

Sistemas Peer-to-Peer

5. Bitcoin

Francisco Sant'Anna
francisco@ime.uerj.br



Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network 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 longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU 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 as proof of what happened while they were gone.

Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the **double-spending problem** using a peer-to-peer network. The network 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 longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU 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 as proof of what happened while they were gone.

**Por quê é tão difícil criar um
sistema monetário distribuído?**

Sistemas Monetários

Sistemas Monetários



Sistemas Monetários



- Uso descentralizado

Sistemas Monetários



- Uso descentralizado
- Emissão centralizada e irrestrita

Sistemas Monetários



- Uso descentralizado
- Emissão centralizada e irrestrita
- Criação difícil / Verificação fácil

Sistemas Monetários



- Uso descentralizado
- Emissão centralizada e irrestrita
- Criação difícil / Verificação fácil

Sistemas Monetários



```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

- Uso descentralizado
- Emissão centralizada e irrestrita
- Criação difícil / Verificação fácil

Sistemas Monetários



```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

- Uso descentralizado
- Emissão centralizada e irrestrita
- Criação difícil / Verificação fácil

Sistemas Monetários



```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

```
$ gpg --sign 10-reais.txt
```

- Uso descentralizado
- **Emissão centralizada e irrestrita**
- Criação difícil / Verificação fácil

Sistemas Monetários



- Uso descentralizado
- Emissão centralizada e irrestrita
- Criação difícil / Verificação fácil

```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

```
$ gpg --sign 10-reais.txt
```

```
$ cp 10-reais.txt ...
```


Sistemas Monetários



- Uso descentralizado
- **Emissão centralizada e irrestrita**
- Criação difícil / Verificação fácil

```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

```
$ gpg --sign 10-reais.txt
```

```
$ cp 10-reais.txt ...
```

double spend

Sistemas Monetários



```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

```
$ gpg --sign 10-reais.txt
```

```
$ cp 10-reais.txt ...
```

double spend

- Uso descentralizado
- **Emissão centralizada e irrestrita**
- Criação difícil / Verificação fácil



Sistemas Monetários



```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

```
$ gpg --sign 10-reais.txt
```

```
$ cp 10-reais.txt ...
```

double spend

- Uso descentralizado
- **Emissão centralizada e irrestrita**
- Criação difícil / Verificação fácil



- Uso descentralizado

Sistemas Monetários



```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

```
$ gpg --sign 10-reais.txt
```

```
$ cp 10-reais.txt ...
```

double spend

- Uso descentralizado
- **Emissão centralizada e irrestrita**
- Criação difícil / Verificação fácil



- Uso descentralizado
- **Emissão descentralizada e restrita**

Sistemas Monetários



```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

```
$ gpg --sign 10-reais.txt
```

```
$ cp 10-reais.txt ...
```

double spend

- Uso descentralizado
- **Emissão centralizada e irrestrita**
- Criação difícil / Verificação fácil



- Uso descentralizado
- **Emissão descentralizada e restrita**
- Criação difícil / Verificação fácil

Sistemas Monetários



```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

```
$ gpg --sign 10-reais.txt
```

```
$ cp 10-reais.txt ...
```

double spend

- Uso descentralizado
- **Emissão centralizada e irrestrita**
- Criação difícil / Verificação fácil



- Uso descentralizado
- **Emissão descentralizada e restrita**
- Criação difícil / Verificação fácil

Sistemas Monetários



```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

```
$ gpg --sign 10-reais.txt
```

```
$ cp 10-reais.txt ...
```

double spend

- Uso descentralizado
- **Emissão centralizada e irrestrita**
- Criação difícil / Verificação fácil



- Uso descentralizado
- **Emissão descentralizada e restrita**
- Criação difícil / Verificação fácil

Sistemas Monetários



```
$ edit 10-reais.txt  
Cédula de 10 reais.
```

```
$ edit 10-reais-2.txt  
Cédula de 10 reais.
```

```
$ gpg --sign 10-reais.txt
```

```
$ cp 10-reais.txt ...
```

double spend

- Uso descentralizado
- **Emissão centralizada e irrestrita**
- Criação difícil / Verificação fácil



Como criar um recurso digital que seja escasso?

- Uso descentralizado
- **Emissão descentralizada e restrita**
- Criação difícil / Verificação fácil

Bitcoin

Bitcoin

- Problema: Não há como distinguir cópias de cédulas.

Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.

Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.

Eu, João, transfiro
1 bitcoin para Maria.

Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.

Eu, João, transfiro
1 bitcoin para Maria.

($t=1001$)

Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.

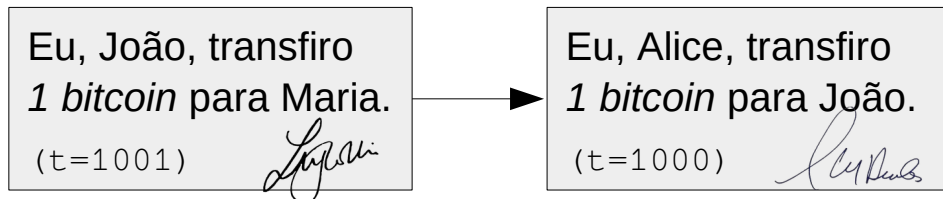
Eu, João, transiro
1 bitcoin para Maria.

(t=1001)



Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.



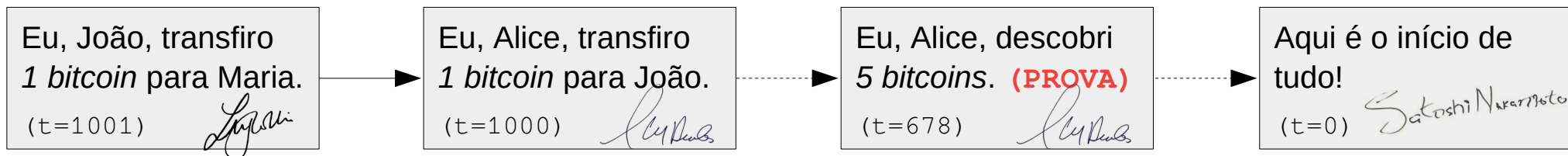
Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.



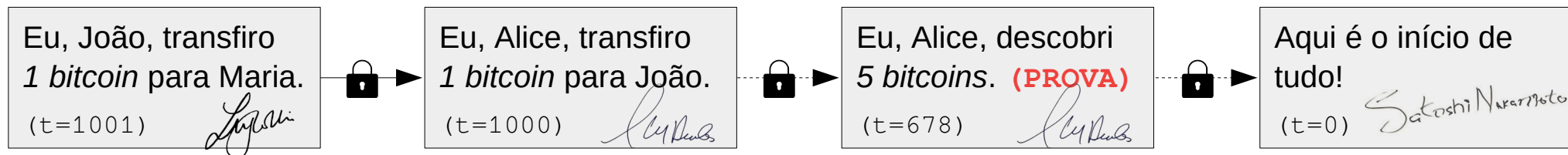
Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.



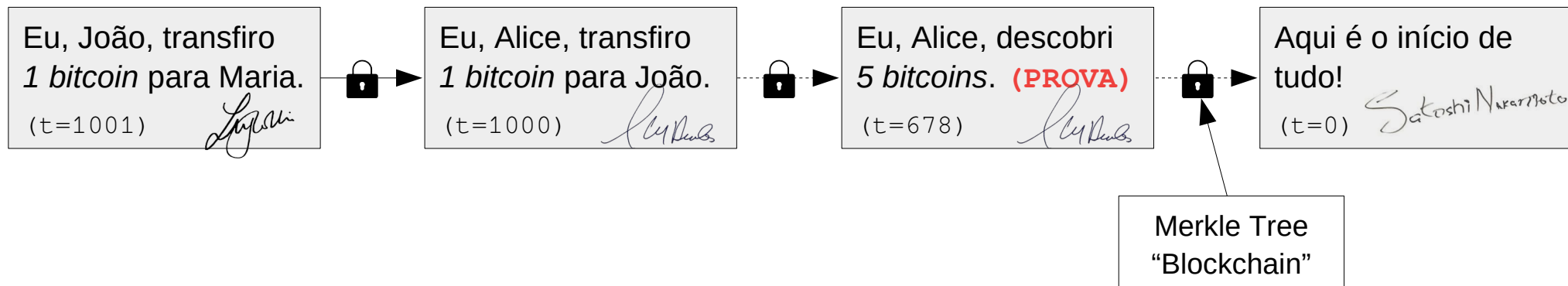
Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.



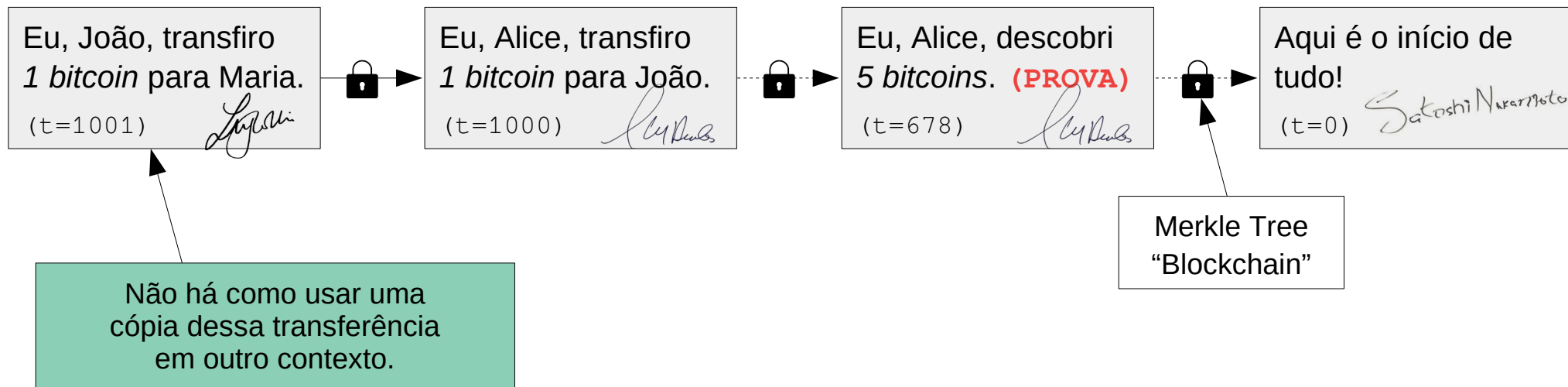
Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.



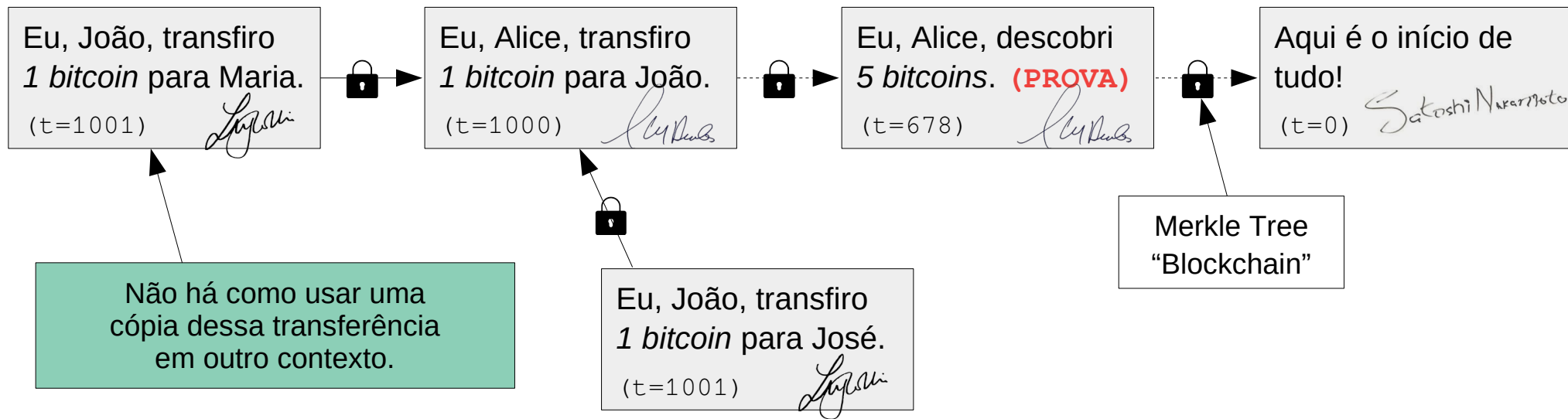
Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.



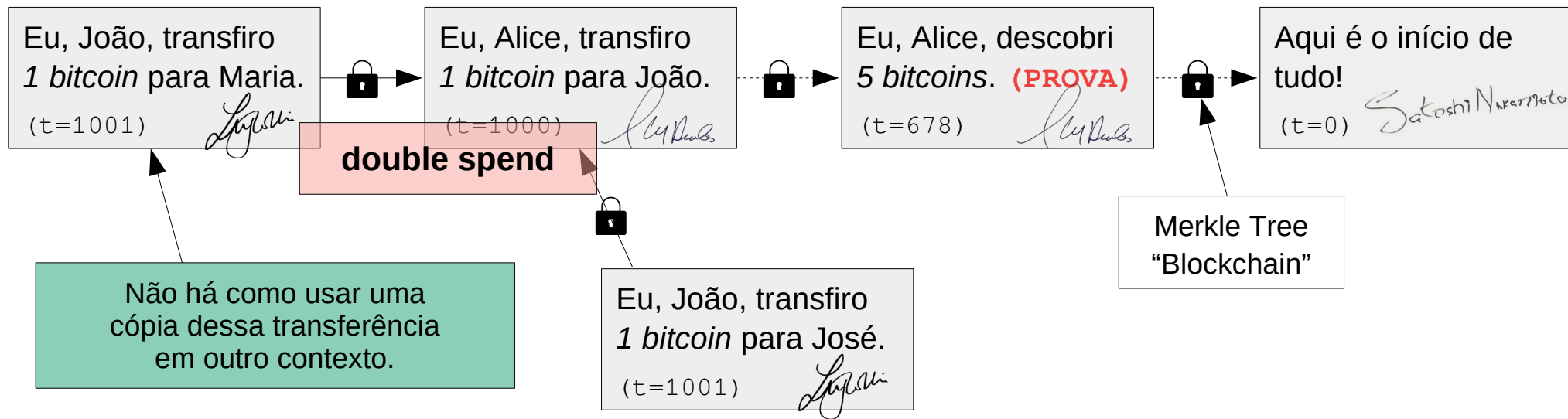
Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.



Bitcoin

- Problema: Não há como distinguir cópias de cédulas.
- Solução: Em vez de assinar as cédulas em si, assinamos as transferências em uma linha do tempo.



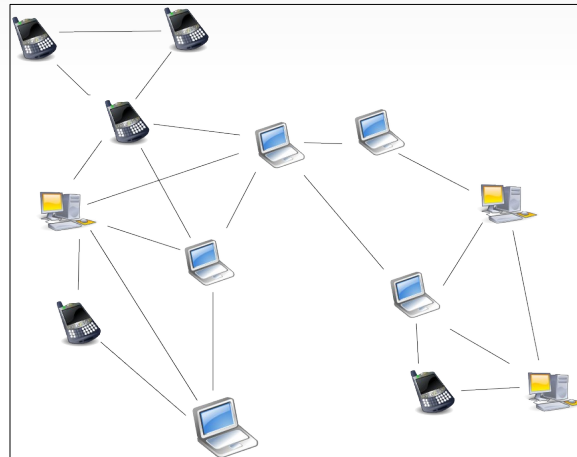
Double Spend

Double Spend

- O Bitcoin é uma rede P2P não estruturada.

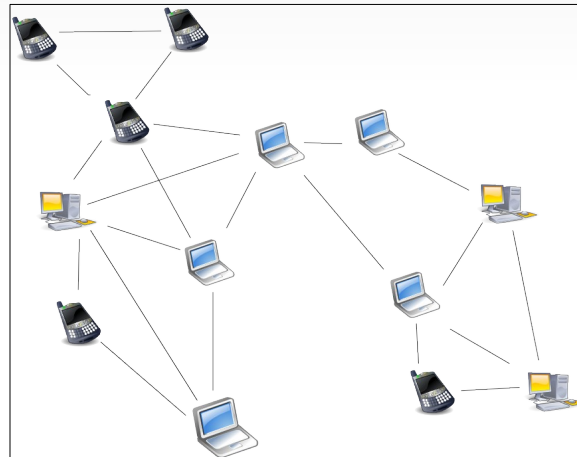
Double Spend

- O Bitcoin é uma rede P2P não estruturada.



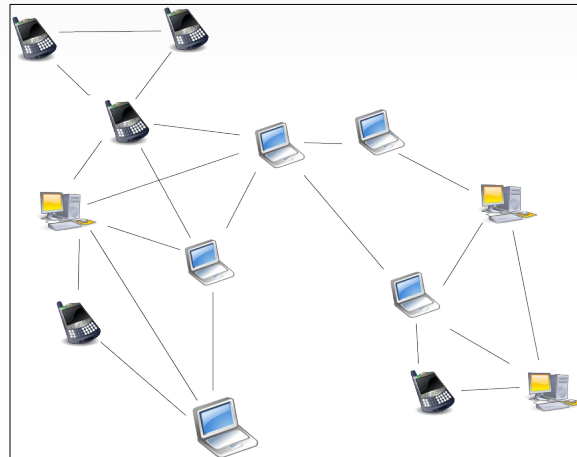
Double Spend

- O Bitcoin é uma rede P2P não estruturada.
- Não existe um estado único global.



Double Spend

- O Bitcoin é uma rede P2P não estruturada.
- Não existe um estado único global.



Eu, Alice, transfiro
1 bitcoin para João.
(t=1000) *Alice*



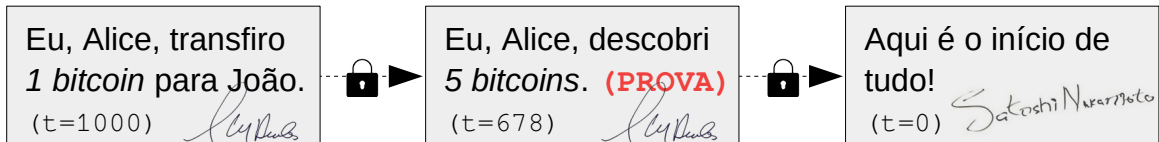
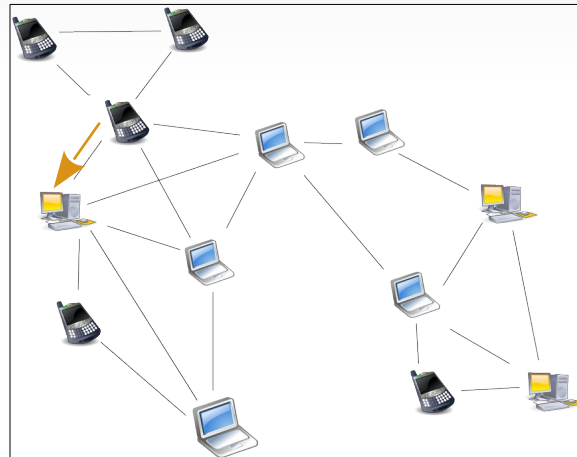
Eu, Alice, descobri
5 bitcoins. **(PROVA)**
(t=678) *Alice*



Aqui é o início de
tudo!
(t=0) *Satoshi Nakamoto*

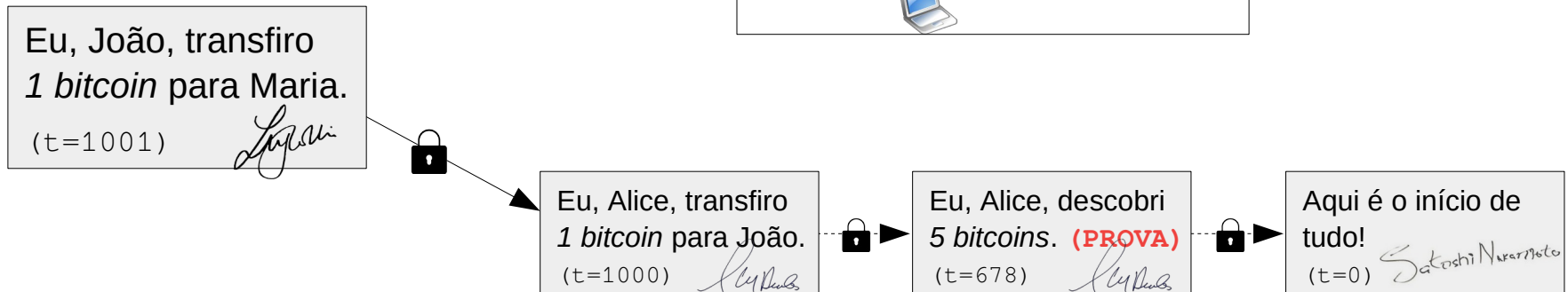
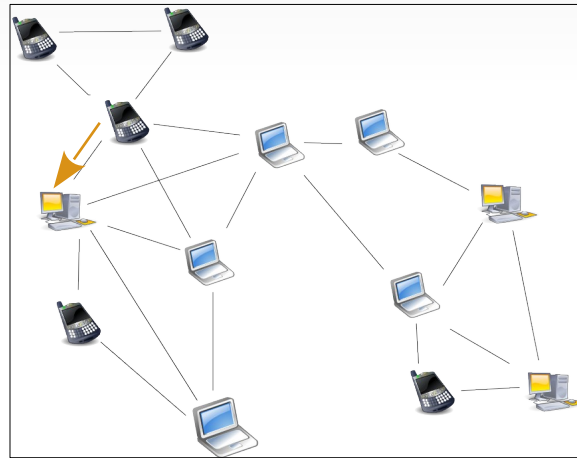
Double Spend

- O Bitcoin é uma rede P2P não estruturada.
- Não existe um estado único global.



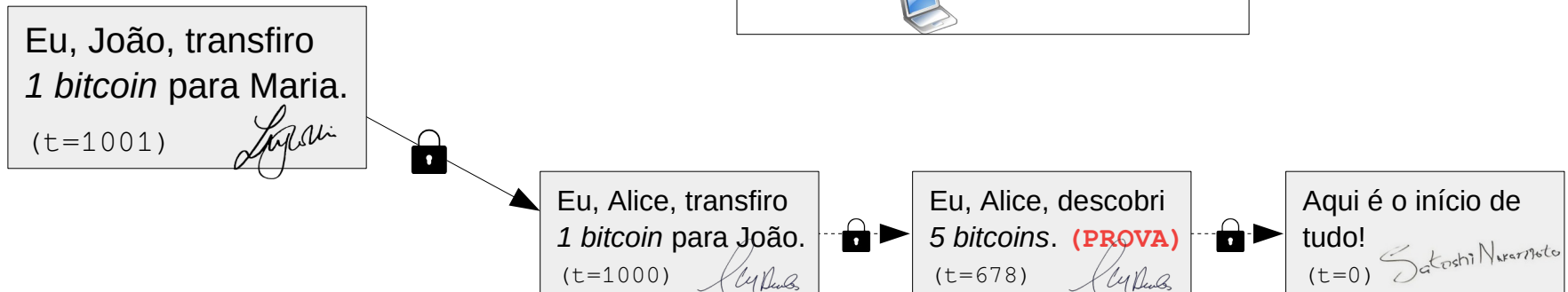
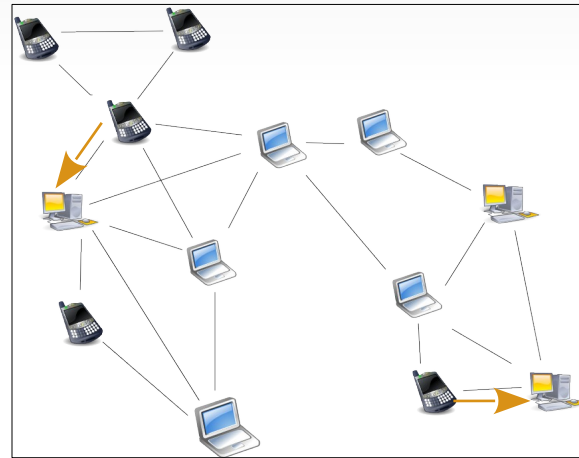
Double Spend

- O Bitcoin é uma rede P2P não estruturada.
- Não existe um estado único global.



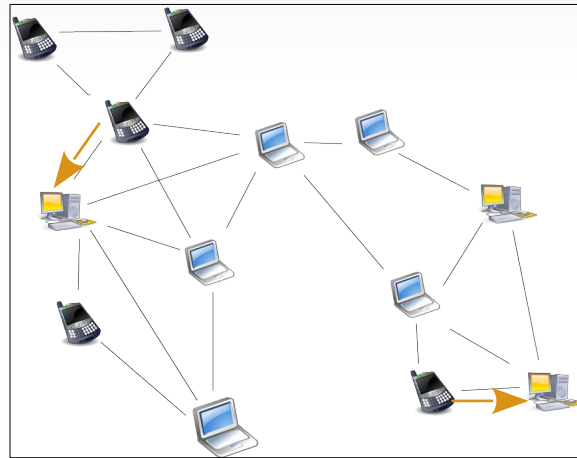
Double Spend

- O Bitcoin é uma rede P2P não estruturada.
- Não existe um estado único global.



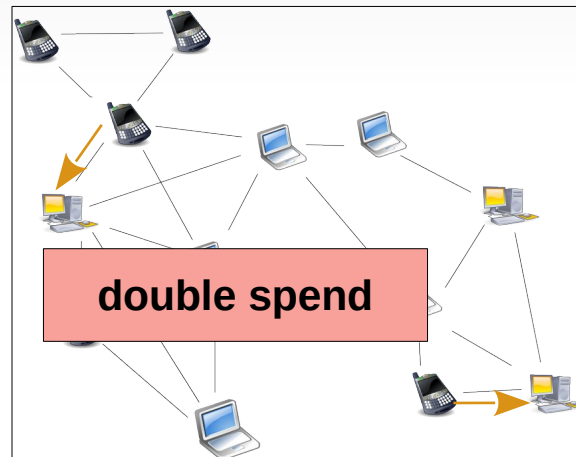
Double Spend

- O Bitcoin é uma rede P2P não estruturada.
- Não existe um estado único global.



Double Spend

- O Bitcoin é uma rede P2P não estruturada.
- Não existe um estado único global.



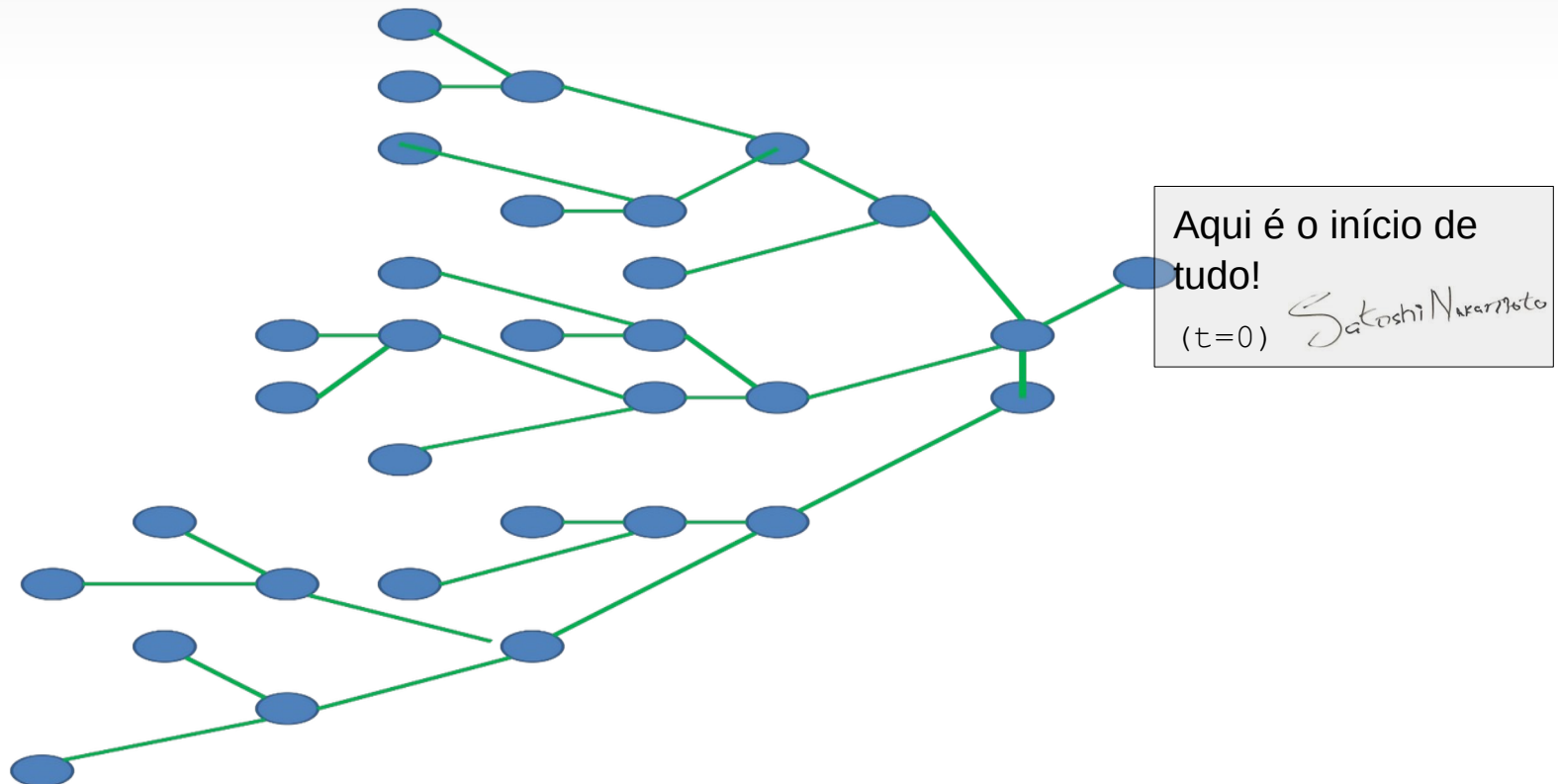
Consenso

Consenso

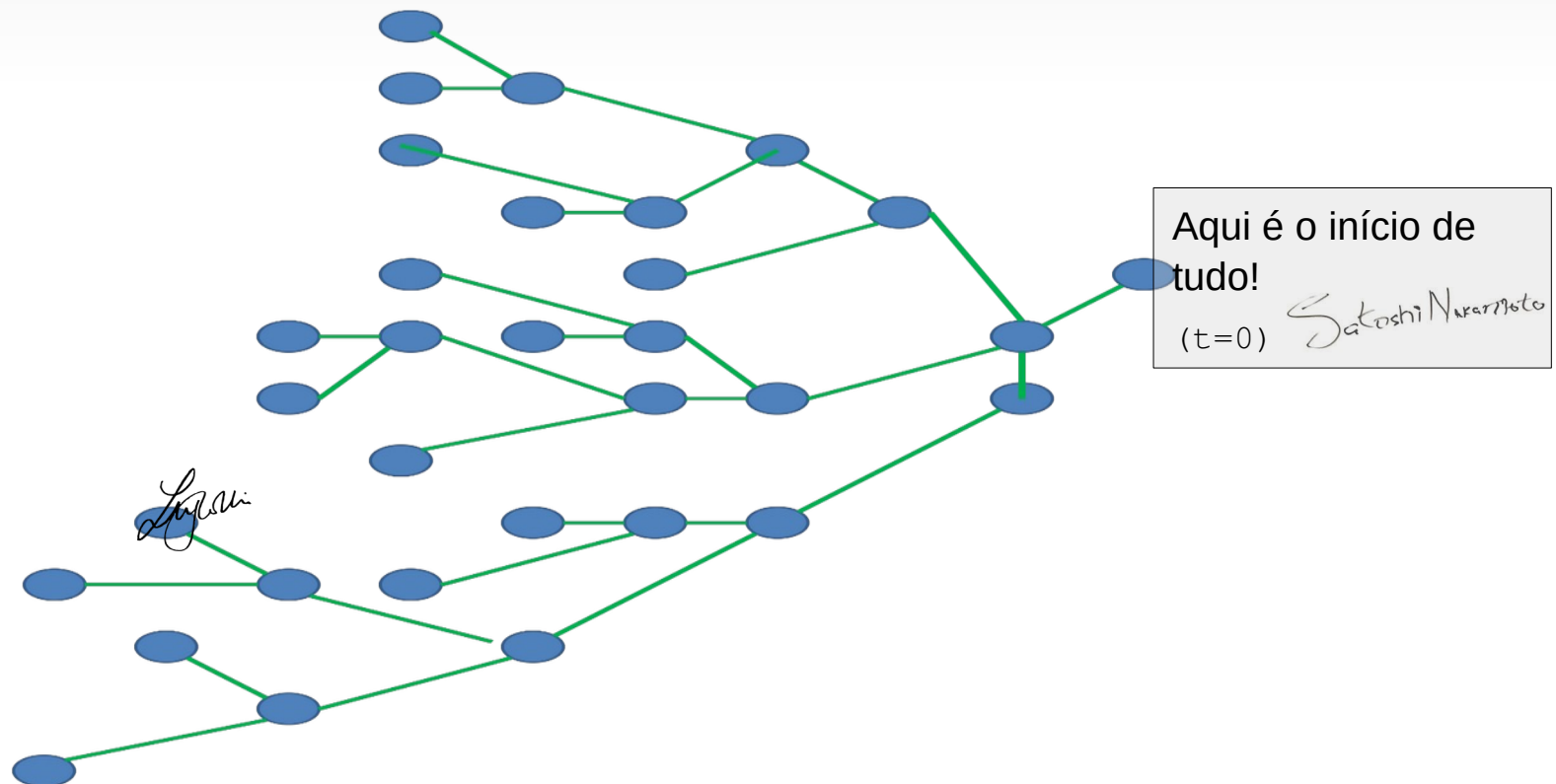
Aqui é o início de
tudo!

(t=0) Satoshi Nakamoto

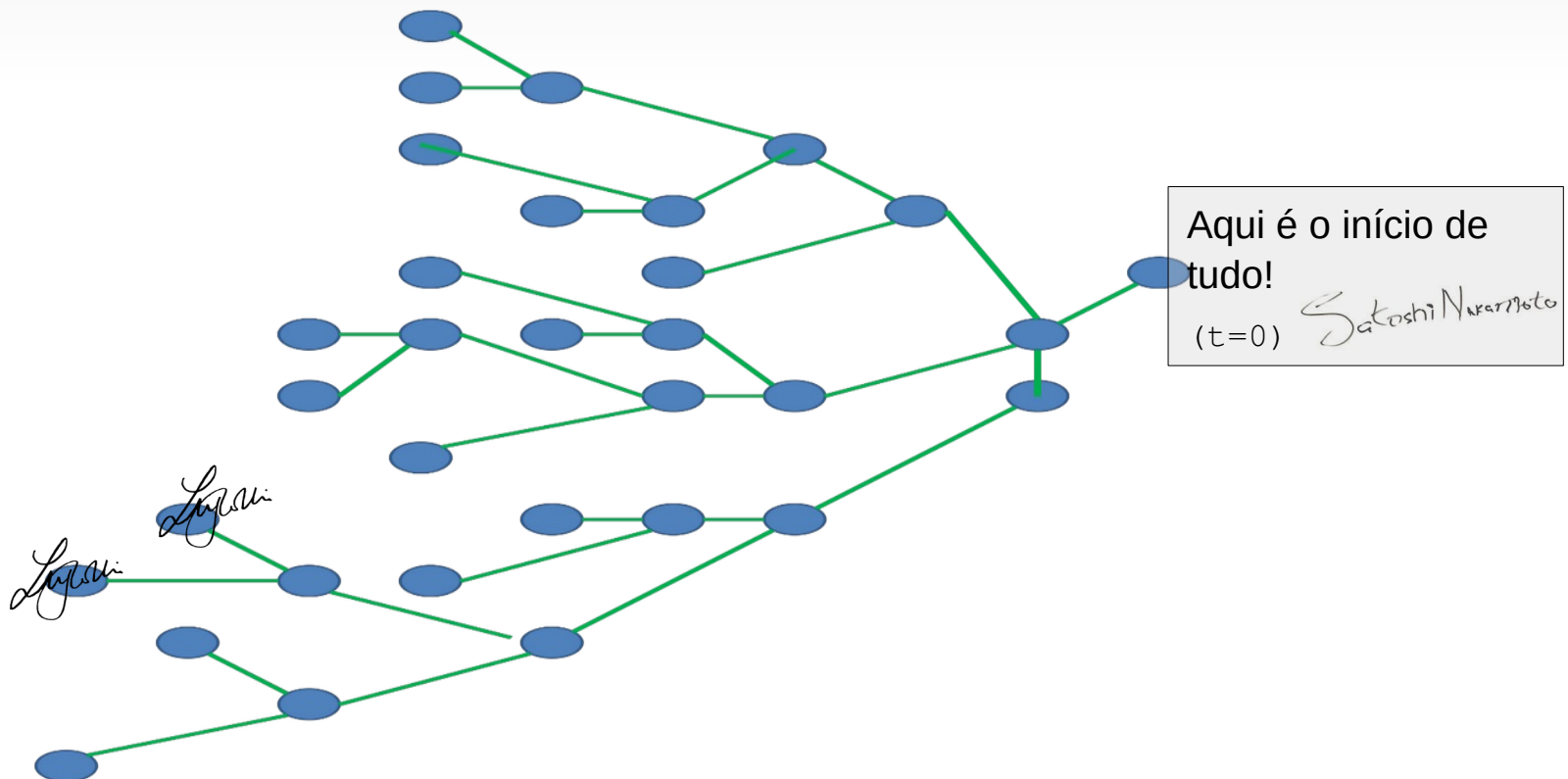
Consenso



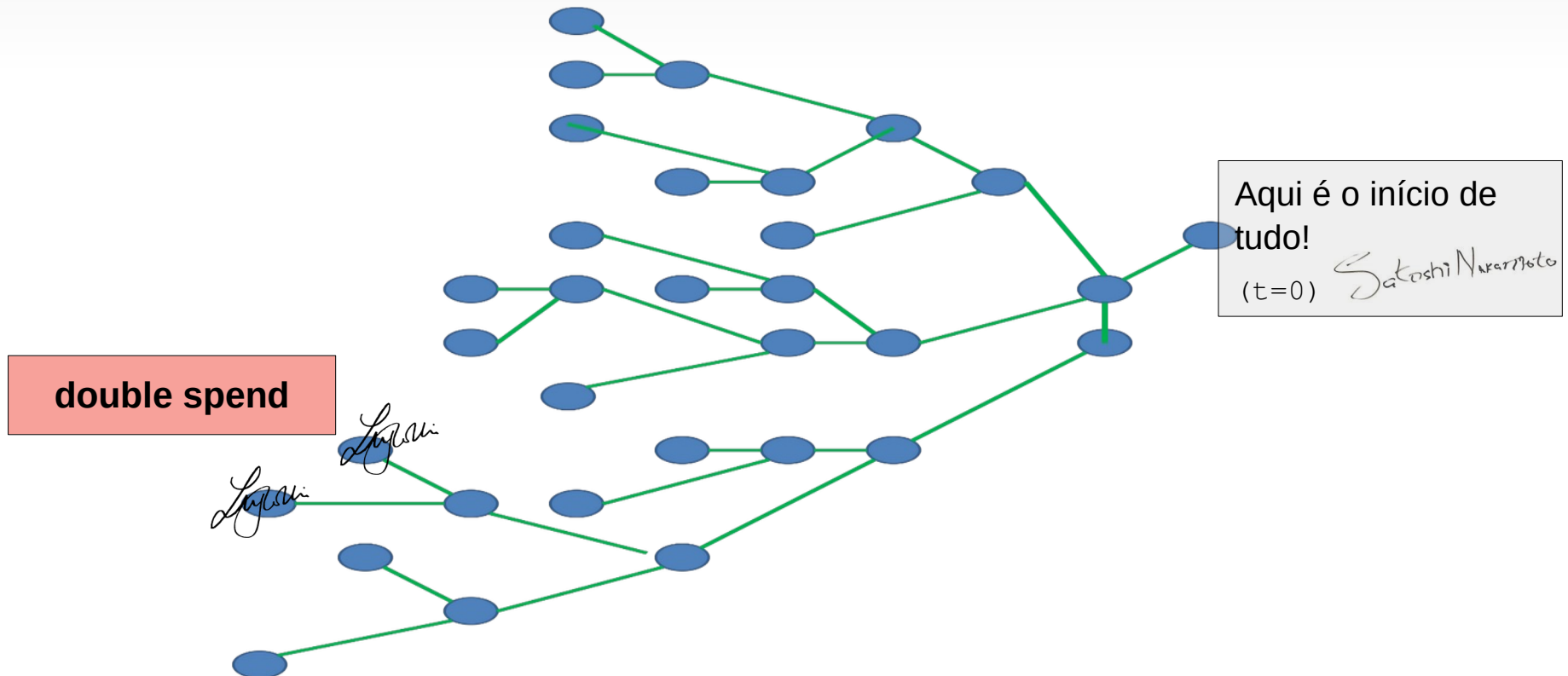
Consenso



Consenso

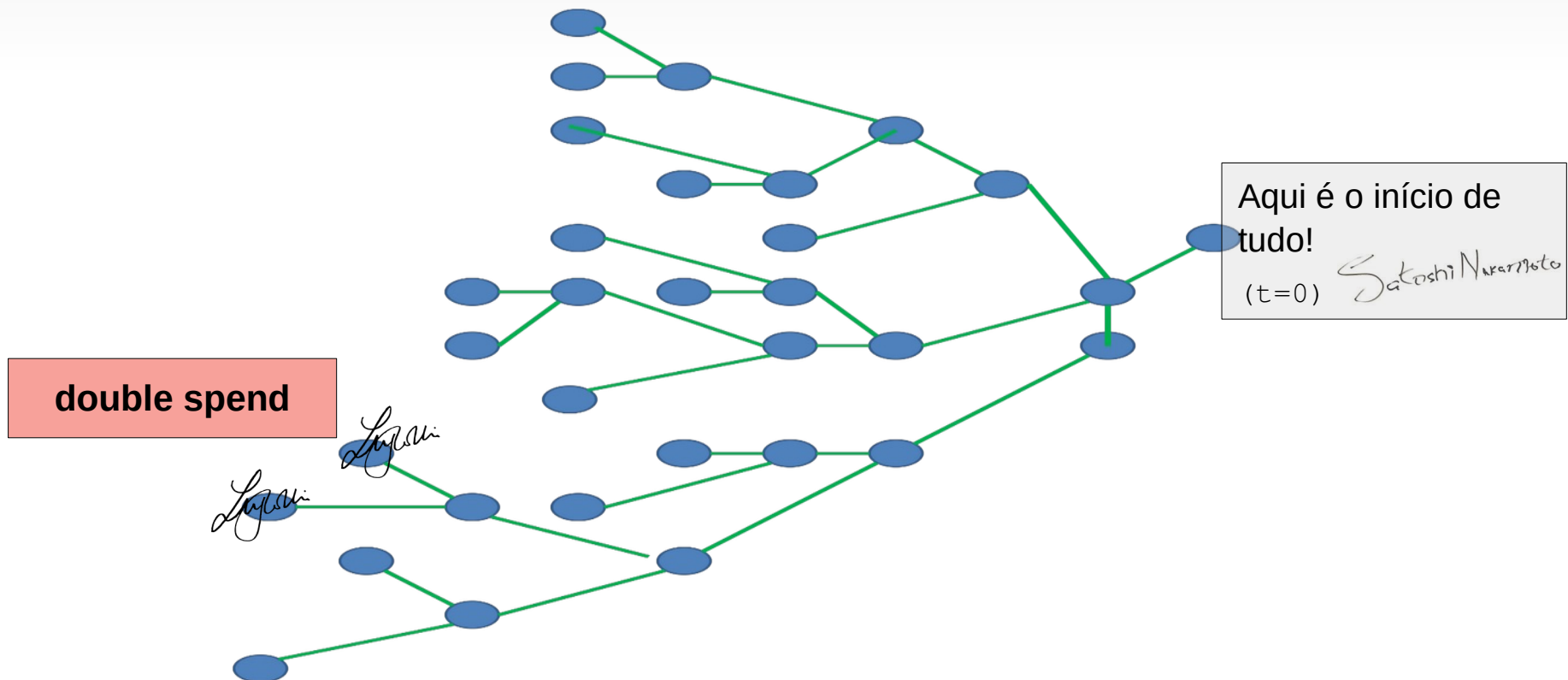


Consenso



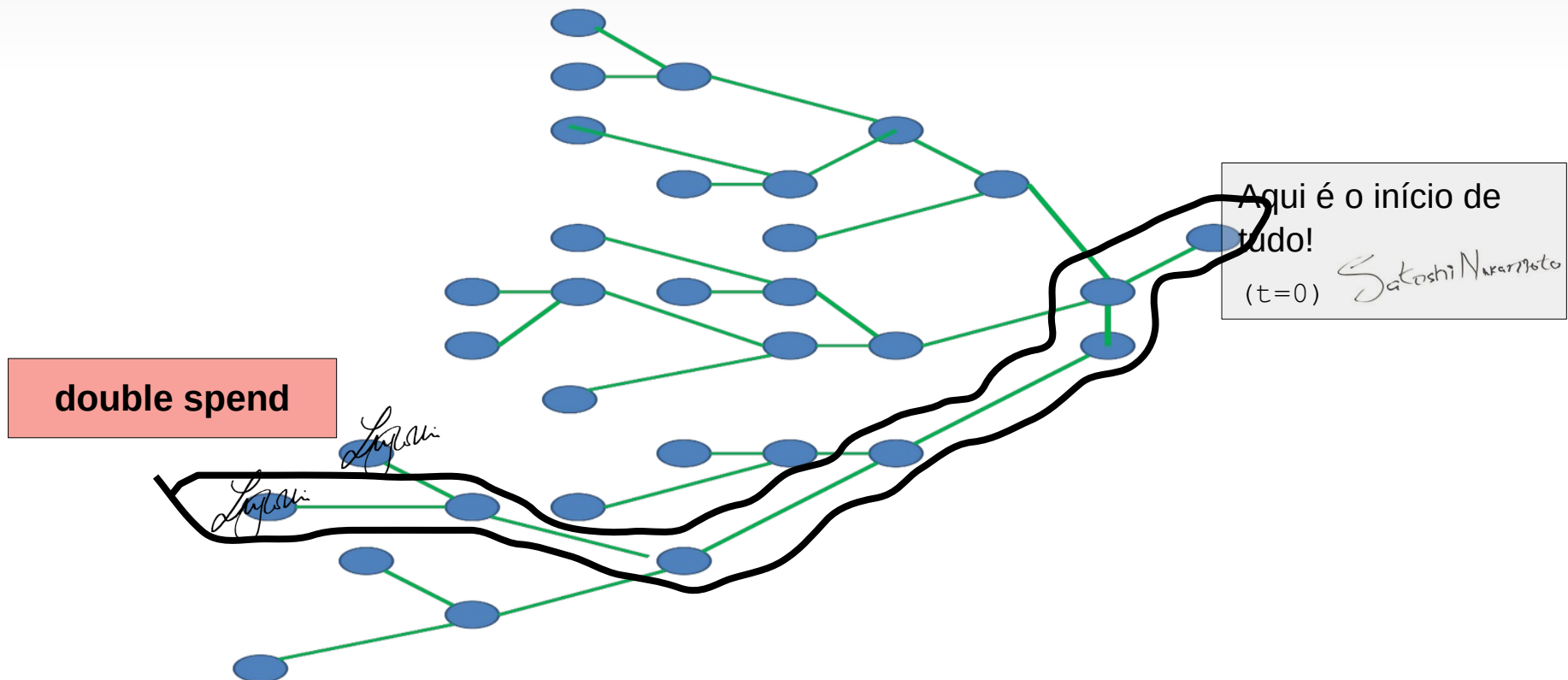
Consenso

- Somente um caminho deve ser válido



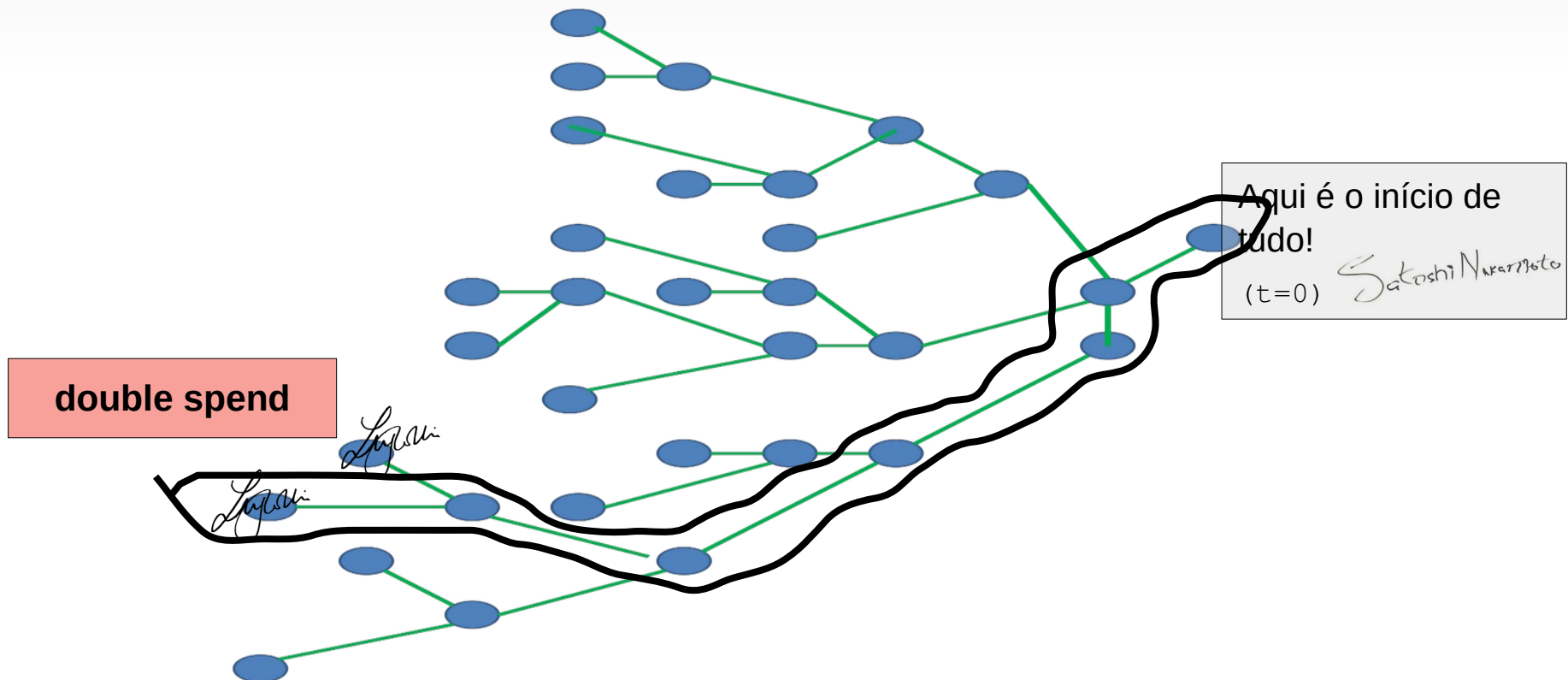
Consenso

- Somente um caminho deve ser válido



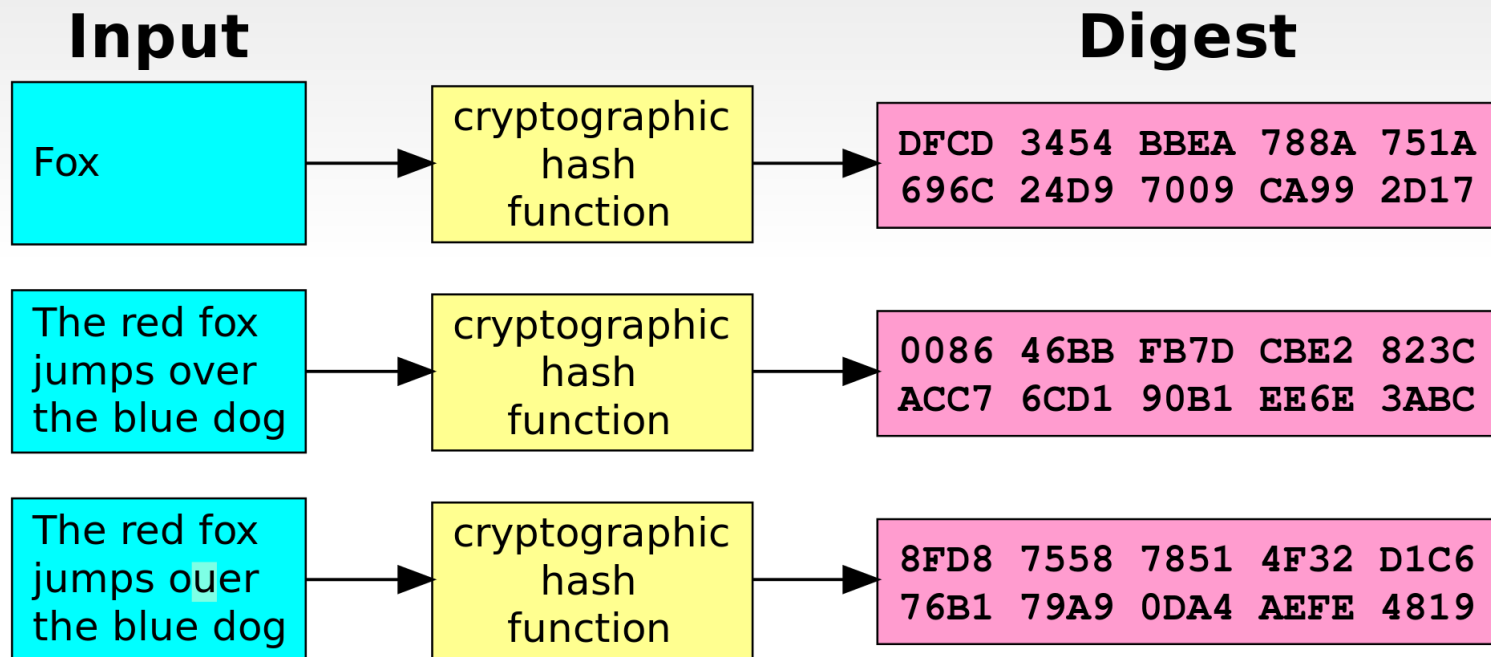
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado



Função Hash Criptográfica

Função Hash Criptográfica



Função Hash Criptográfica

Função Hash Criptográfica

- Determinística:

Função Hash Criptográfica

- Determinística:

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BB EA ... CA99 2D17"}$

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BB EA ... CA99 2D17"}$

- “One-way” (não inversível):

Função Hash Criptográfica

- Determinística:

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BB EA ... CA99 2D17"}$

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BB EA ... CA99 2D17"}$

- “One-way” (não inversível):

- $H^{-1}(\text{"DFCD BB EA ... CA99 2D17"}) \rightarrow ???$

- Sem colisões:

Função Hash Criptográfica

- Determinística:

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BB EA ... CA99 2D17"}$

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BB EA ... CA99 2D17"}$

- “One-way” (não inversível):

- $H^{-1}(\text{"DFCD BB EA ... CA99 2D17"}) \rightarrow ???$

- Sem colisões:

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BB EA ... CA99 2D17"}$

- $H(\text{~~?????~~}) \rightarrow \text{"DFCD BB EA ... CA99 2D17"}$

- Caótica:

Função Hash Criptográfica

- Determinística:

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BBEA ... CA99 2D17"}$

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BBEA ... CA99 2D17"}$

- “One-way” (não inversível):

- $H^{-1}(\text{"DFCD BBEA ... CA99 2D17"}) \rightarrow ???$

- Sem colisões:

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BBEA ... CA99 2D17"}$

- $H(\text{?????}) \rightarrow \text{"DFCD BBEA ... CA99 2D17"}$

- Caótica:

- $H(\text{"Fox"}) \rightarrow \text{"DFCD BBEA ... CA99 2D17"}$

- $H(\text{"Foy"}) \rightarrow \text{"1200 C78A ... EF1A D99B"}$

Mineração

Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



Mineração

Eu, Alice, descobri
5 *bitcoins*. **(PROVA)**

($t=678$)



- Custosa:

Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:
 - Verifica transações pendentes até o gêneseis

Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:
 - Verifica transações pendentes até o gênese
 - Realiza um trabalho com dificuldade artificial (e prova!)

Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:

- Verifica transações pendentes até o gêneseis
- Realiza um trabalho com dificuldade artificial (e prova!)

Eu, Carla, transfiro
1 bitcoin para Érica.

Eu, João, transfiro
1 bitcoin para José.

(t=1001)



Eu, André, transfiro
1 bitcoin para Ana.

Mineração

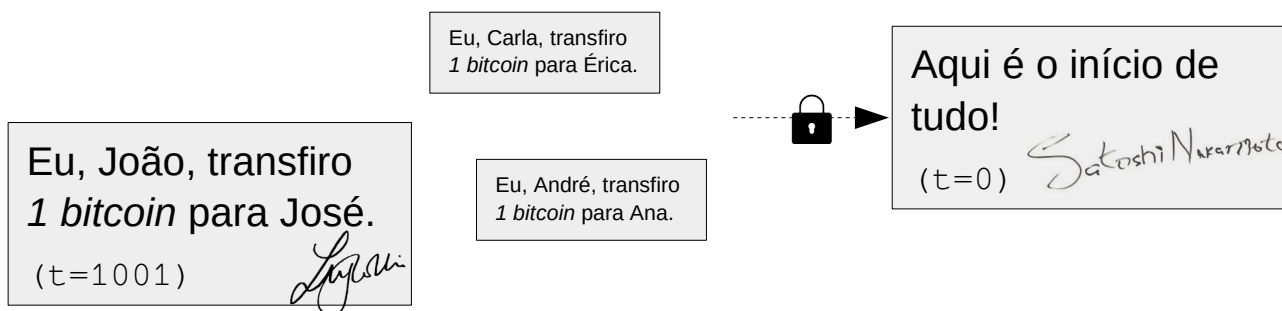
Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:

- Verifica transações pendentes até o gêneseis
- Realiza um trabalho com dificuldade artificial (e prova!)



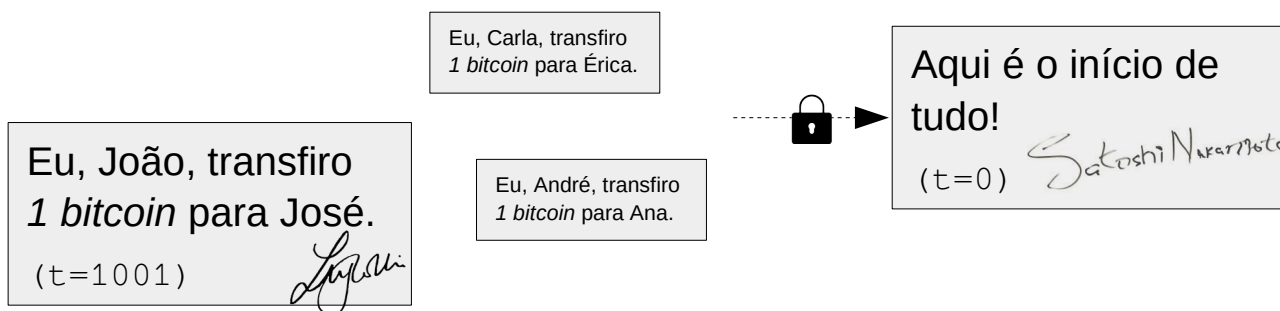
Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:
 - Verifica transações pendentes até o gêneseis
 - Realiza um trabalho com dificuldade artificial (e prova!)
- As transações são compiladas em um bloco.



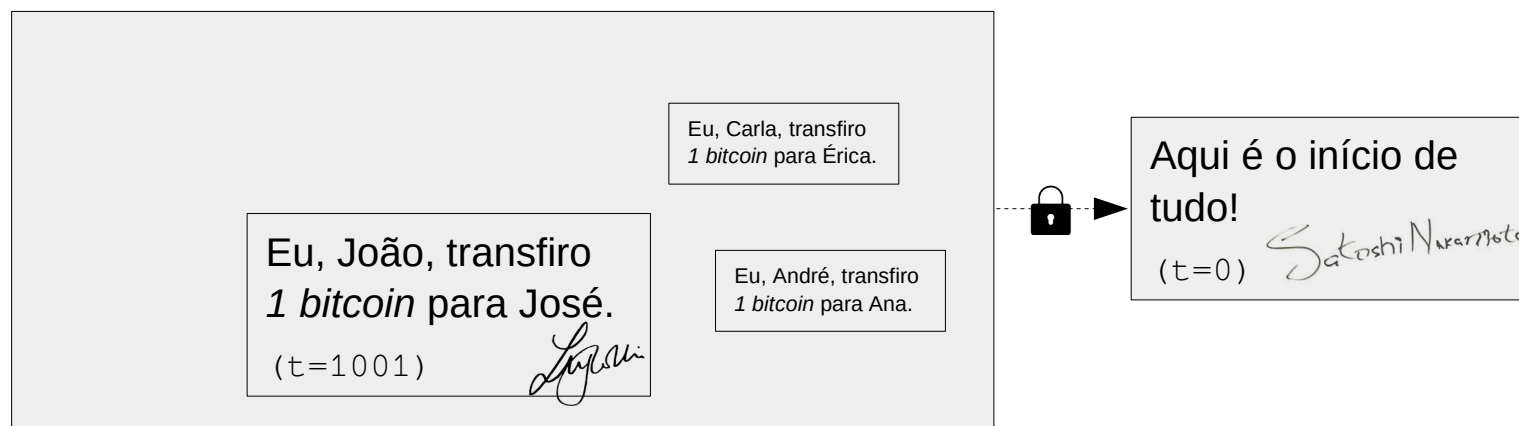
Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:
 - Verifica transações pendentes até o gêneseis
 - Realiza um trabalho com dificuldade artificial (e prova!)
- As transações são compiladas em um bloco.



Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



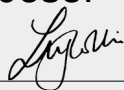
- Custosa:
 - Verifica transações pendentes até o gêneseis
 - Realiza um trabalho com dificuldade artificial (e prova!)
- As transações são compiladas em um bloco.

Eu, Alice, compilo essas transações e
ganho 5 bitcoins de recompensa.

Eu, Carla, transfiro
1 bitcoin para Érica.

Eu, João, transfiro
1 bitcoin para José.

(t=1001)

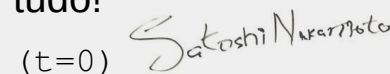


Eu, André, transfiro
1 bitcoin para Ana.



Aqui é o início de
tudo!

(t=0)



Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:
 - Verifica transações pendentes até o gêneseis
 - Realiza um trabalho com dificuldade artificial (e prova!)
- As transações são compiladas em um bloco.

Eu, Alice, compilo essas transações e
ganho 5 bitcoins de recompensa.

Aqui está a prova: N=4823.

(t=678)

Eu, João, transfiro
1 bitcoin para José.

(t=1001)



Eu, Carla, transfiro
1 bitcoin para Érica.

Eu, André, transfiro
1 bitcoin para Ana.



Aqui é o início de
tudo!

(t=0)



Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:
 - Verifica transações pendentes até o gêneseis
 - Realiza um trabalho com dificuldade artificial (e prova!)
- As transações são compiladas em um bloco.

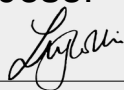
Eu, Alice, compilo essas transações e
ganho 5 bitcoins de recompensa.

Aqui está a prova: N=4823.

(t=678)

Eu, João, transfiro
1 bitcoin para José.

(t=1001)



Eu, Carla, transfiro
1 bitcoin para Érica.

Eu, André, transfiro
1 bitcoin para Ana.



Aqui é o início de
tudo!

(t=0)



Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:
 - Verifica transações pendentes até o gêneseis
 - Realiza um trabalho com dificuldade artificial (e prova!)
- As transações são compiladas em um bloco.

Eu, Alice, compilo essas transações e
ganho 5 bitcoins de recompensa.

Aqui está a prova: N=4823.

(t=678)

Eu, João, transfiro
1 bitcoin para José.

(t=1001)



Eu, Carla, transfiro
1 bitcoin para Érica.

Eu, André, transfiro
1 bitcoin para Ana.

Aqui é o início de
tudo!

(t=0)



$\text{HASH}(\text{alice} + t_1 + t_2 + t_3 + H(1000) + 4823) = 0x0000007F33D3E\dots$

Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:
 - Verifica transações pendentes até o gêneseis
 - Realiza um trabalho com dificuldade artificial (e prova!)
- As transações são compiladas em um bloco.

Eu, Alice, compilo essas transações e
ganho 5 bitcoins de recompensa.

Aqui está a prova: N=4823.

(t=678)



Eu, João, transfiro
1 bitcoin para José.

(t=1001)



Eu, Carla, transfiro
1 bitcoin para Érica.

Eu, André, transfiro
1 bitcoin para Ana.

Aqui é o início de
tudo!

(t=0)



$\text{HASH}(\text{alice} + t_1 + t_2 + t_3 + H(1000) + 4823) = 0x0000007F33D3E\dots$

Mineração

Eu, Alice, descobri
5 bitcoins. **(PROVA)**

(t=678)



- Custosa:
 - Verifica transações pendentes até o gêneseis
 - Realiza um trabalho com dificuldade artificial (e prova!)
- As transações são compiladas em um bloco.

Eu, Alice, compilo essas transações e
ganho 5 bitcoins de recompensa.

Aqui está a prova: N=4823.


(t=678)



Eu, João, transfiro
1 bitcoin para Maria,
1 bitcoin para José.

(t=1001)

(t=1001)



Eu, Carla, transfiro
1 bitcoin para Érica.

Eu, André, transfiro
1 bitcoin para Ana.



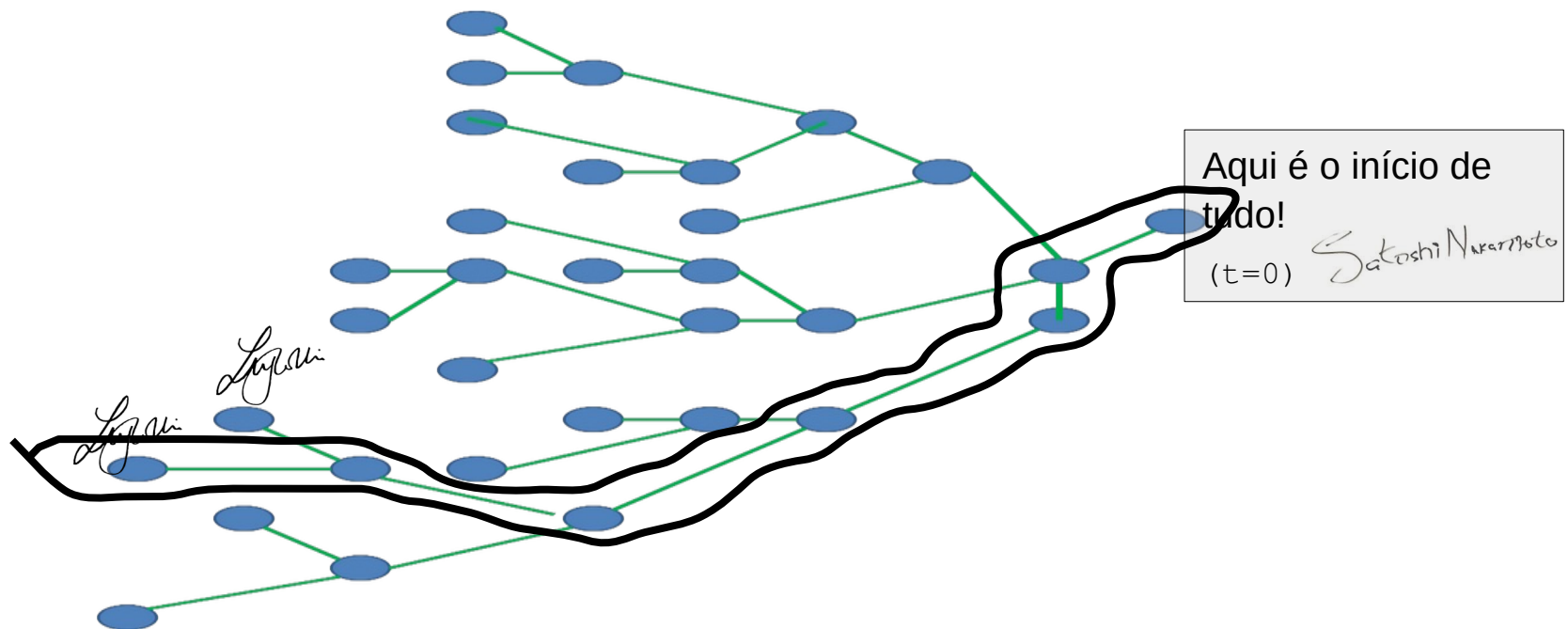
Aqui é o início de
tudo!

(t=0)



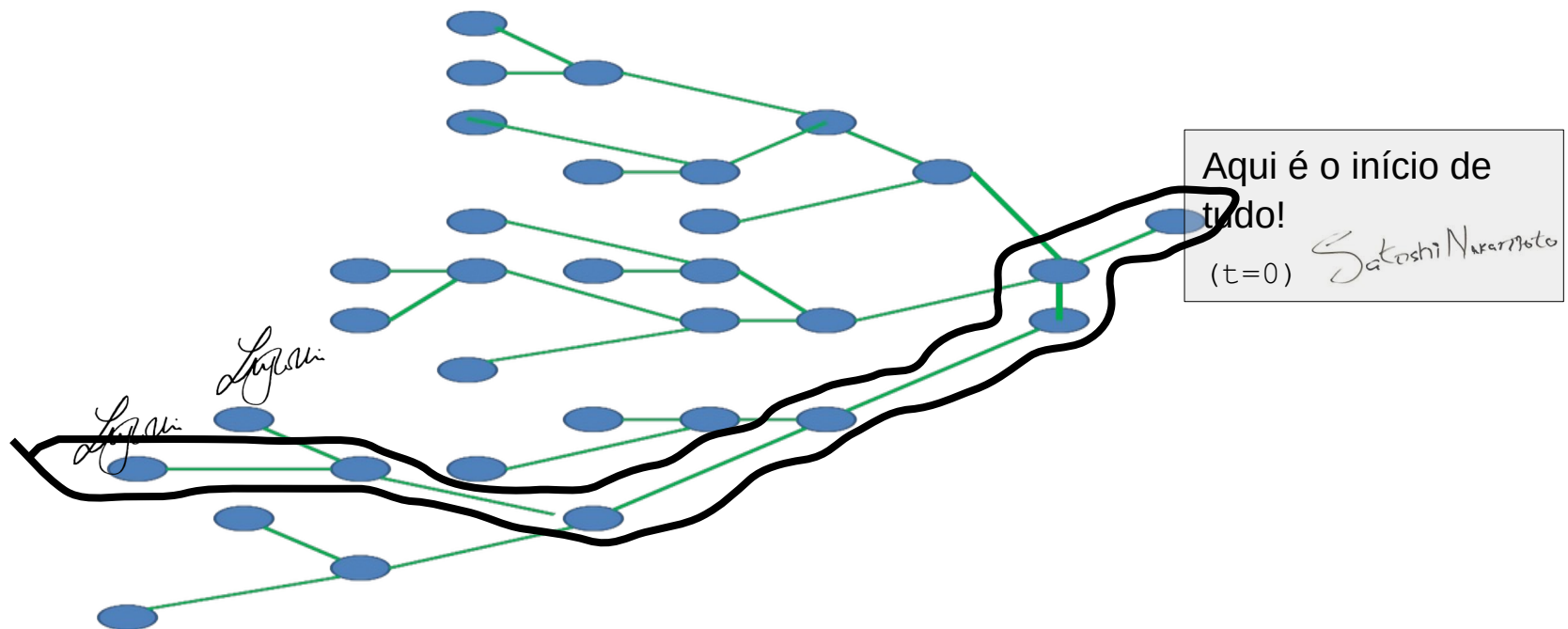
$\text{HASH}(\text{alice} + t1 + t2 + t3 + H(1000) + 4823) = 0x0000007F33D3E\dots$

Consenso



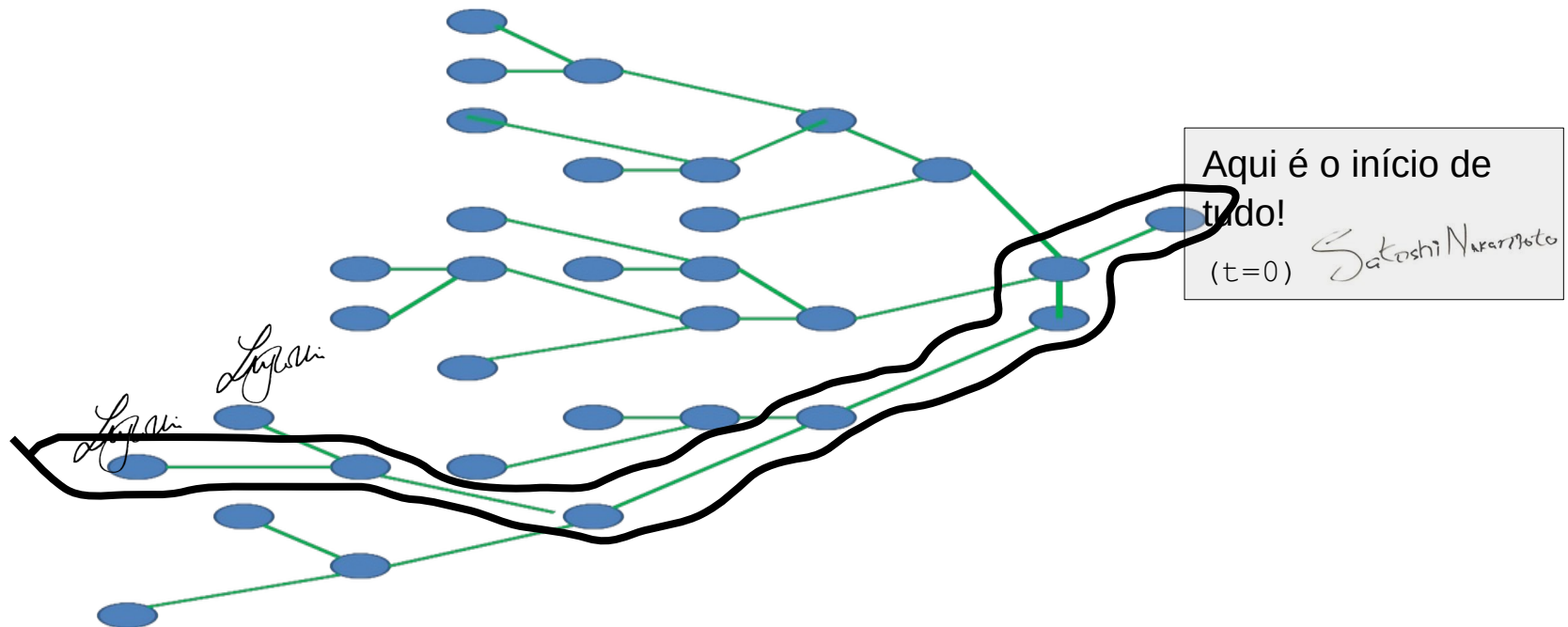
Consenso

- Somente um caminho deve ser válido



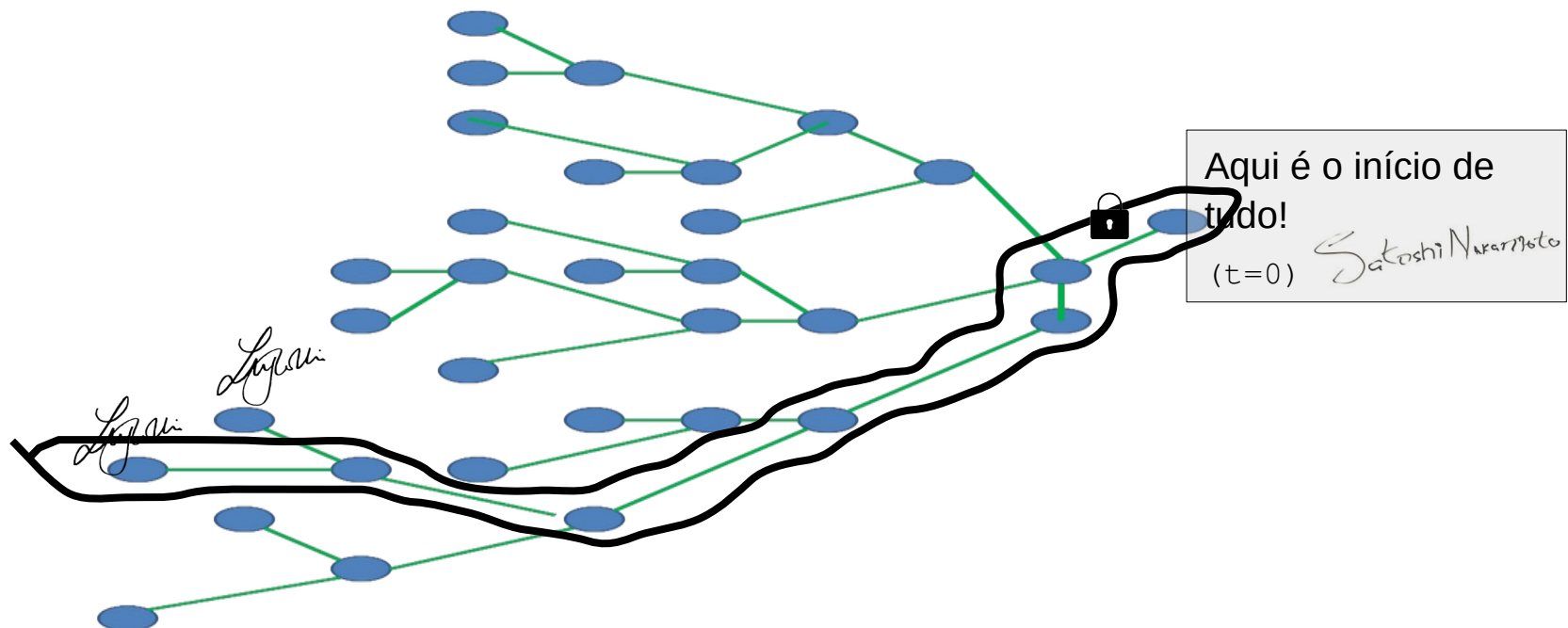
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado



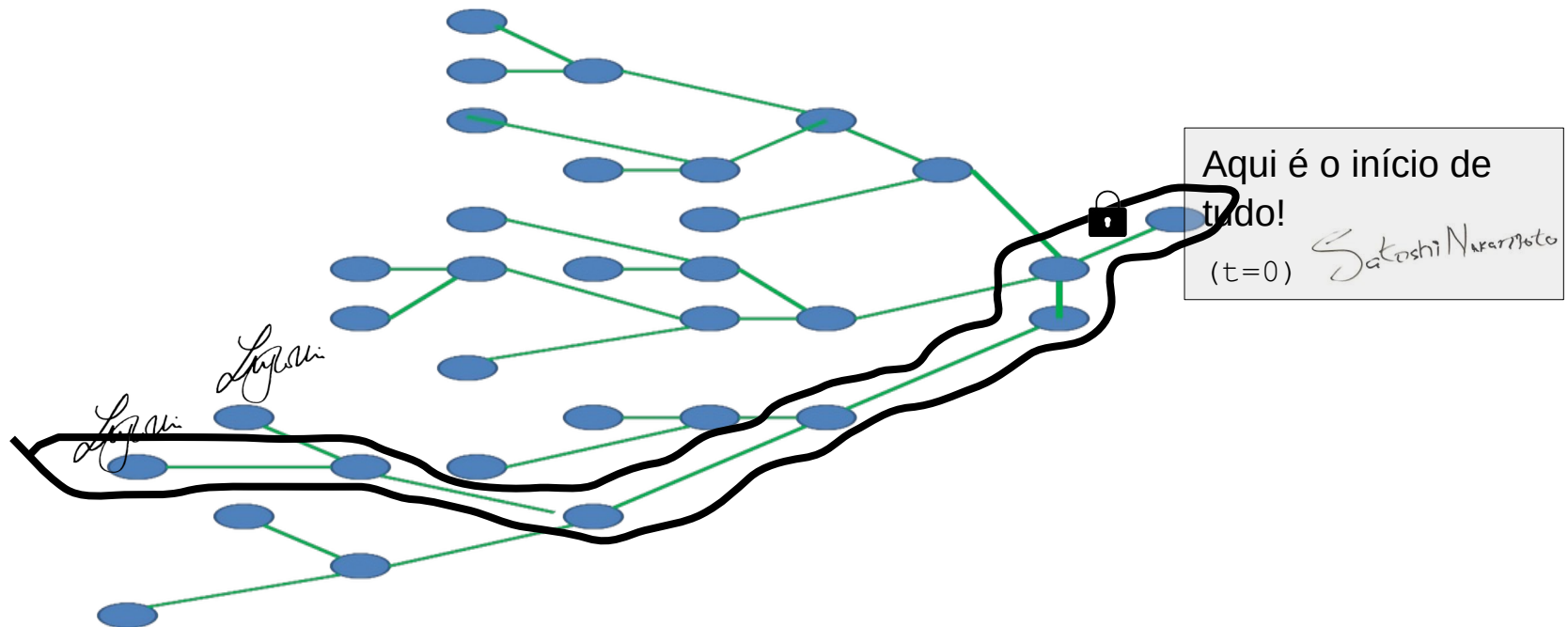
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado



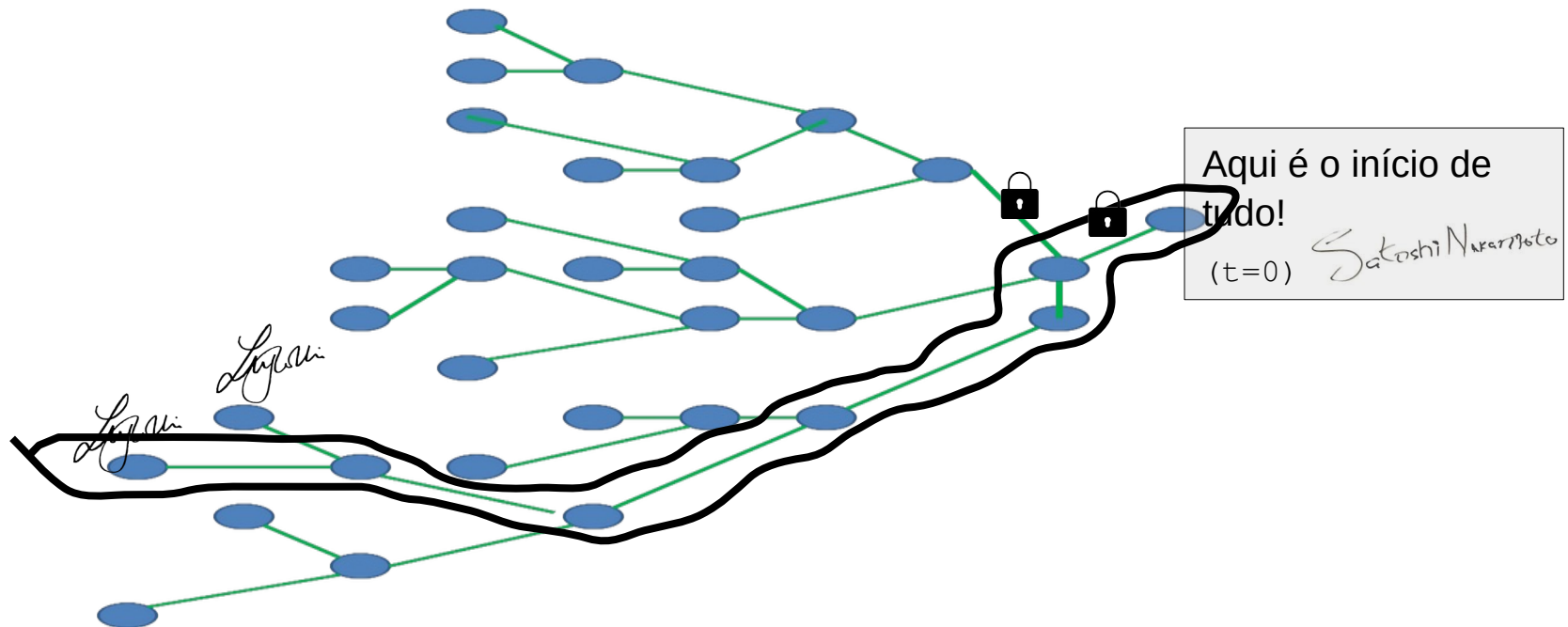
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado
- Trabalho artificial demora ~10 minutos



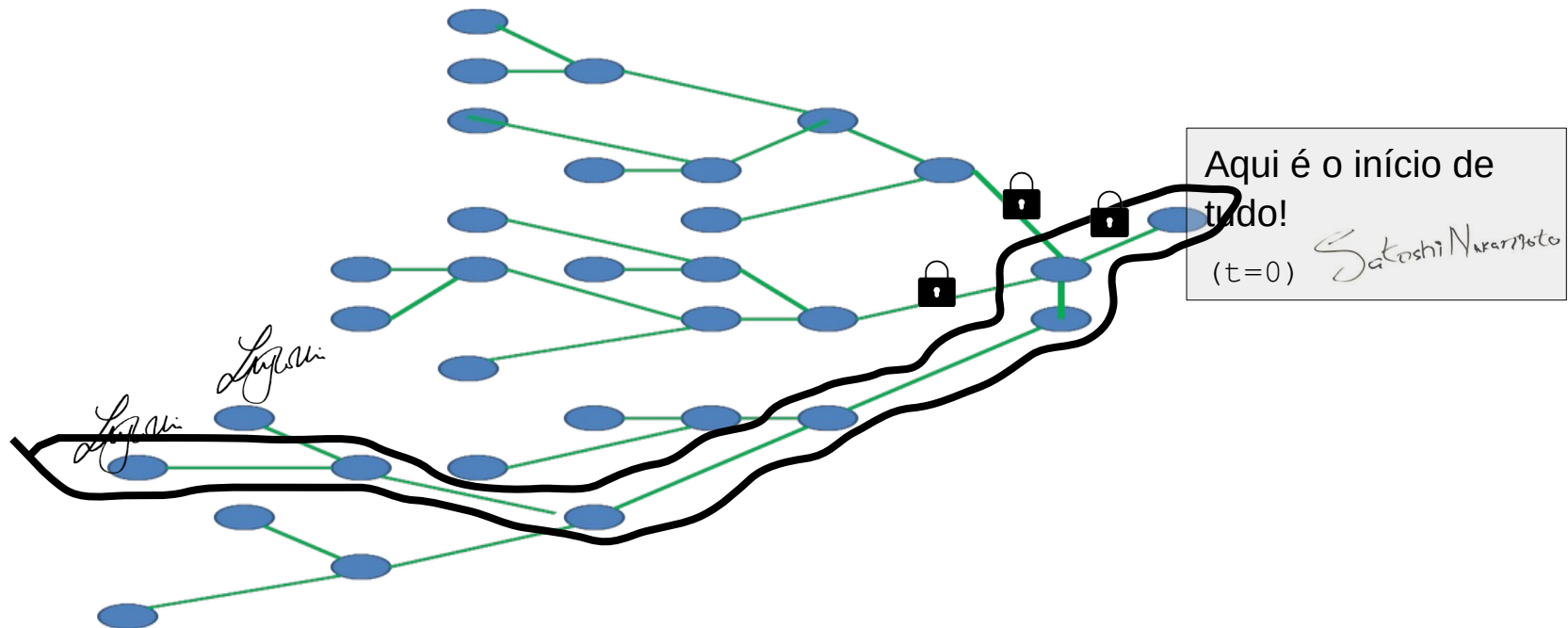
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado
- Trabalho artificial demora ~10 minutos



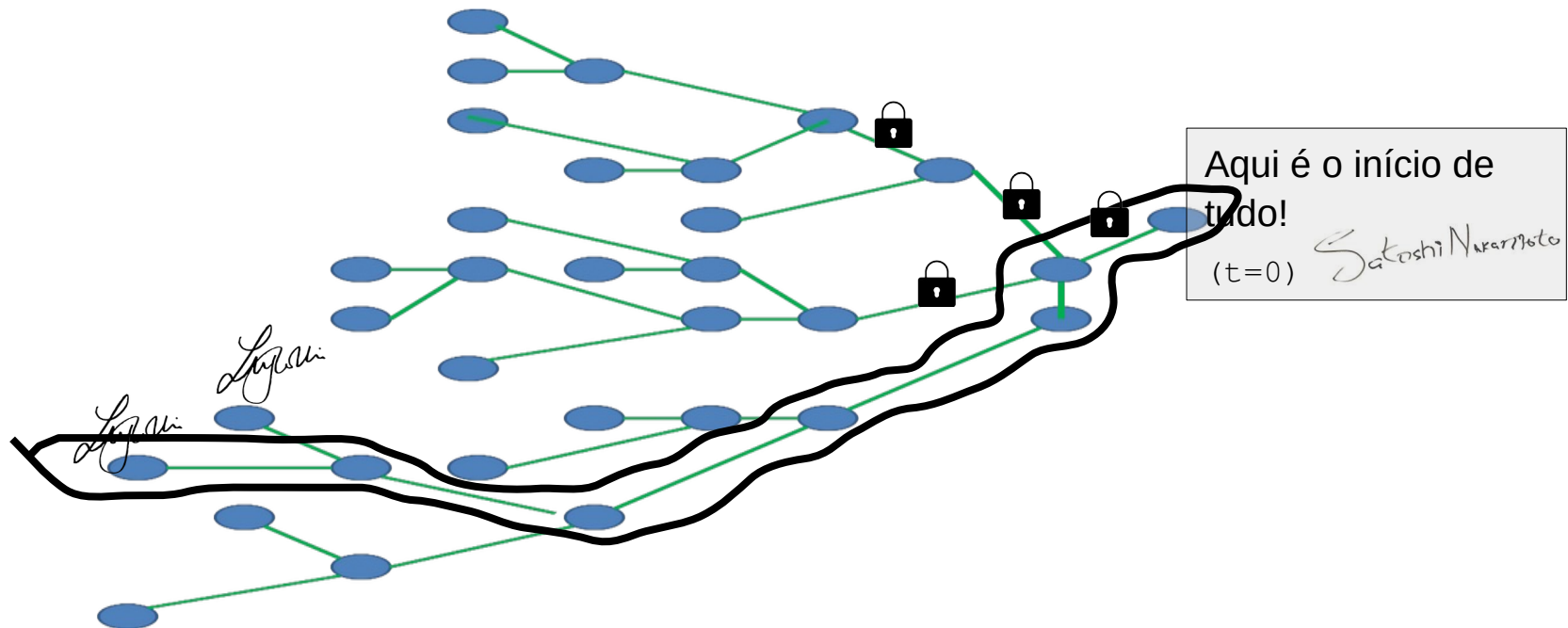
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado
- Trabalho artificial demora ~10 minutos



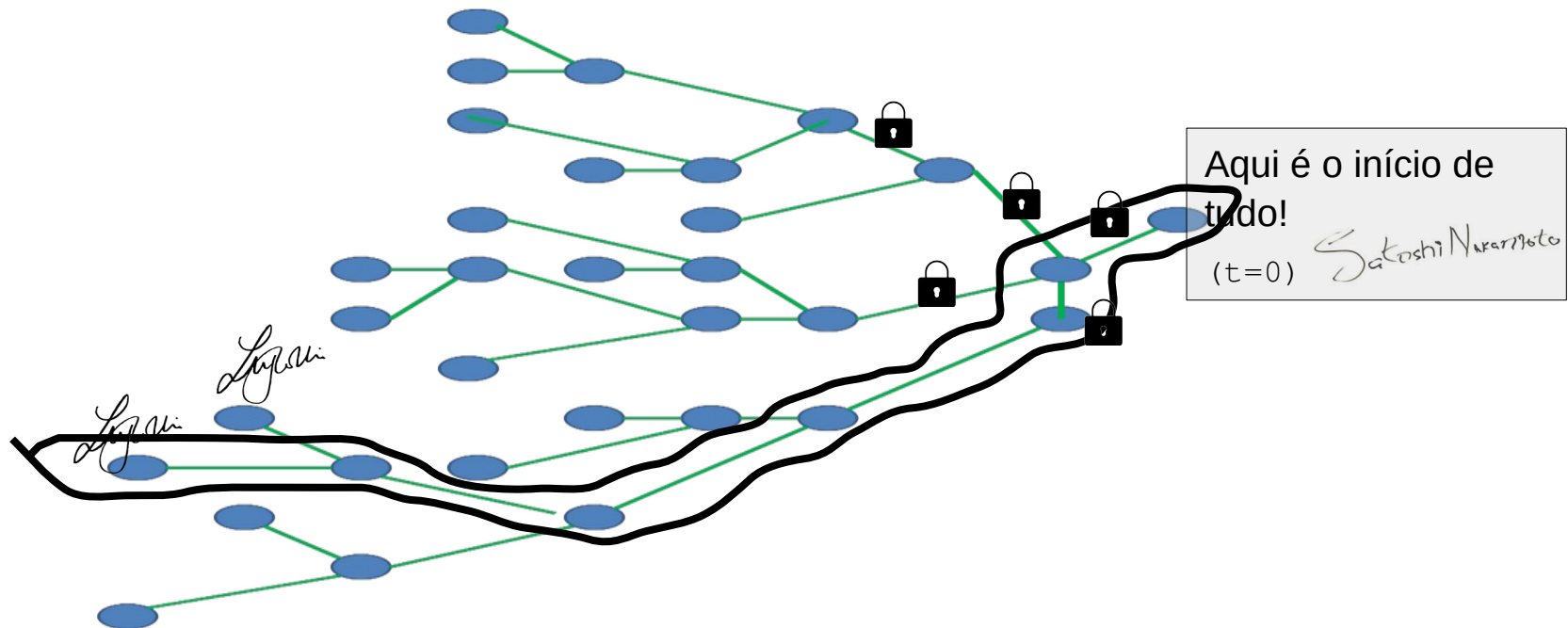
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado
- Trabalho artificial demora ~10 minutos



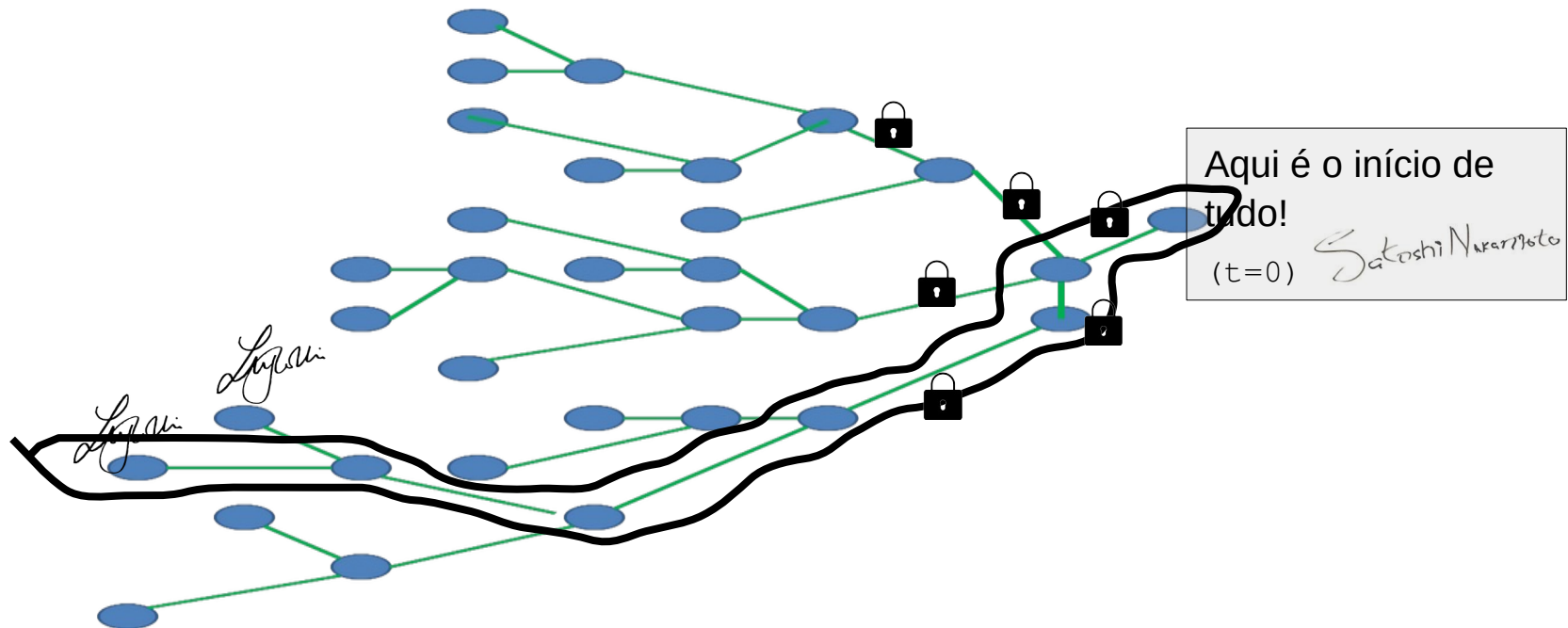
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado
- Trabalho artificial demora ~10 minutos



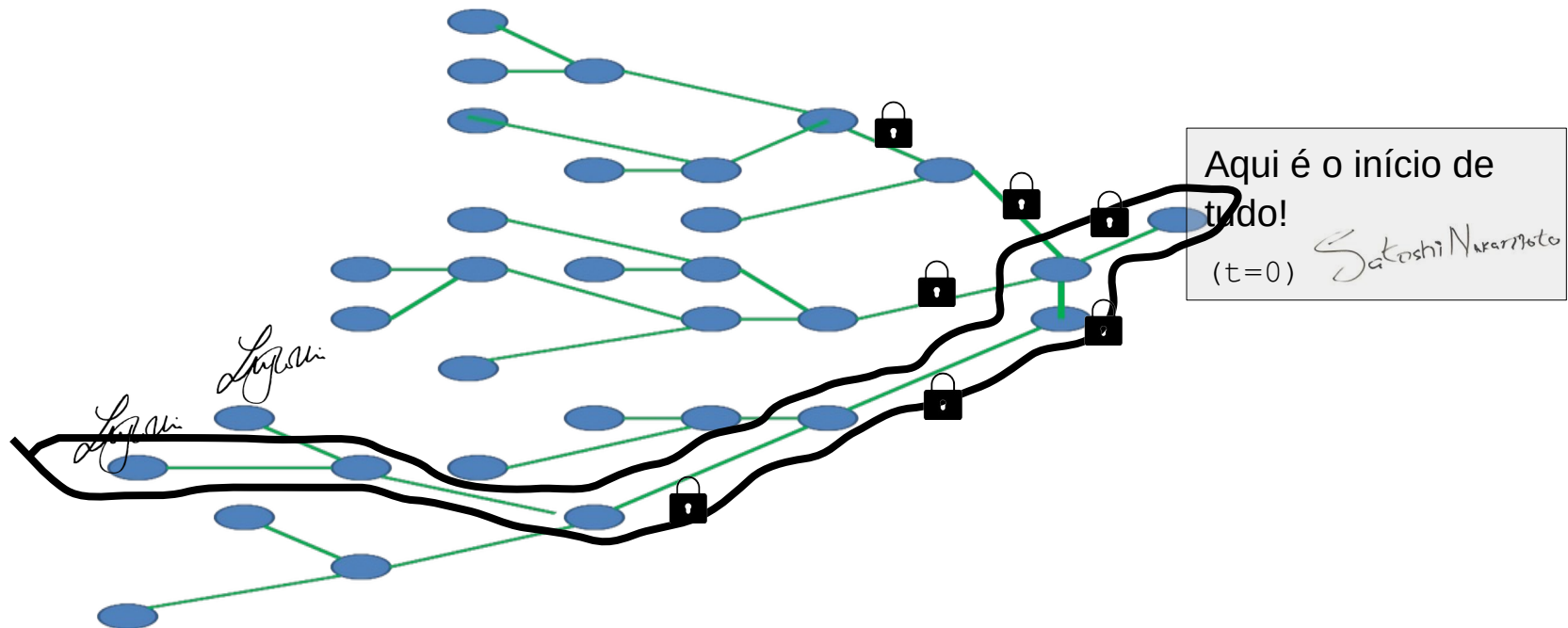
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado
- Trabalho artificial demora ~10 minutos



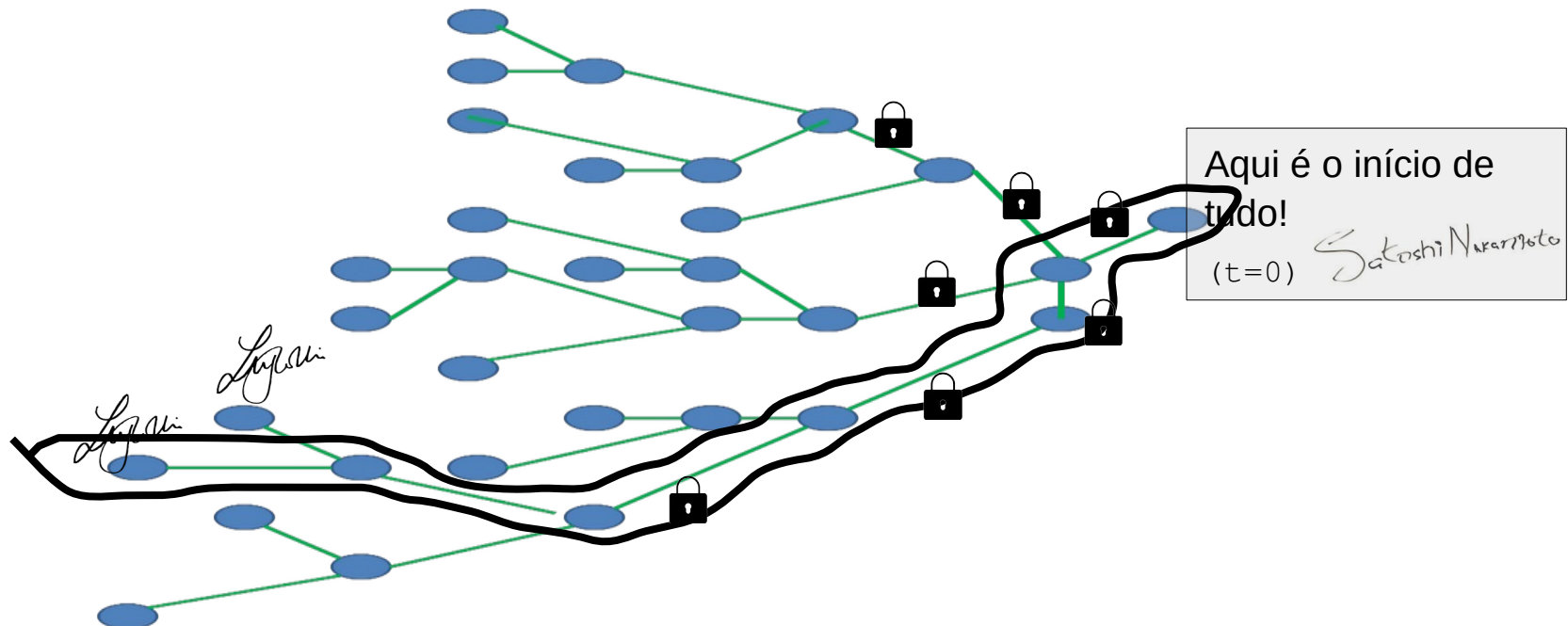
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado
- Trabalho artificial demora ~10 minutos



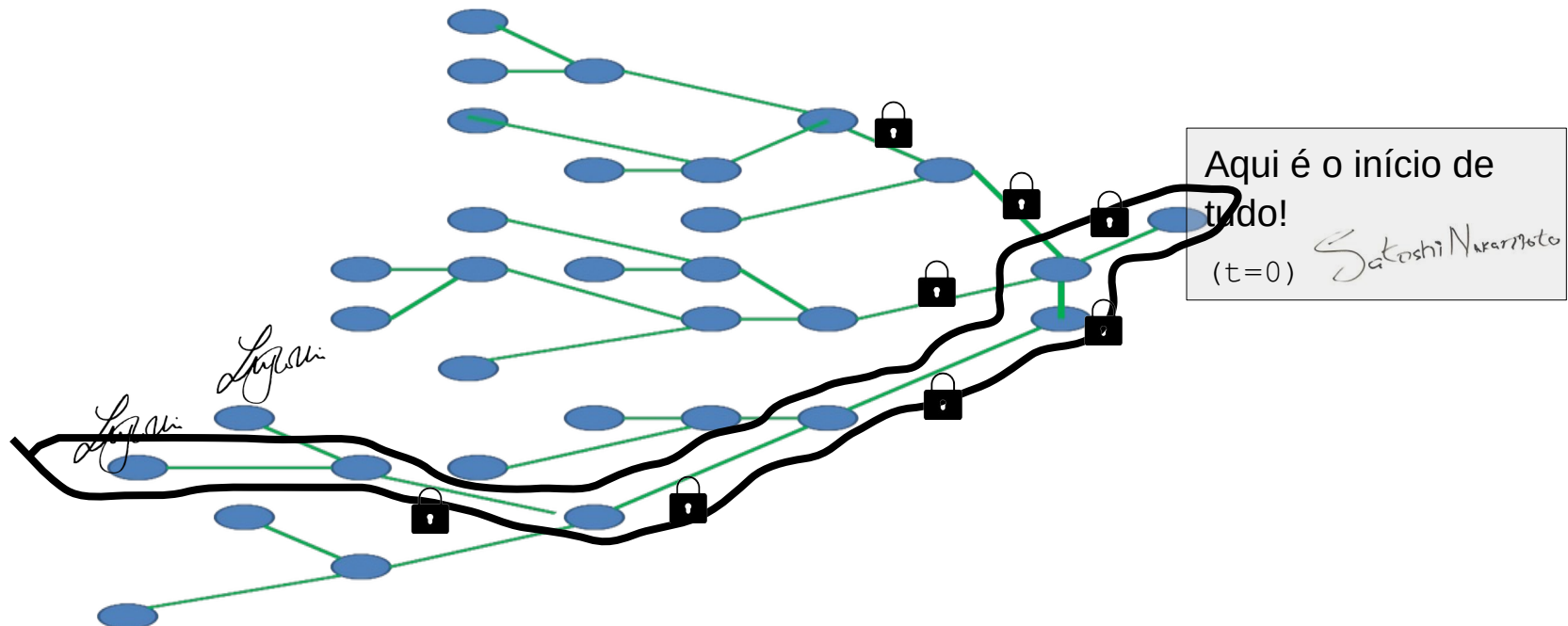
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado
- Trabalho artificial demora ~10 minutos
- Se 51% de CPU da rede for honesta, então a rede é segura



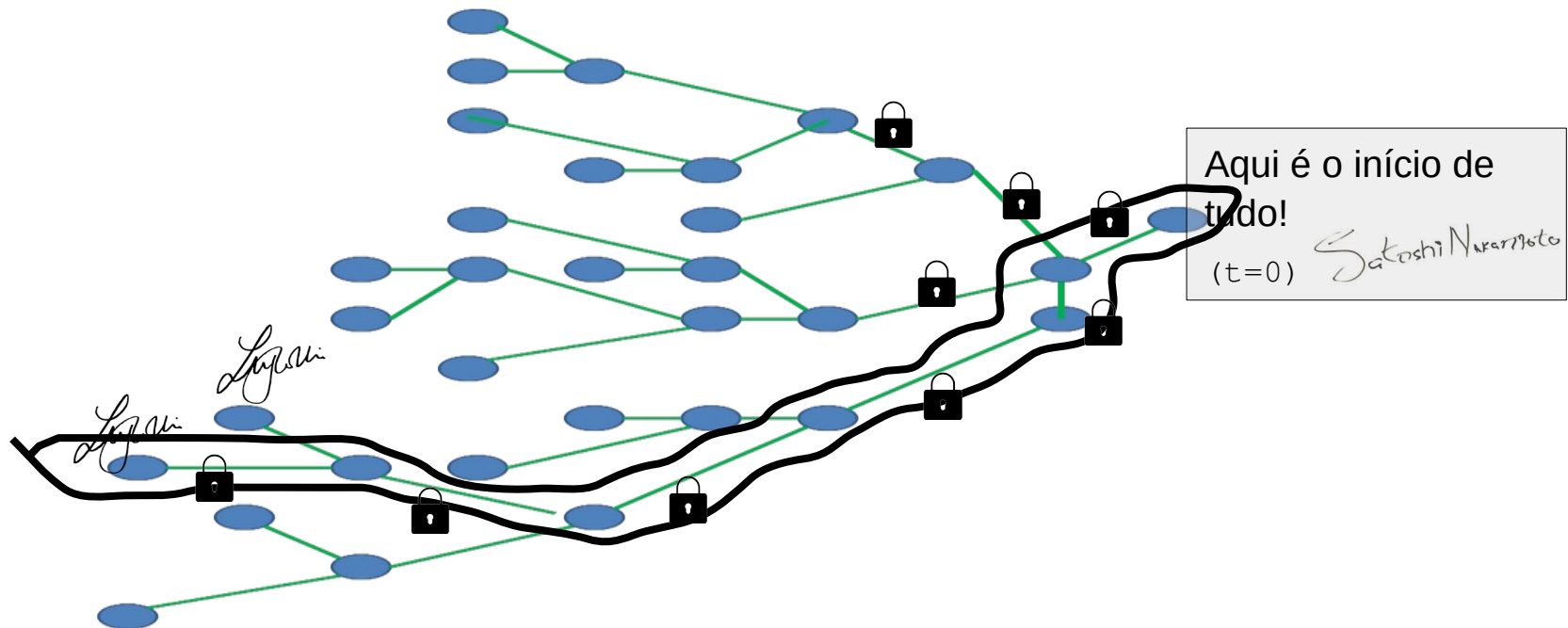
Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado
- Trabalho artificial demora ~10 minutos
- Se 51% de CPU da rede for honesta, então a rede é segura



Consenso

- Somente um caminho deve ser válido
- O caminho com mais trabalho associado
- Trabalho artificial demora ~10 minutos
- Se 51% de CPU da rede for honesta, então a rede é segura



Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network 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 longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU 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 as proof of what happened while they were gone.

Sistemas Peer-to-Peer

5. Bitcoin

Francisco Sant'Anna
francisco@ime.uerj.br

