Recursive Reward Aggregation

Yuting Tang Yivan Zhang Johannes Ackermann
Yu-Jie Zhang Soichiro Nishimori Masashi Sugiyama

Reinforcement Learning Conference 2025

In reinforcement learning, an agent obtains a *sequence* of rewards as it takes actions in a dynamic environment.

$$r_1 \quad r_2 \quad r_3 \quad \dots$$

We use the **discounted sum** to evaluate its performance.

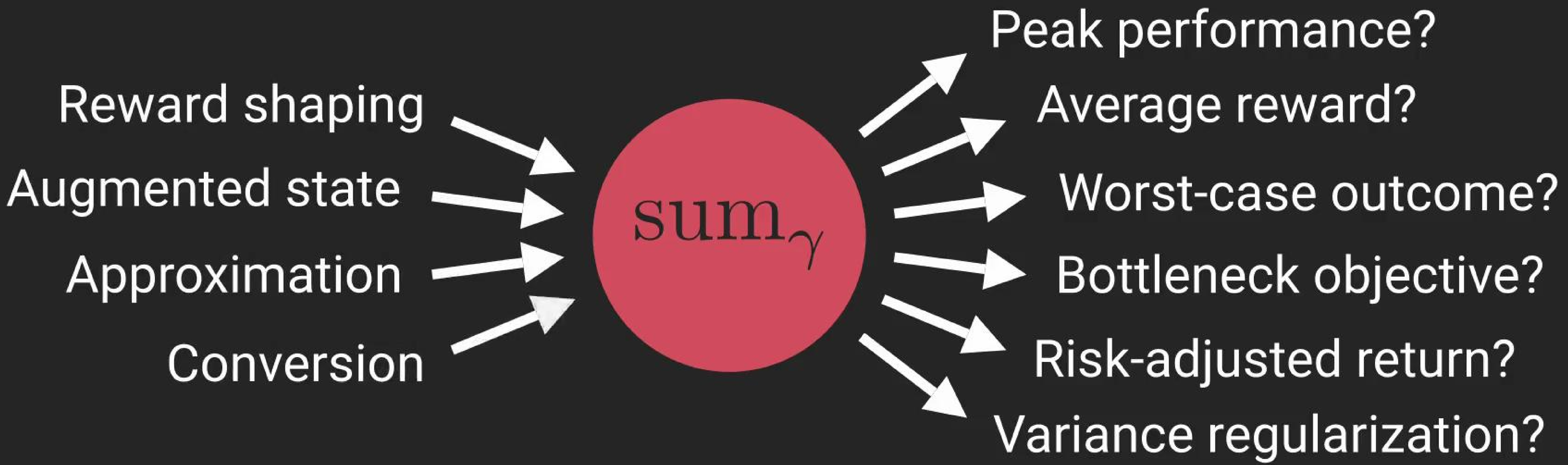
$$\text{sum}_\gamma \left[r_1, r_2, r_3, \dots\right] := r_1 + \gamma r_2 + \gamma^2 r_3 + \dots \quad \gamma \in [0, 1]$$

Why do we optimize the *discounted sum* of rewards?

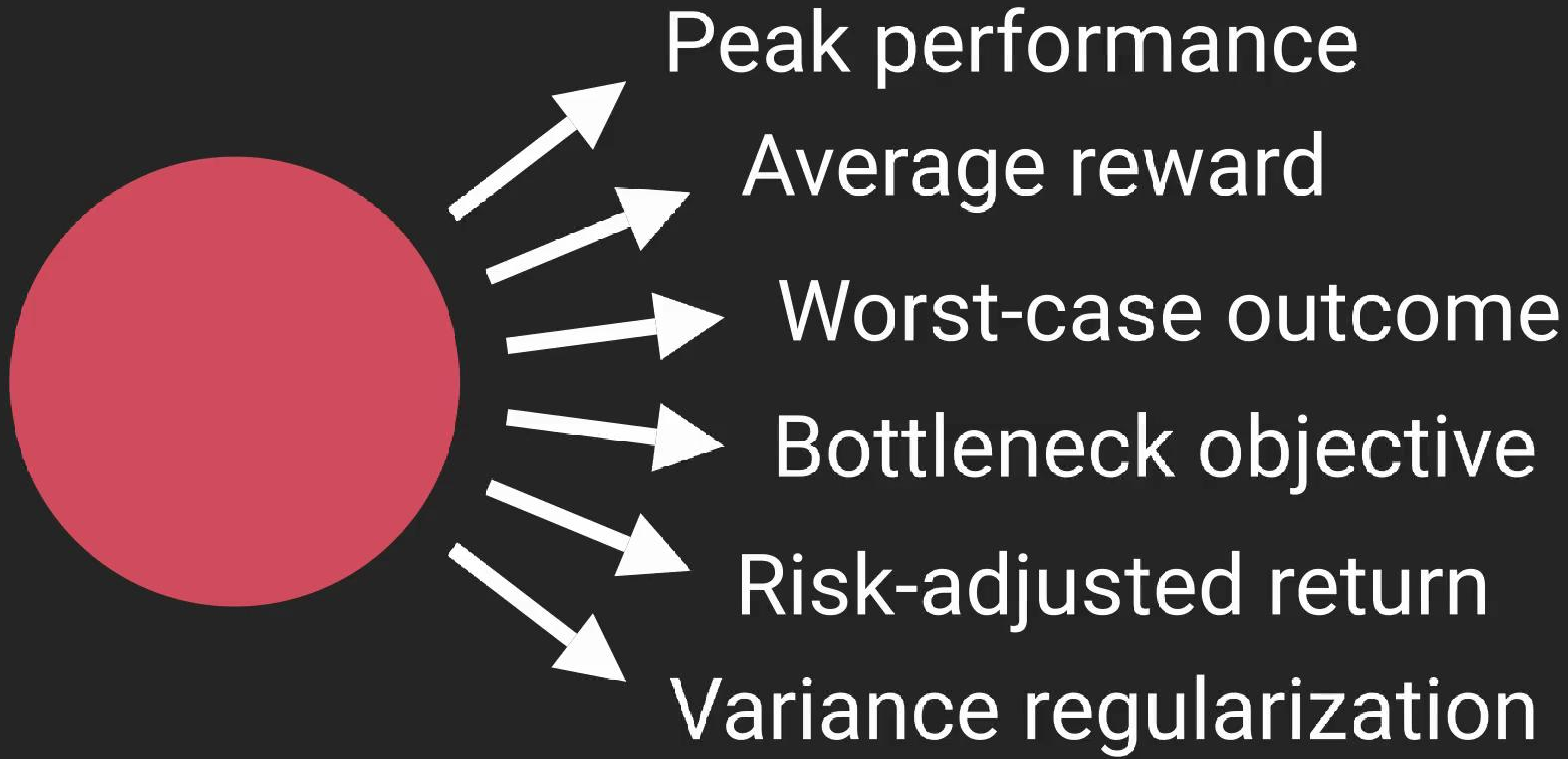
Why do we optimize the *discounted sum* of rewards?

- Standard? Convenient? Mathematically elegant?
- Practical benefits? Higher, sooner rewards preferred
- Theoretical guarantees? Contraction $\rightarrow$ unique fixed point
- Reward hypothesis? If true, all preferences can be represented

But... does it always align with our requirements?



Sobel (1982), Quah & Quek (2006), Wang et al. (2020), Cui & Yu (2023)



Can we optimize other reward aggregations, *directly*, *efficiently*, and *effectively*?

Discounted sum is a *recursive function*

$$\begin{aligned}\text{sum}_\gamma\,[r_1, r_2, r_3, \ldots] &= r_1 + \gamma r_2 + \gamma^2 r_3 + \ldots \\ &= r_1 + \gamma(r_2 + \gamma r_3 + \ldots) \\ &= r_1 + \gamma\,\text{sum}_\gamma\,[r_2, r_3, \ldots]\end{aligned}$$

$$\text{sum}_\gamma\,[\,] = 0$$

Discounted sum is a *recursive function*

$$\text{sum}_\gamma\,[r_1,\, r_2,\, r_3,\, \ldots] := r_1 + \gamma\,\text{sum}_\gamma\,[r_2,\, r_3,\, \ldots]$$

$$\text{sum}_\gamma\,[\,] := 0$$

diagrammatic representation

$$[\mathbb{R}] \xrightarrow{\quad \text{sum}_\gamma \quad} \mathbb{R}$$

the set of

lists of rewards

Discounted sum is a *recursive function*

$$\text{sum}_\gamma\,[r_1, r_2, r_3, \ldots] := r_1 + \gamma\,\text{sum}_\gamma\,[r_2, r_3, \ldots]$$

$$\text{sum}_\gamma\,[\,] := 0$$

diagrammatic representation

$$[\mathbb{R}] \xrightarrow{\text{sum}_\gamma} \mathbb{R}$$

Discounted sum is a *recursive function*

$$\text{sum}_\gamma\,[r_1,\, r_2,\, r_3,\, \ldots] := r_1 + \gamma\,\text{sum}_\gamma\,[r_2,\, r_3,\, \ldots]$$

$$\text{sum}_\gamma\,[\,] := 0$$

diagrammatic representation

Discounted sum is a *recursive function*

$$\operatorname{sum}_\gamma\,[r_1, r_2, r_3, \ldots] := r_1 + \gamma \operatorname{sum}_\gamma\,[r_2, r_3, \ldots]$$

$$\operatorname{sum}_\gamma\,[\,] := 0$$

diagrammatic representation

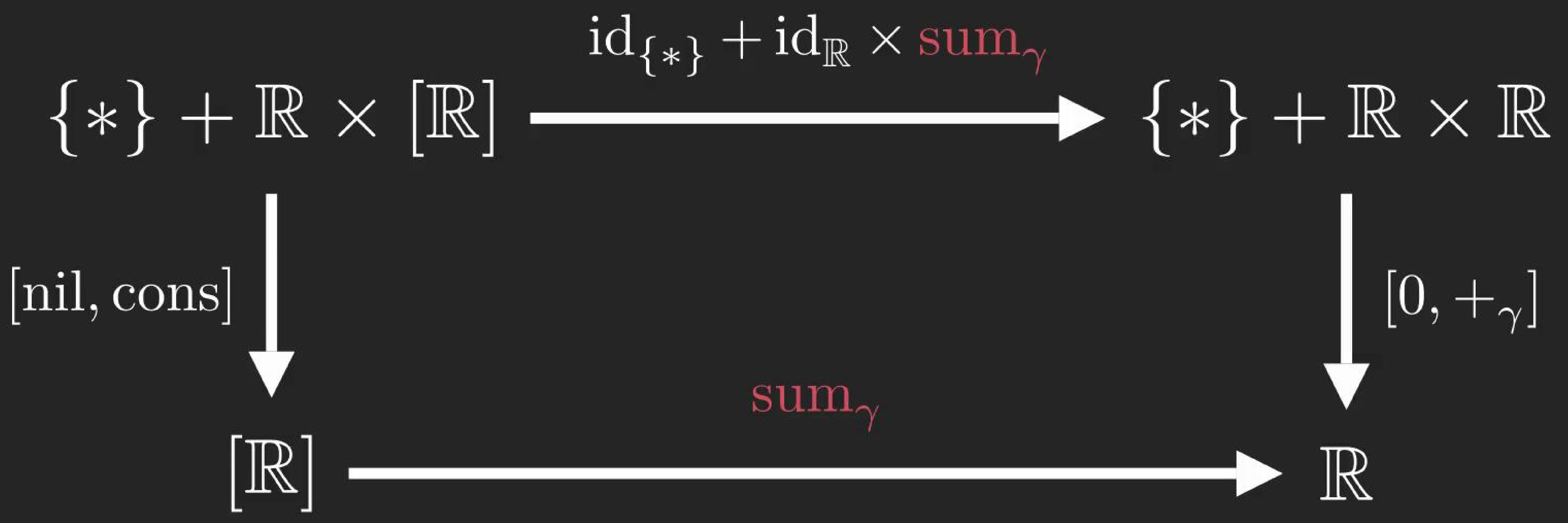
$$[\mathbb{R}] \xrightarrow{\operatorname{sum}_\gamma} \mathbb{R}$$

reward aggregation function

Discounted sum is a *recursive function*

$$\mathrm{sum}_\gamma\ [r_1, r_2, r_3, \ldots] := r_1 + \gamma\, \mathrm{sum}_\gamma\ [r_2, r_3, \ldots]$$

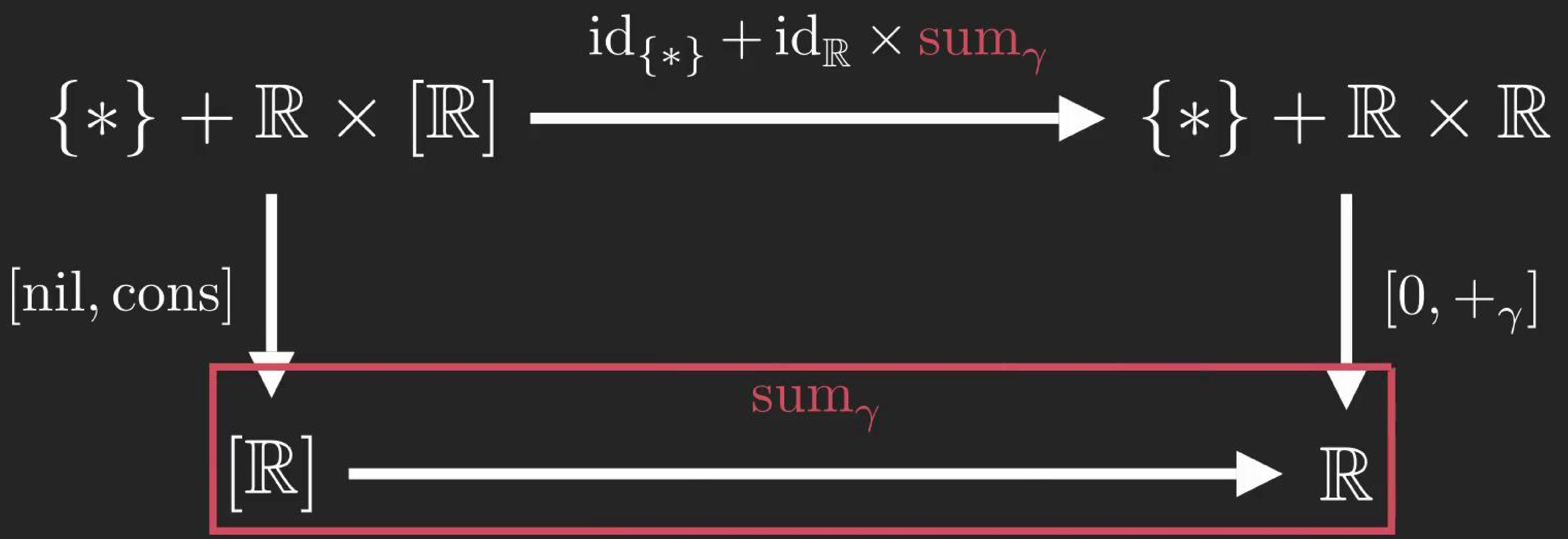
$$\mathrm{sum}_\gamma\ [\,] := 0$$



$\{*\}$: singleton set $+$: disjoint union $\times$: Cartesian product
id: identity function nil: empty list $[\,]$ cons: list constructor

Discounted sum is a *recursive function*

$$\begin{aligned} \mathrm{sum}_\gamma\,[r_1, r_2, r_3, \ldots] &:= r_1 + \gamma\,\mathrm{sum}_\gamma\,[r_2, r_3, \ldots] \\ \mathrm{sum}_\gamma\,[\,] &:= 0 \end{aligned}$$



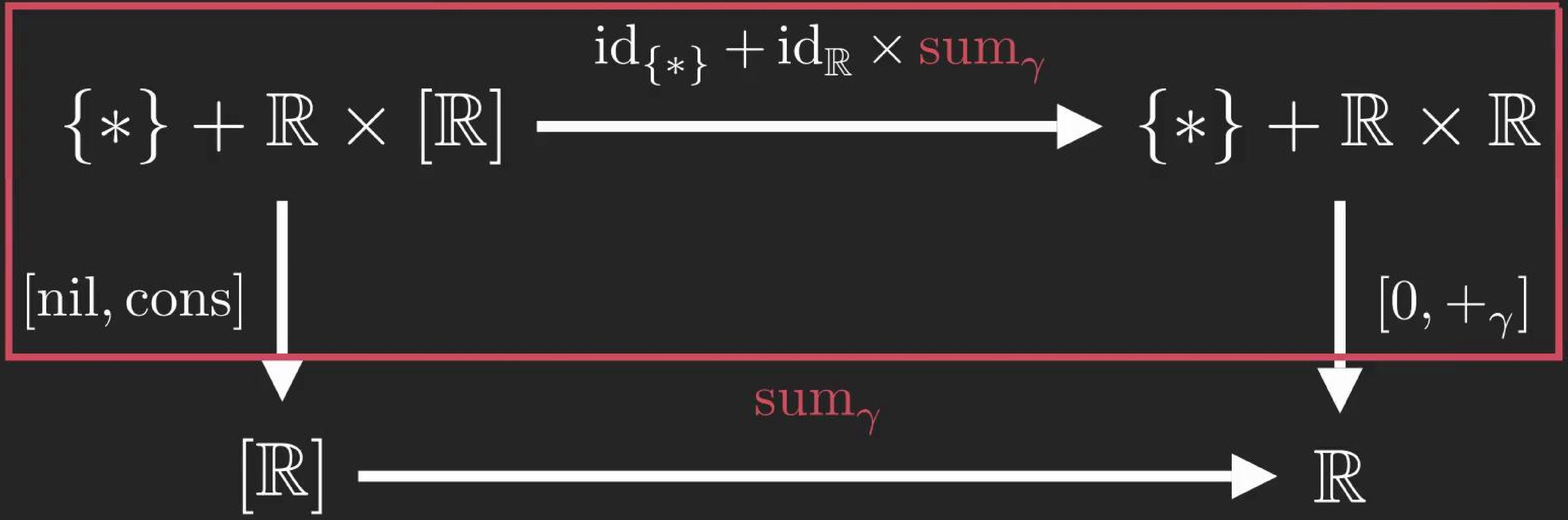
function declaration

$\{*\}$: singleton set $+$: disjoint union $\times$: Cartesian product

id: identity function nil: empty list $[\,]$ cons: list constructor

Discounted sum is a *recursive function*

$$\mathrm{sum}_\gamma\ [r_1, r_2, r_3, \ldots] := r_1 + \gamma\, \mathrm{sum}_\gamma\ [r_2, r_3, \ldots]$$
$$\mathrm{sum}_\gamma\ [\,] := 0$$



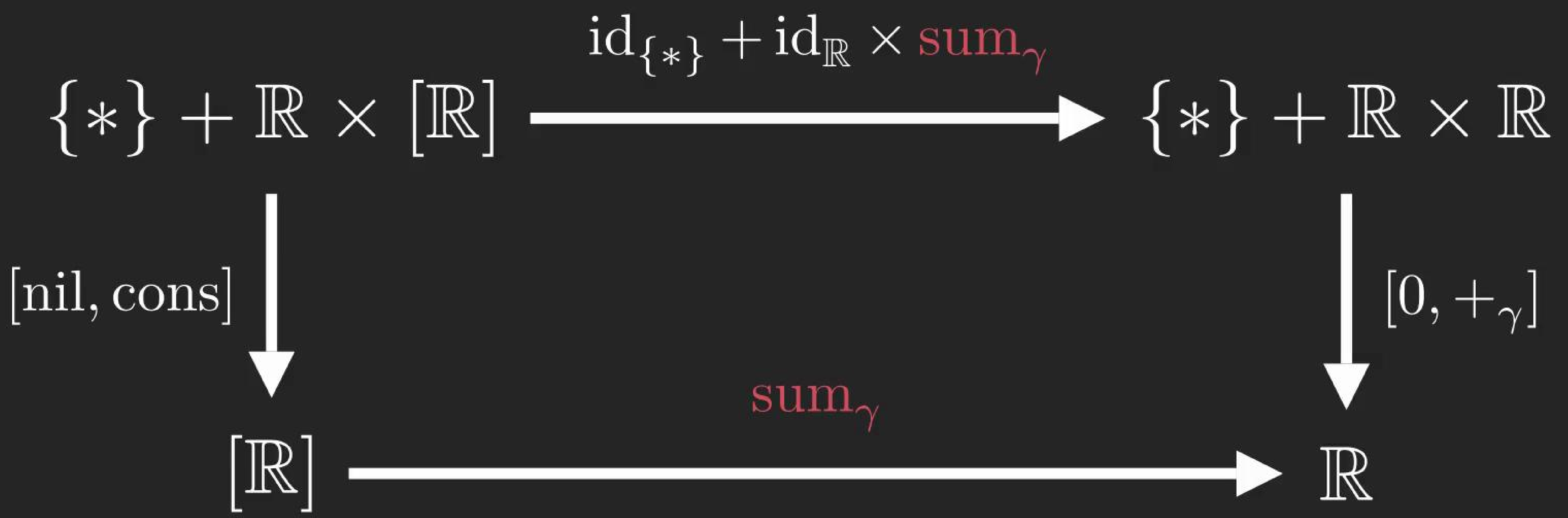
recursive definition

$\{*\}$: singleton set $+$: disjoint union $\times$: Cartesian product
id: identity function nil: empty list $[\,]$ cons: list constructor

Discounted sum is a *recursive function*

$$\mathrm{sum}_\gamma\,[r_1, r_2, r_3, \ldots] := r_1 + \gamma\,\mathrm{sum}_\gamma\,[r_2, r_3, \ldots]$$

$$\mathrm{sum}_\gamma\,[\,] := 0$$

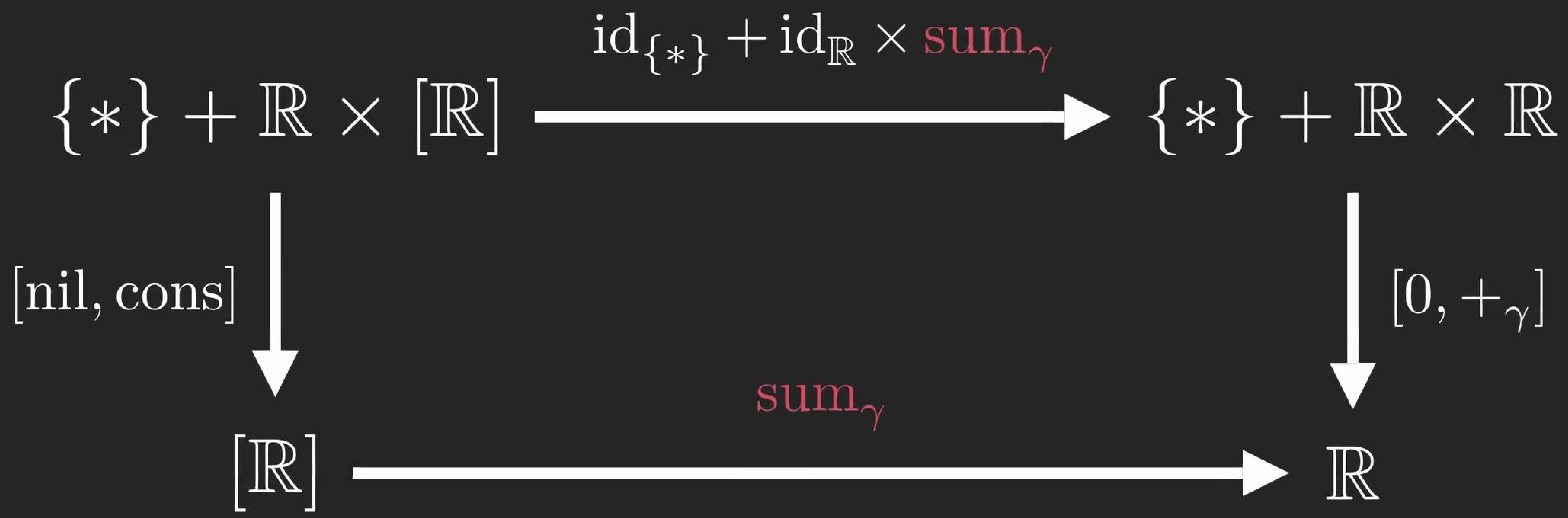


same endpoints $\rightarrow$ same composite

$\{*\}$: singleton set $+$: disjoint union $\times$: Cartesian product
id: identity function nil: empty list $[\,]$ cons: list constructor

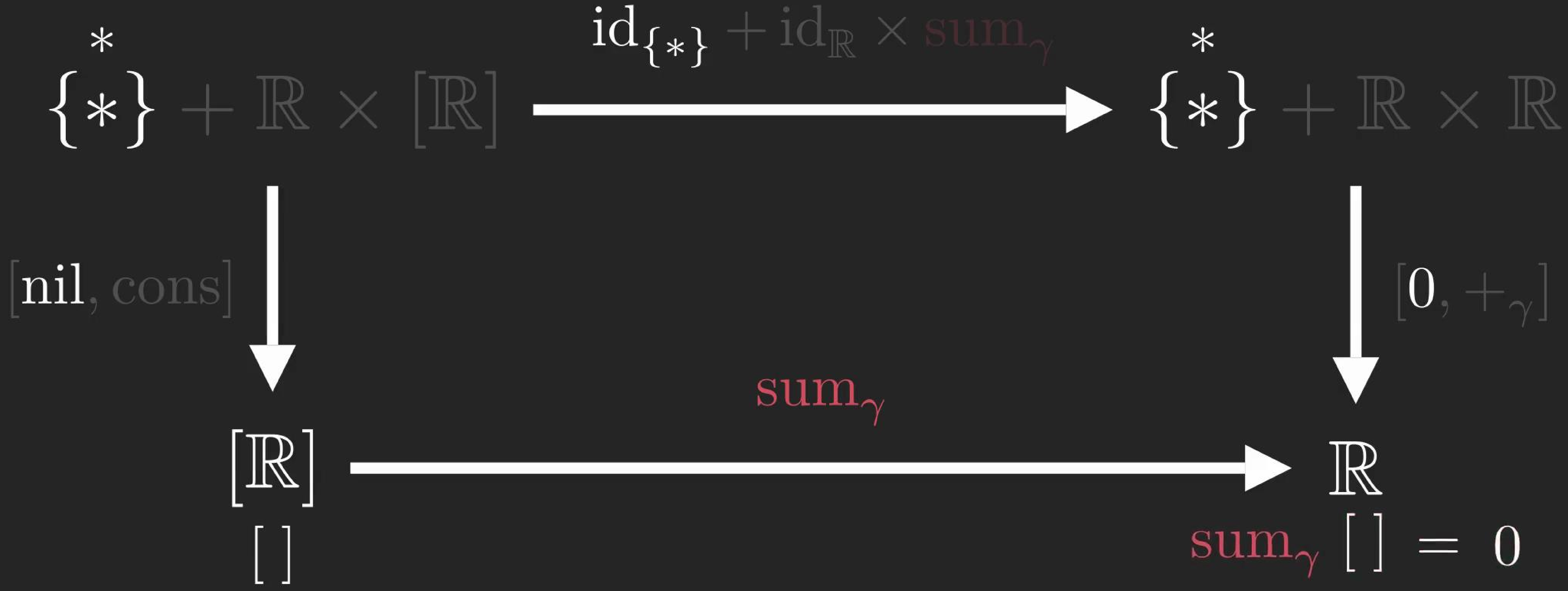
$$\mathrm{sum}_\gamma\,[r_1, r_2, r_3, \ldots] := r_1 + \gamma\,\mathrm{sum}_\gamma\,[r_2, r_3, \ldots]$$

$$\mathrm{sum}_\gamma\,[\,] := 0$$



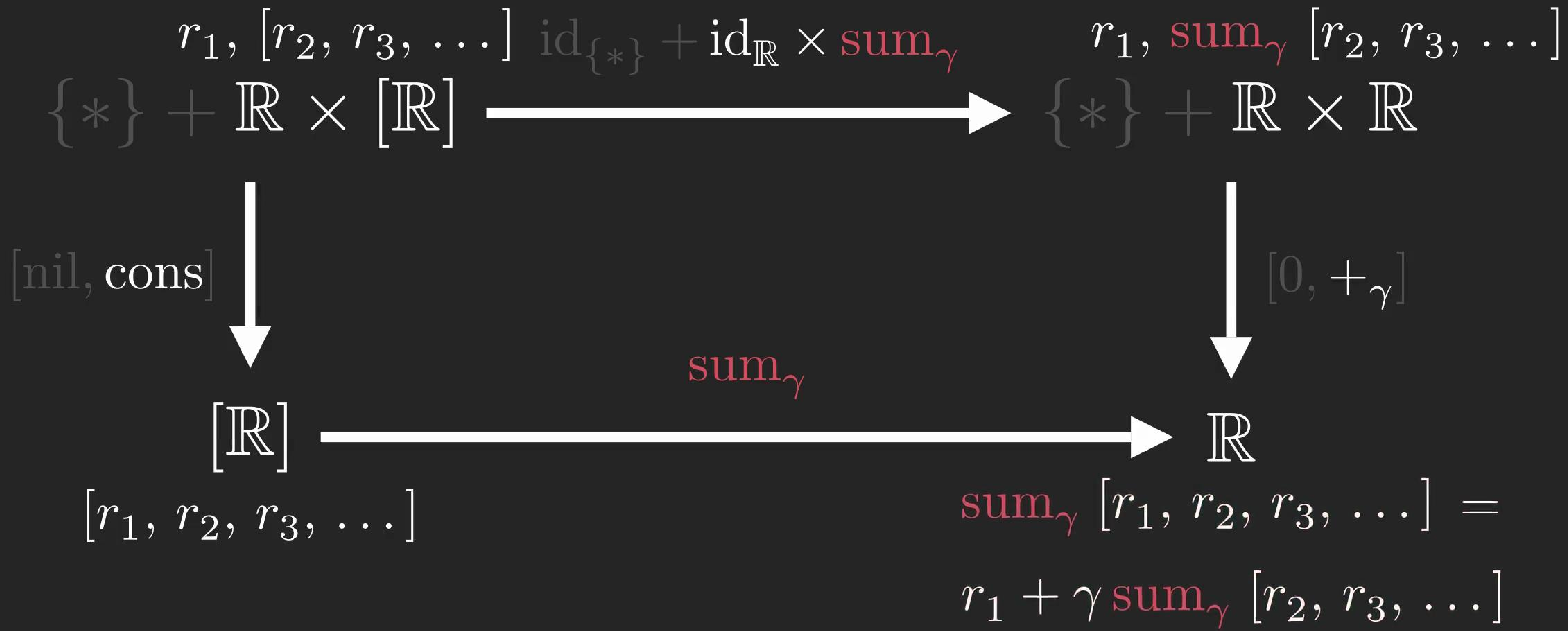
$$\text{sum}_\gamma\ [r_1, r_2, r_3, \ldots] := r_1 + \gamma\, \text{sum}_\gamma\ [r_2, r_3, \ldots]$$

$$\text{sum}_\gamma\ [\,] := 0$$



$$\mathrm{sum}_\gamma\,[r_1, r_2, r_3, \ldots] := r_1 + \gamma\,\mathrm{sum}_\gamma\,[r_2, r_3, \ldots]$$

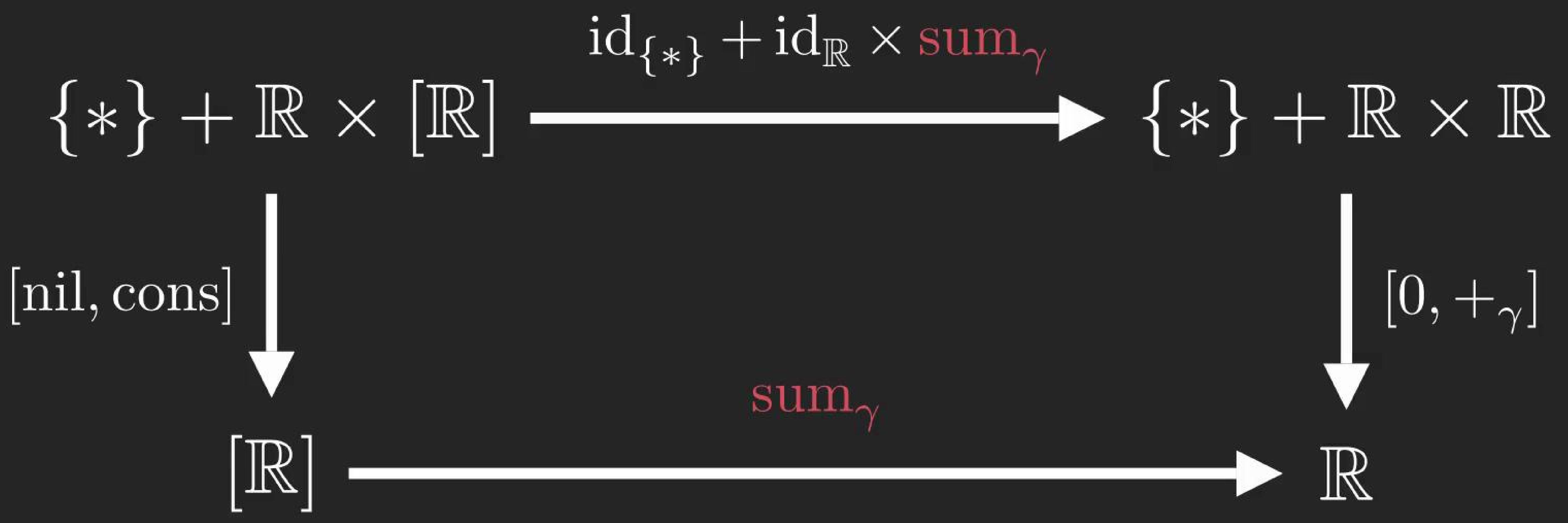
$$\mathrm{sum}_\gamma\,[\,] := 0$$



Discounted sum is a *recursive function*

$$\mathrm{sum}_\gamma\,[r_1, r_2, r_3, \ldots] := r_1 + \gamma\,\mathrm{sum}_\gamma\,[r_2, r_3, \ldots]$$

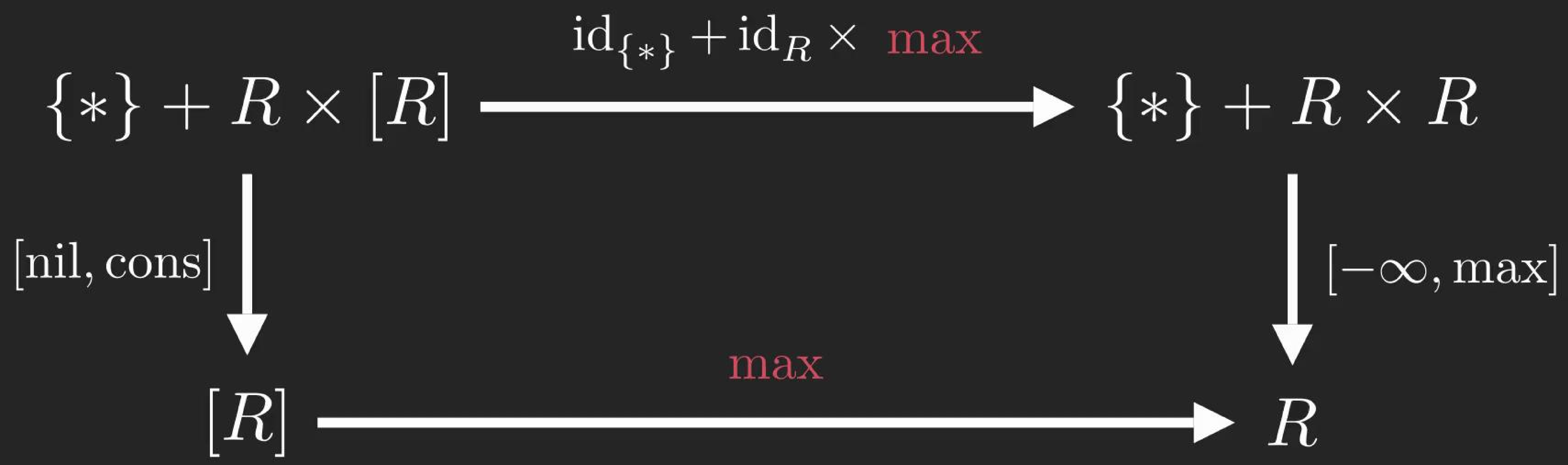
$$\mathrm{sum}_\gamma\,[\,] := 0$$



sum_γ is *uniquely defined* by $[0, +_\gamma]$ via recursion.

... so are many other aggregation functions.

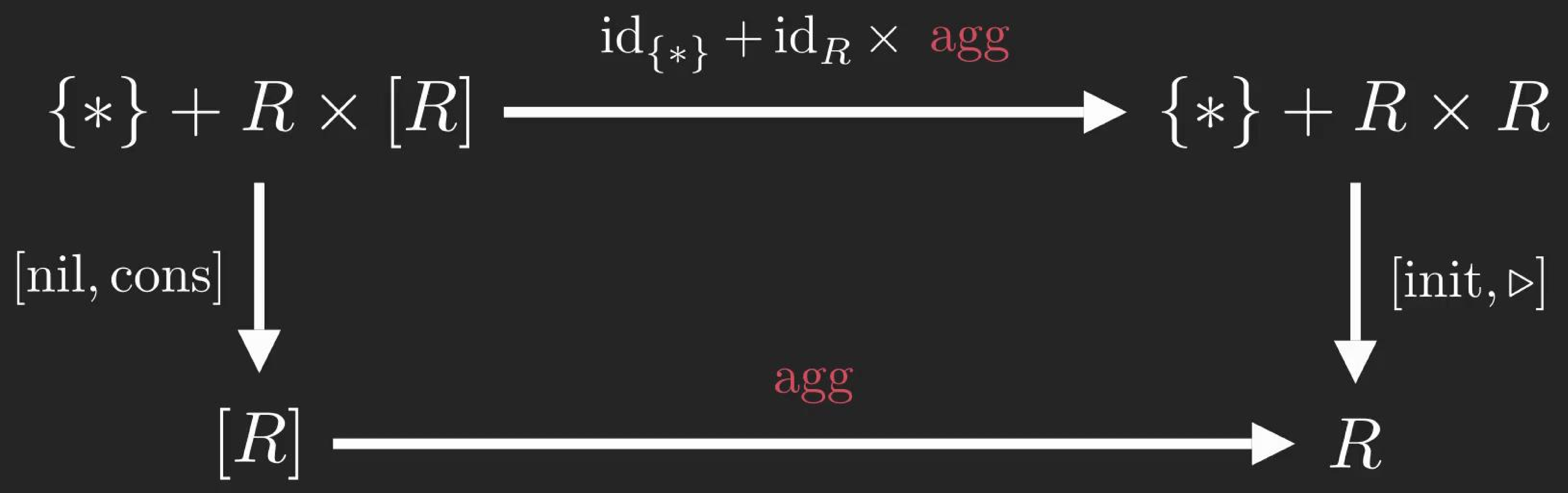
$$\max\,[r_1, r_2, r_3, \ldots] := \max(r_1, \max\,[r_2, r_3, \ldots])$$
$$\max\,[\,] := -\infty$$



$\max$ is *uniquely defined* by $[-\infty, \max]$ via recursion.

... so are many other aggregation functions.

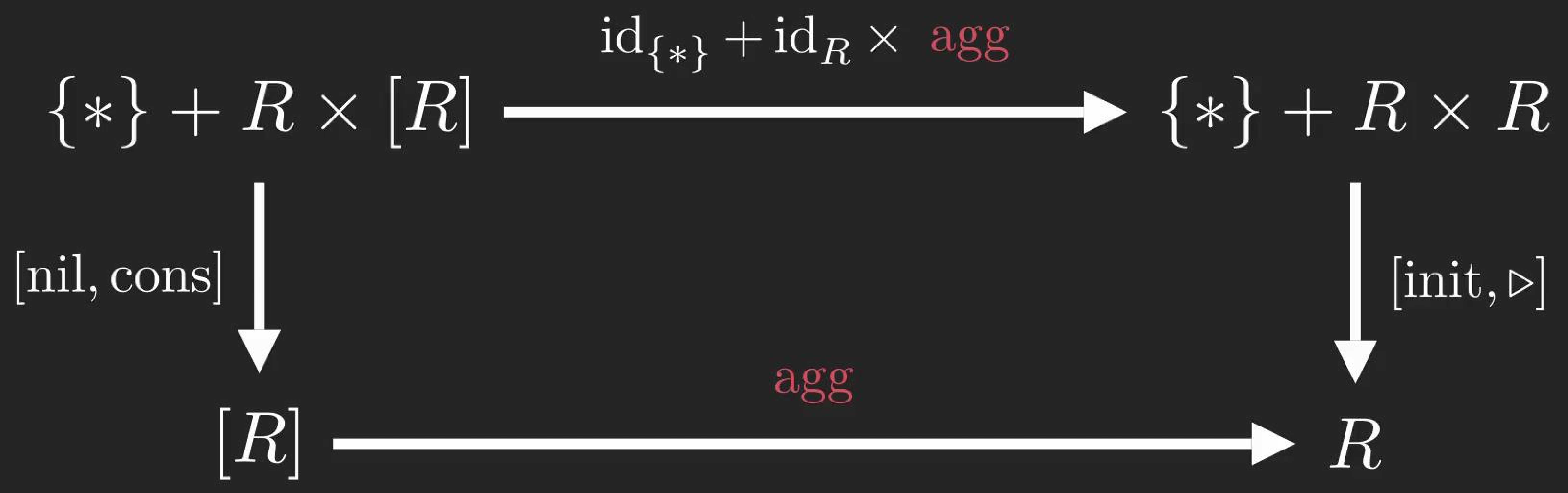
$$\text{agg}\ [r_1, r_2, r_3, \ldots] := r_1 \triangleright \text{agg}[r_2, r_3, \ldots]$$
$$\text{agg}\ [\,] := \text{init}$$



They all have the same *recursive structure*!

... so are many other aggregation functions.

$$\text{agg}\ [r_1, r_2, r_3, \ldots] := r_1 \triangleright \text{agg}[r_2, r_3, \ldots]$$
$$\text{agg}\ [\,] := \text{init}$$



(1) initial value $\text{init} \in R$
(2) update function $\triangleright : R \times R \to R$

We can aggregate multiple statistics at the same time

$$\mathrm{agg}\ [r_1, r_2, r_3, \ldots] := r_1 \triangleright \mathrm{agg}[r_2, r_3, \ldots]$$

$$\mathrm{agg}\ [\,] := \mathrm{init}$$

(1) initial value $\mathrm{init} \in T$

(2) update function $\triangleright : R \times T \to T$

We can aggregate multiple statistics at the same time

$$\text{agg}\ [r_1, r_2, r_3, \ldots] := r_1 \triangleright \text{agg}[r_2, r_3, \ldots]$$

$$\text{agg}\ [\,] := \text{init}$$

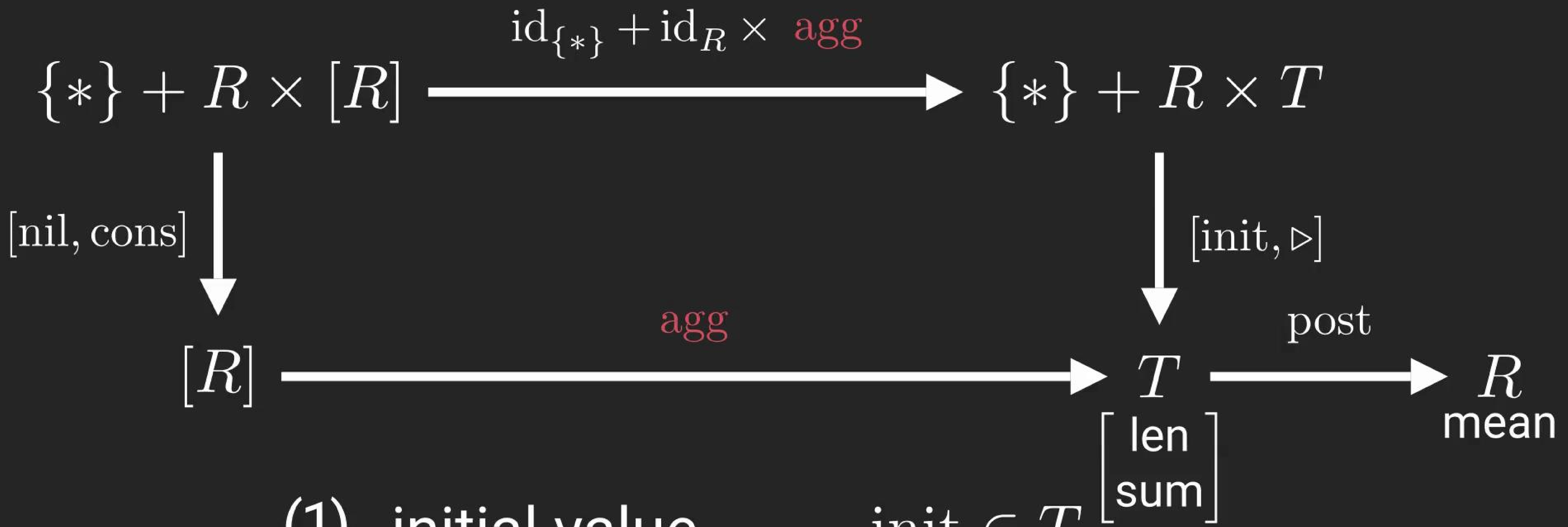
statistic aggregation function

(1) initial value $\text{init} \in T$

(2) update function $\triangleright : R \times T \to T$

... and we can post-process the aggregated statistics.

$$\text{agg}\ [r_1, r_2, r_3, \dots] := r_1 \triangleright \text{agg}[r_2, r_3, \dots]$$
$$\text{agg}\ [\,] := \text{init}$$



(1) initial value $\text{init} \in T$

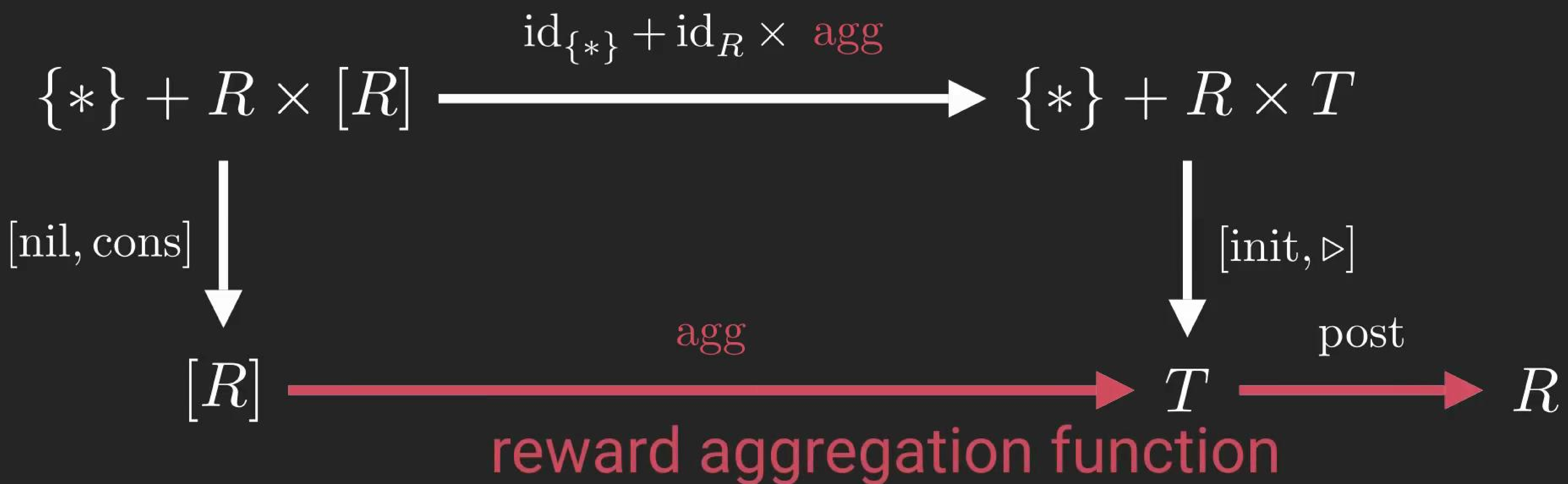
(2) update function $\triangleright : R \times T \to T$

(3) post-processing $\text{post} : T \to R$

... and we can post-process the aggregated statistics.

$$\text{agg}\ [r_1, r_2, r_3, \ldots] := r_1 \triangleright \text{agg}[r_2, r_3, \ldots]$$

$$\text{agg}\ [\,] := \text{init}$$



(1) initial value $\text{init} \in T$

(2) update function $\triangleright : R \times T \to T$

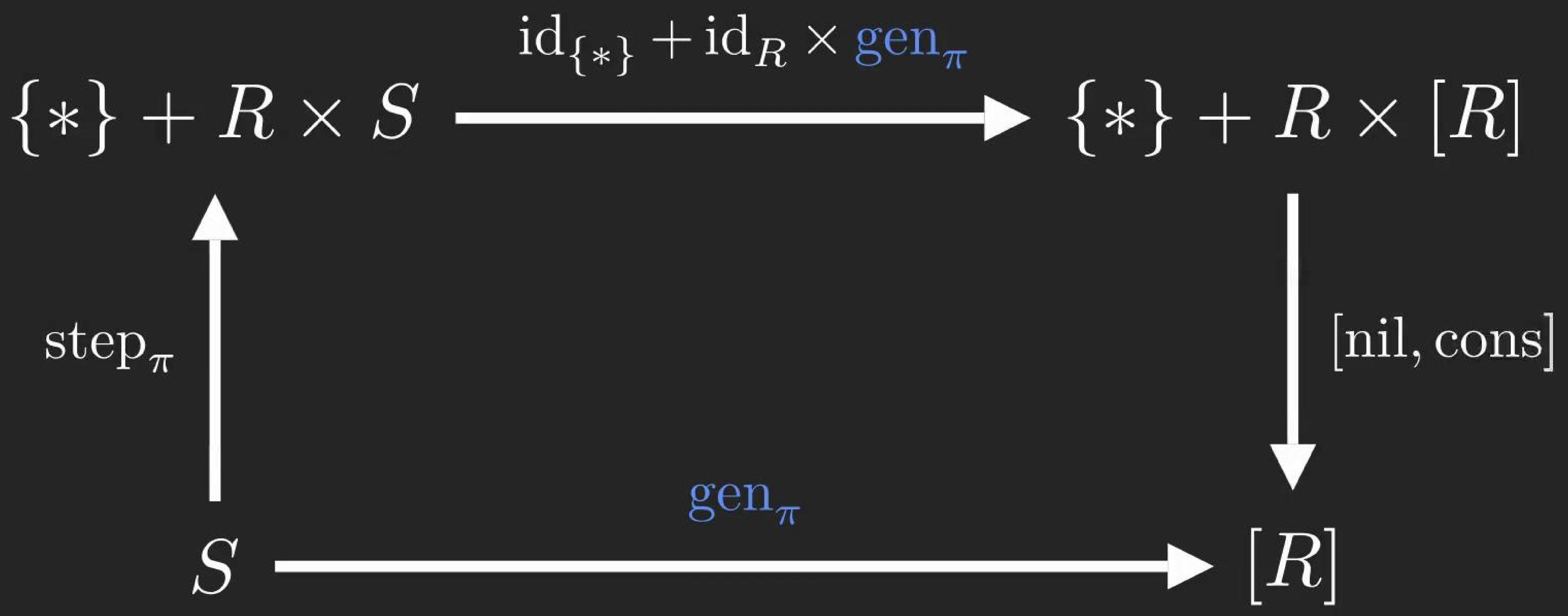
(3) post-processing $\text{post} : T \to R$

	definition $\text{post} \circ \text{agg}_{\text{init},\triangleright} : [R] \to R$	initial value of statistic(s) $\text{init} \in T$	update function $\triangleright : R \times T \to T$	post-processing $\text{post} : T \to R$
discounted sum	$r_1 + \gamma r_2 + \cdots + \gamma^{t-1} r_t$	discounted sum s: $0 \in \mathbb{R}$	$+_\gamma := [r, s \mapsto r + \gamma \cdot s]$	$\text{id}_{\mathbb{R}}$
discounted min	$\min\{r_1, \gamma r_2, \ldots, \gamma^{t-1} r_t\}$	discounted min n: $\infty \in \overline{\mathbb{R}}$	$\min_\gamma := [r, n \mapsto \min(r, \gamma \cdot n)]$	$\text{id}_{\overline{\mathbb{R}}}$
discounted max	$\max\{r_1, \gamma r_2, \ldots, \gamma^{t-1} r_t\}$	discounted max m: $-\infty \in \overline{\mathbb{R}}$	$\max_\gamma := [r, m \mapsto \max(r, \gamma \cdot m)]$	$\text{id}_{\overline{\mathbb{R}}}$
log-sum-exp	$\log(e^{r_1} + e^{r_2} + \cdots + e^{r_t})$	log-sum-exp m: $-\infty \in \overline{\mathbb{R}}$	$[r, m \mapsto \log(e^r + e^m)]$	$\text{id}_{\overline{\mathbb{R}}}$
range	$\max(r_{1:t}) - \min(r_{1:t})$	max m, min n $\begin{bmatrix} -\infty \\ \infty \end{bmatrix} \in \overline{\mathbb{R}}^2$	$\left[r, \begin{bmatrix} m \\ n \end{bmatrix} \mapsto \begin{bmatrix} \max(r, m) \\ \min(r, n) \end{bmatrix}\right]$	$\left[\begin{bmatrix} m \\ n \end{bmatrix} \mapsto m - n\right]$
mean	$\bar{r} := \frac{1}{t} \sum_{i=1}^t r_i$	length n, sum s $\begin{bmatrix} 0 \\ 0 \end{bmatrix} \in \begin{bmatrix} \mathbb{N} \\ \mathbb{R} \end{bmatrix}$	$\left[r, \begin{bmatrix} n \\ s \end{bmatrix} \mapsto \begin{bmatrix} n+1 \\ s+r \end{bmatrix}\right]$	$\left[\begin{bmatrix} n \\ s \end{bmatrix} \mapsto \frac{s}{n}\right]$
		length n, mean m $\begin{bmatrix} 0 \\ 0 \end{bmatrix} \in \begin{bmatrix} \mathbb{N} \\ \mathbb{R} \end{bmatrix}$	$\left[r, \begin{bmatrix} n \\ m \end{bmatrix} \mapsto \begin{bmatrix} n+1 \\ \frac{n \cdot m + r}{n+1} \end{bmatrix}\right]$	$\left[\begin{bmatrix} n \\ m \end{bmatrix} \mapsto m\right]$
variance	$\frac{1}{t} \sum_{i=1}^t (r_i - \bar{r})^2 = \overline{r^2} - \bar{r}^2$	length n, sum s, sum square q $\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \in \begin{bmatrix} \mathbb{N} \\ \mathbb{R} \\ \mathbb{R}_{\geq 0} \end{bmatrix}$	$\left[r, \begin{bmatrix} n \\ s \\ q \end{bmatrix} \mapsto \begin{bmatrix} n+1 \\ s+r \\ q+r^2 \end{bmatrix}\right]$	$\left[\begin{bmatrix} n \\ s \\ q \end{bmatrix} \mapsto \frac{q}{n} - \left(\frac{s}{n}\right)^2\right]$
		length n, mean m, variance v $\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \in \begin{bmatrix} \mathbb{N} \\ \mathbb{R} \\ \mathbb{R}_{\geq 0} \end{bmatrix}$	$\left[r, \begin{bmatrix} n \\ m \\ v \end{bmatrix} \mapsto \begin{bmatrix} n+1 \\ \frac{n \cdot m + r}{n+1} \\ v + \frac{n(r-m)^2 - (n+1)v}{(n+1)^2} \end{bmatrix}\right]$	$\left[\begin{bmatrix} n \\ m \\ v \end{bmatrix} \mapsto v\right]$
top-k	k-th largest in $r_{1:t}$	top-k buffer: top-1, top-2, $\vdots$ $\begin{bmatrix} -\infty \\ -\infty \\ \vdots \end{bmatrix} \in \overline{\mathbb{R}}^k$	$\left[r, b \mapsto \begin{cases} \text{insert}(r, b) & r > \min b \\ b & r \leq \min b \end{cases}\right]$	$[b \mapsto \min b]$

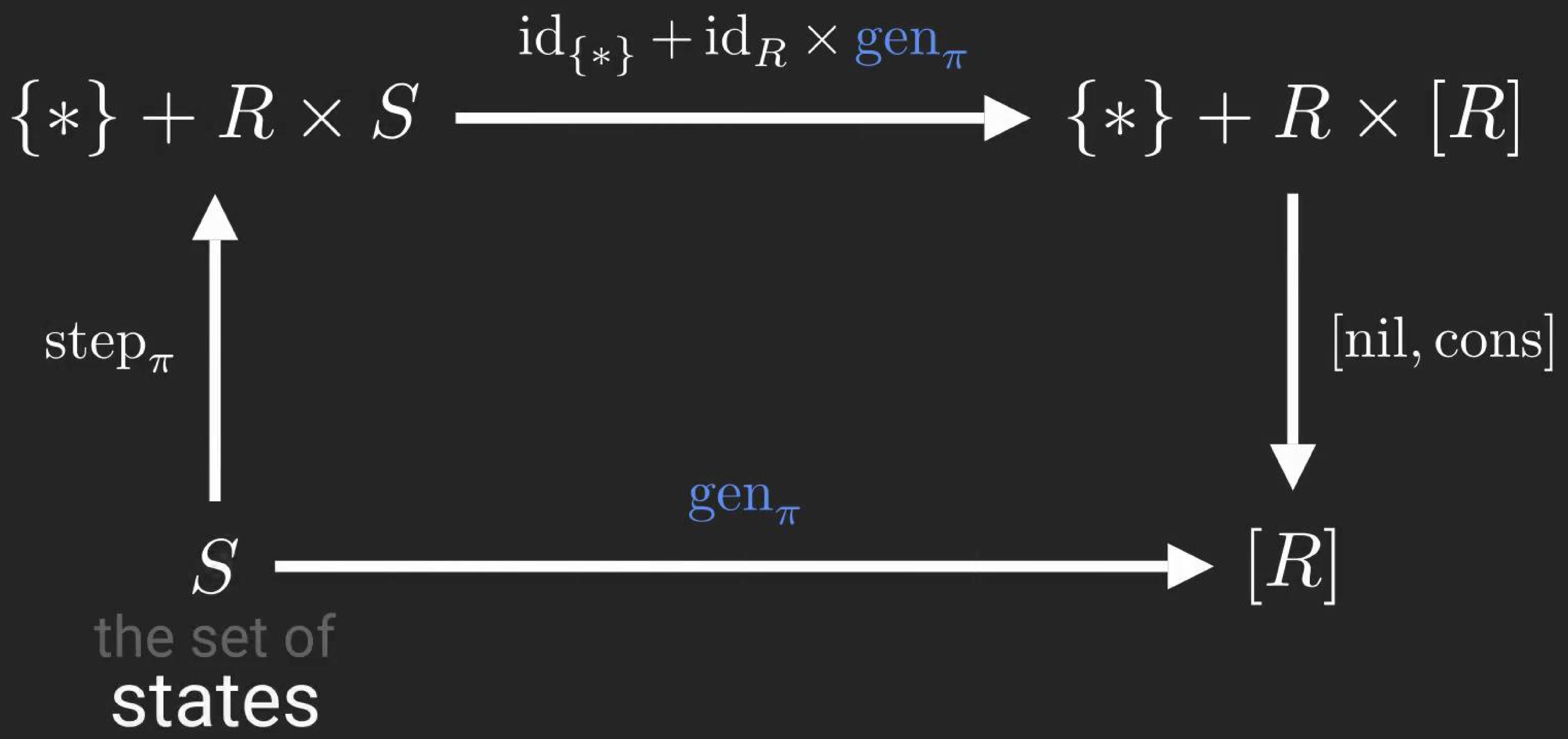
Why does this matter?

	definition $\mathrm{post}\circ\mathrm{agg}_{\mathrm{init},\triangleright} : [R]\to R$	initial value of statistic(s) $\mathrm{init}\in T$	update function $\triangleright : R\times T\to T$	post-processing $\mathrm{post} : T\to R$
discounted sum	$r_1+\gamma r_2+\cdots+\gamma^{t-1}r_t$	discounted sum s: $0\in\mathbb{R}$	$+_\gamma := [r,s\mapsto r+\gamma\cdot s]$	$\mathrm{id}_{\mathbb{R}}$
discounted min	$\min\{r_1,\gamma r_2,\ldots,\gamma^{t-1}r_t\}$	discounted min n: $\infty\in\overline{\mathbb{R}}$	$\min_\gamma := [r,n\mapsto \min(r,\gamma\cdot n)]$	$\mathrm{id}_{\overline{\mathbb{R}}}$
discounted max	$\max\{r_1,\gamma r_2,\ldots,\gamma^{t-1}r_t\}$	discounted max m: $-\infty\in\overline{\mathbb{R}}$	$\max_\gamma := [r,m\mapsto \max(r,\gamma\cdot m)]$	$\mathrm{id}_{\overline{\mathbb{R}}}$
log-sum-exp	$\log(e^{r_1}+e^{r_2}+\cdots+e^{r_t})$	log-sum-exp m: $-\infty\in\overline{\mathbb{R}}$	$[r,m\mapsto\log(e^r+e^m)]$	$\mathrm{id}_{\overline{\mathbb{R}}}$
range	$\max(r_{1:t})-\min(r_{1:t})$	max m, min n $\begin{bmatrix}-\infty\\ \infty\end{bmatrix}\in\overline{\mathbb{R}}^2$	$\left[r,\begin{bmatrix}m\\ n\end{bmatrix}\mapsto\begin{bmatrix}\max(r,m)\\ \min(r,n)\end{bmatrix}\right]$	$\left[\begin{bmatrix}m\\ n\end{bmatrix}\mapsto m-n\right]$
mean	$\bar{r} := \frac{1}{t}\sum_{i=1}^{t} r_i$	length n, sum s $\begin{bmatrix}0\\ 0\end{bmatrix}\in\begin{bmatrix}\mathbb{N}\\ \mathbb{R}\end{bmatrix}$	$\left[r,\begin{bmatrix}n\\ s\end{bmatrix}\mapsto\begin{bmatrix}n+1\\ s+r\end{bmatrix}\right]$	$\left[\begin{bmatrix}n\\ s\end{bmatrix}\mapsto \frac{s}{n}\right]$
		length n, mean m $\begin{bmatrix}0\\ 0\end{bmatrix}\in\begin{bmatrix}\mathbb{N}\\ \mathbb{R}\end{bmatrix}$	$\left[r,\begin{bmatrix}n\\ m\end{bmatrix}\mapsto\begin{bmatrix}n+1\\ \frac{n\cdot m+r}{n+1}\end{bmatrix}\right]$	$\left[\begin{bmatrix}n\\ m\end{bmatrix}\mapsto m\right]$
variance	$\frac{1}{t}\sum_{i=1}^{t}(r_i-\bar{r})^2=\overline{r^2}-\bar{r}^2$	length n, sum s, sum square q $\begin{bmatrix}0\\ 0\\ 0\end{bmatrix}\in\begin{bmatrix}\mathbb{N}\\ \mathbb{R}\\ \mathbb{R}_{\geq 0}\end{bmatrix}$	$\left[r,\begin{bmatrix}n\\ s\\ q\end{bmatrix}\mapsto\begin{bmatrix}n+1\\ s+r\\ q+r^2\end{bmatrix}\right]$	$\left[\begin{bmatrix}n\\ s\\ q\end{bmatrix}\mapsto \frac{q}{n}-\left(\frac{s}{n}\right)^2\right]$
		length n, mean m, variance v $\begin{bmatrix}0\\ 0\\ 0\end{bmatrix}\in\begin{bmatrix}\mathbb{N}\\ \mathbb{R}\\ \mathbb{R}_{\geq 0}\end{bmatrix}$	$\left[r,\begin{bmatrix}n\\ m\\ v\end{bmatrix}\mapsto\begin{bmatrix}n+1\\ \frac{n\cdot m+r}{n+1}\\ v+\frac{n(r-m)^2-(n+1)v}{(n+1)^2}\end{bmatrix}\right]$	$\left[\begin{bmatrix}n\\ m\\ v\end{bmatrix}\mapsto v\right]$
top-k	k-th largest in $r_{1:t}$	top-k buffer: top-1, top-2, $\vdots$ $\begin{bmatrix}-\infty\\ -\infty\\ \vdots\end{bmatrix}\in\overline{\mathbb{R}}^k$	$\left[r,b\mapsto\begin{cases}\mathrm{insert}(r,b) & r>\min b\\ b & r\leq\min b\end{cases}\right]$	$[b\mapsto\min b]$

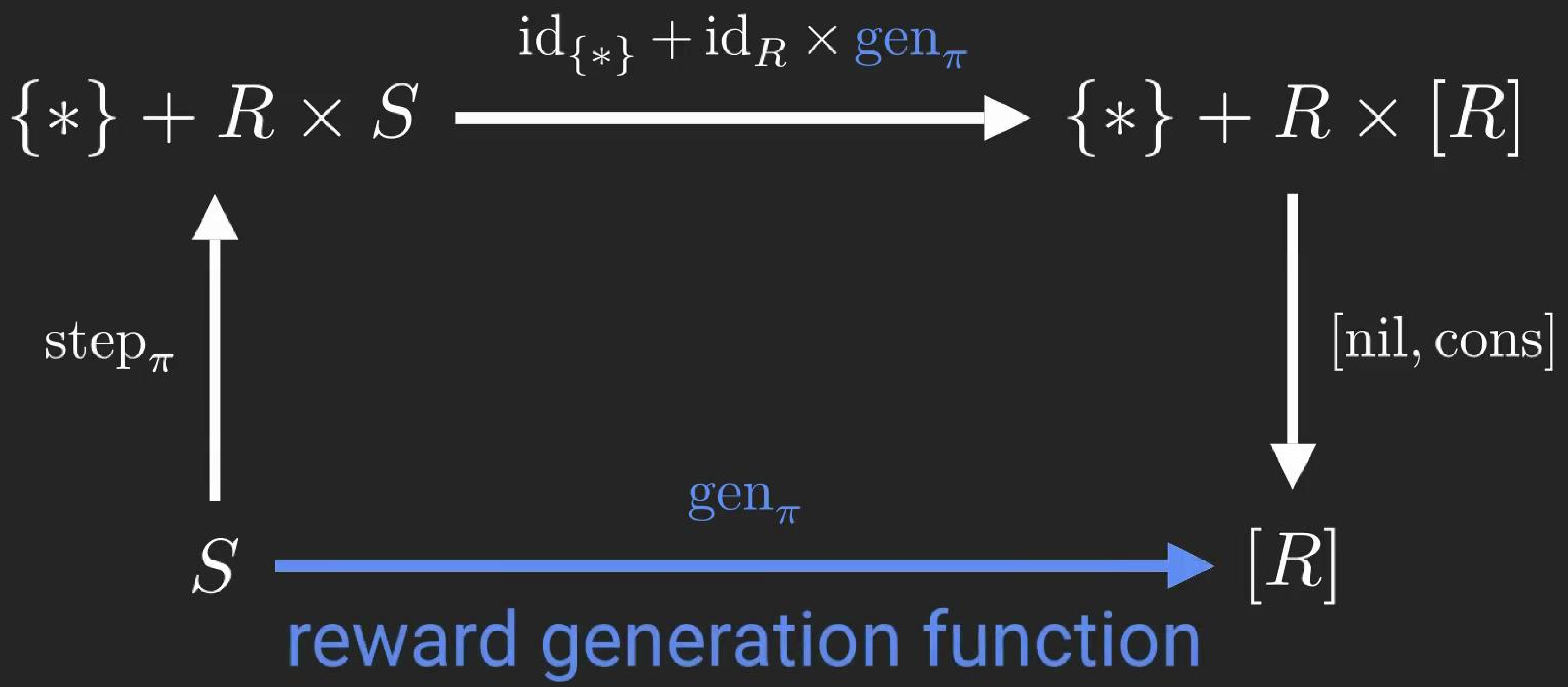
Reward generation is also a *recursive function*

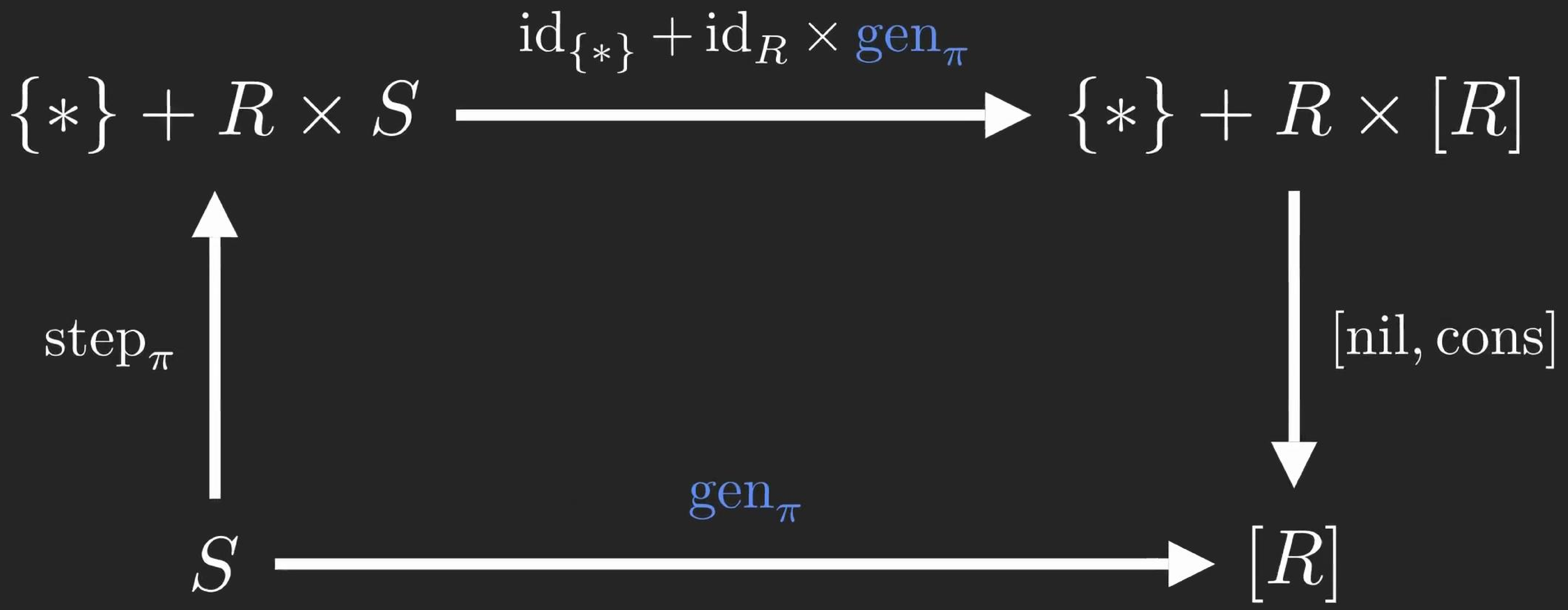


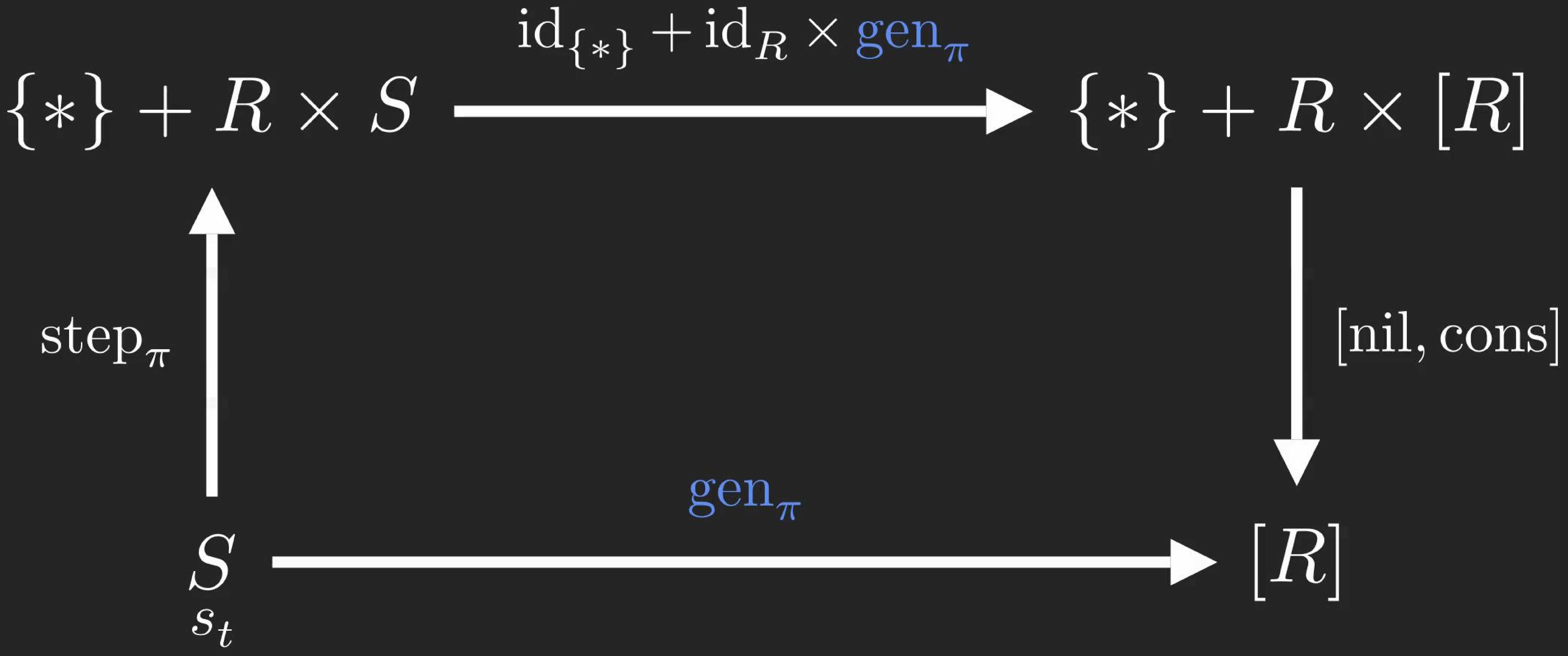
Reward generation is also a *recursive function*



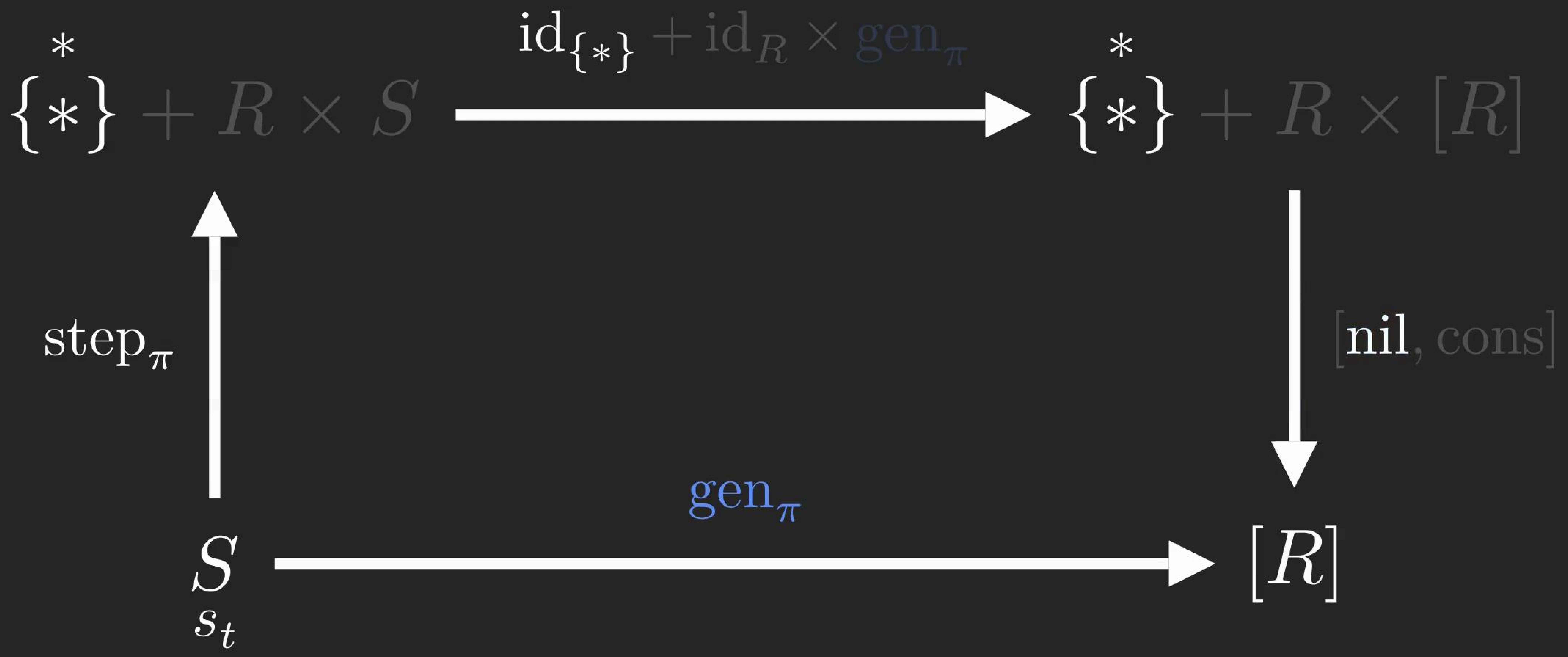
Reward generation is also a *recursive function*



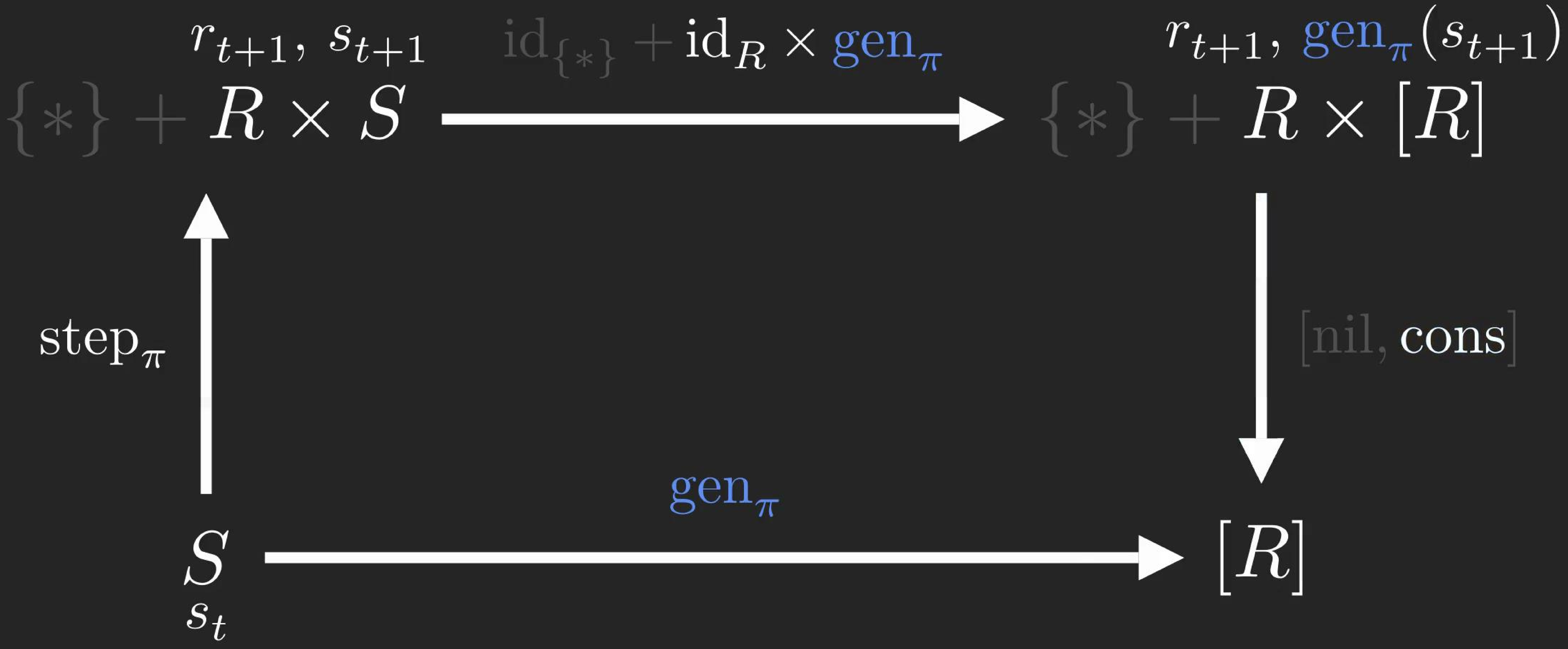

$\mathrm{id}_{\{*\}} + \mathrm{id}_R \times \mathrm{gen}_\pi$
$\{*\} + R \times S$
$\{*\} + R \times [R]$
step_π
$[\mathrm{nil}, \mathrm{cons}]$
gen_π
S
$[R]$



$$\mathrm{gen}_\pi(s_t) = \{$$

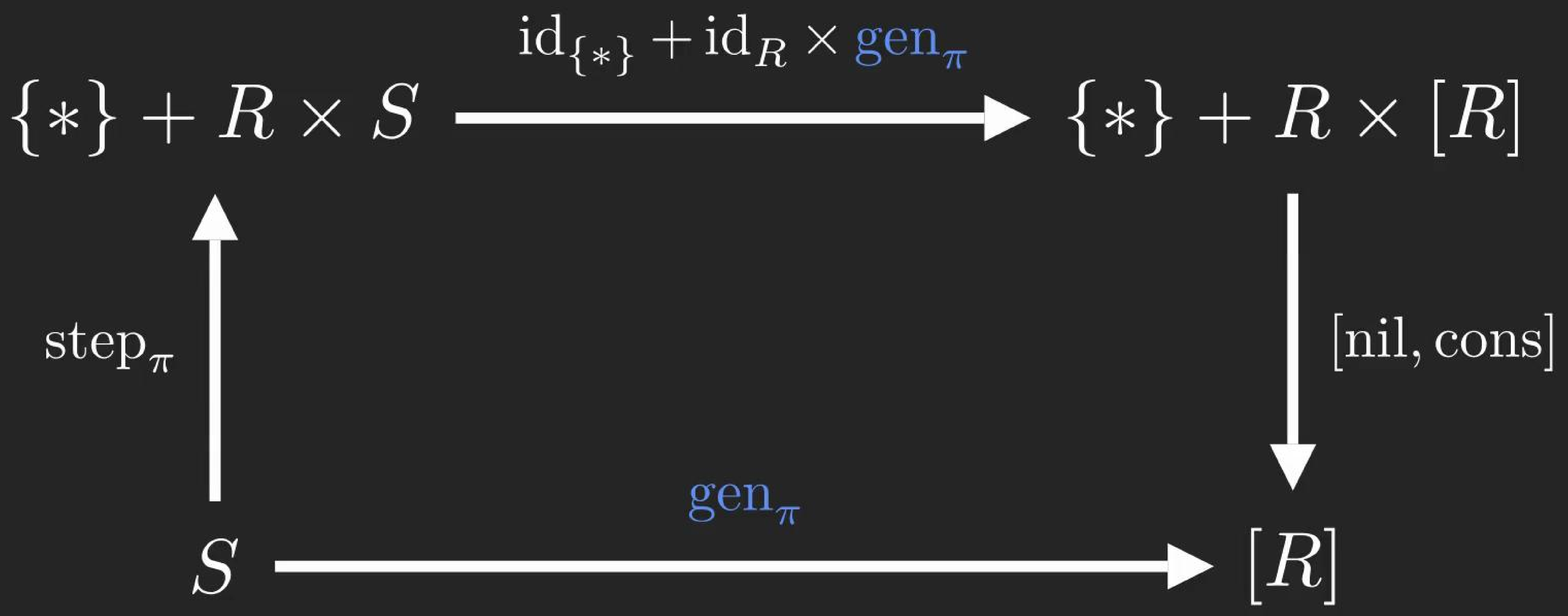


$$\mathrm{gen}_\pi(s_t) = \begin{cases} [\,] & \text{terminal} \\ & \end{cases}$$



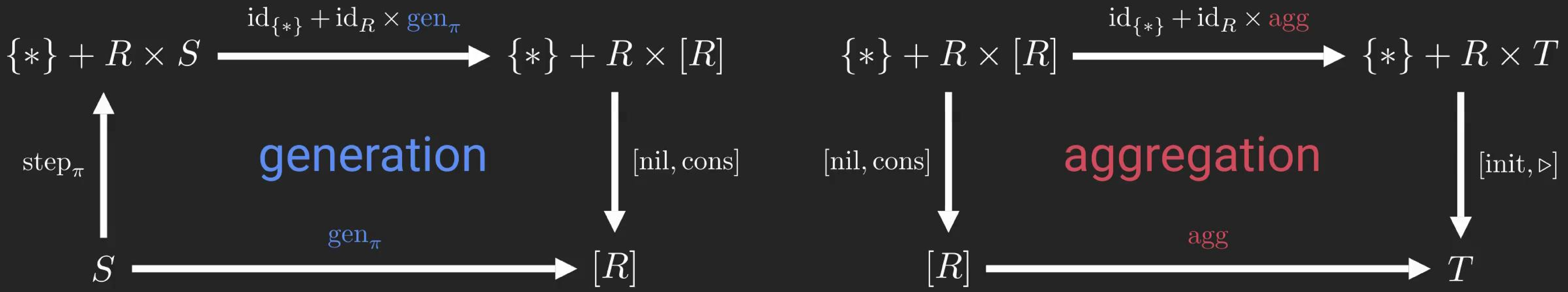
$$\mathrm{gen}_\pi(s_t) = \begin{cases} [\,] & \text{terminal} \\ r_{t+1} : \mathrm{gen}_\pi(s_{t+1}) & \text{otherwise} \end{cases}$$

Reward generation is also a *recursive function*



gen_π is *uniquely defined* by step_π via recursion.

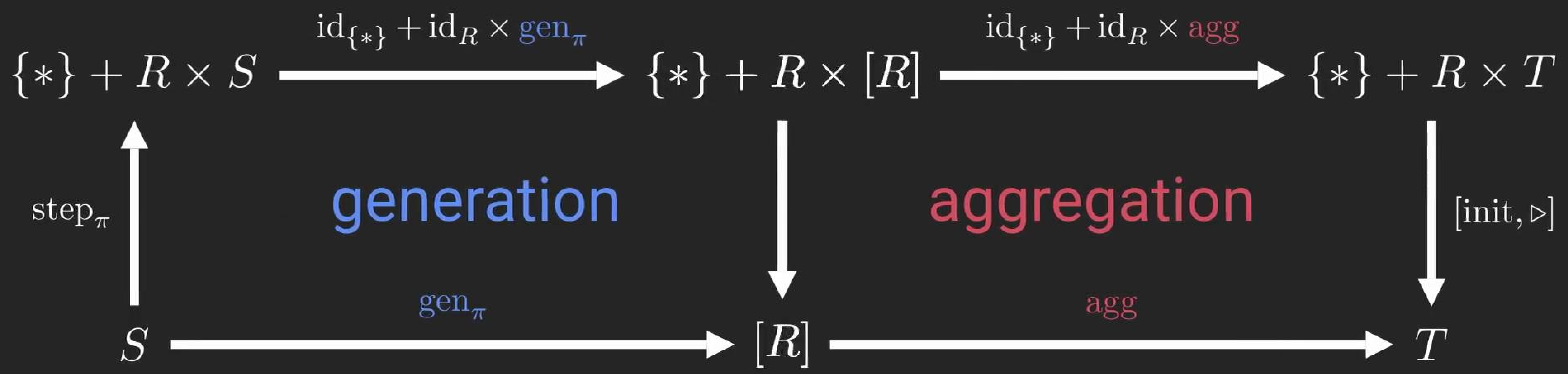
Recursive generation and aggregation of rewards



gen_π is *uniquely defined* by step_π via recursion.

agg is *uniquely defined* by $[\mathrm{init}, \triangleright]$ via recursion.

Recursive generation and aggregation of rewards lead to generalized Bellman equations.



gen_π is *uniquely defined* by step_π via recursion.

agg is *uniquely defined* by $[\mathrm{init}, \triangleright]$ via recursion.

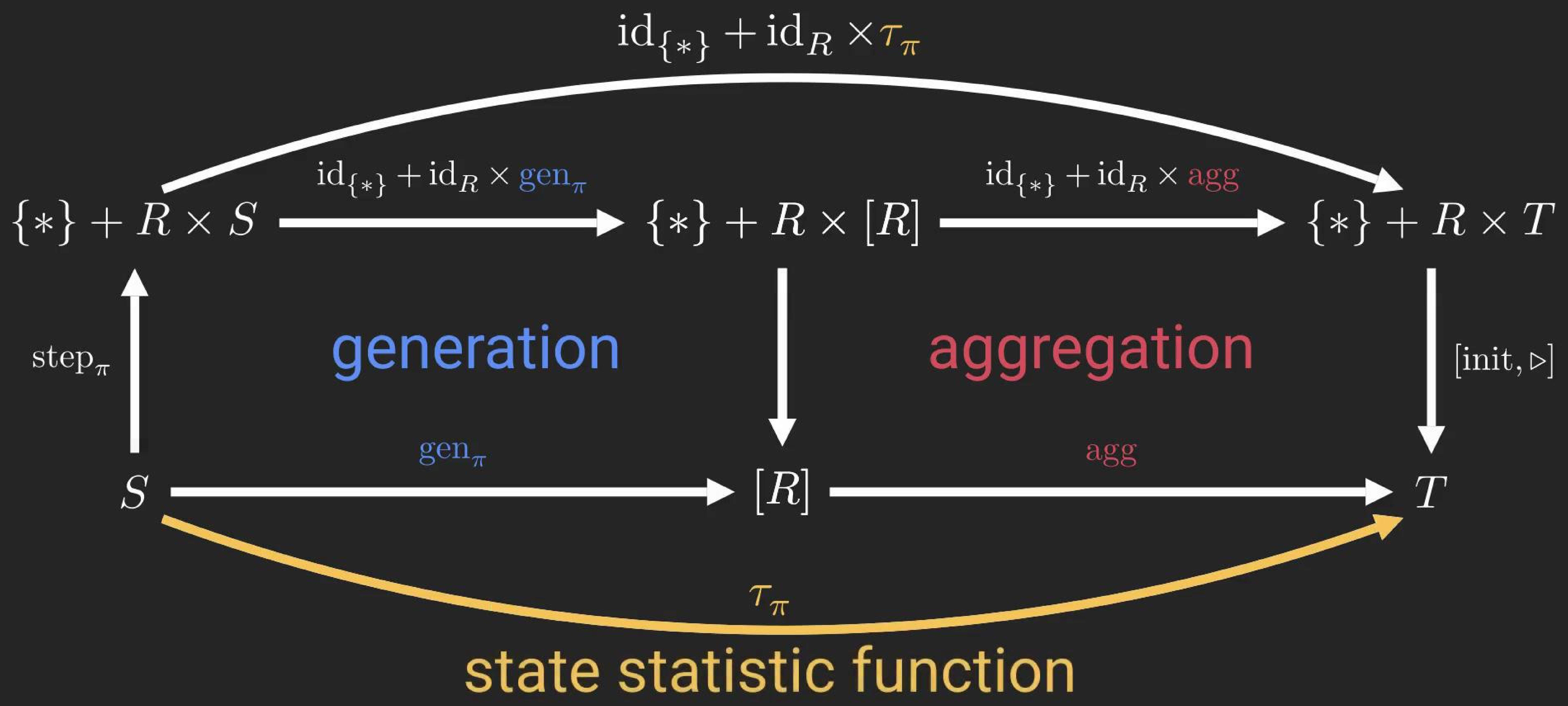
Recursive generation and aggregation of rewards lead to generalized Bellman equations.

Hinze et al. (2010)

gen_π is *uniquely defined* by step_π via recursion.

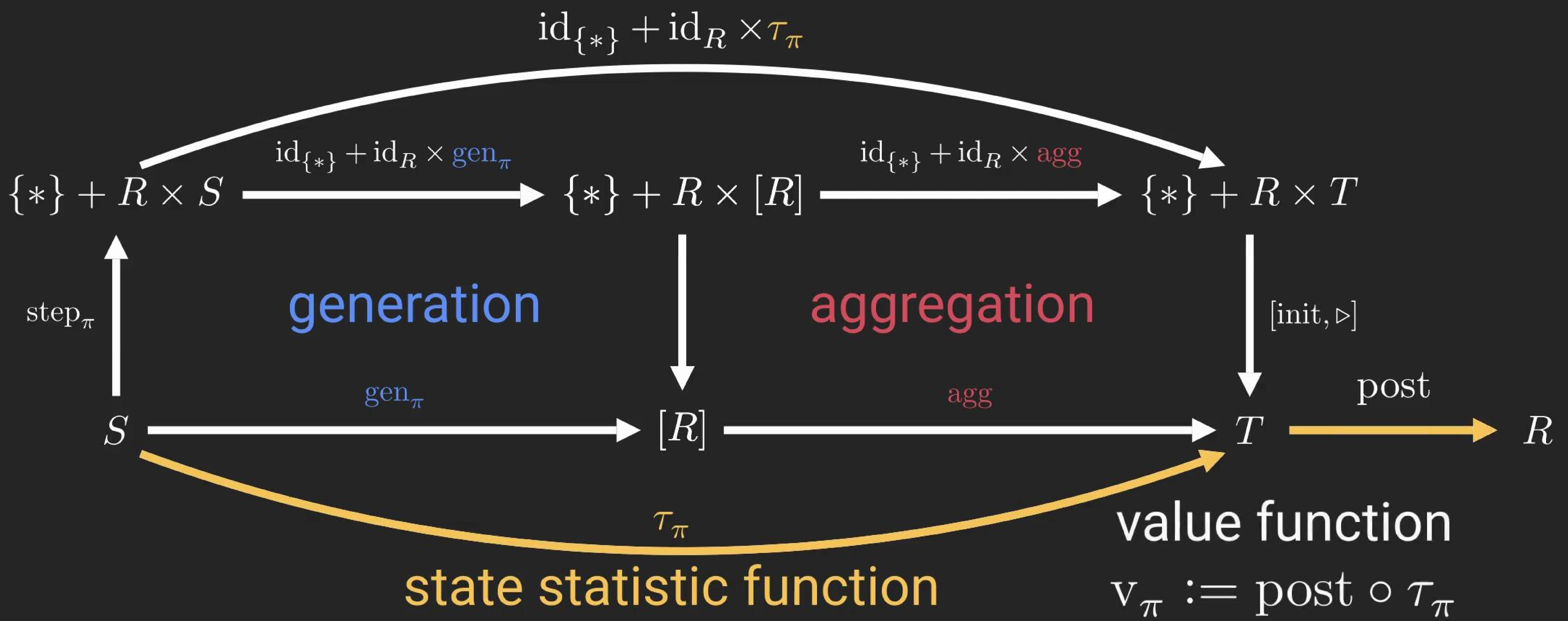
agg is *uniquely defined* by $[\mathrm{init}, \triangleright]$ via recursion.

Recursive generation and aggregation of rewards lead to generalized Bellman equations.



$\tau_\pi = \mathrm{agg} \circ \mathrm{gen}_\pi$ is *uniquely defined* by step_π and $[\mathrm{init}, \triangleright]$ via recursion.

Recursive generation and aggregation of rewards lead to generalized Bellman equations.



$\tau_\pi = \mathrm{agg} \circ \mathrm{gen}_\pi$ is *uniquely defined* by step_π and $[\mathrm{init}, \triangleright]$ via recursion.

Theorem (Bellman equation for state statistic function)

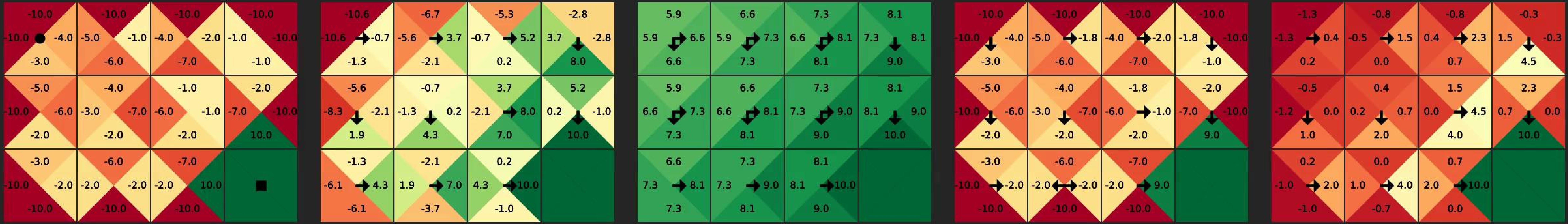
$$\tau_\pi(s_t) = \begin{cases} \text{init} & \text{terminal} \\ r_{t+1} \triangleright \tau_\pi(s_{t+1}) & \text{otherwise} \end{cases}$$

For *theorists*:

- Recursive generation and aggregation $\to$ Bellman equations
- Preorder and premetric structures $\to$ convergence behaviors
- Unified deterministic and stochastic formulations

For *practitioners*:

- You can use your favorite value/critic-based algorithms.
- You can directly optimize the *Sharpe ratio* ($\frac{\text{mean}}{\text{std}}$) in finance, or regularize the velocity *range* ($\max - \min$) in continuous control.



Grid | $\mathrm{sum}_{0.99}$ | $\max_{0.99}$ | $\min_{0.99}$ | mean

$\mathrm{sum}_{0.99}$ | $\max_{0.99}$ | $\min$ | $\mathrm{sum}_{0.99} + \max_{0.99}$ | $\mathrm{sum}_{0.99} - \mathrm{var}$

Q-learning, PPO, TD3, . . .

Mean, variance, range, the Sharpe ratio in portfolio optimization, . . .

Recursive Reward Aggregation

Summary

- An algebraic perspective on Markov decision processes
- Generalized Bellman equations and Bellman operators
- Integration into value-based and actor-critic algorithms

Future work

- Multi-dimensional or non-numerical feedback?
- Agent states? **Automata** as aggregators?
- List $\to$ **tree**, list function $\to$ **tree traversal**?
- Logic, reasoning, safety, and alignment?

Recursive Reward Aggregation

Yuting Tang Yivan Zhang Johannes Ackermann
Yu-Jie Zhang Soichiro Nishimori Masashi Sugiyama

UTokyo RIKEN

Reinforcement Learning Conference 2025