



دانشگاه شهید بهشتی

دانشکده مهندسی و علوم کامپیوتر

## ارائه‌ی یک روش رای‌گیری امن مبتنی بر بلاک‌چین

پایان‌نامه کارشناسی ارشد مهندسی کامپیوتر  
گرایش نرم‌افزار

نگارش

شروین حاجی‌اسمعیلی

استاد راهنما

دکتر مقصود عباس‌پور

تابستان ۹۷



دانشگاه شهید بهشتی  
دانشکده مهندسی و علوم کامپیوتر

پایان نامه کارشناسی ارشد مهندسی کامپیوتر - گرایش نرم افزار  
تحت عنوان:

ارائه‌ی یک روش رای گیری امن مبتنی بر بلاک چین

در تاریخ                      پایان نامه دانشجو،                      ، توسط کمیته تخصصی داوران مورد بررسی و تصویب نهایی قرار گرفت.

امضا	نام و نام خانوادگی	۱- استاد راهنما اول:
امضا (در صورت نیاز)	نام و نام خانوادگی	۲- استاد راهنما دوم:
امضا (در صورت نیاز)	نام و نام خانوادگی	۳- استاد مشاور:
امضا	نام و نام خانوادگی	۴- استاد داور (داخلی):
امضا	نام و نام خانوادگی	۵- استاد داور (خارجی):
امضا	نام و نام خانوادگی	۶- نماینده تحصیلات تکمیلی:

## با سپاس و قدردانی از

پدران و مادرانی که خود را فدای تربیت فرزندان خود کردند و  
اساتید و معلمانی که در تمام دوران زندگی، راهنمای جانسوز ما بودند.

آوردن این صفحه اختیاریست.

کلیه حقوق مادی مترتب بر نتایج مطالعات،  
ابتکارات و نوآوری‌های ناشی از تحقیق موضوع  
این پایان‌نامه متعلق به دانشگاه شهید بهشتی  
می‌باشد.

## به نام خدا

نام و نام خانوادگی: شروین حاجی اسمعیلی

عنوان پایان نامه: ارائه‌ی یک روش رای‌گیری امن مبتنی بر بلاک‌چین

استاد راهنما: دکتر مقصود عباس‌پور

اینجانب شروین حاجی اسمعیلی تهیه‌کننده پایان‌نامه کارشناسی ارشد حاضر، خود را ملزم به حفظ امانت‌داری و قدردانی از زحمات سایر محققین و نویسندگان بنابر قانون Copyright می‌دانم. بدین وسیله اعلام می‌نمایم که مسئولیت کلیه مطالب درج شده با اینجانب می‌باشد و در صورت استفاده از اشکال، جداول و مطالب سایر منابع، بلافاصله مرجع آن ذکر شده و سایر مطالب از کار تحقیقاتی اینجانب استخراج گشته است و امانت‌داری را به صورت کامل رعایت نموده‌ام. در صورتی که خلاف این مطلب ثابت شود، مسئولیت کلیه عواقب قانونی با شخص اینجانب می‌باشد.

نام و نام خانوادگی: شروین حاجی اسمعیلی

تاریخ و امضا:

تقدیم به

رهجویان علم و فناوری و دوستداران علم و دانش

آوردن این صفحه اختیاریست.

# فهرست مطالب

۱	مقدمه	۱
۲	۱.۱ فرایند رای گیری ایده آل	۲
۲	۲.۱ سیستم های رای گیری سنتی	۲
۳	۳.۱ مشکلات و چالش های رای گیری الکترونیک	۳
۵	۴.۱ انگیزه و هدف	۵
۷	۲ تعریف مفاهیم	۷
۸	۱.۲ بلاک چین	۸
۸	۲.۲ درخت مرکب	۸
۹	۱.۲.۲ انواع بلاک چین	۹
۱۰	۳.۲ اثبات های بی دانش	۱۰
۱۰	۱.۳.۲ مثال شهودی	۱۰
۱۱	۲.۳.۲ اثبات های بی دانش بدون تعامل	۱۱
۱۱	ZK-SNARK	۱۱
۱۳	ZK-STARK	۱۳
۱۴	۳ کارهای پیشین	۱۴
۱۵	۱.۳ اعتماد	۱۵

۱۵	توافق	۱.۱.۳
۱۵	توافق	
۱۶	اثبات سهم	
۱۶	Protocol Consensus Ripple	
۱۶	Protocol Consensus Stellar	
۱۷	کاربردهای بلاک چین	۲.۱.۳
۱۷	ارز دیجیتال	
۱۸	سازمان‌های توزیع‌شده‌ی خودکار	۳.۱.۳
۱۹	شناسایی	۴.۱.۳
۱۹	رای‌گیری الکترونیک	۲.۳
۲۰	رای‌گیری الکترونیک متمرکز	۱.۲.۳
۲۰	رای‌گیری الکترونیک توزیع‌شده	۲.۲.۳
۲۱	رای‌گیری بدون بلاک چین	
۲۱	رای‌گیری با بلاک چین عمومی	
۲۲	رای‌گیری با بلاک چین خصوصی	
۲۲	اثبات‌های بی‌دانش	۳.۳
۲۳	کاربردها	۱.۳.۳
۲۳	پرداخت ناشناس	
۲۴	فاز آماده‌سازی	۲.۳.۳

۲۵	روش پیشنهادی	۴
۲۶	تعریف عناصر	۱.۴
۲۶	شرایط مسئله	۲.۴
۲۷	فرضیات مسئله	۳.۴



۲۸	مثال شهودی	۴.۴
۲۹	فرایند رای گیری از نگاه کاربر	۵.۴
۲۹	قبل از رای گیری	۱.۵.۴
۲۹	در حوزه ی رای گیری	۲.۵.۴
۳۰	فرایند رای گیری از دید حوزه	۶.۴
۳۰	تراکنش ها	۱.۶.۴
۳۰	تراکنش ثبت	
۳۰	تراکنش شمارش	
۳۱	فرایند ثبت رای کاربر	۲.۶.۴
۳۲	شمای کلی	۷.۴
۳۳	اضافه شدن بلوک	۱.۷.۴
۳۴	مقادیر اولیه برای ZK-SNARK	۲.۷.۴
۳۴	توافق	۳.۷.۴
۳۴	قضیه ی CAP	
۳۵	Practical Byzantine Fault Tolerance	

## ۵ تحلیل و ارزیابی ۳۸

۳۹	پیاده سازی	۱.۵
----	------------	-----

## مراجع ۴۰

## فهرست تصاویر

۹	یک درخت مرکب	۱.۲
۱۲	یک نمونه مدار محاسباتی	۲.۲
۳۲	فرایند ثبت رای در حوزه	۱.۴
۳۳	شمای منطقی سیستم	۲.۴
۳۶	روش PBFT	۳.۴

## فهرست جداول

۹	.....	انواع بلاک چین	۱.۲
۳۶	.....	روش توافق	۱.۴
۳۹	.....	تنظیمات libsnark	۱.۵

## چکیده

از سال ۲۰۰۹ تاکنون، با فراگیری بیت کوین شاهد افزایش کاربردهای بلاک چین و سیستم‌های توزیع شده و بدون نیاز به اعتماد بوده‌ایم. بعد از انتشار بستر اتریوم تا به امروز قراردادهای هوشمند توزیع شده در این بستر رشد قابل توجهی داشته‌اند. به همین دلیل بررسی امنیتی قراردادهای این بستر اهمیت ویژه‌ای دارد. همچنین با ساخت این بستر فرصت مناسبی است تا سرویس‌های بیشمار مبتنی بر اعتماد فعلی خود و راه‌های جایگزین آن‌ها را در بستر بلاک چین بررسی کنیم.

در این تحقیق ابتدا به معرفی ارزهای دیجیتال و نحوه‌ی کارکرد آن‌ها می‌پردازیم، سپس تحقیقات امنیتی خود بسترها و کاربردهای آن‌ها را بررسی کرده و در نهایت به کاربرد آن‌ها برای رای‌گیری دیجیتال و چالش‌های این کار اشاره خواهیم کرد.

**واژگان کلیدی:** بلاک چین، اتریوم، امنیت، قرارداد هوشمند، رای‌گیری

# فصل ۱

## مقدمه

امنیت در رای گیری همواره یک مسئله پیچیده بوده است که نیازمند یک فرد قابل اعتماد برای برگزاری و یک پروتکل امن برای جلوگیری از تقلب یا اشتباه در فرایند آن است. سیستم های رای گیری الکترونیک از سال ۱۹۶۰ وجود داشتند و اولین استفاده بزرگ از آن ها در چند ایالت آمریکا در سال ۱۹۶۴ برای انتخابات ریاست جمهوری بود. رای گیری الکترونیک به سادگی می تواند هزینه برگزاری انتخابات را از طریق سادگی شمارش کاهش دهد.

## ۱.۱ فرایند رای گیری ایده آل

شروط فرایند رای گیری ایده آل عبارت است از:

- هر فرد واجد شرایط دقیقاً یک بار بتواند رای دهد.
- هیچ کسی نتواند به جای فرد دیگری رای دهد.
- هیچ فردی مجبور به رای دادن نشود.
- هیچ فردی مجبور به رای دادن به کاندیدای خاصی نشود.
- از شمارش هر رای اطمینان حاصل شود.
- نتیجه ی آرا ناشناس باقی بماند.
- بسته به نیاز بتوان نتایج لحظه ای انتخابات را (بدون آسیب به شرط های قبلی) دید.

## ۲.۱ سیستم های رای گیری سنتی

در رای گیری غیر الکترونیکی معمولاً فرایند به شکل زیر است:

فرد برای رای دادن به یکی از حوزه های رای گیری مراجعه کرده و با ارائه ی مدارک شناسایی خود یک برگه ی رای دریافت می کند. برگه رای دارای دو بخش است: قسمتی که برای ردیابی با اطلاعات شخصی فرد پر می شود و یک قسمت بی نام که فرد کاندیدای مورد نظر خود را در آن ثبت کرده و در یک صندوق می اندازد.

با بررسی مدارک شناسایی، شرط دوم فرایند رای گیری ایده آل تایید شده و با ثبت شدن اطلاعات فرد به عنوان یک رای دهنده از رای دادن دوباره ای او جلوگیری می شود. امنیت شخصی افراد در حوزه توسط برگزارکننده ای انتخابات تامین می شود و با وجود گزینه ی «رای سفید» فردی مجبور به رای دادن و یا رای دادن به یک کاندیدای خاص نمی شود.

با وجود یک صندوق برای چندین رای و نبودن هیچ نشانه ی شناسایی در آرا، هیچ راهی برای فهمیدن رای یک فرد خاص - حتی اگر برگه های رای به دست رقیب بیفتد - وجود ندارد.

احراز هویت و شمارش رای ها به عهده ی برگزارکننده ی انتخابات است و تنها از طریق یک شخص ثالث برای بازشماری آرا می توان از اجرای درست آن ها اطمینان حاصل کرد.

با توجه به هزینه ی زیاد شمارش در انتخابات های بزرگ راهی برای اعلام لحظه ای نتایج با هزینه ی معقول وجود ندارد.

همانطور که می بینیم در روش های فعلی انتخابات بسیاری از شرایط مورد نیاز یک انتخابات خوب با هزینه ی نسبتاً زیاد فراهم می شود. از دیگر مشکلات انتخابات به این روش می توان به نیازمندی به یک برگزارکننده ی مورد اعتماد اشاره کرد. باید به برگزارکننده اعتماد شود تا:

۱. امنیت حوزه ی انتخابات را تامین کند.

۲. افراد را به درستی احراز هویت کند.

۳. همه ی رای ها را بشمارد.

۴. تغییری در رای ها ندهد.

## ۳.۱ مشکلات و چالش های رای گیری الکترونیک

دو مسئله ی اساسی در یک سیستم رای گیری امنیت و حریم خصوصی است. مخالفین رای گیری الکترونیک از کم هزینه بودن تقلب و تغییر رای های ثبت شده در انتخابات الکترونیکی می گویند و رد کاغذی در یک انتخابات را یک فاکتور مهم برای امنیت آن می دانند. هزینه تغییر میلیون ها رای در یک سیستم کامپیوتری بسیار پایین تر از

تولید چند میلیون رای کاغذی تقلبی برای تغییر نتیجه‌ی یک انتخابات است.

بزرگترین مسئله در به‌کارگیری رای‌گیری الکترونیک مسئله‌ی اعتماد به یک سیستم کامپیوتری است. از نظر بسیاری از رای‌دهندگان رای دادن با کامپیوتر شخصی می‌تواند ریسک تغییر رای تا رسیدن آن به سرورهای رای‌گیری ایجاد کند. از طرف دیگر عدم امکان بررسی و تایید انسانی عملیات کامپیوتر، حس امنیت کمتری القا می‌کند.

مسئله‌ی دیگر پرهزینه بودن ساخت زیرساخت‌های رای‌گیری الکترونیک و خطر پیدایش مشکلات امنیتی در هر سیستم کامپیوتری - چه از نظر نرم‌افزار و چه سخت‌افزار - است. این مشکل باعث شده تعدادی از کشورها از جمله هلند، ایرلند و آلمان فرایند ایجاد زیرساخت لازم را شروع کرده و در ادامه این فرایند را ملقی کنند. دلیل اصلی اعلام شده برای این مسائل قابل اتکا نبودن سیستم‌های رای‌گیری الکترونیکی اعلام شده است.

برای مثال یک تحقیق معروف از دانشگاه NYU در سال ۲۰۱۵<sup>۱</sup> توضیح داد که ماشین‌های رای‌گیری الکترونیکی که در ۴۳ ایالت آمریکا استفاده می‌شوند در سال ۲۰۱۶ به دهمین سال استفاده شدن می‌رسند و به دلیل نداشتن بودجه‌ی کافی برای تعمیرات و بروزرسانی، در معرض خطر کرش<sup>۲</sup> کردن هستند که می‌تواند باعث کندی فرایند و حتی گاهی از دست رفتن رای‌های مردم شود. علاوه بر این، قدیمی بودن دستگاه‌ها می‌تواند ریسک‌های امنیتی ایجاد کند.

یک مشکل دیگر در پیاده‌سازی‌های بسیاری از رای‌گیری الکترونیک، نیاز به اینترنت و توانایی استفاده از کامپیوتر است. این مسئله می‌تواند دسترسی بسیاری از افرادی واجد شرایط را - به دلیل نقص جسمی و یا عدم توانایی کار با کامپیوتر - محدود کند. در سیستم‌های فعلی که مبتنی بر حوزه‌های رای‌گیری هستند می‌توانند با کمک انسانی در خود حوزه تا حدی این مشکلات را رفع کنند.

مشکلات مطرح شده موانع بزرگی برای فراگیری سیستم‌های رای‌گیری کاملاً الکترونیکی برای انتخابات‌های مهم و بزرگ هستند که یک سیستم رای‌گیری مناسب باید آن‌ها را تا جای ممکن رفع کند.

<sup>۱</sup> <https://www.brennancenter.org/publication/americas-voting-machines-risk>

<sup>۲</sup> crash



## ۴.۱ انگیزه و هدف

هدف این تحقیق، طراحی یک سیستم رای گیری الکترونیک است که شرایط رای گیری ایده آل را تا جای ممکن بدون نیاز به اعتماد به شخص ثالث ایفا کند. با فراگیری تکنولوژی بلاک چین برای ایجاد سیستم های توزیع شده بدون نیاز به اعتماد (برای مثال بیت کوین به عنوان یک ارز دیجیتال بدون نیاز به اعتماد)، پلتفرم هایی برای رای گیری الکترونیک ایجاد شدند که امنیت شمارش آرا را با عمومی ساختن فرایند رای گیری تامین می کردند. با وجودی که راه حل ارائه شده ی این سیستم ها مسئله ی اطمینان از شمارش رای ها را حل می کرد، مسئله ی حریم شخصی در این روش ها حل نشده است و انتخابات های برگزار شده با این سیستم ها امنیت کمتری در قبال ناشناس ماندن رای ها ارائه می کنند.

برای مثال حالتی را فرض کنید که یک رای دهنده تهدید می شود که باید به یک کاندیدای خاص رای بدهد، در سیستم های سنتی رای گیری به دلیل بی نام بودن برگه های رای بعد از اتمام فرایند رای گیری راهی برای اطمینان حاصل کردن از نتیجه ی رای فرد نیست. از طرفی به دلیل امنیت حوزه های رای گیری راهی برای اطمینان از نتیجه ی رای یک نفر در حین فرایند رای گیری هم نیست. پس راهی برای محبور کردن یک نفر که به یک کاندیدای خاص رای بدهد وجود ندارد. اما در سیستم های مبتنی بر بلاک چین هر رای داده شده به امضای الکترونیکی فرد امضا شده است و این موضوع می تواند با عمومی شدن بلاک چین بعد از رای گیری باعث لو رفتن نتیجه ی رای آن فرد شود.

این مشکلات مانع بزرگی برای استفاده ی فراگیر این سیستم ها خواهد بود. هدف ما در این تحقیق ارائه امنیت و هزینه ی کم ناشی از استفاده از این روش های رای گیری، بدون ایجاد ریسک های جدید در حریم خصوصی رای دهندگان است.

نتیجه ی این تحقیق یک سیستم رای گیری الکترونیک است که قیاس با سیستم های سنتی انتخابات هزینه ها را کاهش خواهد داد. در عین حال کمترین تغییر برای رای دهندگان خواهد داشت که باعث افزایش دسترس پذیری این سیستم خواهد شد. همچنین تمامی آرا رای دهندگان در قبال یک مهاجم خارجی و حتی خود برگزار کننده ی انتخابات ناشناس خواهند ماند.

از طرفی این سیستم یک رد الکترونیک غیرقابل انکار از تمام آرا، در قبال یک بلاک چین، ارائه خواهد کرد که

توانایی اثبات درستی شمارش را برای شخص ثالث بدون ایجاد خطری برای ناشناسی رای‌ها خواهد داد. تمامی این قابلیت‌ها بدون نیاز اعتماد به برگزارکننده‌ی انتخابات خواهد بود و هرگونه تخطی از پروتکل ارائه شده توسط حوزه‌های رای‌گیری قابل ردیابی از طریق اطلاعات ثبت شده در بلاک‌چین خواهد بود.

## فصل ۲

### تعريف مفاهيم

در این بخش به معرفی بعضی مفاهیم پایه برای این تحقیق می‌پردازیم. در ابتدا با مفاهیم بلاک چین و انواع و کاربردهای آن آشنا می‌شویم و در ادامه به بررسی اثبات‌های بی‌دانش می‌پردازیم. این دو تکنولوژی ابزارهای تئوری لازم برای ساخت سیستم رای‌گیری امن خواهند بود.

## ۱.۲ بلاک چین

بلاک چین ساختمان داده ایست که به مانند لینک لیست<sup>۱</sup> از بلوک‌های متوالی تشکیل شده ولی در بلاک چین هر بلوک هش<sup>۲</sup> عنصر قبلی خود را نیز نگه می‌دارد. هدف از این کار ساخت یک ساختار داده‌ی صرفاً افزایشی<sup>۳</sup> است که در آن بلوک‌های قبلی تغییرناپذیرند. تغییر هر بلوک باعث تغییر بلوک بعدی خواهد شد و این موضوع تشخیص تغییر در بلوک‌های پیشین را بسیار ساده می‌کند.

## ۲.۲ درخت مرکل

برای پیاده‌سازی یک بلاک چین معمولاً از درخت مرکل<sup>۴</sup> استفاده می‌شود. درخت مرکل یا درخت هش، نوعی درخت دودویی<sup>۵</sup> است که در آن هر راس هش فرزندان خود را نگه داشته و برگ‌ها هش داده‌ی ذخیره شده در خودشان را نگه می‌دارند. این روش نگهداری اطلاعات باعث می‌شود که درچه‌ی زمانی بررسی وجود یک بلوک داده در بلاک چین از  $N$  به  $\log N$  کاهش یابد. به دلیل این نوع ساختار یک درخت مرکل، هر تغییری در درخت باعث تغییر هش در ریشه‌ی آن خواهد شد و به دلیل رندم بودن خروجی یک هش خوب، هش ریشه‌ی درخت مرکل هیچ ویژگی قابل پیشبینی ندارد.

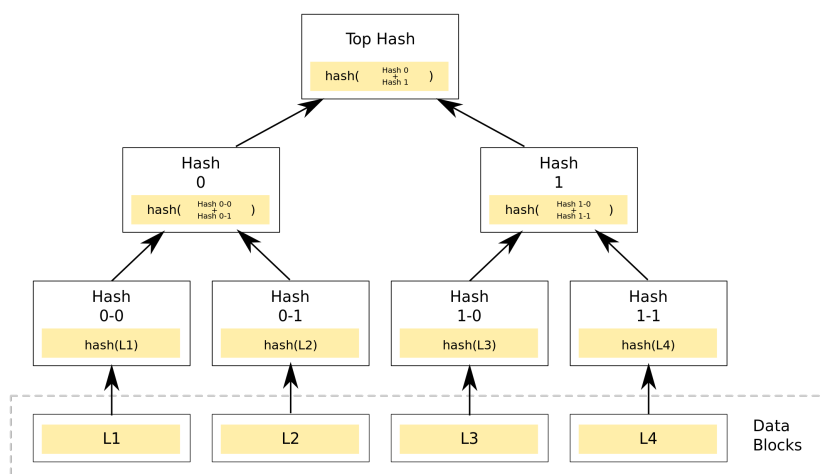
<sup>۱</sup> Linked list

<sup>۲</sup> Hash

<sup>۳</sup> Append only

<sup>۴</sup> Merkle tree

<sup>۵</sup> Binary tree



شکل ۱.۲: یک درخت مرکل

## ۱.۲.۲ انواع بلاک چین

در این تحقیق بلاک چین‌ها را از دو نظر دسته‌بندی می‌کنیم. بلاک چین‌ها می‌توانند عمومی یا خصوصی باشند، در بلاک چین‌های عمومی اضافه کردن بلوک به بلاک چین دسترسی خاصی نمی‌خواهد و هر کسی می‌تواند در آن‌ها بنویسد ولی در بلاک چین‌های خصوصی اضافه کردن بلوک صرفاً توسط افراد خاص ممکن است. روش دیگر تقسیم‌بندی ما باز یا بسته بودن بلاک چین است که این دسته‌بندی در مورد دسترسی خواندن اطلاعات از بلاک چین است. در بلاک چین‌های بسته خواندن اطلاعات توسط عموم آزاد نیست و در بلاک چین‌های خصوصی تمام اطلاعات بلاک چین برای خواندن، در دسترس عموم است. با توجه به کاربرد بلاک چین مورد نظر هر بلاک چین می‌تواند در هر کدام از این دسته‌بندی‌ها قرار بگیرد، جدول ۱.۲ یک کاربرد ممکن برای هر کدام از این دسته‌بندی‌ها را نشان می‌دهد.

جدول ۱.۲: انواع بلاک چین

بسته	باز	
بعضی رای‌گیری‌ها	ارزهای دیجیتال	عمومی
اطلاعات خصوصی یک شرکت	سامانه‌ی مدیریت اطلاعات مالیات	خصوصی

## ۳.۲ اثبات‌های بی‌دانش

اثبات بی‌دانش<sup>۱</sup> روشی است که یک «اثبات‌کننده» می‌تواند به یک «بررسی‌کننده» نشان دهد که او یک راز - مثلاً خروجی یک عملیات کامپیوتری - را می‌داند، بدون این که به بررسی‌کننده هیچ اطلاعات اضافه‌ای، مانند خروجی عملیات، بدهد. به عبارت دیگر اثبات‌های بی‌دانش، صرفاً داشتن اطلاعات را اثبات می‌کنند و خود اطلاعات را محفوظ نگه می‌دارند.

یک اثبات بی‌دانش باید ۳ شرط زیر را داشته باشد:

- کامل بودن: اگر گزاره‌ی مورد اثبات صحیح باشد، بررسی‌کننده‌ای که پروتکل را رعایت کند، باید از درستی گزاره مطمئن شود.
- درستی: اگر گزاره مورد اثبات غلط باشد، هیچ اثبات‌کننده‌ای نتواند اثباتی ارائه کند که گزاره درست است.
- بی‌دانش: اگر اثبات درست باشد، بررسی‌کننده هیچ اطلاعاتی فراتر از این که گزاره صحیح است دریافت نکند.

اثبات‌های بی‌دانش، اثبات‌های احتمالاتی هستند و در واقع احتمال کمی وجود دارد که بتوان یک اثبات نادرست ارائه کرد. به بیان دیگر شرط درستی این است که احتمال تولید یک اثبات نادرست بسیار کم باشد.

## ۱.۳.۲ مثال شهودی

سناریویی را در نظر می‌گیریم که یک توپ سبز و یک توپ قرمز روی یک میز قرار دارد و آلیس می‌خواهد به باب که کوررنگ سبز و قرمز است ثابت کند که این دو توپ با هم تفاوت دارند. برای اثبات آلیس چشمش را می‌بندد و باب یا دو توپ را جابجا می‌کند و یا جابجا نمی‌کند. در ادامه آلیس می‌گوید که آیا جای توپ‌ها با هم عوض شده‌اند یا نه. با یک پاسخ درست باب می‌فهمد که آلیس با احتمال ۵۰٪ درست می‌گوید. این فرایند را تا جایی که باب به احتمال دلخواهش برسد ادامه می‌دهند.

<sup>۱</sup> Zero knowledge proofs

یک نکته‌ی مهم در مثال بالا این است که حتی اگر باب این فرایند را ضبط کرده باشد، نمی‌تواند به کس دیگری اثبات کند که آلیس تفاوت این دو توپ را می‌داند چون که راهی برای اثبات این که سوال و جواب از قبل هماهنگ نشده بوده است ندارد.

این یکی از نیازمندی‌های بی‌دانش بودن اثبات است. اگر در فرایند برای تصمیم‌گیری در تعویض توپ‌ها باب از شیر یا خط کردن یک سکه استفاده می‌کرد، دیگر این اثبات بی‌دانش نبود، چرا که باب می‌توانست با ضبط کردن این فرایند به یک شخص ثالث اثبات کند که آلیس تفاوت این دو توپ را می‌داند.

برای داشتن شرط بالا یک اثبات بی‌دانش همواره تعامل از سمت بررسی‌کننده نیاز دارد. اما با ریلکس کردن این شرط و استفاده از یک ورودی غیرقابل پیش‌بینی برای تولید سوال‌های یک اثبات بی‌دانش - مثلاً هش ریشه‌ی یک درخت مرکب - می‌توان اثبات‌های بی‌دانش بدون نیاز به تعامل بررسی‌کننده ساخت.

## ۲.۳.۲ اثبات‌های بی‌دانش بدون تعامل

منظور از اثبات بدون تعامل، اثباتی است که در آن نیازی به فرستادن پیامی از سمت بررسی‌کننده به اثبات‌کننده نباشد. با این روش‌ها اثبات‌کننده می‌تواند اثبات را مستقل از بررسی‌کننده بسازد و ارسال کند، در ادامه‌ی این تحقیق اثبات‌های بی‌دانش و بی‌تعامل را **شاهد** می‌نامیم. در ادامه دو روش تولید یک شاهد بی‌دانش را بررسی می‌کنیم. این روش‌ها می‌توانند برای خروجی هر محاسبات کامپیوتری شاهد ایجاد کنند.

### ZK-SNARK

این روش مخفف Zero-Knowledge Succinct Non-Interactive Argument of Knowledge است. شاهد‌های این روش علاوه بر بی‌دانش بودن ویژگی‌های زیر را دارند:

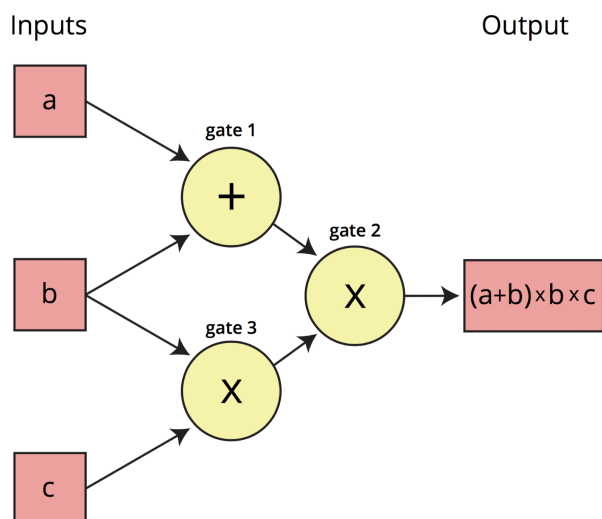
- مختصر<sup>۱</sup>: تولید و بررسی شاهد از انجام خود محاسباتی که اثبات می‌شود کوتاه‌تر (معمولاً از مرتبه‌ی زمانی  $(\log N)^2$ ) است.

- بی‌تعامل<sup>۲</sup>: نیازی به پیامی از بررسی‌کننده برای ایجاد شاهد نیست.

<sup>۱</sup> Succinct

<sup>۲</sup> Non-Interactive

- ادعای دانش<sup>۱</sup>: اثبات ارائه شده در این روش درست<sup>۲</sup> است و نمی‌شود بدون داشتن اطلاعات آن را در زمان محدود ساخت.



شکل ۲.۲: یک نمونه مدار محاسباتی

برای ساختن یک شاهد به این روش ابتدا محاسبات لازم را به یک مدار محاسباتی ریاضی تبدیل می‌کنیم به طوری که اثبات را به عنوان تعدادی شرط روی این مدار نشان دهیم، سپس به کمک یک elliptic curve مقدار مدار را در چند نقطه‌ی تصادفی به عنوان اثبات ارائه می‌کنیم، با صادق بودن شرط‌ها در این نقاط شاهد را بررسی می‌کنیم.

برای انتخاب یکسان این نقاط تصادفی بین اثبات‌کننده و بررسی‌کننده نیاز به تعدادی نقطه‌ی توافق شده روی elliptic curve داریم که باید در قبل از تولید اثبات انتخاب شده باشند. در این فاز آماده‌سازی تعدادی عدد تصادفی برای انتخاب این نقاط تولید می‌شوند که بعد از تولید نقاط باید بلافاصله پاک شوند. کسی که این اعداد (در واقع نقطه‌ی شروع روی منحنی) را داشته باشد می‌تواند شاهد‌های تقلبی ایجاد کند. برای تولید شاهد واقعی نیازی به دانستن این نقاط نیست و بنابراین بعد از فاز آماده‌سازی این اعداد باید پاک شوند.

<sup>1</sup> Argument of Knowledge

<sup>2</sup> Sound



**ZK-STARK**

این روش مخفف Zero-Knowledge Scalable Transparent ARguments of Knowledge است. مهم‌ترین وجه تمایز این روش در مقایسه با ZK-SNARK «شفافیت»<sup>۱</sup> است، به این معنی که نیازی به فاز آماده‌سازی ندارد. عدم نیاز به آماده‌سازی و نداشتن زباله‌ی سمی (اطلاعاتی که باید پاک شوند تا امنیت سیستم تامین شود) این روش را برای کاربردهای حساس مناسب‌تر می‌کند اما در ازای این امنیت، حجم شاهدها از چند صد بایت به چند صد هزار بایت تغییر می‌کند.

از مزیت‌های دیگر این روش استفاده نکردن از Elliptic curveها است. نیازهای کم این روش باعث می‌شود که حتی با کامپیوترهای کوانتومی<sup>۲</sup> راهی برای شکستن این اثبات‌ها وجود نداشته باشد. برای ساختن یک شاهد با این روش، برنامه‌ی مورد نظر را تبدیل به یک چندجمله‌ای درجه بالا می‌کنند، سپس از مقادیر این چندجمله‌ای یک درخت مرکب ساخته می‌شود که مقادیر مختلف خروجی را نشان می‌دهد. سپس بررسی‌کننده چند شاخه از این درخت را به طور تصادفی انتخاب و بررسی می‌کند. برای غیرتعاملی کردن این اثبات می‌توان از هاش ریشه‌ی درخت مرکب به عنوان ورودی به تابع شبه‌تصادفی<sup>۳</sup> استفاده می‌شود که مشخص می‌کند خروجی کدام شاخه‌ها باید در شاهد بیاید.

---

<sup>1</sup> Transparency

<sup>2</sup> Quantum computers

<sup>3</sup> Pseudo random

## فصل ۳

### کارهای پیشین

در این بخش ابتدا به بررسی تحقیقاتی می‌پردازیم که به مسئله حذف اعتماد از سیستم‌های مبتنی بر اعتماد پرداخته‌اند، در ادامه به کارهای مربوط به رای‌گیری الکترونیک و در نهایت به اثبات‌های بی‌دانش می‌پردازیم.

### ۱.۳ اعتماد

مسئله حذف نیاز به یک شخص معتمد را از طریق عمومی ساختن کل اطلاعات مورد نیاز می‌توان حل کرد. اگر تمامی اطلاعات درست باشد هر کسی می‌تواند برای خود درستی تراکنش‌ها را بررسی کند. مسئله‌ای که باقی می‌ماند زمانبست که بین چند شخص اختلاف پیش می‌آید که نسخه‌ی درست اطلاعات کدام است. مثلاً زمانی که چند نسخه‌ی صحیح از نظر فرمت وجود دارند اما نتایج مختلفی را می‌رسانند.

#### ۱.۱.۳ توافق

توصیف رسمی این مسئله، مسئله‌ی ژنرال‌های بیزنتین<sup>۱</sup> [۱] است. در این مسئله چند ژنرال که می‌توانند یک به یک با هم صحبت کنند، در تلاشند تا به توافق برسند که آیا باید حمله کنند یا نکنند، تعدادی از ژنرال‌ها خائن هستند و در تلاشند که نتیجه‌ی توافق ژنرال‌ها را تغییر دهند. ژنرال‌های خائن می‌توانند با جواب ندادن یا جواب غلط دادن تلاش کنند که نتیجه‌ی توافق را تغییر دهند. در ساده‌ترین حالت و بدون استفاده از امضاهای دیجیتال ثابت می‌شود که برای  $3k + 1$  ژنرال، با رای‌گیری می‌توان تا  $k$  خائن را تحمل کرد. راه‌حل‌های متعددی برای توافق<sup>۲</sup> در بستر بلاک‌چین داده شده که در ادامه به تعدادی از آن‌ها می‌پردازیم.

#### توافق

روشی که S.Nakamoto [۲] برای رفع این مسئله در بیت‌کوین استفاده کرده است، اثبات کار<sup>۳</sup> نام دارد. این روش که بر پایه‌ی روش استفاده شده در hashcash [۳] است. در این روش برای اضافه شدن هر بلوک به بلاک‌چین باید یک مسئله‌ی سخت (که نیاز به توان پردازشی بالا دارد) حل شود ولی بررسی درستی جواب ساده است. این روش روش بسیار فراگیری در ارزهای دیجیتال است. از مشکلات این روش می‌توان به توان مصرفی بالا و کندی

<sup>۱</sup> The Byzantine generals problem

<sup>۲</sup> کار اثبات

<sup>۳</sup> Proof of work

نسبی آن اشاره کرد. برای مثال حداکثر توان تئوری بیت کوین، ۷ تراکنش بر ثانیه است.

### اثبات سهم

در روش اثبات سهم<sup>۱</sup> [۴] برای ساخت بلوک‌های جدید باید یک فاکتور مقدار سکه‌های در اختیار ماینتر و سن آن‌هاست. به این صورت که می‌تواند در ازای سن سکه‌های در اختیارش (با زدن یه تراکنش به خود) هش ساده‌تری برای بلوک بعدی اعمال کند. مزیت اصلی این روش توان مصرفی پایین‌تر آن به نسبت اثبات کار است. معمولاً در بلاک‌چین‌ها در بلوک‌های ابتدایی از روش اثبات کار استفاده می‌شود و بعد از مدتی برای کاهش هزینه‌های اضافه کردن بلوک جدید و مقایسه‌پذیری می‌توان از این روش یا ترکیب این روش‌ها استفاده کرد.

### Protocol Consensus Ripple

در این روش [۵] [۶] تعدادی شخص مورد اعتماد وجود دارند که برای اضافه‌شدن بلوک به بلاک‌چین باید درصدی از آن‌ها درستی تراکنش را تایید کنند. این اشخاص در دسته‌های مختلف قرار می‌گیرند و برای تایید باید یک زیردسته‌ی کامل تراکنش‌ها را تایید کنند. با وجود سرعت نسبتاً بالای این روش - تا ۱۰۰۰ تراکنش در ثانیه - منتقدین آن از نیاز به اشخاص مورد اعتماد می‌گویند. این روش تا  $n/5$  خطا در نودهای مورد اعتماد را می‌تواند تحمل کند.

### Protocol Consensus Stellar

روش SCP [۷] مبتنی بر ایجاد افراد مورد اعتماد، به صورت طبیعی و خودجوش، در شبکه است. در این روش با افزایش تراکنش‌های درست توسط هر شخصی، آن شخص به عنوان فرد مورد اعتماد شناخته می‌شود و هر تراکنش را باید تعداد افراد مورد اعتماد تایید کنند. این افراد توسط پرداخت‌کننده‌ی تراکنش انتخاب می‌شوند اما دسته‌بندی آن‌ها در شبکه به گونه‌ای است که خطا در تایید تراکنش باعث حذف شدن فرد از لیست افراد مورد اعتماد شود. تفاوت اصلی این روش با Ripple در توانایی انتخاب تاییدکنندگان تراکنش است و فرض‌های اعتماد کمتر این روش باعث می‌شود که تا  $n/3$  خطا در نودهای مورد اطمینان را بتواند تحمل کند.

<sup>۱</sup> proof of stake

### ۲.۱.۳ کاربردهای بلاک چین

#### ارز دیجیتال

اولین کاربرد بلاک چین، بیت کوین بود که با موفقیت آن چندین محصول دیگر هم تولید شدند. در بستر بیت کوین اثبات کار به صورت زیر استفاده می شود:

هر بلوک جدید شامل تعدادی تراکنش، برای ثبت در بلاک چین است. اما برای پذیرفته شدن این بلوک توسط دیگر دیگران، باید در این بلاک یک رشته بی معنی<sup>۱</sup> قرار گیرد به صورتی که هش بلاک از عددی که توسط پروتکل بیت کوین انتخاب می شود کمتر باشد. این شرط در طول زمان به صورت خودکار به روزرسانی می شود به طوری که در هر لحظه به صورت میانگین اضافه کردن یک بلوک ۱۰ دقیقه از کل شبکه زمان ببرد. از آنجایی که تنها راه یافتن همچنین رشته ای بروت فورس<sup>۲</sup> است، توان محاسباتی بالاتر احتمال یافتن بلوک بعدی را افزایش خواهد داد.

در پروتکل بیت کوین طولانی ترین بلاک چین - یعنی بلاک چینی که بیشترین توان محاسباتی برای آن صرف شده - به عنوان نسخه ی درست در نظر گرفته می شود. در نتیجه فرض می کنیم شخص A یک بیت کوین را به B منتقل کرده، این تراکنش در بلاک چین ثبت شده و در ازای آن کالایی دریافت کرده است، حال قصد دارد این تراکنش را از بلاک چین بیت کوین حذف کند تا بتواند آن را دوباره خرج کند. از آنجایی که نودهای شبکه ی بیت کوین اگر ۲ زنجیره از بلوک ها دریافت کنند زنجیره ی بلندتر را قبول خواهند کرد باید قبل از این که کل شبکه یک بلوک اضافه کند، دو بلوک سالم بسازد.

احتمال موفقیت حمله ی A مساوی<sup>۲</sup>  $\left(\frac{A's\ computational\ power}{Bitcoin\ network's\ computational\ power}\right)^2$  است. اگر توان محاسباتی A از دیگر قسمت های شبکه کمتر باشد، این کسر عددی کوچک تر از 0.5 است. اگر این کار به موقع با موفقیت انجام نشود، سه بلوک عقب می افتد و توان فرمول بالا تبدیل به سه می شود و احتمال موفقیتش کمتر از پیش می شود. این مسئله مسئله ی قمارباز<sup>۳</sup> نام دارد که نشان داده می شود در آن در طول زمان احتمال موفقیت مهاجم به صورت نمایی کاهش پیدا می کند.

<sup>1</sup> Nounce

<sup>2</sup> Brute force

<sup>3</sup> Gambler's Ruin

از پرکاربردترین‌های این محصولات می‌توان به LiteCoin و z-cash [۸] اشاره کرد. مزیت لایت کوین نسبت به بیت کوین در هزینه‌ی کمتر اضافه کردن بلوک و زمان تراکنش‌های کمتر است و زی‌کش امنیت بیشتری در زمینه‌ی حریم خصوصی و ناشناس ماندن ارائه می‌کند.

از نمونه‌های دیگر می‌توان به Ripple اشاره کرد، که برای تراکنش‌های بانک‌ها طراحی شده و همانطوری که بررسی کردیم از اثبات کار برای توافق استفاده نمی‌کند.

### ۳.۱.۳ سازمان‌های توزیع‌شده‌ی خودکار

دسته‌ی بعدی کاربردهای بلاک‌چین ایجاد سازمان‌های توزیع‌شده‌ی خودکار<sup>۱</sup> است. این مفهوم بر اساس مفهوم قرارداد هوشمند<sup>۲</sup> [۹] ساخته شده است. این سازمان‌ها برنامه‌هایی هستند که به طور خودکار و بی‌اعتماد در بستر بلاک‌چین ایجاد می‌شوند و می‌توانند کاربردهای بسیاری داشته باشند. اتریوم<sup>۳</sup> [۱۰] که یک ارز دیجیتال است، یک بستر مناسب برای تولید قراردادهای هوشمند هم هست که در سال‌های اخیر کاربردهای بیشتری پیدا کرده‌اند. با این وجود یک دغدغه‌ای که همچنان در این بستر وجود دارد خطر اشتباه‌های برنامه‌نویسی در این سازمان‌هاست، از آن‌جا که تغییر کد اینگونه سازمان‌ها به دلیل خودکار بودن، همواره قابل تغییر نیست، اشتباهات امنیتی می‌تواند تاثیر مخرب عظیمی داشته باشند. [۱۱] atezi به بررسی مشکلات امنیتی معمول قراردادهای در بستر اتریوم و تله‌ی معمول این زبان برنامه‌نویسی و روش‌های تصحیح آن‌ها پرداخته است.

یک مسئله‌ی دیگر که ناشی از ناشناسی ذاتی این بسترهاست ایجاد شدن سازمان‌های مچرمانه در آن‌هاست. تحقیقات بسیاری [۱۲] [۱۳] در این زمینه شده، و از مثال‌های قراردادهای مخرب ممکن می‌توان به افشای اطلاعات خصوصی و یا حتی دزدین کلیدهای رمزنگاری اشاره کرد. به دلیل عدم وجود نظارت مرکزی در این سیستم‌ها، راهی برای جلوگیری از اینگونه قراردادها وجود ندارد.

<sup>۱</sup> Decentralized autonomous organization (DAO)

<sup>۲</sup> Smart contract

<sup>۳</sup> Ethereum

### ۴.۱.۳ شناسایی

از کاربردهای دیگر بلاک چین ساخت روش های شناسایی<sup>۱</sup> است. ویژگی تغییر ناپذیری بلوک های قدیمی در یک بلاک چین، باعث می شود که ساختار داده ای ایده آلی برای شناسایی باشد. از این نمونه کاربردها می توان به namecoin اشاره کرد. هدف این محصول کاهش نیاز به اعتماد در DNS<sup>۲</sup> است. این محصول که روی بلاک چین بیت کوین ساخته شده است، در متن نوشته شده در تراکنش اطلاعات مربوط به نام دامنه ها را ذخیره می کند. از تحقیقات دیگر در این زمینه می توان به امنیت در داکر<sup>۳</sup> [۱۴] اشاره کرد. در این تحقیق با استفاده از بلاک چین یک بستر توزیع شده برای به اشتراک گذاری فایل های امضا شده ساخته شده است. در این سیستم می توان از یک بلاک چین خصوصی و یا به بلاک چین عمومی استفاده کرد.

### ۲.۳ رای گیری الکترونیک

رای گیری الکترونیک را به طور کلی می توانیم به دو دسته رای گیری تماما الکترونیک و رای گیری به کمک ابزارهای الکترونیکی تقسیم کرد. روش های دسته دوم مبتنی بر رای گیری سنتی هستند و از ابزاری های الکترونیکی صرفا برای کاهش هزینه و افزایش دسترسی پذیری استفاده می کنند. از این روش ها می توان به ابزارهای شمارش رای خودکار و یا دستگاه های ثبت رای الکترونیکی که خروجی آن ها یک برگه رای کاغذی<sup>۴</sup> است اشاره کرد. در این تحقیق این ابزارها را بررسی نمی کنیم و منظور از رای گیری الکترونیک، دسته اول یا رای گیری تماما الکترونیک است.

سیستم های رای گیری الکترونیک را می توان به دو دسته کلی توزیع شده و متمرکز تقسیم کرد. سیستم های مرکزی نیازمند یک ارتباط امن از رای دهنده تا سرویس مرکزی هستند. همچنین نیازمند اعتماد کامل به همان یک سرویس برای درستی انتخابات است. در سیستم های توزیع شده تلاش می کنند تا این دو مسئله را کمرنگ تر کنند.

<sup>۱</sup> Authentication

<sup>۲</sup> Domain Name Service

<sup>۳</sup> Docker

<sup>۴</sup> Direct-recording electronic voting systems or DRE

### ۱.۲.۳ رای‌گیری الکترونیک متمرکز

در این سیستم‌ها یک سامانه‌ی مرکزی وجود دارد که تمامی آرا در آن ذخیره می‌شوند. در این سیستم‌ها حوزه‌های رای‌گیری می‌توانند وجود داشته باشند اما حوزه‌ها صرفاً وظیفه‌ی احراز هویت و ارائه‌ی درگاه امن برای ثبت رای در سامانه‌ی مرکزی را دارند. حریم خصوصی کاربران در این سیستم‌ها مبتنی بر استفاده از کانال‌های ارتباطی ناشناس<sup>۱</sup> بین حوزه‌های رای‌گیری و سامانه‌ی مرکزی است.

اولین تحقیق در رابطه با استفاده از کانال‌های ارتباطی ناشناس برای رای‌گیری در سال ۱۹۸۵ توسط Chaum [۱۵] بود که در آن برای ساخت کانال‌های ارتباطی ناشناس از امضای کورکورانه<sup>۲</sup> [۱۶] برای ایجاد یک ارز دیجیتال استفاده می‌شد. در امضای کورکورانه، برای حفظ حریم خصوصی فردی که باید اطلاعاتی را تایید کند، رمزشده‌ی اطلاعات را امضا می‌کند، به این صورت از اطلاعات پیام باخبر نمی‌شود. اولین تلاش برای ایجاد یک پروتکل رای‌گیری الکترونیک با امضای کورکورانه در سال ۱۹۹۲ [۱۷] بود و در ادامه در سال ۱۹۹۷ [۱۸] نسخه‌ی کامل‌تری از آن ارائه شد. این روش‌ها مبتنی بر وجود یک شمارنده و یک حوزه هستند، حوزه احراز هویت را انجام می‌دهد و برگه رای‌های ناشناس صادر می‌کند و سپس از طریق یک کانال ارتباطی ناشناس رای‌دهنده رای را به شمارنده می‌دهد. از مشکلات این روش می‌توان به نیاز به اعتماد به حوزه اشاره کرد. حوزه می‌تواند که با ارائه رای‌های اشتباه بدون توانایی پیگیری، رای‌گیری را خراب کند، برای حل این مشکل تحقیقاتی [۱۹] در راستای استفاده از چند حوزه انجام شده است. مشکل بزرگ دیگر این روش‌ها [۲۰] سختی ناشناس نگه‌داشتن کانال‌های ارتباطی ناشناس است.

### ۲.۲.۳ رای‌گیری الکترونیک توزیع‌شده

رای‌گیری الکترونیک توزیع‌شده را به دو دسته‌ی رای‌گیری‌های بدون بلاک‌چین و با بلاک‌چین عمومی و با بلاک‌چین خصوصی تقسیم می‌کنیم. با توجه به این که در رای‌گیری احراز هویت یک مسئله‌ی مهم است هم‌هی این سیستم‌ها از بلاک‌چین‌های بسته استفاده می‌کنند.

<sup>۱</sup> Anonymous communication channel

<sup>۲</sup> Blind signature



### رای‌گیری بدون بلاک‌چین

بعضی سیستم‌های طراحی شده برای رای‌گیری الکترونیک [۲۱] [۲۲] [۲۳] از روش‌های تقسیم راز<sup>۱</sup> استفاده می‌کنند. در این روش‌ها فرایند ثبت رای باید به تایید تعدادی از حوزه‌ها برسد که باعث کاهش نیاز به اعتماد می‌شود. در این روش‌ها می‌توان اثبات کرد که برای ردیابی یک رمز حداقل  $k$  حوزه باید تبانی کنند. روش دیگری که برای رای‌گیری توزیع‌شده استفاده شده است، استفاده از بردار بررسی<sup>۲</sup> [۲۴] است. در این روش‌ها بررسی درستی رای‌ها کاملاً توزیع‌شده است اما نیاز به ارتباط دو به دوی تمامی رای‌دهنده‌ها دارد که در یک انتخابات واقعی شدنی نیست. ترکیبی از این روش و تقسیم راز باعث ایجاد پروتکل‌هایی [۲۵] [۲۶] شد که با سطح‌بندی حوزه‌ها و تقسیم رای‌ها و مخلوط کردن آن‌ها از حریم خصوصی حمایت می‌کنند. اما این روش‌ها به حوزه‌ها و رای‌دهندگان توانایی بررسی درستی برگه رای‌ها را نمی‌دهد و امکان ایجاد رای‌های اشتباه و جلوگیری از رای دادن یک فرد خاص را ایجاد می‌کنند.

### رای‌گیری با بلاک‌چین عمومی

با فراگیر شدن تکنولوژی بلاک‌چین [۲۷]، محصولاتی در زمینه رای‌گیری الکترونیک به کمک این تکنولوژی ساخته شدند. تعدادی از این سیستم‌های رای‌گیری در قالب قراردادهای هوشمند ساخته شده‌اند که از آن‌ها می‌توان به‌ویژه<sup>۳</sup> [۲۸] و یا کار E.Yavuz [۲۹] در بستر اتریوم اشاره کرد. مزیت اینجور رای‌گیری‌ها هزینه‌ی اولیه کم استفاده از آن‌هاست، اما همچنان ریسک اشتباه برنامه‌نویسی در این سبک کارها بسیار بالاست. همچنین هزینه اجرای قراردادهای هوشمند به تعداد بالا برای یک رای‌گیری هزینه‌ی بالایی خواهد داشت که در طول زمان باعث افزایش هزینه‌ی رای‌گیری خواهد شد. مسئله‌ی دیگر در بستر اتریوم هم وابستگی سیستم رای‌گیری، به پهنای باند نودهای اتریوم و میزان بار روی شبکه‌ی آن است. این موضوع می‌تواند باعث کند شدن یا حتی در مواردی حذف شدن تعدادی از آرا شود.

<sup>1</sup> Secret sharing

<sup>2</sup> Check vector

<sup>3</sup> Votereum

### رای‌گیری با بلاک‌چین خصوصی

از سیستم‌های رای‌گیری با بلاک‌چین خصوصی می‌توان به VoteBook [۳۰] توسط شرکت Kaspersky که یک شرکت پیشرو در زمینه‌ی امنیت است اشاره کرد. فلسفه‌ی ساخت این سیستم به صورتی است که تلاش می‌کند برای کاربرانی که از سیستم‌هایی رای‌گیری فعلی استفاده می‌کنند کمترین تغییر در رفتار نیاز باشد. از مثال‌های دیگر سیستم‌های رای‌گیری مبتنی بر بلاک‌چین می‌توان به استارت‌آپ Follow My Vote اشاره کرد. نحوه‌ی کار این سیستم با سیستم VoteBook تفاوت اساسی دارد و برای رای‌دادن احتیاج دارد که نرم‌افزاری برای رای‌دادن به روی کامپیوتر و یا تلفن همراه کاربران نصب شود. اینگونه طراحی سیستم، خطرات امنیتی در قالب بدافزار ایجاد می‌کند. همچنین با نبود یک حوزه‌ی رای‌گیری امن راهی برای تامین امنیت رای‌دهندگان و اطمینان حاصل کردن از این که کسی مجبور به رای دادن نشده، نیست.

و در نهایت یکی از موفق‌ترین سیستم‌های رای‌گیری مبتنی بر بلاک‌چین موجود در حال حاضر VoteWatcher ساخته شده توسط یک شاخه از شرکت blockchain Technologies Corporation است که یک شرکت بزرگ برای ارائه‌ی سرویس‌های مبتنی بر بلاک‌چین است. طبق وب‌سایت این محصول تاکنون بیش از صد هزار رای در بیشتر از ۲۰ رای‌گیری مختلف توسط این سیستم شمارش شده است.

مدل اسفاده‌ی VoteWatcher به سیستم VoteBook بسیار شبیه است و تفاوت رفتاری زیادی با مدل‌های رای‌گیری الکترونیکی فعلی برای کاربران ندارد. در این محصول طبق نیاز رای‌گیری می‌توان از یک بلاک‌چین عمومی یا خصوصی استفاده کرد.

از موارد دیگر می‌توان به پیاده‌سازی‌های به کمک قراردادهای هوشمند ولی با استفاده از بلاک‌چین خصوصی [۳۱] که توانایی ردگیری بالاتری از پیاده‌سازی‌های دیگر ارائه می‌کنند، اما به دلیل کندی نسبی، نیاز به تقسیم رای‌گیری به چند رای‌گیری کوچک‌تر دارند.

### ۳.۳ اثبات‌های بی‌دانش

اثبات‌های بی‌دانش اولین بار در سال ۱۹۸۵ [۳۲] به عنوان روشی ساخت یک روش رمزنگاری متقارن با کلید عمومی استفاده شد. با پیشرفت تکنولوژی اثبات‌های بی‌دانش، روش‌های جامع اثبات بی‌دانش مانند ZK-

SNARK [۳۳] و ZK-STARK [۳۴] بوجود آمدند. این روش‌ها توانایی اثبات هر محاسباتی را به صورت بی‌دانش دارند و این موضوع باعث استفاده‌ی آن‌ها در کاربردهای بیشتری شد.

### ۱.۳.۳ کاربردها

با فراگیری ارزهای دیجیتال، مسئله‌ی حریم خصوصی در آن‌ها پررنگ‌تر شد. در ابتدا یکی از بزرگ‌ترین کاربردهای بیت‌کوین پرداخت‌های مجرمانه بود که ناشناسی نسبی در این بستر باعث می‌شد برای این سبک پرداخت‌ها ایده‌آل باشند. اما به دلیل عمومی بودن بلاک‌چین و تمامی تراکنش‌ها در آن دنبال کردن رد پرداخت بسیار ساده است.

برای جلوگیری از این ردگیری در کاربردهای مجرمانه از یک شخص مورد اعتماد برای «مخلوط کردن» سکه‌های افراد استفاده می‌شود. در این روش چندین نفر به یک نفر پرداخت می‌کنند و آن فرد به کلیدهای عمومی که از قبل تعیین شده با سکه‌های جدید پرداخت می‌کند اما در این روش همگی به شخص مخلوط‌کننده اعتماد می‌کنند که به اندازه پرداخت کند و رد واقعی سکه‌ها را جایی ثبت نکند. این روش برای کاربردهای مجرمانه تا حد بسیار خوبی پاسخگو است اما برای افراد عادی که صرفاً دغدغه‌ی حریم خصوصی خود را دارند خطرناک و هزینه‌ی این روش معقول نیست، به همین دلیل کارهای مختلفی در زمینه‌ی پرداخت ناشناس انجام شده است.

### پرداخت ناشناس

از این تحقیقات می‌توان به بستر HAWK [۳۵] اشاره کرد. در این تحقیق به کمک یک تعریف کلی از بلاک‌چین به عنوان سیستمی که همواره در دسترس است و هیچ اطلاعات اشتباه نمی‌پذیرد اما حریم خصوصی را حفظ نمی‌کند یک بستر قرارداد هوشمند ساخته شده است که در آن به ازای کد قرارداد هوشمند، یک کد برای حفظ حریم خصوصی به کمک اثبات‌های بی‌دانش ساخته می‌شود. روش کار این سیستم مبتنی بر ایجاد آدرس‌های مقصد یکتا به ازای هر تراکنش است.

مونرو<sup>۱</sup> که یک ارز دیجیتال است که بر اساس الگوریتم Cryptonote [۳۶] کار می‌کند، به مانند HAWK با یکتا سازی آدرس‌های مقصد و امضای حلقه‌ای<sup>۲</sup> کار می‌کند. در ادامه این ارز دیجیتال با همین روش مدلی [۳۷] برای

<sup>۱</sup> Monero

<sup>۲</sup> Ring Signature

مخفی کردن پرداخت کننده‌ی سکه نیز ارائه کرد.

از کارهای دیگر در این زمینه می‌توان به zerocoin [۳۸] که روش پرداخت ناشناس بر بستر بیت‌کوین به کمک ZK-SNARK ارائه کرد اشاره کرد. روش استفاده شده در آن با بهبود در ارز دیجیتال z-cash [۸] استفاده شد. این ارز دیجیتال دو مدل سکه‌ی قابل ردگیری و غیرقابل ردگیری دارد و هر کسی می‌تواند طی یک تراکنش سکه‌های خود را به سکه‌های ناشناس تبدیل کند.

### ۲.۳.۳ فاز آماده‌سازی

با توجه به این که ZK-SNARK ها، به طور خاص خاص پروتکل پینوکیو<sup>۱</sup> فراگیرترین روش برای ایجاد اثبات‌های بی‌دانش است. تحقیقات زیاد در مورد فاز آماده‌سازی و ایجاد پارامترهای عمومی این مدل اثبات شده است. همانطور که قبلاً اشاره کردیم ورودی‌های این فاز اگر بعد از این فاز پاک نشوند می‌توانند برای ایجاد اثبات‌های تقلبی استفاده شوند.

از این تحقیق‌ها می‌توان به روش‌هایی [۳۹][۴۰] که تلاش در کاهش نیاز به اعتماد در این فاز می‌کنند اشاره کرد. این روش‌ها باعث می‌شوند که برای لو رفت اطلاعات خطرناک احتیاج به تبانی تمامی اعضای موجود در فاز آماده‌سازی باشد.

از دیگر کارهای در این زمینه می‌توان به تلاش‌هایی برای حذف فاز آماده‌سازی به طور کلی اشاره کرد. این روش با تغییر اساسی در پروتکل و استفاده از چندجمله‌ای‌ها [۴۱] مرحله‌ی آماده‌سازی را حذف می‌کند

<sup>۱</sup> Pinocchio

## فصل ۴

### روش پیشنهادی

در این بخش به بررسی روش پیشنهادی این تحقیق می‌پردازیم، در ابتدا شرایطی که در آن مسئله را حل کنیم بررسی می‌کنیم و شرط‌های لازم برای سیستم رای‌گیری خود را بررسی می‌کنیم، در ادامه به بررسی کلی روش رای‌گیری عناصر حاضر در آن می‌پردازیم و در نهایت الگوریتم‌ها و پروتکل دقیق را بررسی می‌کنیم.

## ۱.۴ تعریف عناصر

در این بخش عناصر حاضر در سیستم رای‌گیری و انتظارات خود از آن‌ها را تعریف می‌کنیم:

- **ناظر انتخابات:** این سازمان مسئول بررسی درستی انتخابات و احراز هویت شرکت‌کنندگان در انتخابات است. به این سازمان اعتماد می‌شود تا کار احراز هویت را به درستی انجام دهد. همچنین این سازمان بررسی‌کننده‌ی نهایی درست بودن انتخابات است و باید بتواند از درستی انتخابات اطمینان حاصل کند.
- **رای‌دهنده:** فردی که حق رای به یک کاندیدا را دارد، این فرد می‌تواند از رایش استفاده نکند یا نکند، می‌تواند تلاش کند که چند بار رای دهد. باید بتواند از درستی انتخابات اطمینان حاصل کند. ممکن است توسط یک رقیب بدخواه برای رای دادن تحت فشار قرار بگیرد.
- **حوزه‌ی انتخابات:** محلی که در آن رای داده می‌شود، باید بتواند امنیت فیزیکی افراد را تامین کند. ممکن است برای خراب کردن انتخابات یا نقض حریم خصوصی کاربران تلاش کند.

## ۲.۴ شرایط مسئله

شروط لازم برای سیستم رای‌گیری ارائه شده عبارتند از:

۱. هر فرد واجد شرایط دقیقاً یک بار بتواند رای دهد.
۲. هیچ‌کسی نتواند به جای فرد دیگری رای دهد.
۳. هیچ فردی مجبور به رای دادن نشود.
۴. هیچ فردی مجبور به رای دادن به کاندیدای خاصی نشود.

۵. در صورت نقض حریم خصوصی و یا شماره‌دهی نشدن بعضی رای‌ها ناظر انتخابات بتواند حوزه‌ی متخلف را شناسایی کند. حوزه‌ها به دلیل در اختیار داشتن سامانه‌های کامپیوتری رای‌گیری همواره می‌توانند با روش‌های phishing و یا استفاده از بدافزارها حریم خصوصی کاربر را زیر سوال ببرند یا رای او را ثبت نکنند. به همین دلیل قابل پیگیری بودن تخلفات حوزه‌ها یکی از مهم‌ترین شرایط یک انتخابات درست است.
۶. هر رای‌دهنده بتواند به کمک ابزارهای رمزنگاری اطمینان حاصل کند که رای او شمرده شده است. این شرط به این معنی است که هر کاربری که دانش کافی داشته باشد باید بتواند از درستی انتخابات - بدون نیاز به اعتماد به حوزه یا حتی ناظر انتخابات - اطمینان حاصل کند.
۷. رای‌دهنده نیازی به دانش یا توانایی خاصی برای رای‌دادن نداشته باشد. این نیازمندی برای دسترس‌پذیر نگه‌داشتن انتخابات لازم است.
۸. سیستم رای‌گیری نسبت به روش‌های فعلی رای‌گیری از دید کاربر تفاوت چندانی نداشته باشد. هر تغییر اساسی از نگاه رای‌دهنده باعث سختی نسبی انتخابات خواهد شد و هزینه‌ی اولیه استفاده از این سیستم انتخاباتی را به شدت افزایش خواهد داد.
۹. بتوان نتایج انتخابات را در بازه‌های زمانی معین دید. هر سیستم انتخاباتی که نتایج لحظه‌ای نشان دهد همواره در خطر حمله‌های مبتنی بر زمان در رابطه با حریم خصوصی رای‌دهندگان خواهد بود، به همین منظور نتایج انتخابات را می‌توان در بازه‌های زمانی که حریم خصوصی را به خطر نیندازد نشان داد.
۱۰. بتوان از شماره‌دهی شدن تمامی آرا بعد از انتخابات اطمینان حاصل کرد. به دلیل الزام مخفی نگه‌داشتن زمان حدودی ارسال هر رای، نتایج نهایی انتخابات تنها بعد از اتمام رای‌گیری قابل اتکاست.

## ۳.۴ فرضیات مسئله

در این تحقیق هدف ارائه‌ی یک پروتکل رای‌گیری امن است و در این روش به جزئیات پیاده‌سازی و مسائل مسئله‌ی شناسایی یک مسئله‌ی مهم در هر انتخابات است، با توجه به این که افراد واجد شرایط بسته به هر انتخابات تغییر می‌کنند در این مسئله فرض می‌کنیم که هر رای‌دهنده یک جفت کلید خصوصی و عمومی دارد

که قبل از فرایند انتخابات توسط ناظر انتخابات تایید شده است.

وظیفه‌ی حفظ امنیت کلید عمومی و خصوصی هر کاربر به عهده‌ی خود کاربر خواهد بود چرا نشانگر هویت کاربر در سیستم کلید عمومی او خواهد بود. هر چند که برای رای دادن در حوزه اطلاعات شناسایی کاربر با کلید عمومی او تطابق داده خواهد شد.

هر رای دهنده برای ثبت رای نیاز به یک دستگاه هوشمند دارد، از آن جایی که این دستگاه صرفاً برای امضای کورکورانه استفاده می‌شود برای دسترس‌پذیری بالاتر می‌توان این دستگاه را در حوزه ارائه کرد و رای‌دهندگان تنها کلید خصوصی خود را در آن وارد کنند. بدیهیست که انجام این کار نیاز اعتماد به حوزه را بالاتر می‌برد چرا که رای‌دهنده راهی برای اطمینان حاصل کردن از این که دستگاه برنامه‌ی درستی را اجرا می‌کند ندارد.

## ۴.۴ مثال شهودی

برای بدست آوردن دید کلی در راه حل ابتدا یک مثال شهودی از یک مدل رای‌گیری متمرکز را بررسی می‌کنیم، سپس در ادامه از این روش برای ایجاد سیستم توزیع‌شده و الکترونیکی خود استفاده می‌کنیم.

یک رای‌گیری را فرض می‌کنیم که در آن به ازای هر رای‌دهنده یک کاغذ نام رای‌دهنده و یک برگه‌ی رای وجود دارد. همچنین یک صندوق به ازای هر کاندیدا وجود دارد و همه‌ی این اطلاعات در معرض دید عموم هستند.

آلیس برای رای دادن یکی از کاندیداها را انتخاب می‌کند، مسئول حوزه یک کاغذ رمز شده که در آن ک شماره‌ی تصادفی  $r$  و کاندیدای موردنظر خود، باب، نوشته شده را به آلیس می‌دهد تا امضا کند، آلیس پس از اطمینان حاصل کردن از درستی ثبت کاندیدا (ولی بدون فهمیدن  $r$ ) آن را امضا می‌کند. ثبت مسئول حوزه این عبارت رمز شده و برگه‌ی رای مربوط به آلیس را به تخته می‌چسباند. این فرایند را ثبت رای می‌نامیم.

در ادامه پس از مدتی مسئول حوزه تخته مراجعه‌ی می‌کند و شاهدی بر تخته ثبت می‌کند که ثابت می‌کند او یک کلیدرمزی می‌داند که یکی از کاغذهای روی تخته را باز می‌کند که نتیجه‌ی آن  $r$  و باب است. سپس یکی از رای‌های روی تخته را برمی‌دارد و به صندوق باب می‌اندازد. این فرایند را شمارش رای می‌نامیم.

شاهد آلیس در تخته ثبت شده و دیگر نمی‌توان اثباتی ارائه کرد که رای آلیس را دو بار بشمارد، چرا که عدد  $r$  آن تکراری خواهد بود.



از طرفی از آن جا که مسئول حوزه نشان نداده که کدام عبارت رمز شده به باب رای داده، در نتیجه حریم خصوصی آلیس محفوظ می ماند.

چون حوزه ی انتخابات محل امنی برای رای دادن است راهی برای مجبور کردن آلیس به رای دادن به کاندیدای خاصی نخواهد بود و با اضافه کردن صندوق رای ممتنع می توانیم اطمینان حاصل کنیم که کسی آلیس را مجبور به رای دادن نیز نکرده است.

## ۵.۴ فرایند رای گیری از نگاه کاربر

یکی از مهم ترین قسمت های طراحی یک سیستم رای گیری تاثیر آن بر رای دهنده است چرا که در انتخابات های بزرگ هزینه ی تغییر رفتار رای دهندگان یا سخت تر شدن فرایند رای گیری به هر نحوی می تواند باعث کاهش شرکت رای دهندگان و افزایش هزینه ی تغییر رای گیری شود.

فرایند رای گیری را به دو بخش قبل از رای گیری و در حوزه ی رای گیری تقسیم می کنیم.

### ۱.۵.۴ قبل از رای گیری

قبل از رای گیری هر فرد واجد شرایط باید از یک روش امن از ناظر انتخابات یک جفت کلید عمومی و خصوصی دریافت کند و یا کلید عمومی خود را در سامانه ی مربوط ثبت کند. این کلید می تواند در قالب یک فایل بر روی یک دستگاه هوشمند - مثلاً تلفن همراه یا حتی یک کارت هوشمند - باشد. این مرحله می تواند در هر انتخابات تکرار شود یا یک فرایند ابتدایی باشد و برای انتخابات های بعدی نیز استفاده شود.

### ۲.۵.۴ در حوزه ی رای گیری

در حوزه ی رای گیری رای دهنده پس از ورود کارت شناسایی و کلید عمومی خود را ارائه می کند، در صورت تایید اطلاعات کاربر، کاربر با دستگاه هوشمند خود به سیستم حوزه متصل می شود و رای خود را از بین کاندیداهای ممکن و یا ممتنع وارد می کند و پیام تایید را دریافت می کند.

## ۶.۴ فرایند رای گیری از دید حوزه

در این بخش فرایند رای گیری را از نگاه حوزه بررسی می کنیم، در این بخش صرفاً منطق پروتکل را بررسی می کنیم و جزئیات بلاک چین و عملیات توزیع شده نمی پردازیم.

برای ثبت رای یک نفر در ابتدا آن فرد با کارت شناسایی احراز هویت می شود و از طریق ارتباط با ناظر انتخابات درستی کلید عمومی آن فرد بررسی می شود. در مرحله ی بعدی با ارائه ی کلید خصوصی کاربر یک تراکنش ثبت با نتیجه ی مورد نظر ایجاد می کند و تراکنش ثبت توسط دستگاه هوشمند کاربر کوکورانده امضا می شود. در این مرحله تراکنش شمارش نیز ایجاد می شود و با تاخیر زمانی در بلاک چین ذخیره می شود.

### ۱.۶.۴ تراکنش ها

در این بخش تراکنش ها ثبت و شمارش را تعریف می کنیم.

#### تراکنش ثبت

در تراکنش ثبت رای از یک کلید عمومی به دسته ی رای های منتظر شمارش منتقل می شود. هر تراکنش ثبت شامل یک رای و یک رشته <sup>۱</sup> رمز شده است که حاوی یک عدد تصادفی  $s$  و حساب مقصد  $d$  است که با کلید تصادفی  $k$  رمز شده است. این عبارت رمز شده رای  $CM$  می نامیم. سپس  $CM$  (نه خود  $s$ ) در بلاک چین به همراه رای ثبت می شود.

$$CM = enc_k(r, d) \quad (۱.۴)$$

#### تراکنش شمارش

مدل دیگر تراکنش ممکن تراکنش شمارش است که در آن شاهی برای شمارش رای ارائه می شود. برای این تراکنش رای دهنده اثبات بی دانشی برای دو موضوع ارائه می کند: یک  $C$  می شناسد به طوری که  $C \in$

<sup>۱</sup> string

$C_1, C_2, \dots, C_n$  و یک رشته‌ای  $r$  می‌داند که  $C$  را به  $s$  و  $d$  باز می‌کند. در نتیجه‌ی این اثبات یکی از رای‌های ثبت‌شده به  $d$  منتقل می‌شود.

لازم به ذکر است که از آنجایی که در تراکنش شمارش  $C$  و  $r$  نشان داده نشده‌اند، راهی برای فهمیدن این که رای متعلق به چه کسی است وجود ندارد.

## ۲.۶.۴ فرایند ثبت رای کاربر

در این بخش پروتکل ثبت رای حوزه را به طور دقیق بررسی می‌کنیم، در این بخش منظور از کاربر دستگاه هوشمند اوست.

۱. کاربر گزینه‌ی مورد نظر خود را انتخاب می‌کند، حوزه به تعداد  $n$  تا  $CM$  می‌سازد که رای کاربر به کاندیدای مورد نظر را نشان می‌دهند، هر کدام را با یک کلید تصادفی رمز می‌کند و برای تایید به کاربر می‌دهد.

۲. یکی از عبارات رمز شده به طور تصادفی توسط کاربر  $CM_c$  انتخاب می‌شود و به حوزه اعلام می‌شود.

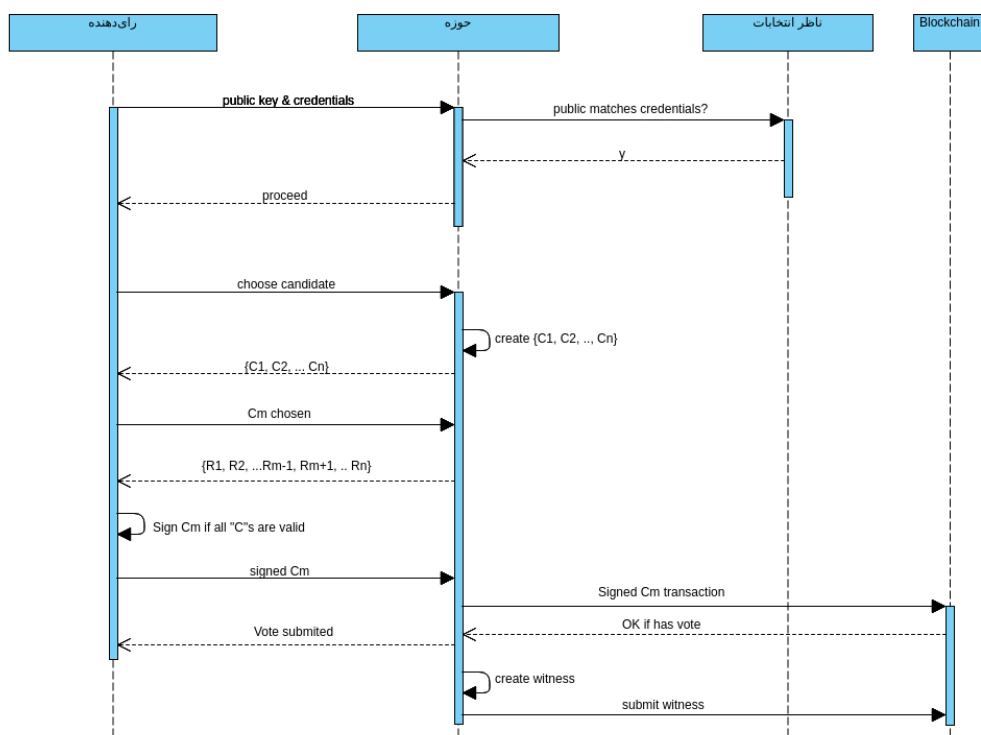
۳. حوزه کلید مربوط به تمامی  $CM$  های دیگر را به کاربر ارائه می‌کند.

۴. کاربر همه‌ی  $CM$  ها را باز می‌کند و چک می‌کند که در همه‌ی آن‌ها رای به نام کاندیدای مورد نظر او ثبت شده باشند. با این روش کاربر با احتمال  $\frac{n-1}{n}$  از درستی رای در  $CM_c$  مطمئن می‌شود. سپس کاربر  $CM_c$  را با کلید خصوصی خود امضا می‌کند و به حوزه می‌دهد. در نتیجه‌ی این کار کاربر راهی برای فهمیدن عدد تصادفی در  $CM_c$ ، که آن را  $s$  می‌نامیم، ندارد.

۵. حوزه  $CM_c$  را به همراه رای موجود در حساب کاربر به عنوان یک تراکنش ثبت در بلاک چین ذخیره می‌کند. در صورتی که رای کاربر قبلاً ثبت شده باشد در این مرحله خطا رخ می‌دهد و رای ثبت نمی‌شود.

۶. حوزه یک شاهد برای شمارش رای کاربر می‌سازد و هر دوی این تراکنش‌ها در بلوک بعدی بلاک چین ذخیره می‌شوند.

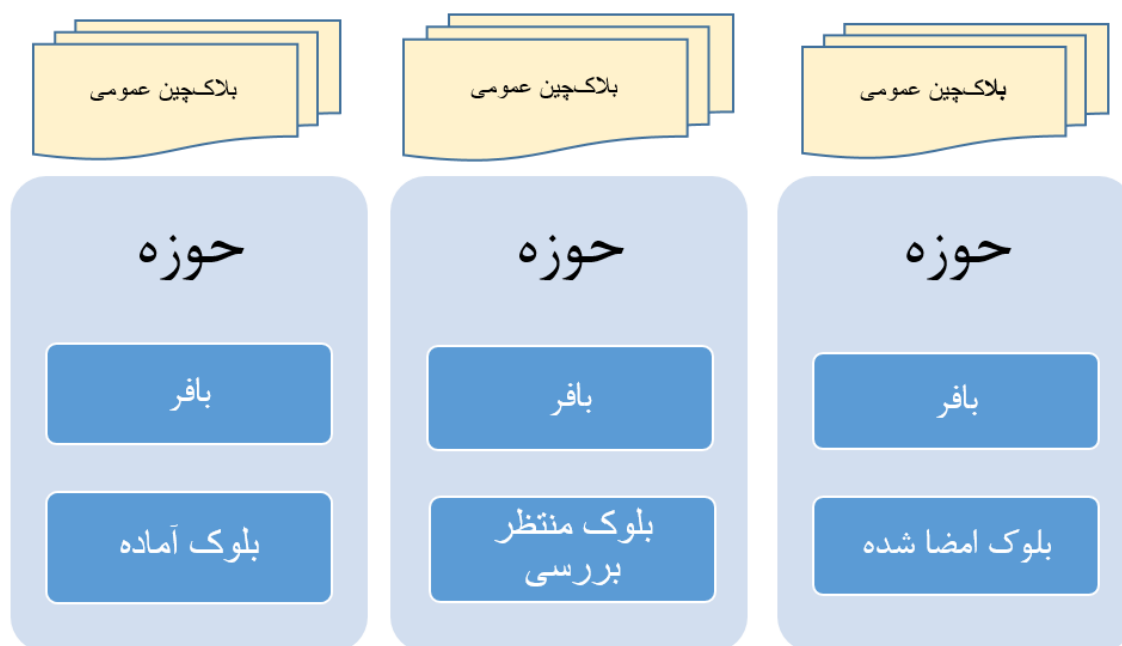
شکل ۱۰۴ این توالی فرایند را نشان می‌دهد.



شکل ۱.۴: فرایند ثبت رای در حوزه

## ۷.۴ شمای کلی

به طور کلی سیستم از تعدادی حوزه تشکیل می‌شود که روی یک بلاک چین توافق می‌کنند. همچنین هر حوزه ممکن است بلوک آماده شده و منتظر گرفتن تایید از بقیه‌ی حوزه‌ها داشته باشد، یا این که بلوکی از حوزه‌ی دیگری گرفته باشد که باید بررسی و امضا کند، شکل ۲.۴ شمای کلی درشت‌دانه‌ی سیستم را نشان می‌دهد. بلوک ابتدایی این بلاک چین به ازای هر فرد واجد شرایط یک کلید عمومی و یک رای دارد. همچنین یک آدرس خروجی به ازای هر کاندیدا وجود دارد که تعداد رای‌هایی که به آن آدرس فرستاده شده باشند رای‌های آن کاندیدا است.



شکل ۲.۴: شمای منطقی سیستم

#### ۱.۷.۴ اضافه شدن بلوک

می‌دانیم که در هر بلوک ثبت‌شده در بلاک‌چین تعدادی از دو مدل تراکنش‌های ثبت و شمارش داریم، همچنین هش بلوک قبلی را نیز برای اطمینان از تغییر نکردن بلوک‌های قبلی نگه‌می‌داریم. مسئله‌ای که در اینجا باقی می‌ماند نحوه‌ی توافق روی یک بلاک‌چین است. در این تحقیق از یک بلاک‌چین عمومی و بسته استفاده می‌کنیم.

هر حوزه یک بافر برای نگهداری تراکنش‌های مربوط به آرا دارد که قبل از ثبت در بلاک‌چین که آن را تبدیل به یک بلوک می‌کند. برای اضافه شدن هر تراکنش روی بلاک‌چین باید حداقل نصف به علاوه‌ی یکی از حوزه‌ها روی آن توافق کنند. هر حوزه برای توافق بررسی می‌کند که تراکنش‌های مربوط به رای دادن و شاهدهای ثبت شده در بلوک جدید درست باشند و هش بلوک قبلی نیز در بلوک صحیح باشد. هر حوزه‌ی بلاک‌چین را با امضای دیجیتال خود تایید می‌کند.

## ۲.۷.۴ مقادیر اولیه برای ZK-SNARK

همانطور که قبلاً اشاره کردیم برای استفاده از روش ZK-SNARK برای ایجاد شاهدهای بی‌دانش احتیاج به توافق روی نقاط اولیه‌ای روی یک elliptic curve داریم. برای انجام این کار از روش ارائه شده در تحقیق [۴۰] استفاده می‌کنیم. در این روش برای ایجاد نقاط اولیه از تقسیم مسئله بین افراد توافق‌کننده استفاده می‌شود، به صورتی که حاصل ضرب اطلاعات همه‌ی افراد نقاط اولیه را تشکیل می‌دهند و برای لورفتن آن باید تمامی حوزه‌ها تبانی کنند. در این روش خود برای ایجاد پارامترها از یک حالت خاص اثبات‌های بی‌دانش استفاده می‌شود.

## ۳.۷.۴ توافق

در بلاک‌چین‌های عمومی معمولاً یک مسئله در توافق حالت‌های بیست که بلاک‌چین دو شاخه می‌شود، یعنی دو مدل از بلاک‌چین وجود داشته باشد که هر کدام را قسمتی از شبکه به عنوان بلاک‌چین درست بشناسند. در بسیاری از ارزهای دیجیتال طولانی‌ترین بلاک‌چین از نظر تعداد بلوک‌ها همواره به عنوان نسخه‌ی درست شناخته می‌شود اما در یک سیستم رای‌گیری این کار باعث کم شدن رای می‌شود و این ریسک قابل قبولی نیست.

## قضیه‌ی CAP

این قضیه‌ی معروف [۴۲] اثبات می‌کند که در یک دیتابیس توزیع‌شده - مانند یک بلاک‌چین - در هر بازه‌ای حداکثر می‌توان دو شرط از سه شرط همخوانی<sup>۱</sup>، در دسترس بودن<sup>۲</sup> و تحمل قسمت‌شدن<sup>۳</sup> را می‌توانند داشته باشند. با توجه به این که هیچ زمانی نمی‌توانیم روی درستی شبکه حساب کنیم سیستم ما در حال CP عمل می‌کند. معمولاً در سیستم‌ها مبتنی بر بلاک‌چین از روش‌های که گارانتی می‌کنند که همه‌ی نودها در نهایت به همخوانی می‌رسند<sup>۴</sup>. اما در یک سیستمی که برای رای‌گیری استفاده می‌شود این مسئله می‌تواند خطر گم شدن تعدادی از آرا ایجاد کند و این خطر معقولی برای یک سیستم رای‌گیری الکترونیکی نیست.

در بلاک‌چین‌های عمومی معمولاً از روش‌های اثبات کار و اثبات سهم استفاده می‌شود که دسترس پذیری

<sup>1</sup> Consistency

<sup>2</sup> Availability

<sup>3</sup> Partition tolerance

<sup>4</sup> Eventual consistency

و تحمل قسمت شدن را به خوبی ارائه می کنند اما ممکن است شاخه های <sup>۱</sup> لحظه ای پیش بیاید و چند نسخه از بلاک چین درست وجود داشته باشد، معمولاً در این روش ها طولانی ترین بلاک چین نسخه ی درست در نظر گرفته می شود و بلاک چین های کوتاه تر حذف می شوند. از آن جایی که این کار در سیستم ما باعث از بین رفتن رای می شود باید در روش دیگری استفاده کنیم.

در این تحقیق سه روش برای تفاوق را بررسی می کنیم. روش اول Aura [۴۳] نام دارد. در این روش نود یک نود رهبر بلوک های جدید ارائه می کند و در طور زمان در دوره های مشخصی رهبر تغییر می کند. برای انتخاب زمان تغییر از ساعت unix استفاده می شود که می تواند در اثر همگام sync نبودن ساعت نودهای حاضر در شبکه در یک لحظه دو رهبر وجود داشته باشد. این باعث از بین رفتن همخوانی در سیستم می شود ولی سیستم همواره در دسترس خواهد بود.

روش دیگر Clique [۴۴] نام دارد. این روش شبیه Aura عمل می کند با این تفاوت که برای همگامی نودها به جای استفاده از ساعت، از تعداد بلوک ثبت شده در بلاک چین استفاده می شود، همچنین در این روش، نودهایی غیر از نود رهبر نیز می توانند بلوک جدید پیشنهاد کنند. این کار می تواند باعث ایجاد شاخه در بلاک چین شود اما در طول زمان با تغییر رهبر یکی از دو شاخه حذف خواهد شد. این روش نیز در نهایت به همخوانی می رسد اما ریسک حذف شدن رای را سیستم وارد می کند.

راه حلی در در پروتکل ارائه شده در این تحقیق برای این موضوع استفاده کردیم PBFT<sup>۲</sup> [۴۵] نام دارد. این روش هزینه ی محاسباتی بسیار کمتری از روش های مبتنی بر اثبات کار دارد و از نظر مقیاس پذیری [۴۶] نیز بسیار بهتر عمل می کند.

### Practical Byzantine Fault Tolerance

در این روش در ابتدای رای گیری حوزه ها در یک زنجیره ی اولویت چیده می شوند و برای ثبت هر بلوک جدید، بلوک آماده شده را به رهبر (حوزه ای با بیشترین اولویت) داده می شود و حوزه آن را به تمامی حوزه های دیگر ارسال می کند. سپس تمامی حوزه های دیگر بعد از تایید نتیجه را برای هم ارسال می کنند و در صورت موفقیت آن را ثبت

<sup>۱</sup> fork

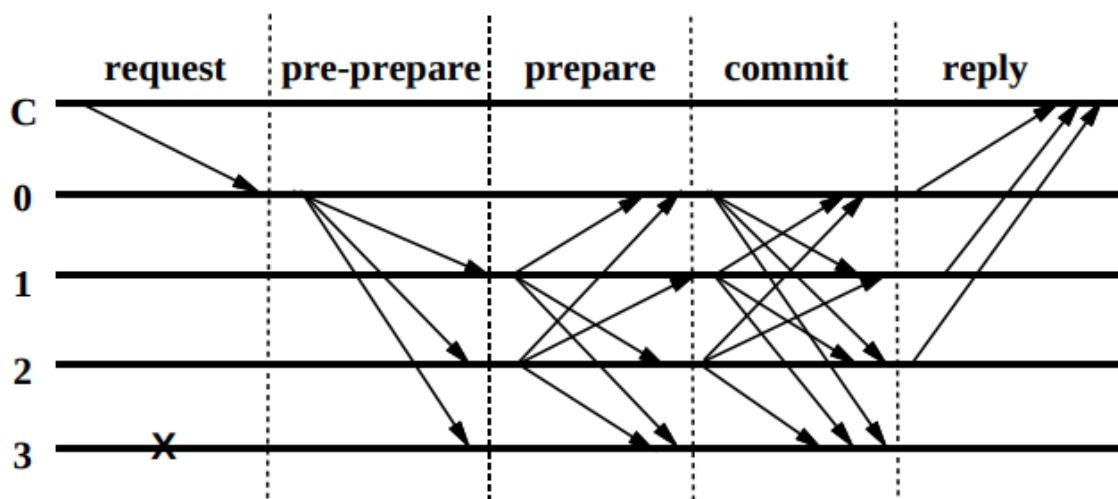
<sup>۲</sup> Practical Byzantine fault tolerance

جدول ۱.۴: روش توافق

اثبات کار	همخوانی	دردسترس بودن	تحمل تقسیم شدن
اثبات سهم	همخوانی در نهایت	دارد (اما ممکن است بلوک حذف شود)	دارد
Aura	گارانتی نمی‌کند	دارد	دارد
Clique	همخوانی در نهایت	دارد (اما ممکن است بلوک حذف شود)	دارد
PBFT	دارد	در صورت تقسیم شدن ندارد	دارد

می‌کنند، بعد از تایید نتایج برای حوزه‌ی ابتدایی ارسال شده و ثبت نهایی می‌شود.

همچنین بعد از اضافه شدن هر بلوک حوزه‌ی رهبر تغییر می‌کند و به نفر بعدی در زنجیره‌ی اولیت می‌رسد. همچنین در صورت جواب ندادن حوزه‌ی رهبر در مدت زمان مشخص یا جواب‌های غلط دادن مسئولیت به نفر بعدی منتقل می‌شود.



شکل ۳.۴: روش PBFT

در این روش حداکثر تعداد تعداد حوزه‌ای که باید خطا کار باشند تا بلوک اشتباهی در بلاک چین ثبت شود برای  $3f + 1$  حوزه  $f$  حوزه است. این تعداد حوزه‌ی خراب کار می‌توانند بلوک‌های اشتباهی ثبت کنند اما به دلیل



استفاده از امضای دیجیتال بعد از اتمام رای گیری با بررسی درستی بلوک های ثبت شده در بلاک چین بلوک های خطا به سادگی قابل شناسایی هستند. در همچنین شرایطی چون نمی توان از تعداد رای دهندگانی که رای آنها ثبت نشده اطمینان پیدا کرد نتایج رای گیری باید مردود محسوب شود.

در روش PBFT اثبات می شود [۴۷] تا زمانی که کمتر از یک سوم حوزه ها خطا کار باشند هیچ زمانی دو نسخه ی مستقل از بلاک چین توسط حوزه ها تایید نمی شود. به عبارت دیگر این روش به طور کامل هم خوانی و توانایی قسمت شدن را دارد.

لازم به ذکر است که در این حالت سیستم همواره در دسترس نیست. به این معنی که ممکن است موقعیتی پیش بیاید که یک حوزه به دلیل قطع شدن شبکه از بقیه ی حوزه ها نتواند بلوک جدید ثبت کند، اما می تواند در بافر خود اطلاعات آرا را نگه دارد و زمان اتصال به شبکه بلوک جدید را بسازد و ثبت کند.

## فصل ۵

### تحليل و ارزیابی

## ۱.۵ پیاده سازی

برای ساخت اثبات‌های بی‌دانش از کتابخانه‌ی `libsnaark`<sup>۱</sup> استفاده شده است. این کتابخانه از الگوریتم پینوکیو برای ساخت اثبات استفاده می‌کند و توسط `Zcash` نیز برای ساخت اثبات استفاده شده است. همچنین برای بالا آوردن سیستم از `docker` استفاده شده است. در فرایند کامپایل این کتابخانه از تنظیمات زیر استفاده شده است:

جدول ۱.۵: تنظیمات `libsnaark`

نام متغیر	مقدار	توضیحات
CURVE	ALT_BN128	مدل خم مورد استفاده با ۱۲۸ بیت امنیت
MULTICORE	ON	استفاده از چند هسته برای موازی سازی
USE_PT_COMPRESSION	OFF	سرعت بالاتر در ازای حجم شاهد‌های بزرگتر
PROFILE_OP_COUNTS	OFF	شمارش تعداد فعالیت روی خم

بقیه‌ی تنظیمات کتابخانه در حالت پیش فرض استفاده شده است.

برای بلاک‌چین و حوزه‌ها از `hyperledger fabric`<sup>۲</sup> که یک بستر قراردادهای هوشمند است استفاده شده است. `hyperledger fabric` توسط شرکت `IBM` طراحی شده و توسط `Linux Foundation` نگهداری می‌شود. این پروژه یک بلاک‌چین خصوصی ارائه می‌کند که قسمت‌های مختلف آن مانند روش توافق به سادگی قابل تغییرند.

## ۲.۵ معیارهای مقایسه

برای مقایسه این روش رای گیری با روش‌های دیگر جنبه‌های زیر را بررسی می‌کنیم:

- هزینه‌ی زیرساخت

<sup>۱</sup> <https://github.com/scipr-lab/libsnaark>

<sup>۲</sup> <https://www.hyperledger.org/projects/fabric>

- هزینه‌ی هر رای‌گیری
- توانایی ردگیری درستی رای‌گیری
- هزینه برای کاربر
- درصد اطمینان از درست شمارده شدن رای
- توانایی بررسی ناظر انتخابات
- سختی مجبور کردن یک کاربر برای رای دادن به یک کاندیدای خاص

## مراجع

- [1] R. L.Lamport and M.Pease, "The byzantine generals problem," *ACM Transactions on Programming Languages and Systems( TOPLAS)*, vol.4, no.3, pp.382-401, 1982.
- [2] S. Nakamoto, "Bitcoin : A peer-to-peer electronic cash system," 2008.
- [3] A. beck, "Hashcash : a denial of service counter-measure," 2008.
- [4] S. S.King, "Ppcoin : Peer-to-peer crypto-currency with proof-of-stake.," *self-published paper*, 2012.
- [5] N. D.Schwartz and A.Britto, "The ripple protocol consensus algorithm," *Ripple Labs Inc White Paper*, 2014.
- [6] B.Chase and E.MacBrough, "Analysis of the xrp ledger consensus protocol," *Ripple Labs Inc White Paper*, 2018.
- [7] D.Mazieres, "The stellar consensus protocol : A federated model for internet-level consensus," 2015.
- [8] E. B. Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza, "Zero-cash : Decentralized anonymous payments from bitcoin," *2014 IEEE Symposium on Security and Privacy*, pp.459-474, 2014.
- [9] N. Szabo, "Smart contracts," [http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Lecture/02\\_11\\_SmartContracts.html](http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Lecture/02_11_SmartContracts.html), 1993.
- [10] "Ethereum foundation, ethereum whitepaper, a next-generation smart contract and decentralized application platform," <https://github.com/ethereum/wiki/wiki/White-Paper>, 2014.
- [11] M. N.Atzei and T.Cimon, "A survey of attacks on ethereum smart contracts," *Proceedings of the 6th International Conference on Principles of Security and Trust*, vol.10204, pp.164-186, 2017.
- [12] A. Juels, A. Kosba, and E. shi, "The ring of gyges : Investigating the future of criminal smart contracts," *Proceedings of ACM CCS*, pp.283-295, 2013.

- [13] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making smart contracts smarter," *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp.254-269.
- [14] M. F. B. M. R. B. Q.Xu, C.Jin and K.M.M.Aung, "Blockchain-based decentralized content trust for docker images," *Multimedia Tools and Applications*, vol.77, no.14, 2018.
- [15] D. Chaum, "Security without identification: Transaction systems to make big brother obsolete," *Communication of the ACM*, vol.28, no.1, pp.1030-1044, 1985.
- [16] D. Chaum, "Security without identification: Transaction systems to make big brother obsolete," *Advances in Cryptology Proceedings of Crypt*, vol.82, no.3, pp.199-203, 1983.
- [17] T. O. B. Fujioka and K. Ohta, "A practical secret voting scheme for large scale elections," *LNCS 718, Advances in Cryptology - ASIACRYPT*, pp.244-251, 1992.
- [18] L. F. Cranor and R. K. Cytron, "Sensus: A security-conscious electronic polling system for the internet," *Proceedings of the Hawai i International Conference on System Sciences*, 1997.
- [19] B. W. DuRette, ""multiple administrators for electronic voting," *B.Sc thesis, MIT*, 1999.
- [20] B. N. L. C. S. M. Wright, M. Adler, ""an analysis of the degradation of anonymous protocols," *In Proceedings of Network and Distributed System Security Symposium*, 2002.
- [21] B. S. R. Cramer, M. Franklin and M. Yung, "Multi-authority secret-ballot elections with hnear work," *LNCS 1070, Advances in Cryptology - EUROCRYPT*, pp.72-83, 1996.
- [22] R. G. R. Cramer and B. Schoenmakers, "A secure and optimally efficient multi-authority election scheme," *LNCS 1233, Advances in Cryptology - EUROCRYPT*, pp.103-118, 1997.
- [23] B. Schoenmakers, "A simple publicly verifiable secret sharing scheme and its application to electronic voting," *LNCS 1666, Advances in Cryptology - CRYPTO*, pp.148-164, 1999.
- [24] N. L. R. DeMillo and M. Merritt, "Cryptographic protocols," *Proceedings of the 14th Annual Symposium on the Theory of Computing*, pp.383-400, 1984.
- [25] D. Malkhi and E. Pavlov, "Anonymity without 'cryptography'," *Proceedings of Financial Cryptography*, pp.117-135, 2001.
- [26] O. D.Malkhi and E.Pavlov, "E-voting without 'cryptography'," *In International Conference on Financial Cryptography*, pp.1-15, 2002.
- [27] R. Osgood, "The future of democracy: Blockchain voting," *COMP116: Information Security*, 2016.

- [28] C.-L.-B. L.Vo-Cao-Thuy, K.Cao-Minh and T.A.Nguyen, "Security without identification : Transaction systems to make big brother obsolete," *IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF)*, pp.1-6, 2019.
- [29] U. . G. E.Yavuz, A.Kaan Koç, "Towards secure e-voting using ethereum blockchain," *6th International Symposium on Digital Forensic and Security (ISDFS)*, pp.1-7, 2019.
- [30] K. kirby, A. Masi, and F. Maymi, "Votebook : A proposal for a blockchain-based electronic voting system," <https://www.economist.com/sites/default/files/nyu.pdf>, 2016.
- [31] M. F.Hjalmarsson, G.Hrei arsson and G.Hjalmtysson, "Blockchain-based e-voting system.," *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, pp.983-986, 2018.
- [32] S. Z.Galil and M.Yung, "Symmetric public-key encryption," *Conference on the Theory and Application of Cryptographic Techniques. Springer.*, 1985.
- [33] Y. E.Ben-Sasson, I.Bentov and M.Riabzev, "Pinocchio : nearly practical verifiable computation," *IEEE Symposium on Security and Privacy*, pp.238-252, 2019.
- [34] Y. E.Ben-Sasson, I.Bentov and M.Riabzev, "Scalable zero knowledge with no trusted setup," *Annual International Cryptology Conference*, pp.701-732, 2019.
- [35] E. Z. A.Kosba, A.Miller and C.Papamanthou, "Hawk : The blockchain model of cryptography and privacy-preserving smart contracts," *IEEE Symposium on Security and Privacy*, pp.839-858, 2016.
- [36] N. Saberhagen, "Cryptonote v 2.0," <https://cryptonote.org/whitepaper.pdf>, 2013.
- [37] B.Goodell1 and S.Noether2, "Thring signatures and their applications to spender-ambiguous digital currencies," *Monero Research Lab*, 2018.
- [38] M. I.Miers, C.Garman and A.D.Rubin, "Zerocoin : Anonymous distributed e-cash from bitcoin," *IEEE Symposium on Security and Privacy*, pp.397-411, 2013.
- [39] A. S.Bowel and M.D.Green, "A multi-party protocol for constructing the public parameters of the pinocchio zk-snark," *International Conference on Financial Cryptography and Data Security*, pp.64-77, 2018.
- [40] M. E. M. V. E.Ben-Sasson, A.Chiesa, "Secure sampling of public parameters for succinct zero knowledge proofs," *EEE Symposium on Security and Privacy*, pp.287-304, 2015.
- [41] A. J. R.S.Wahby, I.Tzialla and M.Walfish, "Doubly-efficient zksnarks without trusted setup," *IEEE Symposium on Security and Privacy (SP)*, pp.926-943, 2018.

- 
- [42] E. Brewer, "Cap twelve years later : How the "rules" have changed," *Computer*, vol.45, no.2, pp.23-29, 2012.
- [43] "Clique," <https://github.com/ethereum/EIPs/issues/225>.
- [45] M.Castro and B.Liskov, "Practical byzantine fault tolerance," *OSDI*, pp.173-186, 1999.
- [46] X. K. H.Sukhwani, J.M.Martinez and A.Rindos, "Performance modeling of pbft consensus process for permissioned blockchain network (hyperledger fabric)," *36th Symposium on Reliable Distributed Systems (SRDS)*, pp.253-255, 2017.
- [47] M. Castro and B. Liskov, "Practical byzantine fault tolerance and proactive recovery," *CM Transactions on Computer Systems (TOCS)*, vol.20, no.4, p.398-461, 2002.



## **Abstract:**

Since Bitcoin's wide adaption in 2009 there has been a abundance of trustless applications based of Bitcoin's use of blockchain technology and after the release of Ethereum's smart contract platform we are seeing more and more usages of smart contracts. With this increase in usage of these platforms we on must be mindful of the security implications of these platforms.

In This research we first review the basics of digital currencies and their underlying technologies and then review the security considerations of their platforms and the applications based on them and finally move to voting as a usecase of these platforms and consider the challenges we face while implementing such a system.

**Keywords:** Blockchain, Ethereum, Security, Smart Contracts, Voting



Shahid Beheshti University

Faculty of Computer Science & Engineering

# **Usage and Security of Blockchain in Smart Contracts**

By

**Shervin Hajiesmaili**

A THESIS SUBMITTED  
FOR THE DEGREE OF  
MASTER OF SCIENCE

Supervisor :

**Dr. Maghsoud Abbaspour**

2018