Tangle

Por qué Tangle?

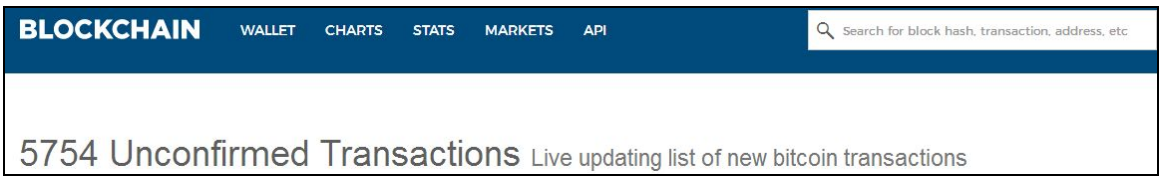
La tangle surge como una solución a los principales problemas que trae aparejada una blockchain. Los creadores de tangle estuvieron desarrollando herramientas relacionadas al mundo de Bitcoin desde 2010. Esas experiencias les permitieron identificar las debilidades y fortalezas de Bitcoin y crear otra tecnología que comparta las fortalezas y enmiende las debilidades.

Los creadores de tangle lo hicieron teniendo en mente su criptomoneda iota, que está pensada para prestación de servicios en el mundo de IOT. Nosotros intentamos darle un uso más general, llevándolo a las transacciones entre personas usando celulares.

Las fortaleza de Bitcoin se basa en su único ledger distribuido, que permite a todos los usuarios verificar transacciones y operar fuera de las regulaciones o limitaciones de ningún organismo. No ahondaremos mucho en estos puntos porque suponemos que todos los equipos y los profesores encargados de calificar este trabajo conocen de sobra las características de la blockchain. A continuación discutiremos sus debilidades:

Escalabilidad

La falta de escalabilidad es un gran problema de Bitcoin. El hecho de tener un único ledger donde todas las transacciones deben ser anexadas secuencialmente provoca un comportamiento cíclico: primero se mina un bloque, luego se agrega a la blockchain, luego se mina el siguiente bloque, luego se agrega a la blockchain. Nuestra transacción tendrá que esperar a que un minero la asocie a un bloque, lo “mine” para poder agregarlo a la blockchain y finalmente lo adjunte. No sabemos cuántos de estos ciclos tendremos que esperar hasta que nuestra transacción sea parte de la blockchain. Al momento de escribir este documento, hay 5754 transacciones pendientes.



Valores de blockchain.info

Esto se traduce como poco en una molestia para los usuarios y como mucho en la imposibilidad de ejecutar la mayoría de los casos de uso que tiene una moneda tradicional.

Las transacciones de nuestra vida diaria que necesitan confirmación rápida no pueden estar sujetas a la disponibilidad de la red en el momento en que necesitamos usarla.

También debe tenerse en cuenta la existencia de mining pools. Estos son grupos de mineros que juntan su poder computacional para minar y luego se dividen las ganancias. Debido al gran poder computacional que requiere minar un bloque, es muy posible que nuestra transacción termine siendo añadida a la blockchain por uno de estos grupos. Qué pasaría si por alguna razón los pools más grandes no quisieran trabajar con transacciones provenientes de nuestra dirección? Nuestras transacciones tardarían muchísimo más en procesarse. Si bien Bitcoin es descentralizado, no podemos ignorar la gran influencia que tienen los mining pools más grandes en el funcionamiento del sistema.

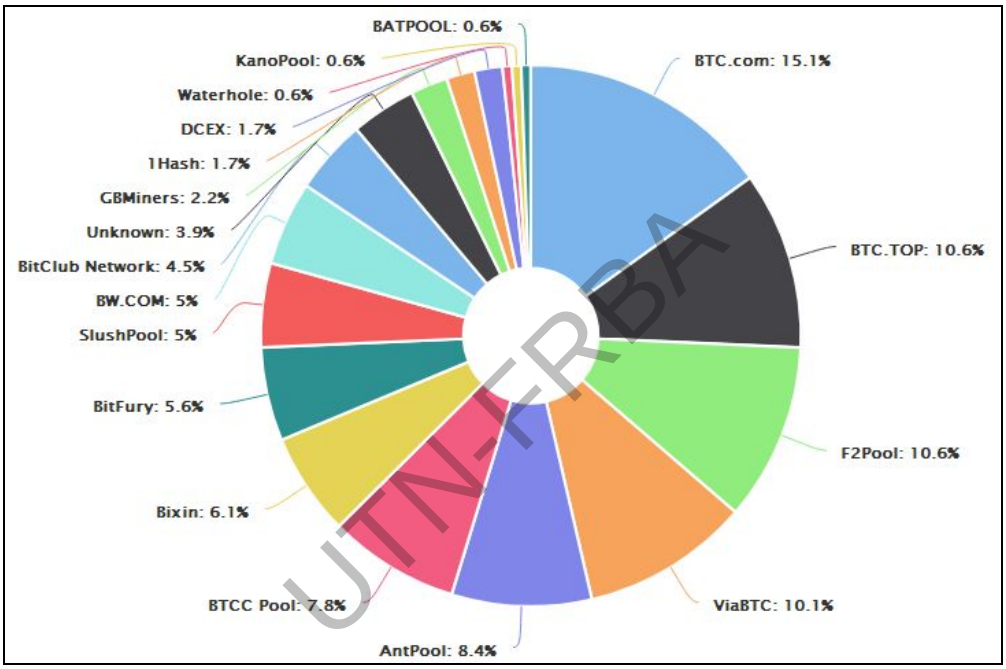


Gráfico de blockchain.info/pools. Este gráfico representa el procentaje de poder de procesamiento que tiene cada pool. Este gráfico no quiere decir que los únicos que minen bloques sean pools, sino éstos están bastante concentrados.

Comisiones

El problema que trae aparejada la falta de escalabilidad de una blockchain son las comisiones que los mineros cobran por agregar las transacciones de los usuarios a un bloque. El promedio de las comisiones de transacción son ahora de 0.0004746 BTC por transacción en promedio. Al momento de escribir esto, ese valor equivale a 34.79 pesos.

> Which fee should I use?
>
> The fastest and cheapest transaction fee is currently **0.0000021 BTC/byte**, shown in green at the top. For the median transaction size of **226 bytes**, this results in a fee of **0.0004746 BTC**.

0.0004746 Bitcoin equals
34.79 Argentine Peso

Valores de bitcoinfees.21.co

El estado de la red u otros factores asociados con la misma pueden variar considerablemente.

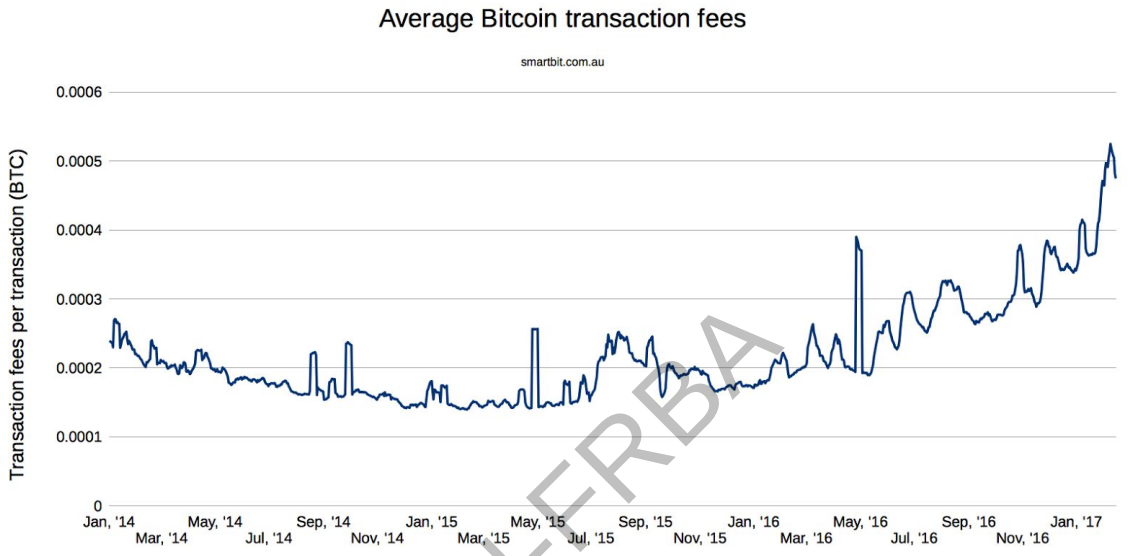


Gráfico de smartbits.com.au

La pregunta de "quién pagaría tanta comisión?" cada vez resuena más en la comunidad de Bitcoin. Las comisiones tan altas no sólo son algo que se presenta como una desventaja teórica, sino que directamente hace que el uso de una blockchain sea descartado para muchos casos de uso de la vida diaria.

La incertidumbre sobre cuánto un comerciante va a recibir sobre una transacción monetaria implica la incertidumbre sobre si su modelo de negocio funciona(especialmente en los casos donde el precio del producto es del mismo orden de magnitud que la comisión que se paga por la transacción). Por qué sumar $35 de comisión al precio de un producto de $5? Si el comerciante absorbe el costo de la comisión, cuánto dinero ganará por vender un producto si la comisión es impredecibles?

Aunque se están viendo muchos avances en esta área, la conclusión es que una tecnología como blockchain no puede cubrir todos los usos que tiene una moneda fiat en nuestro día a día.

Tangle

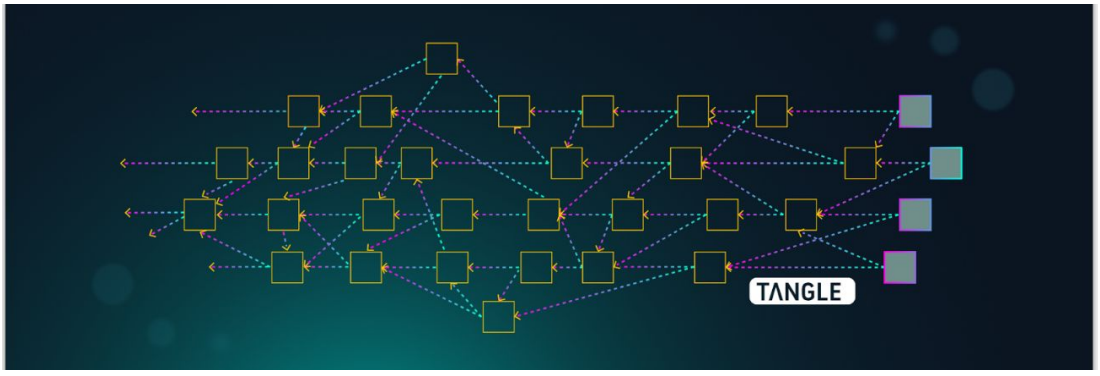
Tangle(una traducción podría ser enredo o maraña) es la solución a estos problemas que propuso un grupo de programadores europeos. Junto con la tecnología de tangle, estas personas crearon una criptomoneda llamada Iota que hace uso de la misma. Tangle es una nueva arquitectura de ledger distribuido basada en un un GAD(grafo acíclico dirigido). La tangle tiene los mismos principios que un blockchain: es una base de datos distribuida, es una red p2p y cuenta con un mecanismo de consenso(cuáles transacciones son válidas y cómo se relacionan entre sí) y validación.

Las principales diferencias entre una tangle y un blockchain son la estructura y la forma en que llegan al consenso. No hay "bloques" sino que cada transacción referencia dos transacciones anteriores. Al referenciar una transacción, damos fe que es válida y se adapta a las reglas del protocolo. No sólo validamos directamente las transacciones que estamos referenciando, sino que también validamos indirectamente todas las transacciones que éstas referenciaron.

En lugar de tener un grupo reducido de usuario responsables de obtener el consenso, toda la red de participantes activos están involucrados en la aprobación de las transacciones. El consenso sobre cuáles transacciones son válidas no es un proceso independiente de la realización de esas transacciones, sino que lo primero es una parte intrínseca de lo segundo. Esto es lo que le permite eliminar las comisiones en este tipo de transacciones.

Estructura

Como ya dijimos, tangle es un grafo acíclico dirigido. Los nodos del grafo representan transacciones y los arcos representan referencias. Cada transacción referencia dos transacciones anteriores, por lo que no puede ser cíclico(para ello alguna transacción debería referenciar otra posterior a sí misma).



En la comunidad de Bitcoin han habido acusaciones de spammers que realizan muchísimas transacciones inútiles por largos períodos de tiempo con el objetivo de agregar carga a la red con alguna agenda oculta. En tangle, dado que cada transacción

valida dos transacciones previas en la red, el spam ayuda a la red: hace que las transacciones se validen más rápido.

Titular de news.bitcoin.com

Saldo

Al igual que en Bitcoin, para calcular el saldo de una persona se cuentan todas las transacciones hacia su dirección que no hayan sido gastadas. Para gastar ese saldo, se hace una transacción hacia otra dirección y se indican cuáles ingresos se usarán para justificarla.

Por ejemplo, si alguien me envía 5 monedas en la transacción #65586, yo puedo enviar 3 monedas a un amigo y 2 a otro referenciando la transacción #65586. Una vez que haya gastado esa transacción, ya no podré hacer otra referenciando la misma transacción. Más adelante hablamos de el intento de ataque en el que alguien gasta dos veces referenciando la misma transacción.

Cómo realizar transacciones

Las transacciones que todavía no fueron referenciadas por nadie se llaman tips o puntas, debido a se encuentran en las puntas del grafo.

Para realizar una transacción deben seguirse los siguientes pasos:

1. **Firmar:** El usuario firma la transacción usando su clave privada

2. **Seleccionar tip:** Se usa una cadena de markov de Monte Carlo (MCMC: Markov chain Monte Carlo) para seleccionar dos tips al azar. Se deberá validar que ambos tips seleccionados sean válidos. Las dos tips que referenciará la transacción se conocen como branchTransaction y trunkTransaction. Esta validación es el corazón de la tangle: todo el que quiera hacer una transacción debe validar otras transacciones, proveyendo un servicio a toda la red. La razón para elegir dos referencias en lugar de cualquier otro número es que los creadores de tangle corrieron simulaciones con distintos valores y 2 fue el que dio mejores resultados.
3. **Prueba de trabajo/Proof of Work:** Para que la transacción sea aceptada por la red, hay que realizar alguna prueba de trabajo. Es importante destacar que, a diferencia con Bitcoin, aquí no hay competencia. No hay otro usuario compitiendo por completar este paso, por lo que no se genera un gasto innecesario de energía por parte de el "perdedor" ni existe posibilidad de que alguien nos quite la oportunidad de realizar la prueba de trabajo.

Una vez realizados estos pasos, la transacción es transmitida a la red en forma de tip. Otro usuario que quiera hacer una transacción seleccionará ese tip y lo validará.

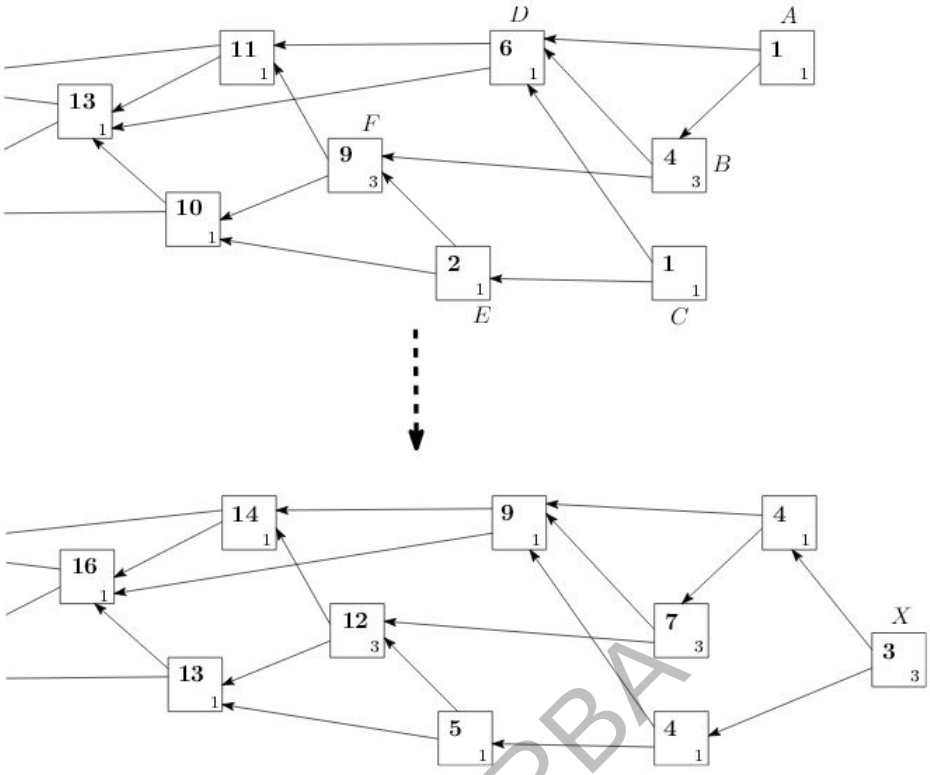
Una vez comprendidos estos pasos, podemos ver cómo la validación es realizada por los propios usuarios en lugar de por otro grupo dedicado a eso. No dependemos de que un minero valide nuestra transacción y la agregue a un ledger único que sólo puede ser modificado secuencialmente: cada usuario valida las transacciones de los demás y agrega su propia transacción al tangle en paralelo en cuanto esté listo para hacerlo.

Propiedades de la transacción

Definamos algunas propiedades de las transacciones:

Peso propio: La cantidad de trabajo que el nodo que realiza la transacción tuvo que hacer. Ese paso sólo puede tener valores de 3^n donde n es un entero mayor o igual a 0.

Peso acumulado: La suma de su peso propio y el peso propio de todas las transacciones que la validan directa o indirectamente. Puede pensarse como la suma de todo el trabajo que generó esa transacción.



El número pequeño es el peso propio y el grande es el acumulado. En la imagen superior A y C son tips. Cuando se agrega el tip X, todas las transacciones suman el peso propio de X(3) a su peso acumulado.

Altura: Longitud del camino más largo entre la transacción y el génesis

Profundidad: Longitud del camino más largo entre la transacción y el tip más cercano

Puntaje: La suma de su peso propio y el peso propio de todas las transacciones validadas por ella directa o indirectamente. Puede pensarse como todo el trabajo que está validando la transacción. También puede pensarse como es trabajo realizado por toda la tangle hasta ese momento.

Selección de tips/MCMC

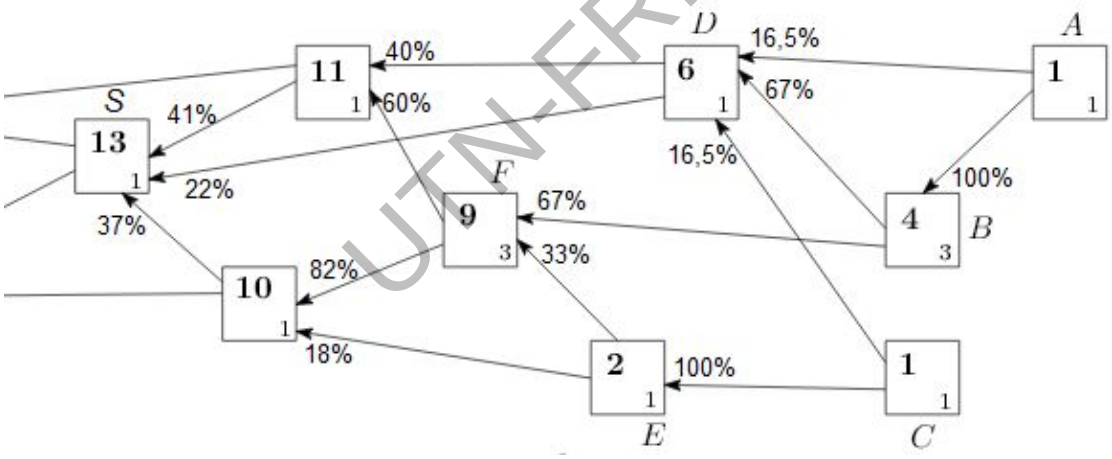
Para seleccionar los tips que se validarán se usa una MCMC(Markov Chain Monte Carlo). Esto es la combinación de una cadena de Markov y una simulación Monte Carlo.

Una cadena de Markov es una forma de modelar situaciones que incluyen un conjunto discreto de estados y probabilidades definidas para saltar de un estado a otro. La probabilidad de pasar de un estado a otro está dada de antemano. Por ejemplo, si yo estoy en el estado A no puedo decir con seguridad a qué estado pasaré después. Puedo tener 70% de probabilidades de pasar al estado B y 30% de pasar al estado C. No importa cómo

llegué al estado A a la hora de decidir el próximo estado, las posibilidades están predefinidas para cada estado y no dependen de nada más.

Una simulación Monte Carlo es una simulación en la que se intenta determinar la distribución de probabilidad de diferentes resultados. Por ejemplo, si estoy jugando con dos dados y quiero saber la distribución de probabilidades de su suma, tengo dos opciones. Puedo calcular para cada resultado cuántas combinaciones posibles hay. Para cada resultado me anoto la cantidad de combinaciones que lo logran y luego divido cada cantidad por la suma de todas las combinaciones. El problema de este método es que tengo que calcular todas las combinaciones que llegan a todos los resultados. Y si estuviese jugando con 10 dados? Y si sólo me interesaba la probabilidad de que salga un resultado? La otra forma de resolverlo es mediante una simulación Monte Carlo. Simplemente tiro los dados un gran número de veces X y anoto cuántas veces salió cada resultado. Luego divido la cantidad de veces que salió cada resultado por X. A mayor X, mejor será la aproximación.

Al combinar ambos, podemos calcular las probabilidades de llegar desde una transacción suficientemente confirmada a cada tip recorriendo una cadena de Markov. Podemos calcular la probabilidad de saltar de una transacción a otra por los pesos acumulados relativos de cada transacción adyacente. Por ejemplo:



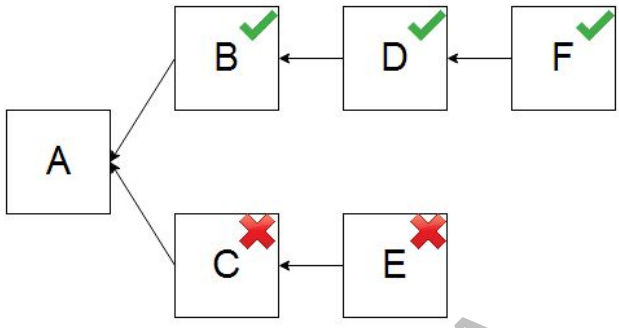
Notemos que, por ejemplo, no importa cómo se llegó a la transacción D. Una vez que se llegó hasta ahí, las probabilidades para seguir son las dadas para esa transacción.

No es necesario empezar desde el génesis(la primer transacción de la tangle, que no valida a nadie). Uno puede elegir cualquier transacción lo suficientemente profunda en la red.

No es necesario usar sólo el peso relativo. Pueden usarse también las otras propiedades para determinar las probabilidades de la cadena de Markov. Sin embargo, se recomienda usar el peso relativo por razones que explicaremos más adelante.

Consenso

Llegar a un consenso en ledgers distribuidos siempre es más complicado que en ledgers centralizados. En blockchain, el consenso sobre si una transacción perteneciente a un bloque es válida depende de la cantidad de bloques que le sigan. Al agregar un bloque B luego de un bloque A, es posible que otro nodo esté agregando otro bloque C luego del bloque A. Eventualmente, la cadena más larga es la que sigue en circulación y la otra se descarta.



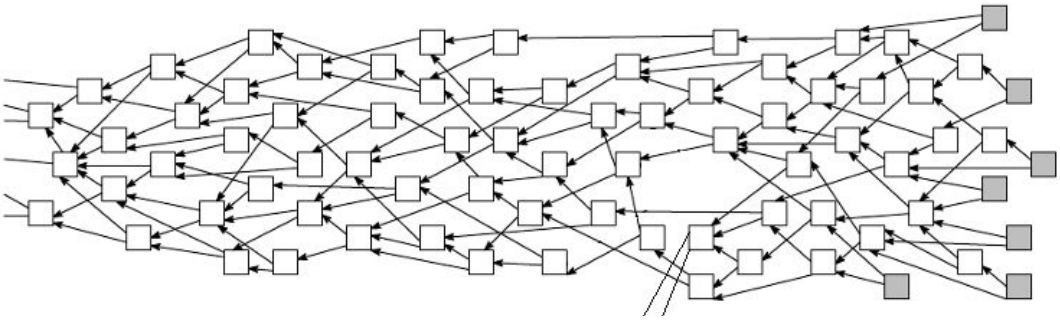
Normalmente se acepta que para asegurarse que una transacción fue confirmada debe haber 6 bloques después de la misma.

También introducimos el concepto del consenso eventual. Podemos discrepar con alguien sobre cuál es el hash del último bloque de la cadena en este momento(ya que quizás al nodo de la otra persona llegó un bloque que todavía no llegó al nuestro) pero no vamos a discrepar sobre cuál es el hash del bloque 20, minado en 2009. Más pertinentemente, podemos discrepar sobre si el último bloque de la cadena incurrió en un doble gasto y será descartado(como en la imagen de arriba), pero podemos acertar con seguridad que el vigésimo bloque no será descartado. Eventualmente, vamos a llegar a un consenso sobre el bloque actual, así como todos estamos de acuerdo en lo que se refiere a los bloques "viejos".

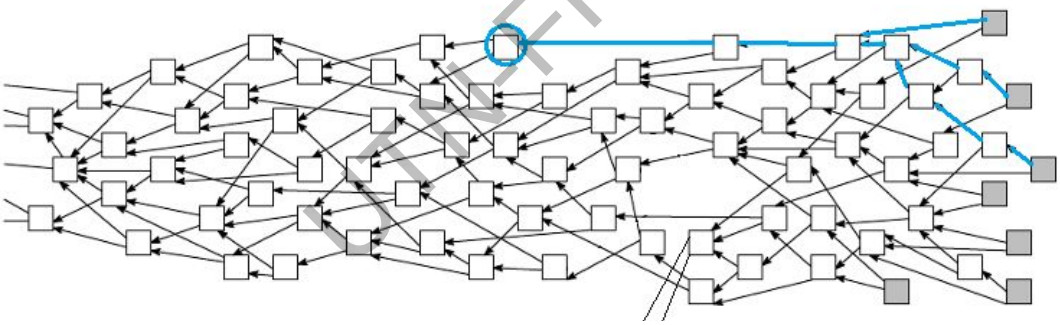
Veamos cómo es posible llegar a ese consenso en tangle. Para una determinada transacción A, se determina qué porcentaje de los tips la validan directa o indirectamente. No siempre es posible tener acceso a todos los tips de la red(después de todo es un sistema distribuido - quizás alguien acaba de agregar un tip del otro lado del mundo y todavía no llegó hasta nuestro nodo), por lo que voy a usar sólo cierto número de nodos (por ejemplo 100). Luego, se calcula si cada tip valida o no la transacción A. Si 20 de esos 100 nodos la validan, se puede decir con un 20% de seguridad que la transacción está confirmada. Si 99 nodos la validan, se podrá decir con un 99% de seguridad.

Cada vendedor puede decidir cuánta seguridad quiere esperar a tener antes de entregar el producto(similar a cómo un vendedor que acepta bitcoins puede esperar a que una transacción tenga 1, 6 o 50 bloques de confirmación).

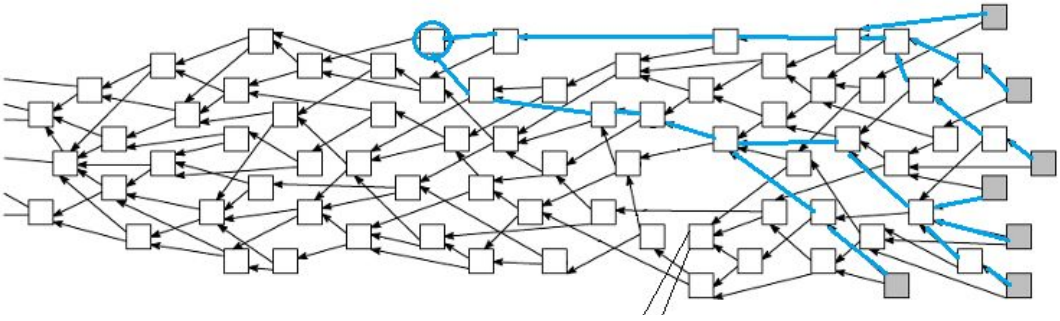
Veamos un ejemplo. Los cuadrados representan transacciones. Los cuadrados grises son tips. Cuáles cuadrados transacciones son referenciadas por todos los tips?



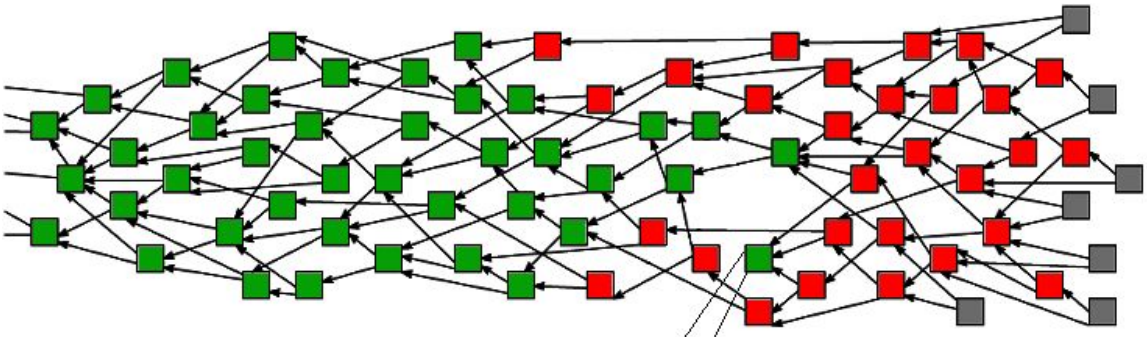
Probemos con el cuadrado redondeado. Sólo 3 de 7 tips lo confirman (43%). A medida que nuevos tips confirmen los anteriores, los 4 tips que no validan la transacción dejarán de ser tips, y por lo tanto dejarán de importan a la hora de validar esa transacción. Como en el ejemplo de Bitcoin, si bien quizás ahora discrepamos sobre la validez de una transacción, eventualmente llegaremos a un consenso.



Ahora probemos con otro. 7 de 7 tips lo confirman (100%)



Si pintamos de verde las transacciones que tienen 100% de confirmación y de rojo el resto, el grafo nos queda así:



Podemos destacar otra ventaja de la red tangle: para validar una transacción A, no nos interesa conocer todas las transacciones anteriores a la misma(las que están a la izquierda en el gráfico). Sólo nos interesa la transacción A y las que la referencian(las que están a la derecha en el gráfico).

Posibles ataques

Double spending

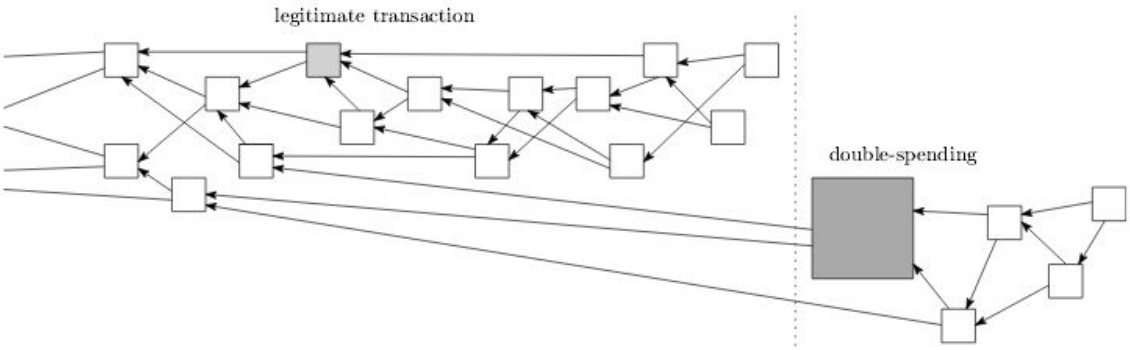
Pensemos en el siguiente ataque a Bitcoin. Una persona con muchísimo poder de procesamiento mina 4 bloques pero no los transmite a la red. Entre esos bloques se encuentra una transferencia enviando Bitcoins a sí mismo. Luego, hace una compra referenciando la misma transferencia. El comerciante espera a que esté confirmado por 2 bloques y le entrega el producto. En ese momento, el atacante transmite a la red sus cuatro bloques. La transferencia con la que compró el producto se descarta por haber una cadena más larga y el comerciante se queda sin el producto y sin el dinero.

Un double spending es tangle sería similar. Comprar algo, recibir el producto y luego volver a gastarlo. La pregunta es cómo evitar que se tenga en cuenta el segundo y se descarte el primero. Como ya dijimos, la forma elegir una tip para referenciar es corriendo un algoritmo MCMC basado en los pesos acumulados. Pero si el atacante validó su transacción con muchas pequeñas transacciones? Y ahora esa nueva transacción tiene más peso acumulado que la original?

La respuesta es similar a la se da en Bitcoin: no permitirle a un atacante llegar a ese nivel de procesamiento comparado con el resto de la red. Al igual que en Bitcoin, se calculó que no se le debe permitir a nadie controlar más del 34% de todo el procesamiento. Es por eso que Iota(la moneda administrada por los creadores de tangle) tiene un “coordinador” y un

gran poder de procesamiento hasta que la red esté lista para *defenderse por sí misma*. No se debe permitir a una misma clave introducir muchos tips en poco tiempo.

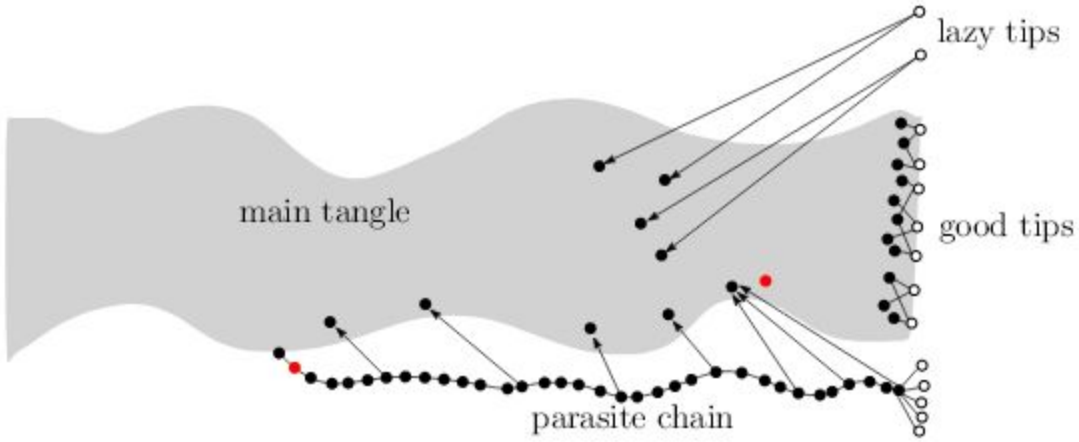
Y si el atacante tiene preparado un doble gasto de antemano con un proof of word muy grande, sin llegar a tener el 34% del poder de procesamiento? En ese caso podría pasar que la red empiece a referenciar esa transacción, ya que podría aumentar repentinamente el peso acumulado de una zona de la red, haciéndola más atractiva al algoritmo MCMC.



Antes habíamos dicho que podíamos elegir nuestro proof of work. Una posible solución es poner un límite bajo sobre qué tan elevado puede ser nuestro proof of work. O también directamente podemos determinar que todas las transacciones requieran el mismo valor de proof of work y que por lo tanto todas las transacciones tengan el mismo peso propio.

Cadena parásito

Un posible ataque similar consiste en armar una subtangle paralela ocasionalmente referenciando la tangle principal. Esta subtangle tendría más puntaje que la principal. Recordemos que el puntaje es equivalente a todo el trabajo validado por la transacción. El puntaje de las transacciones honestas es el trabajo de toda la tangle, y las transacciones de la subtangle paralela podrían sumarle a esto el trabajo realizado en su subtangle. También, como al armar la cadena parásito el atacante no debe preocuparse por la latencia de la red, puede crear más transacciones y tener más altura. También, el atacante puede dejar muchas tips abiertas al momento de lanzar el ataque para que sea más probable lo validen a él comparado a que validen las tips honestas.



Los círculos rojos indican double spending

Para defendernos de esto, usaremos la suposición que la tangle principal tiene más poder de procesamiento que cualquier atacante en cualquier momento dado, y por lo tanto la tangle principal tendrá más peso acumulado que la paralela. La idea será que el algoritmo de selección de tips para validar nunca elija validar las de la cadena paralela.

Para ello utilizaremos el algoritmo MCMC anteriormente descrito. Situaremos N *partículas* en transacciones de la red que consideremos confirmadas. Dejaremos que las partículas "caminen" hacia las tips, siendo parciales hacia las transacciones de mayor peso acumulado y haciendo que cada salto tome cierta cantidad fija de tiempo. Las primeras dos tips en recibir partículas serán referenciadas.

Veamos cómo afecta esto a las **lazy tips**, es decir, tips que referenciaron transacciones "viejas" y que ya están confirmadas con un alto porcentaje de probabilidad. Simplemente eligieron referenciar transacciones que seguramente sean correctas para ahorrarse realizar la validación. Las tips que referencian transacciones más nuevas tendrán más muchísimo más peso acumulado por todas las transacciones intermedias. Es por eso que será muy poco probable que una lazy tip sea seleccionada para ser referenciada.

Es por el mismo motivo que no funciona este tipo de ataque: las transacciones de la cadena paralela tienen muy poco peso acumulado. Consecuentemente es muy poco probable que la partícula salte de la tangle principal a la paralela y luego decida referenciar sus tips. En el raro caso de que una partícula haga ese salto, como la red paralela seguramente tenga más altura y cada salto toma una cantidad fija de tiempo seguramente otra partícula de la red principal llegará primero.

Splitting attack

Probablemente el ataque más extraño que se plantea sea el siguiente. Qué pasaría si alguien hace un doble gasto a propósito en "el medio" de la tangle? Como el peso acumulado total en ambas subtangles son parecidos, el atacante intentará mantener el equilibrio, generando spam en la subtangle con menos peso. El atacante esperará que la

mitad de los nodos honestos realicen transacciones en cada mitad, manteniendo el balance entre ambas ramas. el atacante podría entonces realizar dobles gastos: uno en cada rama.

Primero, debemos considerar las probabilidades de que un nodo honesto use una u otra rama. Si una rama tiene un peso de 550 y la otra de 570, qué tan balanceados serán los nodos honestos? Si son muy balanceados, esto representará un problema. Pero si una gran mayoría trabaja en el lado de la segunda rama, la primera se quedará huérfana eventualmente. El problema es que la decisión de qué rama tomar sólo se dará si el MCMC empieza antes de la separación. Si uno empieza a recorrer desde una de las ramas, no podrá pasarse a la rama de más peso.

Es importante notar que este tipo de ataque es muy difícil, porque el nodo del atacante no necesariamente tiene todos los últimos tips por la naturaleza distribuida de la tangle. Es muy posible que esté viendo una porción del total, y por lo tanto no esté trabajando con los pesos acumulados actuales. También notemos que en caso de detectar algo así, los usuario podría ponerse de acuerdo para aumentar repentinamente el peso de una de las ramas, intentando descartar la otra. Los usuario podrían detectarlo, debido a la forma de detectar las confirmaciones que explicamos anteriormente. Los usuarios verían sólo un 50% de confirmación en sus nuevas transacciones y sería obvio que sólo la mitad de los tips pueden ver su transacción, por lo que la misma debe estar en una rama.

También pueden considerarse cambios al algoritmo de selección de tips, para que al principio sea más determinístico y luego(cerca de los tips, cuando se asegure de no estar en una rama aislada) empiece a ser más probabilístico para que los tips se repartan entre todos los usuarios. De todas formas este tipo de ataques sería muy complejo y sería descubierto muy rápido como para necesitar este tipo de cambios. Sin embargo, definitivamente sería muy molesto para las personas cuyas transacciones quedaron en la rama descartada.

Sobre la adopción del algoritmo MCMC

La sección anterior está basada en que la mayoría de los nodos usen algoritmos similares para elegir sus tips. Cómo sabemos que todos usarán el mismo algoritmo? Si un nodo egoísta hace una simulación Monte Carlo y sabe cuál es “la zona” de los tips que tiene más posibilidades de ser alcanzada por el algoritmo MCMC que usa la mayoría, pondría siempre sus tips ahí para que los referencien más rápido.

Para empezar aclaremos que por los delays computacionales y de red, lo mejor sería correr ese algoritmo en el momento de hacer una transacción sobre un snapshot de la tangle en ese momento. Si el tip al que se llega ya fue referenciado, se puede actualizar el snapshot y seguir avanzando a un tip actual.

Supongamos la siguiente situación:

En un instante *t0* la zona que más probabilidades tiene de ser referenciada tiene el 8% de los nodos honestos recorriendo el MCMC para llegar hasta allí.

En *t1* muchos nodos egoístas(más que el 8% en camino) referencian tips en esa zona con el objetivo de ser validados rápido por todos los nodos en camino. Los tips referenciados ya no son tips sino transacciones, y el enorme número de nuevas tips egoístas esperan que las validen. Como los nodos honestos están trabajando sobre un snapshot no ven este nuevo influjo de peso acumulado sobre la zona, por lo que no aumenta el tráfico hacia esa zona.

En *t2* los nodos honestos llegan a sus destinos en el snapshot. Actualizan el snapshot y descubren que esos tips ya fueron referenciados. Ahora hay tantos tips egoístas comparados con los nodos honestos que la competencia es feroz y los tips egoístas tardan todavía más de lo normal en validarse.

La conclusión de esto es no sólo que usar muchos nodos egoístas para referenciar la misma zona es contraproducente, *sino que la mejor estrategia es tener el mismo algoritmo(o lo más parecido posible) a todos los demás nodos*. De esta forma, uno puede asegurarse que habrá la cantidad justa de nodos recorriendo la tangle usando MCMC como para poder referenciar los tips. Puede hacerse una analogía al punto de equilibrio entre oferta y demanda. Yo estoy dispuesto a ir a comprar todas las manzanas que pueda por $100 por un televisor. Cuando estoy yendo al mercado, los vendedores se enteran que estoy yendo a comprar y se amontonan en la zona donde yo voy a llegar. Cuando llego, la competencia es tan feroz que deben bajar el precio de sus manzanas considerablemente para poder competir. Además, mucho vendedores no podrán vender las manzanas que tenían previstas debido a la reciente suba de la oferta. Si se hubiesen quedado en su lugar inicial, quizá podrían haberlas vendido todas a precio normal.

Conclusiones

Tangle presenta ciertas ventajas sobre una blockchain. Principalmente, el hecho de que todos los nodos puedan escribir en el ledger al mismo tiempo hace que el funcionamiento de la red no sea dependiente de los mineros ni del ciclo de la minería. Cada nodo puede anexar sus transacciones cuando quiera. Esto a su vez hace que no sea necesario cobrar comisiones.

Las desventajas que presenta están asociadas a lo experimental y poco probado de esta tecnología. Como todavía no es usada en gran escala por mucho usuarios, quizás todavía queden ataques posibles por descubrir.