형태의 저장 장치를 포함할 수 있거나, 복구된 콘텐츠가 통신 소켓을 통해 재생기(또는 임의의 적합한 그리고 이용 가능한 디스플레이 디바이스)(355)에 보내질 수 있으며, 이러한 통신 소켓은 UDP/IP 프로토콜 스택(340, 345)과 디스플레이 디바이스(355)에서의 UDP/IP 프로토콜 스택(350)을 포함한다.

[0027] 콘텐츠 서버측에서는, FEC 인코딩 동작이 수행되기 전에, 일정한 개수의 매체 패킷들(K)이 버퍼링된다. 멀티미디어 콘텐츠는 비디오 트랙과 오디오 트랙과 같은 다수의 트랙을 포함할 수 있다. FEC 인코딩 동작은 각각의 트랙에서 각각 수행된다. 상이한 매체 트랙들은 상이한 비트 속도를 가지고, 보통 비디오는 오디오보다 높은 비트 속도를 가진다. 비디오와 오디오 모두에 대해 동일한 FEC 블록 크기를 사용하게 되면 수신기에서 오디오/비디오 동기화 문제를 야기하게 된다. 예컨대, 비디오 버퍼에서의 비디오 패킷들의 개수가 K에 도달하게 되면, FEC 인코딩이 이들 패킷에서 수행되고, 비디오 FEC 패킷들이 내보내지며, 버퍼가 그 다음 블록에 대해서 비워지는데,

즉 이와 동시에 오디오 버퍼에서의 오디오 패킷들의 개수가 여전히 그것이 K에 도달하기 전에 특정되지 않은 시간을 여전히 기다려야 할 수 있다. 한 가지 접근법은 비디오와 오디오에 대해 상이한 블록 크기를 사용하는 것이다. 하지만 특정 블록 크기의 비디오에 대해, 오디오에 대한 블록 크기를 어떻게 선택하는지는 설계 문제가 된다. 상이한 콘텐츠는 오디오 트랙과 비디오 트랙 사이의 상이한 블록 크기 짝짓기(pairing)를 필요로 할 수 있다. 심지어 동일한 콘텐츠의 경우에도, 다양한 비트 속도의 콘텐츠가 상이한 시각에서의 오디오 및 비디오에 대해 상이한 K_s 를 필요로 할 수 있다.

[0028] 다양한 비트 속도 스트림의 경우, 버퍼에서 고정된 개수의 매체 패킷들을 채우기 위한 시간이 달라질 수 있다. FEC 버퍼링 때문에 생길 수 있는 재생 지터(playback jitter)를 추가로 해결하기 위해, 본 발명에서의 FEC 블록에 대해서는 최대 버퍼 시간(maxT)이 설정된다. 비디오 또는 오디오 FEC 버퍼에서의 첫 번째 패킷에 대한 버퍼링 시간이 maxT 이상이거나, 버퍼에서의 한 트랙에 대한 비디오(또는 오디오) 패킷들의 개수가 maxK(최대 버퍼 크기)에 도달한다면, 멀티미디어 콘텐츠의 비디오 트랙과 오디오 트랙 모두의 패킷들에 관한 FEC 인코딩이 수행된다.

[0029] 실시간 애플리케이션의 경우, 다양한 N과 K를 사용하는 것이 오디오 및 비디오 동기화 문제를 해결할 수 있지만, 여분의 인코딩 및 디코딩 오버헤드를 야기할 수 있는데, 이는 상이한 N과 K에 대한 생성 매트릭스가 상이하고, 생성 매트릭스들을 형성하는 것이 매우 계산상 고가이기 때문이다. 인코딩 및 디코딩을 효율적으로 개선하기 위해, 본 발명에서는 maxK와 maxN 모두 고정된다. 생성 매트릭스는 maxK와 maxN에 기초하여 형성되고, 서버 측과 클라이언트 측 모두에 관해 한 번만 초기화된다. 그러므로 모든 (N,K) 코드에 대해, 구멍이 뚫린(punctured)/쪼아진 코드들에 기초한 초기 생성 매트릭스를 사용하여 $K \leq \max K$ 이고, $N \leq \max N$ 이 생성될 수 있고, 이는 인코딩 및 디코딩 시간을 극적으로 감소시킨다.

[0030] 클라이언트 프록시(330)의 기능에 대한 세부 사항은 도 4a, 도 4b, 도 4c, 및 도 4d에 상세히 도시되어 있다. 클라이언트 프록시의 기능/행위/단계들은 소프트웨어, 하드웨어, 펌웨어, 애플리케이션 특정 집적 회로(ASIC: application specific integrated circuit)들 및/또는 필드 프로그래밍 가능한 게이트 어레이(FPGA: field programmable gate array)들, 또는 이러한 것들의 임의의 조합 또는 임의의 다른 적합한 수단으로 구현될 수 있다. 클라이언트 프록시는 FEC와 매체 패킷들을 수신하고 버퍼링하며, 패킷 분실들에 기초하여 채널 상태들을 추정한다. 클라이언트 프록시가 FEC 블록으로부터 본래의 메시지를 복구할 수 없다면, 클라이언트 프록시는 더 많은 FEC 그룹들에 가입하고/가입하거나 더 많은 FEC 또는 본래의 매체 패킷들의 재송신을 위한 ARQ 요청을 보낸다. ARQ 피드백 억제 알고리즘은 또한 클라이언트 프록시에서 구현된다. 일부 ARQ가 분실/손상되지만 FEC 블록이 디코딩될 수 있다면, 클라이언트 프록시는 패리티(FEC) 패킷들로부터 분실/손상된 패킷들을 복구하고 복구된 매체 패킷들을 재생기에 보낸다. 패킷들이 분실/손상되고, 클라이언트 프록시가 FEC 블록을 디코딩할 수 없다면, 버퍼 타이머가 종료할 때, 심지어 일부 매체 패킷들이 분실되고 복구될 수 없는 경우에도 클라이언트 프록시는 수신된 매체 패킷들을 재생기에 보내게 된다.

[0031] 클라이언트 프록시의 일들은 3개의 독립된 줄거리에 의해 수행된다. 주 줄거리는 매체/패리티 패킷들을 수신하고 버퍼링한다. 주 줄거리는 또한 멀티캐스트 그룹에서 다른 클라이언트/모바일 디바이스로부터 ARQ 요청들을 수신 및 처리하게 된다. 주 줄거리는 또한 채널 추정, 적응 FEC 처리, 및 FEC 디코딩을 수행하게 된다. ARQ 처리 및 패킷 발송은 각각 제 2 줄거리와 제 3 줄거리에 의해 수행된다.

[0032] 도 3의 클라이언트 프록시 모듈(330)에서 실시된 방법을 함께 보여주는 도 4a, 도 4b, 도 4c, 및 도 4d를 다시 참조한다. 도 4a는 주 루틴(main routine)이고, 소스 버퍼는 405에서 초기화된다. 410에서는, 패킷 발송 공정(제 2) 줄거리가 불러내어 진다. 415에서는 ARQ 이벤트 큐(event queue)가 초기화된다. ARQ 이벤트 공정(제 3) 줄거리는 420에서 불러내어 진다. 마지막으로, 425에서는 매체 패킷 수신, 매체 패킷들의 버퍼링, 및 에러 복구 등을 다루는 주/제 1 줄거리가 불러내어 진다.

[0033] 도 4b는 도 3의 클라이언트 프록시 모듈(330)에 의해 실시된 방법의 주/제 1 줄거리의 흐름도이다. 430에서 패킷이 수신된다. 채널 상태들은 435에서 추정된다. 440에서는 수신된 패킷이 FEC 패킷인지를 결정하는 테스트가 수행된다. 수신된 패킷이 FEC 패킷이라면, 그 패킷은 465에서 처리된다. 클라이언트/모바일 디바이스가 패리티(FEC) 패킷을 수신하면, 클라이언트/모바일 디바이스는 FEC 블록 헤더로부터 이러한 패리티 패킷이 속하는(패리티/FEC 패킷이 포함되는) FEC 블록에 대한 모든 정보를 얻는다. 클라이언트/모바일 디바이스는 먼저 블록 데이터 구조가 FEC 블록에 할당되었는지를 확인한다. 블록 데이터 구조가 아직 할당되지 않았다면, 블록 데이터 구조가 할당된다. FEC 블록이 그것의 기본 시퀀스 번호에 따라 데이터 구조(예컨대, 링크된 목록)에 삽입된다. FEC 블록의 기본 시퀀스 번호는 이 FEC 블록에 속하는(포함되는) 매체 패킷들의 가장 작은 시퀀스 번호이다.

FEC 블록이 할당된 후, 클라이언트/모바일 디바이스가 소스 버퍼를 조사하고 이 FEC 블록에 속하는 매체 패킷들을 표시하며, 이 블록에 대응하는 정보(얼마나 많은 패킷이 분실되는지, 어느 패킷이 분실되는지 등)를 갱신한다. 485에서 FEC 블록이 디코딩 가능하였는지를 결정하는 테스트가 수행된다. FEC 블록이 할당된 후에는, 클라이언트/모바일 디바이스가 소스 버퍼를 조사하고 이 블록에 대응하는 정보(얼마나 많은 패킷이 분실되는지, 어느 패킷이 분실되는지 등)를 검색하고, FEC 디코딩 공정에서 검색된 정보를 사용할 수 있다. FEC 블록이 디코딩 가능하지 않았다면, 적응 FEC 알고리즘이 495에서 불리내어 진다. 즉, 클라이언트 프록시는 지연된 FEC 멀티캐스트 그룹에 가입하고 지연된 FEC 패킷들을 요청한다. 그런 다음 497에서 ARQ가 필요한지를 결정하는 테스트가 수행된다. 지연된 FEC 패킷들이 분실/손상된 패킷들/데이터를 복구하기에 불충분하다면 ARQ가 필요할 것이다. 본 명세서에서 ARQ는 ARQ FEC 패킷들과 ARQ 소스/콘텐츠 패킷 요청들을 포함한다. 이는 ARQ FEC 및 ARQ 소스/콘텐츠 멀티캐스트 그룹에 가입하는 것을 포함한다. ARQ가 필요하다면, 499에서 ARQ 요청이 ARQ 이벤트 큐에 삽입된다. 그런 다음 처리가 430으로 되돌아간다. ARQ가 필요하지 않으면, 처리는 430으로 되돌아간다.

[0034]

FEC 블록이 디코딩 가능하였다면, 490에서 복구된 패킷(들)이 소스 버퍼에 삽입된다. FEC가 디코딩된 후 수신된 매체 또는 FEC 패킷들이 버려진다. FEC 블록에 대한 모든 매체 패킷들이 디스플레이로 보내질 때, FEC 블록 구조는 클리어되고 메모리는 비게 된다. 그런 다음 491에서 적응 FEC가 활성화되어 클라이언트가 불필요한 패리티 패킷들을 수신하였는지와, 이미 가입한 FEC 그룹들을 떠날지/탈퇴할지를 결정한다. 그런 다음 처리는 430으로 되돌아간다.

[0035]

수신된 패킷이 FEC 패킷이 아니라면, 445에서 수신된 패킷이 매체 패킷인지를 결정하는 테스트가 수행된다. 수신된 패킷이 매체 패킷이라면, 그 매체 패킷은 470에서 처리된다. 클라이언트/모바일 디바이스가 매체 패킷을 수신할 때, 수신된 매체 패킷은 시퀀스 번호에 따라 데이터 구조(예컨대, 링크된 목록)에 삽입된다. 이 패킷이 속하는 FEC 블록이 할당된 후 매체 패킷이 수신되면, 클라이언트/모바일 디바이스가 매체 패킷의 시퀀스 번호, 기본 시퀀스 번호, FEC 블록의 N과 K에 기초하여 링크된 목록으로부터 이러한 FEC 블록 구조의 위치를 정할 수 있다. 클라이언트/모바일 디바이스는 이러한 매체 패킷에 표시를 하고 FEC 블록 정보를 갱신한다. 그런 다음 처리는 485로 진행한다.

[0036]

일정한 FEC 블록에 대한 패리티 패킷들이, 예컨대 핸드오프에서 총체적으로 잃어버려지는 것이 또한 가능하다. 클라이언트/모바일 디바이스는 이러한 FEC 블록이 속하는 매체 패킷들에 대한 임의의 FEC 정보를 얻지 못할 수 있다. 마지막으로 표시되지 않은 매체 패킷과 처음에 표시되지 않은 매체 패킷 사이의 차이가 일정한 임계값(예컨대, K의 추정된 값의 3/2)을 초과한다면, 클라이언트/모바일 디바이스는 이들 매체 패킷들에 대한 모든 FEC 패킷이 분실/손상된다고 결론을 내린다(FEC 블록은 디코딩 가능하지 않음). 그런 다음 클라이언트/모바일 디바이스는 이들 매체 패킷들에 대한 임의의 FEC 정보 없이 본 발명의 적응 FEC 알고리즘이나 ARQ 요청 처리를 활성화하게 된다.

[0037]

수신된 패킷이 매체 패킷이 아니라면, 450에서 수신된 패킷이 ARQ 패킷인지를 결정하는 테스트가 수행된다. 수신된 패킷이 ARQ 패킷이라면, ARQ 피드백의 급격한 증가 문제를 방지하기 위해, 475에서 ARQ 억제가 수행된다. 처리는 그 후 430으로 진행한다.

[0038]

수신된 패킷이 ARQ 패킷이 아니라면, 455에서 수신된 패킷이 RTCP 패킷인지를 결정하는 테스트가 수행된다. 수신된 패킷이 RTCP 패킷이라면, 480에서 RTCP 패킷이 처리된다. 수신된 RTCP 패킷들은 그것들이 수신된 시간에 따라 소스 버퍼에 삽입되는데, 이는 RTCP 패킷들에서의 시퀀스 번호들이 RTP 패킷들에 따라 지정되지 않기 때문이다. 그런 다음 430으로 처리가 되돌아간다. 수신된 패킷이 RTCP 패킷이 아니라면, 수신된 패킷은 460에서 버려진다. 그런 다음 처리는 430으로 되돌아간다.

[0039]

도 4c는 도 3의 클라이언트 프록시 모듈(330)에 의해 실시된 방법의 두 번째 줄거리의 흐름도이다. 이 두 번째 줄거리는 소스 패킷 발송을 다룬다. 433에서는 소스 버퍼가 비어있는지를 결정하기 위한 테스트가 수행된다. 소스 버퍼가 비어 있다면, 처리는 되돌아가고 소스 버퍼가 더 이상 비어 있지 않을 때까지 대기 상태에 있다. 433에서는 소스 버퍼의 상태에 관한 결정이 이루어질 수 있도록, 일부 시간에 기초하여 또는 임의의 다른 편리한 것에 기초하여 테스트가 수행될 수 있다. 소스 버퍼가 비어 있지 않다면, 443에서 헤더 패킷에 대한 패킷 버퍼 시간이 만료되었는지를 결정하는 테스트가 수행된다. 헤더 패킷은 다루어질 버퍼에서의 그 다음 패킷이다. 본 발명의 방법이 링크된 목록 데이터 구조를 사용할 수 있으므로, 버퍼에서의 그 다음 패킷은 그 구조에서의 다음 패킷이 아닐 수 있지만, 오히려 다루어질 그 다음 패킷은 링크된 목록 포인터(pointer)에 의해 지시될 그 다음 패킷일 것이다. 버퍼 시간이 만료되었다면, 453에서 헤더 패킷이 매체 재생기나 매체 디스플레이 디바이스에 발송된다. 즉, 수신된 패킷 또는 RTCP 패킷이 버퍼에서 시간 T보다 긴 시간 동안 남아있다면, 클라이언트/모

파일 디바이스가 그 패킷을 매체 재생기에 발송한다. 헤더 패킷에 관한 포인터는 463에서 소스 버퍼에 있는 그 다음 패킷으로 설정된다. 그런 다음 처리는 433으로 되돌아간다. 헤드 패킷 버퍼 시간이 만료되지 않았다면, 처리는 433으로 되돌아간다.

[0040] 도 4d는 도 3의 클라이언트 프록시 모듈(330)에 의해 실시된 방법의 세 번째 줄거리의 흐름도이다. 이 세 번째 줄거리는 ARQ 처리를 다룬다. 437에서는 이벤트 큐 버퍼가 비어 있는지를 결정하는 테스트가 수행된다. 이벤트 큐 버퍼가 비어 있다면 처리는 되돌아가고, 이벤트 큐 버퍼가 더 이상 비어 있지 않을 때까지 대기 상태에 있다. 437에서의 테스트는 이벤트 큐 버퍼의 상태에 관한 결정이 이루어질 수 있도록, 일부 시간에 기초하여 또는 임의의 다른 편리한 것에 기초하여 테스트가 수행될 수 있다. 이벤트 큐 버퍼가 비어 있지 않다면, 헤드 이벤트 타이머가 만료되었는지를 결정하는 테스트가 447에서 수행된다. 헤드 이벤트 타이머는 이벤트 큐 내의 다음 이벤트를 처리하기 위한 타이머이다. 헤드 타이머가 만료되었다면, 457에서 ARQ 요청이 보내진다. 헤드 이벤트 타이머는 467에서 이벤트 큐에서의 그 다음 이벤트를 지시하도록 설정된다. 그런 다음 처리는 437로 되돌아간다. 헤드 이벤트 타이머가 만료되지 않았다면, 처리는 437로 되돌아간다.

[0041] 클라이언트 측에서의 주 데이터 구조는 도 5에 도시되어 있다. 클라이언트/모바일 디바이스는 작업 시간에서 매체 스트림의 각각의 트랙에 대한 소스 버퍼(매체 버퍼)를 유지한다. 소스 버퍼는 매체 패킷들을 T초간 버퍼링한다. 소스 버퍼는 링크된 목록 데이터 구조를 사용하여 구현될 수 있다. 아래에 설명된 본 발명의 데이터 구조들은 도 4a, 도 4b, 도 4c, 및 도 4의 공정에 의해 사용된다. 이러한 데이터 구조들은 도 4a, 도 4b, 도 4c, 및 도 4의 공정에서 시간상 여러 지점에서, 그리고 여러 목적을 위해 사용된다.

[0042] FEC 블록은 하나 이상의 매체 패킷(k개의 매체 패킷들까지)을 포함한다. 그러므로, 매체 패킷은 FEC 블록에 속한다고 말해진다. FEC 블록은 또한 하나 이상의 FEC(패리티) 패킷(n-k개의 패리티/FEC 패킷들까지)을 포함한다. 그러므로, FEC/패리티 패킷들은 FEC 블록에 속한다고 말해진다. 여기서 n은 FEC 블록 크기이고, k는 FEC 블록에서의 매체 패킷들의 개수이다. 매체 패킷에는 그것이 속하는 FEC 블록이 할당되었는지가 표시된다. 그러므로, FEC 블록이 할당되었다면, 클라이언트 모바일 디바이스는 매체 패킷의 시퀀스 번호, 기본 시퀀스 번호, 및 FEC 블록의 N과 K에 기초한 링크된 목록으로부터 이러한 FEC 블록 구조의 위치를 정할 수 있다.

[0043] 본 발명의 구현예에서, 서버측에서의 FEC 인코딩 동안, 매체 패킷들은 수정되지 않는다. FEC 인코딩 파라미터들에 대한 모든 정보는 FEC 패킷들에 포함된다. 이 구현예는 역 호환 가능한데, 이는 심지어 클라이언트/모바일 디바이스가 FEC를 지원하지 않는 경우에도, 여전히 매체 멀티캐스트 그룹에 가입하고, 매체 패킷들을 수신하며, 수신된 콘텐츠를 재생할 수 있기 때문이다.

[0044] 클라이언트/모바일 디바이스가 매체 패킷을 수신할 때에는, 수신된 매체 패킷이 505a, 505b, ..., 505a+1에서 도시된 시퀀스 번호에 따라 소스 버퍼 데이터 구조(예컨대, 링크된 목록)에 삽입된다. 수신된 매체 패킷이 속하는 FEC 블록 데이터 구조가 아직 할당되지 않았다면, 클라이언트/모바일 디바이스가 이러한 매체 패킷이 속하는 FEC 블록에 대한 정보를 가지고 있지 않다. 매체 패킷은 FEC 블록에 대한 임의의 정보를 포함하지 않는다. 또한, 본 발명은 FEC 인코딩에서 고정된 N과 K를 사용하지 않고(오히려 MaxN과 MaxK가 고정된다), FEC 블록 인덱스 정보는 그러한 시퀀스 번호로부터 유도되지 않는다. 매체 패킷이 속하는 FEC 블록에 대한 정보가 없다면, 매체 패킷은 표시되지 않는다.

[0045] 매체 트랙에 대한 실시간 수송 제어 프로토콜(RTCP: real-time transport control protocol) 패킷들이 또한 대응하는 소스 버퍼 데이터 구조(예컨대, 링크된 목록)에 삽입된다는 점을 주목해야 한다. RTCP 패킷들에서의 시퀀스 번호들은 RTP 패킷들에 따라 지정되지 않는다. 오히려 RTCP 패킷들이 그러한 RTCP 패킷들이 수신된 시간에 따라 소스 버퍼에 삽입된다.

[0046] 매체 또는 RTCP 패킷이 미리 결정된 또는 구성 가능한 시간 T보다 더 긴 시간 동안 버퍼에 남아 있다면, 클라이언트/모바일 디바이스가 그 패킷을 매체 재생기에 보낸다.

[0047] 클라이언트/모바일 디바이스는 또한 FEC 블록 버퍼 데이터 구조를 유지한다. 이러한 데이터 구조는 또한 링크된 목록으로서 구현될 수 있고, 링크된 목록에서의 각각의 요소는 기본 시퀀스 번호, N, 및 K 등과 같은 FEC 블록에 대한 모든 정보를 포함하는 FEC 블록 데이터 구조이다.

[0048] 클라이언트/모바일 디바이스가 패리티(FEC) 패킷을 수신할 때에는, 그 클라이언트/모바일 디바이스가 이러한 패리티 패킷이 속하는 FEC 블록에 대한 모든 정보를 FEC 블록 헤더로부터 얻는다. 클라이언트/모바일 디바이스는 먼저 블록 데이터 구조가 FEC 블록에 할당되었는지를 확인한다. 블록 데이터 구조가 아직 할당되지 않았다면, 블록 데이터 구조가 할당된다. FEC 블록 데이터 구조는 510a, 510b, ..., 510β+1에서 도시된 기본 시퀀스 번호에

따라 데이터 구조(예컨대, 링크된 목록)에 삽입된다. FEC 블록의 기본 시퀀스 번호는 이러한 FEC 블록에 속하는 매체 패킷들의 가장 작은 시퀀스 번호이다. FEC 블록이 할당된 후, 클라이언트/모바일 디바이스는 소스 버퍼를 조사하고, 이러한 FEC 블록에 속하는 매체 패킷들에 표시를 하며, 이러한 블록에 대응하는 정보(얼마나 많은 패킷이 분실되는지, 어느 패킷이 분실되는지 등과 같은)를 갱신한다. 이러한 정보는 본 발명의 적응 FEC에서 사용된다. 이러한 정보는 또한 FEC 디코딩(도 4b의 485,490)에서 사용된다. 그 후, 매체 패킷 또는 FEC 패킷이 이러한 FEC 블록에 대해 수신된다면, 이러한 블록의 정보는 그에 따라 갱신된다.

- [0049] 매체 패킷이 이러한 패킷이 속하는 FEC 블록이 할당된 후 수신되면, 클라이언트/모바일 디바이스가 매체 패킷의 시퀀스 번호, 기본 시퀀스 번호, 및 FEC 블록의 N과 K에 기초하여 링크된 목록으로부터 이러한 FEC 블록 구조의 위치를 정할 수 있다. 클라이언트/모바일 디바이스는 이러한 매체 패킷에 표시를 하고 FEC 블록 정보를 갱신한다.
- [0050] FEC 블록 데이터 구조는 이러한 블록에 속하는 모든 매체 패킷을 가리키는 포인터들의 배열(array)을 포함한다. FEC 블록 데이터 구조는 또한 이러한 블록에 대한 FEC 패킷들을 저장하기 위해 사용되는 디코드 버퍼를 포함한다. FEC 디코딩을 수행하기 위해 요구되는 메모리 버퍼 또한 디코드 버퍼에서 할당된다.
- [0051] FEC 블록이 디코딩 가능하다면, FEC 디코딩이 수행되고, 복구된 매체 패킷들이 소스 버퍼에 삽입된다. FEC 블록이 디코딩된 후 수신된 매체 또는 FEC 패킷들은 버려진다. FEC 블록에 대한 모든 매체 패킷들이 디스플레이에 보내지면, FEC 블록 구조가 클리어되고 메모리 비워진다.
- [0052] 일정한 FEC 블록에 대한 패리티 패킷들이, 예컨대 핸드오프에서 총체적으로 분실되는 것이 또한 가능하다. 클라이언트/모바일 디바이스는 이들 매체 패킷들에 대한 임의의 FEC 정보를 얻지 못할 수 있다. 클라이언트/모바일 디바이스는 또한 소스 버퍼에서의 첫 번째 표시되지 않은 매체 패킷{도 5에서, pkt-(a+1)이 첫 번째 표시되지 않은 패킷이다}으로 포인터를 유지한다.
- [0053] 마지막으로 표시되지 않은 매체 패킷과 첫 번째 표시되지 않은 매체 패킷 사이의 차이가 일정한 임계값(예컨대, K의 추정된 값의 3/2)을 초과한다면, 클라이언트/모바일 디바이스는 이들 매체 패킷들에 대한 모든 FEC 패킷이 분실/손상된다고 결론을 내린다(FEC 블록은 디코딩 가능하지 않음). 그런 다음 클라이언트/모바일 디바이스는 이들 매체 패킷들에 대한 임의의 FEC 정보 없이 본 발명의 적응 FEC 알고리즘이나 ARQ 요청 처리를 활성화하게 된다. 본 발명에서 K는 고정되지 않기 때문에, 클라이언트/모바일 디바이스는 K의 평균값의 추정을 유지한다.
- [0054] 각각의 클라이언트/모바일 디바이스는 이벤트 큐 데이터 구조를 유지한다. 이벤트는 클라이언트/모바일 디바이스가 ARQ 요청을 보내거나 ARQ 역제를 처리하기 위한 모든 정보를 포함하는 ARQ 처리 이벤트이다. 각각의 이벤트는 515a, 515b, ..., 515x+1에서 도식된 그것들의 만료 시간에 따라 이벤트 큐에 삽입된다. 이벤트는 또한 확인(check) 이벤트일 수 있다. 예컨대, 클라이언트/모바일 디바이스가 ARQ 요청을 보낸 후, 그 클라이언트/모바일 디바이스는 그것이 재송신 패리티 패킷들을 수신하였는지와 FEC 블록을 디코딩할 수 있었는지를 결정하기 위해, 그 클라이언트/모바일 디바이스가 나중에 확인하도록 확인 이벤트를 이벤트 큐에 삽입할 수 있다. 클라이언트/모바일 디바이스가 재송신 패킷을 수신하고 디코딩하지 않았다면, 그 클라이언트/모바일 디바이스는 또 다른 ARQ 요청을 보낼 수 있다.
- [0055] 멀티캐스트 작업 시간은 매체 그룹에 의해 나타난 하나의 매체 그룹을 가진다. 멀티캐스트 작업 시간에서의 각각의 매체 트랙은 또한 다수의 FEC 그룹을 가진다. FEC 그룹들의 개수는, 예컨대 위에서 논의된 것처럼 오디오 트랙과 비디오 트랙에 대해 상이할 수 있다. 적응 FEC와 ARQ 알고리즘이 따로따로 각각의 매체 트랙에 대해 배치된다. 다음 논의에서는 적응 FEC와 ARQ 알고리즘을 설명하기 위해 하나의 매체 트랙만을 가지는 멀티캐스트 작업 시간이 가정된다. 다수의 트랙을 구비한 멀티캐스트 작업 시간에 대해서는, 이러한 방법이 각각의 트랙에 대해 사용될 수 있다.
- [0056] 스트림이 총 m+1개의 FEC 그룹(그룹(0), 그룹(1), ..., 그룹(m))을 가진다고 가정하면, 그룹(0) 내지 그룹(m-1)이 적응 FEC 그룹이고, 그룹(m)이 ARQ에 대한 것이다. 첫 번째 FEC 그룹은 매체 그룹(지연되지 않은 FEC 그룹) 직후 보내진다. 클라이언트는 항상 매체 그룹과 첫 번째 FEC 그룹에 가입하게 된다. 이후, 그 그룹 숫자가 그 그룹과 연관된 파라미터들에 색인을 붙이기 위해 사용된다. 일 예로서, 그룹(i)의 오버헤드는 오버헤드(i)가 되고, 그룹(i)의 지연은 지연(i)이 된다. FEC 코드가 n인 블록 크기를 가지면, 각각의 블록에서의 메시지 심벌들의 개수는

$$k = \frac{n}{1 + \sum_{i=0}^m overhead(i)}$$

[0057]

이 된다.

[0058]

각 그룹에서의 패리티 심벌들의 개수는

[0059]

$$num_parity(i) = overhead(i) * k$$

[0060]

이다. 클라이언트/모바일 디바이스는 각각 *avg_loss_rate*와, *avb_variance*로 수신된 FEC 블록들에 대한 평균 손실률과 그것의 분산을 추정한다.

[0061]

클라이언트/모바일 디바이스가 FEC 블록에 대한 첫 번째 FEC 패킷을 수신할 때에는, 그 클라이언트/모바일 디바이스가 현재의 FEC 블록의 마지막 매체 패킷/심벌을 수신하였음을 의미한다.

[0062]

FEC 헤더로부터, 클라이언트/모바일 디바이스는 이러한 FEC 블록에 대한 모든 정보를 추출하고, 임의의 매체 패킷들이 분실되었는지와 이러한 FEC 블록에 대해 얼마나 많은 패킷이 분실되었는지에 대한 지식을 가진다. 이러한 정보는 평균 손실과 분산의 추정을 갱신하기 위해 사용된다. 이러한 점에서, 클라이언트/모바일 디바이스가 현재의 FEC 블록을 디코딩하기 위해 이미 충분한 패킷들/심벌들을 수신하였다면, 그 클라이언트/모바일 디바이스는 임의의 새로운 FEC 그룹에 가입하거나 ARQ 요청을 보낼 필요는 없다. 하지만, 클라이언트/모바일 디바이스가 더 많은 FEC 그룹에 가입하였다면, 클라이언트/모바일 디바이스가 여분의 패리티 패킷/심벌을 수신할 수 있다. 즉, 클라이언트/모바일 디바이스는 그 클라이언트/모바일 디바이스가 필요로 하는 것보다 많은 패리티 패킷/심벌을 수신할 수 있다. 클라이언트는 이들 FEC 그룹들을 떠날(탈퇴할) 필요가 있을 수 있다.

[0063]

클라이언트가 현재의 FEC 블록을 디코딩하기 위해 충분한 심벌을 수신하지 않는다고 가정한다. 분실된 매체/메시지 심벌들의 개수는 $lost_msg = k - recv_msg$ 이다. 클라이언트/모바일 디바이스는 가입할 필요가 있는 FEC 그룹들의 개수를 결정할 필요가 있다. 일 실시예는 한 번에 모든 그룹에 가입하는 것이다. 제 2 실시예는 처음에 하나의 추가 FEC 그룹에 가입하는 것이다. 일부 지연 후, 클라이언트/모바일 디바이스는 그것이 FEC 블록을 디코딩하기 위해 충분한 패킷/심벌을 수신하였는지를 결정하는 것을 확인한다. 충분한 패킷/심벌을 수신하지 않았다면, 클라이언트/모바일 디바이스는 그 다음 FEC 그룹에 가입한다.

[0064]

제 1 실시예에서, 필요로 하는 심벌들의 개수는

[0065]

$$req_sym = lost_msg * (1 + avg_loss_rate) + \alpha * avg_variance$$

[0066]

에 의해 추정될 수 있다.

[0067]

무선 네트워크들에서, 패킷 손실 분산은 매우 클 수 있다. 그러므로, 대부분의 시간 동안 클라이언트/모바일 디바이스는 필요로 하는 패리티 심벌들을 과대추정할 수 있다. 하지만, 여전히 요청된 심벌들을 과소추정할 수 있고, 이는 디코딩할 수 없는 FEC 블록(들)을 야기한다. 클라이언트/모바일 디바이스가 가입할 필요가 있는 그룹들의 개수는

[0068]

수학식 1

$$g = \min(j : \sum_{i=1}^j num_parity(i) \geq req_sym) \quad j \leq m - 1$$

[0069]

이 된다.

[0070]

클라이언트/모바일 디바이스가 멀티캐스트 FEC 그룹에 가입할 때, 그 클라이언트/모바일 디바이스는 그것이 이 그룹에 가입한 채로 있어야 하는 지속 기간을 명시할 필요가 있다. 이는 *grp_expire* 파라미터에 의해 명시된다.

[0071]

수학식 2

[0072] $grp_expire(i) = current_time + delay(i) + Te \quad i \leq m-1$

[0073] 여기서 Te 는 예컨대, $Te=500ms$ 와 같은 구성 가능한 파라미터이다. 클라이언트/모바일 디바이스가 모든 적응 FEC 그룹에 가입하지만 여전히 FEC 블록을 디코딩할 수 없다면, 그 클라이언트/모바일 디바이스는 ARQ 요청을 보내야 한다. 피드백의 급격한 증가에 대처하기 위해, ARQ 요청은 바로 보내지지 않는다. ARQ 요청 타이머가 이벤트 큐에 삽입되고, 타이머의 만료 시간은 $(K - lost_msg) * T_r$ 과 $(K - lost_msg + 1) * T_r$ 사이로 설정된다. 이러한 식으로, 대부분의 패킷을 분실한 클라이언트/모바일 디바이스가 ARQ 요청을 처음에 보내게 된다. ARQ 요청은 ARQ 요청 멀티캐스트 그룹에 멀티캐스팅된다. 다른 클라이언트/모바일 디바이스들은 이러한 ARQ 요청을 수신하게 되면 수신된 ARQ 요청에서의 요청된 패리티 패킷들의 개수를 그 자신이 필요로 하는 것과 비교하게 된다. 또 다른 클라이언트/모바일 디바이스에 의해 요청된 패리티 패킷들의 개수가 그 자신이 요청한 것보다 크다면, 클라이언트/모바일 디바이스가 그 자신의 ARQ 요청을 억제하고, 그 자신이 필요로 하는 것보다 많은 패리티 패킷들을 가지게 될 멀티캐스트를 얻기를 기다린다.

[0074] 제 2 실시예에서는, 클라이언트/모바일 디바이스가 먼저 하나의 추가 지연된 FEC 그룹에 가입한다. 타이머가 설정된다. 타이머는 때때로 그 다음 지연된 FEC 그룹의 지연 시간 전에 만료된다. 클라이언트/모바일 디바이스가 그 자신이 FEC 블록을 여전히 디코딩할 수 없다고 결정하면, 그 클라이언트/모바일 디바이스는 그 다음 FEC 그룹에 가입할 시간을 가진다. 예컨대, IGMP 가입 레이턴시(latency)가 수ms부터 약 50ms까지 사이에 존재한다면, 그 다음 FEC 그룹 지연 시간 전에 타이머 만료 시간이 $T_x=100ms$ 로서 명시된다. 즉,

수학식 3

[0075] $timer_expire(i) = current_time + delay(i+1) - T_x$

[0076] 여기서, T_x 는 구성 가능한 파라미터이다. 또한, FEC 그룹에 이미 가입되었다면, 클라이언트/모바일 디바이스는 단지 그 그룹의 만료 시간을 갱신할 필요가 있지만, 타이머는 여전히 설정되어야 한다. 클라이언트/모바일 디바이스가 마지막 적응 FEC 그룹에 가입하였지만, 여전히 FEC 블록을 디코딩할 수 없다면, 그 클라이언트/모바일 디바이스는 ARQ 요청을 보내야만 하게 된다.

[0077] 일부 시나리오에서, FEC 블록에서의 분실된 매체 패킷들의 개수가 임계값(예컨대, K의 50%)을 초과한다면, 패리티 패킷들만을 요구하는 ARQ를 보내는 것을 효율적이지 못할 수 있다. 이러한 경우, 클라이언트/모바일 디바이스는 본래의 매체 패킷들의 재송신을 요청하는 ARQ를 보낼 수 있다.

[0078] 클라이언트/모바일 디바이스가 FEC 블록을 디코딩하기에 충분한 패킷/심벌을 수신하거나 클라이언트/모바일 디바이스가 그것이 가입한 FEC 그룹들에 의해 제공된 것보다 적은 수의 패리티 심벌을 필요로 한다면, 클라이언트/모바일 디바이스는 그것이 하나 이상의 FEC 멀티캐스트 그룹을 떠나야(탈퇴해야) 할지를 확인할 필요가 있다. 수학식 1은 클라이언트/모바일 디바이스가 가입해야 할 FEC 그룹들의 개수를 제공하고, 현재 가입된 그룹 개수인 l 이 g 보다 크다면, 클라이언트/모바일 디바이스가 그룹($g+1$) 내지 그룹(l)을 탈퇴해야 한다. 가입하고 탈퇴하는 회수를 감소시키기 위해, 또 다른 파라미터가 도입된다.

수학식 4

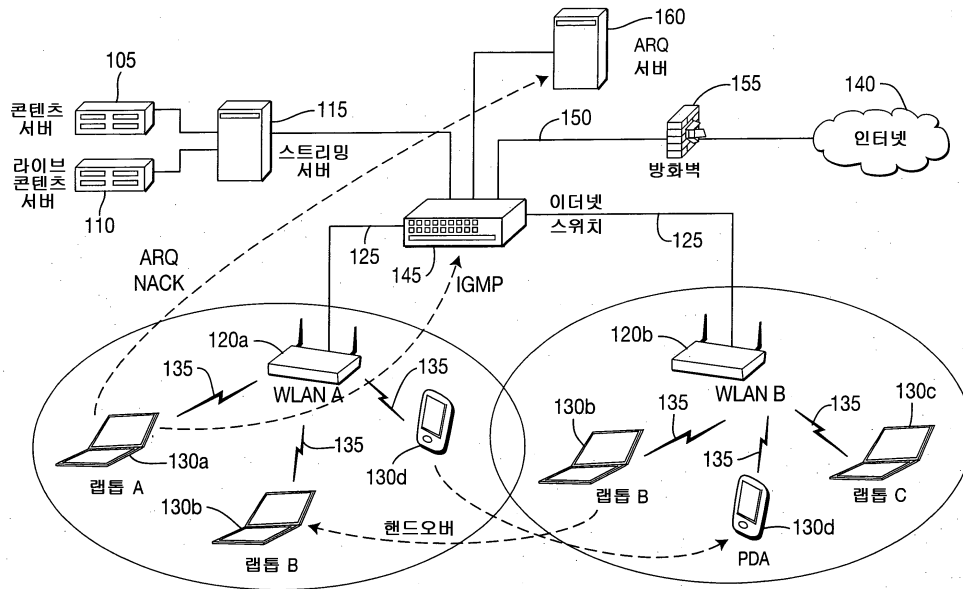
[0079] $h = \min(j : \sum_{i=1}^j over_head(i) \geq avg_loss_rate)$

[0080] 이라고 하고, $t = \max(g, h)$ 라고 할 때, $l > t$ 라면, 클라이언트/모바일 디바이스 그룹($t+1$) 내지 그룹(l)을 떠난다.

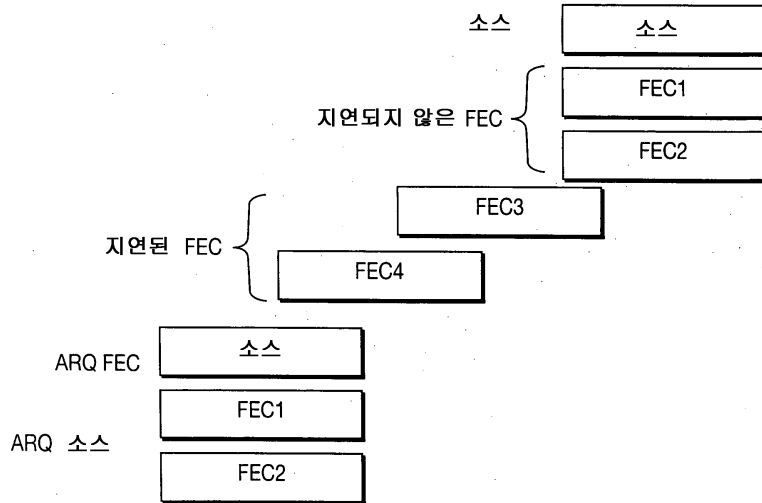
[0081] 도 6은 본 발명의 병합된 하이브리드 ARQ 방법의 서버측의 개략/블록도이다. 특히, 도 6은 본 발명의 스트리밍 서버(105)와 ARQ 서버(160)를 도시한다. 위에서 표시된 바와 같이, 스트리밍 서버와 ARQ 서버는 상이한 기계들

도면

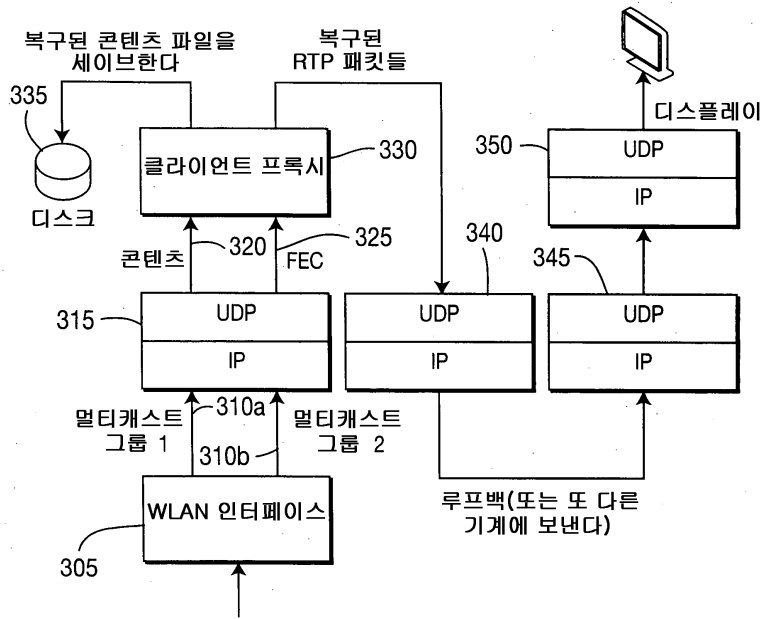
도면1



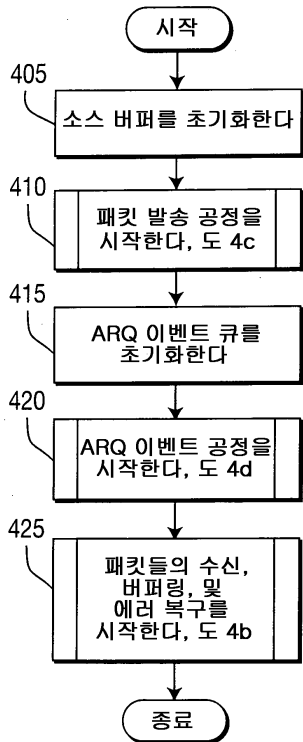
도면2



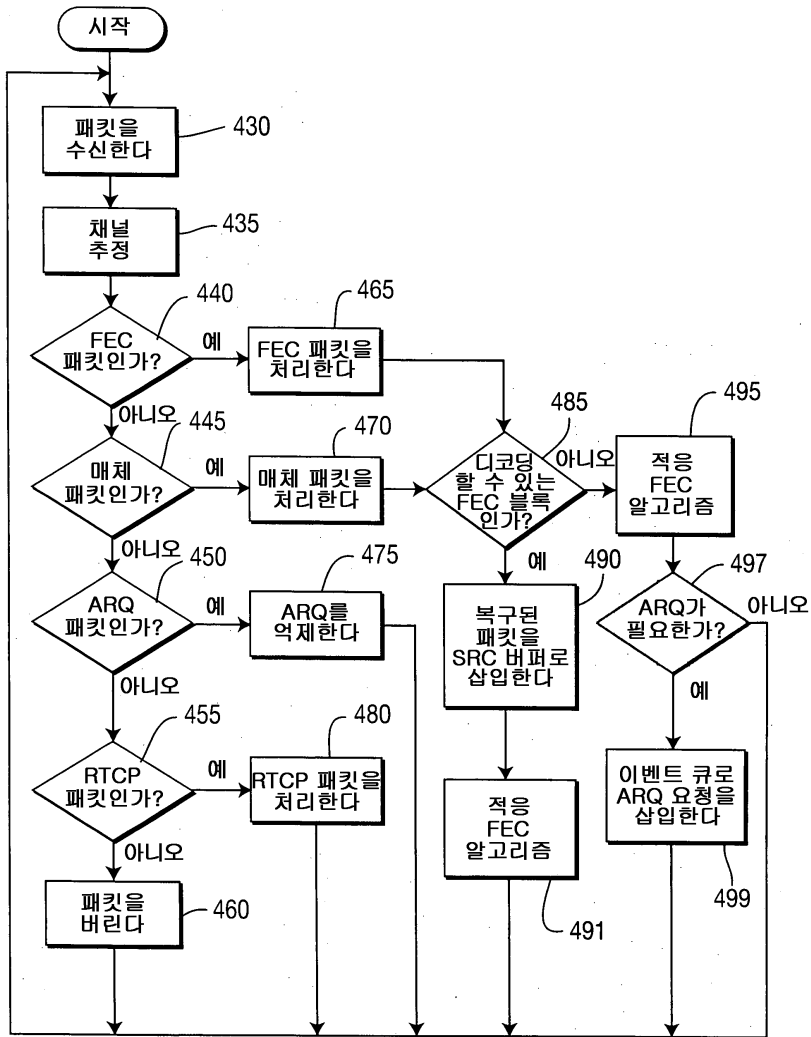
도면3



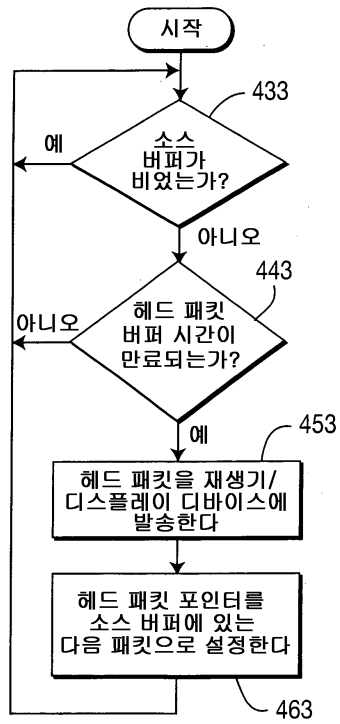
도면4a



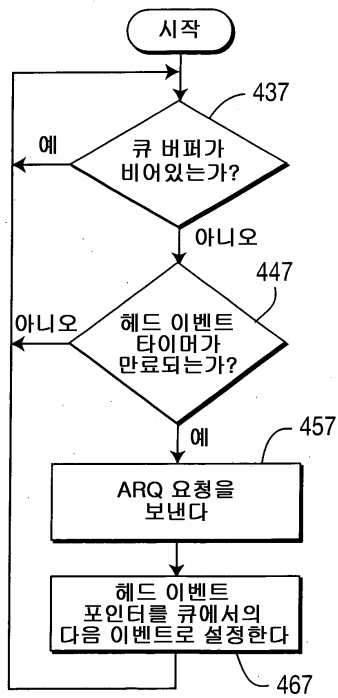
도면4b



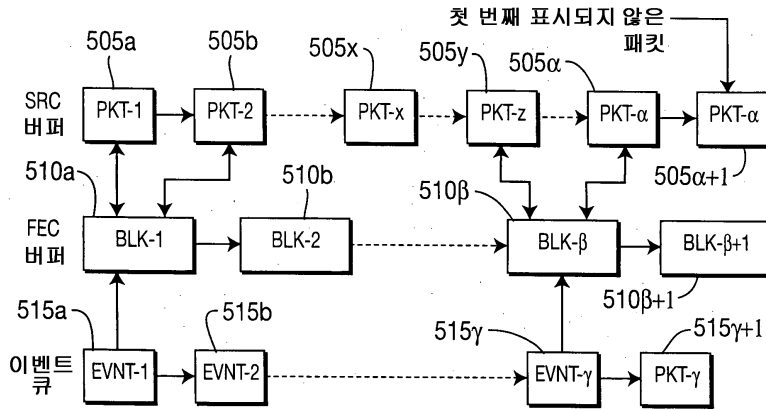
도면4c



도면4d



도면5



도면6

