AlphaTensor: 用增強式學習找出更有效率的矩陣相乘演算法

Speaker: Hung-yi Lee

AlphaTensor

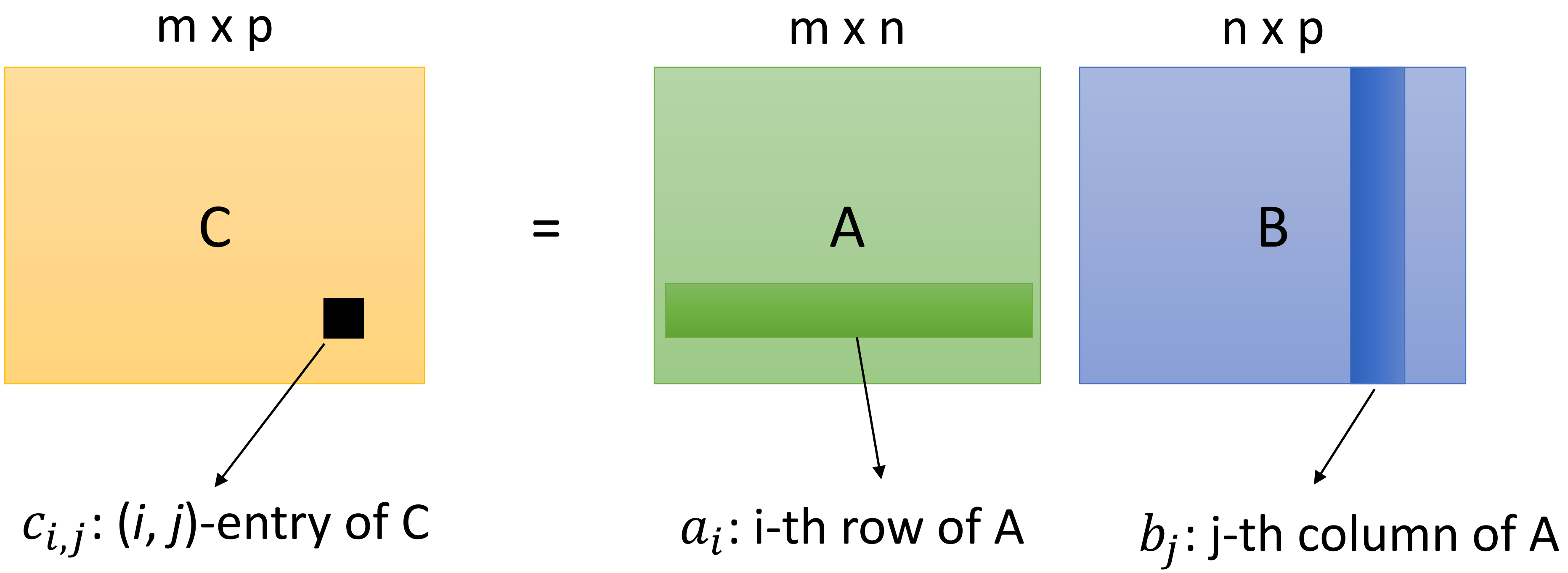
AlphaGo

AlphaStar

AlphaTensor

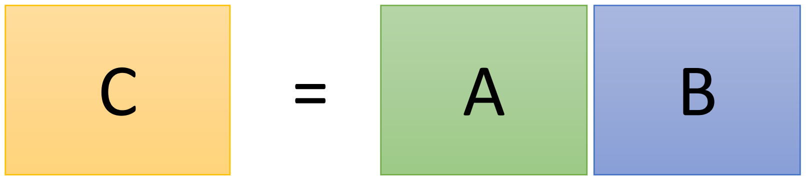
何謂更有效率的
矩陣相乘演算法？

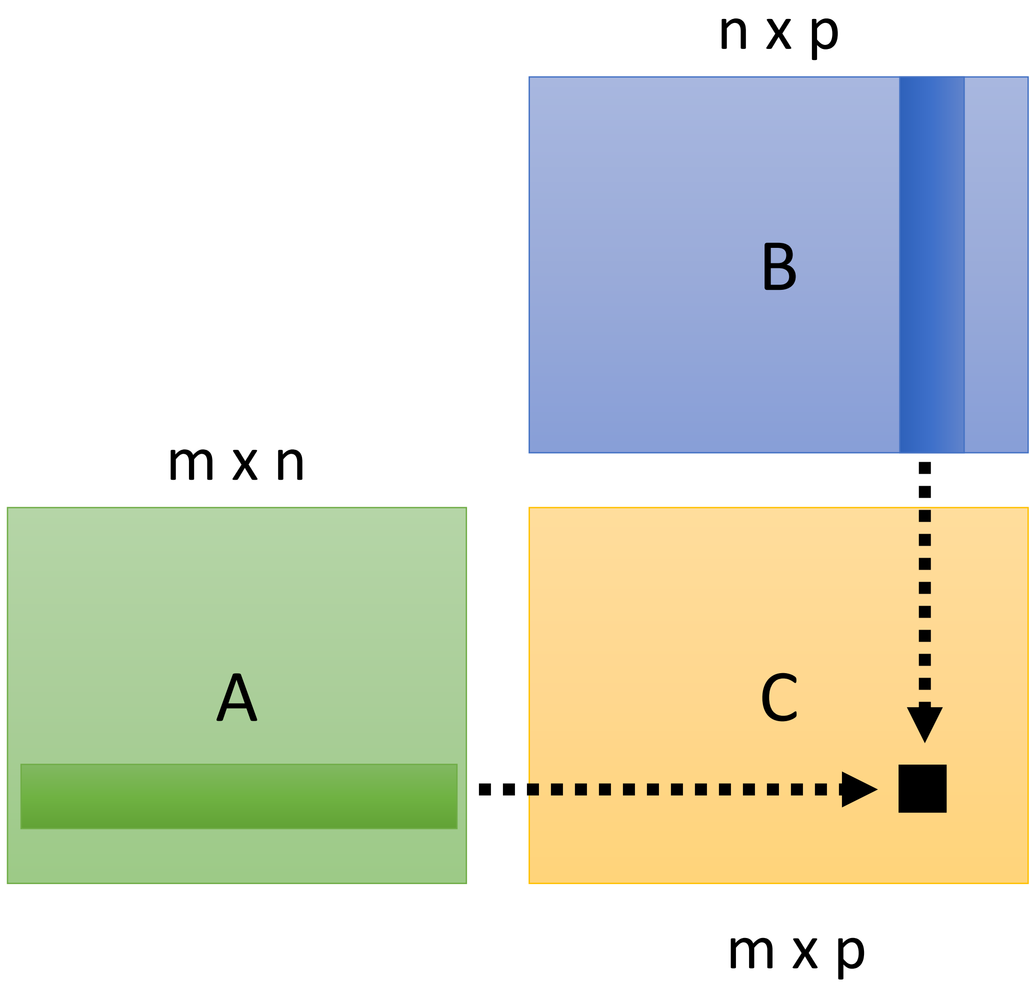
矩陣相乘



$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{in}b_{nj}$$

矩陣相乘





$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{in}b_{nj}$$

Each element in C needs n multiplications

The are m x p elements in C.

Total: m x p x n multiplications

m = p = n

$\boxed{\mathbf{n^3}\text{ multiplications}}$

以 2 x 2 矩陣相乘為例

$$\begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

$$c_{11} = a_{11}b_{11} + a_{12}b_{21}$$

$$c_{12} = a_{11}b_{12} + a_{12}b_{22}$$

$$c_{21} = a_{21}b_{11} + a_{22}b_{21}$$

$$c_{22} = a_{21}b_{12} + a_{22}b_{22}$$

8 multiplications

有沒有更有效率的方法？

(更少的乘法次數)

以 2 x 2 矩陣相乘為例

$$\begin{bmatrix} \boxed{c_{11}} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

$$\boxed{c_{11}} = a_{11}b_{11} + a_{12}b_{21}$$

$$= (a_{11} + a_{22})(b_{11} + b_{22}) + a_{22}(b_{21} - b_{11}) - (a_{11} + a_{12})b_{22} + (a_{12} - a_{22})(b_{21} + b_{22})$$

$$a_{11}b_{11} + \cancel{a_{11}b_{22}} + \cancel{a_{22}b_{11}} + \cancel{a_{22}b_{22}} \qquad \cancel{-a_{11}b_{22}} \cancel{- a_{12}b_{22}}$$

$$\cancel{a_{22}b_{21}} - \cancel{a_{22}b_{11}} \qquad a_{12}b_{21} + \cancel{a_{12}b_{22}} - \cancel{a_{22}b_{21}} - \cancel{a_{22}b_{22}}$$

以 2 x 2 矩陣相乘為例

$$\begin{bmatrix} c_{11} & \boxed{c_{12}} \\ \boxed{c_{21}} & c_{22} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

$$\boxed{c_{12}} = a_{11}b_{12} + a_{12}b_{22} \quad = a_{11}(b_{12} - b_{22}) + (a_{11} + a_{12})b_{22}$$

$$= a_{11}b_{12} \cancel{- a_{11}b_{22}} \cancel{+ a_{11}b_{22}} + a_{12}b_{22}$$

$$\boxed{c_{21}} = a_{21}b_{11} + a_{22}b_{21} \quad = (a_{21} + a_{22})b_{11} + a_{22}(b_{21} - b_{11})$$

$$= a_{21}b_{11} \cancel{+ a_{22}b_{11}} + a_{22}b_{21} \cancel{- a_{22}b_{11}}$$

以 2 x 2 矩陣相乘為例

$$\begin{bmatrix} c_{11} & c_{12} \\ c_{21} & \boxed{c_{22}} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

$$\boxed{c_{22}} = a_{21}b_{12} + a_{22}b_{22}$$

$$= a_{11}(b_{12} - b_{22}) + (a_{11} + a_{22})(b_{11} + b_{22})$$

$$-(a_{21} + a_{22})b_{11} \quad -(a_{11} - a_{21})(b_{11} + b_{12})$$

$$\cancel{a_{11}b_{12}} \cancel{- a_{11}b_{22}} \qquad \cancel{-a_{21}b_{11}} \cancel{- a_{22}b_{11}}$$

$$\cancel{a_{11}b_{11}} \cancel{+ a_{11}b_{22}} \cancel{+ a_{22}b_{11}} + a_{22}b_{22} \qquad \cancel{-a_{11}b_{11}} \cancel{- a_{11}b_{12}} \cancel{+ a_{21}b_{11}} + a_{21}b_{12}$$

$$c_{11} = a_{11}b_{11} + a_{12}b_{21} = \underbrace{(a_{11} + a_{22})(b_{11} + b_{22})}_{h_1} + \underbrace{a_{22}(b_{21} - b_{11})}_{h_2} - \underbrace{(a_{11} + a_{12})b_{22}}_{h_3} + \underbrace{(a_{12} - a_{22})(b_{21} + b_{22})}_{h_4}$$

$$c_{12} = a_{11}b_{12} + a_{12}b_{22} = \underbrace{a_{11}(b_{12} - b_{22})}_{h_5} + \underbrace{(a_{11} + a_{12})b_{22}}_{h_3}$$

$$c_{21} = a_{21}b_{11} + a_{22}b_{21} = \underbrace{(a_{21} + a_{22})b_{11}}_{h_6} + \underbrace{a_{22}(b_{21} - b_{11})}_{h_2}$$

7 multiplications!!!

$$c_{22} = a_{21}b_{12} + a_{22}b_{22} = \underbrace{a_{11}(b_{12} - b_{22})}_{h_5} + \underbrace{(a_{11} + a_{22})(b_{11} + b_{22})}_{h_1} - \underbrace{(a_{21} + a_{22})b_{11}}_{h_6} - \underbrace{(a_{11} - a_{21})(b_{11} + b_{12})}_{h_7}$$

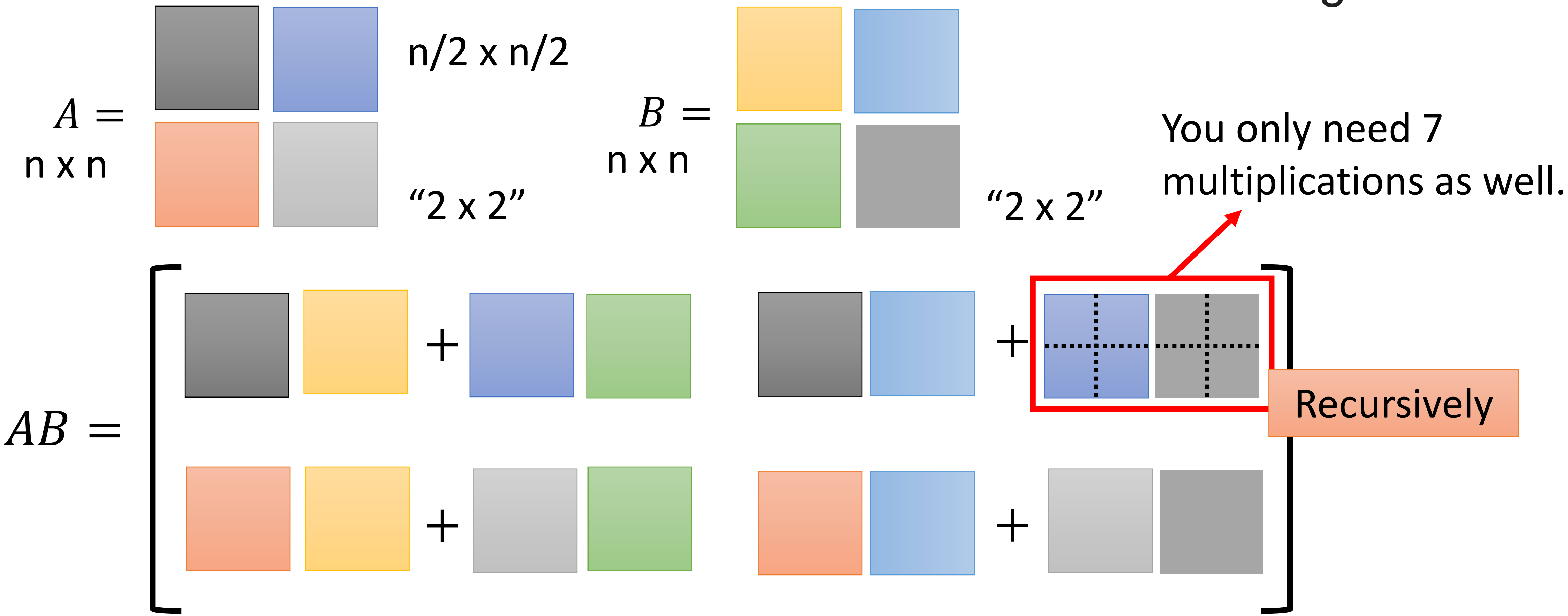
$$c_{11} = h_1 + h_2 - h_3 + h_4 \qquad c_{21} = h_6 + h_2$$

$$c_{12} = h_5 + h_3 \qquad c_{22} = h_5 + h_1 - h_6 - h_7$$

不只是 2 x 2 矩陣相乘而已

Block Multiplication

Strassen algorithm



8 matrix multiplications? No, you only need **7**.

嗯 … 更有效率的矩陣相乘演算法確實是存在的

但這件事為什麼重要？

矩陣相乘的重要性

- Linear System

$$b = A \; x$$

- Deep Neural Network

Non-linear function (e.g., ReLU)

$$y \quad \ldots\ldots \quad \underbrace{c' = f(\,c\,) \quad c = A_2 \; b'}_{\text{one layer}} \quad b' = f(\,b\,) \quad b = A_1 \; x$$

“cat”

Deep = many layers

矩陣相乘的重要性

- To learn more about **deep learning** ……

https://youtu.be/Ye018rCVvOo

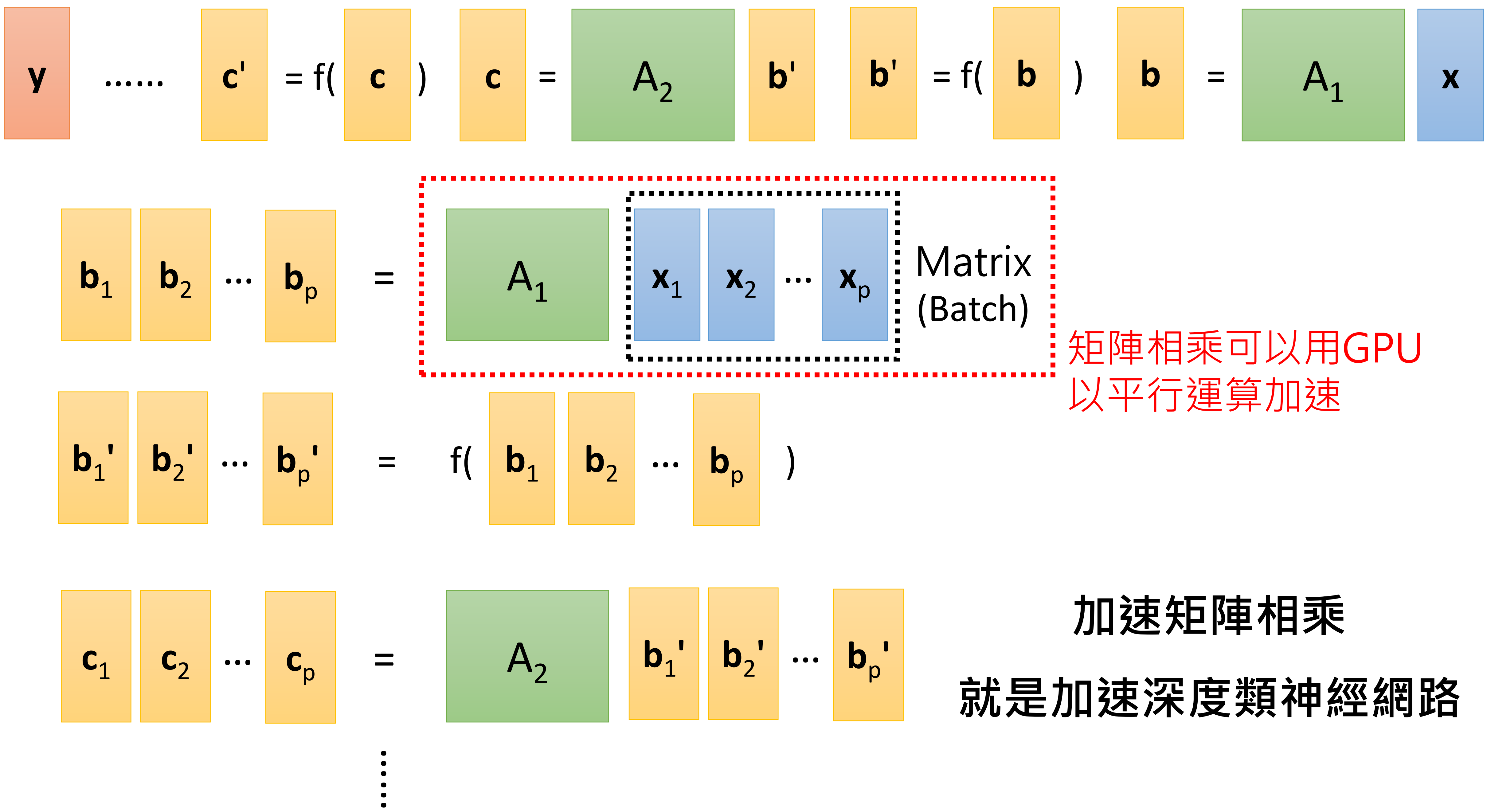
https://youtu.be/bHcJCp2Fyxs

$\mathbf{x}_1 =$  $\mathbf{x}_2 =$  $\mathbf{x}_3 =$ 

$\mathbf{y}_1$ …… $\mathbf{c}_1' = f(\mathbf{c}_1)$ $\quad \mathbf{c}_1 = A_2 \mathbf{b}_1'$ $\quad \mathbf{b}_1' = f(\mathbf{b}_1)$ $\quad \mathbf{b}_1 = A_1 \mathbf{x}_1$

$\mathbf{y}_2$ …… $\mathbf{c}_2' = f(\mathbf{c}_2)$ $\quad \mathbf{c}_2 = A_2 \mathbf{b}_2'$ $\quad \mathbf{b}_2' = f(\mathbf{b}_2)$ $\quad \mathbf{b}_2 = A_1 \mathbf{x}_2$

$\mathbf{y}_3$ …… $\mathbf{c}_3' = f(\mathbf{c}_3)$ $\quad \mathbf{c}_3 = A_2 \mathbf{b}_3'$ $\quad \mathbf{b}_3' = f(\mathbf{b}_3)$ $\quad \mathbf{b}_3 = A_1 \mathbf{x}_3$


y ...... c' = f(c) c = A2 b' b' = f(b) b = A1 x
b1 b2 … bp = A1 x1 x2 … xp
Matrix
(Batch)
矩陣相乘可以用GPU
以平行運算加速
b1' b2' … bp' = f(b1 b2 … bp)
c1 c2 … cp = A2 b1' b2' … bp'
加速矩陣相乘
就是加速深度類神經網路

如何找出更有效率的
矩陣相乘演算法呢？

$$c_{11} = a_{11}b_{11} + a_{12}b_{21}$$
$$c_{12} = a_{11}b_{12} + a_{12}b_{22}$$
$$c_{21} = a_{21}b_{11} + a_{22}b_{21}$$
$$c_{22} = a_{21}b_{12} + a_{22}b_{22}$$

$$h_1 = a_{11}b_{11}$$
$$h_2 = a_{12}b_{21}$$
$$h_3 = a_{11}b_{12}$$
$$h_4 = a_{12}b_{22}$$
$$h_5 = a_{21}b_{11}$$
$$h_6 = a_{22}b_{21}$$
$$h_7 = a_{21}b_{12}$$
$$h_8 = a_{22}b_{22}$$

$$c_{11} = h_1 + h_2$$
$$c_{12} = h_3 + h_4$$
$$c_{21} = h_5 + h_6$$
$$c_{22} = h_7 + h_8$$

$$h_1 = (a_{11} + a_{22})(b_{11} + b_{22})$$
$$h_2 = a_{22}(b_{21} - b_{11})$$
$$h_3 = (a_{11} + a_{12})b_{22}$$
$$h_4 = (a_{12} - a_{22})(b_{21} + b_{22})$$
$$h_5 = a_{11}(b_{12} - b_{22})$$
$$h_6 = (a_{21} + a_{22})b_{11}$$
$$h_7 = (a_{11} - a_{21})(b_{11} + b_{12})$$

$$c_{11} = h_1 + h_2 - h_3 + h_4$$
$$c_{21} = h_6 + h_2$$
$$c_{12} = h_5 + h_3$$
$$c_{22} = h_5 + h_1 - h_6 - h_7$$

$$h_1 = (\overset{\boxed{1}}{u^1_{11}} a_{11} + \overset{\boxed{0}}{u^1_{12}} a_{12} + \overset{\boxed{0}}{u^1_{21}} a_{21} + \overset{\boxed{1}}{u^1_{22}} a_{22})$$
$$(\underset{\boxed{1}}{v^1_{11}} b_{11} + \underset{\boxed{0}}{v^1_{12}} b_{12} + \underset{\boxed{0}}{v^1_{21}} b_{21} + \underset{\boxed{1}}{v^1_{22}} b_{22})$$

$$h_2 = (u^2_{11} a_{11} + u^2_{12} a_{12} + u^2_{21} a_{21} + u^2_{22} a_{22})$$
$$(v^2_{11} b_{11} + v^2_{12} b_{12} + v^2_{21} b_{21} + v^2_{22} b_{22})$$

$$\vdots$$

$$h_R = (u^R_{11} a_{11} + u^R_{12} a_{12} + u^R_{21} a_{21} + u^R_{22} a_{22})$$
$$(v^R_{11} b_{11} + v^R_{12} b_{12} + v^R_{21} b_{21} + v^R_{22} b_{22})$$

$$\boxed{1} \quad \boxed{1} \quad \boxed{-1} \quad \boxed{1} \quad \boxed{0} \quad \ldots\ldots$$

$$c_{11} = w^1_{11} h_1 + w^2_{11} h_2 + \cdots + w^R_{11} h_R$$
$$c_{12} = w^1_{12} h_1 + w^2_{12} h_2 + \cdots + w^R_{12} h_R$$
$$c_{21} = w^1_{21} h_1 + w^2_{21} h_2 + \cdots + w^R_{21} h_R$$
$$c_{22} = w^1_{22} h_1 + w^2_{22} h_2 + \cdots + w^R_{22} h_R$$

$$h_1 = (a_{11} + a_{22})(b_{11} + b_{22})$$
$$h_2 = a_{22}(b_{21} - b_{11})$$
$$h_3 = (a_{11} + a_{12}) b_{22}$$
$$h_4 = (a_{12} - a_{22})(b_{21} + b_{22})$$
$$h_5 = a_{11}(b_{12} - b_{22})$$
$$h_6 = (a_{21} + a_{22}) b_{11}$$
$$h_7 = (a_{11} - a_{21})(b_{11} + b_{12})$$

$$R = 7$$

$$c_{11} = h_1 + h_2 - h_3 + h_4$$
$$c_{21} = h_6 + h_2$$
$$c_{12} = h_5 + h_3$$
$$c_{22} = h_5 + h_1 - h_6 - h_7$$

$$h_1 = (\overset{\boxed{1}}{u_{11}^{1}}a_{11} + \overset{\boxed{0}}{u_{12}^{1}}a_{12} + \overset{\boxed{0}}{u_{21}^{1}}a_{21} + \overset{\boxed{0}}{u_{22}^{1}}a_{22})$$
$$(v_{11}^{1}b_{11} + v_{12}^{1}b_{12} + v_{21}^{1}b_{21} + v_{22}^{1}b_{22})$$

$$h_2 = (\overset{\boxed{1}}{u_{11}^{2}}a_{11} + \overset{\boxed{0}}{u_{12}^{2}}a_{12} + \overset{\boxed{0}}{u_{21}^{2}}a_{21} + \overset{\boxed{0}}{u_{22}^{2}}a_{22})$$
$$(v_{11}^{2}b_{11} + v_{12}^{2}b_{12} + v_{21}^{2}b_{21} + v_{22}^{2}b_{22})$$

$$\vdots$$

$$h_R = (u_{11}^{R}a_{11} + u_{12}^{R}a_{12} + u_{21}^{R}a_{21} + u_{22}^{R}a_{22})$$
$$(v_{11}^{R}b_{11} + v_{12}^{R}b_{12} + v_{21}^{R}b_{21} + v_{22}^{R}b_{22})$$

$$c_{11} = \overset{\boxed{1}}{w_{11}^{1}}h_1 + \overset{\boxed{1}}{w_{11}^{2}}h_2 + \cdots + \overset{\boxed{0}}{w_{11}^{R}}h_R$$
$$c_{12} = w_{12}^{1}h_1 + w_{12}^{2}h_2 + \cdots + w_{12}^{R}h_R$$
$$c_{21} = w_{21}^{1}h_1 + w_{21}^{2}h_2 + \cdots + w_{21}^{R}h_R$$
$$c_{22} = w_{22}^{1}h_1 + w_{22}^{2}h_2 + \cdots + w_{22}^{R}h_R$$

$$R = 8$$

$$h_1 = a_{11}b_{11}$$
$$h_2 = a_{12}b_{21}$$
$$h_3 = a_{11}b_{12}$$
$$h_4 = a_{12}b_{22}$$
$$h_5 = a_{21}b_{11}$$
$$h_6 = a_{22}b_{21}$$
$$h_7 = a_{21}b_{12}$$
$$h_8 = a_{22}b_{22}$$

$$c_{11} = h_1 + h_2$$
$$c_{12} = h_3 + h_4$$
$$c_{21} = h_5 + h_6$$
$$c_{22} = h_7 + h_8$$

$$h_1 = (u^1_{11}a_{11} + u^1_{12}a_{12} + u^1_{21}a_{21} + u^1_{22}a_{22})$$
$$(v^1_{11}b_{11} + v^1_{12}b_{12} + v^1_{21}b_{21} + v^1_{22}b_{22})$$

$$h_2 = (u^2_{11}a_{11} + u^2_{12}a_{12} + u^2_{21}a_{21} + u^2_{22}a_{22})$$
$$(v^2_{11}b_{11} + v^2_{12}b_{12} + v^2_{21}b_{21} + v^2_{22}b_{22})$$

$$\vdots$$

$$h_R = (u^R_{11}a_{11} + u^R_{12}a_{12} + u^R_{21}a_{21} + u^R_{22}a_{22})$$
$$(v^R_{11}b_{11} + v^R_{12}b_{12} + v^R_{21}b_{21} + v^R_{22}b_{22})$$

$$c_{11} = \boxed{w^1_{11}}h_1 + \boxed{w^2_{11}}h_2 + \cdots + \boxed{w^R_{11}}h_R$$
$$c_{12} = w^1_{12}h_1 + w^2_{12}h_2 + \cdots + w^R_{12}h_R$$
$$c_{21} = w^1_{21}h_1 + w^2_{21}h_2 + \cdots + w^R_{21}h_R$$
$$c_{22} = w^1_{22}h_1 + w^2_{22}h_2 + \cdots + w^R_{22}h_R$$

$$\boldsymbol{u^1} = [u^1_{11} \quad u^1_{12} \quad u^1_{21} \quad u^1_{22}]$$
$$\vdots$$
$$\boldsymbol{u^R} = [u^R_{11} \quad u^R_{12} \quad u^R_{21} \quad u^R_{22}]$$
$$\boldsymbol{v^1} = [v^1_{11} \quad v^1_{12} \quad v^1_{21} \quad v^1_{22}]$$
$$\vdots$$
$$\boldsymbol{v^R} = [v^R_{11} \quad v^R_{12} \quad v^R_{21} \quad v^R_{22}]$$
$$\boldsymbol{w^1} = [w^1_{11} \quad w^1_{12} \quad w^1_{21} \quad w^1_{22}]$$
$$\vdots$$
$$\boldsymbol{w^R} = [w^R_{11} \quad w^R_{12} \quad w^R_{21} \quad w^R_{22}]$$

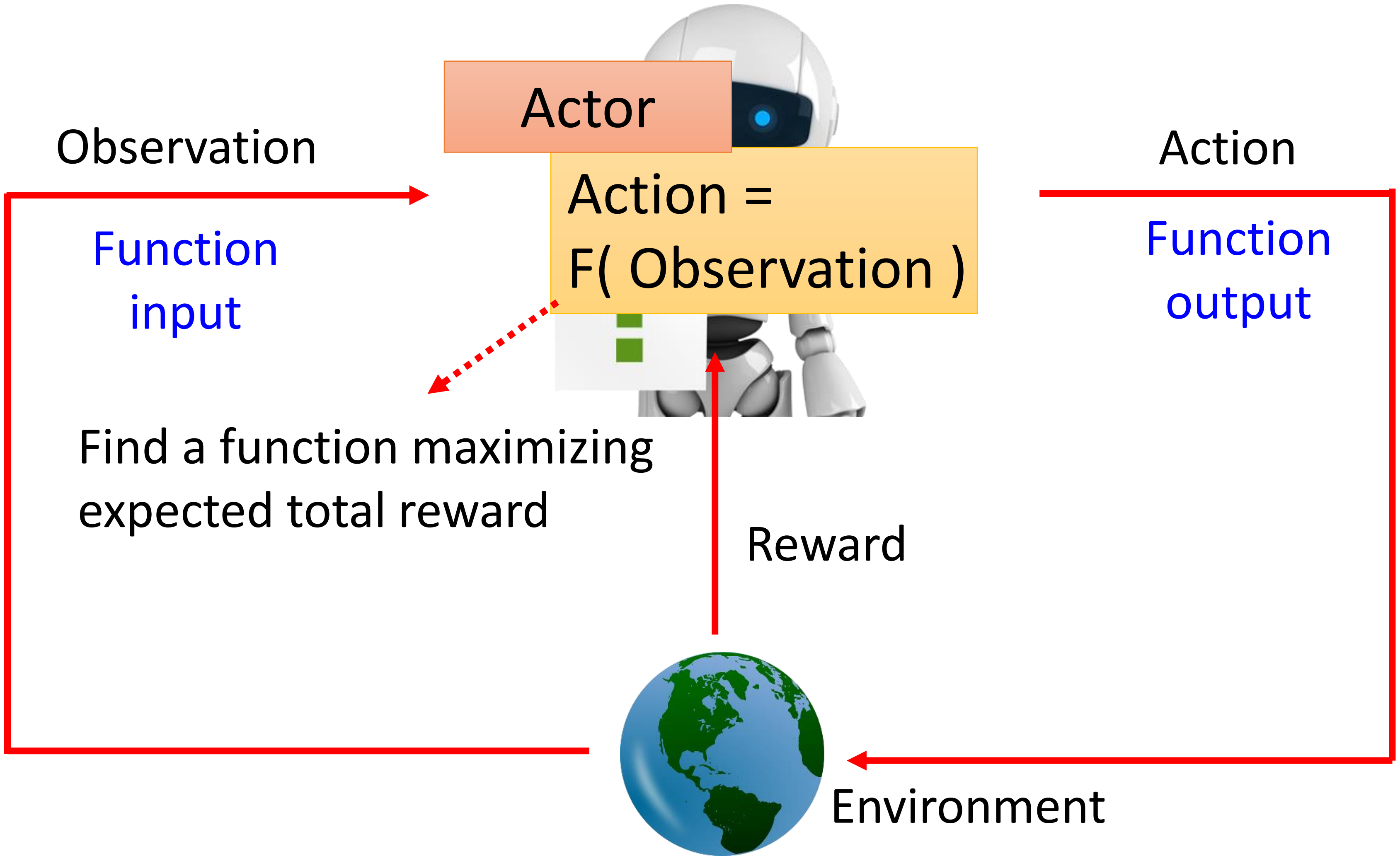
Find $\boldsymbol{u^1} \dots \boldsymbol{u^R}$, $\boldsymbol{v^1} \dots \boldsymbol{v^R}$, $\boldsymbol{w^1} \dots \boldsymbol{w^R}$

Minimizing R

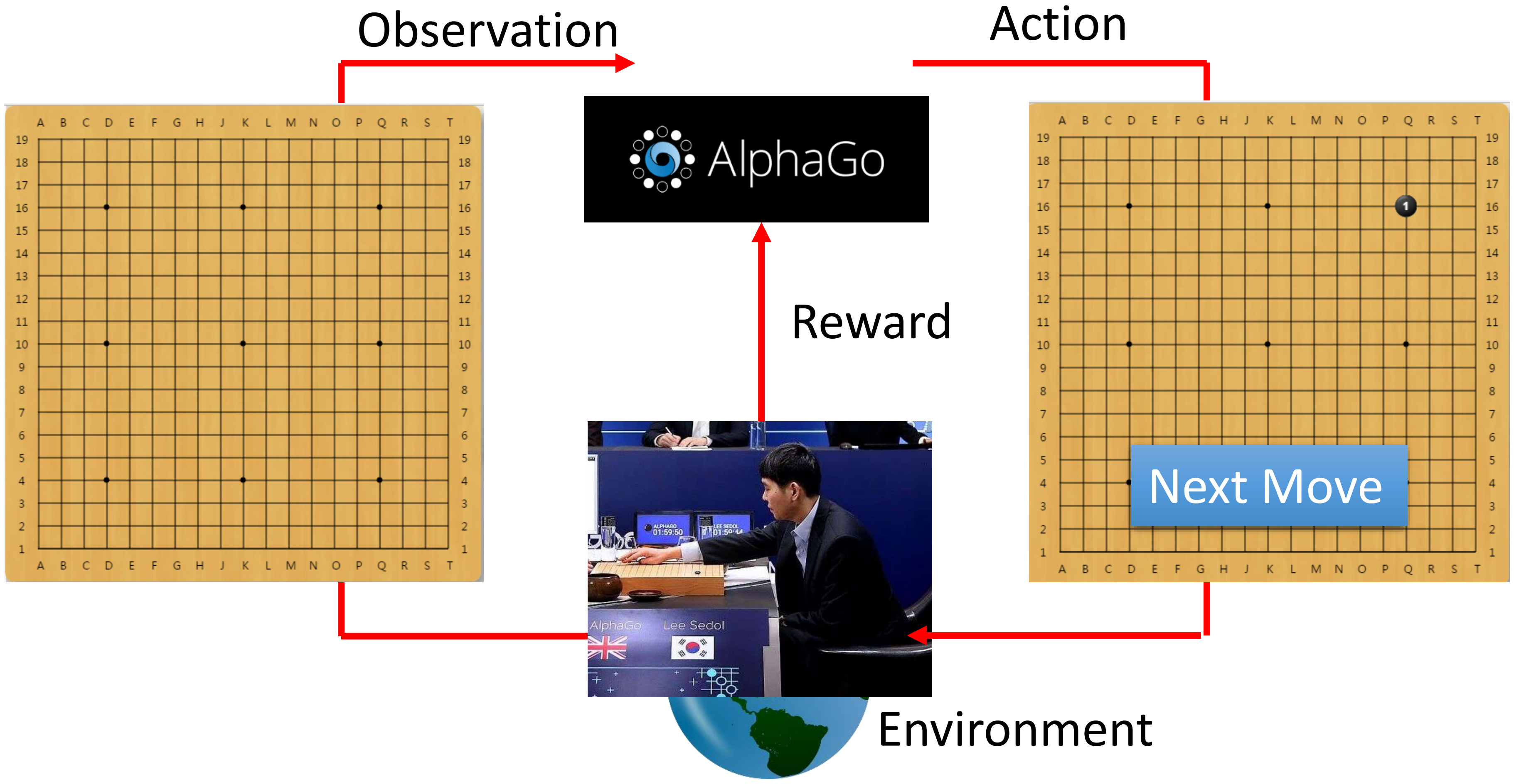
用機器學習中的增強式學習硬 train !?

用增強式學習硬 train 矩陣相乘演算法

Reinforcement Learning, RL (增強式學習)

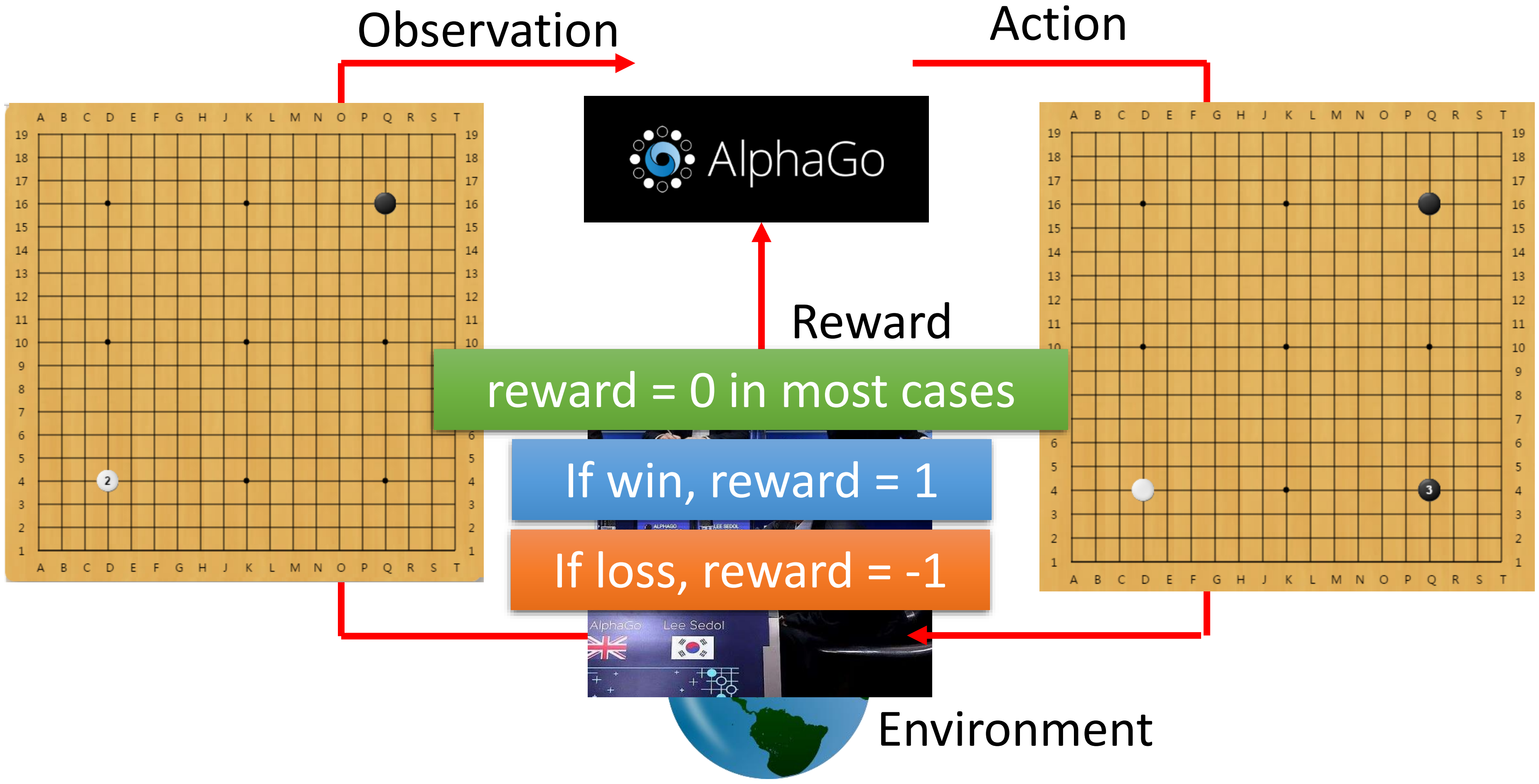


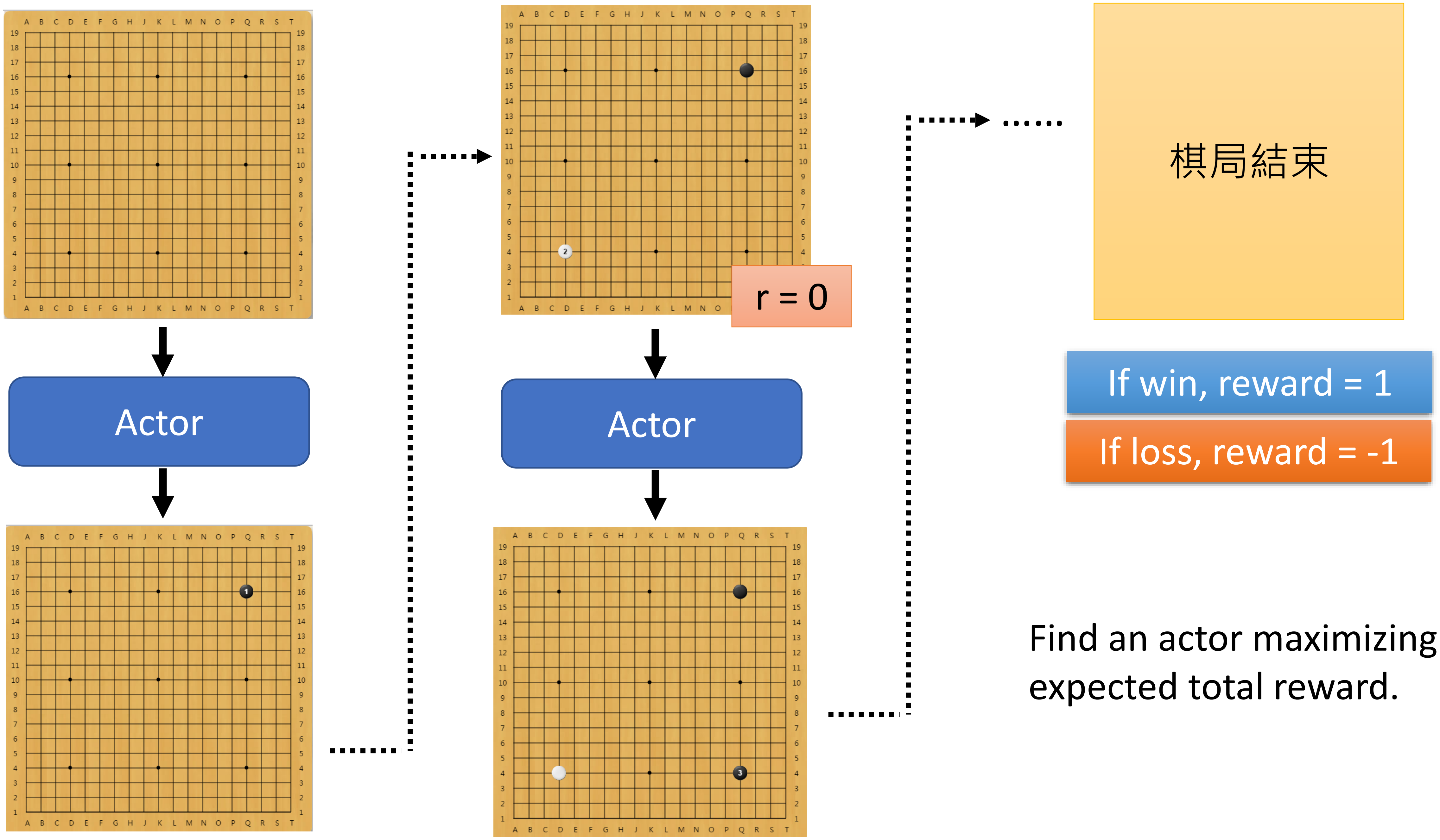
Example: Learning to play Go



Example: Learning to play Go

Find an actor maximizing expected total reward.





Find an actor maximizing expected total reward.

To know more about RL ……

- 【機器學習2021】概述增強式學習 (Reinforcement Learning, RL) (一) – 增強式學習跟機器學習一樣都是三個步驟
  - https://youtu.be/XWukX-ayIrs
- 【機器學習2021】概述增強式學習 (Reinforcement Learning, RL) (二) – Policy Gradient 與修課心情
  - https://youtu.be/US8DFaAZcp4
- 【機器學習2021】概述增強式學習 (Reinforcement Learning, RL) (三) – Actor-Critic
  - https://youtu.be/kk6DqWreLeU

➢ MCTS is not introduced.

$$h_1 = (u_{11}^1 a_{11} + u_{12}^1 a_{12} + u_{21}^1 a_{21} + u_{22}^1 a_{22})$$
$$(v_{11}^1 b_{11} + v_{12}^1 b_{12} + v_{21}^1 b_{21} + v_{22}^1 b_{22})$$

$$h_2 = (u_{11}^2 a_{11} + u_{12}^2 a_{12} + u_{21}^2 a_{21} + u_{22}^2 a_{22})$$
$$(v_{11}^2 b_{11} + v_{12}^2 b_{12} + v_{21}^2 b_{21} + v_{22}^2 b_{22})$$

$$\vdots$$

$$h_R = (u_{11}^R a_{11} + u_{12}^R a_{12} + u_{21}^R a_{21} + u_{22}^R a_{22})$$
$$(v_{11}^R b_{11} + v_{12}^R b_{12} + v_{21}^R b_{21} + v_{22}^R b_{22})$$

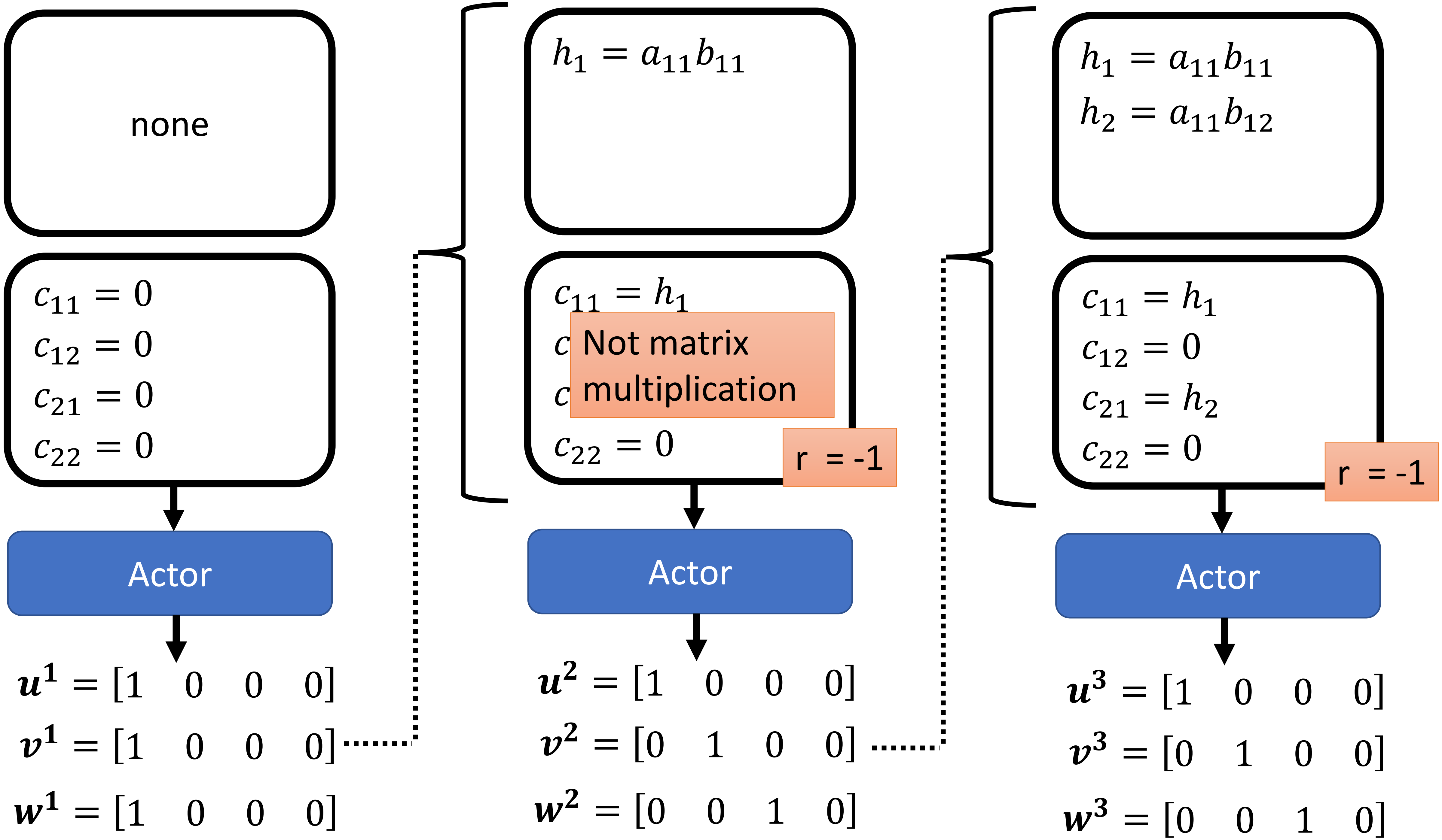
$$c_{11} = w_{11}^1 h_1 + w_{11}^2 h_2 + \cdots + w_{11}^R h_R$$
$$c_{12} = w_{12}^1 h_1 + w_{12}^2 h_2 + \cdots + w_{12}^R h_R$$
$$c_{21} = w_{21}^1 h_1 + w_{21}^2 h_2 + \cdots + w_{21}^R h_R$$
$$c_{22} = w_{22}^1 h_1 + w_{22}^2 h_2 + \cdots + w_{22}^R h_R$$

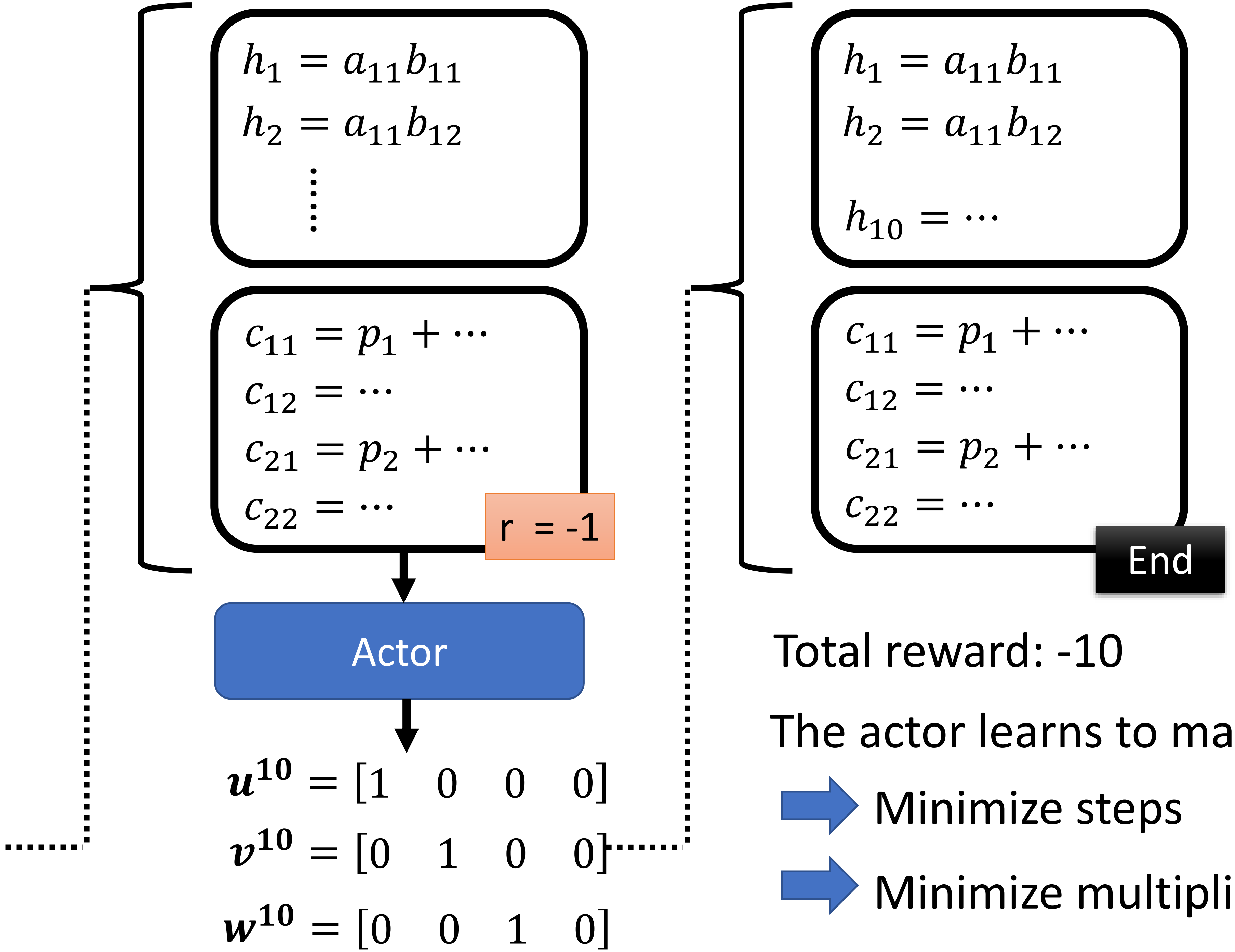
$$\boldsymbol{u^1} = [u_{11}^1 \quad u_{12}^1 \quad u_{21}^1 \quad u_{22}^1]$$
$$\vdots$$
$$\boldsymbol{u^R} = [u_{11}^R \quad u_{12}^R \quad u_{21}^R \quad u_{22}^R]$$
$$\boldsymbol{v^1} = [v_{11}^1 \quad v_{12}^1 \quad v_{21}^1 \quad v_{22}^1]$$
$$\vdots$$
$$\boldsymbol{v^R} = [v_{11}^R \quad v_{12}^R \quad v_{21}^R \quad v_{22}^R]$$
$$\boldsymbol{w^1} = [w_{11}^1 \quad w_{12}^1 \quad w_{21}^1 \quad w_{22}^1]$$
$$\vdots$$
$$\boldsymbol{w^R} = [w_{11}^R \quad w_{12}^R \quad w_{21}^R \quad w_{22}^R]$$

Find $\boldsymbol{u^1} \ldots \boldsymbol{u^R}$, $\boldsymbol{v^1} \ldots \boldsymbol{v^R}$, $\boldsymbol{w^1} \ldots \boldsymbol{w^R}$

Minimizing R

用機器學習中的增強式學習硬 train !?


none
$c_{11} = 0$
$c_{12} = 0$
$c_{21} = 0$
$c_{22} = 0$
Actor
$\boldsymbol{u}^1 = [1 \quad 0 \quad 0 \quad 0]$
$\boldsymbol{v}^1 = [1 \quad 0 \quad 0 \quad 0]$
$\boldsymbol{w}^1 = [1 \quad 0 \quad 0 \quad 0]$
$h_1 = a_{11}b_{11}$
$c_{11} = h_1$
Not matrix multiplication
$c_{22} = 0$
r = -1
Actor
$\boldsymbol{u}^2 = [1 \quad 0 \quad 0 \quad 0]$
$\boldsymbol{v}^2 = [0 \quad 1 \quad 0 \quad 0]$
$\boldsymbol{w}^2 = [0 \quad 0 \quad 1 \quad 0]$
$h_1 = a_{11}b_{11}$
$h_2 = a_{11}b_{12}$
$c_{11} = h_1$
$c_{12} = 0$
$c_{21} = h_2$
$c_{22} = 0$
r = -1
Actor
$\boldsymbol{u}^3 = [1 \quad 0 \quad 0 \quad 0]$
$\boldsymbol{v}^3 = [0 \quad 1 \quad 0 \quad 0]$
$\boldsymbol{w}^3 = [0 \quad 0 \quad 1 \quad 0]$



$$c_{11} = a_{11}b_{11} + a_{12}b_{21}$$
$$c_{12} = a_{11}b_{12} + a_{12}b_{22}$$
$$c_{21} = a_{21}b_{11} + a_{22}b_{21}$$
$$c_{22} = a_{21}b_{12} + a_{22}b_{22}$$

Total reward: -10

The actor learns to maximize total reward.

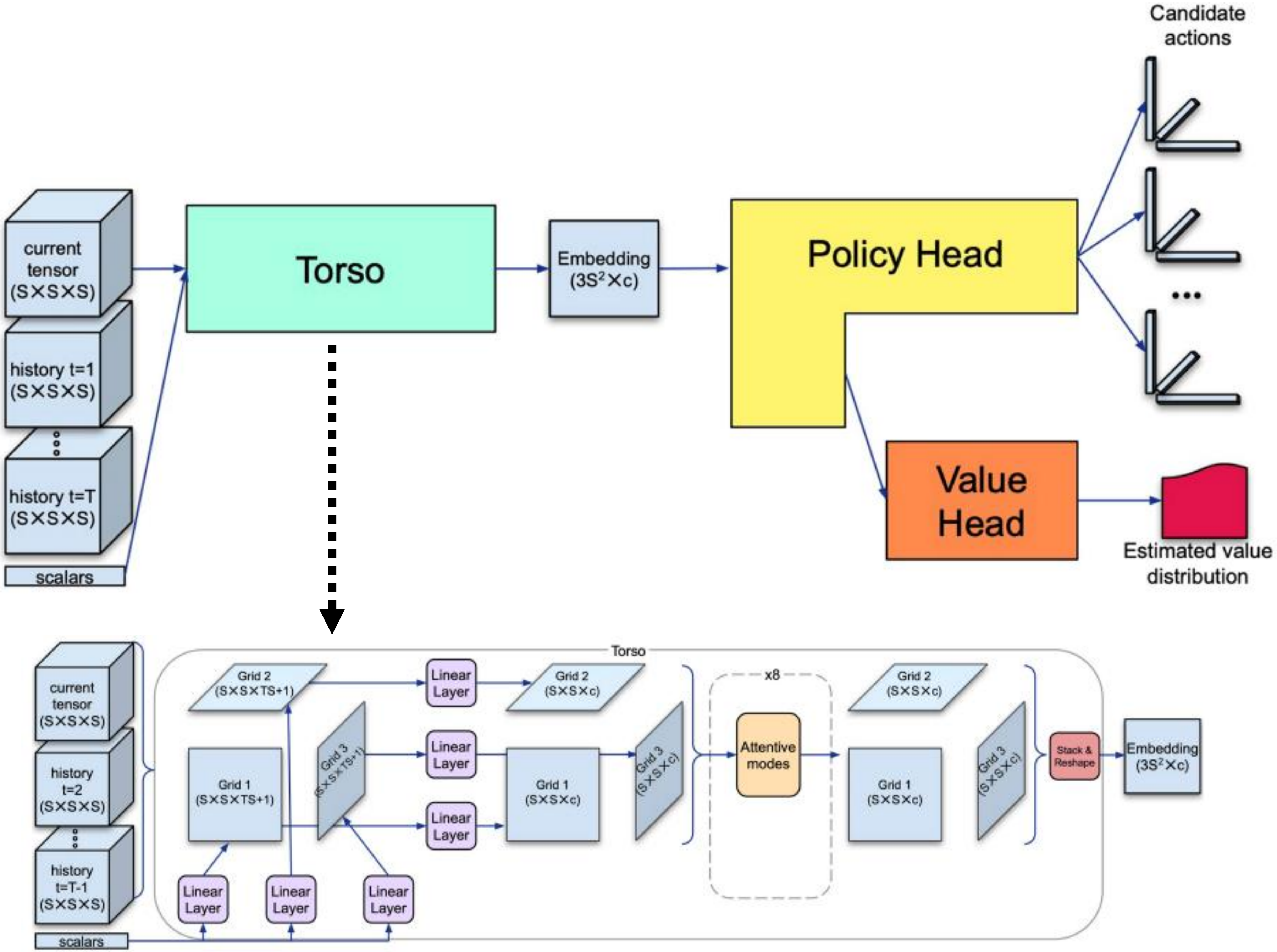
- Minimize steps
- Minimize multiplications

$$h_1 = a_{11} b_{11}$$
$$h_2 = a_{11} b_{12}$$

$$c_{11} = h_1$$
$$c_{12} = 0$$
$$c_{21} = h_2$$
$$c_{22} = 0$$

Actor

$$\boldsymbol{u}^3 = [1 \quad 0 \quad 0 \quad 0]$$
$$\boldsymbol{v}^3 = [0 \quad 1 \quad 0 \quad 0]$$
$$\boldsymbol{w}^3 = [0 \quad 0 \quad 1 \quad 0]$$

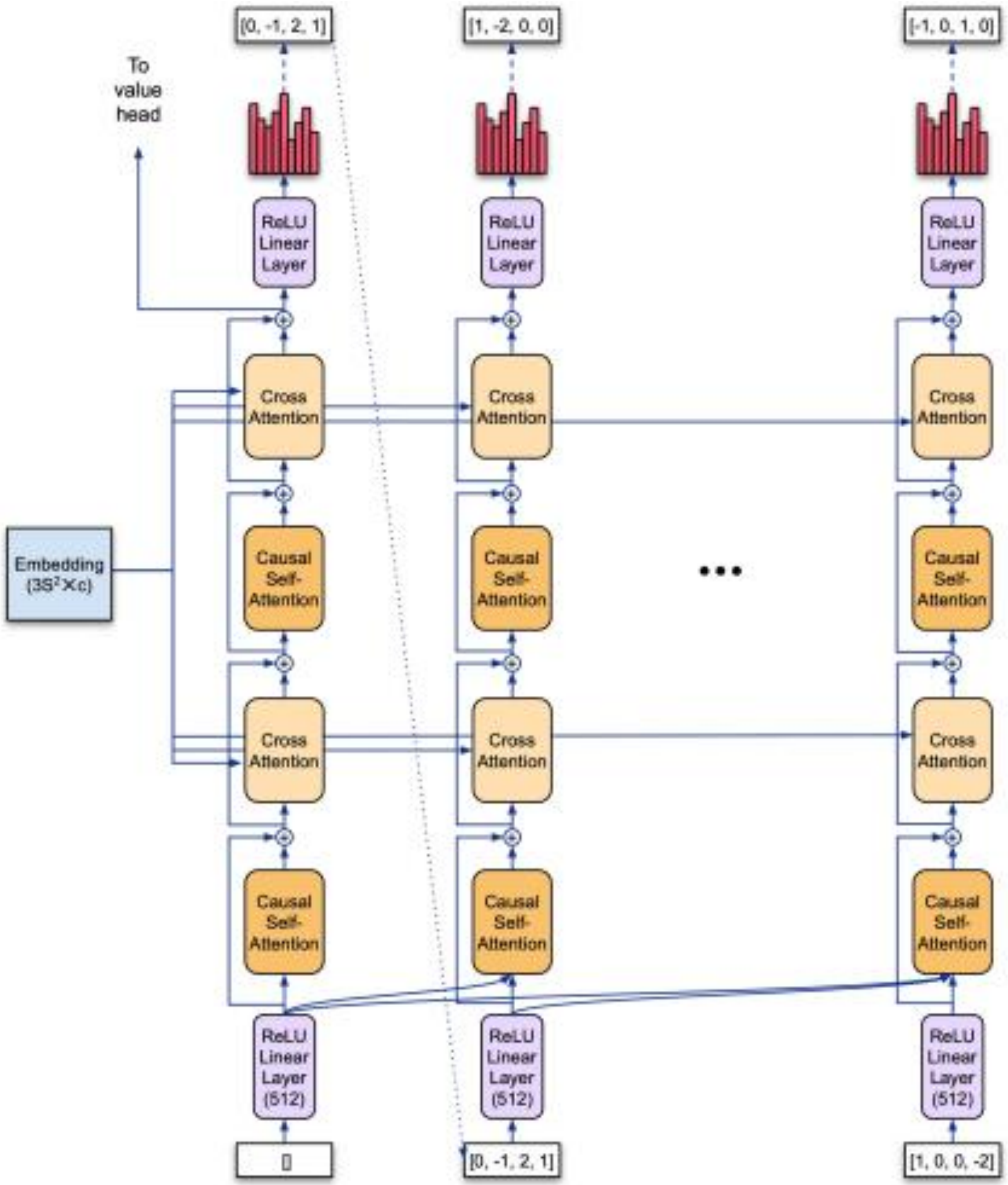


$$h_1 = a_{11} b_{11}$$
$$h_2 = a_{11} b_{12}$$

$$c_{11} = h_1$$
$$c_{12} = 0$$
$$c_{21} = h_2$$
$$c_{22} = 0$$

Actor

$$\boldsymbol{u}^3 = [1 \quad 0 \quad 0 \quad 0]$$
$$\boldsymbol{v}^3 = [0 \quad 1 \quad 0 \quad 0]$$
$$\boldsymbol{w}^3 = [0 \quad 0 \quad 1 \quad 0]$$



結果如何?

Size (n, m, p)	Best method known	Best rank known	AlphaTensor rank Modular	AlphaTensor rank Standard
(2, 2, 2)	(Strassen, 1969)[2]	7	7	7
(3, 3, 3)	(Laderman, 1976)[15]	23	23	23
(4, 4, 4)	(Strassen, 1969)[2] (2, 2, 2) ⊗ (2, 2, 2)	49	47	49
(5, 5, 5)	(3, 5, 5) + (2, 5, 5)	98	96	98
(2, 2, 3)	(2, 2, 2) + (2, 2, 1)	11	11	11
(2, 2, 4)	(2, 2, 2) + (2, 2, 2)	14	14	14
(2, 2, 5)	(2, 2, 2) + (2, 2, 3)	18	18	18
(2, 3, 3)	(Hopcroft and Kerr, 1971)[16]	15	15	15
			20	20
			25	25
			26	26
			33	33
			40	40
(3, 3, 4)	(Smirnov, 2013)[18]	29	29	29
(3, 3, 5)	(Smirnov, 2013)[18]	36	36	36
(3, 4, 4)	(Smirnov, 2013)[18]	38	38	38
(3, 4, 5)	(Smirnov, 2013)[18]	48	47	47
(3, 5, 5)	(Sedoglavic and Smirnov, 2021)[19]	58	58	58
(4, 4, 5)	(4, 4, 2) + (4, 4, 3)	64	63	63
$\boxed{(4, 5, 5)}$	(2, 5, 5) ⊗ (2, 1, 1)	$\boxed{80}$	76	$\boxed{76}$

$$\begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} & a_{1,5} \\ a_{2,1} & a_{2,2} & a_{2,3} & a_{2,4} & a_{2,5} \\ a_{3,1} & a_{3,2} & a_{3,3} & a_{3,4} & a_{3,5} \\ a_{4,1} & a_{4,2} & a_{4,3} & a_{4,4} & a_{4,5} \end{pmatrix} \begin{pmatrix} b_{1,1} & b_{1,2} & b_{1,3} & b_{1,4} & b_{1,5} \\ b_{2,1} & b_{2,2} & b_{2,3} & b_{2,4} & b_{2,5} \\ b_{3,1} & b_{3,2} & b_{3,3} & b_{3,4} & b_{3,5} \\ b_{4,1} & b_{4,2} & b_{4,3} & b_{4,4} & b_{4,5} \\ b_{5,1} & b_{5,2} & b_{5,3} & b_{5,4} & b_{5,5} \end{pmatrix} = \begin{pmatrix} c_{1,1} & c_{1,2} & c_{1,3} & c_{1,4} & c_{1,5} \\ c_{2,1} & c_{2,2} & c_{2,3} & c_{2,4} & c_{2,5} \\ c_{3,1} & c_{3,2} & c_{3,3} & c_{3,4} & c_{3,5} \\ c_{4,1} & c_{4,2} & c_{4,3} & c_{4,4} & c_{4,5} \end{pmatrix}$$

4 x 5　　　　5 x 5

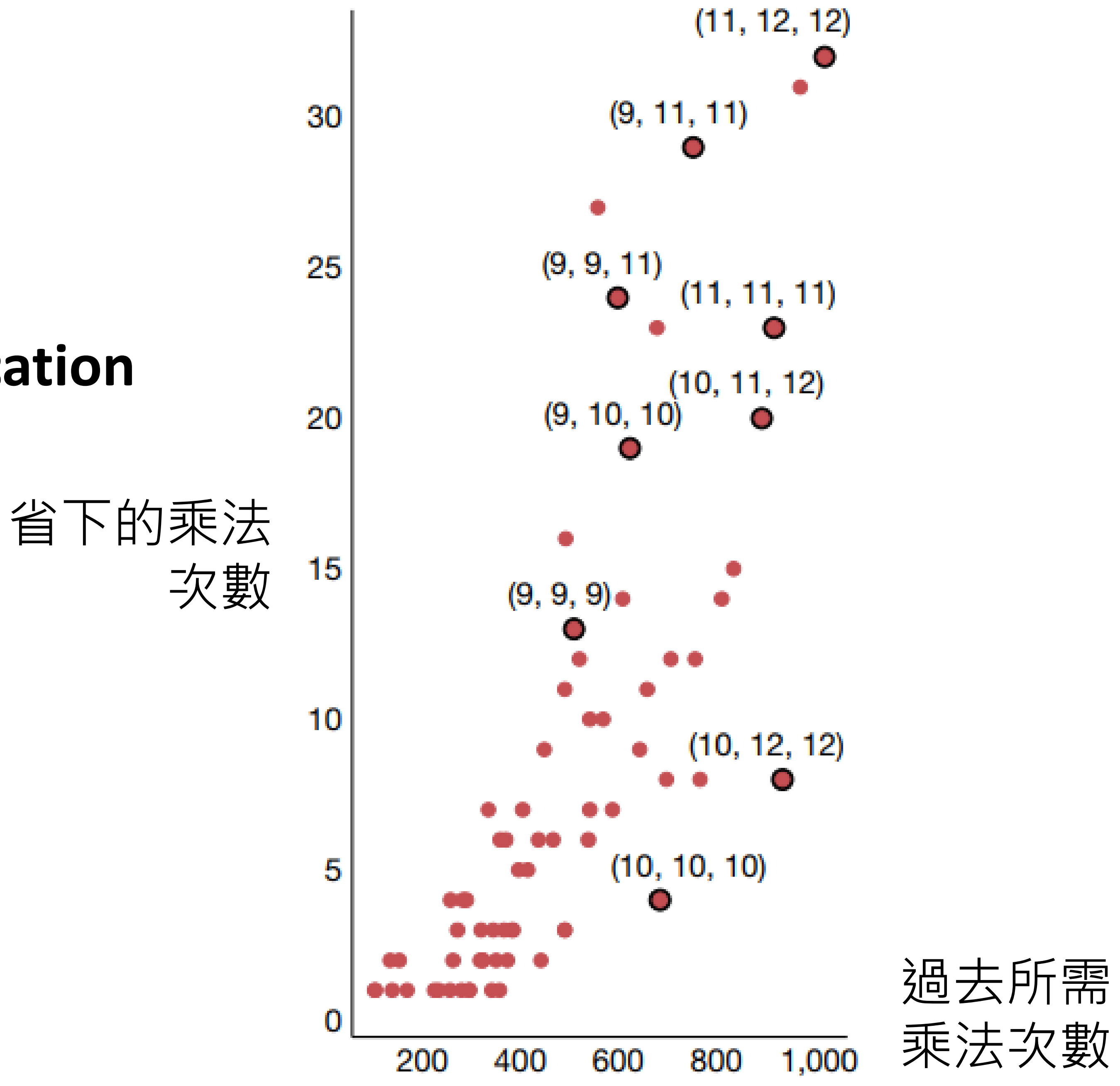
$$h_1 = a_{3,2}\left(-b_{2,1} - b_{2,5} - b_{3,1}\right)$$
$$h_2 = \left(a_{2,2} + a_{2,5} - a_{3,5}\right)\left(-b_{2,5} - b_{5,1}\right)$$
$$h_3 = \left(-a_{3,1} - a_{4,1} + a_{4,2}\right)\left(-b_{1,1} + b_{2,5}\right)$$
$$h_4 = \left(a_{1,2} + a_{1,4} + a_{3,4}\right)\left(-b_{2,5} - b_{4,1}\right)$$
$$h_5 = \left(a_{1,5} + a_{2,2} + a_{2,5}\right)\left(-b_{2,4} + b_{5,1}\right)$$
$$h_6 = \left(-a_{2,2} - a_{2,5} - a_{4,5}\right)\left(b_{2,3} + b_{5,1}\right)$$
$$h_7 = \left(-a_{1,1} + a_{4,1} - a_{4,2}\right)\left(b_{1,1} + b_{2,4}\right)$$
$$h_8 = \left(a_{3,2} - a_{3,3} - a_{4,3}\right)\left(-b_{2,3} + b_{3,1}\right)$$
$$h_9 = \left(-a_{1,2} - a_{1,4} + a_{4,4}\right)\left(b_{2,3} + b_{4,1}\right)$$
$$h_{10} = \left(a_{2,2} + a_{2,5}\right) b_{5,1}$$
$$h_{11} = \left(-a_{2,1} - a_{4,1} + a_{4,2}\right)\left(-b_{1,1} + b_{2,2}\right)$$
$$h_{12} = \left(a_{4,1} - a_{4,2}\right) b_{1,1}$$
$$h_{13} = \left(a_{1,2} + a_{1,4} + a_{2,4}\right)\left(b_{2,2} + b_{4,1}\right)$$
$$h_{14} = \left(a_{1,3} - a_{3,2} + a_{3,3}\right)\left(b_{2,4} + b_{3,1}\right)$$
$$h_{15} = \left(-a_{1,2} - a_{1,4}\right) b_{4,1}$$
$$h_{16} = \left(-a_{3,2} + a_{3,3}\right) b_{3,1}$$
$$h_{17} = \left(a_{1,2} + a_{1,4} - a_{2,1} + a_{2,2} - a_{2,3} + a_{2,4} - a_{3,2} + a_{3,3} - a_{4,1} + a_{4,2}\right) b_{2,2}$$
$$h_{18} = a_{2,1}\left(b_{1,1} + b_{1,2} + b_{5,2}\right)$$
$$h_{19} = -a_{2,3}\left(b_{3,1} + b_{3,2} + b_{5,2}\right)$$
$$h_{20} = \left(-a_{1,5} + a_{2,1} + a_{2,3} - a_{2,5}\right)\left(-b_{1,1} - b_{1,2} + b_{1,4} - b_{5,2}\right)$$
$$h_{21} = \left(a_{2,1} + a_{2,3} - a_{2,5}\right) b_{5,2}$$
$$h_{22} = \left(a_{1,3} - a_{1,4} - a_{2,4}\right)\left(b_{1,1} + b_{1,2} - b_{1,4} - b_{3,1} - b_{3,2} + b_{3,4} + b_{4,4}\right)$$
$$h_{23} = a_{1,3}\left(-b_{3,1} + b_{3,4} + b_{4,4}\right)$$
$$h_{24} = a_{1,5}\left(-b_{4,4} - b_{5,1} + b_{5,4}\right)$$
$$h_{25} = -a_{1,1}\left(b_{1,1} - b_{1,4}\right)$$
$$h_{26} = \left(-a_{1,3} + a_{1,4} + a_{1,5}\right) b_{4,4}$$
$$h_{27} = \left(a_{1,3} - a_{3,1} + a_{3,3}\right)\left(b_{1,1} - b_{1,4} + b_{1,5} + b_{3,5}\right)$$
$$h_{28} = -a_{3,4}\left(-b_{3,5} - b_{4,1} - b_{4,5}\right)$$
$$h_{29} = a_{3,1}\left(b_{1,1} + b_{1,5} + b_{3,5}\right)$$
$$h_{30} = \left(a_{3,1} - a_{3,3} + a_{3,4}\right) b_{3,5}$$
$$h_{31} = \left(-a_{1,4} - a_{1,5} - a_{3,4}\right)\left(-b_{4,4} - b_{5,1} + b_{5,4} - b_{5,5}\right)$$
$$h_{32} = \left(a_{2,1} + a_{4,1} + a_{4,4}\right)\left(b_{1,3} - b_{4,1} - b_{4,2} - b_{4,3}\right)$$
$$h_{33} = a_{4,3}\left(-b_{3,1} - b_{3,3}\right)$$
$$h_{34} = a_{4,4}\left(-b_{1,3} + b_{4,1} + b_{4,3}\right)$$
$$h_{35} = -a_{4,5}\left(b_{1,3} + b_{5,1} + b_{5,3}\right)$$
$$h_{36} = \left(a_{2,3} - a_{2,5} - a_{4,5}\right)\left(b_{3,1} + b_{3,2} + b_{3,3} + b_{5,2}\right)$$
$$h_{37} = \left(-a_{4,1} - a_{4,4} + a_{4,5}\right) b_{1,3}$$
$$h_{38} = \left(-a_{2,3} - a_{3,1} + a_{3,3} - a_{3,4}\right)\left(b_{3,5} + b_{4,1} + b_{4,2} + b_{4,5}\right)$$
$$h_{39} = \left(-a_{3,1} - a_{4,1} - a_{4,4} + a_{4,5}\right)\left(b_{1,3} + b_{5,1} + b_{5,3} + b_{5,5}\right)$$
$$h_{40} = \left(-a_{1,3} + a_{1,4} + a_{1,5} - a_{4,4}\right)\left(-b_{3,1} - b_{3,3} + b_{3,4} + b_{4,4}\right)$$
$$h_{41} = \left(-a_{1,1} + a_{4,1} - a_{4,5}\right)\left(b_{1,3} + b_{3,1} + b_{3,3} - b_{3,4} + b_{5,1} + b_{5,3} - b_{5,4}\right)$$
$$h_{42} = \left(-a_{2,1} + a_{2,5} - a_{3,5}\right)\left(-b_{1,1} - b_{1,2} - b_{1,5} + b_{4,1} + b_{4,2} + b_{4,5} - b_{5,2}\right)$$
$$h_{43} = a_{2,4}\left(b_{4,1} + b_{4,2}\right)$$
$$h_{44} = \left(a_{2,3} + a_{3,2} - a_{3,3}\right)\left(b_{2,2} - b_{3,1}\right)$$
$$h_{45} = \left(-a_{3,3} + a_{3,4} - a_{4,3}\right)\left(b_{3,5} + b_{4,1} + b_{4,3} + b_{4,5} + b_{5,1} + b_{5,3} + b_{5,5}\right)$$
$$h_{46} = -a_{3,5}\left(-b_{5,1} - b_{5,5}\right)$$
$$h_{47} = \left(a_{2,1} - a_{2,5} - a_{3,1} + a_{3,5}\right)\left(b_{1,1} + b_{1,2} + b_{1,5} - b_{4,1} - b_{4,2} - b_{4,5}\right)$$
$$h_{48} = \left(-a_{2,3} + a_{3,3}\right)\left(b_{2,2} + b_{3,2} + b_{3,5} + b_{4,1} + b_{4,2} + b_{4,5}\right)$$
$$h_{49} = \left(-a_{1,1} - a_{1,3} + a_{1,4} + a_{1,5} - a_{2,1} - a_{2,3} + a_{2,4} + a_{2,5}\right)\left(-b_{1,1} - b_{1,2} + b_{1,4}\right)$$
$$h_{50} = \left(-a_{1,4} - a_{2,4}\right)\left(b_{2,2} - b_{3,1} - b_{3,2} + b_{3,4} - b_{4,2} + b_{4,4}\right)$$
$$h_{51} = a_{2,2}\left(b_{2,1} + b_{2,2} - b_{5,1}\right)$$
$$h_{52} = a_{4,2}\left(b_{1,1} + b_{2,1} + b_{2,3}\right)$$
$$h_{53} = -a_{1,2}\left(-b_{2,1} + b_{2,4} + b_{4,1}\right)$$
$$h_{54} = \left(a_{1,2} + a_{1,4} - a_{2,2} - a_{2,5} - a_{3,2} + a_{3,3} - a_{4,2} + a_{4,3} - a_{4,4} - a_{4,5}\right) b_{2,3}$$
$$h_{55} = \left(a_{1,4} - a_{4,4}\right)\left(-b_{2,3} + b_{3,1} + b_{3,3} - b_{3,4} + b_{4,3} - b_{4,4}\right)$$
$$h_{56} = \left(a_{1,1} - a_{1,5} - a_{4,1} + a_{4,5}\right)\left(b_{3,1} + b_{3,3} - b_{3,4} + b_{5,1} + b_{5,3} - b_{5,4}\right)$$
$$h_{57} = \left(-a_{3,1} - a_{4,1}\right)\left(-b_{1,3} - b_{1,5} - b_{2,5} - b_{5,1} - b_{5,3} - b_{5,5}\right)$$
$$h_{58} = \left(-a_{1,4} - a_{1,5} - a_{3,4} - a_{3,5}\right)\left(-b_{5,1} + b_{5,4} - b_{5,5}\right)$$
$$h_{59} = \left(-a_{3,3} + a_{3,4} - a_{4,3} + a_{4,4}\right)\left(b_{4,1} + b_{4,3} + b_{4,5} + b_{5,1} + b_{5,3} + b_{5,5}\right)$$
$$h_{60} = \left(a_{2,5} + a_{4,5}\right)\left(b_{2,3} - b_{3,1} - b_{3,2} - b_{3,3} - b_{5,2} - b_{5,3}\right)$$
$$h_{61} = \left(a_{1,4} + a_{3,4}\right)\left(b_{1,1} - b_{1,4} + b_{1,5} - b_{2,5} - b_{4,4} + b_{4,5} - b_{5,1} + b_{5,4} - b_{5,5}\right)$$
$$h_{62} = \left(a_{2,1} + a_{4,1}\right)\left(b_{1,2} + b_{1,3} + b_{2,2} - b_{4,1} - b_{4,2} - b_{4,3}\right)$$
$$h_{63} = \left(-a_{3,3} - a_{4,3}\right)\left(-b_{2,3} - b_{3,3} - b_{3,5} - b_{4,1} - b_{4,3} - b_{4,5}\right)$$
$$h_{64} = \left(a_{1,1} - a_{1,3} - a_{1,4} + a_{3,1} - a_{3,3} - a_{3,4}\right)\left(b_{1,1} - b_{1,4} + b_{1,5}\right)$$
$$h_{65} = \left(-a_{1,1} + a_{4,1}\right)\left(-b_{1,3} + b_{1,4} + b_{2,4} - b_{5,1} - b_{5,3} + b_{5,4}\right)$$
$$h_{66} = \left(a_{1,1} - a_{1,2} + a_{1,3} - a_{1,5} - a_{2,2} - a_{2,5} - a_{3,2} + a_{3,3} - a_{4,1} + a_{4,2}\right) b_{2,4}$$
$$h_{67} = \left(a_{2,5} - a_{3,5}\right)\left(b_{1,1} + b_{1,2} + b_{1,5} - b_{2,5} - b_{4,1} - b_{4,2} - b_{4,5} + b_{5,2} + b_{5,5}\right)$$
$$h_{68} = \left(a_{1,1} + a_{1,3} - a_{1,4} - a_{1,5} - a_{4,1} - a_{4,3} + a_{4,4} + a_{4,5}\right)\left(-b_{3,1} - b_{3,3} + b_{3,4}\right)$$
$$h_{69} = \left(-a_{1,3} + a_{1,4} - a_{2,3} + a_{2,4}\right)\left(-b_{2,4} - b_{3,1} - b_{3,2} + b_{3,4} - b_{5,2} + b_{5,4}\right)$$
$$h_{70} = \left(a_{2,3} - a_{2,5} + a_{4,3} - a_{4,5}\right)\left(-b_{3,1} - b_{3,2} - b_{3,3}\right)$$
$$h_{71} = \left(-a_{3,1} + a_{3,3} - a_{3,4} + a_{3,5} - a_{4,1} + a_{4,3} - a_{4,4} + a_{4,5}\right)\left(-b_{5,1} - b_{5,3} - b_{5,5}\right)$$
$$h_{72} = \left(-a_{2,1} - a_{2,4} - a_{4,1} - a_{4,4}\right)\left(b_{4,1} + b_{4,2} + b_{4,3}\right)$$
$$h_{73} = \left(a_{1,3} - a_{1,4} - a_{1,5} + a_{2,3} - a_{2,4} - a_{2,5}\right)\left(b_{1,1} + b_{1,2} - b_{1,4} + b_{2,4} + b_{5,2} - b_{5,4}\right)$$
$$h_{74} = \left(a_{2,1} - a_{2,3} + a_{2,4} - a_{3,1} + a_{3,3} - a_{3,4}\right)\left(b_{4,1} + b_{4,2} + b_{4,5}\right)$$
$$h_{75} = -\left(a_{1,2} + a_{1,4} - a_{2,2} - a_{2,5} - a_{3,1} + a_{3,2} + a_{3,4} + a_{3,5} - a_{4,1} + a_{4,2}\right) b_{2,5}$$
$$h_{76} = \left(a_{1,3} + a_{3,3}\right)\left(-b_{1,1} + b_{1,4} - b_{1,5} + b_{2,4} + b_{3,4} - b_{3,5}\right)$$
$$c_{1,1} = -h_{10} + h_{12} + h_{14} - h_{15} - h_{16} + h_{53} + h_5 - h_{66} - h_7$$
$$c_{2,1} = h_{10} + h_{11} - h_{12} + h_{13} + h_{15} + h_{16} - h_{17} - h_{44} + h_{51}$$
$$c_{3,1} = h_{10} - h_{12} + h_{15} + h_{16} - h_1 + h_2 + h_3 - h_4 + h_{75}$$
$$c_{4,1} = -h_{10} + h_{12} - h_{15} - h_{16} + h_{52} + h_{54} - h_6 - h_8 + h_9$$
$$c_{1,2} = h_{13} + h_{15} + h_{20} + h_{21} - h_{22} + h_{23} + h_{25} - h_{43} + h_{49} + h_{50}$$
$$c_{2,2} = -h_{11} + h_{12} - h_{13} - h_{15} - h_{16} + h_{17} + h_{18} - h_{19} - h_{21} + h_{43} + h_{44}$$
$$c_{3,2} = -h_{16} - h_{19} - h_{21} - h_{28} - h_{29} - h_{38} + h_{42} + h_{44} - h_{47} + h_{48}$$
$$c_{4,2} = h_{11} - h_{12} - h_{18} + h_{21} - h_{32} + h_{33} - h_{34} - h_{36} + h_{62} - h_{70}$$
$$c_{1,3} = h_{15} + h_{23} + h_{24} + h_{34} - h_{37} + h_{40} - h_{41} + h_{55} - h_{56} - h_9$$
$$c_{2,3} = -h_{10} + h_{19} + h_{32} + h_{35} + h_{36} + h_{37} - h_{43} - h_{60} - h_6 - h_{72}$$
$$c_{3,3} = -h_{16} - h_{28} + h_{33} + h_{37} - h_{39} + h_{45} - h_{46} + h_{63} - h_{71} - h_8$$
$$c_{4,3} = h_{10} + h_{15} + h_{16} - h_{33} + h_{34} - h_{35} - h_{37} - h_{54} + h_6 + h_8 - h_9$$
$$c_{1,4} = -h_{10} + h_{12} + h_{14} - h_{16} + h_{23} + h_{24} + h_{25} + h_{26} + h_5 - h_{66} - h_7$$
$$c_{2,4} = h_{10} + h_{18} - h_{19} + h_{20} - h_{22} - h_{24} - h_{26} - h_5 - h_{69} + h_{73}$$
$$c_{3,4} = -h_{14} + h_{16} - h_{23} - h_{26} + h_{27} + h_{29} + h_{31} + h_{46} - h_{58} + h_{76}$$
$$c_{4,4} = h_{12} + h_{25} + h_{26} - h_{33} - h_{35} - h_{40} + h_{41} + h_{65} - h_{68} - h_7$$
$$c_{1,5} = h_{15} + h_{24} + h_{25} + h_{27} - h_{28} + h_{30} + h_{31} - h_4 + h_{61} + h_{64}$$
$$c_{2,5} = -h_{10} - h_{18} - h_2 - h_{30} - h_{38} + h_{42} - h_{43} + h_{46} + h_{67} + h_{74}$$
$$c_{3,5} = -h_{10} + h_{12} - h_{15} + h_{28} + h_{29} - h_2 - h_{30} - h_3 + h_{46} + h_4 - h_{75}$$
$$c_{4,5} = -h_{12} - h_{29} + h_{30} - h_{34} + h_{35} + h_{39} + h_3 - h_{45} + h_{57} + h_{59}$$

$$h_1 \ldots h_{76}$$

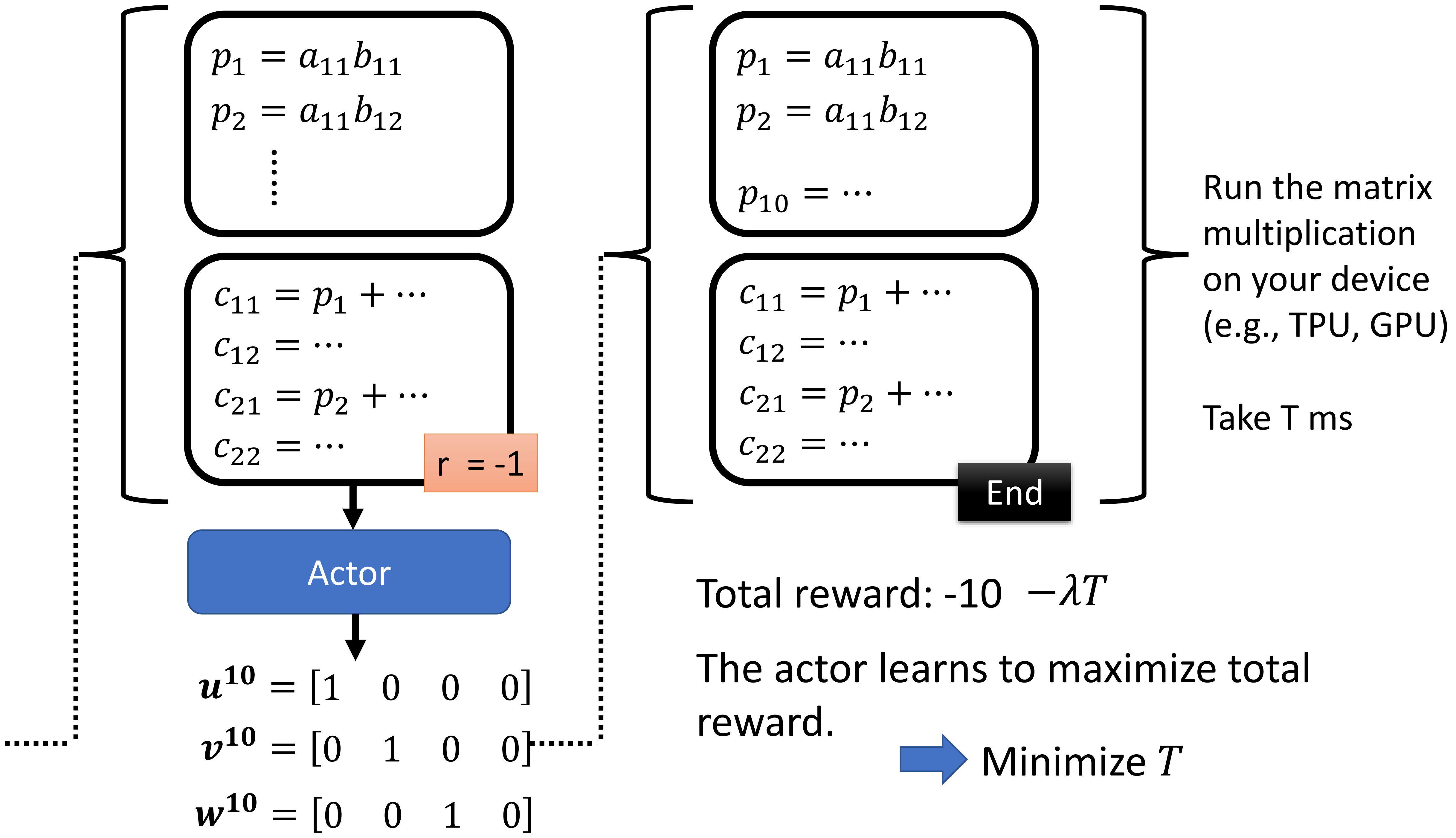
$$\begin{pmatrix} c_{1,1} & c_{1,2} & c_{1,3} & c_{1,4} & c_{1,5} \\ c_{2,1} & c_{2,2} & c_{2,3} & c_{2,4} & c_{2,5} \\ c_{3,1} & c_{3,2} & c_{3,3} & c_{3,4} & c_{3,5} \\ c_{4,1} & c_{4,2} & c_{4,3} & c_{4,4} & c_{4,5} \end{pmatrix}$$

結果如何?

- 稍微再大一點的矩陣
  - Using **block multiplication**



接下來把實際硬體的效能考慮進去

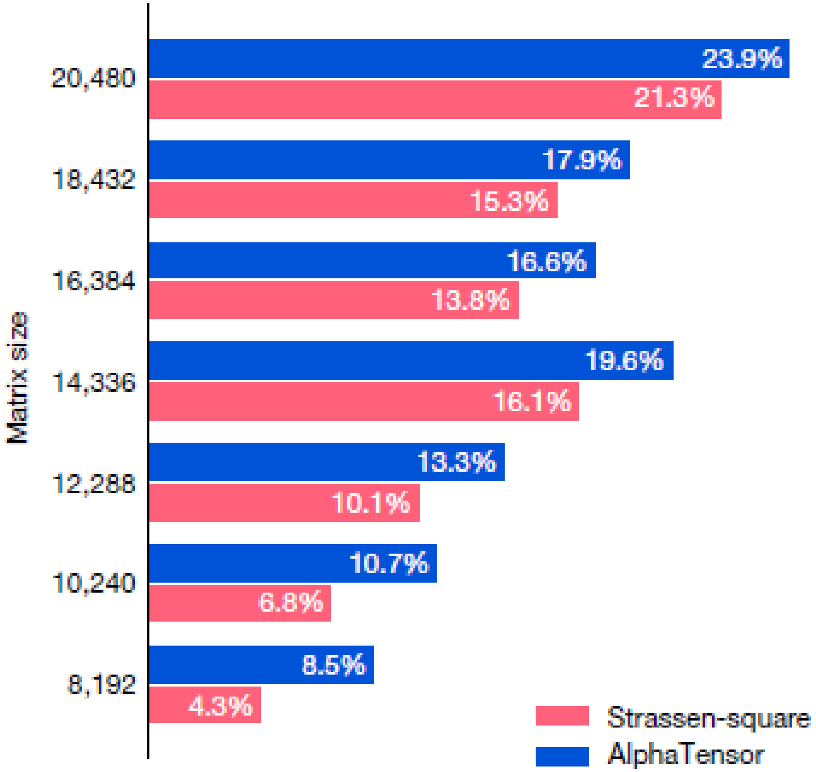

$p_1 = a_{11}b_{11}$
$p_2 = a_{11}b_{12}$
$c_{11} = p_1 + \cdots$
$c_{12} = \cdots$
$c_{21} = p_2 + \cdots$
$c_{22} = \cdots$
r = -1
Actor
$\boldsymbol{u}^{10} = [1 \quad 0 \quad 0 \quad 0]$
$\boldsymbol{v}^{10} = [0 \quad 1 \quad 0 \quad 0]$
$\boldsymbol{w}^{10} = [0 \quad 0 \quad 1 \quad 0]$
$p_1 = a_{11}b_{11}$
$p_2 = a_{11}b_{12}$
$p_{10} = \cdots$
$c_{11} = p_1 + \cdots$
$c_{12} = \cdots$
$c_{21} = p_2 + \cdots$
$c_{22} = \cdots$
End
Run the matrix multiplication on your device (e.g., TPU, GPU)
Take T ms
Total reward: -10 $-\lambda T$
The actor learns to maximize total reward.
Minimize T

結果如何?

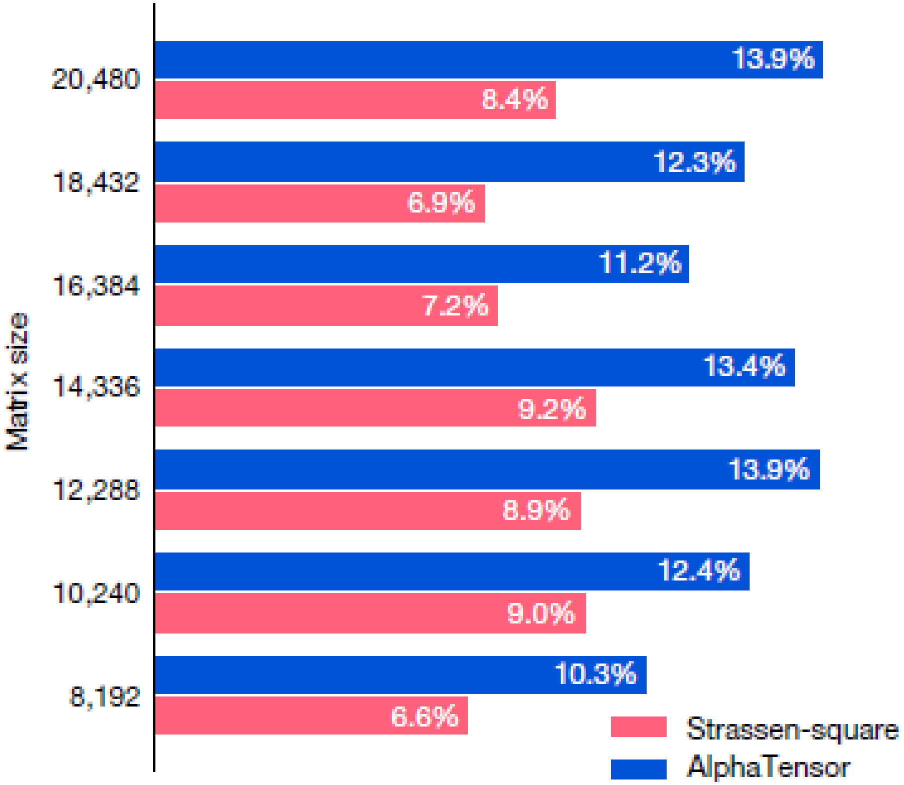
Block multiplication

No recursive

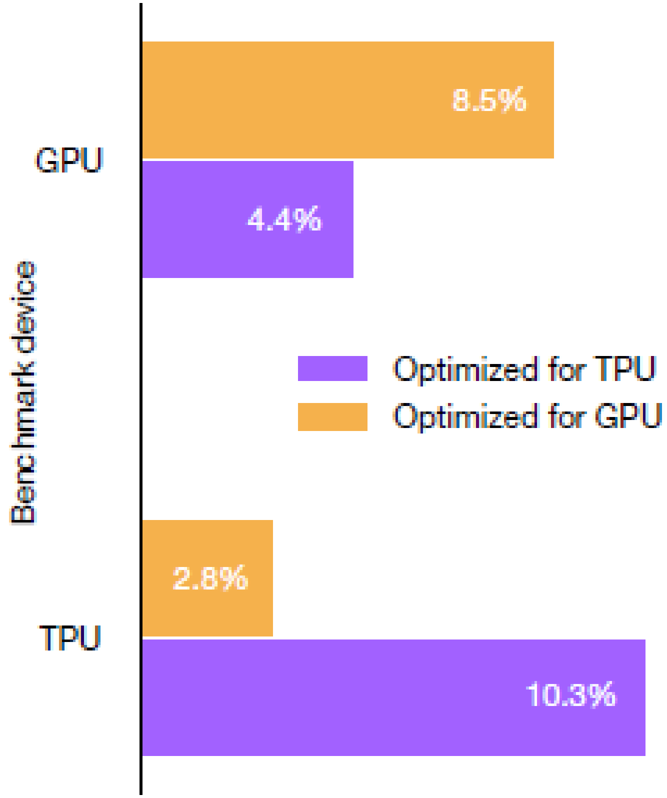
$$A = \begin{array}{|c|c|c|c|}\hline & & & \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline \end{array} \qquad B = \begin{array}{|c|c|c|c|}\hline & & & \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline \end{array}$$



Speed-up on Nvidia V100 GPU



Speed-up on TPU v2



Speed-up of tailored agorithms
on both devices

結語

想找有效率的矩陣相乘演算法

用增強式學習硬 train

還真的找到一些更有效率的演算法

不只是硬 train 一發而已

- Using supervised learning
- Design of network architecture
- Change of basis