논문 22-47-06-04 The Journal of Korean Institute of Communications and Information Sciences '22-06 Vol.47 No.06
https://doi.org/10.7840/kics.2022.47.6.826

비트코인 네트워크에서 압축 블록 전달 방식의 전송 지연 분석

김 애 리◆, 주 홍 택°

Analysis of the Delivery Delay in Compact Block Relay in Bitcoin Networks

Aeri Kim◆, Hongtaek Ju°

요 약

비트코인 네트워크의 노드 간 블록 전달 방식은 주로 압축 블록 전달 방식(CBR: Compact Block Relay)을 사용하여 동작한다. 압축 블록 전달 방식은 블록에 포함된 트랜잭션의 아이디만 송신함으로써 네트워크 대역폭 소모와 블록 전달시간을 줄이는 블록 압축 기술이다. 하지만 기술 도입 후, 정확한 성능을 보여주는 실험 및 분석 결과가 부족하다. 또한, 여전히 블록 전파에는 지연이 존재한다. 따라서 본 논문은 직접 연결된 비트코인 노드 간에 고대역폭(High-bandwidth) 압축 블록 전달시간을 측정하여 압축 블록 전달 방식의 성능을 보여주고, 전달시간 지연 원인을 분석한다. 지연 원인 분석에 사용한 요인은 네트워크, 트랜잭션 요청, 체인 연결과 시스템 메모리양, 블록의 크기와 트랜잭션 보유 수이다. 본 논문의 결과는 다양한 관점에서 압축 블록 전달의 지연 원인을 분석하고 제시한다. 또한, 주요 지연 원인인 트랜잭션 요청에 대하여 심층적인 분석을 제공한다. 분석 결과를 통해 전달 지연 원인을 해결하고 지연시간을 단축하는 방법을 논의해봄으로써 압축 블록 전달 방식의 성능을 개선할 수 있다.

키워드 : 비트코인, 압축 블록, 블록 전파, 지연시간, 원인 분석
Key Words : Bitcoin, Compact Block Relay, Block Propagation, Delay Time, Cause Analysis

ABSTRACT

The block propagation protocol between nodes in the Bitcoin network mainly operates using the Compact Block Relay (CBR) protocol. The CBR is a block compression technology that reduces network bandwidth consumption and block delivery time by transmitting only the transaction ID included in the block. However, after the introduction of the CBR, there are no experimental and analysis results showing an accurate performance. Also, there is still a delay in block propagation. Therefore, this paper shows the performance of the CBR by measuring the high-bandwidth CBR's block delivery time between directly connected bitcoin nodes, and analyzes the cause of the delivery delay. Factors to be used for delay cause analysis are network, transaction request, chain connection and system memory amount, block size and number of transactions. The results of this paper provide analysis results of delay causes of CBR from various viewpoints. In addition, it provides an in-depth analysis of the transaction request, which is the main cause of delay. And the performance of the CBR can be improved by solving the cause of the delivery delay and discussing how to reduce the delay time through the analysis result.

※ 본 연구는 한국연구재단 기초연구사업(NRF-2018R1D1A1B07050380) 지원 및 계명대학교 네트워크 연구실 관리로 수행되었습니다.
◆ First Author : Keimyung University Department of Computer Engineering, AeriKim@stu.kmu.ac.kr, 정회원
° Corresponding Author : Keimyung University Department of Computer Engineering, juht@kmu.ac.kr, 종신회원
논문번호 : 202202-019-B-RN, Received February 7, 2022; Revised April 27, 2022; Accepted May 7, 2022

Ⅰ. 서 론

비트코인은 2008년 나카모토 사토시라는 익명의 개발자에 의해 개발된 블록체인 플랫폼이다[1]. 비트코인에서 트랜잭션(Transaction)은 전파되고, 검증된 후에 새로운 블록(Block)에 포함된다. 새롭게 생성된 블록은 비트코인 네트워크에 참여 중인 노드로 플러딩(Flooding) 방식을 통해 전파된다. 이처럼 블록은 노드를 거쳐서 전체 네트워크로 전파되기 때문에, 전체 네트워크로 블록이 전파되는 시간은 노드 간의 전달 시간에 영향을 받는다. 현재 비트코인 네트워크의 노드 간 블록 전달 방식은 기존의 레거시 방식(Legacy Protocol)과 Matt Corallo가 개발한 압축 블록 전달 방식(CBR : Compact Block Relay)으로 동작한다[2]. 압축 블록 전달 방식 도입을 통해 블록 전달시간과 대역폭의 감소 등 블록 전달의 효율성을 증대시킬 것으로 예상되었지만[3], 정확한 성능 실험 및 분석 결과가 부족하다. 또한, 여전히 블록 전파 지연은 존재한다.

비트코인은 낮은 TPS(Transactions per second) 및 긴 블록 생성 시간, 전파 지연 등 속도 문제로 꾸준히 지적받고 있다[4]. 이러한 문제로 인하여 전자 소매 거래에 사용되지 못하고 한계를 갖는다. 속도 문제가 해결되지 못한다면 비트코인의 지속적인 사용이 불가할 것이다. 따라서 속도 향상 연구는 계속해서 이루어져야 한다.

본 논문에서는 실제 비트코인 네트워크의 고대역폭 압축 블록 전달시간을 측정한다. 측정을 통해 해당 방식의 성능을 보여주고, 전달시간이 지연된 원인을 분석한다. 지연 원인 분석에 사용한 요인은 네트워크, 트랜잭션 요청, 체인 연결과 시스템 메모리양, 블록의 크기와 트랜잭션 보유 수이다. 분석 결과는 다양한 관점에서 압축 블록 전달의 지연 원인을 분석하고 정보를 제공한다는 점에 의의가 있다. 또한, 주요 원인인 트랜잭션 요청에 대하여 트랜잭션의 크기와 요금을 중점으로 심층적인 분석을 제공한다. 이는 블록 전달 시간을 단축하기 위한 연구에 기초 데이터로써 활용 가치가 있으며, 결과를 통하여 전달 지연 원인을 해결하고 지연시간을 단축하는 방법을 논의해봄으로써 압축 블록 전달 방식의 성능을 개선할 수 있다.

본 논문의 구성은 다음과 같다. 2장은 압축 블록 전파에 대한 배경지식과 관련 연구를 제시한다. 3장은 압축 블록 전달시간 실험의 구성 및 방법을 설명한다. 4장에서는 실험과 분석 결과를 정리하고 5장은 본 논문의 결론을 도출하며 향후 연구를 제시하는 것으로 마무리한다.

Ⅱ. 관련 연구

2.1 배경지식

서론에서도 밝힌 바와 같이 현재 블록 전달 방식으로는 기존에 존재하던 레거시 방식과 전달시간 단축을 위해 개발된 압축 블록 전달 방식이 있다. 그림 1의 (a) 레거시 방식은 블록을 전달할 때 블록 헤더(header)와 함께 블록에 포함된 전체 트랜잭션을 전달한다. 압축 블록 전달 방식은 블록에 포함된 트랜잭션의 아이디만 송신함으로써 네트워크 대역폭 소모와 블록 전달시간을 줄이는 블록 압축 기술이다. 최종적으로 블록에 포함된 트랜잭션 중에서 수신하는 노드(Node)의 메모리 풀(Memory-pool; 각 노드가 연결된 노드에게 전달받은 트랜잭션을 모아 놓은 저장소)에 이미 저장된 트랜잭션은 전달하지 않음으로써 블록 전달을 효율적으로 수행한다.

기존 문서에 따르면 60% 이상의 블록 전달에서, 코인베이스(Coinbase; 블록 생성에 대한 보상금을 지급하는 트랜잭션)를 제외하고 전달될 블록에 포함된 트랜잭션들이 수신하는 노드의 메모리 풀에 이미 모두 저장되어 있다. 전달받은 블록에 트랜잭션을 채울 필요 없이 즉시 전달받을 수 있음을 의미한다[2]. 이 방식은 저대역폭 릴레이(Low Bandwidth Relaying)방식과 고대역폭 릴레이(High Bandwidth Relaying)방식이 있다. 특히, 고대역폭 방식에서는 그림 1의 (c)에서 보여주는 바와 같이 주고받는 메시지의 절차가 다른 전달 방식에 비해 간단하다. 또한, 이전 노드에서 아직 블록의 검증이 끝나지 않더라도 다음 노드로 바로 전달하여 블록 전달시간이 짧다[5].

압축 블록 전달 방식의 전달과정을 자세히 서술하면 다음과 같다. 그림 1 (b), (c)의 전달하는 노드(Sender)가 압축 블록(cmpctblock) 메시지를 통해 블록의 헤더와 블록이 포함하고 있는 트랜잭션 목록(shortID)을 보낸다. 전달받는 노드(Receiver)는 자신의 메모리 풀과 전달받은 트랜잭션 목록을 비교하여

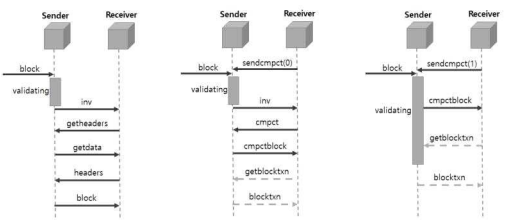

그림 1. 프로토콜 별 블록 전달과정에서 주고받는 메시지
Fig. 1. Bitcoin Protocol (a) Legacy Protocol (b) Low Bandwidth (c) High Bandwidth

존재하지 않는 트랜잭션 데이터를 트랜잭션 요청(getblocktxn) 메시지로 요청한다. 트랜잭션 요청 메시지를 받은 전달하는 노드는 요청한 트랜잭션을 블록 트랜잭션(blocktxn) 메시지로 전달해준다. 전달받는 노드는 필요한 트랜잭션을 전달받으면 메모리 풀에 저장된 트랜잭션을 합쳐서 블록을 완성한다.

2.2 관련 연구

R. Nagayama[6] 등은 압축 블록 전달 방식의 성능 개선 측정 결과를 시뮬레이션으로 보여주고 있으나, 실제 노드 간 전달시간을 측정한 것이 아니다.

C. Decker[5] 등은 비트코인 네트워크의 블록 전파에 대해 분석하고 있으나, 압축 블록 전달 방식을 다루고 있지 않으며 블록체인 포크에 집중하고 있다.

J. Mišić[7] 등은 압축 블록 전달 방식의 저대역폭 모드에서 일반 및 압축 블록 트래픽(traffic)이 혼합된 상태로 비트코인 네트워크의 운영 개선을 평가하고 있다. 그 결과 일반 블록에 비해 압축 블록의 크기는 10배 이상 작고, 전달시간 개선은 0% ~20% 내외라고 밝힌다.

이기영[8] 등은 압축 블록 전달 방식의 고대역폭 모드에서 압축 블록의 전달시간을 측정한다. 노드 간 직접 연결을 통해 블록 전달에 초점을 맞추고 있으며, 측정 결과를 정규분포 및 히스토그램 등을 이용하여 분석한다. 분석 결과 전달 지연이 있음을 밝힌다.

N. Till[9] 등은 압축 블록 전달 방식에서 인벤토리 메시지(Inventory 메시지; inv; 피어가 블록을 광고할 때 사용하는 메시지)를 이용하여 블록 전파의 지연을 측정한다. 측정 결과 지연이 있음을 보여준다. N. Till이 제안한 인벤토리 메시지로 전파시간을 측정하는 방법은 측정 노드의 위치와 네트워크 상황에 영향을 많이 받는다는 단점이 있다.

실제 비트코인 네트워크에서 압축 블록 전달시간을 측정한 논문은 부족하며, 측정하는 방법도 다양하여 제시하는 결과가 다르다. 또한, 압축 블록 전달시간을 측정하는 것에 그치고 전달 지연의 원인을 분석하는 논문은 없다.

본 논문에서는 실제 비트코인 네트워크에서 이기영의 측정방법[8]을 사용하여 압축 블록 전달시간을 측정한다. 측정한 결과를 바탕으로 다양한 시각에서 원인을 분석하고 밝히며, 전달시간을 단축할 수 있는 해결방안의 초석을 제공한다.

Ⅲ. 압축 블록 전달시간 실험 환경구성 및 방법

3.1 환경 구성

본 연구에서 사용된 컴퓨터(PC)는 i5-7500T CPU, 8GB의 메모리를 갖고 있으며, 저장 공간은 SSD 1TB 하드디스크를 사용한다. 운영체제는 리눅스 18.04 LTS 이다. 비트코인 코어를 "Bitcoin Core version v0.20.99.0-30568d3f1"버전으로 설치[10]하고 최신 블록까지 동기화를 진행한 다음에 블록 전달시간을 측정한다.

시스템 성능, 시스템 시간 그리고 네트워크 상태 등을 동일한 환경으로 구성하기 위하여 하나의 컴퓨터에 도커 컨테이너(Docker Container)[11]를 사용해 두 개의 비트코인 노드를 설치한다. 그림 2에서 주 노드(Main Node)는 비트코인 네트워크와 연결하여 일반적인 노드와 같이 8개 이상의 노드들과 연결되고, 부 노드(Sub Node)는 주 노드와 직접 연결한다. 이러한 직접 연결을 통해 다른 노드의 방해 없이 노드 간 데이터를 정확하게 측정한다.

지연 원인 분석의 4.2.1) 네트워크 부분에서 추가적인 검증 실험을 진행한다. 같은 컴퓨터 4대를 각각 대한민국의 포항, 춘천, 대구, 세종, 총 4개의 지역에 설치하여 같은 컴퓨터, 다른 네트워크라는 조건을 준다. 이를 통해 다른 네트워크를 사용함에도 특정 시간대의 트래픽량, 즉 네트워크의 상태가 전달시간에 영향을 미치는지 검증한다. 이때 컴퓨터의 사양은 'AMD Ryzen 7 3700X 8-Core Processor'의 CPU와 32GB 메모리, SSD 1TB의 스토리지를 사용한다.

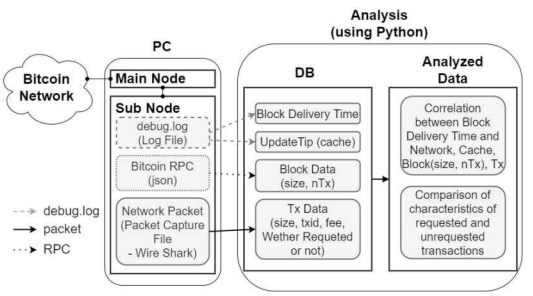

그림 2. 블록 전달시간 측정 환경 및 분석 방법
Fig. 2. Experiment Configuration and analysis method

3.2 실험 방법

본 실험에서는 블록 높이(Block height) 654000부터 656000까지 총 2,000개 블록의 전달 데이터를 수집 및 분석한다. 수집한 날짜는 2020년 10월 23일부터 11월 8일까지로 약 17일이다. 측정한 블록은 고대

역폭 전달과정을 따른다.

실험에 사용한 그림 2의 부 노드(Sub Node)는 블록을 전달받기만 하는 말단 노드에 해당하며 블록 수신 과정을 'debug.log'라는 이름의 로그 파일에 기록한다. 본 연구에서는 부 노드의 로그 정보를 통해 수집한 데이터로 블록 전달시간을 측정하고 분석한다.

그림 1과 2에 따라 블록 전달시간을 측정하는 방법은 다음과 같다. 전파의 시작은 압축 블록(cmpctblock) 메시지 패킷이다. 압축 블록 메시지 패킷을 전달받은 시각부터 블록 트랜잭션(blocktxn) 메시지 패킷을 받은 이후 해당 압축 블록을 성공적으로 조립한 시각을 로그에서 수집한다. 수집한 두 데이터의 시각차가 블록 전달시간(Block Delivery Time)이다. 전달시간을 지속적으로 사용하기 위하여 데이터베이스(Mongo DB 사용)에 저장한다. 또한, 추가로 트랜잭션 요청(getblocktxn) 메시지 패킷과 블록 트랜잭션 패킷 사이의 시각차를 구한다. 이는 블록을 완성하는데 필요한 트랜잭션을 요청하는 시간이다.

블록 전달시간을 알아낸 이후에는 블록 전달시간이 지연되었을 때 영향을 준 원인을 밝혀내기 위하여 블록 전달시간과 비교할 데이터를 최대한 다양하게 수집한다. 추가로 수집하는 데이터는 총 3개로 분류 가능하며 각 그림 2의 블록체인 연결 정보(UpdateTip), 블록 정보(Block Data), 트랜잭션 정보(Tx Data)이다. 블록체인 연결정보와 블록 정보는 비트코인의 RPC를 사용하여 얻는다. 트랜잭션 정보는 자세한 데이터를 얻기 위하여 주 노드와 부 노드 사이의 네트워크 패킷을 캡처한 뒤, 와이어샤크(WireShark)로 분석하여 얻는다. 수집된 데이터는 모두 데이터베이스에 저장하며, 이는 정제되지 않은 원본 데이터(Raw Data)이다. 모든 데이터를 수집하고 나면, 원본 데이터를 가져와 파이썬(Python) 패키지를 활용하여 통계 및 상관관계 분석을 수행한다.

파이썬으로 분석된 정보는 크게 2가지이다. 이는 그림 2의 분석된 데이터(Analyzed Data)로 나타난다. 먼저, 블록 전달시간의 지연 원인을 찾기 위해 블록 전달시간을 '네트워크 상태, 블록체인 연결정보의 캐시(cache), 블록 정보의 크기와 트랜잭션 개수, 트랜잭션 요청 시간, 요청 개수'와 상관관계 분석한 정보가 있다. 분석 결과인 상관 계수를 통해 비례와 반비례 관계를 밝히고 지연에 영향을 미치는지 판단한다.

그리고 주요 지연 원인으로 밝혀진 트랜잭션 요청의 이유를 추가로 분석하기 위해서, 요청한 트랜잭션과 요청하지 않은 트랜잭션의 특징(크기와 요금)을 비교하고 통계 낸 분석 정보가 있다. 비교 결과를 통하여 트랜잭션의 크기와 요금이 트랜잭션의 요청에 영향을 미치는지 판단한다.

분석 정보는 본 논문의 결과를 검증하는 데이터들이며, 표와 그래프를 통해 표현한다.

3.3 압축 블록 전달시간 분석 방법

3.3.1 상관관계 분석

상관 분석은 두 변수 간에 어떤 선형적 또는 비선형적 관계를 갖는지 알아보는 분석 방법이다. 두 변수는 서로 독립적인 관계이거나 상관된 관계일 수 있으며 관계의 강도를 상관 계수로 나타낸다. 상관 계수를 계산하는 식은 수식 1과 같다. X와 Y는 상관관계 분석에 사용되는 두 배열이다.

$$Correl(X,Y)= \frac{\sum(x-\bar{x})(y-\bar{y})}{\sqrt{\sum(x-\bar{x})^2\sum(y-\bar{y})^2}} \quad (1)$$

상관 계수가 +1 또는 -1에 가까울수록 두 값이 관계가 있음을 나타낸다. +1은 정비례, -1은 반비례를 의미한다. 보통 ±0.8 정도면 두 값이 연관이 있다고 판단한다.

상관관계 분석을 통해 전달시간과 지연 원인 요소 간의 상관관계를 밝혀, 해당 요소가 정말 전달시간의 지연 원인인지 파악한다.

Ⅳ. 전달시간 측정 결과와 지연 원인 분석

4.1 압축 블록 전달시간

다음은 비트코인 네트워크 내의 부 노드에서 블록 전달시간을 측정한 결과이다. 블록 높이 654000부터 656000까지 총 2,000개를 측정하여 그래프로 표현한 것이 그림 3이다.

그림 3은 블록 전달시간이 균일하지 않고, 편차가 크다는 것을 보여준다. 정확히 수치로 나타낸 것은 표 1이다. 평균은 0.266874초이고, 최솟값은 0.000275초,

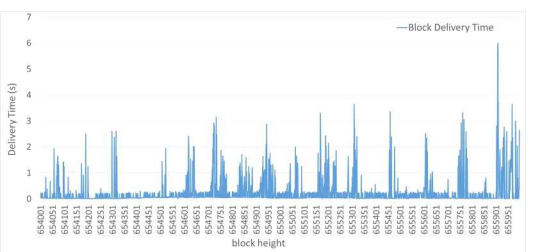

그림 3. 부 노드의 블록 전달시간
Fig. 3. block received time of Sub Node

표 1. 부 노드의 블록 전달시간 통계 (단위 : 초)
Table 1. Block received time statistics of Sub Node (unit: seconds)

Average	Min	Max	Dispersion	Standard Deviation
0.266874	0.000275	5.987413	0.311813	0.558402

최댓값은 5.987413초로, 최댓값이 최솟값의 약 20,000배이다. 분산과 표준편차를 보면 평균에서 변량들이 거리가 멀고, 퍼져 있지 않으며 편차가 크다. 즉, 전달시간은 균일하지 않고, 지연이 존재한다.

4.2 압축 블록 전달시간 지연 원인 분석

4.1의 그림 3 부 노드의 블록 전달시간을 보면 지연시간이 큰 블록들이 특정 구간에 몰려 있는 양상을 갖는다. 해당 구간의 블록 자체에 특징이 있는 것인지, 네트워크 상황에 영향을 받은 것인지 총 4개(네트워크, 체인 연결과 시스템 메모리양, 블록의 크기와 트랜잭션 보유 수, 트랜잭션 요청)의 요인을 통해 분석한다.

4.2.1 네트워크

해당 절에서는 패킷 전달에 큰 영향을 주는 네트워크 상태가 실제로 블록 전달 지연의 원인이 되는지, 원인이 맞으면 구체적으로 어떻게 영향을 미치는지 분석한다.

4.1의 그림 3 그래프에서 전달 지연 발생 구간은 주기성을 갖고 있다. 주기성을 정확하게 파악하기 위하여 주기도[12]를 사용한다. 측정한 날짜(2020년 10월 23일부터 11월 8일)를 바탕으로 일일 블록 생성 개수 평균을 구하면 125.2이며, 주기도 함수의 단위 시간당 표본 개수를 125개로 정한다. 전달시간 데이터와 단위 시간당 표본 개수를 이용하여 그린 주기도 그래프의 피크(Peak)는 1.09이다. 이는 약 하루마다 한 번씩 주기가 있음을 의미하며, 하루에 한 번 전달 지연이 발생함을 뜻한다.

하루 한 번의 전달 지연이 언제 발생하는지 밝히기 위하여 측정 데이터에서 전달시간이 1초 이상인 블록이 있는 구간 총 15개를 선정한다. 선정한 15개의 구간 중 전달시간이 최대로 걸린 블록의 높이와 전달시간, 그리고 받은 시각을 정리하고 분석한다.

분석 결과, 블록을 받은 시각은 주로 오전 1시에서 5시까지이다. 이는 우리나라 시간대로 오전 10시에서 오후 2시까지이며, 인터넷 러시아워(Internet Rush hour)와 연관된다.

대다수의 인터넷 사용자들이 동시에 온라인에 접속하는 기간을 인터넷 러시아워라고 하는데, 이때 네트워크 트래픽이 증가한다. 2020년 "Google Analytics report"[13]에 따르면 인터넷 러시아워는 오전 9시부터 오후 3시까지이며, 트래픽이 가장 높은 시간대는 오후 2시이다(총 세션 300만 중 20만 세션으로 6.4%를 차지). 분석 결과인 블록을 받은 시각과 겹친다. 즉, 블록 전달시간이 오래 걸린 블록들은 네트워크 트래픽이 많은 시간대에 있다는 의미이다.

이를 정확히 하기 위해 한가지 추가적인 실험으로 검증한다. 같은 컴퓨터 4대를 각각 대한민국의 포항, 춘천, 대구, 세종, 총 4개의 지역에 설치한다. 이러한 설치는 연구에서 같은 컴퓨터, 다른 네트워크라는 조건을 주어 네트워크가 전달시간에 영향을 미치는지 검증 가능하다.

블록 높이 656000~656500까지의 블록을 측정한 시간은 2020년 11월 9일 오전 1시부터 11월 12일 오전 6시까지로, 인터넷 러시아워가 총 3번 나타난다.

표 2는 4개 지역에 대하여 전체 블록의 전달시간 평균과 인터넷 러시아워 구간에서 블록 전달시간 평균을 비교한 표이다. 표 2를 통해서 포항, 대구, 춘천, 세종 각 4개의 지역이 전체의 평균보다 인터넷 러시아워 때에 전달시간이 2배에서 5배 정도 더 오래 걸림을 알 수 있다. 즉, 네트워크는 블록 전달 지연에 영향을 준다.

하지만 인터넷 러시아워가 아님에도 전달시간이 오래 걸린 블록들이 있다. 이는 네트워크 이외에도 전달시간에 영향을 주는 요소가 있음을 의미한다. 다음절부터는 네트워크를 제외하고 전달시간에 영향을 주는 요소가 무엇인지 분석한다.

표 2. 4개 지역 별 전체와 인터넷 러시아워 구간의 전달시간 평균 비교 (단위 : 초)
Table 2. Comparison of average received time for all 4 regions and Internet Rush Hour section (unit: seconds)

	Pohang	Daegu	Chuncheon	Sejong
Total (500개)	0.22695	0.999995	0.045059	0.36076
Internet Rush Hour	0.695844	2.051231	0.188847	1.751719

4.2.2 체인 연결과 시스템 메모리양, 블록의 크기와 트랜잭션 수

부 노드의 로그 중에서 블록체인 연결 정보

(UpdateTip) 로그는 새로운 블록을 발견하고 블록체인에 연결했을 때 출력되는 로그로, 블록과 체인에 대한 정보(블록의 해시값, 블록 높이, 마이닝 합의 알고리즘 버전, 블록 완성 시간, 캐시(cache), 체인의 전체 트랜잭션 수)를 담고 있다. 해당 로그 중 캐시를 분석한다. 캐시는 메모리에 할당된 유티엑스오(UTXO; Unspent Transaction Output) 캐시의 양으로, 체인 연결 전 블록을 검증할 때 필요한 유티엑스오를 디스크에서 가져와 메모리에 할당할 수 있을 만큼 할당하여 사용한다[14]. 즉, 검증에 필요한 유티엑스오의 캐시를 메모리에 가져올 때 지연을 유발할 수 있다. 따라서, 전체 2,000개 블록의 캐시 값을 분석한다.

분석 결과는 그림 4로, 블록 높이 별 캐시 값과 전달시간을 그린 그래프이다. 그림 4의 캐시를 보면 로그에 찍힌 캐시 값이 규칙적으로 증가했다가 감소하기를 반복한다. 이는 캐시의 값이 단순히 블록 검증을 위해 가져왔던 유티엑스오의 누적량임을 보여준다. 또한 전달시간과 비교했을 때, 전달시간이 오래 걸린 블록들의 캐시 값은 123, 53, 16 등 최대 캐시 값인 159에 비해 적고, 상관이 없었다. 즉, 캐시 값과 전달시간은 관계가 없음을 알 수 있다.

다음은 블록의 크기와 트랜잭션 수에 대한 분석이다. Bitcoin RPC인 bitcoin-cli의 "getblock" 명령어를 사용하여 블록에 대한 정보(해시, 크기, 높이, 버전, 트랜잭션 아이디 목록, 논스, 블록에 포함된 트랜잭션 수 등 19가지)를 받는다. 블록의 정보는 모니터링 사이트[15]에서도 확인할 수 있다. 이중 블록 전달시간과 밀접한 블록의 크기(size)와 블록이 포함하고 있는 트랜잭션의 수(nTx)를 분석한다.

그림 5는 블록 전달시간과 블록의 크기 분포를, 그림 6은 블록 전달시간과 트랜잭션의 수 분포를 표현한 그래프이다. 두 그래프는 비례관계가 없음을 보여준다. 상관 계수 계산 결과, 전달시간과 트랜잭션 수의 상관 계수는 -0.21971614로 음(-)의 값을 갖고 있

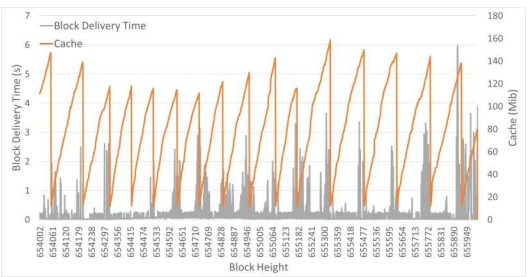

그림 4. 블록 높이 별 캐시
Fig. 4. Cache by block height

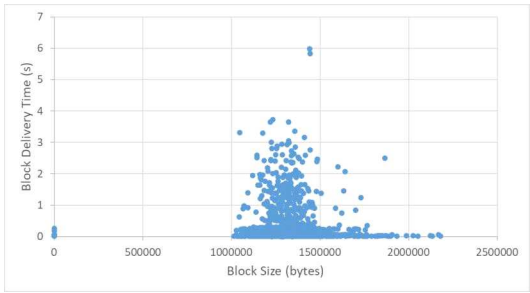

그림 5. 블록 전달시간과 블록 크기 분포 그래프
Fig. 5. Block delivery time and Block Size distribution

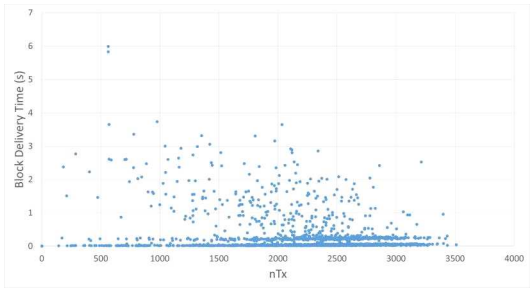

그림 6. 블록 전달시간과 트랜잭션의 수 분포 그래프
Fig. 6. Block delivery time and nTx distribution

으며, 0.8에도 미치지 않는다. 전달시간과 크기의 상관 계수는 0.070191323으로 비례 관계이나 값이 너무 작다. 즉, 블록의 크기와 트랜잭션의 수는 전달시간에 영향을 미치지 않는다.

4.2.3 트랜잭션 요청

주 노드와 직접 연결된 부 노드의 압축 블록 전달 로그(debug.log 파일 사용)를 통해 패킷 흐름과 타임스탬프(Timestamp)를 분석한다.

전달시간이 최대로 걸린 블록들의 로그 흐름은 그림 1의 압축 블록 전달과정과 같이 정석대로 패킷을 주고받는다. 하지만 최소로 걸린 블록들은 트랜잭션 요청(getblocktxn) 메시지를 보낸 로그가 없고, 블록 트랜잭션(blocktxn) 메시지만 있다. 즉 트랜잭션을 요청하지 않는다. 또한, 전달시간이 최대로 걸린 블록의 타임스탬프를 확인해보면 트랜잭션 요청 메시지 패킷과 블록 트랜잭션 메시지 패킷 사이에서 시간이 소요됨을 알 수 있다. 이 시간을 트랜잭션 요청 시간이라 정의하고, 전달시간에서 트랜잭션 요청 시간이 차지하는 비율을 계산한 결과는 98~99%이다. 해당 비율을 그래프로 그린 것이 그림 7이다.

측정 블록 2,000개 중 트랜잭션을 요청한 블록(754개로 전체의 37.7%를 차지)에 대해서 블록 전달시간

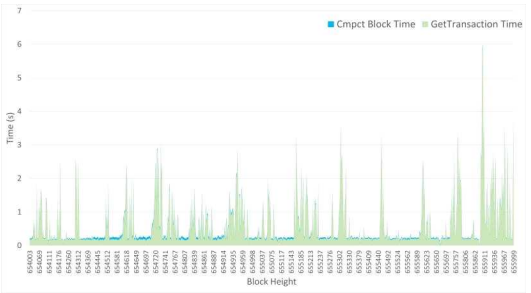

그림 7. 블록 전달시간과 트랜잭션 요청시간 비교
Fig. 7. Comparison of block delivery time and transaction request time

(Cmpct Block Time)과 트랜잭션 요청 시간 (GetTransaction Time)을 영역형 그래프로 표현한 것이다.

분석 결과는 이전 연구[16]의 결과가 검증한다. 전체 전달시간을 압축 블록을 받는 구간과 트랜잭션을 요청하는 구간으로 나누었을 때, 트랜잭션을 요청하는 시간 평균이 압축 블록을 받는 시간보다 약 100배 정도 차이가 있다는 것이 이전 연구의 결론이다. 본 논문의 연구 결과와 이전 연구의 결론을 종합했을 때, 전달 지연은 트랜잭션 요청과 관련이 있다는 걸 알 수 있다.

정확히 트랜잭션 요청을 하지 않았을 때와 트랜잭션 요청을 했을 때로 나누어서 전달시간을 분석하고 통계 낸 결과는 다음과 같다. 트랜잭션을 요청했을 때는 전달시간의 최댓값이 약 6초, 요청하지 않을 때는 최솟값이 약 0.2초 정도로 두 값은 30배 차이가 난다. 트랜잭션을 요청했을 때 전달시간의 평균은 약 0.6초, 요청하지 않았을 때의 평균은 약 0.03초로 12배의 차이가 난다. 표준편차 역시도 의미가 있는데, 트랜잭션을 요청했을 때보다 트랜잭션을 요청하지 않았을 때의 표준편차가 더 작다. 트랜잭션을 요청했을 때는 각 값이 평균에 비해서 고르게 퍼져 있으나, 트랜잭션을 요청하지 않았을 때는 각 값이 평균에 근접하게 뭉쳐 있다는 뜻이다.

종합하자면, 트랜잭션을 요청하지 않은 블록은 요청했을 때 비해 12배나 적은 시간인 0.03초대로 전달시간을 소요한다.

요청한 트랜잭션 개수에 대해서도 추가로 분석한다. 요청한 트랜잭션 개수는 압축 블록을 성공적으로 조립했을 때 찍히는 로그와 와이어샤크 패킷 분석으로 수집한다. 요청한 트랜잭션 개수와 전달시간의 관계를 알아보기 위해 분포도를 그리고 상관관계 분석 방법으로 상관 계수를 구한다. 그림 8은 두 값의 분포

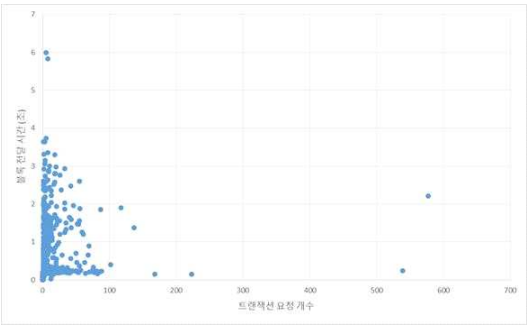

그림 8. 요청한 트랜잭션 개수와 전달시간 분포 그래프
Fig. 8. Distribution of the number of requested transactions and delivery time

도를 그린 그림이다. 그림 8의 그래프는 정비례 관계가 없음을 알려준다. 두 값의 상관 계수는 0.191522246이다. 비례관계이지만 약 0.2 정도의 작은 숫자가 나와 요청한 트랜잭션 개수와 전달시간은 연관이 없음을 의미한다.

앞선 분석을 통해 트랜잭션을 요청하는 것이 블록 전달 지연에 영향을 미친다는 결론을 내렸다. 이를 바탕으로 트랜잭션을 요청하는 이유에 대해서 추가로 분석한다. 분석에 사용한 블록은 이전 실험과 동일한 환경에서 2021년 5월 5일에 측정된 블록으로 총 171개이다.

트랜잭션의 요금(fee)을 책정할 때에 'fee per byte'라는 용어가 있다. 바이트당 요금을 의미하며 트랜잭션의 크기가 크면 보통 요금도 크다. 또한, 마이너들은 요금이 높은 트랜잭션을 우선순위로 블록에 포함하기 때문에 트랜잭션의 요금이 크면 블록에 빠르게 포함될 수 있다. 따라서 요금과 크기가 큰 트랜잭션은 트랜잭션 요청 메시지로 요청할 확률이 높아진다. 블록에 빠르게 포함되어, 아직 메모리 풀에 해당 트랜잭션을 전달받지 않을 확률이 높기 때문이다.

이를 검증하기 위하여 171개의 블록에 포함된 트랜잭션들을 요청한 트랜잭션과 요청하지 않은 트랜잭션으로 나누어서 요금과 크기를 비교한다. 크기를 비교한 표가 표3, 요금을 비교한 표가 표4이다. 요금의 경우, 트랜잭션 생성 후에 요금을 변경시키는 수정된 요금(modified fee)은 값의 변동이 없어 고려하지 않았다.

표 3과 4를 보면 요청하지 않은 트랜잭션의 표준편차가 요청한 트랜잭션보다 작다. 이는 각 크기 평균 292, 요금 평균 0.00018과 유사한 값이 많음을 의미한다. 표3과 4의 Q1, Q2(Median), Q3는 사분위수 범위(IQR)[17] 분석 방법으로 나온 결과이며, 이는 그림 9

표 3. 트랜잭션 크기 비교 (단위 : Bytes)
Table 3. Transaction size comparison (Unit: Bytes)

	Request Tx	Unrequest Tx
Average	333.364	292.777
Standard Deviation	897.34	552.911
Q1	224	222
Q2 (Median)	247	225
Q3	257	249

표 4. 트랜잭션 요금 비교 (단위 : BTC)
Table 4. Transaction fee comparison (Unit: BTC)

	Request Tx	Unrequest Tx
Average	0.00022912	0.00018340
Standard Deviation	0.00075356	0.00037983
Q1	0.00007872	0.00005840
Q2 (Median)	0.00018709	0.00013559
Q3	0.00023399	0.00020317

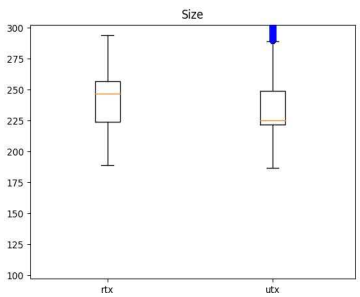

그림 9. 요청한 트랜잭션과 요청하지 않은 트랜잭션의 크기에 대한 사분위수 범위 그래프
Fig. 9. IQR graph of the size of requested and unrequested transactions

와 10의 사분위수 범위 그래프를 통해 더 정확히 보여준다. 사분위수 범위 분석 결과, 요청한 트랜잭션의 크기와 요금의 Q2가 요청하지 않은 트랜잭션보다 Q3에 근접한다. 이는 요청한 트랜잭션의 요금과 크기가 더 큼을 의미한다.

결론적으로 추가 수행한 분석은 요금과 크기가 큰 트랜잭션이 트랜잭션 요청 메시지로 요청될 확률이 높다는 것을 검증하고 보여준다.

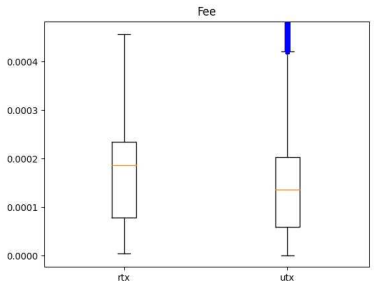

그림 10. 요청한 트랜잭션과 요청하지 않은 트랜잭션의 요금에 대한 사분위수 범위 그래프
Fig. 10. IQR graph of the fee of requested and unrequested transactions

V. 결론 및 향후 연구

5.1 결론

본 논문에서는 실제 비트코인 네트워크의 고대역폭 압축 블록 전달시간을 측정하여 성능을 보여주고 지연 원인을 분석하기 위해, 하나의 컴퓨터에 비트코인 노드 두 개를 설치하고 두 노드를 직접 연결한 후 실험을 진행한다. 2020년 10월 23일부터 11월 8일까지 약 17일 정도 측정했으며 블록 높이 654000부터 656000까지 총 2,000개의 블록 데이터를 측정한다.

측정 결과, 블록 전달시간은 평균 0.0266874초이고, 최대 최솟값의 차이가 약 20,000배 나면서 전달시간이 균일하지 않고 지연이 존재함을 보여준다.

측정 결과를 바탕으로 다양한 관점에서 압축 블록 전달의 지연 원인 분석 결과를 제공한다. 총 4개(네트워크, 트랜잭션 요청, 체인 연결과 시스템 메모리양, 블록의 크기와 트랜잭션 보유 수)의 요인을 통해 찾아낸 지연 원인은 다음과 같다. 블록 전달 지연은 네트워크가 인터넷 러시아워와 겹쳐서 트래픽이 증가함에 따라 발생하는 경향이 있으며, 압축 블록을 조립할 때 트랜잭션 요청 여부에 영향을 받는다.

추가로 주요 원인인 트랜잭션 요청 이유에 대하여 트랜잭션의 요금과 크기를 중점으로 심층적인 분석을 수행한 결과, 요금과 크기가 클 때 트랜잭션 요청 확률이 높아짐을 밝혔다.

본 논문의 분석 결과는 블록 전달시간을 단축하기 위한 연구에 기초 데이터로써 활용 가치가 있다. 분석 결과를 통해 전달 지연 원인을 해결하고 지연시간을 단축하는 방법을 논의해봄으로써 압축 블록 전달 방식의 속도 성능을 개선할 수 있다. 방법은 다음 절에서 제안한다.

5.2 향후 연구

먼저, 인터넷 러시아워로 인한 트래픽 증가의 경우에는 차후 인터넷과 통신의 발달에 따라 지연이 감소할 수 있을 것으로 예측한다. 아니면, 인터넷 러시아워 구간을 제외한 시간대에 사람들이 거래를 하도록 유도하는 방법을 제시할 수 있다. 예를 들어 오후 2시 이후로 거래를 하면 인센티브(incentive)를 주는 방식이다. 이를 통해 트래픽량이 몰리는 것을 완화하고, 지연을 감소할 수 있다.

트랜잭션 요청 여부는 블록에 포함된 트랜잭션을 본인의 메모리 풀에 보유하고 있는지에 따라 지연될 수 있음을 뜻한다. 이와 관련하여 메모리 풀의 동기화에 관한 연구를 J. Mišić[18] 등이 진행한 바 있다. 메모리 풀의 동기화뿐 아니라 트랜잭션 전파에 초점을 두고 향후 연구를 진행할 수 있다.

트랜잭션 요청(getblocktxn) 메시지로 요청하는 트랜잭션이 크기와 요금이 큰 경우임을 안다. 이를 통해 요청할 트랜잭션을 예측할 수 있다. 블록에 포함된 트랜잭션 중 크기와 요금이 큰 트랜잭션을 기존의 블록 전달 방식처럼 블록 전파할 때 함께 전파하여 트랜잭션 요청의 확률을 줄인다. 전파하는 구체적인 방안에 대해서는 논의가 필요하다. 제안하는 방법은 압축 블록 전파방식의 의의를 헤칠 수 있으나, 요청 확률의 감소로 블록 전파시간이 감소한다면 결과적으로 효율은 높아진다. 앞서, 4.2.3에서도 밝혔듯이 트랜잭션 요청 시간은 전달시간의 98% 이상을 차지하기 때문이다. 트랜잭션을 요청하는 블록이 줄어들면 전체적으로 전달시간도 단축되며 압축 블록 전달 프로토콜의 성능과 비트코인 네트워크가 향상될 것이다.

References

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized Business Review*, 21260, 2008.

[2] M. Corallo, *BIP 152: Compact block relay(2020)*, Retrieved Mar. 21, 2021, from https://github.com/bitcoin/bips/blob/master/bip-0152.mediawiki

[3] BitcoinCore, *Compact Blocks FAQ(2016)*, Retrieved Mar. 21, 2021, from https://bitcoincore.org/en/2016/06/07/compact-blocks-faq

[4] N. Courtois, S. Guangyan, and R. Castellucci, "Speed optimizations in Bitcoin key recovery attacks," *Tatra Mountains Math. Publications-The J. Slovak Academy of Sci.*, vol. 67, no. 1, pp. 55-68, 2016.

[5] C. Decker and R. Wattenhofer, "Information propagation in the Bitcoin network," *IEEE P2P 2013,* pp. 1-10, Trento, Italy, Dec. 2013, doi: 10.1109/P2P.2013.6688704.

[6] R. Nagayama, R. Banno and K. Shudo, "Identifying impacts of protocol and internet development on the bitcoin network," *2020 ISCC,* pp. 1-6, 2020, doi: 10.1109/ISCC50000.2020.9219639.

[7] J. Mišić, V. B. Mišić and X. Chang, "On the benefits of compact blocks in bitcoin," *ICC 2020 - 2020 IEEE ICC,* pp. 1-6, 2020, doi: 10.1109/ICC40277.2020.9149418.

[8] K. Lee and H. Ju, "Measurement and analysis of direct connection node delivery time of bitcoin block based on log output," in *Proc. KICS Winter Conf.*, pp. 644-646, 2020.

[9] N. Till, "*Characterization of the Bitcoin Peer-to-Peer Network (2015-2018),*" KIT Karlsruher Institut für Technologie, Fakultät für Informatik, 2019, doi: 10.5445/IR/1000091933.

[10] GitHub, *bitcoin(2021)*, Retrieved Mar. 21. 2021, from https://github.com/bitcoin/bitcoin

[11] docker, *docker(2021)*, Retrieved Mar. 21. 2021, from https://www.docker.com

[12] MathWorks, *MATLAB(2021)*, Retrieved Mar. 21. 2021, from https://kr.mathworks.com/products/ matlab.html

[13] Hallam, *Google Analytics: Hour of day & day of week reports(2021)*, Retrieved Mar. 21. 2021, from https://www.hallaminternet.com/google-analytics-hour-of-day-day-of-week-reports

[14] S. Delgado-Segura, C. Pérez-Solà, G. Navarro-rribas and J. Herrera-Joancomartí, "Analysis of the Bitcoin UTXO Set, in: A. Zohar, et al., Financial cryptography and data security," *FC 2018. LNCS,* vol. 10958, Springer, Berlin, Heidelberg, https://doi.org/10.1007/978-3-662-58820-8_6, 2006.

[15] Blockchain.com, *Bitcoin Explorer(2022)*,

Retrieved Apr. 21, 2022, from https://www.block chain.com/explorer

[16] A. Kim, S. Maeng, E. Meryam, and H. Ju, “Measurement and analysis of single connected bitcoin node’s compact block delivery time based on log output,” in *Proc. KSC 2020*, pp. 1539-1541, Dec. 2020.

[17] Wikipedia, *Interquartile range(2022)*, Retrieved Apr. 21, 2022, from https://en.wikipedia.org/wiki/Interquartile_range

[18] J. Mišić, V. B. Mišić, and X. Chang, “Performance of bitcoin network with synchronizing nodes and a mix of regular and compact blocks,” in *IEEE Trans. Netw. Sci. and Eng.*, vol. 7, no. 4, pp. 3135-3147, Oct.-Dec. 2020, doi: 10.1109/TNSE.2020.3017453.

김 애 리 (Aeri Kim)

2021년 2월 : 계명대학교 컴퓨터공학과 졸업

2021년 3월~현재 : 계명대학교 컴퓨터공학과 석사과정

<관심분야> 컴퓨터 공학, 네트워크, 블록체인

[ORCID:0000-0001-7677-2566]

주 홍 택 (Hongtaek Ju)

1989년 2월 : KAIST 컴퓨터공학과 졸업

1991년 2월 : POSTECH 컴퓨터공학과 석사

2002년 2월 : POSTECH 컴퓨터공학과 박사

2002년 9월~현재 : 계명대학교 컴퓨터공학과 교수

<관심분야> 컴퓨터 공학, 네트워크, 블록체인

[ORCID:0000-0002-8434-485X]