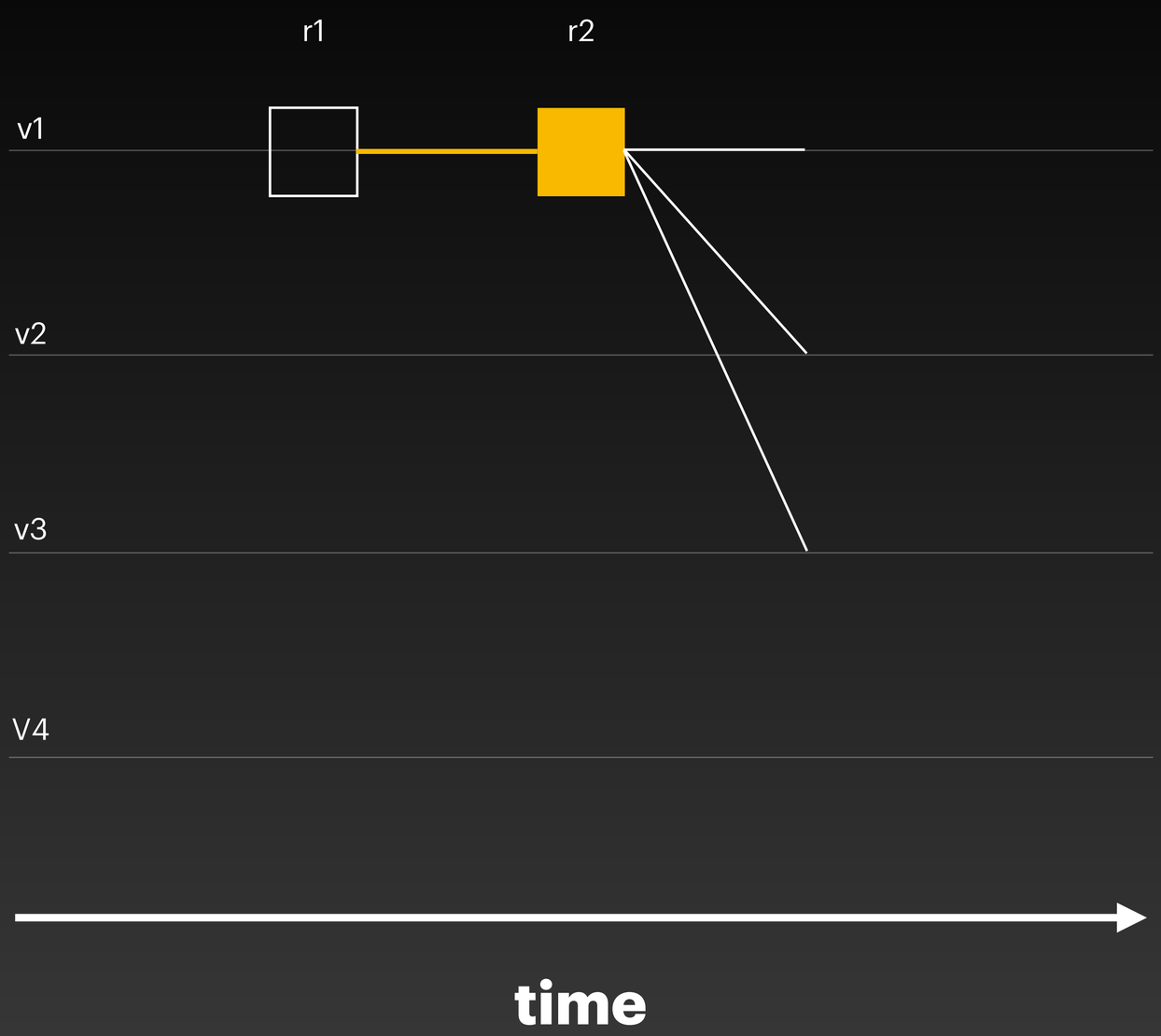
DAG-Based Consensus

Introduction and practical Aspects

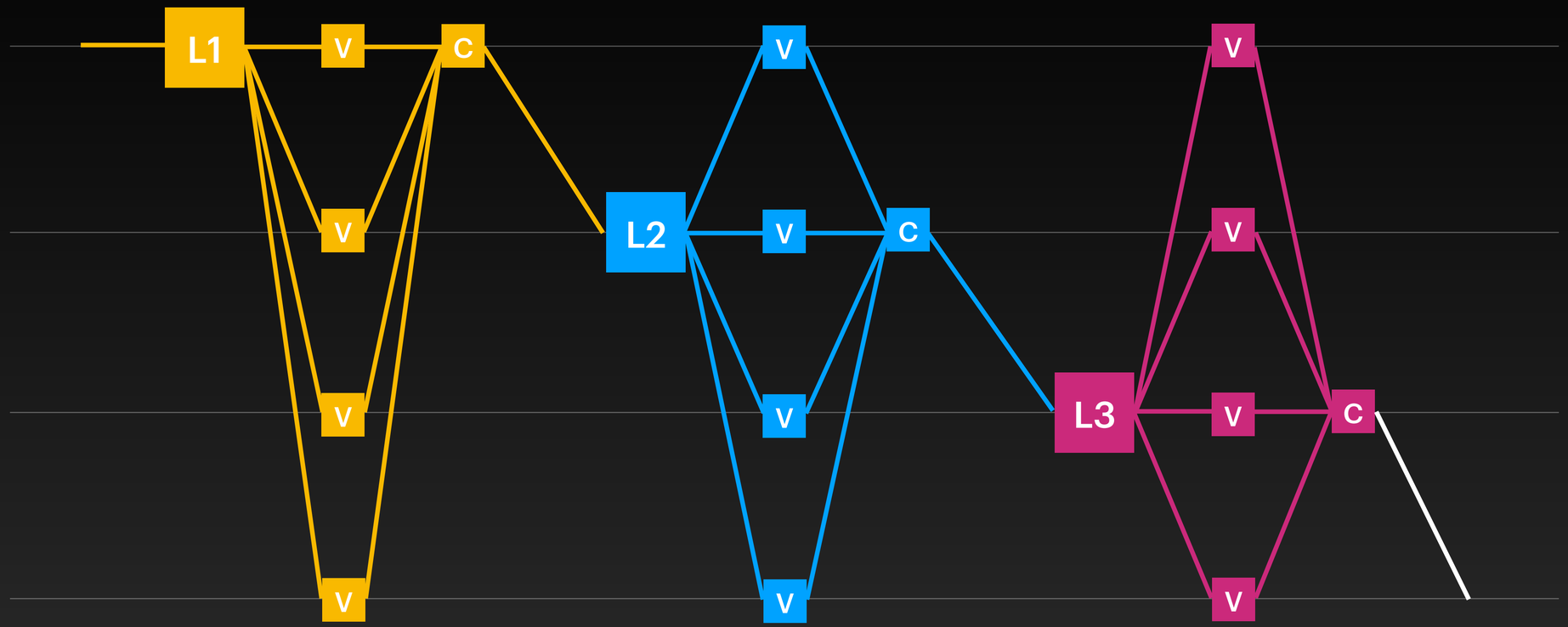
Alberto Sonnino

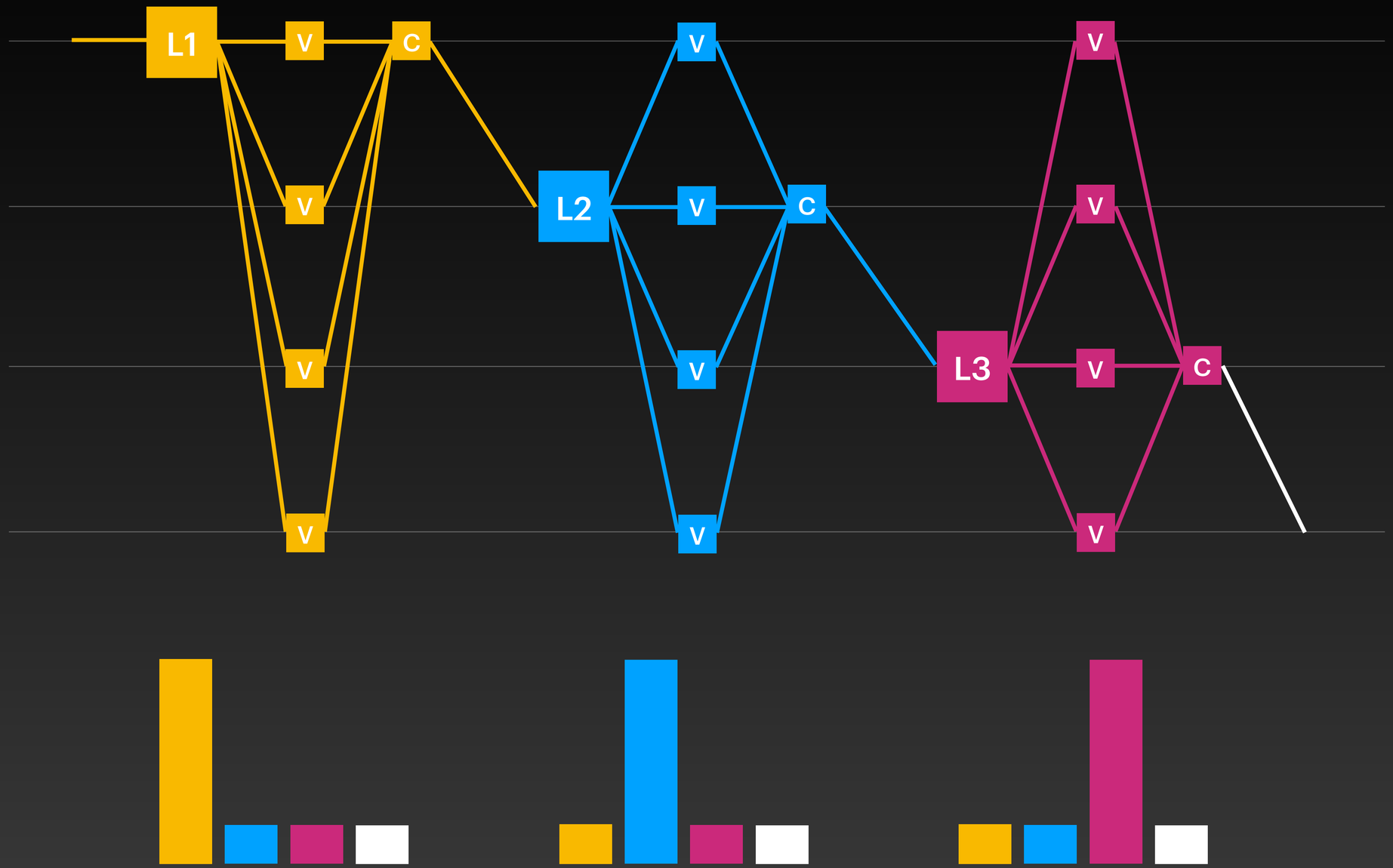
> 2/3

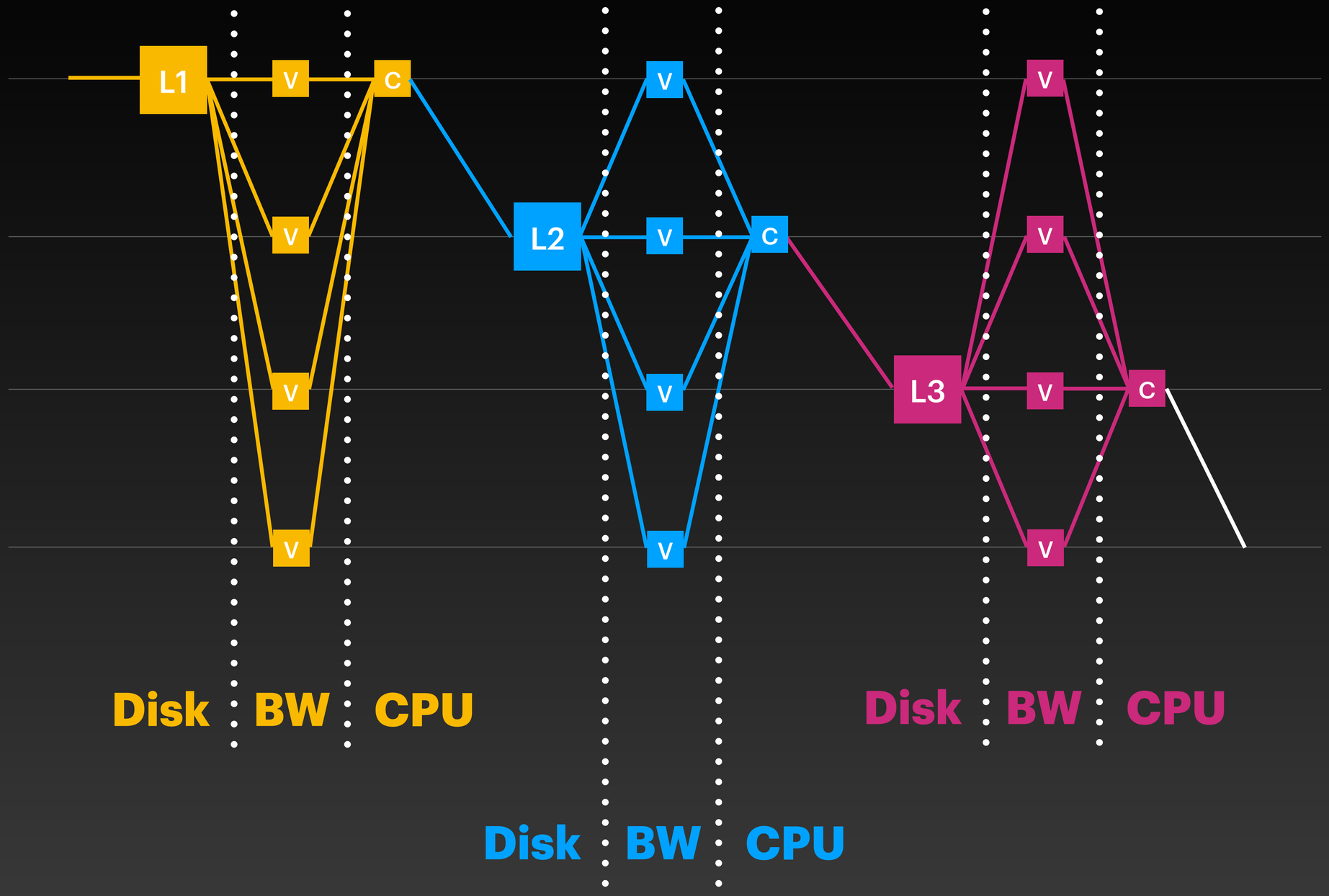
Partial Synchrony

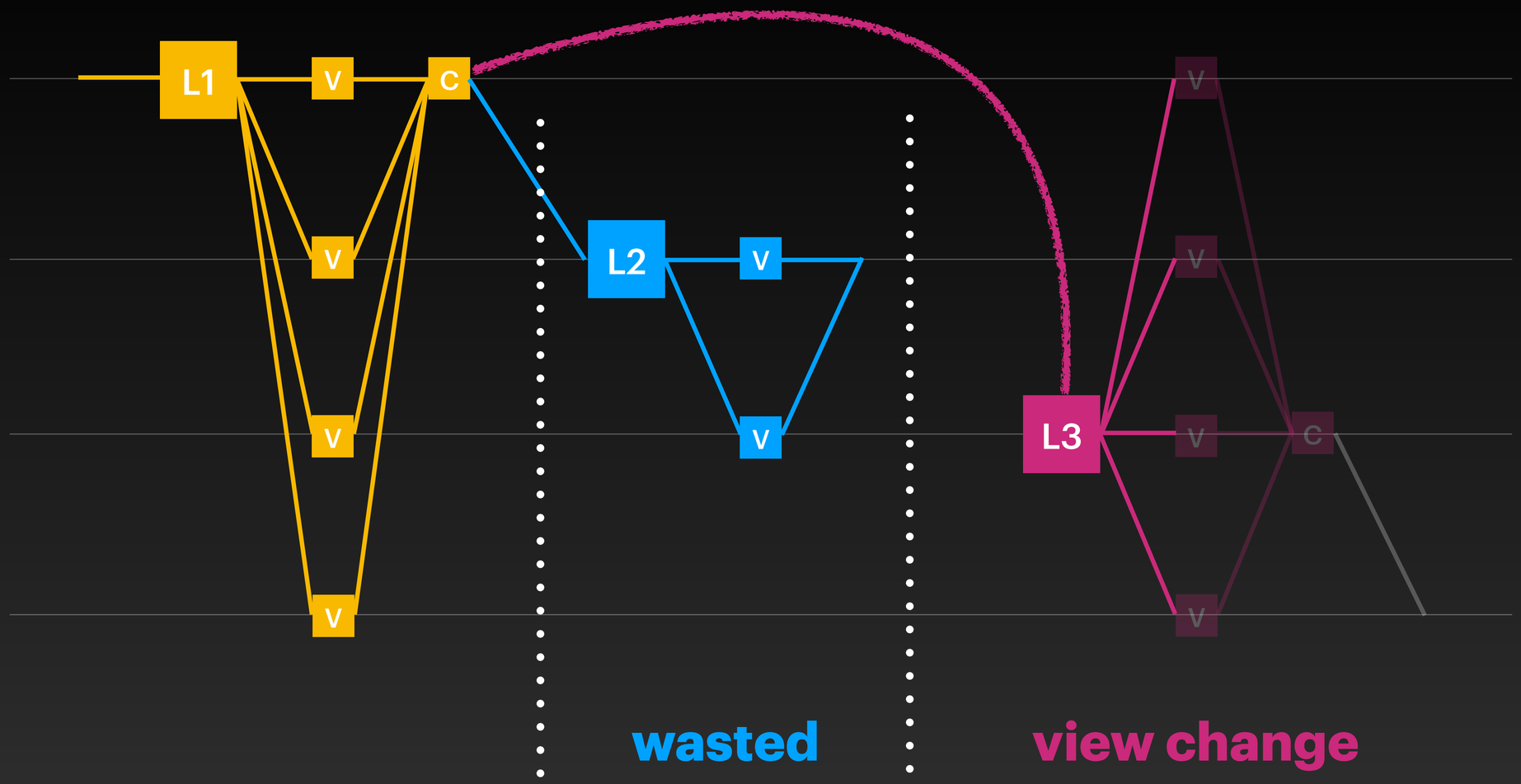


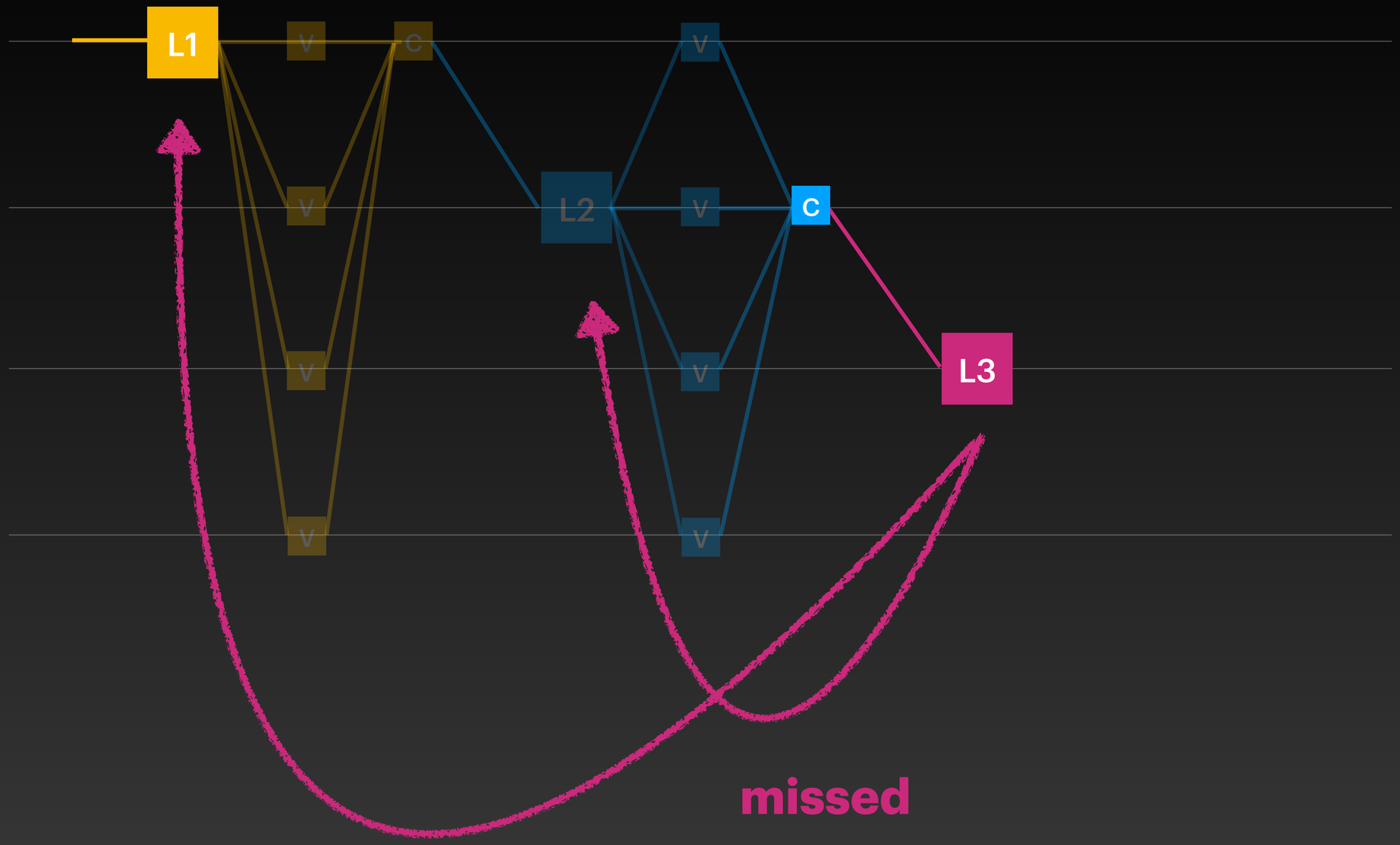
- Round number
- Author
- Payload (txs)
- Parent
- Signature

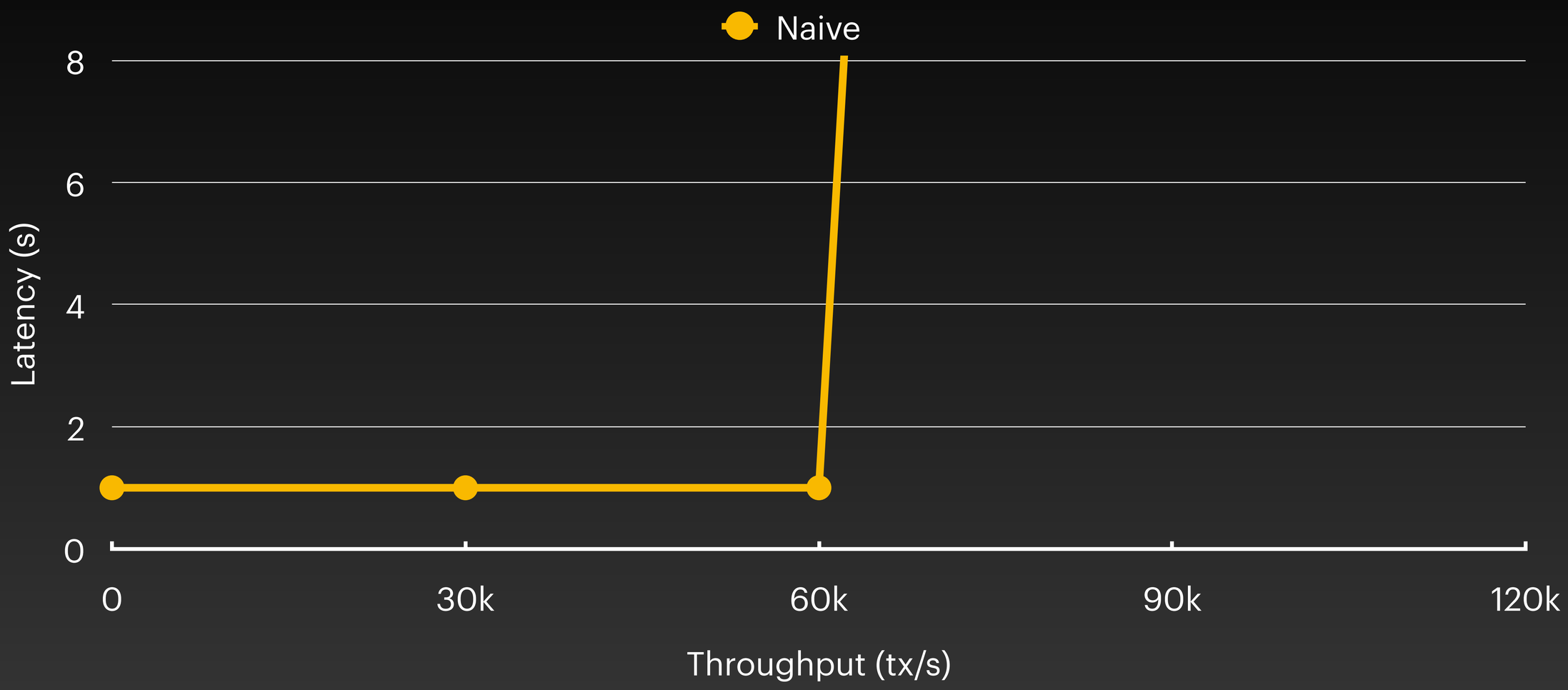

L1
V
V
V
V
C
L2
V
V
V
V
C
L3
V
V
V
V
C

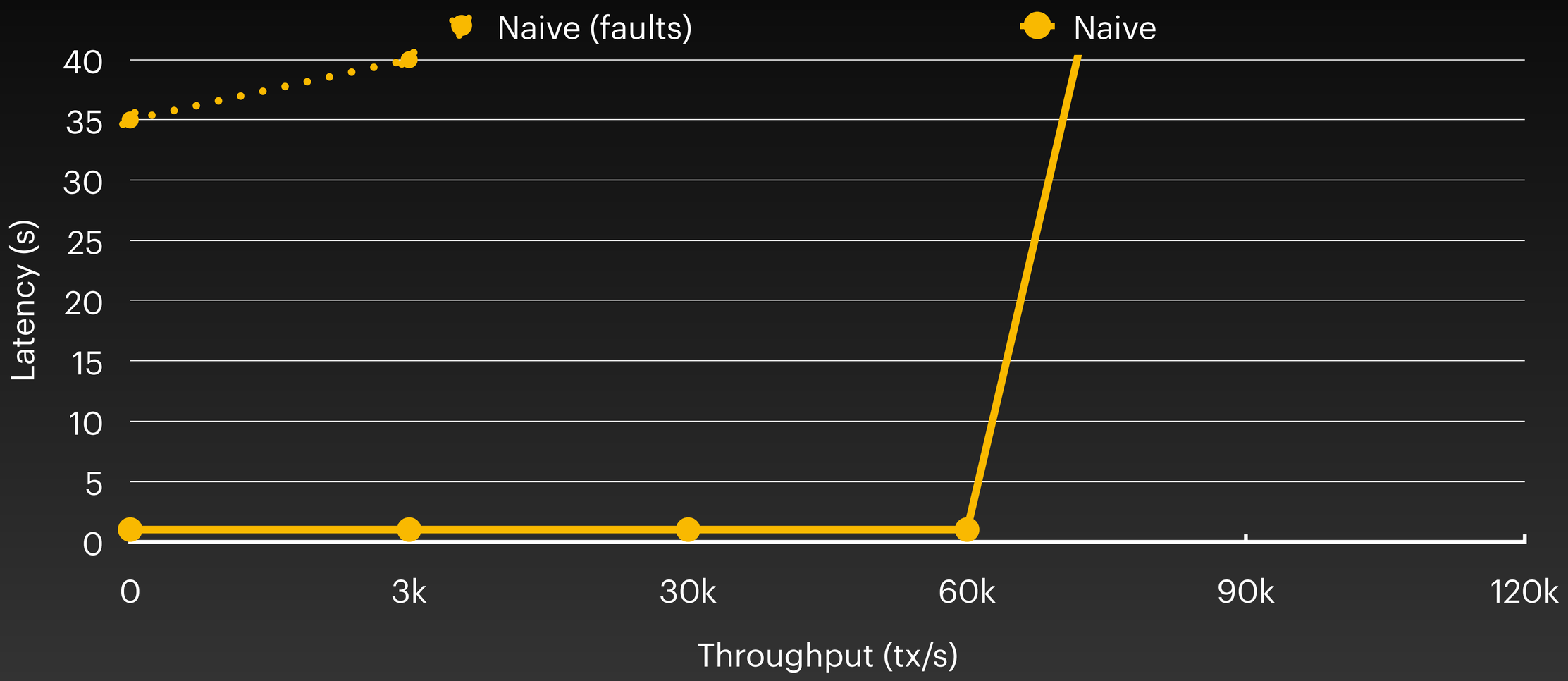

L1
V
V
V
V
C
L2
V
V
V
V
C
L3
V
V
V
V
C

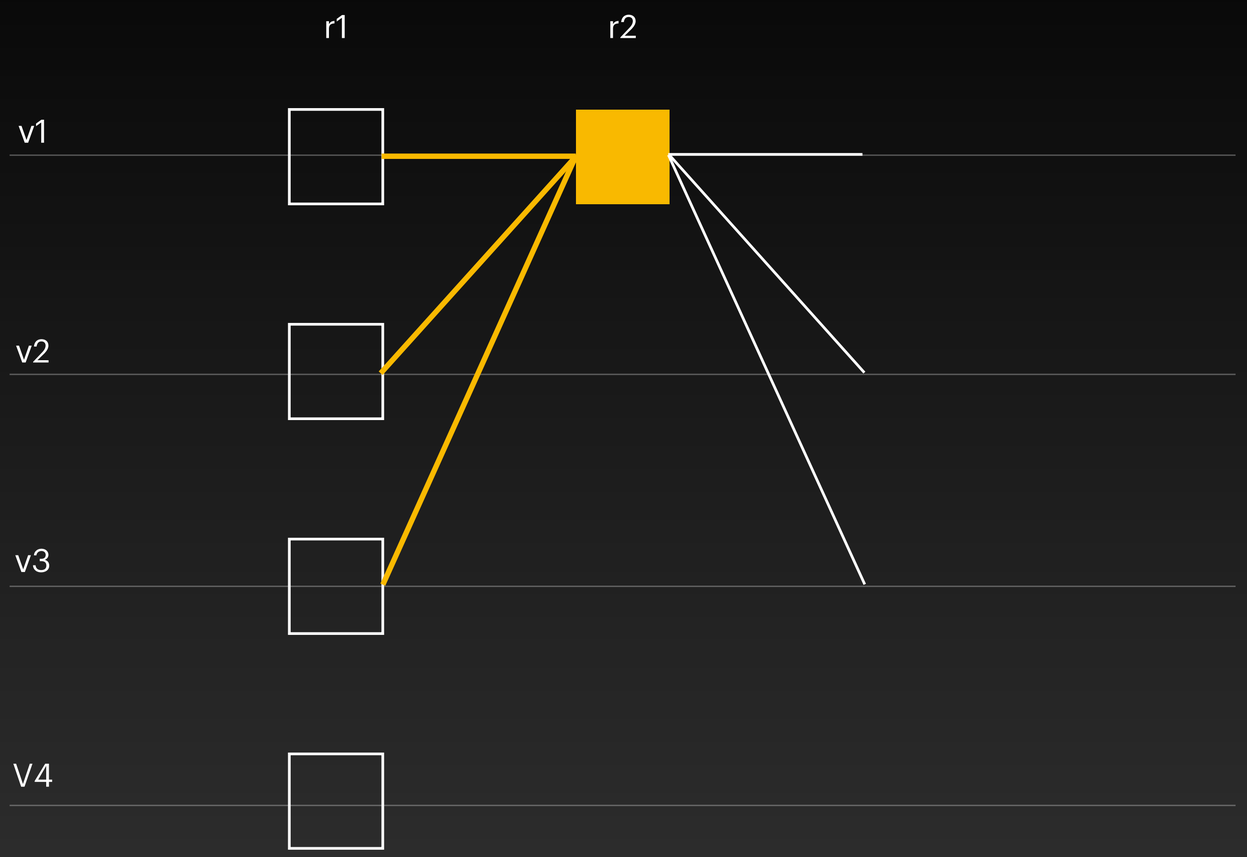

L1
V
V
V
V
C
L2
V
V
V
C
L3
V
V
V
C
Disk
BW
CPU
Disk
BW
CPU
Disk
BW
CPU


L1
V
C
V
V
V
L2
V
V
L3
V
V
C
V
wasted
view change


L1
V
C
V
V
V
V
L2
V
V
V
C
L3
missed

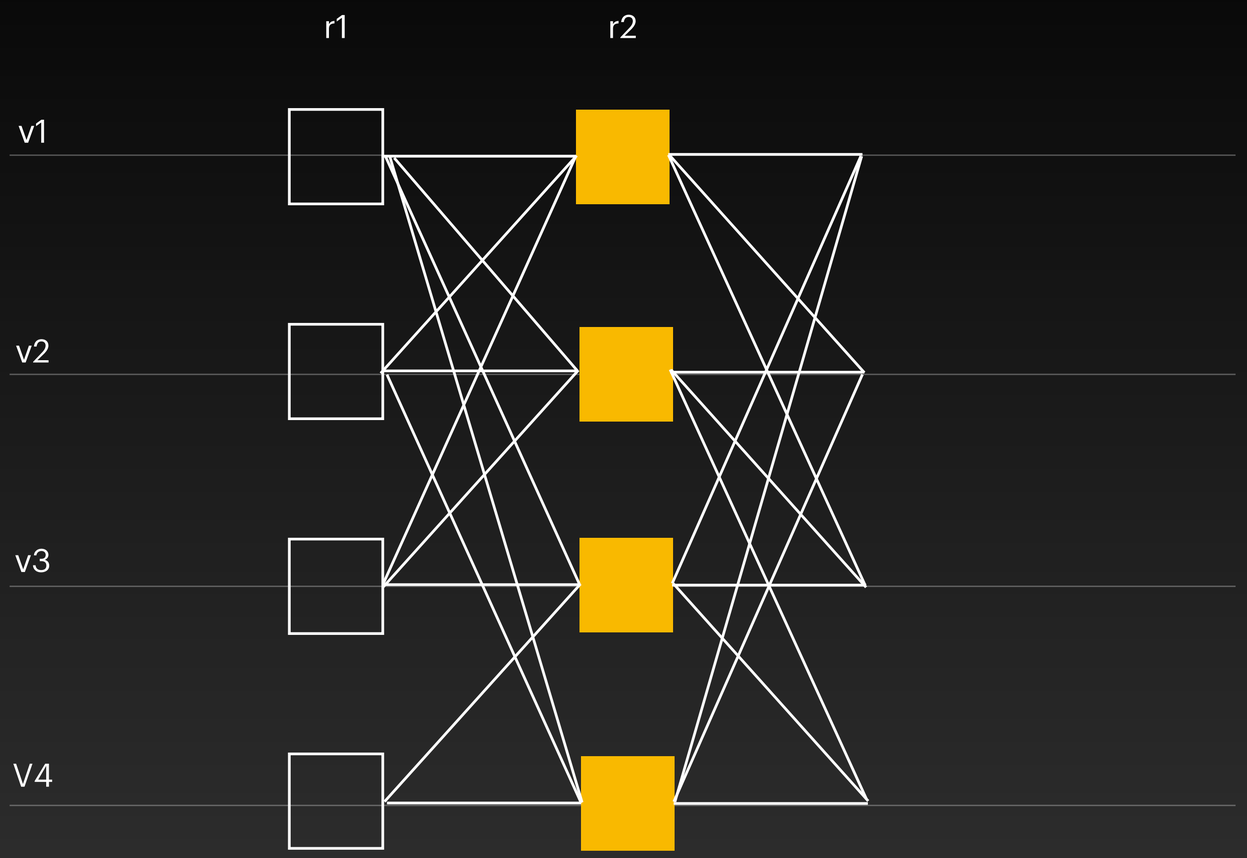

Naive
8
6
4
2
0
Latency (s)
0
30k
60k
90k
120k
Throughput (tx/s)


Naive (faults)
Naive
40
35
30
25
20
15
10
5
0
Latency (s)
0
3k
30k
60k
90k
120k
Throughput (tx/s)



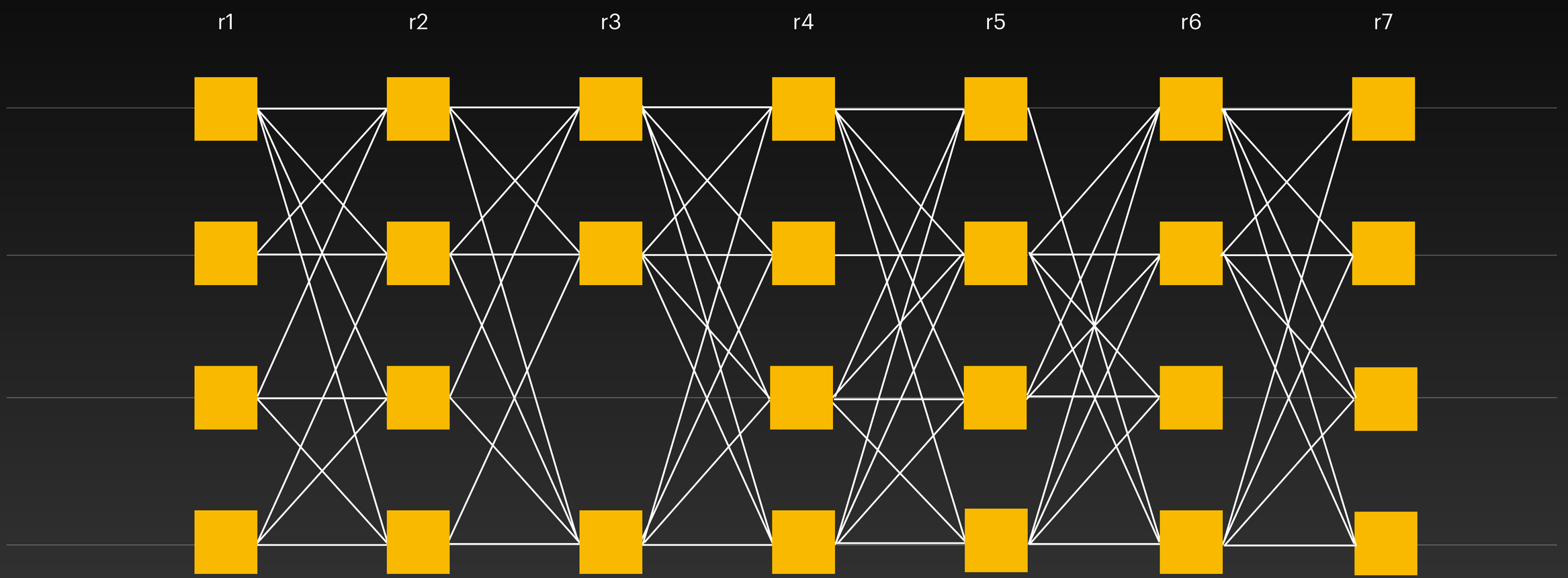
- Round number
- Author
- Payload (txs)
- **Parents**
- Signature

Back Pressure



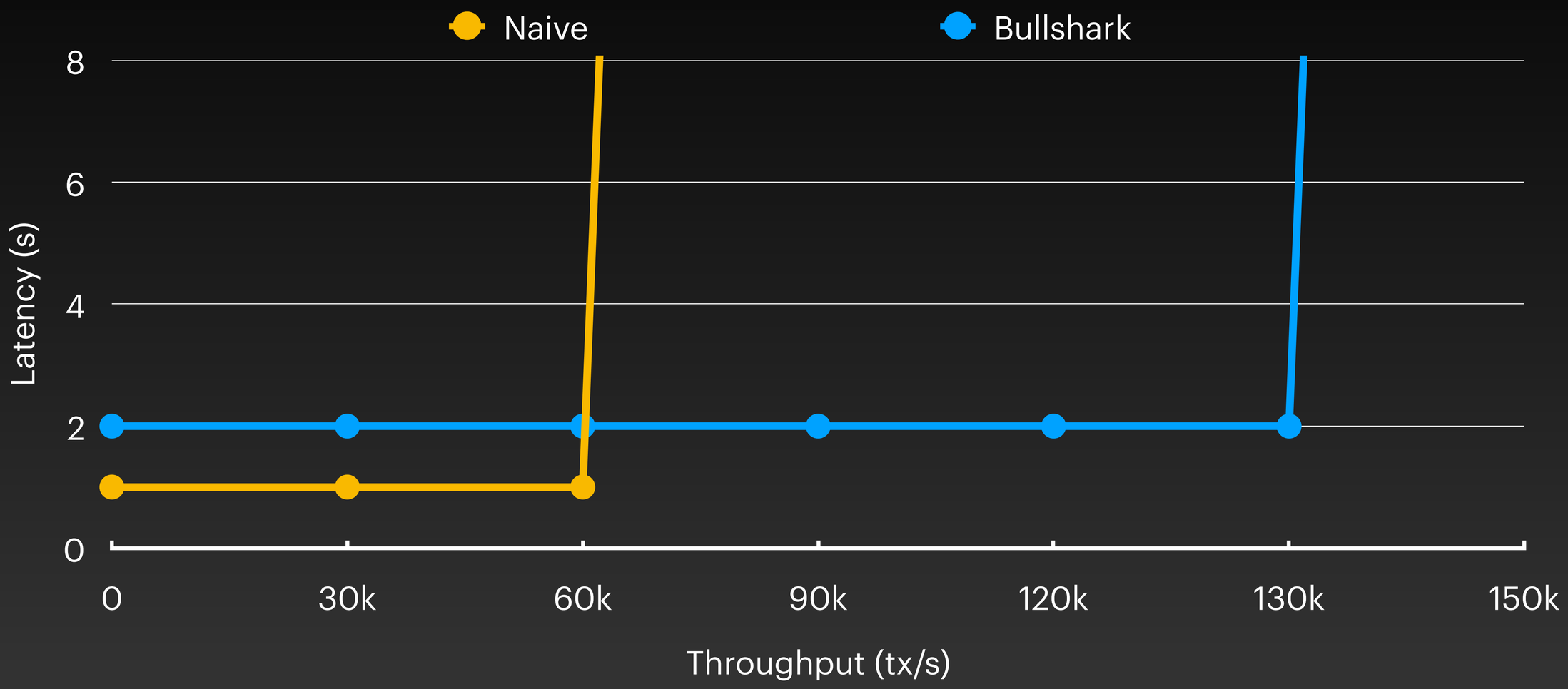
- Round number
- Author
- Payload (txs)
- Parents
- Signature

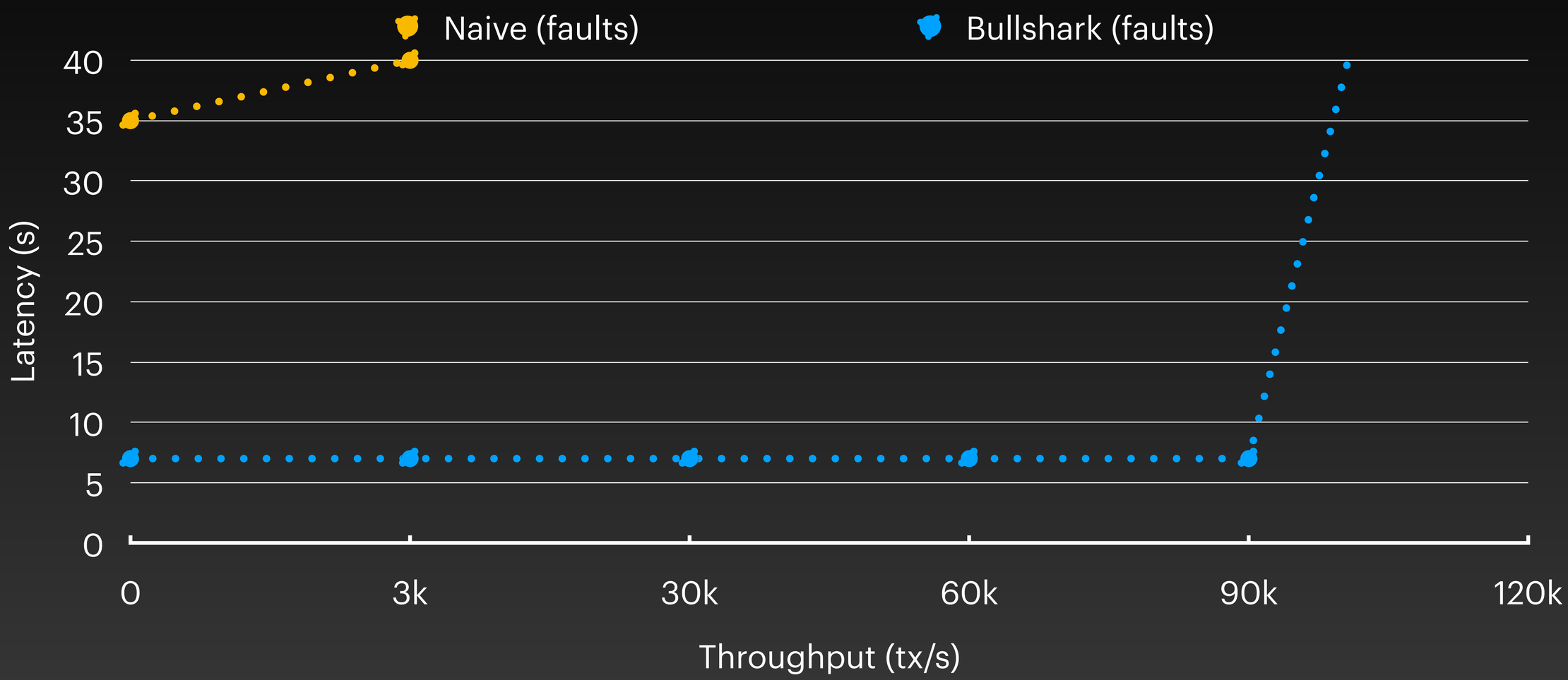
Multi Proposer

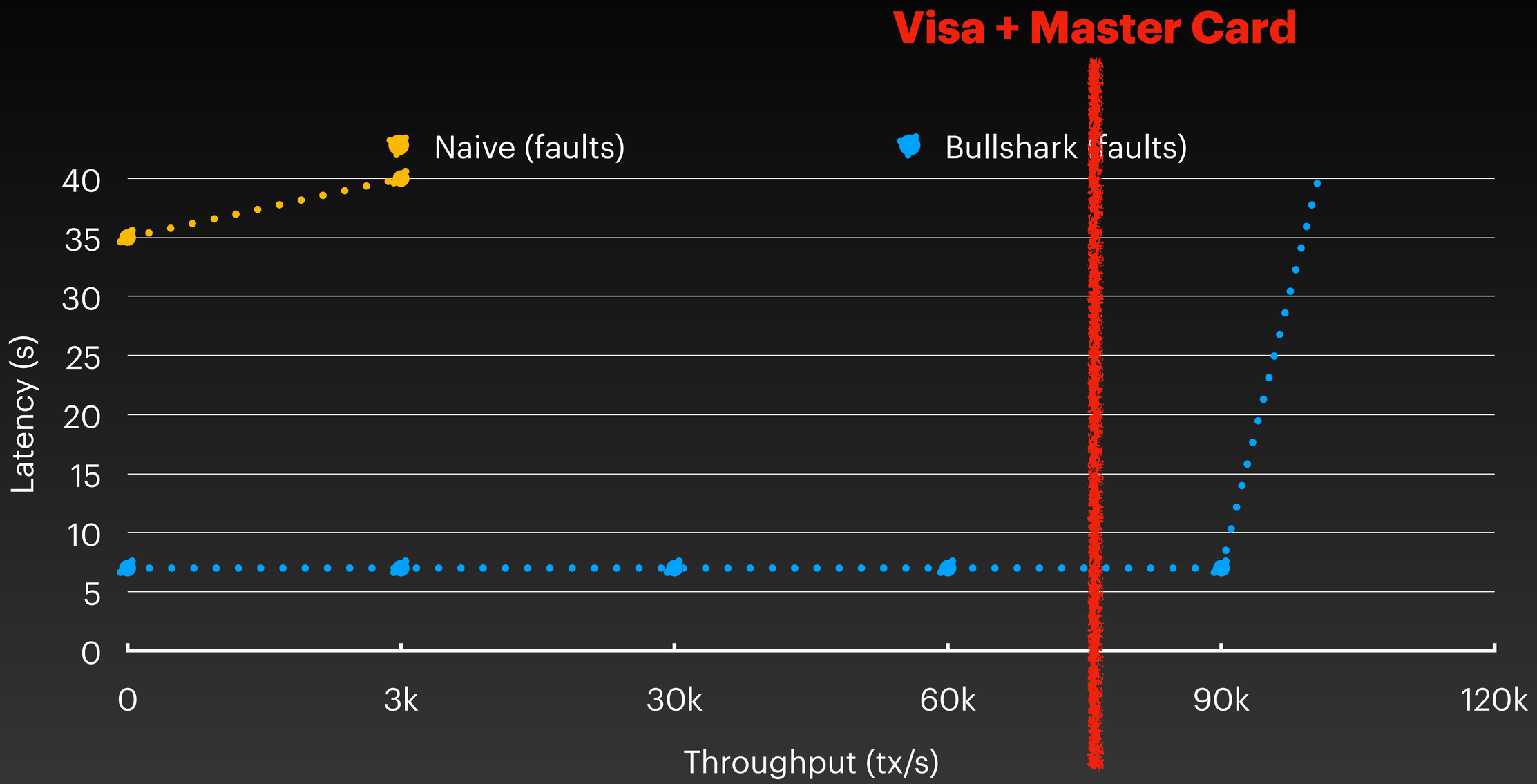

r1
r2
r3
r4
r5
r6
r7

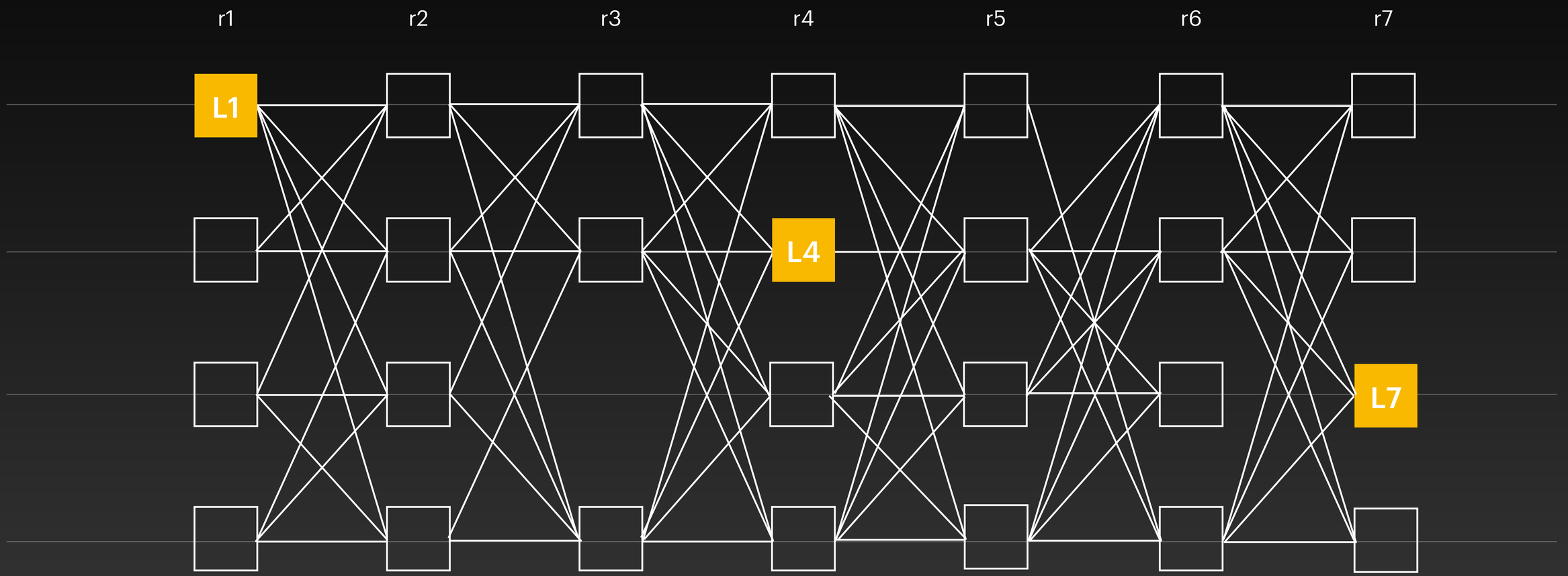
Back Pressure
Robustness

High Resource Utilisation
Throughput


Naive
Bullshark
8
6
4
2
0
Latency (s)
0
30k
60k
90k
120k
130k
150k
Throughput (tx/s)


Naive (faults)
Bullshark (faults)
Latency (s)
40
35
30
25
20
15
10
5
0
0
3k
30k
60k
90k
120k
Throughput (tx/s)

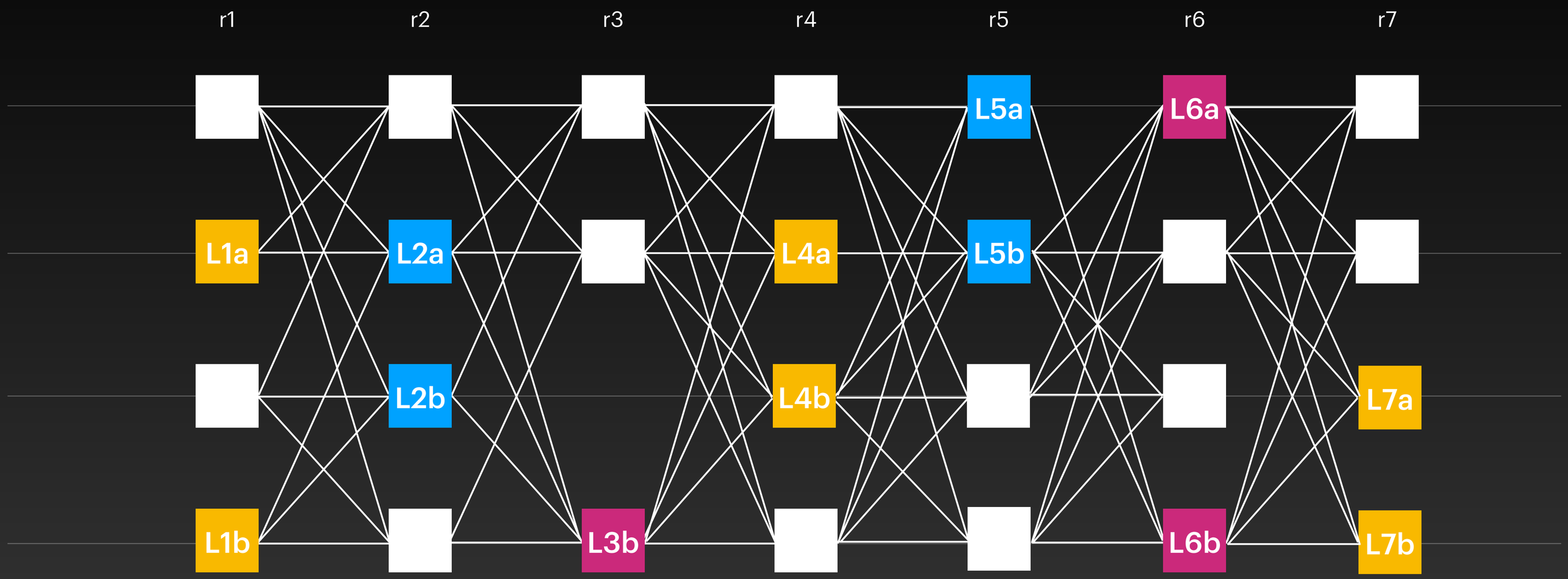

Visa + Master Card
Naive (faults)
Bullshark (faults)
40
35
30
25
20
15
10
5
0
Latency (s)
0
3k
30k
60k
90k
120k
Throughput (tx/s)

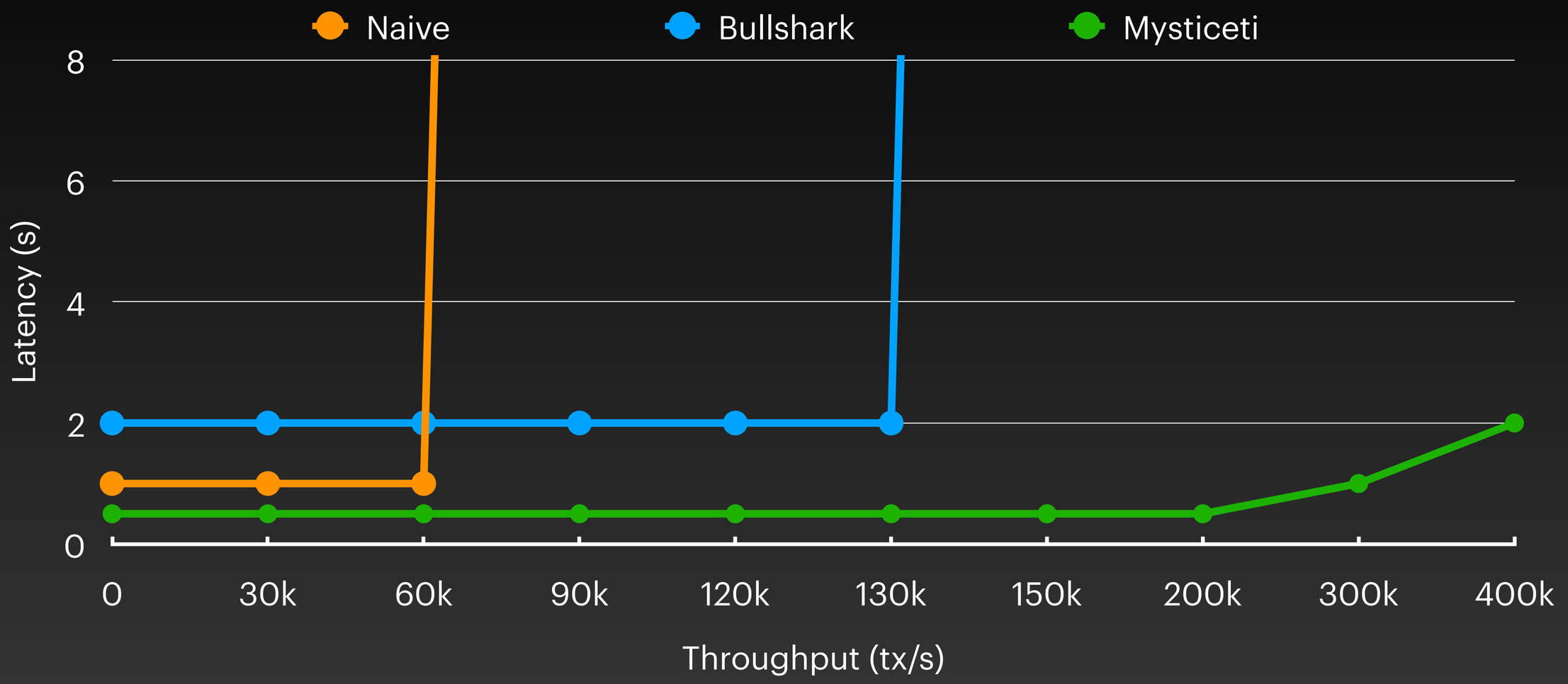

r1
r2
r3
r4
r5
r6
r7
L1
L4
L7

Back Pressure
Robustness

High Resource Utilisation
Throughput

Frequent Commits
Latency


r1
r2
r3
r4
r5
r6
r7
L1a
L1b
L2a
L2b
L3b
L4a
L4b
L5a
L5b
L6a
L6b
L7a
L7b

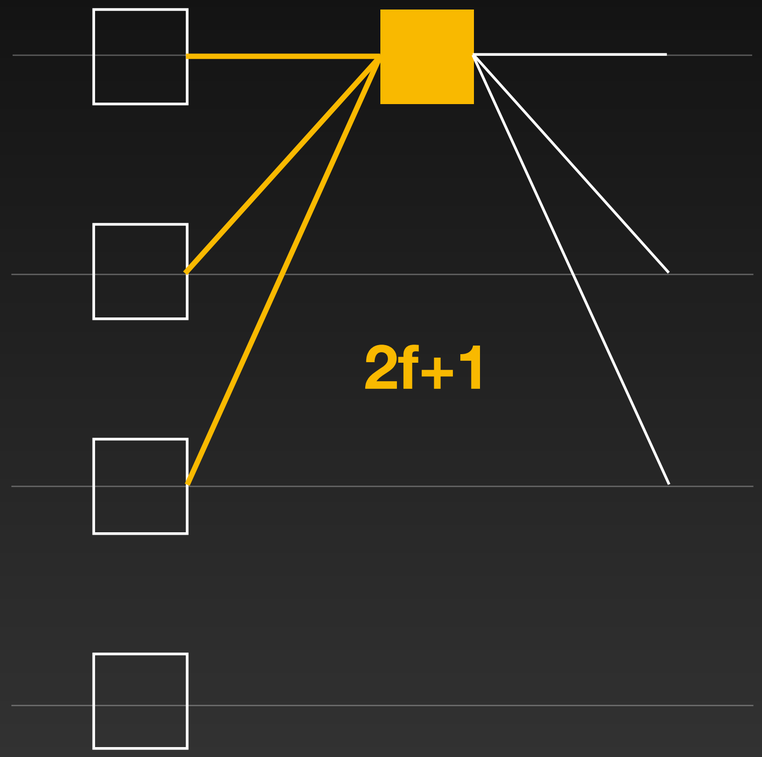

Naive
Bullshark
Mysticeti
Latency (s)
8
6
4
2
0
0
30k
60k
90k
120k
130k
150k
200k
300k
400k
Throughput (tx/s)

Protocol	Committee	Load/TPS	P50	P95
Bullshark	137	5k	2.89 s	4.60 s
Mysticeti	137	5k	397 ms	690 ms

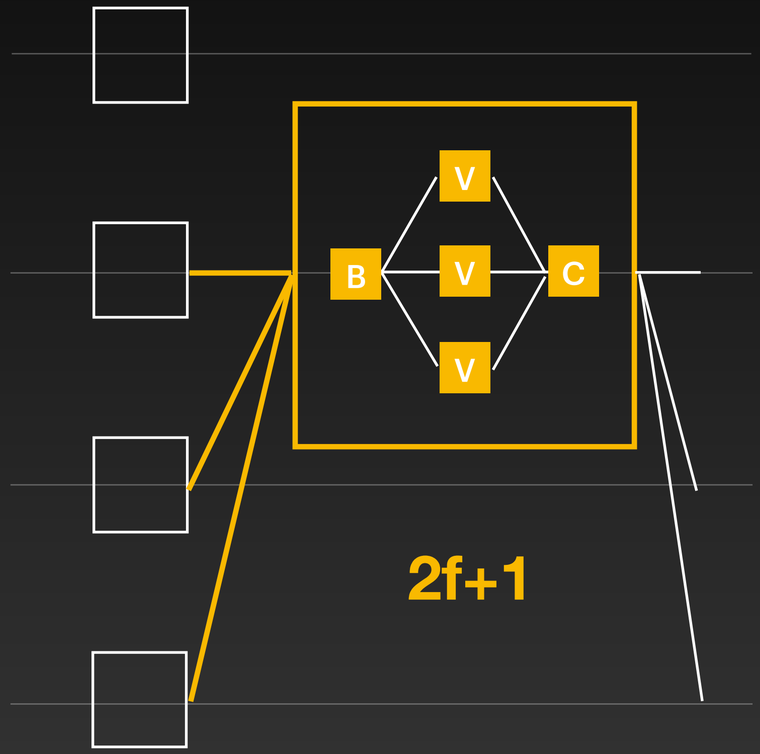
We ran it for 24h: all metrics are fine 👍

Structured DAGs

Uncertified

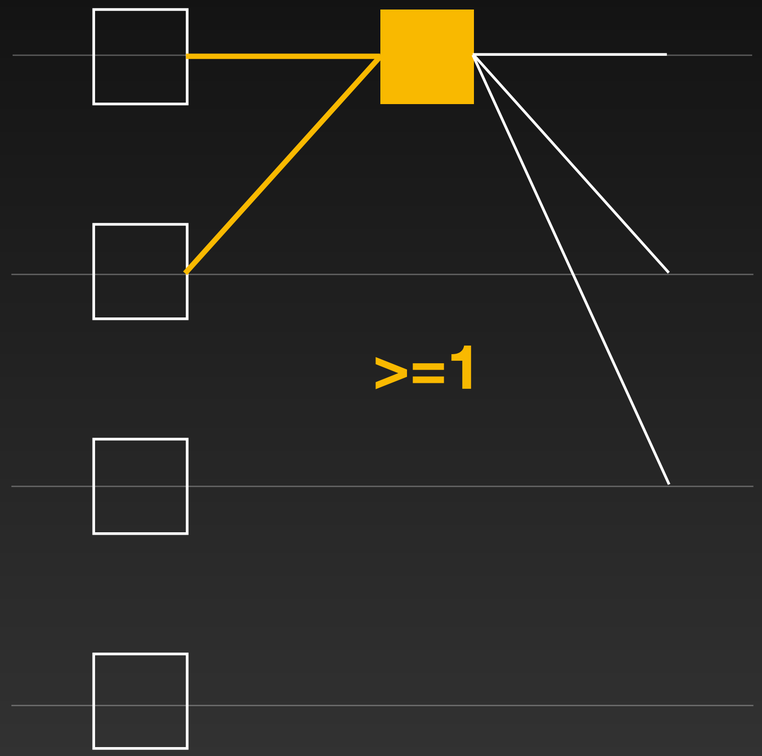


Certified

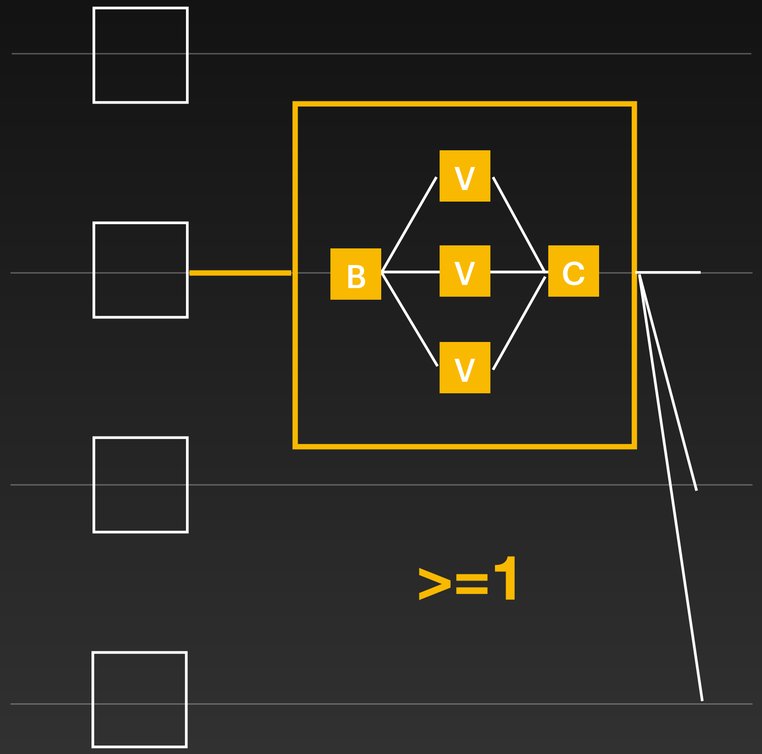


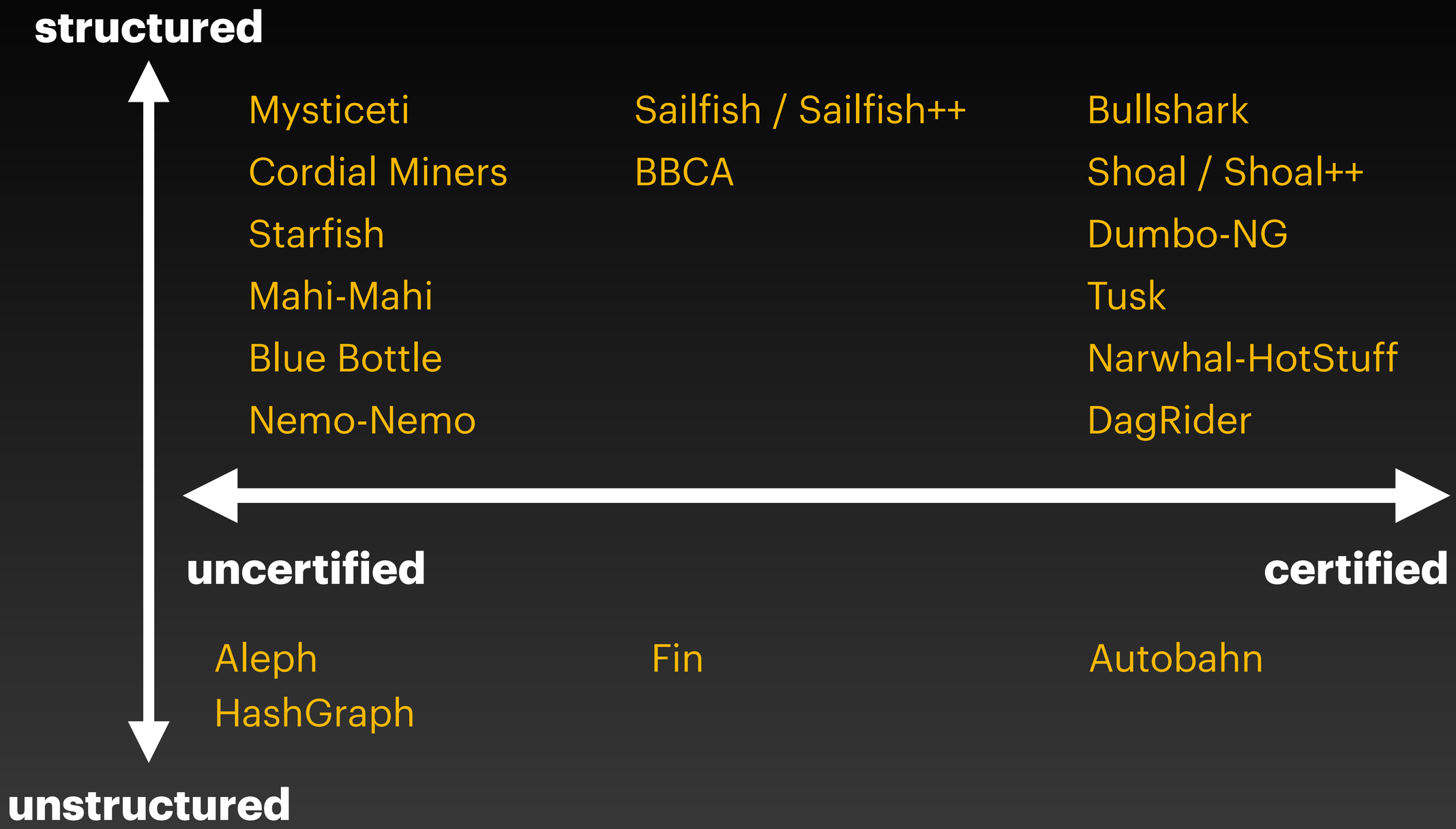
Unstructured DAGs

Uncertified



Certified




structured
Mysticeti
Cordial Miners
Starfish
Mahi-Mahi
Blue Bottle
Nemo-Nemo
Sailfish / Sailfish++
BBCA
Bullshark
Shoal / Shoal++
Dumbo-NG
Tusk
Narwhal-HotStuff
DagRider
uncertified
certified
Aleph
HashGraph
Fin
Autobahn
unstructured

Certified
High Robustness • Easier Block Synchronisation

Uncertified
Low CPU • Graceful Crash Faults

Unstructured
Network Flexibility

Cordial Miners

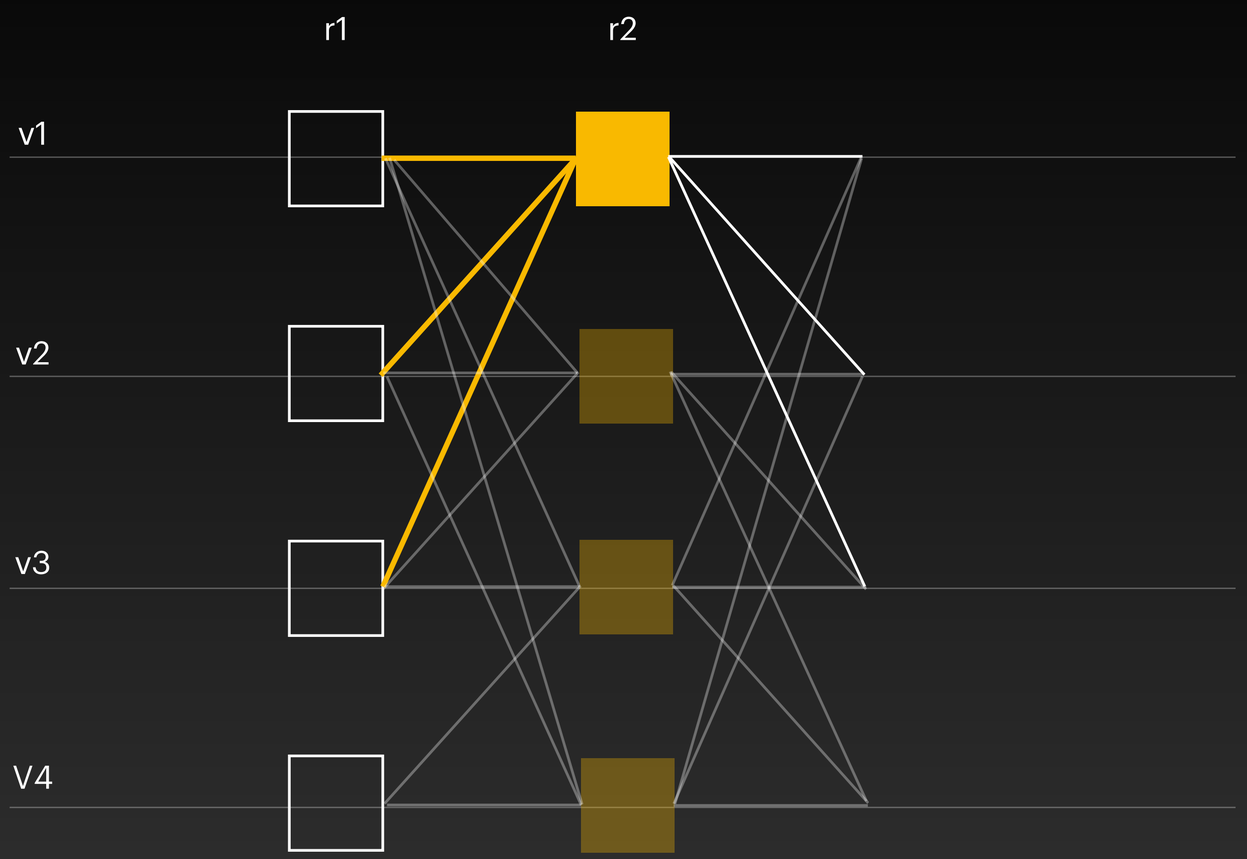
Bullshark

Shoal++

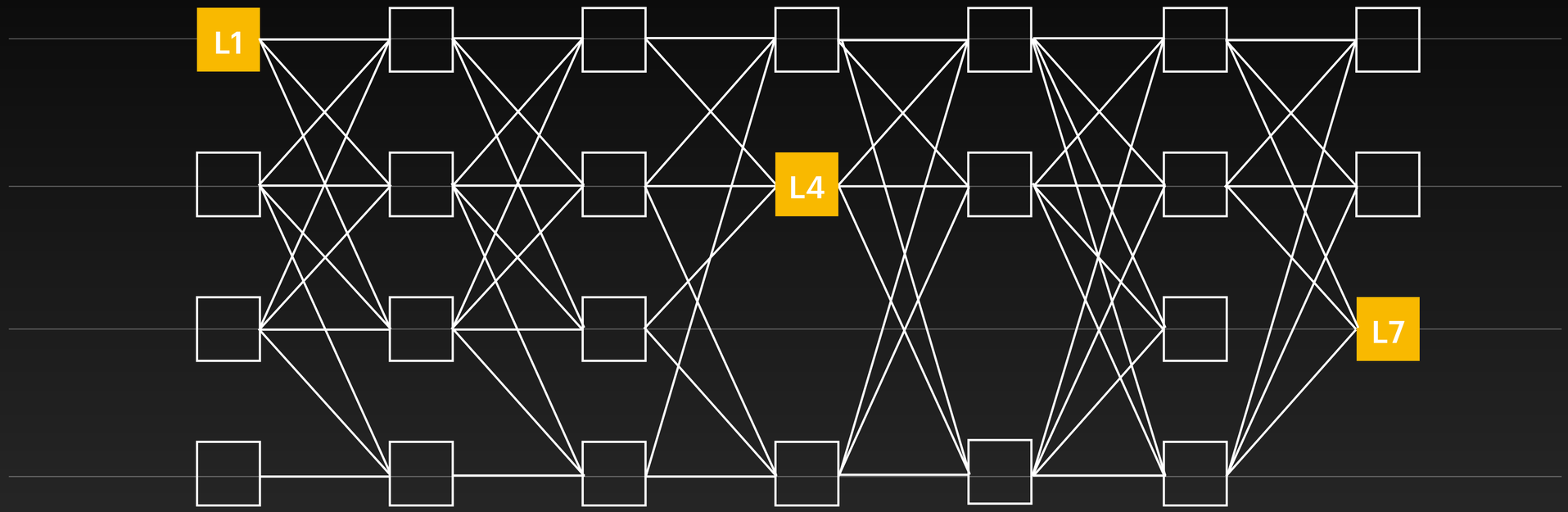
Mysticeti

…

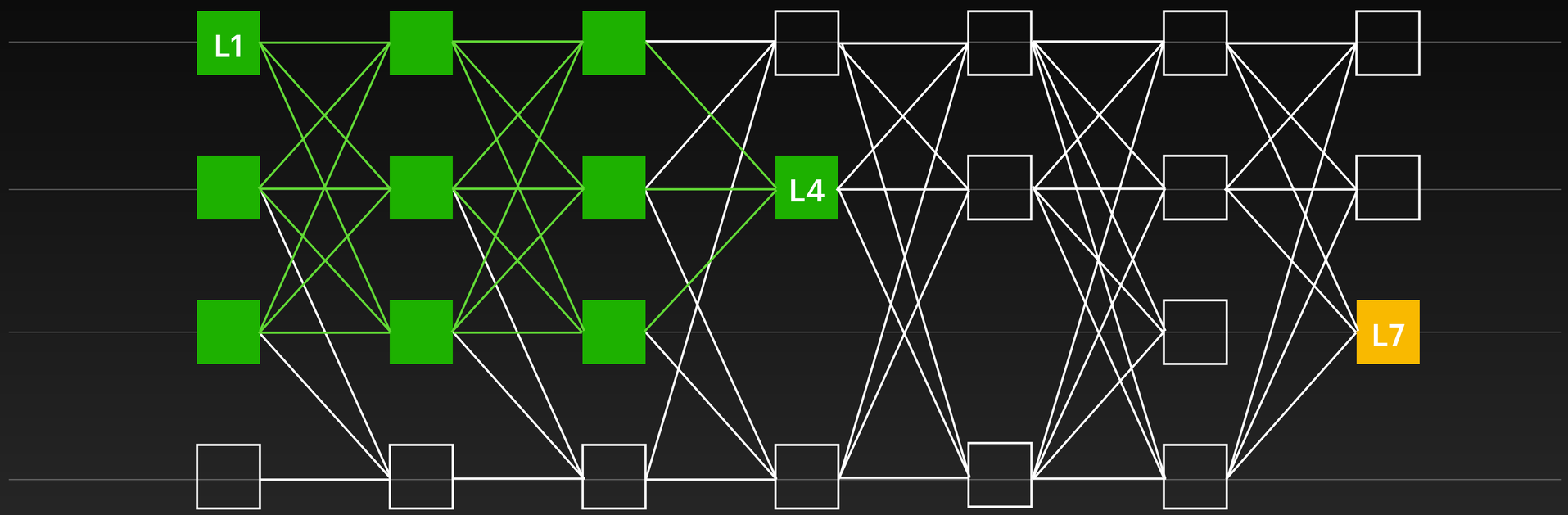
Structured • Uncertified



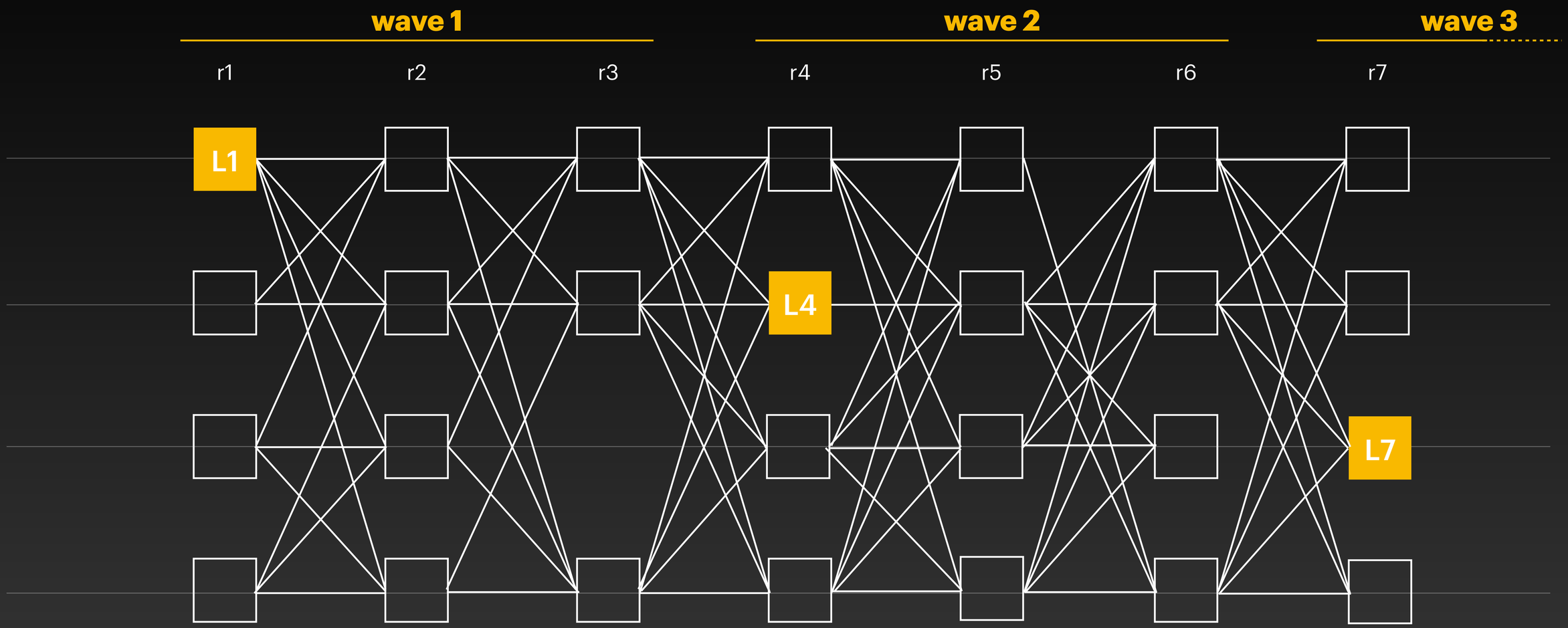
- Round number
- Author
- Payload (txs)
- **2f+1 Parents**
- Signature

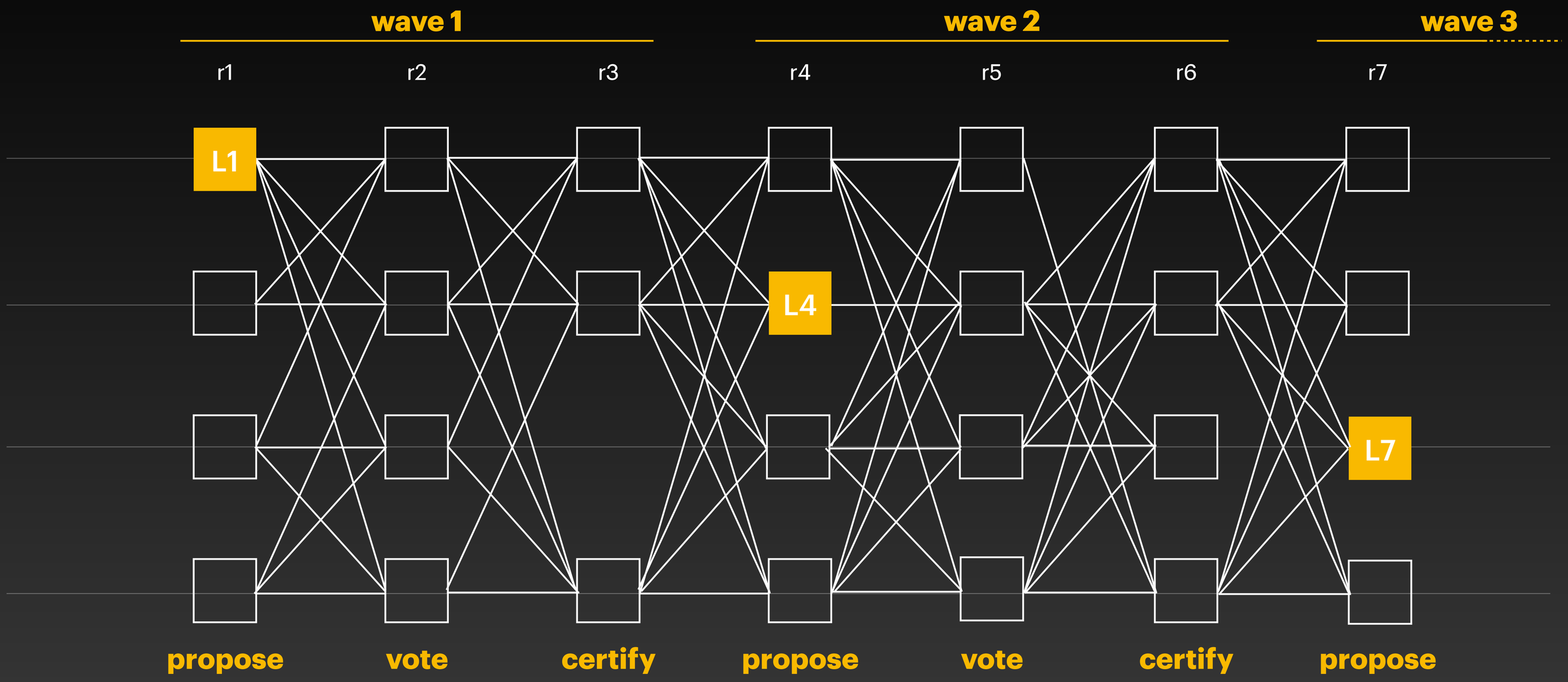


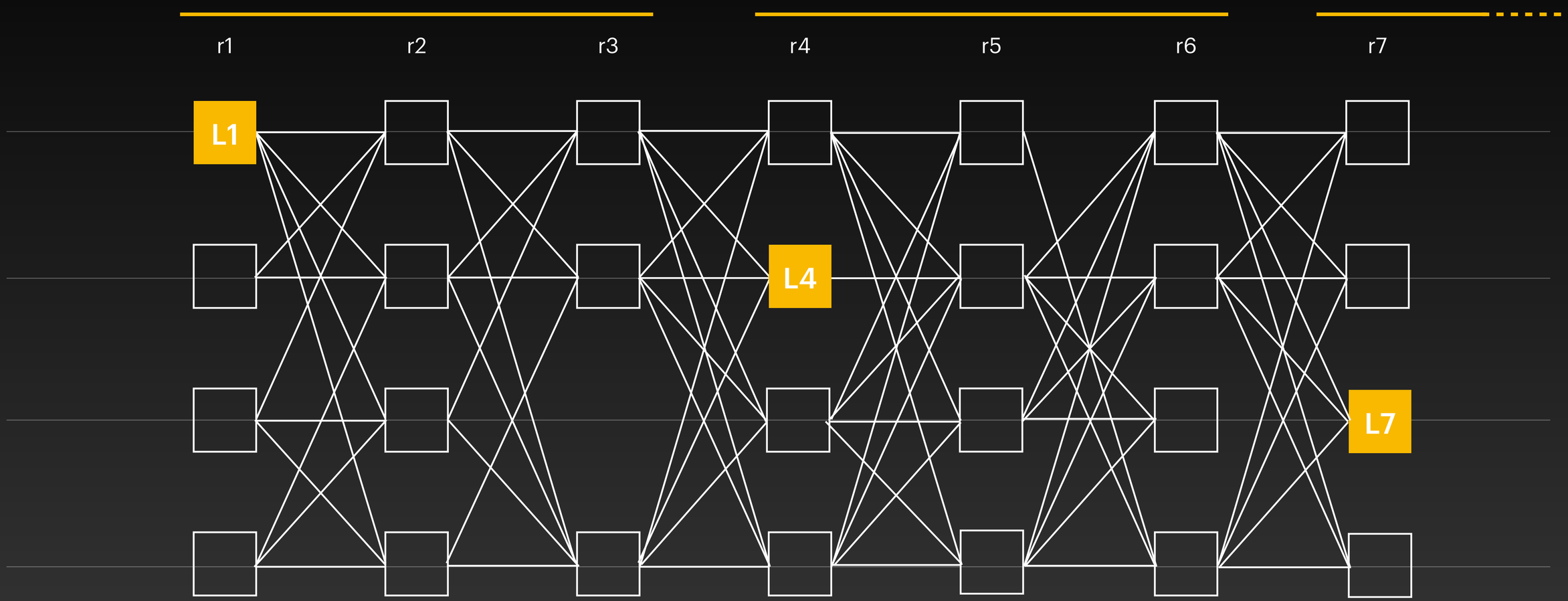
- We focus on ordering leaders: L1 L4 L7

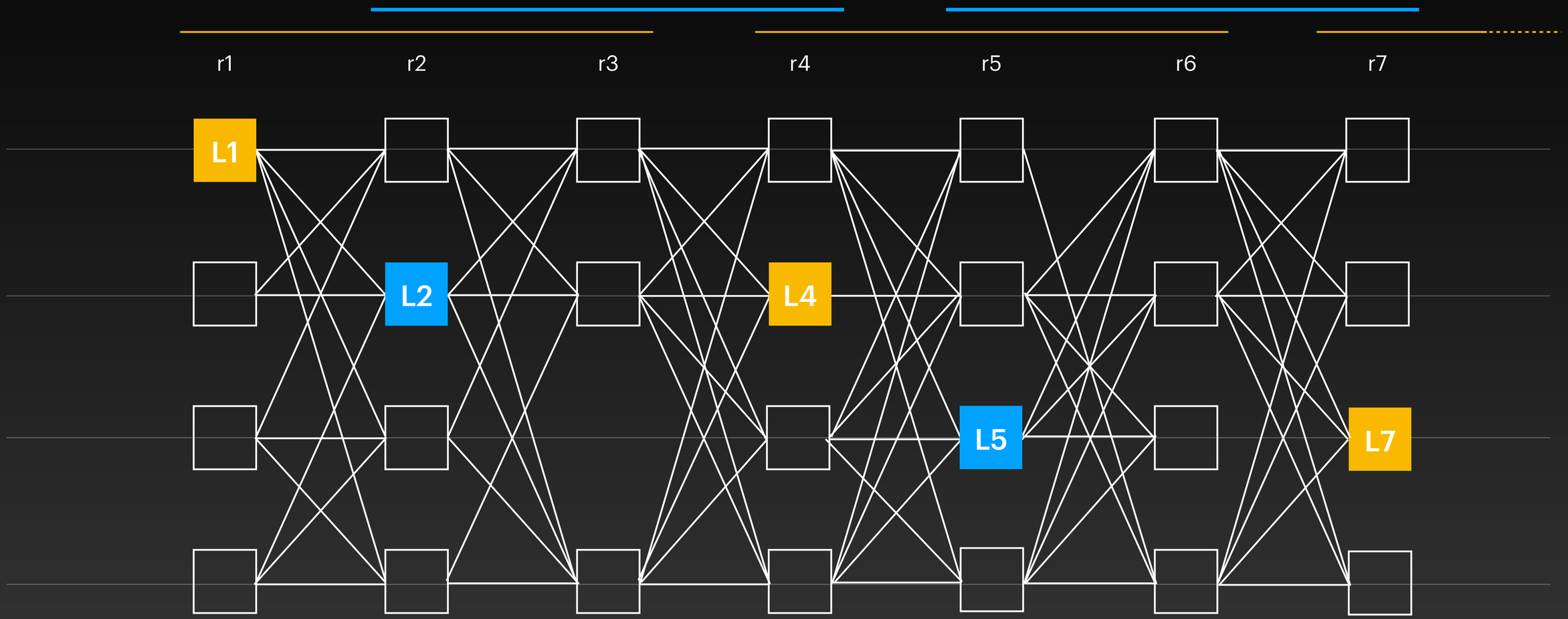


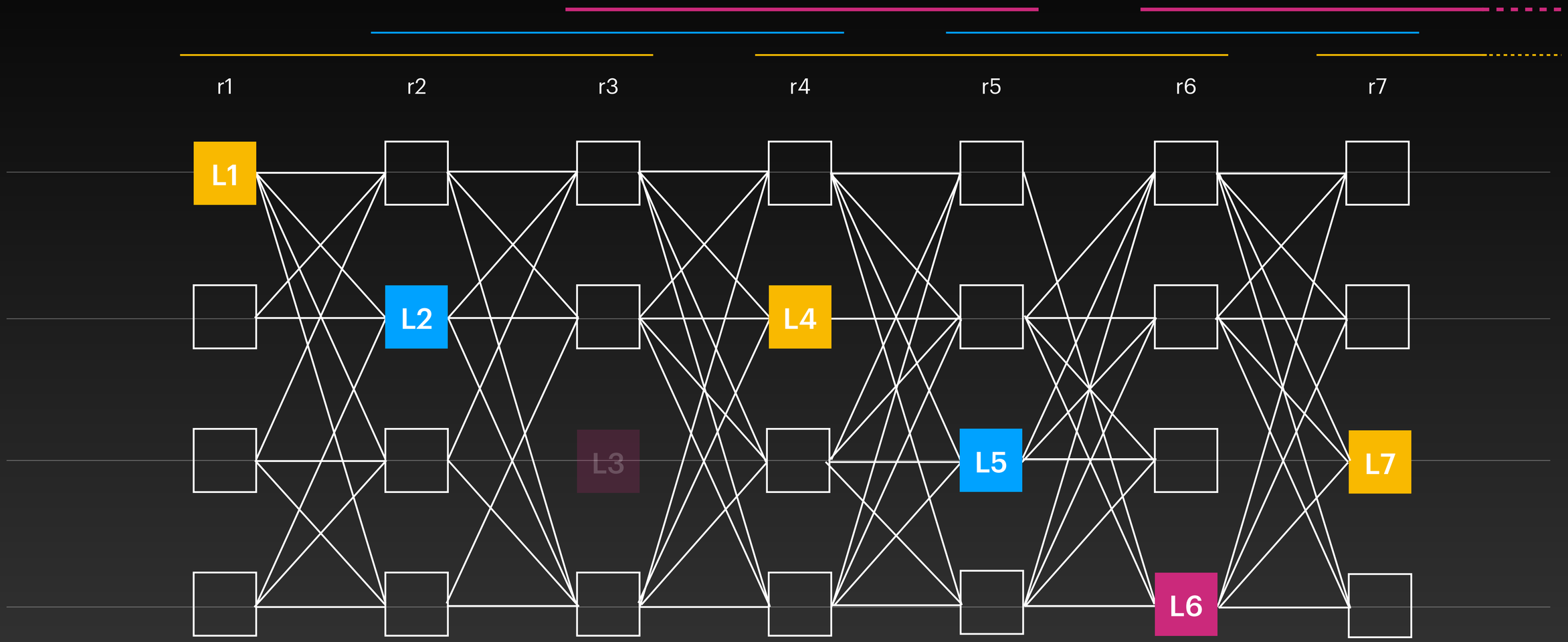
- We focus on ordering leaders: L1 L4 L7
- Linearising the sub-DAG is simple

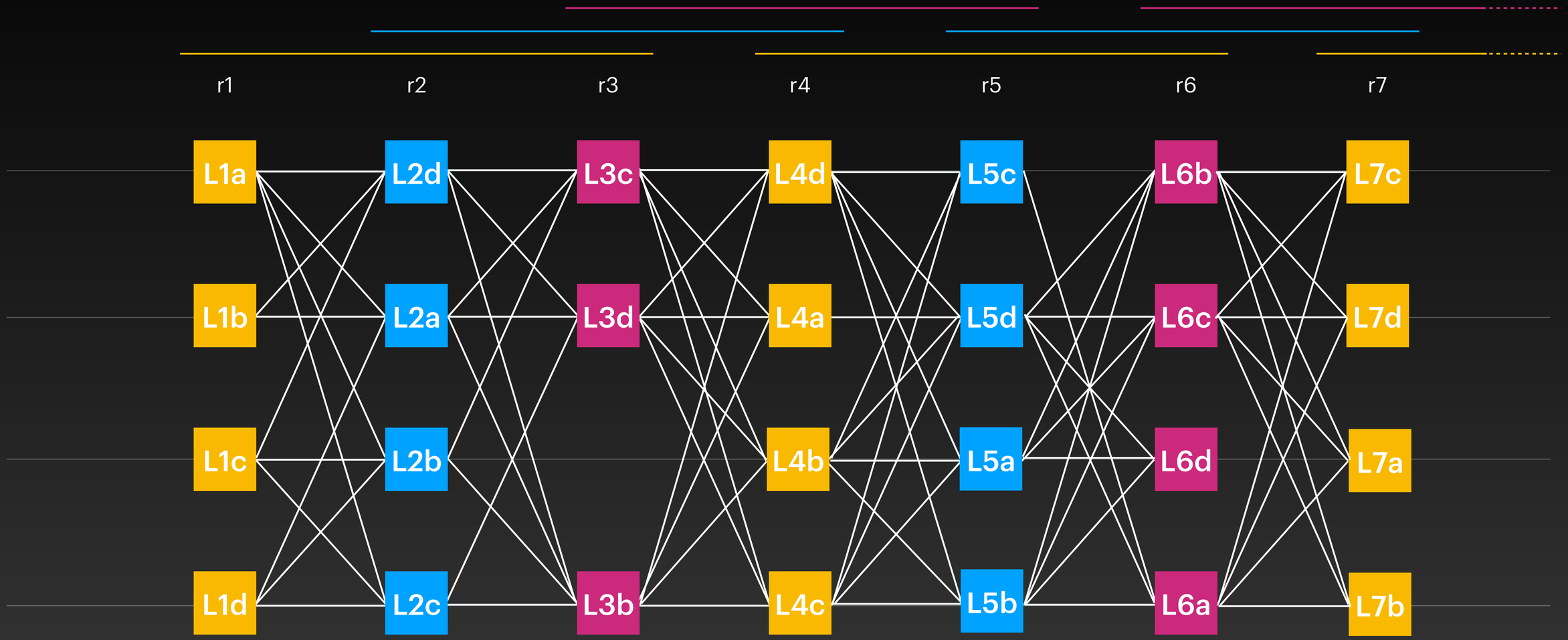

wave 1
wave 2
wave 3
r1
r2
r3
r4
r5
r6
r7
L1
L4
L7

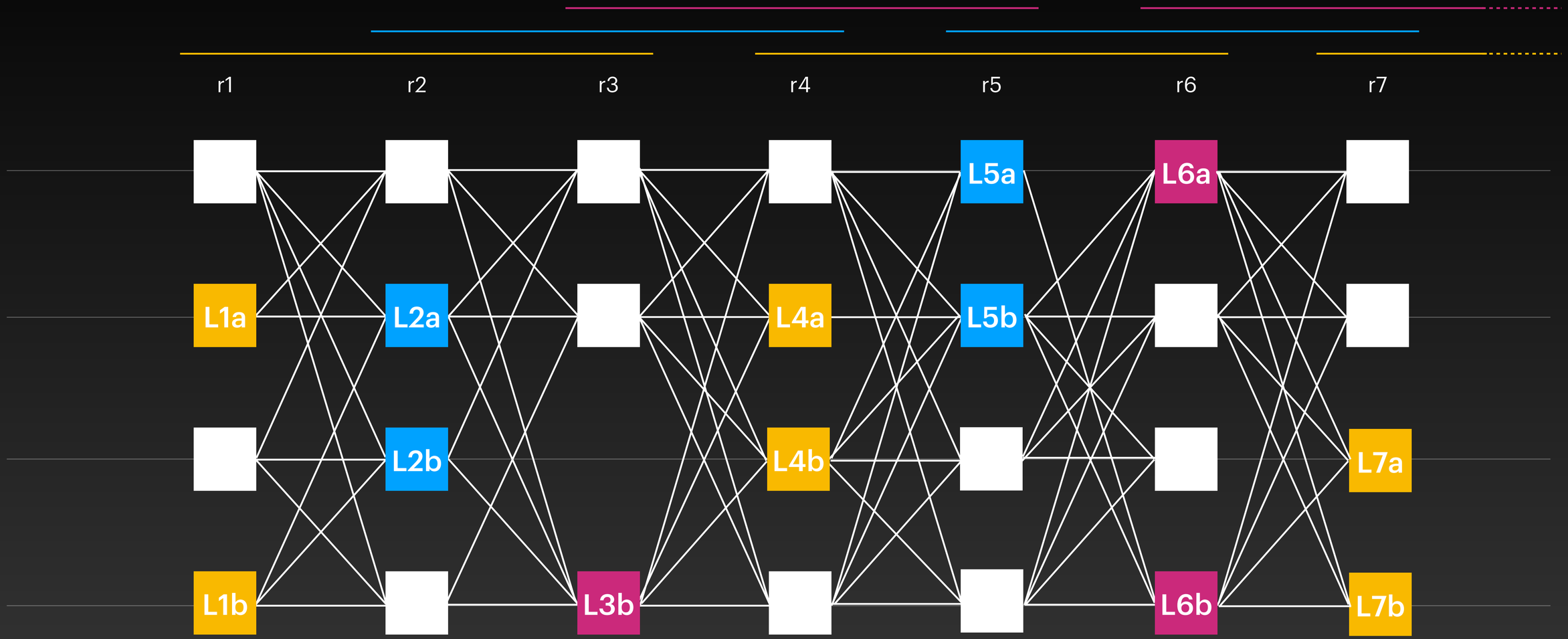

wave 1
wave 2
wave 3
r1
r2
r3
r4
r5
r6
r7
L1
L4
L7
propose
vote
certify
propose
vote
certify
propose

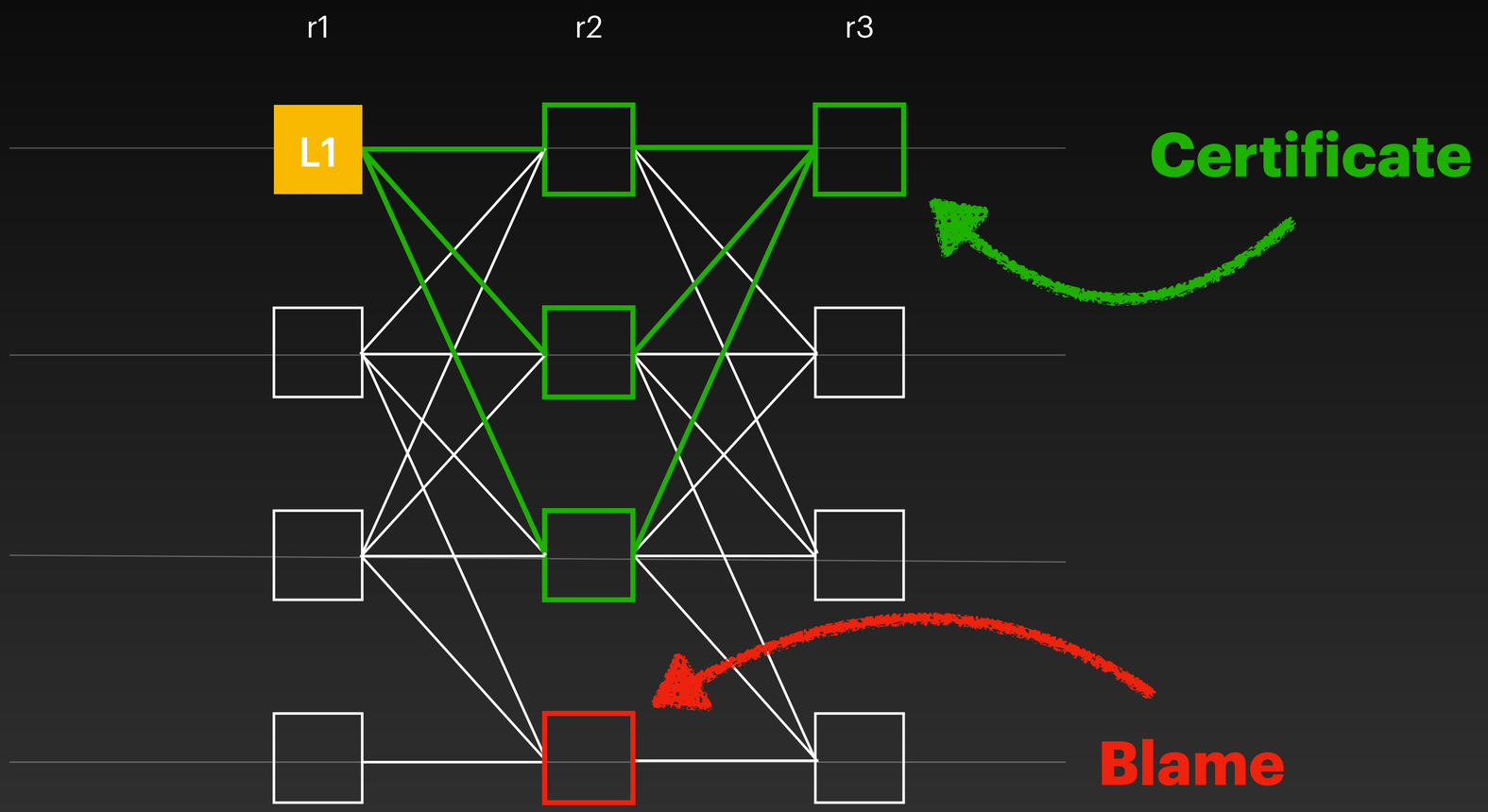

r1
r2
r3
r4
r5
r6
r7
L1
L4
L7


r1
r2
r3
r4
r5
r6
r7
L1
L2
L4
L5
L7


r1
r2
r3
r4
r5
r6
r7
L1
L2
L3
L4
L5
L6
L7


r1
r2
r3
r4
r5
r6
r7
L1a
L1b
L1c
L1d
L2d
L2a
L2b
L2c
L3c
L3d
L3b
L4d
L4a
L4b
L4c
L5c
L5d
L5a
L5b
L6b
L6c
L6d
L6a
L7c
L7d
L7a
L7b


r1
r2
r3
r4
r5
r6
r7
L1a
L1b
L2a
L2b
L3b
L4a
L4b
L5a
L5b
L6a
L6b
L7a
L7b


r1
r2
r3
L1
Certificate
Blame

Direct Decision Rule

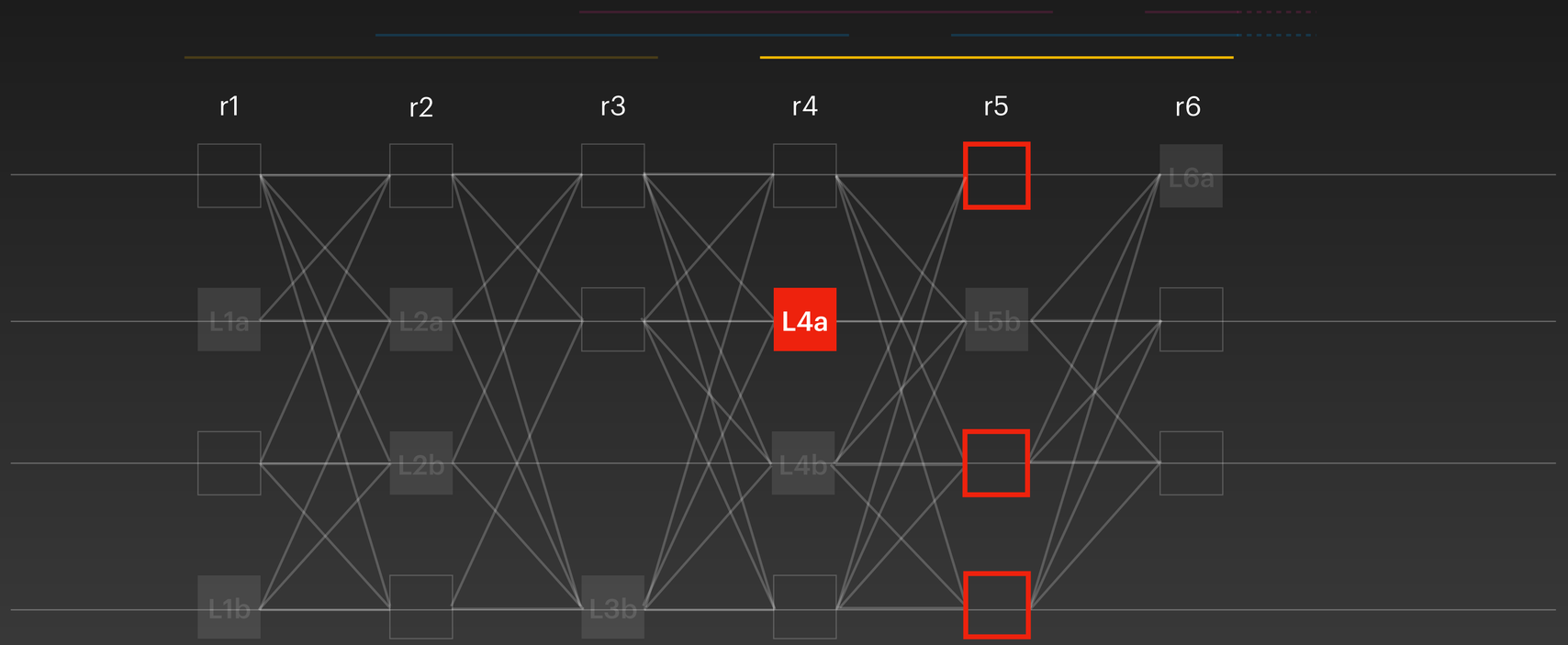
On each leader starting from highest round:

- **Skip** if 2f+1 blames
- **Commit** if 2f+1 certificates
- **Undecided** otherwise

Direct Decision Rule

On each leader starting from highest round:

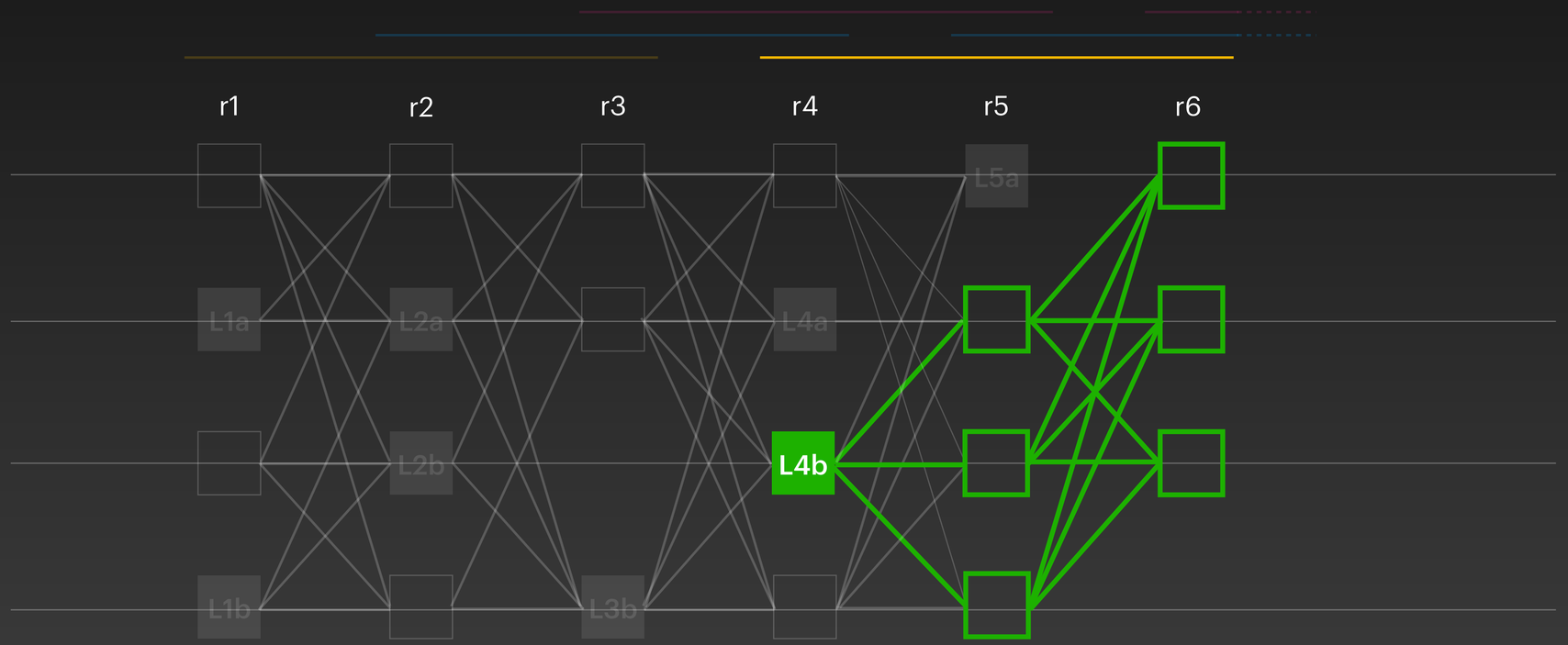
- **Skip** if 2f+1 blames
- **Commit** if 2f+1 certificates
- **Undecided** otherwise



Direct Decision Rule

On each leader starting from highest round:

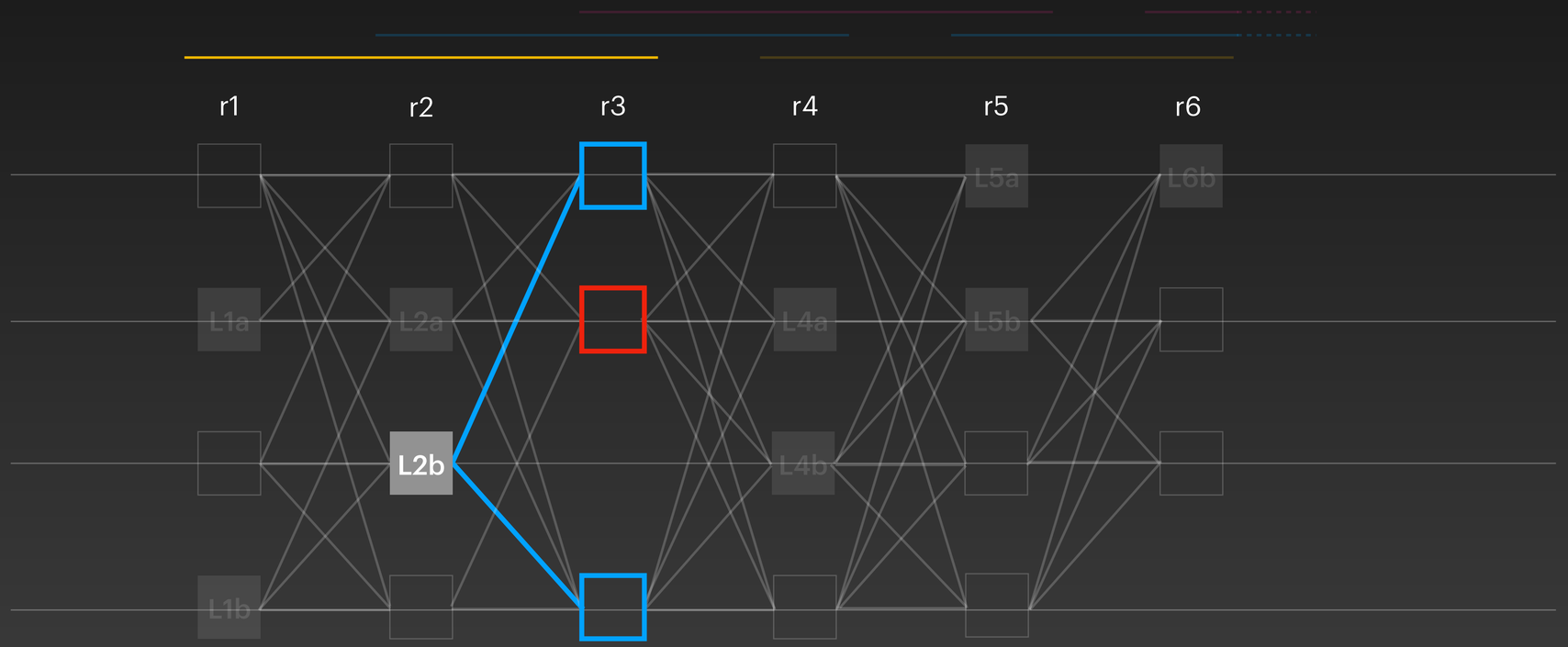
- **Skip** if 2f+1 blames
- **Commit** if 2f+1 certificates
- **Undecided** otherwise



Direct Decision Rule

On each leader starting from highest round:

- **Skip** if 2f+1 blames
- **Commit** if 2f+1 certificates
- **Undecided** otherwise

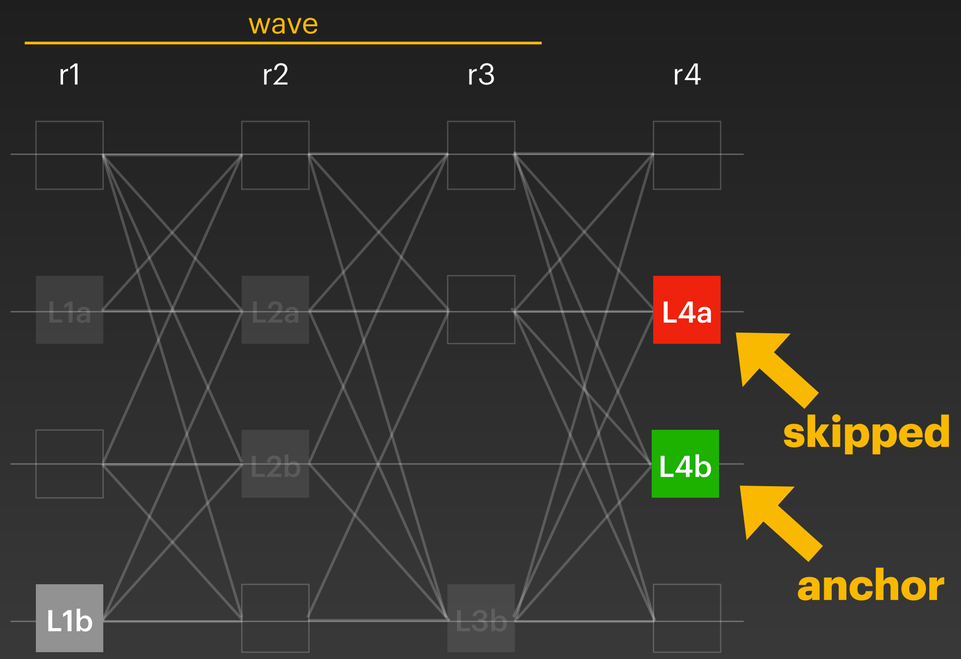


Indirect Decision Rule

Indirect Decision Rule

1. Find Anchor

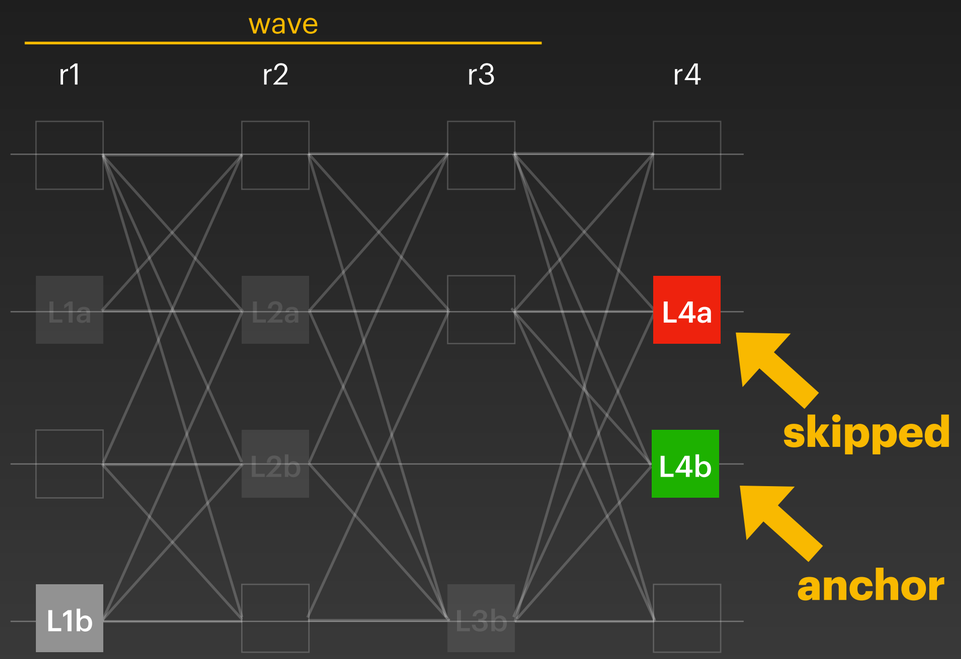
- First block with round > r+2 that is **Commit** or **Undecided**



Indirect Decision Rule

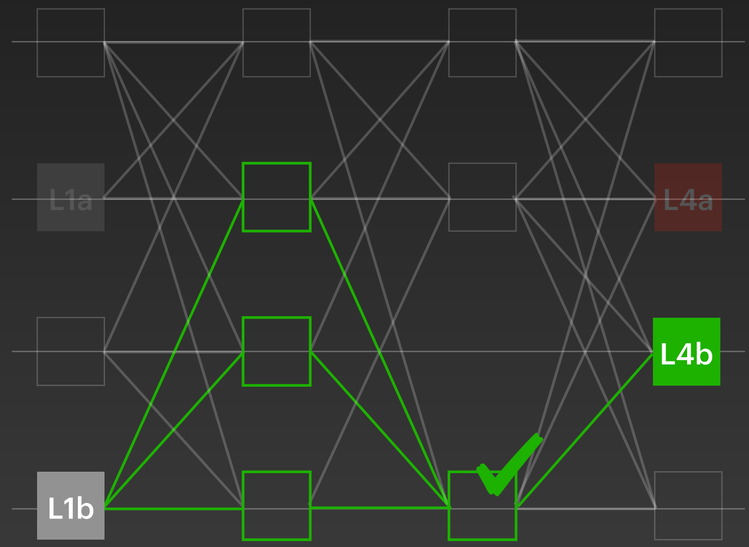
1. Find Anchor

- First block with round > r+2 that is **Commit** or **Undecided**

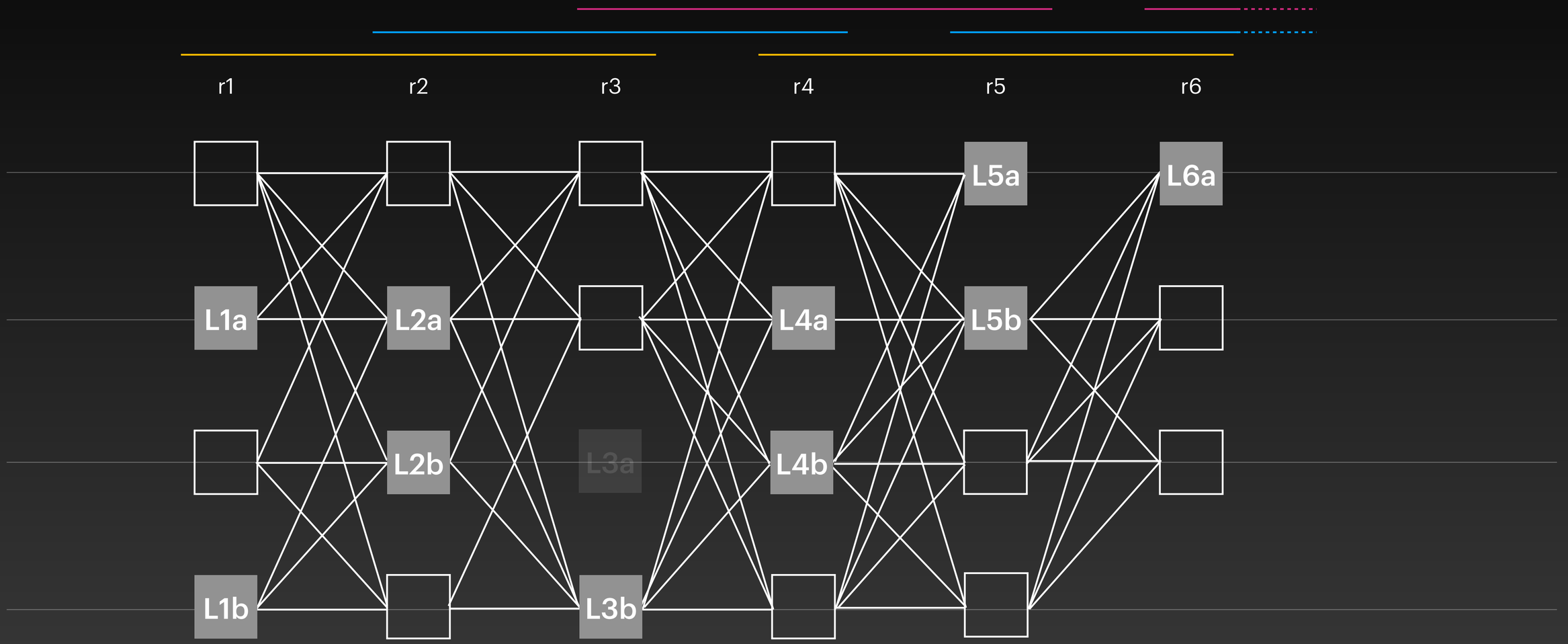


2. Certified link

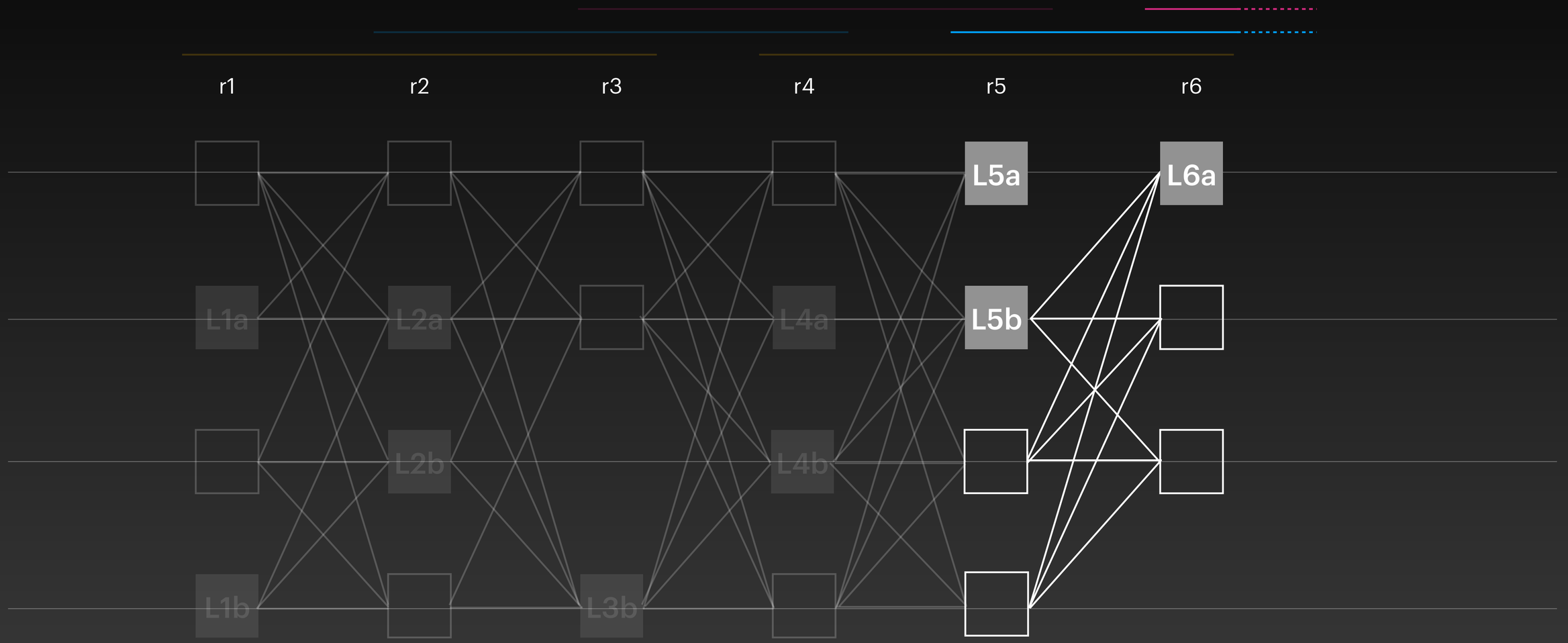
- **Commit** if `B <-> certified link <-> A` otherwise **Skip**



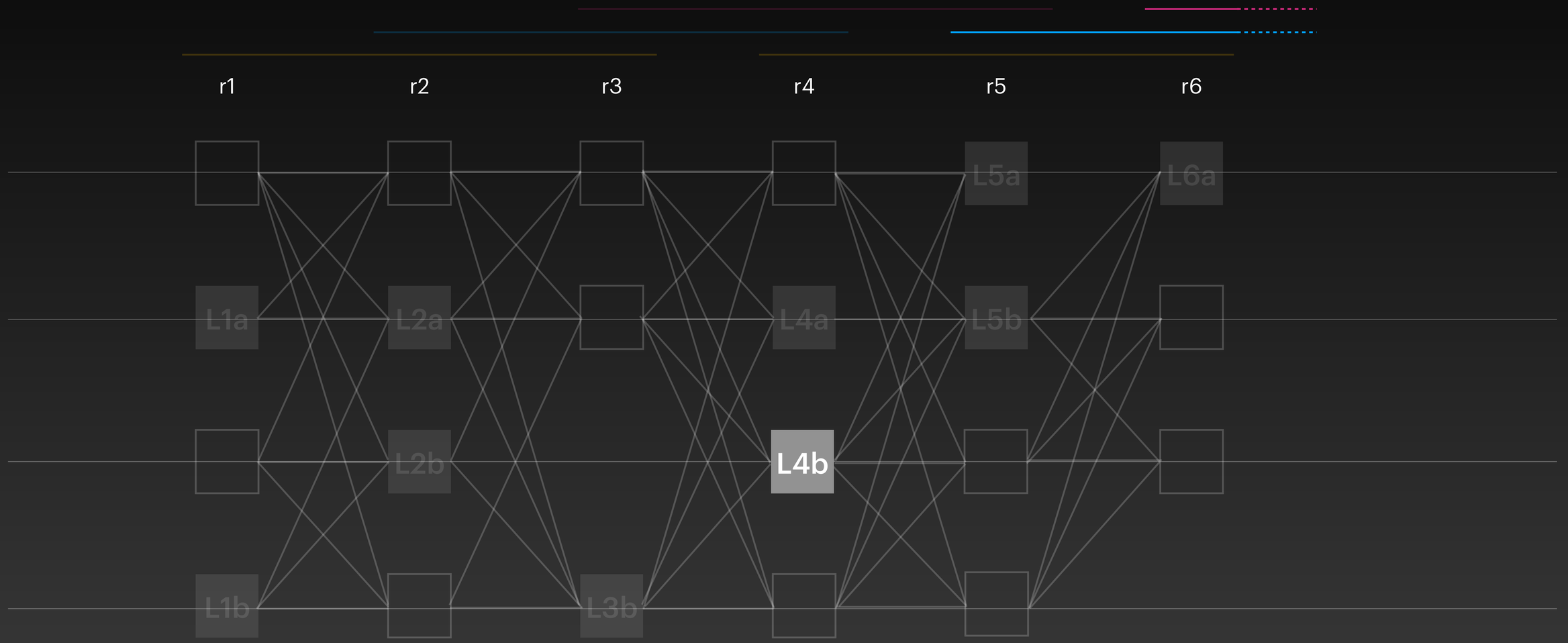
All Start at Undecided



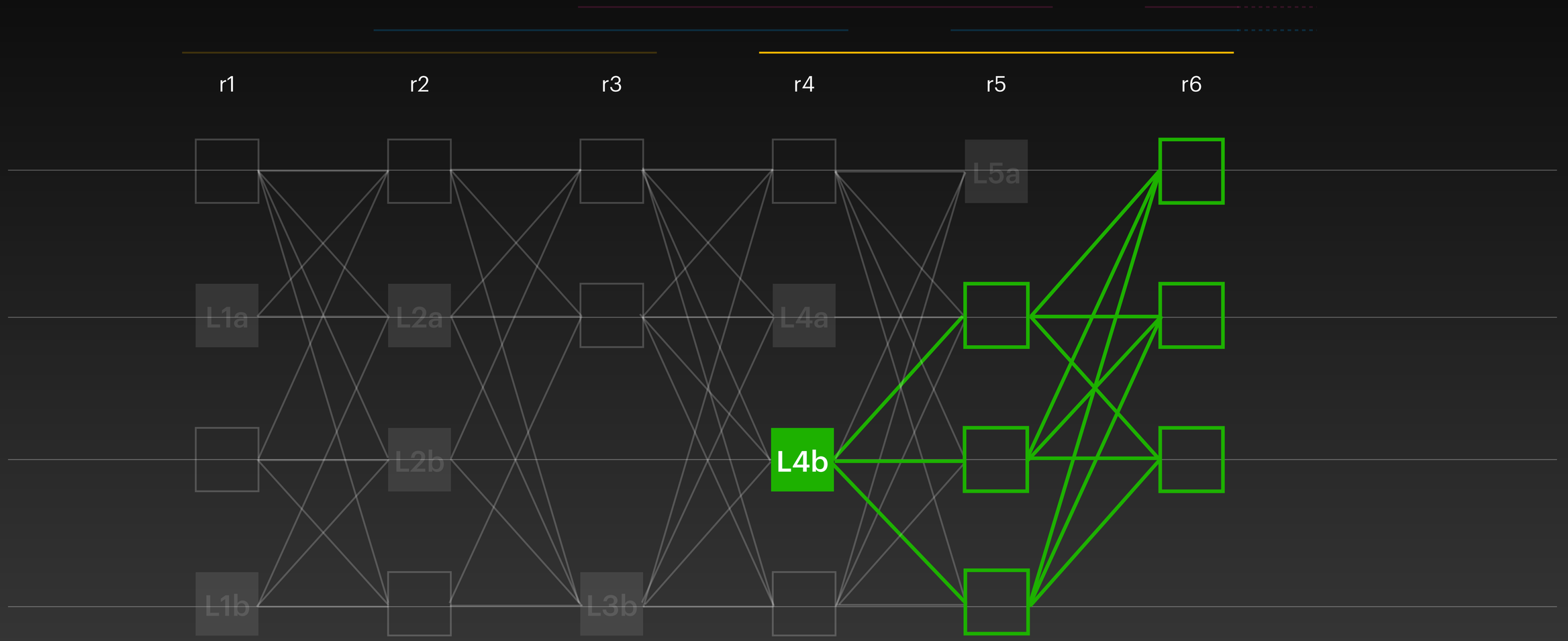
Ignore Incomplete Waves



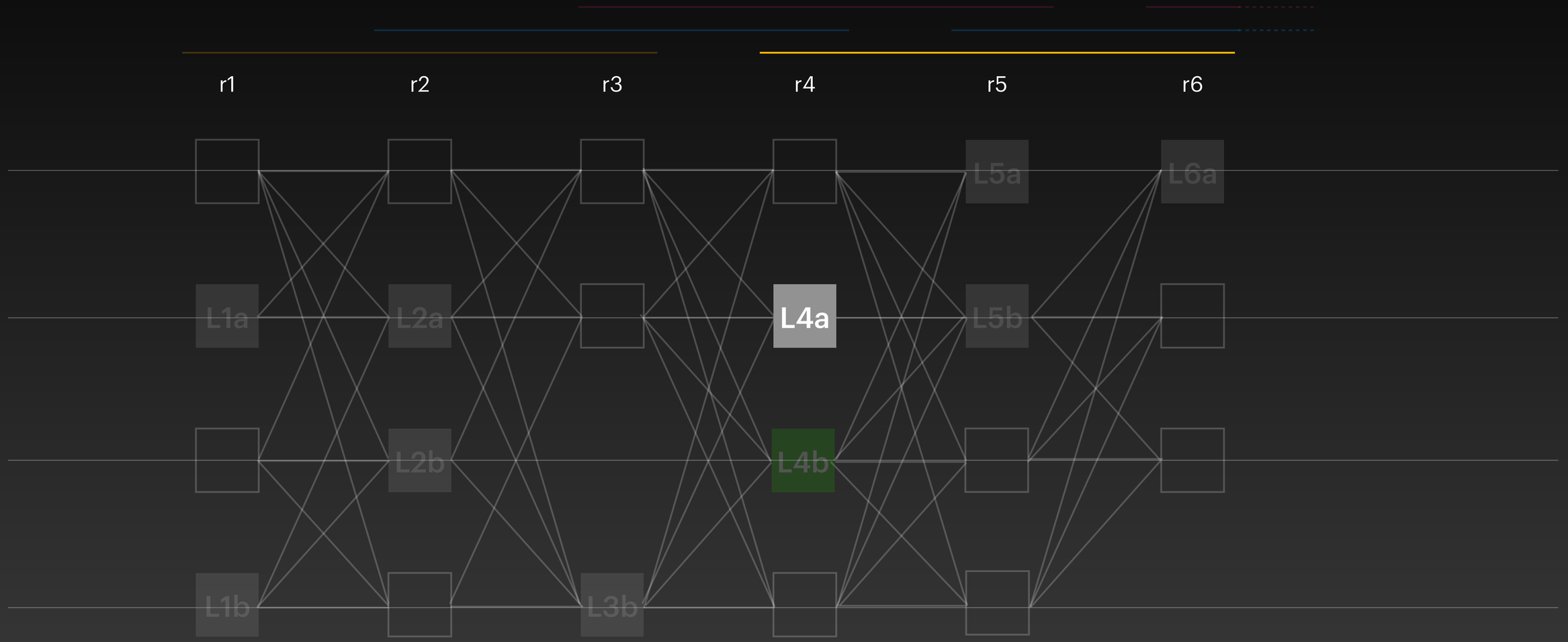
Apply Direct Rule



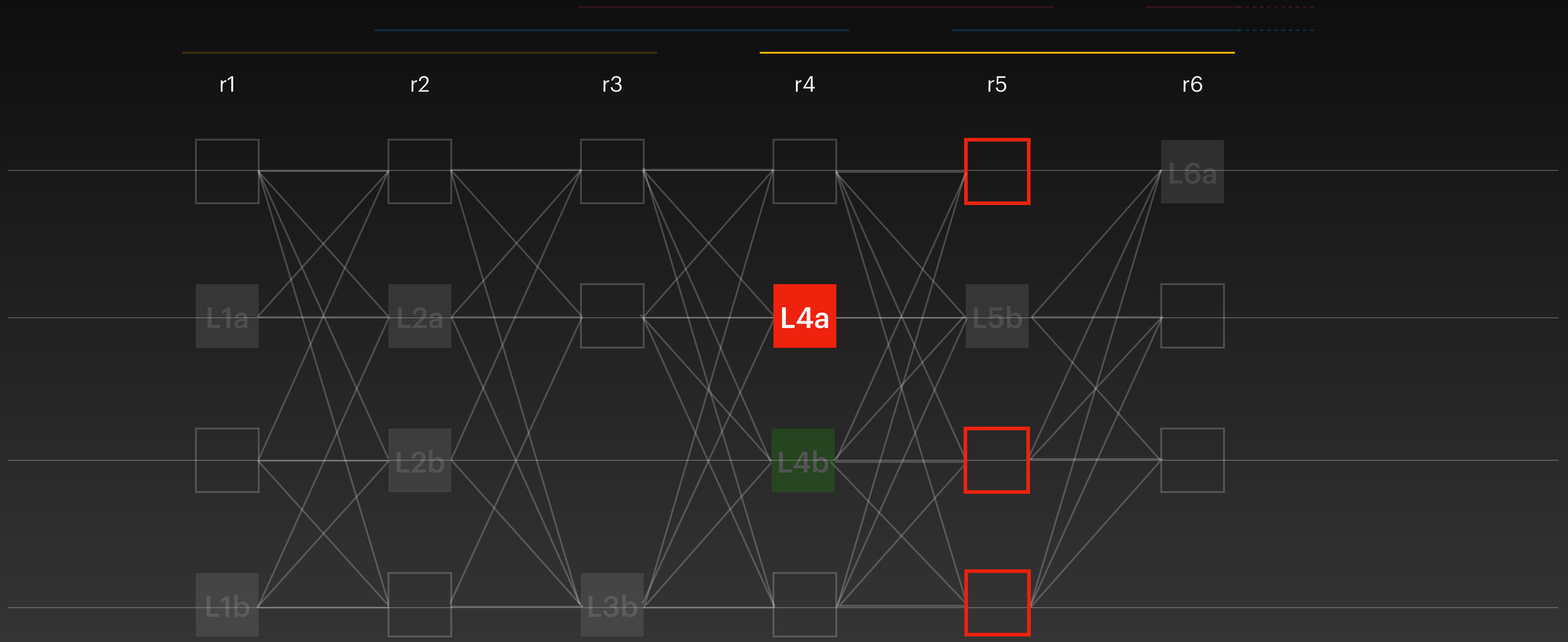
Apply Direct Rule



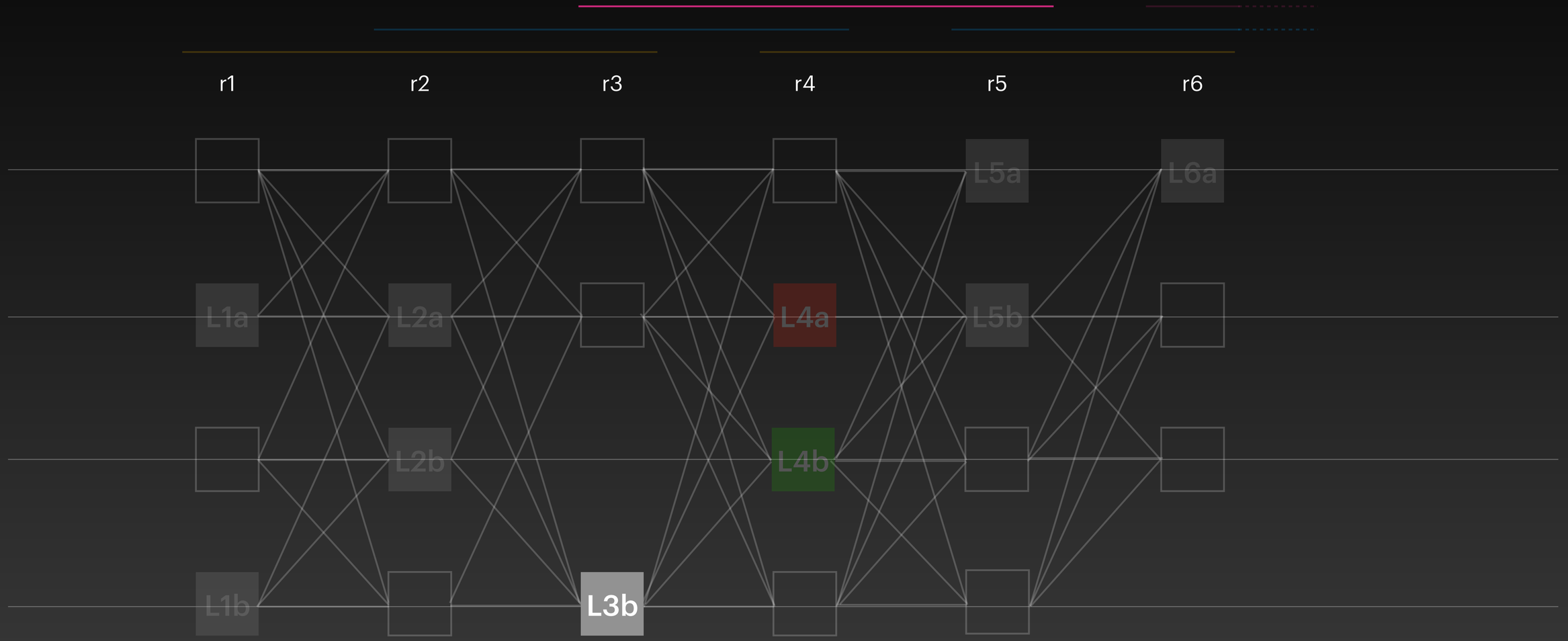
Apply Direct Rule



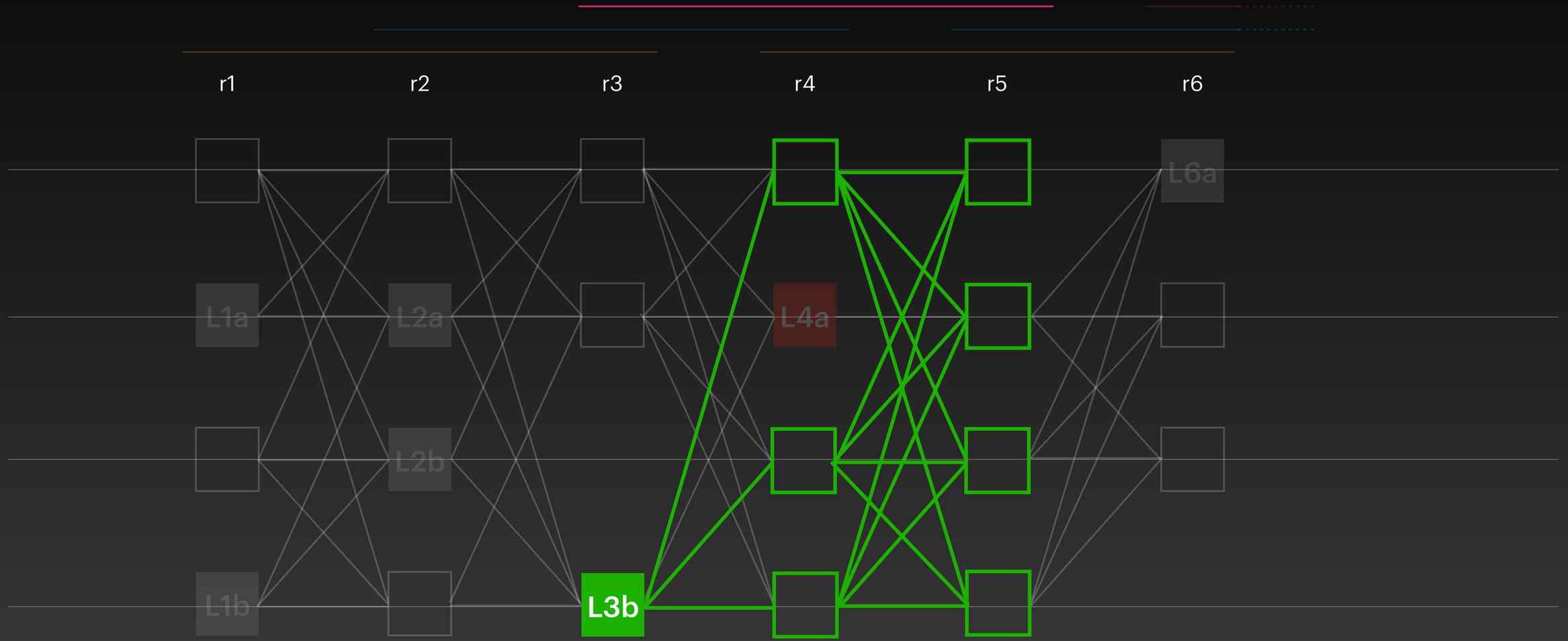
Apply Direct Rule



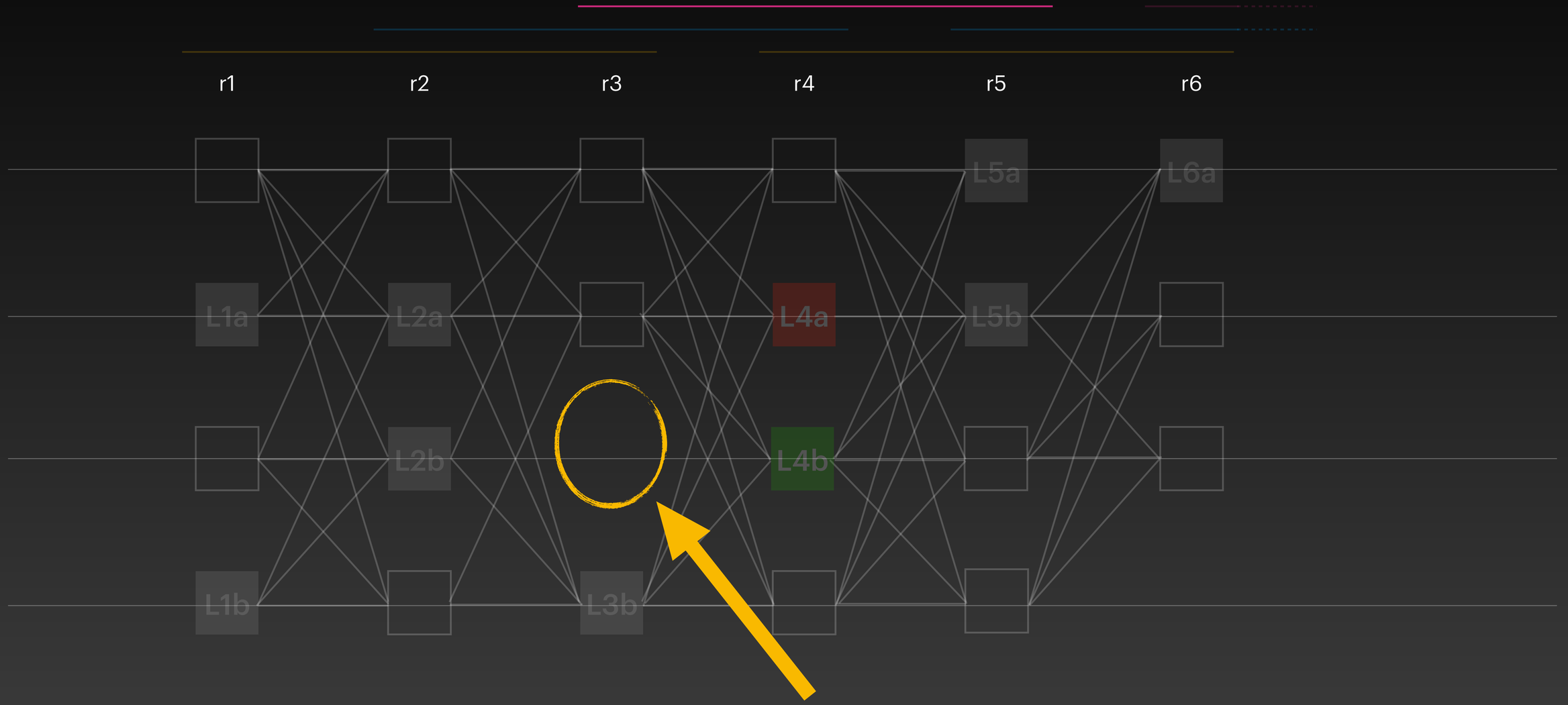
Apply Direct Rule



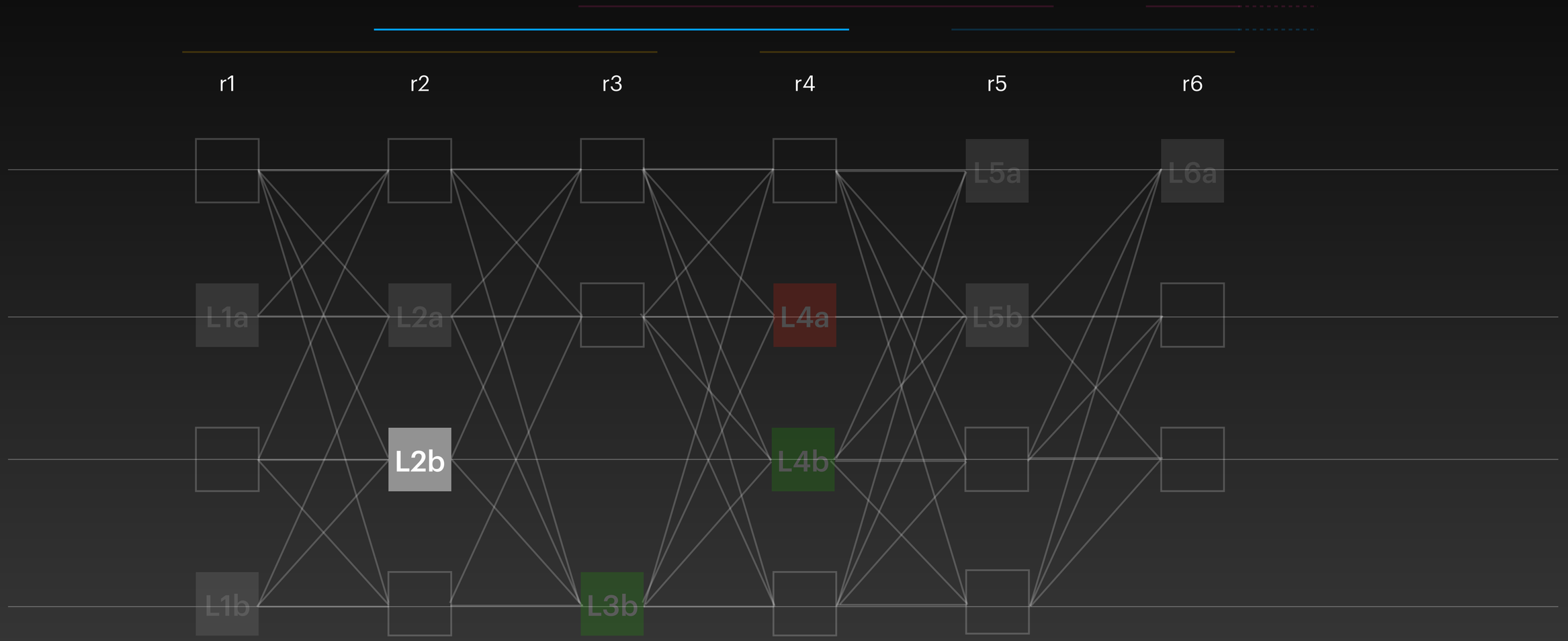
Apply Direct Rule



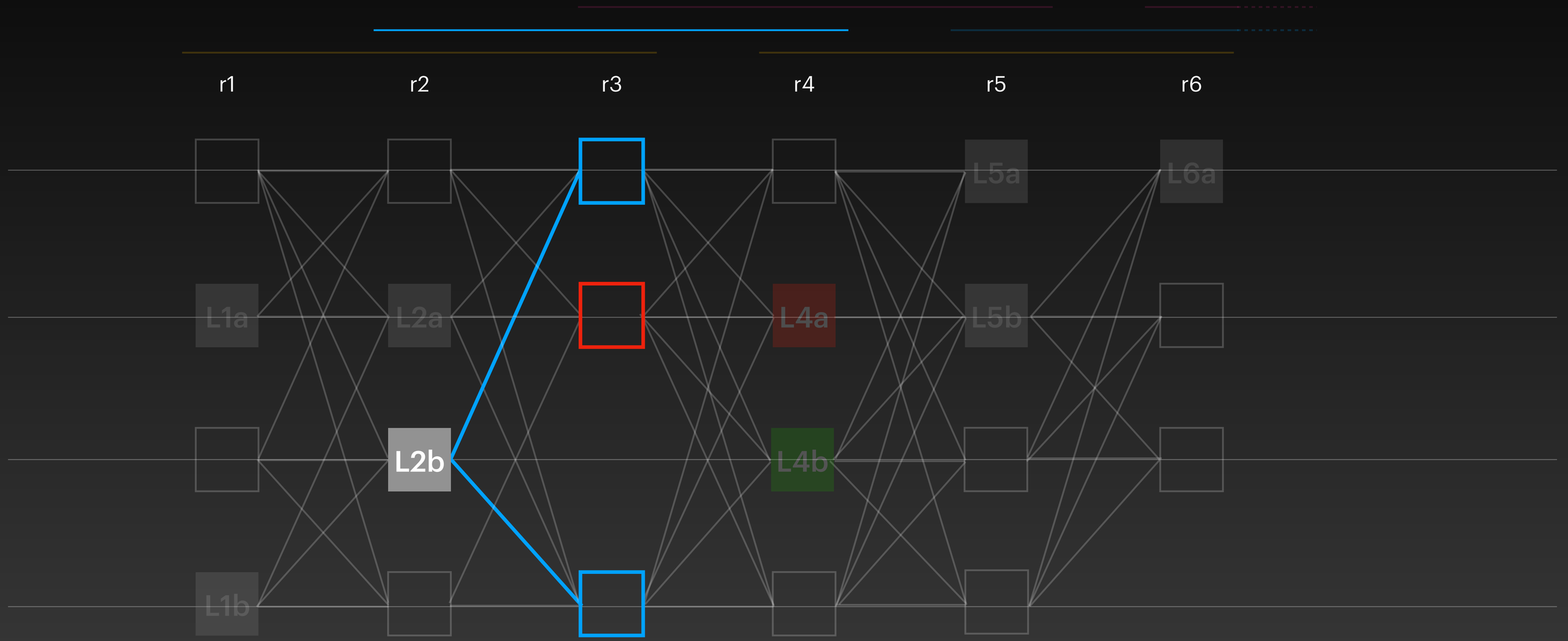
Ignore Missing Leader



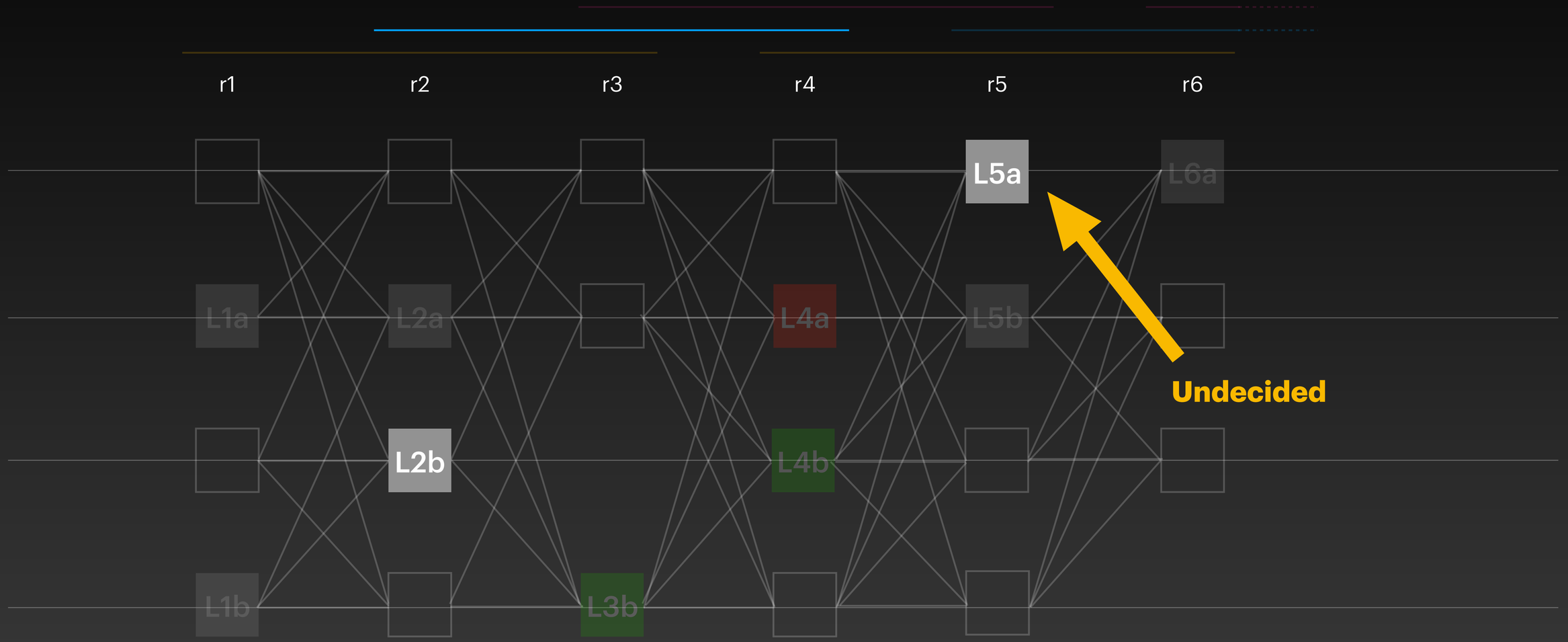
Apply Direct Rule



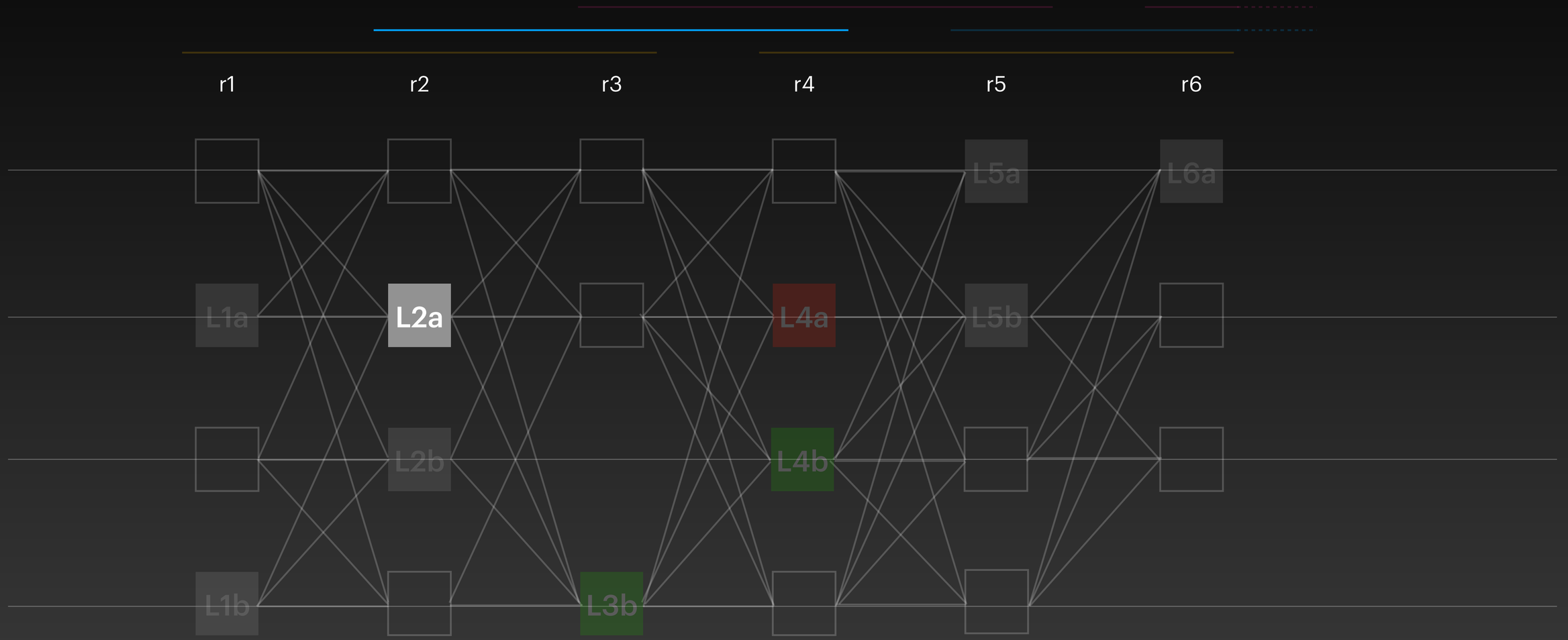
Apply Direct Rule



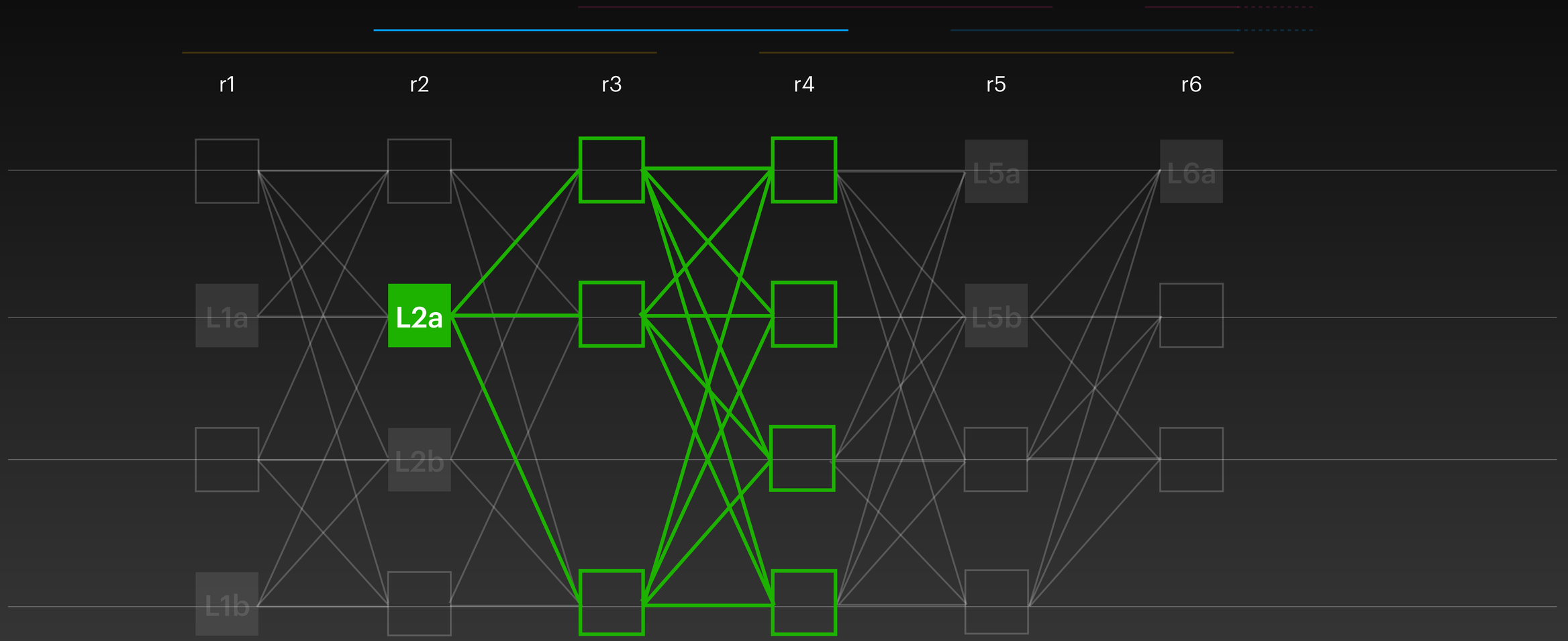
Apply Indirect Rule



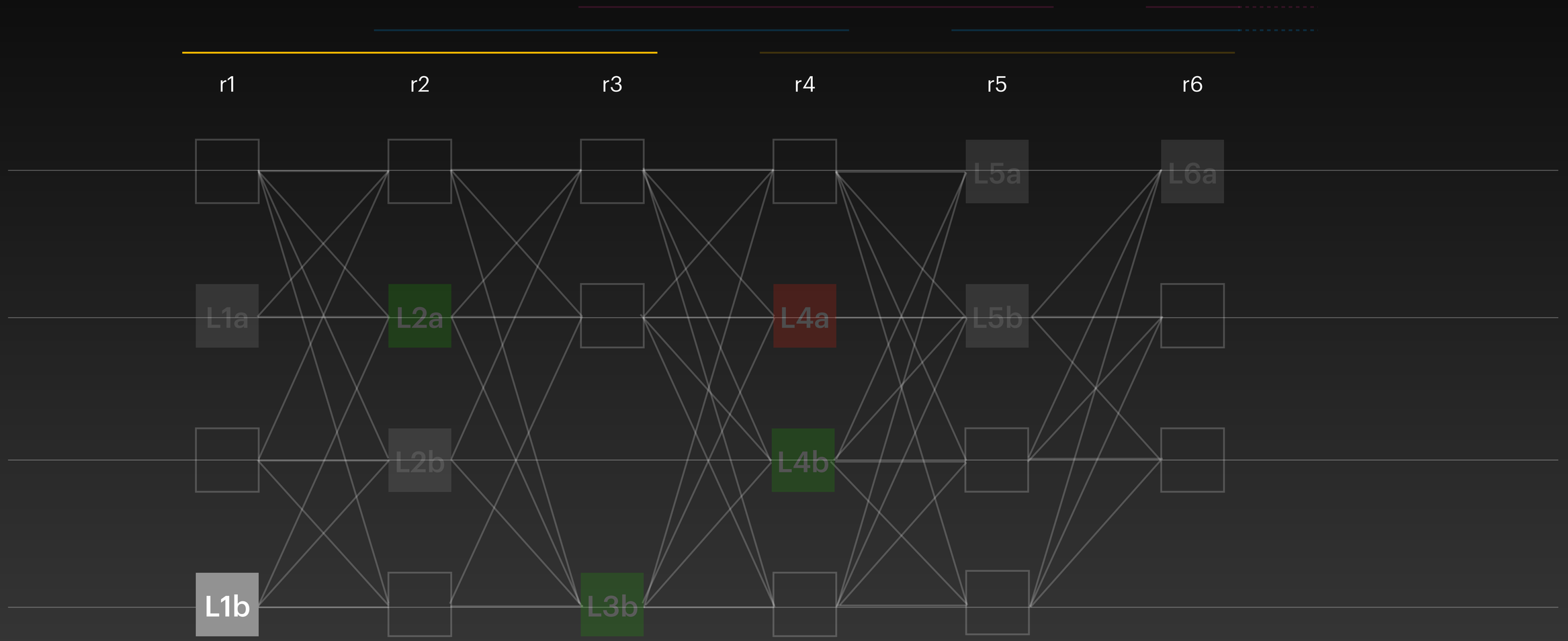
Apply Direct Rule



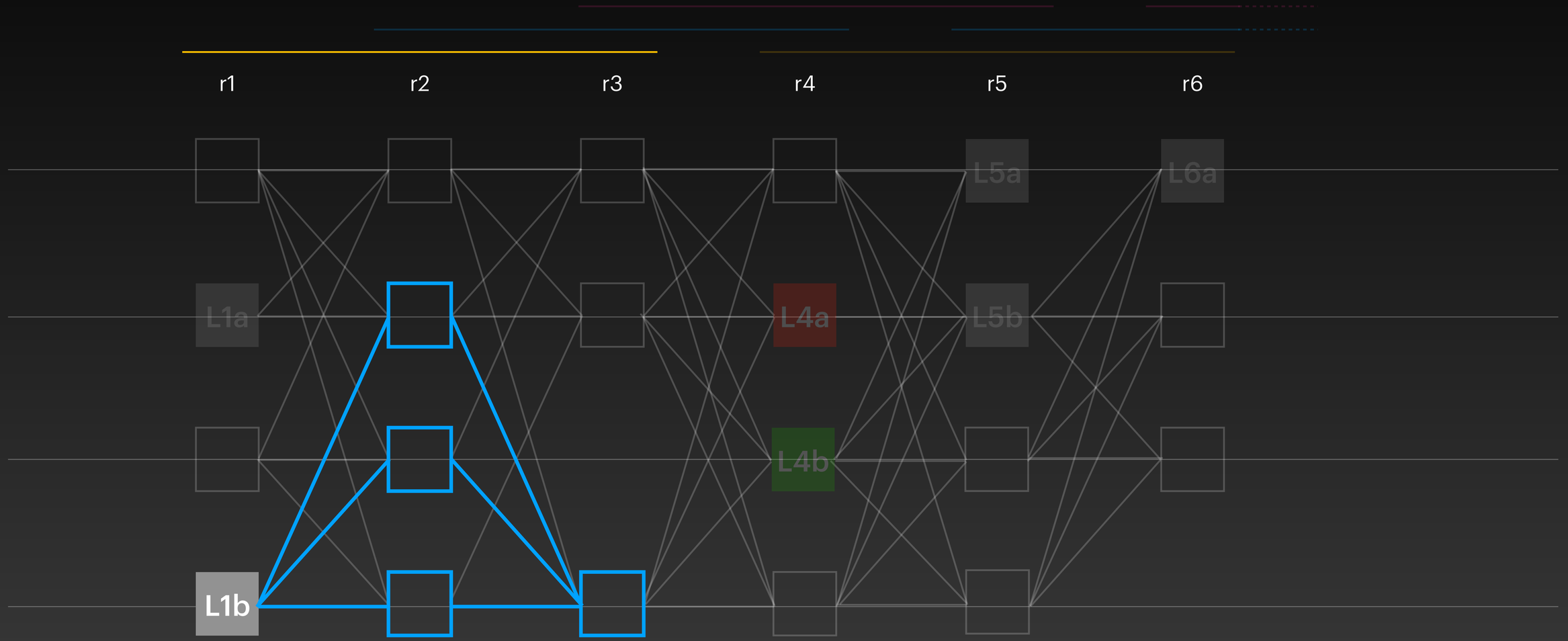
Apply Direct Rule



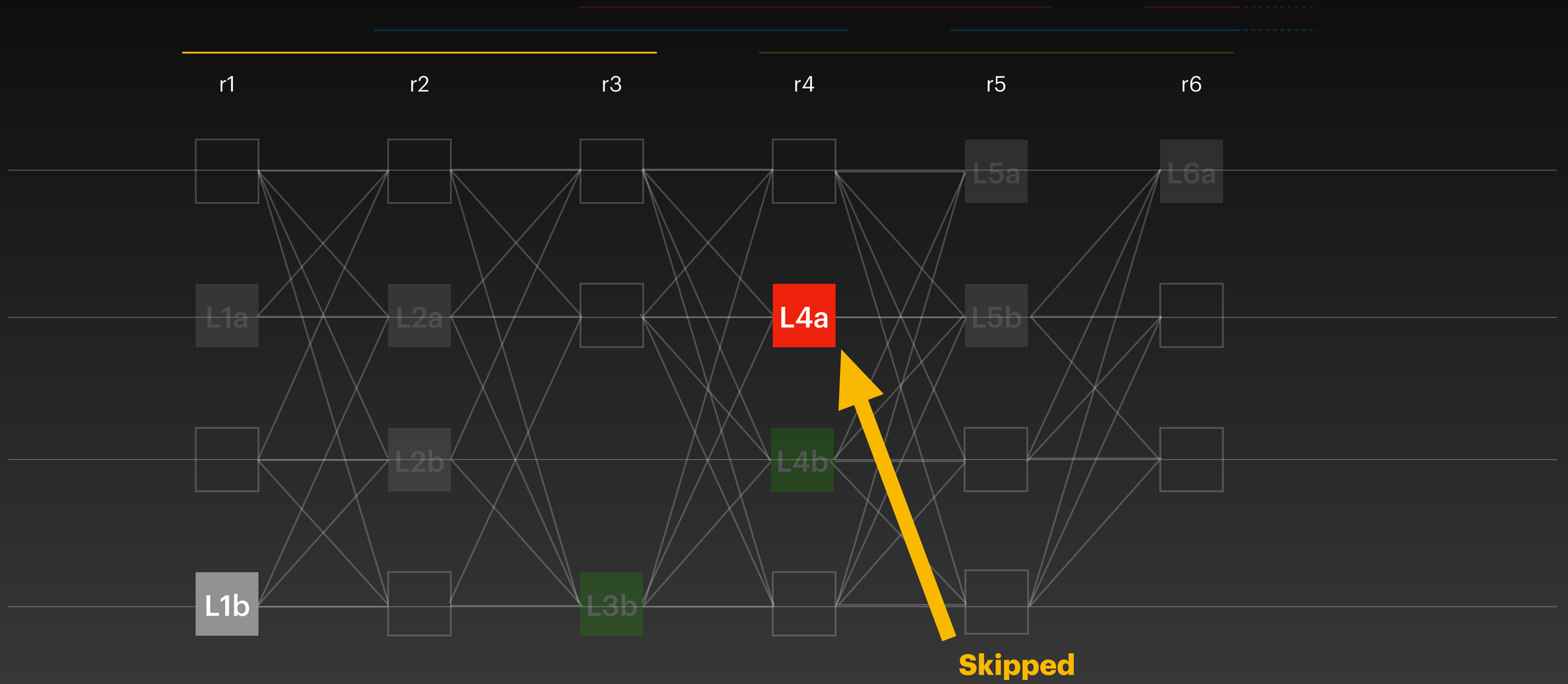
Apply Direct Rule



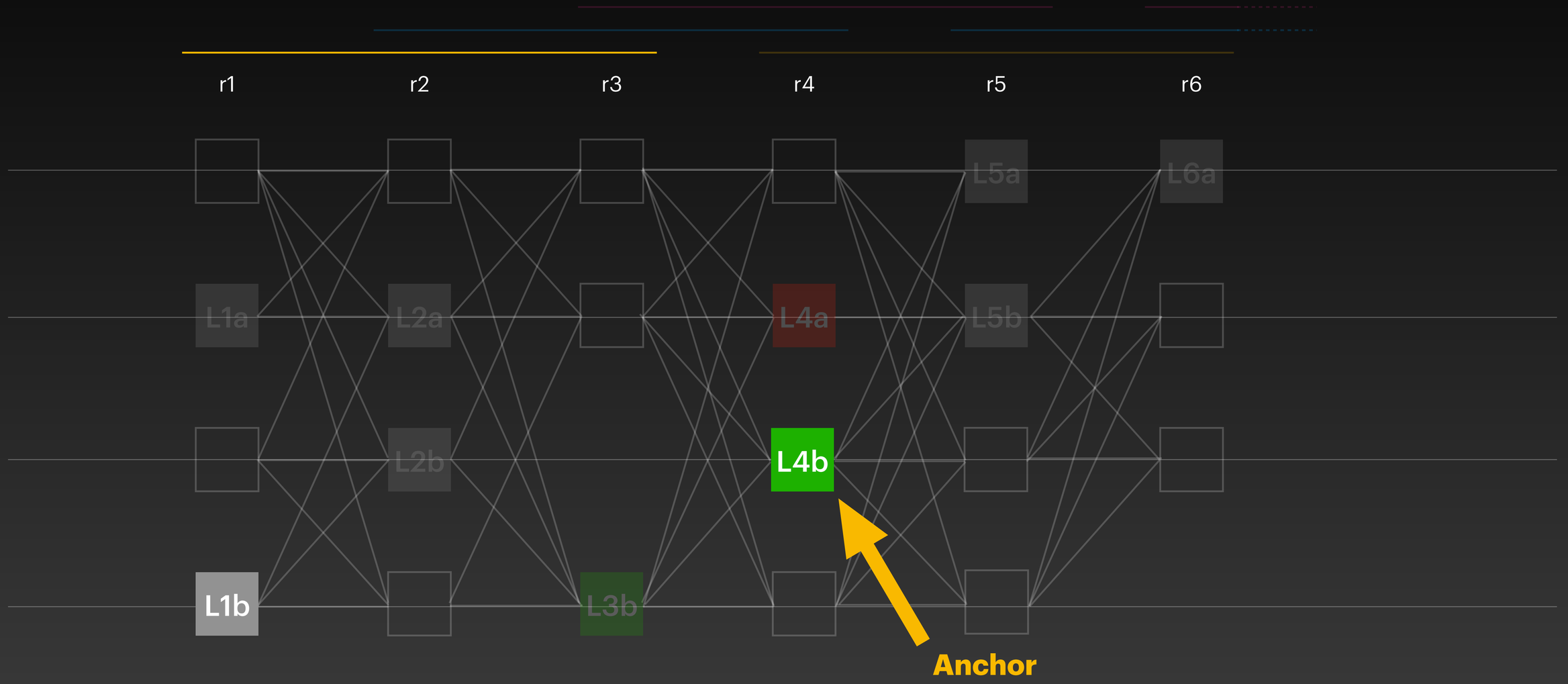
Apply Direct Rule



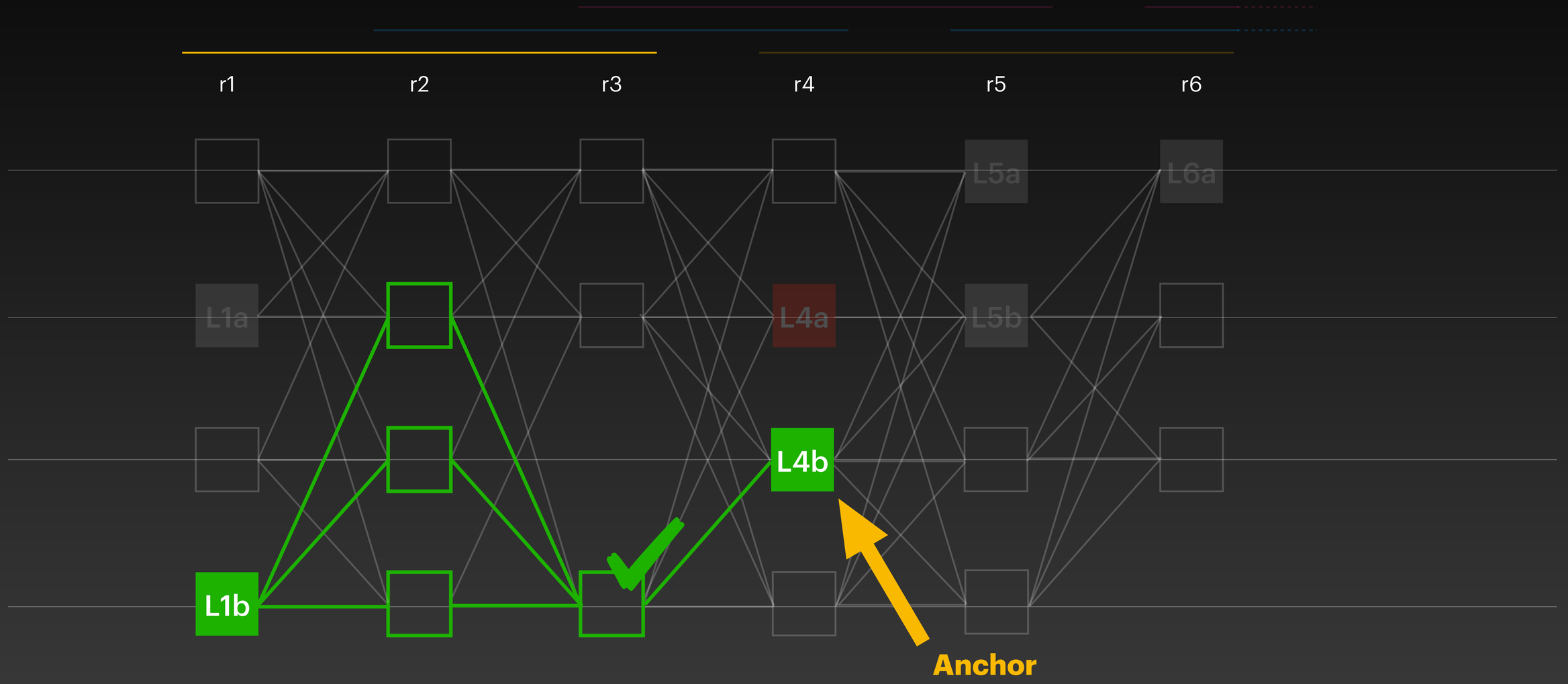
Apply Indirect Rule



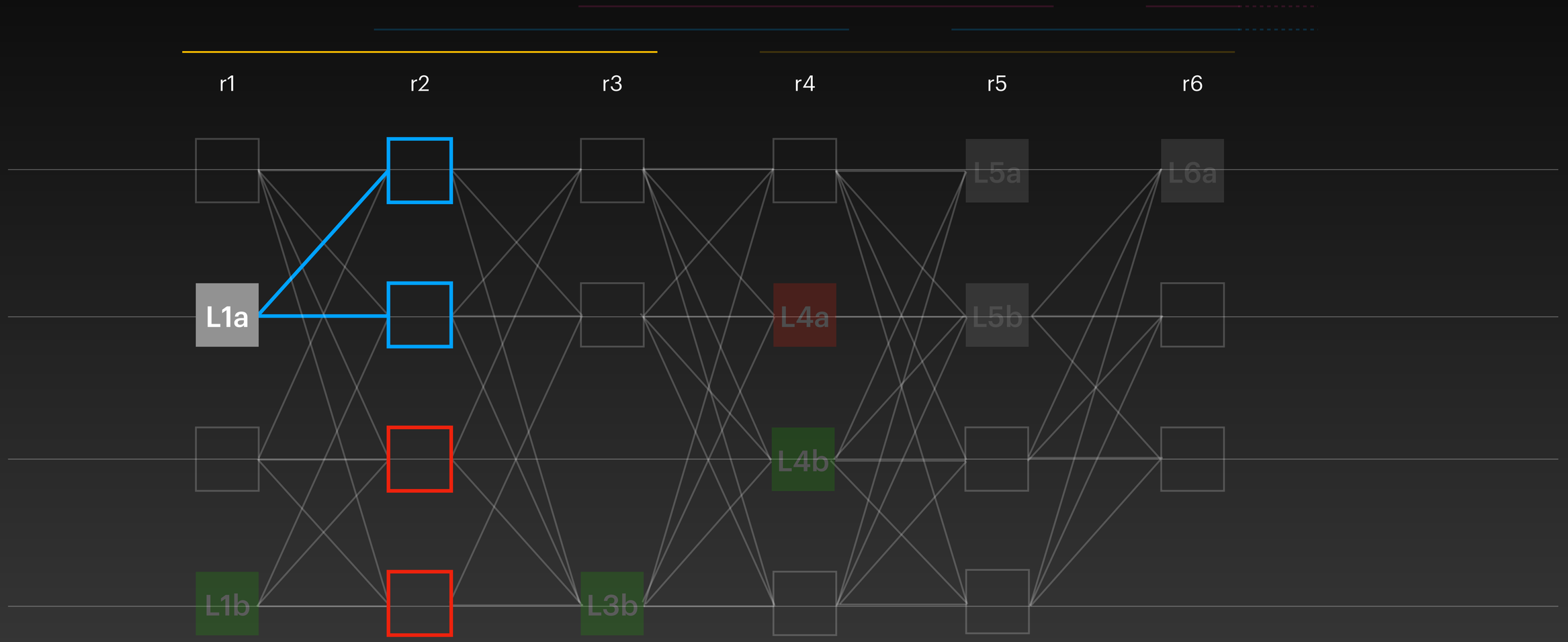
Apply Indirect Rule



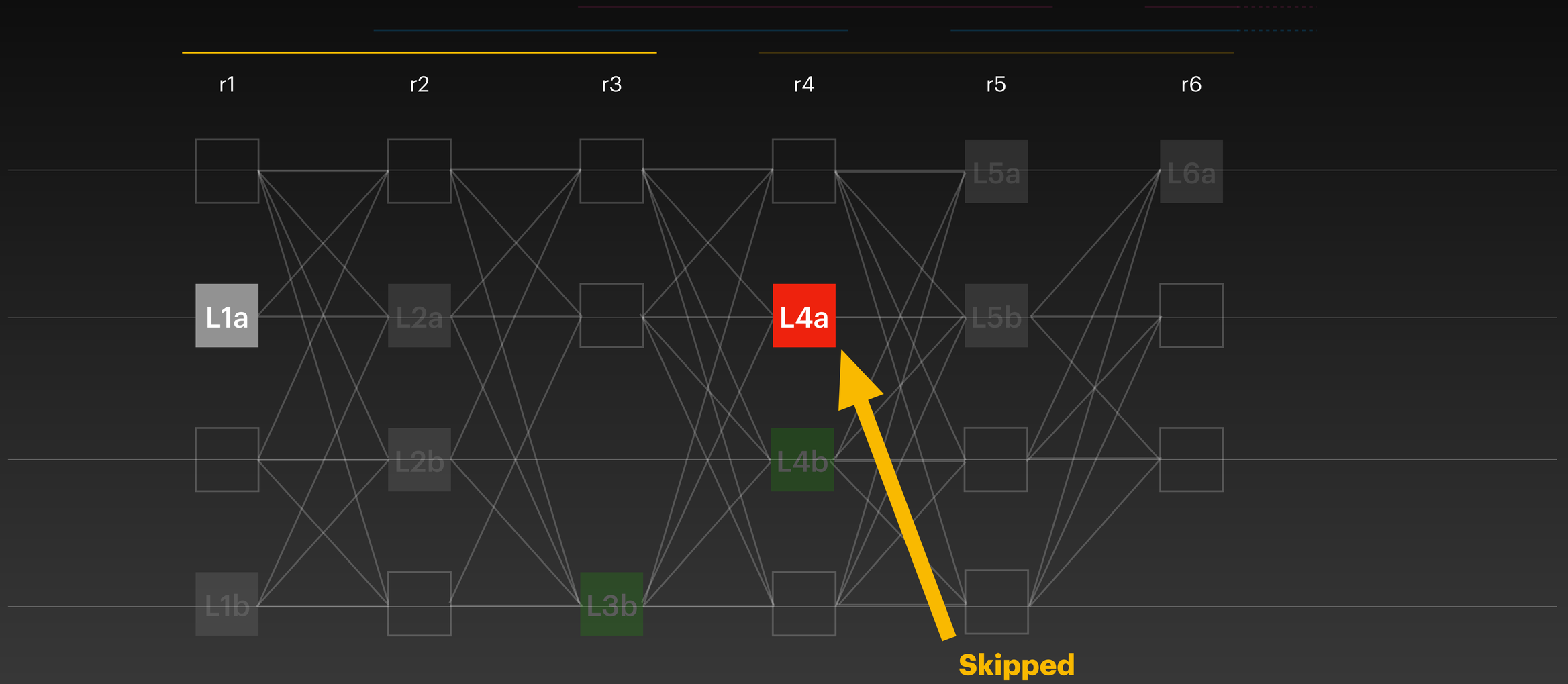
Apply Indirect Rule



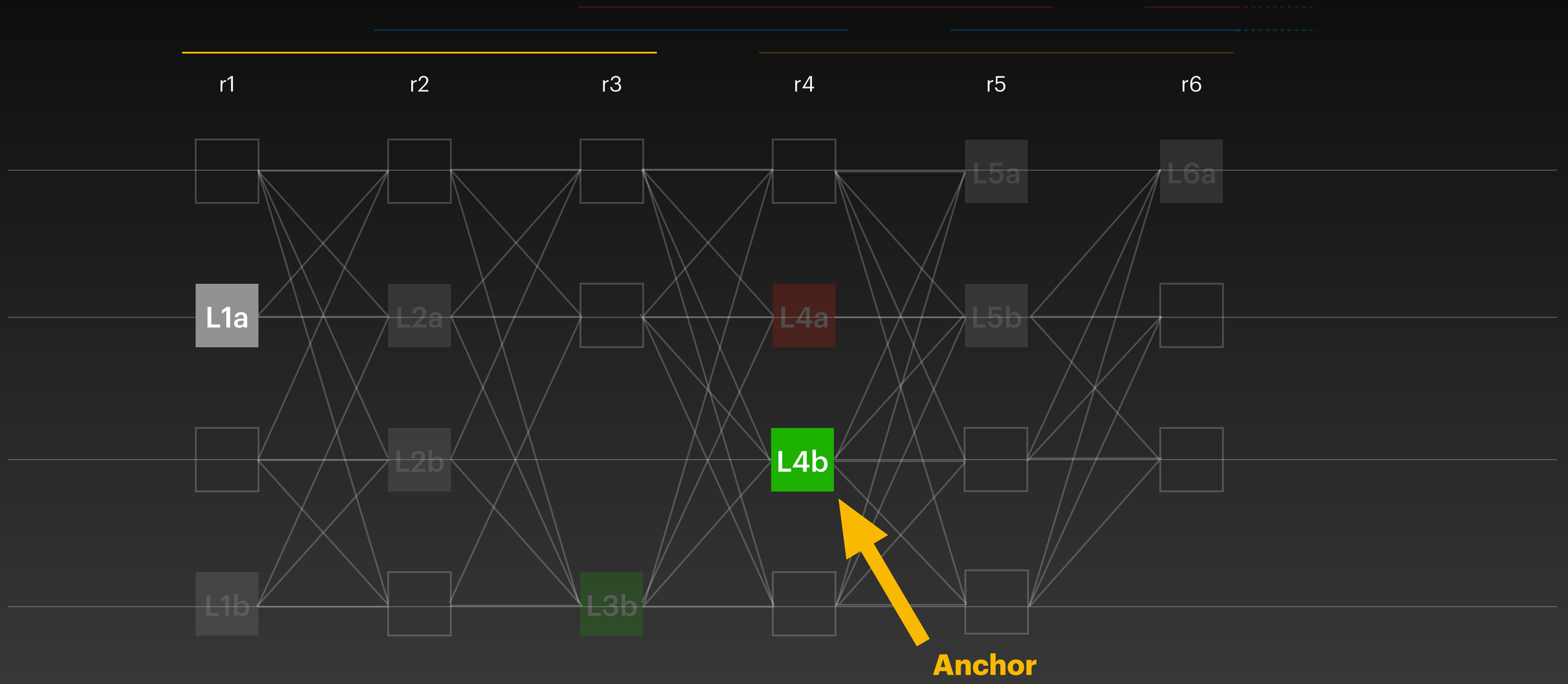
Apply Direct Rule



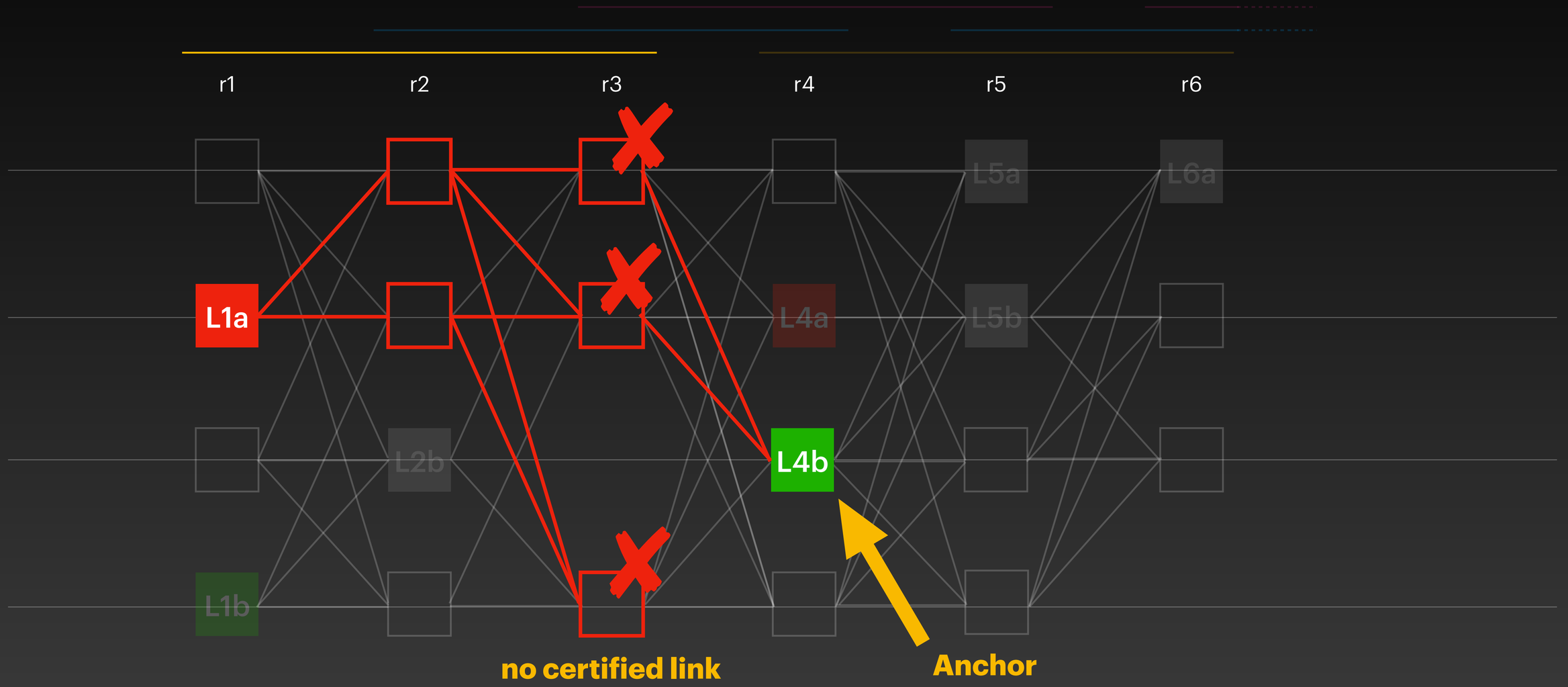
Apply Indirect Rule



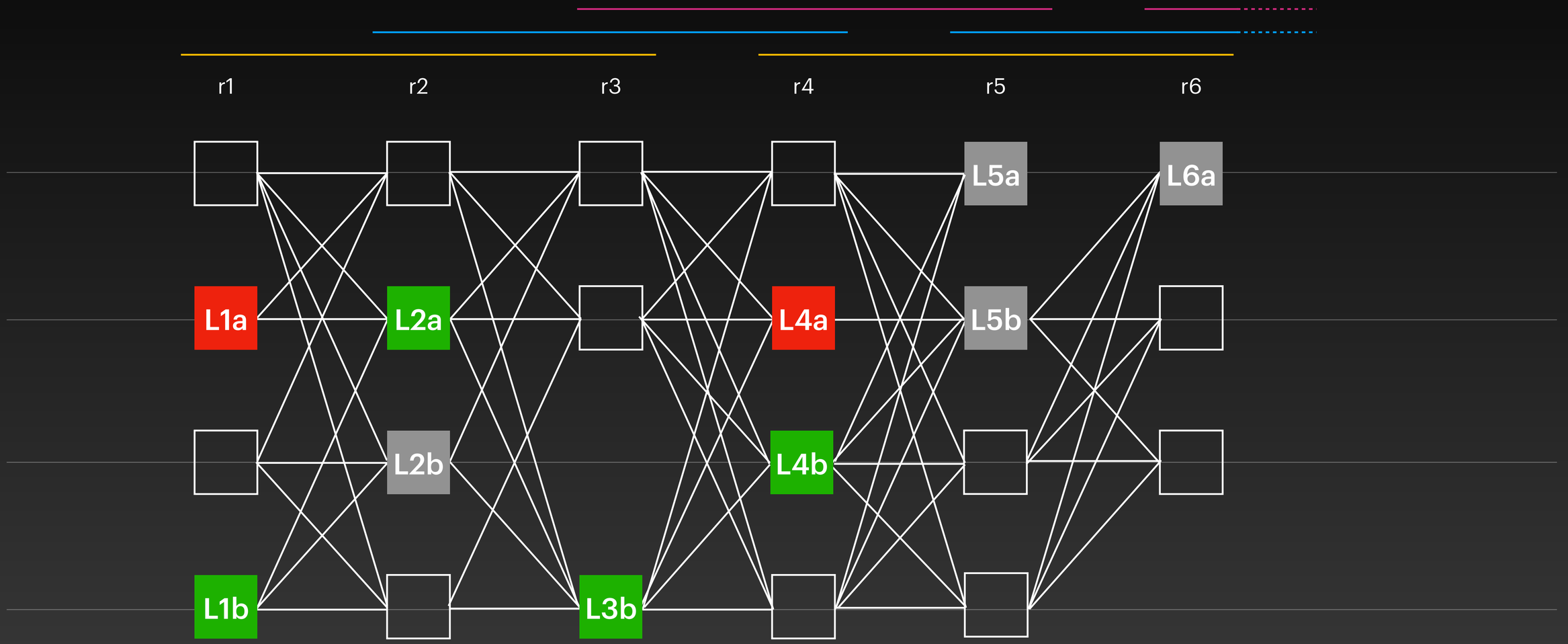
Apply Indirect Rule



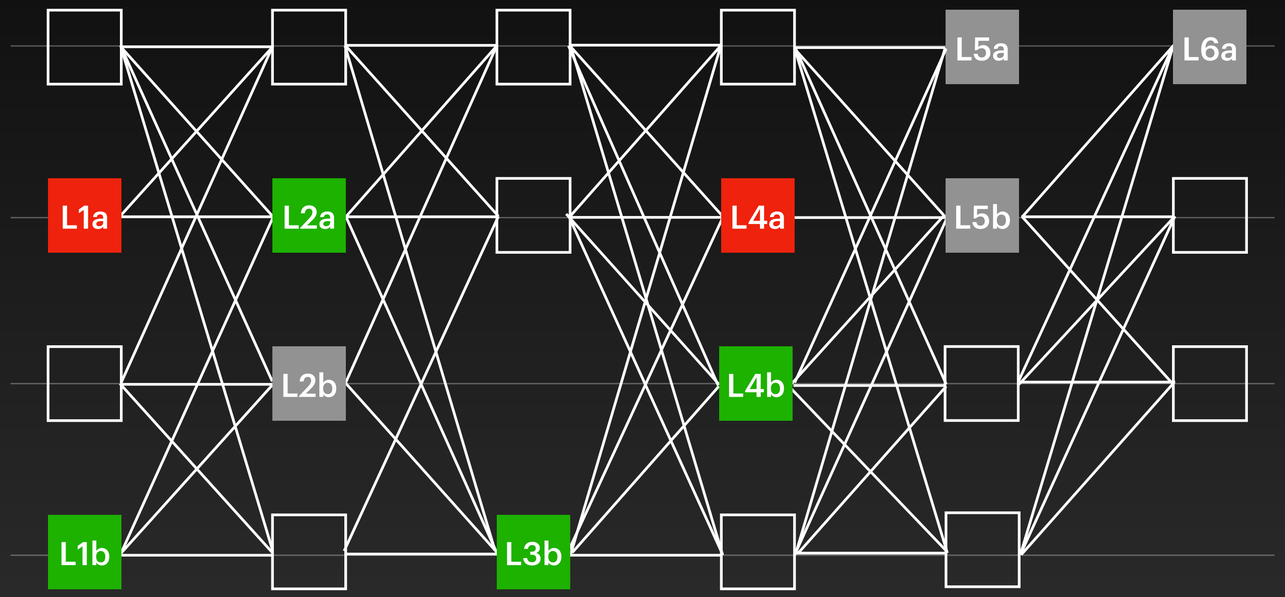
Apply Indirect Rule



Current Status

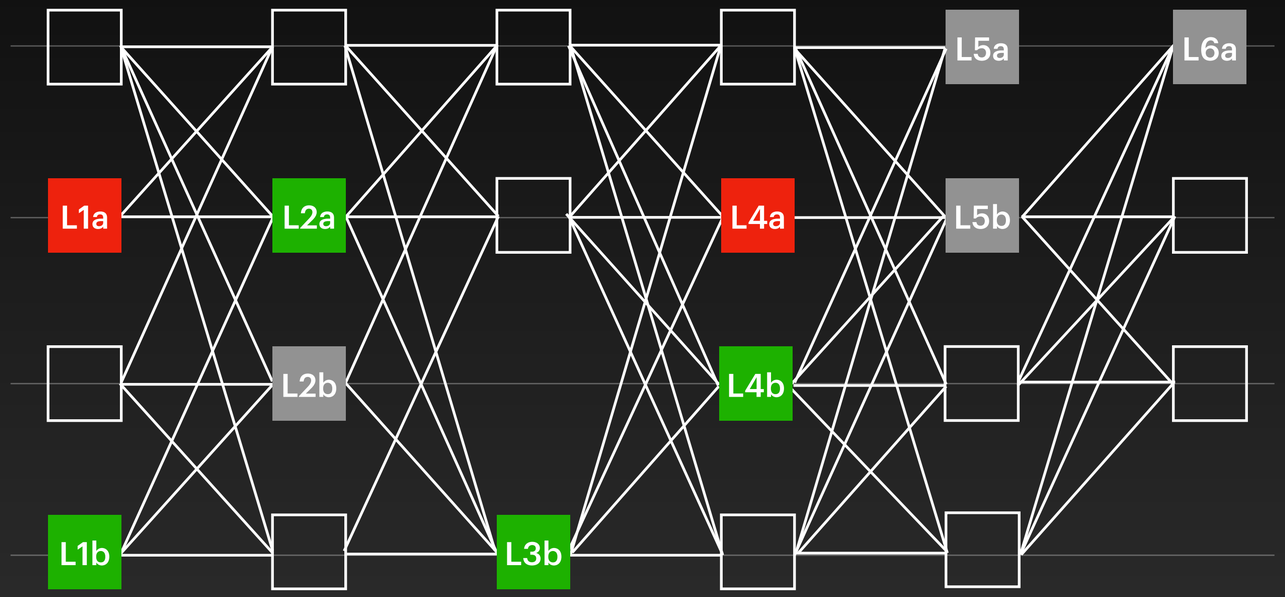


Take all leaders in order

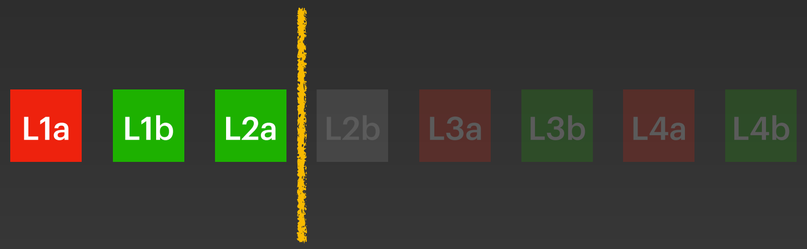


leaders sequence: L1a L1b L2a L2b L3a L3b L4a L4b

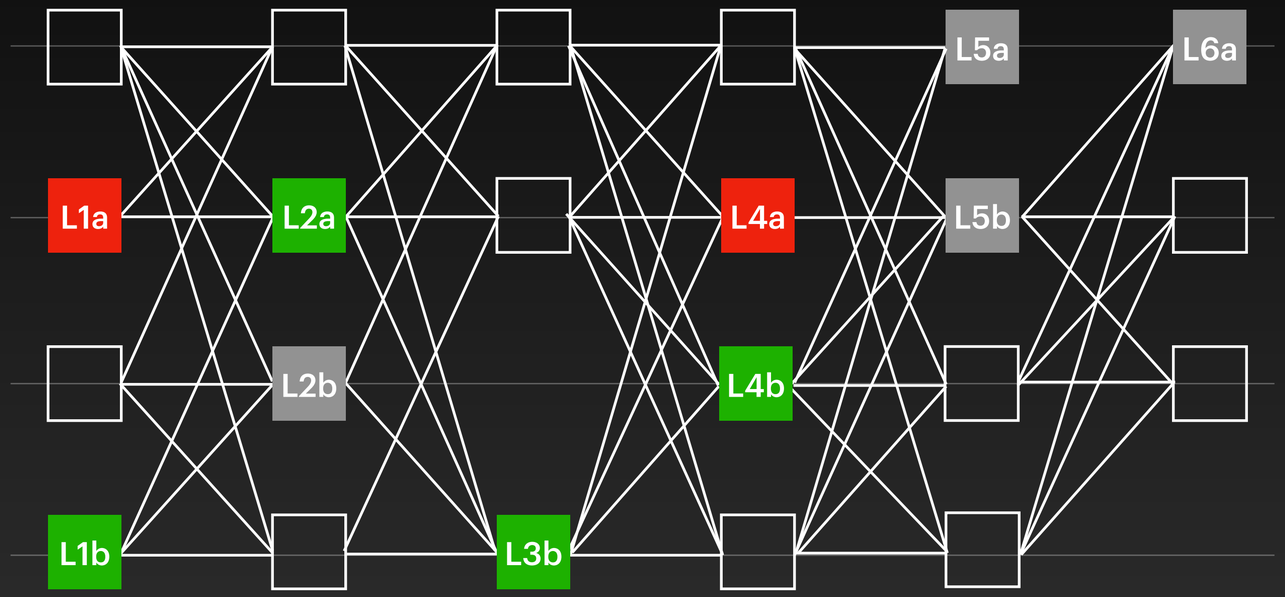
Stop at the first Undecided leader



leaders sequence:

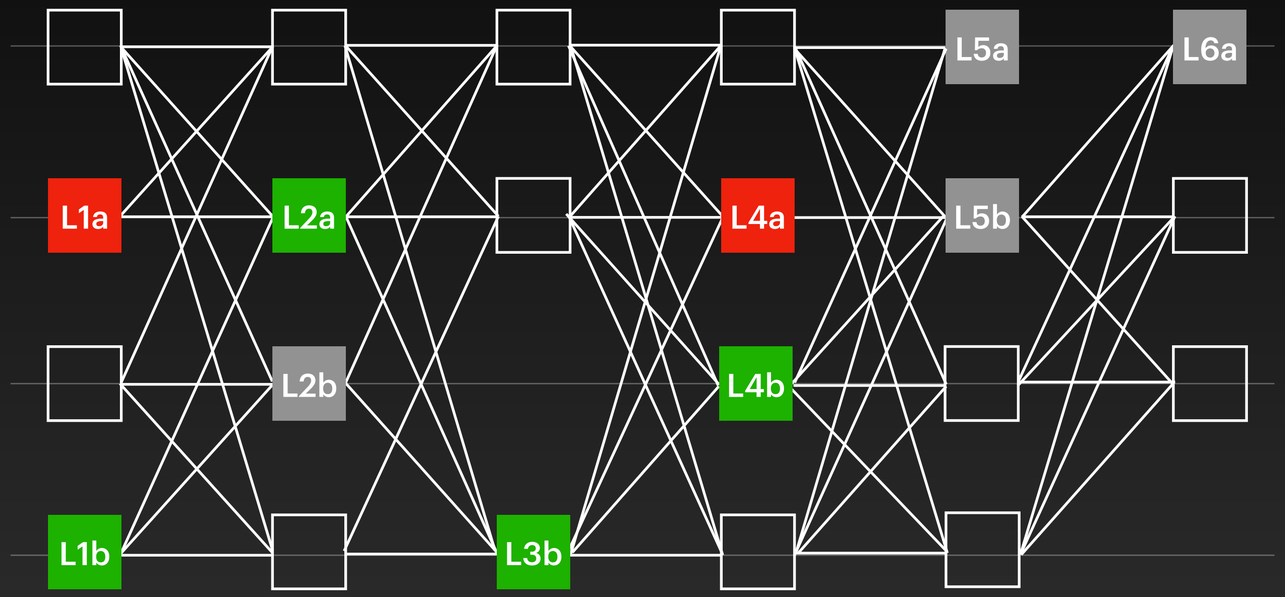


Remove skipped leaders



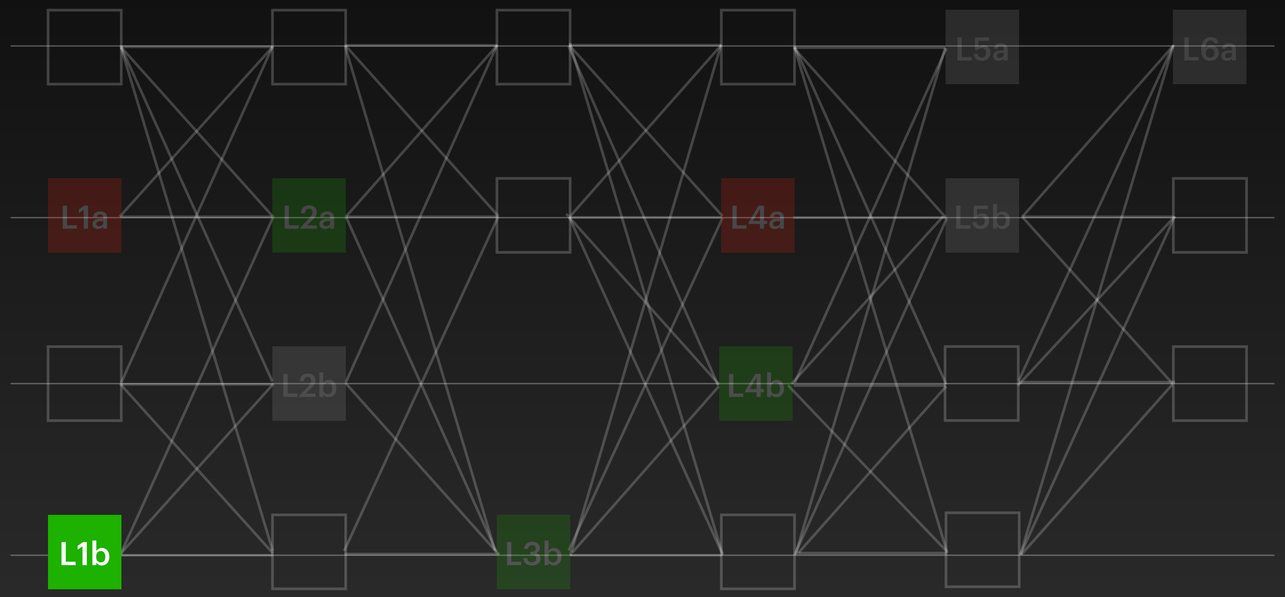
leaders sequence: L1a L1b L2a L2b L3a L3b L4a L4b

Final leader sequence



leaders sequence: L1b L2a

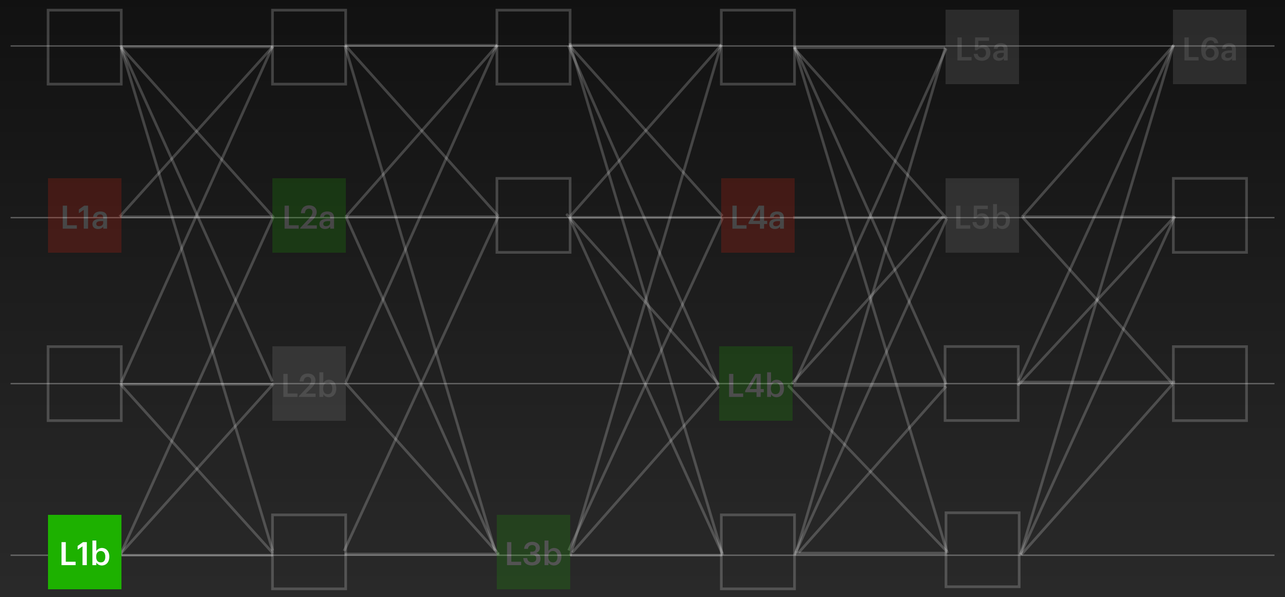
Commit sub-dag



leaders sequence: L1b L2a

output sequence:

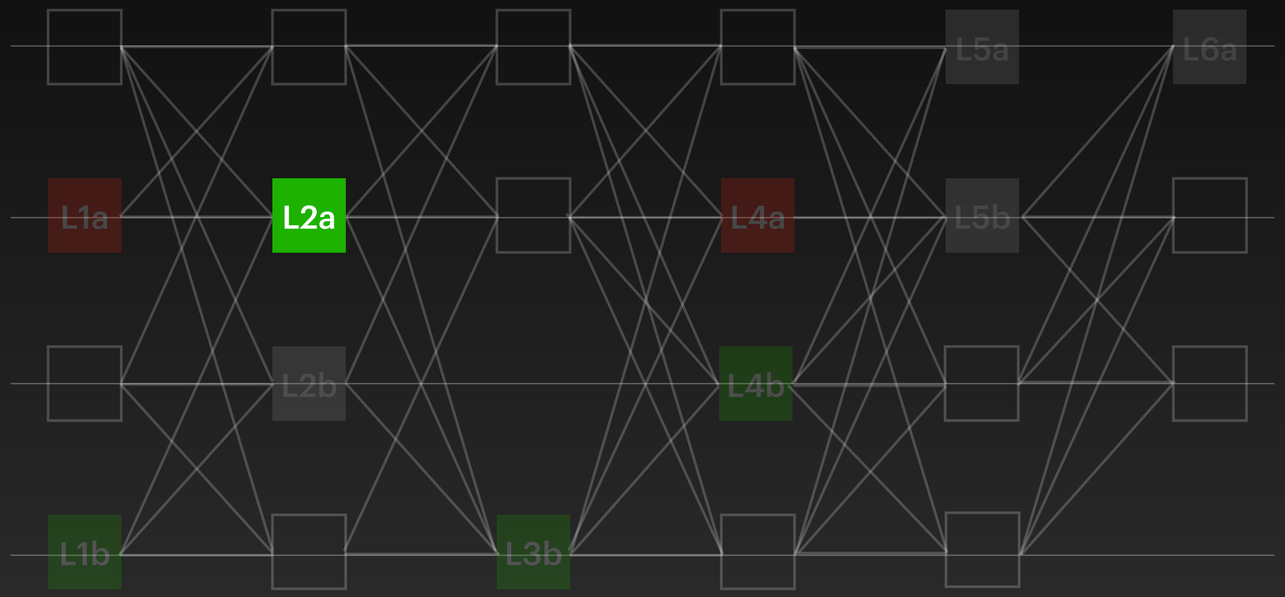
Commit sub-dag



leaders sequence: L2a

output sequence: L1b

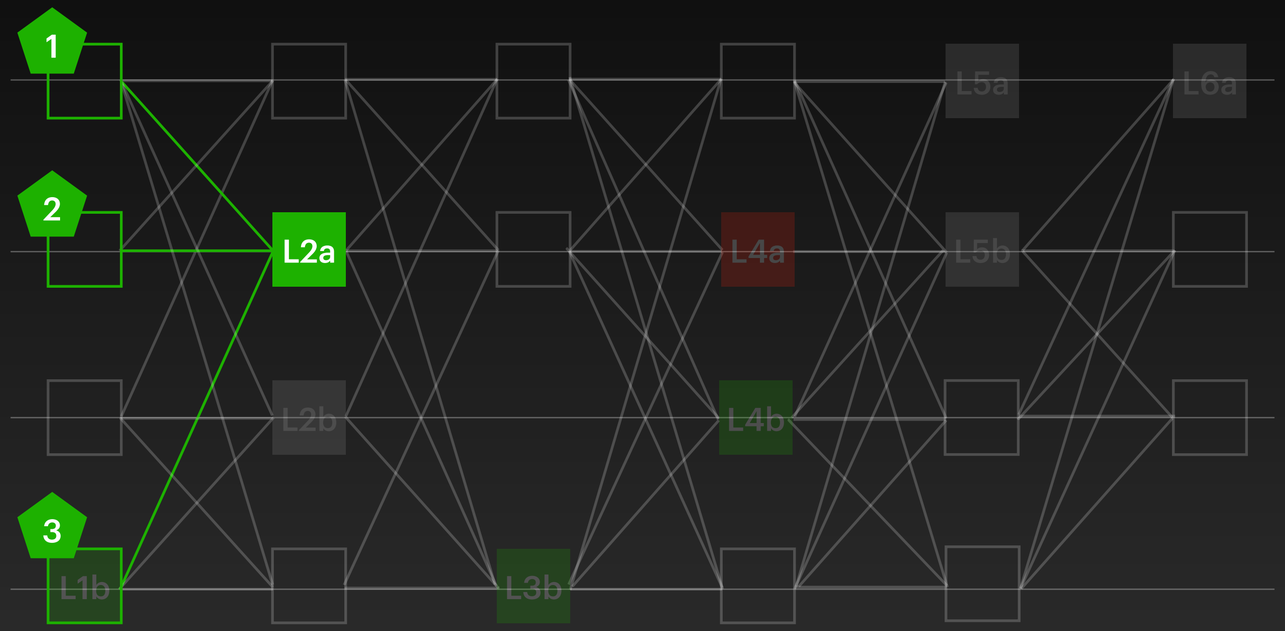
Commit sub-dag



leaders sequence: L2a

output sequence: L1b

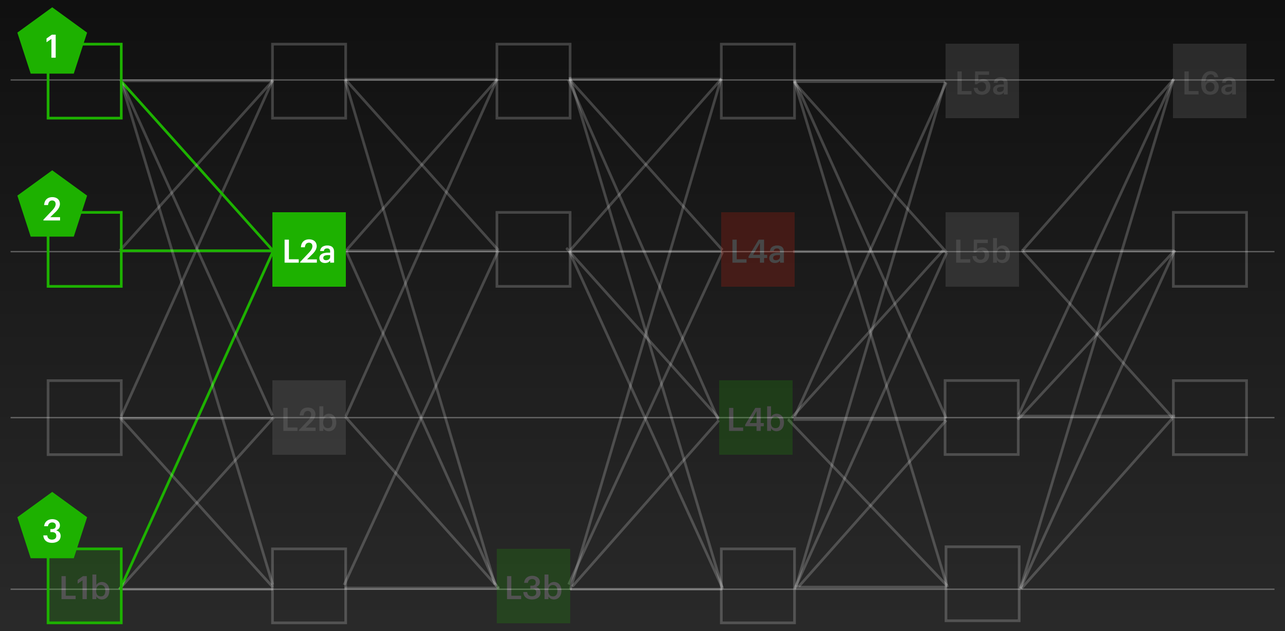
Commit sub-dag



leaders sequence: L2a

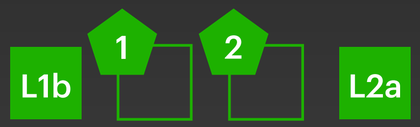
output sequence: L1b

Commit sub-dag

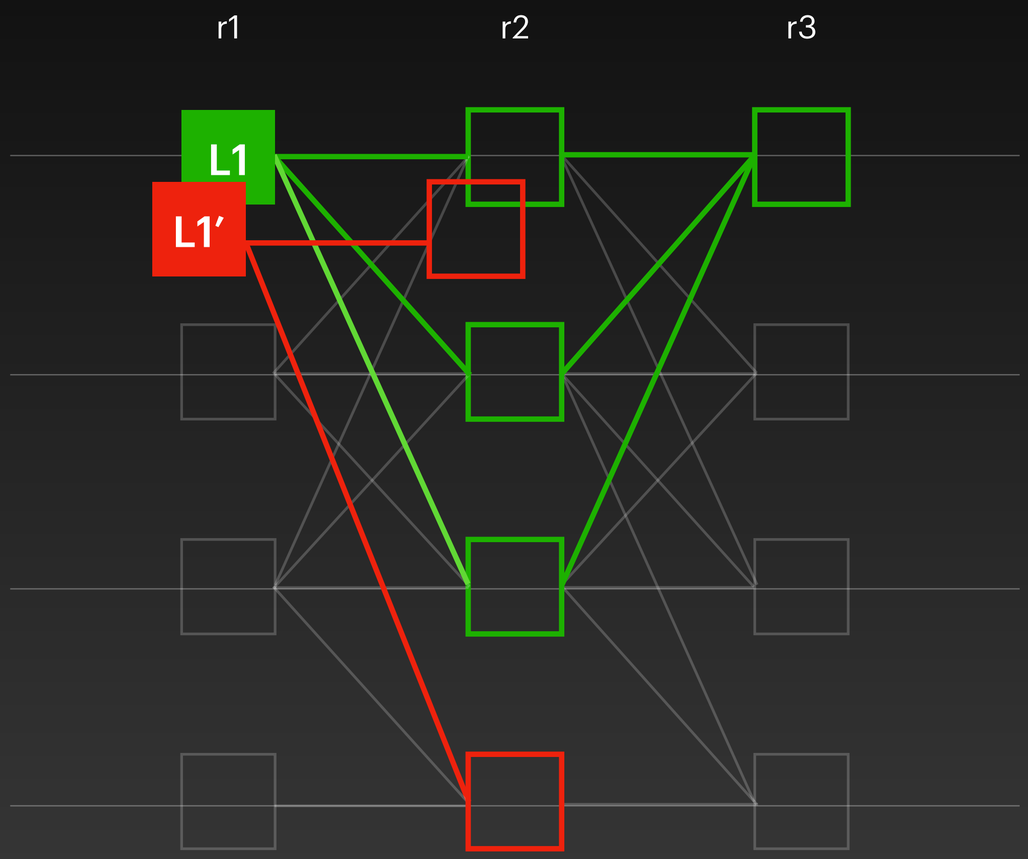


leaders sequence:

output sequence:

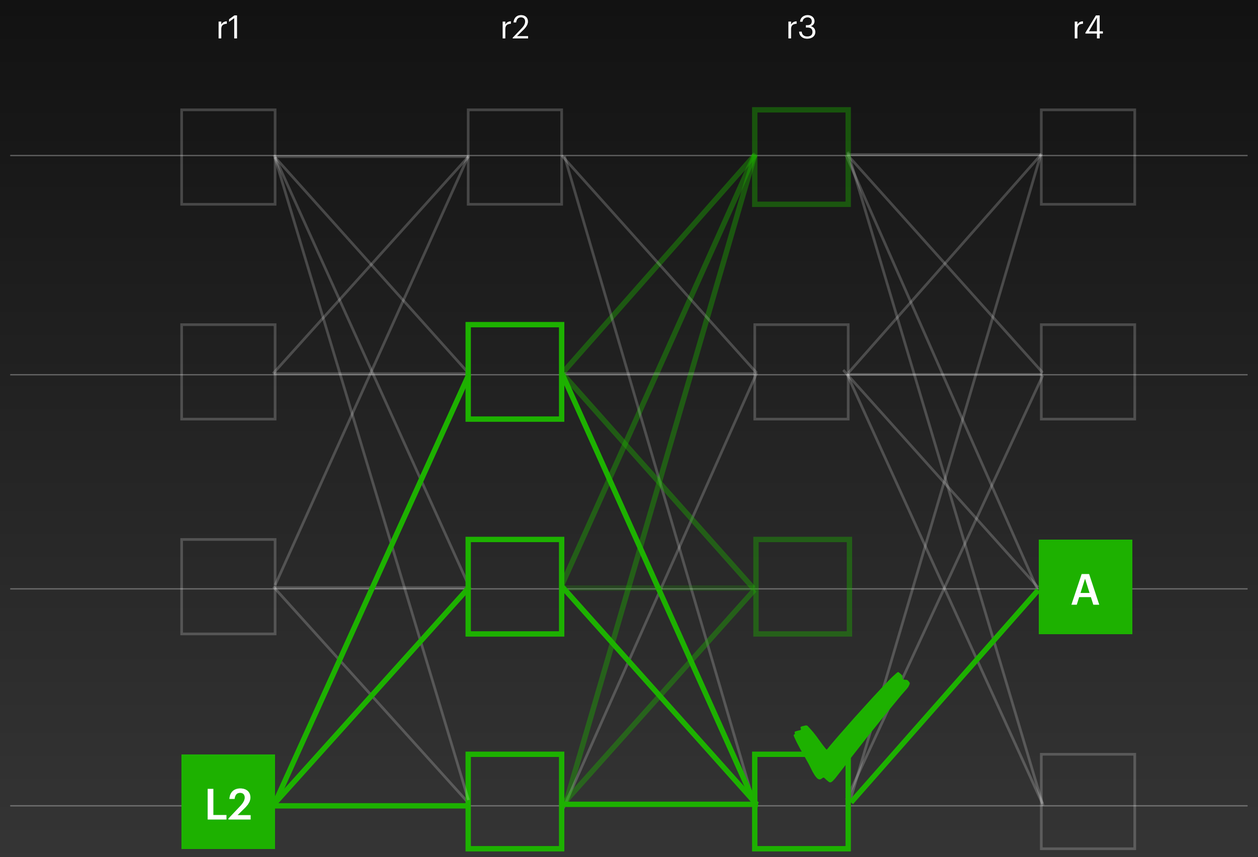


Safety



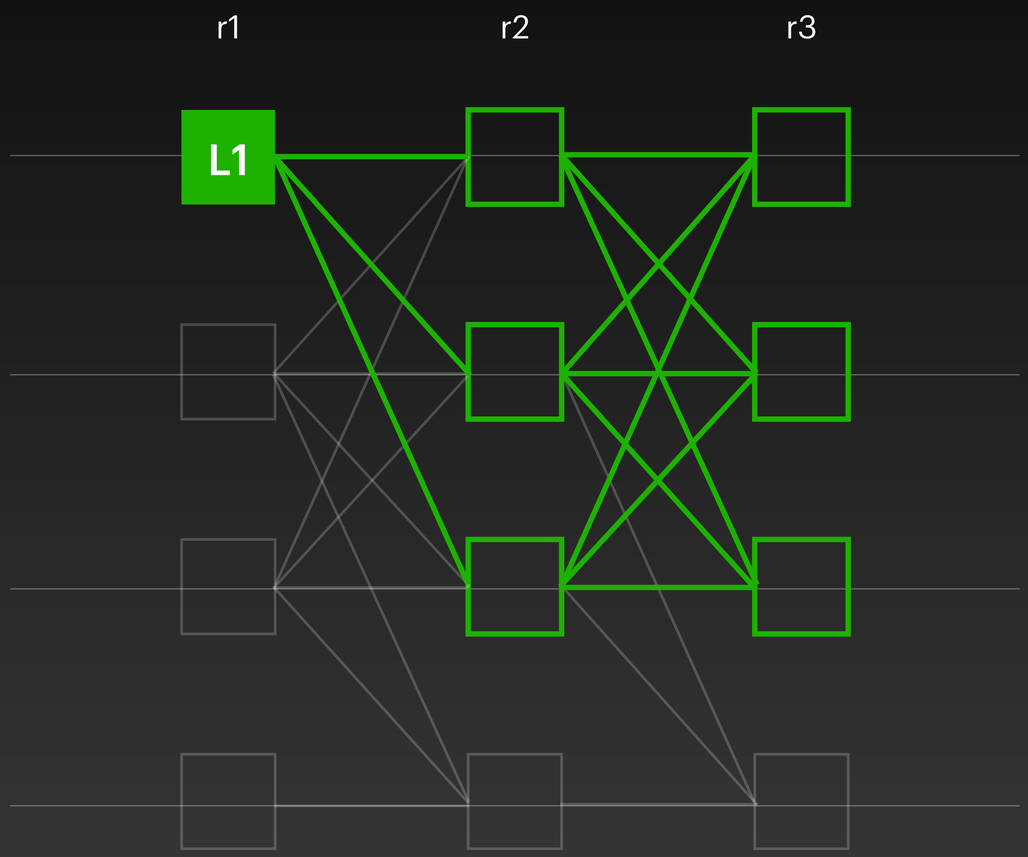
- At most **L1** or **L1’** can have a certificate pattern (quorum intersection)

Safety



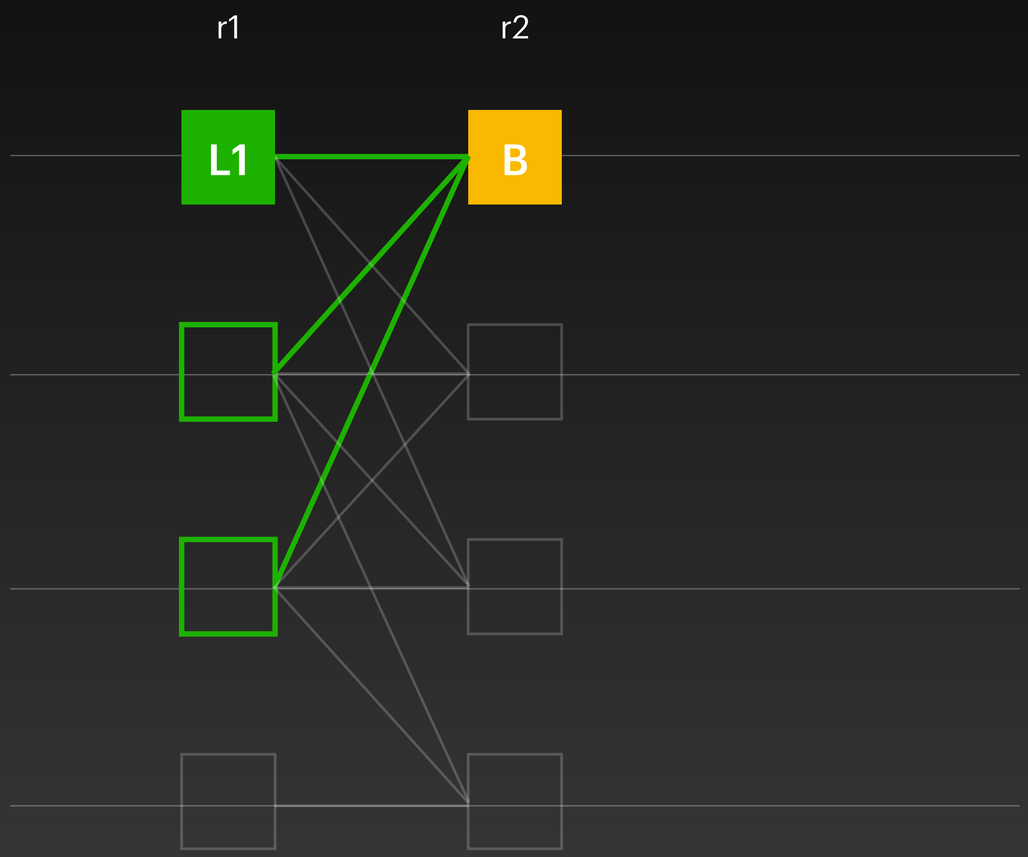
- At most **L1** or **L1'** can have a certificate pattern (quorum intersection
- If **L2** has 2f+1 certificate patterns, **A** always has a certified link to **L2**

Liveness



- After GST, the direct decision rule **commits** a block

Liveness



- After GST, the direct decision rule **commits** a block
- Leader timeout: **B** waits for 2f+1 parents + 250 ms

We are getting limited returns on algorithmic improvements of BFT consensus

controversial

High resource utilisation: throughput

Protocols commit often: latency

Engineering caught up with research

Change the assumptions (vs classic 3f+1)

Specialised chains (vs one chain to rule them all)

Other components are bottleneck (network, execution, etc)

Block Synchroniser

Theory
Reliable Broadcast • Proactive dissemination

Prototype
Optimistic dissemination • Random pull on a timer (if needed)

Production
Optimistic dissemination • Proactive re-sharing on timer • Bulk-sync of past commits

Why DAGs

Narwhal (sections 1-4) • https://sonnino.com/papers/narwhal-and-tusk.pdf

Robustness & Throughput

Bullshark (partially synchronous) • https://sonnino.com/papers/bullshark-simple.pdf

Latency

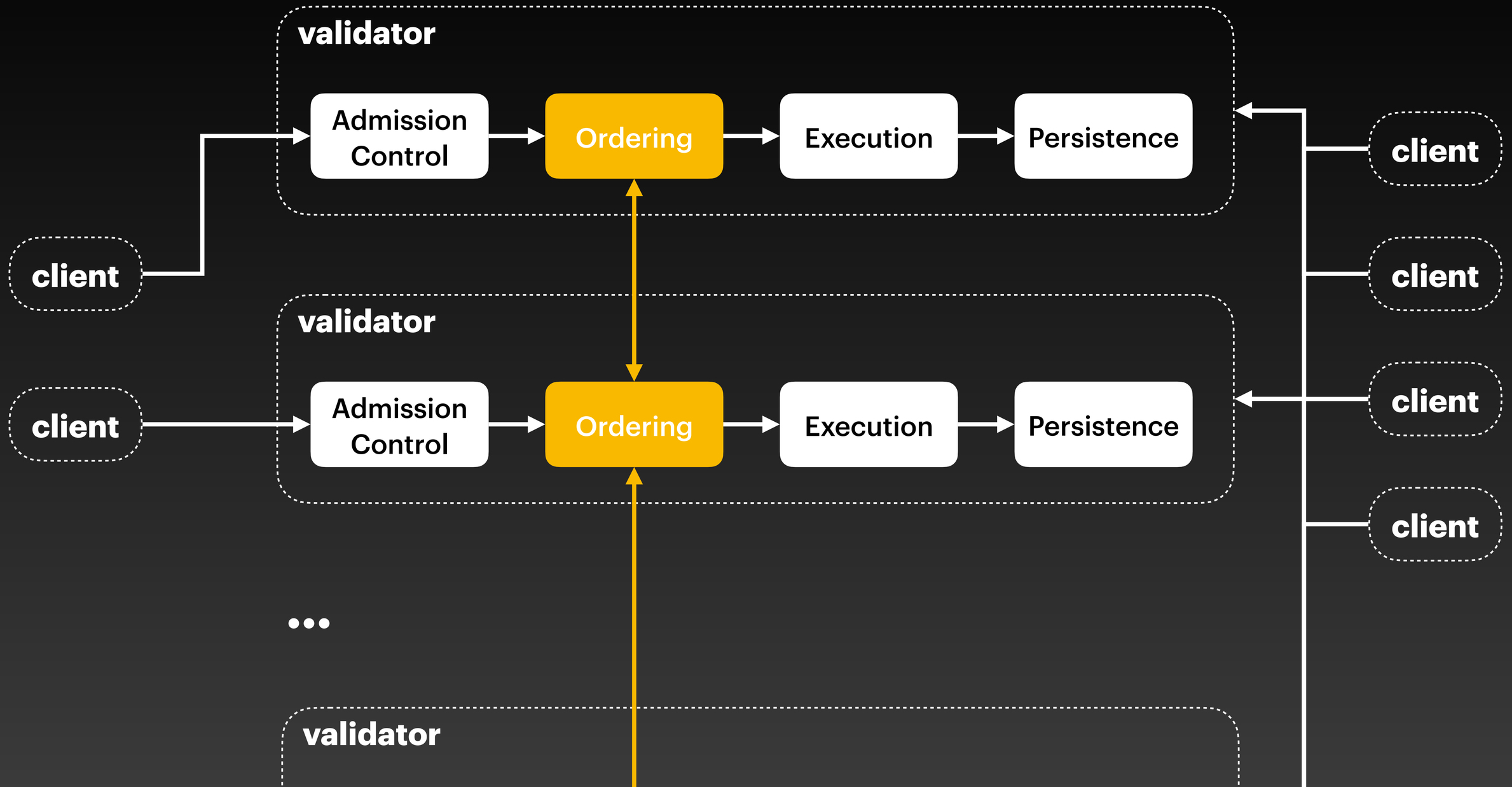
Shoal++ • https://arxiv.org/pdf/2405.20488v1

Uncertified DAG

Mysticeti (skip section 5) • https://sonnino.com/papers/mysticeti.pdf

Unstructured DAG

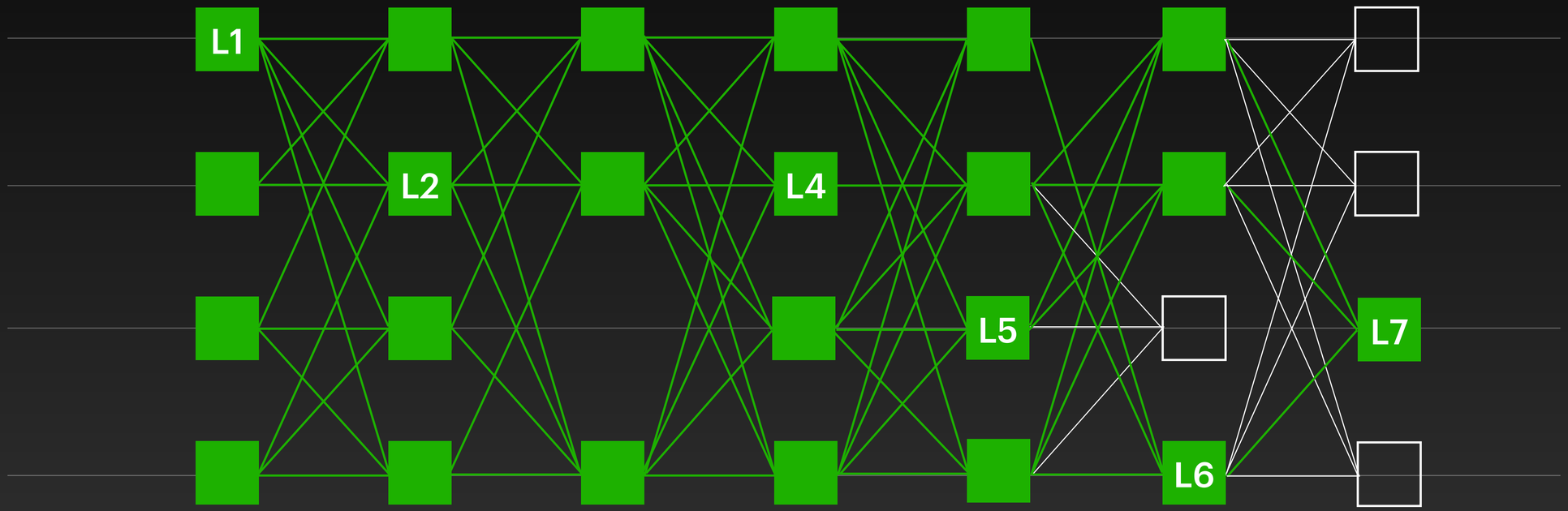
Autobahn • https://arxiv.org/pdf/2401.10369


validator
Admission Control
Ordering
Execution
Persistence
client
validator
Admission Control
Ordering
Execution
Persistence
client
validator
client
client
client
client
...

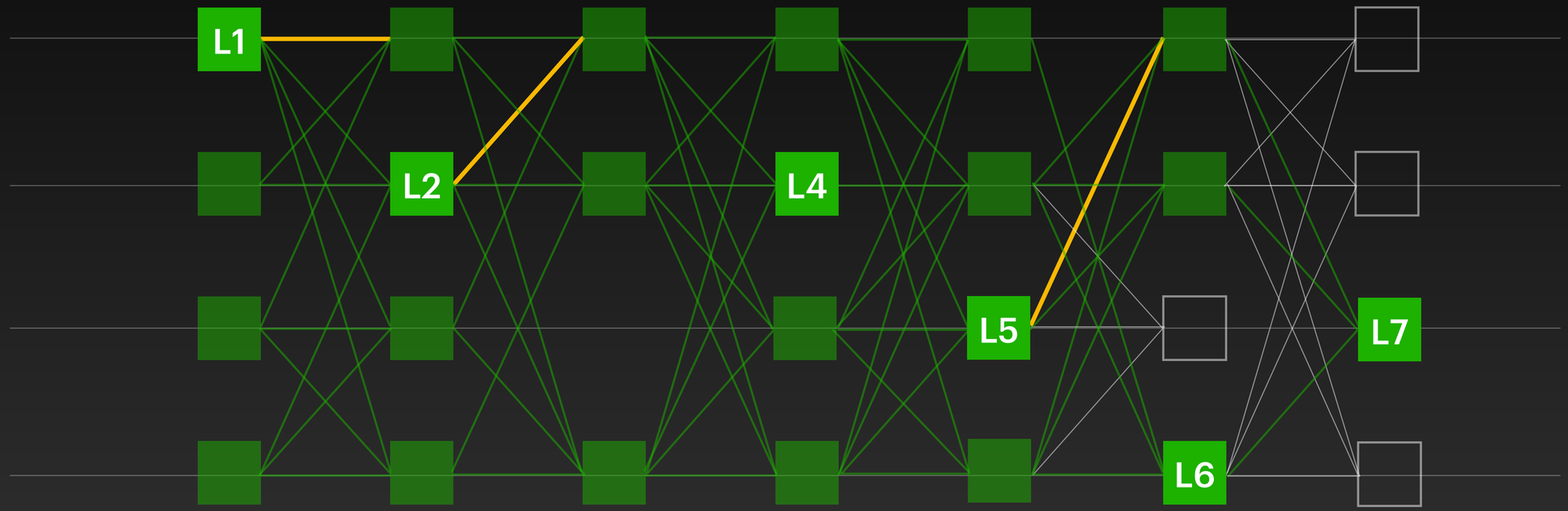
alberto@mystenlabs.com

EXTRA:
Leader Reputation

Past Commits

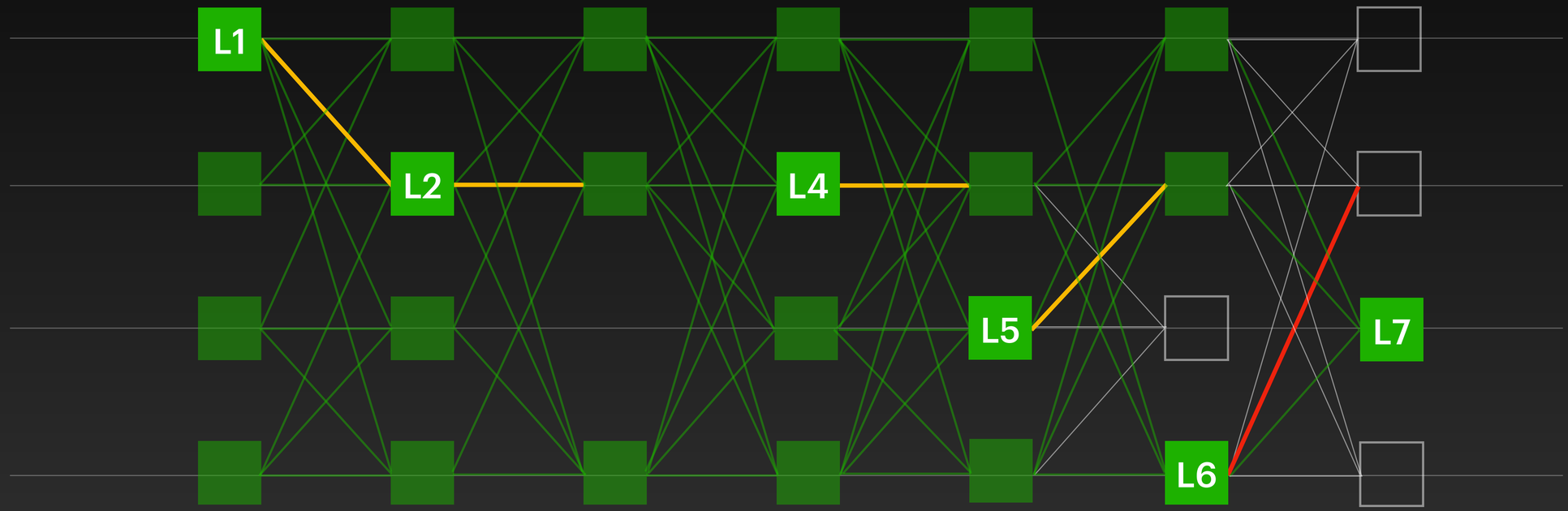


Compute Reputation Scores



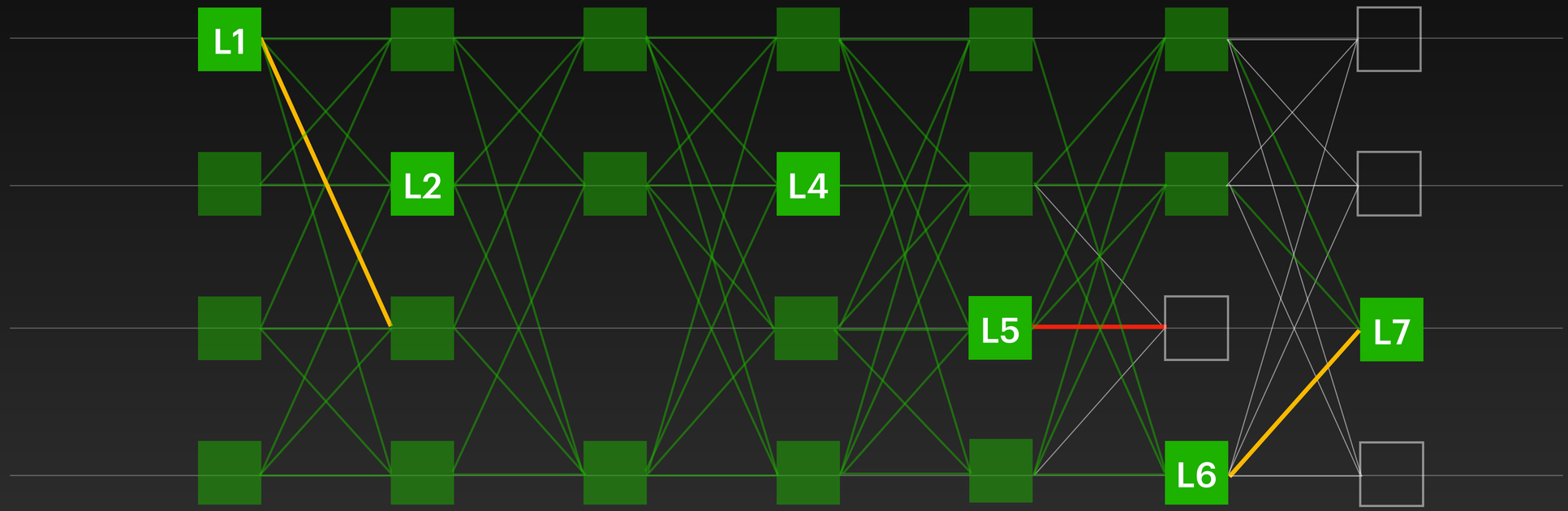
node 1: 3

Compute Reputation Scores



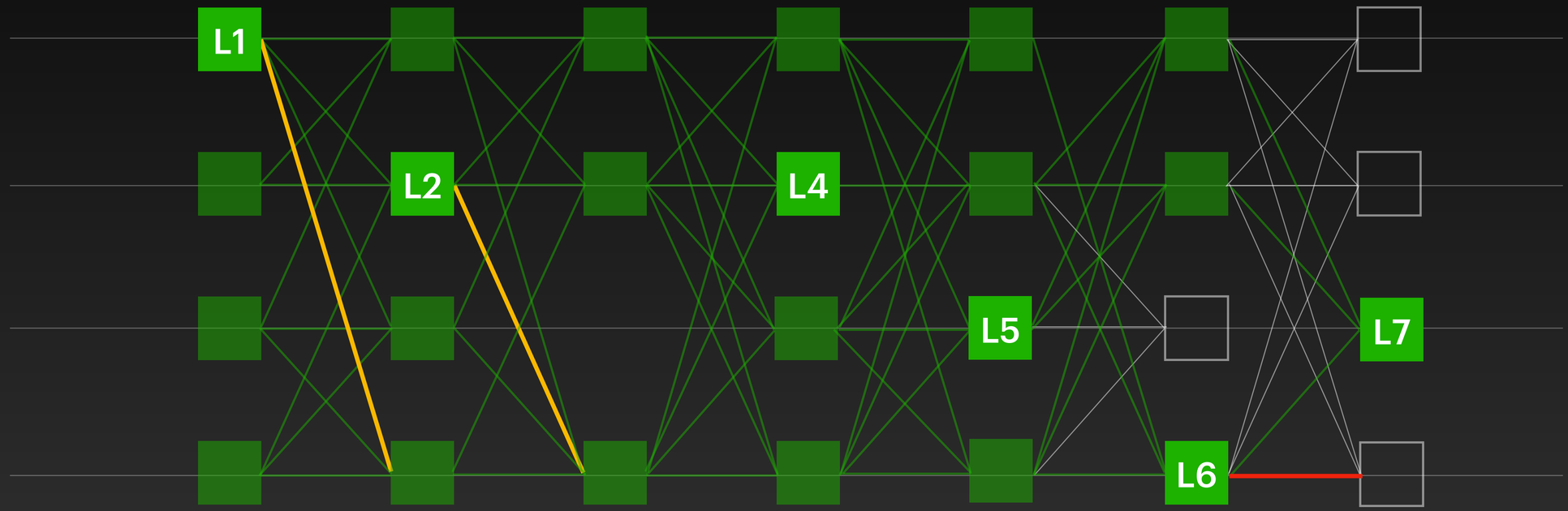
node 1: 3 **node 2:** 4

Compute Reputation Scores



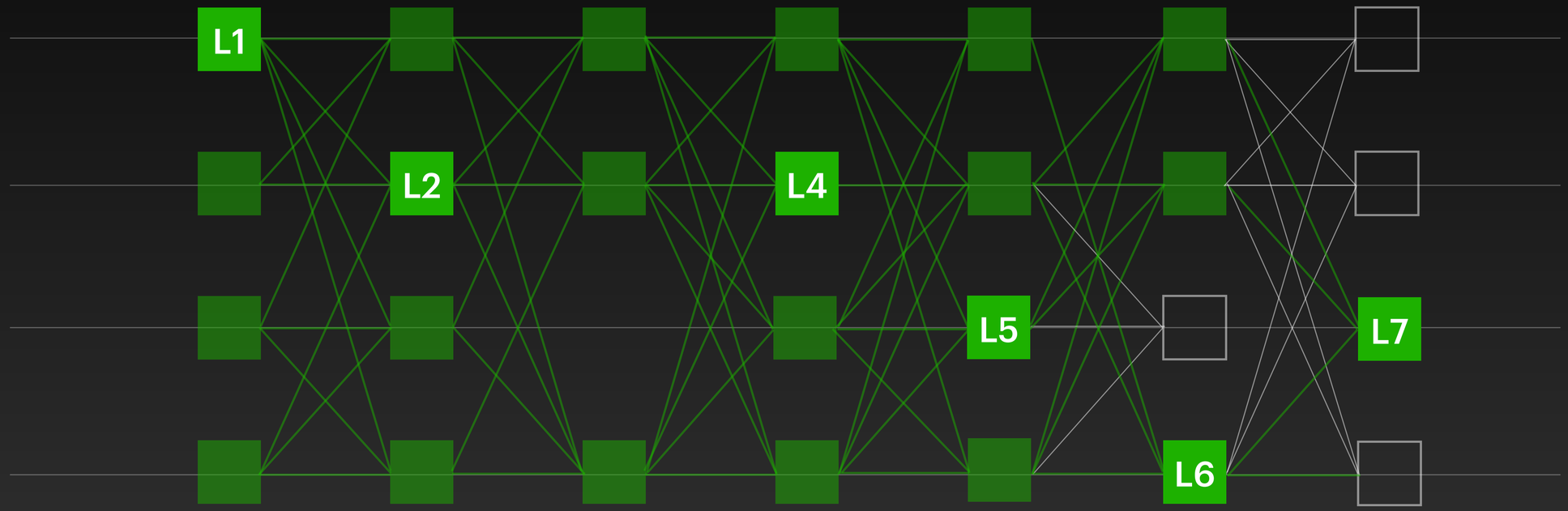
node 1: 3 node 2: 4 node 3: 2

Compute Reputation Scores



node 1: 3 node 2: 4 node 3: 2 node 4: 2

Future Leaders



node 1: 3 node 2: 4 node 3: 2 node 4: 2