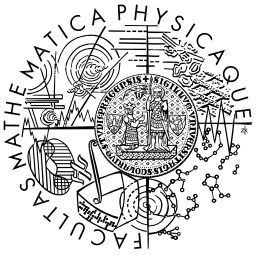


MATEMATICKO-FYZIKÁLNÍ FAKULTA
Univerzita Karlova

DIPLOMOVÁ PRÁCE

David Surma

Analýza blockchainu používaného pro Bitcoin

Informatický ústav Univerzity Karlovy

Vedoucí diplomové práce: Ing. David Hartman, Ph.D.
Studijní program: Informatika
Studijní obor: Umělá inteligence

Praha 2023

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů. Tato práce nebyla využita k získání jiného nebo stejného titulu.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V ..... dne ....

Podpis autora

Děkuji svému vedoucímu Ing. Davidu Hartmanovi, Ph.D., za rady a připomínky vedoucí ke zkvalitnění mé práce. Rovněž děkuji svému bratrovi Martinovi za cennou zpětnou vazbu.

Výpočetní prostředky byly poskytnuty projektem e-INFRA CZ (ID:90140) podporovaným Ministerstvem školství, mládeže a tělovýchovy České republiky.

Název práce: Analýza blockchainu používaného pro Bitcoin

Autor: David Surma

Ústav: Informatický ústav Univerzity Karlovy

Vedoucí diplomové práce: Ing. David Hartman, Ph.D., Informatický ústav Univerzity Karlovy

Abstrakt: Tato práce se zabývá analýzou blockchainu používaného pro Bitcoin. Blockchain je distribuovaná databáze všech uskutečněných transakcí s touto kryptoměnou. Její veřejná dostupnost představuje možnost zkoumání přesunů prostředků mezi veškerými uživateli. Ti však v transakcích vystupují pod anonymními adresami, jejichž počet je prakticky neomezený. Hlavním cílem naší práce je nalézt klastrování adres odpovídající jejich příslušnosti k reálným uživatelům. V práci navrhujeme nové heuristiky, které lze při klastrování využít. Hlavním přínosem je metoda, která využívá vlastnosti velmi rychle po sobě vytvořených transakcí. Dále analyzujeme problém vzniku superklastru obsahujícího neúměrně velkou část adres a navrhujeme způsob, jakým lze klastr vhodně rozdělit.

Klíčová slova: Bitcoin blockchain klastrovací algoritmy transakce směnárna

Title: Analysis of blockchain used for Bitcoin

Author: David Surma

Institute: Computer Science Institute of Charles University

Supervisor: Ing. David Hartman, Ph.D., Computer Science Institute of Charles University

Abstract: This thesis deals with the analysis of the blockchain used for Bitcoin. Blockchain is a distributed database of all transactions made with this cryptocurrency. Its public availability represents the possibility of examining the transfer of funds between all users. However, they appear in transactions under anonymous addresses, the number of which is practically unlimited. The main goal of our work is to find a clustering of addresses corresponding to their belonging to real users. In this work, we propose new heuristics that can be used in clustering. The main benefit is a method that uses the properties of transactions created very quickly one after the other. Furthermore, we analyze the problem of the formation of a supercluster containing a disproportionately large number of addresses and propose a way in which the cluster can be appropriately partitioned.

Keywords: Bitcoin blockchain clustering algorithms transactions exchange

Obsah

Úvod **2**

1 Blockchain **3**
1.1 Adresy . . . 4
1.2 Transakce . . . 5
1.3 Peněženky . . . 8
1.4 Síť . . . 9
1.5 Útoky . . . 10

2 Rozpoznávání entit **12**
2.1 Vlastnosti transakcí . . . 13
2.1.1 Multi-input heuristika . . . 13
2.1.2 Heuristiky pro hledání adresy vratky . . . 14
2.2 Off-chain informace . . . 16
2.3 Vlastnosti chování uživatelů . . . 17

3 Související práce **19**
3.1 Charakteristiky grafu . . . 19
3.2 Charakteristiky entit . . . 21

4 Analýza **23**
4.1 Klastrování . . . 24
4.2 Rozdělení superklastru . . . 29

Závěr **31**

Seznam použité literatury **32**

Seznam použitých termínů **35**

A Přílohy **36**

Úvod

Bitcoin je nejstarší fungující kryptoměna s největší tržní kapitalizací dosahující stovek miliard amerických dolarů. Umožňuje lidem posílat si navzájem virtuální peníze bez nutnosti využití centrální autority. Jako první vyřešil problém, jak v takovém případě zabránit dvojímu utracení stejných peněz. Princip jeho fungování popsal v roce 2008 člověk pod pseudonymem Satoshi Nakamoto. O rok později byl celý projekt decentralizované kryptoměny spuštěn a od té doby běží nepřetržitě dodnes.

Decentralizace je základní myšlenka, na které Bitcoin spočívá. Jednoduše zabraňuje možnosti celý projekt měnit na základě vůle jednotlivců či ho celý zastavit. Uživatelé si bitcoiny posílají v peer-to-peer síti, kde neexistuje žádné centrum schvalující transakce jednotlivým uživatelům. V této síti, kde potenciálně nikdo nikomu nemůže důvěřovat, musí existovat pravidla, jejichž dodržování umožňuje většině uživatelů shodnout se na společné věci. U Bitcoinu jsou tato pravidla nastavena takovým způsobem, že jejich dodržování přináší vyšší benefity než jejich porušování. Implementace těchto jednoduchých myšlenek již řadu let funguje bez větších problémů a díky vyspělejším uživatelům se stává stále bezpečnější. Zaútočit na popsané principy díky decentralizaci buď nelze, nebo se to útočníkům nevyplácí. Proto lze předpokládat, že Bitcoin bude na rozdíl od většiny ostatních kryptoměn fungovat i nadále.

Databáze, do které se zapisují všechny uskutečněné transakce bitcoinů, se nazývá blockchain. Oproti jiným doménám se liší v tom, že je veřejně dostupná všem uživatelům. Díky tomu láká výzkumníky z celého světa, kteří se snaží sledovat toky virtuálních peněz ať už z čistě vědeckých důvodů či z těch praktických. Mezi ně patří odhalování podvodníků nebo třeba snaha o predikci vývoje ceny bitcoinu. Uživatelé však na blockchainu vystupují pod pseudonymy nazývanými adresy a skrývají tak svou reálnou identitu před okolním světem. Takových adres přitom mohou vytvářet prakticky neomezené množství. Přesto existují způsoby, kterými lze tato anonymní data analyzovat a odhalit alespoň část informací. Naše práce se zabývá právě možnostmi této analýzy.

Zkoumaný problém lze obecně popsat jako učení bez učitele, jehož cílem je nalezení ideálního klastrování skupin adres vlastněných stejnou entitou. Vzhledem k počtu adres převyšujícímu jednu miliardu a srovnatelným počtem transakcí souvisí s tímto problémem dobývání znalostí z velkých dat. Vlastnosti transakcí se přitom v čase vyvíjí a někteří uživatelé je mění záměrně. Použité algoritmy tak musí být uzpůsobené nejen extrémnímu množství dat, ale i různým nestandardním jevům, které se v nich vyskytují. Jejich výsledky lze navíc jen složitě testovat, neboť z decentralizace Bitcoinu rovněž plyne nedostupnost většího množství ověřených dat pro interpretaci výsledků.

V kapitole 1 postupně popisujeme prvky, ze kterých se blockchain skládá. Uvádíme rovněž pravidla, kterými se jednotlivé součásti, jakož i celý blockchain řídí. V kapitole 2 jsme přehledně roztřídili možné přístupy, které lze použít pro rozpoznání entit stojících za jednotlivými adresami. Kapitola 3 potom obsahuje přehled o postupech a výsledcích souvisejících prací ostatních autorů. V kapitole 4 popisujeme vlastní přístup ke klastrování a analyzujeme jeho výsledky na aktuálních datech.

1. Blockchain

Bitcoin s velkým počátečním písmenem je komplexní označení decentralizované sítě uzlů, protokolů a softwaru, který uživatelé používají, zatímco bitcoin je označení pro jednotku kryptoměny existující a fungující v této síti. Jeden bitcoin lze rozdělit až na 10^8 dílů. Nejmenší přenositelná jednotka byla pojmenována po tvůrci Bitcoinu Satoshi Nakamotovi. Platí tedy vztah $1\,\mathrm{BTC} = 100\,000\,000\,\mathrm{SAT}$, kde BTC je zkratka pro bitcoin a SAT pro satoshi.

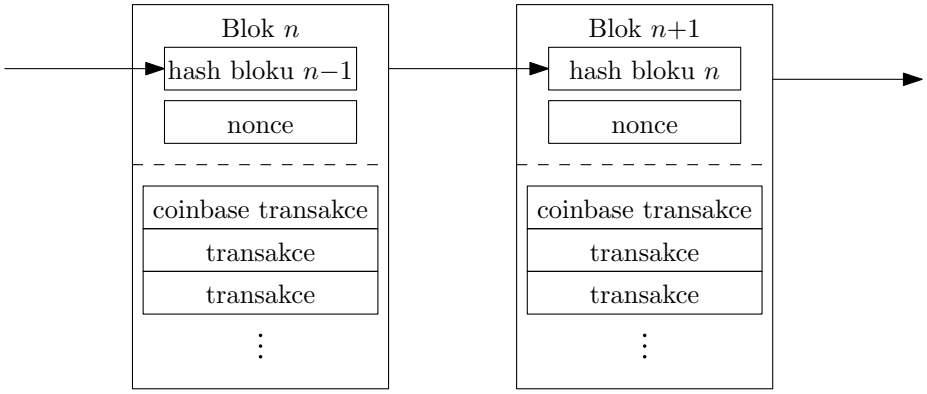
Počet bitcoinů v oběhu se řídí jasnými pravidly. Nové bitcoiny jsou uvolňovány jakožto odměna takzvaným těžařům, kteří ověřují validitu probíhajících transakcí a zabezpečují ochranu sítě před útoky, jako je problém dvojího utracení stejných peněz. Odměna jim je vyplácena v průměru každých deset minut, její velikost se každé čtyři roky zmenšuje o polovinu. Od roku 2009 se tedy snížila z původní hodnoty 50 BTC na současných 6,25 BTC. Dosud bylo vytěženo přes 19,3 milionů bitcoinů, což představuje 92 % z konečného počtu 21 milionů, kterých stanoveným tempem dosáhneme kolem roku 2140.

Uživatelé vystupují na bitcoinové síti pod náhodně vygenerovanými pseudonymy, kterým se říká *adresy*. Ty jsou používány v rámci transakcí pro přesun hodnoty z jedné adresy na druhou. Každý uživatel však nemusí používat jedinou adresu, ale může jich vlastnit prakticky neomezené množství. Ostatně vytvářet novou adresu pro každou příchozí transakci doporučoval již Nakamoto [1]. Takový přístup zvyšuje anonymitu a bezpečnost sítě, protože ostatní uživatelé si obecně nedokáží spojit nově vzniklou adresu s těmi, které už daný uživatel dříve použil. Na druhou stranu pracovat s větším množstvím adres jednotlivě by bylo nepraktické. Proto byly vyvinuty takzvané *peněženky* - softwarové nebo i hardwarové nástroje, které dokáží soubor takových adres zastřešit. Peněženek může mít každý uživatel rovněž více.

Prováděné transakce jsou těžaři ověřovány a seskupovány do bloků o maximální velikosti 1 MB. Tyto bloky jsou následně v rámci procesu těžby připojovány jeden za druhý. Takto vytvářené databázi obsahující veškeré uskutečněné transakce se díky jejímu tvaru říká *blockchain*. Spolu s provedenými transakcemi je v hlavičce každého bloku uložen mimo jiné i hash předchozího bloku, čímž se zabraňuje nepozorovatelnému upravení jednou přijatých transakcí potenciálními útočníky, viz Obrázek 1.1.

Proces těžby je výpočetně náročný. Těžaři musí najít hodnotu takzvané *nonce* o velikosti 32 bitů takovou, že hash těženého bloku obsahujícího tuto nonce bude mít na svém začátku alespoň n nul, kde n je parametr náročnosti těžby. Složitost této kryptografické úlohy se v čase mění podle aktuálního výkonu těžařů tak, aby v průměru trvalo jeden blok vytěžit deset minut. Vzhledem k tomu, že použitá hashovací funkce SHA256[1] je jednosměrná, je jediným efektivním způsobem, jak správnou hodnotu nonce najít, zkoušet hashovat jednu možnost po druhé. Vynaložená energie garantuje správnost a nezměnitelnost transakcí zapsaných do blockchainu, neboť představuje překážku pro potenciální útočníky snažící se údaje o proběhlých transakcích měnit. Ověření správnosti hashů je naopak výpočetně nenáročné a může jej provádět každý uživatel.

[1] Secure Hash Algorithm, hashovací funkce zobrazující vstupní řetězce na výstup s konstantní velikostí, v tomto případě 256 bitů.



Obrázek 1.1: Struktura blockchainu. Bloky transakcí jsou na sebe napojené pomocí hashů. V hlavičce figuruje rovněž nalezená hodnota nonce. První transakce v bloku je speciální coinbase transakce.

Bitcoin nemá žádné centrum, kterému by museli uživatelé důvěřovat, že jím publikovaná posloupnost bloků je ta správná. Blockchain je naopak uložený na velkém množství nezávislých uzlů zároveň. Aby se byla většina uživatelů schopna shodnout na platné verzi blockchainu i bez centrální autority, je potřeba dvou základních věcí. Předně celý blockchain od prvního vytěženého bloku až po ten momentálně poslední musí být veřejně dostupný. To umožňuje uživatelům kontrolovat, že jim posílané prostředky na blockchainu skutečně existují. Dále se musí stanovit, jak rozlišit aktuálně správnou verzi blockchainu od ostatních. Bitcoin využívá jakožto mechanizmus konsenzu takzvaný důkaz o vykonané práci, podle kterého je platný vždy ten nejdelší publikovaný řetězec validních bloků.

1.1 Adresy

Na blockchainu je již použita více než miliarda různých adres. Jelikož lze vyhledat všechny transakce, ve kterých libovolná adresa figuruje, můžeme zjistit, že z nich pouze 43 milionů vlastní alespoň jeden satoshi. Ostatní adresy byly někdy v historii použity, ale veškerá hodnota z nich byla posléze odeslána pryč. Vzhledem k doporučení, aby uživatelé pro každou příchozí transakci vytvářeli novou adresu, se většina z nich už nikdy znovu na blockchainu neobjeví.

Poznámka. Množinu všech adres figurujících na blockchainu budeme značit $\mathcal{A}$. Vzhledem k povaze blockchainu je tato množina závislá na jeho délce a na verzi, neboť nejdelších posloupností bloků může existovat více současně. Pro zachování přehlednosti budeme v celé práci popisovat jednu statickou verzi blockchainu.

Adresy jsou vytvářeny na základě principů asymetrické kryptografie. Nejprve se vygeneruje náhodný soukromý klíč o velikosti 256 bitů a z něj se pomocí algoritmu ECDSA[2] odvodí odpovídající veřejný klíč. Ten bývá často zaměňován se samotnou adresou. Ta je však z veřejného klíče pouze odvozena a reprezentuje jej na blockchainu v uživatelsky přívětivější podobě alfanumerického řetězce tvořeného mezi 26 a 62 znaky. Vlastnictví soukromého klíče představuje jediný způsob, jak nakládat s bitcoiny na blockchainu, neboť každá transakce z libovolné adresy

[2]Elliptic Curve Digital Signature Algorithm.

musí být podepsána pomocí odpovídajícího soukromého klíče. Těžaři i ostatní uživatelé ověřují autenticitu prováděných transakcí kontrolou správnosti přiložených podpisů pomocí veřejného klíče bez toho, aniž by museli znát ten soukromý.

V době vzniku úvodního článku představujícího koncept Bitcoinu (Nakamoto [1]) se jakožto adresy používaly skutečně samotné veřejné klíče. Pro ověření autenticity transakcí stačilo jednoduše zkontrolovat přiložený podpis. Nevýhodou a potenciálním bezpečnostním rizikem však bylo zveřejňování veřejného klíče adresáta prostředků, jelikož prolomení šifrovacího algoritmu ECDSA by v budoucnu znamenalo možnost krádeže všech bitcoinů uložených na takto vypadajících adresách. Proto už od počátku Bitcoinu existovala možnost, jak zveřejňovanou adresu pouze odvodit z veřejného klíče pomocí jednosměrné funkce. Bezpečnější podoba adres však na druhou stranu zesložitila proces ověřování transakcí.

Podle způsobu odvození adresy z veřejného klíče a tím souvisejícího podepisovacího a ověřovacího mechanizmu se adresy dělí na několik typů. Lze je rozeznat podle počátečních znaků. Číslicí 1 začínají původní P2PKH[3] adresy. Ty jsou odvozeny z veřejných klíčů jakožto jejich SHA256 hashe upravené do textové podoby pomocí kódování Base58. Pro ověření transakcí odesílaných z těchto adres však potřebují ostatní uživatelé znát odpovídající veřejný klíč, odesilatel jej proto musí přiložit do transakce spolu s podpisem tajným klíčem. Od okamžiku odeslání prostředků z P2PKH adresy její zabezpečení opět závisí pouze na algoritmu ECDSA. Bezpečnostním rizikem je tedy až její případné znovupoužití.

Nejen z bezpečnostních, ale i z praktických důvodů byly vytvořeny další typy adres. Ty umožnily například spravovat jednu adresu pomocí více párů klíčů nebo zmenšit velikost podpisových dat ukládaných do blockchainu. Jejich detailní popis přesahuje rámec naší práce, podrobně je popsal Zaghloul a kolektiv [2]. Patří mezi ně P2SH[4] adresy, které lze rozpoznat podle počáteční číslice 3. V dnešní době nejrozšířenější jsou P2WPKH[5] neboli Bech32 adresy začínající znaky bc1. Kromě toho ještě existují takzvané multisig adresy, ke kterým je při vzniku vytvořeno $n \geq 2$ párů klíčů. Pro schválení transakce z takových adres je potřeba alespoň m podpisů, kde $m \leq n$ je rovněž při vzniku adresy stanovená konstanta.

1.2 Transakce

Jak už bylo naznačeno v úvodu, veškerá existence bitcoinů je spjata se seznamem transakcí uložených na blockchainu. Uživatel nemá přiřazen jeden zůstatek bitcoinů přímo ke své adrese, má jen možnost nakládat s obnosy, které na jeho adresu byly dříve zaslány. Tyto obnosy může přeposílat dál, narozdíl od klasických mincí či bankovek je přitom může podle potřeby slučovat či dělit a vytvářet tak nové obnosy čítající libovolný počet satoshi. V rámci běžné transakce nemůže být žádná hodnota vytvořena, ale ani ztracena. Standardní transakce tedy nemění počet bitcoinů v oběhu, jen přesouvají právo nakládat s nimi mezi různými adresami.

Definice (TXO)**.** *Transakční výstup (TXO) je dvojice* (a,h)*, kde* $a \in \mathcal{A}$ *je adresa a* $h \in \mathbb{N}$ *je přenášená hodnota vyjádřená v počtu satoshi.*

[3]Pay to Public Key Hash, například 1LQoWist8KkaUXSPKZHNvEyfrEkPHzSsCd.

[4]Pay to Script Hash, například 3JJmF63ifcamPLiAmLgG96RA599yNtY3EQ.

[5]Pay to Witness Public Key Hash, např. bc1qazcm763858nkj2dj986etajv6wquslv8uxwczt.

Transakce se skládají ze dvou částí - vstupů a výstupů. Obě jsou tvořeny posloupnostmi TXO. Vstupem nové transakce může být pouze výstup nějaké předchozí, proto se pro obě části používá stejný název. Zatímco TXO figurující na vstupu se během zapsání transakce do blockchainu utratí a už je nikdy nepůjde použít podruhé, výstupní TXO vzniknou a stanou se použitelnými na vstupech dalších transakcí. Bylo by nepraktické vytvářet TXO pro každý přenášený satoshi, lze jich tedy poslat libovolný počet naráz. Aby se však daly větší obnosy dělit a menší naopak slučovat, může mít transakce vstupních i výstupních TXO více. Pro dosud neutracené transakční výstupy se používá označení UTXO[6].

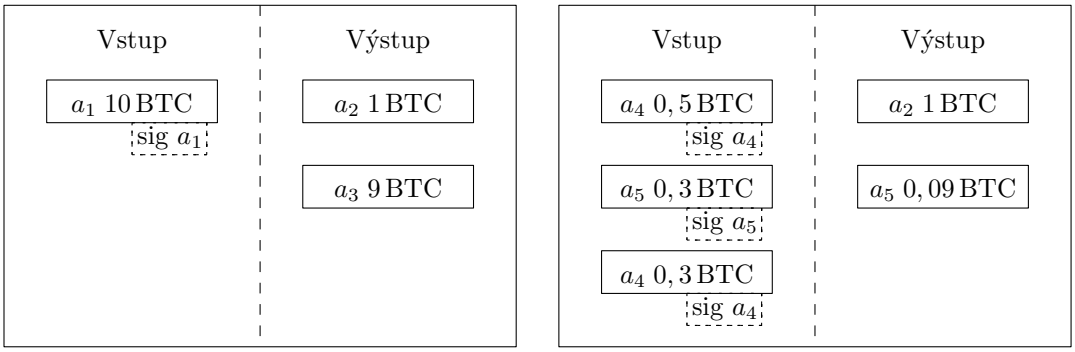
V minulé podkapitole jsme popisovali adresy a s nimi spojený mechanizmus podepisování transakcí. Tyto podpisy jsou přiloženy u všech transakčních vstupů, viz Obrázek 1.2. Jsou vytvořeny z hashe posloupností vstupních a výstupních TXO. Prokazují znalost tajných klíčů k adresám obsaženým v použitých UTXO a tím autorizují jejich utracení. Podpisy nacházející se ve vstupech jedné transakce jsou na sobě nezávislé. Společně utratit několik UTXO tedy může více uživatelů, v tom případě se však musí nejprve domluvit a shodnout na výsledné podobě transakce.

V rámci běžné transakce nemůže vzniknout nová hodnota, takže součet výstupních satoshi může být nejvýše roven součtu satoshi na vstupu. Celkový výstup bývá obvykle o něco menší než vstup, tento rozdíl je brán jako implicitní poplatek těžařům, kteří danou transakci zapíší do blockchainu. Často uživatel nedokáže z jemu patřících UTXO poskládat vstup tak, aby přesně odpovídal částce, kterou potřebuje odeslat. Jelikož se veškerý vstup vždy celý utrácí, má většina transakcí na výstupu jedno zvláštní UTXO tvořené takzvanou *change adresou*, která patří odesilateli, a přebývající částkou, kterou si na ni vrací. Toto UTXO je zvláštní obecně jen pro odesilatele transakce, neboť pouze on s jistotou ví, že adresa použitá pro vratku je jeho. Ostatní uživatelé vidí na blockchainu obyčejnou transakci s alespoň dvěma standardními UTXO na výstupu.

Pro odeslání jednoho bitcoinu na adresu a_2 může například uživatel utratit jediné UTXO obsahující 10 BTC na adrese a_1 a poslat si zbývajících 9 BTC zpátky na nově vytvořenou adresu a_3. Nebo může v rámci vstupu spojit tři menší UTXO na dvou adresách a_4 a a_5, které mají dohromady hodnotu 1,1 BTC, a poslat si vratku v hodnotě 0,09 BTC zpátky na jednu ze vstupních adres, viz Obrázek 1.2. Přebytečných 0,01 BTC získá automaticky úspěšný těžař v rámci své odměny po vytěžení patřičného bloku.

Speciální typ transakcí tvoří takzvané *coinbase* transakce. Ty se nachází na prvním místě v každém bloku, nemají žádné vstupy, mají pouze jeden výstup. Tím je UTXO obsahující adresu těžaře, který daný blok vytěžil a připojil do blockchainu. Hodnota tohoto UTXO je závislá na dvou faktorech. První část tvoří nově vzniklé bitcoiny, jejichž počet se řídí stanovenými pravidly. Její hodnota se tedy postupně sníží až na nulu. I poté však budou těžaři za svou práci odměňováni, neboť druhou část tvoří poplatky implicitně uvedené v transakcích ve vytěženém bloku. Pomocí poplatků jsou těžaři motivováni k přednostnímu zařazení výhodnějších transakcí do jimi těžených bloků, což může vzhledem k omezené velikosti bloků a v průměru konstantní rychlosti jejich těžby způsobovat zvyšování minimálních poplatků potřebných pro schválení transakce.

[6]Unspent Transaction Output.



Obrázek 1.2: Dvě transakce posílající 1 BTC na adresu a_2. V obou případech vznikají na výstupu dvě nová UTXO, první z nich jsou skutečně odeslané novému majiteli a druhá reprezentují vrácení prostředků odesilateli. Vlevo se rozděluje jedno velké UTXO a vratka putuje na nově vytvořenou adresu a_3, vpravo se naopak slučují tři menší a drobné se vrací na jednu ze vstupních adres. Navíc tato transakce obsahuje implicitní poplatek 0,01 BTC. U vstupních UTXO jsou přiloženy podpisy tajnými klíči odpovídajících adres.

Definice (Transakce)**.** *Transakce t je dvojice (I_t, O_t), kde I_t a O_t jsou posloupnosti TXO.*

Poznámka. V pracích ostatních autorů bývají transakce popisovány jen neformálně. O jejich formální definici se pokusili Zhang, Wang a Luo [3], kteří však místo o posloupnostech TXO hovoří o množinách TXO. To je však nesprávné označení, neboť na vstupu i na výstupu se mohou vyskytovat dvě různé TXO, které přitom obsahují stejnou adresu i hodnotu. Ermilov a Panov [4] pro označení vstupů a výstupů použili vhodnější termín multimnožina. V případě utracení jednoho TXO, jenž se na výstupu transakce ve stejné podobě vyskytuje vícekrát, však stále nelze jednoznačně identifikovat, které bylo utraceno a které nikoliv. Proto jsme v naší definici vstupy i výstupy označili jako posloupnosti.

Transakci jsme formálně definovali jen jako dvě posloupnosti TXO. To, které transakce jsou správné a platné, plyne až ze samotného blockchainu. Přitom blockchain je pouze posloupnost bloků a těmi jsou v zásadě jen posloupnosti transakcí, metadata jako podpisy, hashe a nonce jsou do nich přidávána jen z důvodu zabezpečení. Nad blockchainovou databází transakcí lze tedy definovat funkci transakčních výstupů $U : \mathbb{N}^3 \to \mathcal{U}$, kde $\mathcal{U}$ značí množinu všech TXO. Výstupem této funkce $U(i,j,k)$ pro trojici indexů tvořenou pořadovým číslem bloku i, číslem transakce j a číslem výstupu k je platný transakční výstup nebo prázdná množina, pokud některý z indexů převýšil délku odpovídající posloupnosti.

Transakčními vstupy jsou potom posloupnosti tvořené výstupy této funkce splňující soubor dříve popsaných podmínek. Na základě funkce U tedy můžeme pomocí pořadového čísla bloku $i \in \mathbb{N}$ a čísla transakce $j \in \mathbb{N}$ identifikovat všechny platné transakce pomocí funkce $T : \mathbb{N}^2 \to \mathcal{T}$, kde $\mathcal{T}$ značí množinu všech transakcí a $T(i,j) \mapsto (I_t,O_t)$, přičemž $O_t = (U(i,j,k)|k \in \mathbb{N})$ a $I_t = (U(i',j',k'))$ pro vhodné trojice indexů $(i',j',k') \in \mathbb{N}^3$.

Můžeme dále zavést dvě pomocné funkce $I,O : \mathcal{T} \to 2^{\mathcal{A}}$, které budou označovat množinu vstupních, respektive výstupních adres v dané transakci. Pro trans-

akci $t = (I_t,O_t) \in \mathcal{T}$ s m vstupy a n výstupy definujme $I(t) = \{a_k | I_t = (a_k,h_k)_{k=1}^m\}$ a obdobně $O(t) = \{a_l | O_t = (a_l,h_l)_{l=1}^n\}$.

Poznámka. Pomocí $\mathcal{T}$ budeme dále značit jen množinu všech platných transakcí. Zároveň množinou $\mathcal{A}$ budeme rozumět množinu použitých adres, tedy těch, které se nachází ve výstupu nějaké transakce z množiny $\mathcal{T}$. Její význačnou podmnožinu tvoří adresy, které byly použity jen jednou a lze je tedy označit za jednorázové.

Definice (Jednorázová adresa)**.** *Adresa $a \in \mathcal{A}$ je jednorázová právě tehdy, když $\exists!(i,j,k) \in \mathbb{N}^3$ tak, že $U(i,j,k) = (a,h)$ pro nějaké $h \in \mathbb{N}$.*

1.3 Peněženky

Peněženky neboli *walletky* jsou nástroje určené pro správu adres a klíčů. Kromě bezpečného vytváření a ukládání soukromých klíčů řeší za uživatele veškeré technické detaily transakcí, takže pracovat s peněženkou lze i bez znalosti fungování Bitcoinu. Peněženky dokáží ke spravovaným adresám vyextrahovat z blockchainu všechny UTXO a zobrazovat tak uživatelům aktuální počet bitcoinů v jejich vlastnictví, jako by to byl zůstatek na běžném účtu. Představa běžného účtu je umocňována jednoduchostí vytváření transakcí, kdy uživateli stačí specifikovat jen částku a cílovou adresu, peněženka se postará o vše ostatní. Nalezne tedy taková UTXO, která dohromady pokryjí odesílanou částku navýšenou o drobný poplatek těžařům. Pokud je potřeba, vytvoří k uživatelem specifikovanému výstupu automaticky také UTXO s vratkou přebývajících peněz. Pro toto UTXO použije buď jednu z uložených adres, nebo častěji vytvoří úplně novou adresu spolu s patřičnými klíči. K celé transakci poté peněženka připojí podpisy tajnými klíči všech adres, které byly použity ve vstupech, a odešle ji do bitcoinové sítě na ověření a připojení do blockchainu.

Jsou to právě peněženky, které podle nastavených parametrů a použitých algoritmů ovlivňují finální podobu většiny transakcí. Mají ve svém nastavení specifikováno, jestli utrácet raději větší počet menších UTXO, nebo naopak menší počet větších, jestli pro vracení přeplatků znovu používat jednu speciální adresu, nebo vždy vytvářet pro tyto účely novou. Mají rovněž nastaven výchozí typ všech vytvářených adres.

Peněženky můžeme podle míry jejich zabezpečení, které souvisí zejména se způsobem připojení k internetu, rozdělit na dva typy. Tím prvním jsou online peněženky neboli *hot walletky*, které jsou trvale připojené k síti a umožňují tak rychlé zprostředkování transakcí. Na jejich adresách se obvykle uchovává jen menší množství bitcoinů postačující pro běžné platby, neboť je zde větší riziko hackerských útoků a odcizení uložených klíčů. Druhým typem jsou offline peněženky neboli *cold walletky*. To bývají speciální hardwarová zařízení s jediným účelem, kterým je bezpečné vygenerování a uložení klíčů. Používají se k dlouhodobému uchovávání větších obnosů. Pro odeslání transakce je nutné offline peněženky nejprve spojit se zařízením připojeným k síti.

Přestože je nakládání s bitcoiny podmíněno vlastnictvím tajných klíčů k jistým adresám, někteří uživatelé vlastní bitcoiny nepřímo přes zprostředkovatele. Těmi bývají společnosti zabývající se směnou bitcoinů za jiné měny, provozováním online peněženek pro své zákazníky nebo těžbou. Patří mezi ně kryptoměnové

směnárny a burzy nebo takzvané *mining pooly*, což jsou společenství těžařů. Společnosti často poskytují více služeb zároveň. Mívají své peněženky také rozdělené na cold walletky určené pro bezpečné uložení většiny prostředků svých zákazníků a hot walletky, které zabezpečují přijímání vkladů a zároveň odesílání spravovaných bitcoinů na adresy zákazníků, pokud si o to požádají.

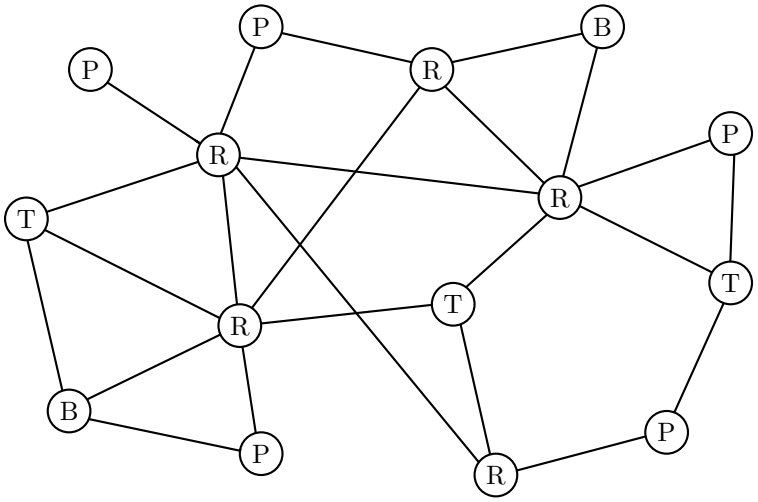
Transakce mezi peněženkami jednotlivců a peněženkami burz a směnáren představují hranici mezi takzvanými *on-chain* a *off-chain* transakcemi. Mezi ty první patří všechny dosud popisované transakce, jsou to tedy přesuny prostředků, které lze vyčíst z veřejného blockchainu. Off-chain transakce může uživatel provádět po zaslání bitcoinů na adresu zprostředkovatele. Ten připíše patřičnou částku k účtu odesilatele, který ji může dále převádět na jiné měny nebo například spekulovat na růst či pokles ceny obchodováním na burze. Garantem všech takových transakcí je samotný zprostředkovatel, ve skutečnosti nemají s blockchainem nic společného.

Pro úplnost dodejme, že existují i jiné off-chain transakce, které se nezapisují do blockchainu. Příkladem může být poslání samotných klíčů novému majiteli nebo použití druhé vrstvy Bitcoinu, takzvané Lightning Network. Ta řeší problém škálovatelnosti bitcoinových transakcí, neboť jejich počet je shora omezen velikostí bloků a frekvencí jejich těžby. Pokud v síti vzniká více než několik málo tisíc transakcí za deset minut, začíná se tvořit fronta nepotvrzených transakcí, ze kterých si těžaři racionálně vybírají ty s největšími poplatky. K potvrzení malých transakcí tak vůbec nemusí dojít. Právě Lightning Network umožňuje posílat menší obnosy bitcoinů rychle a s nízkými poplatky, jelikož většinu těchto transakcí nezapisuje do blockchainu. Využívá přitom konceptu multisig adres, přes které je spojena se základní vrstvou Bitcoinu. Pro správu transakcí na Lightning Network se používají speciální peněženky, ty se však z pohledu uživatele takřka neliší od těch klasických.

1.4 Síť

Myšlenka Bitcoinu je založena na decentralizaci. Bitcoinová síť, přes kterou uživatelé posílají své transakce a těžaři je posléze připojují do bloků, funguje nad internetovým protokolem a má peer-to-peer architekturu, viz Obrázek 1.3. Skládá se tedy z množiny rovnoprávných *uzlů*, které se mohou do sítě libovolně připojovat a odpojovat dle svého uvážení. Přestože jsou si všechny rovny, mohou se od sebe lišit funkcí, ke které v síti slouží a kvůli které je jejich majitelé vytvořili. Popíšeme je obdobně jako Antonopoulos [5]. Každý uzel může zároveň vykonávat libovolnou kombinaci následujících čtyř funkcí.

První funkci jsme již popsali v minulé podkapitole. Je jí tedy funkce peněženky, jenž udržuje přehled o množině UTXO a dokáže z nich vytvářet nové transakce. Po podepsání je rozesílá sousedním uzlům, které se mají postarat o jejich zapsání do blockchainu. Další funkcí uzlů je tedy těžba. Těžaři nové transakce od sousedních uzlů přijímají, kontrolují a ukládají do své prioritní fronty, tzv. *mempoolu*. Z něj pak podle vlastního uvážení vybírají transakce do bloku, který se sami snaží vytěžit. Takový blok musí obsahovat speciální coinbase transakci. Její výstup těžaři směřují na vlastní adresu, aby se v případě úspěšného vytěžení svého bloku stali majiteli nově emitovaných bitcoinů. Pokud se jim podaří najít správnou hodnotu nonce rychleji než ostatním, odešlou ji společně s celým



Obrázek 1.3: Ilustrace bitcoinové sítě. Uzly označené P představují peněženky uživatelů, T těžaře a R routovací uzly přeposílající nové transakce a bloky. Písmeno B značí uzly uchovávající a distribuující celý blockchain. Hrany se nachází mezi uzly, které spolu přímo komunikují.

novým blokem zbytku sítě na ověření. Jestliže ostatním těžařům přijde nový validní blok, tedy blok obsahující validní transakce a správné hashe, aktualizují si podle něj svůj mempool a začnou pracovat na těžbě bloku následujícího. Příklad odeslání a schválení konkrétní transakce je zachycen na Obrázku A.1.

Aby celá síť fungovala a byla robustní vůči případnému výpadku některých uzlů, je potřeba takzvané routovací funkce. Většina uzlů tedy jen přeposílá nově příchozí transakce či celé bloky dále svým sousedům a zabezpečuje tak distribuci informací po celé síti. Obvykle je přitom rovnou kontroluje, aby se sítí zbytečně nešířily neplatné zprávy. Poslední potřebnou funkcionalitou je samotné uložení blockchainu. Část uzlů uchovává celý řetězec bloků od prvního až po ten momentálně poslední přijatý a svou verzi blockchainu dokáže přeposlat sousedním uzlům například při jejich vzniku nebo znovupřipojení. Těm pro získání důvěry ve správnost této verze stačí překontrolovat posloupnost hashů obdržených bloků.

1.5 Útoky

Jak jsme již psali v úvodu kapitoly, způsobem dosažení konsenzu v bitcoinové síti je důkaz o vykonané práci, podle kterého je platný aktuálně nejdelší řetězec validních bloků. Takto jednoduchá podmínka může samozřejmě způsobovat problémy, neboť nejdelších validních řetězců se může v síti objevit současně více. Některé z nich mohou být přitom vytvořeny útočníky snažícími se poslední vytěžené bloky měnit ve svůj prospěch. Veškerá důvěra v blockchain je tedy postavena na předpokladu, že nadpoloviční většina těžařů se chová podle pravidel a nesnaží se na princip Bitcoinu útočit. Míru zabezpečení blockchainu přitom představuje jejich celkový výkon udávaný v počtu prováděných hashů za sekundu, tzv. *hashrate*. Pokud je celkový hashrate příliš nízký, hrozí riziko takzvaného 51% útoku, při kterém jeden těžař dokáže dlouhodobě vytvářet nejdelší verzi blockchainu sám a stává se tak de facto centrální autoritou, viz Puthal a kol. [6]. Pokud útočník nemá vyšší hashrate než všichni ostatní těžaři dohromady, klesá pravděpodob-

nost, že by dokázal vytěžit několik po sobě jdoucích bloků rychleji než všichni ostatní, exponenciálně v závislosti na délce posloupnosti takovýchto napadených bloků, viz Nakamoto [1].

I když se útočníkovi povede připojit do blockchainu několik jím vytvořených bloků, má velmi omezené možnosti podvodu. Aby jeho verzi blockchainu zbytek sítě nakonec přijal, musí v ní být dodržena veškerá pravidla platná pro kteréhokoliv jiného těžaře. Útočník si tedy nemůže připsat na svůj účet libovolné množství bitcoinů, neboť nové bitcoiny mohou vznikat jen v rámci coinbase transakcí a jejich počet je dopředu známý. Nemůže ani ukrást prostředky z cizích adres, protože validní transakce musí být opatřeny správnými podpisy a útočník soukromé klíče většiny adres nezná. Cizí transakce dokáže jedině celé smazat, své odchozí transakce, ke kterým tedy zná odpovídající tajné klíče, dokáže případně modifikovat například změnou jejich adresáta.

Tento podvod se nazývá problém dvojího utracení neboli double-spending útok. Útočník při něm vytvoří běžnou transakci, ve které odesílá své prostředky jiné entitě, a čeká na obdržení protislužby od adresáta transakce například v podobě klasických peněz. Mezitím těží vlastní verzi blockchainu, ve které však zmíněnou transakci přesměroval na vlastní adresu. Pokud protislužbu obdrží ještě v momentě, kdy sám disponuje verzí blockchainu, která je aktuálně nejdelší, může jejím následným zveřejněním původní transakci fakticky stornovat. Proto je obecně doporučováno považovat příchozí transakci za platnou a trvale nezměnitelnou až poté, co se nachází v dostatečné hloubce, tedy když bylo vytěženo dostatečné množství následujících bloků. V literatuře je nejčastěji doporučován počet 6 (viz [2],[5],[7]).

Ještě jednodušší je provést útok na transakci, kterou adresát přijme za platnou už v době, kdy ještě vůbec nebyla zapsána do blockchainu a nachází se jen v mempoolech těžařů. Odesilatel v tom případě sám nemusí vytěžit jediný blok, stačí mu pouze vytvořit jinou transakci utrácející stejné UTXO a motivovat těžaře k jejímu přednostnímu vytěžení například pomocí vyšších poplatků.

Všechna tato rizika jsou známými nevýhodami používaného mechanizmu konsenzu. Pro úplnost dodejme, že jiným typem problémů jsou útoky na samotné peněženky uživatelů. Při jejich nedostatečném zabezpečení se může útočník zmocnit uložených tajných klíčů a tím de facto získat právo nakládat se všemi bitcoiny na odpovídajících adresách. Detailní popis známých útoků a způsobů, jak se jim bránit, vytvořil Zaghloul a kol. [2].

2. Rozpoznávání entit

Na první pohled vypadají bitcoiny jako anonymní peníze, jelikož jejich reální vlastníci skrývají svou identitu za nicneříkajícími náhodně vygenerovanými adresami. Bitcoin bývá spojován s praním špinavých peněz, placením výkupného hackerům a s různými dalšími ilegálními aktivitami, kde je snahou adresáta prostředků zůstat v anonymitě. Na druhou stranu dostupnost veškerých transakcí na blockchainu představuje obrovské množství informací, ze kterých lze spoustu věcí o uživatelích a jejich chování odvodit. Tato oblast přitahuje jak vědecký výzkum, tak i soukromé společnosti, které se zabývají analýzou a monitorováním transakcí na blockchainu nebo se snaží o zjišťování původu peněz.

Snaha o částečnou kontrolu nad posíláním prostředků přes Bitcoin již probíhá. Nejjednodušším způsobem, jak získat vlastní bitcoiny, je vytvořit si účet na některé z kryptoměnových směnáren a poslat jí klasické peníze výměnou za obdrženou transakci s patřičným obnosem bitcoinů z některé z jejích adres. Opačným způsobem lze zase směnit bitcoiny na klasické peníze. Jsou to právě některé směnárny, které musí dodržovat takzvané KYC[1] a AML[2] opatření, kvůli kterým vyžadují ověření totožnosti uživatelů a kontrolují, jestli k nim bitcoiny nepřichází z podezřelých adres. Na druhou stranu vytvoření jediné entity, která by všechny adresy třídila na povolené a zakázané, ze kterých by například směnárny nesměly přijímat vklady, by znamenalo de facto vznik centrální autority v bitcoinové síti, což jde přesně proti základní myšlence Bitcoinu. V rámci evoluce tak probíhá hra na kočku a myš, kdy uživatelé postupně přichází s novými metodami zvyšujícími jejich anonymitu a výzkumníci se je naopak snaží odhalit a rozkrýt.

Základním cílem výzkumu je odhalení identity reálných vlastníků bitcoinových adres. Těmi mohou být jednotlivci či celé společnosti, Ron a Shamir [8] pro ně zavedli obecné označení *entity*. Vzhledem k tomu, že obvykle není možné nebo ani potřebné znát konkrétní identitu vlastníka, autoři vytvořili několik obecných typů entit, které na blockchainu operují a které se snaží rozpoznat. Mezi nejčastěji používané třídy při klasifikační úloze strojového učení patří jednotlivci, burzy a směnárny, online walletky, mining pooly a různě pojmenovávané typy spojené s nelegálními činnostmi, viz například Harlev a kolektiv [9].

Vzhledem k tomu, že uživatelé na blockchainu nevystupují pod jedinou adresou, souvisí s hledáním entit stojících za jednotlivými adresami problém nalezení všech adres patřících jedné entitě. Na tento podproblém *klastrování* se budeme v celé naší práci soustředit. Obvykle se při klastrování používá funkce vzdálenosti mezi dvojicemi objektů, viz Kleinberg [10]. Klastrování bitcoinových adres je odlišné v tom, že přímý vztah mezi adresami lze odvodit jen z jejich společného výskytu v rámci stejné transakce. Přitom se však mohou v jedné transakci zřejmě vyskytovat i adresy odlišných entit. Při klastrování se tedy používají různé heuristiky snažící se využít slabá místa návrhu Bitcoinu nebo využít jisté vzory v chování uživatelů, jenž mohou napovědět o společném vlastnictví dvojic adres.

V následujících podkapitolách podrobně popíšeme možné způsoby rozpoznávání entit rozdělené podle původu použitých informací. Značení přitom zavedeme obdobným způsobem jako Zhang, Wang a Luo [3].

[1] Know Your Customer, ověření identity uživatele.

[2] Anti Money Laundering, ověření původu prostředků.

Definice (Klastrování)**.** *Klastrování C je rozdělení adres do po dvou disjunktních množin $C_1, C_2, \ldots, C_n$ takových, že*

$$\bigcup_{i=1}^{n} C_i = \mathcal{A}.$$

Poznámka. Množinu všech entit budeme značit $\mathcal{E}$. Pro entitu $e \in \mathcal{E}$ budeme značit $C_e \subseteq \mathcal{A}$ klastr adres, které jí patří. Za předpokladu, že každou adresu vlastní právě jedna entita, odpovídají množiny C_e pro $e \in \mathcal{E}$ ideálnímu klastrování množiny adres. Vzhledem k anonymitě uživatelů je v praxi nemožné ideální klastrování najít. Obvyklým postupem je zklastrování jen takové podmnožiny adres, u kterých je jejich příslušnost ke společnému majiteli jistým způsobem prokazatelná. Adresy jedné entity se tak mohou vyskytovat ve více klastrech. Zavedeme proto termín podklastrování.

Definice (Podklastrování)**.** *Podklastrování C' klastrování $C = C_1, \ldots, C_n$ je podrobnější rozdělení adres do množin $C'_1, \ldots, C'_m$ takových, že*

$$\forall i \leq m \; \exists j \leq n : C'_i \subseteq C_j.$$

2.1 Vlastnosti transakcí

Podoba byť jediné transakce může napovědět o společném vlastnictví několika použitých adres. Jak jsme již popsali v podkapitole 1.2, transakce jsou založeny na vytváření a utrácení celých UTXO. Z podstaty věci tak často dochází ke spojování více UTXO na vstupu jedné transakce a naopak k rozdělování výstupu na části odesílané novým majitelům a na vratku, která ve skutečnosti patří i nadále původnímu majiteli. Právě na tyto dva běžné jevy se zaměřují následující heuristiky.

2.1.1 Multi-input heuristika

Aby mohla být transakce schválena, musí obsahovat podpisy tajnými klíči patřícími ke všem adresám uvedeným ve vstupech. Transakce jsou přitom obvykle vytvářeny pomocí peněženek jednotlivých uživatelů. S vysokou pravděpodobností tedy patří všechny adresy vyskytující se společně ve vstupech jedné transakce stejné entitě. Tuto heuristiku lze aplikovat tranzitivně na všechny transakce vyskytující se na blockchainu. Je nejpoužívanější a nejefektivnější metodou pro klastrování adres.

Heuristika 1 (Multi-input heuristika)**.** *Adresy použité ve vstupech jedné transakce patří stejné entitě. Tedy*

$$\forall t \in \mathcal{T} \; \exists! e \in \mathcal{E} \; \forall a \in I(t) : a \in C_e.$$

Důsledek. Tranzitivní uzávěr multi-input heuristiky definuje klastrování C množiny $\mathcal{A}$. Nechť a_1,a_2,$a_3 \in \mathcal{A}$ a $t_1, t_2 \in \mathcal{T}$ jsou takové transakce, že $\{a_1$,$a_2\} \subseteq I(t_1)$ a $\{a_2$,$a_3\} \subseteq I(t_2)$. Potom

$$\exists! e \in \mathcal{E} : \{a_1,a_2,a_3\} \subseteq C_e.$$

O této slabině v anonymitě Bitcoinu věděl od počátku už Satoshi Nakamoto [1]. Multi-input heuristika se stala běžnou součástí prací zabývajících se analýzou blockchainu. Odpovědí na to bylo vytvoření takzvaných *mixovacích služeb*, na kterých se mohou různí uživatelé společně domluvit na finální podobě transakce, k té potom každý z nich nezávisle na ostatních připojí podpis svým soukromým klíčem a takto vytvořenou transakci míchající prostředky více entit pak odešlou k zapsání do blockchainu. Vzhledem k tomu, že uživatelé musí mít při použití mixovací služby hlubší znalosti o fungování Bitcoinu a zároveň snahu o zvýšení anonymity svých prostředků, nejsou tyto služby využívány běžně. Kromě pokročilejších uživatelů je využívají i zločinci ve snaze zabránit vystopování jim poslaných prostředků. Proto se na popsání takovýchto služeb zase zaměřuje část výzkumu, viz například Wu a kol. [11].

Spíše než samotné entity tato heuristika odhaluje jejich peněženky, neboť ty jsou zodpovědné za výběr jimi spravovaných UTXO do vstupů nových transakcí. Pokud nějaká entita používá více peněženek, například hot a cold, potom mezi nimi pravděpodobně provádí transakce. Jejich vstupy však vytváří vždy jen jedna z nich a výstupy spravuje případně peněženka druhá. Ta je bez dalších znalostí nerozlišitelná od peněženek cizích entit. Navíc prostředky z některých adres nemusí být nikdy utraceny společně s jinými, výsledkem klastrování pomocí této heuristiky je tedy jen podklastrování ideálního klastrování. Přitom však mohou být nesprávně zklastrovány adresy entit používajících mixovací služby. Dalším problémem může být změna majitele adresy ať už legální cestou při dobrovolném odeslání tajných klíčů nebo při jejich krádeži, neboť druhý vlastník může tuto původně cizí adresu použít na vstupu s vlastními adresami.

Pro úplnost dodejme, že vlastníky klíčů ke společné multisig adrese budeme považovat za jedinou entitu. Takové adresy vytváří buď skutečně jediná entita za účelem vyššího zabezpečení svých prostředků, nebo se na jejím vytvoření domluví více uživatelů. V tom případě však na sobě nejsou nezávislí, neboť spravují společný majetek a nakládají s ním odděleně od svého vlastního.

2.1.2 Heuristiky pro hledání adresy vratky

Dalším znakem typickým pro bitcoinové transakce je vytváření umělých výstupů reprezentujících vrácení přebytečných prostředků ze vstupu původnímu majiteli. Vzhledem k tomu, že přes 80 % všech transakcí obsahuje na výstupu alespoň dvě TXO, se můžeme domnívat, že change adresa je součástí většiny transakcí, viz předpoklad Mösera a Narayanana [12]. Správné rozpoznání change adres umožňuje sjednotit některé do té doby oddělené podklastry adres patřících společné entitě vytvořené například pomocí multi-input heuristiky.

Heuristik zaměřených na rozpoznání change adresy na výstupu transakce byla publikována celá řada (viz např. [3], [13], [14]). Všechny mají společnou jednu věc, a to nízkou spolehlivost, neboť jsou založeny jen na empirickém výzkumu. Vychází z běžného chování uživatelů, které však může libovolná entita jednoduše schválně či mimoděk změnit. Některé heuristiky mají navíc omezenou použitelnost, třeba jen pro transakce s právě dvěma výstupy nebo vzniklé do určité doby. Například software oficiální bitcoinové peněženky obsahoval podle Spagnuola a kol. [15] chybu, díky které se až do zveřejnění opravy v roce 2013 change adresa vyskytovala vždy v prvním výstupním TXO.

Největší množství poznatků využitelných pro detekci change adres zaznamenali Möser a Narayanan [12]. Stručně je popíšeme v následujících bodech:

Heuristika 2 (Heuristiky pro hledání change adres)**.**

- *Change adresa bývá stejného typu jako adresy na vstupu.*
- *Hodnota vratky bývá oproti ostatním výstupům méně zaokrouhlena.*
- *Pokud má jedno ze dvou výstupních TXO nižší hodnotu než některé vstupní, pak pravděpodobně obsahuje change adresu.*
- *Transakce, ve které vratka vzniká a ve které je později utracena, mívají stejné vlastnosti, jako například počet vstupů a výstupů nebo velikost poplatků.*
- *Change adresy jsou obvykle jednorázové, čili vyskytují se ve výstupu jediné transakce.*

Poslední jmenovanou heuristiku popíšeme podrobněji. Poprvé ji použil Meiklejohn a kol. [16] v roce 2013, od té doby se stala druhou nejpoužívanější heuristikou. Zaměřuje se na detekci jednorázových adres vytvořených automaticky peněženkou za účelem anonymního uložení vratky, tzv. *one-time change* adres. Teoreticky by mohly být jednorázové všechny adresy na blockchainu, pokud by každý uživatel, jak přijímající, tak posílající bitcoiny, dodržoval pravidlo o vytváření a používání neustále nových adres pro výstup každé transakce. Ve skutečnosti jsou uživateli vytvořené adresy náchylnější ke znovupoužití, ať už kvůli jejich nepozornosti či lenosti vytvořit novou adresu, nebo z toho důvodu, že odesilatel může jednoduše poslat v budoucnu prostředky adresátovi na stejnou adresu jako minule, protože ji už zná. Naopak nad jednorázovou adresou vratky má plnou kontrolu peněženka, která se spolehlivě postará o to, aby na tuto adresu už nikdy znovu prostředky neposlala.

Následujícím způsobem formulovali heuristiku využívající jednorázové povahy change adres Neudecker a Hartenstein [14]:

Heuristika 3. *Mějme transakci $t = T(i,j)$. Výstupní adresa $a_{ch} \in O(t)$ je change adresou transakce t, pokud jsou splněny následující čtyři podmínky:*

(1) Transakce t představuje první výskyt adresy a_{ch} na blockchainu. Tedy platí $\forall i',j' \in \mathbb{N} : a_{ch} \in O(T(i',j')) \implies i' \geq i$.

(2) Transakce t není coinbase transakce.

(3) Mezi výstupy transakce t není žádná adresa, která se objevuje na jejím vstupu, tzv. self-change adresa. Tedy $\forall a \in O(t) : a \notin I(t)$.

(4) Všechny ostatní výstupní adresy $a_{ch} \neq a_o \in O(t)$ nesplňují podmínku (1), tedy byly již použity v některé transakci v dřívějším bloku.

Tyto podmínky však zřejmě nejsou postačující pro správnou klasifikaci jednorázových adres vratek. Například k nesprávně negativní klasifikaci dojde v případě, kdy jsou pro všechny výstupy nějaké transakce vytvořeny nové adresy, přičemž jedna z nich je change. Pokud naopak odesilatel pošle veškerý vstup více novým majitelům bez vytvoření vratky a přitom právě jeden z adresátů vytvoří

novou adresu pro danou transakci, potom je jeho adresa nesprávně označena za adresu vratky původního majitele. Někteří autoři proto omezují tuto heuristiku jen na transakce s jistým počtem vstupů či výstupů, u kterých je pravděpodobnější, že byly vytvořeny peněženkou běžného uživatele. Zhang, Wang a Luo [3] použili upravené znění této heuristiky, když obrátili časové vymezení podmínky (4) tak, že se ostatní výstupní adresy musí vyskytovat znovu na výstupu některé z pozdějších transakcí. Nejefektivnější by však mohlo být sjednocení obou podmínek, tedy že se ostatní adresy musí vyskytovat obecně na výstupu alespoň dvou různých transakcí.

2.2 Off-chain informace

Samotný blockchain je anonymní. Pomocí klastrování lze sice odvodit společného vlastníka množiny adres, stále však zůstává skryta jeho identita. Takřka jediný způsob, jak spojit adresy s jejich reálným vlastníkem, je použít takzvané off-chain informace. To jsou obecně informace, které se blockchainu týkají, avšak nejsou v něm přímo uložené. Mezi jejich zdroje patří různá internetová fóra, webové stránky společností zabývajících se kryptoměnami či přímo seznamy adres a jejich vlastníků dříve vytvořené jinými výzkumníky (viz [7], [17], [18]) . Tyto informace mohou obsahovat identitu konkrétní entity stojící za nějakou adresou nebo jen obecně prozrazovat její typ. Například oběti krádeží bitcoinů mohou označit adresy, na které odešly prostředky z jejich vykradených peněženek, nebo mohou organizace zveřejnit své darovací adresy, na kterých je lze podpořit.

Entity spravující větší množství bitcoinů se i podle Gaihra, Lua a Liua [7] snaží zbytečně nezveřejňovat své adresy, které navíc obvykle používají jen jednorázově. Přesto musí některé adresy prozradit protistraně, se kterou obchodují, pro umožnění vkladů prostředků do svého vlastnictví. A naopak při obdržení bitcoinů může adresát zjistit jejich adresu implicitně z obdržené transakce. Interakce s ostatními uživateli tak představuje spolehlivý způsob pro spojení části adres s jejich konkrétními majiteli. Není však příliš efektivní, neboť i větších entit než jsou běžní uživatelé mohou existovat stále tisíce.

Méně používaným zdrojem off-chain informací jsou IP adresy v bitcoinové síti. Jak jsme již popisovali v podkapitole 1.4, tato síť je decentralizovaná a kdokoliv v ní může vytvořit nový uzel. V něm pak může zaznamenávat IP adresy sousedních uzlů, ze kterých se k němu nově vzniklé transakce dostávají poprvé, a snažit se s jejich pomocí lépe rozpoznávat různé entity stojící za těmito transakcemi. Tuto metodu vyzkoušeli například Neudecker a Hartenstein [14]. Její výsledky však nejsou příliš dobré, neboť bitcoinová síť má netriviální topologii a první IP adresa transakce z pohledu jednoho uzlu poskytuje velmi málo informací o samotném odesilateli. Ten může při vytváření transakcí používat více uzlů a z nich navíc mohou transakce putovat různými cestami.

Přestože různých adres existují na blockchainu stovky milionů a stojí za nimi pravděpodobně řádově miliony samostatných entit, dřívější práce byly schopny s využitím dat dohledatelných na internetu odhalit adresy řádově jen stovek z nich. V tomto smyslu nejúspěšnější práce Makarova a Schoarové [19] z roku 2021 vychází z informací o 1043 různých entitách, jejichž adresy však získali čás-

tečně i z databáze soukromé společnosti Bitfury Crystal Blockchain[3], která se na analýzu bitcoinových transakcí specializuje.

Vzhledem k používání mnoha adres a neustálému vytváření nových jednotlivými entitami jsou off-chain informace z povahy věci vždy jen kusé a musí být doplněny vhodnou klastrovací metodou pro odhalení větší části adres spojených s konkrétní entitou. Off-chain informace se dají použít dvěma způsoby. Jednak lze v transakcích obsahujících známé adresy hledat společné vzory, ať už automaticky trénováním zvoleného modelu, nebo ručně například s pomocí některého z blockchainových prohlížečů jako je Blockchair[4]. Druhou možností je využít tyto informace při klastrování jako kontrolní množinu, jejíž adresy by neměly klastrovací heuristiky přiřadit stejné entitě. Tento přístup zvolili a popsali Ermilov, Panov a Yanovich [4].

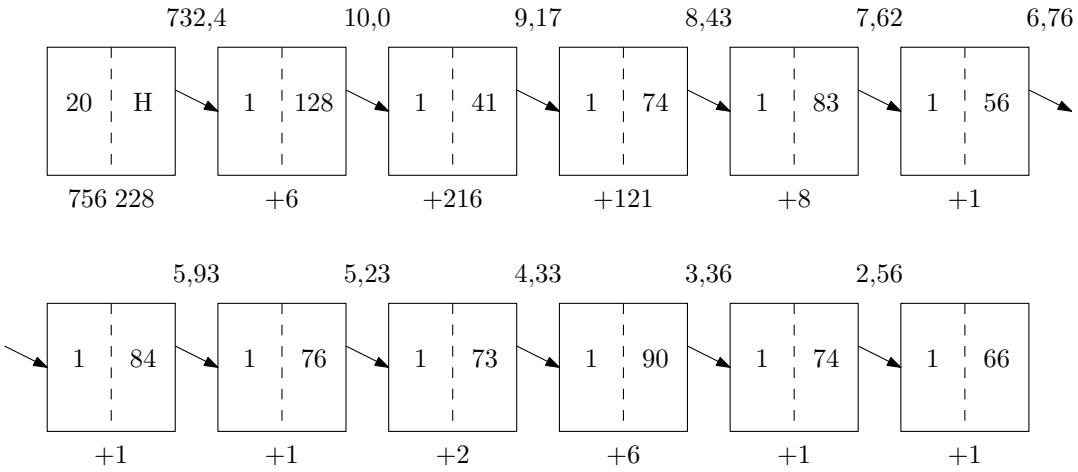
2.3 Vlastnosti chování uživatelů

Bitcoin už funguje řadu let a během této doby se na něm vyprofilovalo několik druhů entit. Ty se od sebe neliší jen v reálném světě, jejich odlišné chování lze vypozorovat i na blockchainu. Abychom však byly schopni sledovat transakce konkrétních entit, potřebujeme znát alespoň nějaké příklady jejich adres. Popis chování uživatelů tedy většinou vychází z poznatků získaných pomocí dříve popsaných heuristik v kombinaci s off-chain informacemi. Podobnosti v chování různých typů entit lze najít z množství, tvaru či frekvence transakcí, které vytváří. Této podobnosti se pak využívá při klasifikaci dosud neznámých klastrů, viz například článek Makarova a Schoarové [19].

Nejsnáze pozorovatelné je chování těžařů. To lze zjistit i ze samotných on-chain dat, neboť některé mining pooly zanechávají své jméno v poznámce u vytěženého bloku. Například blok s číslem 775 028 byl vytěžen společenstvím s názvem F2Pool a odměna byla zaslána na adresu 1KFHE7w8BhaENAswwryaoccDb6qcT6DbYY. Ta je zřejmě používána tímto mining poolem pravidelně, neboť se vyskytuje ve velkém množství coinbase transakcí. Každý den jsou z ní prostředky odesílány na adresu 1GX28yLjVWux7ws4UQ9FB4MnLH4UKTPK2z, ze které jsou dále jedinou transakcí posílány na asi tři tisíce různých adres. Můžeme tedy usoudit, že tímto způsobem rozděluje F2Pool bitcoiny mezi své těžaře.

Jiný způsob vyplácení odměn těžařům zvolilo společenství AntPool. Vytěžené bitcoiny shromažďuje na více adresách, z nichž pak jsou jednotlivým těžařům vypláceny postupně pomocí takzvaného *peeling chainu*. To je obvyklý anonymitu zvyšující způsob, jakým bohaté entity rozdělují jedno velké UTXO na mnoho malých částí. V rámci první transakce v takovém řetězci odesílají prostředky prvním desítkám či stovkám adresátů a zbývající velkou část vstupní hodnoty vrací na nově vytvořenou jednorázovou change adresu. Z ní se obdobný proces mnohokrát opakuje a skončí až ve chvíli, kdy se veškerá hodnota rozešle mezi jednotlivé adresáty, viz Obrázek 2.1. Vyplácení odměn společenstvím Antpool podrobně popsali Makarov a Schoarová [19]. Uvádí rovněž příklad výplatního řetězce, kde se 100 BTC rozeslalo postupně v celkem 152 navazujících transakcích, které se navíc podařilo těžařům umístit do jediného bloku s číslem 590 738. Pro úplnost

[3] https://crystalblockchain.com
[4] https://blockchair.com/bitcoin



Obrázek 2.1: Peeling chain – posloupnost transakcí, ve kterých směnárna vyplácí bitcoiny svým zákazníkům. Čísla oddělená čárkovaně značí počet vstupů, respektive výstupů daných transakcí. Písmeno H značí hot adresu směnárny Coinbase. Šipky pomyslně spojují místo vzniku a utracení konkrétních UTXO. Nad nimi je uvedena jejich hodnota v BTC. Pod transakcemi se nachází čísla bloků, ve kterých jsou zapsány, případně jejich relativní vzdálenost od bloku předchozího.

dodejme, že některé mining pooly jsou spravovány přímo směnárnami, výplata těžařům tedy nemusí chodit na jejich bitcoinové adresy, ale může jim být připsána přímo k účtu na směnárně.

Rovněž směnárny často používají k vyplácení bitcoinů svým zákazníkům peeling chainy. Jejich struktura však není tak přímočará jako u pravidelných výplat těžařů. Obecným znakem pro většinu směnáren a burz je vlastnictví hot a cold walletky. Druhá jmenovaná uchovává v bezpečí mimo síť většinu uživatelských vkladů. Jejich souhrnná výše může dosahovat tisíců bitcoinů. Například na nejbohatší adrese, kterou je již řadu let cold walletka největší kryptoměnové směnárny Binance, bylo na začátku roku 2023 uloženo 250 000 BTC.

Hot walletky směnáren spravují mimo spousty jednorázových adres i několik význačných opakovaně používaných, tzv. *hot adres*. Z nich všechny výplatní řetězce transakcí startují podobně jako ty z těžařské adresy akumulující vytěžené bitcoiny. Hot adresy dlouhodobě neuchovávají žádnou hodnotu, v případě potřeby se na ně posílají prostředky buď z adres shromažďujících vklady uživatelů, nebo přímo z adres napojených na cold walletku. Aby společnosti dokázali jednoznačně identifikovat odesilatele vkladů na jejich adresy, vytváří za tímto účelem obvykle unikátní příjmovou adresu pro každého svého uživatele. Přicházející UTXO následně spojují napříč množstvím takovýchto adres do jednoho hodnotného UTXO, které v případě potřeby odesílají na hot adresu.

Znakem typickým pro obyčejné uživatele je vytváření transakcí s nízkým počtem UTXO. Obvykle mívají jeden nebo dva výstupy podle toho, jestli potřebují nějaké prostředky vracet či nikoliv. Oproti ostatním neprovádí transakce příliš často. Často je naopak provádí podvodníci snažící se zakrýt původ bitcoinů, ke kterým přišli nelegální aktivitou. Využívají při tom opět peeling chainy, které rozdělují kradené prostředky na obrovské množství malých TXO. Ty navíc dále přeposílají a míchají za pomoci mixovacích služeb s prostředky ostatních uživatelů ve snaze setřást případné stopaře.

3. Související práce

Článků zabývajících se vlastnostmi Bitcoinu bylo od jeho spuštění publikováno mnoho. Jejich obsáhlou rešerši provedli v roce 2021 Wu a kol. [20], kteří nalezli a pojmenovali několik základních směrů výzkumu a podle nich pak přehledně roztřídili desítky do té doby vzniklých článků do několika skupin. My popíšeme pouze dvě z nich, které souvisí se zaměřením naší práce nejvíce. První skupinou jsou práce zabývající se různými grafovými reprezentacemi adres a transakcí a následnou analýzou sestrojených grafů. Do druhé skupiny patří články zabývající se samotným klastrováním adres patřících stejné entitě, které je často rozšiřováno o navazující problém klasifikace typů těchto entit.

3.1 Charakteristiky grafu

Vývoj grafových aspektů blockchainové sítě od jejího vzniku až do roku 2020 popsali například Nerurkar a kol. [17]. Evoluce sítě bývá rozdělována do tří částí, jejichž hranice se v literatuře různí. Tou první je ranná fáze trvající přibližně do roku 2011, kdy byla funkčnost Bitcoinu uživateli teprve testována a síť se tak vyznačovala nízkou aktivitou. Následovalo období rozmachu, kdy počet nových adres rostl exponenciálně a vznikaly první směnárny. Bitcoin se na nich obchodoval za jednotky či desítky dolarů. V té době však neexistovala žádná pravidla, uživatelé byli nezkušení. Docházelo tak často k podvodům a ke kradení tajných klíčů z nezabezpečených peněženek. Bitcoiny se platilo při hraní hazardních her. Odtud získalo celé období označení jako období černého trhu. Za jeho konec bývá považováno vykradení a následný krach burzy Mt.Gox, v roce 2014 největší burzy využívané většinou uživatelů. Poslední fáze, která trvá dodnes, je označována za období směnáren či obchodování, jelikož velká část transakcí nyní souvisí s přesuny bitcoinů mezi adresami uživatelů a směnáren. Ty tak představují centra bitcoinové sítě. Žádná z nich však již nezískala takový monopol jako Mt.Gox, uživatelé si mohou vybírat mezi stovkami zavedených směnáren a burz.

Tao a kol. [21] provedli podrobnou analýzu orientovaného grafu $G = (V,E)$ vytvořeného z množiny transakcí $\mathcal{T}$. Zvolili přitom takovou reprezentaci, kde vrcholy V odpovídají množině adres $\mathcal{A}$ a kde hrany spojují vstupní a výstupní adresy daných transakcí. Tedy $(a_1, a_2) \in E$ právě tehdy, když $\exists t \in \mathcal{T}$ tak, že $a_1 \in I(t), a_2 \in O(t)$. Transakce s m vstupy a n výstupy tak může být reprezentována pomocí až $m \times n$ hran. Jejich výsledky shrneme v následujících bodech:

(a) *Stupeň vrcholu* odpovídá počtu různých adres, se kterými je daná adresa přímo spojena nějakou transakcí. Označme $P(k)$ pravděpodobnostní rozložení těchto stupňů vrcholů v grafu reprezentující pravděpodobnost, že náhodně zvolený vrchol má stupeň k. Podobně jako u jiných reálných sítí i u Bitcoinu má toto rozložení power law distribuci $P(k) \sim k^{-\alpha}$, kde α je konstanta. Její hodnota je pro Bitcoin odhadnuta na 1,411. Síť je tedy bezškálová, obsahuje malé množství centrálních vrcholů s mnoha sousedy a naopak velké množství vrcholů se stupněm 1. Pro úplnost dodejme, že obdobné výsledky byly popsány i pro vstupní a výstupní stupně v orientovaném grafu.

(b) *Délka nejkratší cesty* mezi dvěma vrcholy dosahuje průměrné hodnoty 3,822.

(c) *Klastrovací koeficient* popisuje tendence uzlů v síti se shromažďovat. Je definován jako

$$C_G = \frac{1}{|V(G)|} \sum_{i \in V(G)} \frac{2\Delta_i}{k_i(k_i - 1)},$$

kde k_i značí stupeň vrcholu i a Δ_i je počet trojúhelníků v jeho sousedství. Autoři pro bitcoinovou síť nalezli hodnotu klastrovacího koeficientu $C = 0{,}0071$.

(d) *Komponenty souvislosti* dále doplňují analýzu shluků uzlů. Nové jednoprvkové komponenty vznikají v každé coinbase transakci, pokud pro ni těžař vytvoří novou adresu. Největší slabě souvislá komponenta zahrnuje kolem 93 % všech vrcholů a silně souvislá 45 %, graf je tedy poměrně dobře propojený. Počet slabě souvislých komponent je řádově menší než počet silně souvislých, což svědčí o převažující jednosměrné povaze transakcí.

(e) *Centralita* popisuje důležitost jednotlivých uzlů v síti. Lze ji měřit několika odlišnými způsoby. Výsledky všech měření vypovídají o tom, že v bitcoinové síti existuje nemnoho centrálních uzlů, přičemž některé z nich jsou přímo spojeny s velikým počtem dalších uzlů.

(f) *Koeficient asortativity* vypovídá o četnosti propojení vrcholů s podobnými vlastnostmi v našem případě reprezentovanými jejich stupni. Záporná hodnota tohoto koeficientu vypovídá o tom, že vrcholy s nízkými stupni jsou častěji spojené s vrcholy s vysokými stupni a obráceně.

(g) *Rich-club koeficient* dále slouží ke stanovení míry propojení centrálních uzlů. Pro různé prahy stupňů vrcholů je definován jako

$$\phi(k) = \frac{2|E_{>k}|}{|V_{>k}|(|V_{>k}| - 1)},$$

kde $G_{>k} = (V_{>k}, E_{>k})$ je podgraf indukovaný vrcholy stupně alespoň k. Pro lepší vypovídající hodnotu bývá normalizován porovnáním se stejným koeficientem náhodného grafu. Normalizované hodnoty však téměř pro žádné k nepřevyšují hodnotu 1, což svědčí o tom, že vrcholy s velkými stupni nejsou příliš navzájem propojené.

Analýzou *motivů* v síti se zabýval Ranshous a kol. [22]. Motivy jsou vzory podgrafů, které se v síti objevují statisticky významně častěji než v náhodném grafu. Autoři se snažili nalézt motivy charakterizující řetězce transakcí začínající a končící na adresách směnáren. Využili při tom data o entitách stojících za částí adres publikovaná na stránce WalletExplorer[1]. Výsledky jejich práce poukázali na vysokou interní aktivitu směnáren, kdy si v rámci transakcí přeposílají prostředky mezi vlastními adresami.

[1] https://www.walletexplorer.com/

3.2 Charakteristiky entit

Další skupinou jsou práce zabývající se analýzou blockchainu z hlediska spojitosti transakcí s reálnými uživateli. Motivací odpovídajícího segmentu prací je zejména snaha o odhalení a omezení ilegálních aktivit využívajících Bitcoin. Této oblasti se kromě výzkumníků věnují i kryptoanalytické společnosti poskytující AML služby zejména pro směnárny. S problémem rozpoznání kradených bitcoinů přímo souvisí rozpoznávání všech adres patřících stejné entitě a hledání vzorů v posloupnostech navazujících transakcí napovídajících o umělých přesunech prostředků mezi adresami stejného majitele. Z podstaty věci je tedy součástí většiny prací nějaké klastrování, které bývá různými způsoby vylepšováno větším množstvím off-chain informací, případně dále doplňováno o úlohu klasifikace typů entit.

Společným nedostatkem všech prací věnujících se rozpoznávání entit je nedostupnost ground truth dat pro trénování modelů a testování hypotéz. Za ta bývají často vydávána data vytvořená kryptoanalytickými společnostmi či nadšenci. První a v počátcích Bitcoinu nejpoužívanější byla data publikovaná na již zmíněné stránce WalletExplorer. Ta představuje dodnes jeden z nejbohatších zdrojů volně dostupných informací o klastrech adres, přestože deklarovaných rozpoznávaných entit jsou zde jen stovky a navíc od roku 2015 není seznam entit aktualizován. Některým pozdějším autorům se podařilo získat větší množství dat pro vlastní výzkum od soukromých společností. Mikkel a kol. [9] používají data od Chainalysis[2], Makarov a Schoarová [19] od Crystal Blockchain[3], zatímco Weber a kol. [23] od společnosti Elliptic[4]. Vzhledem k tomu, že proces tvorby těchto dat představuje konkurenční výhodu, nejsou jejich metody klastrování ani zdroje off-chain informací známé.

Vzhledem k absenci centrální autority, která by měla přehled o identitě všech uživatelů, nelze s jistotou stanovit precision ani recall zmíněných klasifikací. Lze předpokládat, že všechny používají klastrování pomocí variant nějakých heuristik. Výsledky modelů trénovaných na takových datech pak mohou být zkreslené, protože modely se mohou naučit hledat vzory použité pro klastrování spíše než vzory skutečně charakterizující jednotlivé entity. Například Gomez, Sanchez a Caballero [18] zkoušeli natrénovat různé modely zaměřené na rozpoznání adres směnáren s využitím označkovaných adres právě od Walletexploreru a na testovací části dat dosáhli s pomocí náhodného lesa až překvapivě vysokého F_1 skóre predikce dosahující 95 %. Jourdan a kol. [24] na obdobném datasetu provedli klasifikaci do více tříd pomocí rozhodovacího stromu a rovněž zaznamenali hodnotu F_1 skóre převyšující 90 %. Vzhledem k tomu, že proces vytvoření dat na stránce Walletexplorer není známý a probíhal jen v prvních letech fungování Bitcoinu, kdy se ke klastrování používaly jen základní metody jako multi-input heuristika a heuristika pro rozpoznání jednorázových change adres, domníváme se, že jsou správně klasifikovány jen malé části v té době existujících entit a některé klasifikace mohou být přitom navíc zavádějící.

O automatické vytvoření ověřitelných dat se pokusili Möser a Narayanan [12]. Zaměřili se však specificky jen na rozpoznávání change adres v transakcích se dvěma výstupy. Použili přitom klastrování založené na klasické multi-input heu-

[2] https://www.chainalysis.com/
[3] https://crystalblockchain.com/
[4] https://www.elliptic.co/

ristice, které samo o sobě nalezne v části transakcí výstupní adresy patřící do stejného klastru jako ty vstupní. Takové adresy při splnění několika dalších podmínek prohlásili za ověřené change adresy v odpovídajících transakcích a na takto vytvořené množině 35 milionů transakcí pak trénovali a testovali ostatní heuristiky pro hledání adresy vratky.

Gaihre, Luo a Liu [7] se ve své práci snažili odpovědět na otázku, jestli uživatelům vůbec záleží na jejich anonymitě. Jako metriky přitom použili četnost znovupoužití adres, kdy se vzrůstajícím počtem TXO obsahujících stejnou adresu kromě automatického odvození stejného majitele zároveň roste pravděpodobnost utracení společně s dalšími adresami a jejich odhalení pomocí multi-input heuristiky. Další použitou metrikou byla četnost adres s nulovým zůstatkem. Ukázali, že počty nových adres jsou dokonce menší než znovupoužití starých, tedy že uživatelé zásady pro zachování anonymity příliš neřeší. Dále označili čtvrtinu adres s nejvyššími zůstatky za bohaté a ukázali, že ty dbají o svou anonymitu více.

Makarov a Schoarová [19] v práci z roku 2021 představili spoustu zajímavých zjištění. Vycházeli přitom z off-chain dat od společnosti Crystal Blockchain, která dále rozšířili a pokusili se podle nalezených vzorů odhadnout typy entit i u neoznačených klastrů adres. Představili vlastní definici takzvaného *falešného objemu*. Tak označili množství bitcoinů přeposílaných mezi vlastními adresami stejné entity. Nadále zkoumali jen zbývající ekonomicky významné transakce. Došli k závěru, že kolem 85 % objemu přenášeného v transakcích je falešného a ze zbývající části je nadpoloviční většina spojena s transakcemi, ve kterých figurují směnárny. Poukázali na toky prostředků mezi různými směnárnami navzájem. Dále analyzovali časový vývoj výkonu jednotlivých mining poolů a způsoby vyplácení odměn jejich členům. Zároveň se snažili změřit, jaká část bitcoinů je držena jednotlivci a jaká je spojena s adresami zprostředkovatelů. Ukázali rovněž jistou spojitost mezi on-chain objemy transakcí a off-chain objemy zobchodovanými na největších burzách.

Bovet a kol. [25] hledali vztah mezi vývojem grafových aspektů sítě a cenou bitcoinu. Použili přitom grafy vytvořené z transakcí na denní a týdenní bázi. Nalezli v nich vzory stádového chování uživatelů, kdy při růstu ceny bitcoinu převažují jednoduché transakce pravděpodobně mířící na adresy směnáren. Po dosažení cenového vrcholu opět začínají být transakce heterogenní. Více kauzálních vztahů nalezli v datech do roku 2014, od té doby se podle nich zvýšila efektivita trhu, což ztížilo možnosti predikce vývoje ceny bitcoinu.

4. Analýza

V následující části postupně popíšeme vlastní analýzu blockchainu, která bude vycházet z poznatků předchozích kapitol. Nejprve popíšeme a zdůvodníme možnosti použití klasických i upravených klastrovacích heuristik a jejich výsledky porovnáme na aktuálních datech. Poté z obdržených výsledků odvodíme závěry týkající se největších burz a směnáren. Budeme přitom vycházet takřka výhradně z dat obsažených v blockchainu, neboť off-chain data nejsou snadno dostupná a navíc může být problematické jim důvěřovat.

Vstupní data tvoří blockchain stažený pomocí volně dostupného bitcoinového klienta Bitcoin Core[1]. Jeho velikost dne 5.1.2023 dosahovala 484 GB. Blockchain v syrové podobě je přitom optimalizovaný pro co nejmenší spotřebu paměti a zároveň co nejrychlejší možnost kontroly hashů jednotlivých bloků. Tato forma je ideální pro síťové uzly, ne však pro efektivní analýzu transakcí. K jeho předzpracování a k následné tvorbě vlastních klastrování jsme tedy použili nástroj BlockSci[2] (Kalodner a kol. [26]), do kterého jsme sami implementovali nové typy heuristik. Vzhledem k potenciální možnosti změny nejmladších bloků jsme zaokrouhlili počet bloků použitých pro samotnou analýzu na hodnotě 770 000. To odpovídá takřka přesně veškerým datům za prvních 14 let existence Bitcoinu, neboť nultý blok byl vytěžen 3.1.2009 a poslední námi použitý byl k blockchainu připojen 2.1.2023. Za tu dobu se do blockchainu zapsalo celkem 792 734 480 transakcí, v nichž figuruje 1 123 291 574 různých adres.

Mezi transakcemi jsou nejčastěji zastoupené ty se dvěma výstupy. Tuto vlastnost mají takřka tři čtvrtiny transakcí. Obdobně často se v transakcích vyskytuje pouze jediný vstup, viz Obrázek A.2. Transakcím kombinujícím obě tyto vlastnosti budeme říkat *standardní*. Jejich dominance vypovídá o převaze klasických uživatelských transakcí, při kterých je jedno vstupní UTXO rozděleno na část odeslanou adresátovi a na vratku. Druhým nejčastějším počtem vstupů jsou dva a u výstupů to je jeden. Vyskytují se asi u 15 % transakcí. Jiné počty vstupů či výstupů se v transakcích vyskytují poměrně zřídka. Například pouze 10 % transakcí má alespoň tři výstupy a jen necelá 2 % jich má více než deset. Co se týče počtu výstupů, největší dosud vytvořená transakce jich obsahovala 13 107.

Další zajímavou vlastností transakcí odvoditelnou z blockchainu je stáří utrácených UTXO. To lze jednoduše měřit podle počtu bloků nacházejících se mezi vznikem a utracením stejného UTXO. Zatímco rozsah počtu transakčních vstupů a výstupů byl shora omezen maximální velikostí bloku, stáří utrácených výstupů může teoreticky růst lineárně v závislosti na počtu bloků v blockchainu. V našem případě je tedy shora omezeno hodnotou 770 000. Přesto je většina použitých UTXO utrácena poměrně brzy. Medián jejich stáří dosahuje hodnoty pouze 46 bloků. Polovina všech použitých UTXO je tedy utracena do 8 hodin od jejich vzniku. Ještě rychleji jsou utráceny UTXO v případě, kdy se na vstupu vyskytují samostatně, viz Obrázek A.3. Většina z nich je použita jen několik málo bloků po svém zapsání do blockchainu. Toto chování je překvapivé, neboť jak jsme popisovali v podkapitole 1.5, s klesající hloubkou transakce v blockchainu roste riziko jejího zneplatnění při vytěžení delšího blockchainu potenciálním útočníkem.

[1] https://bitcoin.org/en/bitcoin-core/
[2] https://github.com/citp/BlockSci

Zaměřili jsme se tedy na TXO s velmi nízkým stářím detailněji. Podle našeho nejlepšího vědomí jejich častý výskyt ještě nikdo nepopsal. Pouze Liu, Zhang a Zhao [27] ve svém článku na Obrázku 3 zachycují vývoj stáří utrácených UTXO v čase a potvrzují skutečnost, že kolem 80 % použitých UTXO je utraceno do jednoho dne od svého vzniku. Místo toho, aby velmi mladé UTXO nebyly utráceny prakticky vůbec, jsou ve skutečnosti utráceny častěji než ty starší, viz Obrázek A.4. Dokonce nejčastější stáří je nula bloků, což odpovídá situaci, kdy jsou obě transakce vytvářející a utrácející dané UTXO vytěženy ve stejném bloku. Vzhledem k tomu, že je toto chování mnohonásobně častější u transakcí s jedním vstupem, myslíme si, že je způsobeno vytvářením posloupností transakcí v peeling chainech. Vytvářející entita totiž zná mezivýstupy peeling chainu okamžitě, neboť transakce sama vytváří, a zároveň nemusí čekat několik bloků pro jejich potvrzení, neboť sama sobě důvěřuje. Posloupnost desítek či stovek navazujících transakcí se tedy může objevit v mempoolech těžařů ve stejnou chvíli a ti je do těženého bloku umístí zároveň nebo po několika částech, pokud se všechny nevlezou.

Abychom měli alespoň nějaké důvěryhodné příklady reálných transakcí, ve kterých figurují adresy směnáren, založili jsme vlastní účet na několika z nich a využili jsme možnosti přímé interakce s těmito entitami popisované v podkapitole 2.2. Ze třech směnáren Coinbase, Coinbase Pro a Coinmate jsme si nechali poslat na naši adresu zakoupené bitcoiny. Ve všech případech jsme zpětnou analýzou transakcí, ze kterých nám byly poslány prostředky, nalezli posloupnosti transakcí podobající se peeling chainu, jehož první transakce vedla z nějaké hot adresy pravidelně přeposílající větší objemy bitcoinů. Posloupnost takových transakcí odhalených při výběru ze směnárny Coinbase jsme zachytili na Obrázku 2.1, komplikovanější posloupnosti ze směnárny Coinbase Pro potom na Obrázku A.6. Transakce vytvářené směnárnou Coinmate byly ještě složitější, neboť příchozí transakce i množství jí předcházejících obsahovalo více vstupů. Jen v jednom případě jsme se dostali zpětnou analýzou přímočaře na nějakou hot adresu. Oproti ostatním dvěma odhaleným hot adresám přes ni protekly hodnotnější transakce čítající obvykle stovky bitcoinů, zato však byla používána málo často. Ostatní hot adresy se vyskytují řádově v tisících transakcích uskutečňovaných pravidelně po dobu několika let.

Příklady odhalených transakcí směnáren jsou v souladu s poznatky uvedenými v podkapitole 3.1, podle kterých graf vytvořený z adres a transakcí obsahuje množství nepřímo propojených center. Námi zaznamenané transakce rovněž potvrzují představu o existenci hot adres a z nich vycházejících výplatních řetězců. Ty však zjevně nemusí mít přesně podobu peeling chainů, mohou obsahovat i jiné nestandardní transakce než ty s mnoha výstupy. Zároveň v nich často dochází k utrácení jen několik bloků starých UTXO.

4.1 Klastrování

Naším cílem je odhalit a zklastrovat co nejvíce adres patřících stejné směnárně či burze a zároveň přitom klást důraz na protichůdnou podmínku, aby se do těchto klastrů dostalo co nejméně cizích adres. Vzhledem k tomu, že snadno dostupných off-chain informací je málo a ty navíc nemusí být spolehlivé, budeme při klastrování vycházet z obecných poznatků popsaných v předchozích kapitolách

a z poznatků odvozených z příkladů směnárenských transakcí. Kvůli nedostatku testovacích dat se budeme snažit při klastrování použít jen konzervativní heuristiky, které s vysokou pravděpodobností neumístí adresy patřící dvěma různým entitám do stejného klastru. Algoritmus klastrování bude postupně procházet všechny transakce a při splnění jistých podmínek sjednocovat klastry, do kterých dvě adresy vyskytující se ve stejné transakci patří, viz Algoritmus 1. Podmínky, které musí obě adresy, případně celé TXO či transakce splňovat, budou stanovovat patřičné heuristiky. Formálně můžeme na heuristiku nahlížet jako na funkci $H : \mathcal{T} \times \mathcal{A} \times \mathcal{A} \to \{0, 1\}$, kde $H(t, a_1, a_2) = 1$ indikuje, že adresy a_1 a a_2 patří do stejného klastru podle heuristiky H kvůli transakci t. Pokud budeme používat více heuristik současně, budeme vyžadovat splnění podmínek alespoň jedné z nich.

Základní konzervativní heuristikou je multi-input heuristika. Její potenciální rizika například při použití mixovacích služeb jsme popsali v podkapitole 2.1.1. Na druhou stranu většina směnáren musí dodržovat restriktivní opatření kontrolující původ bitcoinů a nemá tedy důvod podporovat jejich anonymizaci používáním těchto služeb. Použití multi-input heuristiky je prakticky nutné při testování dalších heuristik, neboť jinak není jasné, se kterou vstupní adresou spojit výstupní adresu označenou například jako change adresu původního majitele.

Při implementaci dalších heuristik jsme se inspirovali v práci Makarova a Schoarové [19], kteří při hledání falešného objemu v transakcích použili heuristiku pro hledání jednorázových adres zohledňující stáří utrácených UTXO. Jako jednorázové adresy označili ty, které zůstaly v samostatném klastru po použití multi-input heuristiky a na které právě jednou přišel obnos bitcoinů, který byl později utracen. Jednorázovou adresu přitom spojili s klastrem vstupních adres entity, která danou adresu v transakci vytvořila, pokud bylo patřičné UTXO s jednorázovou adresou utraceno do jednoho týdne nebo do stáří 1068 bloků. Tuto podmínku však splňuje 93 % transakcí s jedním vstupem, je tedy podle našeho názoru příliš benevolentní a pravděpodobně tak spojili například část jednorázových adres patřících uživatelům s klastry směnáren, ze kterých své prostředky obdrželi.

První námi nově představená heuristika se zaměřuje na okamžitě utracené výstupy, jejichž nezvykle častý výskyt jsme popsali v úvodu kapitoly. Tato heuristika tedy klastruje výstupní adresu transakce s entitou jejího odesilatele v případě, že odpovídající UTXO s danou výstupní adresou bylo utraceno ve stejném bloku, tedy při stáří nula. Budeme ji označovat jako *age zero* heuristiku. Důvody, proč za výstupem utráceným při stáří nula stojí pravděpodobně stejná entita, která výstup vytvořila, jsme již částečně popsali. Cizí entita by totiž o příchozí transakci nemusela vůbec vědět, případně by ji nemusela důvěřovat, neboť tato transakce se v době vytvoření té navazující nacházela pravděpodobně teprve v mempoolu těžařů. Druhou méně pravděpodobnou možností je, že první transakce již byla v té době zapsaná v nějaké verzi blockchainu, ale v tom případě se musela nacházet ve velmi nízké hloubce, neboť tato verze byla později nahrazena jinou, delší. V tom případě by cizí entita příchozí transakci ještě nemohla považovat za nezměnitelnou a nepřeposílala by dále obnosy, o které může teoreticky přijít.

Tuto argumentaci lze částečně použít i pro *age six* heuristiku beroucí v potaz výstupy mladší než 6 bloků. Právě tak dlouho by měl minimálně běžný uživatel čekat, než příchozí transakci prohlásí za trvale platnou, jak popisujeme v pod-

kapitole 1.5. Vzhledem k tomu, že se tímto pravidlem, případně tímto pravidlem právě s konstantou 6, všechny entity řídit nemusí, použijeme v tomto případě ještě doplňující podmínku zohledňující velikost transakce. Ta by měla heuristiku dále specializovat na peeling chainy směnáren. Při stanovení této podmínky jsme vycházeli z histogramů na Obrázku A.2 a z následující úvahy. Zatímco nestandardní transakci s více než jedním vstupem může běžně vytvořit každý uživatel při placení větších obnosů a tedy nutnosti spojení více UTXO, transakci se třemi a více výstupy peněženka běžného uživatele nevytvoří. Pro zvýšení konzervativity a zároveň zaměření na správné rozpoznání klastrů směnáren tedy přidáme do age six heuristiky podmínku, že transakce utrácející UTXO musí mít alespoň 3 výstupy.

U transakcí, jejichž výstupy byly utraceny za více než šest bloků nebo nebyly dosud utraceny vůbec, nese vždy označení některé výstupní adresy za adresu patřící entitě na vstupu vyšší míru rizika, že dojde k chybnému spojení dvou klastrů ve skutečnosti různých entit. Heuristiky popsané v podkapitole 2.1.2 jednoduše porovnávající typy adres nebo hodnoty UTXO jsou příliš náchylné k chybám, než abychom je mohli bezpečně použít. Jedinou výjimkou je one-time change heuristika. Při použití společně s dalšími podmínkami na podobu transakce může být dle našeho názoru stále dostatečně konzervativní. Její vhodné použití může být na druhou stranu velmi efektivní, neboť velké množství adres je jednorázových.

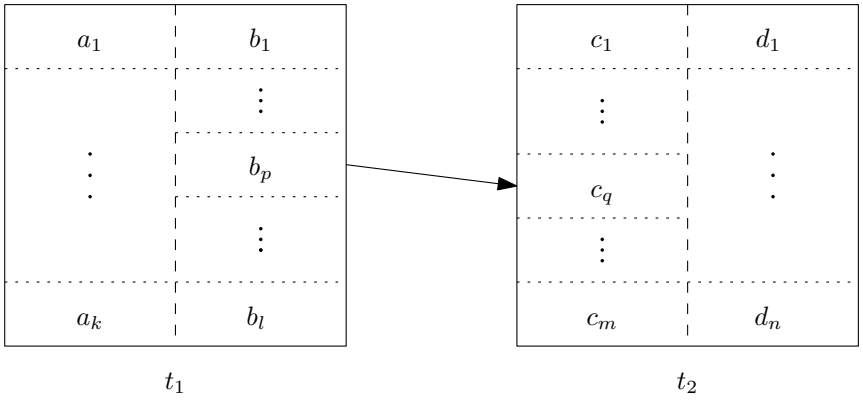
Heuristiku pro rozpoznání jednorázových adres vratky budeme používat společně s podmínkou na stáří, do kterého musí být nově vytvořená adresa utracena, a s omezením na počet vstupů této transakce utrácející prostředky z potenciální change adresy. Tento počet nastavíme na striktní hodnotu 1, abychom snížili riziko, že dojde k chybnému připojení většího klastru adres sjednocených pomocí multi-input heuristiky. Otestujeme dvě varianty této heuristiky, první s časovým omezením utracení do jednoho dne a druhou do jednoho týdne. Čas přitom budeme měřit v počtu vytěžených bloků, kdy jeden den odpovídá v průměru počtu 144 bloků. Budeme je nazývat jako *one-time one day* a *one-time one week* heuristiky. U druhé jmenované, jejíž podmínku na stáří splňuje většina transakcí s jedním vstupem, opět použijeme doplňující limit na počet výstupů transakce utrácející UTXO. Tyto podmínky by měly splňovat jednorázové adresy, které jsou součástí peeling chainů směnáren, ale zároveň by je neměly splňovat první cizí adresy, které se nachází na konci těchto chainů. Při utracení UTXO z jednorázové adresy v transakci s jediným vstupem pravděpodobně běžný uživatel nevytvoří narozdíl od směnárny více než 2 výstupy.

Zmíněné heuristiky nyní definujeme formálně. Mějme tedy transakci $t_1 = T(i_1,j_1)$ s k vstupy a l výstupy. Označme adresy v posloupnosti vstupů jako $(a_1, \ldots, a_k)$ a v posloupnosti výstupů $(b_1, \ldots, b_l)$. Označme $u = (b_p,h) \in \mathcal{U}$ p–tý výstup této transakce, kde $p \leq l$. Pokud již byl utracen, označme $t_2 = T(i_2,j_2)$ transakci utrácející toto UTXO. Analogicky označme m počet jejích vstupů, n počet výstupů a $(c_1, \ldots, c_m)$, $(d_1, \ldots, d_n)$ posloupnost jejích vstupních, respektive výstupních adres. Platí tedy $b_p = c_q$ pro nějaké $q \leq m$, viz Obrázek 4.1.

S takto zavedeným značením můžeme popsat podmínky jednotlivých heuristik pro zklastrování adres transakce t_1 následujícím způsobem:

Heuristika H1 (Multi-input heuristika)**.** *Adresy a_i, a_j patří do stejného klastru kvůli transakci t_1 pro všechna $i,j \leq k$.*

V definici zbývajících heuristik použijeme adresu a_1 jako prvek reprezentující



Obrázek 4.1: Obecné zobrazení dvou transakcí t_1 a t_2, ve kterých vzniká, respektive zaniká stejné UTXO. Levá část transakcí reprezentuje vstupy, pravá výstupy. Pro přehlednost jsou zde zachyceny jen posloupnosti adres.

klastr adres, do kterého patří všechny vstupní adresy. Vzhledem k výchozímu použití multi-input heuristiky volba tohoto reprezentanta neovlivňuje podobu výsledného klastrování.

Heuristika H2 (Age zero heuristika)**.** *Adresa b_p patří do stejného klastru jako a_1, pokud transakce t_2 existuje a platí $i_1 = i_2$.*

Heuristika H3 (Age six heuristika)**.** *Adresa b_p patří do stejného klastru jako a_1, pokud transakce t_2 existuje a navíc platí $i_1 \geq i_2 - 6$ a $n \geq 3$.*

Heuristika H4 (One-time one day heuristika)**.** *Adresa b_p patří do stejného klastru jako a_1, pokud transakce t_2 existuje a přitom $m = 1$, $i_1 \geq i_2 - 144$ a adresa b_p je jednorázová.*

Heuristika H5 (One-time one week heuristika)**.** *Adresa b_p patří do stejného klastru jako a_1, pokud transakce t_2 existuje a přitom $i_1 \geq i_2 - 1008$, adresa b_p je jednorázová, $m = 1$ a $n \geq 3$.*

Z definic všech heuristik je patrné, že využívají informací prozrazených při utracení jednotlivých UTXO. Zmíněné heuristiky lze seřadit podle jejich podmínek na stáří těchto UTXO. Na multi-input heuristiku lze pohlížet jako na speciální případ age zero heuristiky, kdy se dvě či více UTXO utratí zároveň nejen ve stejném bloku, ale dokonce ve stejné transakci. Podle zmíněného pořadí budeme postupně přidávat k multi-input heuristice další a ukážeme výsledky jejich kombinací, abychom získali přehled o efektu jednotlivých heuristik. Pro porovnání ještě použijeme zvlášť heuristiku H4.

Výsledky jsou zaznamenány v tabulce 4.1. Už po použití samotné multi-input heuristiky se počet klastrů snížil o více než polovinu oproti celkovému počtu adres. Naprostá většina klastrů přitom obsahuje jen několik málo prvků. Počet klastrů úměrně klesá v závislosti na jejich velikosti, viz Obrázek A.5. Největší klastry přitom obsahují přes milion adres, ten úplně největší jich obsahuje dokonce 35 milionů. Efektivitě multi-input heuristiky se detailně věnovali Harrigan a Fretter [28], kteří rovněž zaznamenali vznik jednoho nezvykle velkého klastru, jehož existenci zdůvodnili tím, že si burza Mt.Gox nechávala v prvních letech Bitcoinu zasílat privátní klíče od adres svých uživatelů.

Heuristika	# klastrů	# triviálních	největší	redukce
H1	527 M	437 M	35 M	53,13%
H1+H2	440 M	347 M	334 M	60,82%
H1+H2+H3	415 M	330 M	408 M	63,01%
H1+H2+H3+H4	277 M	181 M	500 M	75,31%
H1+H2+H3+H4+H5	276 M	180 M	502 M	75,38%
H1+H4	332 M	204 M	194 M	70,38%

Tabulka 4.1: Porovnání výsledků různých kombinací heuristik. Tabulka zachycuje počet výsledných klastrů, počet triviálních klastrů obsahujících jedinou adresu, velikost největšího klastru a relativní úbytek počtu klastrů oproti počátečnímu stavu, kdy každá adresa patří do triviálního klastru. Písmeno M značí miliony.

Počet triviálních klastrů v tomto případě ukazuje, že se 437 milionů adres nikdy nevyskytlo na vstupu společně s jinou adresou. Více než pětina z nich splňovala podmínky age zero heuristiky, byla tedy obsažena v transakčních výstupech utracených při stáří nula. Následující age six heuristika měla i kvůli použití spolu s podmínkou na velikost výstupu několikanásobně nižší efekt, který je však i přesto nezanedbatelný.

Přidání heuristik zaměřených na jednorázové adresy přineslo očekávaný efekt citelného snížení počtu jednoprvkových klastrů. O většinu tohoto efektu se postarala heuristika H4 klastrující UTXO utracené do jednoho dne od vzniku. Doplňující podmínky u heuristiky H5 již byly pravděpodobně příliš striktní, neboť po jejím použití spolu s dřívějšími heuristikami se výsledky změnily jen zanedbatelně.

Porovnání výsledků vůči ostatním autorům lze provést jen obtížně, neboť neexistuje žádná standardně používaná množina testovacích dat. Různí autoři definují svůj cíl různě a svůj postup popisují často jen neformálně. Navíc není zavedena žádná standardní množina metrik, kterou by každý zobrazoval u výsledků svých klastrování. Ty jsou přitom závislé na velikosti vstupních dat, neboť s rostoucí velikostí blockchainu roste počet transakcí využitelných při klastrování.

V tomto ohledu nejbližší naší práci je postup Zhanga, Wanga a Lua [3], kteří porovnali výsledky multi-input heuristiky a dvou dalších heuristik zaměřených na one-time change adresy na různě velkých verzích blockchainu. Efektivitu heuristik přitom vyhodnocovali pomocí relativního úbytku adres r definovaného jako

$$r = \frac{|\mathcal{A}| - |C|}{|\mathcal{A}|},$$

kde C odpovídá klastrování pomocí dané heuristiky. Efektivita multi-input heuristiky na jejich verzi blockchainu dosáhla 52,05 %, efektivita po použití s heuristikou 3 potom dosáhla 53,02 % a po následovném přidání vylepšené varianty této heuristiky se výsledky změnily jen o desetiny procent na hodnotu 53,23 %. Na naší verzi blockchainu sice už samotná multi-input heuristika dosáhla efektivity 53,12 %, přesto námi navržené heuristiky převyšují míru efektivity 60 % i 70 % a jsou tedy výrazně lepší, viz tabulka 4.1. Existence jediné entity spravující více než třetinu všech existujících adres je však nepravděpodobná. Naše výsledky jsou tedy zřejmě ovlivněny skutečností nazývanou jako kolaps klastrů, kdy kvůli byť jediné nesprávně sjednocené dvojici adres dochází ke spojení dvou velkých klastrů různých entit.

Pro porovnání jsme v tabulce 4.1 zaznamenali zvlášť výsledky multi-input heuristiky použité jen s heuristikou pro rozpoznání části jednorázových adres vratek (H1+H4). I v tomto případě však došlo ke vzniku klastru obsahujícího téměř 200 milionů adres. To nasvědčuje skutečnosti, že více entit do stejného klastru umístila již multi-input heuristika, neboť one-time change heuristika ke klastru adres odesilatele transakce může přidat jedině jednoprvkové klastry obsahující jednorázové adresy.

4.2 Rozdělení superklastru

Vznik superklastru po použití heuristik zaměřených na change adresy byl již v literatuře zaznamenán. Například Neudecker a Hartenstein [14] při spojování dvou klastrů použili omezující podmínku, že se velikost největšího z nich nesmí zvětšit o více než zvolenou konstantu. Ermilov, Panov a Yanovich [4] kolapsu klastrů zabránili díky off-chain informacím, neboť nesjednocovali dva klastry, pokud obsahovaly označené adresy patřící navzájem různým entitám. Möser a Narayanan [12] využili výstup svého modelu rovněž pro stanovení omezujících podmínek pro nespojení dvou klastrů.

Přesto je velikost superklastru vzniklého po použití age zero heuristiky nečekaně velká, neboť superklastr pohltil takřka všechny největší klastry z heuristiky H1. Část z nich přitom odpovídá klastrům největších směnáren. Největší entity přitom jistě znají rizika přijímání transakcí nacházejících se v příliš nízké hloubce na blockchainu. Například burza Coinmate schvaluje transakce přicházející od uživatelů, až když se nachází v hloubce dvou bloků. Vysvětlení vzniku superklastru obsahujícího téměř třetinu všech adres po použití heuristik H1 a H2 se nabízí několik. První možností je, že burzy nedodržují bezpečnostní pravidla striktně za všech okolností. Nebo tato pravidla nedodržují vědomě například při transakcích mezi dvěma burzami navzájem, neboť si důvěřují nebo mají vzájemné transfery právně ošetřené.

Na superklastr obsahující 347 milionů adres jsme se zaměřili detailněji a pokusili jsme se v něm rozlišit podklastry různých entit. Vyšli jsme přitom z hranově ohodnoceného grafu $G = (V,E,w)$ vzešlého z klastrování pomocí multi-input heuristiky s ohodnocovací funkcí $w : E \to \mathbb{N}$. Množina vrcholů V tedy obsahuje klastry $C_1, \ldots, C_n$ vytvořené pomocí multi-input heuristiky. Hrany E mezi jednotlivými klastry jsou přitom vytvořeny na základě age zero heuristiky, přičemž jejich váha je daná maximální velikostí TXO zaslaného mezi patřičnými klastry a utraceného při stáří nula. Váhy lze potom interpretovat jako míru důvěry mezi entitami stojícími za danými klastry, neboť uživatelé mohou být ochotni podstoupit riziko ztráty menšího množství bitcoinů, ale u větších obnosů už takové riziko tolerovat nemusí. Mějme tedy dva klastry $C_1, C_2 \in V$. Hrana $e = \{C_1,C_2\} \in E$ právě tehdy, když $\exists a_1, a_2 \in \mathcal{A}, t_1 = T(i_1,j_1), t_2 = T(i_2,j_2) \in \mathcal{T}$ takové, že $a_1 \in C_1$, $a_2 \in C_2$, $a_1 \in I(t_1), i_1 = i_2$ a zároveň adresu a_2 obsahuje výstupní TXO transakce t_1, které je přitom utraceno v transakci t_2. Váha $w(e)$ potom odpovídá maximální hodnotě těchto výstupních TXO.

Takto vytvořený graf tedy obsahuje 527 milionů vrcholů, z nichž 437 milionů reprezentuje pouze triviální jednoprvkové klastry. Dále budeme pracovat pouze s podgrafem tohoto grafu odpovídajícím původnímu superklastru. Ten představuje největší komponentu souvislosti obsahující 26 milionů vrcholů a takřka 30 mi-

lionů hran. Jejich počet je tedy jen o 15 % vyšší než počet vrcholů. Naším cílem je v tomto podgrafu nalézt odlišitelné komunity a podle nich pak celý superklastr rozdělit. Jako první přístup jsme zvolili odstraňování hran s nejmenší váhou. Motivací k tomuto kroku byla hypotéza, že hrany mezi klastry patřícími dvěma různým entitám by mohly mít na rozdíl od ostatních jen nízkou váhu.

Výsledky počtů a velikostí komponent souvislosti po odstranění všech hran s váhou nižší než je hodnota prahu pohybující se v rozmezí od 10^{-4} BTC do 10 BTC jsou zachyceny na Obrázku A.7. Při odstraňování hran s nejmenšími vahami se až do hodnoty 0,01 BTC velikost největší komponenty snižovala jen pomalu, zatímco počet vytvářených singletonů rostl výrazně rychleji. Přijatelné velikosti v řádu jednotek milionů klastrů největší komponenta dosáhla, až když se z grafu odebralo tolik hran, že v něm zůstala většina vrcholů osamocených. Neprokázala se tedy hypotéza, že by se v grafu vyskytovalo několik desítek či stovek komunit, které by byly navzájem propojeny jen slabě. V superklastru se naopak vyskytuje komponenta několika milionů klastrů, které jsou spojeny pouze hranami s poměrně vysokými váhami převyšujícími hodnotu jednoho bitcoinu. Na základě těchto výsledků se můžeme domnívat, že množina největších směnáren si navzájem důvěřuje a neobává se double-spending útoku ze strany jiné směnárny.

Podobných výsledků jsme vzhledem k relativně nízkému počtu hran v grafu dosáhli rovněž při použití Louvainovy metody pro komunitní detekci zohledňující modularitu jednotlivých komunit. Pomocí ní jsme rovněž dokázali nalézt jednu komunitu zahrnující přes polovinu vrcholů a zbývající vrcholy zůstaly osamoceny ve velkém množství malých či přímo jednoprvkových komunit.

Rozhodli jsme se tedy místo odstraňování hrany postupně přidávat. Vybrali jsme přitom 1000 nejdůležitějších vrcholů z hlediska počtu adres, které odpovídající klastry reprezentují, a z těch spustili upravené prohledávání do šířky, které k jednotlivým centrům hladovým způsobem postupně připojovalo uzly v jejich sousedství. Použili jsme přitom prioritní frontu z důvodu upřednostnění hran s vyššími váhami. Tuto frontu jsme navíc inicializovali sousedstvím všech centrálních uzlů, abychom snížili závislost výsledků algoritmu na pořadí zpracování iniciálních uzlů, viz Algoritmus 2.

Tímto způsobem jsme dosáhli rovnoměrnějšího rozložení velikostí jednotlivých komunit, kdy největší obsahovala přes milion uzlů a několik desítek dalších dosahovalo řádu statisíců. Tato velikost přitom nebyla prokazatelně závislá na počtu adres obsažených ve vybraných centrálních uzlech, neboť i vrcholy reprezentující řádově méně adres než ty největší nakonec dokázaly vytvořit komunity patřící mezi ty největší, viz Obrázek A.8. Přestože některé vytvořené komunity ve skutečnosti zřejmě patří stejným entitám a odpovídají jen například jejich odděleným peněženkám, dokázali jsme tímto způsobem rozdělit superklastr na množství netriviálních částí a vylepšit tak podobu původního klastrování. Pokud bychom disponovali příklady adres větší části směnáren či jiných velkých entit, mohli bychom je použít místo uměle zvolených center a obdobným způsobem vylepšit klastrování představená v předchozí podkapitole do přijatelnější podoby a stejnou metodu aplikovat i na ostatní velké klastry.

Závěr

V práci jsme se věnovali klastrování bitcoinových adres. Našim cílem bylo rozdělit přes miliardu existujících adres do menšího počtu skupin podle jejich příslušnosti ke stejné entitě. Při klastrování jsme vycházeli zejména ze způsobu vytváření transakcí jednotlivými uživateli. Proto jsme v první kapitole detailně popsali fungování celého Bitcoinu. Na rozdíl od většiny ostatních autorů jsme formálně definovali veškeré používané objekty od základních UTXO až po samotný blockchain a popsali pravidla, která pro tyto objekty platí. Vysvětlili jsme přitom rizika pro uživatele plynoucí z nestandardních způsobů vytváření adres a transakcí a uvedli možné dopady podobných okrajových případů na následnou analýzu.

Po zavedení veškerých potřebných termínů a popsání jejich vztahů jsme navázali na teoretické aspekty Bitcoinu obsáhlým výčtem praktických poznatků, které lze využít při rozpoznávání adres patřících stejné entitě. Vzhledem k jejich různorodosti jsme je přehledně roztřídili a podrobně jsme je vysvětlili i s ohledem na jejich potenciální spolehlivost a přínosnost. Složitější vlastnosti jsme názorně vysvětlili na příkladech či přímo zobrazili na přiložených obrázcích. Vycházeli jsme ze souvisejících prací citovaných autorů, jejichž postupy a výsledky jsme rovněž zaznamenali. Formulovali jsme přitom kritiku některých jejich výsledků či nesprávných předpokladů.

Hlavním přínosem této práce je popsání nové metody klastrování založené na dosud nepopsané vlastnosti části transakcí, jenž jsou vytvářeny velmi rychle po sobě. Dále jsme formálně definovali známé i nové heuristiky sloužící k prokázání společného majitele dvojic adres a jejich výsledky jsme vzájemně porovnali. V závislosti na použité kombinaci heuristik jsme analyzovali problém vzniku superklastru obsahujícího neúměrně velkou část adres. Navrhli jsme metodu, pomocí které lze takový superklastr rozdělit na menší části, které již mohou odpovídat realitě. Ukázali jsme přitom, že jej nelze vhodně rozdělit pomocí triviálního odstranění slabě ohodnocených hran ani pomocí Louvainovy metody pro detekci komunit.

V naší práci jsme byli nejvíce limitovaní malým množstvím testovacích dat. Na naši práci tak lze navázat získáním většího množství informací například o adresách jednotlivých směnáren a tyto znalosti pak vhodně využít ať už při samotném klastrování, nebo při následné analýze potenciálních superklastrů. Z výsledků naší práce lze vyjít i v navazující úloze při trénování různých modelů strojového učení sloužících ke klasifikaci typů entit stojících za jednotlivými klastry adres.

Další zlepšení by mohla přinést metoda neaplikující heuristiky globálně na celý blockchain, ale postupně jen na rozšiřující se okolí uzlů reprezentujících význačné adresy směnáren. Nevyřešeným problémem zůstává nalezení správné hranice mezi řetězcem transakcí vytvářených směnárnou a první transakcí vytvořenou jinou entitou. Otevřená zůstává rovněž opačná otázka, jakým způsobem směnárny příchozí transakce zpracovávají. Vzhledem k velikosti dat a jejich neustálému růstu je potřeba použít časově i paměťově efektivní grafové algoritmy, které se navíc budou schopny vypořádat s množstvím na blockchainu se vyskytujících výjimek.

Seznam použité literatury

[1] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf, 2008. Accessed: 2023-04-23.

[2] Ehab Zaghloul, Tongtong Li, Matt W. Mutka, and Jian Ren. Bitcoin and blockchain: Security and privacy. *IEEE Internet of Things Journal*, 7(10):10288–10313, 2020.

[3] Yuhang Zhang, Jun Wang, and Jie Luo. Heuristic-based address clustering in bitcoin. *IEEE Access*, 8:210582–210591, 2020.

[4] Dmitry Ermilov, Maxim Panov, and Yury Yanovich. Automatic bitcoin address clustering. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 461–466, 2017.

[5] Andreas M. Antonopoulos. *Mastering Bitcoin: Unlocking Digital Cryptocurrencies*. O'Reilly Media, Inc., 1st edition, 2014.

[6] Deepak Puthal, Nisha Malik, Saraju Mohanty, Elias Kougianos, and Gautam Das. Everything you wanted to know about the blockchain: Its promise, components, processes, and problems. *IEEE Consumer Electronics Magazine*, 7:6–14, 07 2018.

[7] Anil Gaihre, Yan Luo, and Hang Liu. Do bitcoin users really care about anonymity? An analysis of the bitcoin transaction graph. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 1198–1207, 2018.

[8] Dorit Ron and Adi Shamir. Quantitative analysis of the full bitcoin transaction graph. In *Financial Cryptography and Data Security*, pages 6–24, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[9] Mikkel Alexander Harlev, Haohua Sun Yin, Klaus Christian Langenheldt, Raghava Rao Mukkamala, and Ravikiran Vatrapu. Breaking bad: De-anonymising entity types on the bitcoin blockchain using supervised machine learning. In *Hawaii International Conference on System Sciences*, 2018.

[10] Jon Kleinberg. An impossibility theorem for clustering. *Adv Neural Inform Process Syst (NIPS)*, 15, 2003.

[11] Lei Wu, Yufeng Hu, Yajin Zhou, Haoyu Wang, Xiapu Luo, Zhi Wang, Fan Zhang, and Kui Ren. Towards understanding and demystifying bitcoin mixing services. pages 33–44, 2021.

[12] Malte Möser and Arvind Narayanan. Resurrecting address clustering in bitcoin. In *Financial Cryptography and Data Security*, pages 386–403, Cham, 2022. Springer International Publishing.

[13] Changhoon Kang, Chaehyeon Lee, Kyungchan Ko, Jongsoo Woo, and James Won-Ki Hong. De-anonymization of the bitcoin network using address clustering. In *International Conference on Blockchain and Trustworthy Systems*, 2020.

[14] Till Neudecker and Hannes Hartenstein. Could network information facilitate address clustering in bitcoin? In *Financial Cryptography and Data Security*, pages 155–169, Cham, 2017. Springer International Publishing.

[15] Michele Spagnuolo, Federico Maggi, and Stefano Zanero. Bitiodine: Extracting intelligence from the bitcoin network. In *Financial Cryptography and Data Security*, pages 457–468, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.

[16] Sarah Meiklejohn, Marjori Pomarole, Grant Jordan, Kirill Levchenko, Damon McCoy, Geoffrey M. Voelker, and Stefan Savage. A fistful of bitcoins: Characterizing payments among men with no names. Proceedings of the ACM SIGCOMM Internet Measurement Conference, IMC, pages 127–139, 2013.

[17] Pranav Nerurkar, Dhiren Patel, Yann Busnel, Romaric Ludinard, Saru Kumari, and Muhammad Khurram Khan. Dissecting bitcoin blockchain: Empirical analysis of bitcoin network (2009–2020). *Journal of Network and Computer Applications*, 177:102940, 2021.

[18] Gibran Gomez, Pedro A. Moreno-Sanchez, and Juan Caballero. Watch your back: Identifying cybercrime financial relationships in bitcoin through back-and-forth exploration. *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, 2022.

[19] Igor Makarov and Antoinette Schoar. Blockchain analysis of the bitcoin market. Working Paper 29396, National Bureau of Economic Research, 2021.

[20] Jiajing Wu, Jieli Liu, Yijing Zhao, and Zibin Zheng. Analysis of cryptocurrency transactions from a network perspective: An overview. *Journal of Network and Computer Applications*, 190:103139, 2021.

[21] Bishenghui Tao, Hong-Ning Dai, Jiajing Wu, Ivan Wang-Hei Ho, Zibin Zheng, and Chak Fong Cheang. Complex network analysis of the bitcoin transaction network. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 69(3):1009–1013, 2022.

[22] Stephen Ranshous, Cliff A. Joslyn, Sean Kreyling, Kathleen Nowak, Nagiza F. Samatova, Curtis L. West, and Samuel Winters. Exchange pattern mining in the bitcoin transaction directed hypergraph. In *Financial Cryptography and Data Security*, pages 248–263, Cham, 2017. Springer International Publishing.

[23] Mark Weber, Giacomo Domeniconi, Jie Chen, Daniel Karl I. Weidele, Claudio Bellei, Tom Robinson, and Charles E. Leiserson. Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. *ArXiv*, abs/1908.02591, 2019.

[24] Marc Jourdan, Sebastien Blandin, Laura Wynter, and Pralhad Deshpande. Characterizing entities in the bitcoin blockchain. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 55–62, 2018.

[25] Alexandre Bovet, Carlo Campajola, Francesco Mottes, Valerio Restocchi, Nicolò Vallarano, Tiziano Squartini, and Claudio J. Tessone. The evolving liaisons between the transaction networks of bitcoin and its price dynamics, 2019.

[26] Harry Kalodner, Malte Möser, Kevin Lee, Steven Goldfeder, Martin Plattner, Alishah Chator, and Arvind Narayanan. BlockSci: Design and applications of a blockchain analysis platform. In *29th USENIX Security Symposium (USENIX Security 20)*, pages 2721–2738. USENIX Association, 2020.

[27] Yulin Liu, Luyao Zhang, and Yinhong Zhao. Deciphering bitcoin blockchain data by cohort analysis. 2021.

[28] Martin Harrigan and Christoph Fretter. The unreasonable effectiveness of address clustering. In *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*, pages 368–373, 2016.

Seznam použitých termínů

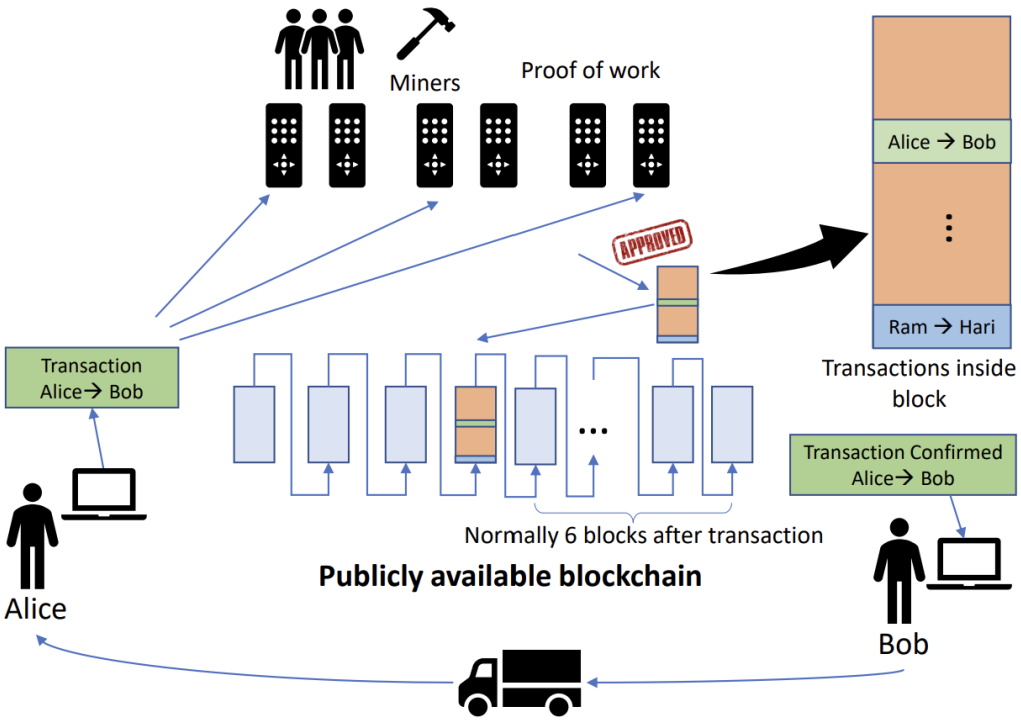
- Adresa - řetězec znaků, který identifikuje adresáta bitcoinů.
- Blockchain - databáze uskutečněných transakcí, řetězec navazujících bloků.
- Blok - skupina transakcí opatřená metadaty.
- Coinbase transakce - speciální transakce uvolňující nové bitcoiny do oběhu.
- Cold walletka - peněženka určená pro bezpečné uložení velkých částek.
- Entita - obecné označení pro majitele skupiny adres.
- Hashrate - výkon těžařů měřený v počtu provedených hashů za určitý čas.
- Hot walletka - peněženka provádějící běžné transakce s nižšími částkami.
- Change adresa - adresa na výstupu transakce určená k vrácení přebytečných bitcoinů odesilateli.
- Mining pool - společenství spolupracujících těžařů.
- Mempool - fronta nových transakcí čekajících na zapsání do blockchainu.
- Mixovací služba - služba propojující uživatele za účelem společného utracení jejich bitcoinů.
- Multisig adresa - adresa opatřená více než jedním párem klíčů.
- Nonce - speciální číslo hledané těžaři figurující v každém bloku.
- Off-chain - označení pro věci týkající se blockchainu, přitom v něm přímo neobsažené.
- On-chain - označení pro věci odvoditelné z dat na blockchainu.
- One-time change - jednorázově použitá change adresa.
- Peeling chain - řetězec transakcí postupně utrácejících velké UTXO.
- Peněženka - nástroj pro správu klíčů k adresám a k vytváření transakcí.
- Satoshi - nejmenší přenositelný díl bitcoinu pojmenovaný po jeho tvůrci.
- Směnárna - společnost poskytující službu směny bitcoinů za jiné měny.
- Těžba - proces přidávání nového bloku na konec řetězce.
- Transakce - záznam o přesunu bitcoinů mezi stanovenými adresami.
- UTXO - neutracený transakční výstup.
- Walletka - ekvivalentní název pro peněženku.

A. Přílohy

Algoritmus 1 Klastrovací algoritmus

Vstup: Množina transakcí $\mathcal{T}$ a adres $\mathcal{A}$, heuristika H
Výstup: Klastrování C

```
for all a ∈ A do
    C_a ← {a}
end for
for all t ∈ T do
    for all a_1, a_2 ∈ I(t) ∪ O(t) do
        if H(t, a_1, a_2) then
            C_{a_1} ← C_{a_1} ∪ C_{a_2}
            C_{a_2} ← C_{a_1}
        end if
    end for
end for
return {C_a | a ∈ A}
```



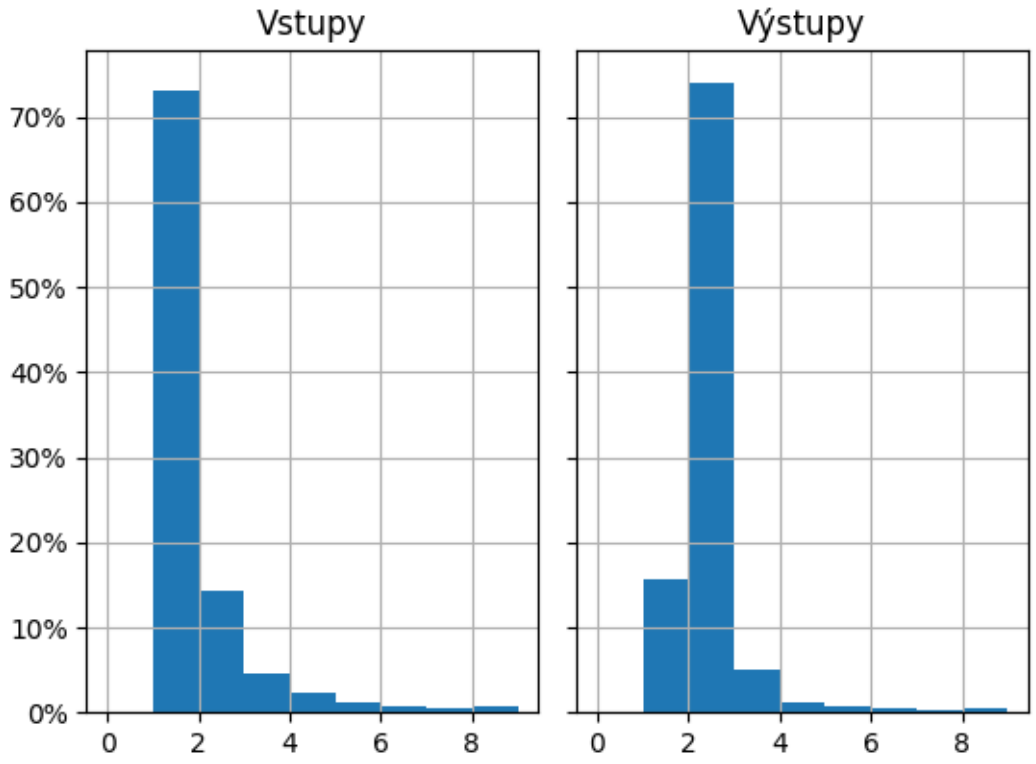
Obrázek A.1: Obrázek 1 ze článku Gaihra, Lua a Liu [7]. Zachycuje průběh možného obchodu mezi Alicí a Bobem od vzniku bitcoinové transakce přes její zveřejnění a ověření těžaři až po její schválení Bobem a poslání reálného zboží na oplátku.

Algoritmus 2 Rozdělení superklastru

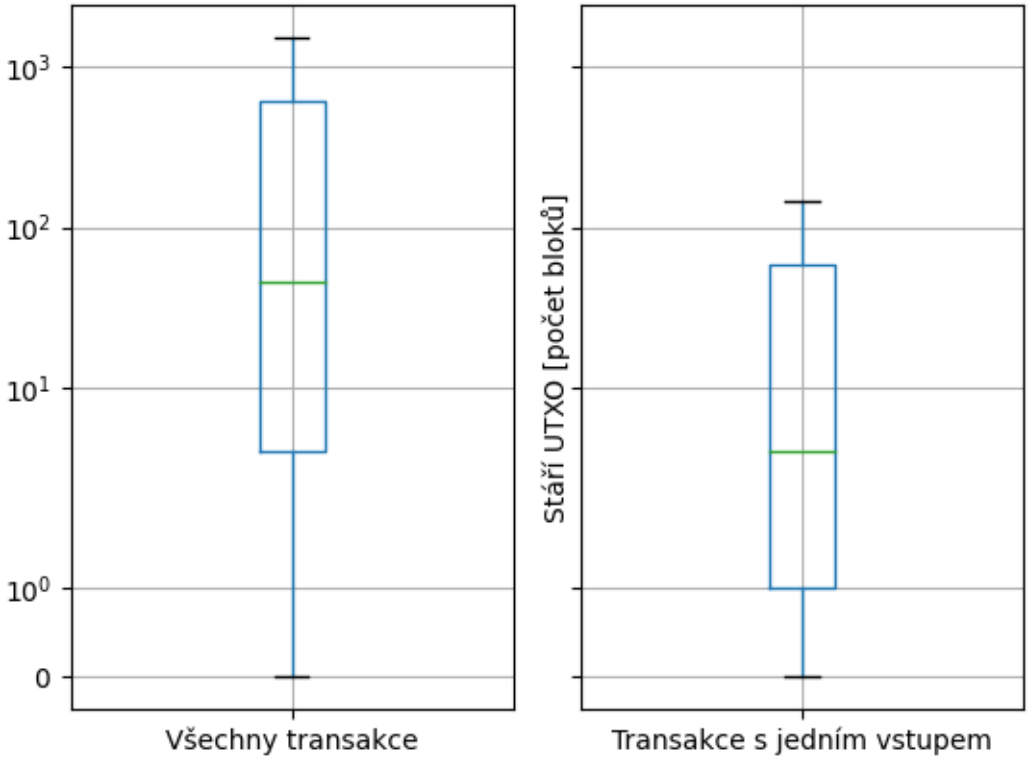
Vstup: Graf $G = (V,E)$ s hranovým ohodnocením $w : E \to \mathbb{N}$ a s označenými uzly $x_1, \ldots x_{1000} \in V$

Výstup: Množina klastrů C

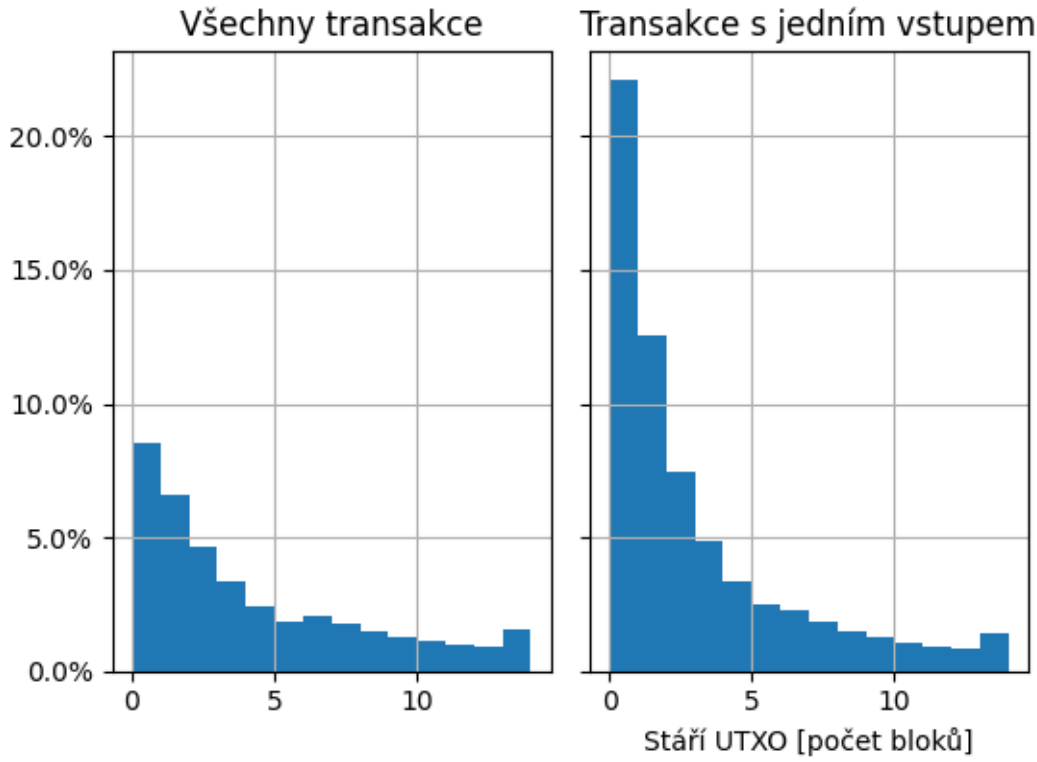
```
Q ← PriorityQueue()                          ▷ according to first parameter
for all u ∈ V do
    b(u) ← 0
end for
for i ← 1 to 1000 do
    C_i ← ∅
    b(x_i) ← i
    for all v ∈ N(x_i) do
        if b(v) = 0 then
            Q.enqueue([w((x_i,v)), v, b(x_i)])
        end if
    end for
end for
while Q.size() > 0 do
    h, u, c ← Q.dequeue()
    b(u) ← c
    for all v ∈ N(u) do
        if b(v) = 0 then
            Q.enqueue([w((u,v)), v, b(u)])
        end if
    end for
end while
for all v ∈ V do
    i ← b(v)
    C_i ← C_i ∪ {v}
end for
return {C_1, ..., C_1000}
```



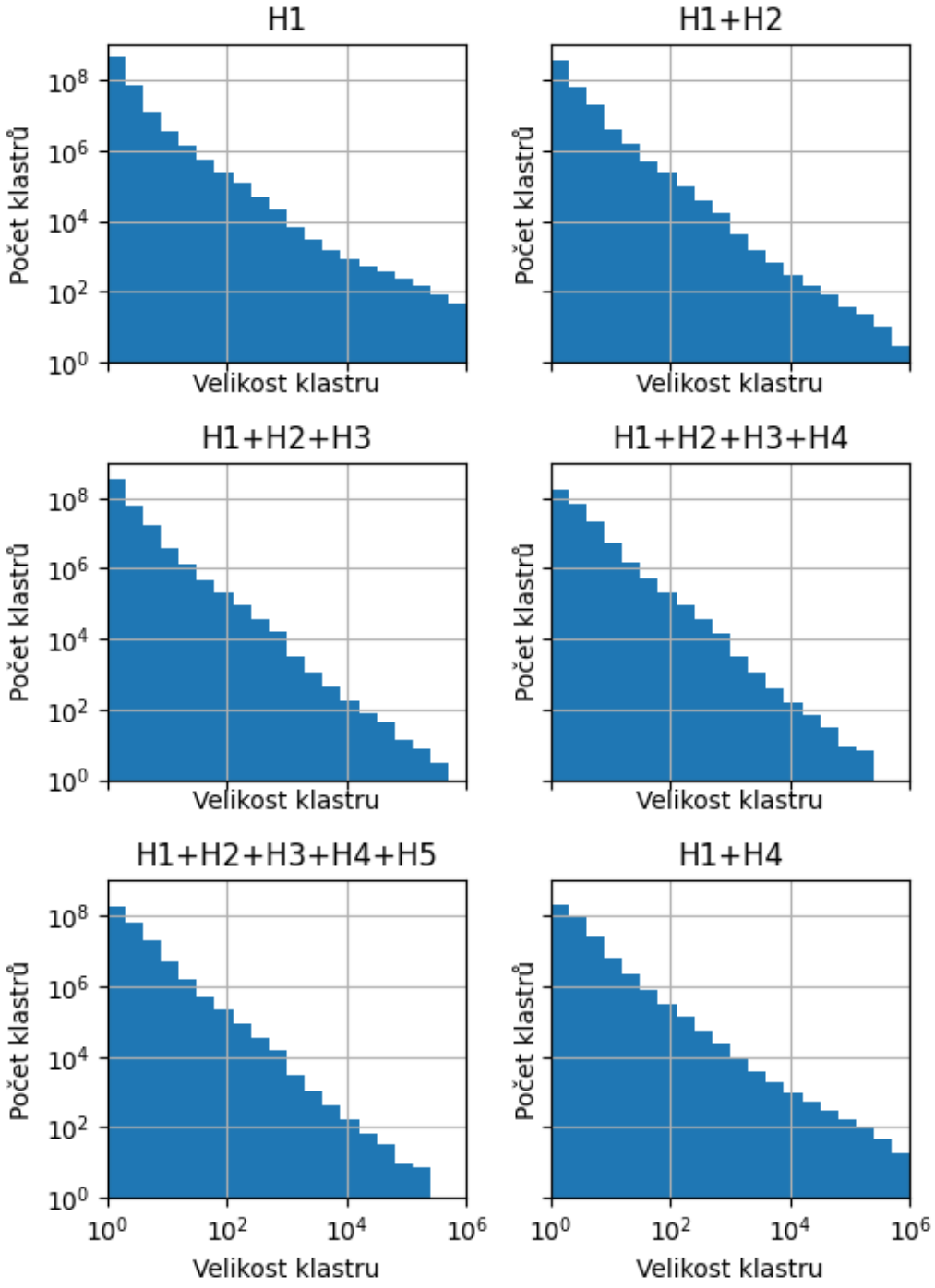
Obrázek A.2: Histogramy relativního zastoupení daných počtů vstupů a výstupů v množině všech transakcí. Vodorovné osy jsou omezeny na hodnoty menší než 10, neboť četnost transakcí s vyšším počtem vstupů či výstupů je zanedbatelná.



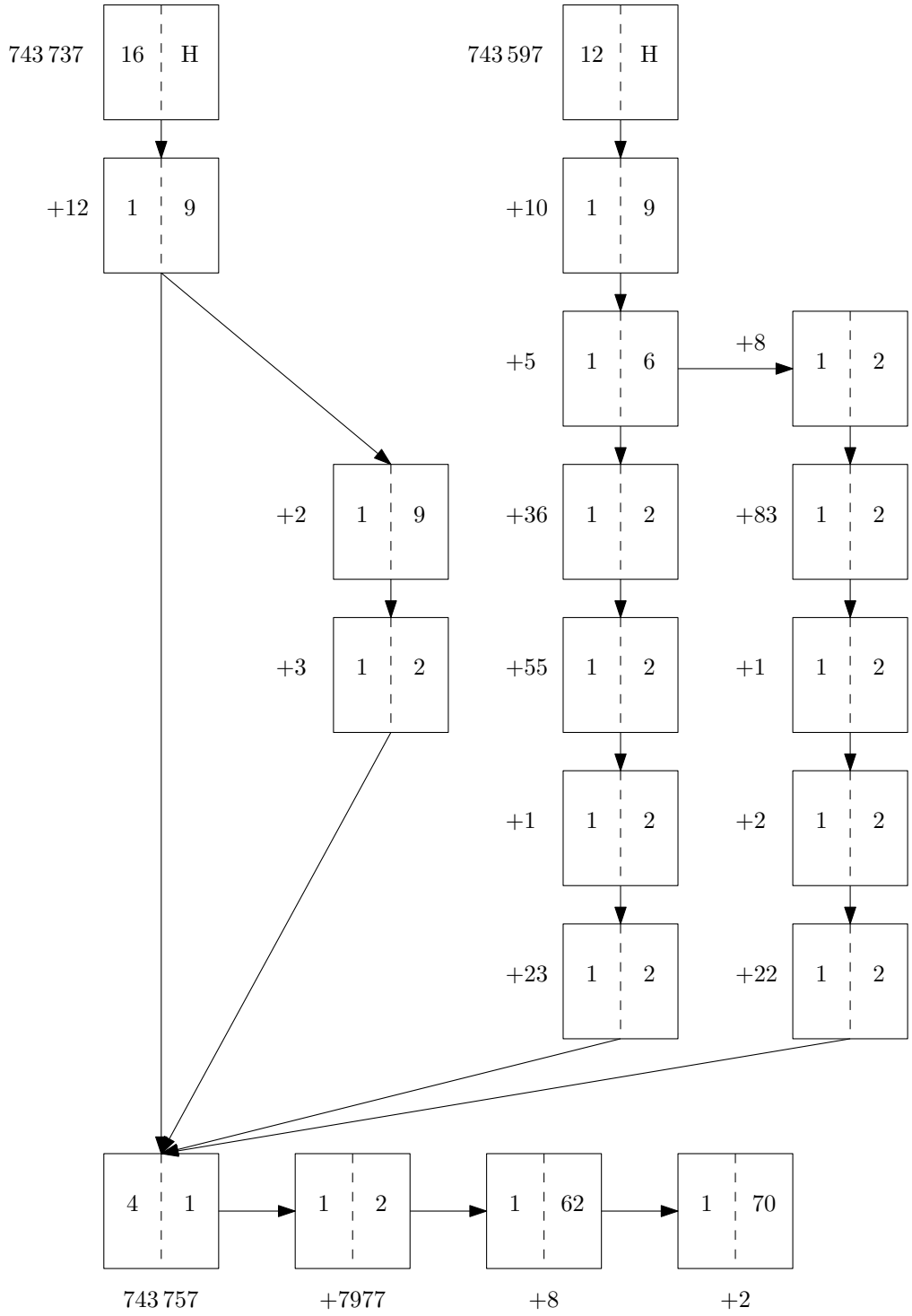
Obrázek A.3: Krabicové grafy bez vykreslených odlehlých hodnot zachycující stáří utrácených UTXO vyjádřené v počtu bloků mezi jejich vznikem a utracením. Levá část je vykreslena ze všech vstupních TXO, pravá část je vytovřena jen ze specifické podmnožiny TXO, které byly utraceny v transakcích s právě jedním vstupem.



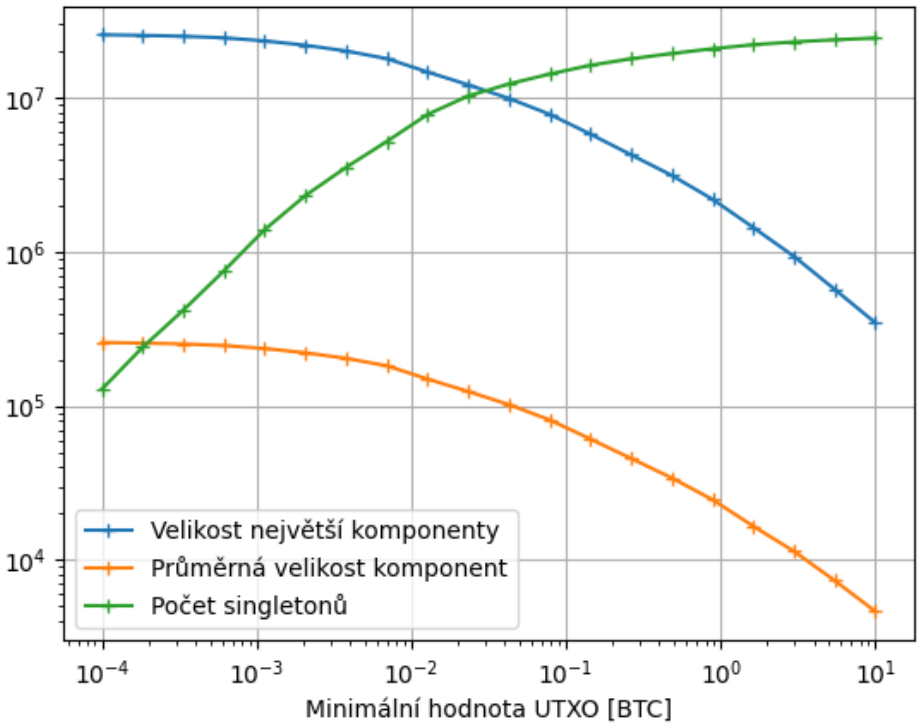
Obrázek A.4: Histogramy relativního zastoupení stáří utrácených UTXO ve všech transakcích, respektive v těch s jedním vstupem. Stáří je vyjádřeno v počtu bloků mezi vznikem UTXO a jeho utracením. Histogramy detailně zachycují jen nejnižší hodnoty, nejsou zde zobrazeny vyšší hodnoty dosahující ve skutečnosti řádů až stovek tisíc bloků.



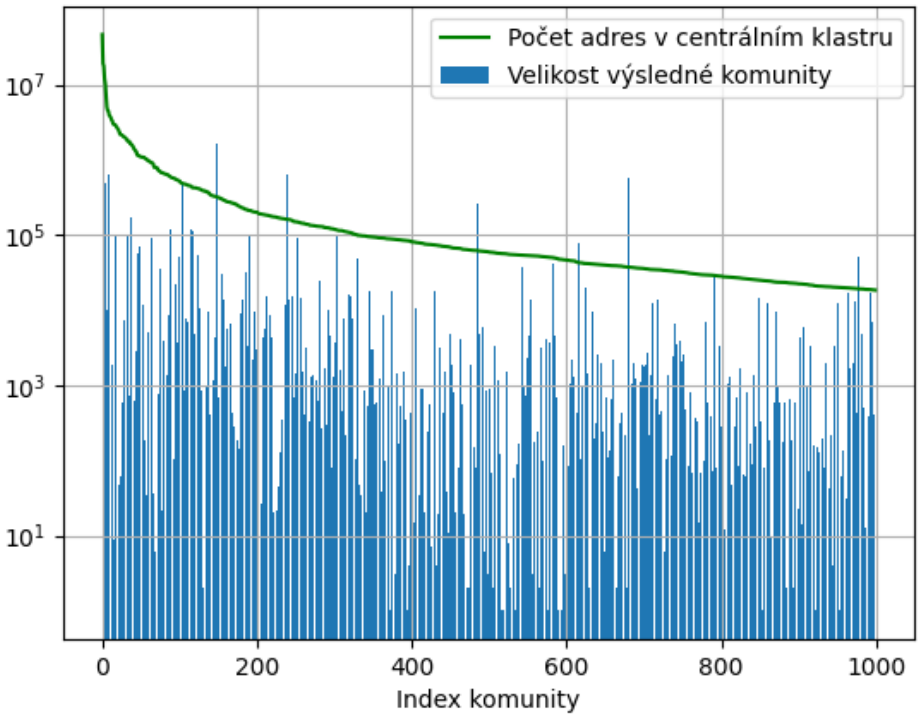
Obrázek A.5: Histogramy zachycující počty klastrů čítajících méně než milion adres pro šest různých kombinací heuristik. Obě osy mají logaritmické měřítko, šířka sloupců tedy roste exponenciálně.



Obrázek A.6: Zpětná analýza původu bitcoinů při výběru prostředků z burzy Coinbase Pro. Poslední transakce vpravo dole obsahuje naši adresu. Písmeno H v horních transakcích značí hot adresu dané burzy. Uvnitř transakcí jsou zobrazeny počty vstupů a výstupů. U transakcí je zobrazeno číslo odpovídajícího bloku nebo jeho relativní vzdálenost od bloku předchozího.



Obrázek A.7: Výsledky různých metrik po odstranění všech hran z grafu G s váhou nižší než je prahová hodnota zobrazená na vodorovné ose. Graf G odpovídá superklastru, jeho vrcholy přitom tvoří klastry vzniklé po použití multi-input heuristiky a hrany zase odpovídají existenci transakce splňující podmínky age zero heuristiky mezi dvěma klastry. První metrikou je velikost největší komponenty souvislosti, druhou je průměrná velikost netriviálních komponent s alespoň dvěma vrcholy a poslední je počet jednoprvkových komponent. Obě osy mají logaritmické měřítko.



Obrázek A.8: Histogram zachycující velikosti komunit nalezených v superklastru pomocí Algoritmu 2. Zeleně je zobrazen počet adres v klastrech, které byly zvoleny jakožto centrální uzly v superklastru. Svislá osa má logaritmické měřítko.