



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2018-0069609
(43) 공개일자 2018년06월25일

(51) 국제특허분류(Int. Cl.)

H04L 9/08 (2006.01)

(52) CPC특허분류

H04L 9/0852 (2013.01)

H04L 9/0819 (2013.01)

(21) 출원번호 10-2016-0171949

(22) 출원일자 2016년12월15일

심사청구일자 없음

(71) 출원인

고려대학교 산학협력단

서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)

(72) 발명자

허준

서울특별시 강남구 광평로10길 6, 206동 103호

박주윤

경기도 가평군 청평면 버들숲로8번길 75-4

(74) 대리인

심경식, 홍성욱

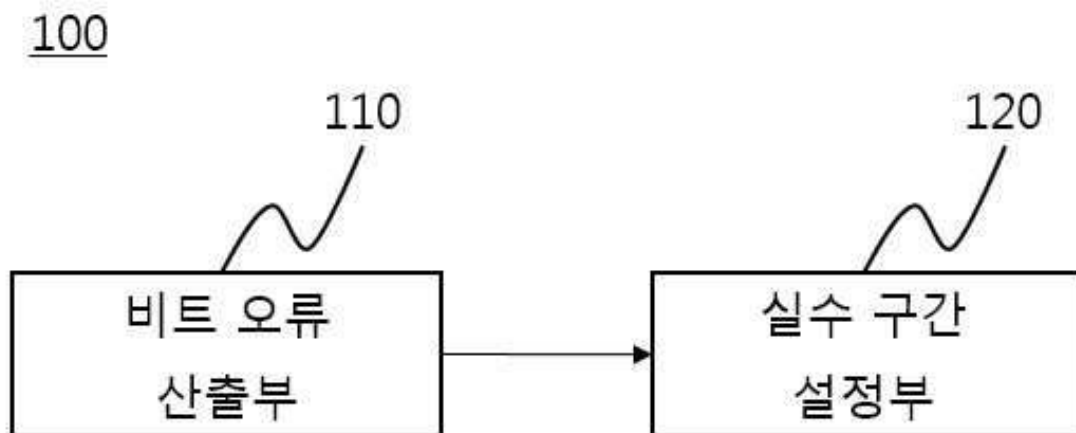
전체 청구항 수 : 총 1 항

(54) 발명의 명칭 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치 및 방법

(57) 요약

연속 변수 양자 키 분배를 위한 슬라이스드 리컨실리에이션 (Sliced Reconciliation)에서 실수값 키를 이진값 키로 매칭하기 위한 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치에 있어서, 실수값 키에 대응되는 각각의 비트에 대한 오류율을 산출하는 비트 오류 산출부 및 전체 실수 구간을 오류율을 최소화하는 하나 이상의 부분 실수 구간으로 분할하는 실수 구간 설정부를 포함하는, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치 및 방법에 관한 것이다.

대표도 - 도1



이 발명을 지원한 국가연구개발사업

과제고유번호 1711035212

부처명 미래창조과학부

연구관리전문기관 정보통신기술진흥센터

연구사업명 정보통신기술인력양성

연구과제명 무결점 보안을 위한 지상/위성 양자통신 기술 개발

기 여 율 1/1

주관기관 고려대학교 산학협력단

연구기간 2015.06.01 ~ 2018.12.31

명세서

청구범위

청구항 1

연속 변수 양자 키 분배를 위한 슬라이스드 리컨실리에이션 (Sliced Reconciliation)에서 실수값 키를 이진값 키로 매칭하기 위한 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치에 있어서,

상기 실수값 키에 대응되는 각각의 비트에 대한 오류율을 산출하는 비트 오류 산출부; 및

전체 실수 구간을 상기 오류율을 최소화하는 하나 이상의 부분 실수 구간으로 분할하는 실수 구간 설정부를 포함하는, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치.

발명의 설명

기술 분야

[0001] 본 발명은 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치 및 방법에 관한 것이다.

배경 기술

[0002] 양자 키 분배 프로토콜은 양자역학의 성질을 이용하여 송신자와 수신자가 서로 안전한 키를 나눠 갖는 프로토콜을 의미하며, 양자 키 분배 프로토콜에서 송신자와 수신자는 서로 양자채널 및 공개채널을 사용하여 키를 나누어 가지며, 양자 키 분배 프로토콜에서 실제 키로 사용하는 메시지를 보내는 채널은 양자 채널이다.

[0003] 양자 키 분배 프로토콜에서 양자채널을 활용하여 송신자는 수신자에게 0 혹은 1로 인코딩된 키를 보내게 되는데, 양자채널의 불완전성 또는 도청자의 존재로 인해 오류가 발생할 수 있으며, 이 때 생긴 오류를 정정하여 송신자 및 수신자가 서로 완벽하게 같은 키를 사용할 수 있도록 하기 위하여, 송신자는 추가 암호 키에 대한 추가 정보를 공개채널로 수신자에게 보낸다.

[0004] 한편, 연속 변수 양자 키 분배 프로토콜은 기존의 단일 광자를 사용하는 양자 키 분배 프로토콜을 개량한 기술로 구현이 용이하고 장비가 저렴한 장점이 있으며, 연속 변수 양자 키 분배 프로토콜은 기존의 양자 키 분배 프로토콜과 달리 송신자 및 수신자 상호 간에 키를 분배하고 난 후의 결과 값이 임의의 실수 중의 하나의 값을 가진다.

[0005] 그러나, 연속 변수 양자 키 분배 프로토콜에서는 양자 채널의 노이즈와 광자 측정기기 등의 불완전성 등에 의해 수신자는 송신자의 키 값과 다른 값을 수신하게 되며, 송신자 및 수신자가 서로 동일한 키 값을 가져야 인증이 가능한 일반적인 양자 키 분배 프로토콜에서 오류를 정정하는 과정인 정보 조정(information reconciliation)은 연속 변수 양자 키 분배 프로토콜에 그대로 적용될 수 없는 문제가 있다.

[0006] 따라서, 연속 변수 양자 키 분배 프로토콜에서는 키 분배 후처리 과정을 통해 서로 나눠가진 실수 값을 이진수 값으로 바꾸는 과정이 필요하며, 그 중 대표적인 방법이 슬라이스드 리컨실리에이션(sliced reconciliation) 프로토콜이다.

[0007] 그러나, 종래의 슬라이스드 리컨실리에이션(sliced reconciliation) 프로토콜에서는 순 정보량을 최대로 하는 분할 방법을 사용하며, 이는 완벽한 오류 정정이 가능함을 전제로 하기 때문에 실제로 키로 사용할 수 있는 비트 수가 적어지는 문제가 있다.

선행기술문헌

특허문헌

[0008] (특허문헌 0001) 공개특허공보 제10-2008-0034211호 (2008.04.18.)

발명의 내용

해결하려는 과제

- [0009] 본 발명의 목적은, 상기 문제점을 해결하기 위한 것으로, 실수값 키에 대응되는 각각의 비트에 대한 오류율을 산출하고, 전체 실수 구간을 오류율을 최소화하는 하나 이상의 부분 실수 구간으로 분할하여 연속 변수 양자 키 분배를 위해 실수 구간을 분할하기 위함이다.
- [0010] 본 발명이 해결하고자 하는 과제는 이상에서 언급한 과제(들)로 제한되지 않으며, 언급되지 않은 또 다른 과제(들)은 아래의 기재로부터 당업자에게 명확하게 이해될 수 있을 것이다.

과제의 해결 수단

- [0011] 상기한 목적을 달성하기 위하여, 본 발명의 실시예에 따른 연속 변수 양자 키 분배를 위한 슬라이스드 리컨실리에이션 (Sliced Reconciliation)에서 실수값 키를 이진값 키로 매칭하기 위한 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치는, 실수값 키에 대응되는 각각의 비트에 대한 오류율을 산출하는 비트 오류 산출부 및 전체 실수 구간을 오류율을 최소화하는 하나 이상의 부분 실수 구간으로 분할하는 실수 구간 설정부를 포함한다.

발명의 효과

- [0012] 본 발명의 일 실시예에 따르면, 실수값 키에 대응되는 각각의 비트에 대한 오류율을 산출하고, 전체 실수 구간을 오류율을 최소화하는 하나 이상의 부분 실수 구간으로 분할하여 연속 변수 양자 키 분배를 위해 실수 구간을 분할함으로써, 실제 사용 가능한 비트 수를 늘릴 수 있다.

도면의 간단한 설명

- [0013] 도 1은 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치를 설명하기 위한 구성도이다.
- 도 2는, 종래의 연속 변수 양자 키 분배에서 실수 값의 키를 이진수 키로 변환하는 방법을 설명하기 위한 도면이다.
- 도 3은, 종래의 연속 변수 양자 키 분배에서 수신자가 송신자의 키를 추측하는 방법을 설명하기 위한 도면이다.
- 도 4a는 종래의 연속 변수 양자 키 분배에서 적용되는 분할(Slicing) 기준을 설명하기 위한 도면이다.
- 도 4b는 종래의 연속 변수 양자 키 분배에서 적용되는 분할(Slicing) 기준을 적용한 결과 사용 가능한 비트 수를 설명하기 위한 도면이다.
- 도 5a는 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치에서 적용되는 분할(Slicing) 기준을 설명하기 위한 도면이다.
- 도 5b는 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치에서 적용되는 분할(Slicing) 기준을 적용한 결과 사용 가능한 비트 수를 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0014] 이하, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자가 본 발명의 기술적 사상을 용이하게 실시할 수 있을 정도로 상세히 설명하기 위하여, 본 발명의 가장 바람직한 실시예를 첨부 도면을 참조하여 설명하기로 한다. 우선 각 도면의 구성요소들에 참조부호를 부가함에 있어서, 동일한 구성요소들에 대해서는 비록 다른 도면상에 표시되더라도 가능한 한 동일한 부호를 가지도록 하고 있음에 유의해야 한다. 또한, 본 발명을 설명함에 있어, 관련된 공지 구성 또는 기능에 대한 구체적인 설명이 본 발명의 요지를 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명은 생략한다.
- [0015] 이하, 본 발명의 실시예에 따른 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치 및 방법을 첨부된 도면을 참조하여 상세하게 설명하면 아래와 같다.
- [0016] 도 1은 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치를 설명하기 위한 구성도이다. 도 2는, 종래의 연속 변수 양자 키 분배에서 실수 값의 키를 이진수 키로 변환하는 방법을 설명하기 위한

도면이다. 도 3은, 종래의 연속 변수 양자 키 분배에서 수신자가 송신자의 키를 추측하는 방법을 설명하기 위한 도면이다. 도 4a는 종래의 연속 변수 양자 키 분배에서 적용되는 분할(Slicing) 기준을 설명하기 위한 도면이다. 도 4b는 종래의 연속 변수 양자 키 분배에서 적용되는 분할(Slicing) 기준을 적용한 결과 사용 가능한 비트 수를 설명하기 위한 도면이다. 도 5a는 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치에서 적용되는 분할(Slicing) 기준을 설명하기 위한 도면이다. 도 5b는 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치에서 적용되는 분할(Slicing) 기준을 적용한 결과 사용 가능한 비트 수를 설명하기 위한 도면이다.

[0017] 도 1에 도시된 바와 같이, 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치(100)는 비트 오류 산출부(110) 및 실수 구간 설정부(120)를 포함한다.

[0018] 예컨대, 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치(100)는 연속 변수 양자 키 분배를 위한 슬라이스드 리컨실리에이션 (Sliced Reconciliation)에서 실수값 키를 이진값 키로 매칭하기 위한 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치를 의미할 수 있다.

[0019] 비트 오류 산출부(110)는 실수값 키에 대응되는 각각의 비트에 대한 오류율을 산출한다.

[0020] 실수 구간 설정부(120)는 전체 실수 구간을 오류율을 최소화하는 하나 이상의 부분 실수 구간으로 분할한다.

[0021] 도 2에 도시된 바와 같이, 슬라이스드 리컨실리에이션(sliced reconciliation)은 연속 변수 양자 키 분배 프로토콜에서 송신자와 수신자가 나눠가진 가우시안 변수를 이진수 값으로 변환시키는 방법 중 하나이다.

[0022] 슬라이스드 리컨실리에이션(sliced reconciliation)에서 송신자와 수신자가 나눠가진 키 값은 실수 전체를 범위로 갖는데, 슬라이스드 리컨실리에이션(sliced reconciliation)은 실수 전체 범위를 기설정된 숫자의 구간으로 나눈 뒤, 나누어진 각 구간에 이진수 번호를 붙이고, 송신자는 실수 값의 키를 그 키가 속해있는 구간의 이진수 값으로 바꾸는 방법을 의미할 수 있다.

[0023] 도 3에 도시된 바와 같이, 슬라이스드 리컨실리에이션(sliced reconciliation)에서 수신자(Bob)는 LSB로부터 차례대로 그 값을 추정하게 되며, 수신자가 받은 실수 값의 키와 송신자(Alice)를 통해 알게 된 이전의 이진수 값을 바탕으로 키를 추정하게 되고, 수신자(Bob)는 추정한 비트의 오류율을 계산하여 기설정된 오류 정정 부호를 활용하여 그 오류를 정정하면서 다음 자리의 비트를 다시 추정하고 오류를 정정하는 과정을 MSB까지 반복하게 된다.

[0024] 그러나, 종래의 슬라이스드 리컨실리에이션(sliced reconciliation)은 이론적으로 송신자와 수신자가 최대로 가질 수 있는 정보량을 계산하여 구간을 나누는 방법을 사용하였지만, 실제로는 LSB와 그 다음의 비트는 사용할 수 없을 만큼 오류율이 높기 때문에 오류정정 과정을 거치지 않고 공개해야 한다.

[0025] 예를 들어, 종래의 슬라이스드 리컨실리에이션(sliced reconciliation)을 활용하여 한 개의 실수 값을 5개의 비트로 바꾸는 경우, 실제로 키로 사용할 수 있을 만큼 오류율이 낮은 비트는 2개에 불과하며, 이론적인 정보량에 비해 사용할 수 있는 정보량이 줄어드는 문제가 있다.

[0026] 예컨대, 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치(100)는 송신자와 수신자 상호 간의 정보량을 극대화하는 구간을 설정하는 방법을 대신하여 각 비트의 오류율을 줄이는 방법을 활용함으로써, 실제로 사용할 수 있는 키의 정보량을 늘릴 수 있으며, 종래의 슬라이스드 리컨실리에이션(sliced reconciliation) 방법에 비하여 실제적으로 더 높은 정보량을 송신자와 수신자가 나눠가질 수 있기 때문에 한번의 통신으로 더 많은 양의 비밀 키를 나눌 수 있다.

[0027] 예컨대, 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치(100)가 오류 정정을 진행하기 위해서는 비트의 오류율이 기설정된 값을 넘어서는 안되고, 예컨대, LDPC code 혹은 CASCADE를 사용하는 경우 각각 오류율이 기설정된 값인 0.1 또는 0.2를 넘는 경우가 발생하기 때문에 오류정정의 효율이 떨어진다.

[0028] 예컨대, 비트 오류 산출부(110)가 산출하는 오류율은 아래 수학적 식 1에 기초하여 산출될 수 있다.

[0029] [수학적 식 1]

$$e_i = \int dx' \sum_{\beta \in GF(2)} i-1 \min_a \Pr[S_i(X) = a \wedge S_{1..i-1}(X) = \beta \wedge X' = x']$$

[0031] 예를 들어, 종래의 슬라이스드 리컨실리에이션(sliced reconciliation) 방법은 완벽한 오류정정이 가능하다는 가정을 하고 송신자와 수신자 사이의 정보량을 최대로 하는 구간을 설정한다.

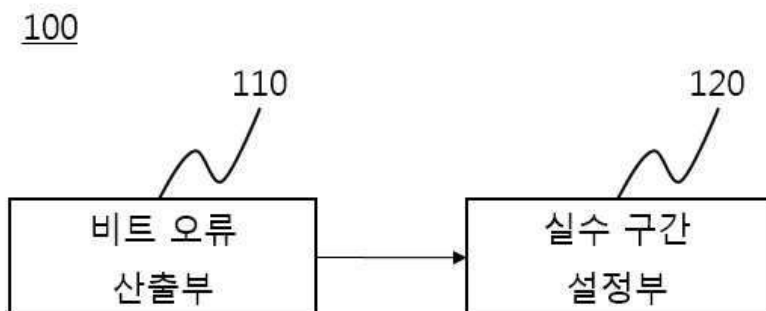
- [0032] 그러나, 실제 상황에서는 완벽한 효율의 오류정정이 이루어지지 않기 때문에 구간을 설정하는 기준을 달리 해야 하며, 실수 구간 설정부(120)는 LSB부터 차례대로 오류율을 기설정된 기준보다 낮도록 설정하는 실수 구간을 특정 시뮬레이션을 통해서 찾을 수 있다.
- [0033] 예를 들어, 키로 생성되는 실수 값은 0을 중심으로 하는 정규분포를 그리기 때문에 0 근처에서 오류가 발생하기 쉬우며, 실수 구간 설정부(120)는 0 근처의 구간의 길이를 기존의 방법보다 상대적으로 길게 설정하여 오류율이 줄어들도록 실수 구간을 분할 할 수 있다.
- [0034] 예컨대, 한 개의 실수 키 값을 5개의 이진수 값으로 변환하는 경우를 가정하면, 종래의 슬라이스드 리컨실리에이션(sliced reconciliation) 방법을 통해 정보량을 최대화 하는 경우를 고려하여 실수 구간을 분할한 결과는 도 4a에 도시된 바와 같다.
- [0035] 이 경우, 이진수 각 자리의 오류율은 도 4b에 도시된 바와 같이, 0, 0.157, 0.221, 0.465 및 0.5가 되기 때문에, 0.2보다 오류율이 큰 0.221, 0.465 및 0.5를 오류율로 갖는 총 3개의 비트가 키로 사용될 수 없게 된다.
- [0036] 반면, 동일한 상황에서, 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치(100) 및 방법을 활용하여 실수 구간을 분할한 결과는 도 5a에 도시된 바와 같다.
- [0037] 본 발명의 실시예에 따른, 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치(100) 및 방법을 활용하는 경우, 오류율이 0, 0.001, 0.091, 0.379 및 0.5 가 되기 때문에, 0.2보다 오류율이 큰 0.379 및 0.5를 오류율로 갖는 총 2개의 비트가 키로 사용될 수 없게 된다.
- [0038] 다시 말해, 본 발명의 실시예에 따른 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치(100) 및 방법을 활용하는 경우, 종래의 슬라이스드 리컨실리에이션(sliced reconciliation) 방법에 비해 더 많은 비트를 키로 활용할 수 있음을 알 수 있다.
- [0039] 이상에서 본 발명에 따른 바람직한 실시예에 대해 설명하였으나, 다양한 형태로 변형이 가능하며, 본 기술분야에서 통상의 지식을 가진 자라면 본 발명의 특허청구범위를 벗어남이 없이 다양한 변형예 및 수정예를 실시할 수 있을 것으로 이해된다.

부호의 설명

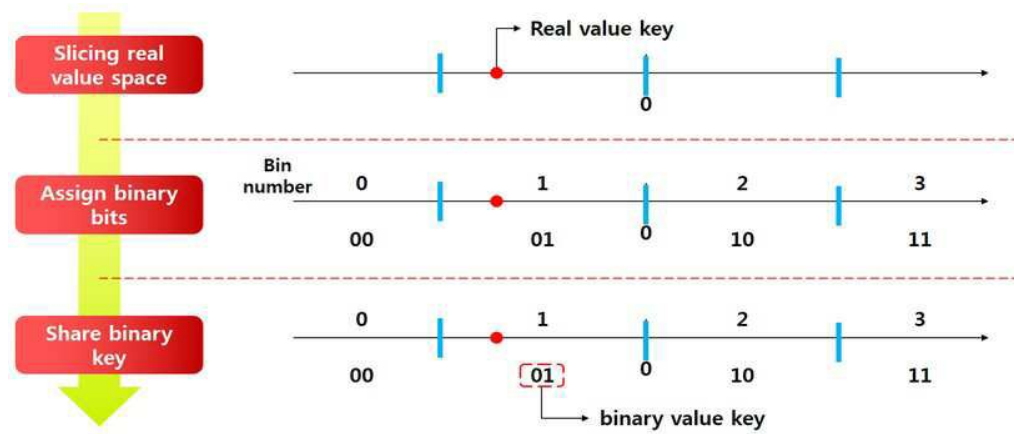
- [0040] 100: 연속 변수 양자 키 분배를 위한 실수 구간 분할 장치
110: 비트 오류 산출부
120: 실수 구간 설정부

도면

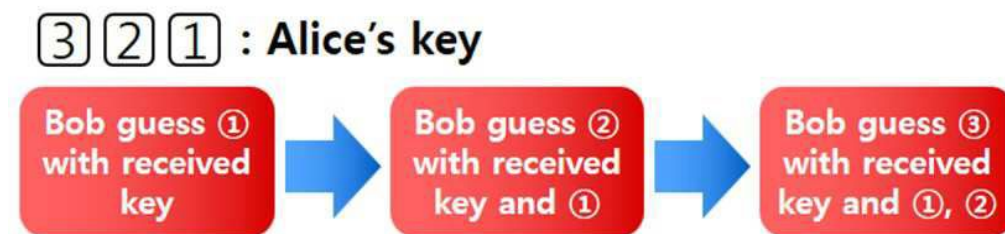
도면1



도면2



도면3



도면4a

τ_{16}	0	$\tau_{21} = -\tau_{11}$	0.725	$\tau_{26} = -\tau_6$	1.56
$\tau_{17} = -\tau_{15}$	0.14	$\tau_{22} = -\tau_{10}$	0.885	$\tau_{27} = -\tau_5$	1.73
$\tau_{18} = -\tau_{14}$	0.28	$\tau_{23} = -\tau_9$	1.05	$\tau_{28} = -\tau_4$	1.9
$\tau_{19} = -\tau_{13}$	0.421	$\tau_{24} = -\tau_8$	1.22	$\tau_{29} = -\tau_3$	2.15
$\tau_{20} = -\tau_{12}$	0.567	$\tau_{25} = -\tau_7$	1.39	$\tau_{30} = -\tau_2$	2.5
				$\tau_{31} = -\tau_1$	3.1

도면4b



도면5a

τ_{16}	0	$\tau_{21} = -\tau_{11}$	1.05	$\tau_{26} = -\tau_6$	2.5
$\tau_{17} = -\tau_{15}$	0.2	$\tau_{22} = -\tau_{10}$	1.33	$\tau_{27} = -\tau_5$	2.8
$\tau_{18} = -\tau_{14}$	0.4	$\tau_{23} = -\tau_9$	1.6	$\tau_{28} = -\tau_4$	3.3
$\tau_{19} = -\tau_{13}$	0.6	$\tau_{24} = -\tau_8$	2.0	$\tau_{29} = -\tau_3$	3.7
$\tau_{20} = -\tau_{12}$	0.824	$\tau_{25} = -\tau_7$	2.3	$\tau_{30} = -\tau_2$	4.3
				$\tau_{31} = -\tau_1$	4.4

도면5b

