

중앙 집중형 네트워크 제어 플랫폼에서 SNMP 연결 관리의 고속화 방안 및 성능 분석

고 영 석⁺ · 권 태 현^{**} · 김 춘 희^{***} · 남 현 순^{**} · 정 유 현^{****} · 차 영 옥^{*****}

요 약

차세대 네트워크의 성공적인 추진을 위하여 트래픽 엔지니어링이 보장되는 중앙 집중형 제어 및 관리 기술이 네트워크 제어 플랫폼인 NCP(Network Control Platform)와 서비스 품질을 보장하는 스위치인 QSS(Quality of Service Switch)로 실현되고 있다. 본 논문에서는 NCP와 QSS 사이의 SNMP 인터페이스에서 고속의 연결 관리를 위하여 병렬형 기법과 쓰레드 및 객체 풀을 도입하였다. 연결 관리의 테스트-베드를 구축하여 본 논문에서 도입한 고속화 방안을 실험실 환경에서 확인하였으며, 연결 설정의 지연과 완료율을 측정하여 성능을 비교 및 분석하였다. NCP와 QSS 사이의 SNMP 인터페이스에서 연결 관리의 고속화를 위하여 병렬형 방식과 객체 풀의 사용이 중요한 성능 파라미터임을 확인하였다.

키워드 : 중앙 집중형, 병렬형 연결 관리, 네트워크 제어 플랫폼 (NCP), SNMP, 서비스 품질 스위치 (QSS)

The Performance Analysis of A High-speed Mechanism for SNMP Connection Management in Centralized Network Control Platform

YoungSuk Ko⁺ · TaeHyun Kwon^{**} · ChoonHee Kim^{***} · HyunSoon Nam^{**}
YouHyeon Jeong^{****} · YoungWook Cha^{*****}

ABSTRACT

Network Control Platform (NCP) and Quality of Service Switch (QSS) are being developed to realize centralized control and management technology, which is essential for guaranteeing traffic engineering and service quality in a next generation network. This paper adopts a parallel mechanism, and a thread and object pool to achieve high-speed connection management in the existing SNMP interface between NCP and QSS. We built up a connection management test-bed in laboratory environment to validate the functionality of high-speed connection management. We also measured and analyzed a performance of connection setup delay and a completion ratio using the test-bed. We ascertain that the parallel mechanism and the object pool are the most important performance parameters to achieve high-speed connection management in the SNMP interface between NCP and QSS.

Key Words : Centralization, Parallel Connection Management, NCP(Network Control Platform), SNMP, QSS(Quality of Service Switch)

1. 서 론

최근의 정보 통신 환경과 통신 시장은 차세대 네트워크의 실현이라는 목표를 향하여 기술의 통합과 융합이 이루어지고 있다. 해외에서는 NGN(Next Generation Network)[1]이라는 일반적인 용어로, 국내에서는 BcN(Broadband convergence Network)[2]이라는 용어로 통합과 융합에 대한 최적의 해법

과 방안을 찾기 위한 연구 활동이 활발히 진행 중에 있다.

국내에서 추진 중인 BcN의 성공적인 추진을 위하여 트래픽 엔지니어링이 보장되는 중앙 집중형 제어 및 관리 기술이 네트워크 제어 플랫폼인 NCP(Network Control Platform)와 서비스 품질을 보장하는 스위치인 QSS(Quality of Service Switch)로 실현되고 있다[3]. 제어 및 관리 평면과 전달 평면의 기능이 분리된 네트워크에서 QSS 스위치들을 중앙 집중화된 방식으로 제어 및 관리하는 NCP는 토폴로지 관리, 경로 계산 및 자원 예약, 연결 관리와 같은 기능을 수행한다. QSS 스위치는 인접 노드에 대한 정보 수집 및 서비스 품질과 트래픽 엔지니어링을 제공하기 위한 흐름기반의 스위칭을 제공하는 전달 및 관리 평면의 기능을 수행한다[3,4].

* 본 논문은 한국과학재단 우수연구센터(OIRC) 사업과 ETRI 정보통신 연구 개발사업 위탁과제의 연구결과임

+ 준 회 원: 안동대학교 컴퓨터공학과 석사과정

** 정 회 원: 한국전자통신연구원 광대역통합망연구단 NCP기술팀 선임연구원

*** 정 회 원: 대구사이버대학교 컴퓨터정보학과 조교수

**** 정 회 원: 한국전자통신연구원 광대역통합망연구단 NCP기술팀 팀장, 책임연구원

***** 정 회 원: 안동대학교 컴퓨터공학과 부교수(교신직자)

논문접수: 2007년 5월 15일, 심사완료: 2007년 8월 6일

서버가 ATM 스위치들에게 동시에 연결의 설정 및 해제를 요구하는 것이다. 하지만, 중앙 집중화된 연결 서버가 경로 계산을 위하여 스위치 망의 토폴로지만 관리할 뿐, 링크 상태와 스위치의 자원들은 각 스위치에 탑재된 스위치 자원 서버가 관리하므로 자원 예약 절차와 연결 구성 절차의 2 단계 절차가 요구된다. 즉, 연결 제어 서버는 단계 1에서 각 스위치에 탑재된 스위치 자원 서버에게 자원에 대한 가용 여부를 확인하며, 단계 2에서 스위치들에게 연결의 설정을 수행하게 하므로 전체적인 연결 설정의 지연이 길어지는 단점이 있다.

한편, 콜롬비아 대학에서는 ATM 스위칭 플랫폼을 위한 코바 기반의 개방형 프로그래머블 신호 시스템의 xbind를 위한 고성능 연결 제어 시스템을 제안하고 성능을 분석하였다[12]. 고성능의 연결 제어를 위하여 xbind 연결 매니저에 네트워크 상태 캐싱(caching), 연결 요청 메시지의 통합(aggregation) 그리고 연결 매니저와 스위치 서버 사이의 병렬형 연결 설정 방식을 적용하였다. 연결 식별자 캐싱을 이용하는 연결 매니저는 경로상의 스위치들에 대하여 가용한 출력 연결 식별자를 캐시에서 발견한다면 단일 단계(single phase)의 연결 설정을 수행하며, 캐시에서 발견하지 못하면 2 단계의 연결 설정을 수행하게 된다. 연결 식별자의 캐싱 크기와 메시지 통합 임계치(threshold)를 성능 제어 파라미터로 사용하였으며, 캐시 크기에 따른 연결 식별자의 히트율을 제시하고 있다. 또한, 병렬형 연결 제어와 더불어 부과되는 호의 도착율에 따라 메시지 통합 임계치를 적절히 조정함으로써 처리율과 지연의 개선 효과를 제시하였다.

AT&T나 콜롬비아 대학의 연구에서는 경로상의 특정 스위치에서 연결 설정이 실패하는 경우에 대한 연결의 재설정 메커니즘이 정의되어 있지 않다.

3. 중앙 집중형 네트워크 제어 플랫폼에서 SNMP 연결 관리의 고속화 방안

본 장에서는 중앙 집중형 제어 및 관리 기능을 수행하는 NCP와 흐름기반의 스위칭을 지원하는 QSS 스위치들 사이에 적용되는 SNMP 인터페이스에서 연결 관리의 고속화를 위하여 단일 단계 기반의 병렬형 연결 관리 방안을 제시한다.

3.1 사실 MIB 기반 SNMP 연결 관리

SNMP 기반의 연결 관리는 표준 MIB(Management Information Base)을 사용하는 방식과 사실 MIB을 사용하는 방식이 가능하다. IETF에서는 MPLS(Multiprotocol Label Switching)/GMPLS(Generalized MPLS)의 망 관리를 위하여 MPLS-FTN-STD-MIB, MPLS-LSR-STD-MIB, MPLS-TE-STD-MIB 그리고 GMPLS-LSR-STD-MIB, GMPLS-TE-STD-MIB 등의 표준 MIB을 정의하고 있다 [13,14]. 이들 표준 MIB들은 LER(Label Edge Router) 및 LSR(Label Switching Router)에 모두 적용될 수 있으나, MPLS-FTN-STD-MIB은 입구 LER에만 적용된다. 표준

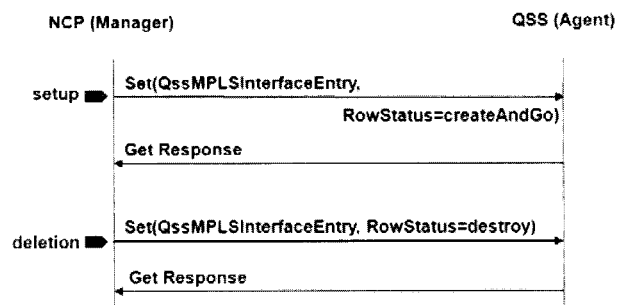
```

qssMPLSInterfaceTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF QssMPLSInterfaceEntry
    Max-ACCESS  read-create
    STATUS      current
    DESCRIPTION "MPLS Transit Node Configuration Info"
    ::= { qss120MplsMIB 3 }

qssMPLSInterfaceEntry OBJECT-TYPE
    SYNTAX      QssMPLSInterfaceEntry
    Max-ACCESS  read-create
    STATUS      current
    DESCRIPTION "Row Description"
    INDEX       ( qssIfName, qssInLabel )
    ::= { qssMPLSInterfaceTable 1 }

QssMPLSInterfaceEntry ::= SEQUENCE {
    qssIfName          DisplayString,
    qssInLabel         Integer32,
    qssNextHopIpAddr  DisplayString,
    qssLabelAction     Integer32,
    qssSwapOutLabel   Integer32,
    qssMPLSIFLSPName  DisplayString,
    qssMPLSISPTType   Integer32,
    qssMPLSIngressAddr DisplayString,
    qssMPLSEgressAddr DisplayString,
    qssMPLSLSPID      Integer32,
    qssMPLSInterfaceRowStatus RowStatus
}
    
```

(그림 3) QSS120-MPLS-MIB의 연결 관리 테이블 및 엔트리



(그림 4) QSS120-MPLS-MIB을 사용한 연결 설정 및 해제 절차

MPLS-LSR-STD-MIB[15]을 사용하는 연결 관리에서는 입력 및 출력 세그먼트 테이블의 엔트리와 크로스-커넥트 테이블의 엔트리를 각각 별도의 SNMP 요청 및 응답 메시지를 이용하여 설정하므로 연결 설정의 지연이 발생 할 수 있다[16].

이러한 연결 설정 지연의 최소화를 위하여 하나의 SNMP 요청 및 응답 메시지로 연결을 설정할 수 있도록 표준 MPLS-LSR-STD-MIB을 단순화한 사실 QSS120-MPLS-MIB을 정의하였다. (그림 3)은 사실 QSS120-MPLS-MIB에 정의된 연결 관리 테이블 및 엔트리에 포함된 관리 정보를 나타낸다. qssMPLSInterfaceTable의 인덱스는 qssIfName과 qssInLabel를 사용한다. qssIfName과 qssNextHopIpAddr은 MPLS-LSR-STD-MIB의 입력 및 출력 세그먼트 테이블의 입력 및 출력 세그먼트 인터페이스와 동일한 의미를 가진다.

(그림 4)는 QSS120-MPLS-MIB을 사용하여 망 관리 매니저가 탑재된 NCP와 QSS 스위치사이에서 연결을 설정하고 해제하는 절차를 나타낸다.

연결 설정을 위하여 표준 MPLS-LSR-STD-MIB을 사용할 경우, 망 관리 매니저는 크로스-커넥트, 입력 및 출력 세그먼트 테이블 엔트리의 설정을 에이전트에게 요구하여야

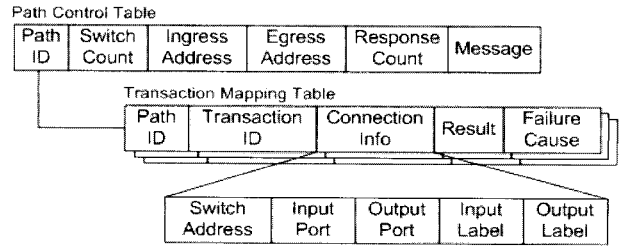
Request Path 메시지는 연결의 생성 요구 메시지를 수신한 PCC가 발신지 및 착신지 주소와 QoS 정보를 제공하여 PAC에게 경로 계산을 요청하는 메시지이다. Recalculate Path 메시지는 PCC가 PAC에게 경로의 재계산을 요청하는 메시지이다. 스위치의 내부 오류 또는 프로토콜의 오류로 인하여 연결 설정이 실패하거나 우회 경로로 연결을 재설정하는 경우 그리고 계산된 경로상의 스위치에 대한 자원 예약이 실패한 경우에 경로의 재계산 요청이 사용된다. Reserve Resource 메시지는 PCC가 경로상의 스위치들에 대한 주소 정보와 QoS 정보를 제공하여 CAC에게 연결 수락 제어 및 자원 예약을 요청하는 메시지이다. CAC는 각 스위치에 대한 연결 수락 여부와 경로상의 스위치들에 대한 입력 및 출력 포트와 레이블 정보를 PCC에게 반환한다. PCC는 성공적인 Reserve Resource 메시지를 CAC에게서 수신한 후 연결 설정을 요구하는 SNMP 메시지들을 QSS 스위치들에게 동시에 전달한다. 경로상의 모든 QSS 스위치로부터 연결 설정의 완료를 SNMP 메시지로 통보받으면 PCC는 CAC에게 Assign Resource 메시지를 전달하여 스위치들의 자원이 할당되었음을 알린다. Release Resource 메시지는 PCC가 CAC에게 스위치에 예약되었거나 할당되었던 자원의 해제를 요구하는 메시지이다.

3.4 병렬형 연결 관리를 위한 자료 구조

NCP의 PCC는 SNMP 매니저와 상호 동작하여 경로상에 있는 QSS 스위치들에게 연결 설정을 위한 SetRequest 메시지를 병렬로 보내게 된다. 병렬형 연결 관리를 위하여 NCP는 수신한 GetResponse 메시지가 어느 QSS 스위치로부터, 어떤 SetRequest 메시지에 대한 응답인지를 구분할 수 있어야 한다. 또한, NCP는 경로에 있는 모든 스위치들에 대한 연결 설정이 모두 완료되었음을 판단하기 위해 경로상의 스위치 개수와 스위치들로부터 수신한 응답의 개수가 일치하는지 확인할 수 있어야 한다. (그림 7)은 병렬형 연결 관리를 위하여 PCC에서 유지하는 경로 제어 테이블(PCT: Path Control Table)과 트랜잭션 매핑 테이블(TMT: Transaction Mapping Table)의 자료 구조를 나타낸다.

PCT 테이블의 각 엔트리에는 경로 식별자(PathID), 입구 및 출구 주소(Ingress/Egress Address), 경로상에 선택된 스위치의 개수(Switch Count), 연결 설정 요청에 응답한 스위치의 개수(Response Count) 그리고 메시지(Message) 필드로 구성된다. 메시지 필드에는 PCC가 수신한 연결의 생성 요구 메시지의 원본이 저장되며, 경로의 재계산이나 연결의 재설정 시에 사용된다. PCT 테이블의 엔트리 하나에 대하여 해당 경로상에 선택된 스위치의 개수만큼 TMT 테이블의 엔트리들이 생성되며 경로 식별자 필드는 동일한 값을 갖는다.

TMT의 트랜잭션 식별자(TransactionID)는 SNMP 매니저와 에이전트 사이의 메시지를 식별하기 위한 필드로 SNMP 메시지의 RequestID와 매핑된다. 연결 설정과 해제 시에 사용되는 TMT의 연결 정보(Connection Info) 필드는



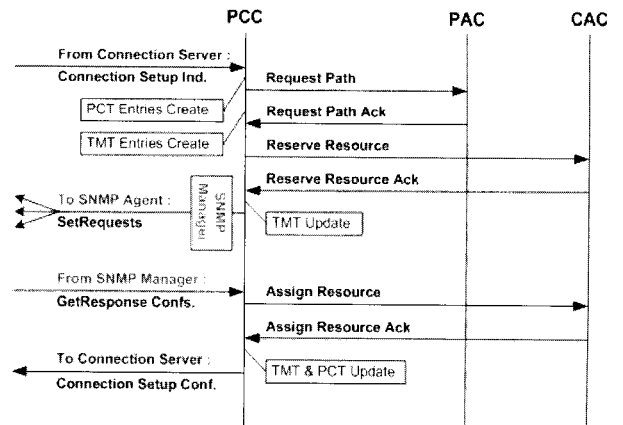
(그림 7) 병렬형 연결 관리를 위한 테이블

경로상의 특정 스위치에 대한 연결 정보를 나타내며, 스위치 주소, 입력 및 출력 포트와 레이블 정보를 갖는다. TMT의 결과(Result) 필드는 연결 설정 요청에 대한 결과를 나타내며, 실패 원인(Failure Cause) 필드는 스위치로부터 실패 응답을 수신한 경우에 대한 실패 원인을 나타낸다. PCC는 SNMP 매니저로부터 받은 응답 메시지가 어느 스위치, 어느 경로에 대한 응답인지를 구분하기 위하여 TMT의 TransactionID와 PathID 필드를 사용하며, PCT와 TMT의 PathID 필드를 매핑시킴으로써 내부 경로에 대한 연결 설정이 모두 완료되었는지 판단하게 된다.

3.5 병렬형 연결 설정 및 연결 설정 실패 시 재설정 절차

3.5.1 병렬형 연결 설정 절차

(그림 8)은 NCP가 QSS 스위치들에게 성공적으로 병렬형 연결 설정을 수행한 경우의 절차를 나타낸다. PCC가 연결 생성 요구(Connection Setup Ind.)를 수신하면, PCT 테이블의 엔트리를 생성한 후, Request Path 메시지를 PAC에게 전달한다. PAC는 전달 망에 대하여 계산된 최적의 경로를 Ack 메시지에 포함하여 PCC에게 응답한다. PCC는 수신한 Request Path Ack 메시지의 정보를 기반으로 TMT의 엔트리들을 생성한 후, CAC에게 Reserve Resource 메시지를 전달한다. CAC는 연결 수락과 자원 예약을 수행한 후, 응답으로 Ack 메시지를 PCC에게 전달한다. PCC는 SNMP 매니저와 상호 작동하여 TMT 엔트리들에 기록된 연결 정보를 기반으로



PCC : Parallel Connection Control, PAC : Path Computation, CAC : Connection Admission Control, PCT : Path Control Table, TMT : Transaction Mapping Table

(그림 8) 병렬형 연결 설정 절차

설정 지연이 병렬형 동기 요청별 쓰레드 방식(PCC-STR)보다 약 1.2배 감소되었으며, 순차형 동기 쓰레드 풀 방식(SCC-STP)보다는 약 2.8배 감소되었다. 하지만, 쓰레드 풀을 이용하는 병렬형 동기 방식(PCC-STP)은 병렬형 비동기 방식(PCC-ASY)보다는 약 1.27배가 증가되었다. 스위치의 수가 3대인 경우와 동일하게 SnmpTarget API를 사용하여 동기 방식으로 동작하는 연결 관리에서는 병렬형 쓰레드 풀 방식이 가장 우수하지만, 전체적으로는 하위 레벨 SNMP API를 사용하는 병렬형 비동기 방식이 가장 우수함을 알 수 있다.

순차형 쓰레드 풀 방식이 초당 연결의 요청이 300일 때 연결 설정 지연이 급격하게 증가한 이유는 쓰레드 간에 문맥 교환이 빈번히 일어나기 때문이다. 순차형 연결 관리를 담당하는 SEQCCM 모듈은 요청별 쓰레드 방식으로 동작하면서 스위치들에 대하여 순차적으로 연결 설정을 수행한다. 즉, SEQCCM 모듈에서 동작하는 요청별 쓰레드에서 연결 설정을 완료하는 시간보다 더 짧은 주기로 연결 요청이 생성된다. 이는 짧은 시간에 요청별 쓰레드의 수가 많아지므로 쓰레드 간의 문맥 교환이 빈번히 이루어지게 된다.

병렬형 동기 요청별 쓰레드 방식에서 연결 요청의 수가 200부터 연결 설정 지연이 증가하는 이유 또한 쓰레드 간의 문맥 교환에 의하여 발생하는 오버헤드가 원인인 것으로 분석된다. 병렬형 연결 관리 방식을 사용함으로써 순차형 연결 관리 방식보다는 연결을 설정하는 시간이 많이 감소하여 급격히 쓰레드의 수가 증가되지는 않았지만, 200 아래의 경우보다는 다소 많은 쓰레드가 생성되었기 때문이다.

연결 설정 지연을 비교 및 분석한 결과, 순차형 쓰레드 풀 방식은 스위치가 증가하고 초당 연결 요청의 수가 증가하면 할수록 연결 설정 지연도 급격히 증가하게 된다. 반면, 병렬형 방식들은 다수의 QSS 스위치에게 동시에 연결 설정을 수행함으로써 연결 설정 지연을 최소화 할 수 있다. 하지만, 동기 방식을 사용하는 연결 관리에서 병렬형 방식을 사용하더라도 쓰레드 풀을 도입하지 않은 병렬형 동기 요청별 쓰레드 방식은 스위치 및 초당 연결 요청의 수가 증가하면 할수록 짧은 시간에 요청별 쓰레드의 수가 많아지므로 쓰레드

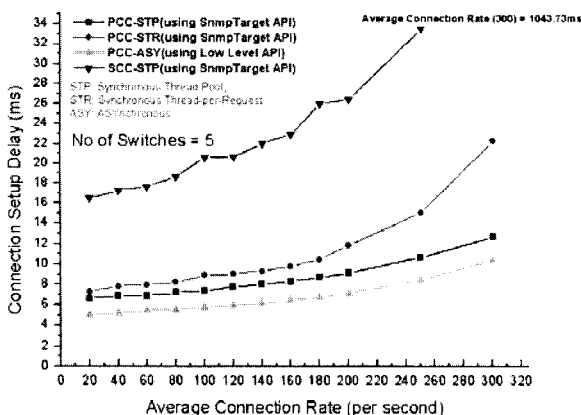
풀을 도입한 병렬형 쓰레드 방식보다는 연결 설정 지연이 증가하게 된다.

5.2.2 연결 설정 완료율

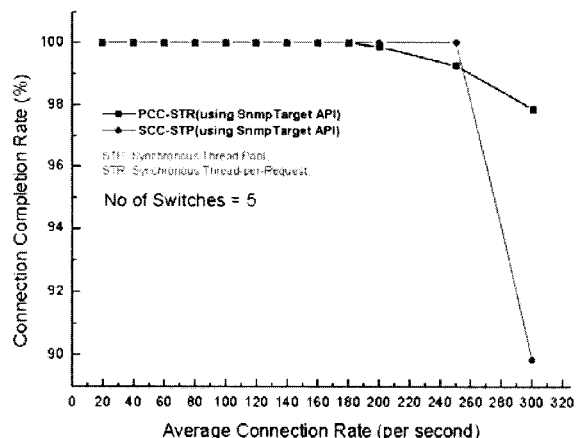
실험 환경에서 QSS 스위치의 수가 3대인 경우에 연결 설정의 완료율은 (그림 14)의 방식들에서 모두 동일하게 100%로 측정되었다. (그림 15)는 QSS 스위치의 수가 5대인 경우에 대하여 측정된 연결 설정의 완료율을 그래프로 나타낸 것이다. 완료율이 100%인 방식은 그래프에서 나타내지 않았다. 병렬형 동기 요청별 쓰레드 방식(PCC-STR)과 순차형 동기 쓰레드 풀 방식(SCC-STP)은 초당 연결의 평균 도착 개수에 따라 다소 감소하였다. 초당 연결의 도착 개수와 스위치의 수에 따른 연결 설정 지연과 더불어 쓰레드 및 객체 풀과 메시지-큐의 크기는 연결 설정의 완료율과 밀접한 관계를 가진다. 내부 통신을 위하여 메시지-큐를 사용하는 구현 구조에서 연결 요청의 평균 도착 시간보다 연결 설정 지연이 긴 경우에 메시지-큐의 크기는 완료율에 영향을 미치게 된다. 실험 환경에서 메시지-큐의 크기는 시스템의 기본 값(약 16KB)을 사용하였으며, 메시지-큐 크기의 85%를 가용범위의 임계치로 설정하였다. 병렬형 동기 요청별 쓰레드와 순차형 동기 쓰레드 풀 방식에서 메시지-큐 크기를 증가하면 완료율을 개선할 수 있음을 실험 과정에서 확인할 수 있다.

5.2.3 객체 풀의 성능 개선 효과

NCP의 SNMPMM 모듈은 AdventNet사의 Web NMS에서 제공하는 SNMP API인 SnmpTarget 객체를 사용한다. 매번 연결 설정 시에 생성되는 SnmpTarget 객체의 생성 오버헤드를 개선하기 위하여 객체 풀의 개념을 사용하였으며, 객체 풀의 사용과 미사용 시의 성능을 측정하여 비교하였다. <표 2>는 병렬형 쓰레드 풀 방식(PCC-STP)에서 SnmpTarget 객체 풀을 사용하는 경우와 사용하지 않은 경우의 연결 설정 지연과 완료율을 측정된 결과이다. 객체 풀을 사용하지 않은 경우의 연결 설정 지연은 20에서 300까지의 연결 요청에서 70.32와 380.95ms 사이의 값이 측정되었다. SnmpTarget



(그림 14) 스위치의 수가 5대인 경우의 연결 설정 지연



(그림 15) 완료율

