39th 2022全国青少年信息学奥林匹克冬令营
National Olympiad in Informatics Winter Camp

第一课堂

杂谈：多重时空下的算法竞赛

罗哲正
麻省理工学院

什么是算法竞赛?

- 考察算法
- 规定时间内编程解决数学问题
- 测试数据
- 时间/空间限制

什么是OI

OI 即 Olympiad in Informatics，是高中阶段主要的算法竞赛，有较为明确考纲（如 NOI，IOI 大纲）

- 多为4~5小时解决3~4道难度不同的题
  较为正式的比赛中为5小时三道题
- 每题都有明确标注的部分分
- IOI 赛制有及时反馈
  CNOI 赛制无及时反馈，赛后统一评测

OI 以外的算法竞赛

区域赛　　CCPC

World Finals

ICPC系列

Atcoder UOJ Codechef

Codeforces Topcoder

… …

短周期网络赛

蓝桥杯

百度之星　　美团杯

… …

线下比赛

Topcoder Open (TCO)

Google Code Jam (GCJ)

HashCode … …

长周期网络赛
(年度)

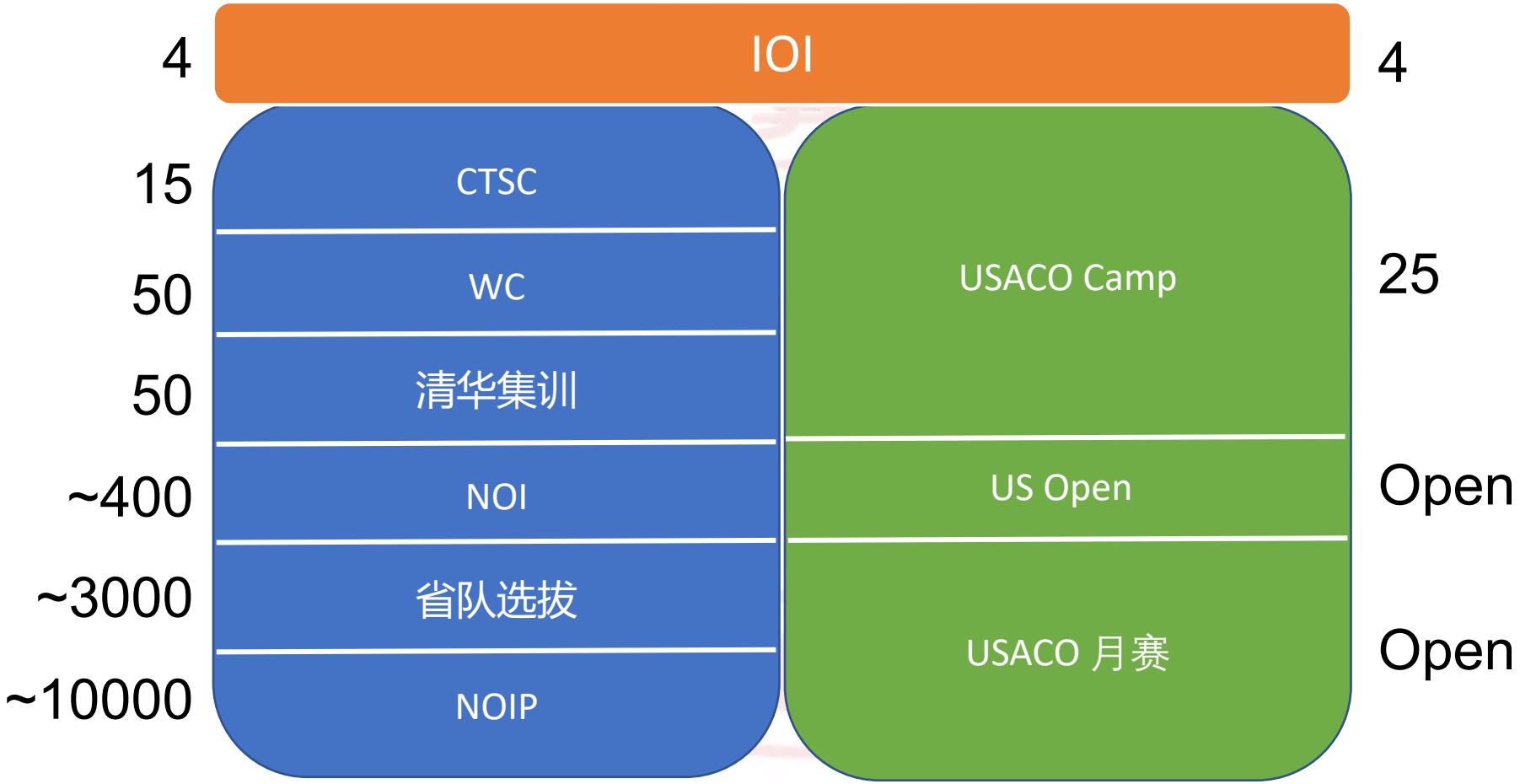
多种赛制：ICPC，CF，提交答案

大洋彼岸的启示

从交换生经历谈中美信息学竞赛差异

我的高三到底做了啥?

选拔


4
IOI
4
15
CTSC
50
WC
50
清华集训
~400
NOI
~3000
省队选拔
~10000
NOIP
USACO Camp
25
US Open
Open
USACO 月赛
Open

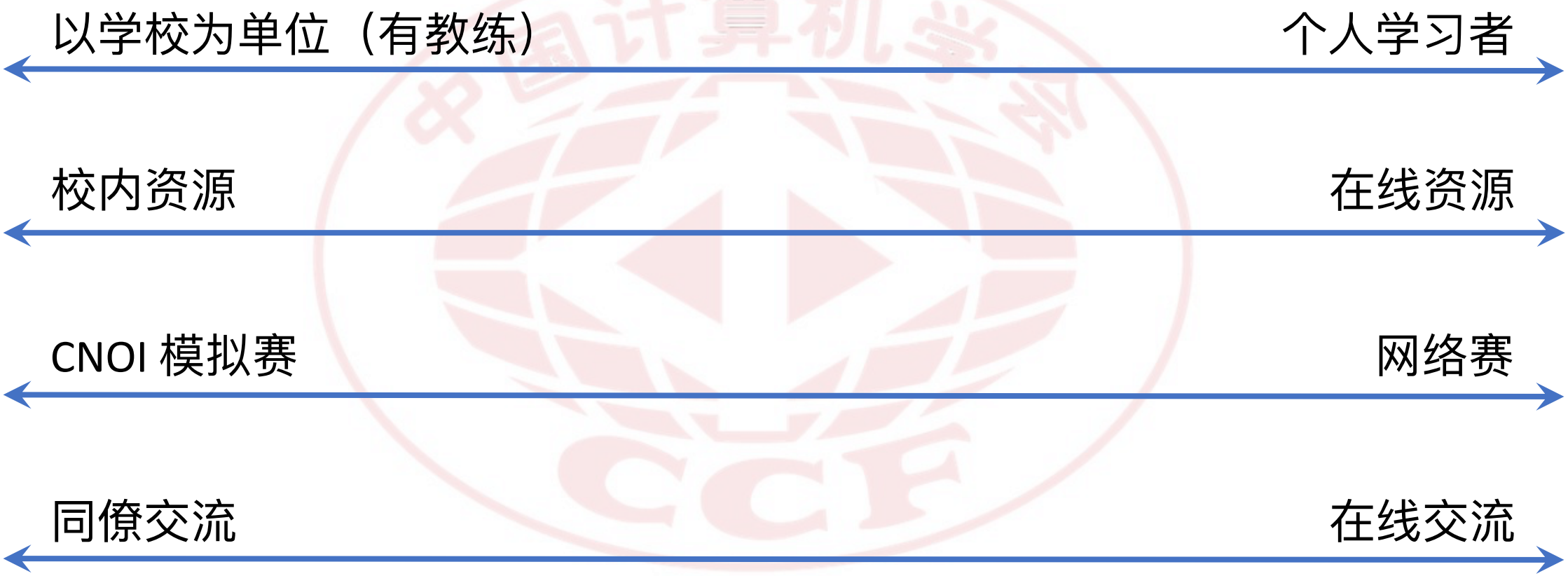
赛制

CNOI	USACO	IOI
按测试点给分	按测试点给分	按子任务给分
文件输入输出	标准输入输出	交互式输入输出
赛后统一评测	即时评测	即时评测
有详细部分分	有粗略部分分	严格按照子任务给部分分

命题特点与选手科技树

- IOI 的命题精神是什么？
  - *请认真听一听彭老师《IOI 题型与去向分析》*
- 我们的命题与组题科学吗？USACO 这方面如何？
  - 谈谈你印象中最“偏”的一道正式比赛题。
  - 谈谈你见过的好题。
  - 谈谈你心中最优秀的一场比赛。
  - USACO 的题是什么样的？
  - 你觉得近年的比赛难度在如何变化？应该如此吗？
- 我们的科技领先吗？领先在哪里？
  - 谈谈你认为最惊艳的中国 OI 科技。

训练模式



总结与建议

对比国外选手的训练选拔方式，我们可以学到什么？

如何训练（面向 IOI）

面向IOI的训练模式

- 可以继续通过模拟赛训练，但注意不要过度拟合 CNOI 赛制
- 利用网络在线比赛适应即时反馈的赛制
- 提高整体的做题效率，IOI 赛制的节奏远快于 CNOI 赛制

资源的选择与利用

- 认真对待且珍惜 IOI 真题，尤其是交互类型
- 刷题时侧重 IOI 考纲内的题目
- 其他国家的决赛，选拔赛真题（如美国，日本，波兰）
- ICPC 训练赛可极大提升综合能力，可以尝试

如何考试（面向 IOI）

时间管理

减小方差

擅于挑战

- 利用即时反馈尽快拿到暴力/部分分，不要优化暴力
- 优先做得分率高的部分分，平时注意提升代码稳定性
- 提前 90 分钟做好全部收场规划
- 善于从 low-level 写码状态切换回 high-level 思考状态
- 谨慎但不要惧怕投入大量时间思考
- 训练对题目难度的感知

如何选拔

- 尽可能采用 IOI 赛制。
- 增多选拔赛的题目数量，平均权值（需要更多成本投入）。
- 增加 CNOI 中缺乏的题型与知识点的考察，如交互。

着眼未来：OI 与大学

包含我对一些 OI 选手的采访

中学阶段：OI 的投入与回报

- OI 与文化课的冲突：大多数集训队员都不同程度的停了文化课。
- OI 对升学的价值：加分，保送，申请时作为重要的课外活动。

OI 的学习方式

- 自主学习能力
- 自主在网上找资源的能力（如 blog，google scholar）
- 快速阅读并筛选资源的能力
- 不要硬想问题，学会从题解中学习
- 做题往往比看书有用

OI 的思维模式

- 逻辑思维能力
- 建立对题目难度的良好估计
- 模块化的拆分问题
- 先考虑问题的特殊情况，看看能不能把做法一般化
- 先暴力解决问题再思考优化
- 学会问“为什么想到这个”，关注直觉（intuition）

OI 中的知识点

- 代码能力，调试技巧，常数优化
- 可以快速学会一门编程语言，并看懂他人的代码
- 算法思维：把计算过程写成算法，提升理解复杂算法的能力
- 分析算法的复杂度及正确性
- 概率，组合与线性代数
- 初级而广泛应用的思想：贪心，动态规划，分治，递归

OI in Research

- Theory
- System
- Artificial Intelligence and Machine Learning
- Other areas

OI in my Research

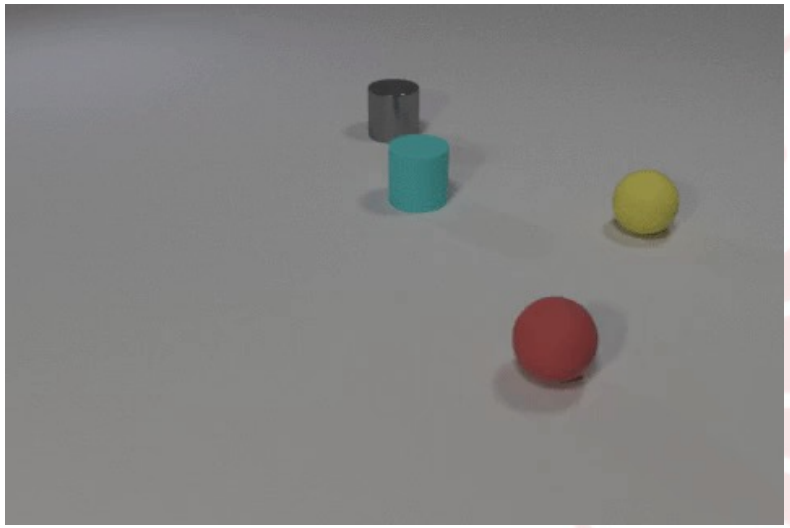
- Temporal and Object Quantification Networks

  Jiayuan Mao*, Zhezheng Luo*, Chuang Gan, Joshua B. Tenenbaum, Jiajun Wu, Leslie Pack Kaelbling, and Tomer D. Ullman (*: First two authors contributed equally.)

- Project Page:
  - http://toqnet.csail.mit.edu/

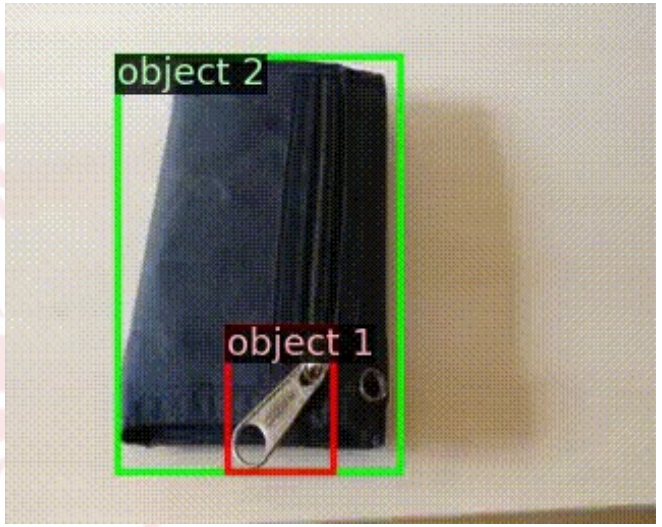
Scan Me

Event Grounding



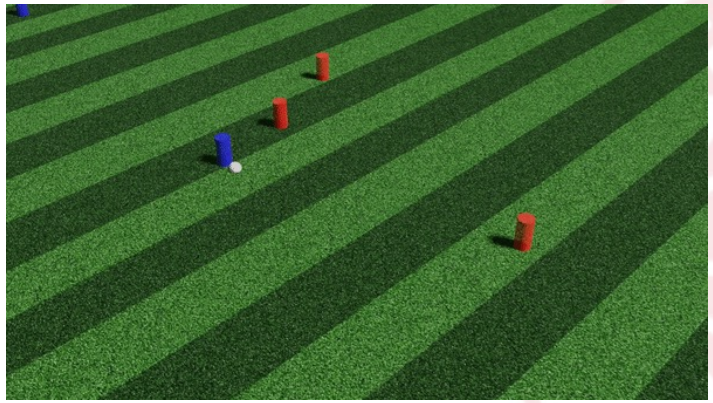
Collision

Grasp

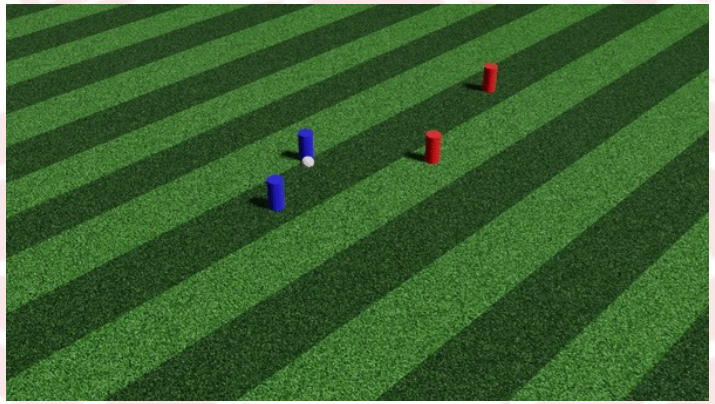


Flip

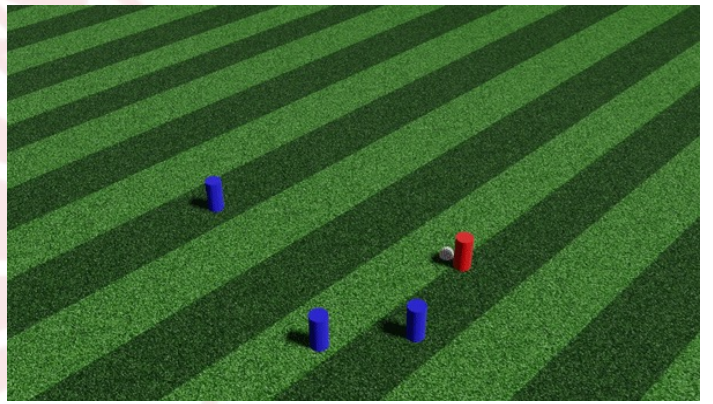
Event Recognition in Soccer Game



Shot

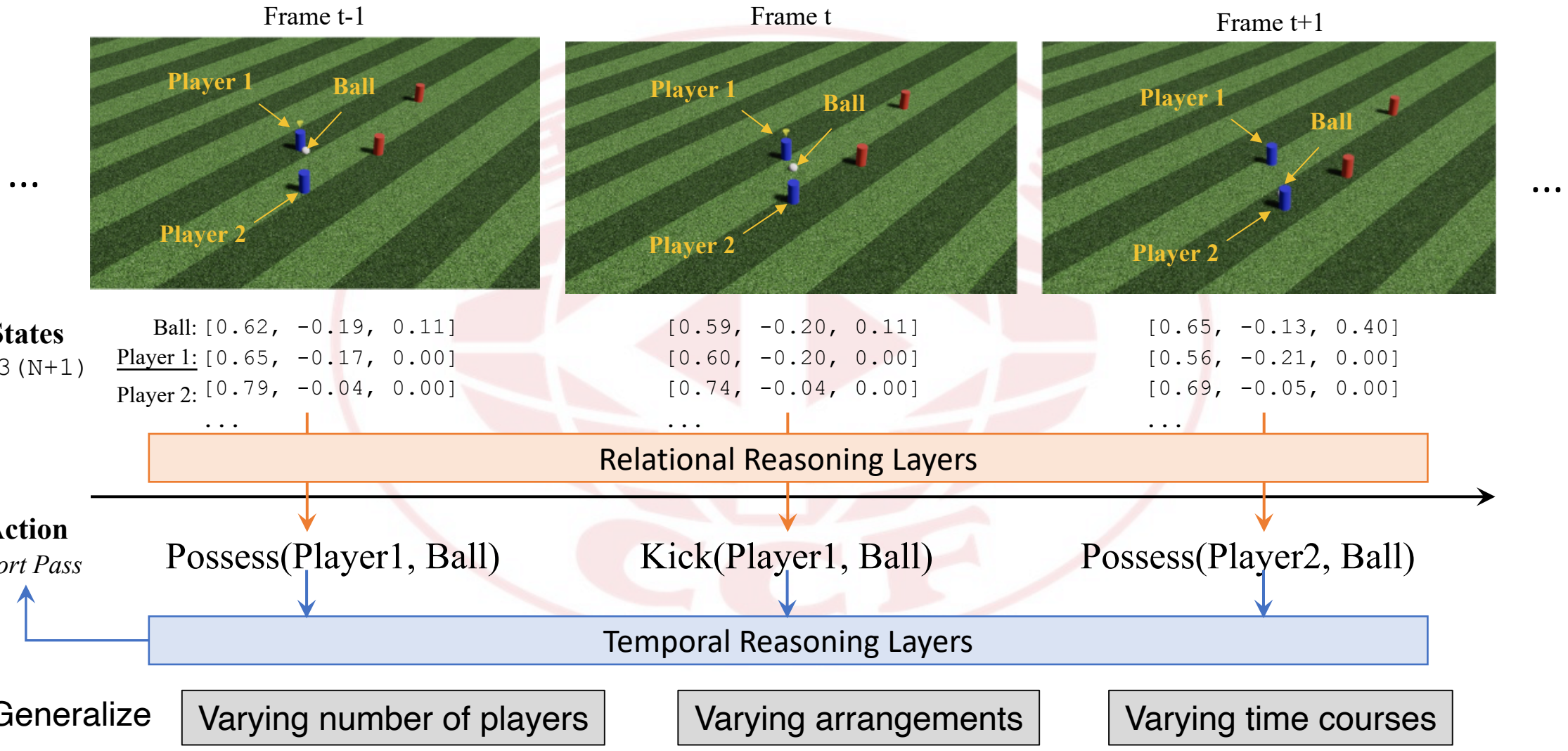


Short pass

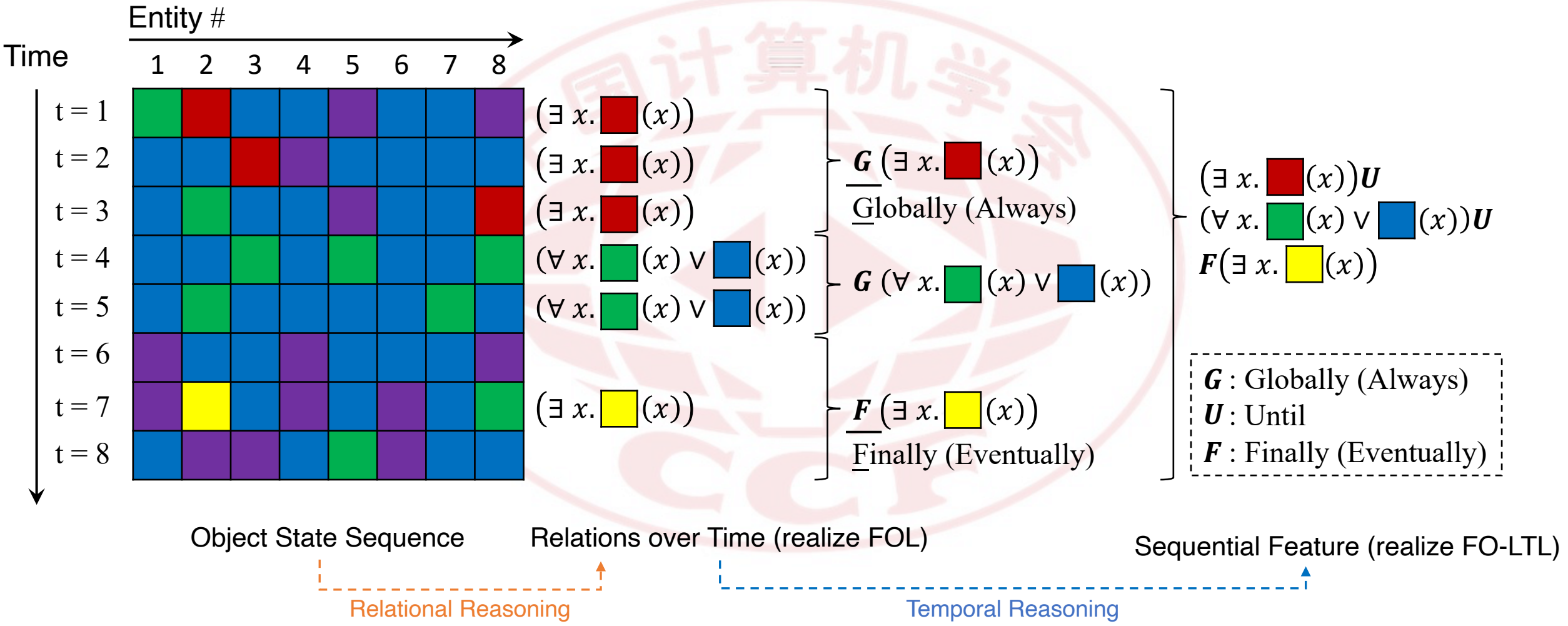


Deflect

Relational and Temporal Action Concept



Action Concept Representation — First-Order-Logic (FOL) and Linear-Temporal-Logic (LTL)



Relational Reasoning – Feature Representation

We store relational features at each time t.
Internal relational features are represented by tensors.

$\mathrm{red}(x,t)$:
x is a red team player

$\mathrm{blue}(x,t)$:
x is a blue team player

$\mathrm{ball}(x,t)$:
x is a the ball

$\mathrm{close}(x,y,t)$:
x and y are close to each other (distance < 1m) at time t

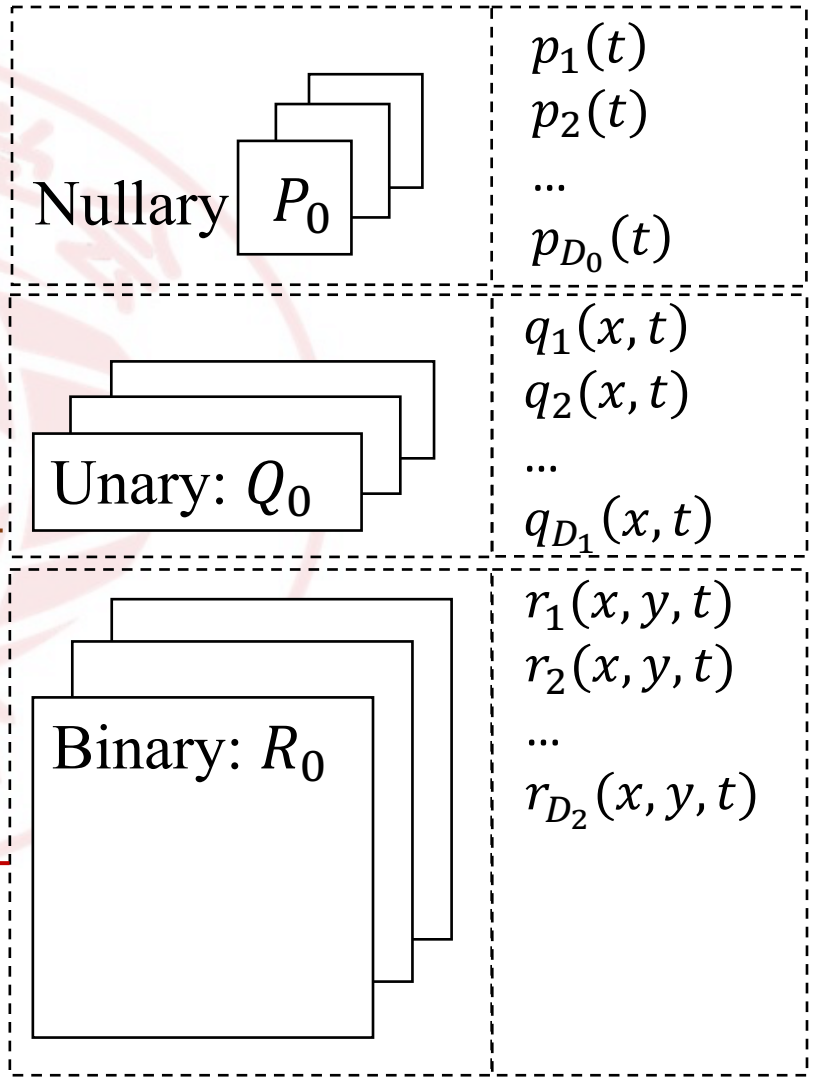
...

Globally Features (Nullary)
Dimension: D_0

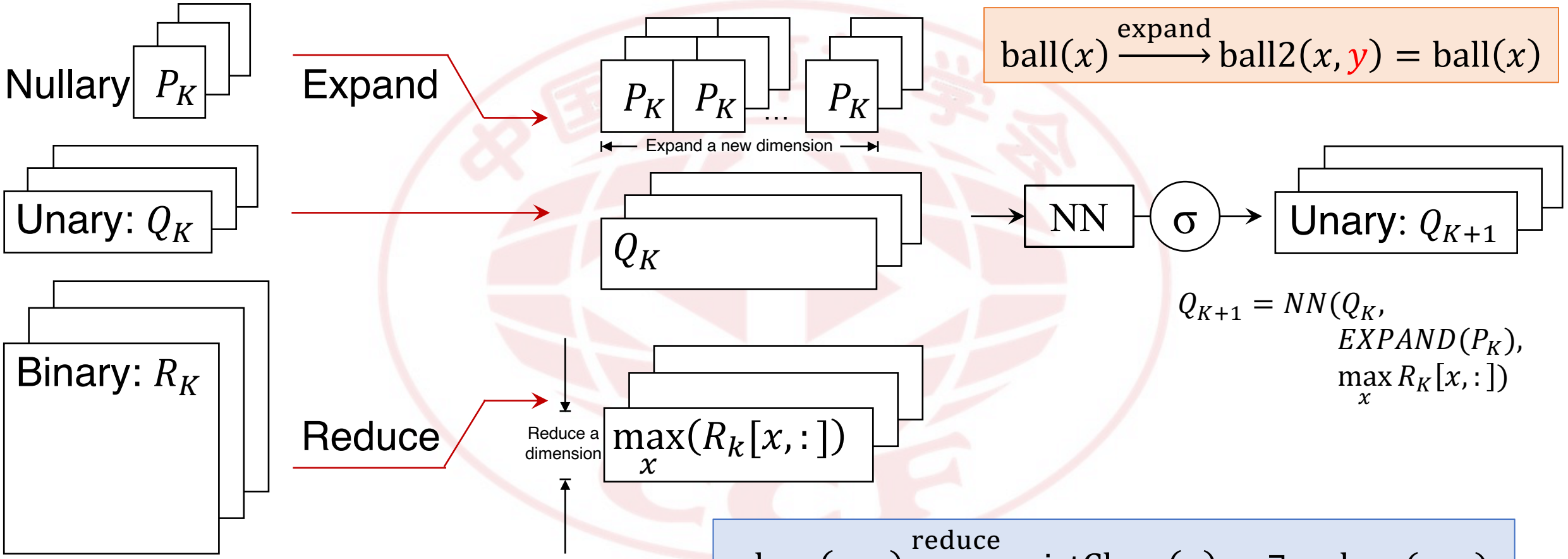
Object Features (Unary)
Dimension: $N \times D_1$

Object Pair Features (Binary)
Dimension: $N \times N \times D_2$

…



Relational Reasoning Layer – Expander and Reducer

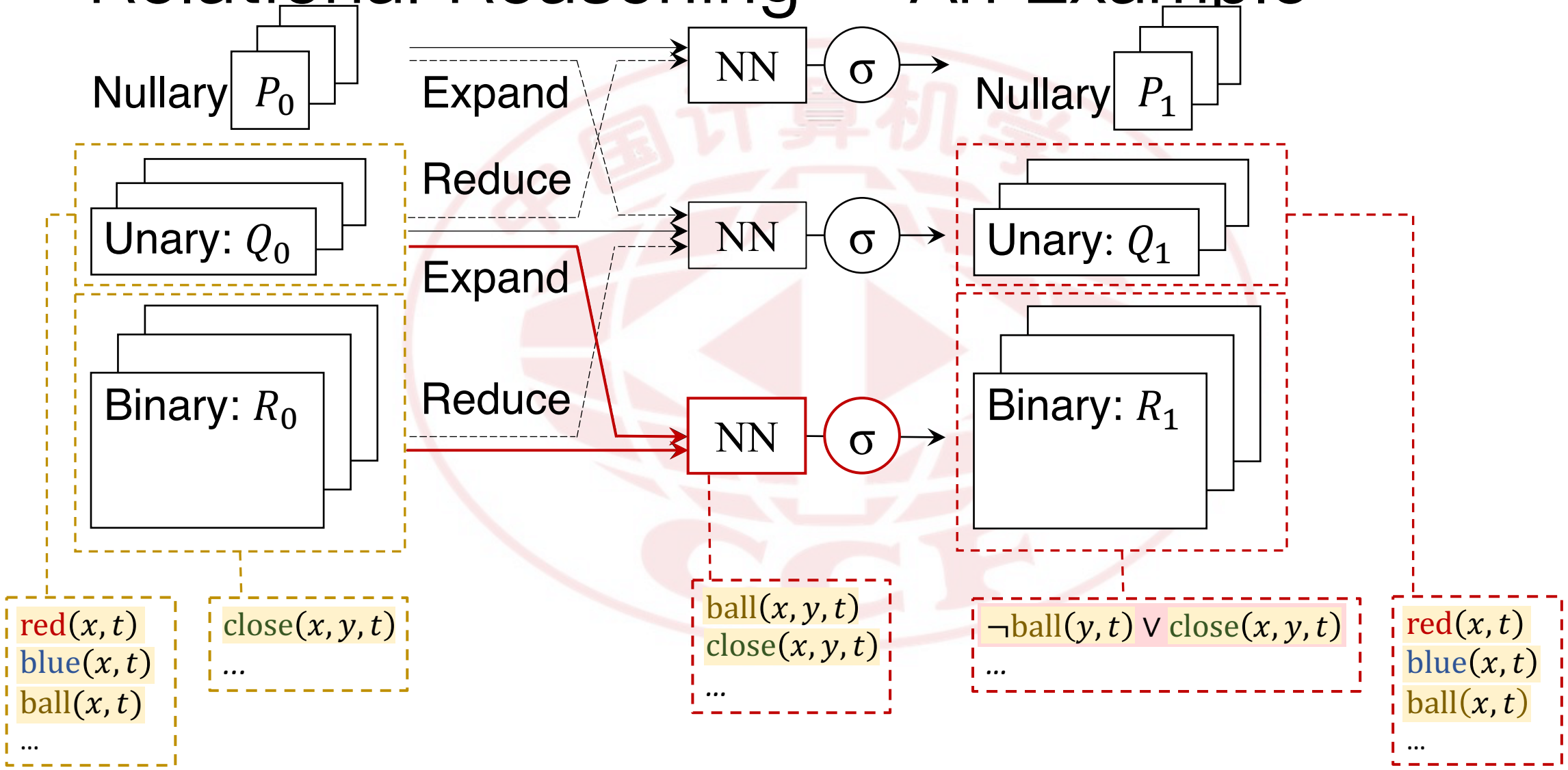


$$\text{ball}(x) \xrightarrow{\text{expand}} \text{ball2}(x, y) = \text{ball}(x)$$

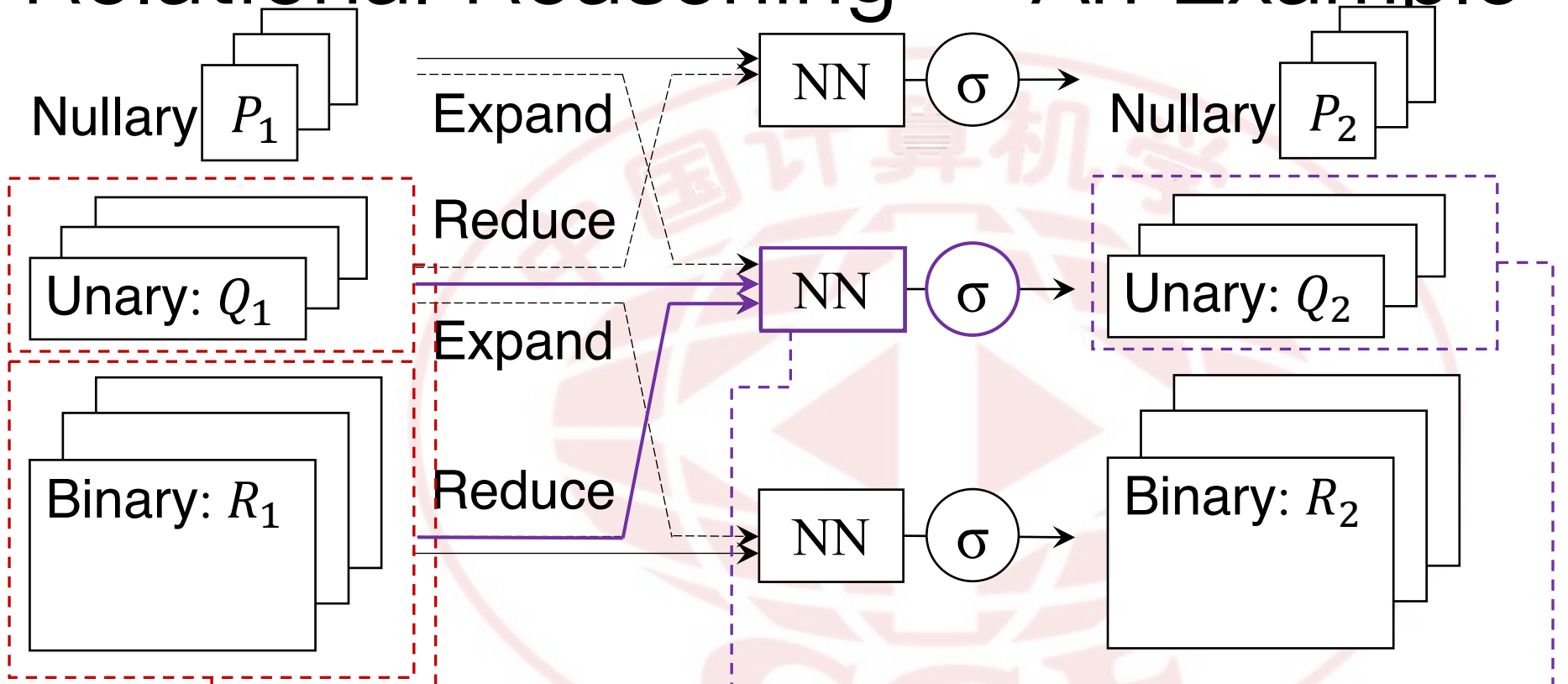
$$Q_{K+1} = NN(Q_K, EXPAND(P_K), \max_x R_K[x,:])$$

$$\text{close}(x, y) \xrightarrow{\text{reduce}} \text{existClose}(y) = \exists x.\, \text{close}(x, y)$$

Relational Reasoning – An Example



Relational Reasoning – An Example



$\neg \text{ball}(y,t) \vee \text{close}(x,y,t)$
…

$\text{red}(x,t)$
$\text{blue}(x,t)$
$\text{ball}(x,t)$
…

$\forall y.\ \neg \text{ball}(y,t) \vee \text{close}(x,y,t)$
$\text{red}(x,t)$
…

$\text{red}(x,t) \wedge \forall y.\ \neg \text{ball}(y,t) \vee \text{close}(x,y,t)$
…

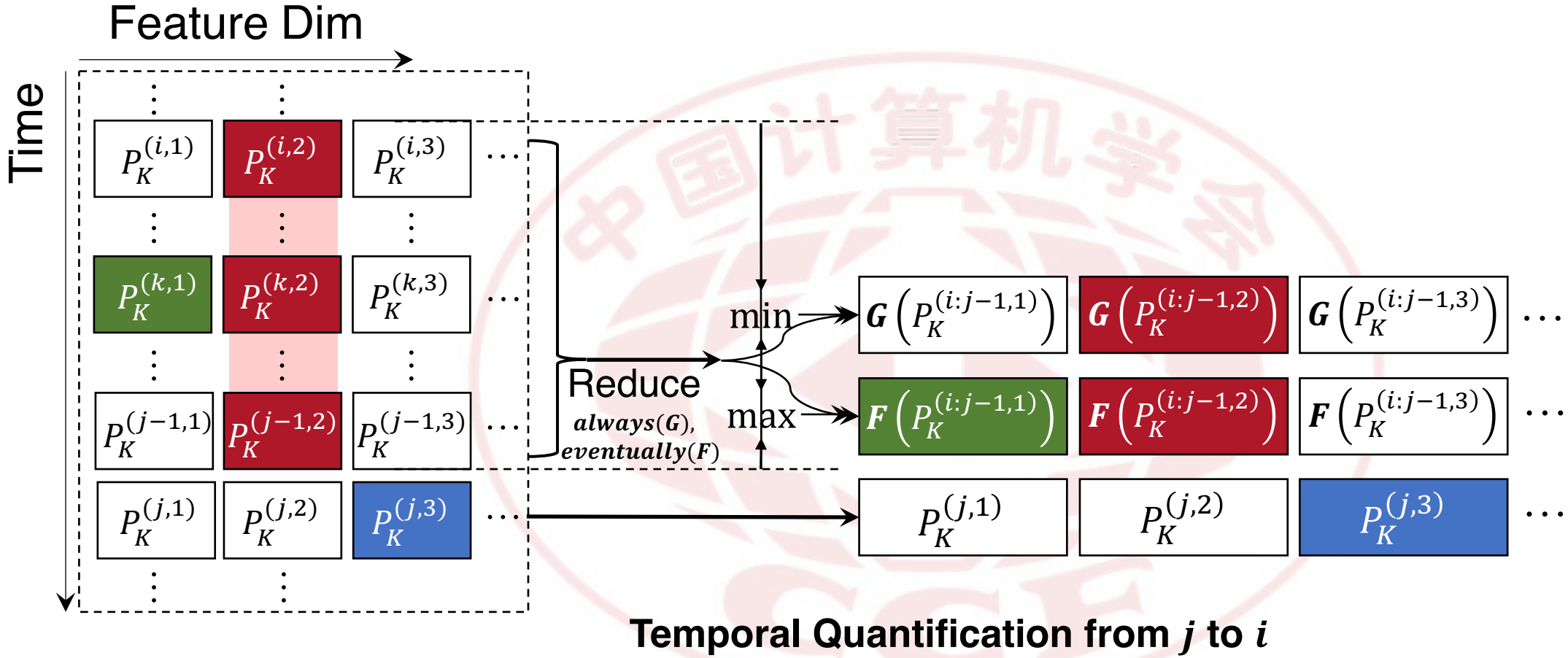
如何抽取时序关系?

- $possess \rightarrow kick \rightarrow otherPlayerGet$

- $f_1[t] = otherPlayerGet[t]$
- $f_2[t_1] = \max\limits_{t_2>t_1}\{f_1[t_2]\ \&\&\ \max\limits_{t_1\le k<t_2} kick[k]\}$
- $f_3[t_1] = \max\limits_{t_2>t_1}\{f_2[t_2]\ \&\&\ \max\limits_{t_1\le k<t_2} possess[k]\}$

- $f[i][t_1] = \max_{t_2>t_1} func1\big(f[i-1][t_2], \max_{t_1\le k<t_2} f[i-1][k]\big)$

Temporal Reasoning — Temporal Quantification

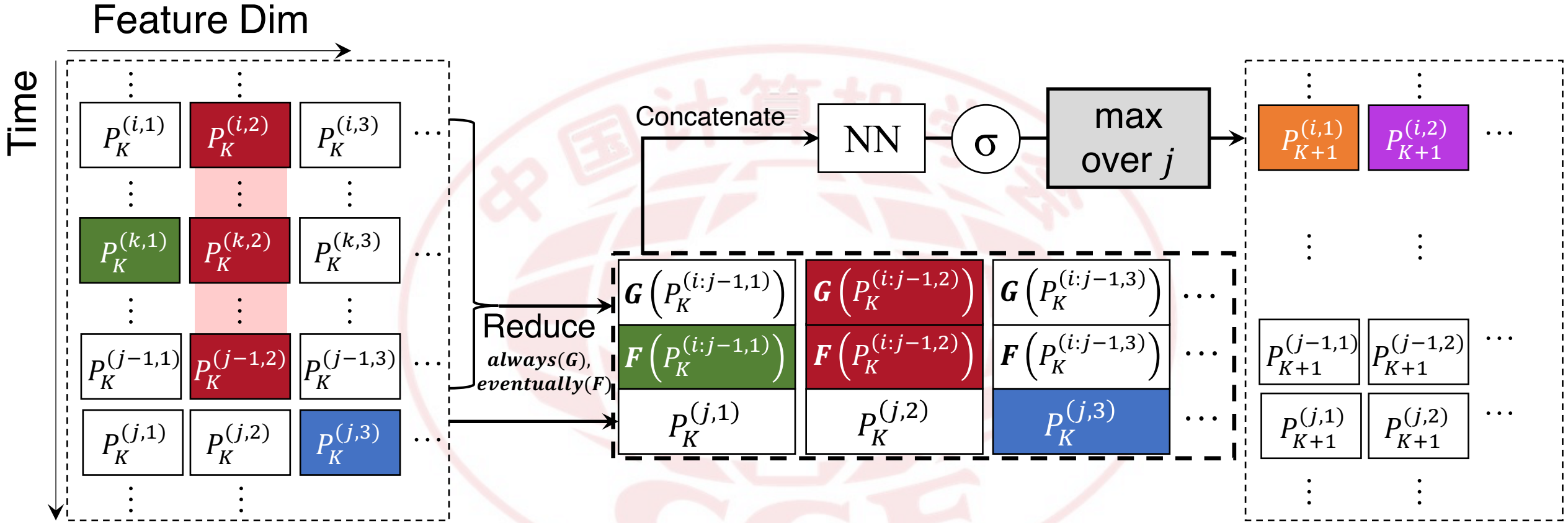


Temporal Quantification from j to i

$P_K^{(t,1)}$: e_1

$P_K^{(t,2)}$: e_2

Temporal Reasoning — Temporal Quantification



Temporal Quantification from j to i

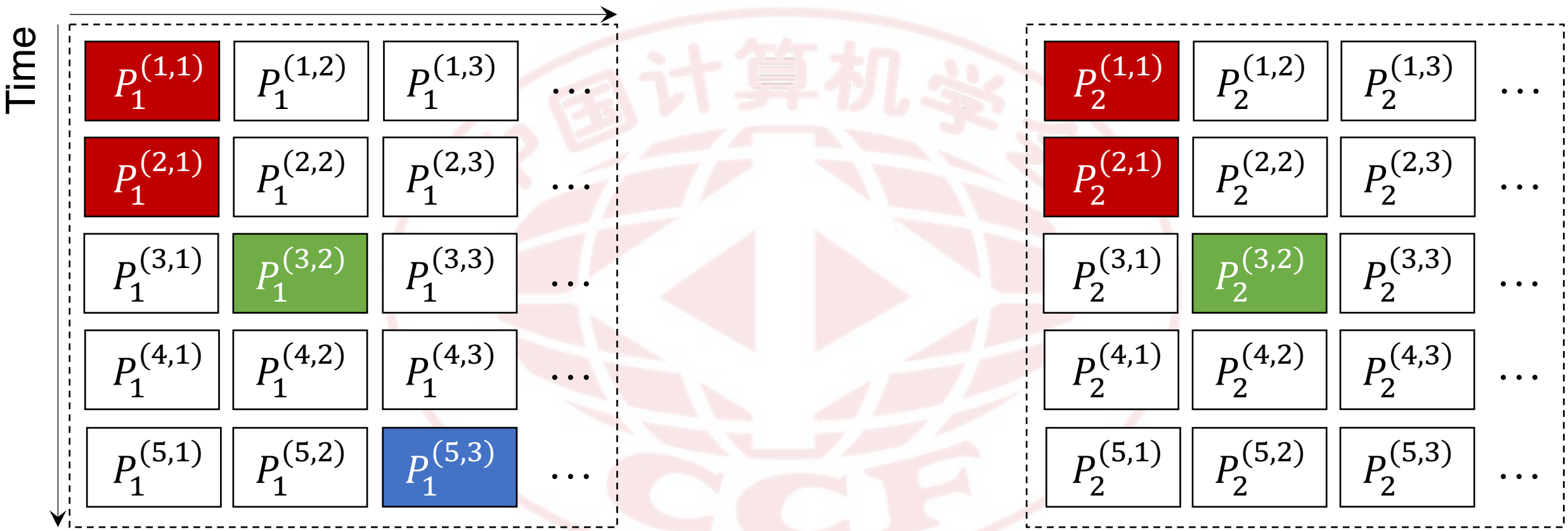
$P_K^{(t,1)} : e_1$

$P_K^{(t,2)} : e_2$

$e_3(i) = (e_1 \ \textbf{until} \ e_2)(i) \Leftrightarrow \exists j > i. \left(\boldsymbol{G}_{i:j-1} e_1\right) \wedge e_2(j)$

$e_4(i) = (e_1 \ \textbf{then} \ e_2)(i) \Leftrightarrow \exists j > i. \left(\boldsymbol{F}_{i:j-1} e_1\right) \wedge e_2(j)$

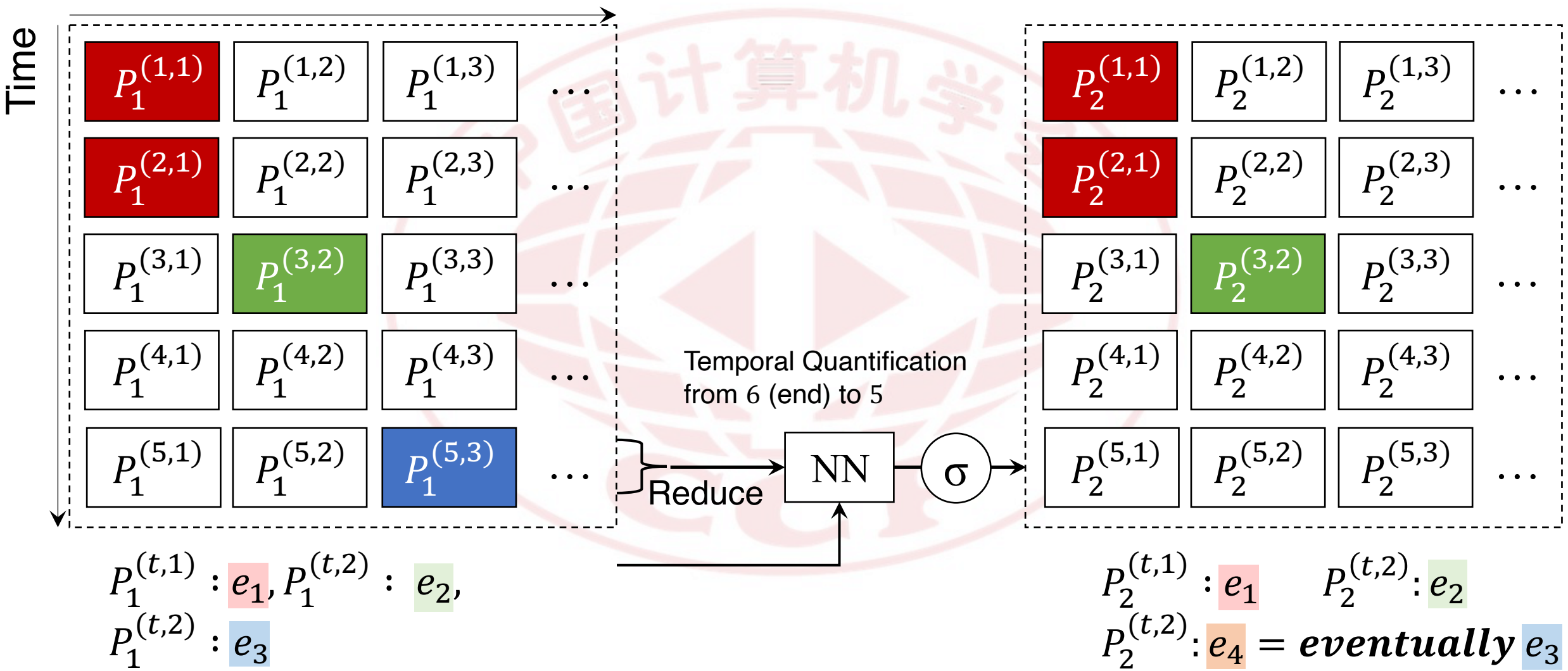
Temporal Reasoning — Temporal Quantification



$P_1^{(t,1)} : e_1, P_1^{(t,2)} : e_2,$
$P_1^{(t,2)} : e_3$

$P_2^{(t,1)} : e_1 \quad P_2^{(t,2)} : e_2$
$P_2^{(t,2)} : e_4 = \boldsymbol{eventually}\ e_3$

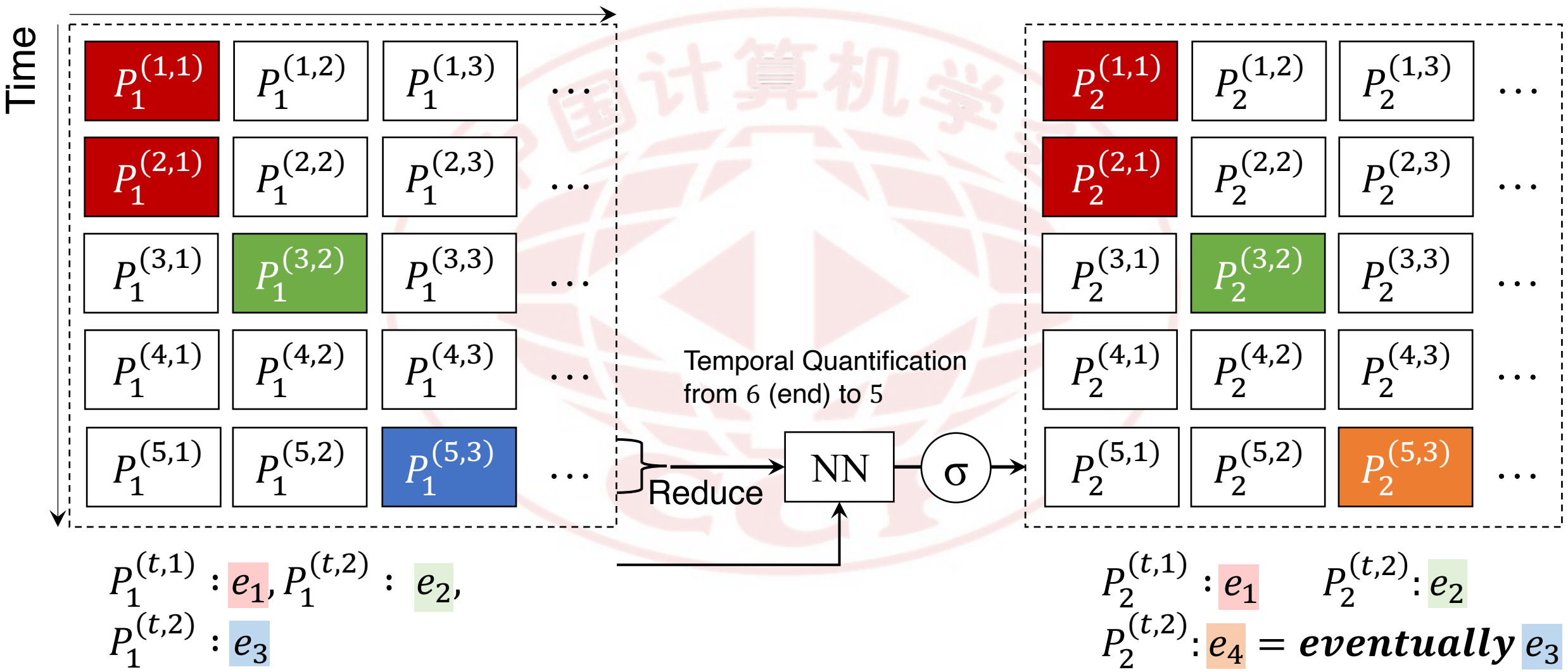
Temporal Reasoning — Temporal Quantification



$P_1^{(t,1)} : e_1, P_1^{(t,2)} : e_2,$
$P_1^{(t,2)} : e_3$

$P_2^{(t,1)} : e_1 \quad P_2^{(t,2)} : e_2$
$P_2^{(t,2)} : e_4 = \boldsymbol{eventually}\ e_3$

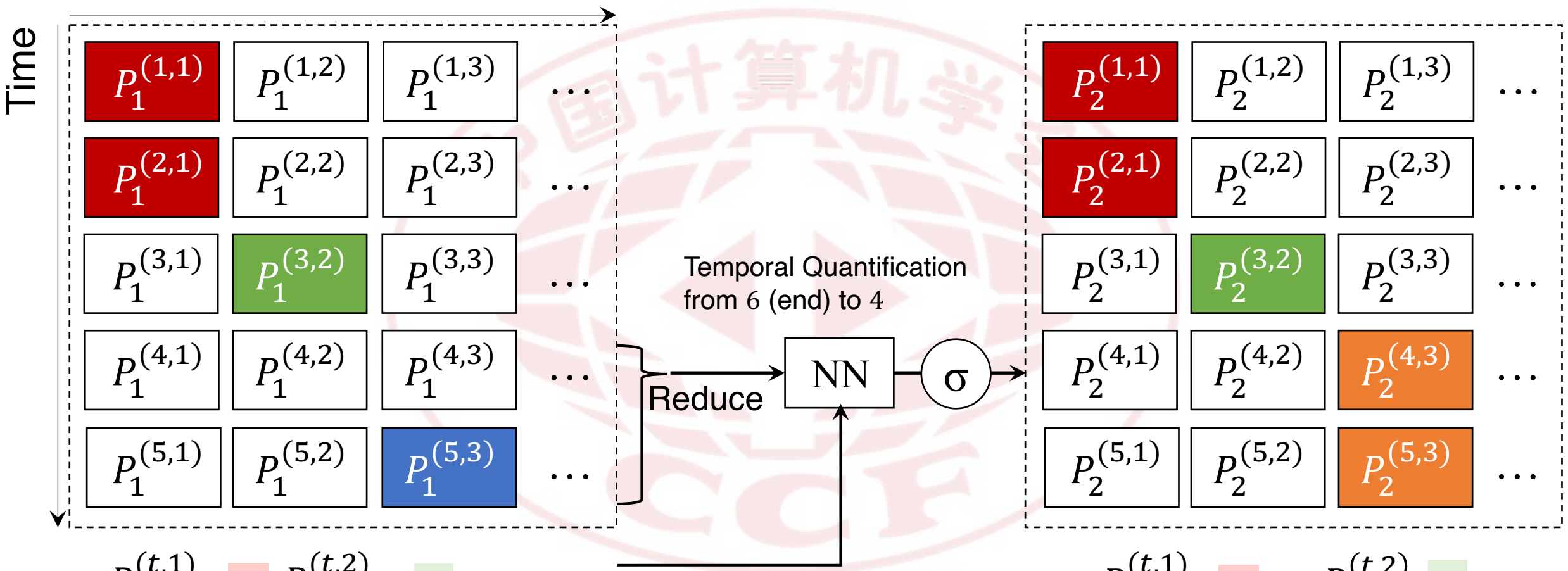
Temporal Reasoning — Temporal Quantification



$P_1^{(t,1)} : e_1, P_1^{(t,2)} : e_2,$
$P_1^{(t,2)} : e_3$

$P_2^{(t,1)} : e_1 \quad P_2^{(t,2)} : e_2$
$P_2^{(t,2)} : e_4 = \boldsymbol{eventually}\ e_3$

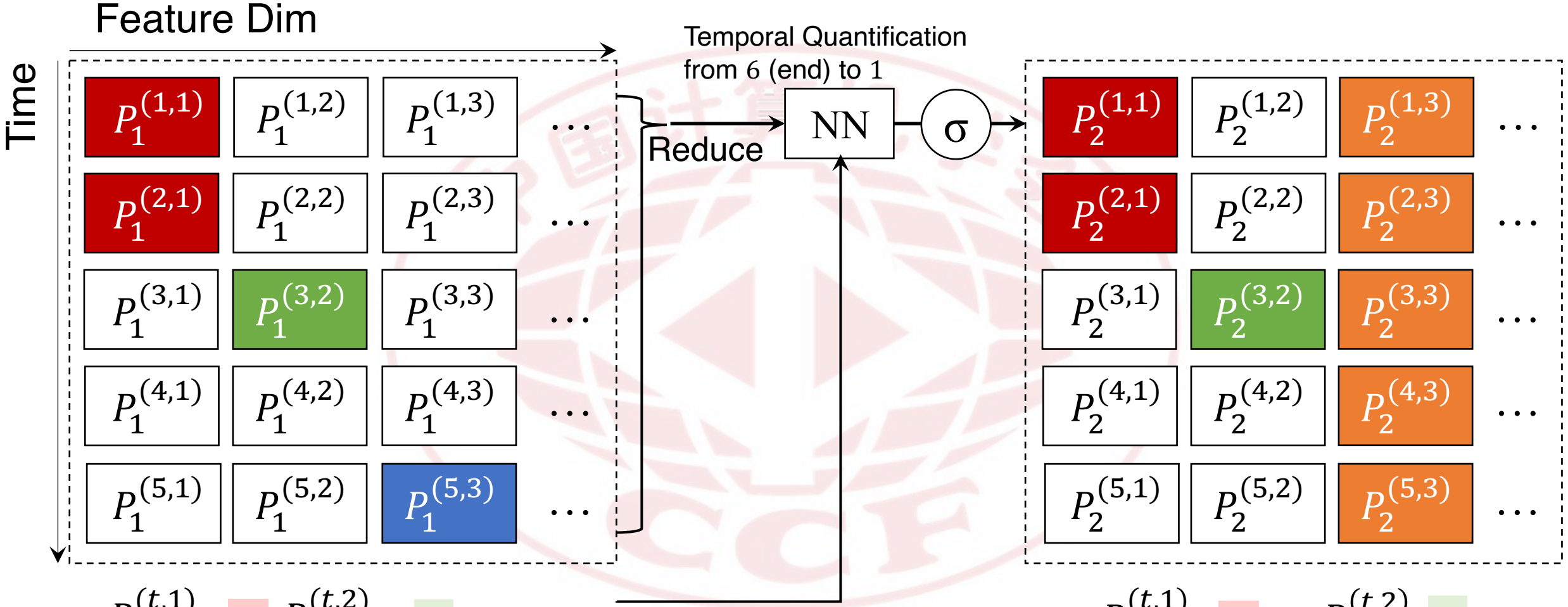
Temporal Reasoning — Temporal Quantification



$P_1^{(t,1)} : e_1, P_1^{(t,2)} : e_2,$
$P_1^{(t,2)} : e_3$

$P_2^{(t,1)} : e_1 \quad P_2^{(t,2)} : e_2$
$P_2^{(t,2)} : e_4 = \boldsymbol{eventually}\ e_3$

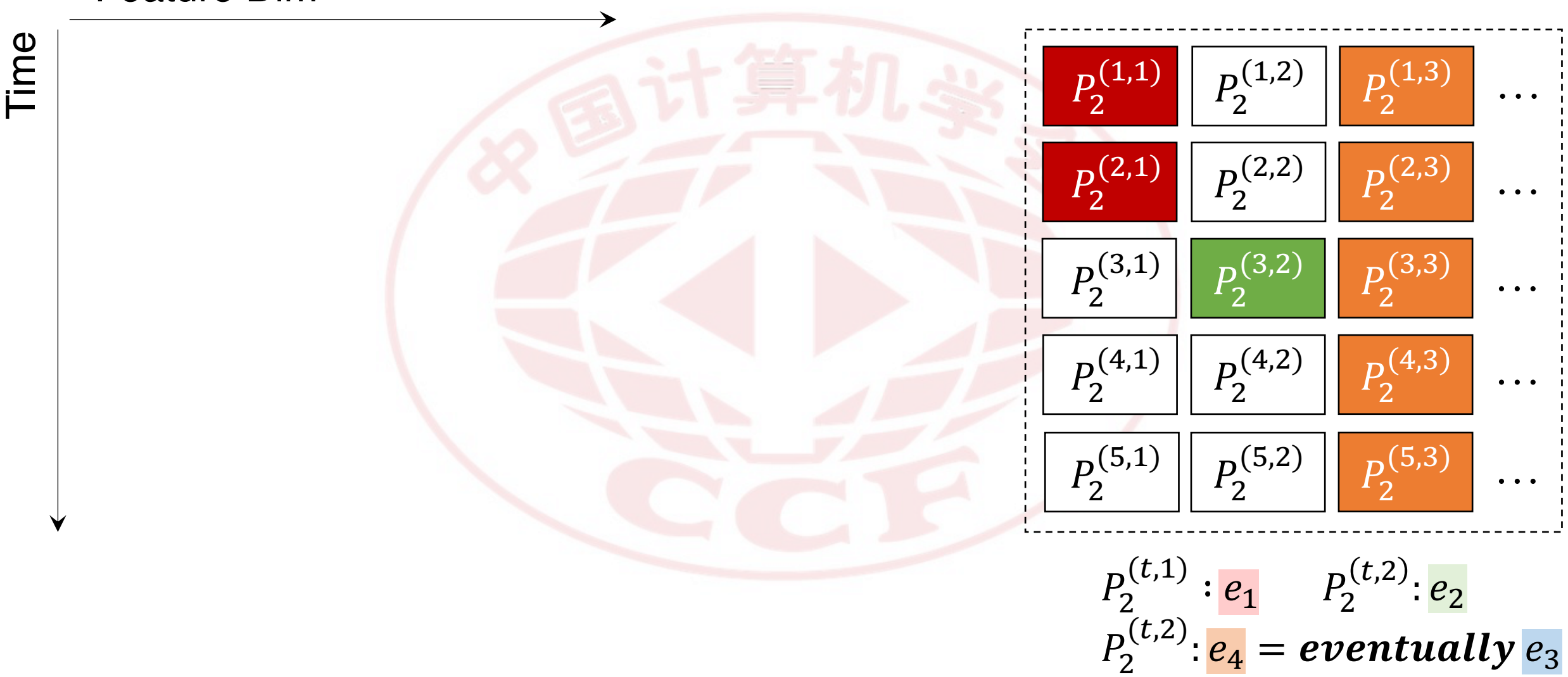
Temporal Reasoning — Temporal Quantification



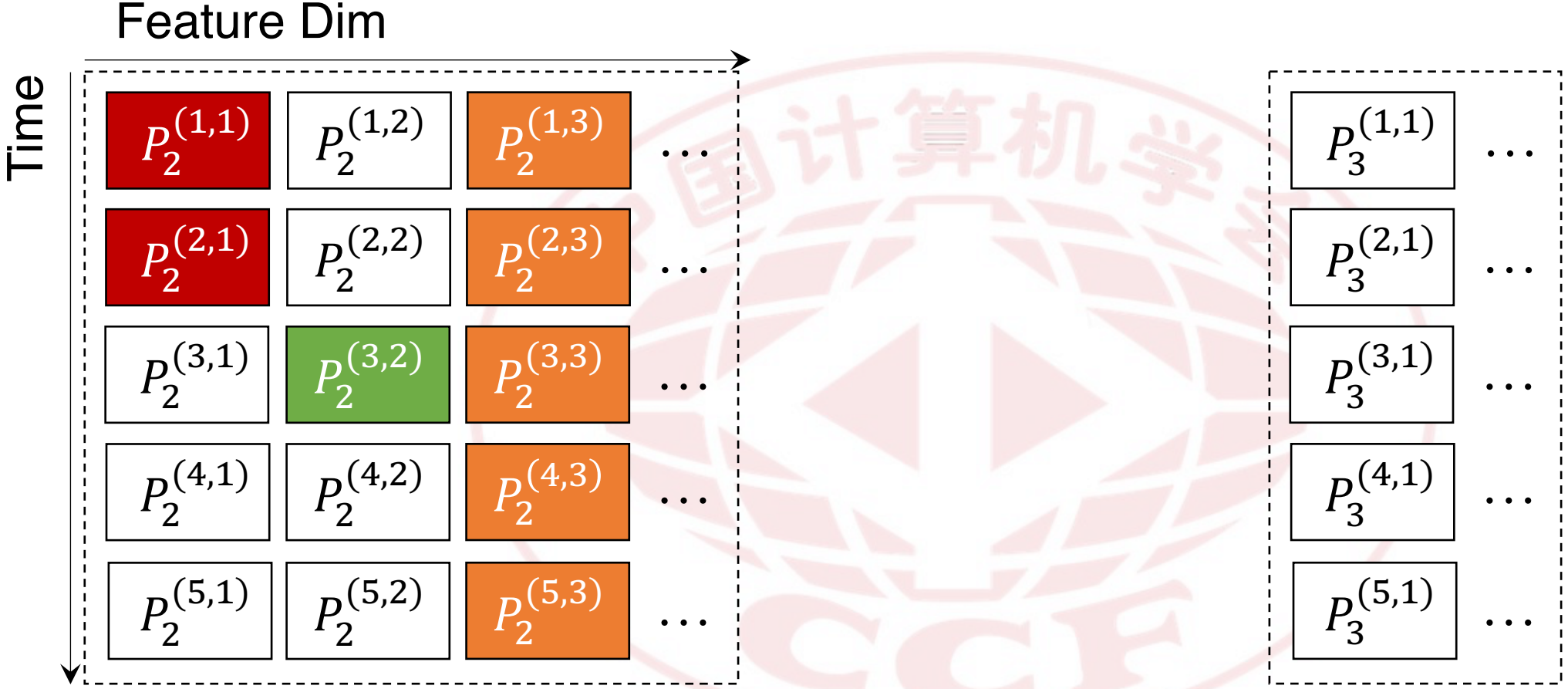
$P_1^{(t,1)} : e_1, P_1^{(t,2)} : e_2,$
$P_1^{(t,2)} : e_3$

$P_2^{(t,1)} : e_1 \quad P_2^{(t,2)} : e_2$
$P_2^{(t,2)} : e_4 = \boldsymbol{eventually}\ e_3$

Temporal Reasoning — Temporal Quantification



Temporal Reasoning — Temporal Quantification



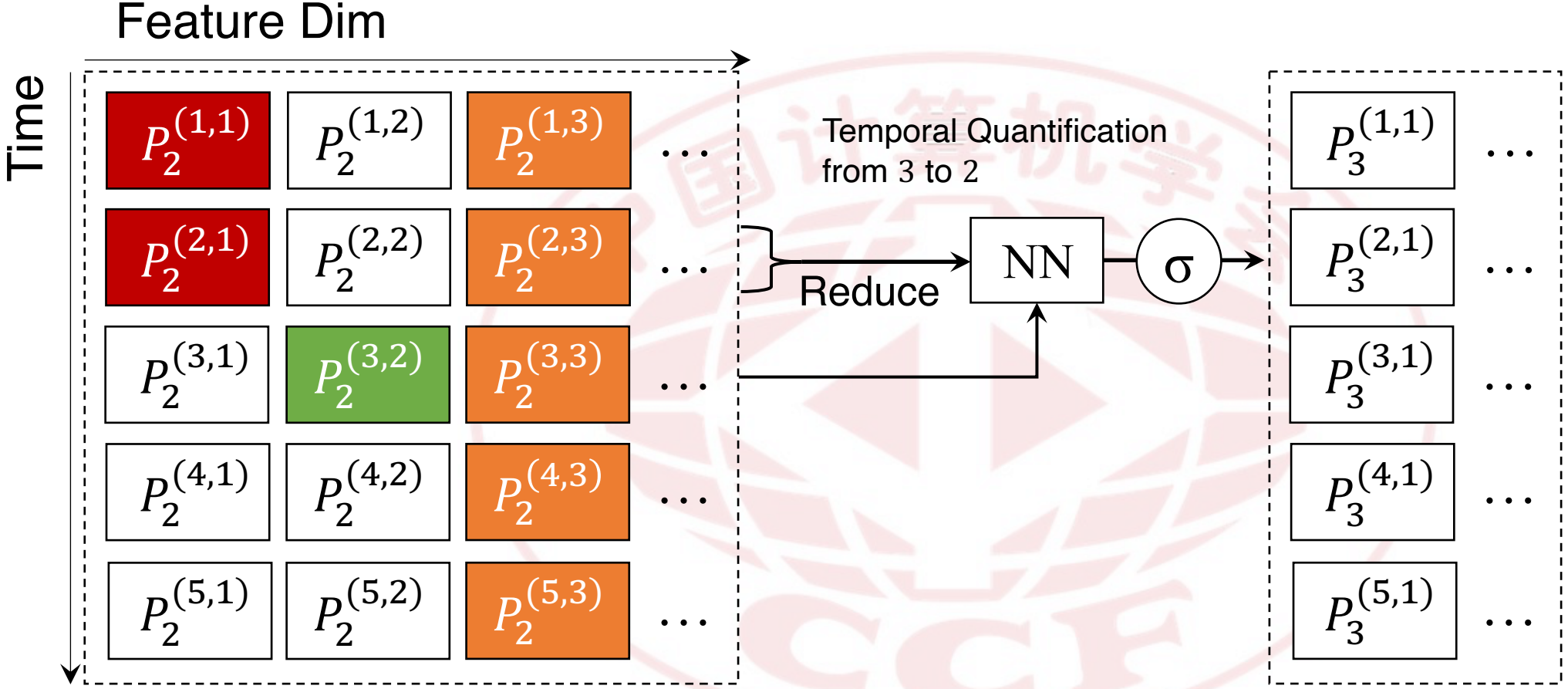
$P_2^{(t,1)} : e_1 \qquad P_2^{(t,2)} : e_2$

$P_2^{(t,2)} : e_4 = \boldsymbol{eventually}\ e_3$

$P_3^{(t,1)} : e_5 = e_1\ \boldsymbol{until}\ (e_2 \wedge e_4)$

$= e_1\ \boldsymbol{until}\ \big(e_2 \wedge (\boldsymbol{eventually}\ e_3)\big)$

Temporal Reasoning — Temporal Quantification



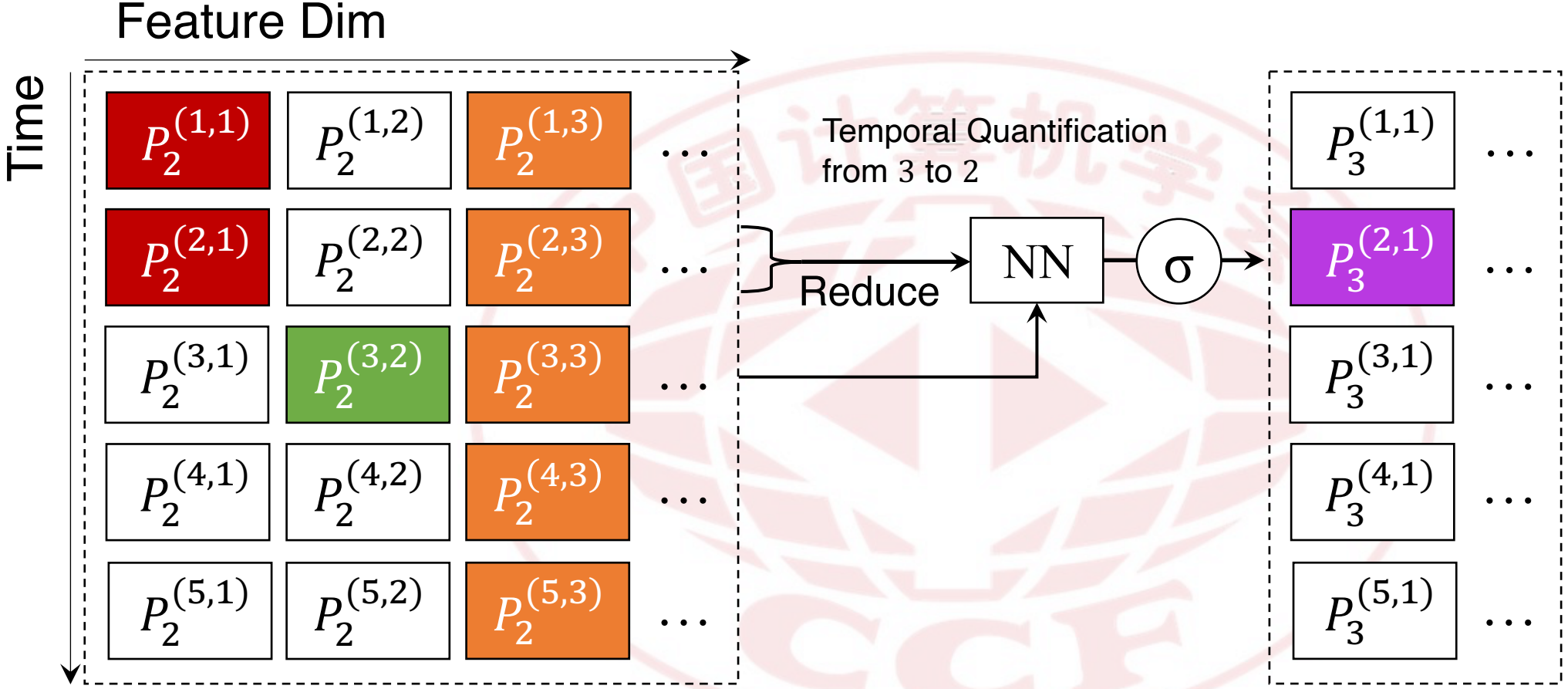
$P_2^{(t,1)} : e_1 \quad P_2^{(t,2)} : e_2$

$P_2^{(t,2)} : e_4 = \boldsymbol{eventually}\ e_3$

$P_3^{(t,1)} : e_5 = e_1\ \boldsymbol{until}\ (e_2 \wedge e_4)$

$= e_1\ \boldsymbol{until}\ \left(e_2 \wedge (\boldsymbol{eventually}\ e_3)\right)$

Temporal Reasoning — Temporal Quantification



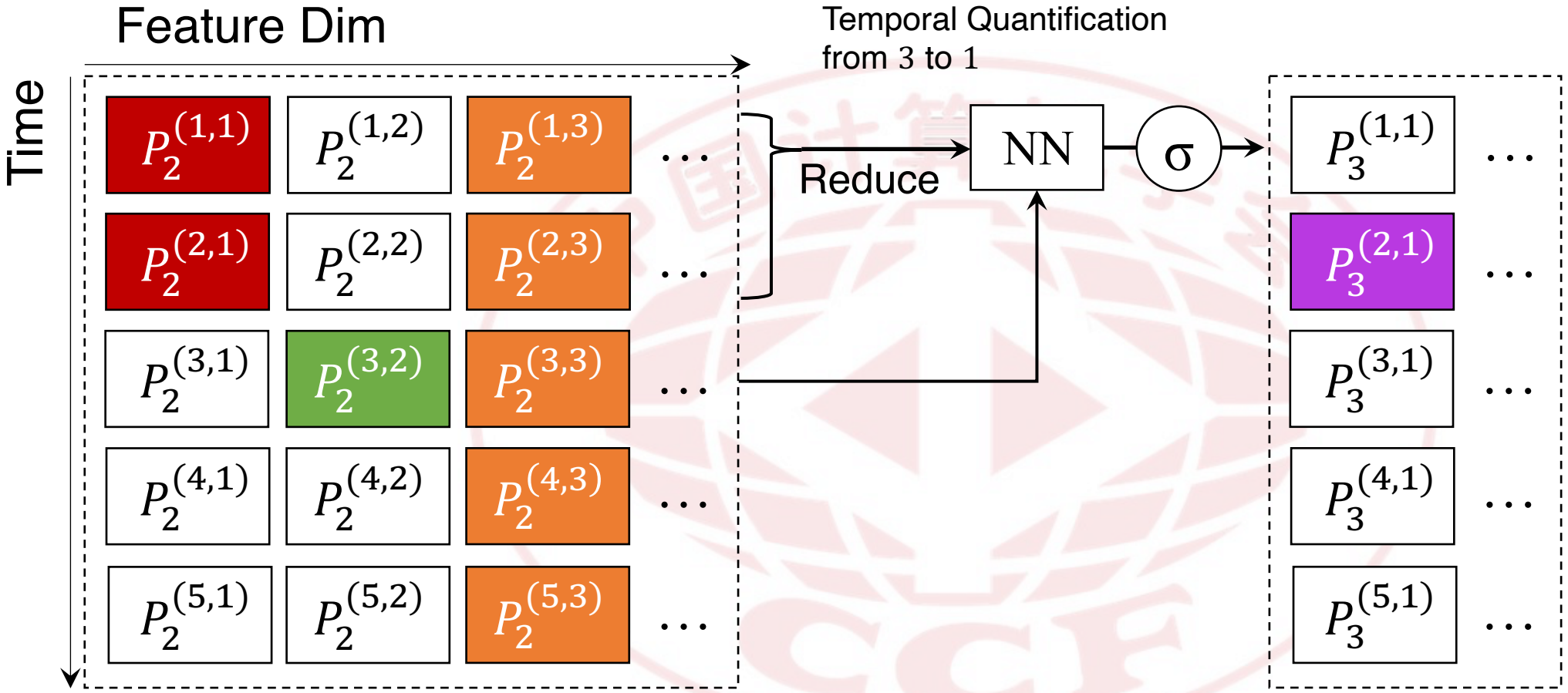
$P_2^{(t,1)} : e_1 \quad P_2^{(t,2)} : e_2$

$P_2^{(t,2)} : e_4 = \boldsymbol{eventually}\ e_3$

$P_3^{(t,1)} : e_5 = e_1\ \boldsymbol{until}\ (e_2 \wedge e_4)$

$= e_1\ \boldsymbol{until}\ \big(e_2 \wedge (\boldsymbol{eventually}\ e_3)\big)$

Temporal Reasoning — Temporal Quantification



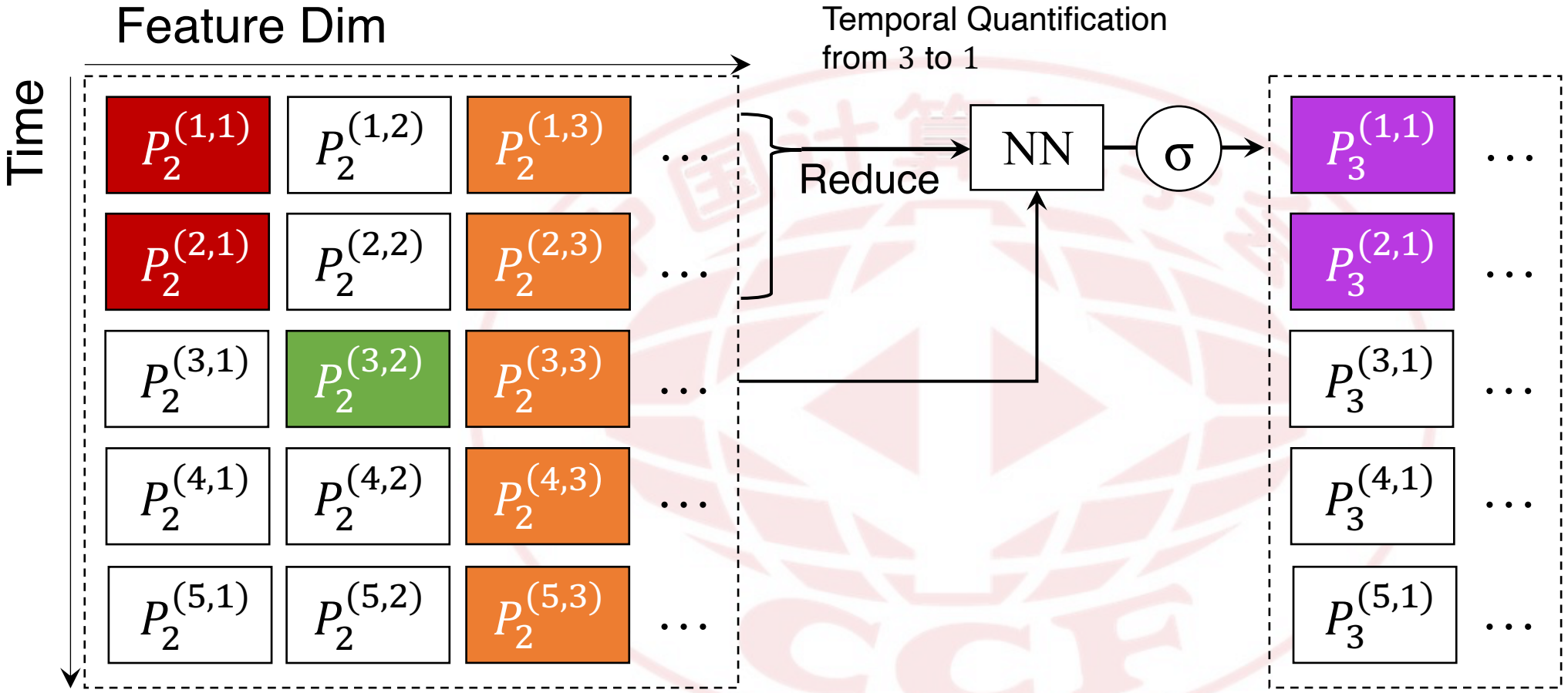
$P_2^{(t,1)} : e_1 \quad P_2^{(t,2)} : e_2$

$P_2^{(t,2)} : e_4 = \boldsymbol{eventually}\ e_3$

$P_3^{(t,1)} : e_5 = e_1\ \boldsymbol{until}\ (e_2 \wedge e_4)$

$= e_1\ \boldsymbol{until}\ (e_2 \wedge (\boldsymbol{eventually}\ e_3))$

Temporal Reasoning — Temporal Quantification



$P_2^{(t,1)} : e_1 \quad P_2^{(t,2)} : e_2$

$P_2^{(t,2)} : e_4 = \boldsymbol{eventually}\ e_3$

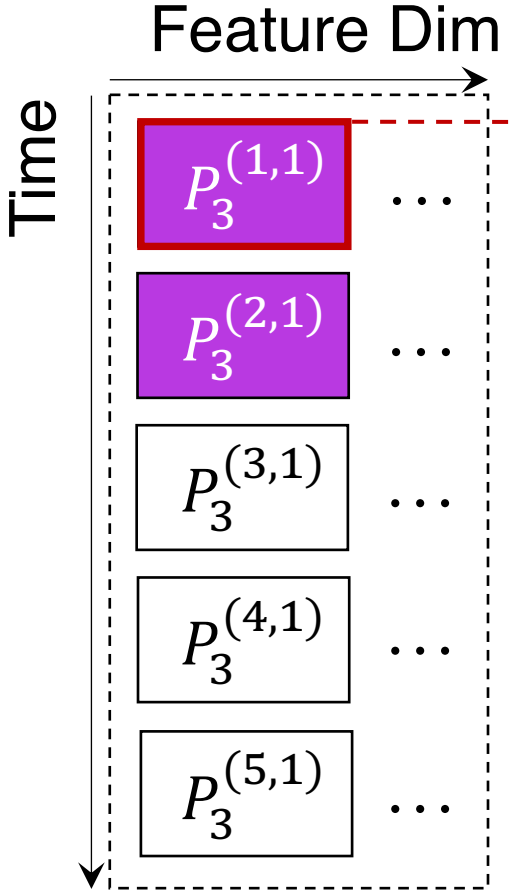
$P_3^{(t,1)} : e_5 = e_1\ \boldsymbol{until}\ (e_2 \wedge e_4)$

$= e_1\ \boldsymbol{until}\ (e_2 \wedge (\boldsymbol{eventually}\ e_3))$

Temporal Reasoning — Temporal Quantification

$$P_3^{(t,1)}: e_5 = e_1 \ \textbf{until}\ (e_2 \wedge e_4)$$
$$= e_1 \ \textbf{until}\ \big(e_2 \wedge (\textbf{eventually}\ e_3)\big)$$

Temporal Reasoning — Temporal Quantification



e_1: Red team is possessing the ball

- $\exists x.\left(\text{red}(x) \wedge \forall y.\left(\text{close}(x,y) \vee \neg\text{ball}(y)\right)\right)$

e_2: Ball is moving fast while is near the blue teams gate

- $\exists x.\left(\text{ball}(x) \wedge \text{fast}(x) \wedge \text{near\_blue\_gate}(x)\right)$

e_3: Blue team goal keeper is controlling the ball

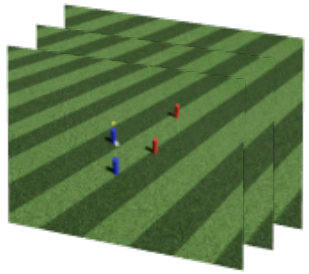
- $\exists x.\left(\text{blue}(x) \wedge \text{is\_goalkeeper}(x) \wedge \forall y.\left(\text{close}(x,y) \vee \neg\text{ball}(y)\right)\right)$

e_5: Blue team deflected: Red team was possessing the ball, until the ball was moving fast near blue teams gate, and then the goal keeper of the blue team got the ball

- $\exists x.\left(\text{red}(x) \wedge \forall y.\left(\text{close}(x,y) \vee \neg\text{ball}(y)\right)\right)$
  $\boldsymbol{U}\ \exists x.\left(\text{ball}(x) \wedge \text{fast}(x) \wedge \text{near\_blue\_gate}(x)\right) \wedge$
  $F\ \exists x.\left(\text{blue}(x) \wedge \text{is\_goalkeeper}(x) \wedge \forall y.\left(\text{close}(x,y) \vee \neg\text{ball}(y)\right)\right)$

$$P_3^{(t,1)}: e_5 = e_1\ \boldsymbol{until}\ (e_2 \wedge e_4)$$
$$= e_1\ \boldsymbol{until}\ \left(e_2 \wedge (\boldsymbol{eventually}\ e_3)\right)$$

TOQ–Nets — Overview



Input Trajectories

$\text{xpos}(x,t)$ $\text{tplayer}(x,t)$
$\text{ypos}(x,t)$ $\text{ball}(x,t)$
$\text{zpos}(x,t)$ .....

Action Label

Short Pass

Supervision

Predicted Action

Back-Propagation

Input Feature Extractor

Nullary P_0

Unary: Q_0

Binary: R_0

... ...

K **Relational Reasoning Layers**

Nullary P_K

Unary: Q_K

Binary: R_K

$P_K^{(1)}$ $P_K^{(2)}$ $P_K^{(3)}$ ⋮ $P_K^{(T)}$

L **Temporal Reasoning Layers**

$P_{K+L}^{(1)}$ $P_{K+L}^{(2)}$ $P_{K+L}^{(3)}$ ⋮ $P_{K+L}^{(T)}$

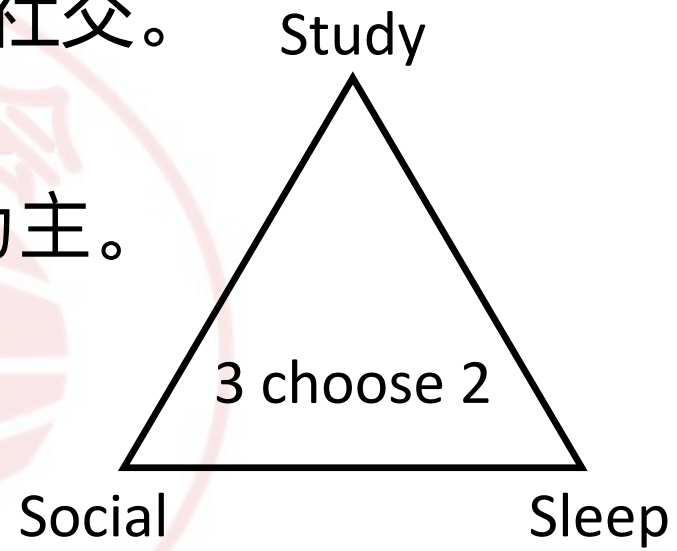
Time

如何避免 OI 模式的“副作用”

- 要学会“拉长战线”，OI 题都可以在几个小时内解决，而科研和工程上的问题并非如此。要学会做没有短期回报的事情。
- 要学会听课，读材料，而不是仅仅自己做题。避免纯粹的“面向作业学习”。
- 学会提问，学会合作，学会从别人的视角理解问题。
- 要跨越“AC思维”，不要觉得只要做对了题，懂不懂并不重要，要从定义开始理解。透彻的理解对课程，考试，科研都很重要。
- 并不是每个问题都有简洁优美高效的解法，不要看不起暴力。

高中 OI 选手的生活与社交

- 事实：很多集训队员的高中生活中几乎没有社交。
- 多数集训队员的社交对象以机房同学/网友为主。
- 部分 OI 选手生活不规律，经常熬夜



对 OI 选手大学生活的建议

- 重视一起在 OI 中成长的同学，你们会有很多共同语言。
- 不要只和 OI 选手打交道，走出舒适圈，多样化自己的社交圈。与更多的人交流会帮助你认识到自己社交和心智上的局限，并扩展自己的社交能力。
- 有能力可以主动参与学校活动，培养其他的爱好并结识同好。

杂谈：一些宽泛的建议

- 享受生活，这可能是前后三年内唯一有空玩一玩的时间。保持纯粹，到了大学之后人会变得复杂。珍惜现在只想快乐做点题目的时光。
- 早点思考自己要做什么，大一大二就可以去了解一下，可以试错，多获取一些信息，多和前辈聊天。
- 尽可能想清楚自己要做什么。如果要做科研的话，不要只做TCS（尤其是组合算法），多看别的领域。
- 放平心态，享受过程，关注自己和周边人的心理健康，有困难及时求助。相信自己，大胆尝试。

致谢

- 感谢 CCF 提供交流与分享的平台。
- 感谢陈立杰，罗浩宽，王蕴韵，吕凯风，高继扬和杨家齐同学接受采访。
- 感谢在座各位的聆听。

谢谢大家!

2022 年 1 月 24 日

有争议的例子

- [WC2015] 混淆与破解
- [WC2015] 未来程序
- [CTSC2015] CTEX
- [清华集训2015] 恐怖的奴隶主
- [清华集训2015] 多边形下海
- [清华集训2015] 斗地主
- [WC2017] 挑战

“偏题”应该怎么出

- 数学题应该怎么出？
  - 例子：[浙江省选2015]地震后的幻想乡, Endless Spin
- 物理题应该怎么出？
  - 例子：[清华集训2015] V
- 游戏题应该怎么出？
  - 例子：[JLOI2014] 2048, [UR#9] 包子晚宴
- 提交答案题应该怎么出？
  - 例子：[APIO2013] TASKSAUTHOR, [IOI2017] Nowruz

什么样的题叫好题?

[NOI2013] 向量内积

- 给n个d维向量，问是否存在两个向量的内积为k的倍数。
- $n \leq 100000; d \leq 30; k \in \{2,3\}$.
- $O(n^2 d)$做法有60，使用bitset做$k = 2$的有80。

[NOI2015] 寿司晚宴

- 问有多少种办法把$2,3,\cdots,n$分成三个集合A,B,C，使得A集合中任何一个数都和B中任何一个数都互质。

- $n \leq 500.$

-

30pts	$n \leq 30$
20pts	$n \leq 100$
20pts	$n \leq 200$
30pts	$n \leq 500$

[NOI2017] 整数

- 一个整数x刚开始为0，有n个操作，操作分为两种
- $1\ a\ b$：把x加上$a \cdot 2^b$
- $2\ k$：询问x二进制从末尾数第k位的值，保证$x \geq 0$

- $n \leq 1000000; |a| \leq 10^9; b, k \leq 30n.$

- 数据中，n从10到1000000按对数大致均匀分布，存在一些$a = 1; |a| = 1; b, k \leq 30; b, k \leq 100; b, k \leq n$的部分分。

[IOI2016] Railroad

- 有n个过山车轨道，每个轨道有两个参数s_i, t_i，进入速度不能高于s_i且离开速度一定为t_i，在轨道之间减速需要花费减速量的代价。火车初速度为1。
- 现在你需要把轨道排成一排，最小化代价。

- $n \leq 200000; 1 \leq s_i, t_i \leq 10^9$.
-

Subtask 1 [11pts]	$n \leq 8$
Subtask 2 [23pts]	$n \leq 16$
Subtask 3 [30pts]	$n \leq 200000$
Subtask 4 [36pts]	$n \leq 200000$

[IOI2017] Ancient Books

- 有一列桌子$0 \sim n-1$，每个桌子上摆了一本古书，古书编号也为$1 \sim n$，你需要整理古书使得古书和所在桌子的编号对应。你刚开始在s，结束时也需要回到s。你有如下操作，求最小时间。
  - 将手上的内容和桌子上的内容交换（即拿起/放下或交换）
  - 走到相邻的桌子
- 前一个操作为瞬时操作，最后一个操作需要花费1的时间。
- $1 \leq n \leq 1000000; 0 \leq s \leq n-1.$
-

Subtask 1 [12pts]	$n \leq 4; s = 0$	Subtask 4 [20pts]	$n \leq 1000$
Subtask 2 [10pts]	$n \leq 1000; s = 0$	Subtask 5 [30pts]	$n \leq 10^6$
Subtask 3 [28pts]	$n \leq 10^6; s = 0$		

[IOI2018] 排座位

- 一个大厅有$H \times W$个座位，行编号为$0 \sim H-1$，列编号为$0 \sim W-1$。一共HW位参赛者，编号为$0 \sim HW-1$，第i个参赛者座位为(R_i, C_i)。对于任意k，若编号为$0 \sim k-1$的参赛者的座位恰好组成了一个实心矩形，那么称为美妙的。一个座位表的美妙度为美妙的k的个数。
- 给定座位表，每次交换两个人的座位，维护美妙度。
- $HW \leq 10^6; Q \leq 50000.$
-

Subtask 1 [5pts]	$HW \leq 100; Q \leq 5000.$	Subtask 4 [6pts]	$Q \leq 5000, \lvert a-b \rvert \leq 10^4.$
Subtask 2 [6pts]	$HW \leq 10^4; Q \leq 5000.$	Subtask 5 [33pts]	$H = 1$
Subtask 3 [20pts]	$H, W \leq 10^3; Q \leq 5000.$	Subtask 6 [30pts]	No Special Constraints

如何组成一场比赛?

- 难度分布合理
  - 不合理的例子：CTSC
- 代码量合理
  - 不合理的例子：AHOI 2016
- 部分分合理
  - 部分分很大程度数影响了题目的实际权重
  - 失衡的例子：NOI 2013 Day 1
  - 好的例子：NOI 2014
- 新颖的思路？

[US Open 2017] Modern Art

- 有一个$N \times N$的格子棋盘，刚开始颜色都是0，颜色$1 \sim N^2$以某种顺序被涂到棋盘上，每次涂色恰好是一个子矩形，并会覆盖之前的颜色。现在给定棋盘的最终状态，问哪些颜色可能是第一个被涂到棋盘上的。

- $N \leq 1000$.

[US Open 2017] Switch Grass

- 给一张N个点M条边的带权无向图。每个点都有一个颜色，现在有Q个操作，每个操作修改一个点的颜色，并询问两个顶点不同色的边的最大权值。

- $N \leq 200000; M \leq 200000; Q \leq 200000.$

[US Open 2017] Cowbasic

- 有一门编程语言，有以下语法：

```
<var>=<exp>
<num> MOO {
    <list of statements>
}
RETURN <var>
```

- 其中：<exp> = <num> | <var> | (<exp> + <exp>)，<num>都是正整数。
- 给一段程序，求答案对10^9+7取模的结果。
- 程序最多100行。

WQS–CLJ 二分

- [国家集训队2012] tree（by 陈立杰）
  - 给一个n个点m条边的无向带权连通图，每条边是黑色或白色，求恰好有c条边的最小权生成树。
  - $n \leq 50000, m \leq 100000, 1 \leq w_i \leq 100.$
- [IOI2016] 外星人