TÉCNICO LISBOA

Planning, Learning and Decision Making

Lecture 4. Hidden Markov models (cont.)

Quick recap

Hidden Markov model

Markov state

The state at instant t is enough to predict the state at instant $t + 1$:

$$\mathbb{P}\left[\mathrm{x}_{t+1} = y \mid \mathbf{x}_{0:t} = \boldsymbol{x}_{0:t}, \mathbf{z}_{0:t} = \boldsymbol{z}_{0:t}\right] = \mathbb{P}\left[\mathrm{x}_{t+1} = y \mid \mathrm{x}_t = x_t\right]$$

State-dependent observations

The state at instant t is enough to predict the observation at instant t:

$$\mathbb{P}\left[\mathrm{z}_t = z \mid \mathbf{x}_{0:t} = \boldsymbol{x}_{0:t}, \mathbf{z}_{0:t-1} = \boldsymbol{z}_{0:t-1}\right] = \mathbb{P}\left[\mathrm{z}_t = z \mid \mathrm{x}_t = x_t\right]$$

Summarizing...

- A HMM can be represented compactly as a tuple

$$(\mathcal{X}, \mathcal{Z}, \mathbf{P}, \mathbf{O})$$

  - $\mathcal{X}$ is the set of possible states
  - $\mathcal{Z}$ is the set of possible observations
  - $\mathbf{P}$ is the transition probability matrix
  - $\mathbf{O}$ is the observation probability matrix

Estimation

- **Filtering:**
  - Given a sequence of observations, estimate the final state
- **Smoothing:**
  - Given a sequence of observations, estimate the sequence of states
- **Prediction:**
  - Given a sequence of observations, predict future states

Estimation

- **Filtering:** Forward algorithm
  - Given a sequence of observations, estimate the final state
- **Smoothing:**
  - Given a sequence of observations, estimate the sequence of states
- **Prediction:**
  - Given a sequence of observations, predict future states

Forward mapping

Forward mapping

Given a sequence of observations $\mathbf{z}_{0:t}$, the forward mapping $\alpha_t : \mathcal{X} \longmapsto \mathbb{R}$ is defined for each t as

$$\alpha_t(x) = \mathbb{P}_{\mu_0}\left[\mathrm{x}_t = x, \mathbf{z}_{0:t} = z_{0:t}\right]$$

How the past relates
to the present

Forward algorithm

Require: Observation sequence $\boldsymbol{z}_{0:T}$

1. Initialize $\boldsymbol{\alpha}_0 \leftarrow \mathrm{diag}(\mathbf{O}_{:,z_0})\boldsymbol{\mu}_0^\top$

2. **for** $t = 1, \ldots, T$ **do**

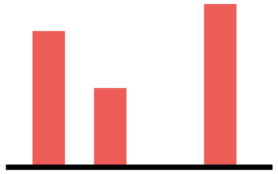
$$\boldsymbol{\alpha}_t \leftarrow \mathrm{diag}(\mathbf{O}_{:,z_t})\mathbf{P}^\top\boldsymbol{\alpha}_{t-1}$$

4. end for

5. **return** $\boldsymbol{\mu}_{T|0:T} = \alpha_T/(\mathbf{1}^\top\boldsymbol{\alpha}_T)$

Forward algorithm

z_0 z_1 z_2 z_T



Initial distribution

Forward algorithm

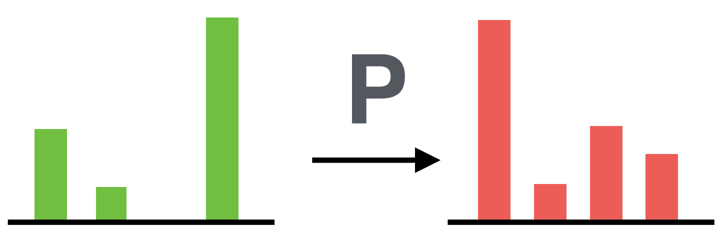
z_0 z_1 z_2 z_T

"Improved" distribution

$$\mathrm{diag}(\mathbf{O}_{:,z_0})\boldsymbol{\mu}_0^\top$$

Forward algorithm

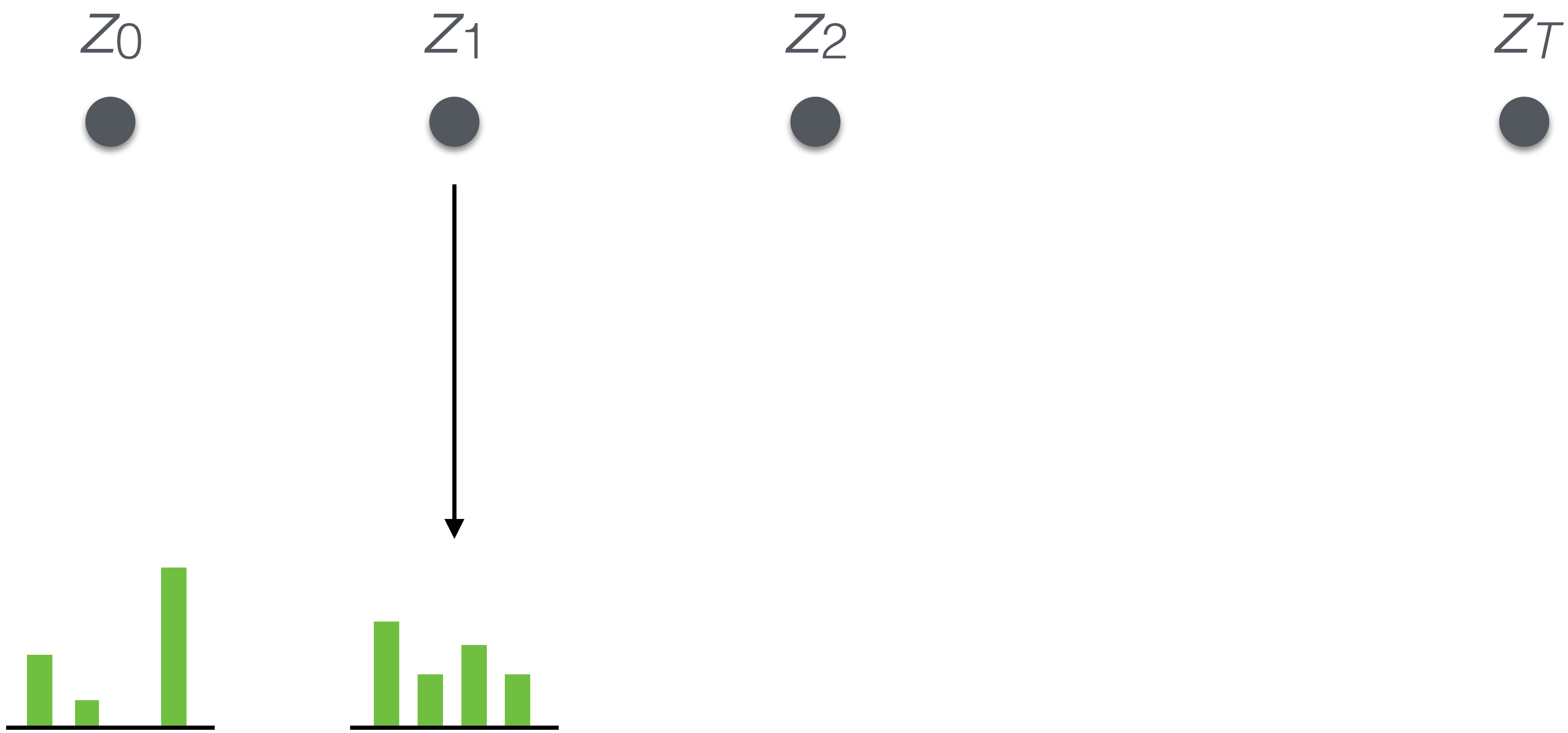
z_0 z_1 z_2 z_T



Predicted distribution

$$\mathbf{P}^\top \boldsymbol{\alpha}_0$$

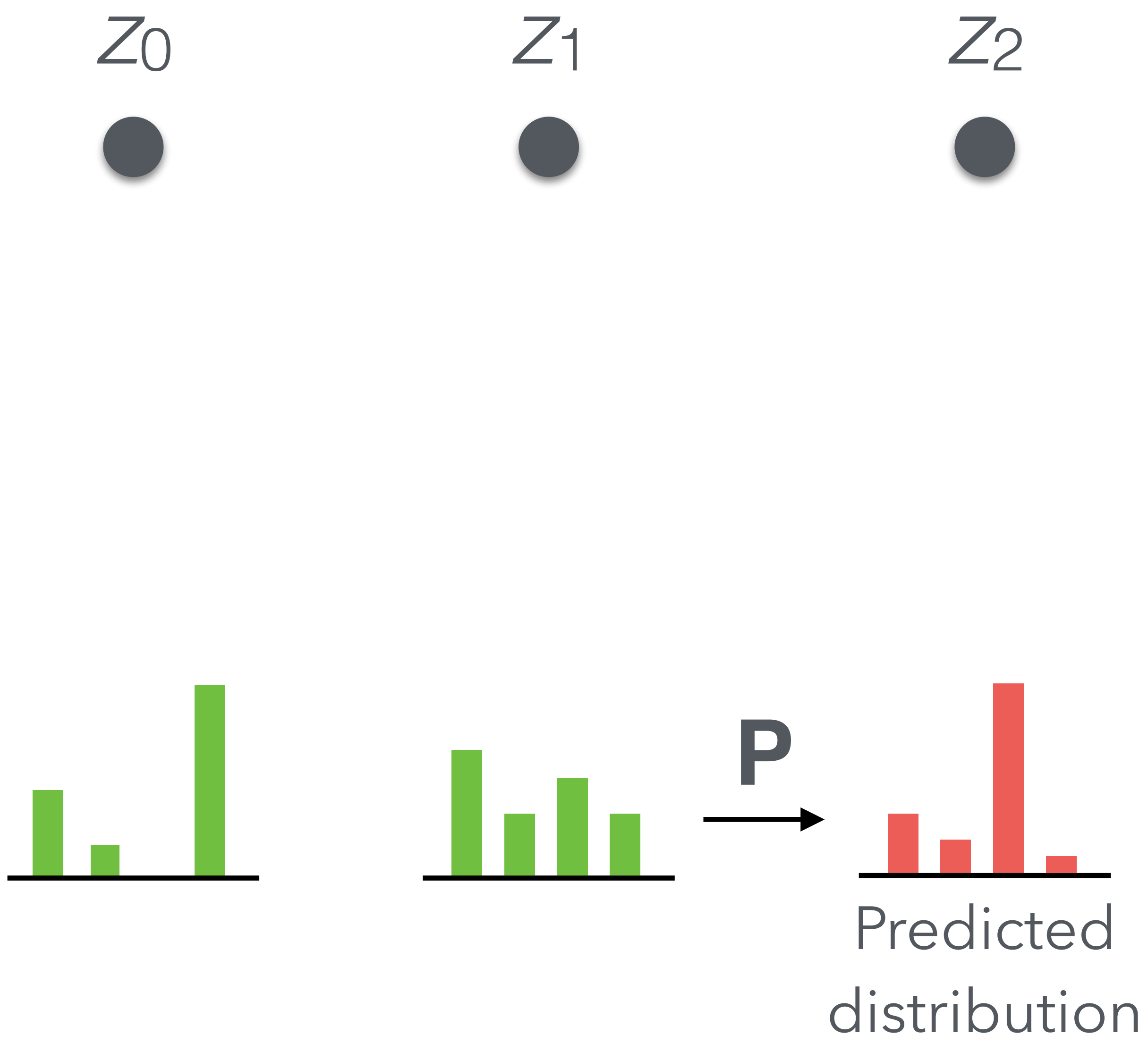
Forward algorithm



"Improved"
distribution

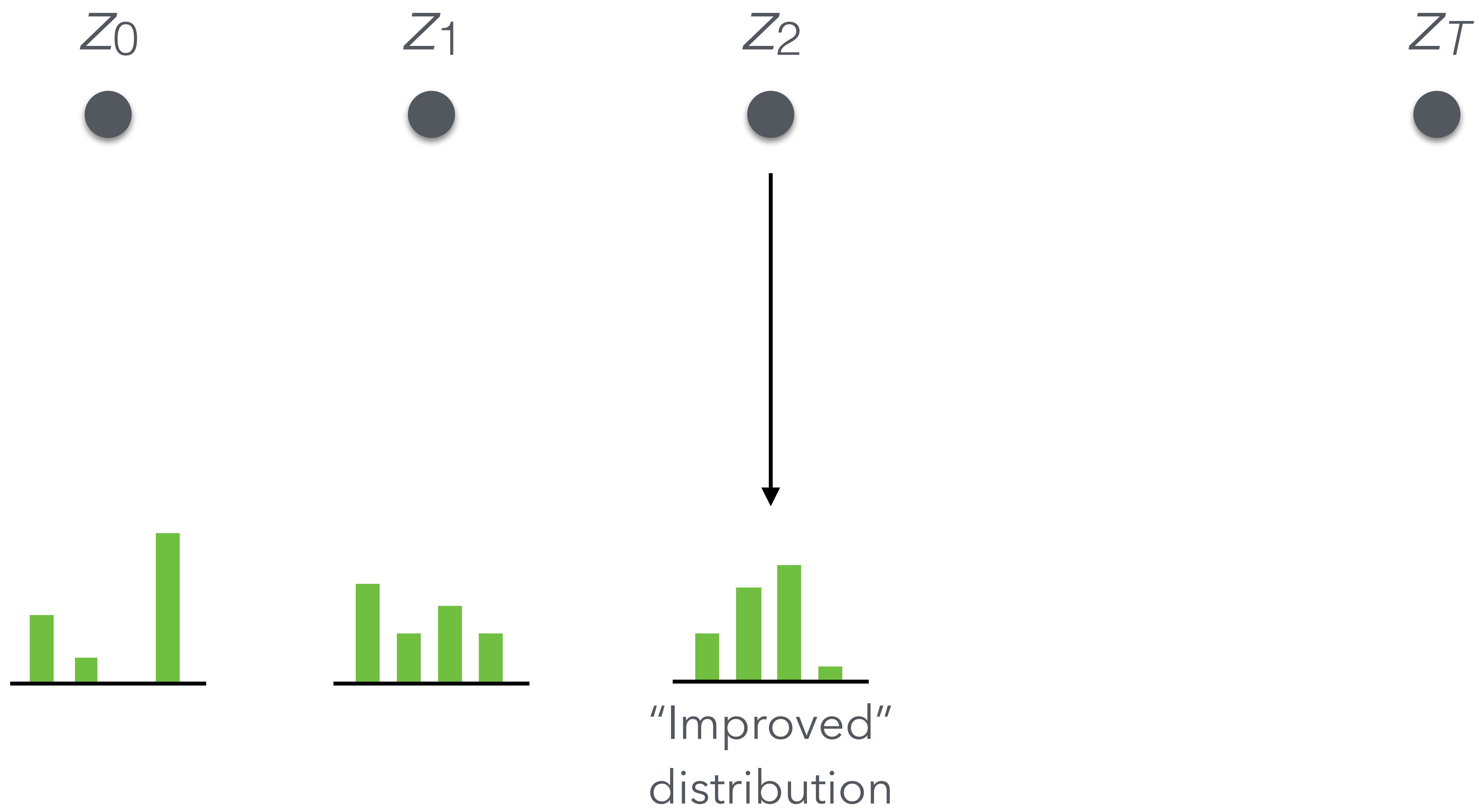
$$\mathrm{diag}(\mathbf{O}_{:,z_1})\mathbf{P}^\top\boldsymbol{\alpha}_0$$

Forward algorithm



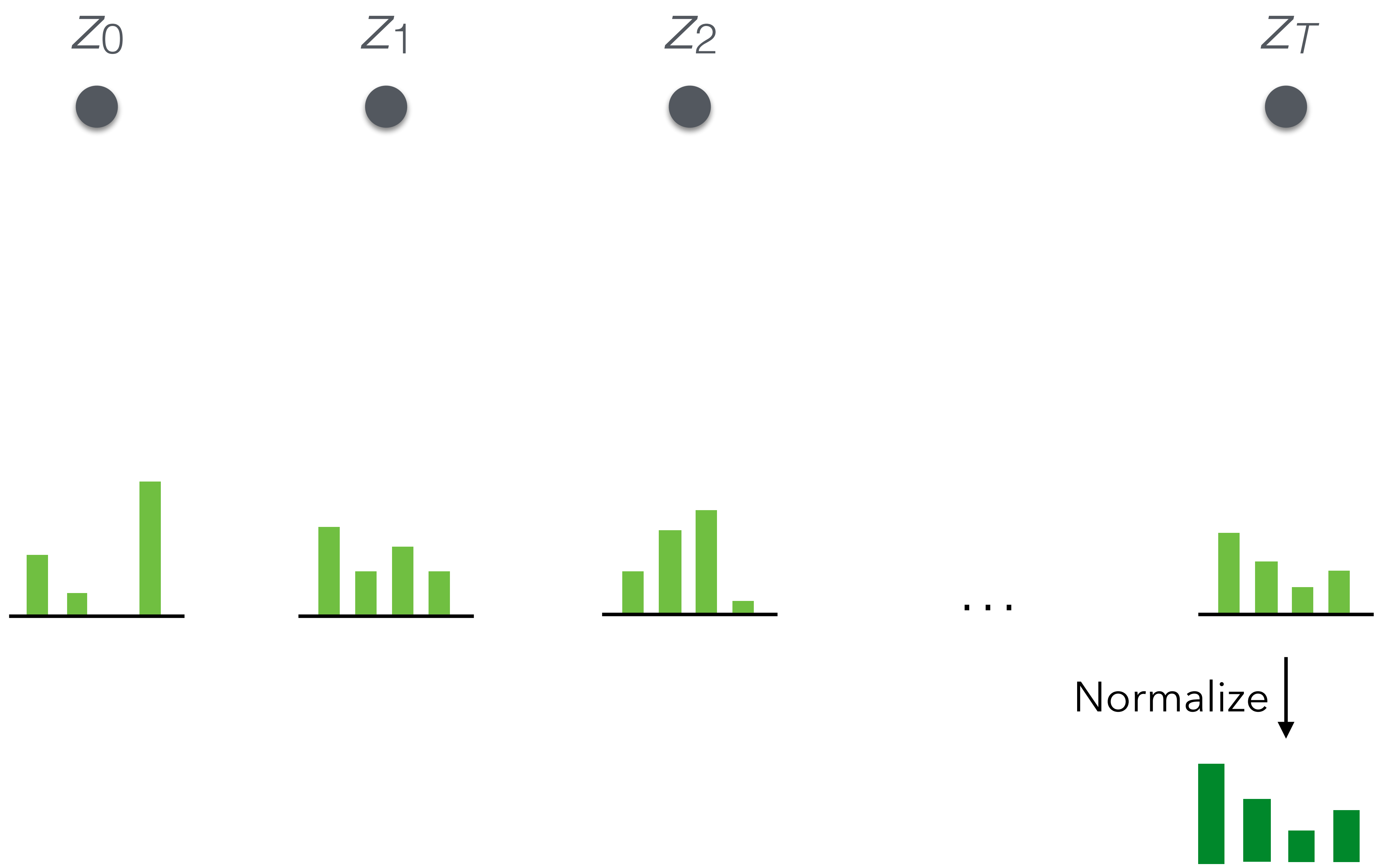
$$\mathbf{P}^\top \boldsymbol{\alpha}_1$$

Forward algorithm



$$\text{diag}(\mathbf{O}_{:,z_2})\mathbf{P}^\top \boldsymbol{\alpha}_1$$

Forward algorithm



Forward algorithm

Given observation sequence $\boldsymbol{z}_{0:T}$

1. Multiply initial distribution by $\mathbf{O}(z_0 \mid :)$

Improve initial distribution

Forward algorithm

Given observation sequence $\boldsymbol{z}_{0:T}$

1. Multiply initial distribution by $\mathbf{O}(z_0 \mid :)$

2. At each time step:

   a. Multiply current distribution by $\mathbf{P}$

Predict 1-step move

Forward algorithm

Given observation sequence $\boldsymbol{z}_{0:T}$

1. Multiply initial distribution by $\mathbf{O}(z_0 \mid :)$
2. At each time step:
   a. Multiply current distribution by $\mathbf{P}$
   b. Multiply by $\mathbf{O}(z_t \mid :)$

Check prediction with observation

Forward algorithm

Given observation sequence $\boldsymbol{z}_{0:T}$

1. Multiply initial distribution by $\mathbf{O}(z_0 \mid :)$
2. At each time step:
   a. Multiply current distribution by $\mathbf{P}$
   b. Multiply by $\mathbf{O}(z_t \mid :)$
3. Normalize

Estimation

- **Filtering:**
  - Given a sequence of observations, estimate the final state
- **Smoothing:**
  - Given a sequence of observations, estimate the sequence of states
- **Prediction:**
  - Given a sequence of observations, predict future states

Smoothing

TÉCNICO LISBOA

Estimation

- **Filtering:**
  - Given a sequence of observations, estimate the final state
- **(Easier) Marginal smoothing:**
  - Given a sequence of observations, estimate **some state** in the middle
- **Prediction:**
  - Given a sequence of observations, predict future states

Smoothing

- We are given a sequence of observations $\boldsymbol{z}_{0:T}$
- We want to estimate, for $t < T$

$$\mathbb{P}_{\mu_0}\left[\mathrm{x}_t = x \mid \mathbf{z}_{0:T} = \boldsymbol{z}_{0:T}\right]$$

where μ_0 is the initial distribution, i.e.,

$$\mu_0(x) = \mathbb{P}\left[\mathrm{x}_0 = x\right]$$

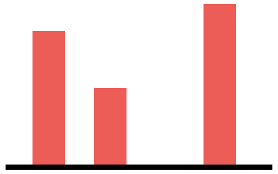
Smoothing

- We use the same notation:

$$\mu_{t|0:T}(x) = \mathbb{P}_{\mu_0}\left[\mathrm{x}_t = x \mid \mathbf{z}_{0:T} = \boldsymbol{z}_{0:T}\right]$$

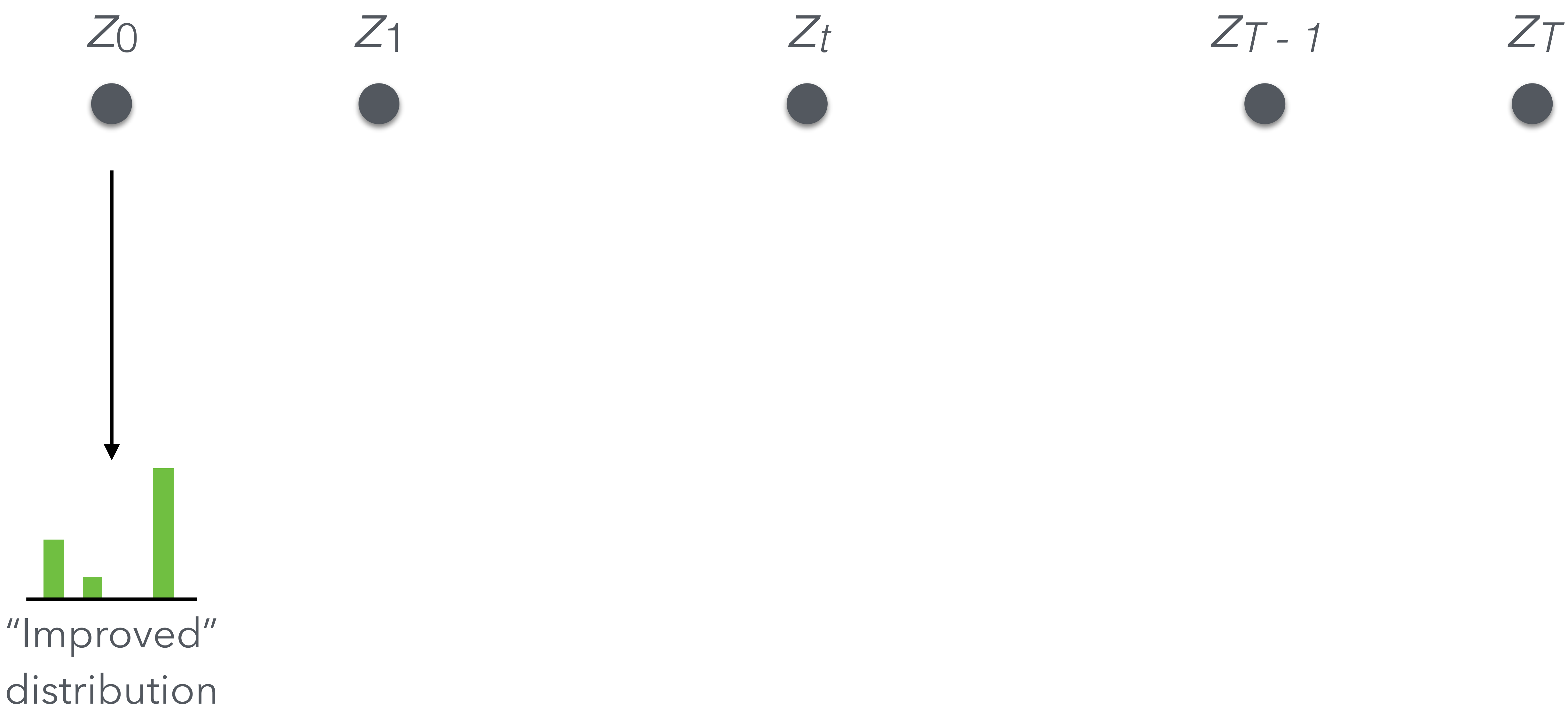
Similar idea

z_0 z_1 z_t z_{T-1} z_T



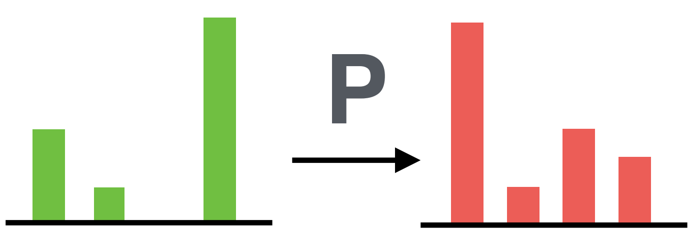
Initial distribution

Similar idea



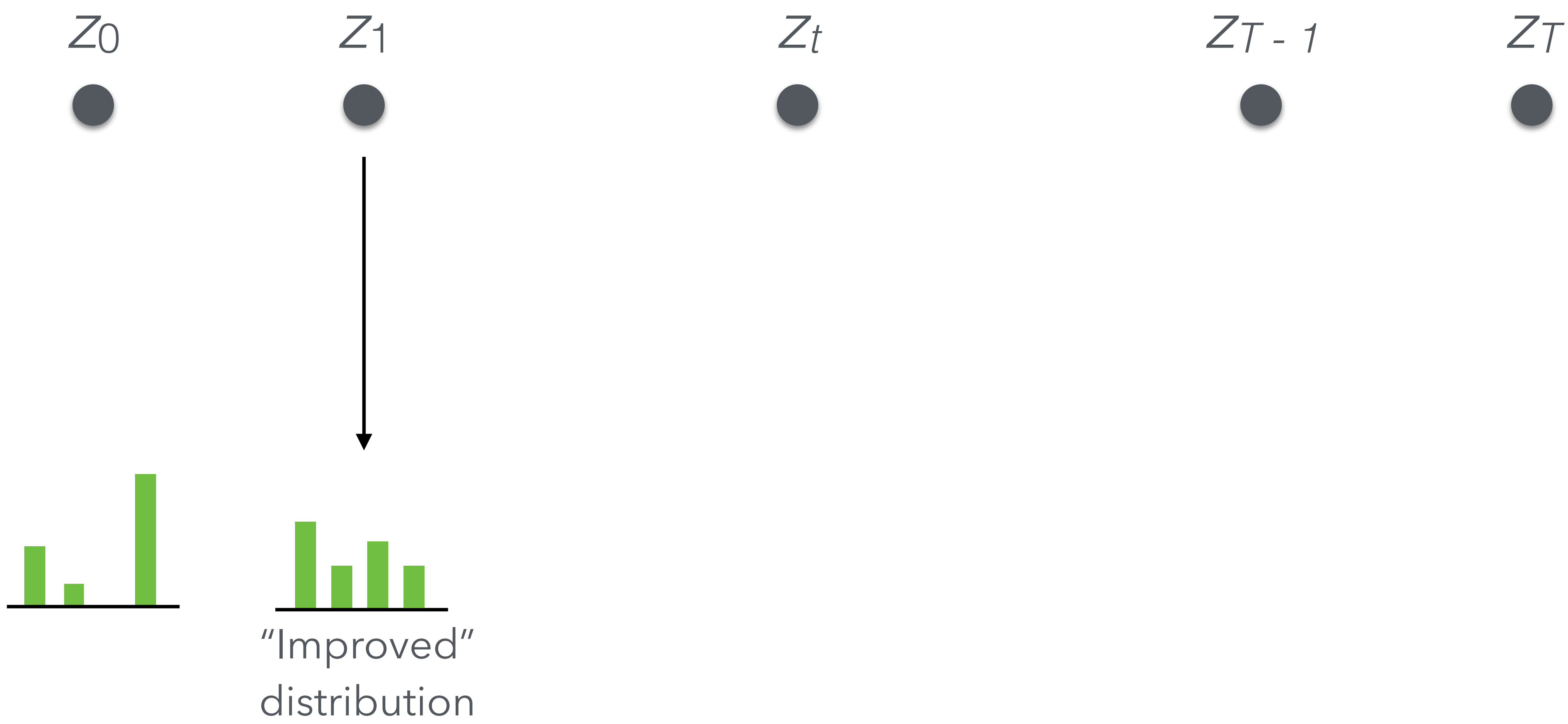
Similar idea

z_0 z_1 z_t z_{T-1} z_T

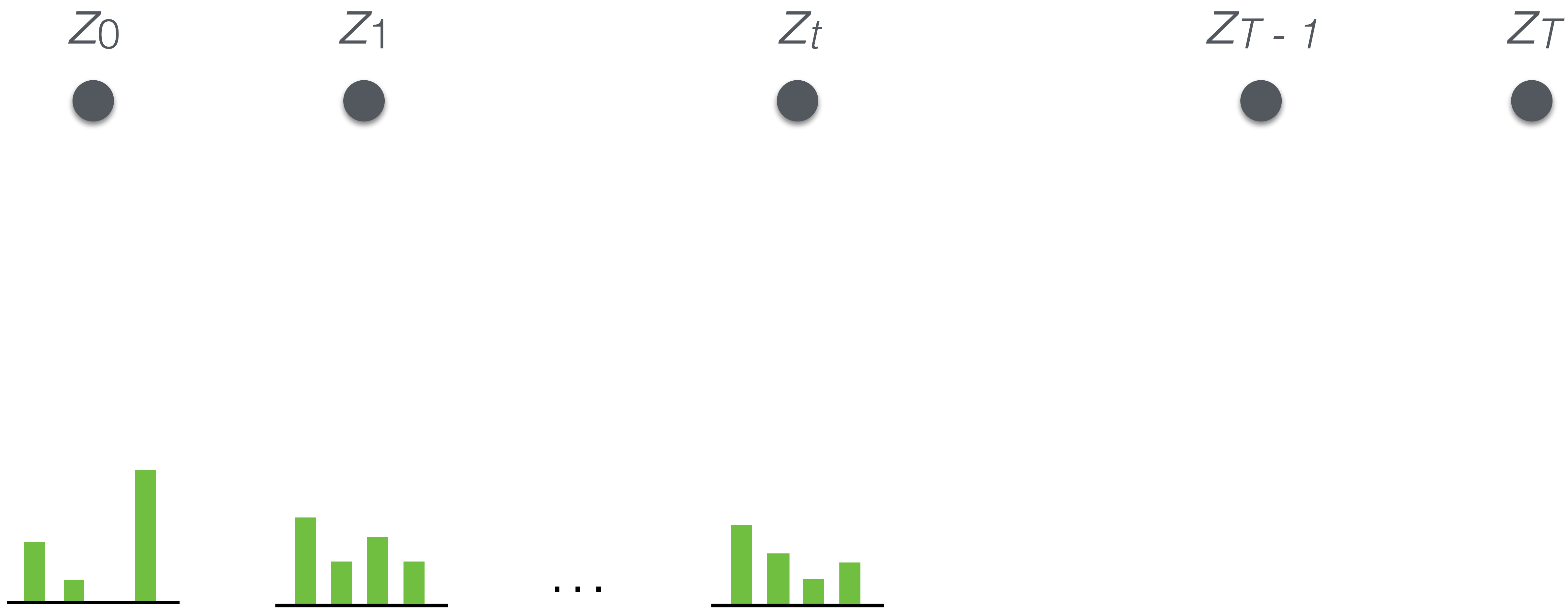


Predicted distribution

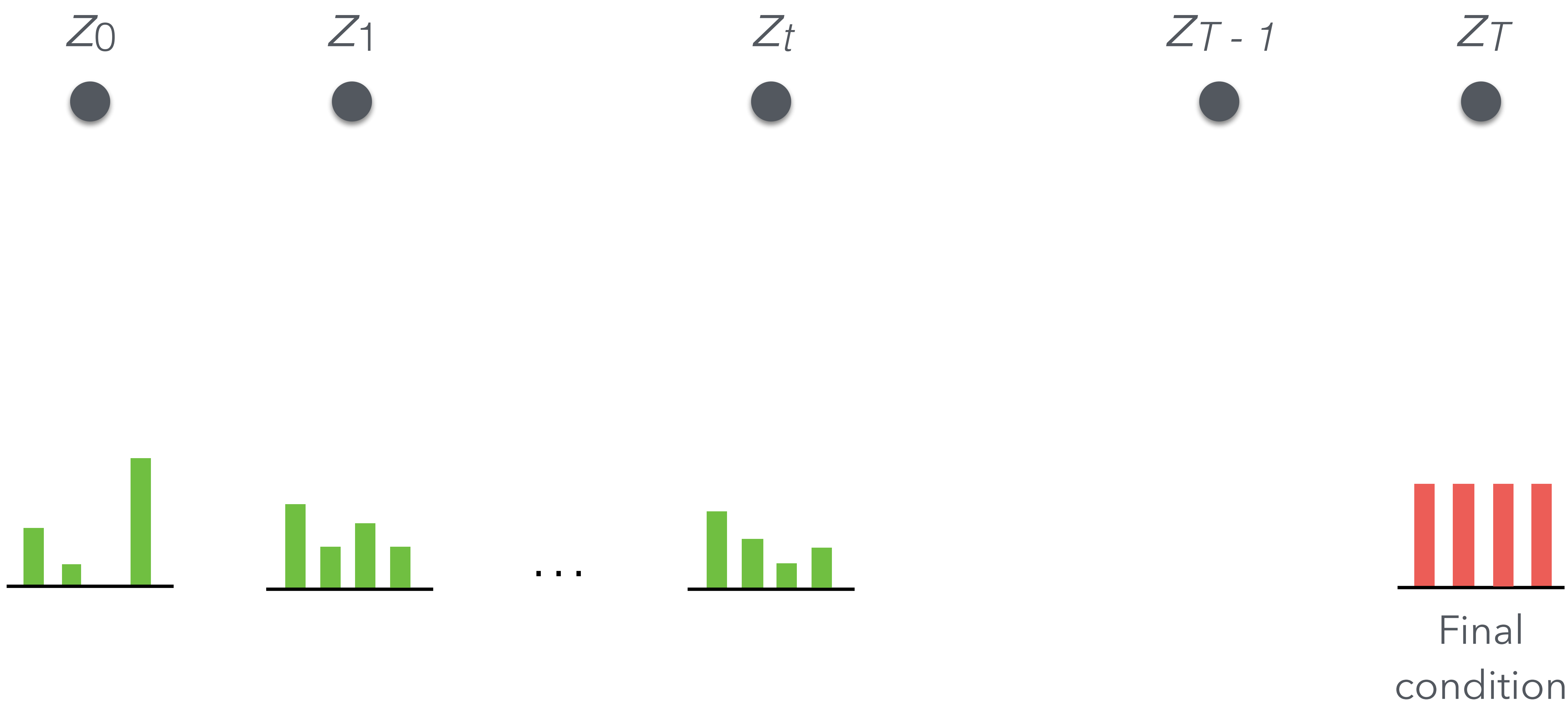
Similar idea



Similar idea

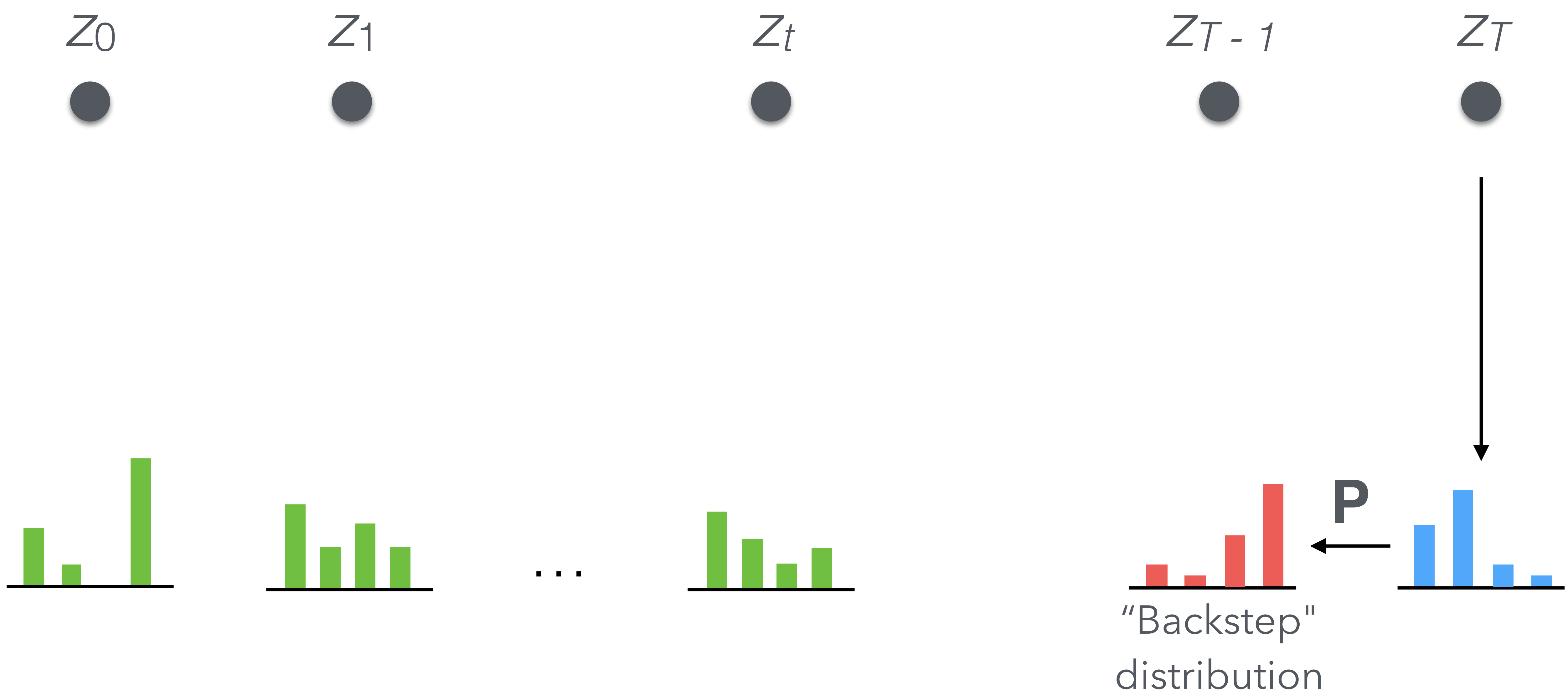


... with a twist

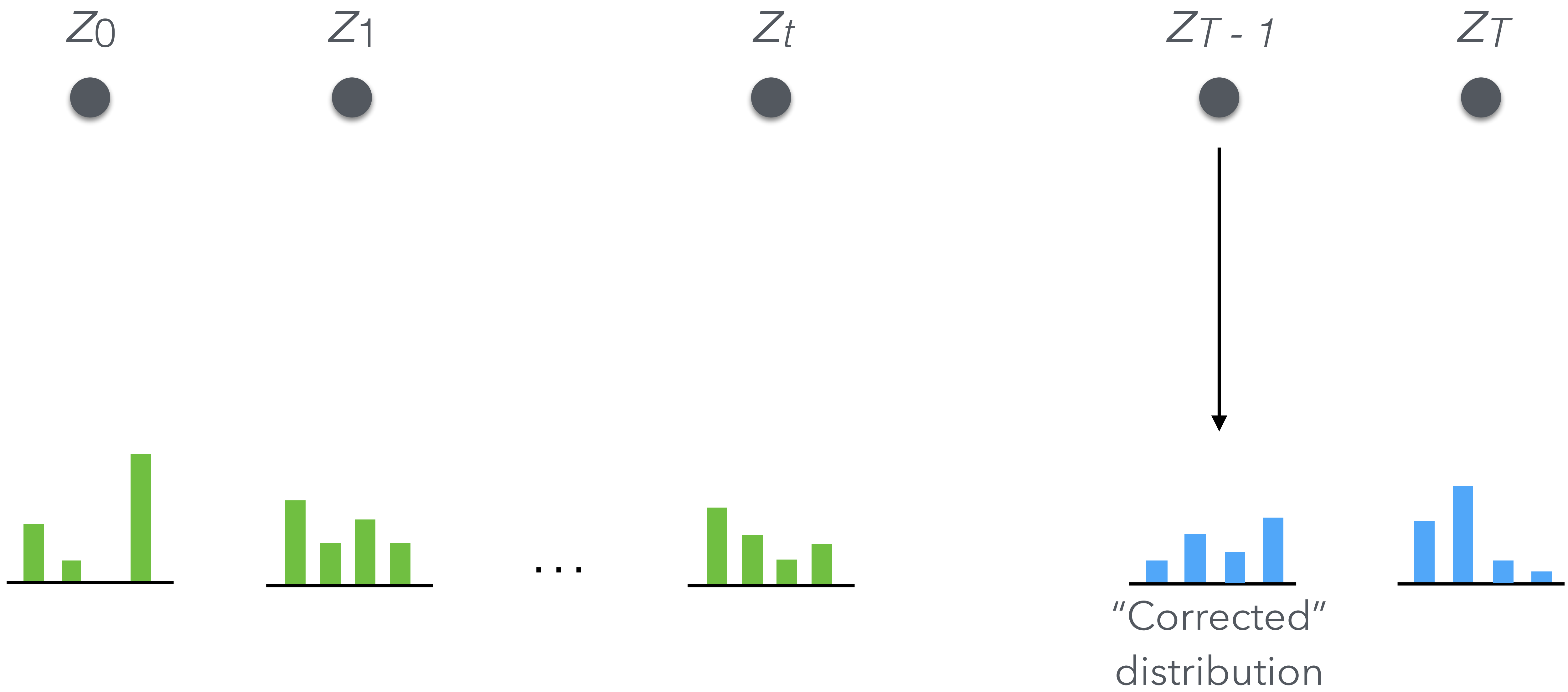


… with a twist

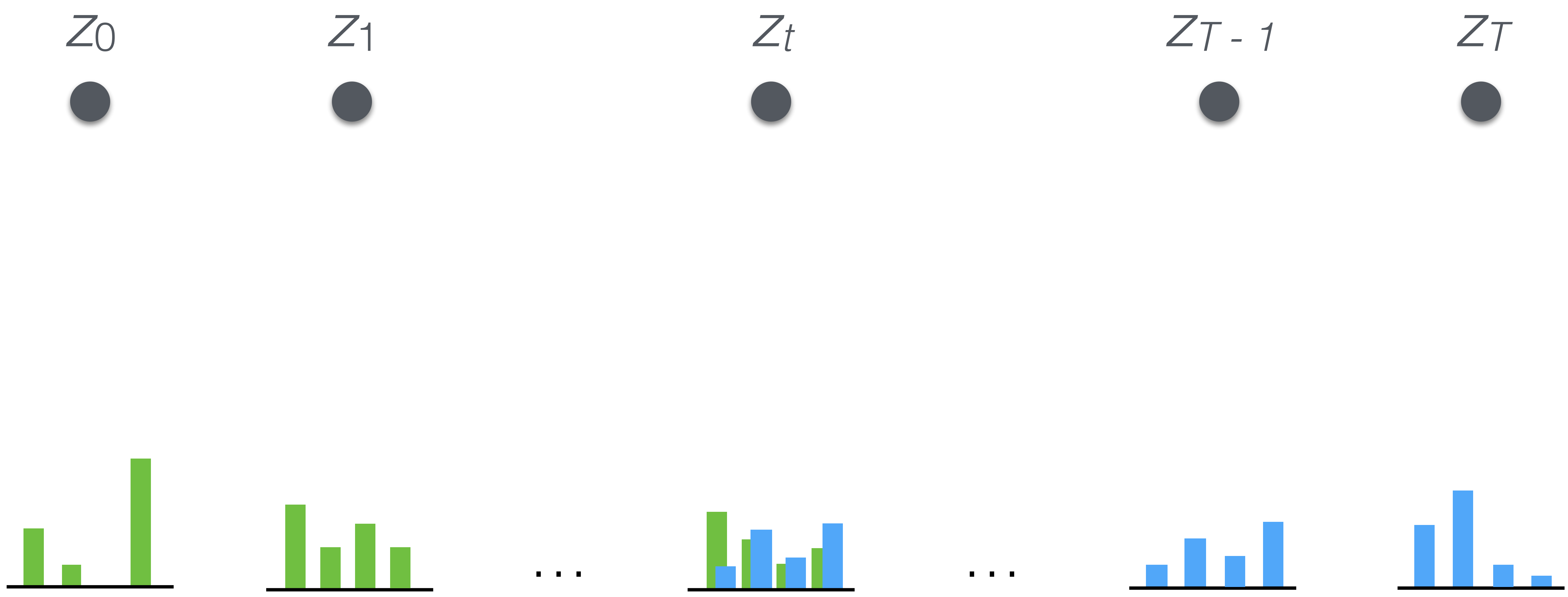
... with a twist



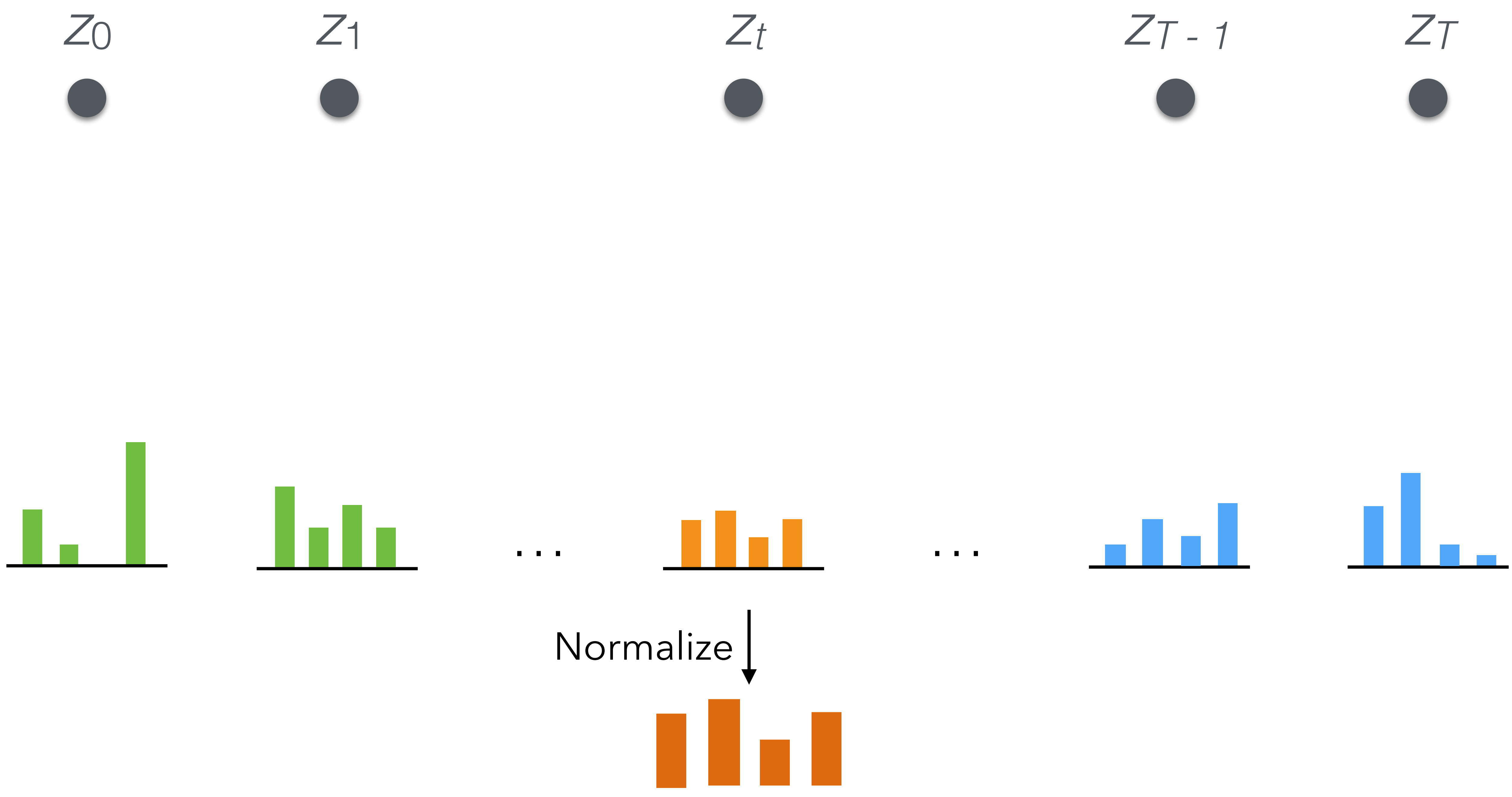
… with a twist



… with a twist



... with a twist



Backward mapping

Backward mapping

Given a sequence of observations $\mathbf{z}_{0:t}$, the backward mapping $\beta_t : \mathscr{X} \longmapsto \mathbb{R}$ is defined for each t as

$$\beta_t(x) = \mathbb{P}_{\mu_0}\left[\mathbf{z}_{t+1:T} = \boldsymbol{z}_{t+1:T} \mid \mathrm{x}_t = x\right]$$

How the present relates
to the future

So what?

- Backward mapping has several useful properties

1. We can compute $\mu_{t\,|\,0:T}$ from α_t and β_t:

$$\mu_{t|0:T}(x) = \frac{\beta_t(x)\alpha_t(x)}{\sum_{y\in\mathcal{X}} \beta_t(y)\alpha_t(y)}$$

So what?

- Backward mapping has several useful properties

  1. We can compute $\mu_{t \mid 0:T}$ from α_t and β_t

  2. The backward mapping can be computed recursively:

$$\beta_t(x) = \sum_{y \in \mathcal{X}} \mathbf{O}(z_{t+1} \mid y)\beta_{t+1}(y)\mathbf{P}(y \mid x)$$

Forward-backward algorithm

Require: Observation sequence $z_{0:T}$

1. Initialize $\boldsymbol{\alpha}_0 \leftarrow \text{diag}(\mathbf{O}_{:,z_0})\boldsymbol{\mu}_0^\top$, $\boldsymbol{\beta}_T \leftarrow \mathbf{1}$
2. **for** τ = 0, …, *t* **do**
3. $\quad \boldsymbol{\alpha}_{\tau+1} \leftarrow \text{diag}(\mathbf{O}_{:,z_\tau})\mathbf{P}^\top \boldsymbol{\alpha}_\tau$ ← Forward update
4. **end for**
5. **for** τ = T - 1, …, *t* **do**
6. $\quad \boldsymbol{\beta}_\tau \leftarrow \mathbf{P}\,\text{diag}(\mathbf{O}_{:,z_{\tau+1}})\boldsymbol{\beta}_{\tau+1}$ ← Backward update
7. **end for**
8. **return** $\boldsymbol{\alpha}_t \otimes \boldsymbol{\beta}_t / (\boldsymbol{\alpha}_t^\top \boldsymbol{\beta}_t)$ ← Combine & normalize

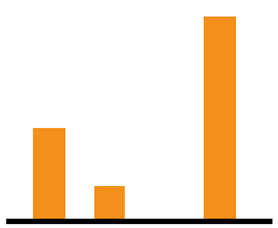
Estimation

- **Filering:**
  - Given a sequence of observations, estimate the final state
- **(Joint) Smoothing:**
  - Given a sequence of observations, estimate the **whole** sequence of states (most likely sequence)
- **Prediction:**
  - Given a sequence of observations, predict future states

Any ideas?

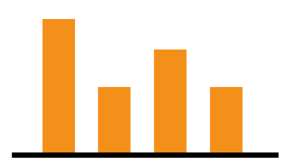
Naive approach

Use FB to compute $\mu_{0\,|\,0:T}$



$t = 0$

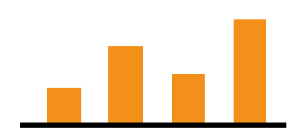
Use FB to compute $\mu_{1\,|\,0:T}$



$t = 1$

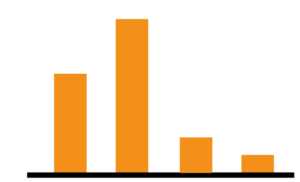
…

Use FB to compute $\mu_{T-1\,|\,0:T}$



$t = T - 1$

Use FB to compute $\mu_{T\,|\,0:T}$



$t = T$

Naive approach

Compute most likely state x_0

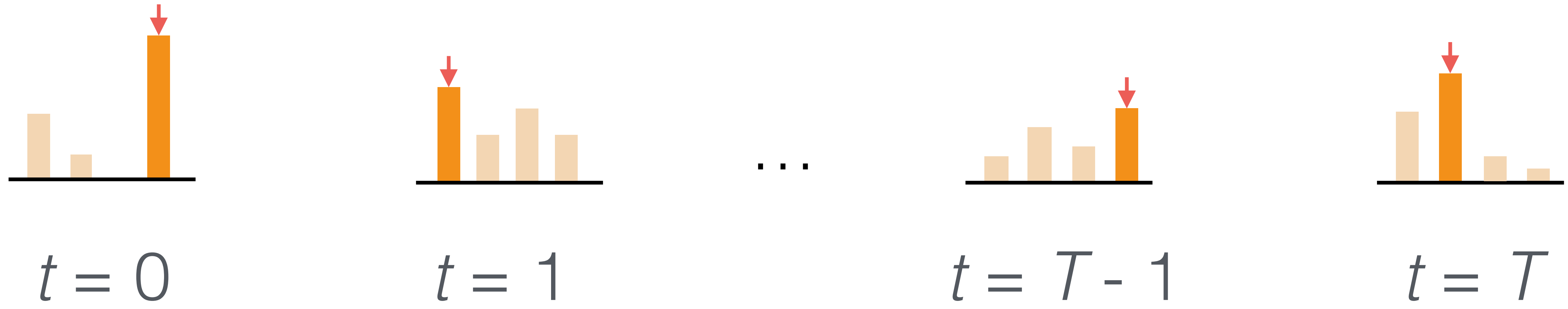
Compute most likely state x_1

Compute most likely state $x_{T\text{-}1}$

Compute most likely state x_T

$t = 0$ $\quad$ $t = 1$ $\quad$ … $\quad$ $t = T - 1$ $\quad$ $t = T$

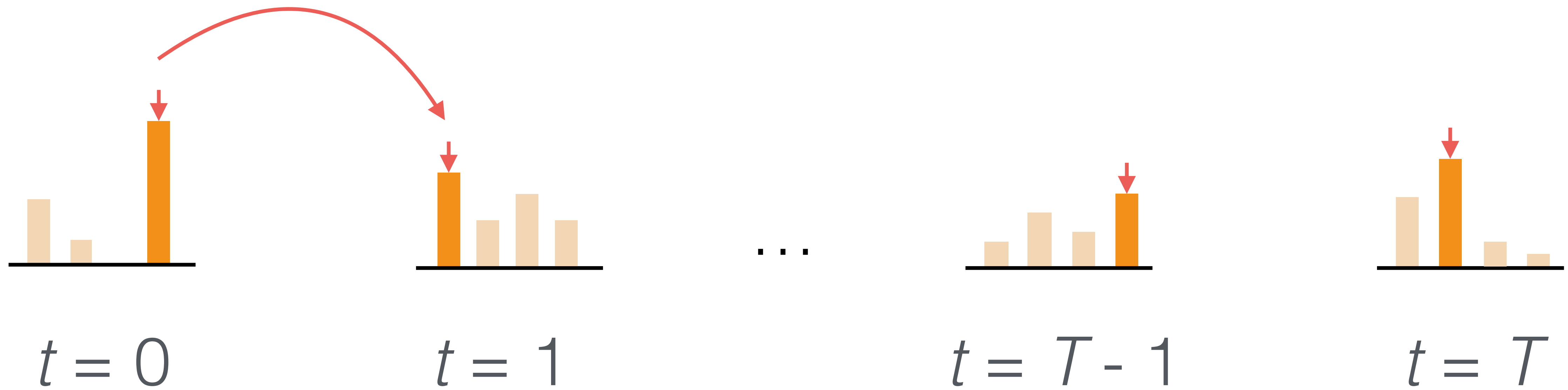
Naive approach



Problem?

Inconsistency…

These transitions may be impossible



TÉCNICO LISBOA

Smoothing

- We are given a sequence of observations $\boldsymbol{z}_{0:T}$

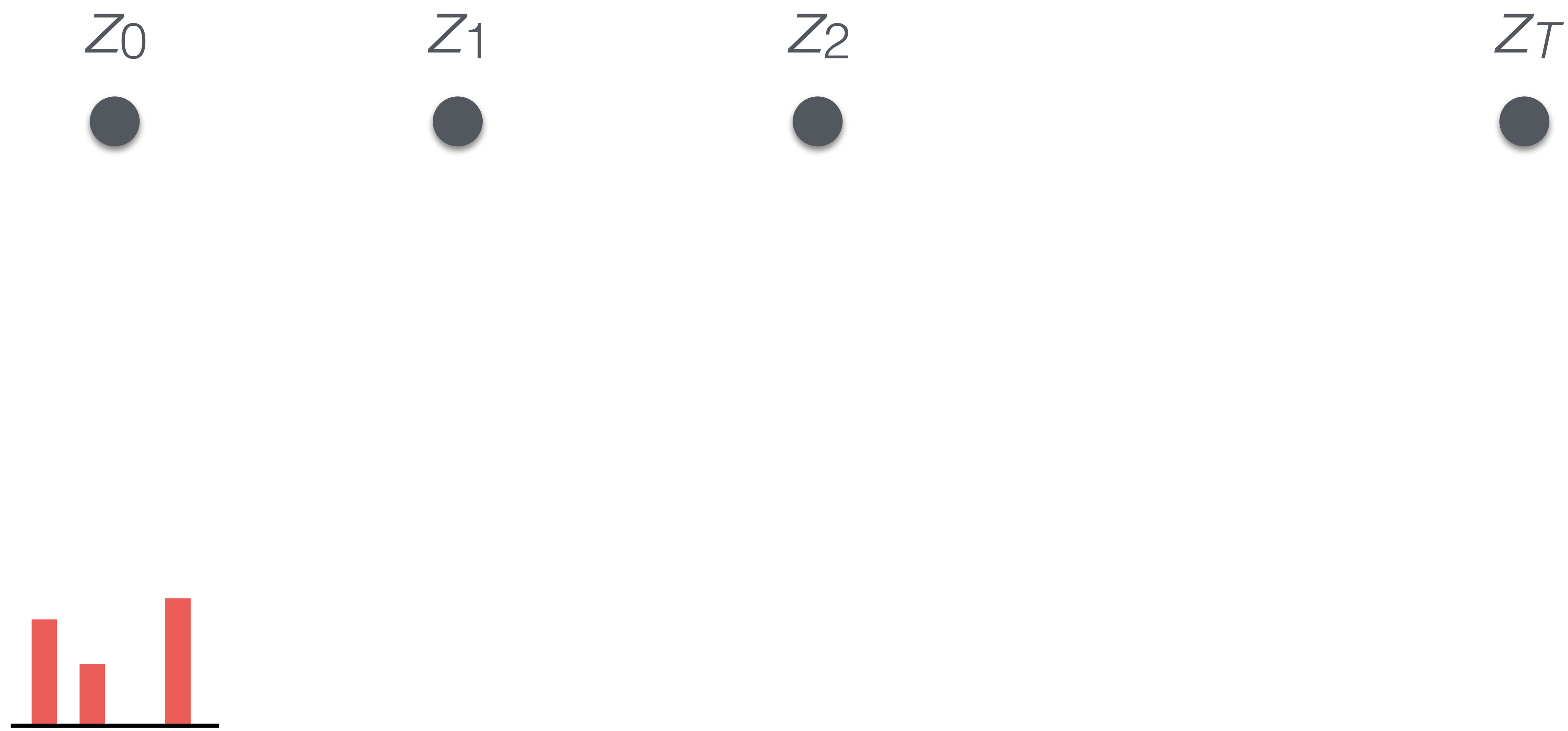
- We want to estimate the **most likely sequence**, i.e.,

$$\boldsymbol{x}^*_{0:T} = \operatorname*{argmax}_{\boldsymbol{x}_{0:T}} \mathbb{P}_{\mu_0}\left[\mathbf{x}_{0:T} = \boldsymbol{x}_{0:T} \mid \mathbf{z}_{0:T} = \boldsymbol{z}_{0:T}\right]$$

where μ_0 is the initial distribution, i.e.,

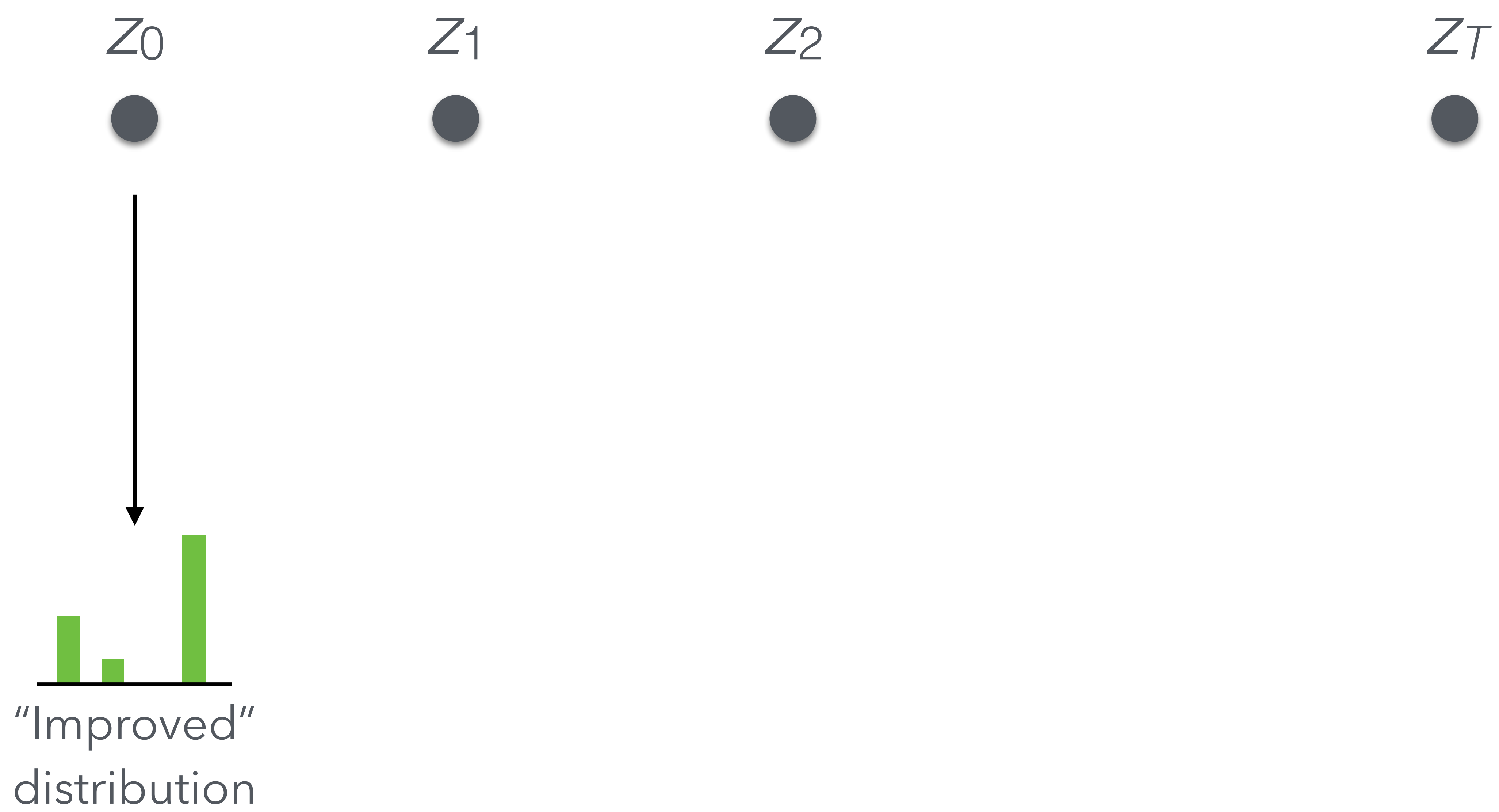
$$\mu_0(x) = \mathbb{P}\left[\mathrm{x}_0 = x\right]$$

Idea

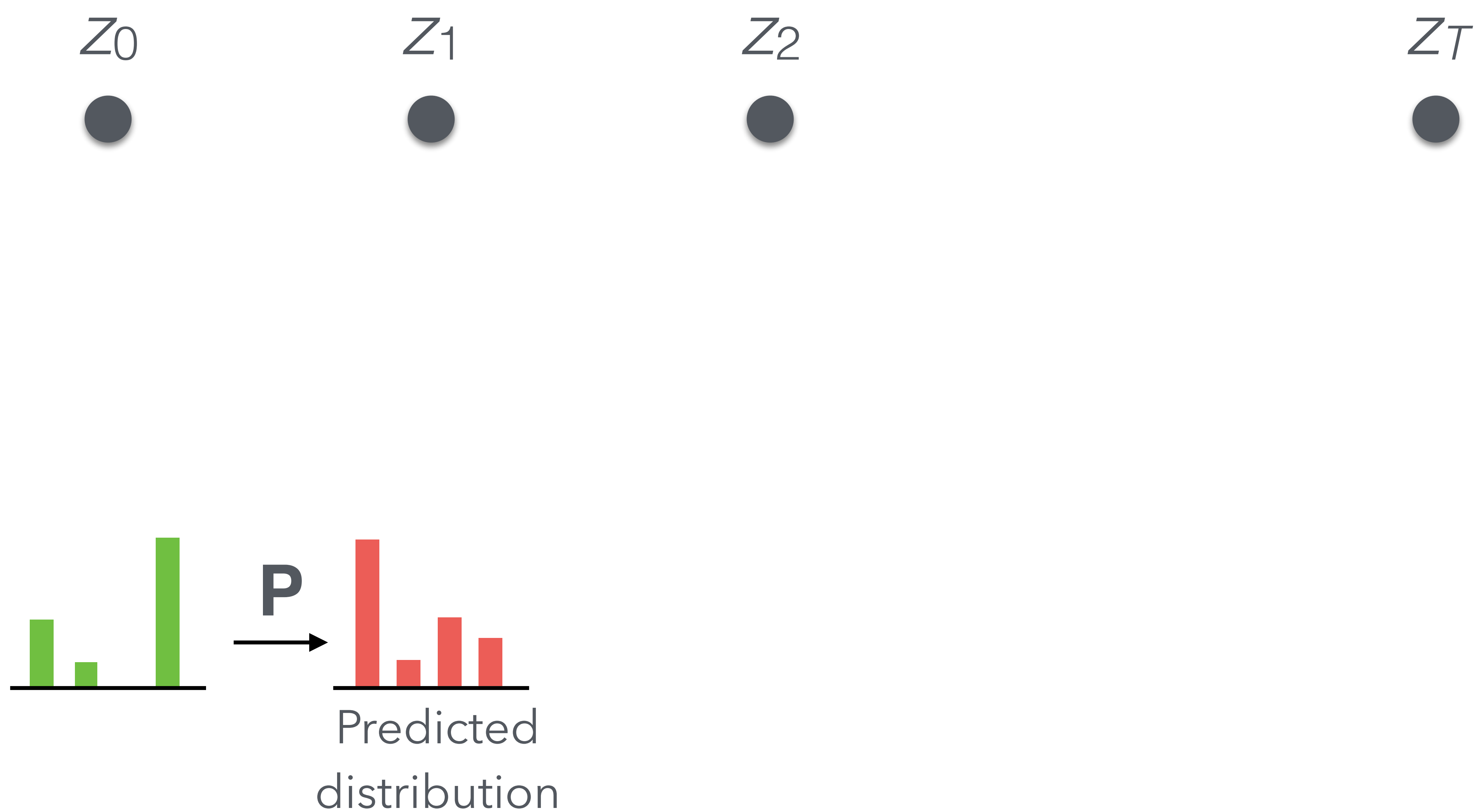


Initial distribution

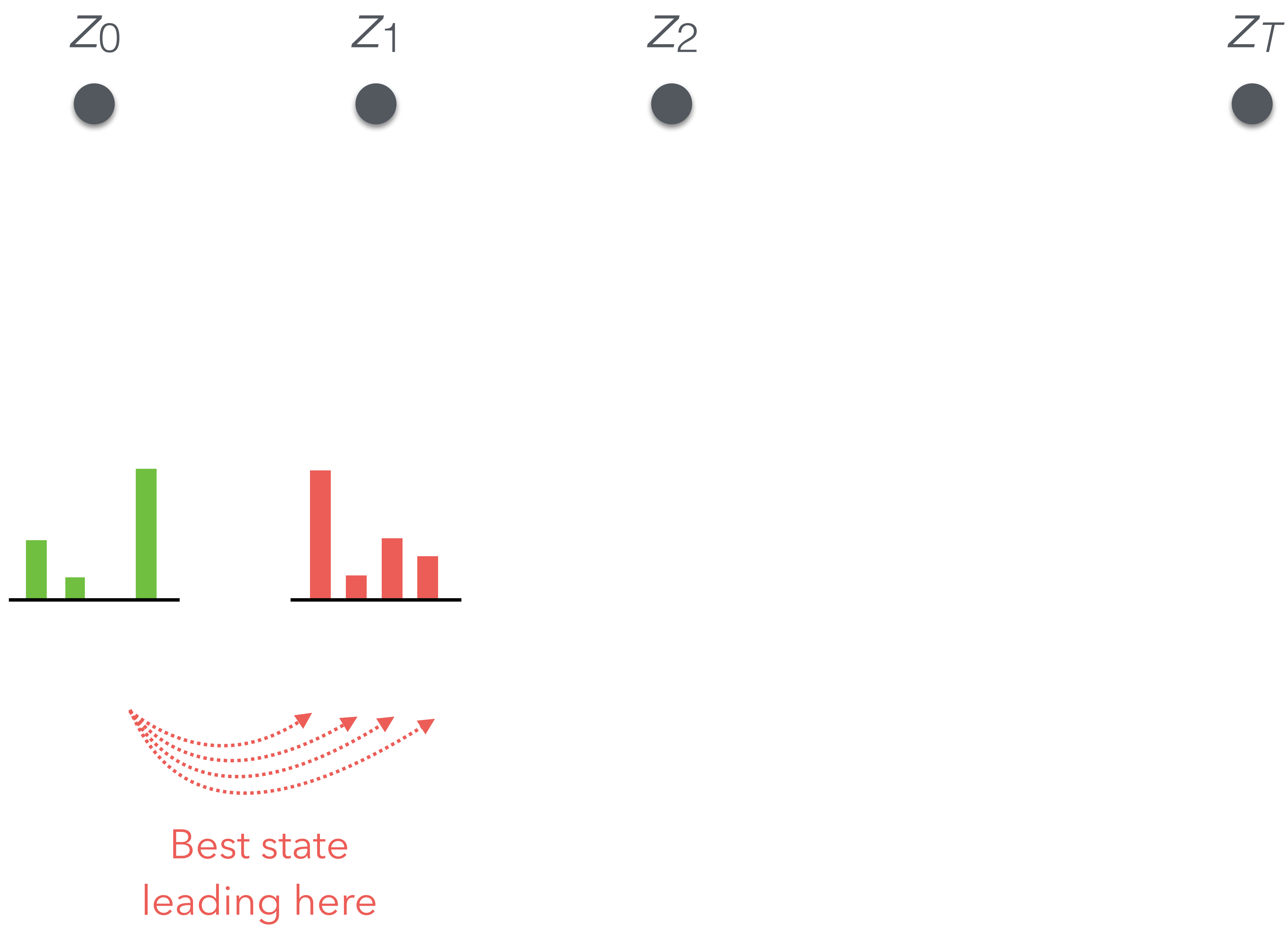
Idea



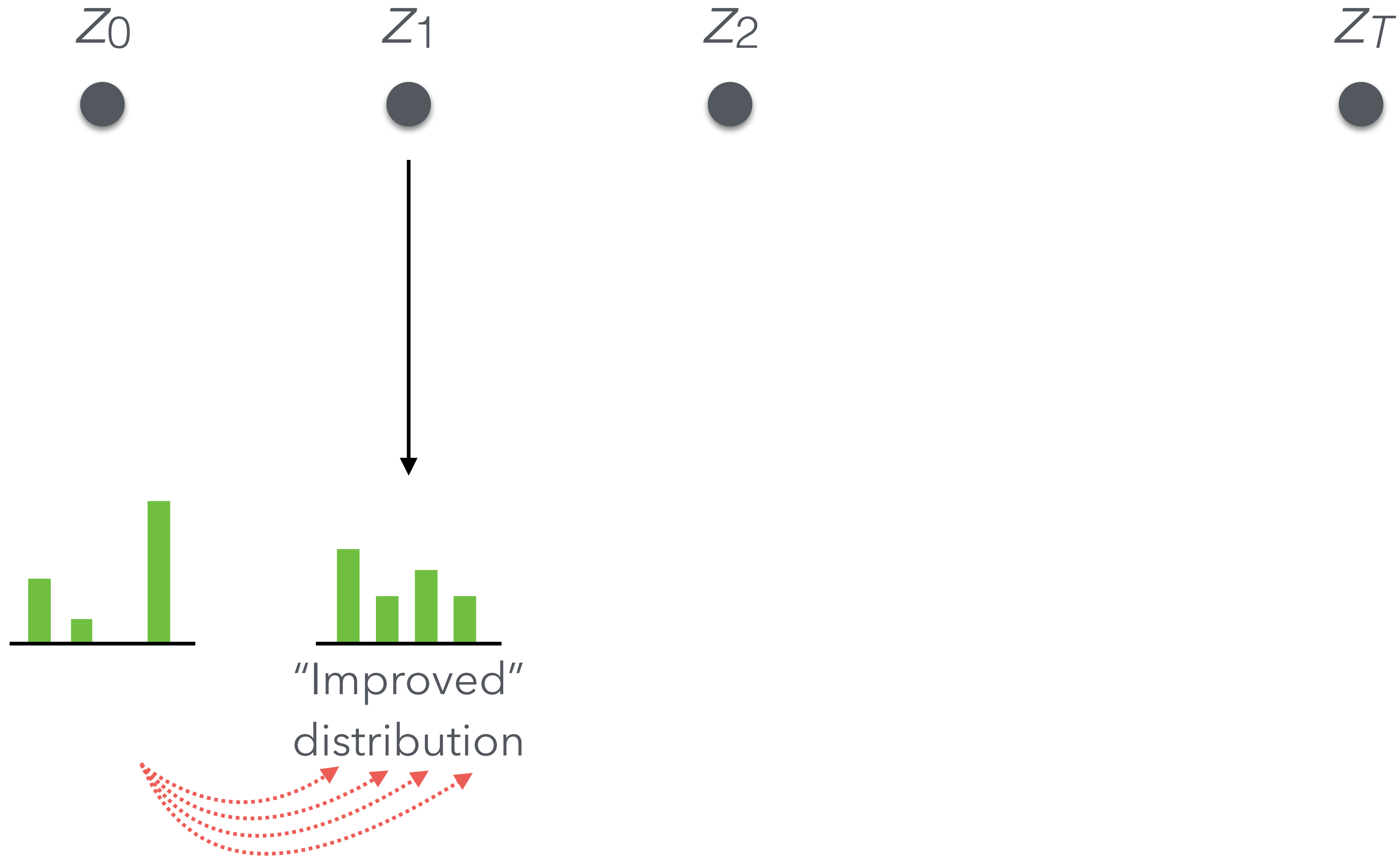
Idea



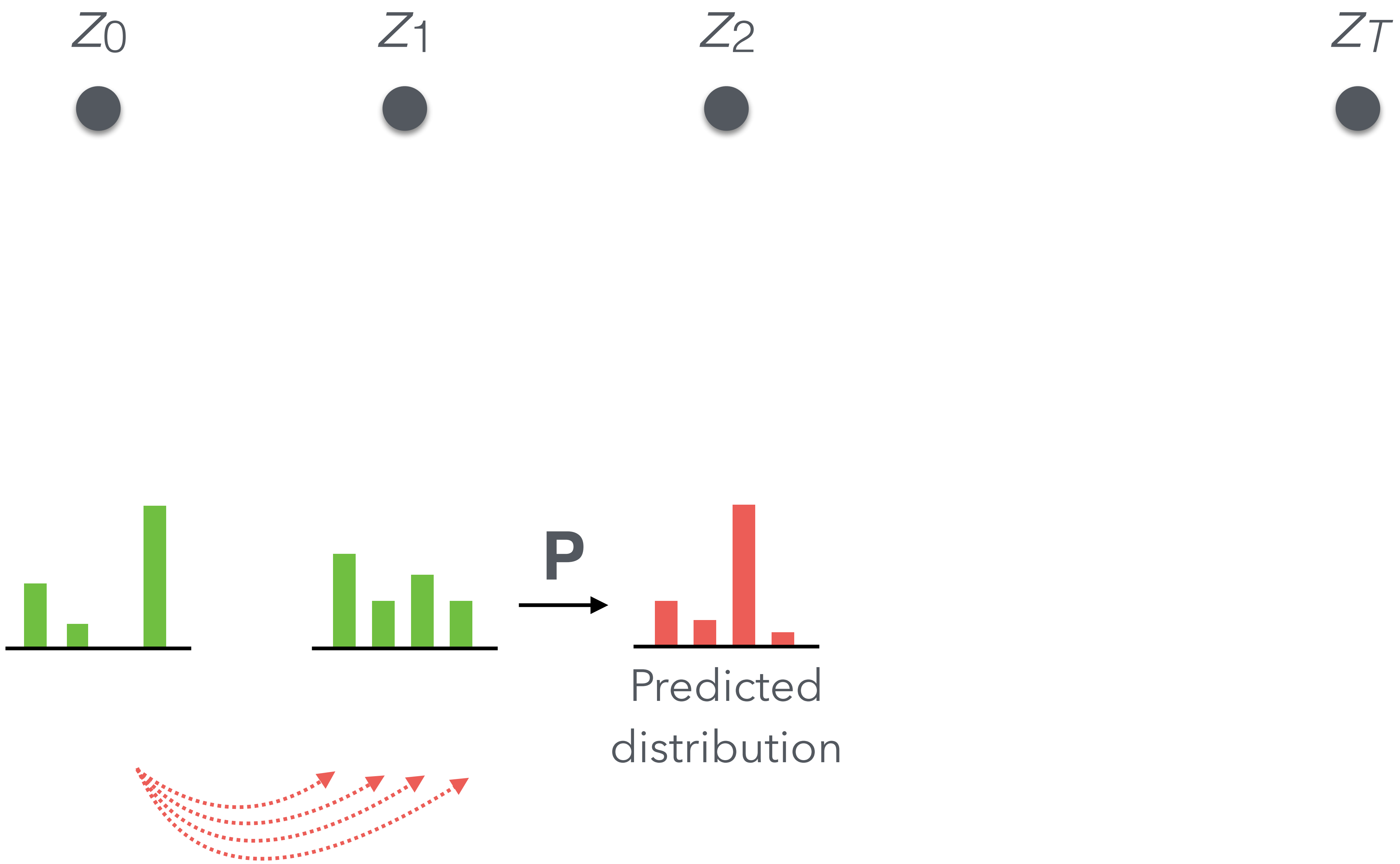
Idea



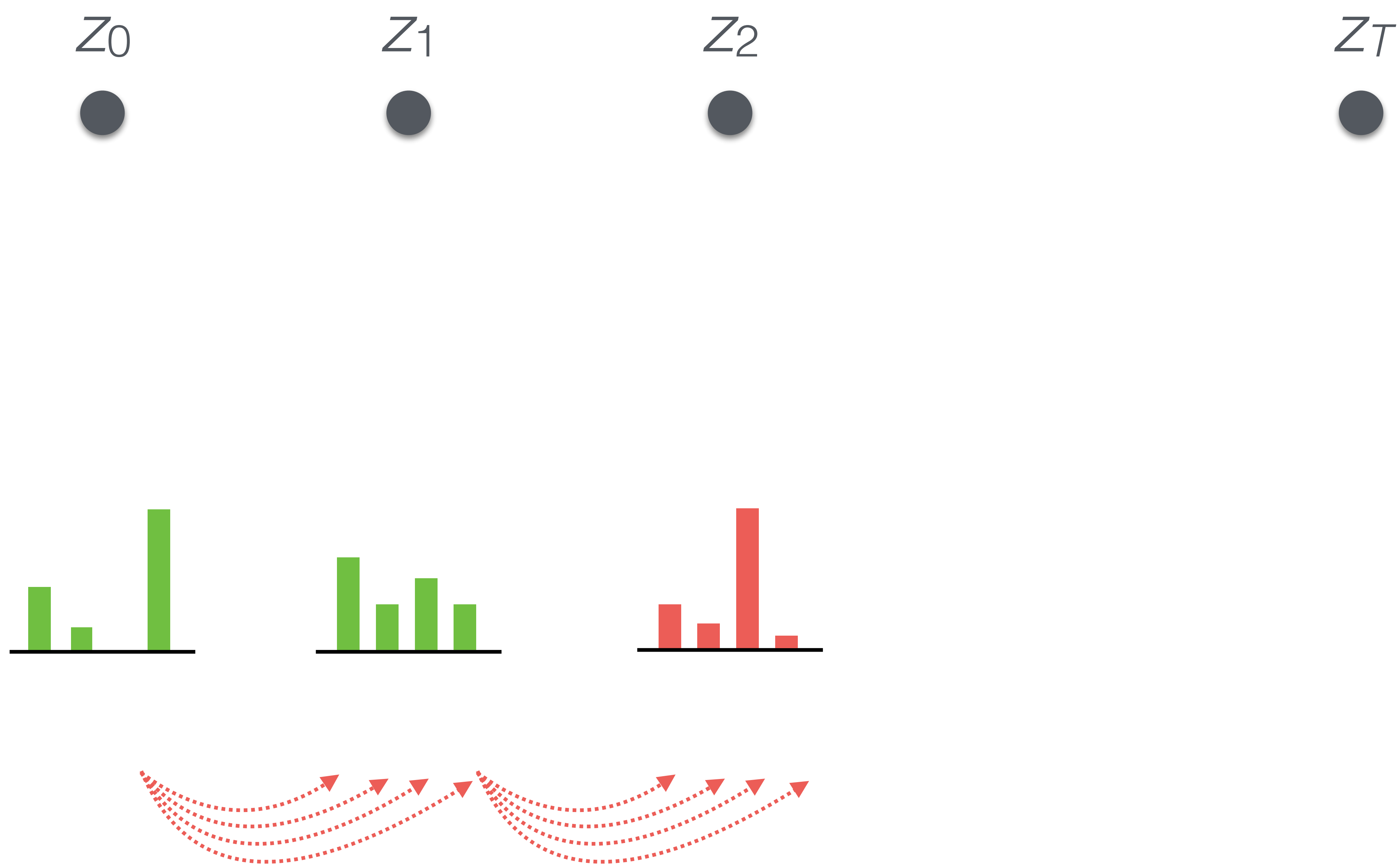
Idea



Idea

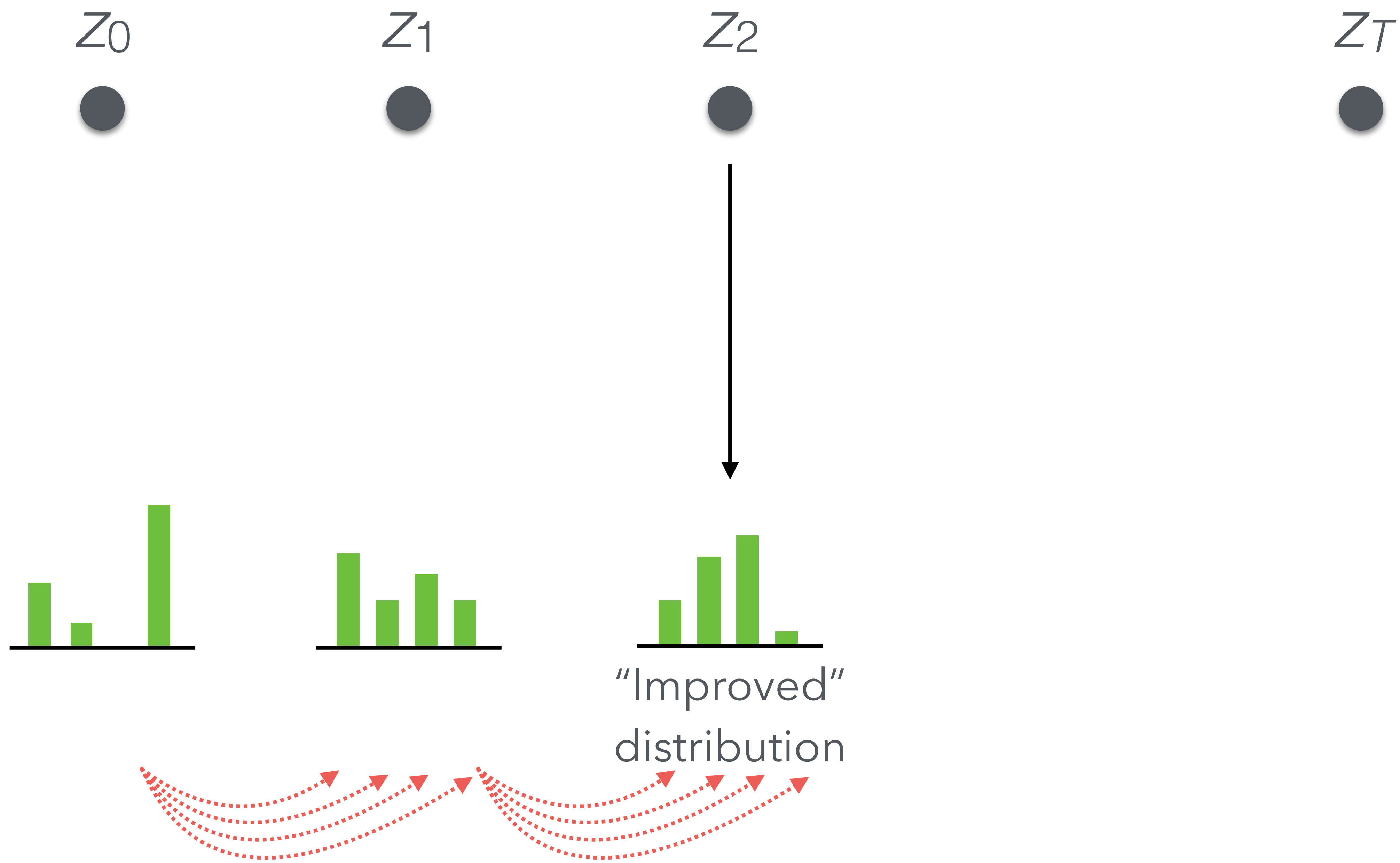


Idea

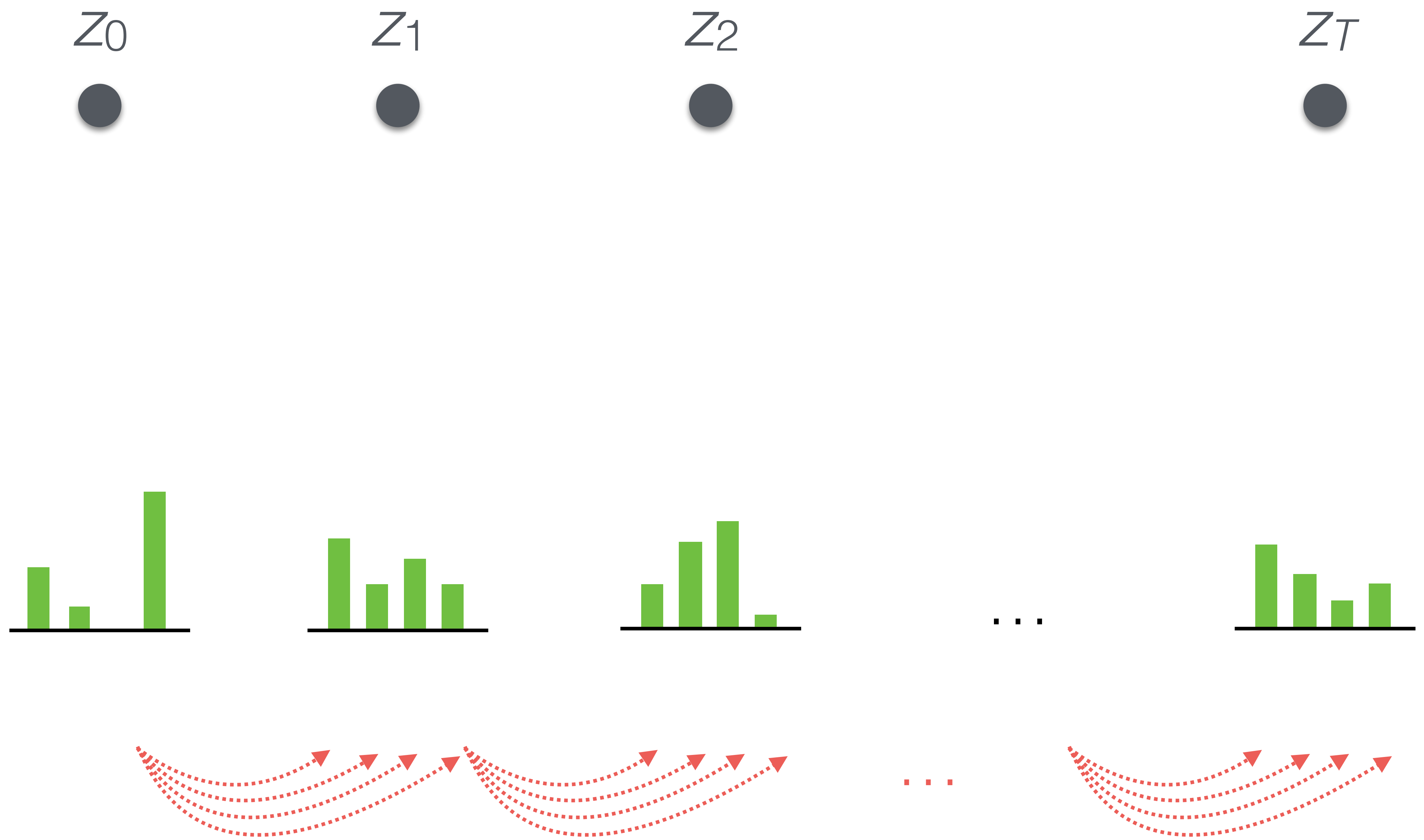


Best state
leading here

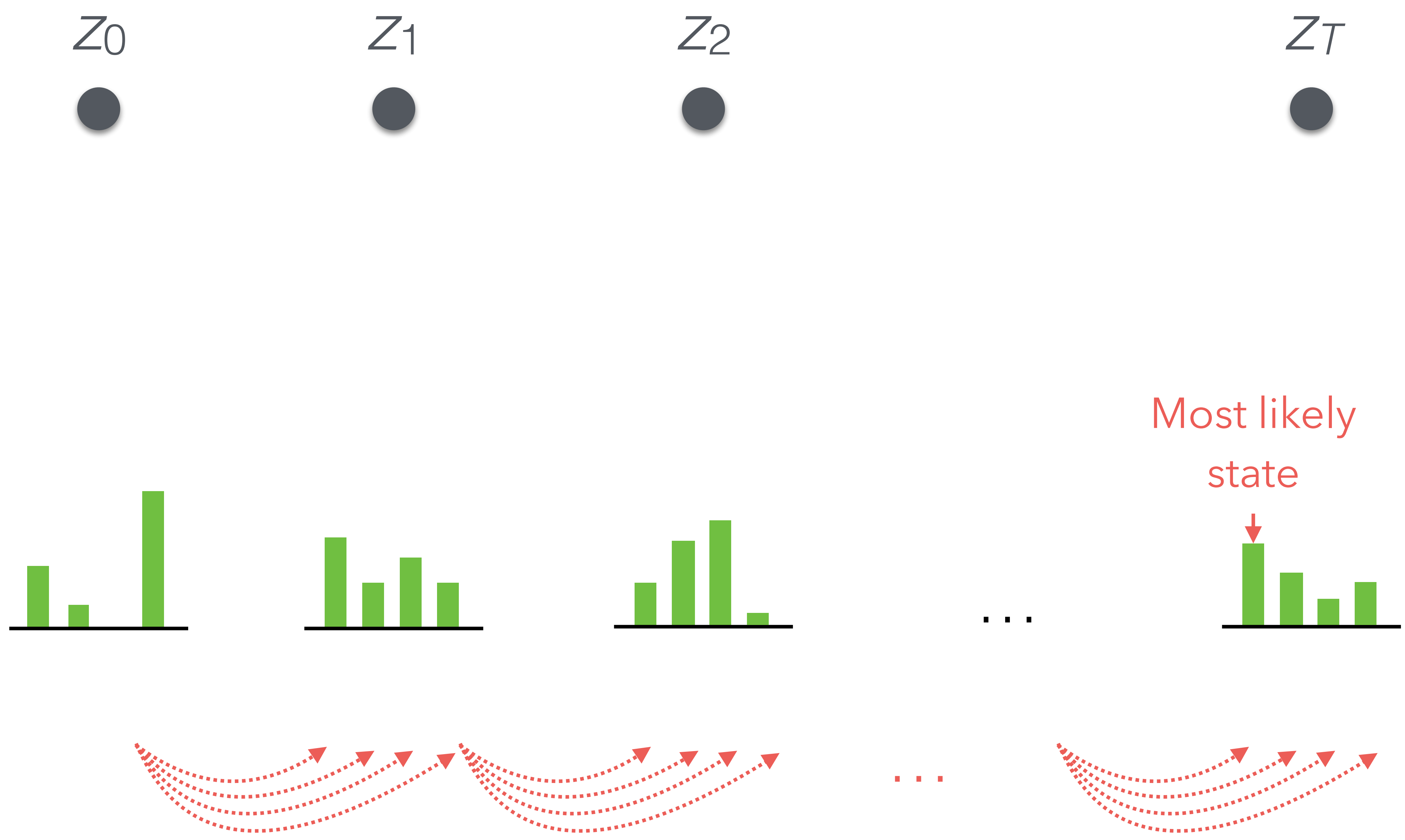
Idea



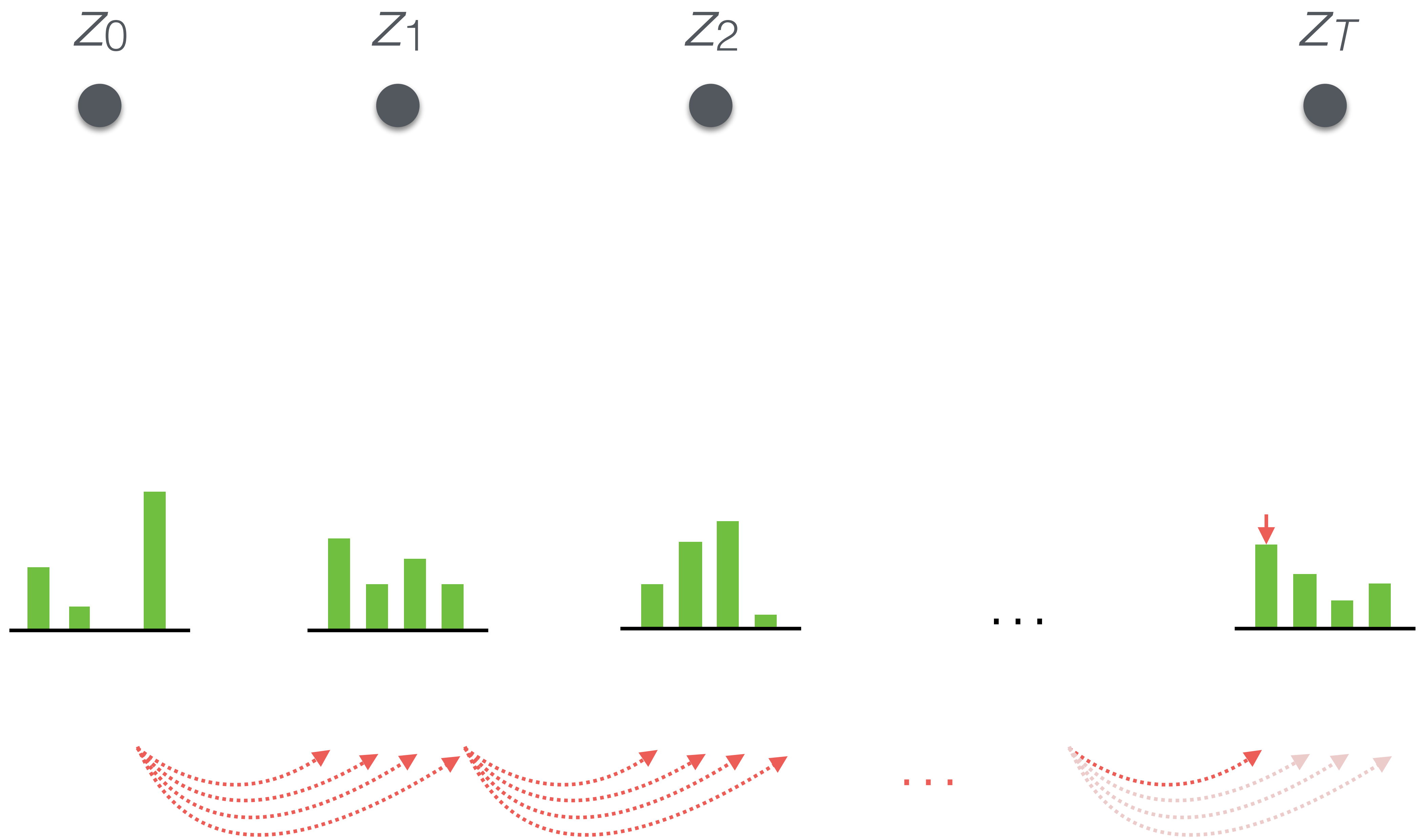
Idea



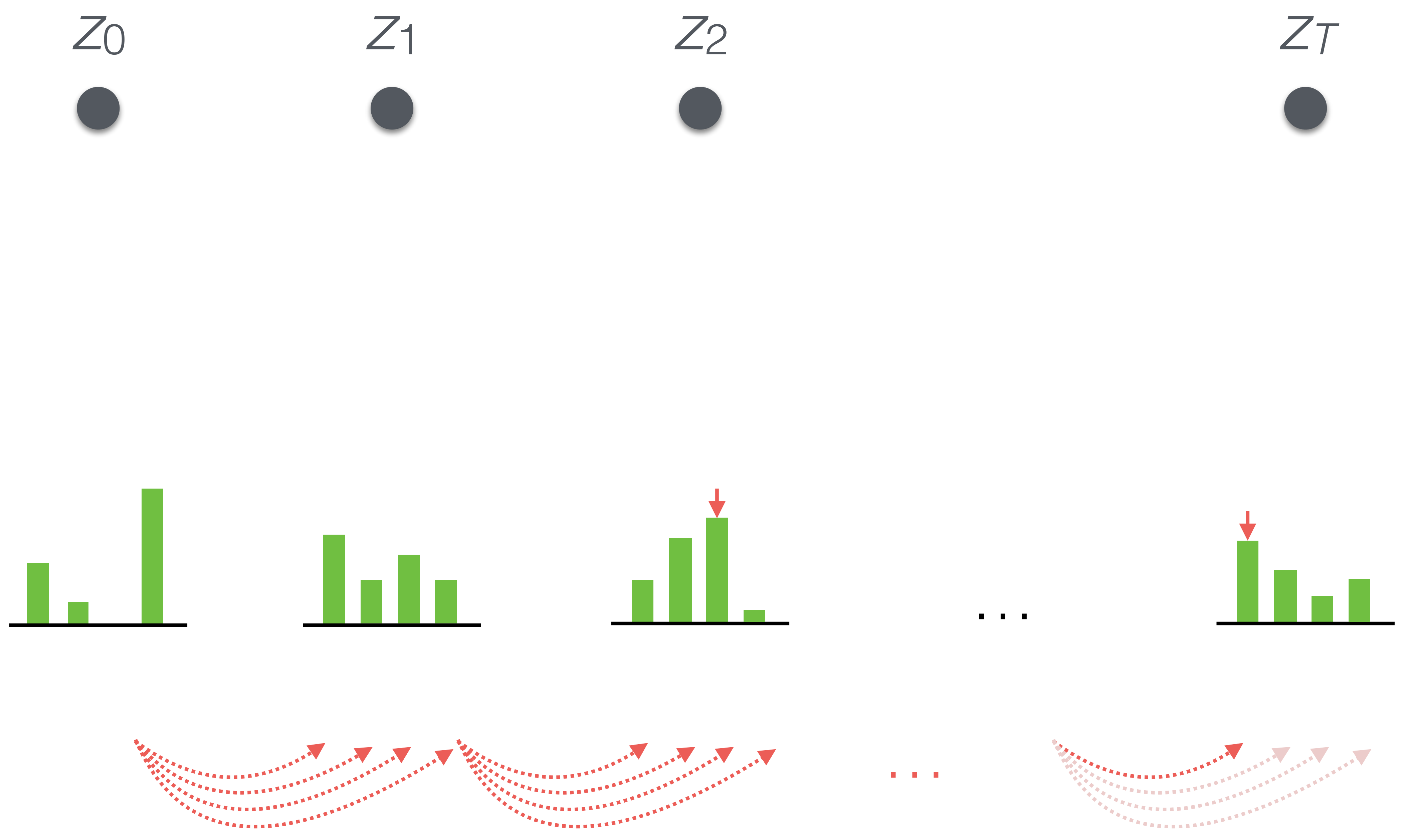
Idea



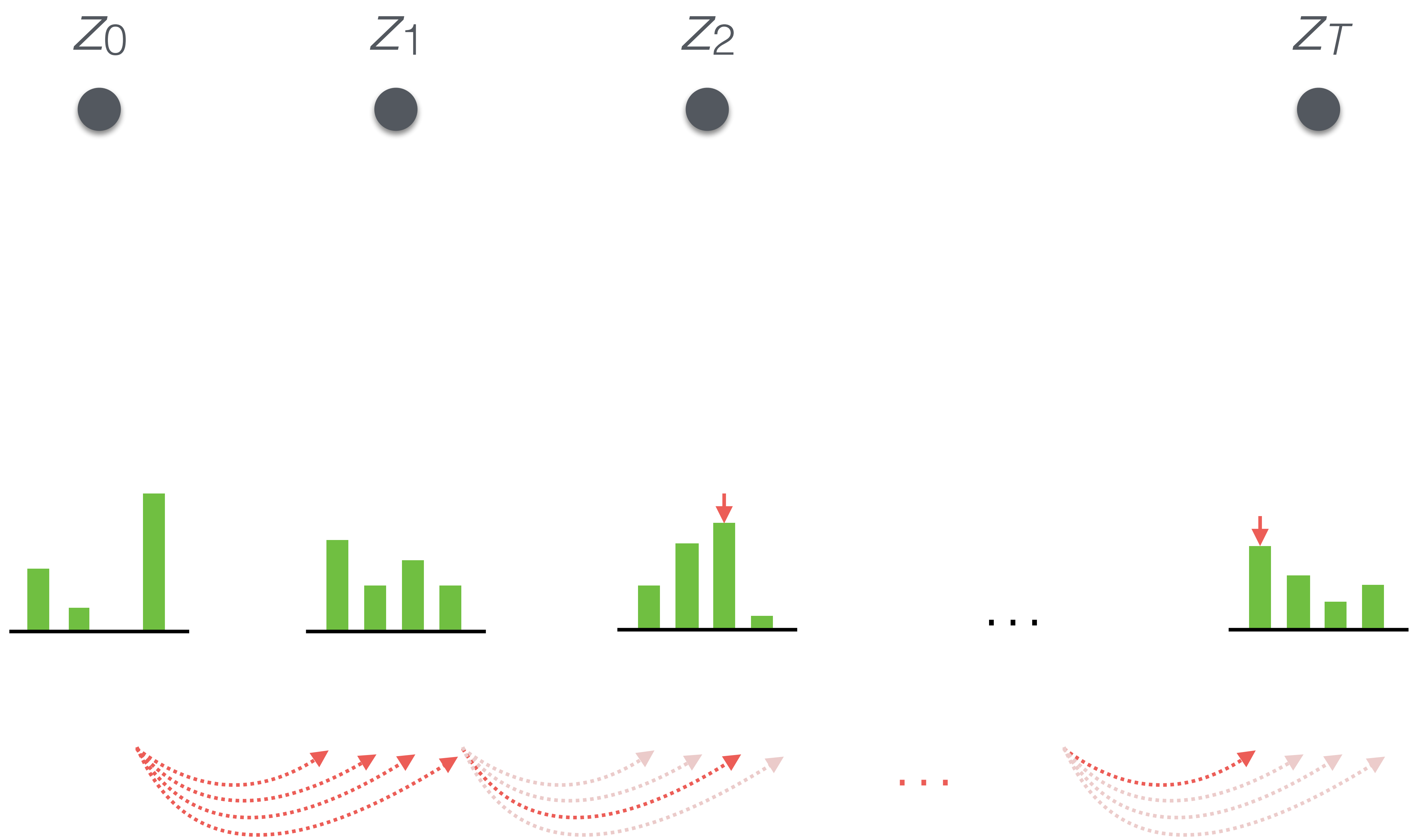
Idea



Idea

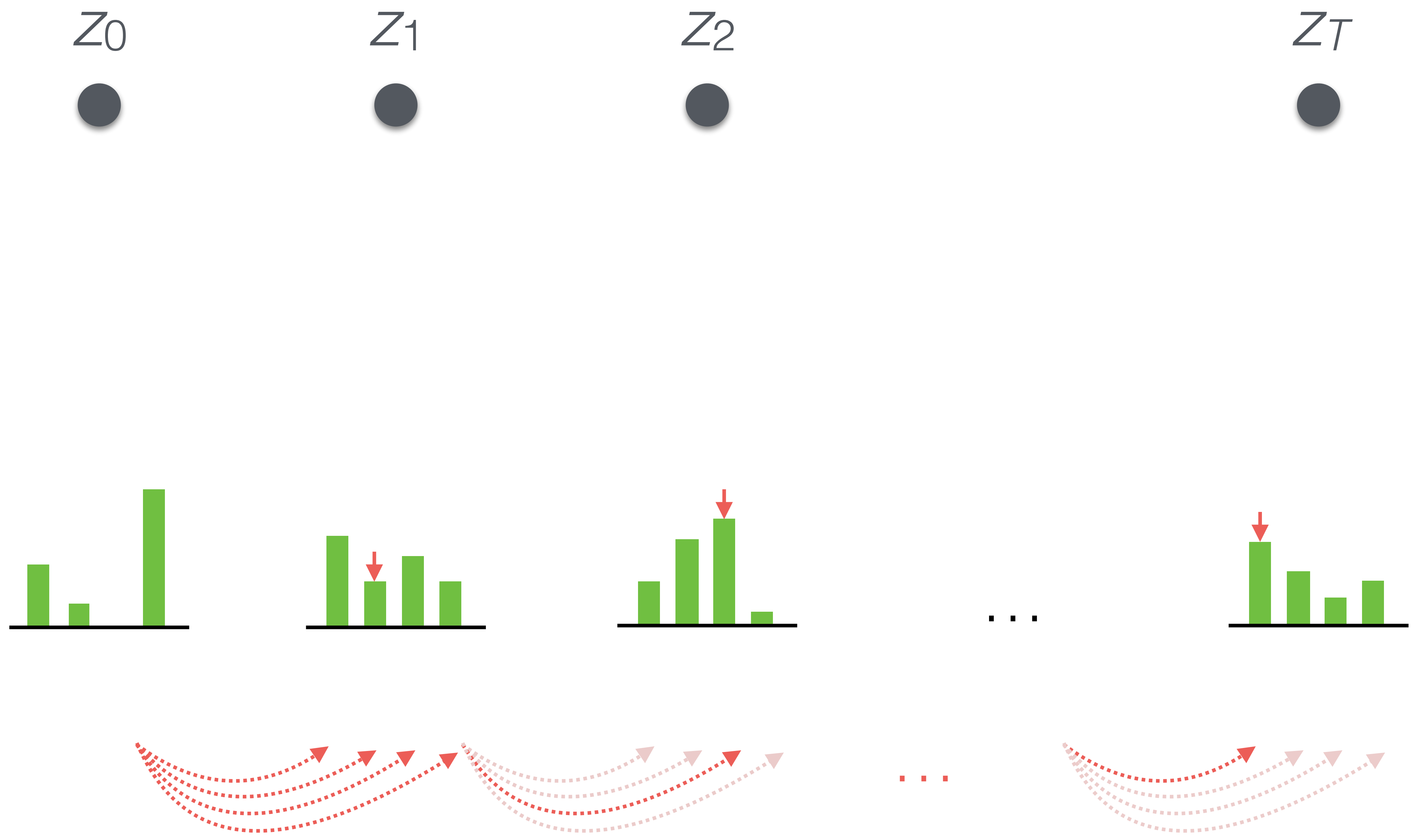


Idea

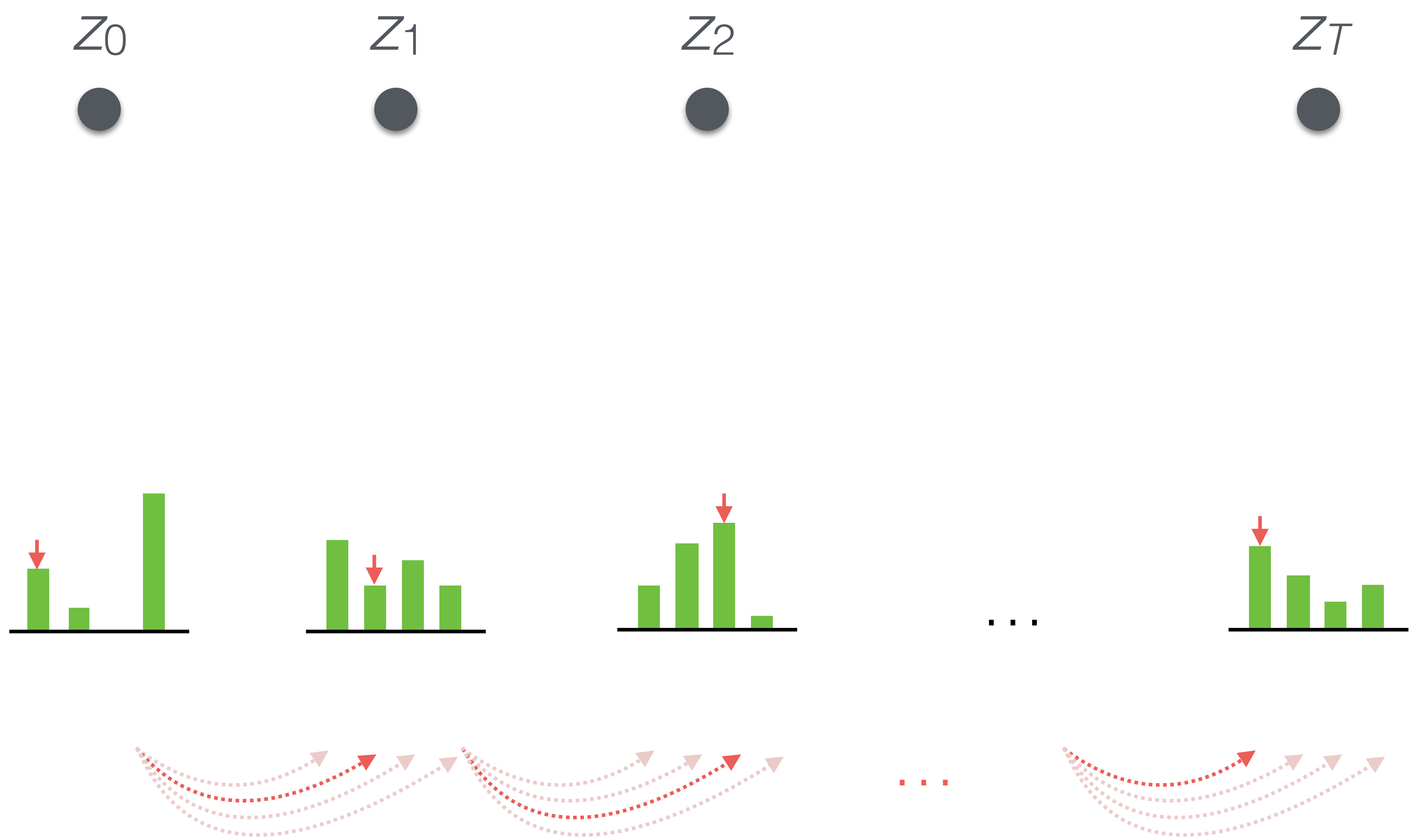


TÉCNICO LISBOA

Idea



Idea



Maximizing forward mapping

Maximizing forward mapping

Given a sequence of observations $\mathbf{z}_{0:t}$, the maximizing forward mapping $\mathrm{m}_t : \mathscr{X} \longmapsto \mathbb{R}$ is defined for each t as

$$m_t(x) = \max_{\boldsymbol{x}_{0:t-1}} \mathbb{P}_{\mu_0}\left[\mathrm{x}_t = x, \mathbf{x}_{0:t-1} = \boldsymbol{x}_{0:t-1}, \mathbf{z}_{0:t} = \boldsymbol{z}_{0:t}\right]$$

Maximizing sequence ending in x

Viterbi algorithm

Require: Observation sequence $z_{0:T}$

1. Initialize $\boldsymbol{m}_0 \leftarrow \mathrm{diag}(\mathbf{O}_{:,z_0})\boldsymbol{\mu}_0^\top$
2. **for** T = 1, …, *T* **do**
3. $\quad \boldsymbol{m}_t \leftarrow \mathrm{diag}(\mathbf{O}_{:,z_t}) \max\{\mathbf{P}^\top \mathrm{diag}(\boldsymbol{m}_{t-1})\}$ — Forward update
4. $\quad \boldsymbol{i}_t = \mathrm{argmax}\{\mathbf{P}^\top \mathrm{diag}(\boldsymbol{m}_{t-1})\}$ — Index tracking
5. **end for**
6. $x_T^* = \underset{x \in \mathcal{X}}{\mathrm{argmax}}\, m_T(x)$
7. **for** *t* = *T* - 1, …, *0* **do**
8. $\quad x_t^* = i_{t+1}(x_{t+1}^*)$ — Backtrack
9. **end for**
10. **return** $x_{0:T}^*$

Example: The urn problem

- Suppose that

$$\boldsymbol{\mu}_0 = \begin{bmatrix} 0.125 & 0.375 & 0.375 & 0.125 \end{bmatrix}$$

- We observe the sequence of observations

$$\boldsymbol{z}_{0:2} = \{w, w, b\}$$

- What is the most likely sequence up to time $t = 2$?

Step 1: Initialize m_0

- $\boldsymbol{m}_0 \leftarrow \operatorname{diag}(\mathbf{O}_{:,z_0})\boldsymbol{\mu}_0^\top$

$$\begin{array}{c}\begin{bmatrix}1&0\\0&1\\1&0\\0&1\end{bmatrix}\\ \begin{array}{cc}z_0&\end{array}\end{array} \rightarrow \begin{bmatrix}1&0&0&0\\0&0&0&0\\0&0&1&0\\0&0&0&0\end{bmatrix}\begin{bmatrix}0.125\\0.375\\0.375\\0.125\end{bmatrix} \rightarrow \begin{bmatrix}0.125\\0\\0.375\\0\end{bmatrix}$$

Step 2: Compute m_1

- $\boldsymbol{m}_1 \leftarrow \operatorname{diag}(\mathbf{O}_{:,z_1}) \max\{\mathbf{P}^\top \operatorname{diag}(\boldsymbol{m}_0)\}$

$$\begin{bmatrix} 0.2 & 0.2 & 0.05 & 0.05 \\ 0.6 & 0.6 & 0.15 & 0.15 \\ 0.15 & 0.15 & 0.6 & 0.6 \\ 0.05 & 0.05 & 0.2 & 0.2 \end{bmatrix} \begin{bmatrix} 0.125 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.375 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0.025 & 0 & 0.019 & 0 \\ 0.075 & 0 & 0.056 & 0 \\ 0.019 & 0 & 0.225 & 0 \\ 0.006 & 0 & 0.075 & 0 \end{bmatrix}$$

Step 2: Compute m_1

- $\boldsymbol{m}_1 \leftarrow \mathrm{diag}(\mathbf{O}_{:,z_1})\max\{\mathbf{P}^\top \mathrm{diag}(\boldsymbol{m}_0)\}$

$$\begin{bmatrix} \boxed{0.025} & 0 & 0.019 & 0 \\ \boxed{0.075} & 0 & 0.056 & 0 \\ 0.019 & 0 & \boxed{0.225} & 0 \\ 0.006 & 0 & \boxed{0.075} & 0 \end{bmatrix} \longrightarrow \begin{bmatrix} 0.025 \\ 0.075 \\ 0.225 \\ 0.075 \end{bmatrix}$$

Step 2: Compute m_1

- $\boldsymbol{m}_1 \leftarrow \text{diag}(\mathbf{O}_{:,z_1}) \max\{\mathbf{P}^\top \text{diag}(\boldsymbol{m}_0)\}$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.025 \\ 0.075 \\ 0.225 \\ 0.075 \end{bmatrix} \longrightarrow \begin{bmatrix} 0.025 \\ 0 \\ 0.225 \\ 0 \end{bmatrix}$$

Step 3: Compute i_1

- $i_1 = \text{argmax}\{\mathbf{P}^\top \text{diag}(\boldsymbol{m}_0)\}$

$$\begin{bmatrix} \boxed{0.025} & 0 & 0.019 & 0 \\ \boxed{0.075} & 0 & 0.056 & 0 \\ 0.019 & 0 & \boxed{0.225} & 0 \\ 0.006 & 0 & \boxed{0.075} & 0 \end{bmatrix} \longrightarrow \begin{bmatrix} 1 \\ 1 \\ 3 \\ 3 \end{bmatrix}$$

Step 4: Compute m_2

- $\boldsymbol{m}_2 \leftarrow \operatorname{diag}(\mathbf{O}_{:,z_2}) \max\{\mathbf{P}^\top \operatorname{diag}(\boldsymbol{m}_1)\}$

$$\begin{bmatrix} 0.2 & 0.2 & 0.05 & 0.05 \\ 0.6 & 0.6 & 0.15 & 0.15 \\ 0.15 & 0.15 & 0.6 & 0.6 \\ 0.05 & 0.05 & 0.2 & 0.2 \end{bmatrix} \begin{bmatrix} 0.025 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.225 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0.005 & 0 & 0.011 & 0 \\ 0.015 & 0 & 0.034 & 0 \\ 0.004 & 0 & 0.135 & 0 \\ 0.001 & 0 & 0.045 & 0 \end{bmatrix}$$

Step 4: Compute m_2

- $\boldsymbol{m}_2 \leftarrow \mathrm{diag}(\mathbf{O}_{:,z_2})\max\{\mathbf{P}^\top\mathrm{diag}(\boldsymbol{m}_1)\}$

$$\begin{bmatrix} 0.005 & 0 & \boxed{0.011} & 0 \\ 0.015 & 0 & \boxed{0.034} & 0 \\ 0.004 & 0 & \boxed{0.135} & 0 \\ 0.001 & 0 & \boxed{0.045} & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0.011 \\ 0.034 \\ 0.135 \\ 0.045 \end{bmatrix}$$

Step 4: Compute m_2

- $\boldsymbol{m}_2 \leftarrow \operatorname{diag}(\mathbf{O}_{:,z_2}) \max\{\mathbf{P}^\top \operatorname{diag}(\boldsymbol{m}_1)\}$

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.011 \\ 0.034 \\ 0.135 \\ 0.045 \end{bmatrix} \rightarrow \begin{bmatrix} 0 \\ 0.034 \\ 0 \\ 0.045 \end{bmatrix}$$

Step 5: Compute i_2

- $\boldsymbol{i}_1 = \text{argmax}\{\mathbf{P}^\top \text{diag}(\boldsymbol{m}_0)\}$

$$\begin{bmatrix} 0.005 & 0 & \boxed{0.011} & 0 \\ 0.015 & 0 & \boxed{0.034} & 0 \\ 0.004 & 0 & \boxed{0.135} & 0 \\ 0.001 & 0 & \boxed{0.045} & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 3 \\ 3 \\ 3 \\ 3 \end{bmatrix}$$

Step 6: Maximize m_2

- $x_2^* = \underset{x \in \mathcal{X}}{\operatorname{argmax}}\, m_2(x)$

$$m_2$$

$$\begin{bmatrix} 0 \\ 0.034 \\ 0 \\ \boxed{0.045} \end{bmatrix}$$

$$x_2^* = 4$$

Step 7: Backtrack

- $x_t^* = i_{t+1}(x_{t+1}^*)$

$$\begin{array}{ccc} i_1 & i_2 & m_2 \\ \begin{bmatrix}1\\1\\\boxed{3}\\3\end{bmatrix} & \begin{bmatrix}3\\3\\3\\\boxed{3}\end{bmatrix} & \begin{bmatrix}0\\0.034\\0\\\boxed{0.045}\end{bmatrix} \\ x_0^* = 3 & x_1^* = 3 & x_2^* = 4 \end{array}$$

Finally…

- Most likely sequence:

$$x_0^* = 3 \qquad x_1^* = 3 \qquad x_2^* = 4$$

(2, w) (2, w) (2, b)

Estimation

- **Filtering:**
  - Given a sequence of observations, estimate the final state
- **Smoothing:**
  - Given a sequence of observations, estimate the sequence of states
- **Prediction:**
  - Given a sequence of observations, predict future states

Prediction

- We are given a sequence of observations $\boldsymbol{z}_{0:T}$
- We want to estimate, for $t > T$

$$\mathbb{P}_{\mu_0}\left[\mathrm{x}_t = x \mid \mathbf{z}_{0:T} = \boldsymbol{z}_{0:T}\right]$$

where μ_0 is the initial distribution, i.e.,

$$\mu_0(x) = \mathbb{P}\left[\mathrm{x}_0 = x\right]$$

Prediction

- Easy:
  - We compute $\mu_{T\,|\,0:T}$ using the forward algorithm
  - We use the Markov property:

$$\boldsymbol{\mu}_{T+1|0:T} = \boldsymbol{\mu}_{T|0:T}\mathbf{P}$$