Blockchain

Cours

Master 2 La Sorbonne Université

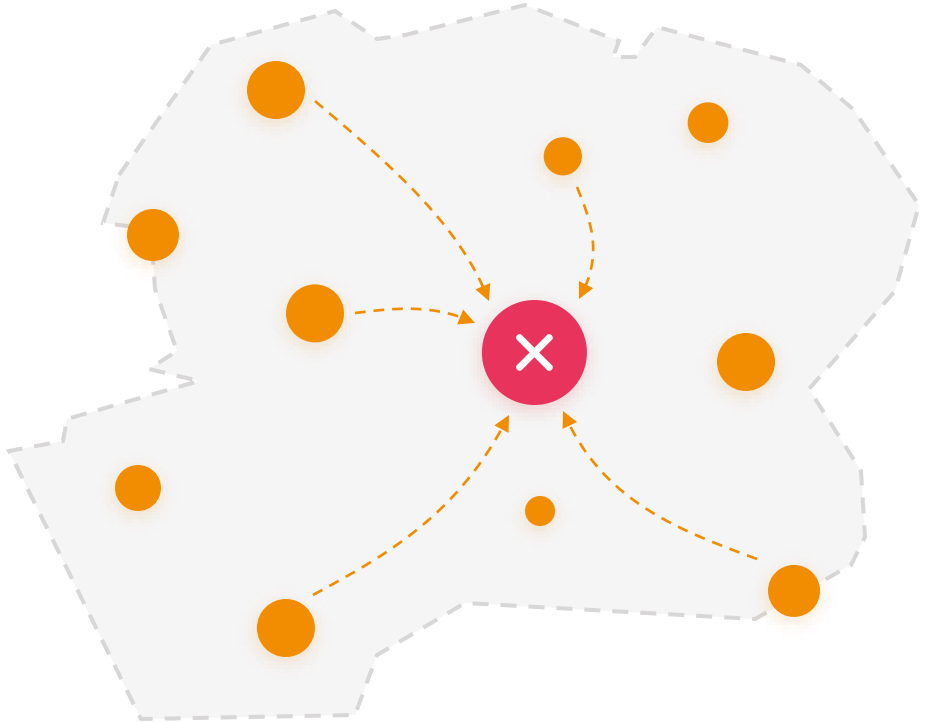
1

Problème des généraux byzantins

*Le problème des **généraux byzantins***

Le problème des généraux byzantins consiste en 2 objectifs contradictoires. Vous jouez ici le rôle d'un dirigeant d'une cité Etat.

1

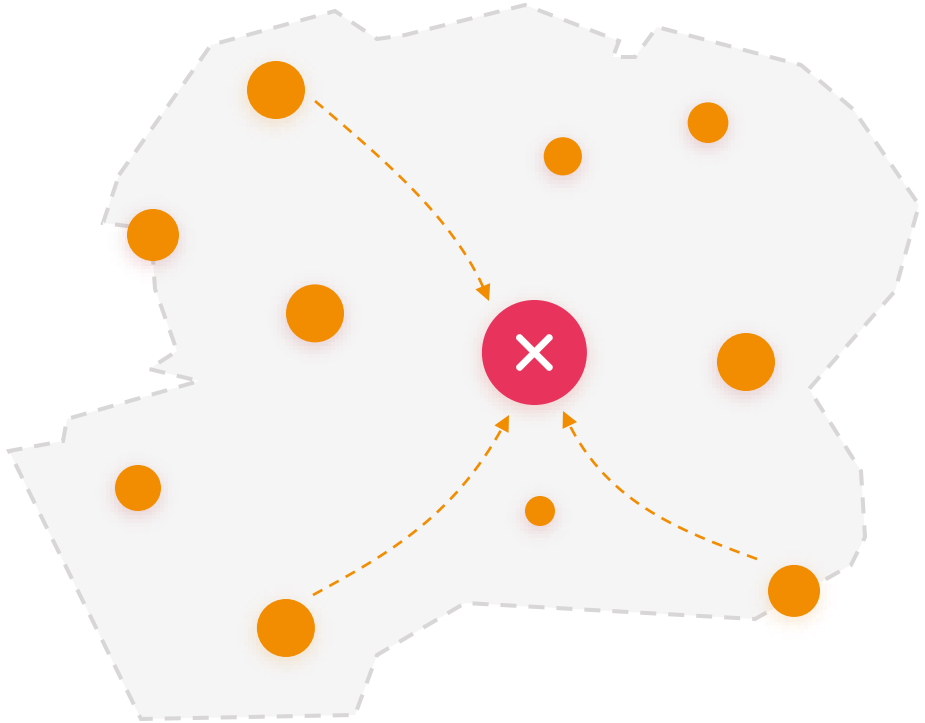


10 cités Etats pourraient empocher un immense butin en pillant une très grande ville, Byzance. Pour ce faire, elles doivent attaquer Byzance à **au moins 5**.

*Le problème des **généraux byzantins***

Le problème des généraux byzantins consiste en 2 objectifs contradictoires. Vous jouez ici le rôle d'un dirigeant d'une cité Etat.

2



Si elles attaquent à **moins de 5**, **elles sont vulnérables au pillage de leurs voisines**. Comment, avec éventuellement des traîtres parmi leurs émissaires, les dirigeants des cités peuvent-ils se coordonner efficacement?

2

La cryptographie de Bitcoin

Sécurité du Bitcoin 1/2

- La sécurité du Bitcoin repose sur l'utilisation de **l'algorithme ECDSA** (Elliptic Curve Digital Signature Algorithm) dans la génération d'une clé publique à partir d'une clé privée et dans la signature d'une transaction à partir d'une clé privée.

- L'algorithme ECDSA utilise une **courbe elliptique** d'équation :

$$y^2 = x^3 + ax + b$$

munie de la loi + présentée en annexe.

- Cette courbe est définie sur un corps fini d'entiers $\mathbb{F}_p = \mathbb{Z}/\mathbb{Z}_p$ où $p > 3$ est un nombre premier.

Sécurité du Bitcoin 2/2

- Pour la cryptographie du Bitcoin : courbe elliptique secp256k1 proposée par Certicom en 2000. Définir sur $\mathbb{F}_p$ avec $p = 2^{256} - 2^{32} - 977$ et des coefficients $a = 0$ et $b = 7$.
- Comme décrit en annexe, **on travaillera dans un sous-groupe composé des solutions $(x, y) \in \mathbb{F}_p^2$ de l'équation de la courbe elliptique et d'un point à l'infini qui sera l'élément neutre de la loi de transformation de ce groupe.**
- On définit l'ordre de G comme le plus petit entier n tel que la transformation nG (addition n fois du point G, au sens du sous-groupe de $\mathbb{F}_p$ considéré (corde-tangente)) donne O, point du sous groupe de $\mathbb{F}_p$ considéré.
- Les points utilisés dans le processus cryptographique de signature étant obtenus à partir d'addition de G avec lui-même, *on travaillera ensuite modulo n* afin que tous ces points ne soient pas égaux à O.
- Une **fonction de hachage** (ou hash function) est une fonction qui permet, à partir d'une donnée fournie en entrée, de calculer un résultat de taille fixe. La fonction de hachage utilisée dans les algorithmes de Bitcoin est SHA-256. Cette fonction produit des résultats d'une taille de 256 bits.
  Une fonction de hachage est déterministe et très complexe à inverser. Elle permet, à partir d'un hash, de vérifier l'authenticité d'un document en calculant le hash de celui-ci en le comparant au hash donné.
  Toutefois, à partir du hash d'un document, il est impossible de remonter aux données composant celui-ci.

Génération clé publique-clé privée et d'une signature 1/2

- L'utilisateur **génère une paire** $(\boldsymbol{s}, \boldsymbol{Q})$ composée d'une clé privée s comprise entre 0 et $p-1$ et d'une clé publique $Q = sG$, le générateur G ainsi que son ordre n étant connus.
  Remarque : Q et G ne permettent pas d'aboutir à s car sG correspond à s fois l'addition du point G avec lui-même. Il est donc **très complexe d'envisager de retrouver $\boldsymbol{s}$ par une opération du type $\boldsymbol{s = QG^{-1}}$.**
- Les données constituant la transaction (destinataire, montant de la transaction,…) composent un message que nous notons m. **Ce message est fonction de $\boldsymbol{Q}$ mais pas de $\boldsymbol{s}$.**
- Dans ce système cryptographique à clé privée-clé publique, l'identité d'un utilisateur est attestée par possession de la clé privée. Afin d'attester son identité sans partager sa clé privée, l'utilisateur peut générer une signature propre à chaque transaction, qui permettra aux autres utilisateurs de vérifier que l'émetteur de la transaction possède bien la clé privée qui y est liée.
- Si bien utilisé (voir diapositive suivante), ce procédé ne permet pas de remonter à la clé privée à partir de la signature d'une transaction. C'est pourquoi on parle de **cryptographie asymétrique.**
- Pour chaque message m, l'utilisateur choisit un entier k tel que $1 \leq k \leq n-1$, où n est l'ordre de G
- Il calcule les coordonnées du point $(i, j) = kG$ puis calcule $x = integer(i)\ mod(n)$. Si $x = 0$, on repart de la première étape.
- Le message m est hashé en une chaîne de bits de longueur inférieure ou égale à la longueur en bits de l'entier n. **Cette chaîne de bits est ensuite transformée en un entier $\boldsymbol{e = H(m)}$ avec H fonction de hachage cryptographique.**
- L'utilisateur calcule ensuite $y = \frac{(e+sx)}{k}\ mod(n)$. Si $y = 0$, on repart de la première étape.
- La signature du message m est alors la paire d'entiers (x, y).

Génération clé publique-clé privée et d'une signature 2/2

- Lors de cette opération, la clé privée s et l'entier k propre à chaque message ne sont pas révélés au réseau par l'émetteur du message.
- **L'entier k ne doit pas être communiqué**. S'il est communiqué, un autre utilisateur pourra obtenir la clé privée s de l'émetteur de la transaction par la formule

$$s = \frac{(ky - e)}{x} \; mod(n)$$

car x et y sont donnés dans la signature et e peut être recalculé.

- **L'entier k ne doit pas être utilisé afin de signer plusieurs messages différents.** Si un utilisateur utilise le même entier k et la même clé privée s pour signer deux messages m_1 et m_2 différents, produisant deux signatures (x, y_1) et (x, y_2) (on voit en effet, en reprenant le calcul précédent, que la première composante de la signature ne dépend pas du message m envoyé), l'entier k peut alors être retrouvé par la formule

$$k = \frac{(e_1 - e_2)}{(s_2 - s_1)} \; mod(n)$$

ce qui permet ensuite de remonter à la clé privée.

Vérification authenticité signature 1/2

L'authenticité d'une signature peut être vérifiée en appliquant l'algorithme suivant :

1. On vérifie que Q est différent de O (le point à l'infini) et que Q appartient bien à la courbe elliptique
2. On vérifie que nQ donne O
3. On vérifie que les entiers x et y sont compris entre 1 et $n-1$
4. On calcule $(i,j) = \left(H(m)y^{-1}(\mathrm{mod}\ n)\right)G + \left(xy^{-1}(\mathrm{mod}\ n)\right)Q$
5. On vérifie que $x = \mathrm{integer}(i)\ \mathrm{mod}(n)$

Vérification authenticité signature 2/2

On a :

$$\begin{aligned}
&\big(H(m)y^{-1}(\mathrm{mod}\ n)\big)G + \big(xy^{-1}(\mathrm{mod}\ n)\big)Q \\
&= \big(H(m)y^{-1}(\mathrm{mod}\ n)\big)\mathrm{G} + \big(xy^{-1}(\mathrm{mod}\ n)\big)sG \\
&= \big((H(m) + sx)y^{-1}\big)\ (\mathrm{mod}\ n)G \\
&= \big((H(m) + sx)k(H(m) + sx)^{-1}\big)\ (\mathrm{mod}\ n)G \\
&= \big(k\ (\mathrm{mod}\ n)\big)G \\
&= kG \\
&= (i, j)
\end{aligned}$$

Remarque : On a utilisé dans ces calculs le grand sous-groupe d'ordre n premier du groupe $E(\mathbb{F}_p)$ des $\mathbb{F}_p$- points rationnels de la courbe E sous-groupe composé des solutions $(x, y) \in \mathbb{F}_p^2$ de l'équation de la courbe et d'un point à l'infini qui sera l'élément neutre de la loi de transformation de ce groupe.

3

Le minage de Bitcoin

Introduction au Proof of Work

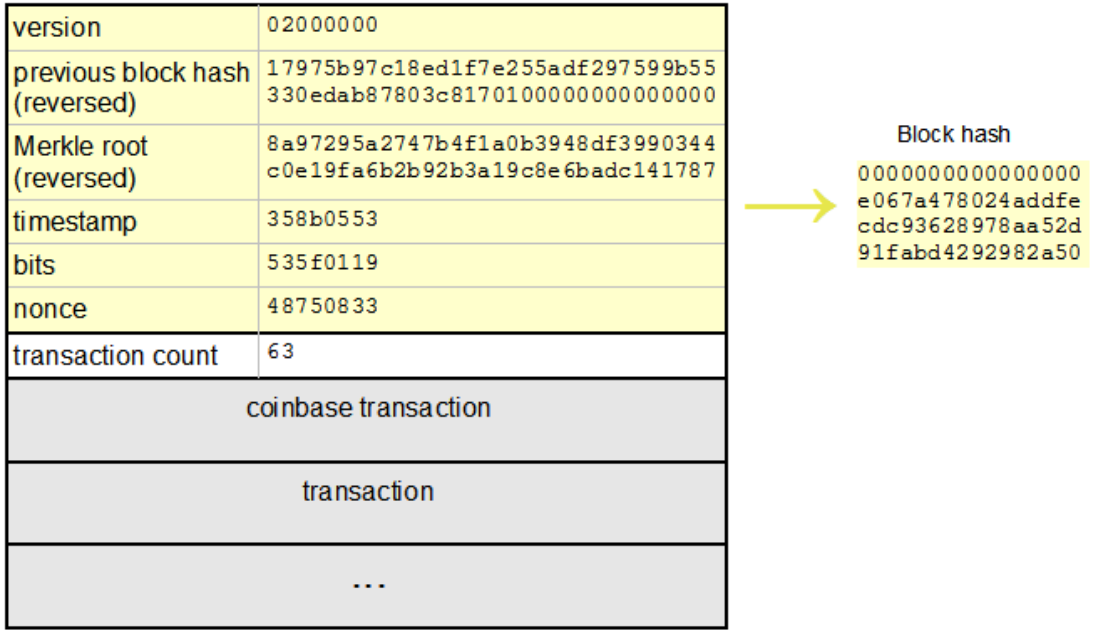
En plus de ces protocoles cryptographiques permettant l'utilisation sécurisée d'un système clé privée-clé publique, l'immutabilité de la blokchain Bitcoin et l'absence de fraudes ou mauvaises transactions est assurée par la structure de la base de données blockchain et l'utilisation d'un protocole dit de

« **Proof of Work** » (PoW).

La structure de la blockchain Bitcoin 1/3

Base de donnée structurée en blocs contenant plusieurs éléments :

- Le hash du bloc considéré
- La version du protocole respecté par le bloc
- Le hash du bloc précédent
- La Merkle root des transactions présentes dans le bloc
- Le timestamp du bloc
- La taille en mémoire du bloc
- Le « nounce » du bloc



version	02000000
previous block hash (reversed)	17975b97c18ed1f7e255adf297599b55 330edab87803c8170100000000000000
Merkle root (reversed)	8a97295a2747b4f1a0b3948df3990344 c0e19fa6b2b92b3a19c8e6badc141787
timestamp	358b0553
bits	535f0119
nonce	48750833
transaction count	63
coinbase transaction	
transaction	
...	

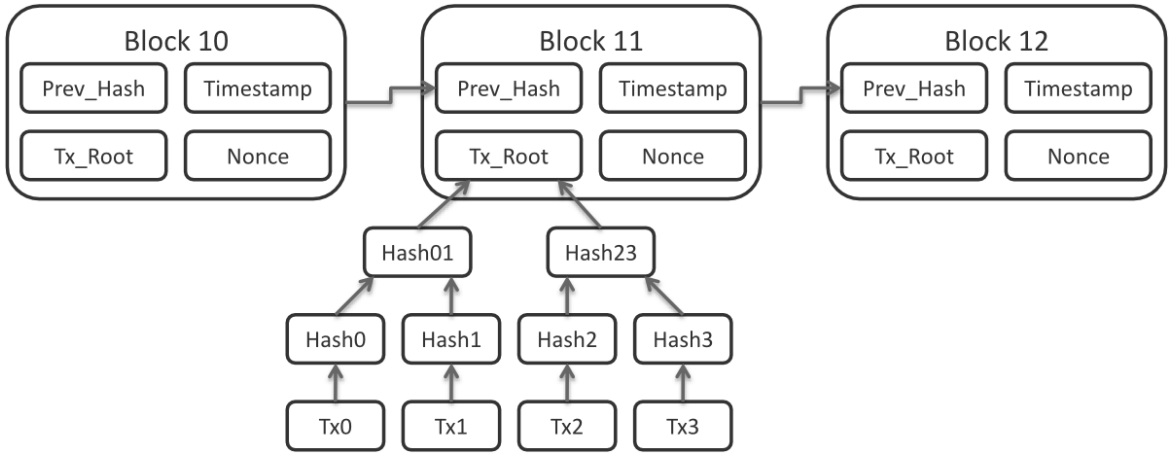
Block hash

0000000000000000 e067a478024addfe cdc93628978aa52d 91fabd4292982a50

La structure de la blockchain Bitcoin 2/3

La **Merkle root**, notée « Tx_Root » sur la figure ci-dessous est un nombre obtenu en calculant le hash de chaque transaction puis en calculant le hash de ces hash concaténés deux à deux, et ainsi de suite… Cette structure permet de pouvoir déterminer rapidement (en $o(\log n)$ où n est le nombre de transactions présentes dans le bloc) si une transaction est présente dans un bloc.

hash précédent|Merkle root|timestamp|taille|nounce $\xrightarrow{\text{SHA256}}$ hash du bloc



La structure de la blockchain Bitcoin 3/3

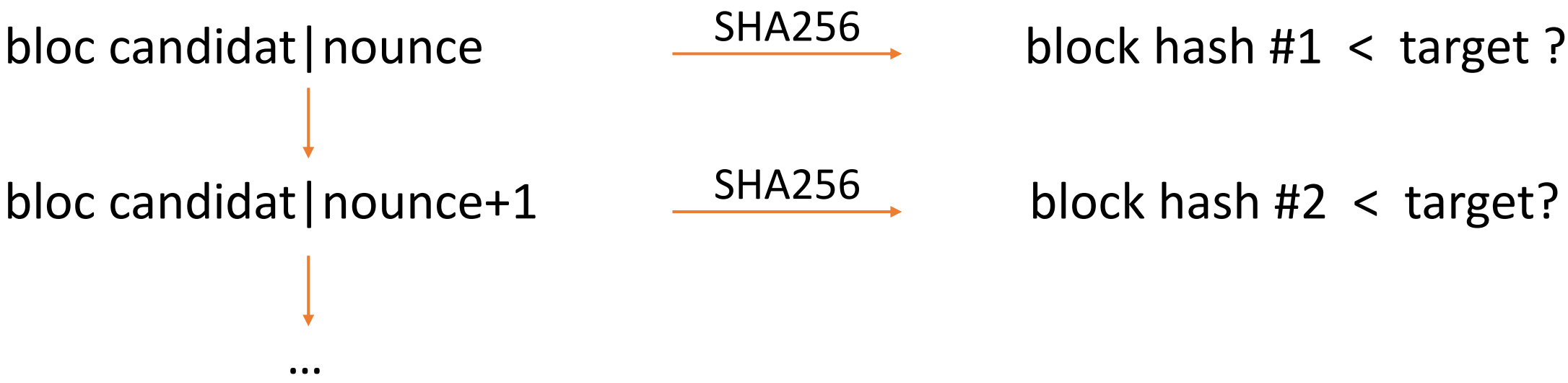
Changement du contenu d’un bloc

Changement de son hash

Changement du contenu du bloc suivant

…

Le protocole de minage du Bitcoin 1/4



Difficulté ajustée périodiquement (tous les 2016 blocs) afin de s'adapter à la puissance de calcul du réseau et conserver un temps moyen inter-bloc égal à 10 minutes.

Le protocole de minage du Bitcoin 2/4

Exemple de **blockchain imaginaire**:

Nous sommes le seul mineur d'une nouvelle blockchain. Le protocole cible un temps moyen entre les blocs égal à 2 secondes. Afin de miner, il faut construire un bloc candidat puis tirer à pile ou face avec une pièce de monnaie jusqu'à obtenir une pile. Nous tirons à pile ou face à la vitesse d'un lancer par seconde. Nous minons donc un bloc toutes les deux secondes en moyenne (en négligeant le temps de construction du bloc candidat), ce qui satisfait le réseau.

Attiré par cette blockchain, un autre mineur, qui tire à pile ou face à la même vitesse que nous, nous rejoint et commence à miner sur cette blockchain. Le temps moyen entre les blocs sera maintenant égal à 1 seconde, ce qui n'est plus satisfaisant pour le réseau. Après un temps suffisant pour que la moyenne du temps entre les blocs soit représentative, la difficulté du minage sur notre blockchain s'adaptera alors afin de ramener le temps moyen entre bloc à 2 secondes, par exemple en exigeant qu'un mineur tire deux piles afin de pouvoir inclure un bloc dans la blockchain.

Le protocole de minage du Bitcoin 3/4

Sur le Bitcoin, la difficulté est ajustée tous les 2016 blocs, en prenant en compte le temps qu'il a fallu pour obtenir ces blocs (normalement 2 semaines au rythme d'un bloc toute les 10 minutes).

Si les 2016 derniers blocs ont mis **plus** de 2 semaines à être minés, la difficulté **baisse.**

Si les 2016 derniers blocs ont mis **moins** de 2 semaines à être minés, la difficulté **augmente.**

Le protocole de minage du Bitcoin 4/4

Une approximation du temps moyen nécessaire pour trouver un nouveau bloc est donnée par l'équation suivante :

$$\text{time} = \frac{\text{difficulty } 2^{32}}{\text{hashrate}}$$

Lorsqu'un bloc est reçu par les membres du réseau, **ceux-ci vérifient qu'il est correct** :

1. Ils vérifient que le « bloc précédent » référencé par ce bloc existe
2. Ils vérifient que l'horodatage de ce bloc est supérieur à celui du bloc précédent et inférieur à deux heures dans le futur
3. Ils vérifient que la preuve de travail fournie est correcte, i.e que le hash fourni correspond bien à l'application de SHA256 aux informations du bloc concaténées avec le nounce fourni
4. Ils vérifient par ordre d'inclusion dans le bloc que les transactions présentes dans ce bloc sont valides

Les UTXO 1/4

La motivation des mineurs à fournir ce travail afin de sécuriser la blockchain Bitcoin provient du fait que chaque mineur a le droit d'inclure dans son bloc candidat une transaction lui versant un montant de Bitcoin prédéterminé ex nihilo. Ces transactions sont appelées les « **coinbase transactions** » et sont le seul moyen d'émission de nouveaux bitcoins.

Une **transaction standard** est composée d'un ou plusieurs inputs (bitcoins entrants dans la transaction) et d'un ou plusieurs outputs (bitcoins sortants de la transaction).

Les UTXO 2/4

Les inputs de transactions sont composés d'outputs de transactions précédentes.

Les outputs des transactions n'ayant pas été mobilisés par une nouvelle transaction portent le nom d'UTXO, pour Unspent Transaction Output.

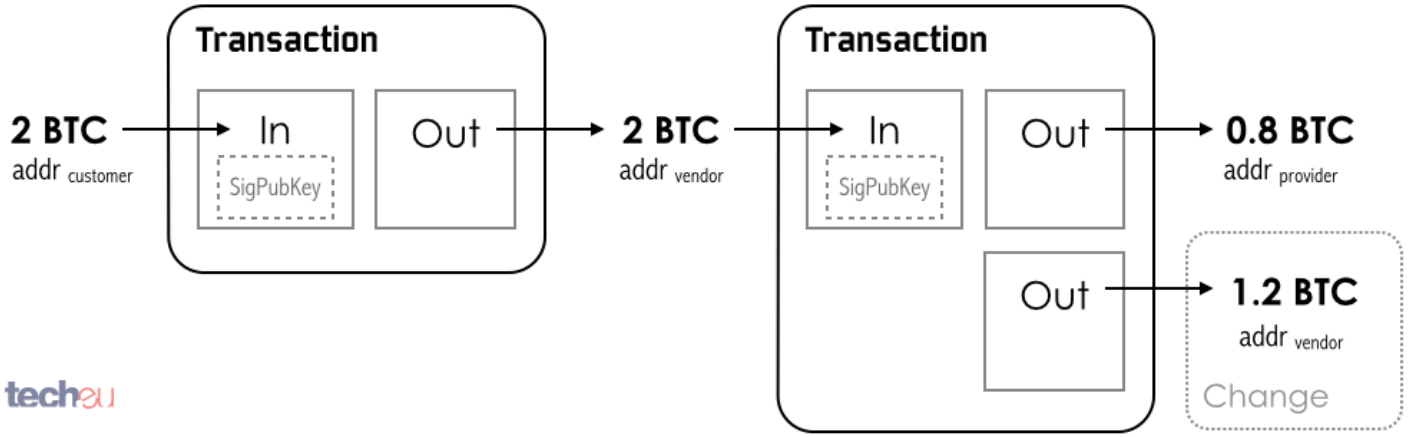
Ces UTXO sont composés d'un montant en bitcoin ainsi que d'un script n'autorisant l'utilisation de ces bitcoins qu'à l'utilisateur pouvant fournir une signature valide correspondant à l'adresse du récipiendaire de cet output.

Les UTXO 3/4

Sur la blockchain Bitcoin, il n'existe donc pas de compte contenant les bitcoins possédés par un utilisateur. Le portefeuille d'un utilisateur est constitué par la somme des UTXO possédés par cet utilisateur.

Fondamentalement, un bitcoin est donc un couple formé d'un identifiant de la transaction dans laquelle l'UTXO contenant ce bitcoin est présente et d'un index correspondant à la position de cette UTXO parmi les outputs de la transaction.

Les mineurs sont les seuls utilisateurs du réseau autorisés à créer une transaction vers leur adresse et n'ayant pas d'input.

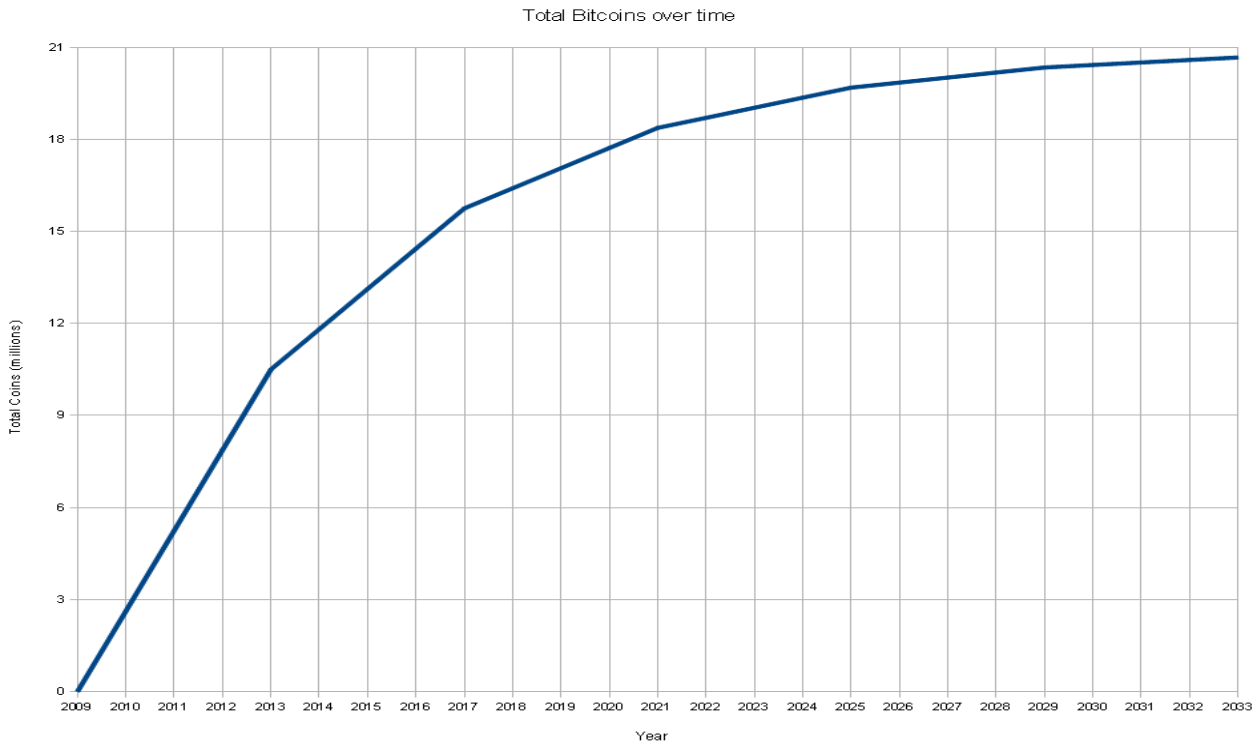


Les UTXO 4/4

Le nombre de bitcoins que peuvent s'attribuer les mineurs minant un bloc était de 50 bitcoins à sa création et est divisé par deux tous les 210000 blocs (environ 4 ans) jusqu'à atteindre zéro.

Lorsque la récompense sera quasi-nulle, le nombre total de bitcoins devrait s'élever à 21 millions environ. Ce système d'émission en fait une monnaie dit **déflationnaire**.

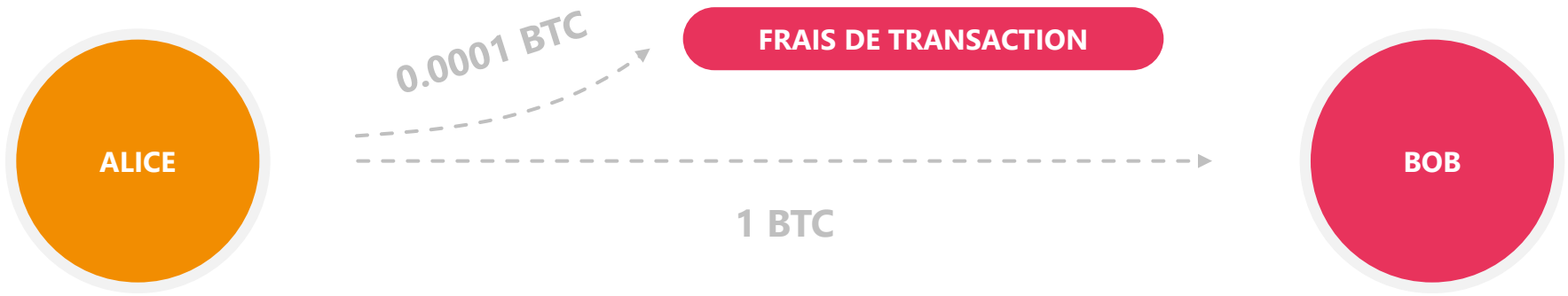
La récompense actuelle est de 3.125 bitcoins par bloc miné.



4

Généralités

*Une **transaction de BTC***



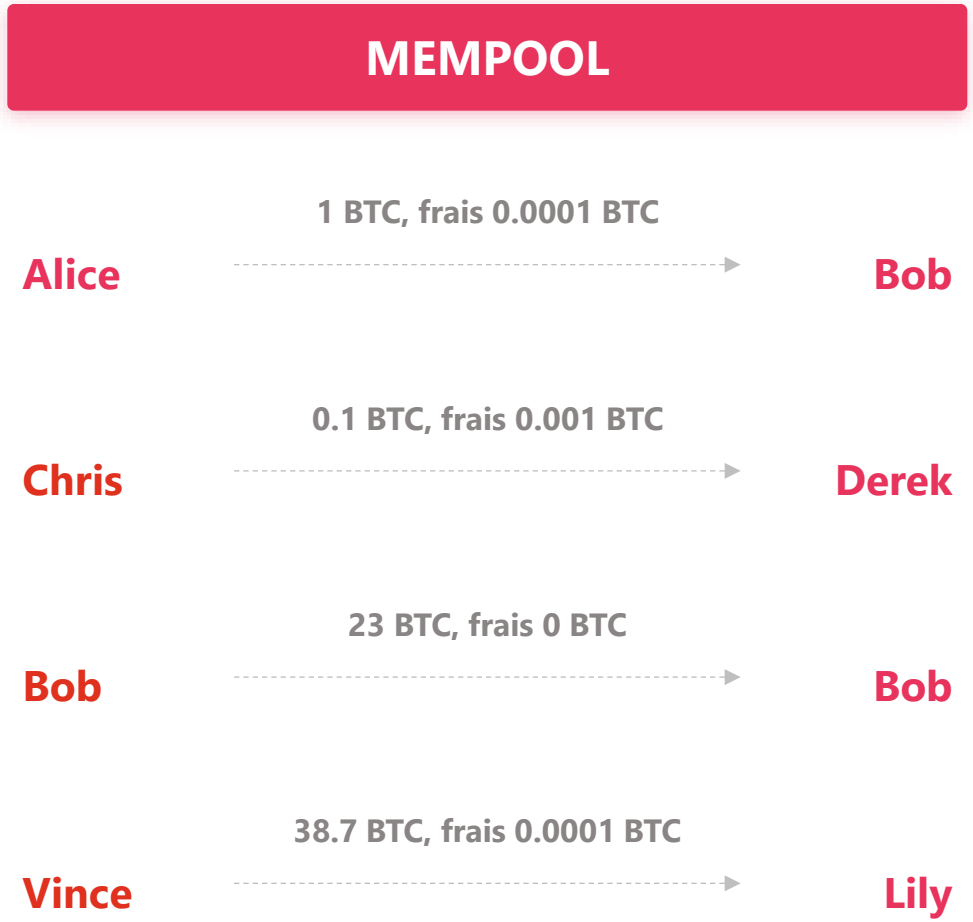
Chacun peut se créer une adresse Bitcoin avec clé privée/clé publique.

Une adresse créée permet de recevoir et d'envoyer des BTC.
Il n'y a pas de découvert : on ne peut envoyer plus de BTC qu'il n'y en a en stock à l'adresse utilisée.

Le mempool

Une fois qu'Alice a tapé son mot de passe, l'adresse de Bob, le montant qu'elle veut envoyer à Bob et les frais de transactions qu'elle choisit de payer, la transaction est enregistrée dans le mempool.

C'est l'ensemble des transactions en attente de confirmations.



Les confirmations ***des transactions***

Chaque mineur qui parvient à ajouter un nouveau bloc à la blockchain choisit les transactions enregistrées dans le mempool pour les inclure dans son bloc.

Les confirmations ***des transactions***

Le mineur reçoit sur l'adresse Bitcoin de son choix :

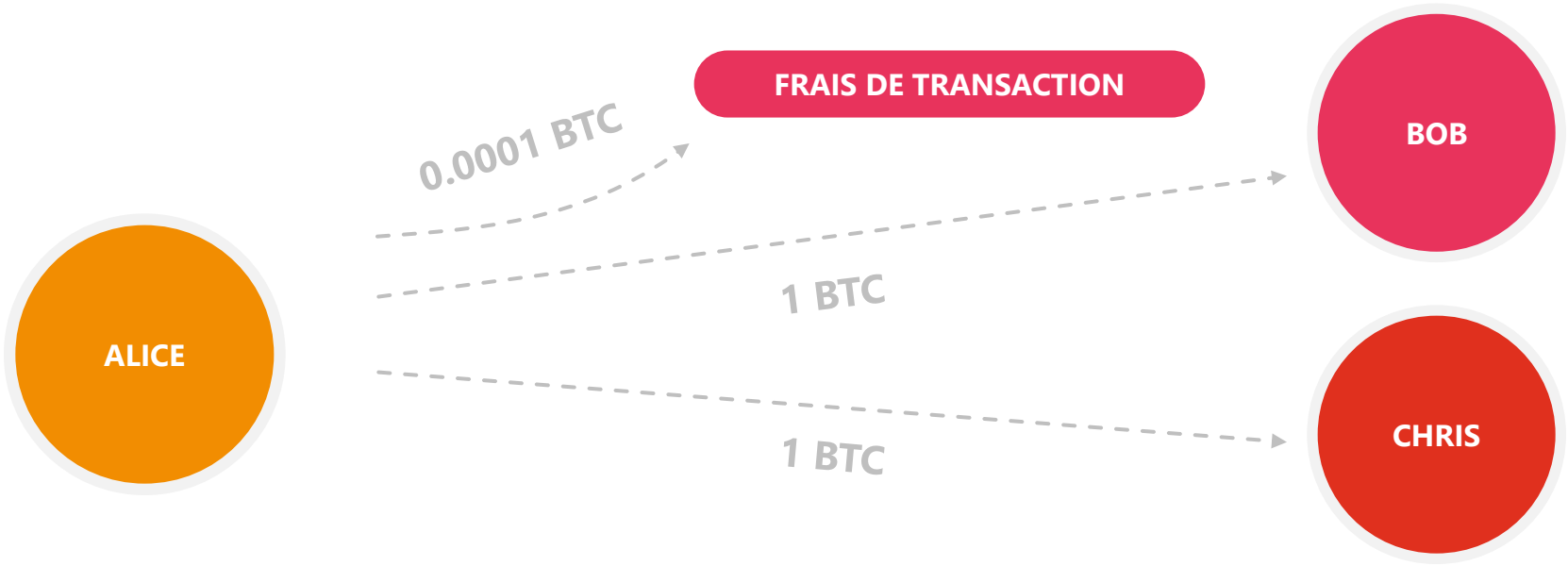
- ***La récompense** correspondante à son bloc*
- ***Les frais** des transactions du mempool qu'il ajoute dans son bloc*

Ces transactions choisies par le mineur ont alors 1 confirmation.

Les confirmations ***des transactions***

En pratique, choisir un frais de transaction élevé assure que la transaction sera ajoutée rapidement dans la blockchain.

Les confirmations ***des transactions***

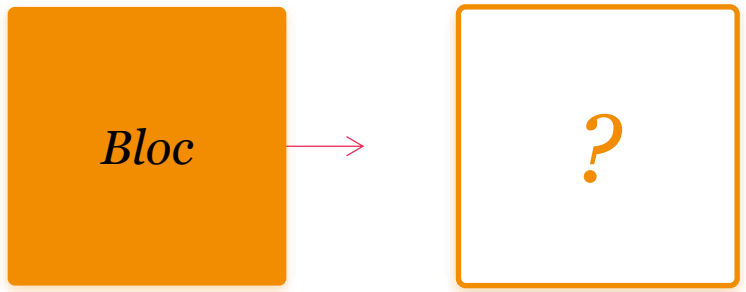


PAS DE DOUBLE SPEND:

Si Alice n'a que 1.0001 BTC sur son compte, elle peut faire 2 transactions non confirmées de 1 BTC avec 0.0001 BTC de frais, à Bob et à Chris. Mais au moins une des 2 transactions ne sera jamais confirmée.

Résolution des ***conflits***

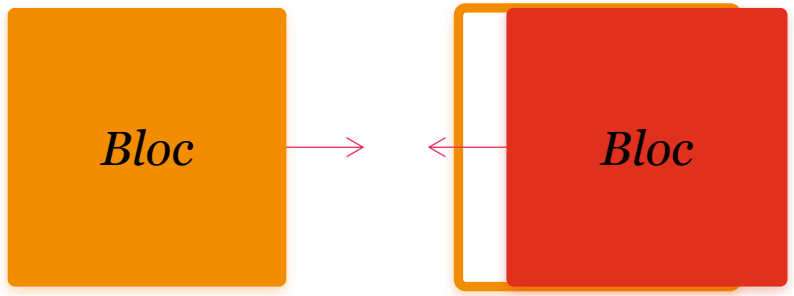
Les mineurs tendent à vouloir s'assurer la valeur de leur récompense



Ce qui le décidera, c'est si les autres mineurs vont chercher à **ajouter un bloc à « leur » nouveau bloc**… ***ou pas.***

Résolution des ***conflits***

Les mineurs tendent à vouloir s'assurer la valeur de leur récompense



Il est donc **très risqué** pour un mineur de proposer un bloc en conflit avec les règles admises par les autres mineurs.

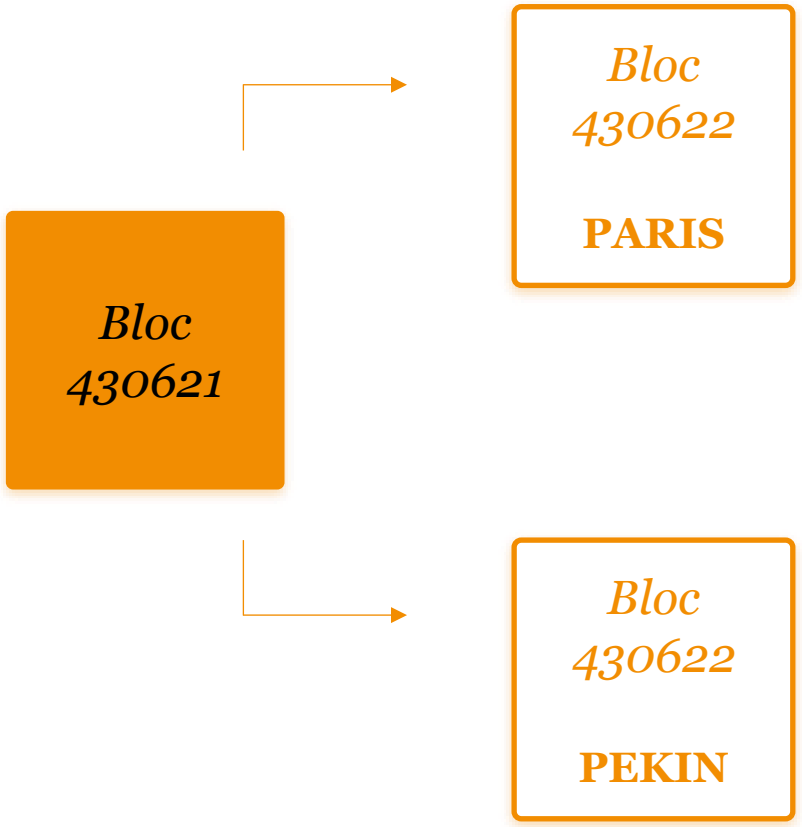
Résolutions des ***conflits***

Les mineurs tendent à vouloir s'assurer la valeur de leur récompense

En pratique, **3 règles** (parmi beaucoup d'autres) sont respectées depuis 2009:

1. Pas de double spend
2. Pas de découvert
3. Les récompenses des mineurs ne sont pas supérieures à celles convenues

Résolution des ***conflits***



« Stale block »

En pratique, il arrive aussi parfois que 2 mineurs obtiennent un bloc en même temps. Les autres mineurs choisissent chacun « à quel bloc » ils cherchent à en ajouter un.

Le perdant a miné un « stale block ».

*Adresses **M of N***

Une adresse M of N est constituée de N clés, et l'accès des fonds correspondants est débloqué avec seulement M de ces N clés.

Scripts de Bitcoin :

- Multisig : Payer à un script qui nécessite m signatures parmi n clés publiques définies
- P2SH : Payer au hash d'un script qui sera révélé au moment de la dépense de l'utxo (souvent utilisé avec un redeem script de type Multisig)

EXEMPLE : UNE ADRESSE 3 SUR 5

*Une adresse 5 of 10 est **la solution** du problème des généraux byzantins évoqué précédemment.*

La Blockchain Bitcoin : un écosystème

Interactions entre plusieurs aspects :

- ***Technologiques*** : protocole de validation, blockchain
- ***Économiques*** : unités de comptes avec courbe d'émission, algorithmes de consensus
- ***Humains*** : interactions entre les différents acteurs
    1. Mineurs
    2. Noeuds
    3. Développeurs
    4. Utilisateurs
- ***Politiques*** : enjeux de **consensus** entre ces différents acteurs, avec une gouvernance à déterminer

Les altcoins : Litecoin

- Le genesis block de Litecoin a été produit le 13 octobre 2011.

C’est un fork du code source de Bitcoin Core, avec comme modifications principales :

- ➢Un intervalle de temps entre deux blocs de 2,5 minutes au lieu de 10 minutes
- ➢Un passage de la quantité totale de coins de 21 millions à 84 millions
- ➢L’utilisation de l’algorithme de hashage Scrypt au lieu de l’algorithme SHA256, ceci dans le but de favoriser la décentralisation du mining. Cet algorithme était en effet supposé rendre plus difficile la production de ASICS (mais finalement des ASICS spécifiques à cet algorithme ont également vu le jour)
- Litecoin a atteint une capitalisation de $10 milliards.

Les altcoins : Dogecoin

- Le genesis block de Dogecoin a été produit le 06 décembre 2013.

Son logo et son nom font référence à un meme Internet, l'objectif était d'être une « joke currency » qui reposait essentiellement sur le fun.

- Les créateurs et la communauté espéraient atteindre par ce mécanisme humoristique une base d'utilisateurs plus importante que les autres crypto-monnaies de l'époque.
- A la différence de nombreuse autres crypto-monnaies déflationnistes, il n'y a pas de limite maximale à la création de nouveaux dogecoins. Ainsi, c'est un exemple de crypto-monnaie inflationniste.
- C'est un fork du code source de Litecoin. Son intervalle de temps entre deux blocs est de 1 minute.
- Elle a connu un succès important à son lancement, lors d'une journée de janvier 2014 le volume trading sur 24h a depassé celui de Bitcoin.
- Elle a atteint plus de $10 milliards en terme de capitalisation.

Les altcoins : Monero

- Le genesis block de Monero (abrégé XMR) a été produit le 18 avril 2014.
- Monero est une crypto-monnaie fongible basée sur le protocole CryptoNote, et **axée sur la confidentialité des transactions et leur non-traçabilité**.
- Elle utilise l'algorithme de consensus Proof-of-Work.
- La taille de la blockchain Monero est supérieure à 60GB et grossit plus rapidement que les autres.
- Monero utilise un registre partagé mais les adresses d'envoi et de réception sont masquées, tout comme les montants échangés.
- En moyenne, l'intervalle de temps entre les blocs est de 2 minutes.
- La capitalisation globale du projet a dépassé le milliard de dollars.

Segregated Witness (SegWit) 1/2

- SegWit est un **changement du format des transactions** qui a été introduit dans Bitcoin suite à un Soft Fork le 24 août 2017. A ce jour, c'est l'une des modifications du protocole Bitcoin les plus importantes qui a eu lieu.
- Cette modification a été **controversée** : elle a été intégrée à la version 0.13.1 de Bitcoin Core à partir d'octore 2016 mais a été activée par le réseau Bitcoin seulement en août 2017.
- C'est un changement qui permet notamment de résoudre le bug de la malléabilité des transactions (*ie.* la possibilité d'altération des signatures sans invalidation des transactions, ce qui entraîne une modification de l'identifiant de la transaction) qui empêchait la mise en place d'autres technologie d'améliorations du protocole et de scaling telles que Lightning Network ou MAST.

Segregated Witness (SegWit) 2/2

- Une transaction SegWit sépare les éléments de base de la transaction (également appelés *Transaction data*) et les signatures des transactions (également appelées *Witness data*) en deux structures de données différentes.
- La limite de “taille des blocs” de 1 est remplacée par une limite de “hauteur de block” de 4.
- Dans le calcul de cette nouvelle métrique de“hauteur de block” la *Witness data* bénéficie d’un discount de 75% par rapport à la *Transaction data.*
- La *Witness data* n’a plus besoin d’être réutilisée une fois que les blocks sont validés, alors que la *Transaction data* doit être conservée en mémoire car elle fait partie de l’UTXO set. **La *Witness data* est donc moins consommatrice en ressources pour le réseau**, c’est la justification économique de ce discount.

Lightning Network 1/2

- C’est un protocole qui permet d’avoir **un réseau de canaux de paiement bi-directionnels** afin d’étendre les fonctionalités du réseau Bitcoin. L’ouverture de canaux de paiement utilise notamment le langage de script de Bitcoin.
- Deux parties doivent ouvrir un canal entre elles en effectuant une transaction standard sur la blockchain Bitcoin.
- Une fois un canal de paiement établi entre ces deux parties, elles peuvent effectuer des transactions entre elles sans les diffuser au réseau P2P. Ces transactions restent donc privées et n’apparaissent pas dans la blockchain.
- Ainsi, à l’intérieur d’un tel canal, des transactions peuvent être effetuées quasi-instantanément, à très faibles coûts et avec une capacité de l’ordre du million de transactions par seconde.
- Via un système de routage, le Lightning Network permet d’établir un réseau entre les canaux bidirectionnels existants.

Lightning Network 2/2

- Ainsi **un utilisateur ne sera pas obligé de payer sa contrepartie via un canal de paiement directement existant avec elle**. Le réseau Lightning trouvera pour lui le chemin existant le plus efficient parmi tous les canaux de paiement existants.
- Si l'une des parties d'un canal pense que sa contrepartie a été malhonnête, elle peut décider à tout moment de broadcaster la transaction litigieuse dans la blockchain et ainsi récupérer le contrôle de ses fonds. Un canal est donc "trustless" mais nécessite néanmoins une surveillance constante de la part des participants.
- Cette idée, dont le but est de permettre le scaling de Bitcoin, a émergé en 2015. Les spécifications du protocole ont été publiées en décembre 2017, la première transaction sur le réseau Bitcoin a eu lieu le même mois.

Confidential Transactions

- Confidential Transactions (CT) est une proposition de modification de Bitcoin pour **améliorer la confidentialité des transactions**. Elle date de 2015.
- CT permet de faire en sorte que le montant transféré soit visible uniquement par les participants de la transaction. Toutefois, les addresses de l'émetteur et du destinaire de la transaction restent visibles de tous.
- Une Zero Knowledge Proof (ZKP) est contenue dans chaque CT afin de permettre à des observateurs de verifier la validité des transactions sans connaitre leur montant.
- Cependant, les transactions deviennent ainsi **16 fois plus grosses que les transactions normale**s, d'où l'impossibilité d'implémenter ce schéma cryptographique dans Bitcoin qui a des problèmes de passage à l'échelle.

Annexe

Compléments sur la cryptographie de Bitcoin

Cubiques rationnelles 1/3

- **Une courbe cubique** définie sur un corps K est une équation $f(x, y) = 0$ où $f(x, y)$ est un polynôme de degré 3 des variables x et y à coefficients dans K.
- Courbes planes C_f : locus de l'équation $f(x, y) = 0$ pour tout polynôme non constant à coefficients complexes $f(x, y) \in \mathbb{C}[x, y]$.
- Courbe plane rationnelle : courbe C_f telle que $f(x, y)$ est un polynôme à coefficients rationnels.
- Objectif : décrire les points rationnels d'une cubique en partant d'un certain point rationnel O, se déplacer de point rationnel en point rationnel en donnant une structure algébrique à la cubique.

Rappels sur la géométrie projective

- **Définition** : le plan projectif $\mathbb{P}_2$ est l'ensemble de tous les triplets $w : x : y$, où w, x et y ne sont pas tous nuls et les points $w : x : y$ et $w' : x' : y'$ sont considérés égaux s'il existe une constante k non nulle telle que

$$w' = kw \qquad x' = kx \qquad y' = ky$$

- Comme dans le cas des plans affines et courbes planes, nous avons trois cas basiques :

$$\mathbb{P}_2(\mathbb{Q}) \subset \mathbb{P}_2(\mathbb{R}) \subset \mathbb{P}_2(\mathbb{C})$$

$w : x : y \in \mathbb{P}_2(\mathbb{C})$ est également dans $\mathbb{P}_2(\mathbb{Q})$ si et seulement s'il est possible de « rescale » $w, x, y \in \mathbb{C}$ en un élément de $\mathbb{Q}$.

- **Remarque** : une ligne C_f de $\mathbb{P}_2$ est le locus de tous les $w : x : y$ satisfaisant l'équation

$$F(w, x, y) = aw + bx + cy = 0$$

La ligne à l'infini L_∞ est donnée par l'équation $w = 0$. Un point de $\mathbb{P}_2 - L_\infty$ a la forme $1 : x : y$.

Le point $1 : x : y$ du plan projectif correspond au point (x, y) du plan Cartésien. Etant données deux lignes L donnée par l'équation $aw + bx + cy = 0$ et L' donnée par l'équation $a'w + b'x + c'y = 0$, on aura $L = L'$ si et seulement si $a : b : c = a' : b' : c'$ dans le plan projectif.

Cubiques rationnelles 2/3

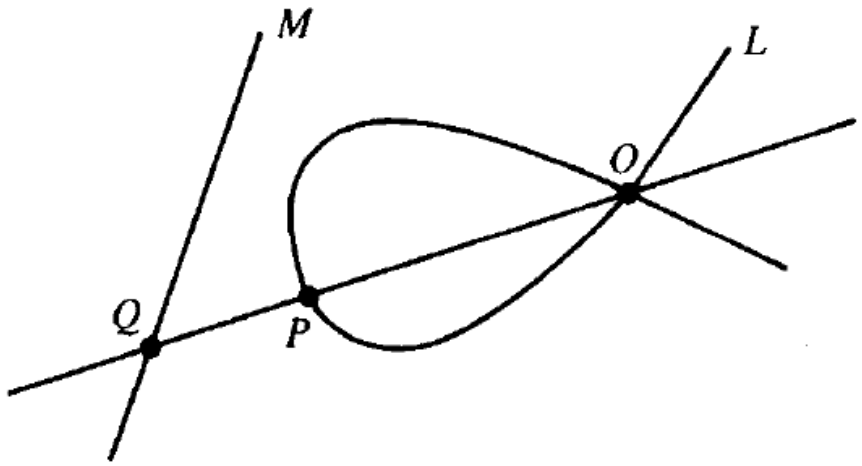
- **Proposition 1** : Si deux des trois points d'intersection d'une cubique avec une ligne rationnelle sont des points rationnels, alors le troisième point d'intersection est aussi un point rationnel.

- Cubique irréductible : une cubique irréductible est une cubique ne pouvant pas être factorisée dans le corps des nombres complexes

- Un point O d'une cubique C est appelé un point singulier si chaque droite passant par O intercepte C en au plus un point.

Cubiques rationnelles 3/3

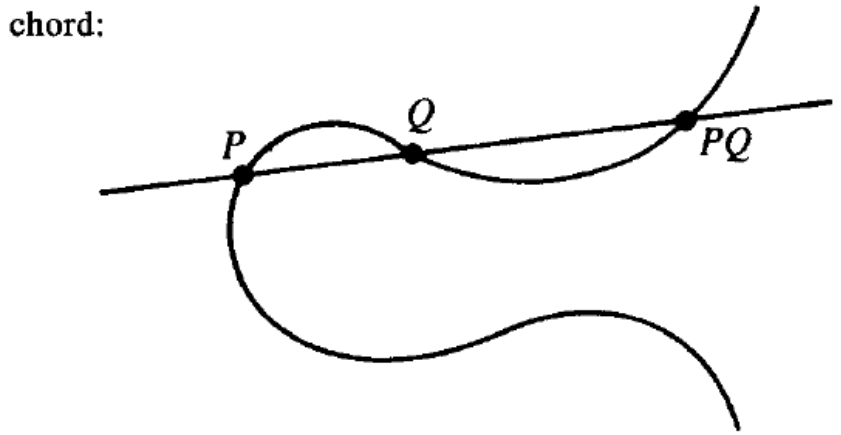
- Exemple de cubique singulière : $y^2 = x^2(x + a)$ et le point $O = (0,0)$



- O point singulier, chaque droite L passant par O coupe la cubique en un second point P rationnel

Transformation corde-tangente 1/4

- Etant donnés deux points P et Q rationnels sur une cubique, on peut d'après la propriété 1 en construire un troisième en traçant la droite reliant ces deux points

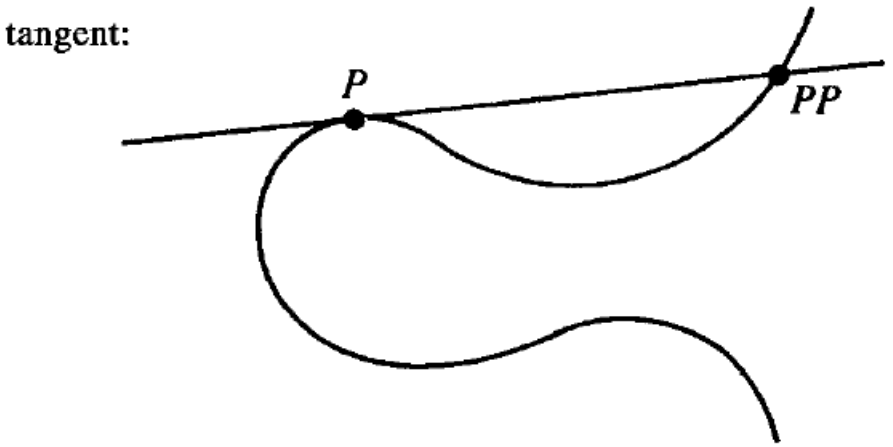


- Le troisième point d'intersection de cette droite avec la cubique, noté PQ, sera rationnel. C'est l'opération dite de « corde ».

Transformation corde-tangente 2/4

Avec un seul point rationnel, on peut tracer la tangente en ce point (droite reliant ce point à lui-même).

Par la propriété 1, le « troisième » point d'intersection de la droite avec la cubique est un point rationnel.



Transformation corde-tangente 3/4

La transformation associant à deux points P et Q le point PQ, ou à un point P le point PP, est appelée la loi de composition corde-tangente.

Première forme du théorème de Mordell : Sur une cubique rationnelle non singulière, il existe un ensemble fini de points rationnels tels que tous les points rationnels de la courbe sont générés à partir de cet ensemble par la loi de composition corde-tangente.

Preuve : cf. David Husemöller. Elliptic curves. Springer-Verlag New York, 2004. Voir chapitre 6.

Transformation corde-tangente 4/4

Il existe donc un ensemble fini X de points rationnels sur une cubique rationnelle non singulière tel que tout point rationnel P de cette cubique peut être décomposé sous la forme :

$$P = (\dots((P_1P_2)P_3)\dots P_r)$$

Où $P_1, \dots, P_r$ sont des éléments non nécessairement différents de l'ensemble fini X.

La loi de composition corde-tangente est commutative mais **ne comprend pas d'élément neutre** et n'est donc pas une loi de groupe.

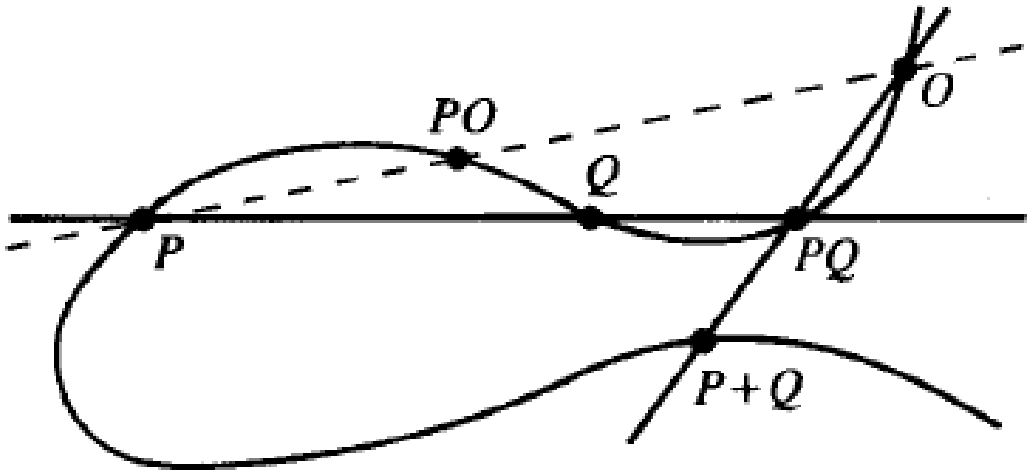
On va cependant construire une loi de groupe à partir de cette loi de composition.

La loi de groupe

Pour compléter la loi de composition corde-tangente, on peut choisir **arbitrairement** un point rationnel O de notre courbe comme « zéro » et définir une nouvelle loi de transformation $P + Q$ de deux points rationnels P et Q :

$$P + Q = O(PQ)$$

Qui est commutative, par définition de la loi corde-tangente.



La loi de groupe – élément neutre

On peut montrer qu'avec cette définition $\boldsymbol{O}$ **est réellement un élément neutre** par la loi de composition $+$.

En effet, soit P un point rationnel quelconque de la cubique, on a :

$$P + O = O(PO) = P$$

Car les points O, P et PO sont alignés.

La loi de groupe – inverse

On peut montrer que pour tout point rationnel P de la cubique, il existe un point rationnel $-P$ tel que $P + (-P) = O$

On doit donc avoir :

$$\begin{aligned} & \mathrm{O}\big(\mathrm{P}(-\mathrm{P})\big) = \mathrm{O} \\ \Leftrightarrow\ & O\big(P(-P)\big) = O + O \\ \Leftrightarrow\ & O\big(P(-P)\big) = O(OO), \forall P \text{ point rationnel} \\ \Leftrightarrow\ & P(-P) = OO \end{aligned}$$

D'après la définition de la transformation corde-tangente, le point $-P$ est donc le troisième point d'intersection de la droite passant par P et OO avec la cubique (existe toujours dans le cas d'une cubique non singulière).

La loi de groupe – associativité

L'associativité est plus compliquée à obtenir et à démontrer, on se contentera d'énoncer le théorème suivant sans le démontrer :

Théorème 1 : Soit C une cubique non singulière définie sur un corps k et O un point de $C(k)$. Alors la loi de composition définie sur $C(k)$ par $P + Q = O(PQ)$ donne une structure de groupe à $C(k)$ avec l'élément neutre O et $-P = P(OO)$. De plus, O est un point d'inflexion ssi $P + Q + R = O$ pour tout triplet (P, Q, R) où P, Q et R sont les points d'intersection d'une droite avec C. Dans ce cas, on a également $-P = PO$ et $OO = O$.

Courbe elliptique

Définition : Une courbe elliptique E définie sur un corps k est une courbe cubique non singulière E sur k munie d'un point $O \in E(k)$. La loi de groupe sur $E(k)$ est définie par le point O et la loi de composition corde-tangente PQ par la relation $P + Q = O(PQ)$ pour deux points $P, Q \in E(k)$.

Le théorème de Mordell se reformule alors en terme plus naturel de courbes elliptiques :

Théorème : Soit E une courbe elliptique rationnelle. Le groupe des points rationnels $E(\mathbb{Q})$ est un groupe abélien finiment engendré (engendré par une partie finie).

Rappels sur la théorie des corps

Théorème : Tout corps fini est commutatif.

Théorème : Soit F un corps fini. Il existe un plus petit entier p tel que la somme constituée de p termes égaux à 1 soit nulle. De plus p est un nombre premier et $\mathbb{F}_p$ est un sous corps de F. Ce nombre p est appelé la caractéristique du corps F.

Théorème : Soit F un corps fini de caractéristique p. Le nombre d'éléments de F est de la forme

$$\#F = p^n$$

Définition: Pour tout élément a de F^* (ensemble des éléments non nuls de F) il existe un plus petit entier non nul e tel que $a^e = 1$. De plus, e divise $p^n - 1$. Ce nombre e est appelé l'ordre de a.

Remarque: Si e est l'ordre de a, alors pour tout entier $1 \leq i \leq e$ l'ordre de a^i est égal à

$$\frac{ppcm(i, e)}{i}$$

Théorème : Le groupe multiplicatif F^* est cyclique.

Corollaire : Tout corps fini a un élément primitif G, c'est-à-dire un générateur de ce corps.

Sécurité

Remarque : Les algorithmes connus pour résoudre le problème du logarithme discret sur les courbes elliptiques sont en $o(\sqrt{n})$, la taille du corps sur lequel on définit la courbe elliptique doit donc être approximativement deux fois plus grande que le paramètre de sécurité souhaité. Pour un degré de sécurité de 128-bits, on choisira donc une courbe sur un corps $\mathbb{F}_q$ où $q \approx 2^{256}$. L'utilisation de ces algorithmes permet l'utilisation d'un système clé privée-clé publique sûr et la sécurisation des fonds stockés sur une clé privée, qui ne peuvent être dépensés que si une signature valide est fournie.

Références

- Courbes elliptiques :
  - David Husemöller. *Elliptic curves*. Springer-Verlag New York, 2004. isbn : 978-0-387-95490-5. doi : 10.1007/b97292.
    url : https://dx.doi.org/10.1007/b97292
  - Joppe W. Bos et al. *"Elliptic Curve Cryptography in Practice"*. *Financial Cryptography and Data Security : 18th International Conference, FC 2014, Christ Church, Barbados, March 3-7, 2014, Revised Selected Papers*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2014, p. 157-175. isbn : 978-3-662-45472-5. doi : 10.1007/978-3-662-45472-5_11. url : https://doi.org/10.1007/978-3-662-45472-5_11
  - Certicom. *"SEC 2 : Recommended Elliptic Curve Domain Parameters"* (2010).
  - Ralph C. Merkle. *"A Certied Digital Signature"*. *Proceedings on Advances in Cryptology*. CRYPTO '89. Santa Barbara, California, USA : Springer-Verlag New York, Inc., 1989, p. 218-238. isbn : 0-387-97317-6.
    url : http://dl.acm.org/citation.cfm?id=118209.118230
- Fonctions de hachage :
  - Ivan Bjerre Damgard. *"A Design Principle for Hash Functions"*. *Proceedings on Advances in Cryptology*. CRYPTO '89. Santa Barbara, California, USA : Springer-Verlag New York, Inc., 1989, p. 416-427. isbn : 0-387-97317-6.
    url : http://dl.acm.org/citation.cfm?id=118209.118248