VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INTELIGENTNÍCH SYSTÉMŮ
DEPARTMENT OF INTELLIGENT SYSTEMS

TESTOVÁNÍ BEZPEČNOSTI A VÝKONU PROOF-OF-STAKE PROTOKOLŮ POMOCÍ SIMULACE

SECURITY AND PERFORMANCE TESTBED FOR SIMULATION OF PROOF-OF-STAKE PROTOCOLS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE **Bc. FILIP BORČÍK**
AUTHOR

VEDOUCÍ PRÁCE **Ing. IVAN HOMOLIAK, Ph.D.**
SUPERVISOR

BRNO 2021

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav inteligentních systémů (UITS) Akademický rok 2020/2021

Zadání diplomové práce

Student: **Borčík Filip, Bc.**
Program: Informační technologie
Obor: Kybernetická bezpečnost
Název: **Testování bezpečnosti a výkonu Proof-of-Stake Protokolů pomocí simulace**
Security and Performance Testbed for Simulation of Proof-of-Stake Protocols
Kategorie: Bezpečnost
Zadání:

1. Get familiar with existing proof-of-stake protocols and their popular hybrid variants. Study existing simulation testbeds for blockchain-oriented consensus protocols.
2. Make a theoretical comparison of these protocols in terms of throughput, scalability, security, privacy, failure-tolerance, liveness, safety, finality, etc. In security analysis, consider all existing proof-of-stake vulnerabilities as well as general ones.
3. Select at least three protocols (including Algorand and Casper FFG) and implement them as part of simulation testbed.
4. Make a comparative study of the selected protocols using the results obtained from simulations. Experiment with the simulation of various threats.
5. Based on the simulation results and theoretical analysis, propose a few improvements, which you can validate using the testbed.

Literatura:

- Homoliak, Ivan, et al. "The security reference architecture for blockchains: Towards a standardized model for studying vulnerabilities, threats, and defenses." *arXiv preprint arXiv:1910.09775* (2019).
- Gilad, Y., Hemo, R., Micali, S., Vlachos, G. and Zeldovich, N., 2017, October. Algorand: Scaling byzantine agreements for cryptocurrencies. In Proceedings of the 26th Symposium on Operating Systems Principles (pp. 51-68). ACM.
- Buterin, Vitalik, and Virgil Griffith. "Casper the friendly finality gadget." *arXiv preprint arXiv:1710.09437* (2017).
- Aoki, Y., Otsuki, K., Kaneko, T., Banno, R. and Shudo, K., 2019. SimBlock: a blockchain network simulator. arXiv preprint arXiv:1901.09777.

Při obhajobě semestrální části projektu je požadováno:

- Items 1 and 2.

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce: **Homoliak Ivan, Ing., Ph.D.**
Vedoucí ústavu: Hanáček Petr, doc. Dr. Ing.
Datum zadání: 1. listopadu 2020
Datum odevzdání: 19. května 2021
Datum schválení: 11. listopadu 2020

Abstrakt

Táto práca sa zaoberá testovaním výkonu a bezpečnosti blockchainových protokolov založených na Proof-of-Stake (PoS) modele konsenzu. Opisuje vlastnosti, problémy, ale aj využitie blockchainových systémov. Na teoretickej úrovni porovnáva PoS protokoly Algorand, Casper, Gasper, Snow White, Stellar a Decred z pohľadu vlastností a odolnosti voči rôznym útokom. Práca tiež implementuje simulátor protokolov Algorand, Casper FFG a Gasper. Ako základ vytvoreného simulátoru používa simulačný nástroj Bitcoin Simulator, ktorý je postavený na simulátore diskrétnych sieťových udalostí NS-3. Následne porovnáva vlastnosti implementovaných protokolov pomocou diskrétnej simulácie.

Abstract

This work deals with performance and security testing of blockchain protocols based on the Proof-of-Stake (PoS) consensus model. It describes properties, problems, but also the use of blockchain systems. On theoretical levels, this thesis compares the properties and resistance to various attacks of numerous PoS protocols, specifically Algorand, Casper, Gasper, Snow White, Stellar and Decred. Additionally, this work implements a protocol simulator of Algorand, Casper FFG and Gasper. The simulator is built on top of the Bitcoin Simulator simulation tool, which is based on the NS-3 discrete network event simulator. Then, it compares the properties of the implemented protocols using discrete simulation.

Kľúčové slová

blockchain, PoS, Proof of Stake, Algorand, Casper, Casper FFG, Gasper, bezpečnosť, výkon, útoky, simulácia, NS-3, Bitcoin Simulator

Keywords

blockchain, PoS, Proof of Stake, Algorand, Casper, Casper FFG, Gasper, security, performance, attacks, simulation, NS-3, Bitcoin Simulator

Citácia

BORČÍK, Filip. *Testování bezpečnosti a výkonu Proof-of-Stake Protokolů pomocí simulace.* Brno, 2021. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Ivan Homoliak, Ph.D.

Testování bezpečnosti a výkonu Proof-of-Stake Protokolů pomocí simulace

Prehlásenie

Prehlasujem, že som túto diplomovú prácu vypracoval samostatne pod vedením pána Ing. Ivana Homoliaka, Ph.D. Uviedol som všetky literárne pramene, publikácie a ďalšie zdroje, z ktorých som čerpal.

.....
Filip Borčík
12. mája 2021

Poďakovanie

Ďakujem vedúcemu práce, pánovi Ing. Ivanovi Homoliakovi, Ph.D., za cenné rady a pomoc. Taktiež ďakujem výpočtovým centrám MetaCentrum a CERIT/SC za poskytnuté zdroje, bez ktorých by testovanie protokolov nebolo možné uskutočniť.
Computational resources were supplied by the project "e-Infrastruktura CZ" (e-INFRA LM2018140) provided within the program Projects of Large Research, Development and Innovations Infrastructures.

Obsah

Kapitola 1

Úvod

Začiatkom tohoto storočia vznikla technológia blockchain. Jej prvotným účelom bolo vytvoriť systém pre podporu finančných transakcií bez potreby centrálnej autority. Zároveň ledger, do ktorého boli transakcie zapisované, poskytoval ich transparentnosť a integritu. Vznikla tak prvá kryptomena – Bitcoin, ktorá svoj základ postavila práve na blockchaine. Vďaka jedinečným vlastnostiam, ako sú dostupnosť, nepopierateľnosť, anonymita, či odolnosť voči cenzúre, našli blockchainové systémy uplatnenie aj v iných odvetviach. Využívajú sa v priemysle IoT (internet of things), no aj pri elektronických voľbách.

V posledných rokoch vzniká čoraz viac blockchainových protokolov. Pre zapisovanie transakcií používajú rôzne modely konsenzu. Najznámejším je koncept Proof-of-Work (PoW), ktorý sa zakladá na vykonávaní náročných výpočtových operácií. Problémom tohoto konceptu je jeho energetická neefektívnosť. Napríklad už spomenutý Bitcoin, používajúci model PoW, spotrebuje za rok väčšie množstvo energie ako Holandsko [26]. Ďalším problémom je nízky počet transakcií, ktoré je možné pri tomto type konsenzu dosiahnuť. Oproti centralyzovaným systémom s výkonom približne 50 000 transakcií za sekundu, dosahujú systémy založené na PoW primalý výkon (5 až 10 tx/s).

Potenciálnym riešením sú protokoly založené na modele Proof-of-Stake (PoS). Namiesto náročných operácií sa pomocou rôznych prístupov vyberú účastníci, ktorí sú zodpovední za správne vytvorenie nových blokov. Nielenže tento prístup šetrí elektrickú energiu, ale aj zvyšuje celkovú priepustnosť blockchainu. Avšak protokoly PoS majú taktiež svoje zraniteľnosti. Z tohoto dôvodu je nutné ich dôkladne študovať a porovnávať. Práve to je účelom tejto práce, pričom bol vopred určený cieľ vzájomne porovnať tri protokoly z hľadiska výkonu a bezpečnosti pomocou diskrétnej simulácie.

1.1 Štruktúra práce

Kapitola 2 približuje blockchain bežnému užívateľovi. Informuje o jeho architektúre (2.1) a kategóriách (2.2), do ktorých sa rozdeľujú blockchainové systémy. Popisuje modely konsenzu (2.3) Proof-of-Work, Proof-of-Stake, Proof-of-Activity, ale aj hybridné modely. Zhrňuje informácie o jeho jedinečných vlastnostiach (2.4), no taktiež o problémoch (2.5), s ktorými sa stretáva. Posledná sekcia tejto kapitoly uvádza rôzne oblasti využitia blockchainu (2.6).

Ďalšia časť (kapitola 3) bližšie popisuje protokoly založené na Proof-of-Stake. Konkrétne sa zameriava na protokoly Algorand (3.1), Casper (3.2), Gasper (3.3), Snow White (3.4),

Stellar (3.5) a Decred (3.6). V závere tejto kapitoly je zhrnutie ich teoretických vlastností a odolnosti voči rôznym útokom (3.7).

Pre potreby simulácie nami zvolených protokolov sme sa pozreli na niektoré z existujúcich simulačných nástrojov (kapitola 4). Okrem popisu sme vytvorili tabuľku, ktorá zhrňuje podporu simulácie Proof-of-Stake protokolov jednotlivými nástrojmi (4.6).

V kapitole 5 je opísaný návrh a spôsob implementácie nástroja schopného simulovať vybrané protokoly. Okrem popisu simulačného jadra (5.1) je v tejto kapitole možné nájsť aj konkrétny spôsob simulácie Algorandu (5.2), Caspera FFG (5.3) aj Gaspera (5.4).

Na konci práce (kapitola 6) je opísaný spôsob testovania a získané výsledky zo simulácií. Medzi testami sa nachádzajú simulácie zamerané na vlastnosti protokolov (6.1, 6.2, 6.3), ale aj odolnosť voči útokom s použitím rôznych scenárov (6.4).

Kapitola 2

Blockchain

Aby sme bližšie pochopili problematiku Proof-of-Stake protokolov, potrebujeme mať základnú znalosť blockchainu. Nasledujúca kapitola nám objasní princíp fungovania tejto technológie a taktiež možnosti využitia v rôznych oblastiach.

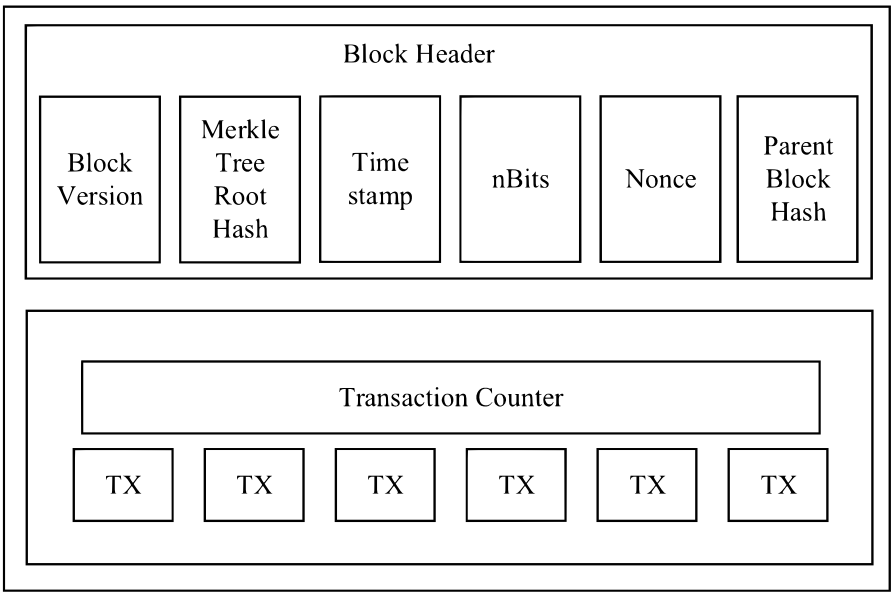
2.1 Architektúra

Na počiatku deväťdesiatych rokov dvadsiateho storočia bolo trendom ukladať dokumenty v digitálnej podobe. Tento spôsob umožnil znížiť cenu ukladania dát mnohonásobne voči použitiu papiera. Avšak, zachovanie integrity a validity digitalizovaných dát bolo v tej dobe dosť problematické, a preto sa hľadali spôsoby ich ochrany. V roku 1991, Dr. W. Scott Stornetta spolu s kolegom Dr. Stuartom Haberom publikovali článok, v ktorom predstavili "blockchain". Jednalo sa o decentralizovanú a kryptovanú databázu s nasledujúcim princípom fungovania. Dokument bol na serveri podpísaný časovou známkou a pomocou ukazovateľov spojený s predchádzajúcim a nasledujúcim dokumentom. Tieto ukazovatele smerovali na konkrétne dáta a nie na umiestnenie dokumentu. Hneď ako by sa teda dáta zmenili, tak by ukazovateľ prestal byť validný. Dokumenty, ktoré týmto serverom prešli vytvorili akúsi reťaz, pričom ich integrita zostala zabezpečená. Onedlho, v roku 1993 túto techniku zlepšili a aby mohli zvýšiť množstvo spracovaných dokumentov, tak ich ukladali do blokov [10, 48].

Blockchain taký, ako ho dnes poznáme, bol prvýkrát popísaný v októbri roku 2008 v publikácii *Bitcoin: A Peer to Peer Electronic Cash System* [1]. Autor, známy pod pseudonymom Satoshi Nakamoto, vychádzal z už spomenutého konceptu, pričom využíval peer to peer siete. O pár mesiacov neskôr, v januári roku 2009, bola týmto autorom zriadená kryptomena Bitcoin, ktorá stojí na technológii blockchain. Od tohoto momentu sa začala vyvíjať snaha použiť blockchain v rôznych produktoch a službách. Dôvodom bola najmä jeho schopnosť zabezpečiť dôveru aj bez centrálnej autority [10, 88].

Verejný ledger, v preklade účtovná kniha, je hlavnou podstatou tejto technológie. Jedná sa o kompletný distribuovaný zoznam transakcií, ktoré sú uložené do blokov tvoriacich sekvenciu. Obsah blokov ale nemusí tvoriť zoznam informácií o finančných tokoch, ale taktiež čokoľvek, čo je možné spracovať do digitálnej podoby (fotografie, audio, video, ...). Každý blok sa odkazuje na rodičovský blok. Transakcie sú po verifikácii nemenné a je z nich vytvorený hash, uložený v jednotlivých blokoch. Zároveň každý blok nesie kombináciu predošlých hashov, a teda pokiaľ sa informácia uložená v niektorom z predchádzajúcich blokov

[1] Bitcoin: A Peer-to-Peer Electronic Cash System, `https://git.dhimmel.com/bitcoin-whitepaper/`



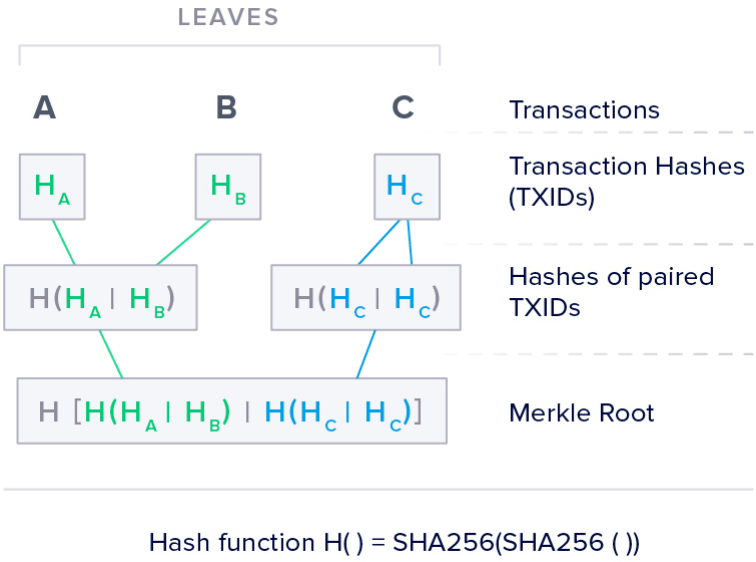
Obr. 2.1: Štruktúra blockchainového bloku [93]

zmení, tak sa o tom dozvie každý jeho následník. Na počiatku tejto sekvencie je vopred nakonfigurovaný blok nazývaný *genesis*. Každý nový účastník blockchainovej siete súhlasí s počiatočným stavom definovaným v tomto bloku [10, 88, 93].

Štruktúra každého bloku sa skladá z niekoľkých položiek tak, ako to môžeme vidieť na obrázku 2.1. Tieto položky sú rozdelené do dvoch hlavných častí, ktorými sú hlavička bloku, obsahujúca informácie dôležité pre fungovanie blockchainu a telo bloku, nesúce jednotlivé transakcie [4, 93, 51].

V hlavičke môžeme nájsť nasledujúce údaje:

- verzia bloku - vyjadruje verziu softwaru/protokolu, ktorá bola použitá na vybudovanie tohoto bloku, a ktoré pravidlá je nutné použiť pre validáciu bloku,
- hash koreňa Merkle stromu - reprezentuje zhrnutie všetkých transakcií obsiahnutých v bloku,
- časová značka - približný čas začiatku vytvárania bloku, vyjadrený v počte sekúnd unixového času,
- nBits - zakódovaná hodnota cieľovej obtiažnosti/prahu nájdenia nového platného bloku,
- nonce - štvorbitové pole obsahujúce hodnotu, ktorú mineri menia tak, aby hash hlavičky bloku zodpovedal zvolenej náročnosti,
- hash rodičovského bloku - referencia na predchádzajúci blok, taktiež zabezpečujúca integritu predchádzajúcich blokov. Tento hash je generovaný dvojitou aplikáciou algoritmu SHA256.



Obr. 2.2: Merkle strom [51]

Telo bloku obsahuje počítadlo transakcií a samotné transakcie. Tie sú uložené v dátovej štruktúre zvanej strom *Merkle*. Samotné transakcie sú uschované v listoch stromu a nad každou z nich je v prvej fáze tvorby stromu vytvorený identifikátor v podobe hashu, ktorý označujeme *TRXID*. Jednoznačné identifikátory sa párovo kombinujú do nových hashov, čím sa postupne vytvára binárny strom, na ktorého vrchole je koreň Merkle stromu. Vždy keď uzol nenájde partnera, tak nový hash je tvorený dvomi hashmi tohoto uzla. Pre priblíženie tejto problematiky je na obrázku 2.2 zobrazená ukážka Merkle stromu. Počet transakcií nie je obmedzený, a teda môže ich byť nula alebo viac. Každá transakcia sa skladá zo vstupov a výstupov. Vstupmi rozumieme digitálne aktíva, ktoré sa majú preniesť spolu so zdrojom, z ktorého sa prevádzajú. Výstupmi sú zväčša adresy prijímateľov spolu s informáciou, ktorá časť aktív sa prevedie na akú adresu. Na zachovanie validity a autenticity dát sa využívajú moderné kryptografické metódy [51, 4, 88].

2.2 Kategorizácia

Aby bolo zaručené dostatočné riadenie blockchainu, musia sa stanoviť určité pravidlá, ktorými sa riadia užívatelia, uzly v sieti a celý systém. Tieto pravidlá musia zodpovedať najmä otázku prístupu k blokom a ich vytváraniu. Určujú kto bude mať možnosť čítania, kto sa bude podieľať na vytváraní nových blokov, ale taktiež, kto bude spravovať konsenzus. V blockchainoch podľa spôsobu prístupu rozlišujeme dve hlavné kategórie. Ak nový blok môže publikovať ktokoľvek, tak hovoríme o *"permissionless"* modele. V opačnom prípade, kde sú možnosti užívateľa obmedzené, používame model *"permissioned"* [80, 87, 88].

2.2.1 Permissionless model

Táto varianta je decentralizovaná a otvorená pre účastníkov. Každý z nich má práva na čítanie ale aj zápis, pričom neexistuje centrálna autorita, kontrolujúca priebeh vytvárania nových blokov. Do siete sa môže kedykoľvek pripojiť nový účastník, ktorý má možnosť rozhodnúť sa či prispeje do budovania blockchainu alebo nie. S veľkou slobodou však pri-

chádzajú aj zlomyseľní užívatelia, ktorí by radi publikovali bloky škodiace systému a hrajúce v ich prospech. Permissionless modely preto častokrát využívajú konsenzus, kedy pri publikovaní nového bloku je nutné schválenie tejto akcie viacerými užívateľmi. S použitím kryptografických techník je taktiež možné zabrániť čítaniu dát nedôvernými účastníkmi. Stratégie založené na permissionless modele majú častokrát nižšiu efektivitu a spotrebujú vyššie množstvo energie ako je to pri modeli permissioned. Schválenie transakcie môže trvať aj niekoľko minút. Tento model využívajú napríklad protokoly Bitcoin, Zerocash alebo Ethereum [80, 87, 88].

2.2.2 Permissioned model

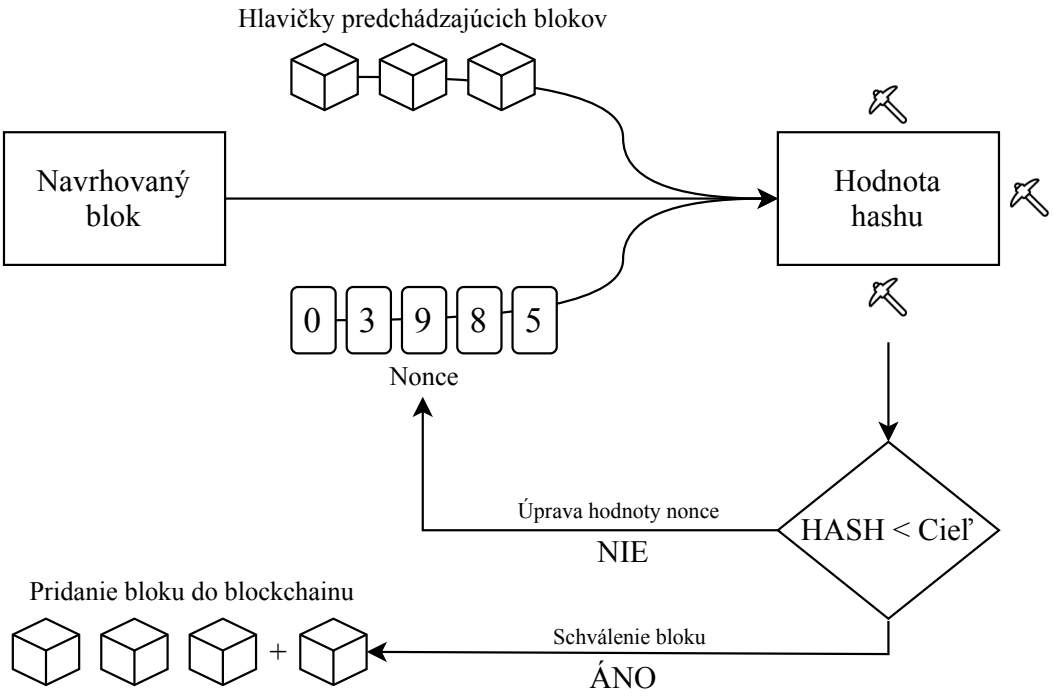
Užívatelia protokolov využívajúcich permissioned model musia pri publikovaní bloku mať pridelený prístup k tejto akcii od vyššej autority, či centralizovanej alebo decentralizovanej. Riadenie prístupu môže taktiež ovplyvniť potvrdzovanie a validáciu transakcií. Účastníci môžu mať taktiež odmietnutý prístup k čítaniu dát. Vtedy tento model označujeme ako **privátny permissioned model**. Právo na čítanie však smie prideľovať len centralizovaná autorita. Pokiaľ dáta môžu čítať všetci účastníci, tak sa model označuje ako **verejný permissioned model**. Rovnako ako permissionless model, aj v tomto sa využíva proces konsenzu, no vďaka voľbe dôveryhodných užívateľov vyššou autoritou je tento proces mnohokrát rýchlejší a lacnejší na výpočtový výkon. Pokiaľ totiž niektorý z účastníkov sklame, tak mu môže byť štatút dôvery odobraný. Niektoré z blockchainov tohoto typu môžu dokonca vyžadovať, aby všetci užívatelia boli pri prijímaní a odosielaní transakcií autorizovaní. Napríklad, po prebehnutí transakcie medzi dvoma stranami by jednotliví účastníci mohli mať čítanie údajov o jej priebehu obmedzené len do určitej miery. Okrem dôvery permissioned model navyšuje transparentnosť, ktorá v prípade zlomyseľných činov značne napomáha vyvodzovaniu zodpovednosti. Príkladmi takýchto blockchainov sú Hyperledger Fabric a R3 Corda [80, 87, 88].

2.3 Modely konsenzu

Blockchainová technológia sa pri svojom fungovaní musí zaoberať niekoľkými problémami. Tým najväčším je rozhodnutie, kto smie publikovať nový blok a ako sa zachovať, keď viacero účastníkov publikuje nový blok v rovnaký čas. Mnoho užívateľov má snahu publikovať nové bloky takým spôsobom, aby dosiahli čo najväčší zisk bez ohľadu na následky. K vyriešeniu tejto problematiky slúži *model konsenzu*. Je to súbor pravidiel a mechaník umožňujúci bezpečné budovanie ledgera. Existuje niekoľko rôznych typov modelov, ktoré sa snažia docieliť spoľahlivosť, autenticitu a presnosť pri vytváraní blokov. Každý z nich prináša iné výhody a nevýhody, založené na rôznych charakteristikách ako sú škálovateľnosť, rýchlosť transakcií, odolnosť voči neoprávnenej manipulácii, či energetická náročnosť. V nasledujúcej časti práce si priblížime niekoľko rôznych modelov [80, 88, 93].

2.3.1 PoW (Proof-of-Work)

Permissionless stratégia PoW je postavená na decentralizovanej topológii siete, využívajúc P2P (peer-to-peer) architektúru. Idea tejto metódy je využiť prácu uzlov na dokázanie platnosti nového publikovaného záznamu o transakciách. Účastník, nazývaný “miner”, má za úlohu spočítať atribút nonce v hlavičke bloku tak, aby výsledný hash bloku dosiahol menšiu hodnotu ako je cieľová. Overovanie správnosti riešenia je však výpočtovo jednoduché.



Obr. 2.3: Princíp fungovania modelu Proof-of-Work [57]

Vďaka tejto vlastnosti môžu účastníci jednoducho zistiť validitu ďalšieho bloku. Na vypočítanie správneho kryptografického hashu je možné použiť jedine metódu brute force. To znamená, že sa skúša generovať každá možná kombinácia hodnoty nonce a následne sa overuje validita bloku. Logicky sa nedá prísť na skratku ku chcenému riešeniu a v niektorých prípadoch je nutné vyskúšať desiatky miliónov rôznych kombinácií bitov. Je preto nutné mať značne veľké výpočtové zdroje a mnoho spoločností sa usiluje o vybudovanie hardvérových fariem slúžiacich na ťaženie. U kryptomien je publikácia bloku odmeňovaná zväčša jednou mincou, a preto má o ťaženie záujem čoraz viac užívateľov [80, 88, 93].

Ako sme už spomínali, hash každého bloku sa vypočítava pomocou algoritmu SHA256, a teda jeho výsledná dĺžka je 256 bitov (32 bajtov). Cieľová hodnota pre vypočítanie hodnoty hashu môže byť časom upravovaná. V prípade technológie Bitcoin je táto hodnota zmenená každých 2016 blokov aby časová náročnosť potrebná pre vznik nového bloku bola približne desať minút. Cieľová hodnota sa najčastejšie upravuje modifikáciou počtu núl na začiatku hashu. Ich zvýšením dosiahneme nárast možných riešení a naopak znížením počtu núl dokážeme zmenšiť výpočtovú náročnosť. Tento proces modifikácie cieľovej hodnoty je nutný z dôvodu technologického pokroku a taktiež väčšieho počtu účastníkov podieľajúcich sa na publikovaní nových blokov [80, 88, 93].

Dôležitou vlastnosťou tohoto prístupu je, že práca jedného uzla nenapomáha ostatným uzlom pri ich výpočtoch. Každý účastník, ktorý chce zverejniť nový blok je tak nútený vypočítať správnu hodnotu nonce sám. V prípade, že niektorý z uzlov vypočíta správnu hodnotu, tak publikuje tento nový blok. Ostatní účastníci blockchainu vykonajú validáciu bloku a v prípade, že je všetko v poriadku, tak sú nútení zahodiť svoju prácu na výpočte validného hashu. Následne musí vytvoriť nový blok, ktorý bude následníkom publikovaného bloku, a teda celý proces opakovať od začiatku [88].

2.3.2 PoS (Proof-of-Stake)

Model Proof-of-Stake je novšia permissionless varianta modelu konsenzu ako Proof-of-Work. Hlavným dôvodom pre vznik tohoto modelu bola predovšetkým snaha znížiť energetické náklady na fungovanie blockchainového systému. Základná myšlienka PoS stavia na psychológii človeka a vraví, že čím viac užívateľ investuje do systému, tým viac chce dosiahnuť jeho úspech, a teda sa menej snaží o útoky na systémovú sieť. Takéto modely majú vopred rozdelenú kryptomenu medzi účastníkov. Generovanie blokov je spojené s množstvom aktív, ktoré sa užívateľ rozhodol do siete staviť. Následná pravdepodobnosť publikovania bloku týmto užívateľom je založená na celkovom množstve aktív v blockchainovej sieti a hodnotou, ktorú užívateľ stavil. Odmena za publikovanie nového bloku je v niekoľkých systémoch nahradená získaním stávok účastníka naspäť, a teda sa negenerujú žiadne nové koruny kryptomeny. Tieto protokoly si bližšie popíšeme v nasledujúcej kapitole (3) [80, 88].

2.3.3 Proof-of-Authority

Tento model je založený na prepojení digitálneho sveta blockchainu s reálnou identitou užívateľov. Fungovanie tejto stratégie je možné len v blockchainových permissioned sieťach s vysokou úrovňou dôvery. Vybraní účastníci majúci splnomocnenie získajú prístup k publikovaniu nových blokov a ku dohliadaniu na bezpečnosť systému. Nazývame ich *dôvernými signatármi*. Každý generovaný blok je podpísaný súkromným kľúčom signatára, a teda užívatelia môžu zistiť, kto stál za podpisom jednotlivých blokov, hoci aj zlomyseľným. V prípade, že došlo k publikovaniu bloku, ktorý nebol v zhode s blockchainovou sieťou, tak jeho signatár príde k strate reputácie. Tá môže viesť k nemožnosti publikovať ďalšie bloky, a preto sa užívatelia snažia udržiavať si reputáciu na vysokej úrovni [80, 88].

2.3.4 Hybridné modely

Okrem štandardných modelov konsenzu existujú aj rôzne hybridy, ktoré sa snažia vylepšiť ich kombináciou svoje vlastnosti. Ako príklad môžeme uviesť hybridný model PoW/PoS. Funguje podobne ako klasické PoW modely. Avšak, pri nájdení platného bloku minerom nedôjde ku jeho automatickej validácii systémom a ku vloženiu do blockchainu. V tento moment prichádzajú piati PoS voliči, ktorí hlasujú o platnosti a zlomyseľnom charaktere daného bloku. Blok je schválený až keď tak rozhodne väčšina hlasujúcich voličov, to je minimálne traja. V prípade, že voliči odmietnu schváliť tento blok, tak mineri prichádzajú o svoju odmenu a nie je im dovolené zapísať transakcie do blockchainu. Hlasovanie je vykonávané pomocou hlasovacích lístkov, ktoré sa podobajú na aktíva štandardných PoS modelov. Odmeňovanie PoS voličov prebieha klasickým spôsobom a navyše sú oprávnení si brať od PoW minerov časť ich odmeny. V systéme je samozrejme obmedzený počet hlasovacích lístkov a užívateľ si smie zakúpiť len isté množstvo. S vyšším množstvom narastá aj ich nákupná cena a po ich nákupe má užívateľ na určitý čas zamknuté aktíva pre ďalšie nákupy. Tento dizajn výrazne zvyšuje bezpečnosť modelu, no jeho rýchlosť nemusí byť ideálna [70].

2.4 Vlastnosti

Blockchain prináša niekoľko výhod, kvôli ktorým je využívaný v rôznych oblastiach. Jeho vlastnosti sa samozrejme môžu líšiť na základe protokolu, ktorý je použitý. Medzi hlavné z nich patria nasledovné.

- **Decentralizácia:** zmeny v ledgeri sú tvorené viacerými účastníkmi, ktorí navzájom spolupracujú. Neexistuje centrálna autorita, ktorá schvaľuje transakcie a ani žiaden súbor pravidiel, na základe ktorého by prebiehal ich schvaľovací proces. Účastníci sú si rovní a ich cieľom je sa z väčšiny zhodnúť na platnosti transakcií [6, 50].

- **Integrita:** transakcie uložené v ledgeri je náročné zmazať alebo upraviť. K tomuto procesu je potrebná väčšina uzlov, ktorá danú zmenu schvália. Pre dosiahnutie integrity sa využívajú kryptografické funkcie zabezpečujúce nemennosť odkazov medzi jednotlivými záznamami. Vďaka integrite ledgera značne narastá bezpečnosť dát [6, 50].

- **Dostupnosť:** blockchain je distribuovaný a využíva replikáciu kompletných dát medzi uzlami. Každý z uzlov obsahuje plnú kópiu ledgera. V prípade výpadku spôsobeného napríklad útokmi je história transakcií naďalej dostupná. Pre úspešný DDoS (distributed denial of service) útok by bolo nutné vynaložiť veľa prostriedkov, a preto je jeho hrozba značne potlačená. Dostupnosť dát má zároveň pozitívny vplyv na dobu trvania zdieľania dát. Negatívnym dôsledkom tejto vlastnosti je problém so spôsobom ukladania dát [6, 50, 66].

- **Odolnosť voči cenzúre:** distribúcia informácií o transakciách medzi uzlami napomáha k odolnosti voči cenzúre. To znamená, že ak existuje transakcia, ktorá je platná, tak sa určite zapíše do ledgera a nie je možné blokovať informácie o nej pred uzlami v sieti [50].

- **Prehľadnosť:** transakcie a aktivity užívateľov sú viditeľné a dostupné pre všetkých [50].

- **Priepustnosť:** počas distribúcie dát o transakciách sa tieto informácie dostanú do celej blockchainovej siete. Následne sa čaká na ich spracovanie a v prípade platných transakcií aj na zápis do ledgera. Priepustnosť reprezentuje rýchlosť spracovania transakcií vyjadrenú v počte za jednotku času (najčastejšie uvádzaná v sekundách). Niektoré protokoly sú schopné spracovať len pár desiatok transakcií za sekundu, no iné v testoch dosahujú priepustnosť o veľkosti stoviek tisíc transakcií [6, 50].

- **Nepopierateľnosť:** každá transakcia vložená do blockchainu je označená časovým razítkom a podpisom. Je možné sledovať ktorý užívateľ, prípadne skupina užívateľov, podpísal transakciu v danom čase, keďže užívatelia majú pridelený práve jeden verejný podpis. Zároveň s každou novou transakciou sa mení stav ledgera a je tak možné uchovať ich kompletnú históriu [66].

- **Anonymita:** pri vytváraní transakcií sa skrýva skutočná identita odosielateľa a príjemcu za unikátne kľúče. Túto vlastnosť je možné využiť napríklad pri elektronických voľbách, ale aj pri ilegálnych nákupoch. Pri použití rôznych techník je anonymita ešte viac posilnená. Príkladom je kryptomena Monero, ktorú si popíšeme neskôr [6, 50].

2.5 Problémy

Popri mnohých možnostiach využitia má blockchain taktiež niekoľko problémov, s ktorými sa musí zaoberať. Niektoré z protokolov sú navrhnuté tak, aby im predišli. Všetko má svoju cenu, ktorou je buďto strata rýchlosti alebo niektorej z ďalších skvelých vlastností tejto technológie.

2.5.1 Škálovateľnosť

Digitalizácia spoločnosti vťahuje do sveta informačných technológií čoraz viac užívateľov. Rovnako je tomu aj pri blockchaine, ktorý neustále nachádza nové možnosti uplatnenia. Zvyšuje sa množstvo transakcií, ktoré čakajú na svoje spracovanie. To je v niektorých protokoloch značným problémom. Ako príklad si môžeme uviesť kryptomenu Bitcoin, v ktorej je nový blok tvorený z bezpečnostných dôvodov raz za približne 10 minút. Zároveň sa do jedného bloku ukladá len niekoľko transakcií, kvôli jeho nízkej kapacite. To má za následok, že vykonanie platby trvá dlhú dobu, a teda nie je možné využiť Bitcoin pri bežných platbách tak, ako je to u bežných mien [20, 93].

Existuje niekoľko rôznych spôsoboch ako predísť týmto problémom. Jedným z prístupov je optimalizácia uložených dát. Počas vytvárania nového bloku je pri kopírovaní záznamov z ledgera možné vynechať staršie z nich. Čím nižší je počet transakcií ukladaných uzlami, tým viac je možné zvýšiť proces validácie blokov [20, 93].

Tento cieľ je možné dosiahnuť aj prístupom, s ktorým prišiel protokol *Bitcoin - Next Generation*. Využíva zmenu návrhu blockchainu. Blok sa rozdelí na dva, pričom v prvom kroku sa určuje líder spomedzi účastníkov. Informácia o výbere lídra sa zapíše do prvého bloku a následne zvolený uzol vytvorí druhý blok obsahujúci záznamy o transakciách [20, 93].

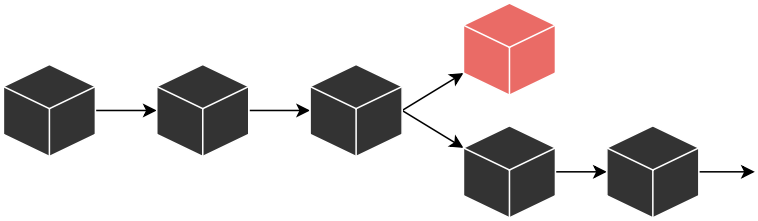
2.5.2 Súkromie

V blockchainoch je nutné vyvažovať pomer medzi prehľadnosťou a súkromím dát. Hrozbu úniku citlivých dát sa snažíme potlačiť využívaním kryptografických techník, ktoré však znižujú výkonnosť systému. Aby sme zachovali anonymitu užívateľov a transakcií, ktoré boli nimi vykonané, je možné využiť niekoľko metód. Hoci užívatelia používajú pseudonymné adresy, tak pri častom používaní rovnakej IP adresy je stále možné spojiť tieto pseudonymy s ich skutočnou identitou [93].

Na potlačenie tejto zraniteľnosti slúži metóda zvaná **mixing**. Pri prevádzaní transakcie medzi dvomi účastníkmi sa pridá ako medzičlánok aj tretia strana. Aktíva sa neodošlú priamo adresátovi ale najskôr sa zašlú prostrednému účastníkovi. Ten prijaté dáta spolu s ďalšími podobne získanými pridá do výstupov niektorej zo svojich transakcií. Skutočná identita odosielateľa transakcie je takto o niečo viac v bezpečí. Mixing je používaný napríklad kryptomenami *Mixcoin, Coinjoin* alebo *CoinShuffle* [63, 93].

Inou voľbou môže byť odstránenie pôvodu aktív pri vykonávaní transakcií. Validácia teda neoveruje či užívateľ mal dostatočné množstvo aktív pred jej vykonaním ale zisťuje sa, či tieto aktíva skutočne existujú v systéme. Platba tak obsahuje iba prijímateľa a množstvo aktív, ktoré sa majú previesť. Tento prístup je využívaný kryptomenou *Zerocoin* [93].

Kryptomena *Monero* využíva k zachovaniu anonymity transakcií techniku zvanú **prstencové podpisovanie** (ring signatures). Pri vytváraní transakcie odosielateľ aktív zvolí veľkosť prstenca určujúcu počet privátnych kľúčov, ktoré vytvoria jeden podpis. Jeden z kľúčov patrí odosielateľovi a zvyšné náhodným užívateľom blockchainu. Všetky z týchto kľúčov



Obr. 2.4: Fork ledgera na dve vetvy [88]

sú jednorázové a pre príjemcu je nemožné rozhodnúť, kto reálne podpísal danú transakciu. Aby sa predišlo dvojnásobnému utrácaniu rovnakých aktív, tak je k transakciám podpísaným užívateľmi pridaný špeciálny kľúč zvaný **"key image"**. Tento kľúč je zaznamenaný priamo v blockchaine a ak by sa niekto pokúsil použiť rovnaký key image, tak by systém túto transakciu zamietol [86].

2.5.3 Fork ledgera

Pri budovaní ledgera občas dochádza k jeho rozdvojeniu, často nazývanému ako **fork** (obrázok 2.4). Tento jav vzniká najčastejšie v permissionless systémoch vytvorením viacerých nových blokov v rovnaký moment. Problémom je, že tieto nové bloky neobsahujú informáciu o sebe navzájom a každý z nich vytvorí samostatnú vetvu. Chýbajúce informácie, obsiahnuté v blokoch z inej vetvy, môžu viesť napríklad k viacnásobnému použitiu aktív rovnakým užívateľom. Blockchainové systémy sa zväčša vedia z tejto situácie rýchlo zotaviť. Pri detekcii rozdvojenia ledgera sa počká na nový blok a podľa toho, do ktorej vetvy sa tento blok pridá, tak sa bude táto najdlhšia vetva považovať za oficiálnu. Ďalšie bloky sa budú pridávať už do nej. Chýbajúce transakcie, ktoré sa dostali do iných vetiev sa vložia do fronty, v ktorej budú čakať na opätovné vloženie do bloku [88].

Aby sa zamedzilo viacnásobnému použitiu aktív, tak sa transakcia nestáva platnou až dokým nie je vytvorených niekoľko ďalších blokov. Čím viac nových blokov je vytvorených z istého bloku, tým menšia je pravdepodobnosť, že dôjde k prepísaniu tohoto bloku z dôvodu forku. To však vedie k ďalšiemu problému. Pokiaľ by mal istý užívateľ dostatočné množstvo výpočtových zdrojov, mohol by popri existujúcom ledgeri vytvoriť z bloku genesis vytvoriť celú novú reťaz, dlhšiu ako existujúci ledger, a tak zmazať celú históriu pôvodného blockchainu [88].

2.5.4 Aktualizácia uzlov

Okrem forku ledgera zvykne dochádzať aj ku forku blockchainových protokolov a to v troch možných variantách. **Soft fork** je zmena v implementácii protokolu, ktorá je spätne kompatibilná. To znamená, že v prípade aktualizácie niektorých z uzlov sú zvyšné uzly schopné vykonávať transakcie aj medzi aktualizovanými uzlami. Príkladom môže byť napríklad zmenšenie veľkosti blokov. Neaktualizované uzly so starou hodnotou vidia nové bloky, vytvorené aktualizovanými uzlami, ako validné nakoľko veľkosť týchto blokov nepresahuje pôvodnú maximálnu veľkosť blokov. Aktualizované uzly už vytvárajú iba bloky s novou maximálnou veľkosťou a väčšie bloky považujú za neplatné [88].

Opačnou variantou je **hard fork**. Pri zmene v implementácii neaktualizované uzly nie sú schopné spolupracovať na transakciách s aktualizovanými a rovnako to platí aj naopak. Teda uzly, ktoré prešli zmenou neakceptujú transakcie vytvorené starou verziou pro-

tokolu a neaktualizované uzly nebudú akceptovať transakcie publikované aktualizovanými uzlami. Dochádza k rozdeleniu systému na dva blockchainy existujúce zároveň. Väčšina takýchto aktualizácií je úmyselná, no môže sa stať, že dôjde k softvérovej chybe, ktorá vytvorí neúmyselný fork blockchainu [88].

Tretia, menej známa, varianta je **velvet fork**. Podobá sa na soft fork, no na rozdiel od neho vyžaduje, aby zmeny neobsahovali žiadny zásah do pravidiel daného konsenzového protokolu. Zmeny tak môžu upravovať iba parametre pomocou premenných namiesto konštantne nastavených hodnôt. Užívatelia majú možnosť rozhodnúť sa, či budú alebo nebudú akceptovať bloky, ktoré sú vytvorené novými pravidlami. Reálne tak nedochádza k rozdvojeniu ledgera ale využíva sa vnútorné odkazovanie na bloky. Pri overovaní užívateľ môže buď akceptovať vnútorný odkaz na bloky podľa nových parametrov, alebo tento odkaz ignorovať a nasledovať klasické odkazovanie na bloky [54].

2.5.5 Sebecké ťaženie (selfish mining)

Jednou z veľkých slabín blockchainových PoW systémov je sebecké ťaženie. Jeho princíp je založený na vytváraní nových blokov bez ich publikovania. Zlomyseľní účastníci investujú výkon svojich hardvérových staníc do blokov bez hrozby akejkoľvek konkurencie. Na počiatku je útočník v stave *mine* a pracuje na vytvorení nového bloku spolu s čestnými uzlami. V momente jeho vytvorenia, si útočník tento blok prevezme a buduje ďalšie bloky z neho. Týmto krokom sa dostáva do stavu *adopt*. Ak sa mu podarí nájsť blok, tak o tejto skutočnosti neinformuje ostatné uzly a tvorí si privátnu reťaz. Ak čestné uzly nájdu blok, tak on tiež publikuje nový blok. Dostáva sa tak do stavu *release*, zníži možnosť prijatia čestne vytvorenej vetvy systémom a vyrovná šancu na zisk za nový blok [36, 93].

Cieľom útočníka je vytvoriť si náskok pred čestnými uzlami a publikovať až vo chvíli, keď sú splnené isté podmienky. Tieto bloky následne prevezmú ostatný užívatelia a útočník získa odmenu. Vďaka tomuto prístupu dokáže s istou pravdepodobnosťou zarobiť na poplatkoch a odmenách za nové bloky viac, ako investuje výpočtových prostriedkov [36].

Ak chceme predísť tejto hrozbe je nutné pridať do blokov časové značky a čestné uzly budú pri vytváraní nových blokov naväzovať na novšie pridané existujúce bloky. Protokol **ZeroBlock** zaviedol taktiež maximálne časové intervaly, v ktorých môže byť nový blok akceptovaný systémom [93].

Ďalším riešením je protokol **StrongChain**, ktorý spája slabé a silné hlavičky blokov pri ich vytváraní. Sila vetvy nezávisí len od jej dĺžky, ale taktiež od počtu oboch typov hlavičiek, ktoré sa v jej blokoch vyskytujú. Tým je zvýšená náročnosť na vytvorenie tohoto typu útoku [79].

Podobný prístup má protokol **FruitChains** kombinujúci klasické bloky s ťažením takzvaného "ovocia". Náročnosť ťažby blokov a ovocia je vyvažovaná tak, aby sa s čo najvyššou pravdepodobnosťou podarilo predísť útokom [79].

2.5.6 Ďalšie problémy

Okrem iných problémov, na ktoré môžeme naraziť v technológii blockchainov si ukážeme ešte dva ďalšie, ktoré sa vyskytujú najmä v modeloch Proof-of-Stake. K prvému, nazývanému problém **"Nothing at stake"** dochádza ak sa v sieti nachádza rozdvojený ledger. Užívateľ má možnosť zapojiť sa do nárastu všetkých vetví s použitím rovnakých aktív, čím je možné obdržať odmenu viacnásobne. Avšak spojenie do samostatnej vetvy v prípade algoritmov PoS môže trvať dlhšiu dobu, čo hrá v prospech zlomyseľných uzlov [88].

Fenomén **"Rich gets richer"** je problémom systémov, v ktorých najvyššiu pravdepodobnosť publikovania nového bloku majú uzly s najväčším množstvom aktív. Takéto uzly sa publikovaním obohatia o ďalšie aktíva, čím získavajú pri ďalšom hlasovaní ešte vyššiu pravdepodobnosť výhry [88].

V prípade vypnutia blockchainovej siete (**blockchain death**) sa môže stať, že kvôli decentralizácii zostanú niektoré uzly naďalej publikovať. Týchto uzlov je však tak málo, že nie sú schopné udržať celú históriu blockchainu. Zlomyseľné uzly získavajú príležitosť k mazaniu a prepisovaniu akéhokoľvek množstva blokov [88].

2.6 Využitie

Okrem rôznych kryptomenových platforiem, nachádza blockchainová technológia svoje uplatnenie aj v iných oblastiach. Aplikovať ju je možné vo finančníctve, IoT priemysle či v sociálnej sfére. Nižšie je uvedených niekoľko z možných využití.

2.6.1 Medzibankové transakcie

Niekoľko bankových systémov sa snaží využiť potenciál k uľahčeniu medzibankových finančných transakcií. V centralizovaných bankových systémoch, ktoré v dnešnej dobe prevládajú, sú prevody medzi subjektmi z rôznych bánk mierne komplikované. Zatiaľ čo pri prevode v rámci jednej banky sa z jedného účtu financie odčítajú a do adresovaného účtu pričítajú, pri medzibankových transakciách do tohoto procesu vstupuje niekoľko ďalších krokov. Tie zahŕňajú prevod financií z účtu odosielateľa na účet banky, odtiaľ na účet centrálnej banky, ktorá financie prevedie na účet banky príjemcu a tá následne zvýši čiastku na účte o potrebnú sumu. Tento proces však znižuje nielen rýchlosť prevodu financií ale taktiež pre každý medzikrok je nutná istá dôvera medzi subjektmi [87].

Blockchainová technológia má vďaka distribuovanej sieti potenciál potlačiť tieto riziká. Niektoré centrálne banky sú nahradené ledgerom, v ktorom majú jednotlivé banky aktíva v hodnote, v ktorej by ich mali na štandardných účtoch centrálnej banky. Pri medzibankovom prevode sa vynechá prevod medzi bankami a centrálnou bankou, pričom s pomocou blockchainu sa transakcie medzi bankami vykonajú priamo. Problém v tomto prípade je, že v rámci jedného blockchainu nie je možné prevádzať financie v rôznych menách [87].

2.6.2 Smart Contracts

Pojem "smart contracts" v súčastnosti má trochu nejasný význam, nakoľko sa používa pre označenie rôznych technológií. Z vyššieho pohľadu sa jeho definície delia do niekoľkých skupín. V prípade technológie blockchainov sa jedná o kategóriu označujúcu akýkoľvek komplexný programový kód, ktorý je uložený, overený a následné spustený blockchainom. Pri uložení v blockchaine tento kód získa jeho vlastnosti, a teda je odolný voči cenzúre a je nemenný. Vďaka týmto vlastnostiam sa vždy spustí tak ako bol napísaný a nikto nemôže do jeho behu zasahovať. S pomocou technológie smart contracts je možné vykonávať operácie rýchlejšie, automatizovane a na vyššej úrovni bezpečnosti. Najčastejšie je určený k správe niečoho dôležitého. Kód má možnosť manipulovať s aktívami blockchainu, vrátane ich ukladania a vykonávania kryptomenových transakčných prevodov. Často je použitý ako súčasť rozsiahlejších aplikácií. Prvýkrát bol pojem smart contracts použitý v roku 1997 známym kryptografom Nickom Szabom. S jeho funkcionalitou pracujú napríklad protokoly *Hyperledger Fabric* a *Ethereum* [75, 78].

2.6.3 Priemysel IoT (Internet of things)

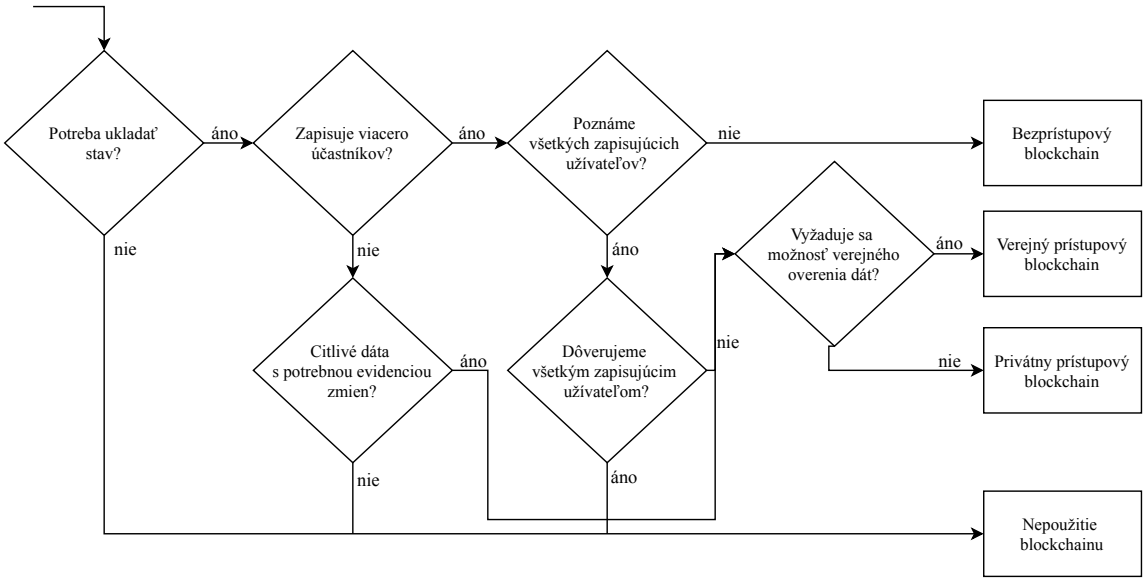
IoT priemysel dnes nachádzame takmer na každom kroku. Automatické zapínanie svetla na chodbe, autá jazdiace bez zásahu vodiča či inteligentné skleníky, v ktorých je možné pestovať zeleninu celoročne. Tieto systémy medzi sebou zdieľajú množstvo dát a ich centralizované riadenie má niekoľko značných nevýhod. S využitím blockchainu je možné predísť nedostupnosti dát pri útokoch ako DoS a zvýšiť ich celkovú integritu. Zároveň jednotlivé zariadenia nie sú viazané na centrálnu autoritu akými sú v súčastnosti rôzne cloudové riešenia. Narušenie ktorejkoľvek z týchto vlastností by bolo s použitím blockchainových protokolov výrazne predražené. Okrem výhod má tento prístup aj niekoľko problémov, ako napríklad spôsob efektívneho ukladania dát na uzloch. Tie sa však dajú potlačiť správnym návrhom konkrétneho riešenia [7, 10].

V spojení s funkcionalitou smart contracts je taktiež možné nechať IoT zariadenia aby platili za skonzumované zdroje či boli odmenené za prácu, ktorú odviedli. To všetko bez akéhokoľvek zásahu tretej strany bez rozdielu na to, či si zariadenia navzájom dôverujú alebo nie. Príkladom využitia blockchainu v IoT je systém ADEPT (Autonomous Decentralised P2P Telemetry) vyvinutý spoločnosťou IBM a slúžiaci ku decentralizovanému prepojeniu zariadení v sieti [10, 20, 87, 92].

2.6.4 Elektronické voľby

Vo viacerých krajinách sveta bol schválený elektronický spôsob hlasovania pri voľbách alebo referendách. Jeho výhody spočívajú najmä v zrýchlení volebného procesu a znížení nákladov na hlasovanie. Vyžaduje však vysokú úroveň súkromia a zároveň je nutné dohliadať na korektný priebeh, v ktorom nikto nebude mať možnosť zmeniť alebo ovplyvniť výsledky. Je potrebné zachovať demokraciu pri takýchto voľbách na čo najvyššej úrovni. Zároveň chceme, aby hlasujúci mali dôveru v korektne vykonané sčítanie hlasov. Práve pri takýchto problémoch je blockchain možným riešením. Výhody, ktoré poskytuje sa snažilo implementovať už niekoľko systémov ako sú napríklad *BitCongress* alebo *Liquid Democracy*. Avšak stále sa nenašlo riešenie, ktoré by dokázalo zvládnuť všetky výzvy elektronického hlasovania. Najbližšie sa dostali systémy, ktoré tento proces rozdelili na dve časti. V prvej majú voliči za úlohu hlasovať a v druhej sa využije verejný permissioned blockchain tretej strany na sčítanie hlasov. Táto technológia tak môže ponúknuť voličom ale aj organizátorom volieb jednoduchý a rýchly spôsob hlasovania, v ktorý môžu dôverovať obe strany [20, 87].

V súčastnosti existuje niekoľko protokolov pre elektronické hlasovacie systémy, ktoré sú založené na blockchaine. B. K. Roy spolu s S. Panjom vytvorili riešenie, v ktorom je možné overovať záznamy o hlasovaní koncovými užívateľmi. Navyše rozšírili pôvodný model aj o biometrický podpis hlasu, ktorý zvyšuje silu autentizácie voličov. Patrick McCorry s jeho tímom predstavili implementáciu protokolu, ktorý nazvali OVN (The Open Vote Network). Využíva Ethereum, do ktorého zapísali OVN vo forme smart kontraktu. Protokol bol navrhnutý tak, aby zachoval súkromie voliča a zároveň dokázal sám sčítavať výsledky hlasovania. Jeho slabosťou je však možnosť výberu len z dvoch hodnôt pri hlasovaní. I. Homoliak s jeho kolegami z Univerzity technológie a dizajnu v Singapure vytvorili protokol BBB-Voting (Blockchain Base Boardroom Voting). Tento protokol zachováva súkromie hlasov, férový prístup, možnosť overovania priebehu hlasovania, je odolný voči poruchám a okrem ďalších vlastností dokáže taktiež sám sčítavať hlasy. Pri hlasovaní podporuje taktiež ľubovoľný počet možností, z ktorých môže volič vybrať jednu. Okrem spomenutých existuje samozrejme aj niekoľko ďalších protokolov [60, 65, 84].



Obr. 2.5: Rozhodovací strom pre výber vhodného blockchainového modelu [34].

2.6.5 Ďalšie možnosti využitia

Blockchainové technológie nachádzajú svoje uplatnenie aj na iných miestach. Pri rozhodovaní toho, aký typ blochcainu použijeme v našom riešení, nám môže pomôcť rozhodovací strom, ktorý môžeme vidieť na obrázku 2.5. Medzi ďalšie spôsoby využitia blockchainu patria napríklad tieto:

- **Big Data** - ukladanie dôležitých informácii pre zachovanie ich integrity a dostupnosti. Príkladom je technológia Streamr, ktorá zbiera množstva dát z rôznych zariadení a umožňuje ich zdieľanie pomocou blockchainovej siete. Producenti týchto dát, prípadne majitelia produktov, ktoré dáta vyprodukovali, majú prehľad o tom, do akých rúk sa tieto dáta dostali [77, 92].

- **Oblasť nehnuteľností** - registrácia pozemkov a zaznamenávanie akýchkoľvek zmien jednotlivých pozemkových parciel. S pomocou tokenizácie nehnuteľností je možné zjednodušiť napríklad rozdeľovanie zisku z nájmu medzi viacerých vlastníkov. K tomuto účelu sú využívané predovšetkým nezameniteľné tokeny, ktoré predstavujú blockchainový pohľad na podiel z aktív, akými sú aj nehnuteľnosti [14, 81].

- **Hudobný priemysel** - umožnenie umelcom propagovať tvorbu medzi zákazníkov a pomocou kombinácie kryptomien a technológie smart contract priamo získavať tržby bez strát na províziách distribútorom [9].

- **Kontrola dovozu** - spotrebitelia získavajú možnosť overiť si skutočný pôvod a dodávateľa tovaru. Často je táto služba využívaná aj v oblasti bioproduktov. Kontrola dovozu a pôvodu tovaru spadá do vyššej aplikačnej vrstvy blockchainu, nazývanej pôvod údajov. Pôvod údajov je postavený na funkcionalite správy identít a súborového systému blockchainového ekosystému [20, 50].

- **Crowdfunding** - umožnenie podieľať sa na riadení projektov podľa toho, ako užívatelia prispeli k ich uskutočneniu skrz platformy ako napríklad Kickstarter. S pomocou smart kontraktov by bolo taktiež možné kontrolovať akési míľniky pri vývoji projektov a kampaní. Cieľom tejto kontroly je predísť premrhaniu finančných projektov, ktoré môžu spôsobiť osoby so zlým úmyslom alebo s nedostatočnou kvalifikáciou [10, 20].
- **Zdravotná starostlivosť** - blockchainová technológia môže pomôcť pri dlhodobom ukladaní záznamov, zdieľaní medicínskych dát ale aj zabránení falšovania liekov. Integrita a transparentnosť dát zaistí, že pacienti budú o ich zdravotnom stave dostatočne informovaní a zdravotné posudky budú vyhodnocované presnejšie. Decentralizáciou taktiež zaistíme, že potrebné dáta budú dostupné aj pri napadnutí organizácie hackermi [10, 20, 92].

Kapitola 3

Proof-of-Stake protokoly

Protokoly PoS využívajú pre dosiahnutie svojich cieľov niekoľko rôznych metód, ktorými volia publikujúci uzol. Prvou je **metóda náhodného výberu**, ktorá pseudonáhodne vyberie užívateľa na základe stávky, ktorú vložil do systému. Pokiaľ napríklad užívateľ vložil 30 % z celkovej hodnoty stávok, ktoré sú v blockchainovej sieti, tak bude volený v tridsiatich percentách prípadov. Tento prístup je však značne zraniteľný [88].

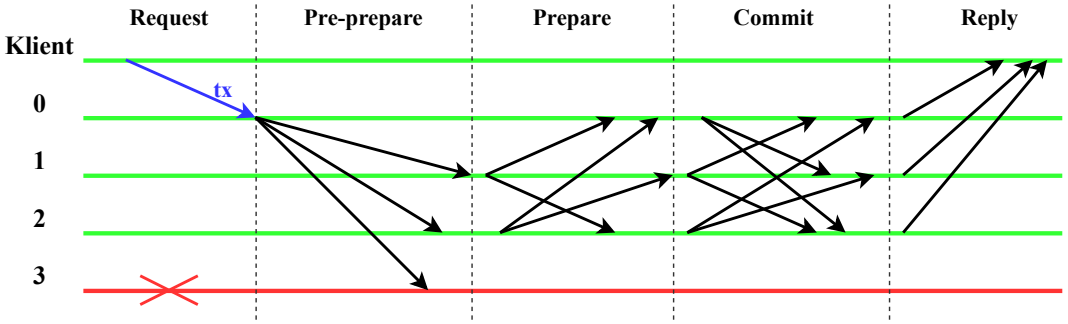
Systémy využívajúce *viackolové hlasovanie*, častokrát nazývané **PBFT** (practical byzantine fault tolerance), vyberajú niekoľkých účastníkov, ktorí sa podieľajú na hlasovaní o užívateľa generujúceho nový blok. Výber hlasujúcich uzlov je založený na určitých pravidlách a toto hlasovanie prebieha v niekoľkých kolách. Každý uzol tak má možnosť zúčastniť sa na procese výberu nového bloku. Tento prístup využíva napríklad Hyperledger Fabric, ktorý dokáže znížiť počet zlomyseľne vytvorených blokov na dve tretiny [88, 93].

Operácia v PBFT sa skladá z nasledujúcich fáz (obrázok 3.1):

- request - klient zasiela transakciu primárnemu uzlu,
- pre-prepare - primárny uzol formuje návrh (napríklad nového bloku) a zasiela ho replikám,
- prepare - repliky overia návrh a v prípade jeho validity odošlú správu *prepare* ostatným uzlom pomocou broadcastu,
- commit - ak aspoň 2/3 replík súhlasili s návrhom, tak sa broadcastom odošle správa *commit*,
- reply - klient vidí výsledok konsenzu [21].

Výber publikujúceho uzla môže byť založený aj na veku stavených kryptomenových aktív (**coin age PoS**). Čím staršiu mincu užívateľ staví, tým má väčšiu pravdepodobnosť, že práve on publikuje nový blok. Aby sa predišlo použitiu rôznych taktík, ktoré by mohli využívať bohatší jedinci, tak sa pridali isté pravidlá. Po stavení aktív do systému sa ich vek vynuluje a zároveň aktíva nebudú môcť byť po istú dobu znovu použité. Navyše sa obmedzuje maximálna pravdepodobnosť úspechu, ktorú môže vek aktív priniesť, aby nedochádzalo ku hromadeniu starých aktív u bohatších účastníkov [88].

V **delegovaných systémov PoS** (DPOS) užívatelia volia spomedzi svojich uzlov zástupcov, ktoré majú publikovať nový blok. Proces generovania a validácie sa s nižším počtom publikujúcich uzlov stáva rýchlejším. Zástupcov zväčša vyberajú uzly s najvyšším množstvom stavených aktív a majú možnosť hlasovať nielen za publikáciu uzla ale aj proti nej.



Obr. 3.1: Priebeh PBFT operácie skladajúca sa z fáz: request, pre-prepare, prepare, commit, reply. Uzol 0 predstavuje primárny uzol, uzly 1 až 3 sú repliky. U repliky číslo 3 došlo k výpadku. [21]

Pri zlomyseľnom zachovaní sa je účastník zbavený možnosti voliť publikovaný uzol, čo prináša vyššiu motiváciu ku korektnému správaniu sa v sieti. Na tomto prístupe je založená napríklad kryptomena *Bitshares* [88, 93].

Táto kapitola nám priblíži metodiku niekoľkých protokolov Proof-of-Stake.

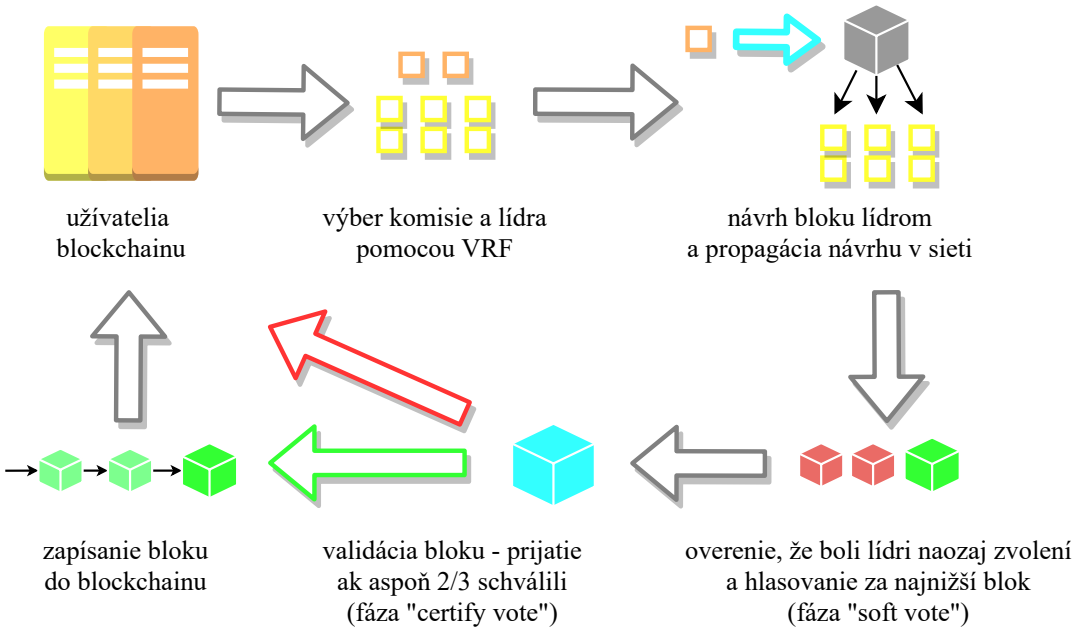
3.1 Algorand

Algorand je v dnešnej dobe známy pod dvomi podobami. Nie je to len blockchainový protokol ale taktiež je to nová menová platforma, podobne ako napríklad Bitcoin, stojaca na tomto protokole. Na rozdiel od iných populárnych protokolov dokáže zabezpečiť všetky tri hlavné vlastnosti (škálovateľnosť, decentralizácia, bezpečnosť). Za jeho vznikom stál taliansky matematik Silvio Micali so svojimi spolupracovníkmi. Od prvého zverejnenia prešiel niekoľkými úpravami, aby mohol dosahovať čo najlepšie výsledky. Vôbec prvý blok kryptomeny Algorand bol vyťažený 11. júna 2019. [45, 95]

Pri tvorbe blockchainového ledgera sa Algorand sústredí na naplnenie troch vlastností. Pokiaľ by niektorá z nich nebola naplnená, tak by sa nedalo vytvoriť dostatočné prostredie pre fungovanie blockchainu. Ich znenie je nasledovné.

1. Každý nový blok sa stane veľmi rýchlo známym pre všetkých užívateľov.
2. Transakcie uložené v bloku sú platné vzhľadom na stav množstva aktív užívateľov k počiatočnému stavu a transakciám v predchádzajúcich blokoch.
3. Každá platná transakcia sa rýchlo uloží do bloku [24].

Tento protokol je možné implementovať v systémoch typu permissionless aj permissioned, pričom permissioned systémy dosahujú podľa jeho autorov lepšie výsledky. Stavia na použití randomizovaných algoritmov pre dosiahnutie byzantskej dohody (**byzantine agreement**), ktoré sa začali rozvíjať už v 80. rokoch 20. storočia. Pri vzniku nových blokov teda používa synchrónny kryptografický protokol, často označovaný ako $\mathbf{BA}^{\star}$. Algorand je binárny protokol, čo znamená, že pri zaslaní správy ostatným účastníkom môže nadobúdať dva stavy, celkový súhlas alebo nesúhlas. Ak je pravdepodobnosť prechodu medzi týmito stavmi väčšia ako 1/3 a zároveň je počet zlomyseľných účastníkov menší ako 1/3 z celkového počtu, tak protokol stále skončí súhlasom s podobou nového bloku. Schválený



Obr. 3.2: Priebeh jedného kola tvorby nového bloku podľa protokolu Algorand, skladajúceho sa z fáz: návrh bloku, soft vote, certify vote [23].

blok je následne podpísaný digitálnym podpisom niekoľkých overovateľov a takto rozšírený do celej blockchainovej siete. Priebeh publikácie nového bloku je zobrazený na obrázku 3.2 [23, 82].

Kryptografické losovanie

Proces byzantského dohadovania podľa spomínaného protokolu prebieha zvlášť pre každý nový blok. Napriek tomu, že byzantské dohadovanie je samo o sebe rýchle, je možné tento proces ešte viac zrýchliť výberom tajnej skupinky užívateľov, ktorá bude rozhodovať o schválení bloku. Aby sme zamedzili možnosti predvídať budúcich overovateľov, sú užívatelia vyberaní náhodne v procese označovanom ako **kryptografické losovanie**. Náhodný výber je v centralizovaných systémoch celkom jednoduchý. Blockchainové systémy sa však snažia byť decentralizované a to robí tento proces zložitejším. Výber skupiny overovateľov spolu s lídrom, ktorý navrhne nový blok, je preto dosiahnutý použitím kryptografie v podobe **VRF (Verifiable Random Functions)** [1, 23].

VRF boli navrhnuté Silviom Micalim už v roku 1999. Skladajú sa z troch funkcií, ktoré sú v Algorande použité nasledovne. Prvá vygeneruje každému užívateľovi v blockchainovej sieti súkromný kľúč, ktorý pozná len on, a verejný kľúč, ktorý je známy aj ostatným užívateľom. Každý blok Algorandu obsahuje špeciálny atribút, ktorým je semienko pre pseudonáhodný výber overovateľov. Oba kľúče užívateľa musia byť vygenerované už pred vytvorením semienka. Pri vytváraní komisie pre schvaľovanie nového bloku sa vezme semienko z predchádzajúceho bloku a spolu so súkromným kľúčom užívateľa sa vložia do druhej funkcie. Tá na základe týchto dvoch vstupov vyprodukuje pseudonáhodnú hodnotu Y a dôkaz ρ. Užívateľ sa stáva účastníkom komisie, prípadne lídrom, ak hodnota Y spadá do rozsahu [0,P], založenom na účastníkovej stávke. K overeniu platnosti hodnoty Y môže každý z účastníkov použiť tretiu funkciu, ktorá vezme verejný kľúč účastníka, semienko z predchádzajú-

ceho bloku, hodnotu Y a dôkaz ρ. Výstup tejto funkcie je binárny, a teda 1 ak je Y platné a 0 v opačnom prípade. Následne komisia spolu s lídrom rozhodujú o novom bloku, ktorý bude vložený do ledgera. V prípade, že je nový blok neschválený, tak sa líder spolu s overovateľmi volia odznovu [46, 61].

Ohrozenie procesu tvorby bloku

V niektorých prípadoch sa stáva, že je zvolený zlomyseľný uzol ako líder komisie, ktorý navrhuje nový blok ledgera. Ten smie poslať každému účastníkovi komisie iný návrh bloku, čo vyvrcholí v pridaní prázdneho bloku do ledgera. Tento stav zamedzí schváleniu reálne vykonaných transakcií. Tvorcovia Algorandu tvrdia, že pravdepodobnosť zvolenia zlomyseľného lídra je nanajvýš $1 - h$, pričom h je počet čestných účastníkov, ktorý je vyšší ako 2/3 [45].

Každé kolo byzantského dohadovania vyžaduje istý počet hlasov T, ktoré označia blok za platný pri istej veľkosti komisie τ. Ak chceme, aby dohoda bola vytvorená rýchlo, tak je ideálne zvoliť nízku hodnotu T. To však spôsobí nárast τ, kvôli zachovaniu bezpečnosti pri riziku prijatých hlasov od zlomyseľných účastníkov. Preto je dôležité tieto hodnoty správne zvoliť. Pokiaľ označíme počet dobrých účastníkov komisie g a počet zlomyseľných účastníkov b, tak pre zachovanie bezpečnosti a rýchlosti dohody potrebujeme dodržať nasledovné podmienky:

$$T \cdot \tau \geq \frac{1}{2} \cdot g + b \tag{3.1}$$

$$T \cdot \tau < g \tag{3.2}$$

Avšak, pomer dobrých a zlých účastníkov v komisii sa pri každom hlasovaní mení. Z tohto dôvodu nie je možné hodnoty T a τ určiť presne. Tvorcovia Algorandu tvrdia, že pokiaľ chceme zachovať pravdepodobnosť útoku na hodnotách nižších ako $5x10^{-9}$, tak je nutné udržať v sieti viac ako 2/3 účastníkov čestných. Z meraní autorov protokolu vychádza, že pokiaľ chceme udržať spomenutú pravdepodobnosť bezpečnosti na hodnote $5x10^{-9}$, tak pri komisii veľkej 2000 uzlov sa v sieti smie nachádzať nanajvýš 20% zlomyseľných uzlov [45].

Jednou z hrozieb blockchainových sietí je vytvorenie nových blokov zlomyseľnými uzlami. V Algorande by bolo potrebné, aby bol škodlivý blok nielen vytvorený lídrom komisie ale taktiež je nutné, aby ho táto komisia schválila. Pri dostatočnom počte čestných overovateľov tento stav nie je možné dosiahnuť a bolo by nutné, aby v komisii spolupracovala väčšina zlomyseľných uzlov. Tým, že sú členovia komisie volený pri každom hlasovaní nanovo a náhodne je možné potlačiť túto hrozbu na minimum. Tvorcovia Algorandu garantujú, že pri veľkosti komisie τ väčšej ako 1000 uzlov je pravdepodobnosť tohoto útoku menšia ako 2^{-166}, a teda je tento útok teoreticky nemožné zrealizovať [45].

Obrana proti DoS

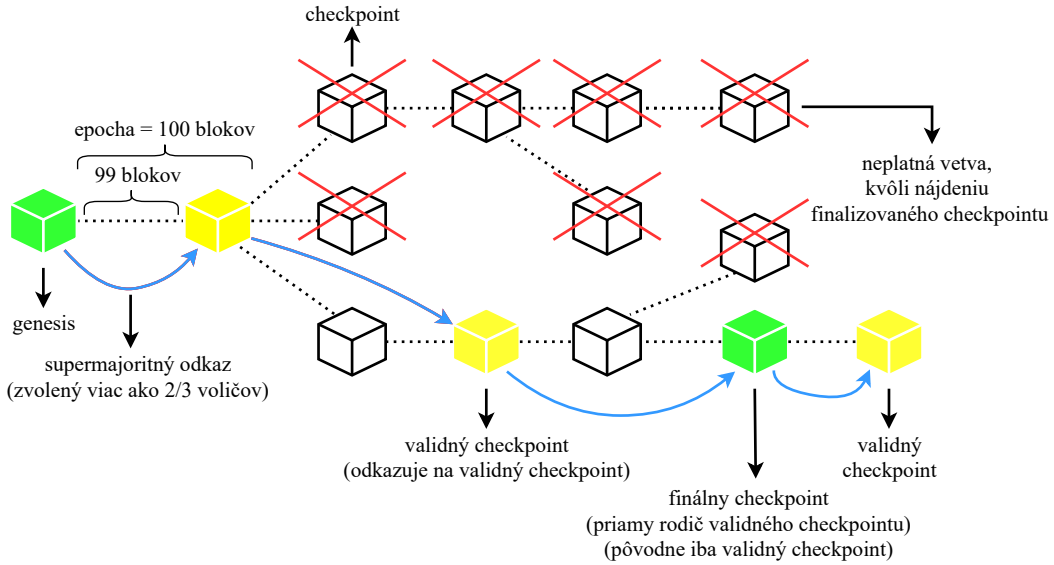
Algorand pri komunikácii uzlov využíva protokol Gossip. Každý užívateľ sa snaží vytvoriť blok z transakcií, o ktorých sa dopočul od iných uzlov. Následne vybraný líder navrhuje tento blok komisii, ktorá rozhodne o jeho schválení. Útočník sa môže snažiť ale zabrániť zaslaniu tohoto bloku. Keďže je blok príliš veľký na to, aby bol zaslaný jedným paketom, tak líder zasiela najskôr potrebné informácie pre jeho správne prijatie. Tento úvodný paket odhalí útočníkovi identitu lídra a je naň možné spustiť útok DoS (denial of service) správami o vymyslených transakciách. Tento útok však musí byť vykonaný skôr ako obeť stihne

odoslať blok. Jeho veľkosť je dostatočne nízka na to, aby ho bolo možné po častiach odoslať do niekoľkých sekúnd skrz spoľahlivé TCP spojenie. Táto nízka veľkosť blokov je sama o sebe dostatočnou prevenciou proti DoS. Aby Algorand hrozbu blokovania lídra ešte viac minimalizoval, tak pridáva povinnosť overiť a podpísať každú správu pred jej odoslaním špeciálnymi algoritmami. Útočník tak nemôže odosielať nezmyselné správy dostatočne rýchlo na vykonanie útoku DoS [23, 45].

3.2 Casper

Počiatky protokolu Casper siahajú už do roku 2015, kedy bola predstavená jeho prvá podoba "Casper the Friendly Ghost", skrátene Casper TFG. Idea bola použiť niektoré z princípov PoW protokolu GHOST (Greedy Heaviest-Observed Sub-Tree) a prispôsobiť ich svetu Proof-of-Stake. Týmto krokom by bolo možné získať prijateľnejšie podmienky fungovania pre dosiahnutie podobných vlastností. V dnešnej dobe poznáme dve rôzne verzie Casper. Jednou je rodina prokolov Casper CBC ("Correct by Construction"), do ktorej spadá už spomínaný Casper TFG a ešte Casper the Friendly Binary Consensus (skr. Casper FBC). Protokoly CBC sú v svojej podstate plne postavené na Proof-of-stake. Druhá verzia Casper Friendly Finality Gadget (Casper FFG) je odlišná a nepokrýva kompletné fungovanie blockchainu. Jedná sa o hybrid medzi PoW a PoS, ktorý je len akousi nadstavbou nad ľubovoľným Proof-of-work algoritmom [13, 91].

3.2.1 Casper Friendly Finality Gadget



Obr. 3.3: Strom kontrolných bodov obsahujúci blok genesis (prvý validný finálny checkpoint), validné body (žlté), finálne body (zelené) a supermajoritné odkazy (modré). Bodkované čiary medzi kontrolnými bodmi predstavujú 99 blokov vytvorených PoW algoritmom. Po nájdení finálneho checkpointu sa ostatné vetvy považujú za neplatné [16, 62].

Verzia **Casper FFG** slúži primárne k zlepšieniu vlastností algoritmov PoW. Tie používajú rôzne mechanizmy na vytváranie blokov, pričom nadstavba FFG označí každý stý blok v reťazi ako kontrolný bod. Overovatelia algoritmu FFG hlasujú o validite jednotlivých

kontrolných bodov a snažia sa vytvoriť jednu reťaz. Broadcastovaný hlas obsahuje zdrojový bod s, cieľový bod t, ich výšku $h(s)$ a $h(t)$, udávanú v počte kontrolných bodoch v danej reťazi, a signatúru súkromného kľúča overovateľa. Žiadny overovateľ nesmie publikovať dva hlasy, ktoré by splnili niektorú z nasledujúcich podmienok:

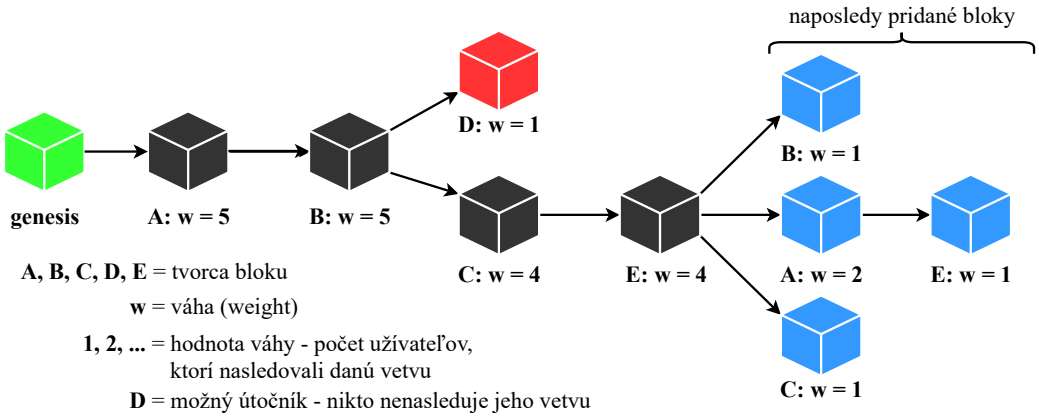
$$h(t_1) = h(t_2) \tag{3.3}$$

$$h(s_1) < h(s_2) < h(t_2) < h(t_1) \tag{3.4}$$

Pokiaľ aspoň 2/3 voličov zvolilo rovnaký zdroj a cieľ, tak sa medzi týmito vytvára **supermajoritný odkaz**. Každý kontrolný bod, na ktorý existuje supermajoritný odkaz z iného bodu, ktorý je označený ako validný, je tiež **validný** (justified). Prvým takýmto blokom je samozrejme blok genesis. Keď sa podarí vytvoriť kontrolný bod, na ktorý odkazuje supermajoritný odkaz práve z validného rodičovského kontrolného bodu, tak je tento kontrolný bod označený ako **finálny** (finalized). Cieľom je vytvoriť reťaz, v ktorej je väčšina kontrolných bodov finálna. Počet finálnych blokov určuje dynastiu, pomocou ktorej sa sleduje príchod a odchod overovateľa z komisie. Pomocou dynastie je umožnené dynamicky meniť členov komisie a určovať pravidlá na kontrolu toho, ktorý overovateľ môže schvaľovať jednotlivé kontrolné bloky. V prípade porušenia pravidiel prichádza uzol o všetky aktíva vložené do systému. Pre lepšie pochopenie môžeme vidieť strom kontrolných bodov na obrázku 3.3 [16, 62].

3.2.2 Casper Correct by Construction

Casper CBC je o trochu komplexnejšia v porovnaní s verziou FFG. Pri rozširovaní ledgera sa v niektorých prípadoch vytvárajú vetvy. Pokiaľ aspoň 3/4 účastníkov nenasleduje správnu vetvu, tak môže dôjsť k rôznym komplikáciám a taktiež k úspešným útokom. Rozhodnutie, do ktorej z vetiev je vhodné pridať nový blok, sa v klasických PoW blockchainoch určuje podľa ich dĺžky. Protokoly Casper CBC využívajú pri voľbe vetvy pravidlo **LMD Ghost** ("latest message driven"). Namiesto dĺžky vetvy rozhoduje to, koľko účastníkov ju považuje za validnú. Pri vytváraní nových blokov sa sledujú naposledy vytvorené bloky jednotlivými užívateľmi. Tí získavajú reputáciu podľa počtu ďalších účastníkov, ktorí nadviazali svojimi blokmi na tie ich a touto cestou ich podporilo. Tento princíp môžeme vidieť na obrázku 3.4. Každý z blokov má pod sebou uvedené dva údaje, publikovateľa a váhu w. Váha udáva koľko z účastníkov publikujúcich ďalšie bloky podporilo tento blok a nasledovalo jeho vetvu.



Obr. 3.4: Princíp ohodnocovania blokov a výberu vetvy podľa pravidla LMD Ghost [13, 17].

Ako môžeme vidieť, tak za predpokladu, že užívatelia A, B, C a E sú čestní, tak užívateľ D je s najväčšou pravdepodobnosťou zlomyseľný, nakoľko jeho vetvu nikto z nich nepodporil. Táto metóda je vhodná na využitie v sieťach s vyššou latenciou, kedy pri publikovaní často dochádza k forku. Pravidlo najdlhšej vetvy nie je práve tou správnou voľbou práve kvôli možnosti publikovania niekoľkých uzlov minoritnou časťou účastníkov, ktorí nehrajú podľa pravidiel [13, 17].

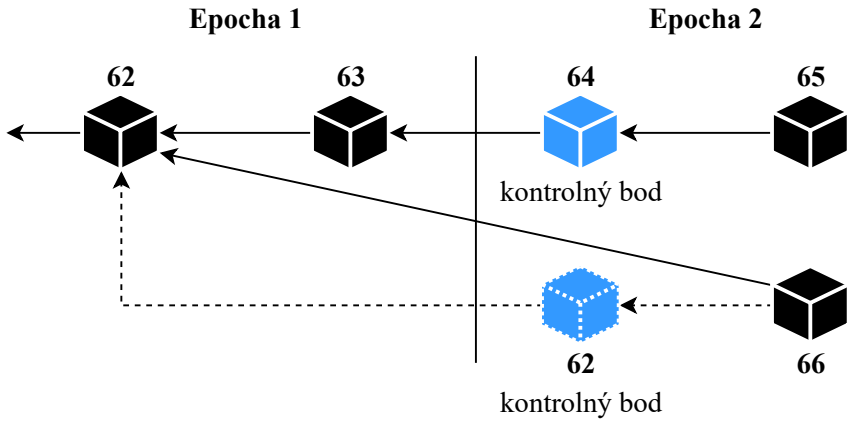
3.2.3 Útoky na Casper

Rovnako ako je to pri iných protokolov, aj Casper sa musí zaoberať problematikou rôznych útokov. Medzi najčastejšie patria útoky **LRR** (Long Range Revision attack) a riziko zlyhania množstva uzlov. V LRR útoku je hrozbou komisia, ktorá v minulosti vlastnila viac ako 2/3 aktív, no v priebehu doby členovia komisie stihli všetky svoje aktíva zo siete vybrať. Takto môžu finalizovať konfliktné kontrolné bloky podľa seba a v prípade porušenia pravidiel nemajú o čo prísť. Problém LRR je primárne vo verzii FFG, kde účastníci pri prihlásení do systému považujú ledger za stabilný od posledného finálneho kontrolného bodu. Pohľad na neho je podobný ako u genesis bloku, a teda nie je nutné mať povedomie o blokoch pred týmto kontrolným bodom. Prevencia pred LRR je poskytnutie užívateľom kompletného pohľadu na ledger v istých časových intervaloch, napríklad raz za dva až štyri týždne [16, 30].

V istom katastrofickom scenári môže nastať, že viac ako tretina overovateľov zlyhá v rovnaký čas. Zlyhanie môže byť spôsobené rôznymi príčinami od straty internetového pripojenia až po cielený mohutný útok DoS (denial of service). Následkom je znemožnenie vytvárania supermajoritných odkazov a finalizovania kontrolných bodov. Ideálnym riešením môže byť zníženie množstva aktív v depozite týmto overovateľom natoľko, aby bolo možné vytvoriť supermajoritnú väčšinu zo zvyšných uzlov. Zníženie aktív môže prebehnúť napríklad akýmsi zmrazením účtov, kedy sa ich aktíva nastavia na nulu a po niekoľkých dňoch sa znovu obnovia na predchádzajúcu hodnotu. O správnom riešení tohoto problému sa však stále uvažuje [16].

3.3 Gasper

Tím autorov Casper protokolu chcel docieliť ešte niečo viac. Aby dosiahli čo najlepšie výsledky, tak skombinovali prístupy Casper FFG a CBC. Týmto vznikol protokol Gasper, ktorý je momentálne implementovaný v aktuálnej verzii kryptomeny Ethereum. Tvorba blockchainu prebieha pomocou čistého Proof-of-Stake prístupu, ktorý je zabezpečený hybridným LMD pravidlom (HLMD). Pre každý blok je náhodne vybraná komisia. Gasper nedefinuje správny prístup pre zvolenie jednotlivých členov a ponecháva túto voľbu plne na konkrétnej implementácii. Jeden z členov komisie vytvára nový blok na základe hlasov od ostatných členov komisie. HLMD hlasy obsahujú informáciu o tom, ktorý blok je podľa daného člena najvhodnejším rodičom nového bloku. Okrem tejto informácie, stavanej na klasickom LMD z verzie Casper CBC, hlas obsahuje aj informáciu o FFG odkaze medzi kontrolnými bodmi, čím sa účastníci uisťujú o rovnakej voľbe kontrolných bodov. Na začiatku každej epochy sú prijaté hlasy prepočítané a účastníci označujú bloky za validné a finalizované pomocou rovnakých princípov aké sú definované v protokole Casper FFG [17].



Obr. 3.5: Spôsob tvorby kontrolných bodov v protokole Gasper. Bloky 64 a 62 sú označené ako kontrolné body. Blok 64 je práve 64. blokom, ktorý bol pridaný do blockchainu. Blok 62 je kontrolný bod v epoche 2 pre vetvu bloku 66 [17].

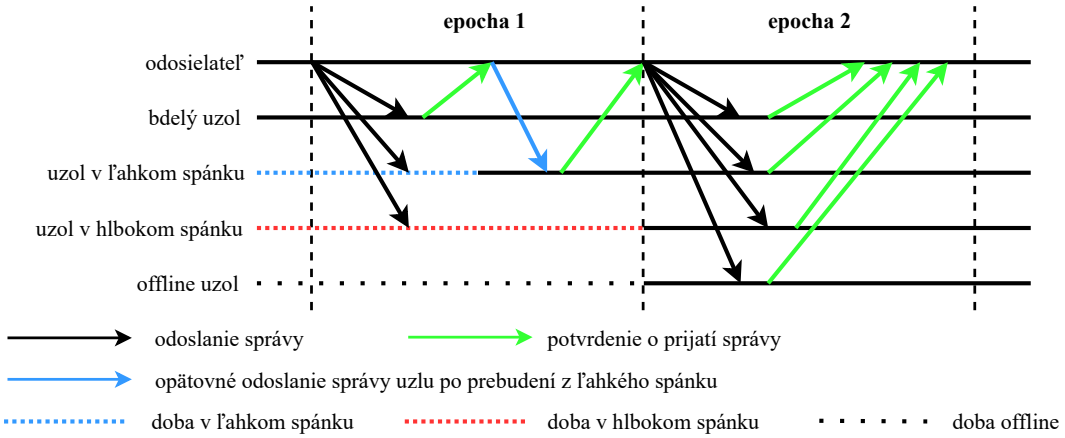
EBB (Epoch Boundary Block)

Na rozdiel od štandardného prístupu, aký má Casper FFG, prináša Gasper trochu iný pohľad na tvorbu kontrolných bodov. Blok sa nestane kontrolným bodom na základe svojej výšky, ale na základe počtu vytvorených blokov. Napríklad môžeme určiť, že každý 64. blok bude novým kontrolným bodom. Ich výška teda nie je rozhodujúca. Gasper vytvorený kontrolný bod nazýva EBB (epoch boundary block). Je to preto, lebo existuje aj iný spôsob, ktorým sa blok môže stať kontrolným bodom. Majme napríklad epochu určenú každým 64. blokom. V epoche číslo 2 dôjde k vytvoreniu bloku 66 v inej vetve akú nasleduje blok 64, označený ako kontrolný blok. Rodičom bloku 66 je blok 62, a teda tento blok (62) je preto taktiež označený za kontrolný bod. Občas preto môže dôjsť k stavu, že jeden blok je kontrolným bodom v niekoľkých epochách. Pre lepšie pochopenie je dostupný obrázok 3.5 [17].

3.4 Snow White

Ďalším zo zaujímavých protokolov Proof-of-Stake je **Snow White**, patriaci do skupiny permissioned algoritmov. Primárnymi cieľmi, ktoré sa snaží dosiahnuť sú robustnosť a formálne dokázateľná bezpečnosť. Jeho autori nadviazali na svoju predchádzajúcu prácu a vyvinuli ho z protokolu **Sleepy**. Počíta s tým, že v niektorých prípadoch môže nastať stav, v ktorom sa istí užívatelia nepodieľajú na fungovaní blockchainu. Môžu sa odpojiť, prípadne zostať neaktívni po potrebnú dobu a neskôr sa znovu zapojiť do práce na budovaní ledgera bez toho, aby boli považovaní za zlomyseľných. Sleepy preto rozdeľoval účastníkov do dvoch skupín: bdelé a spiace uzly. Po vzniku protokolu Snow White viacero protokolov vylepšilo niektoré so svojich vlastností. Patria medzi ne protokoly ako Algorand či Casper [11, 27].

Základný exekučný model Sleepy bol rozšírený o niekoľko nových funkcií. Pokiaľ účastník spí, tak protokol rozlišuje *ľahký spánok* a *hlboký spánok* podľa jeho dĺžky, aby bolo možné lepšie rozhodovať o prístupe k nemu. Taktiež je umožnené užívateľom vstupovať do systému dynamicky aj po jeho spustení. Dôležitá podmienka na fungovanie tohoto protokolu je dôkladná synchronizácia času medzi všetkými uzlami. Počíta sa samozrejme s možným oneskorením správ a je stanovená hranica, do ktorej musí byť správa prijatá všetkými čest-



Obr. 3.6: Doručovanie správ uzlom v ľahkom a hlbokom spánku podľa protokolu Snow White. Uzlom v ľahkom spánku sú správy doručené po ich prebudení. K uzlom v hlbokom spánku sa pristupuje rovnako ako k offline uzlom, čo znamená, že sú zapojení do procesu dosahovania konsenzu až v ďalšej epoche [27].

nými užívateľmi, ktoré práve nespia. Správy, ktoré sú odoslané uzlom v ľahkom spánku sú doručené po ich prebudení. Uzly v hlbokom (dlhotrvajúcom) spánku však po prebudení správy nedostávajú a je k nim pristupované ako ku novým uzlom, ktoré vstúpili do systému. Hranica, ktorá rozdeľuje ľahký a hlboký spánok, nie je pevne určená protokolom a záleží od konkrétnej implementácie. Je potrebné zmieniť, že skorumpované uzly nikdy nespia. Pri vstupe uzla do systému je prostredie informované o tejto udalosti inicializačnou správou. Ak ale útočník vloží do systému skorumpovaný uzol, tak nemusí prostredie informovať, čím sa narúša bezpečnosť [27].

Tvorba ledgera

Ledger je rozširovaný podobne ako v protokole Algorand. Spomedzi aktívnych uzlov sa vyberá náhodne komisia, ktorá si zvolí svojho lídra. Líder má nárok na pridanie nového bloku do blockchainu. Každý z členov má silu hlasu podľa miery stávky, ktorú do systému vložil. Výber členov komisie záleží na konkrétnej implementácii aplikáciou, ktorá protokol Snow White využíva. Pokiaľ bude väčšina vybraných účastníkov čestná, tak sa predpokladá, že je tento prístup bezpečný.

Snow White samozrejme má istý postup, ktorý je treba dodržať. Je ním **Strawmanova schéma**, podľa ktorej sa nová komisia vyberá pre jednotlivé epochy. Každá epocha e, vznikajúca v perióde T_{epoch}, je definovaná časom začiatku a konca $[start(e),\ end(e)]$. Výber komisie je vykonaný funkciou $extractpks(chain[:l])$. Parameter $chain[:l]$ je reťaz obsahujúca blok nachádzajúci sa na indexe l. Index je hodnota posledného bloku, ktorého timestamp je menší ako $start(e) - 2\omega$. Hodnota parametra 2ω musí byť dostatočne veľká na to, aby pre všetky aktívne uzly bol blok na indexe l v rovnakej reťazi [12].

Hrozby a obrana proti nim

V niektorých prípadoch je možné predpovedať spôsob voľby členov. Ak útočník zistí aké verejné kľúče budú mať členovia vybratej komisie, tak môže vložiť do systému uzol s potrebne nastaveným kľúčom. Ten by bol zvolený namiesto čestného uzla a získal by právo zapojiť sa

do tvorby ledgera. Snow White preto ustanovil pravidlo, že do výberu komisie sa môžu zapojiť len uzly, ktoré vstúpili do systému pred začatím novej epochy. Uzly vstupujúce neskôr dostanú svoju príležitosť až v ďalšej epoche. Tento problém sa v pôvodnej verzii Sleepy nevyskytuje, pretože kvôli chýbajúcej dynamickej zložke všetky uzly vstúpili do systému pred jeho spustením [12].

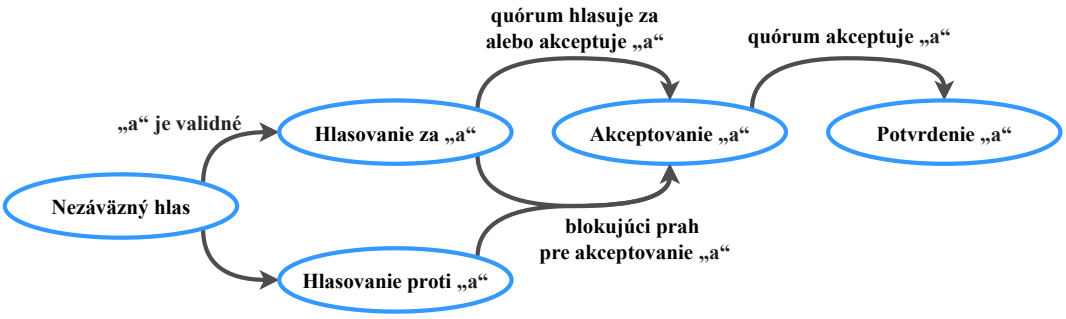
Snow White dokáže byť bezpečný za predpokladu, že nedochádza k príliš častým transakciám. Tento predpoklad je prevenciou napríklad pred možnosťou dvojitého utrácania aktív zlomyseľnými uzlami. Predpokladajme, že v istom časovom okne je limitovaná čiastka, ktorá sa môže presúvať medzi uzlami. Skorumpované uzly tak nemajú možnosť vykonať dostatočné transakcie na vytvorenie škodlivej vetvy, ktorá by mohla viesť k ohrozeniu systému. Dôvodom je skutočnosť, že stále je dostatok aktívnych čestných uzlov, ktoré so svojimi aktívami v daný moment nehýbu a zaisťujú potrebnú stabilitu systému [27].

Okrem protokolu Sleepy sa autori inšpirovali aj protokolom **Fruitchains**. Popri ťažení klasických blokov sa ťaží taktiež ovocie a odmena za ťaženie je udeľovaná len za ovocnú zložku a nie za vytváranie nových blokov. Ide o spôsob ochrany proti nárastu podielu aktív, ktoré vlastnia skorumpované uzly na hodnotu vyššiu ako parameter férovosti ϵ. Táto prevencia je samozrejme účinná jedine za predpokladu, že je v systéme viac aktívnych čestných uzlov ako skorumpovaných [12].

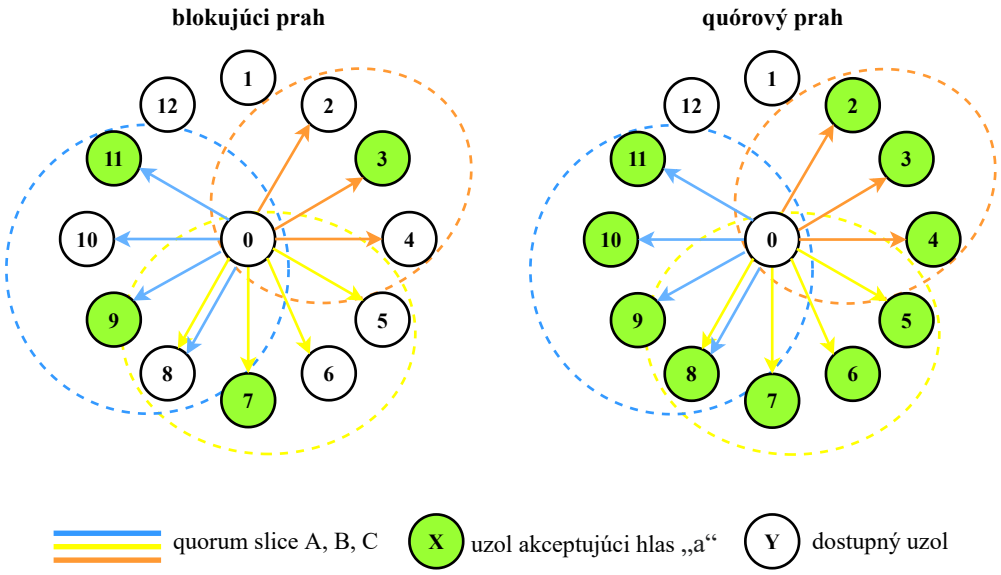
3.5 Stellar

Podobne, ako je to u predchádzajúcich algoritmov aj Stellar používa byzantskú dohodu. Avšak, narozdiel od klasickej byzantskej dohody, **SCP** (Stellar Consensus Protocol) používa variantu **FBA** (Federative Byzantine Agreement). Hlavným cieľom SCP je dosiahnutie bezpečnej decentralizácie s nízkou latenciou a možnosťou rozhodovať o dôvere v jednotlivých účastníkov. Jednou z dôležitých vlastností, ktoré prináša FBA, je otvorenosť komisie. To znamená, že uzly sa samé rozhodnú, ktorým uzlom dôverujú. Každý uzol si vytvára vlastný zoznam, nazývaný *quorum slice*. Zoznamy sú broadcastované a ich spojením vzniká *quorum*. Uzly tvoriace quorum následne schvaľujú nový blok pomocou hlasovacích lístkov (obrázok 3.7).

Uzol akceptuje hlasovací lístok v dvoch možných situáciách. Dosiahnutie **quorového prahu** (quorum threshold) alebo **blokujúceho prahu** (blocking threshold). Blokujúci prah je dosiahnutý vtedy, keď každý slice, do ktorého uzol patrí, obsahuje aspoň jeden iný uzol, ktorý schválil daný hlas. Pri blokujúcom prahu nezáleží na tom, či náš uzol schválil hlas



Obr. 3.7: Priebeh schvaľovania hlasovacieho lístka pomocou federatívnej byzantskej dohody FBA [59].



Obr. 3.8: Dosiahnutie blokujúceho prahu (vľavo) a quórového prahu (vpravo) uzlom 0. Uzly 1 a 12 nepatria do quóra uzla 0 nakoľko nepatria ani do jedného z quorum slice A, B alebo C [94].

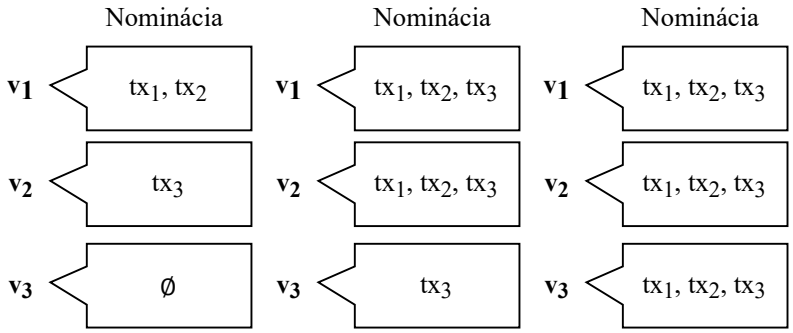
alebo nie. Quorový prah je dosiahnutý, ak každý uzol v quóre, do ktorého patrí ten náš, schválil daný hlas. Rozdiel medzi týmito prahmi môžeme vidieť na obrázku 3.8.

Pre potvrdenie akceptovania hlasu je nutné dosiahnuť ďalší quórový prah. Hlasovanie je považované za úspešné ak jeden alebo viacero uzlov neakceptuje hlas za hodnotu pre daný slot. Táto vlastnosť je dôsledkom uprednostňovania *bezpečnosti* (safety) voči *živosti* (liveness) FBA konsenzu, čím predchádza vzniku zbytočných vetiev. O dosiahnutie živosti systému, teda rýchle schvaľovanie nových hodnôt, sa taktiež snaží, avšak nie je to považované za prioritu. Uzly, ktoré sa snažia docieliť aj živosť aj bezpečnosť systému bez prejavu zlomyseľného správania sú považované za *korektné uzly*. S pomocou FBA vytvára len jednu reťaz bez možnosti ďalších vetiev. Tým predchádza rôznym útokom ako napr. sebecké ťaženie, LRA alebo SRA [59].

Fázy konsenzu

Stellar pracuje v dvoch fázach, **nominácia** kandidátnych hodnôt pre jednotlivé sloty a hlasovanie za finálne hodnoty. Prvá fáza je z bezpečnostných dôvodov synchronizovaná, v druhej to už nie je vyžadované. Nominácia prebieha iteratívnym zdieľaním hlasov medzi uzlami. Cieľom je zhodnúť sa na množine hodnôt, z ktorých sa bude vyberať výsledná. Nominácia beží až dokým táto množina nebude konvergovať. Po dosiahnutí konvergencie sa z množiny vytvorí jedna zložená hodnota. Príkladom môže byť súbor zjednotenia transakcií, o ktorých jednotlivé uzly vedia a označenie súboru najvyššou hodnotou časovej značky z tohoto setu. Každá z hodnôt musí spĺňať isté pravidlá, ako napríklad to, že čas transakcie nesmie byť v budúcnosti. Konvergencia je dosiahnutá vďaka tomu, že neporušené uzly sa snažia primárne o schválenie existujúcich nominovaných hodnôt namiesto pridávania nových. Priblíženie nominačného procesu je zobrazené na obrázku 3.9 [59].

Druhá fáza protokolu Stellar je hlasovanie za finálne hodnoty, označovaná ako fáza **SCP ballot**. Uzly sa do druhej fázy zapájajú hneď po vytvorení zloženej hodnotu x. Každý



Obr. 3.9: Proces nominácie hodnôt medzi uzlami v_1, v_2 a v_3 s dosiahnutím konvergencie v troch krokoch. Vytvorenie zloženej nominovanej hodnoty $x = \{tx_1, tx_2, tx_3\}$ [8].

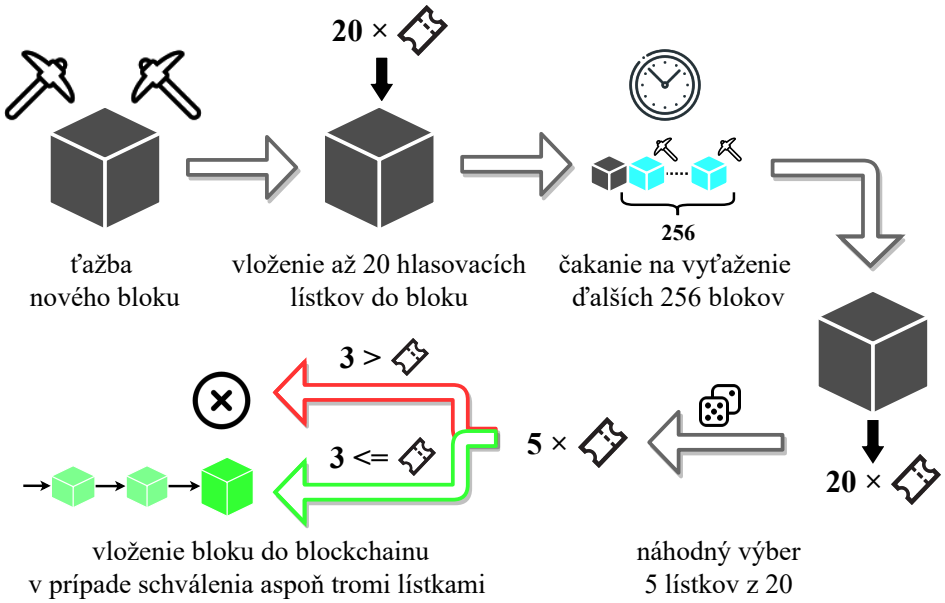
hlasovací lístok b obsahuje zloženú hodnotu x a poradovú hodnotu n. Uzly hlasujú o prijatí alebo odmietnutí lístka b. Pokiaľ sú uzly čestné, tak sa všetky zhodnú na lístku s rovnakou hodnotou. Proces schvaľovania predchádza príprava hlasovacích lístkov jednotlivými uzlami. Lístok je v pripravenom stave ak neexistuje zamietnutý lístok s menším alebo rovnakým poradovým číslom obsahujúci rovnakú hodnotu x. Hneď, ako všetky uzly potvrdia, že je tento hlasovací lístok pripravený, tak čestný uzol hlasuje za prechod tohoto lístka b do stavu *commit*. Po potvrdení stavu commit všetkými uzlami s ním môže čestný uzol pracovať a vytvoriť z hodnôt na tomto hlasovacom lístku nový blok. V prípade, že sa hlasovanie nezdarí, napríklad z'dôvodu nereagujúcich uzlov, tak uzol zvýši poradové číslo hlasovacieho lístka a skúsi znovu spustiť hlasovanie s novým vyšším lístkom [59].

Nedostatky protokolu

Stellar je protokol, ktorý síce je považovaný za bezpečný (safety) ale jeho výkon (liveness) je slabší. Prvý z ďalších problémov je stav, v ktorom všetky uzly naraz opustia systém a už sa nevrátia. Systém by pri opätovnom spustení potreboval centrálnu koordináciu spolu s ľudským zásahom. Ďalším z problémov je zmena konfiguračných parametrov. Jednou z možností je vytvoriť špeciálne nominačné správy, ktoré by okrem klasických hodnôt obsahovali aj nové parametre. Tento prístup ale umožňuje zlomyseľným uzlom ľubovoľne upravovať parametre. Korektný spôsob pre úpravu konfigurácie medzi uzlami v SCP zatiaľ nebol nájdený. Jedným z významných problémov je kaskádové zlyhanie systému, ktorý môže docieliť v kompletné zlyhanie celého SCP, ak dva, alebo viac uzlov riadených protokolom Stellar sú odstránené [55, 59].

3.6 Decred

Kryptomenový protokol Decred bol vytvorený s cieľom minimalizovať rozdvojenie blockchainu do formy hard fork. Spôsob jeho fungovania bol inšpirovaný protokolom Bitcoin, z ktorého prevzal niekoľko zaujímavých vlastností. Patrí medzi ne decentralizácia v podobe peer-to-peer siete a taktiež rýchla odozva na útoky, či žiadosti komunity o nové funkcie. Na rozdiel od Bitcoinu je postavený na hybridnom dizajne, kde kombinuje Proof-of-Work a Proof-of-Stake. Tento dizajn je často označovaný ako *Proof-of-Activity*. Používa nový hlasovací systém, v ktorom každý hlasovací lístok musí byť zakúpený a je platný len po určitú dobu. Doba vytvorenia nového bloku je približne 5 minút [41].



Obr. 3.10: Priebeh schvaľovania vyťaženého bloku proof-of-stake voličmi [53].

Po vyťažení nového bloku minermi štandardným PoW spôsobom získa tento užívateľ privilégium do neho vložiť informácie o transakciách. Tento blok je pred pridaním do ledgera nutné schváliť PoS overovateľmi. O schválení rozhodujú pomocou hlasovacích lístkov. Každý blok môže obsahovať 20 lístkov, ktoré môžu byť zvolené až po vyťažení tohoto bloku a vložení ďalších 256 blokov do ledgera. Následne sa z platných lístkov vyberá pseudonáhodným spôsobom 5 lístkov. Ak aspoň tri z nich schvália daný blok, tak je tento blok vložený do ledgera. Tento postup je zobrazený na obrázku 3.10. Pravdepodobnosť zvolenia bloku v dobe 28 dní je približne 50 % a najviac sa môže predĺžiť až na 142 dní. Odmeny za zvolenie bloku sú rozdelené medzi minera (60 % odmeny), PoS overovateľov (30 % odmeny) a zvyšok (10 % odmeny) ide do fondu na podporu projektu Decred. Namiesto pôvodnej hashovacej funkcie SHA256 používa Decred funkciu BLAKE256, ktorá je značne rýchlejšia na mikroarchitektúrach x86-64 a taktiež bezpečnejšia [19, 41, 53].

Rozšírenia

Pre zvýšenie rýchlosti a škálovateľnosti implementuje Decred sieť zvanú **Lighning Network**. Ide o mimoblockchainový spôsob transakcií medzi dvomi vzájomne si dôverujúcimi uzlami. Tie si medzi sebou vytvoria tretí účet, na ktorý každý z nich vloží rovnako vysoký obnos aktív. V prípade, že chcú vykonať transakciu medzi sebou, tak stačí ak jeden z uzlov pošle druhému správu s novým rozdelením obnosu, ktorý je na tomto účte. Nie je nutné čakať na vloženie nového bloku s touto transakciou do blockchainového ledgera. Platobný systém využívajúci sieť Lightning implementuje okrem Decredu aj niekoľko ďalších systémov, vrátane Bitcoinu [41].

Decred umožňuje užívateľom medzimenové obchodovanie kryptomenových aktív výmenou blokov priamo v ledgeroch, označované ako **on-chain atomic swap**. Hlavná podmienka je, aby v oboch blockchainoch bola používaná rovnaká hashovacia funkcia. Taktiež je potrebné, aby druhý systém tiež kontroloval správne poradie časových značiek rovnako

ako to je implementované v Decrede. Táto funkcionalita je vhodná pre väčšie transakcie, v ktorých nezáleží na jej dobe trvania a ktorá nie je vykonávaná príliš často [89].

Okrem klasického hlasovania pomocou uzlov ponúka Decred aj hlasovanie užívateľov mimo blockchain, systémom nazývaným **Politeia**. Pri vytvorení hlasu v tomto systéme sa návrh musí schváliť. Držitelia aktívnych hlasovacích lístkov rozhodujú o prijatí tohoto návrhu priamo v ich Decred peňaženkách [19].

3.7 Teoretické porovnanie

Každý zo spomenutých protokolov má svoje výhody a taktiež nevýhody. V tabuľke 3.2 je zobrazené teoretické zhrnutie ich vlastností. Je potrebné brať do úvahy skutočnosť, že teória je niekedy iná ako vlastnosti, ktoré sme schopný dosiahnuť pri reálnom behu. Preto je dôležité taktiež vykonať simuláciu blockchainových algoritmov, v ktorej je možné porovnať reálne vlastnosti s teoretickými. Nakoľko je protokol Gasper kombináciou protokolov Casper FFG a CBC, tak sú jeho vlastnosti zhodné s touto rodinou protokolov.

Odolnosť voči útokom

	Algorand	Casper / Gasper	Snow White	Stellar	Decred
Nothing at Stake	Áno [15]	Áno [15, 25]	Áno [15]	Áno [8]	Áno [69]
Fake Stake	—	—	—	Áno [85]	Nie [47]
51 % útok	—	Nie [83]	—	Áno [85]	Áno [47]
Dvojnásobné utrácanie	Áno [45]	Nie [52]	Áno [15]	Áno [8]	Áno [38]
Sybil	Áno [55]	Áno [35, 73]	Nie [12]	Áno [55]	Áno [68]
DoS	Áno [50]	—	—	Nie [55]	Nie [64]
Sebecké ťaženie	Áno	—	Áno [15]	Áno [8]	Áno [28]
LRA	Áno [42]	Áno [25]	Áno [42]	Áno [8]	Áno [49]
SRA	Áno [42]	—	—	Áno [8]	Áno [49]
Grinding attack	Áno [50]	—	Áno [12]	Áno [8]	Áno [2]

Tabuľka 3.1: Odolnosť algoritmov voči útokom. Hodnota "—" je použitá v prípade, keď odpoveď na otázku odolnosti nebola nájdená.

Na protokoly Proof-of-Stake existuje niekoľko typov útokov. Odolnosť nami opísaných protokolov voči týmto útokom je uvedená v tabuľke 3.1. Princíp jednotlivých útokov je vysvetlený nižšie.

- **Nothing at Stake** je útok, ktorý sme si už raz opísali. Pre pripomenutie sa jedná o možnosť prispieť do tvorby viacerých vetiev zároveň. Útočníci využívajú vlastnosť PoS algoritmov, v ktorých nie je nutné vyvinúť pre budovanie ledgera žiadne úsilie. Účastníci tak získajú odmenu v každom prípade, nech je blok pridaný do ktorejkoľvek vetvy. Vyšší podiel útokov nothing at stake, vrátane neúspešných, vedie k nárastu tvorby vetiev, čo výrazne spomaľuje proces finalizácie [50].

Riešení tohto problému je niekoľko. Algoritmus môže byť navrhnutý tak, že pri každej stávke účastníka sa jeho aktíva zmrazia na určitú dobu a nebude môcť hlasovať za bloky na rovnakej úrovni. Ďalším prístupom je použitie rôznych variant byzantskej dohody BFT, ktoré takmer úplne potlačujú možnosť vetvenia ledgera [50].

- **Fake Stake** je útok, ktorý sa prejavil už u niekoľkých existujúcich blockchainových sietí. Cieľom je pomocou minimálnej stávky odstaviť, prípadne prebrať kontrolu nad iným uzlom tým, že zahltíme jeho dostupné zdroje (disk, RAM, ...). Tento útok môže vyústiť do útoku 51 %. Pôvodne bol objavený v protokoloch Qtum, Particl, Navcoin, HTMLcoin a Emercoin. Každý z nich je postavený na princípe podobnom Bitcoinu, u ktorého uzol najskôr prijme hlavičky blokov. Až neskôr prijíma telo s transakciami a na základe ich validity blok môže rozhodnúť o zahodení bloku. Overenie hlavičky je až na jednu položku rýchle. Touto položkou je stávka voliča, ktorá je síce uložená v hlavičke ale musí byť overená spoločným úsilím viacerých blokov. Preto, po čiastočnom overení hlavičky si uzol ukladá hlavičky bloku do štruktúry *mapBlockIndex*. V prípade zahlcovania uzla hlavičkami blokov môže byť výsledkom vyčerpanie jeho pamäte RAM [56].

- **Útok 51 %** predstavuje možnosť ovládnutia blockchainového systému pri vlastníctve viac ako polovice jeho aktív. Užívateľ má tak vyššiu silu ako akýkoľvek iný účastník. Tento útok sa primárne vyskytuje v Proof-of-Work algoritmoch, no pri využití napr. Fake Stake útoku je ho možné docieliť aj v PoS. V nich je totiž náročné získať potrebné množstvo aktív na dosiahnutie väčšinového hlasu. Avšak systémy založené na protokoloch Proof-of-Stake stavajú týchto útočníkov do nevýhody. V prípade, že by totiž volili neplatné bloky, tak znehodnocujú danú kryptomenu. Oplatí sa preto hlasovať za platné bloky bez tvorenia vetiev. Za jeho silný hlas obdrží spravodlivú odmenu, ktorá si naďalej udrží svoju cenu. V niektorých protokoloch ale môže predsa viesť útok 51 % k úspešnému útoku. Pri vytvorení vetiev v blockchaine môže totiž dosiahnuť stav dvojnásobného utrácania aktív [39, 56].

- **Dvojnásobné utrácanie** je stav, v ktorom uzol utráca aktíva, ktoré minul už v inej vetve. Najčastejšou príčinou tohoto problému je dlhá doba na dosiahnutie konečnosti (finalizácie bloku). Ak by totiž účastník nevedel o tom, že bol vytvorený konečný blok, tak by sa mohol pokúsiť vytvoriť druhý blok obsahujúci rovnaké transakcie, ale v druhej vetve. O nej pri pomalej finalizácii blokov tento účastník nemusí vedieť. Útočník to samozrejme môže zneužiť a použiť rovnaké aktíva na rôzne účely. V podstate takto nakupuje za financie, ktoré nemá. Prevenciou môže byť pridanie čakacej doby pri tvorbe nových blokov, ktorá zaistí, že sa o konečnom bloku dozvedia všetci účastníci pred tým než začnú tvoriť ďalší blok [50].

- **Sybil** je útok, v ktorom útočník vytvorí značné množstvo uzlov. S ich pomocou dokáže získať potrebnú moc v systéme na usmerňovanie tvorby blockchainu podľa jeho vlastnej vôle. Základom je použitie falošnej identity pri vstupe do blockchainu. Následne ním vytvorené uzly môžu zneužiť tento stav dvomi spôsobmi. Prvým je bránenie prenosu správ o blokoch, hlasoch alebo transakciách medzi čestnými uzlami v sieti. Okrem blokovania môžu dosiahnuť aj lepších výsledkov pri útokoch DoS (denial of service). Druhým spôsobom je zvýšenie pravdepodobnosti byť zvolený do komisie, a tak vytvoriť perfektné prostredie pre útoky 51 % [18, 50].

Obranou proti útokom Sybil je použitie napríklad už spomínaných VRF (verified random functions), ktoré vyberajú členov komisie úplne náhodným spôsobom. V prípade protokolov využívajúcich reputáciu uzlov je možné v pravidelných intervaloch reputáciu vynulovať. Útočiace uzly si tak budú musieť reputáciu znovu budovať čestným spôsobom pred tým, ako budú môcť silnú reputáciu použiť k útoku [18].

- **DoS (Denial of Service)** je označovaný stav, v ktorom útočník odoprie uzlom byť aktívnymi. Dosiahne toho napríklad zasielaním enormného množstva správ o transakciách, ktoré obeť nebude stíhať spracovávať dostatočne rýchlo na to, aby mohla vykonávať aj inú činnosť. Útočník môže odstaviť aj celú sieť v prípade aj celkom lacným spôsobom. Dosiahol by to vytváraním minimálnych transakcií dostatočnou rýchlosťou (tzv. penny-flooding). Preto niektoré z protokolov zaviedli minimálnu hodnotu, ktorú musí transakcia mať. Je možné použiť aj ďalšie sieťové zariadenia na filtrovanie dát, ktoré by kontrolovali prevádzku medzi uzlami a v prípade identifikácie DoS útoku by túto prevádzku presmerovali [50].

- **Sebecké ťaženie** je možnosť budovať vlastnú tajnú vetvu útočníkom. Jej publikovanie je až v stave, keď je táto vetva najdlhšia, a teda všetky odmeny za budovanie blokov sú pridelené útočníkovi. Zároveň ďalšie nové bloky budú pridávané do tejto vetvy. Čestné uzly tak na základe pravidla najdlhšej vetvy zahodia celú svoju prácu, ktorú vyvinuli na vytvorenie pôvodnej blockchainovej vetvy [50].

  Riešením je použitie pravidiel, ktoré nevyberajú z vetiev na základe ich dĺžky ale na základe ich kvality. Ak sú napríklad v dvoch vetvách takmer rovnaké bloky, tak sa môže nasledovať vetva, ktorej blok má nižšiu hodnotu hashu v svojej hlavičke. Ďalším z riešení, ktoré je možné uplatniť v protokoloch Proof-of-Work, je ich kombinácia s protokolmi BFT (byzantine fault tolerant). Takýmto riešením je napríklad už spomínaný protokol Casper FFG [50].

- **LRA (Long Range Attack)** je podobný útoku 51 %. Útočník, prípadne skupina útočiacich uzlov, siaha ďaleko do histórie (tisíce až milióny blokov), kde mal potrebnú väčšinu aktív. Tu začne budovať svoju vlastnú vetvu, ktorou prichádza k chcenému zisku. Môže taktiež využiť súkromné kľúče užívateľov, ktorí mali v danej dobe veľkú silu hlasu avšak neskôr sa z blockchainovej siete odpojili. Znalosť ich súkromného kľúča mohla byť zistená aj celkom jednoducho a to jeho kúpou priamo od daného užívateľa. Predajom kľúča totiž užívateľ neprichádza k ujme, nakoľko svoje aktíva už s najväčšou pravdepodobnosťou utratil [31, 50].

  Tomuto útoku je možné predísť niekoľkými spôsobmi. Jedným je napríklad pravidelná zmena kľúčov užívateľov počas vývoja blockchainu. Pri vytváraní histórie kľúčov je ale nutné mať k dispozícii ďalšie pamäťové zdroje. Preto boli vyvinuté algoritmy, ktoré spájajú viacero kľúčov do jedného. Príkladom takéhoto riešenia je napríklad *Pixel* [32].

  Iným riešením je vkladanie hashu posledného známeho bloku priamo do transakcií, čím prakticky nie je možné upravovať históriu blockchainu (*context-sensitive transacions*) [50].

- **SRA (Short Range Attack)** je založený na rovnakom princípe ako LRA, ale na rozdiel od neho ide iba do histórie niekoľkých blokov [31].

- **Grinding attack**. Útočník môže zvýšiť šancu na svoje publikovanie bloku ovplyvnením výberu lídra komisie. To môže nastať v stave, keď je užívateľ publikujúci nový blok zverejnený ešte pred začatím nového kola. Útočník analýzou môže napríklad zistiť, že o výbere lídra je rozhodnuté len na základe hashu posledného bloku v blockchaine. Ten ale vie pozmeniť jednoduchou modifikáciou jeho obsahu, takže v nasledujúcom kole je pravdepodobnosť zvolenia útočníka za lídra vyššia. Riešením je výber lídra z pomedzi členov komisie. Algorand k tomuto účelu používa funkcie VRF, ktoré náhodným spôsobom vyberajú členov aj lídra komisie. K ich rozoznaniu slúži vopred stanovený prah pre hodnoty vygenerované funkciou VRF. Čím nižšia hodnota je vygenerovaná pre daného užívateľa, tým vyššiu má pravdepodobnosť, že sa stane lídrom komisie [50].

	Algorand	Casper	Gasper
	Snow White	**Stellar**	**Decred**
Priepustnosť	$> 1\ 000$ tx/s	< 50 tx/s	$> 1\ 000$ tx/s
	> 125 tx/s [27]	až do 10 000 tx/s [33, 67]	14 tx/s [67]
Škálovateľnosť	vysoká, možnosť škálovania miliónov užívateľov	vysoká škálovateľnosť [71]	vysoká škálovateľnosť
	snaha o vysokú škálovateľnosť	vysoká škálovateľnosť [90]	stredne vysoká (bez Lightning siete)
Súkromie	verejné kľúče a VRF	TRS (Traceable Ring Signature) [40]	TRS (Traceable Ring Signature) [40]
	slabé, iba verejné kľúče	rôzne úrovne autentizácie	vysoké zameranie na súkromie [58]
Tolerancia zlyhania uzlov	áno, byzantská dohoda (PBFT)	áno, byzantská dohoda (PBFT)	áno, byzantská dohoda (PBFT)
	podpora veľkého množstva offline užívateľov	nízka tolerancia	zabezpečená stabilita siete [29]
Živosť (liveness)	konsenzus za menej ako 1 minútu	len za dokonalých podmienok, inak nie [74]	konsenzus za menej ako 1 minútu
	garantovaná	nie je garantovaná	nízka, nový blok raz za 5 minút
Živosť (safety)	áno, ak je synchronizácia na aspoň niekoľko hodín	optimálna, rôzne mechanizmy	záleží od spôsobu výberu lídra
	áno, špeciálny prístup po prebudení uzlov [12]	uprednostňovaná pred živosťou	garancia bezpečnosti [29]
Konečnosť	všetky transakcie sú konečné	dosiahnutá pomocou kontrolných bodov	dosiahnutá pomocou kontrolných bodov
	nezabezpečená	necieli na dosiahnutie konečnosti	vyššia ako v systéme Bitcoin [22]
Čas do konečnosti	jednotky sekúnd (rýchlo)	dlhšia doba, až niekoľko hodín	jednotky minút (rýchlo)
	dlhšia doba	dlhšia doba	niekoľko dní (niekedy až 142 dní)

Tabuľka 3.2: Teoretické porovnanie vlastností protokolov Algorand, Casper, Gasper, Snow White, Stellar a Decred.

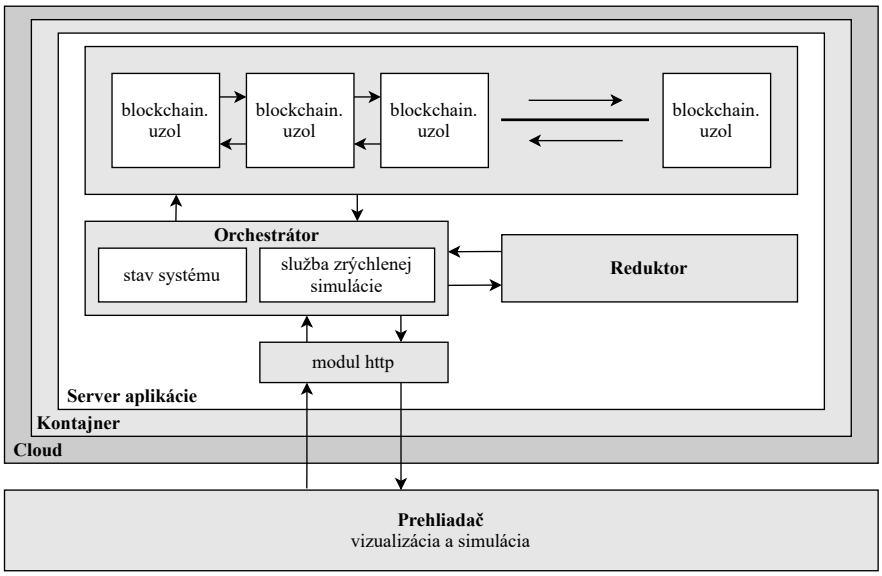
Kapitola 4

Simulačné nástroje

Pre potreby testovania vlastností blockchainových protokolov bolo vyvinutých niekoľko simulačných nástrojov. Väčšina z nich bola vytvorená pre simuláciu protokolov zo skupiny Proof-of-Work hlavne kvôli ich značnej rozšírenosti. V našom prípade budeme skúmať výkonnosť algoritmov a ich správanie pod rôznymi útokmi. Poďme sa spoločne pozrieť na spôsob fungovanie niekoľkých zaujímavých simulačných nástrojov slúžiacich na testovanie blockchainových sietí.

4.1 VIBES

VIBES je simulátor blockchainových sietí postavených na technológii peer-to-peer. Bez potreby stoviek reálnych uzlov dokáže simulovať aj beh sietí o rozmere tisícok uzlov. Jeho veľkou výhodou sú optimálna škálovateľnosť a vysoká rýchlosť. Vďaka týmto vlastnostiam ponúka objektívny pohľad na rýchlo dosiahnuté výsledky. Podporuje konfiguráciu rôznych vstupných parametrov, ako napríklad latencia, topológia siete, počet útočiacich uzlov, či cena energie. Výstupom simulácie sú hodnoty predstavujúce štatistické informácie o transakciách, ale aj pravdepodobnosť úspešného útoku v jednotlivých štádiách [76].



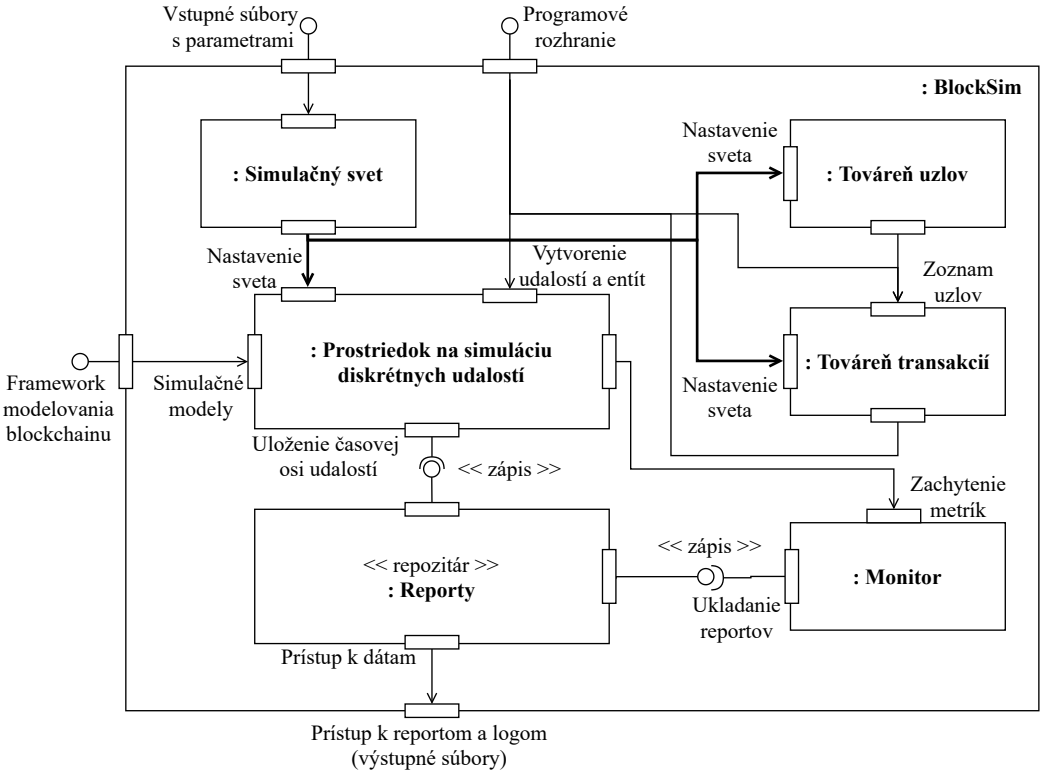
Obr. 4.1: Architektúra simulačného nástroja VIBES [76].

Jednou z veľkých výhod, ktoré poskytuje je možnosť zrýchlenej simulácie. Pokiaľ sú pri spustení zadané parametre s teoretickými hodnotami, ktoré predstavujú jednotlivé výpočty, tak VIBES môže preskočiť náročné výpočty. Napríklad, pokiaľ má byť čas trvania vytvorenia bloku 10 minút, tak VIBES v režime zrýchlenej simulácie vykoná potrebné výpočty a povolí uzlu vytvorenie bloku rýchlosťou niekoľkých milisekúnd. Celá doba simulácie je teda výrazne kratšia a užívateľ má možnosť častejšie meniť potrebné parametre a získať chcené výsledky. Zároveň, VIBES dokáže zabezpečiť zrýchlenie simulácie bez straty kvality výsledkov [76].

Architektúra simulátora VIBES využíva cloudové služby pre beh aplikačného serveru. V ňom je webový http modul prijímajúci parametre simulácie, ktoré predáva orchestrátoru. Ten využíva reduktor a na základe parametrov riadi beh blockchainových uzlov. Náhľad na túto architektúru je zobrazený na obrázku 4.1 [76].

4.2 BlockSim

Nástroj BlockSim je postavený na stochastickom simulačnom modele. Dokáže modelovať udalosti na základe distribúcie pravdepodobnosti ich vzniku. To vyžaduje zmeny stavu systému v diskrétnom čase. BlockSim kontroluje operácie bežiace v spojitom čase v istých časových intervaloch, čím dochádza k ich diskretizácii a zároveň nižším nárokom na beh simulátora. Vďaka tomu je možné sledovať stovky uzlov zároveň. Užívatelia smú využiť tento nástroj k simulácii existujúcich protokolov v už vytvorených modelov ale taktiež môžu vytvoriť nové modely. BlockSim umožňuje zadanie rôznych parametrov vrátane la-



Obr. 4.2: Architektúra simulačného nástroja BlockSim [37].

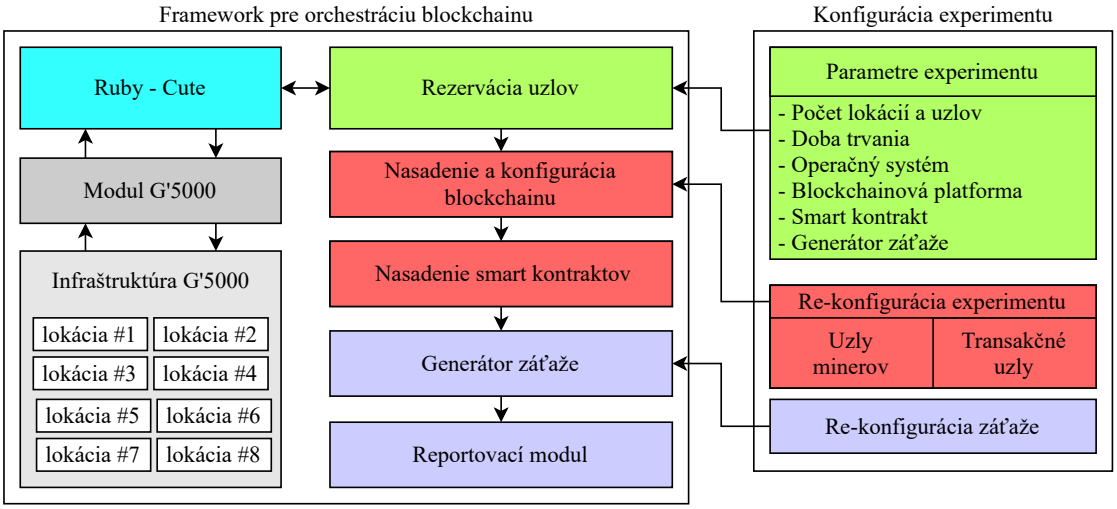
tencie alebo trvania tvorby blokov. Taktiež je možné vytvoriť továrne na uzly či transakcie, ktoré budú neskôr broadcastovo rozosielané medzi uzly. Vďaka dynamickej povahe umožňuje uzlom pripájanie a taktiež odpájanie zo simulovaného systému. Architektúra obsahuje modul na monitorovanie simulácie zo zachytených metrík a modul slúžiaci na vytváranie reportov. Celá simulácia môže byť užívateľom upravovaná v programovacom rozhraní BlockSimu za pomoci jazyka Python. Je určený predovšetkým pre testovanie výkonu protokolov a nepodporuje testovanie správania pri rôznych typoch útokov na blockchain. Architektúra nástroja BlockSim je zobrazená na obrázku 4.2 [37].

4.3 Bitcoin Simulator

Tento simulačný nástroj slúži k skúmaniu vlastností a správania blockchainových Proof-of-Work systémov pri použití rôznych parametrov. Pri jeho vytvorení boli použité štatistiky z reálnych dát už existujúcich sietí a to vrátane latencie. K svojmu behu využíva sieťový simulačný nástroj ns-3, ktorý podporuje udalosti v diskrétnom čase. Simulácie merajú hodnoty ako čas šírenia bloku, či priepustnosť siete. Na základe výstupných hodnôt použitých v Markovových rozhodovacích procesoch je možné zisťovať mieru bezpečnosti protokolov. Môžeme tak sledovať správanie pri rôznych útočných stratégiách a meniť parametre tak, aby sme dosiahli optimálny výkon a bezpečnosť PoW protokolov. Bitcoin Simulator nemodeluje výmenu transakcií medzi uzlami, pretože tento proces neprináša adekvátne informácie potrebné pre skúmanie parametrov konsenzu. Transakcie sú preto implicitne vložené do bloku o veľkosti zadanej v parametroch [43, 37, 44].

4.4 BlockZoom

Podobne ako nástroj VIBES aj BlockZoom podporuje simulovanie rozsiahlych blockchainových systémov. Simulačné prostredie stojí na platforme Grid'5000 a snaží sa vytvoriť podmienky podobné tým v produkčnom prostredí. Užívateľ má možnosť hrať sa s konfi-



Obr. 4.3: Architektúra simulačného nástroja BlockSim. Orchestračný framework riadi simulačné prostredie. Konfigurácia experimentu obsahuje parametre pre spustenie a rekonfiguráciu experimentu [72].

guráciou jednotlivých uzlov a následne porovnávať výsledky vykonaných simulácií. Okrem konfigurácie uzlov je možné upravovať aj záťaž systému čím môžeme vytvárať rôzne stresové podmienky pre blockchain. K simuláciám sa využívajú zdroje uložené na niekoľkých geografických miestach vo Francúzsku a Holandsku, medzi ktorými sa vytvára privátna sieť. Vďaka tomu sú vlastnosti simulácie, ako je napríklad latencia siete, veľmi podobné tým reálnym. Tento nástroj je v dobe písania práce stále vo vývoji, a teda jeho vlastnosti sú hlavne teoretické. Architektúra nástroja BlockZoom je zobrazená na obrázku 4.3 [72].

4.5 SimBlock

SimBlock je simulačný nástroj založený na udalostiach, podobne ako to je u nástroja Bitcoin Simulator. Každý uzol generuje správy a udalosti informujúce o ťažení. Užívateľ má možnosť upravovať parametre, ako veľkosť bloku, počet uzlov a ich susediacich uzlov alebo šírku sieťového pásma. Simulácia transakcií zatiaľ nie je implementovaná, no autori ju plánujú v skorej budúcnosti doplniť. Podporuje vytvorenie modelu pre ľubovolný typ konsenzu, pričom autormi už implementované modely sú pre Proof-of-Work protokoly Bitcoin, Litecoin a Dogecoin. Tento nástroj je napísaný v jazyku Java a okrem klasickej simulácie podporuje taktiež jej vizualizáciu z výstupu uloženého vo formáte JSON [5].

4.6 Podpora Proof-of-Stake

	Simulované protokoly	**Podpora PoW a PoS**
VIBES	Bitcoin	implementácia zameraná na PoW
BlockSim	Bitcoin, Ethereum	PoW
BlockZoom	Ethereum	PoW
SimBlock	Bitcoin, Dogecoin, Litecoin	PoW, podpora ďalších konsenzových algoritmov
Bitcoin Simulator	Bitcoin, Dogecoin, Litecoin	PoW, možnosť rozšírenia o PoS

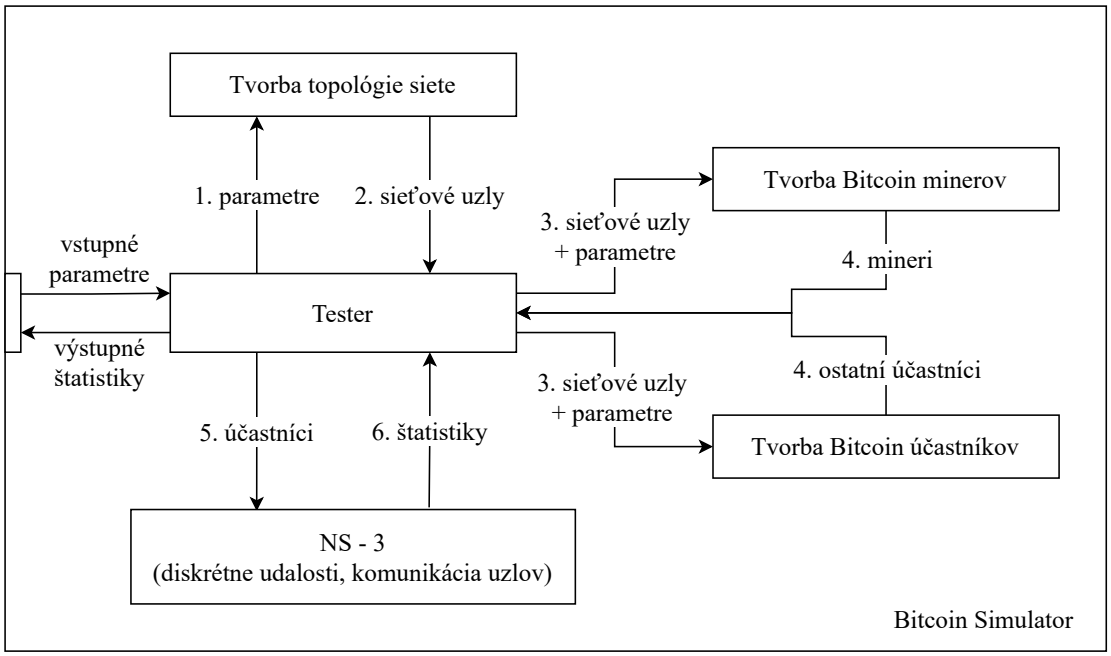
Tabuľka 4.1: Porovnanie podporovaných protokolov u simulátorov VIBES, BlockSim, BlockZoom, SimBlock a Bitcoin Simulator [5, 37, 44, 72, 76].

Simulačné nástroje neuvádzajú či sú priamo určené pre potreby simulovania Proof-of-Work protokolov alebo podporujú aj protokoly založené na Proof-of-Stake. Zväčša ale uvádzajú, ktoré protokoly boli nimi simulované. Prípadne, ak je dostupná implementácia, tak sa táto vlastnosť dá určiť zo zdrojového kódu. V tabuľke 4.1 je uvedený súhrn protokolov, ktoré boli simulované autormi jednotlivých nástrojov a taktiež, či podporujú len protokoly založené na PoW alebo aj na PoS.

Kapitola 5

Návrh a implementácia

Z vyššie opísaných protokolov boli pre účel testovania zvolené tri: Algorand, Casper FFG a Gasper. Základom simulovania je voľba správneho nástroja. Každý z nástrojov, ktoré boli spomenuté, sa v niečom odlišuje a má svoje výhody či nevýhody. Pre testovanie výkonu a bezpečnosti je potrebné vytvoriť nový nástroj, ktorý bude schopný simulovať nami vybrané protokoly. Ako základ tohto nástroja bol zvolený nástroj Bitcoin Simulator, ktorý bol rozšírený o podporu simulácie týchto protokolov. Diagram tried na obrázku A.2 v prílohe A zobrazuje najdôležitejšie vlastnosti a metódy tried slúžiacich pre simuláciu protokolov Algorand, Casper FFG a Gasper. Ukážka výstupu vytvoreného simulátoru je na obrázku A.1 v prílohe A.



Obr. 5.1: Schéma nástroja Bitcoin Simulator zobrazujúca jednotlivé moduly a zjednodušený proces simulácie.

5.1 Jadro simulačného nástroja

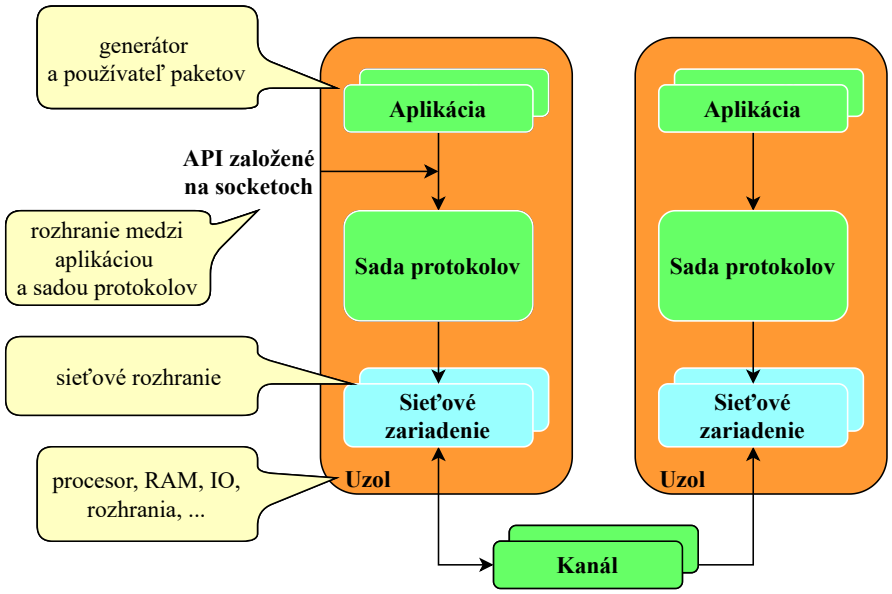
Aby bolo možné opísať implementáciu výsledného nástroja, tak je nutné pochopiť spôsob fungovania jeho jadra, ktorým je nástroj Bitcoin Simulator stojaci na frameworku NS-3. Na obrázku 5.1 je schéma tohoto nástroja zobrazujúca hlavné komponenty a zjednodušený proces behu programu.

Základom je modul zvaný *Tester*. Jeho úlohou je riadenie celého priebehu programu na základe vstupných parametrov. Celá simulácia prebieha s využitím frameworku NS-3, ktorý dokáže simulovať diskrétne udalosti a sieťovú komunikáciu medzi uzlami. Prvou úlohou je konfigurácia siete a vytvorenie sieťových uzlov, ktoré sú schopné medzi sebou komunikovať. K tomuto účelu slúži trieda *BitcoinTopologyHelper*. Tester v ďalšom kroku volá dve triedy, *BitcoinMinerHelper* a *BitcoinNodeHelper*. Obe z nich sú určené k vytvoreniu objektov, predstavujúcich účastníkov blockchainového systému. Štandardní účastníci, zastúpení triedou *BitcoinNode*, sú schopní odosielať a prijímať správy o jednotlivých blokoch a taktiež ukladať bloky do blockchainu. Objekty triedy BitcoinMiner sú rozšírením štandardných účastníkov. Slúžia k ťaženiu blokov, ktoré následne zasielajú medzi ostatné uzly. Účastníci prijímajú bloky a zapisujú si ich do svojich blockchainových databáz. Pre účely útokov na blockchain a simulovanie iných protokolov je nutné vytvoriť ďalší typ účastníkov, ktorí sa budú správať podľa našich potrieb.

Počas simulácie je možné zbierať informácie o priemernej dobe prenosu bloku od minera k jednotlivým účastníkom, či priemernej rýchlosti prijímania blokov. Na konci simulácie sa vyhodnocuje počet blokov v blockchaine, počet vetiev a dĺžka najdlhšej vetvy. Všetky tieto informácie sú po behu simulácie vypísané na štandardný výstup.

5.1.1 Sieťová simulácia

Hlavnou zložkou simulačného nástroja pre testovanie blockchainových protokolov je framework NS-3. Jedná sa o simulátor diskrétnych udalostí prebiehajúcich na sieti. NS-3 je open



Obr. 5.2: Architektúra simulátoru NS-3 [3].

source projekt udržiavaný komunitou dobrovoľníkov z celého sveta. Bol vytvorený primárne pre výskumné a edukačné účely.

Tento nástroj poskytuje rôzne komponenty ako sú napríklad sieťové uzly, zariadenia, kanály. Jednotlivé komponenty môže užívateľ implementovať podľa svojej potreby alebo môže využiť už vstavané riešenia. Všetky procesy sú sériovo spúšťané kernelom NS-3, pričom k tomuto účelu je použitý mechanizmus plánovania udalostí. Udalosti sú spúšťané na základe času simulácie a nie reálneho času. Simulácia tak môže trvať kratšiu dobu. Samozrejme, nie je potrebné mať reálny počet uzlov, ale je možné ich počet simulovať. Framework podporuje implementáciu simulácií v jazykoch C a C++.

Architektúra nástroja NS-3 je na obrázku 5.2. Hlavnou zložkou sú uzly, ktoré medzi sebou komunikujú. Uzol predstavuje akýkoľvek prvok, ktorý sa nachádza v sieti. V našom riešení zastupujú uzly účastníkov blockchainu. Komunikácia uzlov prebieha skrz rôzne kanály. Pre použitie jedného kanálu potrebujú jedno sieťové zariadenie. Pakety obdržané sieťovým zariadením sú spracované jednotlivými protokolmi. Na každom uzle môže bežať taktiež niekoľko aplikácií. Táto schéma je navrhnutá tak, aby sa čo najviac podobala fungovaniu zariadení v reálnych sietiach. Aplikácie sú prepojené s protokolmi pomocou socketov. Každý socket slúži pre odosielanie a prijímanie správ medzi aplikáciou a sadou protokolov na danom uzle. Uzly sú prepojené kanálmi, ktoré predstavujú buď jednotku zabezpečujúcu bezdrôtovú komunikáciu alebo štandardný sieťový kábel. Samozrejme jeden kanál môže mať niekoľko spojení so sieťovými uzlami. Vďaka tomu je možné vytvárať rôzne sieťové topológie [3].

Náš nástroj využíva pre účel simulácie IPv4 adresový priestor, z ktorého prideľuje uzlom ich adresy. Transport správ obsahujúcich dáta medzi uzlami je riadený protokolom TCP (transmission control protocol). Výhodou frameworku NS-3 je jeho podpora simulovania rýchlosti prenosu dát medzi uzlami, čím simulácia dosahuje výsledky podobné tým, ktoré by boli dosahované pri reálnych podmienkach.

5.1.2 Distribuovaná simulácia

Pre zvýšenie rýchlosti programu je možné využiť paralelného behu na viacerých procesorových jadrách. Simulácia je rozdelená na niekoľko logických procesov, kde každý z nich má pridelený jeden procesor. Pokiaľ sú dostupné dostatočné procesorové a pamäťové zdroje, tak paralelná a distribuovaná simulácia je veľkou výhodou. Pre presnosť simulácie je nutné zabezpečiť komunikáciu medzi logickými procesmi. K tomuto účelu používa NS-3 knižnicu MPI (Message Passing Interface). Podpora pre paralelnú a distribuovanú simuláciu je však dostupná len pre point-to-point spojenia.

NS-3 podporuje dva prístupy pre zabezpečenie synchronizácie logických procesov. Prvým je stratégia globálnej synchronizácie hodín medzi všetkými logickými procesmi. Druhou stratégiou je odosielanie takzvaných null správ medzi logickými procesmi, ktoré navzájom zdieľajú point-to-point spojenie. Tento prístup je výhodný vtedy, keď uzly nie sú usporiadané do topológie typu mesh, v ktorej má každý uzol spojenie s každým. Implementované riešenie je navrhnuté tak, aby užívateľ mohol zvoliť minimálny a maximálny počet spojení na jeden uzol. Preto je v parametroch spustenia taktiež podpora výberu stratégie pre synchronizovanie logických procesov, ktorá bude použitá v prípade spustenia na viacerých procesoroch. Z testovania sme zistili, že druhá stratégia, využívajúca null správy, dosahuje vyšších rýchlostí. Voľba stratégie je ponechaná na užívateľovi.

5.2 Algorand

Simulácia protokolu Algorand prebieha nasledovne. Hlavnou komponentou je trieda *AlgorandParticipant* predstavujúca účastníkov systému, ktorí sa podieľajú na hlasovaní a tvorbe blockchainu. Táto trieda rozširuje už opísanú triedu BitcoinNode. Na rozdiel od simulácie Bitcoinu, kde je celková doba vypočítaná na základe priemernej doby generovania bloku, v našom riešení je možné dobu simulácie v minútach (simulačného času) určiť parametrom. Každý z účastníkov sa riadi plánom udalostí, ktoré sú spúšťané v pravidelných intervaloch. Týmito udalosťami sú jednotlivé fázy protokolu:

1. návrh nového bloku,
2. soft vote,
3. certify vote.

Doba, ktorá je medzi týmito fázami je daná v jednotkách sekúnd simulačného času. Program je navrhnutý tak, aby mohol užívateľ tieto časy upraviť podľa svojich potrieb. Voľba správnych intervalov medzi fázami je nutná preto, aby po odoslaní návrhu bloku pomocou protokolu Gossip nezačala ďalšia fáza priskoro. Ak by došlo k tomuto stavu, mohlo by sa stať, že v ďalšej fáze nebude mať uzol správny súbor blokov, či hlasov. To by spôsobilo, že hlas účastníka v tejto fáze by bol nezámerne neplatný. Pokiaľ by bolo takýchto účastníkov priveľa alebo by bola doba prenosu dát medzi uzlami značne väčšia ako doba medzi fázami, tak by mohlo dôjsť k stavu, kedy by každý z účastníkov mal iné dáta v svojom blockchainovom ledgeri. Preto je potrebné voliť dobu medzi fázami úmerne veľkosti siete, počtu hlasujúcich uzlov (členov komisie) a počtu uzlov, s ktorými sú spojení.

5.2.1 VRF

Každý z účastníkov má pri jeho vytvorení vygenerovaný 64 bytový súkromný a 32 bytový verejný kryptografický kľúč. Pre ich generovanie je použitá knižnica **libsodium**[1] upravená autormi Algorandu, ktorí ju využívajú pri implementácii tejto kryptomeny. Z knižnice boli využité aj funkcie na generovanie hodnôt VRF, ktoré sú potrebné pri výbere členov komisie a lídrov tvoriacich nové bloky. Účastníci na začiatku fáz vyhodnocujú VRF a výstup z nej určí, či sa stanú členmi komisie alebo nie. Výstupom VRF je 64 bytová hodnota, ktorá sa porovnáva s globálne nastaveným prahom. Ak je hodnota nižšia ako prah, tak sa účastník stáva členom komisie. Prah musí byť preto určený tak, aby bol počet členov komisie pre jednotlivé fázy primeraný. Ak by bolo zvolených priveľa členov, tak by klesla živosť (liveness) tvorby blockchainu. Ak by ich bolo málo, ohrozilo by to bezpečnosť blockchainu. Treba taktiež počítať s tým, že občas môže dôjsť k neplatnosti hlasov spôsobených napríklad chybami pri prenose dát v sieti.

Implementované riešenie má podporu čiastočnej voľby prahu. Čiastočnou je z dôvodu, že užívateľ neurčuje každý z 512 bitov prahu ale pomocou parametrov určuje počet núl, ktoré sú na začiatku prahu. Na základe počtu núl je pred behom programu vyhodnotená hodnota prahu. Pre každú z fáz je určený prah samostatne, a teda existujú až tri parametre určené k vyhodnocovaniu VRF.

[1] libsodium: A modern, portable, easy to use crypto library, `https://github.com/algorand/libsodium`

5.2.2 Spracovanie správ

V jednotlivých fázach sú členmi komisie generované správy obsahujúce návrh bloku alebo hlas. Dáta posielané v správach sú ukladané do formátu JSON. Keďže správa obsahuje dáta rôzneho typu, tak pre prácu je použitá knižnica **RapidJSON**. Účastníci pri prijatí správy rozlišujú aký typ dát správa nesie a podľa toho ich predáva príslušným metódam. Každá z týchto metód najskôr overuje či bol účastník, ktorý podpísal prijatý blok alebo hlas, skutočne zvolený pomocou VRF. Ak po overení nie je správa zahodená, tak sa dáta z nej spracúvajú pomocou princípov uvedených pri popise protokolu Algorand. Na záver je táto správa odoslaná ďalším uzlom pomocou Gossip protokolu.

5.3 Casper the Friendly Finality Gadget

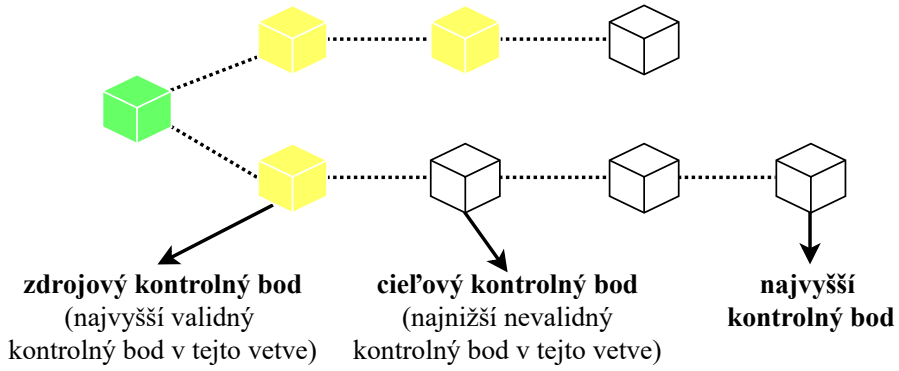
Ďalším z protokolov, ktoré boli implementované je Casper FFG. Okrem iných, vyžaduje implementáciu dvoch nových tried, *CasperParticipant* a *CasperMiner*. Objekty triedy CasperParticipant zastupujú účastníkov blockchainu, ktorých hlavnou úlohou je hlasovanie o kontrolných bodoch vedúce k finalizovaniu blokov. Nezodpovedajú za vytváranie nových blokov ale k zaisteniu vyššej bezpečnosti blockchainu. Pri spustení môže užívateľ zvoliť počet minerov a počet voličov. Okrem ďalších parametrov, je možné zmeniť počet blokov, ktoré určujú jednu epochu. Čím vyššia je táto hodnota, tým väčšia je pravdepodobnosť, že všetky uzly obdržia väčšinu hlasov aj pri nižšej rýchlosti prenosu dát. Predvolená je hodnota 50 blokov, no táto voľba je plne na užívateľovi.

Pred vložením nového bloku do blockchainu, kontroluje každý účastník, volič aj miner, jeho výšku. Ak je výška násobkom počtu blokov v epoche, tak je tento blok označený za kontrolný bod. Následne dochádza k sčítaniu hlasov, ktoré tento účastník prijal. Vyberie sa odkaz, teda zdrojový a cieľový kontrolný bod, s najvyšším počtom hlasov. Ak za tento odkaz hlasovalo väčšinové kvórum voličov, tak sa odkaz stáva supermajoritným a upravuje sa stav blokov v blockchaine podľa pravidiel protokolu Casper FFG. V skratke sa jedná o nasledujúce body:

- zmena cieľového kontrolného bodu na validný bod, ak je zdrojový kontrolný bod validný,
- zmena kontrolného bodu z validného na finalizovaný, ak nasledovný kontrolný bod je tiež validný,
- zmena blokov, ktoré sú predkami nového finalizovaného kontrolného bodu, na finalizované bloky.

5.3.1 Voľba kontrolných bodov odkazu

Až je blockchain upravený, vkladáme do neho nový kontrolný blok. Táto časť je rovnaká pre minerov aj voličov. Voliči v tomto momente prechádzajú k hlasovaniu za ďalší odkaz. Spôsob výberu najvhodnejšieho odkazu medzi kontrolnými blokmi nie je súčasťou protokolu Casper FFG a je závislý na konkrétnom použití. Implementované riešenie preto nasleduje pravidlo používané kryptomenou Ethereum. Toto pravidlo bolo niekoľkokrát upravované, aby dosahovalo čo najlepšie výsledky. Konečným riešením sa stala voľba vetvy, ktorej najnovší kontrolný bod má najvyššiu výšku zo všetkých. Cieľovým kontrolným bodom je zvolený najnižší kontrolný bod medzi týmto a finalizovaným bodom, ktorý ešte nie je označený



Obr. 5.3: Výber zdrojového a cieľového kontrolného bodu podľa pravidla najvyššieho kontrolného bodu používaného v kryptomene Ethereum.

ako validný. Zdrojovým kontrolným bodom je následne vybraný validný kontrolný bod s najvyššou výškou, ktorý je predkom zvoleného cieľového bodu. Týmto zdrojovým bodom môže byť rovnako aj najvyšší finalizovaný bod, nakoľko aj on je považovaný za validný. Samozrejme, hlas nesmie porušovať základné pravidlá stanovené protokolom Casper FFG, ktoré sme si už opísali. Tento princíp voľby zdrojového a cieľového kontrolného bloku je zobrazený na obrázku 5.3.

5.3.2 Ťaženie blokov

Nakoľko je tento protokol považovaný za hybrid medzi Proof-of-Work a Proof-of-Stake blockchainom, tak sme ako základ použili pôvodnú implementáciu simulácie Bitcoinu. Aby sme dosiahli chcených výsledkov, bolo nutné upraviť správanie minerov podľa protokolu Casper. Trieda CasperMiner rozširuje pôvodných minerov triedy BitcoinMiner o zvýšenú kontrolu blokov. Cieľom tejto kontroly je zabránenie stavu, v ktorom sa do blockchainu dostávajú bloky z inej ako finalizovanej vetvy. Znižujeme tým riziko útokov, ako napríklad dvojnásobné utrácanie alebo LRA. Mineri, tak ako ostatní účastníci prijímajú hlasy a vyhodnocujú na ich základe stav blokov v blockchaine už vyššie opísaným procesom. Vďaka tomu vedia, ktoré bloky sú finalizované a ak sú čestné, tak neťažia na blokoch, ktoré by nemali medzi svojimi predkami kontrolný bod s najvyššou výškou. Ťaženie sa nevykonáva lámaním hashu hlavičky bloku nakoľko by to bolo nákladné na výpočtové zdroje. Je však nahradené čakaním a diskrétnymi udalosťami. Priemerná doba generovania nových blokov v minútach simulačného času môže byť zadaná parametrom pri spustení programu.

5.4 Gasper

Posledným z implementovaných protokolov je Gasper. Jeho voľba bola ovplyvnená tým, že protokol patrí medzi aktuálne používané riešenie v kryptomene Ethereum. Implementácia je štruktúrou podobná Algorandu, nakoľko sa jedná o čistý Proof-of-Stake protokol. Objekty triedy GasperParticipant striedajú dve fázy. Hlasovanie za blok, ktorý sa má stať rodičom nového bloku a návrh nového bloku, na základe prijatých hlasov. Obe fázy sú spúšťané plánovačom udalostí na základe intervalov, ktoré zadá užívateľ pri spustení programu. Každá z fáz má samostatný parameter pre interval a ich voľba ovplyvňuje tvorbu ledgera rovnakým spôsobom ako v Algorande.

5.4.1 Voľba členov komisie

Pre každý slot, do ktorého je vložený nový blok je zvolená komisia. Spôsob voľby jej členov nie je definovaný protokolom Gasper. Aby bolo možné zaručiť, že je výber členov náhodný, tak bol použitý už overený spôsob pomocou VRF funkcií, použitých v Algorande. Prah hodnôt pre VRF v oboch fázach je taktiež možné upravovať vstupnými parametrami. Protokol Gasper vo fáze návrhu bloku ale definuje výber iba jedného lídra. Presnejšie povedané, lídrom sa má stať prvý zvolený člen komisie. Je zložité definovať spôsob výberu členov tak, aby každý z členov vedel, v akom poradí boli zvolení. Preto bol pri výbere lídra ponechaný prístup pomocou VRF, ktorý určí niekoľkých členov. Na začiatku ďalšej fázy účastníci ešte pred vyhodnotením VRF vkladajú do ledgera blok, ktorý pre daný slot navrhol užívateľ s najnižšou hodnotou VRF výstupu.

5.4.2 Hlasovanie

Hlasy členov komisie obsahujú dve zložky. Návrh najvhodnejšieho rodičovského bloku pre nový blok a odkaz medzi dvomi kontrolnými bodmi. Výber najvhodnejšieho rodičovského bloku je založený na hybridnom LMD pravidle (HLMD). Kombinuje prístup protokolov Casper CBC a Casper FFG. Prvým krokom je nájdenie najvyššieho validného kontrolného bodu, ktorý je platným v danej epoche. Platným je, ak niektorým z jeho predkov je posledný finalizovaný kontrolný bod. Následne sa hľadá vždy potomok, za ktorého vetvu v predchádzajúcej iterácii hlasovalo najviac voličov. Tento postup je opakovaný, až kým výsledný blok nemá ďalších potomkov. Algoritmus 1 uvádza spôsob vyhodnotenia HLMD podľa vyššie opísaného princípu.

Algoritmus 1: Hybridné LMD pravidlo výberu rodičovského bloku

Input: epocha e
Output: B

1: $B_J \leftarrow$ najvyšší validný kontrolný bod v epoche e
2: $M \leftarrow$ hlasy prijaté v predchádzajúcej iterácii (jeden na člena komisie)
3: **while** B *nie je listový blok* **do**
4: $C \leftarrow$ priamy potomkovia bloku B
5: **if** $1 = C.size()$ **then**
6: $B \leftarrow C.at(0)$
7: **else**
8: $B \leftarrow \arg\max w(C, M)$
9: **end if**
10: **end while**
11: **return** B

Po voľbe rodičovského bloku je na rade voľba odkazu medzi kontrolnými bodmi. Pre výber zdrojového a cieľového kontrolného bodu bol implementovaný rovnaký princíp ako pri protokole Casper FFG. Teda, pravidlo najvyššieho kontrolného bodu, ktoré používa kryptomena Ethereum.

5.4.3 Budovanie ledgera

Vo fáze návrhu bloku sú zvolení lídri, ktorí vytvárajú nový blok. Výber rodičovského bloku prebieha opäť pomocou HLMD pravidla s pomocou hlasov od ostatných členov komisie. Do bloku sa taktiež vkladá nové semienko slúžiace na vyhodnotenie funkcie VRF.

Ako sme už spomenuli, pred novým hlasovaním účastníci vkladajú do ledgera nový blok. Po jeho vložení je kontrolované číslo slotu, do ktorého bol nový blok vložený. Ak je toto číslo deliteľné počtom blokov v epoche, tak je blok označený za kontrolný bod. Po vložení takéhoto bloku do ledgera prebieha úprava blokov v blockchainovom ledgeri podobne implementácii v protocole Casper FFG. Ako sme si však mohli všimnúť, narozdiel od protokolu Casper FFG neberieme už do úvahy výšku bloku ale číslo slotu. Preto, je implementácia vkladania obyčajného bloku do ledgera odlišná. Totiž, je nutné skontrolovať predkov bloku a nájsť príslušný kontrolný bod EBB. Princíp hľadania EBB, ktorý bol v riešení implementovaný, je opísaný v sekcii 3.3.

Kapitola 6

Testovanie a experimenty

Po implementovaní simulátorov boli vykonané testy, ktoré umožnili priblížiť vlastnosti jednotlivých protokolov. Okrem štandardných testov, zameraných napríklad na vplyv veľkosti komisie na celkovú sieťovú prevádzku, boli uskutočnené aj ďalšie experimenty. Pomocou nich sme sledovali toleranciu protokolov na zlyhanie uzlov, a taktiež sme simulovali útoky na protokol Algorand. V tejto kapitole si opíšeme spôsob testovania, priebehu útokov a pozorovaných výsledkov.

6.1 Priepustnosť a živosť

V blockchainových systémoch je dôležité, aby bolo možné rýchlo spracúvať veľké množstvo transakcií. Čím viac transakcii dokáže byť protokolom zapísaných do ledgera, tým lepšie. Priepustnosť je však úzko spätá so živosťou (liveness) protokolu, pretože ak sme schopní vkladať do ledgera bloky rýchlejšie, tak sa zároveň zvýši aj počet zapísaných transakcií.

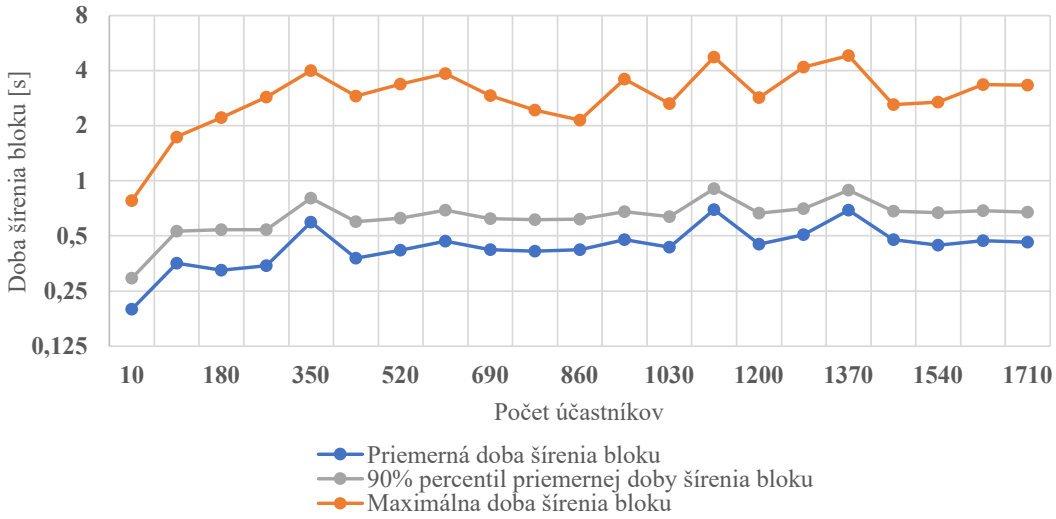
Protokol Casper FFG nebol testovaný na priepustnosť, nakoľko tento protokol slúži ako rozšírenie iných Proof-of-Work protokolov. Jeho priepustnosť preto záleží na použitom základnom protokole. Napríklad ak je pre základ použitý protokol Bitcoin, ktorý tvorí nový blok o veľkosti zhruba 2 500 transakcií každých 10 minút, tak je jeho priepustnosť len okolo 4 tx/s. Preto nás viac zaujímajú čisté Proof-of-Stake protokoly Algorand a Gasper.

Oba z týchto algoritmov sa skladajú z niekoľkých fáz, pričom v jednej dochádza ku návrhu nového bloku a v ďalších sa rozhoduje o validite navrhnutých blokov pomocou hlasov. Hlasy majú veľkosť jednej transakcie a oproti veľkosti navrhnutých blokov je minimálna. Doba prenosu bloku je teda najväčšou brzdou pri dosiahnutí konsenzu. Po prenose bloku ku všetkým účastníkom je možné dosiahnuť konsenzus v ďalších fázach algoritmov behom jednej až pár sekúnd. Skúmali sme, ako závisí doba šírenia bloku na počte účastníkov a veľkosti komisie, ktorá je vybraná vo fáze návrhu bloku.

6.1.1 Vplyv počtu účastníkov na dobu šírenia bloku

Testovaná fáza návrhu bloku bola simulovaná protokolom Algorand, nakoľko nezáleží od spôsobu vytvorenia bloku ale od jeho veľkosti a sieťových podmienok. Výsledky tohoto testu sú preto aplikovateľné aj na protokol Gasper. Pri simulácii bolo uzlom pridelených 8 iných uzlov, s ktorými boli spojení. Tak, ako aj pri ďalších testoch, bola využívaná distribuovaná simulácia so synchronizáciou pomocou null správ.

Simulovali sme dobu 35 minút behu protokolu Algorand. Aby sme zaručili, že každý uzol dostane blok práve vo fáze návrhu bloku, tak bol zvolený interval tejto fázy 10 se-



Obr. 6.1: Vplyv počtu účastníkov na dobu šírenia bloku v logaritmickom merítku. Simulácia 35 minút protokolu Algorand s blokmi pevnej veľkosti 500 KB. Počet členov komisie pre návrh bloku sa pohyboval v rozmedzí 30-50 účastníkov vybraných pomocou VRF. Pri počte účastníkov 10 je každý z nich zvolený do komisie.

kúnd. Veľkosť bloku sme nastavili na pevnú hodnotu 500 KB (512 000 bytov), čo odpovedá približne 2048 transakciám na jeden blok. Simulácia bola vykonaná na počte účastníkov v rozmedzí od 10 do 1710 s rozostupmi o veľkosti 85 účastníkov. Pre každú hodnotu bol určený prah pre vyhodnotenie VRF tak, aby bolo do komisie náhodne vybraných približne 30-50 účastníkov. Samozrejme, pri počte účastníkov 10 boli do komisie zvolení všetci.

Výsledné hodnoty, týkajúce sa doby šírenia bloku, sú zobrazené v grafe na obrázku 6.1. Skúmali sme celkovo 3 hodnoty: priemernú dobu šírenia bloku, maximálnu dobu šírenia bloku a 90% percentil priemernej doby šírenia bloku. Posledná zo spomenutých hodnôt vyjadruje dobu, od ktorej len 10 % účastníkov malo nameranú vyššiu priemernú dobu šírenia bloku počas simulácie.

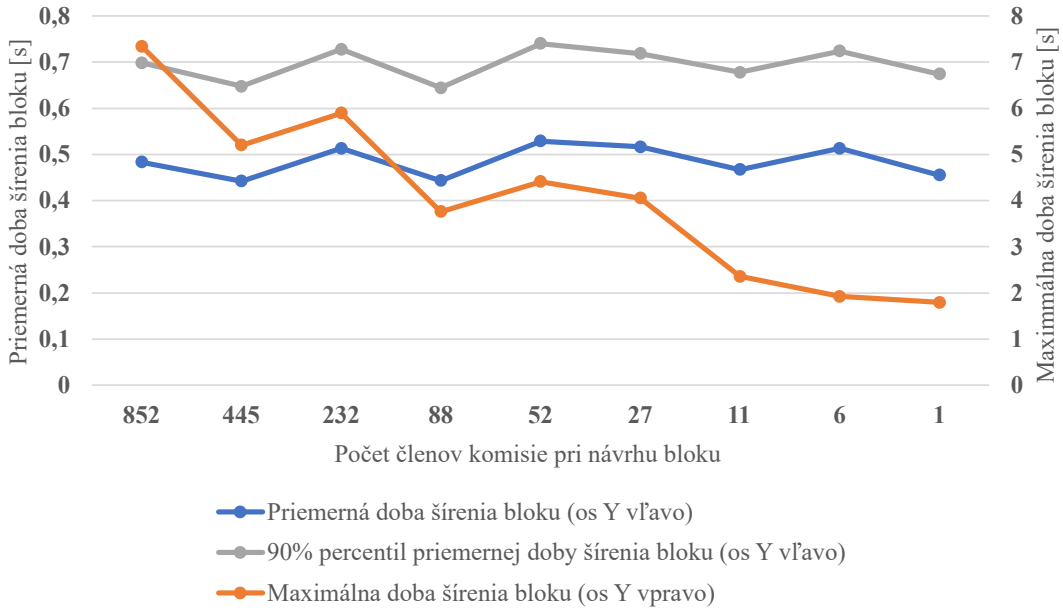
Z grafu na obrázku 6.1 môžeme usúdiť, že protokoly Algorand a Gasper ovplyvňuje počet účastníkov minimálne. Hodnoty dôb šírenia bloku, či už maximálne alebo priemerné, síce so zvyšujúcim sa počtom účastníkov lineárne stúpajú, no udržujú sa na pomerne nízkych hodnotách. Pri maximálnej dobe šírenia je to približne 5 sekúnd, čo znamená, že minimálne raz za beh programu nameral niektorý z účastníkov túto dobu šírenia bloku. Tento stav mohol byť spôsobený tým, že bol zvolený líder na jednom konci siete a správa obsahujúca blok, putovala až na opačný okraj siete.

Priemerná doba šírenia bloku sa udržuje pod jednou sekundou, čo je dôležité. Celková doba pre dosiahnutie konsenzu preto môže byť pomerne nízka. Ak budeme počítať s istými odchýlkami a zvolíme na základe nameraných hodnôt dĺžku fázy návrhu bloku na 4 sekundy a dobu ďalších fáz slúžiacich pre hlasovanie na 1 sekundu, tak sa dostávame na 6 sekúnd u protokolu Algorand a 6 sekundy u protokolu Gasper. To by znamenalo zhruba 340 spracovaných transakcií Algorandom a 410 transakcií spracovaných Gasperom za jednu sekundu pri veľkosti bloku 500 KB. Tieto hodnoty značia o vysokej priepustnosti oboch algoritmov.

Pri implementácii protokolu Gasper a tiež protokolov Casper v kryptomene Ethereum je vplyv počtu účastníkov vyšší, pretože hlasovanie musí rešpektovať gas limit blokov.

To obmedzuje počet hlasovacích volaní smart kontraktu. V práci sme túto vlastnosť nebrali do úvahy, pretože je závislá na konkrétnej implementácii.

6.1.2 Vplyv veľkosti komisie na dobu šírenia bloku



Obr. 6.2: Vplyv veľkosti komisie na dobu šírenia bloku. Simulácia 34 minút protokolu Algorand s blokmi pevnej veľkosti 500 KB. Počet účastníkov bol pri každej simulácii 1700.

Pri vyhodnocovaní závislosti veľkosti komisie na dobu propagácie bloku sú opäť uvedené výsledky simulácie protokolu Algorand. Testy boli vykonané aj na protokole Gasper s totožným výsledkom. Simulácia bola vykonaná podobne ako predchádzajúca zameraná na počet účastníkov. Veľkosť bloku bola fixne nastavená na 500 KB, počet spojení uzlov na 8 a dĺžka fázy návrhu bloku na 10 sekúnd. Simulovali sme 34 minút behu algoritmu s celkovým počtom účastníkov 1700. Prah pre vyhodnocovanie funkcie VRF bol pri každom spustení simulácie upravený s cieľom postupne znižovať počet zvolených členov komisie vo fáze návrhu nového bloku.

Výsledky testovania vplyvu veľkosti komisie na dobu šírenia bloku sú uvedené v grafe na obrázku 6.2. Z nameraných hodnôt môžeme vidieť, že priemerná doba sa nejak zvláštne nemení a 90% percentil priemernej doby šírenia bloku sa udržiava na hodnotách okolo 0, 7 sekundy. Avšak, na hodnotách maximálnej doby propagácie bloku môžeme vidieť, že so zväčšujúcim sa počtom členov komisie sa maximálna doba šírenia lineárne zväčšuje. Je preto vhodné voliť nižší prah pre vyhodnotenie VRF, čím dosiahneme nižší počet členov komisie a narastá priepustnosť algoritmov. Ak vezmeme do úvahy veľkosť komisie 10 až 20 členov, tak sa maximálna doba šírenia bloku hýbe v okolí hodnoty 3 sekúnd. Pri Algorande týmto dosiahneme celkovú dobu pre uzavretie konsenzu približne 5 sekúnd, čo značí približne 400 zapísaných transakcií za sekundu pri veľkosti bloku 500 KB. U protokolu Gasper je to ešte o niečo lepšie, keďže má len jednu fázu, v ktorej sa hlasuje, a teda sa dostávame na hodnotu približne 4 sekúnd pri uvažovaní o najhoršom prípade. Pri tejto rýchlosti je možné zapísať do ledgera až okolo 500 transakcií za jednu sekundu, pri rovnakej veľkosti bloku.

V protokole Gasper samozrejme záleží aj na tom, akým spôsobom je volený líder komisie. Pre tieto výpočty bol braný do úvahy nami implementovaný prístup s použitím VRF. Iný algoritmus môže dobu pre uzavretie konsenzu o niečo zrýchliť či spomaliť. Stále však bude tento protokol, spolu s protokolom Algorand, dosahovať vysokú priepustnosť.

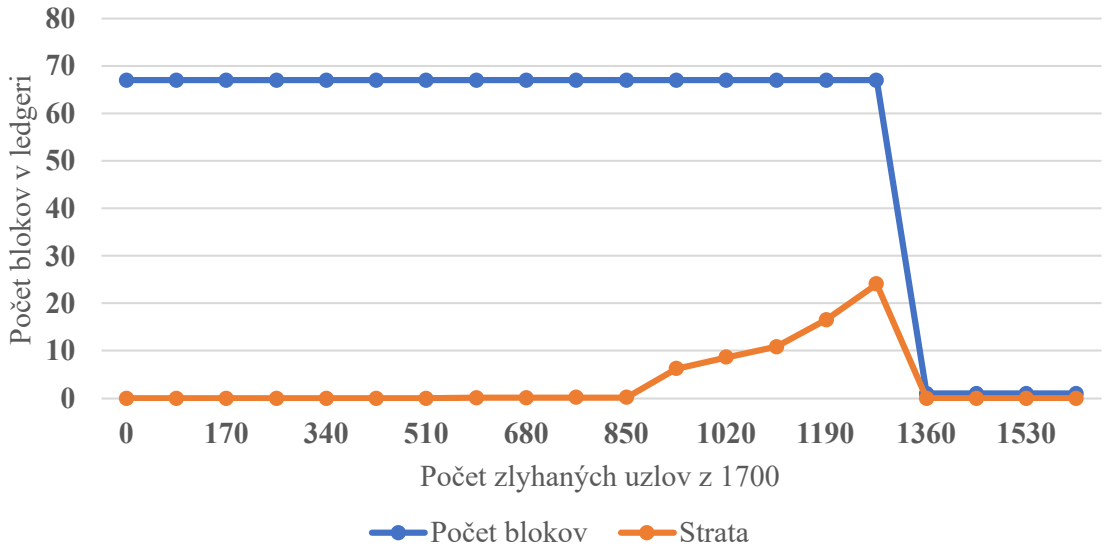
6.2 Tolerancia zlyhania uzlov

Každý z implementovaných protokolov má vysokú teoretickú toleranciu zlyhania uzlov. Je zabezpečená použitím byzantskej dohody (PBFT) pri dosahovaní konsenzu. V našom riešení bol implementovaný spôsob simulácie zlyhania uzlov. Na počiatku behu programu sa vytvorí požadovaný počet účastníkov, no následne sa niektorým z účastníkov nastaví parameter informujúci účastníka o jeho zlyhaní. Počet zlyhaných uzlov je možné zadať ako argument pri spustení programu. Uzol, ktorý má informáciu o zlyhaní nenasleduje kroky protokolu a pri prijatí akýchkoľvek správ od susedných uzlov, tieto pakety zahadzuje a na správy neodpovedá.

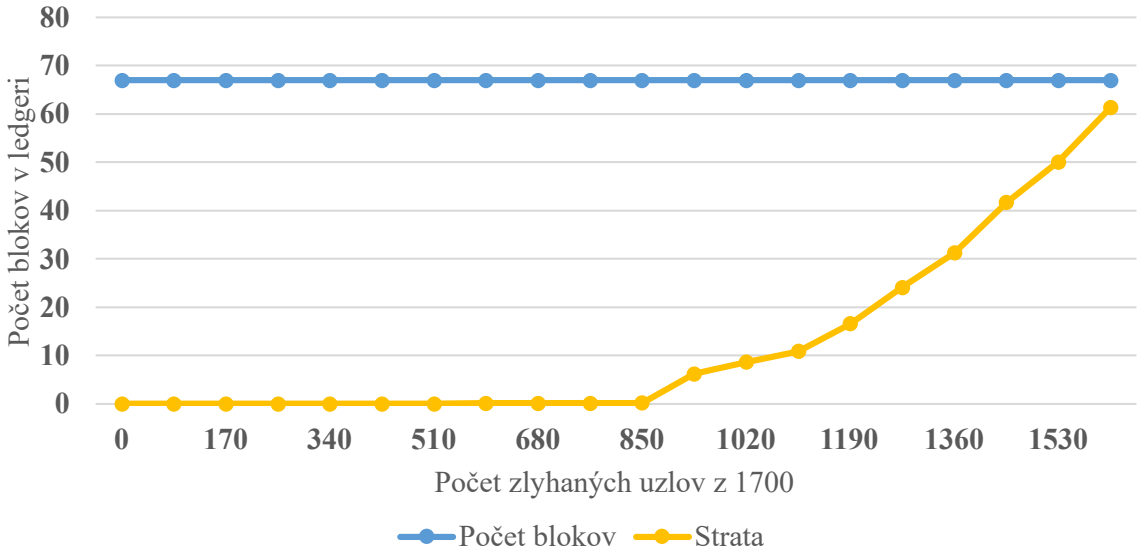
6.2.1 Algorand

Testovanie tolerancie zlyhania Algorandu prebiehalo opäť simuláciou 1700 účastníkov, kde každý mal 8 susedných uzlov. Postupne sme pridávali počet zlyhaných uzlov po 85 a sledovali sme dve hodnoty. Prvou bol počet blokov v ledgeri, určený maximálnou hodnotou nameranou u plne funkčných účastníkoch. Druhou bola strata vyhodnotená rozdielom medzi maximálnym počtom blokov a priemernou hodnotou počtu blokov medzi účastníkmi. Počas tejto simulácie sme nemenili pôvodnú hodnotu prahu pre výpočet VRF.

Získané údaje zo simulácie sú uvedené v grafe na obrázku 6.3. Ako môžeme z nameraných dát vidieť, Algorand je pri nami zvolených parametroch schopný pracovať bez viditeľných problémov až dokým je plne funkčných aspoň 50 % účastníkov. Následne sa začínajú



Obr. 6.3: Vplyv počtu zlyhaných uzlov na počet blokov uložených do ledgera pri stálom prahu funkcie VRF. Simulácia 10 minút protokolu Algorand s blokmi pevnej veľkosti 500 KB. Počet účastníkov bol pri každej simulácii 1700.



Obr. 6.4: Vplyv počtu zlyhaných uzlov na počet blokov uložených do ledgera pri prahu funkcie VRF menenom podľa počtu funkčných uzlov. Simulácia 10 minút protokolu Algorand s blokmi pevnej veľkosti 500 KB. Počet účastníkov bol pri každej simulácii 1700.

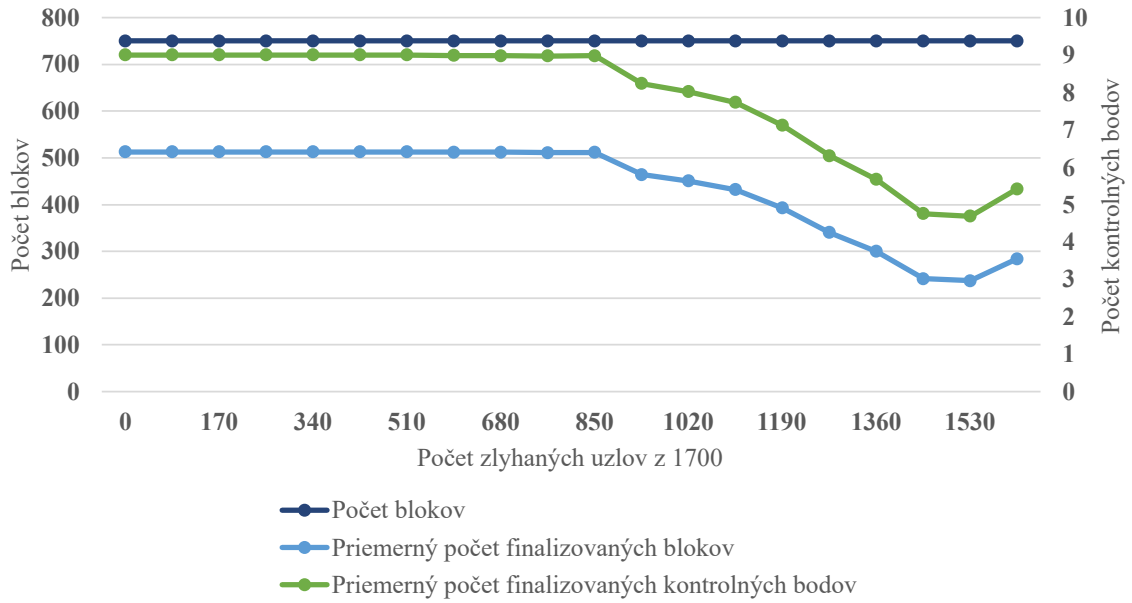
vytvárať sieťové partície. Sú to uzly, ktoré sú od ostatných uzlov oddelené a nie sú plne schopné ukladať dáta do blockchainu. Pri hodnote 1360 (80 % účastníkov) sa začína prejavovať priveľmi vysoká hodnota VRF prahu pre daný počet funkčných účastníkov, určujúca počet členov komisie, ktorá navrhuje nové bloky. Do komisie preto nie je zvolený žiaden z účastníkov, čo má za následok nerozvíjajúci sa ledger.

Vykonali sme taktiež simulácie podľa scenára, v ktorom sme použili rovnaké parametre až na jeden. Podľa počtu plne funkčných uzlov sme menili parameter prahu pre VRF jednotlivých fáz protokolu tak, aby sa vždy dostali do komisie aspoň niekoľkí účastníci. Výsledkom bolo, že pri akomkoľvek počte zlyhaných uzlov bol protokol schopný tvoriť ledger. Hodnoty straty, spôsobenej sieťovými partíciami, sa vyvíjali takmer rovnakým spôsobom, ako pri predchádzajúcej simulácii. To nám však nevadí, pretože za pomoci rôznych techník by mohli byť pri obnovení zlyhaných uzlov partície odstránené a ich ledger zosynchronizovaný s ostatnými uzlami.

V reálnom riešení by uzly mohli reagovať na veľkosť komisie dynamicky. Ak klesne počet členov n-krát po sebe pod istú hodnotu, tak by si účastníci zvýšili hodnotu prahu VRF. Taktiež pri presiahnutí inej hodnoty by sa prah VRF opäť znížil. Pre potlačenie zvyšovania straty je nutné implementovať algoritmus, ktorý bude schopný potlačiť vznik sieťových partícií. Pri odpojení zlyhaní niektorého zo susedných uzlov by sa napríklad mohol uzol napojiť na iný funkčný uzol. Tým by zostal stále v spojení s ostatnými uzlami, ktoré pracujú na budovaní ledgera.

6.2.2 Gasper

Pri overovaní tolerancie zlyhania uzlov protokolom Gasper sme vykonali podobné simulácie ako pri protokole Algorand. Simulovali sme však až 100 minút behu protokolu, kvôli získaniu dát týkajúcich sa tvorby kontrolných bodov a finalizovania blokov. Veľkosť jednej epochy bola nastavená na 64 blokov. Sledovali sme celkovo tri rôzne hodnoty. Prvou



Obr. 6.5: Vplyv počtu zlyhaných uzlov na počet blokov uložených do ledgera pri prahu funkcie VRF menenom podľa počtu funkčných uzlov. Simulácia 100 minút protokolu Gasper s blokmi pevnej veľkosti 500 KB. Počet účastníkov bol pri každej simulácii 1700.

bol opäť počet blokov, určený maximálnou hodnotou nameranou u plne funkčných účastníkoch. Druhá sledovala priemerný počet finalizovaných blokov pri simuláciách. Posledná hodnota sledovala priemerný počet finalizovaných kontrolných bodov, ktoré sa nachádzali v ledgeroch účastníkov.

Výsledky testovania boli zaznamenané do grafu na obrázku 6.5. Z nameraných hodnôt môžeme určiť, že rovnako ako Algorand aj Gasper je vďaka byzantskej dohode vysoko odolný proti výpadkom uzlov. Po celú dobu je schopný tvoriť rovnaký počet blokov v ledgeri. Taktiež, počet finalizovaných kontrolných bodov, a teda aj finalizácia blokov prebieha značne dobre. Ku stratám, spôsobených sieťovými partíciami, začína dochádzať opäť okolo hodnoty 850 účastníkov. Mierne zlepšenie pri hodnote 1615 účastníkov nastalo vďaka inak vygenerovanej sieťovej topológii.

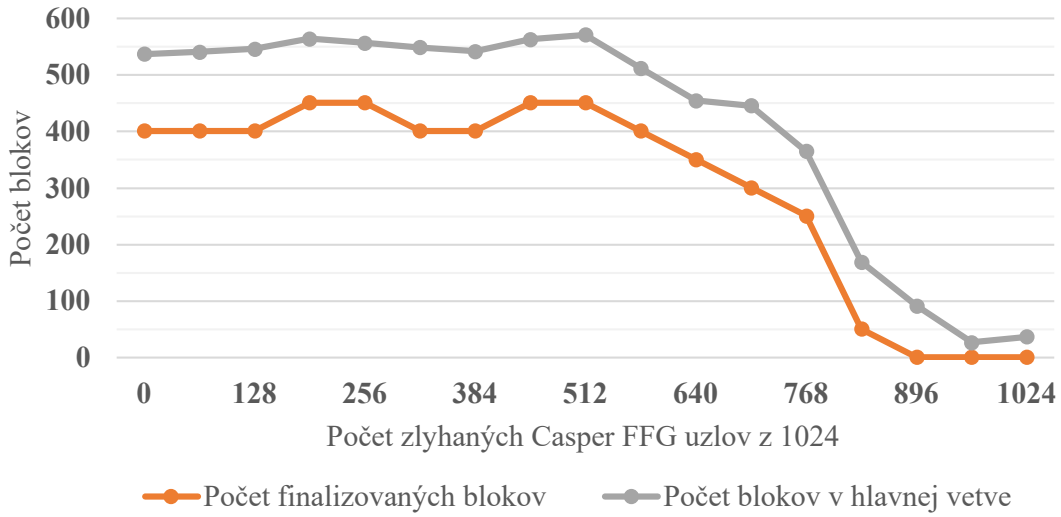
6.2.3 Casper FFG

Simulácia tolerancie zlyhania uzlov protokolom Casper FFG bola spúšťaná nasledovne. Pre test sme zvolili 64 minerov, ktorí približne každých 10 minút generovali nové bloky o veľkosti 500 KB pomocou protokolu Bitcoin. Mimo týchto 64 minerov bolo simulovaných ďalších 1024 validátorov finalizujúcich bloky v ledgeri pomocou protokolu Casper FFG. Každý beh programu simuloval 5 000 minút tvorby blockchainu, pričom pri jednotlivých behoch sme stupňovali počet zlyhaných Casper FFG uzlov v rozostupoch 64 validátorov.

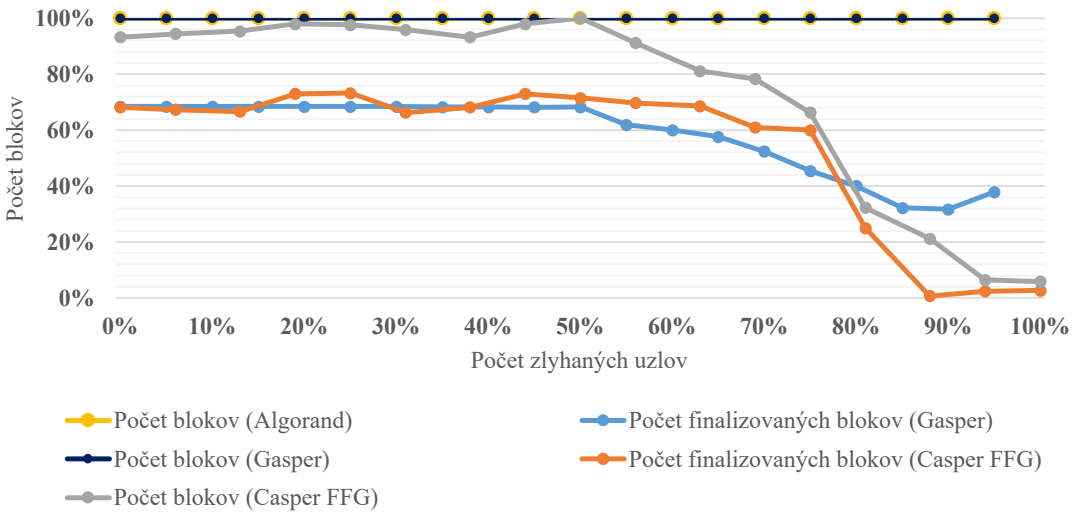
Pri vyhodnocovaní sme sa zamerali na dve hodnoty. Počet blokov v hlavnej vetve a počet blokov, ktoré boli finalizované. Ak niektorí z účastníkov mal na konci simulácie nižšiu hodnotu ako iní účastníci, v dôsledku väčšieho množstva zlyhaných uzlov v sieti, tak sme túto hodnotu nebrali do úvahy.

Výsledky tohoto testu sú zobrazené v grafe na obrázku 6.6. Ako môžeme vidieť, protokol Casper FFG dokáže zabezpečiť finalizáciu blokov aj pri nízkom počte plne funkčných uzlov.

Vďaka byzantskej dohode a vyššej miere počtu validátorov dokáže zachovať proces validácie a finalizácie kontrolných bodov aj pri nižšom počte vygenerovaných blokov, spôsobených tvorbou sieťových partícií.



Obr. 6.6: Vplyv počtu zlyhaných Casper FFG uzlov na finalizáciu blokov. Simulácia 5000 minút protokolu Casper FFG so základným protokolom Bitcoin s blokmi pevnej veľkosti 500 KB. Počet Bitcoin minerov bol pri každej simulácii 64 a počet Casper FFG validátorov bol 1024.



Obr. 6.7: Porovnanie výsledkov v grafe s normalizovanými hodnotami počtu vytvorených blokov. Počet finalizovaných blokov je vyjadrený ako pomer ku celkovému počtu blokov vytvorených daným protokolom. Tvorba blokov v Casper FFG je založená na protokole Bitcoin.

6.2.4 Porovnanie protokolov

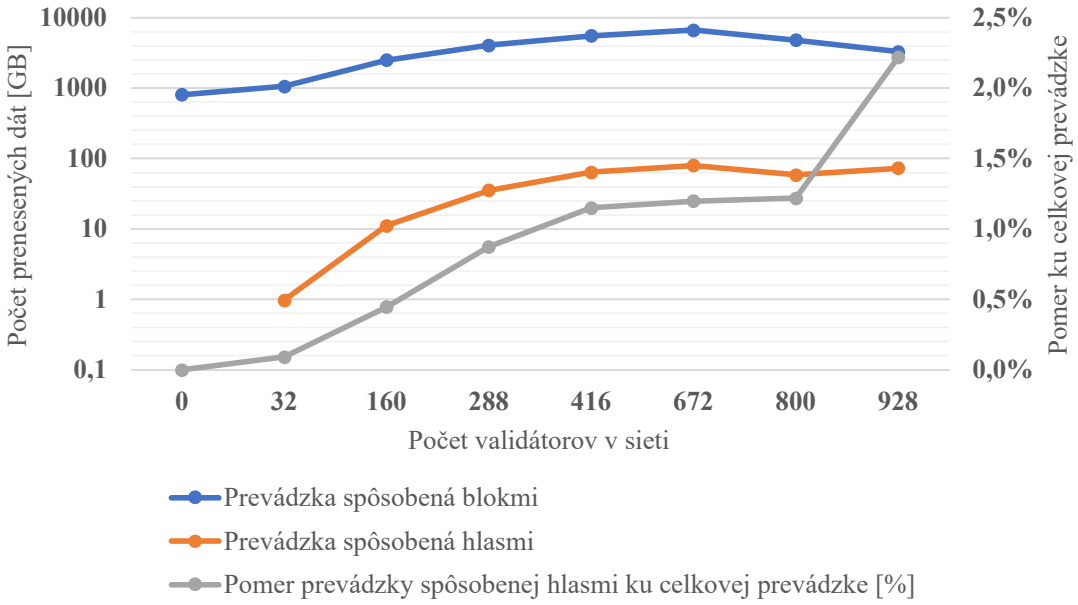
Graf na obrázku 6.7 obsahuje zhrnutie predchádzajúcich výsledkov. Hodnoty sú normalizované podľa jednotlivých protokolov a zobrazené v percentách. Počet zlyhaných uzlov je taktiež uvedený v percentuálnom merítku, pretože pri protokole Casper FFG sme simulovali len 1024 validátorov a v protokoloch Algorand a Gasper sme ich simulovali 1700. Počet finalizovaných blokov je uvedený v percentách z celkového množstva vytvorených blokov daným protokolom.

Z grafu môžeme vidieť, že čisté Proof-of-Stake protokoly Algorand a Gasper boli schopné tvoriť blockchainový ledger aj pri väčšom množstve výpadkov medzi uzlami. U protokolu Casper FFG boli bloky generované protokolom Bitcoin, ktorého značne ovplyvnila tvorba sieťových partícií.

Proces finalizácie blokov prebiehal pri protokoloch Casper FFG a Gasper takmer bez problémov. Výraznejší vplyv malo až vytváranie sieťových partícií. Kým sa darilo vytvárať dostatočný počet blokov na vznik kontrolných bodov, tak finalizácia blokov bola uskutočnená. Ako už bolo vyššie uvedené, zákmit v počte finalizovaných blokov protokolom Gasper na konci grafu je spôsobený dosiahnutím novej epochy tesne pred koncom behu simulácie.

6.3 Casper FFG - sieťová prevádzka spôsobená hlasmi

Casper FFG protokol, ktorý rozširuje iné blockchainové protokoly o schopnosť finalizácie blokov. K tomuto účelu sú využívané hlasy. Pomocou niekoľkých simulácií sme zisťovali, akú veľkú časť sieťovej prevádzky spôsobí práve prenos hlasov medzi účastníkmi pri rôznych počtoch validátorov.



Obr. 6.8: Vplyv počtu Casper FFG validátorov na sieťovú prevádzku zobrazený v logaritmickom merítku. Simulácia 3500 minút protokolu Casper FFG so základným protokolom Bitcoin. Bloky boli pevnej veľkosti 500 KB a intervalom tvorby blokov 15 minút. Počet Bitcoin minerov bol pri každej simulácii 64.

Základom simulácie je rozšírený protokol Bitcoin. V každom behu programu simulovali dobu 3500 minút, pri pevnej dobe generovania blokov 15 minút. V simulovanej sieti bolo 64 minerov, ktorí tvorili bloky o veľkosti 500 KB. Okrem týchto minerov sme do siete vložili validátorov Casper FFG. Prvá simulácia bola vykonaná bez validátorov čím prebehla simulácia bez finalizácie blokov. Ďalší beh programu bol s 32 validátormi a následne sme ich počet v každom behu programu zvýšili o 128 až po 928 validátorov.

Sledovali sme hodnoty počtu prenesených bytov spôsobených prenosom blokov, a taktiež spôsobených hlasovaním. Výsledky testovania sú v grafe na obrázku 6.8. Ako môžeme z grafu vidieť, tak počet validátorov, ktorí rozhodujú o finalizácii kontrolných bodov, a teda aj blokov, má výrazný vplyv na celkovú prevádzku siete. Pri vyšších veľkostiach komisie sa však počet prenesených hlasov v pomere ku celkovej prevádzke postupne stabilizuje. Výkyv pri hodnote 928 validátorov je spôsobený vytvorením novej epochy pri konci simulácie. Pri nižších veľkostiach komisie (do zhruba 300 validátorov) spôsobujú hlasy maximálne 1 % celkovej prevádzky. Pri vyšších počtoch validátorov je prevádzka spôsobená hlasmi taktiež prijateľná, ak berieme do úvahy, že sme týmto krokom schopní zaistiť finalizáciu blokov.

6.4 Testovanie bezpečnosti protokolov

Okrem testovania vlastností implementovaných protokolov, sme sa pozreli aj na ich odolnosť voči útokom. Niektoré z útokov nemalo zmysel testovať, kvôli implementovanej prevencii proti nim. Z tohoto dôvodu a taktiež časovej tiesni boli simulované len útoky na protokol Algorand. Prehľad jednotlivých útokov a dôvody, kvôli ktorým neboli implementované v našom riešení sú nasledovné.

- **Nothing at Stake** je útok založený na možnosti prispievania vo viacerých vetvách ledgera. Implementácia je odolná voči tomuto útoku. Účastníci pri prijímaní hlasov overujú, či už účastník nehlasoval za iný blok na rovnakej úrovni. Taktiež, je v algoritmoch použitá byzantská dohoda, ktorá tento typ útoku potláča.

- **Fake Stake** využíva minimálne stávky na odstavenie iných uzlov pomocou zahltenia ich zdrojov. Avšak, pri simulácii pomocou NS-3 takýto útok nie je možné uskutočniť. Procedúry vykonávané na jednotlivých uzloch v simulovanej sieti sú spracovávané sekvenčne.

- **Dvojnásobné utrácanie** nebolo implementované z dvoch dôvodov. Naše riešenie simulovalo transakcie v blokoch pomocou zmeny veľkosti blokov a skutočný prenos transakcií nebol vykonávaný. Taktiež algoritmy preferujú dlhšie intervaly medzi fázami hlasovania, ktoré zabezpečujú prenos nového bloku ku všetkým účastníkom. Dĺžku intervalu je možné meniť pri spustení programu.

- **LRA** je útok zameraný na budovanie novej vetvy z bloku, ktorý je v histórii tisícov až miliónov blokov. Naše protokoly podporujú finalizáciu blokov, čo znamená, že blok pre vytvorenie novej vetvy by sa musel nachádzať takmer na vrchu ledgera. Taktiež by bol tento krok nevýhodný podobne ako je to u útoku 51 %.

- **Sybil** nebol implementovaný z dôvodu použitia VRF pri protokoloch Algorand a Gasper, čím sme zabezpečili náhodný výber členov komisie. Protokol Casper v reálnych podmienkach zaručuje pomocou trestov, že útoky, ako je práve Sybil, sú až príliš drahé na to, aby ich útočník vykonával [35].

- **Grinding attack** nebol implementovaný z dôvodu použitia VRF v protokoloch Algorand a Gasper, čím sme docielili skutočne náhodný výber členov komisie. U protokolu Casper FFG sme neprišli na spôsob ako ovplyvniť výber hlasujúcich členov.

6.4.1 Algorand - ovplyvňovanie vývoja ledgera

V protokole Algorand bola implementovaná možnosť útoku podobnému útoku 51 %. Útočník sa pomocou stávky snaží ovplyvniť vývoj ledgera. Zamerali sme sa na dva typy hrozieb. Prvou je zamedzenie vloženia bloku do ledgera. Druhá je vloženie neplatného bloku do ledgera. Na vykonanie oboch typov útokov je nutné, aby bol útočník zvolený do komisie vo fáze Soft vote. Pred začiatkom behu simulácie označíme jedného z účastníkov ako útočníka, ktorý sa bude snažiť správať zlomyseľne.

Útočník nemá možnosť kedykoľvek vložiť svoj zlomyseľne vytvorený blok do ledgera a to ani ako návrh, nakoľko výber lídrov je založený na použití VRF, ktoré je zložité ovplyvniť. Za predpokladu, že útočník funguje v zločineckej skupine, kde sa aspoň jednému z členov podarilo navrhnúť neplatný blok, môže tento útočník podporiť navrhnutý blok svojim hlasom vo fáze Soft vote. Neplatný blok bol simulovaný voľbou bloku s najvyššou hodnotou VRF (namiesto voľby najnižšej). Za tento blok náš útočník hlasuje a snaží sa ovplyvniť vývoj blockchainového ledgera.

Správanie simulovaného útočníka je nasledovné. Pri zvolení do komisie vo fáze Soft vote vyberie z návrhov už opísaným spôsobom neplatný blok. Pri hlasovaní je veľkosť jeho stávky pri prvom zvolení vyhodnotená štandardne, ako u ostatných členov komisie. To znamená, že buď vloží pevne nastavenú hodnotu alebo je stavená čiastka vygenerovaná. V každom kole si po obdržaní hlasov ostatných útočníkov uloží celkovú čiastku, ktorá bola stavená. Pri ďalšom zvolení do tejto komisie porovná aký pomer mala jeho posledná stávka ku celkovej stávke v minulom kole. Následne vyhodnotí aká veľká musí byť jeho stávka, aby dosiahla silu zadanú ako parameter pri spustení.

Napríklad, zvoľme silu útoku 0.5 (50 %), celkovú stávku v minulom kole bola 20 000 Algo jednotiek a útočníkova posledná stávka bola 4 500 jednotiek. Na dosiahnutie sily 50 % musí jeho ďalšia stávka mať hodnotu 15 500 Algo jednotiek. Samozrejme, predpokladáme, že útočník má dostatočne veľké finančné zdroje.

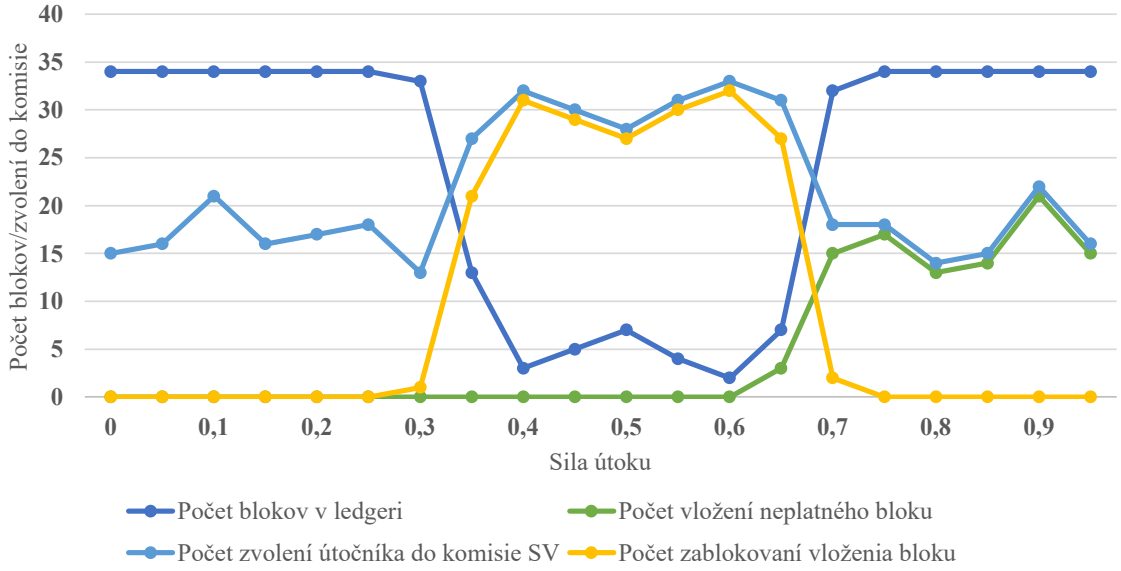
Na začiatku nasledujúceho kola útočník zaznamená výsledok jeho útoku. Ak sa mu podarilo zablokovať vloženie validného bloku do ledgera alebo vložiť neplatný blok s najvyššou hodnotou VRF, tak jeho útok považuje za úspešný. Štatisticky zaznamenávame obe varianty samostatne.

Simulovanie útokov

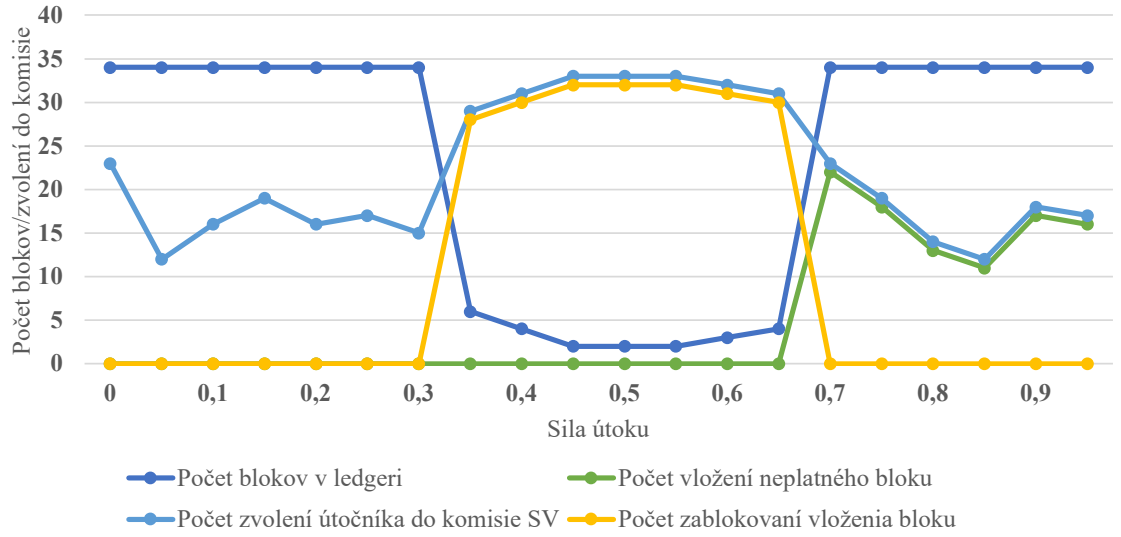
Simulovali sme hrozby s rôzne veľkou silou útoku, ktorá udávala podiel stávky útočníka voči celkovej hodnote ostatných stávok pri hlasovaní. Napríklad, sila útoku $0,3$ udáva, že útočníkova stávka by mala mať veľkosť približne 30 % všetkých stávok pri hlasovaní v danom kole. Tieto simulácie boli vykonané v dvoch scenároch. Prvý mal nastavenú náhodnú voľbu stávky čestných útočníkov a v druhom sme ich stávku pevne definovali na hodnotu 50 Algo jednotiek.

Výsledky simulácií sú zaznačené v grafoch na obrázkoch 6.9 a 6.10. Úspešnosť útočníka nebola ovplyvnená scenárom, nakoľko jeho finančné zdroje boli neobmedzené a záležalo iba na počte zvolení do komisie vo fáze Soft vote. Z tohoto dôvodu sú výsledky oboch scenárov podobné. Z grafov môžeme vidieť, že kým útočníkova stávka nedosiahla aspoň 1/3 všetkých stávok, tak útoky boli neúspešné. Čestné uzly dosiahli v týchto prípa-

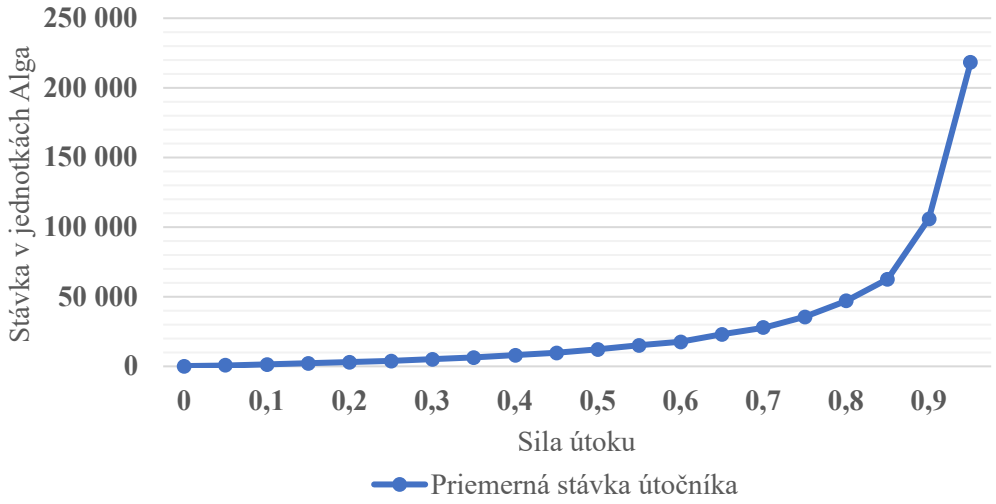
doch vždy potrebné kvórum. V intervale sily útoku od približne 1/3 až do 2/3 začína byť útočník úspešný a blokuje vkladanie platných blokov do ledgera. Pri stávkach nad 2/3 sa darí vkladať neplatné bloky do ledgera.



Obr. 6.9: Vplyv sily útoku na počet úspešných vložení neplatných blokov útočníkom, či zablokovaní vloženia platných blokov do ledgera pri náhodnej veľkosti stávky čestných voličov. Graf zobrazuje aj počet zvolení útočníka do komisie vo fáze Soft vote.



Obr. 6.10: Vplyv sily útoku na počet úspešných vložení neplatných blokov útočníkom, či zablokovaní vloženia platných blokov do ledgera pri pevne danej veľkosti stávky čestných voličov. Graf zobrazuje aj počet zvolení útočníka do komisie vo fáze Soft vote.



Obr. 6.11: Veľkosť priemernej stávky útočníka, ktorú použil pri útokoch s vyžadovanou silou. Sila stávky určuje podiel stávky útočníka voči celkovej hodnote ostatných stávok pri hlasovaní.

Pri vyhodnocovaní údajov získaných zo simulácii útokov bola objavená dôležitá vlastnosť byzantskej dohody v Algorande. Pri zväčšujúcej sa sile útokov stúpa potrebná veľkosť stávky útočníka na vykonanie útoku exponenciálne. Táto vlastnosť je zaznamenaná v grafe na obrázku 6.11, ktorý bol vytvorený z výsledkov simulácie pri pevnej stávke čestných uzlov vo výške 50 Algo jednotiek. Na vykonanie útoku, ktorým by útočník vložil neplatný blok do ledgera, by bolo potrebné veľmi veľké množstvo financií. Táto vlastnosť výrazne zvyšuje bezpečnosť Algorandu a taktiež protokolov, ktoré používajú byzantskú dohodu (PBFT) pre dosiahnutie konsenzu.

Kapitola 7

Záver

V práci boli opísané blockchainové systémy, vrátane ich vlastností, problémov a možného použitia. Približuje spôsob fungovania protokolov založených na Proof-of-Stake modele konsenzu. Konkrétne opisuje protokoly Algorand, Casper, Gasper, Snow White, Stellar a Decred. Na teoretickej úrovni porovnáva ich vlastnosti (tabuľka 3.2) a odolnosť voči rôznym útokom (tabuľka 3.1). V rámci práce bol implementovaný simulátor diskrétnych udalostí slúžiaci pre simuláciu protokolov Algorand, Casper FFG (the Friendly Finality Gadget) a Gasper, založený na nástroji Bitcoin Simulator. Jeho základom je simulátor diskrétnych sieťových udalostí NS-3, ktorý podporuje taktiež paralelnú a distribuovanú simuláciu.

Implementovaný simulátor slúžil pre testovanie vlastností týchto protokolov. Zo simulácií sme získali nasledovné poznatky. Protokoly Algorand a Gasper ovplyvňuje počet účastníkov v sieti minimálne. Doba šírenia lineárne narastala so zvyšujúcim počtom účastníkov, ale udržovala sa na pomerne nízkych hodnotách (obrázok 6.1). Naopak, veľkosť komisie má na dobu propagácie bloku vyšší vplyv (obrázok 6.2) a je preto vhodné voliť nižší prah pre vyhodnotenie VRF, čím dosiahneme nižší počet lídrov. V protokole Gasper samozrejme záleží aj na tom, akým spôsobom je volený líder komisie. Nami implementovaný prístup používal overené VRF funkcie. Výsledok tohoto testu je, že protokoly Algorand a Gasper dosahujú vysokú priepustnosť.

Ďalšie z testov sa zamerali na toleranciu zlyhania uzlov jednotlivými protokolmi. Výsledkom bolo zistenie, že protokoly Algorand aj Gasper sú vysoko odolné voči zlyhaniu uzlov pri tvorbe blockchainového ledgera (tabuľky 6.4 a 6.5). Taktiež, finalizácia blokov protokolmi Gasper aj Casper FFG prebieha značne dobre aj pri vyššom počte zlyhaných uzlov. Ku stratám dochádza až pri vzniku väčšieho počtu sieťových partícií, ku ktorým dochádza najmä pri výpadku viac ako polovici celkového počtu účastníkov (tabuľky 6.5 a 6.6).

Pomocou simulácií sme zisťovali, akú veľkú časť sieťovej prevádzky spôsobí prenos hlasov medzi účastníkmi protokolu Casper FFG pri rôznych počtoch voličov. Z výsledkov (obrázok 6.8) sme zistili, že počet voličov, ktorí rozhodujú o finalizácii kontrolných bodov, a teda aj blokov, má výrazný vplyv na celkovú prevádzku siete. Pri vyšších veľkostiach komisie sa počet prenesených hlasov v pomere ku celkovej prevádzke postupne stabilizuje. Pri nižších veľkostiach komisie (do zhruba 300 voličov) spôsobujú hlasy maximálne 1 % celkovej prevádzky. Pri vyšších počtoch voličov je prevádzka spôsobená hlasmi tiež prijateľná.

V protokole Algorand bola implementovaná možnosť útoku podobnému útoku 51 %, kedy sa útočník pomocou veľkosti stávky snaží ovplyvniť vývoj ledgera. Zamerali sme sa na dva typy hrozieb, a to na zamedzenie vloženia bloku do ledgera a vloženie neplatného bloku do ledgera. Pri oboch typoch útokov sme simulovali dva scenáre. V prvom bola na-

stavená náhodná voľba stávky čestných útočníkov a druhom bola veľkosť stávky pevne definovaná na hodnotu 50 Algo jednotiek. Úspešnosť útočníka ale nebola ovplyvnená scenárom, nakoľko mal neobmedzené finančné zdroje. Výsledky ukázali, že ak útočníkova stávka nedosiahla aspoň 1/3 všetkých stávok, tak útoky boli neúspešné. Čestní účastníci dosiahli v týchto prípadoch vždy potrebné kvórum.

Pri vyhodnocovaní údajov získaných zo simulácii útokov bola objavená dôležitá vlastnosť byzantskej dohody v Algorande. Pri zväčšujúcej sa sile útokov stúpa potrebná veľkosť stávky útočníka na vykonanie útoku exponenciálne (obrázok 6.11), čo výrazne zvyšuje bezpečnosť Algorandu, a taktiež protokolov, ktoré používajú byzantskú dohodu (PBFT) pre dosiahnutie konsenzu.

Vytvorený nástroj je vhodný pre simuláciu rôznych scenárov behu blockchainových protokolov, čím pomáha overovať ich vlastnosti. V budúcnosti by bolo vhodné rozšíriť nástroj o simuláciu vykonávania transakcií medzi štandardnými uzlami a podporu ďalších blockchainových protokolov.

Literatúra

[1] *Algorand - Algorand Protocol Overview*. 2021. Dostupné z: https://www.algorand.com/technology/protocol-overview.

[2] Almeida Martins, M. de. *Blockchain governance: reducing trusted third parties with Decred Project*. 2020. Dizertačná práca. Universidade do Porto.

[3] Amdouni, I., Masucci, A., Baccouch, H. a Adjih, C. *(GETRF Delivrable 3: Network Coding) DragonNet: Specification, Implementation, Experimentation and Performance Evaluation*. 2014. Dizertačná práca. Inria Paris Rocquencourt.

[4] Antonopoulos, A. M. *Mastering Bitcoin: unlocking digital cryptocurrencies*. 1. vyd. Ö'Reilly Media, Inc.", 2014. ISBN 9781449374044.

[5] Aoki, Y., Otsuki, K., Kaneko, T., Banno, R. a Shudo, K. SimBlock: A blockchain network simulator. In: IEEE. *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 2019, s. 325–329. ISBN 9781728118789.

[6] Atlam, H. F., Alenezi, A., Alassafi, M. O. a Wills, G. Blockchain with internet of things: Benefits, challenges, and future directions. *International Journal of Intelligent Systems and Applications*. 1. vyd. 2018, zv. 10, č. 6, s. 40–48.

[7] Banafa, A. IoT and blockchain convergence: benefits and challenges. *IEEE Internet of Things*. -. Január 2017, -.

[8] Barry, N., Losa, G., Mazieres, D., McCaleb, J. a Polu, S. The Stellar Consensus Protocol (SCP). *Internet Engineering Task Force, Internet-Draft draft-mazieres-dinrg-scp-05*. 1. vyd. Marec 2018, č. 1.

[9] Beer, N. a Rahman, N. *Blockchain and The Music Industry: A NEST HQ Documentary*. 2017. Dostupné z: https://www.youtube.com/watch?v=lAMLYP0Nx2A.

[10] Benton, M. C. a Radziwill, N. M. Quality and Innovation with Blockchain technology. *ArXiv preprint arXiv:1710.04130*. -. 2017, -.

[11] Bentov, I., Pass, R. a Shi, E. The Sleepy Model of Consensus. *IACR Cryptol. ePrint Arch.* -. 2016, -, s. 918.

[12] Bentov, I., Pass, R. a Shi, E. Snow White: Provably Secure Proofs of Stake. *IACR Cryptol. ePrint Arch.* -. 2016, zv. 2016, -, s. 919.

[13] Bonello, S. *Ethereum proof of stake. Casper FFG vs. Casper CBC.* Jan 2019. Dostupné z: https://www.chubbydeveloper.com/ethereum-proof-of-stake-casper-ffg-vs-casper-cbc/.

[14] Brenn. *Noobs Guide to Understanding ERC-20 vs ERC-721 Tokens.* Medium, Nov 2018. Dostupné z: https://medium.com/@brenn.a.hill/noobs-guide-to-understanding-erc-20-vs-erc-721-tokens-d7f5657a4ee7.

[15] Brown Cohen, J., Narayanan, A., Psomas, A. a Weinberg, S. M. Formal barriers to longest-chain proof-of-stake protocols. In: Assotiation for Computing Machinery. *Proceedings of the 2019 ACM Conference on Economics and Computation.* 2019, s. 459–473. ISBN 9781450367929.

[16] Buterin, V. a Griffith, V. Casper the friendly finality gadget. *ArXiv preprint arXiv:1710.09437.* -. 2017, -.

[17] Buterin, V., Hernandez, D., Kamphefner, T., Pham, K., Qiao, Z. et al. Combining GHOST and Casper. *ArXiv preprint arXiv:2003.03052.* -. 2020, -.

[18] Cao, M. *How to prevent Sybil Attacks without PoW/PoS.* The Witnet Foundation Blog, Jul 2020. Dostupné z: https://medium.com/witnet/how-to-prevent-sybil-attacks-without-pow-pos-d4d72436dc85.

[19] Caruso, C. *DECRED (DCR).* Medium, Apr 2019. Dostupné z: https://medium.com/@caseycaruso/decred-dcr-1c809eb8bc5d.

[20] Casino, F., Dasaklis, T. K. a Patsakis, C. A systematic literature review of blockchain-based applications: current status, classification and open issues. *Telematics and Informatics.* -. Elsevier. 2019, zv. 36, -, s. 55–81.

[21] Castro, M., Liskov, B. et al. Practical byzantine fault tolerance. In: Laboratory for Computer Science, Massachusetts Institute of Technology. *OSDI.* 1999, sv. 99, č. 1999, s. 173–186. ISBN 9780136386773.

[22] Checkmate. *Decred, Hyper-secure and Unforgeably Scarce.* Medium, Dec 2020. Dostupné z: https://medium.com/@_Checkmatey_/decred-hypersecure-unforgeably-scarce-e076b91a2be.

[23] Chen, J. a Micali, S. Algorand. *ArXiv preprint arXiv:1607.01341.* -. 2016, -.

[24] Chen, J. a Micali, S. Algorand: A secure and efficient distributed ledger. *Theoretical Computer Science.* -. Elsevier. 2019, zv. 777, -, s. 155–183.

[25] Choi, J. *Ethereum Casper 101.* Medium, Feb 2018. Dostupné z: https://medium.com/@jonchoi/ethereum-casper-101-7a851a4f1eb0.

[26] Criddle, C. *Bitcoin consumes 'more electricity than Argentina'.* BBC, Feb 2021. Dostupné z: https://www.bbc.com/news/technology-56012952.

[27] Daian, P., Pass, R. a Shi, E. Snow white: Robustly reconfigurable consensus and applications to provably secure proof of stake. In: Springer. *International Conference on Financial Cryptography and Data Security.* 2019, s. 23–41. ISBN 978-3-030-32100-0.

[28] *Decred - Decred Documentation: Miscellaneous Improvements*. 2020. Dostupné z: https://docs.decred.org/research/miscellaneous-improvements/.

[29] *Decredible - How to mine Decred (DCR)*. 2017. Dostupné z: https://decredible.com/mining/.

[30] Deirmentzoglou, E. *Rewriting History: A Brief Introduction to Long Range Attacks*. -. Positive, 2018.

[31] Deirmentzoglou, E., Papakyriakopoulos, G. a Patsakis, C. A survey on long-range attacks for proof of stake protocols. *IEEE Access*. -. IEEE. 2019, zv. 7, -, s. 28712–28725.

[32] Drijvers, M., Gorbunov, S., Neven, G. a Wee, H. Pixel: Multi-signatures for Consensus. In: {USENIX} Association. *29th USENIX Security Symposium (USENIX Security 20)*. {USENIX} Association, August 2020, s. 2093–2110. ISBN 978-1-939133-17-5. Dostupné z: https://www.usenix.org/conference/usenixsecurity20/presentation/drijvers.

[33] Eha, B. P. *How Barclays Aims to Bring a Billion Unbanked into the Fold*. American Banker, Jan 2017. Dostupné z: https://www.americanbanker.com/news/how-barclays-aims-to-bring-a-billion-unbanked-into-the-fold.

[34] Emmadi, N., Vigneswaran, R., Kanchanapalli, S., Maddali, L. a Narumanchi, H. Practical deployability of permissioned blockchains. In: Springer. *International Conference on Business Information Systems*. 2018, s. 229–243. ISBN 978-3-030-04849-5.

[35] *ETH Staking - All you need to know: Casper Sybil Attack*. 2018. Dostupné z: https://ethstaking.io/guide-to-ethereum-proof-of-stake-and-casper/casper-sybil-attack/.

[36] Eyal, I. a Sirer, E. G. Majority is not enough: Bitcoin mining is vulnerable. In: Springer. *International conference on financial cryptography and data security*. 2014, s. 436–454. ISBN 978-3-662-45472-5.

[37] Faria, C. a Correia, M. BlockSim: Blockchain Simulator. In: IEEE. *2019 IEEE International Conference on Blockchain (Blockchain)*. 2019, s. 439–446. ISBN 978-1-7281-4694-2.

[38] FiachDubh. *Comparing Double Spend Resistance: Decred VS Bitcoin*. Coinmonks, Aug 2020. Dostupné z: https://medium.com/coinmonks/comparing-double-spend-resistance-decred-vs-bitcoin-part-1-330c8081b2a9.

[39] Frankenfield, J. *Proof of Stake (PoS)*. Investopedia, Sep 2020. Dostupné z: https://www.investopedia.com/terms/p/proof-stake-pos.asp.

[40] Fujisaki, E. a Suzuki, K. Traceable ring signature. In: Springer. *International Workshop on Public Key Cryptography*. 2007, s. 181–200. ISBN 978-3-540-71677-8.

[41] Garner, B. *What Is Decred (DCR)?: A Guide on Decentralized Blockchain Governance*. Jan 2019. Dostupné z: https://coincentral.com/decred-lowdown-decentralized-blockchain-governance/.

[42] Gaži, P., Kiayias, A. a Russell, A. Stake-bleeding attacks on proof-of-stake blockchains. In: IEEE. *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*. 2018, s. 85–92. ISBN 978-1-5386-7205-1.

[43] Gervais, A. *Bitcoin Simulator*. 2016. Dostupné z: https://arthurgervais.github.io/Bitcoin-Simulator/get_started.html.

[44] Gervais, A., Karame, G. O., Wüst, K., Glykantzis, V., Ritzdorf, H. et al. On the security and performance of proof of work blockchains. In: Association for Computing Machinery. *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 2016, s. 3–16. ISBN 9781450341394.

[45] Gilad, Y., Hemo, R., Micali, S., Vlachos, G. a Zeldovich, N. Algorand: Scaling byzantine agreements for cryptocurrencies. In: {USENIX} Association. *Proceedings of the 26th Symposium on Operating Systems Principles*. 2017, s. 51–68. ISBN 9781450350853.

[46] Gorbunov, S. *Algorand Releases First Open-Source Code of Verifiable Random Function*. Algorand, Mar 2019. Dostupné z: https://medium.com/algorand/algorand-releases-first-open-source-code-of-verifiable-random-function-93c2960abd61.

[47] Gupta, K. D., Rahman, A., Poudyal, S., Huda, M. N. a Mahmud, M. P. A hybrid pow-pos implementation against 51% attack in cryptocurrency system. In: Institute of Electrical and Electronics Engineers (IEEE). *11th IEEE International Conference on Cloud Computing Technology and Science, CloudCom 2019, 19th IEEE International Conference on Computer and Information Technology, CIT 2019, 2019 International Workshop on Resource Brokering with Blockchain, RBchain 2019 and 2019 Asia-Pacific Services Computing Conference, APSCC 2019*. 2019, s. 396–403. ISBN 9781728150116.

[48] Haber, S. a Stornetta, W. S. How to time-stamp a digital document. In: Springer. *Conference on the Theory and Application of Cryptography*. 1990, s. 437–455. ISBN 978-3-540-38424-3.

[49] Hansen, W. *Decred: An Investment Thesis*. Coinmonks, Aug 2020. Dostupné z: https://medium.com/coinmonks/decred-an-investment-thesis-bf9ba3cd1042.

[50] Homoliak, I., Venugopalan, S., Hum, Q. a Szalachowski, P. A security reference architecture for blockchains. In: IEEE. *2019 IEEE International Conference on Blockchain (Blockchain)*. 2019, s. 390–397. ISBN 978-1-7281-4694-2.

[51] *Horizen Academy - Blockchain as a Data Structure*. 2019. Dostupné z: https://academy.horizen.io/technology/expert/blockchain-as-a-data-structure/.

[52] Jax.Network. *Double spend attacks in the PoS network*. Jax.Network Blog, Sep 2020. Dostupné z: https://medium.com/jax-network/transaction-finality-in-the-pos-network-9700d659a38c.

[53] Jepson, C. DTB001: Decred Technical Brief. *Available at https://cryptorating.eu/whitepapers/Decred/decred.pdf Additional information available at https://www.decred.org*. -. 2015, -.

[54] Kiayias, A., Miller, A. a Zindros, D. Non-interactive proofs of proof-of-work. In: Springer. *International Conference on Financial Cryptography and Data Security*. 2020, s. 505–522. ISBN 978-3-030-51280-4.

[55] Kim, M., Kwon, Y. a Kim, Y. Is Stellar as secure as you think? In: IEEE. *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. 2019, s. 377–385. ISBN 978-1-7281-3027-9.

[56] Lab, D. S. *"Fake Stakeättacks on chain-based Proof-of-Stake cryptocurrencies*. Medium, Feb 2019. Dostupné z: https://medium.com/@dsl_uiuc/fake-stake-attacks-on-chain-based-proof-of-stake-cryptocurrencies-b8b05723f806.

[57] *Ledger - What is Proof-of-Work*. Október 2019. Dostupné z: https://www.ledger.com/academy/blockchain/what-is-proof-of-work.

[58] LeFebvre, D. *Decred Privacy: Taking The Long Road*. Decred, Sep 2019. Dostupné z: https://medium.com/decred/decred-privacy-taking-the-long-road-62d218223db6.

[59] Mazieres, D. The stellar consensus protocol: A federated model for internet-level consensus. *Stellar Development Foundation*. -. Citeseer. 2015, zv. 32, -.

[60] McCorry, P., Shahandashti, S. F. a Hao, F. A smart contract for boardroom voting with maximum voter privacy. In: Springer. *International Conference on Financial Cryptography and Data Security*. 2017, s. 357–375. ISBN 978-3-319-70972-7.

[61] Micali, S., Rabin, M. a Vadhan, S. Verifiable random functions. In: IEEE. *40th annual symposium on foundations of computer science (cat. No. 99CB37039)*. 1999, s. 120–130. ISBN 0-7695-0409-4.

[62] Moindrot, O. a Bournhonesque, C. Proof of Stake Made Simple with Casper. *ICME, Stanford University*. -. 2017, -.

[63] Möser, M., Böhme, R. a Breuker, D. An inquiry into money laundering tools in the Bitcoin ecosystem. In: Ieee. *2013 APWG eCrime researchers summit*. 2013, s. 1–14. ISBN 978-1-4799-1158-5.

[64] Paganini, P. *INVDoS, a severe DoS issue in Bitcoin core remained undisclosed for two years*. Sep 2020. Dostupné z: https://securityaffairs.co/wordpress/108188/hacking/invdos-dos-bitcoin-core.html.

[65] Panja, S. a Roy, B. K. A secure end-to-end verifiable e-voting system using zero knowledge based blockchain. *IACR Cryptol. ePrint Arch.* -. 2018, zv. 2018, -, s. 466.

[66] Piscini, E., Dalton, D. a Kehoe, L. *Blockchain and Cyber Security. Let's Discuss*. Sep 2017. Dostupné z: https://www2.deloitte.com/content/dam/Deloitte/tr/Documents/technology-media-telecommunications/Blockchain-and-Cyber.pdf.

[67] Raczyński, M. *What Is The Fastest Blockchain And Why? Analysis of 43 Blockchains*. Dec 2020. Dostupné z: https://alephzero.org/blog/what-is-the-fastest-blockchain-and-why-analysis-of-43-blockchains/.

[68] Red, R. *The role of Decred voters in defending against majority attacks*. Medium, Jan 2019. Dostupné z: https://richardred.medium.com/the-role-of-decred-voters-in-defending-against-majority-attacks-ec658af0a8fd.

[69] Red, R. *Hybrid PoW/PoS Consensus Explained*. Binance Academy, Dec 2020. Dostupné z: https://academy.binance.com/en/articles/hybrid-pow-pos-consensus-explained.

[70] Red, R. a Academy, B. *Hybrid PoW/PoS Consensus Explained*. Binance Academy, Jan 2020. Dostupné z: https://academy.binance.com/en/articles/hybrid-pow-pos-consensus-explained.

[71] Salimitari, M. a Chatterjee, M. A survey on consensus protocols in blockchain for iot networks. *ArXiv preprint arXiv:1809.05613*. -. 2018, -.

[72] Shbair, W. M., Steichen, M., François, J. a State, R. BlockZoom: Large-Scale Blockchain Testbed. In: IEEE. *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. 2019, s. 5–6. ISBN 978-1-7281-1329-6.

[73] Skidanov, A. Fast Finality and Resilience to Long Range Attacks with Proof of Space-Time and Casper-like Finality Gadget. *Near Protocol*. -. 2019, -.

[74] Sorensen, D. *CBC Casper Proof-of-Stake Consensus Algorithm Liveness Issue*. Pyrofex, Mar 2019. Dostupné z: https://medium.com/pyrofex/cbc-casper-proof-of-stake-consensus-algorithm-liveness-issue-c965e88d163e.

[75] Stark, J. *Making Sense of Blockchain Smart Contracts*. CoinDesk, Jun 2016. Dostupné z: https://www.coindesk.com/making-sense-smart-contracts.

[76] Stoykov, L., Zhang, K. a Jacobsen, H.-A. Vibes: fast blockchain simulations for large-scale peer-to-peer networks. In: {USENIX} Association. *Proceedings of the 18th ACM/IFIP/USENIX Middleware Conference: Posters and Demos*. 2017, s. 19–20. ISBN 9781450352017.

[77] *Streamr Network - The decentralized platform for real-time data*. 2021. Dostupné z: https://streamr.network/.

[78] Szabo, N. Formalizing and securing relationships on public networks. *First Monday*. 1. vyd. 1997, -.

[79] Szalachowski, P., Reijsbergen, D., Homoliak, I. a Sun, S. Strongchain: Transparent and collaborative proof-of-work consensus. In: {USENIX} Association. *28th {USENIX} Security Symposium ({USENIX} Security 19)*. 2019, s. 819–836. ISBN 9781939133069.

[80] Tasca, P. a Tessone, C. J. Taxonomy of blockchain technologies. Principles of identification and classification. *ArXiv preprint arXiv:1708.04872*. -. 2017, -.

[81] *The Block Box - Real Estate Tokenization, How Blockchain Technology Could Revamp and Streamline an Entire Industry*. Dec 2020. Dostupné z: https://theblockbox.io/blog/real-estate-tokenization-how-blockchain-technology-could-revamp-and-streamline-an-entire-industry/.

[82] Toueg, S. Randomized byzantine agreements. In: Association for Computing Machinery. *Proceedings of the third annual ACM symposium on Principles of distributed computing*. 1984, s. 163–178. ISBN 0897911431.

[83] Vbuterin. *Responding to 51% attacks in Casper FFG*. Oct 2019. Dostupné z: https://ethresear.ch/t/responding-to-51-attacks-in-casper-ffg/6363/5.

[84] Venugopalan, S., Homoliak, I., Li, Z. a Szalachowski, P. BBB-Voting: 1-out-of-k Blockchain-Based Boardroom Voting. *ArXiv preprint arXiv:2010.09112*. -. 2020, -.

[85] voidburn. *Stellar - Vulnerabilities of Stellar*. Jun 2019. Dostupné z: https://www.reddit.com/r/Stellar/comments/c02pvc/vulnerabilities_of_stellar/?utm_source=share.

[86] Wijaya, D. A., Liu, J. K., Steinfeld, R., Liu, D. a Yu, J. On the unforkability of monero. In: Association for Computing Machinery. *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*. 2019, s. 621–632. ISBN 9781450367523.

[87] Wüst, K. a Gervais, A. Do you need a blockchain? In: IEEE. *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*. 2018, s. 45–54. ISBN 978-1-5386-7205-1.

[88] Yaga, D., Mell, P., Roby, N. a Scarfone, K. Blockchain technology overview. *ArXiv preprint arXiv:1906.11078*. -. 2019, -.

[89] Yocom Piatt, J. *On-Chain Atomic Swaps*. Sep 2017. Dostupné z: https://blog.decred.org/2017/09/20/On-Chain-Atomic-Swaps/.

[90] Zamboglou, D. *Stellar: Explained*. Data Driven Investor, Apr 2019. Dostupné z: https://medium.com/datadriveninvestor/stellar-explained-e5d28fbec238.

[91] Zamfir, V. *Introducing casper, the friendly ghost*. 2015. Dostupné z: https://blog.ethereum.org/2015/08/01/introducing-casper-friendly-ghost.

[92] Zheng, Z., Xie, S., Dai, H.-N., Chen, X. a Wang, H. Blockchain challenges and opportunities: A survey. *International Journal of Web and Grid Services*. -. Inderscience Publishers (IEL). 2018, zv. 14, č. 4, s. 352–375. ISSN 1741-1114.

[93] Zheng, Z., Xie, S., Dai, H., Chen, X. a Wang, H. An overview of blockchain technology: Architecture, consensus, and future trends. In: IEEE. *2017 IEEE international congress on big data (BigData congress)*. 2017, s. 557–564. ISBN 978-1-5386-1997-1.

[94] Zoi, A. *Study of consensus protocols and improvement of the Federated Byzantine Agreement (FBA) algorithm*. 2019. Diplomová práca. Universitat Politècnica de Catalunya.

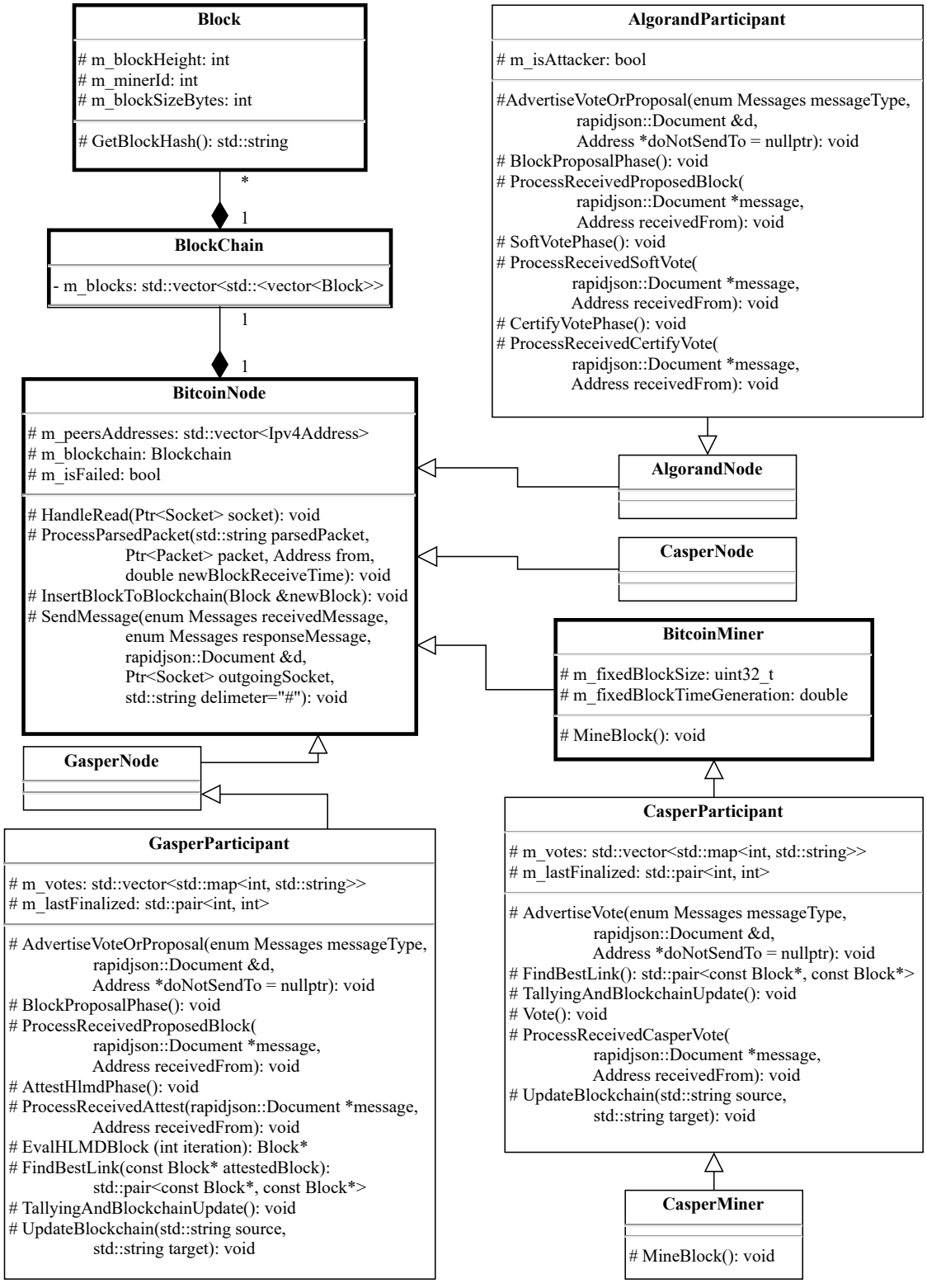
[95] Česák, D. *Pohled na kryptoměnu: Algorand*. mCoins, Jan 2020. Dostupné z: https://www.mcoins.cz/pohled-na-kryptomenu-algorand.

Príloha A

Obrázky a tabuľky

```
Total Stats:
Total Nodes = 50
Miners (participants) = 0
Failed Nodes = 0
Average Connections/node = 7.96
Average Connections/miner = 0
Mean Block Receive Time = 5.87228 or 0min and 5.87228s
Mean Block Propagation Time = 0.329514s
Max Block Propagation Time = 1.59886s
Median Block Propagation Time = 0.280738s
10% percentile of Block Propagation Time = 0.248829s
25% percentile of Block Propagation Time = 0.25974s
75% percentile of Block Propagation Time = 0.429616s
90% percentile of Block Propagation Time = 0.48388s
Miners Mean Block Propagation Time = 0s
Miners Median Block Propagation Time = 0s
Mean Block Size = 531.6 KB
Total Blocks = 30
Stale Blocks = 0 (0%)
Loss = 0
The size of the longest fork was 0 blocks
There were in total 0 blocks in forks
Mean Block Proposal Committee Size = 25
Mean Soft Vote Committee Size = 50
Mean Certify Vote Committee Size = 50
Mean NonAttacker Soft vote Stake = 536.273
Mean Attacker Soft vote Stake = 0
Attackers = 0
Mean attacker member of Soft vote committee = 0x
Total Successful Insertions = 0
Total Successful Insertion Blocks = 0
The average received BLOCK messages were 2.7 GB (49.909%)
The average sent BLOCK messages were 2.7 GB (49.909%)
The average received VOTE messages were 5.0 MB (0.0910011%)
The average sent VOTE messages were 5.0 MB (0.0910011%)
Total average traffic due to BLOCK messages = 5.4 GB (99.818%)
Total average traffic due to VOTE messages = 10.1 MB (0.182002%)
Total average traffic/node = 5.4 GB (266325 Kbps and 195.1 KB/block)
Total traffic due to BLOCK messages = 269.2 GB (99.818%)
Total traffic due to VOTE messages = 502.7 MB (0.182002%)
Total traffic = 269.7 GB
25.1007s per generated block
```

Obr. A.1: Ukážka výstupných štatistík simulátoru. Položky výstupu sa môžu meniť na základe typu protokolu, ktorý bol simulovaný. Na ukážke je výstup simulácie protokolu Algorand.



Obr. A.2: Diagram tried slúžiacich pre simuláciu protokolov Algorand, Casper FFG a Gasper. Hrubo zvýraznené sú pôvodné triedy nástroja Bitcoin Simulator. V diagrame sú zaznamenané iba najdôležitejšie vlastnosti a metódy tried.

Zoznam tabuliek

Zoznam obrázkov

Príloha B

Obsah priloženého CD

Priložené CD obsahuje nasledujúce položky:

- text diplomovej práce,
- zdrojové kódy pre LaTeX,
- zdrojové kódy programu,
- dáta s výsledkami simulácií.