Strategie sobeckého těžení v Bitcoinu

Teorie her (THE) 2020

31. prosince 2020 Vladimír Dušek, xdusek27

Obsah

1 Úvod

Bitcoin je nestátní, decentralizovaný, digitální finanční systém založený na kryptografii. Ze své podstaty funguje bez jakékoliv centrální autority. Narozdíl od státních *fiat* měn[1], které jsou řízeny centrálními bankami. Bitcoinová síť je čistě *peer-to-peer*, kde si mezi sebou uživatelé vyměňují hodnotu bez potřeby jakéhokoliv prostředníka [2]. Je navržen tak, aby nikdo, ani autor, jiní jednotlivci, skupiny či státy, nemohl měnu ovlivňovat, padělat, konfiskovat, devalvovat, cenzurovat nebo ovlivňovat růst peněžní zásoby (ta je pevně definovaná a všem dopředu známá). V protokolu neexistuje žádný centrální ani nijak privilegovaný bod, všichni uživatelé operují jako uzly na stejné úrovni.

Bitcoin byl představen 31. října 2008, kdy člověk nebo skupina lidí pod pseudonymem Satoshi Nakamoto zaslal e-mail kryptografickému mailing listu s odkazem na *white paper* [3]. Bitcoinová síť pak byla spuštěna 9. ledna roku 2009, kdy byl vytěžen první blok [5]. Ke konci roku 2020 se tržní kapitalizace Bitcoinu pohybuje kolem 500 miliard amerických dolarů [6]. Výpočetní výkon (*hash rate*) celé sítě činí zhruba $130 \times 10^{18}\,\mathrm{H/s}$ (*hashes per second*) [7].

Článek *Majority is not Enough: Bitcoin Mining is Vulnerable* [1] ukazuje, že v bitcoinovém protokolu nejsou ekonomické incentivy těžařů nastaveny korektně. Představuje strategii tzv. sobeckého těžení (z anglického *Selfish Mining Strategy*), která vede k většímu zisku, než by mělo těžaři náležet na základě jeho výpočetní síly pokud by se choval tak, jak protokol předpokládá.

V kapitole 2 jsou vysvětleny základy fungování Bitcoinu jakožto decentralizovaného finančního systému. Ty jsou nutné k pochopení strategie sobeckého těžení a následnému sestavení modelu. Ten je formalizován v kapitole 3 a analyzován v kapitole 4. Výsledky analýzy jsou vyhodnoceny v kapitole 5 a na závěr jsou stručně okomentovány.

2 Bitcoin jako decentralizovaný finanční systém

Klíčovým bodem fungování Bitcoinu je veřejná distribuovaná „účetní kniha“, zvaná *blockchain*. V té jsou zaznamenány všechny transakce, které kdy byly v bitcoinové síti uskutečněny. Každý uzel disponuje svojí vlastní kopií, kterou si aktualizuje. Velmi zjednodušeně jsou transakce zapisovány následovně: z adresy X je odesláno N bitcoinů na adresu Y. Uživatel disponuje typicky mnoha adresami, v ideálním případě novou adresou pro každou platbu.

2.1 Autentizace

V systému je bezpochyby nutné autentizovat vlastníky jednotlivých mincí. To je zajištěno pomocí asymetrické kryptografie. Jednotlivé mince uživatelů jsou přiřazeny k pseudonymům – bitcoinovým adresám. Adresa je pouze derivátem veřejného

[1] Státní peníze vznikající z příkazu. Z lanského *fiat* = budiž. Nejčastěji používán jako termín pro papírové peníze, nekryté drahými kovy [4].

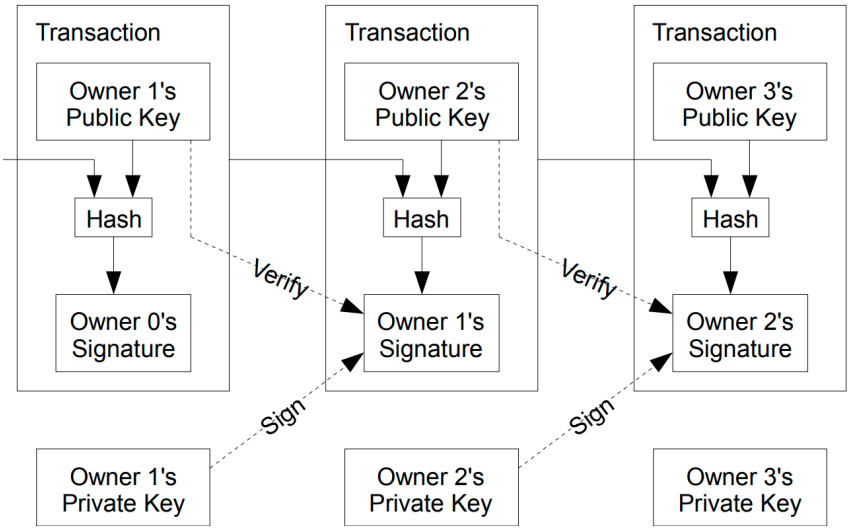
klíče. Výhradně ten, kdo disponuje soukromým klíčem, který náleží danému veřejnému klíči, může vygenerovat validní digitální podpis. Tím se v síti autentizuje jako vlastník mince na dané adrese a může vytvořit platnou transakci.

Protokol je navržen tak, aby každý uživatel mohl disponovat unikátní adresou pro každou svoji minci. Klíče k jednotlivým adresám uživatelé mají uložené ve svých peněženkách. Peněženka je software pro správu klíčů a vytváření transakcí. Vytvořená transakce je zaslána konkrétnímu bitcoinovému uzlu, který, je-li transakce validní, ji zašle dalším uzlům.

2.2 Opětovné utracení

Opětovné utracení (*replay attack*) znamená, že uživatel svoji konkrétní minci utratí vícekrát. Nic uživateli nebrání, vytvořit transakci s validním podpisem, který utrácí minci, kterou již někdy dřívě utratil. V systému je nutné rozpoznávat, které mince již byly utraceny a které nikoliv.

V bitcoinovém protokolu je toto řešeno sledováním historie mincí. Jednotlivé mince řetězíme za sebe, tak jak v jednotlivých transakcích putují napříč různými adresami. Známe tedy historii všech mincí od jejich vzniku (viz sekce 2.5) až po jejich současnost – tzv. neutracený transakční výstup (UTXO – *Unspent transaction output*). Výhradně UTXO může figurovat jako vstup transakce, tím je zajištěno, že vlastník nemůže stejnou „minci“ utratit vícekrát.



Obrázek 1: Bitcoinové transakce [2].

2.3 Dvojí utracení

Dvojí utracení (*double spend attack*) je takový problém, kdy uživatel disponující UTXO X, vytvoří transakci, ve které utrácí UTXO X na adresu Y a zároveň jinou transakci, ve které utrácí UTXO X na adresu Z. Každou transakci pošle jinému uzlu. Ten transakci zvaliduje a pošle dále do sítě. Jelikož jsme si doposud nepředstavili žádný systém časových razítek, síť není schopná vyhodnotit, která transakce

je platná (byla zadána jako první) a která je neplatná (má jako vstup již utracené UTXO).

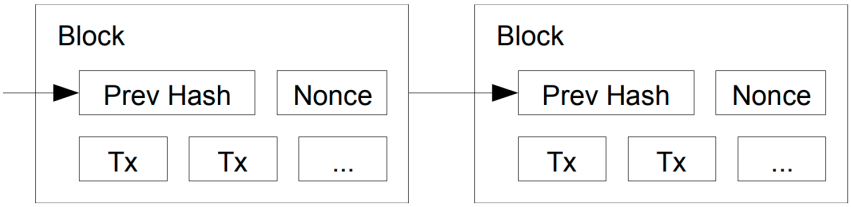
Ochrana před dvojím utracením je řešena právě *blockchainem*. Transakce jsou agregovány do bloků, které jsou zhruba každých 10 minut uzavírány. Každý blok obsahuje kromě samotných transakcí také hash (SHA-256) předchozího bloku. Tím jsou bloky zřetězeny za sebe (odtud název blockchain) a vzniká chronologické uspořádání. Pokud by někdo chtěl měnit informace v již uzavřeném bloku (blok nad kterým staví již další blok), musí přepočítat hashe všech následujících bloků[2].

2.4 Důkaz prací

V bitcoinovém protokolu je konsenzus nastaven tak, že nejdelší blockchain je pravda. V tuto chvíli by však bylo možné změnit jakoukoliv již zapsanou transakci, přepočítat hashe všech následujících bloků a prohlásit tuto verzi blockchainu za nejdelší řetězec a tím pádem pravdu.

Abychom tomuto zamezili, musí být zápis do blockchainu nákladná operace. Toho je v Bitcoinu docíleno tak, že na hash bloku jsou kladeny nějaké nároky. Konkrétně takové, že výsledný hash musí být malé číslo. K tomuto účelu je v bloku místo pro náhodné číslo, tzv. *nonce*. Pokud chce někdo přidat nový blok do blockchainu, musí najít takové číslo, které způsobí, že výsledný hash bloku bude odpovídat současným nárokům sítě. Tyto nároky se v průběhu času mění podle výpočetního výkonu celé sítě tak, aby bloky byly uzavírány v průměru každých 10 minut.

Tento koncept se nazývá důkaz prací (z anglického *proof of work*).



Obrázek 2: Chronologické zřetězení bloků za sebe – *blockchain* [2].

2.5 Těžení

Proces hledání správných *nonce* a zapisování nových bloků do blockchainu se nazývá těžení (*Bitcoin Mining*). Entity, které tuto činnost vykonávají se nazývají těžaři (*Bitcoin Miners*). Bitcoinový protokol motivuje uživatele tuto činnost provádět tak, že za uzavření bloku poskytuje odměnu v podobě nově vzniklých bitcoinů. Ty jsou vytvořeny v rámci tzv. *coinbase* transakce, což je vždy první transakce v bloku.

Těžař poskládá transakce dalších uživatelů do bloku, přidá hash předcházejícího bloku a generuje náhodná čísla jako potenciální *nonce*. Z toho pomocí hashovací funkce SHA-256 spočítá hash a zkontroluje, zda náhodou výsledný hash není dostatečně malé číslo. Pokud měl štěstí a je, blok je vytěžen a těžař ho rozpošle dalším uživatelům. Ti ho přijmou a začnou těžit další bloky nad ním.

[2]Z tohoto hlediska funguje podobným způsobem verzovací distribuovaný systém Git.

Aby těžaři získávali odměnu v pravidelných a více predikovatelných intervalech, začaly se formovat tzv. **těžařské pooly** (*Mining Pool*). Několik těžařů spojí svůj výpočetní výkon a figurují jako jedna entita. Pokud některý z nich vytěží blok, odměnu si rozdělí s ostatními proporciálně dle jejich výpočetního výkonu.

Odměna za vytěžený blok se skládá ze dvou složek. Kromě již zmíněné *coinbase* transakce s nově vzniklými bitcoiny, dostane těžař také transakční odměny ze všech transakcí v bloku. Uživatelé, chtějí-li motivovat těžaře, aby zahrnuli jejich transakci do bloku, platí transakční poplatky. Ty jsou definovány jako rozdíl mezi transakčními vstupy a transakčními výstupy, viz rovnice 1.

$$in_1 + \ldots + in_n - out_1 - \ldots - out_m = tx\_fee \tag{1}$$

3 Model

V této kapitole je formalizován model, který popisuje systém těžení a koncept těžebních poolů. Dále je představena strategie sobeckého těžení.

3.1 Těžení a těžební pooly

Systém je tvořen množinou těžařů $N = 1, \ldots, n$. Každý těžař disponuje výpočetním výkonem p_i takovým, že $\sum_{i=1}^{n} p_i = 1$. Těžař si může vybrat libovolný blok z blockchainu nad kterým začne těžit. Následující blok vytěží za časový interval, který odpovídá exponenciálnímu rozdělení se střední hodnotou $E(X) = (6p_i)^{-1}$ hod. Předpis funkce hustoty pravděpodobnosti tohoto rozdělení je zachycen na rovnici 2. Předpokládáme, že těžaři se chovají racionálně – snaží se maximalizovat svoji odměnu a to i za cenu odchýlení se od protokolu.

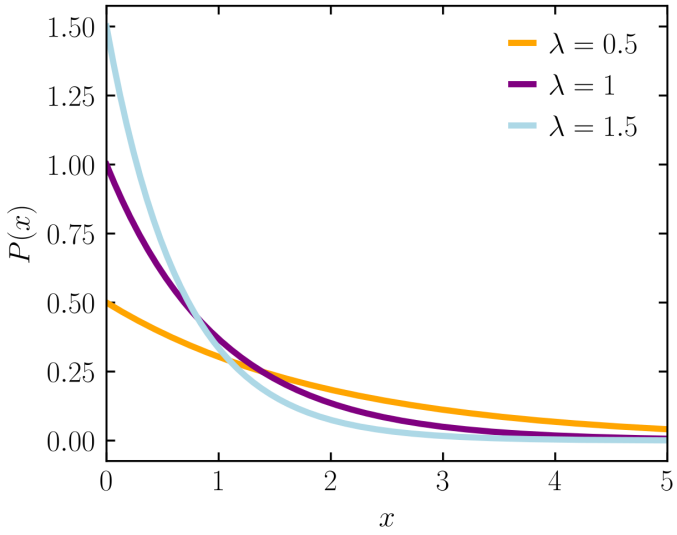
$$f(x) = \begin{cases} 6p_i \cdot e^{-6p_i \cdot x} & ; x > 0 \\ 0 & ; x \leq 0 \end{cases} \tag{2}$$

Skupina těžařů se může spojit a zformovat tzv. *pool*, který vystupuje v síti jako jedna entita. *Pool*, který se skládá z učástníků $M \subseteq N$ disponuje výpočetním výkonem $\sum_{i \in M} p_i \leq 1$. Utržený zisk *pool* rozděluje mezi své členy proporciálně dle jejich výpočetního výkonu. Očekávaný zisk *poolu* je poměr očekávaného počtu vytěžení bloků *poolem* ku celkovému počtu bloků v nejdelším řetězci. Odměna za vytěžený blok je $R = X + Y$, kde X je počet nově vzniklých bitcoinů a Y je suma všech transakčních poplatků v bloku. Počet nových bitcoinů za blok se každých 210000 vytěžených bloků zmenšuje na polovinu, to v modelu není nutné zohledňovat.

3.2 Strategie sobeckého těžení

Jak je ukázáno v kapitole 4, strategie sobecké těžení (z anglického *Selfish Mining*) umožňuje těžebnímu aktérovi získat větší odměnu než odpovídá jeho výpočetnímu výkonu. Nezávisí na tom, zda strategii uplatňuje jednolivec či těžební *pool*.

Pro zjednodušení uvažujme rozdělení těžebních aktérů do dvou skupin. Jedna skupina jako *pool* S se strategií sobeckého těžení, který disponuje minoritním výpočetním výkonem ($p_S < 0,5$). A zbytek sítě jako skupinu poctivých těžařů (*honest*



Obrázek 3: Funkce hustoty pravděpodobnosti pro různá exponenciální rozdělení ($\lambda = E(X)^{-1} = 6p_i$).

miners) s ostatním výpočetním výkonem $(1 - p_S)$. Není podstatné, zda zbytek figuruje jako jedna entita, nebo několik menších skupin.

Podstata sobeckého těžení spočívá v přimětí poctivých těžařů ke zbytečnému těžení. Spotřebovávání zdrojů na aktivitu, za kterou nebudou odměněni. Konkrétně způsobit, že poctiví těžaři budou těžit takovou větev blockchainu, která ve výsledku nebude součástí nejdelšího řetězce. Taková situace může nastat i ve standardních podmínkách, kdy se všichni chovají poctivě, avšak strategie sobeckého těžení bude způsobovat, že poctiví těžaři budou takové situaci vystaveni častěji.

Sobečtí těžaři tohoto cíle dosahují specifickou strategií odhalování jimi vytěžených bloků. Pokud *pool* S vytěží blok výšky n, nepošle ho do sítě ostatním ihned, nýbrž si ho nechá pro sebe a blok $n+1$ těží sám. Disponuje tak nejdelším řetězcem, o kterém nikdo další neví. Zbytek sítě dále pokračuje na veřejné větvi blockchainu a těží blok n. Situace se dále může vyvíjet různě, avšak jelikož *pool* S disponuje pouze minoritou výpočetního vykonu, jeho soukromá větev blockchainu nebude nejdelší do nekonečna. *Pool* S s tím kalkuluje a uvážlivě odhaluje bloky ze své soukromé větve tak, aby způsoboval, že poctiví těžaři opustí veřejnou kratší větev, nad kterou nějaký čas těžili. To znamená že investovali výpočetní výkon zbytečně.

Se znalostí této intuice je možné plně specifikovat algoritmus sobeckého těžení.

Listing 1: Algoritmus sobeckého těžení – inicializační fáze.

```
public chain = publicly known blocks
private chain = publicly known blocks
privateBranchLenght = 0
do: Mine at the head of the private chain
```

Listing 2: Algoritmus sobeckého těžení – situace, kdy sobecký *pool* vytěžil blok.

```
diff = length(private chain) - length(public chain)
```

```
append new block to the private chain
privateBranchLenght += 1

// selfish pool wins due to the lead of 1
if diff == 0 and privateBranchLenght == 2:
    do: publish all the private chain
    privateBranchLenght = 0

do: Mine at the head of the private chain
```

Listing 3: Algoritmus sobeckého těžení – situace, kdy zbytek sítě vytěžil blok.

```
diff = length(private chain) - length(public chain)
do: append the new block to the public chain

// they win
if diff == 0:
    private chain = public chain
    privateBranchLenght = 0

// now it's the same length, luck decides
else if diff = 1:
    do: publish last block of the private chain

// selfish pool wins due to the lead of 1
else if diff = 2:
    do: publish all the private chain
    privateBranchLenght = 0

// diff > 2
else:
    do: publish first unpublished block in private block

do: Mine at the head of the private chain
```

Strategie je řízena těžebními událostmi v síti. Rozhodnutí sobeckého *poolu* závisí na rozdílu délek privátní a veřejné větve blockchainu. V následujících odstavcích je chování sobeckého těžaře popsáno v konkrétních situací.

Veřejná vetěv blockchainu je delší než soukromá větev sobeckého *poolu*. V tomto případě sobecký pool přijme nový blok vytěžený zbytkem sítě a začne těžit nad ním. Vzhledem k distribuci výpočetního výkonu bude tato situace nastávat nejčastěji.

Sobecký těžař vytěží blok a má náskok jednoho bloku nad veřejným blockchainem. Místo toho, aby ho naivně ihned zveřejnil a poslal do sítě, si ho nechá pro sebe a těží nad ním sám. Následovat mohou dvě situace.

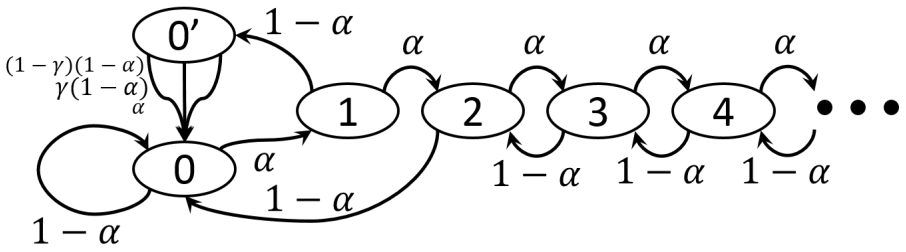
Další blok vytěží zbytek sítě, v tomto případě nastává *fork* blockchainu a následně hraje klíčovou roli latence mezi jednotlivými uzly. V momentě, kdy se informace o nově vytěženém bloku někým dalším dostane k sobeckému těžaři, ihned zveřejní svůj již dřívě vytěžený blok. V tuto chvíli nastává dočasná nekonzistence mezi uzly v síti. Část sítě těží nad blokem vytěženým sobeckým těžařem a část nad blokem vytěženým zbytkem sítě. Kdo vytěží další blok vyhrál, má nejdelší blockchain a zbytek sítě akceptuje jeho verzi.

Pokud další blok vytěží opět sobecký těžař, dosahuje pohodlného náskoku dvou bloků. V takovém případě sobecký pool pokračuje dále v těžení nad svojí větví blockchainu. Další blok zveřejní vždy, pokud zbytek sítě vytěží blok na veřejné větvi a tím sníží jeho náskok. Jelikož sobecký pool má minoritní výpočetní výkon, v určitém bodě bude jeho náskok snížen natolik, že zveřejní svoji soukromou větev celou. Jelikož jeho větev je nejdelší, zbytek sítě ji akceptuje. Sobecký pool získá odměnu všech bloků ve své větvi, zatímco zbytek sítě spotřebovával elektřinu zbytečně.

4 Analýza

V této kapitole je analyzován model, který byl popsán v kapitole 3. Sobecký pool disponuje výpočetním výkonem α a zbytek sítě (poctivý těžaři) $1 - \alpha$.

Na obrázku 4 je definován potenciální náskok sobeckého poolu jako stochastický proces pomocí Markovova řetězce. Přechody mezi stavy odpovídají vytěžení bloku buďto sobeckým poolem nebo poctivými těžaři. Výskyt těchto událostí odpovídá exponenciálním rozdělení tak, jak bylo popsáno v sekci 3.1. Názvy jednotlivých stavů reprezentují náskok sobeckého poolu. Ten je definován jako rozdíl délek privátní větve sobeckého poolu a veřejné větve blockchainu. Pro shodnou délku obou větví existuje stav 0 a $0'$. Stav 0 je stav, ve kterém není blockchain rozvětven. Existuje pouze jeden nejdelší veřejný blockchain a všichni těžaři těží nad stejným blokem.



Obrázek 4: Možný náskok sobeckého poolu popsán jako stochastický proces pomocí Markovova řetězce.

Stav $0'$ je stav, ve kterém mají obě větve délku od rozvětvení 1. Sobecký pool vytěžil blok první, ale nechal si ho pro sebe. Následně někdo ze zbytku sítě vytežil blok stejné výšky na veřejném blockchainu. Jakmile se sobecký pool o tomto doví, ihned publikuje svůj dříve vytěžený blok. Vzniká rozvětvení (*fork*) na veřejném blockchainu. Sobecký pool těží svoji větev. Zbytek sítě podle bitcoinového protokolu těží tu větev, o které se dozvěděl jako první. Parametrem γ definujeme část sítě, která těží větev sobeckého poolu a $1 - \gamma$ část sítě, která těží větev vytěženou někým z poctivých těžařů. Tento parametr ovlivňuje především latence mezi jednotlivými uzly v síti.

Pro stavy $s = 0, 1, 2, ...$ platí, že sobecký pool vytěží blok s pravděpodobností α a tím zvýší svůj náskok na $s + 1$. Ve stavech $s = 3, 4, ...$ platí, že blok vytěží někdo z upřímných těžařů s pravděpodobností $1 - \alpha$ a tím sníží náskok sobeckého poolu na $s - 1$. Pokud někdo z poctivých těžařů vytěží blok ve stavu $s = 2$, sobecký pool zveřejní svoji větev. Jelikož se jedná o nejdelší blockchain, zbytek sítě opustí

svoji práci a dle protokolu přijmou větev sobeckého poolu. Pokud někdo z poctivých těžařů vytěží blok ve stavu $s = 1$ dostaváme se do již zmíněného stavu $0'$. Z toho vedou 3 přechody do stavu $s = 0$: (1) sobecký pool vytěží blok nad svoji předchozí privátní větví (pravděpodobnost α), (2) někdo z poctivých těžařů vytěží blok nad větví sobeckého poolu (pravděpodobnost $\gamma \cdot (1 - \alpha)$), (3) někdo z poctivých těžařů vytěží blok nad veřejnou větví blockchainu (pravděpodobnost $(1 - \gamma) \cdot (1 - \alpha)$).

4.1 Pravděpodobnost jednotlivých stavů

Nyní spočítejme pravděpodobností rozdělení jednotlivých stavů napříč Markovovým řetězcem. Na základě toho bude možné spočítat očekávané odměny ze sobeckého těžení a porovnat je s odměnami při standardním chování.

$$init\_prob\_dist = (P_0, P_{0'}, P_1, P_2, ...) = (1, 0, 0, 0, ...) \tag{3}$$

$$transition\_matrix = \left(\begin{array}{c|ccccc} S_0 & 1-\alpha & 0 & \alpha & 0 & \cdots \\ S_{0'} & 1 & 0 & 0 & 0 & \cdots \\ S_1 & 0 & 1-\alpha & 0 & \alpha & \cdots \\ S_2 & 1-\alpha & 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array}\right) \tag{4}$$

$$init\_prob\_dist \cdot transition\_matrix^n = steady\_state \tag{5}$$

$$steady\_state \cdot transition\_matrix = steady\_state \tag{6}$$

Na rovnici 3 je definována výchozí rozdělení pravděpodobnosti všech stavů. Na rovnici 4 je přechodová matice. Je třeba najít stabilní stav (*steady state* nebo také *stable state*). Ten je definován na rovnici 5 pro $n \to \infty$. Pro *steady_state* musí platit rovnice 6. Na základě těchto znalostí, je možné sestavit následující rovnice:

$$\begin{aligned} P_0 &= P_0 \cdot (1 - \alpha) + P_2 \cdot (1 - \alpha) + P_{0'} \\ P_{0'} &= P_1 \cdot (1 - \alpha) \\ P_1 &= P_0 \cdot \alpha \\ \forall k \geq 2 : P_k &= P_{k-1} \cdot \alpha + P_{k+1} \cdot (1 - \alpha) \\ 0 &= \sum_{i=0}^{\infty} P_i + P_{0'} \end{aligned} \tag{7}$$

Vyřešením těchto rovnic získáme *steady state* – výsledné rozdělení pravděpodobnosti všech stavů. Vyřešení celé soustavy je v příloze článku [1].

$$P_0 = \frac{\alpha - 2\alpha^2}{\alpha \cdot (2\alpha^3 - 4\alpha^2 + 1)} \tag{8}$$

$$P_{0'} = \frac{(1-\alpha) \cdot (\alpha - 2\alpha^2}{1 - 4\alpha^2 + 2\alpha^3} \tag{9}$$

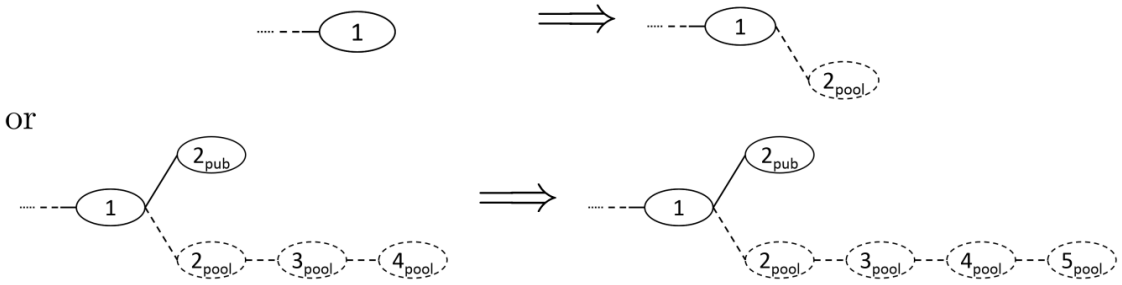
$$P_1 = \frac{\alpha - 2\alpha^2}{2\alpha^3 - 4\alpha^2 + 1} \tag{10}$$

$$\forall k \ \geq 2 : P_k = \left(\frac{\alpha}{1-\alpha}\right)^{k-1} \cdot \frac{\alpha - 2\alpha^2}{2\alpha^3 - 4\alpha^2 + 1} \tag{11}$$

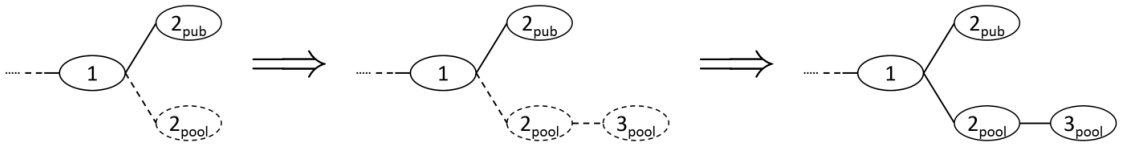
4.2 Očekávané odměny

S výsledným rozdělením pravděpodobnosti napříč stavovým prostorem již můžeme spočítat očekávané odměny pro sobecký pool a poctivé těžaře. Odměna náleží za vytěžený blok, který je součástí nejdelšího blockchainu. Dále jsou analyzovány odměny pro každý případ, který může nastat.

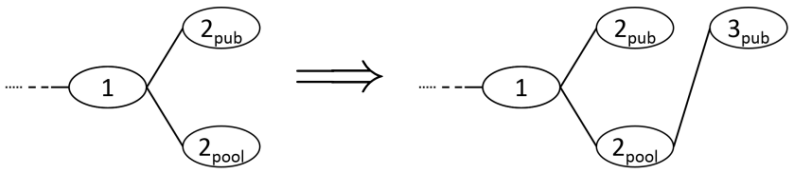
(a) *Jakýkoliv stav, kromě toho, že délka obou větví je* 1*, sobecký pool vytěží blok.* Tím zvyší svůj náskok z k na k+1. Blok přidá do své privátní větve blockchainu. Odměna bude rozdělena později.



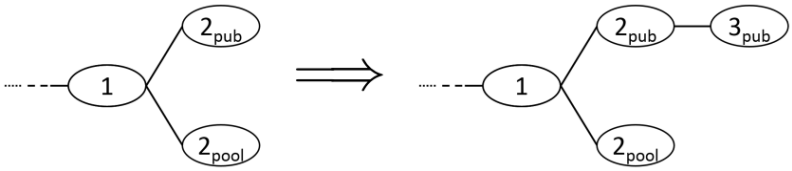
(b) *Délka obou větví je* 1*, sobecký pool vytěží blok.* Tím dosáhne nejdelšího blockchainu, zveřejní svoji větev a získává odměnu za 2 vytěžené bloky.



(c) *Délka obou větví je* 1*, ostatní vytěží blok nad blokem sobeckého poolu.* Sobecký pool i zbytek sítě získají každý odměnu za 1 vytěžený blok – ostatní za no nově vytěžený blok a sobecký pool za předka.

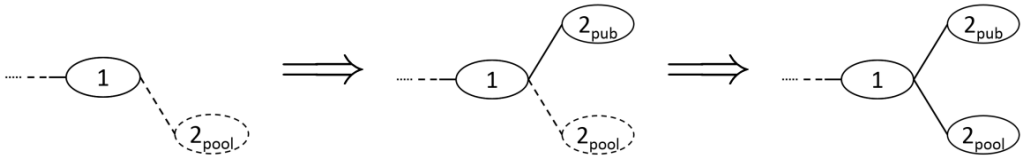


(d) *Délka obou větví je* 1*, ostatní vytěží blok nad blokem ostatních.* Poctiví těžaři získávají odměnu za 2 vytěžené bloky.

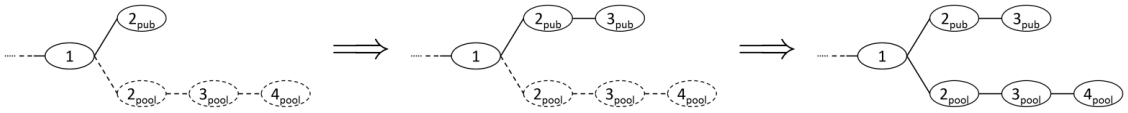


(e) *Žádná privátní větev neexistuje, všichni v síti těží nad stejným blokem, upřímní těžaři vytěží blok.* Upřímní těžaři získávají odměnu za 1 vytěžený blok. Všichni začínají těžit nad novým blokem.

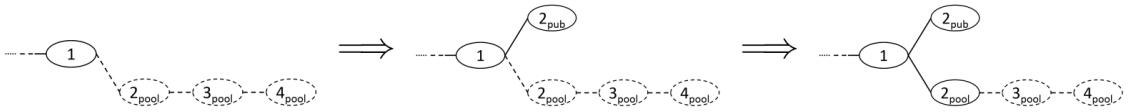
(f) *Sobecký pool vede o* 1*, ostatní vytěží blok.* Sobecký pool zveřejní svoji větev a vzniká *fork*. Sobecký pool těží nad svoji větví a zbytek sítě se rozdělí podle parametru γ – která verze se k ním dostane dříve, tu těží. Odměna z této situace bude rozdělena později, podle toho, která větev „vyhraje“.



(g) *Sobecký pool vede o* 2*, ostatní vytěží blok.* Zbytek sítě sníží náskok sobeckého poolu na 1. Ten jakmile se to dozví, zveřejní celou svoji větev. Jelikož má nejdelší blockchain, všichni přijmou jeho verzi. Sobecký pool obdrží odměnu za 2 vytěžené bloky.



(h) *Sobecký pool vede o více než* 2*, ostatní vytěží blok výšky* k*.* Zbytek sítě sníží náskok sobeckého poolu, ten zůstává aspoň 2. Nový blok k od upřímných těžařů nakonec nebude součástí nejdelší blockchainu. Jakmile sobecký pool zveřejní celou svoji větev, ostatní přijmou jeho verzi a on obdrží odměnu za všechny bloky. V tento okamžik sobecký pool zveřejňuje svůj blok k a získává odměnu za 1 vytěžený blok.



Nyní je možné spočítat očekávané odměny sobeckého poolu a poctivých těžařů na základě rozdělení pravděpodobnosti stavů a pravděpodobnosti jednotlivých přechodů.

$$r_{honest} = \overbrace{1 \cdot P_{0'} \cdot \gamma(1-\alpha)}^{\text{case (c)}} + \overbrace{2 \cdot P_{0'} \cdot (1-\gamma)(1-\alpha)}^{\text{case (d)}} + \overbrace{1 \cdot P_0 \cdot (1-\alpha)}^{\text{case (e)}} \tag{12}$$

$$r_{selfish} = \overbrace{2 \cdot P_{0'} \cdot \alpha}^{\text{case (b)}} + \overbrace{1 \cdot P_{0'} \cdot \gamma(1-\alpha)}^{\text{case (c)}} + \overbrace{2 \cdot P_2 \cdot (1-\alpha)}^{\text{case (g)}} + \overbrace{1 \cdot P_i \cdot (1-\alpha)}^{\text{case (h)}} \tag{13}$$

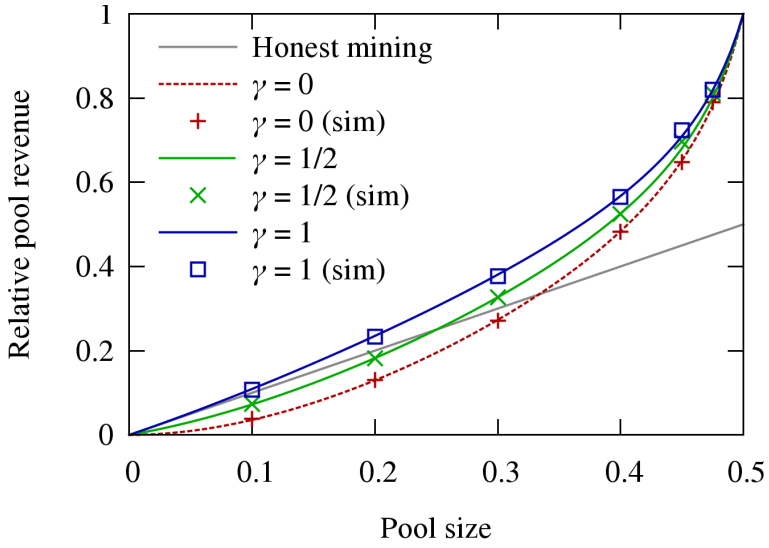
Podle očekávání záměrné větvení blockchainu sobeckým poolem způsobilo, že poctiví těžaři těžili významnou dobu bloky, které skončily mimo nejdelší blockchain. Toto chování mimo jiné způsobí, že celkové tempo generování nových bloků bude nižší. Obtížnost těžby (nároky na hash bloku) se bude snižovat, aby průměrná doba vytěžení bloku byla 10 minut.

Dosazením rozdělení pravděpodobnosti stavů z rovnic 8, 9, 10, 11 do rovnic 12 a 13 je možné spočítat poměr vytěžených bloků sobeckým poolem ku celkovému počtu vytěžených bloků.

$$R_{selfish} = \frac{r_{selfish}}{r_{selfish} + r_{honest}} = \cdots = \frac{\alpha(1-\alpha)^2(4\alpha + \gamma(1-2\alpha)) - \alpha^3}{1 - \alpha(1 + (2-\alpha)\alpha)} \tag{14}$$

4.3 Simulace

Pro ověření teoretické analýzy byly výsledky porovnány se simulátorem Bitcoinového protokolu. V simulaci bylo těžení nahrazeno Monte Carlo simulátorem. Ten simuluje těžení bloků bez skutečného počítání hashů SHA-256 pro různé *nonce*. V simulaci figurovalo 1000 těžařů, z toho 1000α zformovalo sobecký pool. Podrobnější popis simulace je popsán v článku [1] v sekci 4.3. Obrázek 5 ukazuje, že výsledky simulace odpovídají výsledkům z teoretické analýzy.



Obrázek 5: Odměny ze strategie sobeckého těžení pro různé hodnoty parametru γ v porovnání s poctivým těžením. Výsledky simulace odpovídají teoretické analýze. A ukazují, že strategie sobeckého těžení vede od určité hranice k větším odměnám. Hranice zavisí na parametru γ.

4.4 Dopady parametrů α a γ

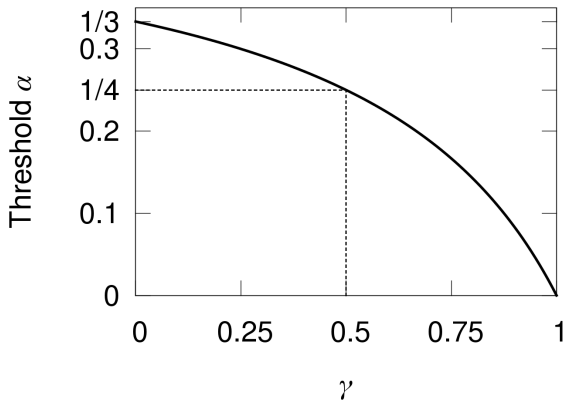
Pokud odměna sobeckého poolu definovaná rovnicí 14 je vyšší než α, tak je strategie sobeckého těžení pro pool výhodnější. Jednotliví těžaři získají vyšší odměnu v sobeckém poolu než by získali v poctivých poolech. Nerovnice je platná pouze pro $0 \leq \alpha \leq 0,5$. Nerovnice 16 je pouze zjednodušení nerovnice 15.

$$\frac{\alpha(1-\alpha)^2(4\alpha+\gamma(1-2\alpha))-\alpha^3}{1-\alpha(1+(2-\alpha)\alpha)} > \alpha \tag{15}$$

Theorem 1 *Pro konkrétní hodnoty parametru γ, pool velikosti α získá odměnu větší než náleží jeho relativnímu výpočetnímu výkonu pro α v rozsahu:*

$$\frac{1-\gamma}{3-2\gamma} < \alpha < \frac{1}{2} \tag{16}$$

Toto odpovídá výsledkům z obrázku 5, kde je zobrazena odměna poolu pro rozdílne hodnoty parametru γ pro velikosti poolu od 0 do $0,5$.



Obrázek 6: Hranice pro jaké hodnoty parametrů α a γ se vyplatí provozovat strategii sobeckého těžení.

Sklon výše odměny poolu R_{pool} jako funkce velikosti poolu je větší než hranice, kdy se vyplatí provozovat strategii sobeckého těžení. Z toho vyplývá druhý teorém:

Theorem 2 *Pro pool provozující strategii sobeckého těžení, se odměna každého člena poolu zvyšuje s velikostí poolu pro pool větší než je hranice výhodnosti sobeckého těžení nad poctivým těžením.*

5 Výsledky

V kapitole 4 bylo ukázáno, že pokud výpočetní výkon poolu překročí určitou hranici, pool by mohl zvýšit svoji celkovou odměnu zvolením strategie sobeckého těžení (teorém 1). V takovém případě by racionální těžaři měli preferovat připojit se k sobeckému poolu, kde mohou zvýšit svoji odměnu. Navíc v zájmu členů sobeckého poolu je přijímat nové členy, jelikož i to zvyšuje jejich odměny (teorém 2).

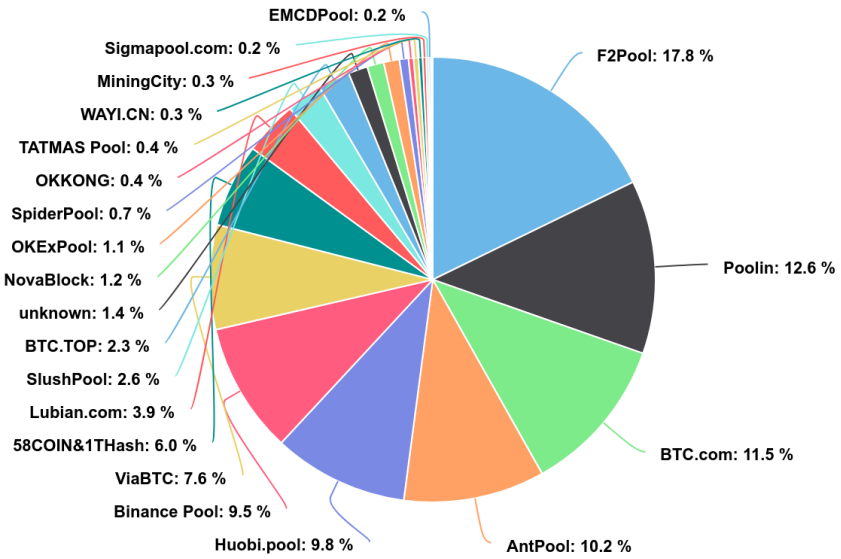
Sobecký pool tímto způsobem bude nabývat na velikosti, až se stane majoritní ($> 0,5$). Jakmile se nějaký pool stane majoritním, získává absolutní kontrolu nad blockchainem. Strategie sobeckého těžení se stává nerelevantní, jelikož zbytek sítě již není rychlejší než pool. V takovém případě majoritní pool může pohodlně ignorovat bloky zbytku sítě a vytěžit všechny bloky sám. Tím získá odměny ze všech bloků. Může také rozhodovat jaké transakce budou vytěženy a jaké ne – cenzura transakcí. V tento moment přestává bitcoinová síť být decentralizovaná.

V článku [1] v kapitole 6 je navrhnuto, jak by bitcoinový protokol mohl být upraven a být více rezistentní proti strategie sobeckého těžení. Jedná se však pouze o posunutí hranice, kdy se sobecké těžení vyplatí, ne kompletní vyřešení problému.

6 Závěr

Z výsledků vyplývá že bitcoinový protokol není incentivně kompatibilní (*incentive compatible*). Pro racionálního hráče je výhodnější přidat se k sobeckému poolu a tím zvýšit svoji odměnu. Pro těžaře sobeckého poolu je racionální přijímat nové členy mezi sebe, čímž zvýší svoji odměnu. Dominantní strategie jdou proti protokolu a vedou k centralizaci.

Článek *Majority is not Enough: Bitcoin Mining is Vulnerable* [1] byl publikován již v listopadu 2013. Do současnosti se tržní kapitalizace zvedla z tehdejších 10×10^9 amerických dolarů na 500×10^9 amerických dolarů [6]. Výpočetní výkon dokonce z 5×10^{15} H/s na 130000×10^{15} H/s [7]. Na bitcoinovém fóru bitcointalk se tento typ slabiny diskutoval již v roce 2010[3].



Obrázek 7: Rozdělení výpočetního výkonu bitcoinové sítě v prosinci 2020 [9].

Formulace modelu, kdy těžař maximalizuje pouze počet utržených bitcoinů se zdá příliš zjednodušující. Těžaři své náklady (pořízení hardwaru, elektřina) pravděpodobně hradí v dolarech (či jiné státní měně). Musí tedy bitcoiny prodat, z toho

[3] `https://bitcointalk.org/index.php?topic=2227.msg30083#msg30083`

zaplatit náklady a zbytek je jeho utržený zisk. Pokud by těžař uplatňoval strategii sobeckého těžení, tak sice za jistých podmínek utrží více bitcoinu, nicméně jeho celkový zisk se nezvýší, jelikož by bitcoin tratil na ceně. Těžař by spíše maximalizoval počet utržených bitcoinů × cena jednoho bitcoinu (např. v amerických dolarech). Strategií sobeckého těžení by sice jednu složku tohoto výrazu zvýšil, ale druhou velmi pravděpodobně snížil.

Naopak se ukazuje, že v zájmu těžařů je v první řadě aby síť fungovala a lidé v ní měli důvěru. V roce 2014 pool Ghash.io pravděpodobně přesáhl hranici 50% výpočetního výkonu sítě. Mohl tak teoreticky ignorovat práci všech ostatních, všechny bloky vytěžit sám a získat výrazně vyšší odměnu. Nic takového ale nenastalo. Samotný pool vydal prohlášení, že v žádném případě nemá v úmyslu Bitcoinu ublížit a že v budoucnu už nepřesáhne 40 %. Těžaři poolu začali dobrovolně přecházet do jiných poolů [8]. V současnosti žádný pool nepřesahuje těžební výkon 20 % (viz obrázek 7).

7 Reference

[1] Ittay Eyal and Emin Gun Sirer: *Majority is not Enough: Bitcoin Mining is Vulnerable*. Dostupné online na `https://www.cs.cornell.edu/~ie53/publications/btcProcFC.pdf`, navštíveno 28. 12. 2020.

[2] Satoshi Nakamoto: *Bitcoin: A Peer-to-Peer Electronic Cash System*. Dostupné online na `https://bitcoin.org/bitcoin.pdf`, navštíveno 28. 12. 2020.

[3] Satoshi Nakamoto Institute: *Cryptography Mailing List; Bitcoin P2P e-cash paper*. Dostupné online na `https://satoshi.nakamotoinstitute.org/emails/cryptography/1/`, navštíveno 28. 12. 2020.

[4] Josef Tětek: *Bitcoin: Odluka peněz od státu*. Září 2020. Dostupné online na `https://odlukapenez.cz`, navštíveno 28. 12. 2020.

[5] Satoshi Labs: *Bitcoin Explorer*. Dostupné online na `https://btc1.trezor.io/block/00000000839a8e6886ab5951d76f411475428afc90947ee320161bbf18eb6048`, navštíveno 28. 12. 2020.

[6] YCharts: *Bitcoin Market Cap*. Dostupné online na `https://ycharts.com/indicators/bitcoin_market_cap`, navštíveno 28. 12. 2020.

[7] YCharts: *Bitcoin Network Hash Rate*. Dostupné online na `https://ycharts.com/indicators/bitcoin_network_hash_rate`, navštíveno 28. 12. 2020.

[8] Coindeck: *Ghash.io: We Will Never Launch a 51 % Attack Against Bitcoin*. Dostupné online na `https://www.coindesk.com/ghash-io-never-launch-51-attack`, navštíveno 30. 12. 2020.

[9] BTC.com: *Pool Distribution*. Dostupné online na `https://btc.com/stats/pool`, navštíveno 31. 12. 2020.