

분산 포커스 제어 방식에 의한 대규모 컨퍼런스 서비스

A Large-Scale Conference Service by Distributed Focus Control Method

장춘서

금오공과대학교 컴퓨터공학과

Choonsoo Jang(csjang@kumoh.ac.kr)

요약

컨퍼런스 서비스에서 컨퍼런스 세션의 설정 및 유지 기능을 제공하는 포커스는 참가자 수가 증가 할수록 부하가 커지며 이는 대규모 컨퍼런스 서비스의 확장성을 제한하는 주요 요소가 된다. 본 논문에서는 이와 같은 문제점을 해결하기 위하여 포커스의 부하를 여러 컨퍼런스 노드로 분산 시켜 대규모 컨퍼런스 서비스의 구현을 용이하게 하는 새로운 분산 포커스 제어 방식을 제안하였다. 여기서 컨퍼런스 노드들 중 포커스 기능을 가진 노드는 자신의 정보를 컨퍼런스 서버의 컨퍼런스 정보 데이터베이스에 등록하며 시스템 전체 컨퍼런스 참가자 수의 증감에 따라서 필요한 포커스가 능동적으로 할당되어 참가자의 처리를 담당한다. 이를 위하여 본 논문에서는 분산 포커스 제어를 위한 컨퍼런스 제어 이벤트 패키지가 제시되었고 아울러 포커스와 컨퍼런스 참가자들 사이의 SIP 메시지 교환 절차도 함께 제시되었다. 제안된 시스템의 성능은 시뮬레이션 실험을 통하여 분석하였다.

■ 중심어 : | 컨퍼런스 서비스 | 포커스 | 세션시작프로토콜 |

Abstract

The focus maintains and manages conference session in the conference service. Load of focus increases by the number of participants and it becomes the major reason that limits the extendability of the large-scale conference service. In this paper, a new distributed focus control method has been proposed to solve this problem. Load of focus has been distributed to several conference nodes to implement large-scale conference service in this architecture. Conference nodes which have focus function subscribe to conference server's conference information database, and focus has been allocated dynamically to process participants needs according to total numbers of the conference participants. For this purpose a new conference control event package for focus load control has been suggested. Furthermore the exchange procedure of SIP messages between focuses and participants also been suggested. The performance of the proposed system has been evaluated by simulation.

■ keyword : | Conference Service | Focus | SIP(Session Initiation Protocol) |

* 본 연구는 금오공과대학교 학술연구비에 의하여 연구된 논문입니다.

접수일자 : 2014년 03월 26일

심사완료일 : 2014년 06월 17일

수정일자 : 2014년 06월 17일

교신저자 : 장춘서, e-mail : csjang@kumoh.ac.kr

I. 서론

SIP 환경에서의 컨퍼런스 서비스[1-4]에서 참가자 수가 증가 할수록 컨퍼런스 세션의 설정 및 유지 기능을 제공하는 포커스의 부하는 커지게 되고 이는 전체 컨퍼런스 시스템의 성능을 저하 시키고 확장성을 가로막는 주요 원인이 된다. 기존 연구에서는 SIP 메시지 개수를 줄이거나 복수개의 믹서를 사용하는 방식이 제안되었으나[5][6] 본 논문에서는 대규모 컨퍼런스 서비스의 구현을 용이하게 하기 위하여 포커스의 부하를 컨퍼런스 서버대신 여러 컨퍼런스 노드로 분산 시켜 처리하는 새로운 분산 포커스 제어 방식을 제안하였다.

제안된 시스템에서는 컨퍼런스 노드들 중에서 포커스 기능을 가진 노드는 자신의 정보를 본 연구에서 설계된 컨퍼런스 이벤트 제어 패키지를 사용하여 컨퍼런스 정보 데이터베이스에 등록한다. 컨퍼런스 참가자가 SIP INVITE 메시지로 컨퍼런스 참가 요청을 하면 이 컨퍼런스 정보 데이터베이스를 조사하여 해당 참가자를 처리 할 수 있는 포커스를 찾는다. 다음 이 컨퍼런스 노드의 SIP URI를 담은 Redirection SIP 메시지가 해당 참가자에게 보내지고 이 참가자는 해당 포커스와 SIP 세션 연결을 맺어 컨퍼런스에 참가하게 된다.

컨퍼런스 정보 데이터베이스에 등록된 정보는 컨퍼런스 노드들 사이에서 본 논문에서 제안한 컨퍼런스 이벤트 제어 패키지에 의해 동기화 되며 시스템 전체 컨퍼런스 참가자 수의 증감에 따라서 참가자의 처리를 담당할 포커스가 능동적으로 할당된다. 이를 위하여 본 연구에서는 서버 및 포커스와 컨퍼런스 참가자들 사이의 SIP 메시지 교환 절차가 제시되었고 분산 포커스 제어에 필요한 요소들이 추가된 새로운 컨퍼런스 정보 데이터 포맷이 설계되었다.

본 논문의 구성은 다음과 같다. II장에서는 관련 연구로서 먼저 컨퍼런스 서비스 시스템에 대해 설명한다. 이어서 컨퍼런스 서버들과 컨퍼런스 참가자 간의 컨퍼런스 정보 데이터를 서로 교환하기위해 사용되는 기본적인 컨퍼런스 이벤트 패키지에 대해 설명한다. III장에서는 먼저 본 논문에서 제안하는 포커스의 부하를 여러 컨퍼런스 노드로 분산 시켜 대규모 컨퍼런스 서비스의

구현을 용이하게 하는 새로운 분산 포커스 제어 구조를 설계한다. 다음 이와 같은 분산 포커스 제어를 위한 이벤트 제어 패키지의 설계 및 포커스와 컨퍼런스 참가자들 사이의 SIP 메시지 교환 절차를 설명한다. IV장에서는 시뮬레이션을 통하여 본 논문에서 제안한 시스템에 대한 성능 분석을 한 후 V장에서 결론을 맺는다.

II. 관련연구

1. 컨퍼런스 서비스 시스템

[그림 1]에 컨퍼런스 서버와 참가자로 구성된 기본적인 컨퍼런스 서비스 시스템을 보였다. 여기서 컨퍼런스 포커스는 컨퍼런스 참가자들과 서버 사이의 컨퍼런스 세션에 대한 설정, 변경 및 종료를 관리하고 컨퍼런스에서 발생하는 각종 정보의 전달을 제어하며 SIP 프락시(proxy) 서버 기능을 포함한다[7].

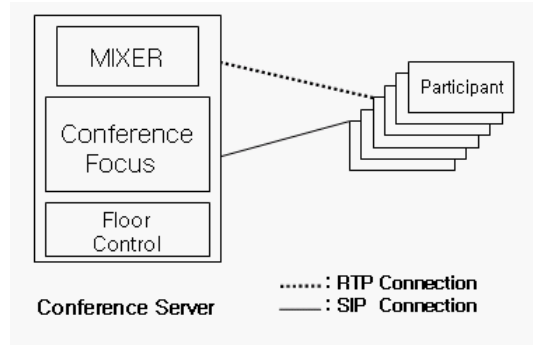


그림 1. 컨퍼런스 서비스 시스템

여기서 믹서는 RTP(Real-Time Transport Protocol) 프로토콜을 사용하여 컨퍼런스 서버와 참가자사이에서 미디어 패킷 스트림을 전송하는 기능을 한다. 플로어 컨트롤은 컨퍼런스 서비스에서 공유 자원에 대한 접근을 제어한다[8].

이와 같은 구조에서 전체 컨퍼런스 참가자 수가 증가 할수록 포커스가 처리해야 하는 SIP 세션 연결 및 관리에 들어가는 데이터 량의 증가와 믹서가 처리해야 하는 미디어 패킷 스트림의 증가로 인해 컨퍼런스 확장성에

제약을 주게 된다.

따라서 다수의 참가자 처리를 할 수 있도록 컨퍼런스 확장성을 향상시키기 위하여 복수개의 컨퍼런스 서버를 사용하여 컨퍼런스 참가자의 부하를 분산 처리하는 방식이 연구되고 있다[6][9]. 이들 연구에서는 SIP 메시지 개수를 줄이거나 시스템의 부하를 분산시키기 위해 컨퍼런스 포커스를 복수개의 서버에 위치시키는 방식을 사용하는 반면 본 연구에서는 확장성을 더욱 높이기 위하여 컨퍼런스 포커스를 복수개의 클라이언트 노드에 위치시키는 방식을 사용하였다.

본 연구에서 제안한 이와 같은 분산 포커스 제어 방식 구조에서는 시스템 전체 컨퍼런스 참가자 수의 증감에 따라서 포커스가 능동적으로 할당되어 참가자의 처리를 담당함으로써 컨퍼런스 포커스의 부하를 여러 노드로 쉽게 분산 시킬 수 있어 대규모 컨퍼런스 서비스의 구현을 용이하게 한다.

2. 컨퍼런스 이벤트 패키지

컨퍼런스 이벤트 패키지는 컨퍼런스 노드들 사이에서 application/conference-info+xml 형태의 컨퍼런스 정보 데이터[10]를 교환하기 위하여 사용된다. 컨퍼런스 정보 데이터의 최상위 요소(element)는 'conference-info'이며 이 요소의 속성으로 'entity', 'state'와 'version'을 가진다. 'entity'는 컨퍼런스 URI 값을 나타내며 'state'는 이전 컨퍼런스 정보에 비해 변화된 부분만을 담은 정보인지 여부를 나타내고 'version'은 통지 메시지의 버전을 각각 나타낸다.

하위 요소(child element)로는 컨퍼런스 전체에 대한 정보를 기술하는 'conference-description', 컨퍼런스를 주관하는 호스트에 대한 정보를 나타내는 'host-info', 현재 컨퍼런스 상태를 표시하기 위한 'conference-state' 및 컨퍼런스 참가자들에 대한 개별 정보를 나타내기 위한 'users' 등이 있다. 이들은 모두 세부적인 기능에 따라 각각 하위 요소들을 가진다.

'conference-info'의 하위 요소인 'users'는 다시 하위 요소로 'user'를 가지며 이는 각 참가자들의 정보를 나타낸다. 이를 위해 속성으로 'entity'를 가지며 이는 컨퍼런스 사용자에게 대한 URI를 나타낸다. 'user'의 하위

요소로는 'endpoint'가 있고 이는 각 참가자들이 컨퍼런스에 참여하면서 사용하는 디바이스 및 SIP 신호 세션에 대한 정보를 나타낸다. 'endpoint'는 다시 여러 하위 요소를 가지는데 주요한 하위 요소로는 디바이스의 연결 상태를 나타내는 'status', 컨퍼런스에 참가한 방식을 나타내는 'joining-method', 컨퍼런스 서버와의 미디어 스트림 정보를 나타내는 'media' 등이 있다.

본 논문에서는 이와 같은 기본적인 컨퍼런스 정보 데이터 포맷을 확장하여 대규모 컨퍼런스 서비스에 적용시킬 수 있도록 분산 포커스 부하 제어에 필요한 새로운 요소들이 추가된 컨퍼런스 제어 이벤트 패키지를 제안하였다.

III. 시스템 설계 및 구현

1. 새로운 분산 포커스 제어 구조 설계

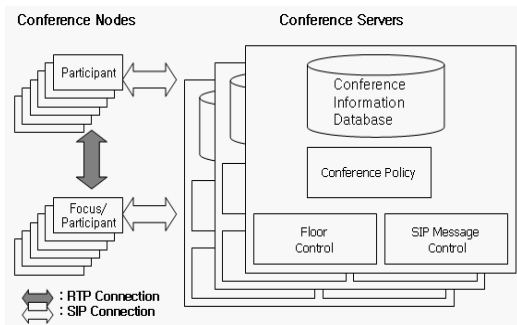


그림 2. 새로운 분산 포커스 제어 구조

[그림 2]에 본 논문에서 설계한 새로운 분산 포커스 제어 구조를 보였다. 여기에서 컨퍼런스 서버는 본 논문에서 확장 설계된 컨퍼런스 정보 데이터 포맷을 처리하는 데이터베이스를 포함하며, 컨퍼런스 정책 제어 모듈, SIP 메시지 제어 모듈 및 컨퍼런스 서비스에서 공유 자원에 대한 접근을 제어하는 플로어 제어 모듈을 구성요소로 가진다. 이 구조에서 컨퍼런스 노드는 포커스 기능을 가진 노드 및 단순 참가자로 구성되며 포커스 기능을 가진 노드는 자신의 컨퍼런스 URI를 SIP SUBSCRIBE 메시지를 사용해 컨퍼런스 정보 데이터

베이스에 등록한다. 이때 SUBSCRIBE 메시지에 포함되는 Contact 헤더의 속성으로 'isfocus', 'focus-capacity' 및 'mixer-capacity'를 사용하여 자신이 포커스 기능이 있음을 알리고 이와 함께 포커스 처리 용량 및 믹서 처리 용량도 각각 알려준다.

컨퍼런스 서버는 참가자 숫자가 증가 할수록 가장 많은 시스템 부하가 걸리는 포커스 및 믹서 기능을 다수의 다른 컨퍼런스 노드로 분산 처리함으로써 부하가 크게 경감되며 따라서 소수의 서버만으로 대규모 컨퍼런스 서비스 구성이 가능해 컨퍼런스 확장성을 크게 개선할 수 있다.

컨퍼런스 참가자가 컨퍼런스 URI를 담은 SIP INVITE 메시지를 사용하여 컨퍼런스 참가 요청을 하면 컨퍼런스 서버는 본 논문에서 제안한 확장된 컨퍼런스 정보 데이터 포맷을 사용해 구현된 데이터베이스를 액세스하여 해당 참가자를 처리 할 수 있는 포커스를 찾은 후 이에 대한 응답으로 상태 코드 번호가 302이고 이 포커스를 가리키는 Redirection SIP 메시지를 보내어 참가자가 이 포커스와 연결을 맺도록 한다.

참가자의 증가로 해당 포커스의 현재 부하 상태를 나타내는 'focus-message-load-index'와 'focus-media-load-index' 값이 주어진 범위를 넘을 경우 컨퍼런스 정보 데이터베이스에서 부하가 낮은 다른 포커스를 찾아 이의 SIP URI 주소를 알려준다. 이와 같은 과정을 거쳐 포커스와 연결된 참가자는 양단간의 SIP 세션 다이얼로그 교환 과정을 거쳐 RTP 세션을 맺어 컨퍼런스에 참가하게 된다.

서버는 컨퍼런스 제어 이벤트 패키지를 사용해 컨퍼런스 정보를 각 포커스 사이에 교환한다. 서버의 부하 상태는 'svr-load-index'를 통해 나타내며 주어진 기준 값을 넘어 새로운 컨퍼런스 서버를 추가해야 할 경우에는 컨퍼런스 정보 데이터베이스에 등록된 후보 중 새로운 컨퍼런스 서버를 선택하여 이 서버에게 INVITE 메시지를 보낸다. 해당 컨퍼런스 서버가 응답하면 컨퍼런스 제어 이벤트 패키지를 사용하여 현 컨퍼런스 정보 데이터를 보내어 데이터베이스를 생성하도록 하고 이후 컨퍼런스 정보 데이터의 교환도 제어 이벤트를 사용하여 실행된다. 이와 같은 컨퍼런스 동작 과정

에 대한 순서도를 [그림 3]에 보였다.

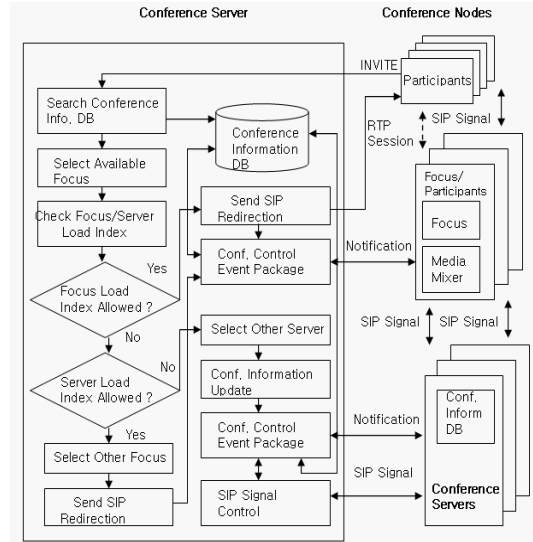


그림 3. 새로운 분산 포커스 제어 구조 동작 과정 순서도

[그림 3]에서 컨퍼런스 참가자가 SIP INVITE 메시지로 컨퍼런스 참가 요청을 하면 컨퍼런스 정보 데이터베이스를 조사하여 포커스 기능을 가진 컨퍼런스 노드를 찾는다. 이 컨퍼런스 노드의 부하 인덱스 값이 허용된 값이면 이 노드의 SIP URI를 담은 Redirection SIP 메시지를 해당 참가자에게 보내고 이어서 이 참가자는 포커스와 SIP 세션 연결을 맺어 컨퍼런스에 참가하게 된다. 이 처리 과정에서 변화된 컨퍼런스 정보는 컨퍼런스 제어 이벤트 패키지를 사용하여 통지된다.

컨퍼런스 노드의 부하 인덱스 값이 허용된 값을 초과하면 컨퍼런스 서버의 부하 인덱스 값을 조사하여 이 값이 허용된 값이면 현재 가장 부하 상태가 낮은 포커스를 선택하여 해당 참가자에게 알려준다. 컨퍼런스 서버의 부하 인덱스 값이 기준치 이상이면 컨퍼런스 정보 데이터베이스의 서버 후보군을 나타내는 'uri-available-servers' 항목에서 다른 서버를 선택하여 이 서버에게 현 컨퍼런스 정보 데이터베이스 내용을 보내고 컨퍼런스 제어 이벤트를 사용하여 컨퍼런스 노드들에게 이를 알린다.

2. 분산 포커스 제어를 위한 컨퍼런스 이벤트 패키지 설계

컨퍼런스 이벤트 패키지는 컨퍼런스 노드들 사이에서 application/conference-info+xml 형태의 컨퍼런스 정보 데이터를 전달하기 위하여 사용되며 설계된 컨퍼런스 이벤트 패키지에서는 분산 포커스 제어를 위하여 다음과 같은 필요 요소들이 추가되었다. 먼저 포커스의 부하 제어를 위하여 컨퍼런스 정보 데이터의 최상위 요소인 'conference-info'의 하위 요소로 'focus-message-load-index'와 'focus-media-load-index'가 새롭게 추가되었다.

여기서 요소 'focus-message-load-index'는 각 포커스에서 SIP 메시지 처리를 위한 부하 지수이며 여기에는 컨퍼런스 세션 설정과 유지 과정에서 발생하는 SIP 메시지와 부하 제어를 위해 발생하는 SIP 메시지 개수를 더하여 구해진다. 이 요소의 속성으로는 최대 부하 허용값을 나타내는 'max-message-load-index'와 각 포커스를 구분하기 위한 'focus-id'가 있다. 요소 'focus-media-load-index'는 RTP 세션에서 사용하는 오디오 샘플 및 비디오 프레임의 페이로드 타입에서 결정되는 초당 전송량의 전체 합을 기준으로 계산되며 미디어 스트림 처리를 위한 부하 지수이다. 이 요소의 속성으로는 최대 부하 허용값을 나타내는 'max-media-load-index'와 각 포커스를 구분하기 위한 'focus-id'가 있다. 이들 요소들의 값은 컨퍼런스 이벤트 패키지를 통하여 컨퍼런스 노드들 사이에서 전달된다.

다음으로는 등록된 포커스의 컨퍼런스 URI를 나타내는 'focus-conf-uri'와 각 포커스가 현재 담당하고 있는 참가자 목록인 'current-participants-list' 및 해당 포커스가 현재 처리하고 있는 참가자수를 나타내는 'current-participants-count'가 새롭게 추가되었다. 요소 'current-participants-list'의 속성으로는 각 포커스를 구분하기 위한 'focus-list-id'가 있고, 요소 'current-participants-count'의 속성으로는 각 포커스를 구분하기 위한 'focus-count-id'가 있다.

요소 'conference-state'는 현재 컨퍼런스의 상태를 표시하기 위하여 추가 되었고 요소 'total-users-count'와 'users-info'가 현재 전체 컨퍼런스 참가자의 수와

컨퍼런스 참가자들에 대한 개별 정보를 나타내기 위하여 이의 하위 요소로써 각각 추가되었다. 컨퍼런스 정보 데이터베이스를 관리하고 제어하는 컨퍼런스 서버의 부하를 나타내기 위한 요소로 'server-load-index'가 추가되었다. 이 요소는 서버에서 처리해야하는 초당 SIP 메시지 개수의 합과 헤더를 제외한 메시지 부분의 크기를 기준으로 계산된다. 아울러 컨퍼런스 서버의 개수를 나타내는 'servers-count'와 시스템의 부하 상황에 따라 새롭게 추가 될 수 있는 컨퍼런스 서버들의 URI를 가지고 있는 'server-list-uri'도 설계되었다. 컨퍼런스 등록 및 통지에 사용되는 컨퍼런스 제어 이벤트 패키지는 이와 같은 필요 요소들이 새롭게 추가된 컨퍼런스 정보 데이터 포맷을 사용하도록 구성된다. [그림 4]에 이를 사용한 SIP 메시지 및 컨퍼런스 정보 교환 절차를 보였다.

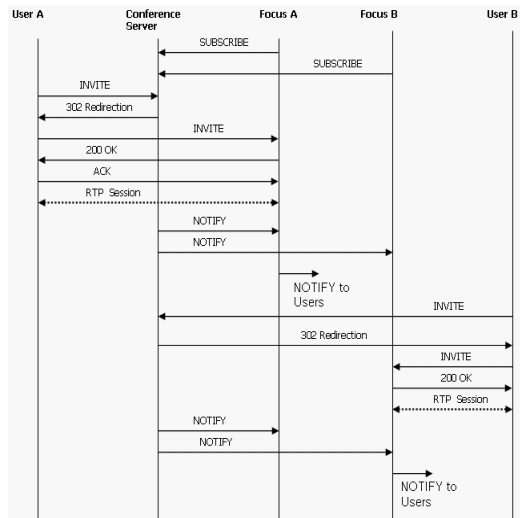


그림 4. SIP 메시지 및 컨퍼런스 정보 교환 절차

여기에서 먼저 컨퍼런스 노드 중 포커스 능력을 가진 노드는 SIP SUBSCRIBE 메시지를 사용하여 본 연구에서 새롭게 설계된 컨퍼런스 제어 이벤트 패키지를 등록한다. 새로운 참가자(User A)가 SIP INVITE 메시지를 사용하여 컨퍼런스 서버에게 컨퍼런스 참가 요청을 하면 컨퍼런스 서버는 확장된 컨퍼런스 정보 포맷으로 구현된 데이터베이스에서 해당 참가자를 처리 할 수 있는

포커스를 찾는다. 이 포커스의 부하 상태가 기준값 이내이어서 새로운 참가자를 처리 할 수 있는 상태이면 이 새로운 참가자에게 해당 포커스(Focus A)의 주소를 넣은 SIP 302 Redirection 메시지를 보낸다. 이 참가자는 이 포커스와 SIP 세션 및 RTP 세션을 생성하여 컨퍼런스에 참가한다. 이후 추가된 참가자(User B)에 대해 현 포커스의 부하가 이를 처리 할 수 있는 상태를 넘어서면 데이터베이스에서 검색한 다른 포커스(Focus B)를 가리키는 SIP Redirection 메시지를 보내어 이 포커스가 처리하도록 한다. 이와 같이 참가자 수가 증가할수록 부하 상태에 따라 다수의 포커스가 이를 분산 처리하게 되며 이와 함께 변화된 모든 컨퍼런스 정보는 컨퍼런스 제어 이벤트 패키지를 사용한 SIP 통지 메시지로 전달되어 항상 최신 상태를 유지할 수 있게 된다.

IV. 성능 분석

본 논문에서 제안한 분산 포커스 제어 방식을 사용한 대규모 컨퍼런스 서비스의 성능을 시뮬레이션으로 분석하였다. 컨퍼런스 노드들 사이에서 RTP 세션을 통한 미디어 스트림의 믹서 기능을 시뮬레이션하기 위해서 RTP 프로파일에 페이로드 타입 18로 정의된 CS-ACE LP G.729 코덱이 사용되었다. 시뮬레이션에서 사용한 샘플링 주파수는 8KHz이고, 프레임 길이는 10msec, 패킷 간격은 60msec, 프레임 크기는 80비트로 하였다. 그리고 미디어 패킷은 초당 16개가 생성되는 것으로 하였다. 실험은 먼저 기존의 중앙 집중형 컨퍼런스 구조와 분산 포커스 제어 구조와의 평균 스트림 지연시간을 비교 측정하였다. [그림 5]에 이때의 측정 결과를 보였다. 여기서 분산 포커스 제어 방식을 사용하는 경우 포커스의 최대 부하 허용값을 나타내는 'focus-message-load-index'와 'focus-media-load-index'가 각각 200이 되도록 컨퍼런스 정책을 설정하였다.

여기서는 처음에 하나의 포커스로 동작하다가 두 개의 부하 인덱스 값 중 하나라도 이 값을 넘으면 새로운 포커스가 추가되도록 하였다. 컨퍼런스 참가자 수가 20명일 때는 양쪽 모두 1개의 포커스 만을 사용하므로 동

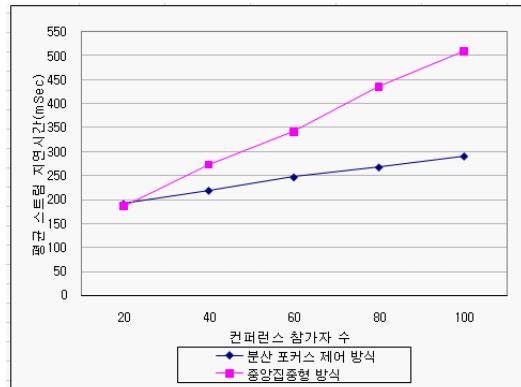


그림 5. 평균 스트림 지연시간 비교(최대 부하 허용값 =200)

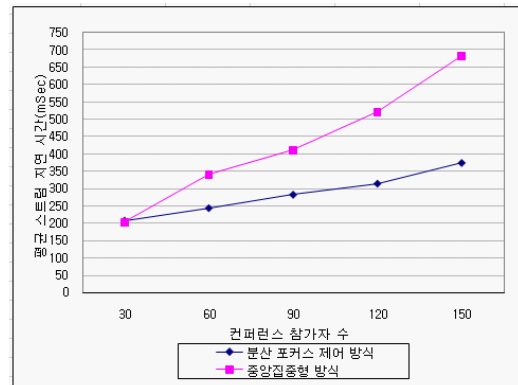


그림 6. 평균 스트림 지연시간 비교(최대 부하 허용값 =300)

일 한 성능을 보이나 40명 일 때부터는 분산 포커스 제어 구조의 경우 포커스가 추가되어 부하를 분담하게 되므로 성능의 향상을 보여준다. 컨퍼런스 참가자 수가 40명 일 때는 평균 지연 시간이 19.8% 감소하고, 60명 일 때는 27.8% 감소하고, 100명 일 때는 41.9%가 감소하여 참가자가 수가 증가할수록 본 논문에서 제안된 분산 포커스 제어 구조에서의 평균 스트림 지연시간의 개선 효과가 커짐을 알 수 있다.

[그림 6]은 부하 인덱스 최대값을 300으로 하여 측정 한 결과이다. 여기서는 컨퍼런스 참가자 수가 30명 일 때까지는 1개의 포커스만으로 동작하고 60명 일 때부터 포커스가 추가되며 따라서 컨퍼런스 참가자 수 60명에서 28.2%, 90명에서 31.1%, 150명에서 44.1% 성능 향

상을 각각 보여준다.

다음으로 컨퍼런스 참가자 수를 변화 시키며 평균 SIP 메시지 지연 시간을 비교한 결과를 [그림 7]에 보였다. 여기서는 컨퍼런스 참가자 수가 60명일 때부터 포커스가 추가되어 처리를 분담하여 60명일 때 14.2% 감소하고, 90일 때 24.5% 감소하고, 150명일 때 36.5% 감소함을 보여주고 있다.

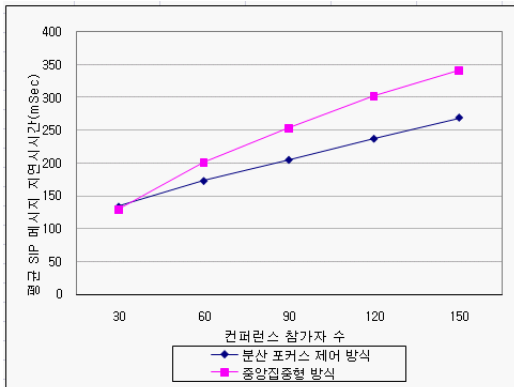


그림 7. 평균 SIP 메시지 지연 시간 비교

V. 결론

본 논문에서는 대규모 컨퍼런스 서비스를 위하여 컨퍼런스 시스템 사용자가 증가함에 따라 포커스의 부하를 능동적으로 여러 컨퍼런스 노드로 분산 시켜 처리하도록 하는 새로운 분산 포커스 제어 구조를 연구하였다. 이 구조에서 컨퍼런스 노드들 중 포커스 기능을 가진 노드는 자신의 정보를 컨퍼런스 정보 데이터베이스에 등록하고 컨퍼런스 참가자가 SIP INVITE 메시지로 컨퍼런스 참가 요청을 하면 이 컨퍼런스 정보 데이터베이스를 조사하여 해당 참가자를 처리 할 수 있는 포커스를 찾는다. 이 포커스의 부하 상태가 기준값 이내에서 새로운 참가자를 처리 할 수 있는 상태이면 이 노드의 SIP URI를 담은 Redirection SIP 메시지가 해당 참가자에게 보내지고 이 참가자는 해당 포커스와 SIP 세션 연결을 맺어 컨퍼런스에 참가하게 된다.

이와 같이 제안된 구조에서는 전체 시스템의 부하에

따라서 포커스가 능동적으로 할당되어 참가자의 처리를 담당하며 이를 위하여 분산 포커스 제어에 필요한 요소들이 추가된 새로운 컨퍼런스 제어 이벤트 패키지가 제시되었다. 아울러 포커스와 컨퍼런스 참가자들 사이의 SIP 메시지 교환 절차도 함께 제시되었다.

제안된 시스템의 성능 분석을 위하여 시뮬레이션 환경에서 컨퍼런스 참가자 수를 증가시켜 가면서 평균 스트림 지연 시간 및 평균 SIP 메시지 지연 시간을 측정하였다. 측정 결과 본 연구에서 제안한 포커스 제어 구조에서 평균 스트림 지연 시간의 경우 참가자 수에 따라서 19.8%에서 44.1% 까지 개선 효과가 있음을 보여주고 있다. 또 평균 SIP 메시지 지연 시간의 경우 참가자 수에 따라서 14.2%에서 36.5% 까지 개선 효과가 있음을 보여주고 있다. 향후 과제로는 대역폭이 제한된 무선 모바일 환경에서의 컨퍼런스 서비스에서 본 논문이 제안한 구조의 개선 효과를 측정하고 이를 바탕으로 대규모 컨퍼런스 서비스를 위한 보다 개선된 구조를 연구 할 계획이다.

참고 문헌

- [1] J. Sim and Y. Kim, "Conferencing Service Interworking in Peer-to-Peer and IMS Networks," Transactions on Internet and Information Systems, Vol.6, No.9, 2012(9).
- [2] R. Shekh-Yusef and M. Barnes, "Indication of Conference Focus Support for the Centralized Conferencing Manipulation Protocol," RFC 7082, 2013(12).
- [3] Deguo Yang, "An Application layer multicast based SIP conference model," Proc. of 10th Inter. Conf. on Computer Supported Cooperative Work in Design, pp.1-5, 2006(5).
- [4] P. Koskelainen, "A SIP-based Conference Control Framework," Proc. of Int. on Network and OS, pp.53-61, 2002(5).
- [5] 조현규, 이기수, 장춘서, "SIP 기반의 효율적인 혼

- 성형 멀티미디어 컨퍼런스 구조”, 한국콘텐츠학회논문지, 제7권, 제3호, pp.17-24, 2007(3).
- [6] 장춘서, 이기수, “리소스 리스트를 사용한 새로운 분산형 컨퍼런스 구조”, 한국컴퓨터정보학회논문지, 제16권, 제9호, pp.123-129, 2011(9).
- [7] M. Barnes, C. Boulto, and O. Levin, “A Framework for Centralized Conferencing,” RFC 5239, 2008(6).
- [8] G. Camarillo, “Connection Establishment in the Binary Floor Control Protocol,” RFC 5018, 2007(9).
- [9] Y. H. Cho and M. S. Jeong, “Policy-based distributed management architecture for large-scale enterprise conferencing service using SIP,” IEEE Journal on Communications, pp.43-53, 2009(2).
- [10] H. Schulzrinne, H. Tschfenig, and M. Thomson, “Dynamic Extensions to the Presence Information Data Formate,” RFC 5962, 2010(9).

저 자 소 개

장 춘 서(Choonseo Jang)

정회원



- 1993년 : 한국과학기술원(공학 박사)
- 1981년 ~ 현재 : 금오공과대학교 컴퓨터공학과 교수

<관심분야> : SIP, 인터넷텔레포니, 임베디드시스템