

数字货币和区块链

- 中心化数字货币

山东大学网络空间安全学院

钱宸 2023年11月27日

数字货币

- 中心化数字货币定义与安全性
- 基于离散对数的数字货币
- 可传递数字货币定义
- 可传递数字货币构造

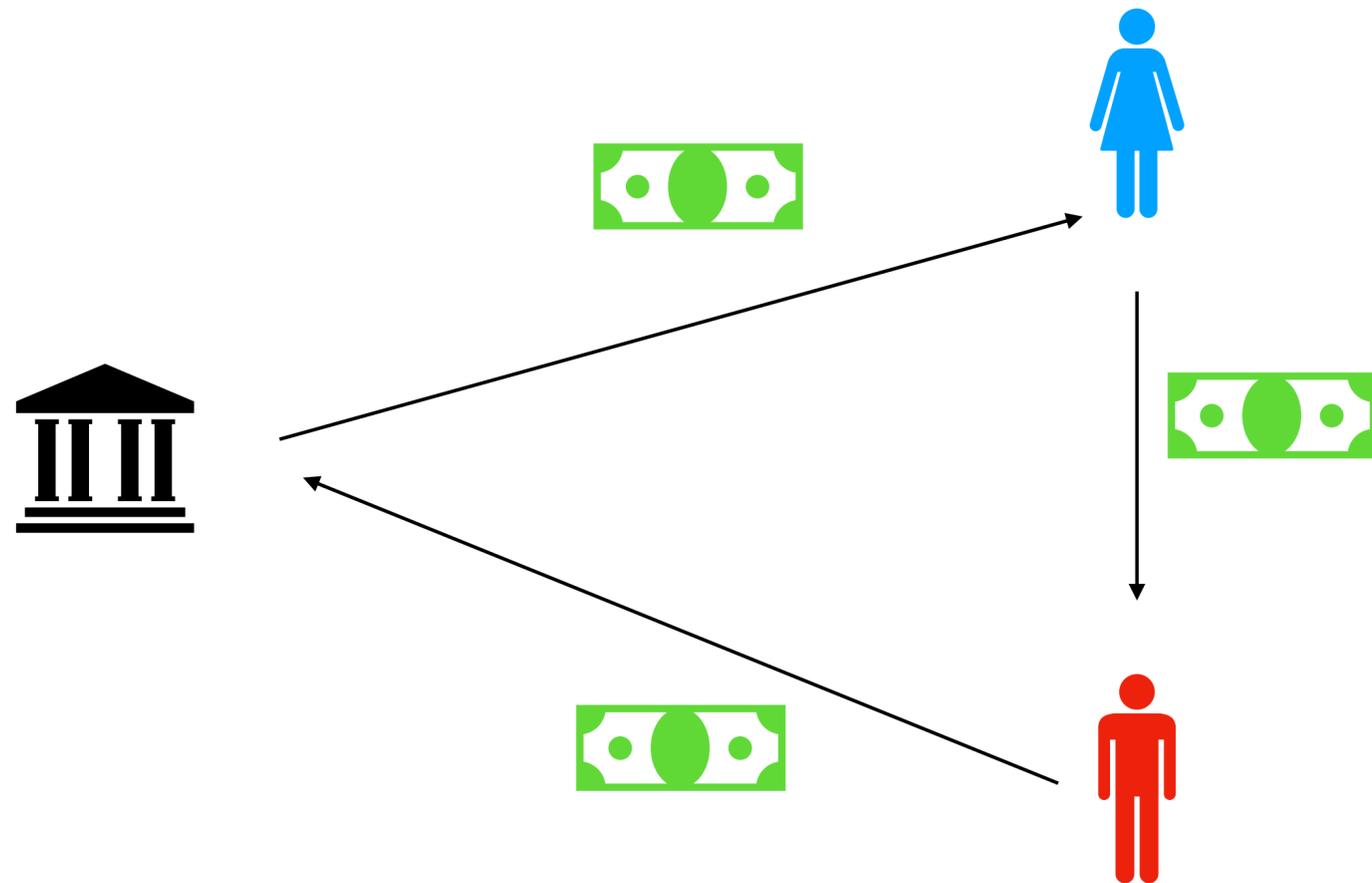
Chaum 数字货币

优点:

- 高效
- 匿名性
- 抗双支付攻击
- 可离线交易

缺点:

- 交易过程无法延续
- 数字货币无法分割



改进数字货币

- 我们这里考虑改进数字货币的可传递性：

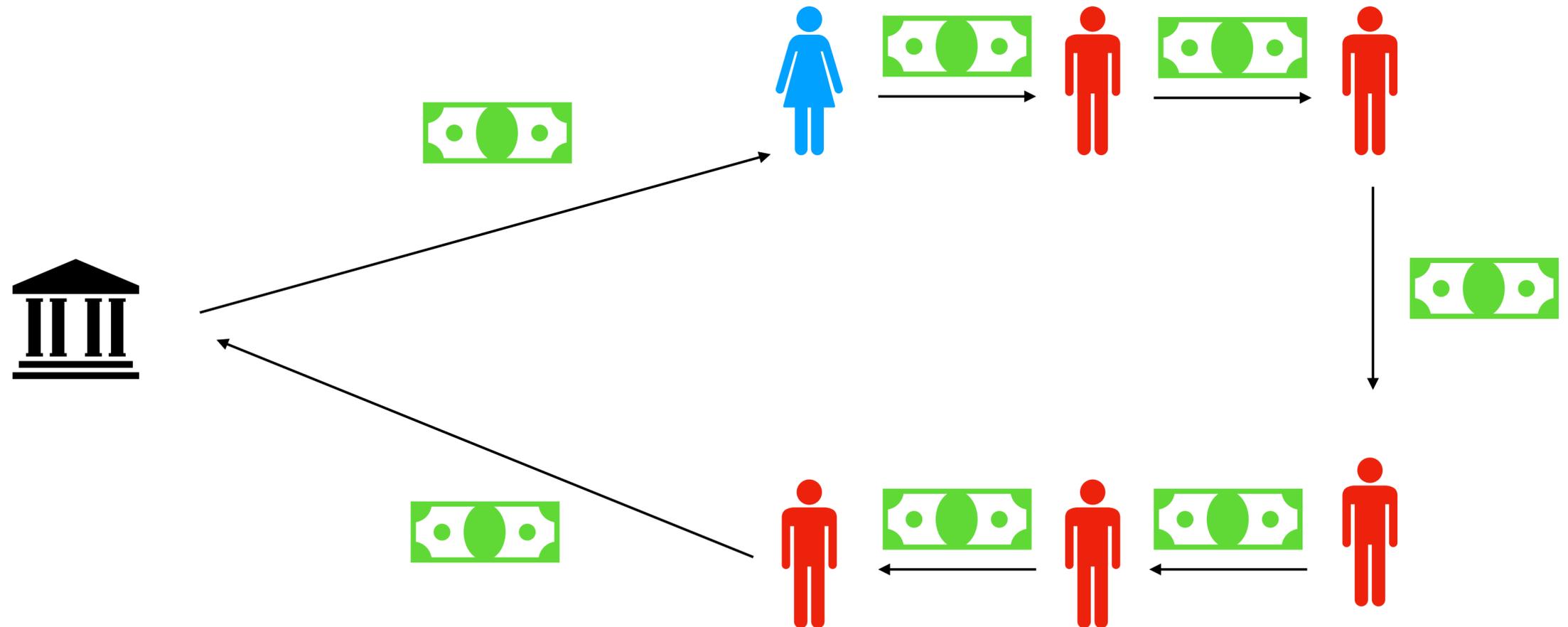
- 理想情况：

- 货币可传递

- 匿名性

- 抗双支付

- 不可抵赖



可传递货币 - 可靠性

- 对应Chaum数字货币的正确性
- Chaum数字货币：
 - 正常生成的货币，能够被银行承认。
- 可传递数字货币：
 - 正常生成的货币，能够被传递给下一个人，并最终被银行承认。

匿名性回顾

- 在Chaum数字货币中，匿名性
 - Bank \xrightarrow{C} Alice \xrightarrow{D} Bob \xrightarrow{E} Bank
 - 银行没办法将C和E联系起来。
 - 性质主要由盲签名算法来保证

可传递数字货币 - 匿名性

- 可传递货币的匿名性?
 - 不同级别的安全性定义
 - 货币匿名性
 - 货币透明性
 - 用户匿名性

可传递数字货币 - 货币匿名性

- Chaum数字货币匿名性的直接拓展

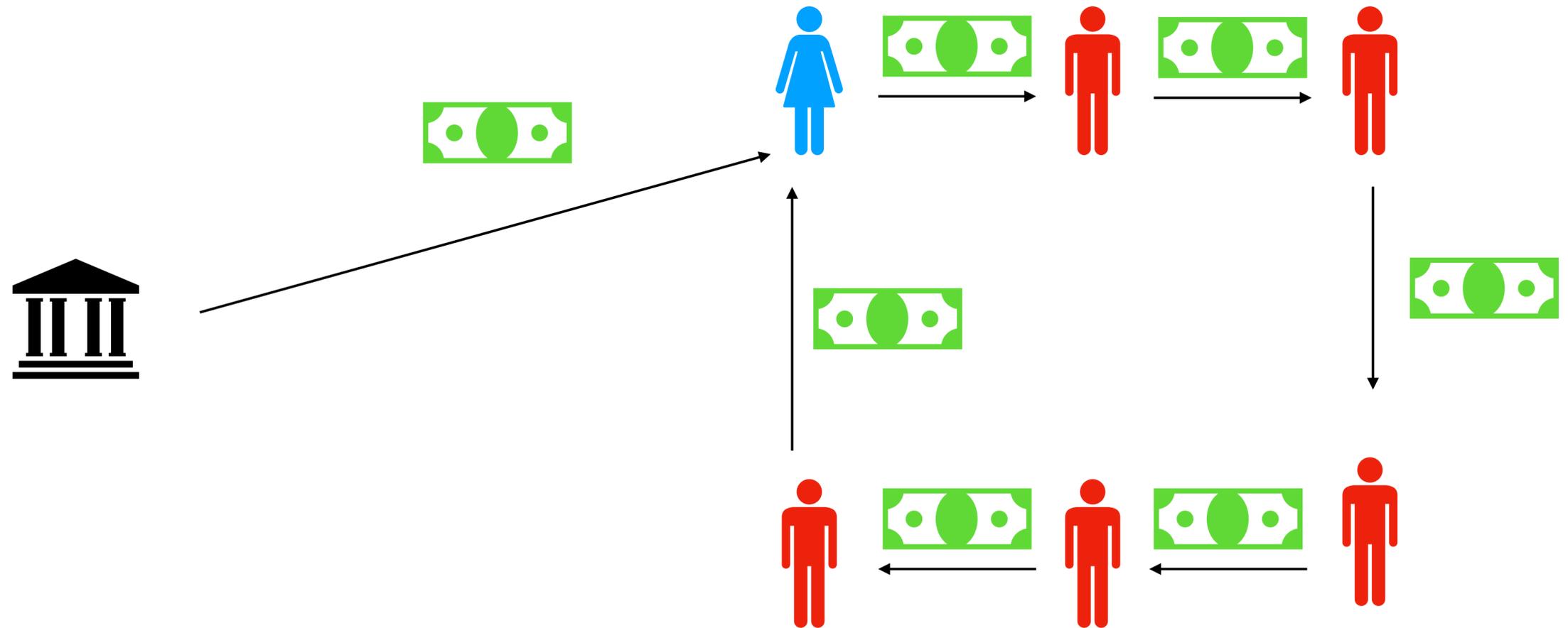
- Bank $\xrightarrow{C_1}$ A₁ $\xrightarrow{C_2}$ A₂ $\xrightarrow{C_3}$... $\xrightarrow{C_k}$ Bank

- 银行无法将 C_1 与 C_k 联系起来

- 然而改匿名性定义看上去并不是很足够

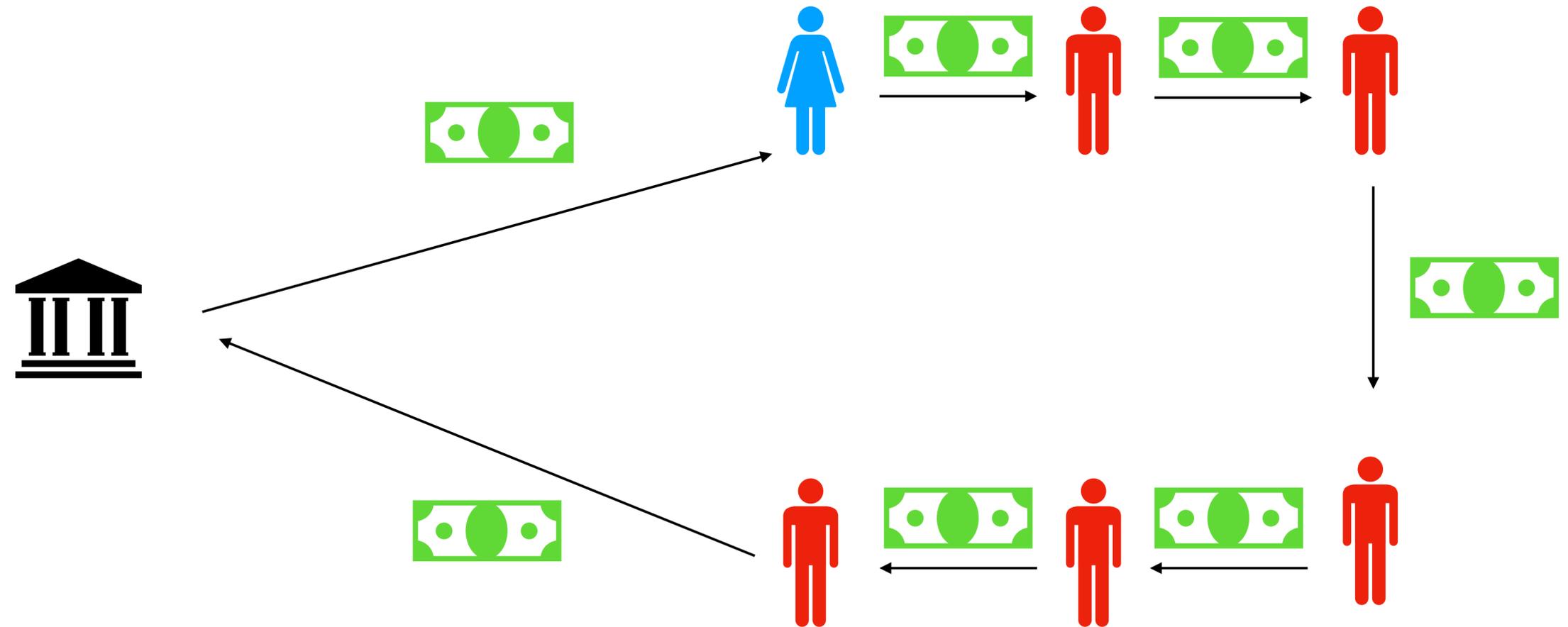
可传递数字货币 - 货币透明性

- 在传递的过程中每一个节点都具有匿名性



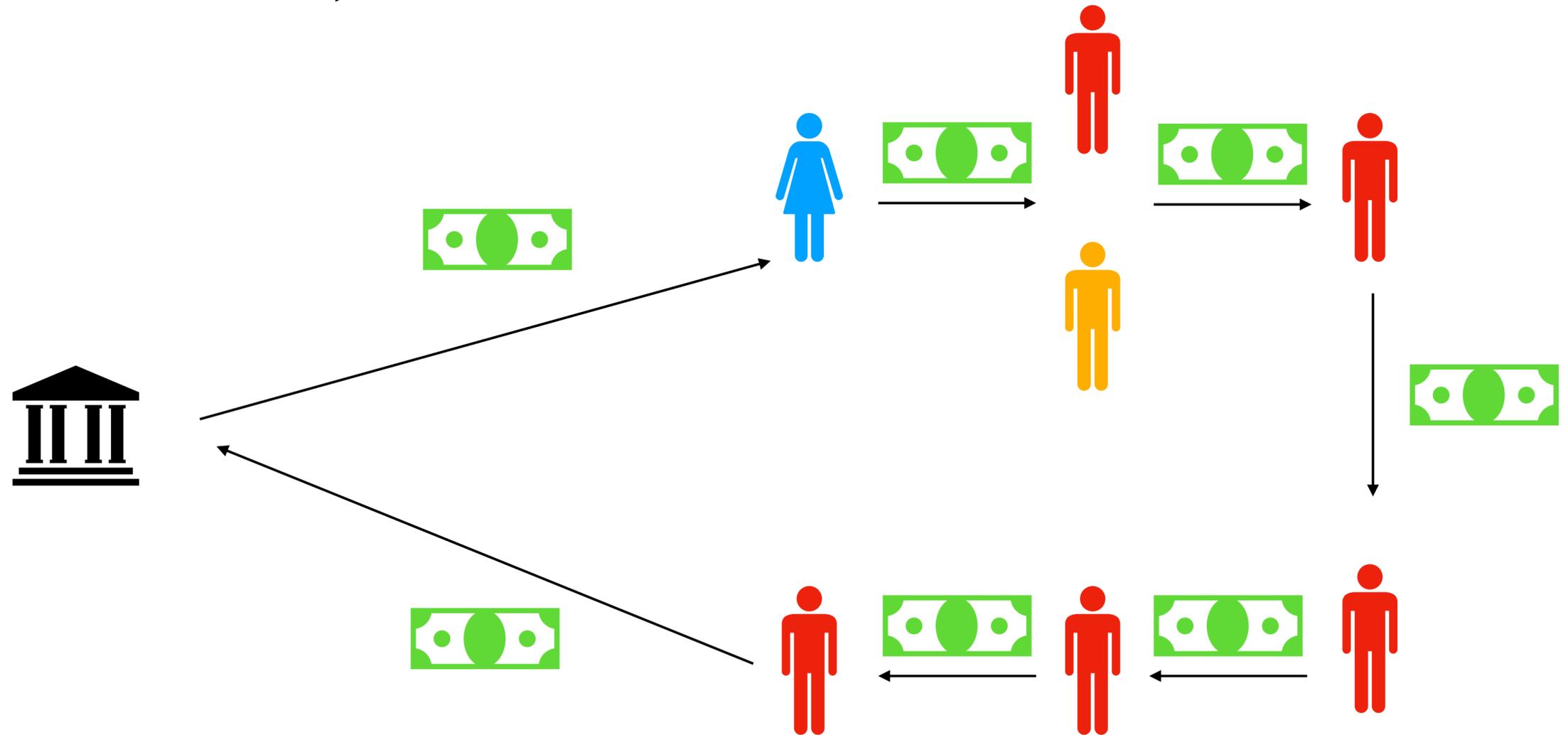
可传递数字货币 - 共谋攻击

- 然而，在有多方参与货币支付的时候，共谋攻击无法避免



可传递数字货币 - 用户匿名性

- 考虑上述共谋攻击存在情况下，定义用户匿名性



可传递数字货币 - 匿名性总结

- 完全照搬Chaum数字货币匿名性
 - 货币匿名性
- 每个用户节点都保证匿名性
 - 货币透明性
- 共谋攻击无法避免
 - 用户匿名性

可传递货币 - 抗双支付

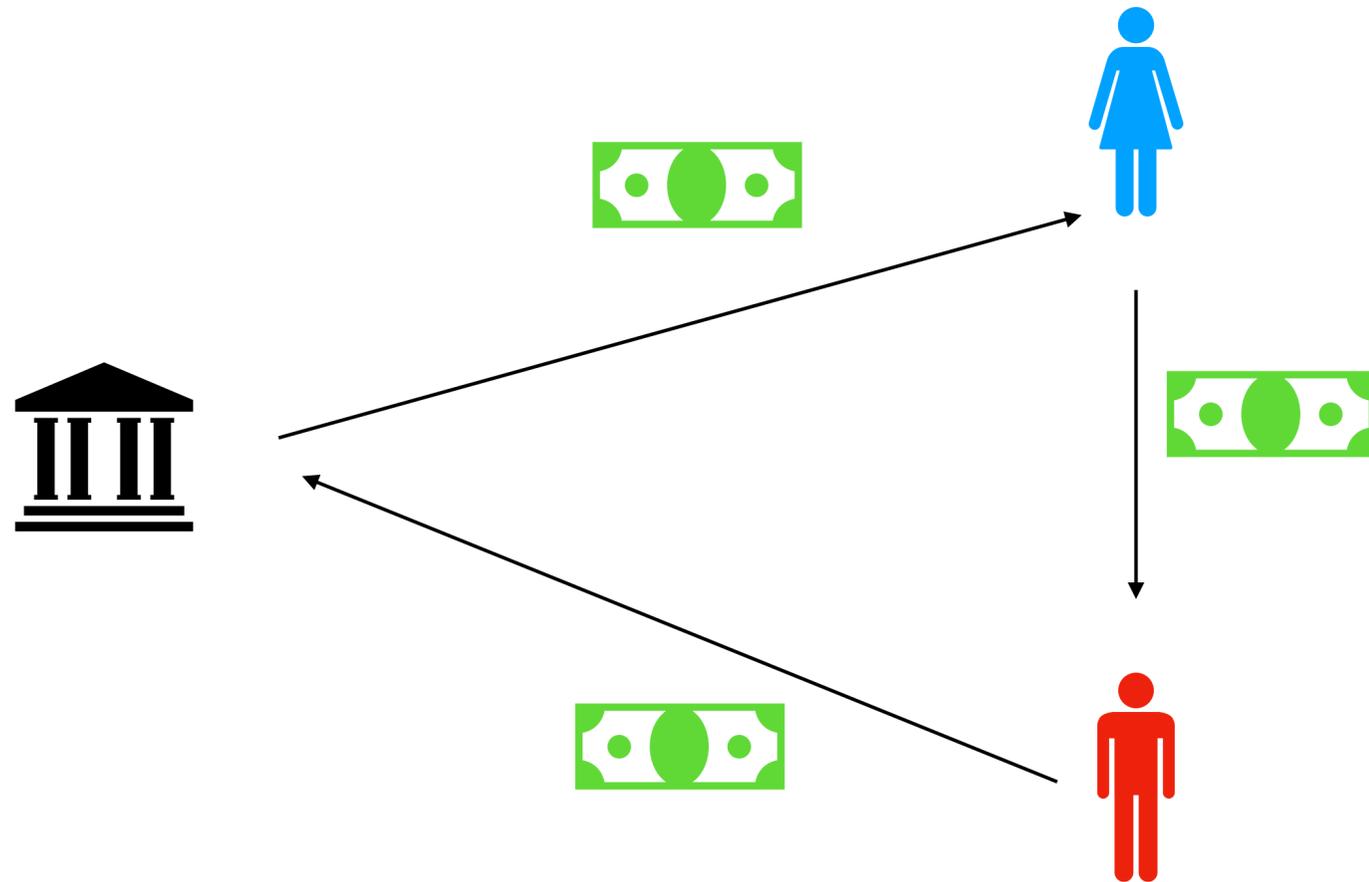
- 如果在货币支付过程中Alice发生了双支付
 - 则兑付的时候Alice的身份被银行识别

可传递货币 - 不可抵赖性

- 和Chaum数字货币类似
 - 假设银行判定Alice双支付攻击行为存在
 - Alice无法抵赖

可传递数字货币 - 第一次尝试

- 直接支付给第三方呢?
- Chaum数字货币:
 - $f_{SN}(pk_a, sk_a, pk_b), \pi_{a,b}, \sigma$
 - 问题?
 - 如果Bob进行了双支付→
 - 银行仍然追查到Alice



可传递数字货币 - 双支付攻击的简单拓展

- 仔细观察Chaum数字货币算法:
- $f_{SN}(pk_a, sk_a, pk_b), \pi_{a,b}, \sigma$
- $= pk_A^{pk_B} H(sk_A)$
- 双支付攻击:
- $\sigma, f_{SN}(pk_a, sk_a, pk_b), f_{SN}(pk_b, sk_b, pk_c), f_{SN}(pk_c, sk_c, pk_d)$
- 必须保证每一个序列号都是正确产生的, 但是又不能和银行进行交互

可传递数字货币 - 可变换签名

- Chase, Kohlweiss, Lysyanskaya, Meiklejohn CSF'14
 - Malleable Signatures: New Definitions and Delegatable Anonymous Credentials
- 提出了可变换签名：
 - $\text{Setup}(1^\lambda) \rightarrow (\text{vk}, \text{sk})$
 - $\text{Sig}(\text{sk}, m, T) \rightarrow \sigma$, 签名的同时规定一个明文变换。
 - $\text{Ver}(\text{vk}, \sigma, m) \rightarrow \{0, 1\}$
 - $\text{Trans}(T, \sigma) \rightarrow \sigma'$

可传递数字货币

- $\sigma, f_{SN}(pk_a, sk_a, pk_b), f_{SN}(pk_b, sk_b, pk_c), f_{SN}(pk_c, sk_c, pk_d)$
- 这里 σ 为一个可变换签名，其中变换T被定义为诚实的序列号生成
- 即，对于诚实产生的签名满足
 - $Ver(vk, \sigma, \{f_{SN}(pk_i, sk_i, pk_{i+1})\}_{i \in \{1, \dots, k\}}) = 1$

双支付攻击的匿名性考虑 - 零知识证明

- 现有数字货币:
- $\sigma, f_{SN}(pk_a, sk_a, pk_b), f_{SN}(pk_b, sk_b, pk_c), f_{SN}(pk_c, sk_c, pk_d)$
 - 货币匿名性 YES
 - 货币透明性 NO
- 如何解决?
 - 将每个序列号都加密, 并加上零知识证明 (Groth-Sahai EC'08)
 - $ct_\sigma, ct_1, ct_2, ct_3, \dots, ct_k, \pi$, 利用加密与零知识的可随机性

防止双支付攻击 - 公钥加密算法

- $ct_{\sigma}, ct_1, ct_2, ct_3, \dots, ct_k, \pi$
- 然而，银行并不知道序列号本身，所以无法进行双支付检测。
- 解决方法：
 - 用银行的公私钥对来进行加密

补充：双支付实现

- 这里零知识证明可随机化的条件→具有代数结构
 - 所以之前利用哈希函数的双支付无法实现
- Anonymous Transferable E-Cash
 - Baldimtsi, Melissa Chase, Georg Fuchsbauer, and Kohlweiss (PKC'15)
- 前期知识：双线性对
 - $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$, 其中 $g_1 \in \mathbb{G}_1, g_2 \in \mathbb{G}_2$
 - 满足
 - $e(g_1^x, g_2) = e(g_1, g_2^x)$
 - $e(g_1^0, g_2) = e(g_1, g_2^0) = g_T^0$

具有代数结构的序列号

- $(e, p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, g_T)$
- $pp = g_1, g_2, h_1, h_2, \bar{h}_1, \bar{h}_2$
- $f_{\text{SN}}(n_{i+1}, sk_{i+1}) = (N_{i+1}, M_{i+1})$, 其中 $N_{i+1} = g_1^{n_{i+1}}, M_{i+1} = g_2^{sk_{i+1} \cdot n_{i+1}}$
- $f_{\text{DS}}(ID_i, n_i, sk_i, (N_{i+1}, M_{i+1})) = (A_i, B_i, \bar{A}_i, \bar{B}_i)$
 - 其中 $A_i = N_{i+1}^{ID_i} h_1^{n_i}$ $B_i = M_{i+1}^{ID_i} h_2^{n_i}$
 $\bar{A}_i = N_{i+1}^{sk_i} \bar{h}_1^{n_i}$ $\bar{B}_i = M_{i+1}^{sk_i} \bar{h}_2^{n_i}$

具有代数结构的序列号

- $f_{\text{DS}}(ID_i, n_i, sk_i, (N_{i+1}, M_{i+1})) = (A_i, B_i, \bar{A}_i, \bar{B}_i)$

- 其中
$$A_i = N_{i+1}^{ID_i} h_1^{n_i} \quad B_i = M_{i+1}^{ID_i} h_2^{n_i}$$
$$\bar{A}_i = N_{i+1}^{sk_i} \bar{h}_1^{n_i} \quad \bar{B}_i = M_{i+1}^{sk_i} \bar{h}_2^{n_i}$$

- 抗碰撞
- 抗双支付