How is Your Contract Executed in Ethereum?

@juinc

Video Link

Outline

- Overview
- Section 6: Transaction Execution
- Section 7: Contract Creation
- Section 8: Message Call
- Section 9: Execution Model
- Conclusion

Overview

- What is Blockchain?
    - Destributed State Machine (See more in my previous talk)
- What is the difference between Bitcoin and Ethereum?
    - State Modeling (UTXO v.s. World State)
    - Rewarding Mechanism(uncle blocks are introduced)
    - Programmability (smart contract)
- What is Ethereum Virtual Machine (EVM)?
    - A machine that performs instructions from compiled bytecode (smart contract) which is stored in world state

EVM Overview

- Stack-based Machine
- Memory Model: Word-addressed Byte Array, Volatile
- Storage Model: Word-addressed Word Array, Non-volatile
- Quasi-Turing-Complete Machine, will raise out-of-gas(oos) exception if the paid gas is insufficient

Section 6: Transaction Execution

Overview

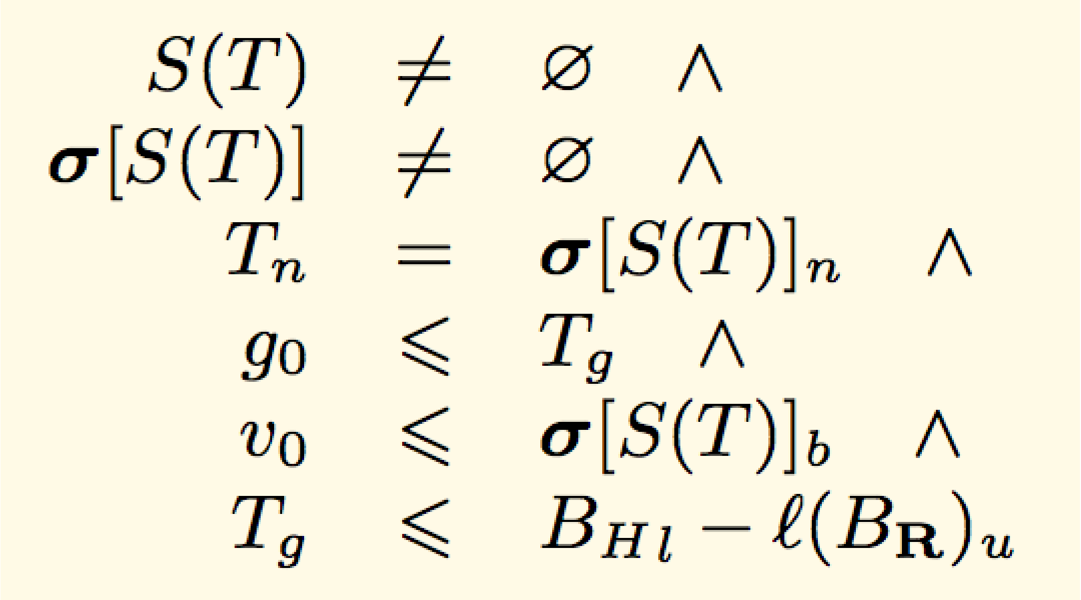
$$\boldsymbol{\sigma}' = \Upsilon(\boldsymbol{\sigma}, T)$$

There are totally 5 phases of state during transaction execution:

- Initial State => (Substrate gas limit x gas price and increment nonce)
- Checkpoint State => (Process message call or contract creation)
- Post-execution Provisional State => (Apply refunds and rewards)
- Pre-final State => (Delete destructed accounts)
- Final State

Section 6: Transaction Execution

Validation



$$\begin{aligned}
S(T) &\neq \varnothing \quad \wedge \\
\boldsymbol{\sigma}[S(T)] &\neq \varnothing \quad \wedge \\
T_n &= \boldsymbol{\sigma}[S(T)]_n \quad \wedge \\
g_0 &\leqslant T_g \quad \wedge \\
v_0 &\leqslant \boldsymbol{\sigma}[S(T)]_b \quad \wedge \\
T_g &\leqslant B_{H\,l} - \ell(B_{\mathbf{R}})_u
\end{aligned}$$

Section 6: Transaction Execution

Validation (Cont.)

Where

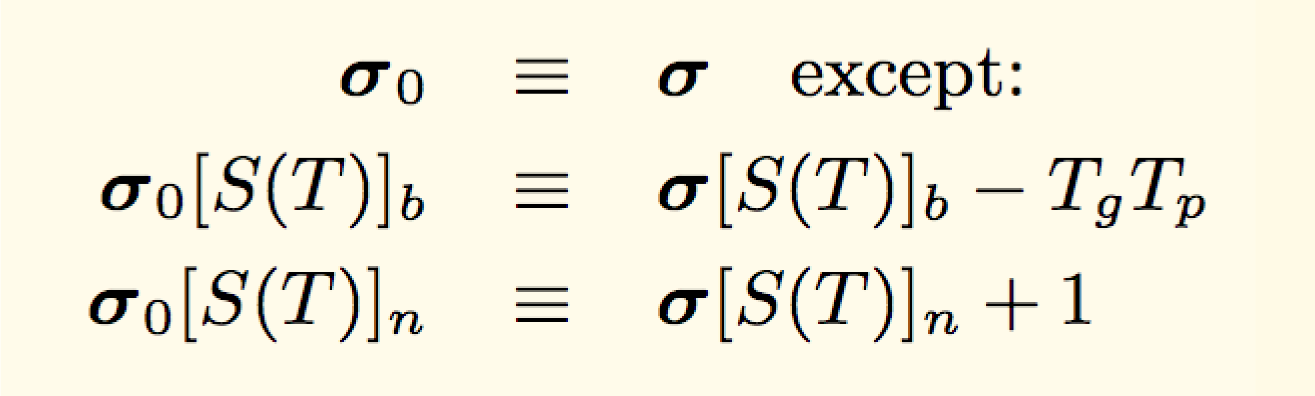
$$g_0 \equiv \sum_{i \in T_\mathbf{i}, T_\mathbf{d}} \begin{cases} G_{txdatazero} & \text{if} \quad i = 0 \\ G_{txdatanonzero} & \text{otherwise} \end{cases} \\ + \begin{cases} G_{\text{txcreate}} & \text{if} \quad T_t = \varnothing \\ 0 & \text{otherwise} \end{cases} \\ + G_{transaction}$$

$$v_0 \equiv T_g T_p + T_v$$

Section 6: Transaction Execution

(1) Substrate Gas Limit x Gas Price and Increment Nonce

Initial State => Checkpoint State



$$\begin{aligned}
\boldsymbol{\sigma}_0 &\equiv \boldsymbol{\sigma} \quad \text{except:} \\
\boldsymbol{\sigma}_0[S(T)]_b &\equiv \boldsymbol{\sigma}[S(T)]_b - T_g T_p \\
\boldsymbol{\sigma}_0[S(T)]_n &\equiv \boldsymbol{\sigma}[S(T)]_n + 1
\end{aligned}$$

Section 6: Transaction Execution

(2) Process Message Call or Contract Creation

Checkpoint State => Post-execution Provisional State

$$(\boldsymbol{\sigma}_P, g', A) \equiv \begin{cases} \Lambda(\boldsymbol{\sigma}_0, S(T), T_o, \\ \qquad g, T_p, T_v, T_{\mathbf{i}}, 0) & \text{if} \quad T_t = \varnothing \\ \Theta_3(\boldsymbol{\sigma}_0, S(T), T_o, \\ \qquad T_t, T_t, g, T_p, T_v, T_v, T_{\mathbf{d}}, 0) & \text{otherwise} \end{cases}$$

Where

$$g \equiv T_g - g_0$$

*These functions will be explained more in section 7 & 8

Section 6: Transaction Execution

(3) Apply Refunds and Rewards

Post-execution Provisional State => Pre-final State

$$\begin{aligned} \boldsymbol{\sigma}^* &\equiv \boldsymbol{\sigma}_P \quad \text{except} \\ \boldsymbol{\sigma}^*[S(T)]_b &\equiv \boldsymbol{\sigma}_P[S(T)]_b + g^* T_p \\ \boldsymbol{\sigma}^*[m]_b &\equiv \boldsymbol{\sigma}_P[m]_b + (T_g - g^*) T_p \\ m &\equiv B_{H c} \end{aligned}$$

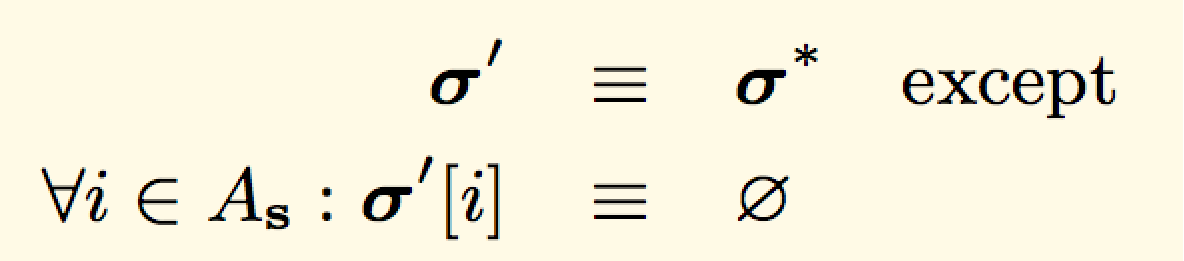
Where

$$g^* \equiv g' + \min\{ \left\lfloor \frac{T_g - g'}{2} \right\rfloor, A_r \}$$

Section 6: Transaction Execution

(4) Delete Destructed Accounts

Pre-final State => Final State



$$
\begin{aligned}
\boldsymbol{\sigma}' &\equiv \boldsymbol{\sigma}^* \quad \text{except} \\
\forall i \in A_{\mathbf{s}} : \boldsymbol{\sigma}'[i] &\equiv \varnothing
\end{aligned}
$$

Section 7: Contract Creation

Overview

$$(\boldsymbol{\sigma}', g', A) \equiv \Lambda(\boldsymbol{\sigma}, s, o, g, p, v, \mathbf{i}, e)$$

Where

$$A \equiv (A_{\mathbf{s}}, A_{\mathbf{l}}, A_r)$$

Section 7: Contract Creation

(1) Create a New Account

$$\boldsymbol{\sigma}^* \equiv \boldsymbol{\sigma} \quad \text{except:}$$

$$\begin{aligned} \boldsymbol{\sigma}^*[a] &\equiv \big(0, v + v', \mathtt{TRIE}(\varnothing), \mathtt{KEC}\big(()\big)\big) \\ \boldsymbol{\sigma}^*[s]_b &\equiv \boldsymbol{\sigma}[s]_b - v \end{aligned}$$

Section 7: Contract Creation

(1) Create a New Account (Cont.)

Where

$$a \equiv \mathcal{B}_{96..255}\Big(\mathtt{KEC}\Big(\mathtt{RLP}\big(\;(s, \boldsymbol{\sigma}[s]_n - 1)\;\big)\Big)\Big)$$

$$v' \equiv \begin{cases} 0 & \text{if} \quad \boldsymbol{\sigma}[a] = \varnothing \\ \boldsymbol{\sigma}[a]_b & \text{otherwise} \end{cases}$$

Section 7: Contract Creation

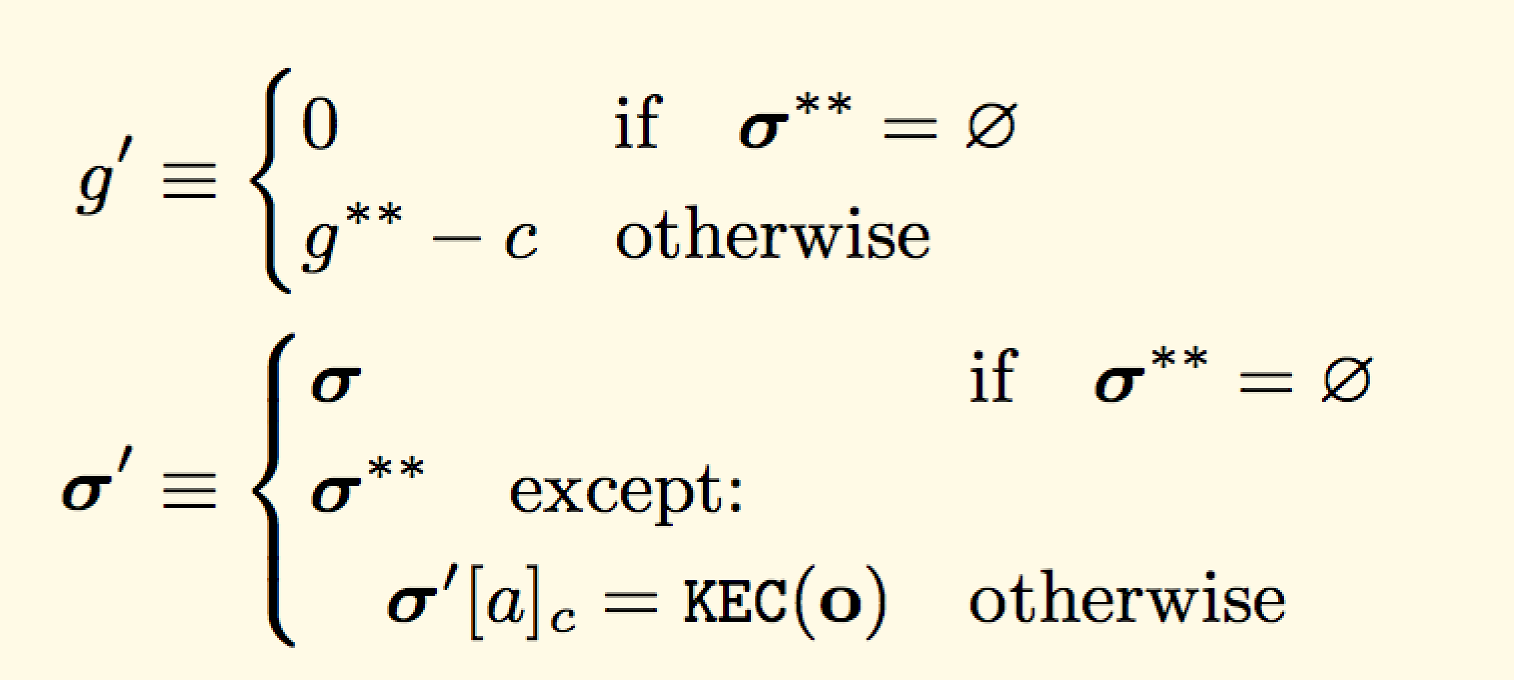
(2) Initialize

$$(\boldsymbol{\sigma}^{**}, g^{**}, A, \mathbf{o}) \equiv \Xi(\boldsymbol{\sigma}^{*}, g, I)$$

*The function will be expalined more in section 9

Section 7: Contract Creation

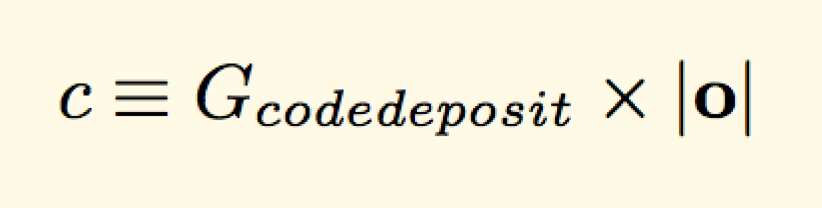
(3) Determine Final State



$$g' \equiv \begin{cases} 0 & \text{if} \quad \boldsymbol{\sigma}^{**} = \varnothing \\ g^{**} - c & \text{otherwise} \end{cases}$$

$$\boldsymbol{\sigma}' \equiv \begin{cases} \boldsymbol{\sigma} & \text{if} \quad \boldsymbol{\sigma}^{**} = \varnothing \\ \boldsymbol{\sigma}^{**} \quad \text{except:} & \\ \quad \boldsymbol{\sigma}'[a]_c = \mathtt{KEC}(\mathbf{o}) & \text{otherwise} \end{cases}$$

Where



$$c \equiv G_{codedeposit} \times |\mathbf{o}|$$

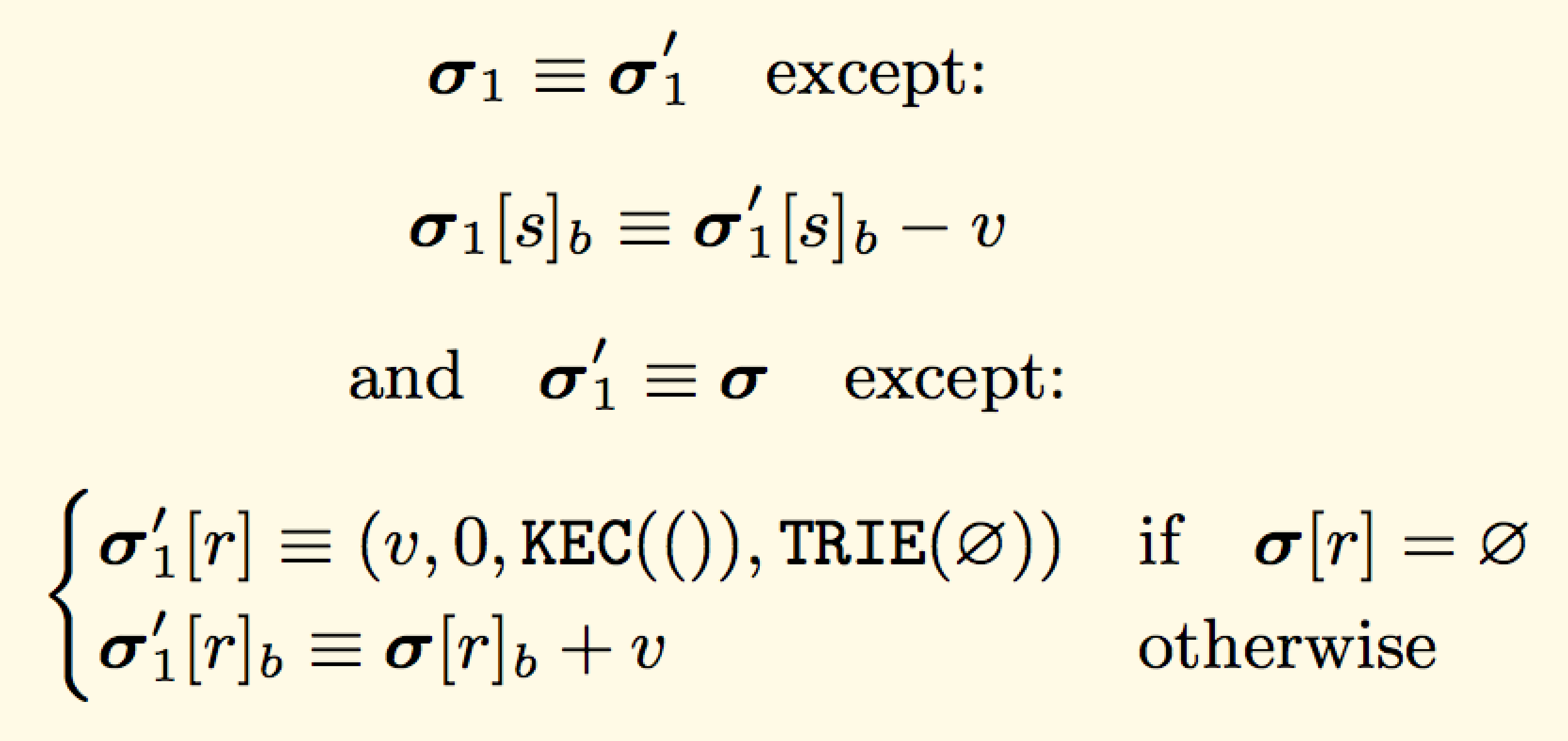
Section 8: Message Call

Overview

$$(\boldsymbol{\sigma}', g', A, \mathbf{o}) \equiv \Theta(\boldsymbol{\sigma}, s, o, r, c, g, p, v, \tilde{v}, \mathbf{d}, e)$$

Section 8: Message Call

(1) Transfer Value



$$\boldsymbol{\sigma}_1 \equiv \boldsymbol{\sigma}'_1 \quad \text{except:}$$

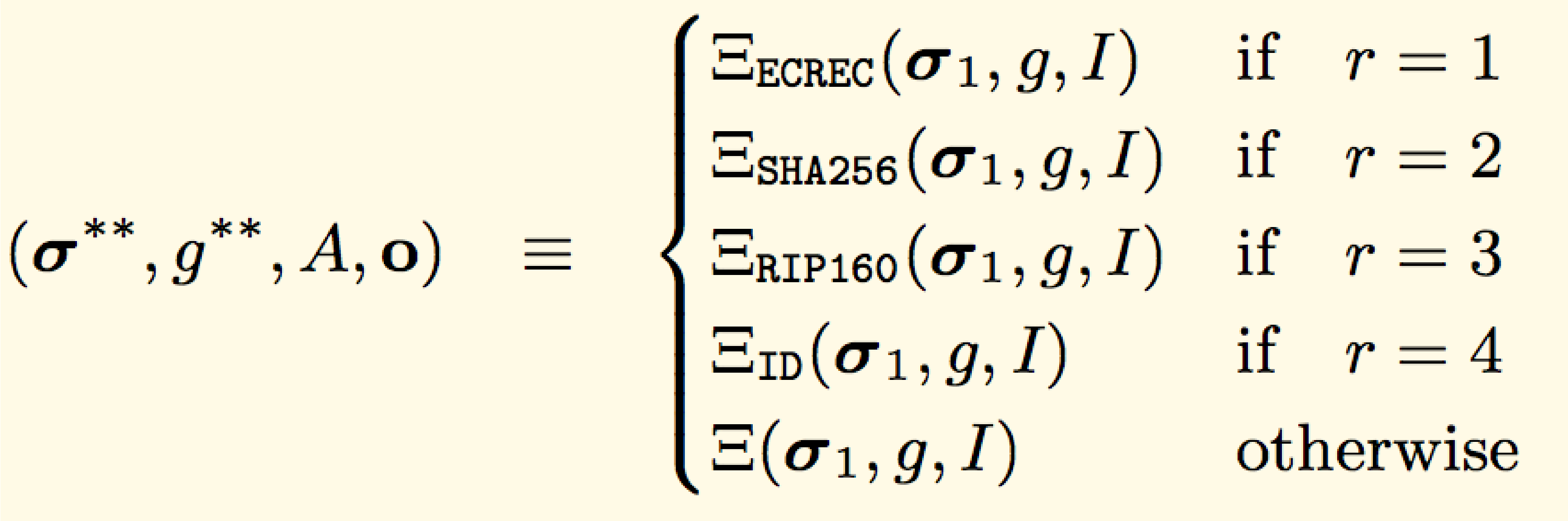
$$\boldsymbol{\sigma}_1[s]_b \equiv \boldsymbol{\sigma}'_1[s]_b - v$$

$$\text{and} \quad \boldsymbol{\sigma}'_1 \equiv \boldsymbol{\sigma} \quad \text{except:}$$

$$\begin{cases} \boldsymbol{\sigma}'_1[r] \equiv (v, 0, \mathtt{KEC}(()), \mathtt{TRIE}(\varnothing)) & \text{if} \quad \boldsymbol{\sigma}[r] = \varnothing \\ \boldsymbol{\sigma}'_1[r]_b \equiv \boldsymbol{\sigma}[r]_b + v & \text{otherwise} \end{cases}$$

Section 8: Message Call

(2) Execute

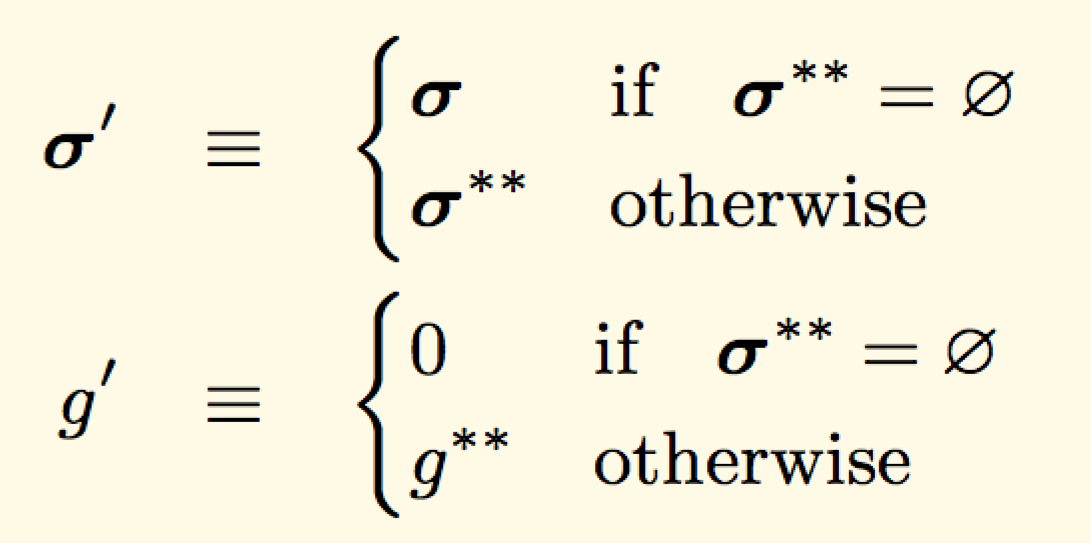


$$(\boldsymbol{\sigma}^{**}, g^{**}, A, \mathbf{o}) \equiv \begin{cases} \Xi_{\mathtt{ECREC}}(\boldsymbol{\sigma}_1, g, I) & \text{if} \quad r = 1 \\ \Xi_{\mathtt{SHA256}}(\boldsymbol{\sigma}_1, g, I) & \text{if} \quad r = 2 \\ \Xi_{\mathtt{RIP160}}(\boldsymbol{\sigma}_1, g, I) & \text{if} \quad r = 3 \\ \Xi_{\mathtt{ID}}(\boldsymbol{\sigma}_1, g, I) & \text{if} \quad r = 4 \\ \Xi(\boldsymbol{\sigma}_1, g, I) & \text{otherwise} \end{cases}$$

Where r=1, r=2, r=3, r=4 are Precompiled Contracts

Section 8: Message Call

(3) Determine Final State

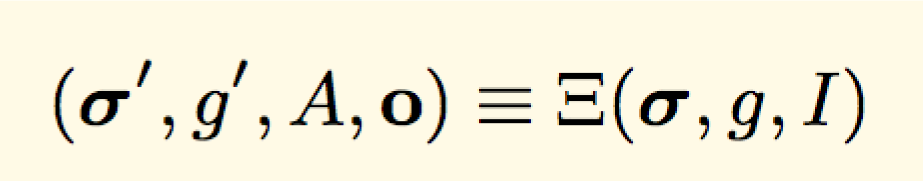


$$\boldsymbol{\sigma}' \equiv \begin{cases} \boldsymbol{\sigma} & \text{if} \quad \boldsymbol{\sigma}^{**} = \varnothing \\ \boldsymbol{\sigma}^{**} & \text{otherwise} \end{cases}$$

$$g' \equiv \begin{cases} 0 & \text{if} \quad \boldsymbol{\sigma}^{**} = \varnothing \\ g^{**} & \text{otherwise} \end{cases}$$

Section 9: Execution Model

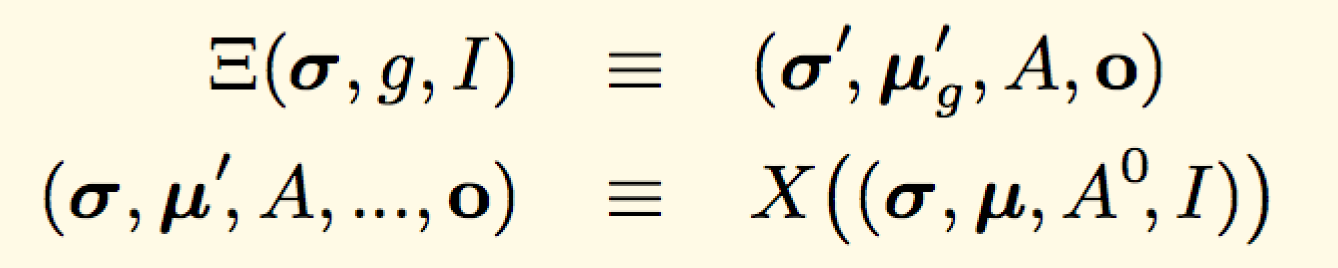
Overview



$$(\boldsymbol{\sigma}', g', A, \mathbf{o}) \equiv \Xi(\boldsymbol{\sigma}, g, I)$$

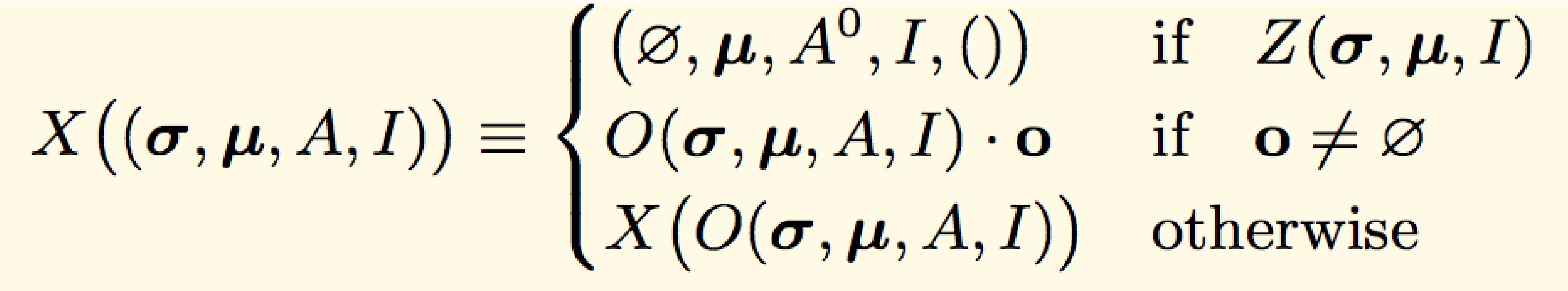
Section 9: Execution Model

(1) The Execution Function

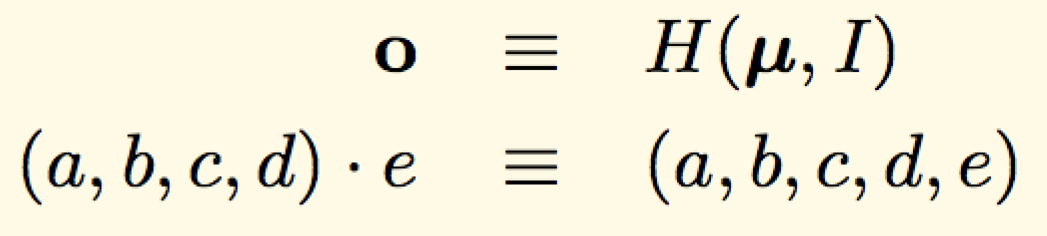


$$\begin{aligned}\Xi(\boldsymbol{\sigma}, g, I) &\equiv (\boldsymbol{\sigma}', \boldsymbol{\mu}'_g, A, \mathbf{o}) \\ (\boldsymbol{\sigma}, \boldsymbol{\mu}', A, ..., \mathbf{o}) &\equiv X\big((\boldsymbol{\sigma}, \boldsymbol{\mu}, A^0, I)\big)\end{aligned}$$

Where



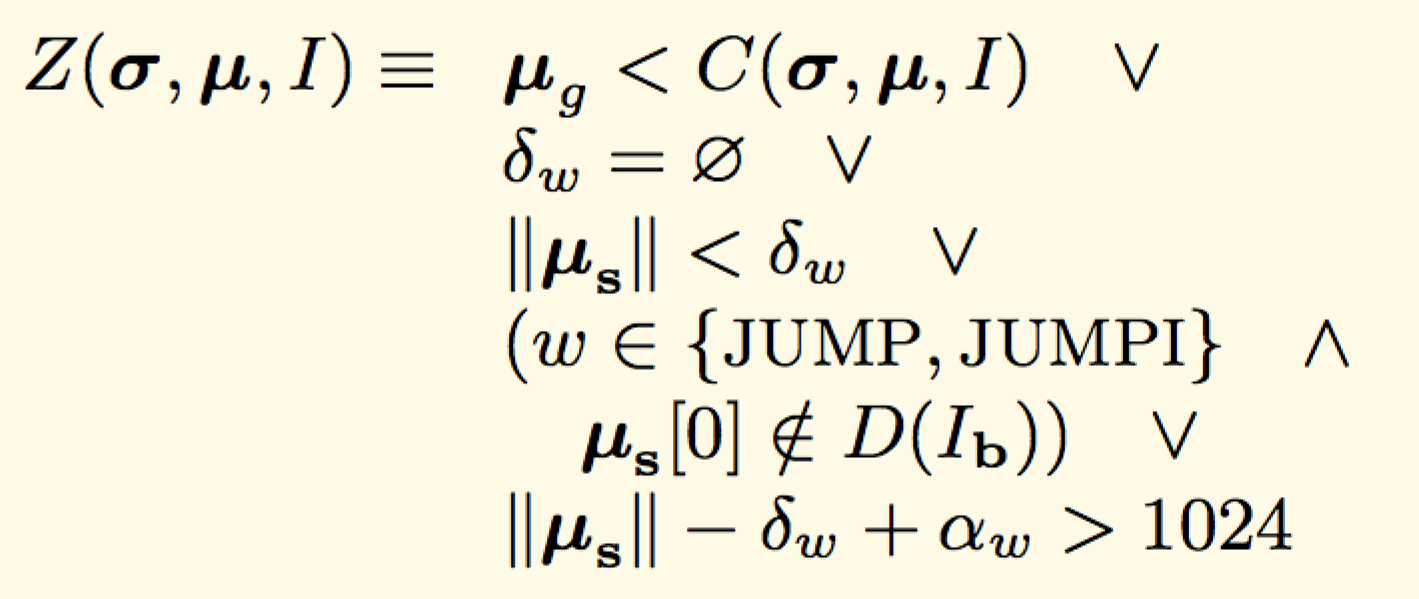
$$X\big((\boldsymbol{\sigma}, \boldsymbol{\mu}, A, I)\big) \equiv \begin{cases} \big(\varnothing, \boldsymbol{\mu}, A^0, I, ()\big) & \text{if} \quad Z(\boldsymbol{\sigma}, \boldsymbol{\mu}, I) \\ O(\boldsymbol{\sigma}, \boldsymbol{\mu}, A, I) \cdot \mathbf{o} & \text{if} \quad \mathbf{o} \neq \varnothing \\ X\big(O(\boldsymbol{\sigma}, \boldsymbol{\mu}, A, I)\big) & \text{otherwise} \end{cases}$$



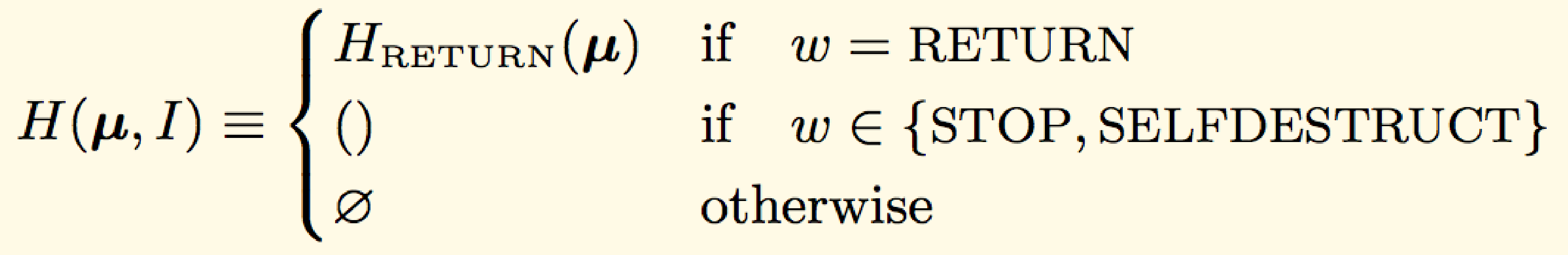
$$\begin{aligned}\mathbf{o} &\equiv H(\boldsymbol{\mu}, I) \\ (a, b, c, d) \cdot e &\equiv (a, b, c, d, e)\end{aligned}$$

Section 9: Execution Model

(2) Conditions



$$
\begin{aligned}
Z(\boldsymbol{\sigma}, \boldsymbol{\mu}, I) \equiv \quad & \boldsymbol{\mu}_g < C(\boldsymbol{\sigma}, \boldsymbol{\mu}, I) \quad \vee \\
& \delta_w = \varnothing \quad \vee \\
& \lVert \boldsymbol{\mu}_{\mathbf{s}} \rVert < \delta_w \quad \vee \\
& (w \in \{\mathrm{JUMP}, \mathrm{JUMPI}\} \quad \wedge \\
& \quad \boldsymbol{\mu}_{\mathbf{s}}[0] \notin D(I_{\mathbf{b}})) \quad \vee \\
& \lVert \boldsymbol{\mu}_{\mathbf{s}} \rVert - \delta_w + \alpha_w > 1024
\end{aligned}
$$



$$
H(\boldsymbol{\mu}, I) \equiv \begin{cases} H_{\text{RETURN}}(\boldsymbol{\mu}) & \text{if} \quad w = \mathrm{RETURN} \\ () & \text{if} \quad w \in \{\mathrm{STOP}, \mathrm{SELFDESTRUCT}\} \\ \varnothing & \text{otherwise} \end{cases}
$$

Section 9: Execution Model

(3) The Execution Cycle

$$\begin{aligned} O\big((\boldsymbol{\sigma}, \boldsymbol{\mu}, A, I)\big) &\equiv (\boldsymbol{\sigma}', \boldsymbol{\mu}', A', I) \\ \Delta &\equiv \alpha_w - \delta_w \end{aligned}$$

Section 9: Execution Model

(3) The Execution Cycle (Cont.)

Where

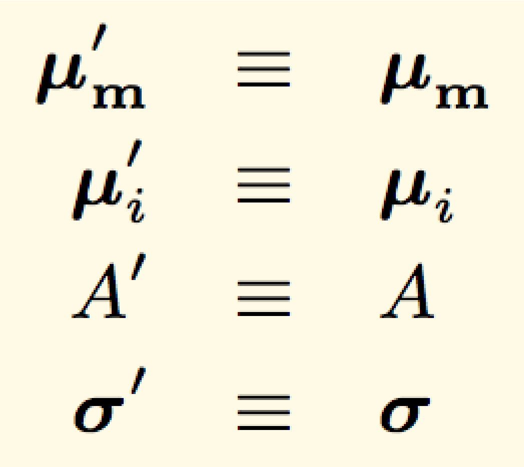
$$\begin{aligned}
\lVert \boldsymbol{\mu}'_{\mathbf{s}} \rVert &\equiv \lVert \boldsymbol{\mu}_{\mathbf{s}} \rVert + \Delta \\
\forall x \in [\alpha_w, \lVert \boldsymbol{\mu}'_{\mathbf{s}} \rVert) : \boldsymbol{\mu}'_{\mathbf{s}}[x] &\equiv \boldsymbol{\mu}_{\mathbf{s}}[x + \Delta]
\end{aligned}$$

$$\begin{aligned}
\boldsymbol{\mu}'_g &\equiv \boldsymbol{\mu}_g - C(\boldsymbol{\sigma}, \boldsymbol{\mu}, I) \\
\boldsymbol{\mu}'_{pc} &\equiv \begin{cases} J_{\text{JUMP}}(\boldsymbol{\mu}) & \text{if} \quad w = \text{JUMP} \\ J_{\text{JUMPI}}(\boldsymbol{\mu}) & \text{if} \quad w = \text{JUMPI} \\ N(\boldsymbol{\mu}_{pc}, w) & \text{otherwise} \end{cases}
\end{aligned}$$

Section 9: Execution Model

(3) The Execution Cycle (Cont.)

Where



$$
\begin{aligned}
\boldsymbol{\mu}'_{\mathbf{m}} &\equiv \boldsymbol{\mu}_{\mathbf{m}} \\
\boldsymbol{\mu}'_{i} &\equiv \boldsymbol{\mu}_{i} \\
A' &\equiv A \\
\boldsymbol{\sigma}' &\equiv \boldsymbol{\sigma}
\end{aligned}
$$

Conclusion

- Formulas are concise and elegant
- Focused on the whole picture thus many details are skipped
- Focused the EVM thus the consensus strategy are skipped