OFF-CHAIN SOL.

CHANNELS

ROUTING

ROUTING

- 만일 A가 B에게 채널을 통해 지불하고자 하면
  - A에서 B에 이르기까지 하나 혹은 그 이상의 개설된 채널의 경로를 찾을 필요가 있음

ROUTING

- 만일 지불이 오직 하나의 경로만을 활용한다면
  - 모든 채널이 충분한 담보를 가질 필요가 있음
- 만일 지불이 여러 경로로 분할된다면
  - 각 경로가 지불의 일부를 다룰 수 있도록 잘 나눠야 함

ROUTING

- 지불 또는 상태 채널의 네트워크에서 경로를 찾는 라우팅 알고리즘이 필요

NETWORK ROUTING ALGORITHM FOR DATA TRANSMISSION

- 데이터 전송 관련 네트워크 라우팅 알고리즘을
  단순하게(naive) 지불 채널 네트워크에 적용하면 고유한 문제들이 발생
  - 데이터 라우팅 알고리즘의 목표는 데이터를 한 노드에서 다른 노드로 전송하는 것
  - 라우팅 알고리즘은 정보를 전달함으로써 노드의 상태를 변경

NETWORK ROUTING ALGORITHM FOR DATA TRANSMISSION

- 데이터 전송 관련 네트워크 라우팅 알고리즘을
  단순하게(naive) 지불 채널 네트워크에 적용하면 고유한 문제들이 발생
  - 데이터 네트워크에서 노드 연결 및 대역폭 용량은 개인 정보로 간주되지 않음
  - 데이터 재전송은 TCP 등 프로토콜의 고유한 특징
    - 일반적으로 송신자나 수신자에게 경제적 손실을 유발하지 않음

PAYMENT CHANNEL ROUTING ALGORITHM

- 대조적으로, 지불 채널 라우팅 알고리즘은 다음과 같은 특징을 가짐
  - 지불 채널 라우팅 알고리즘의 목표는 횡단하는 채널의 상태를 변경함으로써 안전하게 자산을 송신자로부터 수신자에게 전달하는 것
  - 트랜잭션의 값에 따라, 어떤 채널들은 경로에 적합하지 않을 수 있음
    - 따라서 채널 잔액은 라우팅 알고리즘이 고려해야 하는 장애물이 됨

PAYMENT CHANNEL ROUTING ALGORITHM

- 대조적으로, 지불 채널 라우팅 알고리즘은 다음과 같은 특징을 가짐
  - 채널 트랜잭션이 실행되면 영구적으로 경로 상의 모든 채널의 상태가 변경됨
  - 대역폭 및 네트워크 지연과 같은 추가적인 매개변수들
    - 채널 경로의 지연 특성에 영향을 미침

PAYMENT CHANNEL ROUTING ALGORITHM

- 대조적으로, 지불 채널 라우팅 알고리즘은 다음과 같은 특징을 가짐
  - 사용자 프라이버시를 보호하기 위해 채널의 전체 용량만이 공개
    - 두 채널 참여자의 자금은 공개되지 않음
  - 채널 트랜잭션은 실패할 수 있음
    - 라우팅 알고리즘은 한 번의 성공이 있기까지 서로 다른 경로를 시도

DATA VS. PCN ROUTING

	데이터 라우팅	지불 채널 라우팅
목표	데이터를 한 노드에서 다른 노드로 전송	송신자로부터 수신자에게 자산을 안전하게 전달
고려사항	대역폭 및 네트워크 지연	대역폭 및 네트워크 지연 + 트랜잭션 수량이 경로 상 한 채널의 용량을 넘을 수 없음
재전송	송신자나 수신자에게 경제적 손실을 유발하지 않음	한 번의 성공이 있기까지 여러 경로를 시도
상태	정보 전달로 노드의 상태를 변경	경로 상의 모든 채널의 상태를 영구적으로 변경
프라이버시	노드 연결 및 대역폭은 개인 정보로 간주되지 않음	채널의 전체 용량만이 공개, 참여자의 자금 분포는 공개되지 않음

ROUTING

- PCN 라우팅 알고리즘은 만족스러운 경로 추천을 제공하기 위해
  - 채널의 고유한 특성을 고려해야 함
    - 채널의 용량 등
  - 무작위 경로 선택과 같은 Tor식 라우팅 방식은 적절하지 않음

ROUTING

- 안전한 지불 채널을 위한 라우팅 알고리즘의 다섯 속성
- 유효성(Effectiveness)
  - 주어진 PCN 스냅샷(snapshot)과 채널 잔액에 대해,
    알고리즘은 지불 성공 확률을 최대로 하는 경로를 찾아야 함
  - 알고리즘은 채널 잔액이 변하더라도 유효성을 유지해야 함

ROUTING

- 안전한 지불 채널을 위한 라우팅 알고리즘의 다섯 속성
- 효율성(Efficiency)
  - 경로 탐색의 오버헤드는 지연(latency), 통신, 연산에서 낮아야 함
  - PCN 토폴로지(topology)의 변경에는
    낮은 업데이트 오버헤드 비용이 수반되어야 함

ROUTING

- 안전한 지불 채널을 위한 라우팅 알고리즘의 다섯 속성
- 확장성(Scalability)
  - 라우팅 알고리즘은 큰 규모의 PCN과 높은 트랜잭션 빈도에서도 유효하고 효율적이어야 함

ROUTING

- 안전한 지불 채널을 위한 라우팅 알고리즘의 다섯 속성
- 비용-유효성(Cost-Effectiveness)
  - 알고리즘은 낮은 트랜잭션 수수료를 가진 경로를 찾아야 함
  - 레이어-2 트랜잭션의 수수료는 레이어-1 트랜잭션보다 낮아야 함

ROUTING

- 안전한 지불 채널을 위한 라우팅 알고리즘의 다섯 속성
- 프라이버시(Privacy)
  - 라우팅 경로는 트랜잭션 값을 공개하지 않고 찾아질 수 있어야 함
    - 값 프라이버시
  - 거래 당사자를 공개하지 않고서도 찾아질 수 있어야 함
    - 송신자 및 수신자 프라이버시

ROUTING

- 라우팅 알고리즘을 두 부류로 분류
  - 글로벌(global) 라우팅
  - 로컬(local) 라우팅

ROUTING

- 글로벌 라우팅 또는 소스(source) 라우팅
  - 각 노드는 전체 PCN 토폴로지의 지역적 스냅샷을 유지
- 로컬 라우팅
  - 알고리즘은 지역적 정보만으로 연산
  - 알고리즘은 노드와 채널을 형성한 이웃 정보만을 알 수 있음

GLOBAL VIEW

GLOBAL VIEW

- 라이트닝 및 라이덴(Raiden)
  - 지불의 출처(소스)가 지불의 전체 경로를 지정
    - 소스 라우팅

GLOBAL VIEW

- 만일 모든 노드의 글로벌 뷰가 정확하다면
  - 소스 라우팅은 노드 간 모든 경로를 탐색
  - 높은 유효성

GLOBAL VIEW

- 기본 소스 라우팅은 채널 잔액을 고려하지 않음
  - 라우팅 경로 선택에 낮은 잔액 혹은
  - 내재적으로 단방향으로 바뀐 채널을 선택하게 될 수도 있음
  - 동적 PCN의 경로 가용성을 감소

GLOBAL VIEW

- SpiderNetwork
- 세 가지 핵심 변경을 통해 동적 PCN에서 소스 라우팅의 유효성을 증대
  - 경로 선택은 잔액을 최적화하는 경로에 대한 편향을 포함
  - 라우팅은 잔액을 증대시키기 위해 추가 코인을 예치할 수 있는 온체인 재조정을 포함
  - 라우팅은 패킷 교환 방식(packet-switch)의 네트워크에 의존

GLOBAL VIEW

- 라우팅은 패킷 교환 방식(packet-switch)의 네트워크에 의존
  - 지불 그 자체를 완전히 라우팅하는 대신에
  - 알고리즘은 지불을 상수 크기의 단위로 분할
  - 각각을 독립적으로 라우팅

GLOBAL VIEW

- SpiderNetwork
  - 심지어 잔액이 지속적으로 변경되는 환경과
  - 온체인 재조정이 사용되는 높은 지연 상황에서도
  - 높은 유효성을 획득

GLOBAL VIEW

- 글로벌 뷰에 기반한 알고리즘
  - 지역적으로 미리 계산한 경로 덕분에
    낮은 지연과 낮은 통신 오버헤드를 자랑
  - 모든 완전한 스냅샷을 저장하기 위한 로컬 메모리 비용 및
    경로를 찾기 위한 연산 비용이 높음

GLOBAL VIEW

- 글로벌 뷰에 기반한 알고리즘
  - 온체인에 채널을 개설하고 폐쇄하는 것에도 동일한 오버헤드 요구
  - 노드 수에 따라 오버헤드가 초선형(super-linear)으로 증가
    - 확장성을 제한

LOCAL VIEW

LOCAL VIEW

- 지역적 정보에 기반한 알고리즘은 잘 연구된 개념들을 이용
  - 분산 해시 테이블(Distributed Hash Table, DHT)
  - 유량(flow) 알고리즘
  - 랜드마크 라우팅
  - 네트워크 임베딩

DISTRIBUTED HASH TABLE

- Flare
- 카뎀리아(Kademlia) DHT를 활용
- 원래 카뎀리아는 전략적으로 선택된 노드 간 새 채널을 개설
  - 이는 지연 및 온체인 수수료 관점에서 비쌈

DISTRIBUTED HASH TABLE

- 수정된 카뎀리아 버전을 사용
  - 직접적인 채널을 새 채널을 개설할 필요가 없는 멀티-홉 경로로 대체
- 단점
  - 전통적인 DHT 대비 더 긴 경로, 더 높은 지연 및 통신 오버헤드를 야기
  - 토폴로지 변경을 지원할 수 없다는 점이 가장 큰 한계

FLOW ALGORITHM

- cRoute
- 최적화 문제는 지역적 혼잡 문제로 공식화
- 혼잡 그레디언트(gradient) 해결책으로 해결 가능
  - 효율적(이라고 주장하지만)
  - 채널 잔액을 유지
- 관련 연구들에서 효율성과 확장성에 문제가 있음을 지적받음

LANDMARK ROUTING

- SilentWhispers
- 랜드마크: 헌신적인 노드를 의미
- 각 노드는 모든 랜드마크에 다다르기 위해 이웃과의 접촉을 추적

LANDMARK ROUTING

- 두 노드 사이의 지불
  - 우선 송신자에서 랜드마크로의 경로를 횡단
  - 랜드마크에서 수신자로의 경로를 횡단

LANDMARK ROUTING

- 여러 랜드마크를 조합해서 사용하면
  - 프라이버시-보존의 의미에서 지불을 나누어 여러 경로로 전송하는 것이 가능
- 각 노드는 토폴로지 변화에 대응하기 위해
  - 주기적으로 랜드마크에 다다르는 방법을 재계산
  - 데이트의 비용이 소스 라우팅보다 낮음

LANDMARK ROUTING

- 실제 데이터셋을 대상으로 진행한 SilentWhispers의 평가
  - 다른 알고리즘에 비해 낮은 유효성과 적당한 지연을 가짐
  - 트랜잭션 수의 증가에 따라 연산 비용의 증대와 낮은 확장성을 보임

NETWORK EMBEDDING

- SpeedyMurmurs
- SilentWhispers의 한계를 극복하고자 등장
- 임베딩-기반 라우팅과 지역적 토폴로지 업데이트를 처리하기 위한 프로토콜을 사용

NETWORK EMBEDDING

- 노드는 좌표를 통해 신장 트리(spanning tree)에서의 자신의 위치를 표시
- 모든 충분한 잔액을 가진 인접 채널들을 고려
  - 지불 경로 상의 다음 노드를 지역적으로 선택
  - 수신자의 좌표로부터 가장 가까운 노드에 대한 채널을 선택
  - 경로는 신장 트리의 일부가 아닌 채널을 포함할 수 있음

NETWORK EMBEDDING

- 정적 PCN에 대한 높은 유효성과 낮은 지연
- 채널이 개설되거나 폐쇄되는 경우
  - 오직 관련된 부분 트리의 좌표만이 조정
  - 일반적으로 네트워크 사이즈에 대해 로그 스케일의 오버헤드

NETWORK EMBEDDING

- 지불 중에 자금을 차단하기 때문에 트랜잭션 빈도가 영향을 받을 수 있음
- 잔액을 고려하지 않음
  - 동적 PCN에서 충분하지 않은 잔액으로 인한 낮은 효율성을 초래

SUMMARY

- 멀티-홉(Multi-hop) 지불을 위한 라우팅 알고리즘

		Global View		Local View			
		Lightning [27]/Raiden [29]	SpiderNetwork [34]	Flare [37]	cRoute [82]	SilentWhispers [32]	SpeedyMurmurs [33]
Effectiveness	Snapshot	High	High	High	High	Medium	High
	Dynamic	Medium	High	NA[1]	High	Low	Low
Efficiency	Latency	Low	Low/High[2]	High	High	High	Low
	Communication	Low	Low/High[2]	High	High	High	Low
	Computation	High	High	Low	Low	High	Low
	Update	High	High	NA[1]	Low	High	Low
Scalability	Nodes	Low	Low	NA[1]	?[3]	High	High
	Transactions	High	High	NA[1]	High	Low	Low
Cost-Effectiveness	Considered	✓	×	×	×	×	×
Privacy	Guarantees	×	×	×	×	✓	✓

[1] Dynamic behavior not specified. [2] High for on-chain re-balancing, otherwise low. [3] No evaluation, only tested for 77 nodes.

SUMMARY

- 오직 한 종류의 알고리즘만이 비용-유효성을 고려 (Lightning/Raiden)

		Global View		Local View			
		Lightning [27]/Raiden [29]	SpiderNetwork [34]	Flare [37]	cRoute [82]	SilentWhispers [32]	SpeedyMurmurs [33]
Effectiveness	Snapshot	High	High	High	High	Medium	High
	Dynamic	Medium	High	NA[1]	High	Low	Low
Efficiency	Latency	Low	Low/High[2]	High	High	High	Low
	Communication	Low	Low/High[2]	High	High	High	Low
	Computation	High	High	Low	Low	High	Low
	Update	High	High	NA[1]	Low	High	Low
Scalability	Nodes	Low	Low	NA[1]	?[3]	High	High
	Transactions	High	High	NA[1]	High	Low	Low
Cost-Effectiveness	Considered	✓	×	×	×	×	×
Privacy	Guarantees	×	×	×	×	✓	✓

[1] Dynamic behavior not specified. [2] High for on-chain re-balancing, otherwise low. [3] No evaluation, only tested for 77 nodes.

SUMMARY

- 일부는 동적 상황을 고려하지 않았으며 (윗첨자 1)

		Global View		Local View			
		Lightning [27]/Raiden [29]	SpiderNetwork [34]	Flare [37]	cRoute [82]	SilentWhispers [32]	SpeedyMurmurs [33]
Effectiveness	Snapshot	High	High	High	High	Medium	High
	Dynamic	Medium	High	NA[1]	High	Low	Low
Efficiency	Latency	Low	Low/High[2]	High	High	High	Low
	Communication	Low	Low/High[2]	High	High	High	Low
	Computation	High	High	Low	Low	High	Low
	Update	High	High	NA[1]	Low	High	Low
Scalability	Nodes	Low	Low	NA[1]	?[3]	High	High
	Transactions	High	High	NA[1]	High	Low	Low
Cost-Effectiveness	Considered	✓	×	×	×	×	×
Privacy	Guarantees	×	×	×	×	✓	✓

[1] Dynamic behavior not specified. [2] High for on-chain re-balancing, otherwise low. [3] No evaluation, only tested for 77 nodes.

SUMMARY

- Spider Network의 경우 일부 효율성이 온체인 재조정에서는 높고, 다른 측면에서는 낮음

		Global View		Local View			
		Lightning [27]/Raiden [29]	SpiderNetwork [34]	Flare [37]	cRoute [82]	SilentWhispers [32]	SpeedyMurmurs [33]
Effectiveness	Snapshot	High	High	High	High	Medium	High
	Dynamic	Medium	High	NA[1]	High	Low	Low
Efficiency	Latency	Low	Low/High[2]	High	High	High	Low
	Communication	Low	Low/High[2]	High	High	High	Low
	Computation	High	High	Low	Low	High	Low
	Update	High	High	NA[1]	Low	High	Low
Scalability	Nodes	Low	Low	NA[1]	?[3]	High	High
	Transactions	High	High	NA[1]	High	Low	Low
Cost-Effectiveness	Considered	✓	×	×	×	×	×
Privacy	Guarantees	×	×	×	×	✓	✓

[1] Dynamic behavior not specified. [2] High for on-chain re-balancing, otherwise low. [3] No evaluation, only tested for 77 nodes.

SUMMARY

- cRoute는 확장성에 대한 평가가 없고, 오직 77개 노드에서만 테스트 됨

		Global View		Local View			
		Lightning [27]/Raiden [29]	SpiderNetwork [34]	Flare [37]	cRoute [82]	SilentWhispers [32]	SpeedyMurmurs [33]
Effectiveness	Snapshot	High	High	High	High	Medium	High
	Dynamic	Medium	High	NA[1]	High	Low	Low
Efficiency	Latency	Low	Low/High[2]	High	High	High	Low
	Communication	Low	Low/High[2]	High	High	High	Low
	Computation	High	High	Low	Low	High	Low
	Update	High	High	NA[1]	Low	High	Low
Scalability	Nodes	Low	Low	NA[1]	?[3]	High	High
	Transactions	High	High	NA[1]	High	Low	Low
Cost-Effectiveness	Considered	✓	×	×	×	×	×
Privacy	Guarantees	×	×	×	×	✓	✓

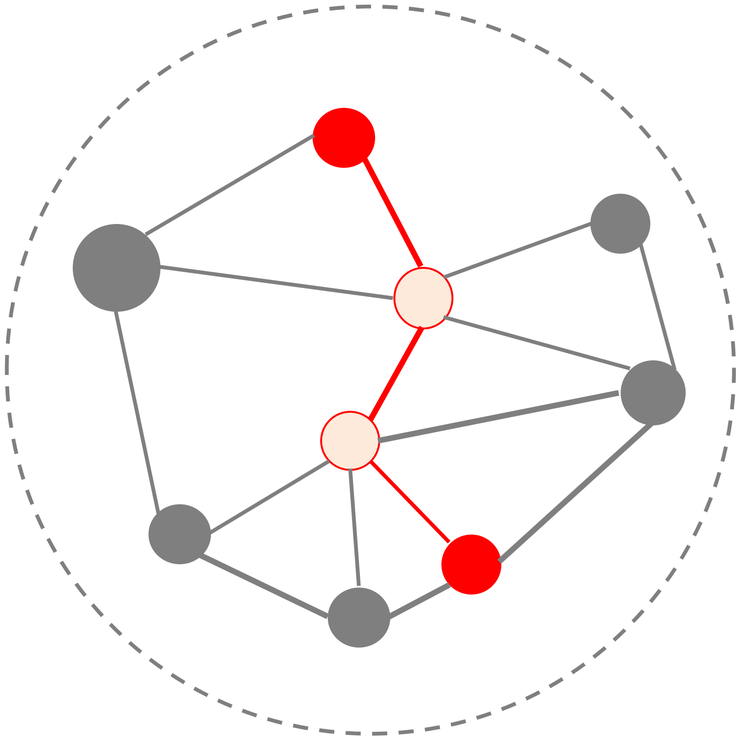
[1] Dynamic behavior not specified. [2] High for on-chain re-balancing, otherwise low. [3] No evaluation, only tested for 77 nodes.

CHANNEL HUB

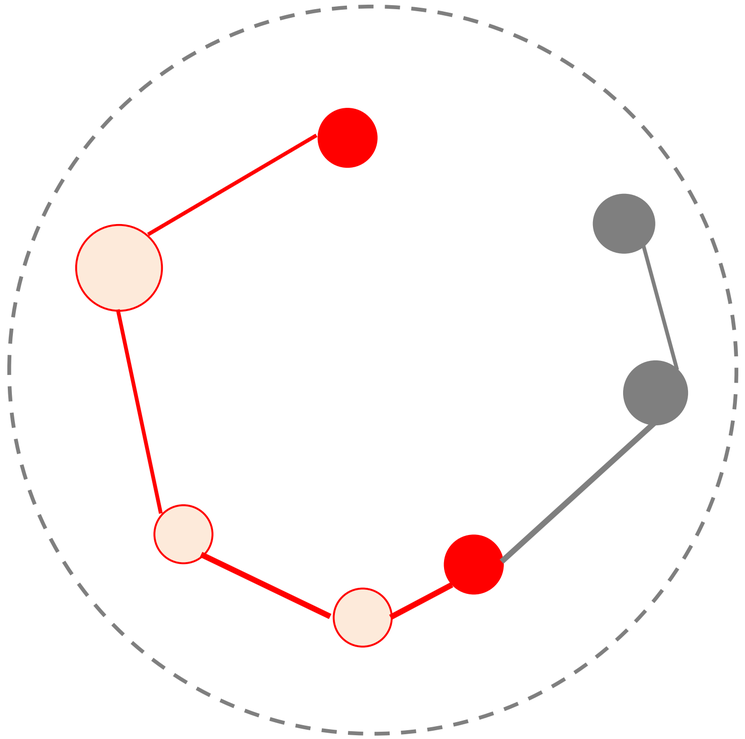
CHANNEL HUB

- 담보 잠금 비용을 줄이고 라우팅 복잡도를 줄이기 위해
  - 중앙화된(그렇지만 비구금형) 별모양 토폴로지의 수혜를 받음
- 별모양의 중심: 지불 채널 허브(Payment Channel Hub, PCH)
  - 본질적으로 다른 피어와의 많은 채널을 유지하는 PCN에서의 노드
  - 여러 상호연결된 PCH를 가진 네트워크는 평균적으로 낮은 경로 길이를 가짐
  - 줄어든 경로 길이는 담보 비용의 및 경로 탐색의 복잡도 감소를 수반

CHANNEL HUB



지불 채널 허브를 통한 라우팅



지불 채널 허브가 없는 상황

CHANNEL HUB

- PCH는 각 채널에 대해 상당한 자금 잠금을 요구
- 가정: 1M 채널을 가진 PCN, 각 채널은 트랜잭션의 값으로 평균 $1000를 전송
  - 허브는 총 $1B의 잠금이 필요
  - 재조정 작업은 비용이 많이 들고 느린 부모 체인 트랜잭션을 통해서만 가능
- 사용자 등록 역시 비용이 많이 드는 절차
  - 1M 사용자를 가진 PCN 노드는 1M 부모 체인 설정 트랜잭션을 요구
  - 이더리움에서 $100k 이상 비용이 소모됨

SUMMARY

SUMMARY

- 블록체인은 레이어-1의 변경 없이도 레이어-2 기술을 활용해 확장성 증가가 가능
- PCN 역시 한계를 가지고 있음
  - 아직 얼마나 확장 가능한지 정확하게 정량화되지 않음

SUMMARY

- 모든 레이어-2 프로토콜은 다음 측면에서 확장성을 고려해야 함:
  - 노드의 수
  - 네트워크에서 채널의 총 개수
  - 노드 당 채널 수 분포
  - 네트워크 내 누적 트랜잭션의 수
  - 각 트랜잭션의 값 분포

SUMMARY

- 레이어-1 트랜잭션 비용
  - 크기(UTXO 블록체인) 또는
  - 연산 복잡도(스마트 컨트랙트 블록체인)로 요약
- 레이어-2 트랜잭션 비용
  - 재화의 가치($)와 관련이 있음
  - 더 높은 레이어-2 트랜잭션의 값은 더 많은 온체인 담보의 예치를 요구
  - 잠재적으로 상당한 양의 자금이 잠길 것으로 예상 가능

OFF-CHAIN SOL.

CHANNELS