WHITE PAPER

Pure Pound: A Peer-to-Peer Electronic Cash System, Loyalty Program, and Virtual Currency

AN OPEN-SOURCED PROTOCOL ANCHORED ON CRYPTOLOGY, TRAP DOOR MATHEMATICS, & GAME THEORY

The 4th Generation Cryptocurrency for the IoT and the 4th Industrial Revolution

Introduction. The first industrial revolution in 1765 marked the transition of world economy from small-scale agriculture to mechanized industry. This resulted to the invention of the steam engine that boosted transportation and global trade in all aspects. Over a century after, the second wave started following the synthesis of gas, oil, and electricity. This discovery made mass production possible. The third wave in 1969 came at the dawn of nuclear age, where transistors and circuits became the new commodity, giving rise to computers and redefining commerce.

To quote the white paper source of BTC, "…there is one big problem - commerce on the internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments. While the system works well enough for most transactions, it still suffers from the inherent weaknesses of the trust based model. Completely non-reversible transactions are not really possible, since financial institutions cannot avoid mediating disputes. The cost of mediation increases transaction costs, limiting the minimum practical transaction size and cutting off the possibility for small casual transactions, and there is a broader cost in the loss of ability to make non-reversible payments for non- reversible services. With the possibility of reversal, the need for trust spreads. Merchants must be wary of their customers, hassling them for more information than they would otherwise need. A certain percentage of fraud is accepted as unavoidable. These costs and payment uncertainties can be avoided in person by using physical currency, but no mechanism exists to make payments over a communications channel without a trusted party.

What is needed is an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact directly with each other without the need for a trusted third party. Transactions that are computationally impractical to reverse would protect sellers from fraud, and routine escrow mechanisms could easily be implemented to protect buyers. In this paper, we propose a solution to the double-spending problem using a peer-to-peer distributed timestamp server to generate computational proof of the chronological order of transactions. The system is secure as long as honest nodes collectively control more CPU power than any cooperating group of attacker nodes."

When Satoshi Nakamoto created the BTC protocol in 2008, he never imagined the impact of mobile devices in the near future. At that time, Android was nothing but a small company knocking at Google's door steps. Apple was just warming up on its own version of a phone. Everyone else was highly reliant on desktops and laptops as aids in computing. This explains the foundation on which BTC was established and on why, it is not adaptable to innovation.

In 2018, the world is a completely different place. Mobile phones become as powerful as high-end desktops released in 2008. On their specifications alone, performance have increased nearly fifty folds. Following this trend, people have likewise become highly mobile, to the point that most applications are more valuable than their desktop counterparts. Put it simply, the world will become more and more mobile in the years to come.

The MDG of the United Nations in year 2000 included a provision of enhancing communication for underdeveloped nations as means of improving lives and economic activities. Realizing this, Facebook and Google started the roll-out of various platform to bring internet around the world. The idea is simple – since generations to come are expected to be highly mobile, having internet wherever in the world is the next logical step.

This inevitable trend in mobility around the world, paves way for the next generation of cryptocurrencies that are highly tied to the mobile platform. This means, every capable phone out there can mine, giving people more options.

The fourth industrial revolution is seen to catalyze all of these, and in turn, will change the world as we see and experience it. It will decentralize finance and global trade, giving way to new types of industries. It will blur the line between the real, unreal, physical, chemical, biological, and other aspects perceived by the senses.

This wave of development will bring digitalization to almost all aspects of human life, with key emphasis to currency and speedy inter-industry communication.

Abstract. Pure Pound (PUP) is a purely peer-to-peer version of electronic cash that would allow online payments to be sent directly from one party to another without going through a financial institution. It is anchored on the POUND Protocol (PP). Digital signatures and verification provide part of the solution, but the main benefits are lost if a trusted

third party is still required to prevent double-spending. PP proposes a solution to the double-spending problem using a secured peer-to-peer network verification. The network signs and timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The most dominant record, hereby referred to as a log, not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of mobile computing power. As long as a majority of mobile power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain possessed by majority of the network users, as proof of what happened while they were gone.

PUP is the next level cryptocurrency that combines the stability of fiat currency along with the efficiency of virtual currency. In essence, it mimics the natural macro-economic properties of fiat currency making it an ideal replacement. Though cryptocurrencies were not originally planned as substrate for trading, people soon discovered that the best way to maximize its value potential is to hold it while its cheap and sell it when its expensive.

The nature of cryptocurrencies as a decentralized unit of value not only possess advantages but also disadvantages. For one, the lack of a central authority results to fluctuating trading prices which means, investors and believers are at the mercy of illiquid market effects. Secondly, mining has become extremely difficult that the cost toppled the reward at a ratio of 10:3.

PUP also answers many of the problems that current cryptocurrencies face namely – scalability, speed, and accessibility. The protocol behind PUP allows it to reach a maximum number capped and tied to the total gold existing in the global economy. This is way bigger than BTC, which remains as one of the most popular cryptocurrency in the world.

Speed is one area where PUP challenges the status quo. Using a two-way verification process, it removes the burden from the network by preventing unverified transactions to be resolved by others' computing power. It relies on personal attesting, a process that shortlists a transaction based on the secured log present within the author's device. It is based on the **PP First Premise** which states that "If a device contains the most recent log and is secured at the point in time where the transaction is authored, that same device can be trusted to verify a transaction, including the one it authored."

The Premises. These are decision pillars that serve as guide for resolving disputes within the transfer network.

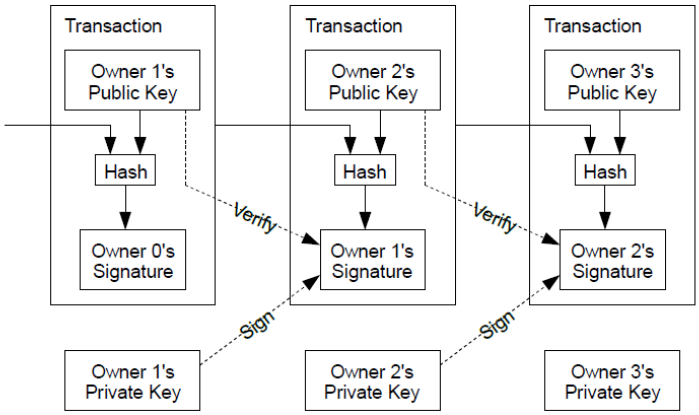
1. If a device contains the most recent log and is secured at the point in time where the transaction is authored, that same device can be trusted to verify a transaction, including the one it authored;
2. If device A is the author and it contains the very same log that majority of the network possesses, and that a random set of devices named after A, say B is chosen, containing the same log as what the network possesses, both A and B MUST share the same and exact rights for verification;
3. Following the first and second premise, a verification and validation performed by device A is self-sufficient and thus, the same step undertaken by devices B, C, or D are all confirmatory in nature and should conform with the condition set forth by device A's verification;
4. The primary determinant for a transaction to be validated is the account balance. Hence, so long as the account balance is the same as compared to that of the APL and the SSLS that majority of the network users possess, the account can be therefore trusted;
5. The impact of cracking and hacking a server is lessened with the network possess a summary of the transactions being compromised – when, and if, over half of the network believes their personal logs to be accurate, a compromised APL or SSLS could not severely affect the network operations;

The Protocol. POUND is an acronym for Progressive Over/Under and Nod/Denial protocol in which the algorithmic foundation is anchored. In theory, the protocol is based on the premise of a "community network" that collectively receives and process transactions authored by any member of the network. In practice, the protocol organizes a transaction chain called "**the matrix**".

The currency in use is called as PURE. Publicly, it is referred to as Pure Pound, whose supply is finite.

Supply and Demand. The POUND protocol is patterned after real-life economic models, reflecting trading activities and exchanges. The total number of coins that can be mined is 7.5 trillion. This is based on the conservative estimate value of all gold mined across the course of human history. The approximate value of a pure pound coin is at one cent of a US Dollar. Since the supply is finite, future transactions at the time when all coins are mined will have transaction fees shouldered by the author.

Transactions. We define an electronic coin as a verified request guaranteed by an inter-dependent and decentralized network community. Each owner transfers the coin to the next by digitally signing the coin with a hash, verifying it locally, and endorsing it to the network for the same purpose, before signing and writing it to the public log.



The problem of course is the payee can't verify that one of the owners did not double-spend the coin. A common solution is to introduce a trusted central authority, or mint, that checks every transaction for double spending. After each transaction, the coin must be returned to the mint to issue a new coin, and only coins issued directly from the mint are trusted not to be double-spent. The problem with this solution is that the fate of the entire money system depends on the company running the mint, with every transaction having to go through them, just like a bank. We need a way for the payee to know that the previous owners did not sign any earlier transactions. For our purposes, the earliest transaction is the one that counts, so we don't care about later attempts to double-spend. The only way to confirm the absence of a transaction is to be aware of all transactions. In the mint based model, the mint was aware of all transactions and decided which arrived first. To accomplish this without a trusted party, transactions must be publicly announced [1], and we need a system for participants to agree on a single history of the order in which they were received. The payee needs proof that at the time of each transaction, the majority of nodes agreed it was the first received.

The Process. The user's phone will serve as an apparatus to mine, store, or trade PUP. Two major platforms will serve as the launching point of the application - namely Android and iOS. Desktop versions and web-based solutions will follow soon. In essence, using a mobile phone for computing power is more efficient and fairly equal as compared to using desktops. For one, monetary resources create the playing field unequal when PCs are used for mining tokens. The upgradability of computer parts, processors, and other components required for computing power creates an advantage to those who can afford it. Since today's mobile phones lack the flexibility of upgrades, it makes the field equal for all.

All transactions within the PUP network is arranged and controlled by a network of mobile phones interlaced by the platform they use. Each mobile phone downloads a copy of the log report. The log report is a list of all events that happened in the network.

Each mobile phone is considered as a node.

Upon downloading, the node will stamp the application using the device's model number, phone number, carrier, name of user, birthday, secret question, answer to secret question, time, date, and zone when the application was downloaded. This set of information will be hashed using SHA-256 and will become the users private key.

The private key is therefore static.

The public key is randomly given to an account holder through a code embedded within the program protocol. For security, the public key randomly changes every transaction to preserve anonymity. The public key is dynamic based on SHA-256.

The node holder will have a unique username and password stored locally. Three incorrect password attempts will disable the account which resolves to dissolution of the tokens within the wallet. Accounts maybe opened at different nodes, but the security property is shared across the platform regardless of the node. A correct password entry will reset the incorrect password value to zero. No password reset is offered.

Steps. The procedure to run the network are as follows:

1. All transactions require an internet connection;
2. The node will download the latest SSLS as commanded by the protocol;
3. Only then can a particular node author a transaction;
4. The transaction will be shortlisted (locally-verified) using the current SSLS;
5. Once the transaction is shortlisted (i.e., the app agrees that the current balance is equal or greater than the requested amount), the protocol will endorse the transaction to the matrix;
6. Along with this, the local balance of the author will be deducted which prevents the possibility of double-spending by ensuring that only the amount synchronized by the trusted SSLS is spent;
7. Even if an author attempts to fool the system, it will be virtually impossible since the local balance restricts him from doing so, based on the shortlisting principle;
8. Even if the author can crack the local code, the network will detect it; and transaction will be denied;
9. The protocol will randomly choose verifiers among the senders or stand-by minders that will authenticate the shortlisting performed by the author;
10. Once authenticated, the transaction will be encoded to the APL, and the SSLS will be updated;
11. The process will sync

Community Server. A central repository of previous logs, independent from any third-party control, is maintained by the network through a wallet governed by a condition. The community server allows the retrieval of logs, and its processing to extract information that are relevant to the network.

Timestamp Server. The solution that PUP proposes begin includes a timestamp server. A timestamp server works by taking a transaction to be timestamped and then publishing the stamp, such as in a newspaper post. The timestamp proves that the data must have existed at the time, obviously, in order to get into the matrix.

Conditions. This is an algorithm within the POUND protocol that actively executes once triggered by pre-set conditions. It is permanent in nature and cannot be modified once created.

Characters. The PP identifies each member of the network based on the role that they portray in each transaction. An author is the principal writer of a transaction. The verifier is any member of the network that was assigned to verify a certain transaction.

Shortlisting. Double payment is an issue that plagues almost all cryptocurrency based on the block chain technology. This impacts the customer experience. To be a truly global platform, the speed of processing transactions should significantly improve. The strength of block chain relies on its public ledger. Ironically, it also causes its major weakness. Since each block can only contain a finite number of transactions, the speed at which it is processed is

relatively slower as compared to centralized systems like that of Visa. Centralized processing is more than 1000% faster that block chain-based system.

To address this, the processing procedure is to be divided into segments. The first segment is called shortlisting. Once an author publishes a transaction, the application on his mobile phone will locally verify the request against the recently downloaded summary of the APL. A successful transaction gets a **local nod** and is **endorsed** to the matrix.

The APL. All transactions that occurred within the network is listed in the Actual Public Log (APL). This is a raw collection of data that is stored in a community server maintained by the community. This list is perpetually permanent and cannot be changed by anyone.

In BTC standards, this is referred to as the full ledger. The APL is not distributed passively to the network due to bandwidth and storage limitations. It can be viewed publicly and can be downloaded by any member of the network as needed. In lieu of an APL download, a processed data summary is broadcasted.

The SSLS. Every time a mobile application connects to the internet, the application proactively downloads a processed and organized copy of the APL. This is called a Super Secured Log Summary (SSLS). The SSLS contains processed information ready for use and reference by any network device.

This process makes local shortlisting possible. The application will sync the users wallet balance with the recent copy of the SSLS. Once a request is made, the shortlisting procedure will dictate local processing comparing the transaction versus the updated balance from the SSLS.

If the shortlisting procedure is denied, the transaction will not enter the matrix. Only transactions that get a nod are endorsed for network processing.

Security. Users have the option to attach meta-data to their chains whenever they desire to. The same is true for conditions that can be used to program a particular transaction.

P.O.U.N.D. A protocol unique to the PUP system ensures timely and accurate processing of all transactions. The process is progressive and is based on two unique conditions – over/under and nod/denial. Once an author publishes a transaction, the shortlisting procedure ensures that only valid and pre-screened transactions entre the matrix.

The Matrix. A list of both new and old transactions is called a matrix. It is arranged in an algorithmic order. Once an author's transaction is published, his mobile phone is required to contribute by offering computing power to validate and verify other transactions. The author could not verify its own transaction in the matrix as it is self-serving. Doing so is also redundant as the LVP ensures that the author's balance is enough to augment the transaction published.

The Matrix is composed of super blocks, each containing a cryptographic hash, a data, and a time stamp. Once encoded to the APL, the super block adds the hash of the super block before it, and the hash of the super block after it if any. A chain also cements the position of each block. A chain contains the data and time stamp of the block before and after any given super block.

Problem Solving and Proof of Work. Cryptocurrencies' biggest challenge is managing double spending. To address this, all transactions within the network are verified locally and by the nodes using mathematical computation. In exchange of computing power, successful transactions organized by the node will yield a token reward.

Trust is a primary issue in a P2P Network like PUP. A distributed ledger in the form of APL via the SSLS provides a solution. The APL contain raw data, while the SSLS contain processed information. To implement a distributed ledger (APL) server on a peer-to-peer basis, we will need to use a proof- of-work system similar to Adam Back's Hashcash, rather than newspaper or Usenet posts. The proof-of-work involves providing computing power to order transactions and generating rewards for doing so.

For the PUP protocol, we implement permanent recording of transaction orders and account balances such that, once a node was able to satisfy the required proof of work and the corresponding reward has been encoded, it becomes impossible to change the entry in the APL (and SSLS too) without redoing the proof of work. As later blocks are chained after it, the work to change the block would include redoing all the blocks after it.

The proof-of-work also solves the problem of determining representation in majority decision making. If the majority were based on one-node-one-vote, it could be subverted by anyone able to allocate many nodes. Proof-of-work is essentially one-node-one-vote. Since verifiers are chosen on a random order, the chance of a user being chosen per verification relatively becomes lesser as more users join the network. Even if we assume that an author tries to fool a network by enrolling more nodes, the cost of maintaining such will overpower the advantages of doing so. If the network is composed mostly of honest nodes, dishonest nodes will become insignificant and least likely that an anomaly will occur.

To modify a past block in the APL, an attacker would have to redo the proof-of-work of the block and all blocks after it and then catch up with and surpass the work of the honest nodes. We will show later that the probability of a slower attacker catching up diminishes exponentially as subsequent blocks are added. To compensate for increasing hardware speed and varying interest in running nodes over time, the proof-of-work difficulty is determined by a moving average targeting an average number of blocks per hour. If they're generated too fast, the difficulty increases.

New transaction broadcasts do not necessarily need to reach all nodes. As long as they reach many nodes, they will get into a block before long. Block broadcasts are also tolerant of dropped messages. If a node does not receive a block, it will request it when it receives the next block and realizes it missed one.

Mining. The author will be paid for every transaction verified. The protocol will dictate the author's device to verify a transaction two levels below his line, while simultaneously verifying the transaction over it, and under it. This means verifying three transactions progressively and simultaneously at the same time. An approved transaction, unanimously verified by at least three miners, gets a **system nod** and is ready for **encoding**.

By convention, the first transaction in a block is a special transaction that starts a new coin owned by the creator of the block. This adds an incentive for nodes to support the network, and provides a way to initially distribute coins into circulation, since there is no central authority to issue them. The steady addition of a constant of amount of new coins is analogous to gold miners expending resources to add gold to circulation. In our case, it is mobile computing power, time, effort, and electricity that is expended.

Every time a transaction is solved and verified, a PUP is generated as a reward and is split among the verifiers (miners). In this sense, all transactions are paid for by the protocol, which results to zero cost for the author.

The algorithm will randomly choose someone among the three miners, to encode the verified transaction in the APL. In certain instances, more verifiers maybe required to validate a certain transaction.

Reclaiming Disk Space. Once the latest transaction in a coin is buried under enough blocks, the spent transactions before it can be discarded to save disk space. To facilitate this without breaking the block's hash, transactions are hashed in a Merkle Tree [7][2][5], with only the root included in the block's hash. Old blocks can then be compacted by stubbing off branches of the tree. The interior hashes do not need to be stored.
Block
Block Header (Block Hash)

Transactions Hashed in a Merkle Tree After Pruning Tx0-2 from the Block
A block header with no transactions would be about 80 bytes. If we suppose blocks are generated every 10 minutes, 80 bytes * 6 * 24 * 365 = 4.2MB per year. With computer systems typically selling with 2GB of RAM as of 2008, and Moore's Law predicting current growth of 1.2GB per year, storage should not be a problem even if the block headers must be kept in memory.

Order. Ideally, an author is required to solve (process) transactions and is given top priority. However, if he failed to show up, unwilling, unable to solve transactions due to bandwidth or storage issues, or have voluntarily given up, a special lottery will be held. All verifiers in stand-by mode will attempt to solve a mathematical problem and the winner takes over the slot. The process should be no more than ten seconds.

Speed and Efficiency. The PUP system relies on mobility and network cooperation. All transactions anchored on the protocol should be no more than ten seconds in duration to complete. The goal of the PUP protocol is to process infinite number of transactions per second.

8. Simplified Payment Verification
It is possible to verify payments without running a full network node. A user only needs to keep a copy of the block headers of the longest proof-of-work chain, which he can get by querying network nodes until he's convinced he has the longest chain, and obtain the Merkle branch linking the transaction to the block it's timestamped in. He can't check the transaction for himself, but by linking it to a place in the chain, he can see that a network node has accepted it, and blocks added after it further confirm the network has accepted it.
Longest Proof-of-Work Chain As such, the verification is reliable as long as honest nodes control the network, but is more vulnerable if the network is overpowered by an attacker. While network nodes can verify transactions for themselves, the simplified method can be fooled by an attacker's fabricated transactions for as long as the attacker can continue to overpower the network. One strategy to protect against this would be to accept alerts from network nodes when they detect an invalid block, prompting the user's software to download the full block and alerted transactions to confirm the inconsistency. Businesses that receive frequent payments will probably still want to run their own nodes for more independent security and quicker verification.

The Software. The development will focus on four platforms – two for mobile devices, and two for non-mobile. Versions for Android and iOS are to be developed first, followed by Windows and Mac OS shortly thereafter. For mobile versions, the application should emulate a messaging platform where authors could communicate real time to recipients and process transactions from there. It should provide a seamless, streamlined, and wonderful experience to the user.

Philosophy. Trust, reliability, security, and ease of use are the for key pillars of user experience that the PUP protocol aims to achieve. Our vision is to connect all business and industries together, using smart technology and unified payment systems.

Timeline. The launch of the PUP protocol in real-life application will be divided into periods, corresponding to the major objective anchored on each. The periods are: introductory, ICO, mobile, general launch, desktop, and progressive.

March 11 & 18, 2018 – Start of Introductory Period
This period will focus on marketing and public awareness. Seminars, symposiums, meetings, trainings, and talks will be organized to introduce POUND algorithm in the public. At this period, genesis tokens and pre-mined coins can be sold at a pre-selling price fifty centavos or more, but less than three pesos. The building of third-party exchanges are also seen in this period.

May 5a, 2018 – Start of ICO Period
Following the Introductory Period is selling tokens more than one peso. At this period, the mobile application is expected to launch and trading is expected to commence.

September 1, 2018 – Mobile Period
Other features of the mobile application is expected to roll-out. At this period, trading is at its mature phase.

December 1, 2018 – General Period
The PUP protocol is sorted-out for application to other industries, most especially to devices enabling IoT.

March 11, 2019 – Desktop Period
Desktops, mobile devices, and servers need to connect with one another for seamless customer experience.

May 1, 2019 - Progressive Period
Characterized by optimizations and improvements for other uses of the POUND protocol.

Privacy. The traditional banking model achieves a level of privacy by limiting access to information to the parties involved and the trusted third party. The necessity to announce all transactions publicly precludes this method, but privacy can still be maintained by breaking the flow of information in another place: by keeping public keys anonymous. The public can see that someone is sending an amount to someone else, but without information linking the transaction to anyone. This is similar to the level of information released by stock exchanges, where the time and size of individual trades, the "tape", is made public, but without telling who the parties were.
Traditional Privacy Model
Transactions
New Privacy Model
Identities Transactions
As an additional firewall, a new key pair should be used for each transaction to keep them from being linked to a common owner. Some linking is still unavoidable with multi-input transactions, which necessarily reveal that their inputs were owned by the same owner. The risk is that if the owner of a key is revealed, linking could reveal other transactions that belonged to the same owner.

Risk Management Protection. Security protocol is the prime foundation of cryptocurrency. Unlike centralized systems, securities on block chain networks are heavily reliant on encryption, proof of work, and network consensus. Like any other system currently available, the POUND Protocol is vulnerable to these challenges.

In order to address these issues, the following security protocols are embedded:

Super Majority Wins – if a segment of the network is compromised, the prevailing SSLS on most of the “secured” or “non-compromised” will overwrite the SSLS on the “unsecured” or “compromised” segment of the network. This process is called **healing**.

Network Nods Win – if a user attempts to pirate the local application on his device, and the transaction is endorsed to the network, the protocol and algorithmic principle will detect fraud, posing a blockade of the compromised account. The account is prevented from endorsing any transaction in the matrix until it is healed. The user is prompted to uninstall the current application and reinstall a new version.

Download then Transact (DTT) Principle – this dictates the application to download the recent SSLS immediately after being disconnected from the network. This ensures that updated balances are reflected accurately in the user’s account. Only after successful sync will transactions be allowed.

Local then Network (LTN) Principle – this ensures that local verification (shortlisting) is performed before endorsing any transaction to the network. This process decongests the matrix, minimizes file size, and conserves internet bandwidth.

Freezing - a process that halts activity in one account to prevent it from compromising other nodes in the network. This may also be done reversely by freezing unaffected nodes as a precautionary measure.

Risk Calculations. We consider the scenario of a dishonest node (an attacker) trying to generate an alternate transaction log faster than that organized by the honest nodes. Even if this is accomplished, it does not throw the system open to arbitrary changes, such as creating value out of thin air or taking money that never belonged to the attacker. Nodes are not going to accept an invalid transaction as payment, and honest nodes will never accept an entry containing them. An attacker can only try to change one of his own transactions to take back money he recently spent. However, doing so entails a lot of work and computing power.

The race between the honest nodes and a dishonest node can be characterized as a Binomial Random Walk. The success event is the honest nodes' logs being extended by one block, increasing its lead by +1, and the failure event is the attacker's chain being extended by one block, reducing the gap by -1.

The probability of an attacker catching up from a given deficit is analogous to a Gambler's Ruin problem. Suppose a gambler with unlimited credit starts at a deficit and plays potentially an infinite number of trials to try to reach breakeven. We can calculate the probability he ever reaches breakeven, or that an attacker ever catches up with the honest chain, as follows:

Let p = the probability an honest node solves a transaction
Let q = the probability a dishonest node solves a transaction
Let q_z = the probability a dishonest node will ever catch up from z blocks behind

$$q_z = \begin{cases} 1 & \text{if } p \le q \\ (q/p)^z & \text{if } p > q \end{cases}$$

Given our assumption that $p > q$, the probability drops exponentially as the number of blocks the attacker has to catch up with increases. With the odds against him, if he doesn't make a lucky lunge forward early on, his chances become vanishingly small as he falls further behind.
We now consider how long the recipient of a new transaction needs to wait before being sufficiently certain the sender can't change the transaction. We assume the sender is an attacker who wants to make the recipient believe he paid him for a while, then switch it to pay back to himself after some time has passed. The receiver will be alerted when that happens, but the sender hopes it will be too late.
The receiver generates a new key pair and gives the public key to the sender shortly before signing. This prevents the sender from preparing a chain of blocks ahead of time by working on it continuously until he is lucky enough to get far enough ahead, then executing the transaction at that moment. Once the transaction is sent, the dishonest sender starts working in secret on a parallel chain containing an alternate version of his transaction.
The recipient waits until the transaction has been added to a block and z blocks have been linked after it. He doesn't know the exact amount of progress the attacker has made, but assuming the honest blocks took the average expected time per block, the attacker's potential progress will be a Poisson distribution with expected value:

$$\lambda = z\frac{q}{p}$$

To get the probability the attacker could still catch up now, we multiply the Poisson density for each amount of progress he could have made by the probability he could catch up from that point:

$$\sum_{k=0}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \cdot \begin{cases} (q/p)^{(z-k)} & \text{if } k \le z \\ 1 & \text{if } k > z \end{cases}$$

Rearranging to avoid summing the infinite tail of the distribution...

$$1 - \sum_{k=0}^{z} \frac{\lambda^k e^{-\lambda}}{k!} \left(1 - (q/p)^{(z-k)}\right)$$

Converting to C code...

```
#include <math.h>
double AttackerSuccessProbability(double q, int z)
{
double p = 1.0 - q;
double lambda = z * (q / p);
```

```
double sum = 1.0;
int i, k;
for (k = 0; k <= z; k++)
{
double poisson = exp(-lambda);
for (i = 1; i <= k; i++)
poisson *= lambda / i;
sum -= poisson * (1 - pow(q / p, z - k));
}
return sum;
}
```

Running some results, we can see the probability drop off exponentially with z.

```
q=0.1
z=0 P=1.0000000
z=1 P=0.2045873
z=2 P=0.0509779
z=3 P=0.0131722
z=4 P=0.0034552
z=5 P=0.0009137
z=6 P=0.0002428
z=7 P=0.0000647
z=8 P=0.0000173
z=9 P=0.0000046
z=10 P=0.0000012
q=0.3
z=0 P=1.0000000
z=5 P=0.1773523
z=10 P=0.0416605
z=15 P=0.0101008
z=20 P=0.0024804
z=25 P=0.0006132
z=30 P=0.0001522
z=35 P=0.0000379
z=40 P=0.0000095
z=45 P=0.0000024
z=50 P=0.0000006
```

Solving for P less than 0.1%...

```
P < 0.001
q=0.10 z=5
q=0.15 z=8
q=0.20 z=11
q=0.25 z=15
q=0.30 z=24
q=0.35 z=41
q=0.40 z=89
q=0.45 z=340
```

Other Applications. POUND is expected to open the gates to new portals of technology such as voluntary data selling for individuals.

Trading. Holders may opt to open a trading account at their discretion, with a minimum wallet value of PhP 5,000 pesos relative to the XR_c. The price at which someone is willing to buy refers to the bid price, while the price at which someone is looking to sell is the ask price. The current pricing refers to the price of the last successful transaction incurred, regardless if it is a buy or a sell. The current pricing also refers to the XR_c.

Intermediaries. Third-party institutions are allowed to transact PUP subject to the laws and regulations of the country where they seek to operate. Brokers report directly to intermediaries, which maybe in the form of VC Exchange Centers or other analogous services. Intermediaries are subject to their own licensure examination protocols.

Advantages. Pure Pound follows the mathematical pattern of macro-economic principle:

Pure Pound value is highly affected by trade volume. Since Pure Pound has an existing data base that regularly requires the conversion if legal tender to VC, the value is preserved and non-erratic;

Value Preservation. Pure Pound value is affected by the economic rule of supply and demand which translates to the actual supply of the currency versus the actual demand.

Factors that upticks the value of Pure Pound:

Buying of PUP through Loyalty Program
Trading PUP higher than the current exchange rate (CXR)
Mining then selling mined PUP (mPUP) higher than the CXR
Positive news
Increasing performance

Factors that downticks the value of Pure Pound

Encashment of PUP
Trading PUP higher than the CXR
Selling mPUP lower than CXR
Negative news
Decreasing performance

Owning. Activities will yield a varying amount of PUP.

Solving circular polynomial functions using computing power of mobile phones;
Participating in the advertising program of intermediaries
Participating in the data mining program of intermediaries
Buying of products or PUP from the partner companies
Other activities where a company intends to pay for services rendered

Combining and Splitting Value. Although it would be possible to handle coins individually, it would be unwieldy to make a separate transaction for every cent in a transfer. To allow value to be split and combined, transactions contain multiple inputs and outputs. Normally there will be either a single input from a larger previous transaction or multiple inputs combining smaller amounts, and at most two outputs: one for the payment, and one returning the change, if any, back to the sender.

It should be noted that fan-out, where a transaction depends on several transactions, and those transactions depend on many more, is not a problem here. There is never the need to extract a complete standalone copy of a transaction's history.

Conclusion. We have proposed a system for electronic transactions without relying on trust. We started with the usual framework of coins made from digital signatures, which provides strong control of ownership, but is incomplete without a way to prevent double-spending. To solve this, we proposed a peer-to-peer network using proof-of-work to record a public history of transactions that quickly becomes computationally impractical for an attacker to change if honest nodes control a majority of CPU power. The network is robust in its unstructured simplicity. Nodes work all at once with little coordination. They do not need to be identified, since messages are not routed to any particular place and only need to be delivered on a best effort basis. Nodes can leave and rejoin the network at will, accepting the proof-of-work chain as proof of what happened while they were gone. They vote with their CPU power, expressing their acceptance of valid blocks by working on extending them and rejecting invalid blocks by refusing to work on them. Any needed rules and incentives can be enforced with this consensus mechanism.

OSCD (Open Source Community Development). Pure Pound assumes innovation as a vital ingredient in creating a truly global economy. As such, its community is leading the development of the platform by ensuring its adaptation to the prevalent demands of globalization.