Creació de graf de transaccions de la blockchain de Bitcoin

Francesc Escolà Bargués

Resum—Aquest Treball de Fi de Grau (TFG) té com a objectiu el desenvolupament d'una eina capaç de representar i analitzar la blockchain de Bitcoin mitjançant estructues de graf. Per tal d'abordar el problema d'escalabilitat i complexitat de la xarxa, es proposa una estratègia híbrida que combina la construcció d'un graf de blocs, que representa les dependències entre blocs a partir de les transaccions que contenen, i un graf de transaccions parcial, que permet analitzar la traçabilitat del bitcoin des d'una transacció inicial. Els resultats obtinguts mostren el potencial d'aquest enfocament per a l'anàlisi forense, detecció de patrons i visualització de fluxos de bitcoins de manera escalable i eficient.

Paraules clau— Bitcoin, blockchain, graf de blocs, graf de transaccions, Neo4J, traçabilitat

Abstract— This Final Degree Project aims to develop a tool capable of representing and analyzing the Bitcoin blockchain using graph-based structures. To address scalability and complexity issues, a hybrid strategy is proposed combining the construction of a block graph, representing transactional dependencies between blocks, and a partial transaction graph that allows tracing bitcoins from an initial transaction. The results demonstrate the effectiveness of this approach for forensic analysis, pattern detection, and scalable visualization of Bitcoin flows.

Index Terms— Bitcoin, blockchain, block graphs, transaction graphs, Neo4J, traceability

1 Introducció - Context del treball

BITCOIN va aparèixer l'any 2009, des de aleshores s'ha consolidat com la criptomoneda més influent i utilitzada a nivell global. El seu funcionament es basa en una arquitectura descentralitzada peer-to-peer i en una estructura de dades anomenada blockchain, que actua com un llibre de registre públic i immutable on es desen totes les transaccions realitzades.

A mesura que l'ús de Bitcoin ha anat creixent, també ho ha fet el seu ecosistema i la quantitat de dades generades. Amb més de 900.000 blocs [1] i més de mil milions de transaccions registrades [2], la blockchain de Bitcoin constitueix una font massiva d'informació oberta i consultable. Aquesta obertura ha permès el desenvolupament d'estudis des de diferents disciplines, com la informàtica i la criptografia fina a l'economia, la seguretat o la criminologia, amb l'objectiu d'analitzar patrons de comportament, identificar activitats sospitoses, o avaluar el grau de privadesa del sistema.

Tanmateix, l'estructura original de la blockchain no està pensada per facilitar l'anàlisi directa del flux de bitcoins ni la reconstrucció de relacions entre entitats. Les transaccions es troben encapsulades dins de blocs de forma seqüencial, i no hi ha cap indexació explícita que relacioni inputs i outputs entre transaccions. Per aquest motiu, molts treballs proposen representar la blockchain com un graf, on els nodes poden ser transaccions o adreces, i les arestes representen les transferències de valor entre elles.

Aquestes representacions han estat especialment útils per a tasques com el clustering d'adreces, el seguiment de fluxos de bitcoins (en casos de robatoris, blanqueig de capitals o ransomware), o l'anàlisi econòmica del comportament dels usuaris dins la xarxa. No obstant això, aquestes tècniques presenten problemes d'escalabilitat, ja que tant els grafs de transaccions com els d'adreces poden arribar a tenir centenars o milers de milions de nodes, cosa que dificulta el processament i l'anàlisi eficient, especialment en entorns amb recursos computacionals limitats.

En aquest context, el present treball proposa una estratègia alternativa basada en la representació combinada de la blockchain com a graf de blocs i graf de transaccions. El graf de blocs ofereix una visió global i compacta de la xarxa, on cada node representa un bloc i les arestes indiquen dependències transaccionals entre blocs. Aquesta simplificació permet treballar amb un conjunt molt més reduït de dades (aproximadament 900.000 nodes [1], respecte els mil milions de nodes inicials [2]), mantenint la capacitat d'analitzar l'evolució temporal i els fluxos econòmics de manera eficient.

- *E-mail de contacte: Francesc.Escola@autonoma.cat*
- *Menció realitzada: Tecnologies de la Informació*
- *Treball tutoritzat per: Julián Salas Piñón (DEIC)*
- *Curs 2024/25*

A partir d'aquest graf de blocs, és possible construir un graf de transaccions parcial sobre un subconjunt seleccionat de blocs, permetent aprofundir l'anàlisi només en les àrees rellevants i evitant el cost computacional de processar tota la xarxa. Aquest enfocament híbrid cerca trobar un equilibri entre precisió analítica i escalabilitat, facilitant així el desenvolupament d'eines d'anàlisi estructurada, visualització i traçabilitat de dades en l'ecosistema Bitcoin.

2 Objectius

L'objectiu principal d'aquest treball ha estat el de desenvolupar un eina capaç de representar i analitzar la blockchain de Bitcoin mitjançant grafs, aprofitant el seu caràcter públic i estructurat per tal de facilitar-ne l'exploració i traçabilitat. Els objectius específics que s'han abordat al llarg del desenvolupament han estat els esmentats a continuació.

2.1 Construcció del graf de blocs

Dissenyar i implementar una representació en forma de graf, on cada node representa un bloc de la blockchain i una aresta entre nodes indicant que hi ha almenys una transacció en el bloc més actual que gasta un output del bloc més antic. Aquesta estructura permet identificar com es relacionen els blocs més enllà de la seva seqüència temporal, mostrant com es mouen els bitcoins entre ells.

2.2 Construcció del graf de transaccions

Crear una representació de les transaccions com a graf, on els nodes son les transaccions i les arestes representen la transferència de bitcoins entre elles, on els pesos son els bitcoins enviats. Aquesta estructura fa possible traçar el recorregut dels bitcoins de la blockchain, obrint la possibilitat de diferents tipus d'anàlisis.

2.3 Integració amb Neo4J

Durant tot el desenvolupament del projecte, s'ha treballat en paral·lel amb la integració del projecte a Neo4J per tal de poder fer servir aquest servei com base de dades del projecte. El potencial que ens ofereix Neo4J es tal que també ens permet poder realitzar consultes complexes, i poder visualitzar totes les dades que siguin necessàries de la blockchain de forma ràpida e intuïtiva.

3 Estat del Art

L'anàlisi de la blockchain de Bitcoin s'ha consolidat com una àrea de recerca rellevant en àmbits com els sistemes distribuïts, la privadesa i seguretat informàtica. El caràcter públic i immutable de la cadena de blocs de Bitcoin permet extreure grans volums de dades i representar-les com a estructures orientades a grafs, fet que facilita l'estudi detallat del comportament de les transaccions i usuaris que hi intervenen.

3.1 Representació de la blockchain com a graf

La blockchain de Bitcoin es pot entendre com una seqüència lineal de blocs enllaçats, on cada bloc conté un conjunt de transaccions i està connectat criptogràficament amb l'anterior. Aquesta estructura, en què els blocs més recents se situen al final de la cadena, permet dur a terme anàlisis històrics de l'activitat a la xarxa, ja que cada bloc representa un instant concret en la línia del temps del sistema.

Cada bloc, conté nombroses transaccions, on totes elles consumeixen bitcoins que han estat generats prèviament per altres transaccions. És a dir, els inputs d'una transacció han de fer referència a outputs generats en transaccions de blocs anteriors, o bé a transaccions prèvies dins del mateix bloc. Això indica que aquesta relació de dependència entre transaccions permet, almenys en teoria, reconstruir el camí que segueixen els bitcoins des del seu origen fins al seu destí actual, fixant-se en els identificadors de la transacció (txid) i els inputs de les transaccions.

Tanmateix, quan s'intenta dur a terme aquesta traçabilitat en un conjunt de dades tan extens com és la blockchain de Bitcoin, que ja supera els centenars de gigabytes [3], es fa evident una limitació significativa, l'únic ordre explicit existent és el temporal dels blocs, i no existeix cap índex directe que relacioni les transaccions que contenen els outputs i les que contenen els inputs que gasten aquests outputs, fet que complica l'anàlisis de fluxos de valor o clustering d'adreces.

Per tal de superar aquesta limitació, diversos treballs proposen representar la blockchain com un graf dirigit, on els nodes poden correspondre a **transaccions o adreces**, i les arestes indiquen relacions de dependència o de transferencia de valor. Aquesta representació gràfica transforma l'estructura lineal, en una xarxa que reflecteix millor la dinàmica real dels fluxos de bitcoins, i permet l'aplicació de tècniques d'anàlisi pròpies de la teoria de grafs i de les xarxes complexes [4].

Un dels primers estudis rellevants són de **Ron i Shamir (2013)** [5], que van construir un graf de transaccions, i van proposar una metodologia de clustering d'adreces, mostrant com identificar wallets i fluxos de bitcoins a través de la blockchain. Una de les contribucions més destacades del treball és la identificació de transaccions de gran valor aparentment aïllades però que, mitjançant l'anàlisi estructural del graf, es va revelar com deriven d'una transacció massiva realitzada al novembre del 2010. Aquest descobriment demostra el potencial de la representació en forma de graf per representar patrons amagats en la xarxa, tot i els intents d'ofuscació per part dels usuaris.

Meiklejohn et al. (2013) [6], van ampliar els estudis previs, amb la premisa que si bé Bitcoin ofereix un cert grau de pseudoanonimat, el seu flux és públic i rastrejable. El seu treball es va centrar en l'anàlisis empíric de Bitcoin mitjançant la recopilació d'adreces i l'aplicació d'heurístiques per identificar entitats. Un dels casos de seguiment més rellevants va ser el del mercat negre Silk Road, demostrant que es poden rastrejar activitats

potencialment il·lícites. Aquest estudi posa en evidencia els límits del pseudoanonimat de Bitcoin i els reptes que afronten aquells que intenten utilitzar-ho com a eina d'ocultació.

Posteriorment, **Di Francesco Maesa et al. (2018)** [7] analitzen la blockchain de Bitcoin mitjançant grafs d'adreces (usuaris) amb l'objectiu d'estudiar propietats estructurals com la distribució de graus, la densitat, la centralitat i la concentració de riquesa. A més, examinen com aquestes propietats reflecteixen patrons socio-econòmics, com la tendència a la concentració de riquesa en determinats nodes. Aquest estudi sorgeix en un moment de creixement exponencial del volum de transaccions a partir de 2015, una tendencia que continua actualment amb més de **mil milions de transaccions** registrades a la xarxa [2].

Tot i que la majoria de treballs es centren en la representació de la blockchain mitjançant grafs de transaccions o d'adreces, en aquest treball plantegem la construcció de grafs de blocs, que permet analitzar com les transaccions interconnecten diferents blocs i observar l'evolució dels fluxos econòmics al llarg del temps. Aquesta representació és computacionalment més eficient, ja que redueix l'abast del problema agrupant les transaccions dins dels blocs corresponents, passant d'un potencial graf de transaccions de més de mil milions de nodes [2], a un graf de blocs de nou-cents mil nodes [1]. Això facilita una visió global de la xarxa a nivell temporal i permet el rastreig de dependències entre blocs sense necessitat de mostrar cada transacció de manera individual, fet que la fa especialment adequada per a entorns amb recursos computacionals limitats.

3.2 Clustering d'adreces i anàlisi forense

Tot i que Bitcoin es concep com un sistema pseudoanònim, on les adreces no estan explícitament vinculades a la identitat real dels usuaris, múltiples estudis han demostrat que és possible inferir agrupacions d'adreces que pertanyen a un mateix individu o entitat mitjançant tècniques d'anàlisi de grafs i heurístiques de comportament.

Una de les heurístiques més conegudes és la "multi-input", estudis com el ja presentat de **Ron i Shamir** [5], consisteixen en assumir que si diverses adreces signen una mateixa transacció d'entrada, molt probablement pertanyen al mateix propietari. Aquesta idea ha estat explotada i refinada per diversos treballs, com els de **Meiklejohn et al.** [6].

Aquestes tècniques han estat fonamentals per a la recerca forense, ja que permeten identificar patrons de comportament, agrupacions d'entitats i fins i tot seguir la traçabilitat de fons il·lícits. A més, han posat en qüestió el grau real de privadesa que ofereix Bitcoin, mostrant que amb suficient informació externa com adreces conegudes associades a serveis, és possible establir relacions entre usuaris i activitats.

Altres treballs com els de **Androulaki et al. (2013)** [8] evidencien encara més els riscos de privadesa a través de l'anàlisi de dades de xarxa i comportament d'ús. En aquest estudi, s'aconsegueix reidentificar el 40% dels usuaris en un entorn de prova tot i l'ús de nous parells de claus en cada transacció i seguint les recomanacions de privacitat de Bitcoin, posant de manifest que la reutilització d'adreces i patrons d'ús poden comprometre l'anonimat.

3.3 Eines i infraestructura per a l'anàlisis

Diverses eines han estat desenvolupades per facilitar la representació i exploració de la blockchain com a graf. Un dels projectes més destacats es l'eina **BlockSci** [9], una biblioteca de Python i C++ altament optimitzada per analitzar la cadena de blocs, orientada a la investigació acadèmica. Permet crear grafs d'adreces i transaccions, extreure estadístiques i aplicar heurístiques de clustering.

També cal mencionar el projecte **Bitcoingraph de Haslhofer i Karl (2015)** [10], el qual ha servit com a referencia directa per aquest treball. Aquest repositori proposa una arquitectura modular basada en Python per exportar blocs i transaccions de Bitcoin a una base de dades de grafs, usant una interfície RPC de Bitcoin Core [11] i una interfície gràfica usant la plataforma Neo4J on es poden visualitzar els grafs creats i fer-ne peticions personalitzades.

4 METODOLOGIA

Per dur a terme els objectius del projecte, s'ha seguit una metodologia centrada en la programació en Python, la construcció modular del software i l'ús de drivers per poder integrar el codi amb Neo4J.

4.1 Entorn de desenvolupament

El entorn de desenvolupament s'ha realitzat amb l'IDE PyCharm Community, usant Python 3.12, que ha permès estructurar el projecte en mòduls de manera clara i extensible. El projecte té dos paquets:

- **bitcoingraph**: conté les estructures i funcionalitats per a la representació i gestió dels blocs i transaccions de la blockchain.
- **scripts**: conté scripts que fan ús del paquet anterior per dur a terme accions concretes com la sincronització de dades amb Neo4J o l'exportació de dades de la blockchain.

Per poder connectar-se amb la blockchain de Bitcoin, s'utilitza un node remot proporcionat per la mateixa universitat, al qual s'hi accedeix mitjançant VPN (WireGuard) i SSH. Les dades s'obtenen fent crides JSON-RPC al node de Bitcoin Core de la UAB i s'analitzen localment per construir els grafs.

4.2 Bitcoingraph

Com s'ha comentat anteriorment, el projecte té un paquet principal, que mou la lógica principal del projecte. Aquest es desglosa en diferents mòduls en diferents arxius Python:

- **bitcoingraph.py:** actua com a controlador principal del paquet, proporcionant una classe amb les funcions principals com exportar la blockchain, crear el graf de blocs de forma local, crear i actualitzar el graf de blocs a Neo4J, i crear el graf de transaccions a Neo4J. En el mateix ordre, aquestes funcions reben el nom:
  - **def export**(self, start, end, step, output_path, separate_header)
  - **def blockgraph**(self, start, end, step, output_path, separate_header)
  - **def synchronize**(self, max_blocks)
  - **def transactiongraph**(self, tx_id, depth)

  A més, la classe disposa de dos atributs clau: **blockchain** que facilita l'accés a la informació de la blockchain i **graph_db** que permet la interacció amb el de la base de dades Neo4J.

- **blockchain.py:** aquest mòdul actua com a API per a la navegació i consulta de la blockchain. Proporciona funcionalitats per obtenir blocs tant per seu hash com per la seva altura, així com per recuperar blocs dins d'un rang especificat. També permet obtenir transaccions a partir del seu identificador (txid), tant individualment com en grup, i consultar l'altura de l'últim block de la xarxa. Aquest mòdul ha estat desenvolupat per Bernhard Haslhofer, MIT [10].

- **bitcoind.py:** serveix com a interfície que utilitza blockchain.py per executar comandes RPC al node Bitcoin Core de la universitat. Aquest mòdul ha estat desenvolupat per Bernhard Haslhofer, MIT [10].

- **model.py:** defineix les estructures de dades utilitzades per la resta de mòduls per representar els elements fonamentals de la blockchain, com blocs, transaccions, inputs i outputs.
  - **Block**: representa un bloc amb atributs com ara l'altura (**height**), referències als blocs adjacents (**height_prev_block** i **height_next_block**) i una llista de transaccions associades (**transactions**)
  - **Transaction:** inclou atributs per identificar la transacció (**txid**), el hash del bloc al qual pertany (**blockhash**), així com un llistat dels inputs (**inputs**) i dels outputs (**outputs**). Disposa del mètode **def is_coinbase**(self) per indicar si la transacció és coinbase.
  - **Input:** conté informació sobre si l'input es coinbase (**is_coinbase**), l'identificador de la transacció d'origen dels bitcoins (**txid**) i la referència a l'output específic d'aquesta transacció (**vout**).
  - **Output:** conté l'identificador de la transacció al que pertany l'output (**tx**), l'index del output dins la transacció (**index**) i els bitcoins que se li assigna a l'adreça destí (**value**).

- **writer.py:** aquest mòdul s'encarrega de parsejar els blocs proporcionats per paràmetre extrets de la blockchain **def parse_block**(self, block), també s'encarrega d'emmagatzemar les relacions entre blocs, facilitant així la construcció i persistència local del graf de blocs **def write_block**(self, block).

- **neo4jcontroller.py:** mòdul encarregat de la interàcció directa amb el driver oficial de Neo4J per Python. Proporciona funcionalitats per afegir blocs i transaccions, així com les seves respectives relacions (bloc-bloc com transacció-transacció). També permet obtenir els camins de relacions entre blocs.

- **graphdb.py:** actua com a interfície d'alt nivell per accedir al mòdul neo4jcontroller.py. Les seves funcionalitats inclouen afegir un bloc i les seves relacions **def add_block**(self, block, blockchain), **def get_sub_graph**(self, height, depth) permet obtenir el camí de blocs a partir d'una altura concreta fins a una profunditat especificada. El métode **def add_tx_graph_from_dict**(self, tx_graph) permet pujar un graf de transaccions.

- **transactiongraph.py: def build_transaction_graph**(self, init_tx_id, depht) construeix el bloc de transaccions a partir del txid inicial . Té com atributs el graf de blocs del qual es genera el graf de transaccions (**adjacency_block_matrix**), la transacció d'on parteix el graf (**graph_tx_id_genesis**) i el graf de transaccions en format diccionari (**tx_graph**). Paral·lelament té un altre funció **def get_txgraph_leaves**(self) que només quan tx_graph és creat retorna les fulles del graf, útil en cas que es vulgui aprofundir en algún camí del graf.

4.3 Estructures de dades

Per representar la informació de la blockchain, com s'ha mencionat al punt anterior, s'han definit dues estructures:

- **Graf de blocs:** a partir de la identificació d'una transacció que gasta l'output d'una transacció anterior, s'extreuen les altures dels blocs a les que pertanyen aquestes dues transaccions i es crea la relació **height-TO-height**, és a dir es crea una relació entre el bloc amb la primera altura i el bloc amb la segona altura, funciona contemplant l'existència de llaços és a dir, que una transacció faci referencia a un altre del mateix bloc.

- **Graf de transaccions:** quan es detecta que un input gasta l'output d'un altre transacció, es crea una relació entre aquestes dues transaccions i se li afegeix un pes, el qual és la quantitat de bitcoins enviats d'una adreça a un altre. També afegir que als grafs creats, els identificadors de les transaccions no son els txid, sinó que s'extreu l'altura del bloc que pertanyen i la posició de la transacció en el bloc i es concatenen formant un nou identificador per la transacció. L'identificador acaba tenint aquest aspecte: **height_pos**.

4.4 Emmagatzematge i consultes a Neo4J

Neo4J es un eina realment potent i interactiva per treballar amb grafs. Un cop generats, aquests grafs es sincronitzen amb Neo4J Aura mitjançant crides al driver per Python neo4j-driver. Un cop sincronitzada la base de dades remota, la seva visualització i anàlisi es poden dur a terme a través de la interfície web Neo4J Browser i realitzar consultes complexes gràcies al al llenguatge Cypher. Aquest llenguatge de consultes està especialment dissenyat per a grafs i proporciona una gran flexibilitat per a explotar relacions i estructures, tal com s'evidencia a la secció de resultats del present treball.

A més, el sistema també permet obtenir els nodes pujats a la base de dades i emmagatzemar-los localment en fitxers CSV, facilitant-ne, si escau, l'anàlisis posterior fora de l'entorn de Neo4J.

5 RESULTATS

Un cop completades les fases de desenvolupament i integració del projecte, s'han generat les dues estructures principals: el graf de blocs i el graf de transaccions. Aquest apartat presenta els resultats obtinguts durant l'anàlisi del conjunt de dades extret de la blockchain de Bitcoin.

5.1 Graf de blocs

El primer objectiu ha estat la construcció d'un graf de blocs, on cada node representa un bloc de la blockchain de Bitcoin, i s'estableix una aresta dirigida entre dos blocs $b_i \rightarrow b_j$ si una transacció dintre de b_j gasta un output creat en b_i. Aquest enfocament permet identificar relacions entre blocs no necessàriament consecutius, basant-se en la dependència real entre transaccions en lloc de la seva successió temporal.

Per garantir un rendiment òptim durant l'anàlisi i visualització del graf, es va limitar el conjunt de dades a **83.232 blocs consecutius de la blockchain**. Aquesta decisió respon a consideracions d'eficiència computacional, ja que es va detectar una degradació progressiva del rendiment de l'entorn de treball basat en Neo4J a mesura que augmenta el volum d'informació processada. Així doncs, aquest subconjunt es va establir com a banc de proves estable, permetent explorar i validar el model proposat sense comprometre la capacitat de resposta d'entorn de treball.

A partir del conjunt extret de la blockchain, s'han identificat 57.485 relacions inter-bloc, corresponents a casos en què almenys una transacció d'un bloc consumeix un output generat en un altre. És important remarcar que aquesta xifra no representa el nombre total de transaccions connectades sinó al nombre de parells de blocs que mantenen almenys una dependència d'aquest tipus.

La representació completa del graf de blocs emmagatzemat a Neo4J es pot visualitzar mitjançant la Consulta 1 (vegeu Apèndix), que retorna tots els blocs i les seves relacions inter-bloc en cas que tinguin. La figura 1 mostra el resultat gràfic d'aquesta exploració.

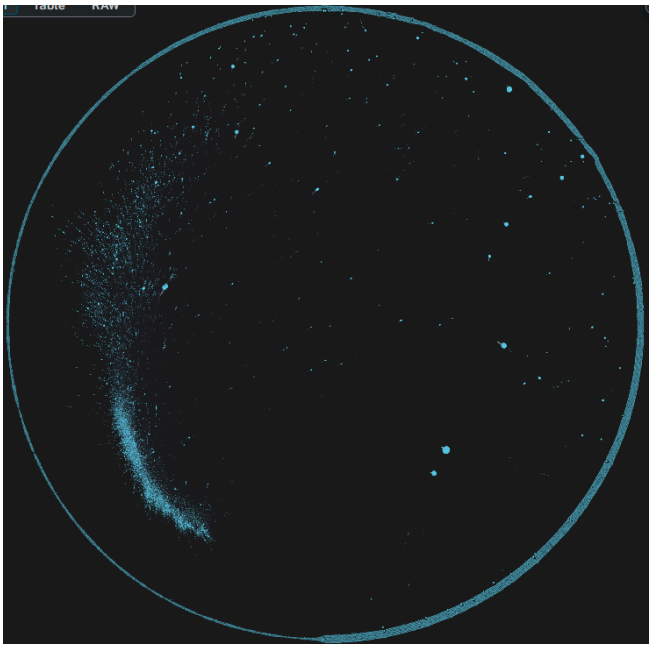

Fig. 1: Graf de blocs complert

Donat l'elevat nombre de nodes del graf, la seva visualització completa en una única imatge dificulta l'anàlisi detallat de les relacions específiques. Per tal de comprendre millor la naturalesa de la xarxa, s'ha seleccionat una regió situada a la part inferior esquerra del graf, caracteritzada per una alta densitat de connexions entre blocs, i se n'ha realitzat una ampliació, com es mostra a la figura 2.

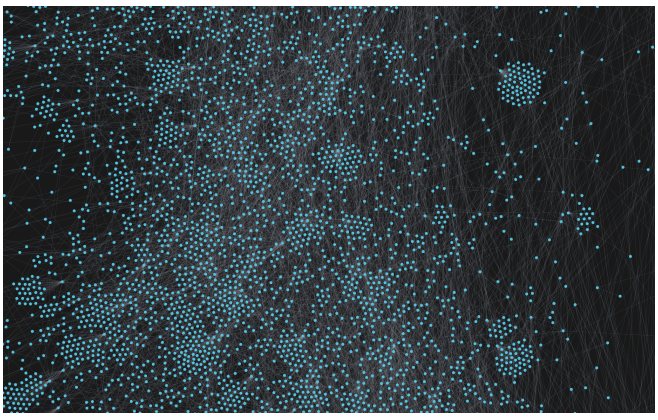

Fig. 2: Graf de blocs complert ampliat

Aquesta àrea ampliada permet observar patrons estructurals més definits. Per exemple, s'identifiquen grups de blocs amb connexions concentrades entorn a un bloc central, fet que pot indicar la recepció de bitcoins procedents de múltiples blocs anteriors, o bé el contrari, la recepció de bitcoins cap a múltiples blocs posteriors. També es detecten blocs amb poques o cap connexió, que poden correspondre a blocs amb baixa activitat transaccional o amb un comportament aïllat dins la xarxa. Malgrat aquests exemples, el graf mostra una forta connectivitat generalitzada, evidenciant la naturalesa altament interrelacionada de la blockchain.

Amb l'objectiu d'obtenir una visió encara més precisa de les relacions entre blocs, s'ha ampliat encara més aquesta imatge, tal com s'il·lustra a la figura 3.

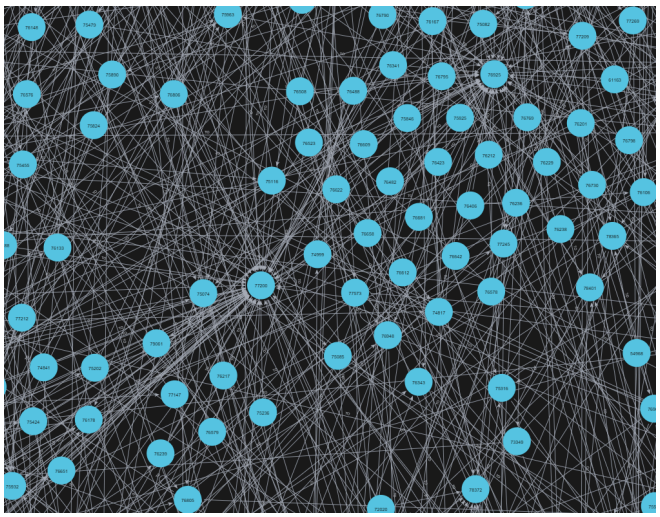

Fig. 3: Graf de blocs complert molt més ampliat

Un cop visualitzat el conjunt complet del banc de proves, s'ha explorat una funcionalitat proporcionada per Neo4j a través del llenguatge Cypher: la generació de camins entre blocs a partir de la seva altura. Aquesta funcionalitat permet traçar rutes a través del graf segons diferents criteris de navegació: mitjançant arestes de sortida **(b)-[:TO]->(b)**, d'entrada **(b)<-[:TO]-(b)**, o bé en ambdós sentits **(b)-[:TO]-(b)**.

Per il·lustrar aquestes noves peticions, s'han escollit dos blocs d'origen per a l'anàlisi. El primer és el bloc 9, ja que la primera transacció que gasta un output a la blockchain, gasta l'output d'aquest bloc. El segon és el bloc 74087, seleccionat aleatòriament d'entre els visualitzats a la figura 2. Les figures 4 i 5 mostren els camins resultants generats a partir d'aquests blocs respectivament.

El camí del bloc 9 es pot visualitzar mitjançant la Consulta 2 (vegeu Apèndix), alterant el valor HEIGHT pel valor de l'altura del bloc, en aquest cas 9.

Fig. 4: Camí de blocs partint del bloc 9

La figura 4 mostra un arbre de relacions de blocs amb diverses ramificacions. Per exemple, blocs com el 187 i el 248 gasten outputs d'una mateixa transacció, generant una bifurcació. També s'observen casos de convergència, com en el bloc 496, on múltiples camins convergeixen. Cal tenir en compte, però, que aquest tipus d'operacions poden incloure camins que no estan exclusivament relacionats amb el bloc d'origen. Per exemple, si un bloc intermedi (com el bloc 170) gasta outputs provinents tant del bloc 9 com d'un tercer bloc no representat al camí, la continuació del recorregut inclou per tant transaccions no associades al bloc d'origen. Aquest comportament és especialment rellevant quan pretén construir un graf de transaccions acurat, aspecte que es resol verificant la procedencia del input gastat.

Per evitar problemes de rendiment associats a camins excessivament llargs, s'ha limitat la profunditat de la cerca. En el cas del camí generat a partir del bloc 74087, s'ha fixat una profunditat màxima de 26. Aquest recorregut s'ha obtingut mitjançant la Consulta 3 (vegeu Apèndix), substituint els paràmetres HEIGHT i DEPTH per l'altura del bloc i la profunditat definida.

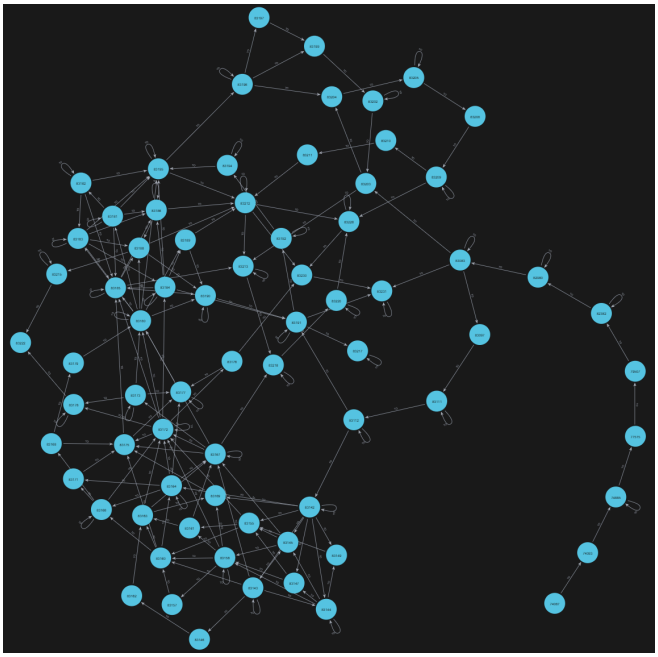

Fig. 5: Camí de blocs partint del bloc 74087

Aquest resultat presenta una estructura més complexa, amb múltiples interconnexions entre blocs i llaços. El bloc 74087, el bloc generador del camí, es localitza a la part inferior dreta del graf.

5.2 Graf de transaccions

El segon objectiu ha estat el poder construir un graf de transaccions, on cada node es una transacció, i s'estableix una aresta dirigida entre dues transaccions $t_i \rightarrow t$ si almenys un dels inputs de t consumeix un output generat per t_i. El pes de l'aresta correspon a la quantitat de bitcoins transferida entre aquestes dues transaccions. Aquesta estructura permet una representació clara de la traçabilitat dels bitcoins, des de la seva emissió fins a la seva destinació final dins del conjunt de dades analitzat, és a dir, fins al bloc d'altura 83.231 (amb altura inicial 0).

Tanmateix cal tenir en compte dues consideracions importants en la construcció i interpretació d'aquest graf:

- **Outputs no gastats:** és important remarcar que no tots els outputs generats per una transacció són necessàriament gastats immediatament. Això implica que si, per exemple, una transacció rep 5 BTC però només se n'ha gastat 3, no es generarà cap aresta corresponent als 2 BTC restants fins que siguin utilitzats en una futura transacció. Aquest comportament pot donar lloc a aparent desajust entre el total d'entrada i el de sortida de certes transaccions, però és una característica esperada del funcionament de la xarxa Bitcoin.

- **Inputs combinats de diverses fonts:** en molts casos, les transaccions que consumeixen outputs d'una transacció determinada també poden incorporar inputs provinents de transaccions alienes al camí analitzat. Com que la construcció del graf es basa en partir d'una transacció inicial, aquesta situació provoca que, en nivells posteriors del graf, s'estiguin també seguint bitcoins procedents d'aquestes fonts externes. Tot i que aquest comportament pot ser útil per a tasques de clustering d'adreces, en el context de seguiment de bitcoins pot introduir soroll i càrrega computacional addicional. Aquest fenomen es manifesta en forma d'una major quantitat de bitcoins a les arestes de sortida que a les d'entrada d'una transacció concreta.

Un cop definit l'objectiu, a continuació es mostren els resultats obtinguts amb la construcció del graf de transaccions. Per garantir la comparabilitat amb l'anàlisi realitzada a l'apartat anterior, s'han seleccionat els mateixos blocs: el bloc 9 i el bloc 74.087.

A continuació, es tria una transacció representativa de cada bloc. Del bloc 9, s'utilitza la primera transacció (identificada com 9_0), mentre que del bloc 74087 s'utilitza la segona (74087_1). A partir d'aquestes transaccions, s'ha aplicat la funció transactiongraph del paquet bitcoingraph per construir el graf de transaccions corresponent.

En el cas de la transacció 9_0, no s'ha imposat cap límit de profunditat, atès que es tracta d'una transacció antiga amb una profunditat esperada relativament reduïda. En canvi, per la transacció 74087_1 s'ha fixat una profunditat màxima de 8, amb l'objectiu de limitar la mida del graf de blocs implicat i evitar un cost computacional excessiu durant la generació local del graf de transaccions.

Un cop generat i sincronitzat a Neo4J cada un dels grafs de transaccions per separat, s'ha executat la Consulta 4 (vegeu Apèndix), la qual permet recuperar les relacions entre transaccions del graf. Les figures 6 i 7 recullen els resultats obtinguts per a cada cas.

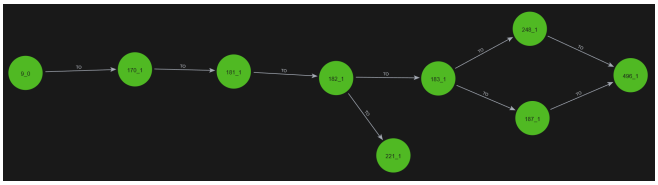

Fig. 6: Graf de transaccions partint de la transacció 9_0

A la figura 6 s'observa com el graf mostra una estructura molt senzilla i lineal, la qual cosa concorda amb l'estructura del graf de blocs que parteix del bloc 9. Aquest comportament és esperable, ja que en les primeres etapes de la blockchain hi havia poques transaccions que gastessin outputs d'altres transaccions anteriors.

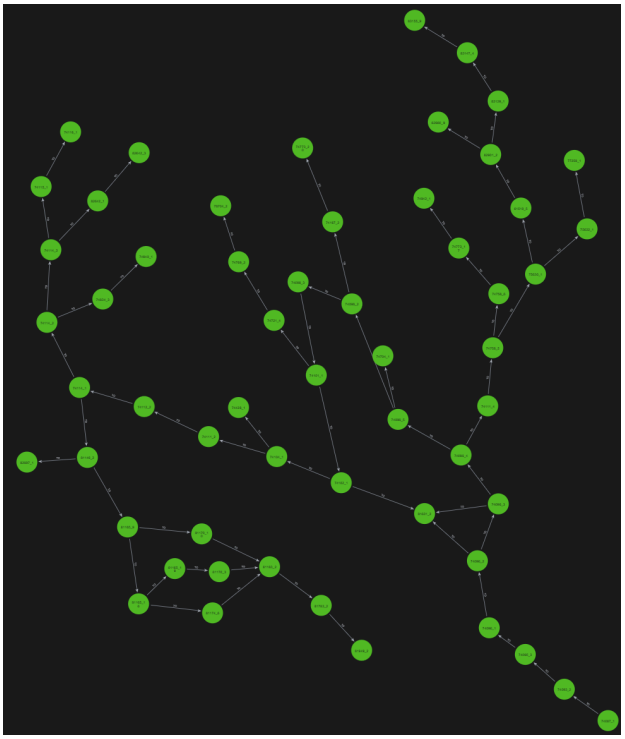

Fig. 7: Graf de transaccions partint de la transacció 74087_1

A la figura 7, s'observa un graf amb estructura d'arbre que s'origina a partir de la transacció 74087_1 (situada a la part inferior dreta) i s'estén fins a les seves fulles. Aquest graf evidencia una activitat transaccional més complexa i ramificada pròpia d'una etapa més madura de la blockchain. En cas de voler continuar l'expansió del graf,

el mètode transactiongraph retorna els identificadors de les transaccions corresponents a les fulles, permetent així repetir el procés i aprofundir en la direcció desitjada.

Una particularitat de la figura 7, és que es pot observar com una transacció es bifurca en diverses transaccions per, finalment després d'unes quantes iteracions convergir en una mateixa transacció novament. Aquest fenomen, que podem veure en detall a la figura 8, és especialment interessant, ja que pot indicar un intent d'un usuari de dificultar la traçabilitat dels seus moviments per la xarxa. És a dir, podria ser que diverses de les adreces involucrades en aquestes transaccions pertanyin a una mateixa entitat o usuari.

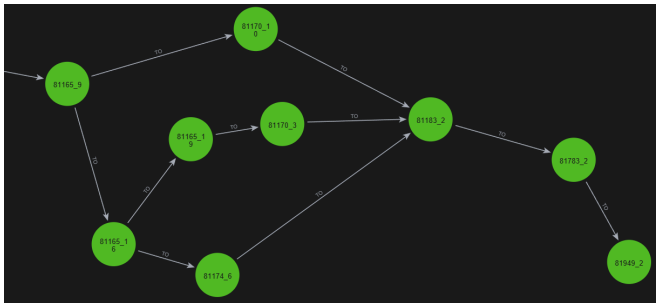

Fig. 8: Graf de transaccions partint de la transacció 74087_1 ampliat

Aquest petit exemple serveix per corroborar que el present treball també té aplicacions potencials en l'àmbit del clustering d'adreces. L'estructura del graf de transaccions, tal com s'ha implementat, permet no només el seguiment de la traçabilitat dels fons, sinó també la detecció de patrons sospitosos o recurrents dins l'activitat transaccional.

6 Conclusió

El treball ha tingut com a objectiu principal el desenvolupament d'una eina capaç de representar i analitzar la blockchain de Bitcoin mitjançant estructures de graf, amb la finalitat de facilitar el seguiment i la traçabilitat de les transaccions.

S'han assolit amb èxit els dos objectius plantejats:

- **Construcció del graf de blocs:** s'ha implementat un algoritme que construeix un graf de blocs que permet connectar els blocs de la blockchain entre ells, on s'estableix una relació entre dos blocs $b_i \rightarrow b$? quan una transacció del bloc b? gasta un output creat en el bloc b_i. D'aquesta manera, es proporciona una perspectiva més funcional de la blockchain, més enllà de la seva naturalesa seqüencial. L'eina permet consultar i visualitzar aquest graf de manera eficient mitjançant Neo4j Browser.

- **Construcció del graf de transaccions:** s'ha desenvolupat un sistema que segueix el bitcoin a partir d'un punt d'origen, d'una transacció inicial, mostrant com es transfereixen els bitcoins a través del temps.

Els resultats obtinguts mostren que l'eina desenvolupada facilita l'anàlisi de patrons de moviment de bitcoins, permetent identificar trajectòries, bifurcacions i possibles concentracions de valor dins la blockchain. Aquesta capacitat ha demostrat també obrir la porta a futures aplicacions com en detecció de comportaments sospitosos, l'anàlisi de fluxos massius o clustering d'adreces. A més gràcies al disseny modular del software, permet incloure noves funcionalitats de manera senzilla en futures versions del projecte.

Aquest projecte, no només ha permès assolir els objectius tècnics, sinó també ha servit com a marc per entendre millor el funcionament intern de la blockchain de Bitcoin i el seu potencial d'anàlisi mitjançant tècniques de graf.

Agraïments

Vull expressar el meu sincer agraïment a tots els professors que, al llarg del grau, han sabut transmetre amb passió els seus coneixements i, en especial, aquells que han despertat en mi l'interès pel món del Bitcoin i la tecnologia que hi ha al darrere. Sense aquesta motivació inicial, probablement no hauria triat aquest tema per al meu TFG.

També voldria agrair als professors que m'han donat un cop de mà puntual durant el desenvolupament del projecte, oferint-me orientació o resolent dubtes en moments clau.

Un agraïment molt especial al meu tutor, Julián Salas Piñón, per la seva dedicació i per saber guiar-me amb idees clau que han enriquit significativament el treball. La seva supervisió ha estat fonamental per donar forma i coherència al projecte.

Finalment, una menció molt especial a la meva família i amics més propers pel seu suport incondicional durant tot el procés. Gràcies per acompanyar-me, animar-me i donar-me forces quan més ho necessitava. En especial, vull agrair a la meva mare, Mª Montserrat Bargués Companys, per estar sempre al meu costat.

Bibliografia

[1] "Latest BTC Blocks", Blockchain.com, June 20, 2025. https://www.blockchain.com/es/explorer/blocks/btc

[2] "Bitcoin scalability problem", Wikipedia, June 20, 2025. https://en.wikipedia.org/wiki/Bitcoin_scalability_problem

[3] R. de Best, "Bitcoin (BTC) blockchain size as of May 13, 2025", Statista, June 20, 2025. https://www.statista.com/statistics/647523/worldwide-bitcoin-blockchain-size/

[4] Barabási, A. L. (2013). Network science. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 371(1987), 20120375.

[5] Ron, D., Shamir, A. (2013). Quantitative Analysis of the Full Bitcoin Transaction Graph. In: Sadeghi, AR. (eds) Financial Cryptography and Data Security. FC 2013. Lecture Notes in Computer Science, vol 7859. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-39884-1_2

[6] Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G. M., & Savage, S. (2013, October). A fistful of bitcoins: characterizing payments among men with no names. In Proceedings of the 2013 conference on Internet measurement conference (pp. 127-140).

[7] Di Francesco Maesa, D., Marino, A., & Ricci, L. (2018). Data-driven analysis of bitcoin properties: exploiting the users graph. International Journal of Data Science and Analytics, 6, 63-80.

[8] Androulaki, E., Karame, G. O., Roeschlin, M., Scherer, T., & Capkun, S. (2013). Evaluating user privacy in bitcoin. In Financial Cryptography and Data Security: 17th International Conference, FC 2013, Okinawa, Japan, April 1-5, 2013, Revised Selected Papers 17 (pp. 34-51). Springer Berlin Heidelberg.

[9] Kalodner, H., Möser, M., Lee, K., Goldfeder, S., Plattner, M., Chator, A., & Narayanan, A. (2020). {BlockSci}: Design and applications of a blockchain analysis platform. In 29th USENIX Security Symposium (USENIX Security 20) (pp. 2721-2738).

[10] Haslhofer, B. & Karl, R. (2015). Bitcoingraph [Source code]. GitHub. https://github.com/behas/bitcoingraph

[11] "Bitcoin RPC API Reference", Bitcoin.org, June 20, 2025. https://developer.bitcoin.org/reference/rpc/

APÈNDIX

Consulta 1 - Visualització del graf de blocs

La següent consulta en llenguatge Cypher mostra tots els blocs i les seves relacions inter-bloc:

```
MATCH (b:Block)
OPTIONAL MATCH (b)-[r:TO]->(b2:Block)
RETURN b, r, b2
```

Consulta 2 - Visualització de camí de blocs a partir d'un bloc inicial

La següent consulta en llenguatge Cypher mostra el camí de blocs relacionats amb el bloc d'altura HEIGHT:

```
MATCH (start:Block {height: HEIGHT})
MATCH path = (start)-[:TO*]->(end)
RETURN path
```

Consulta 3 - Visualització de camí de blocs a partir d'un bloc inicial amb profunditat màxima

La següent consulta en llenguatge Cypher mostra el camí de blocs relacionats amb el bloc d'altura HEIGHT i profunditat màxima DEPTH:

```
MATCH (start:Block {height: HEIGHT})
MATCH path = (start)-[:TO*1..DEPTH]->(end)
RETURN path
```

Consulta 4 - Visualització de graf de transaccions

La següent consulta en llenguatge Cypher mostra el graf de transaccions:

```
MATCH (t1:Tx)-[r:TO]-> (t2:Tx)
RETURN t1, t2, r
```