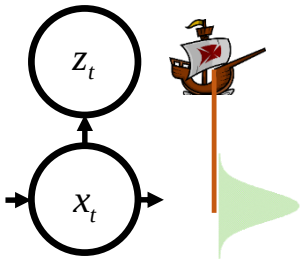
Important Distributions (Particle Filter)

Observation Model:

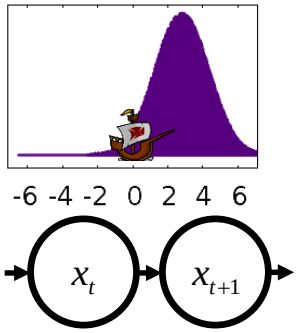
$p(z_t|x_t)$



- Likelihood of observation given state
- Continuous Gaussian around real depth

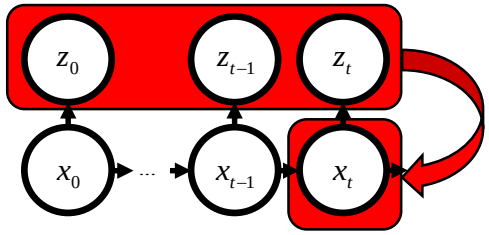
Motion Model:

$p(x_{t+1}|x_t)$



- Probability of new state given old one
- **Continuous Gaussian**

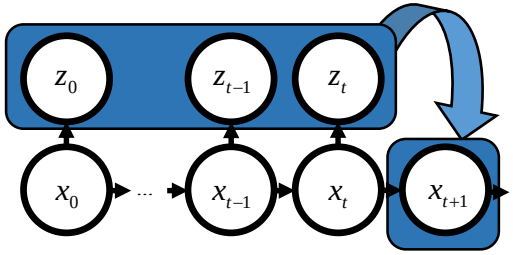
Posterior:



$p(x_t|z_{0:t})$

- Probability of state given previous and current observations
- **Continuous, represented as set of Samples (Particles)**

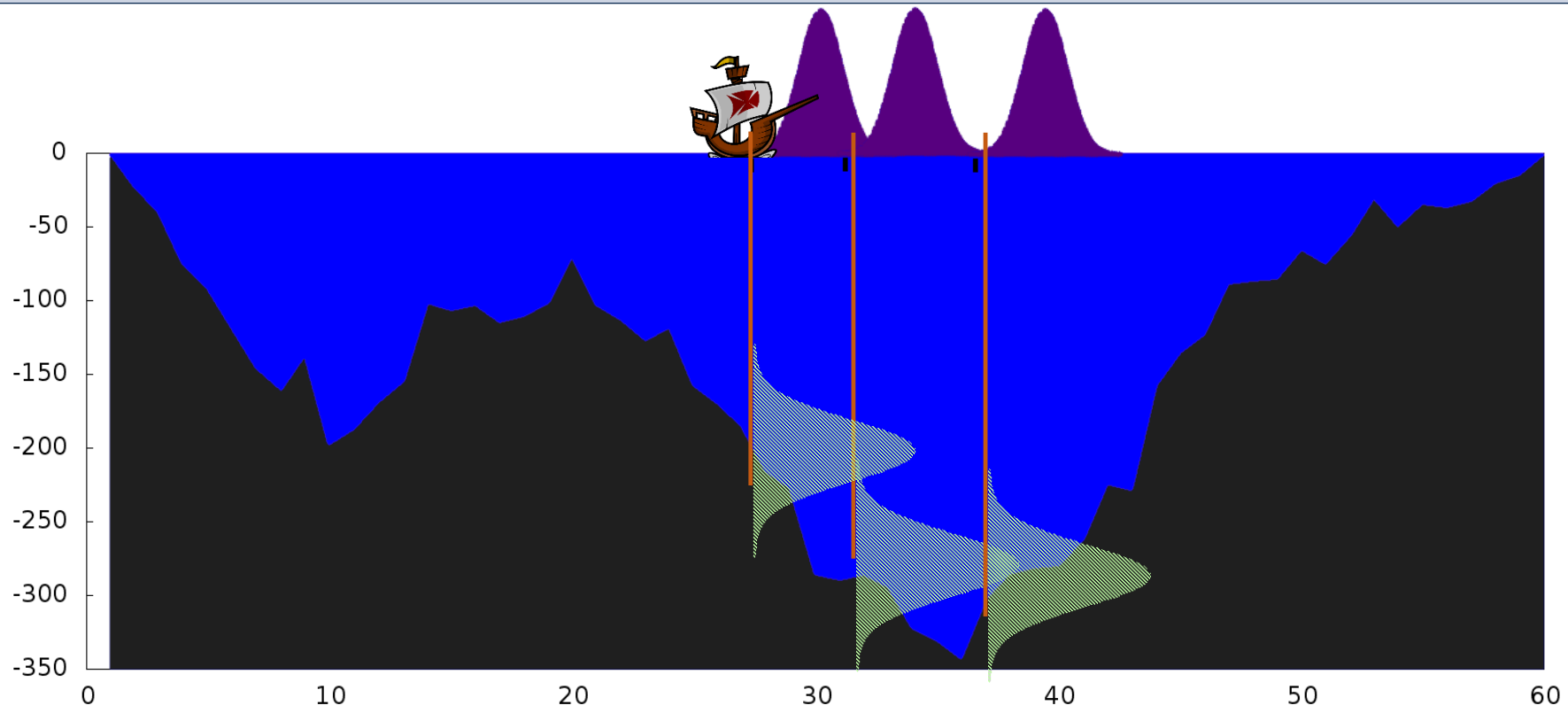
Prior:



$p(x_{t+1}|z_{0:t})$

- Probability of state given only previous observations
- **Continuous, represented as set of Samples (Particles)**

Particle Filter



Discrete Bayes Filter vs. Particle Filter

- Discrete Bayes Filter:

1. Make observation

2. Calculate likelihood for every position

3. **Multiply** with last prior and normalize

4. **Convolution** with motion model

5. Go to 1.

- Particle Filter:

1. Make observation

2. Calculate likelihood for every **sample** -> weights

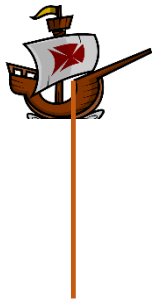
3. **Resampling** according to weights

4. Randomly move samples according to motion model (**Sampling**)

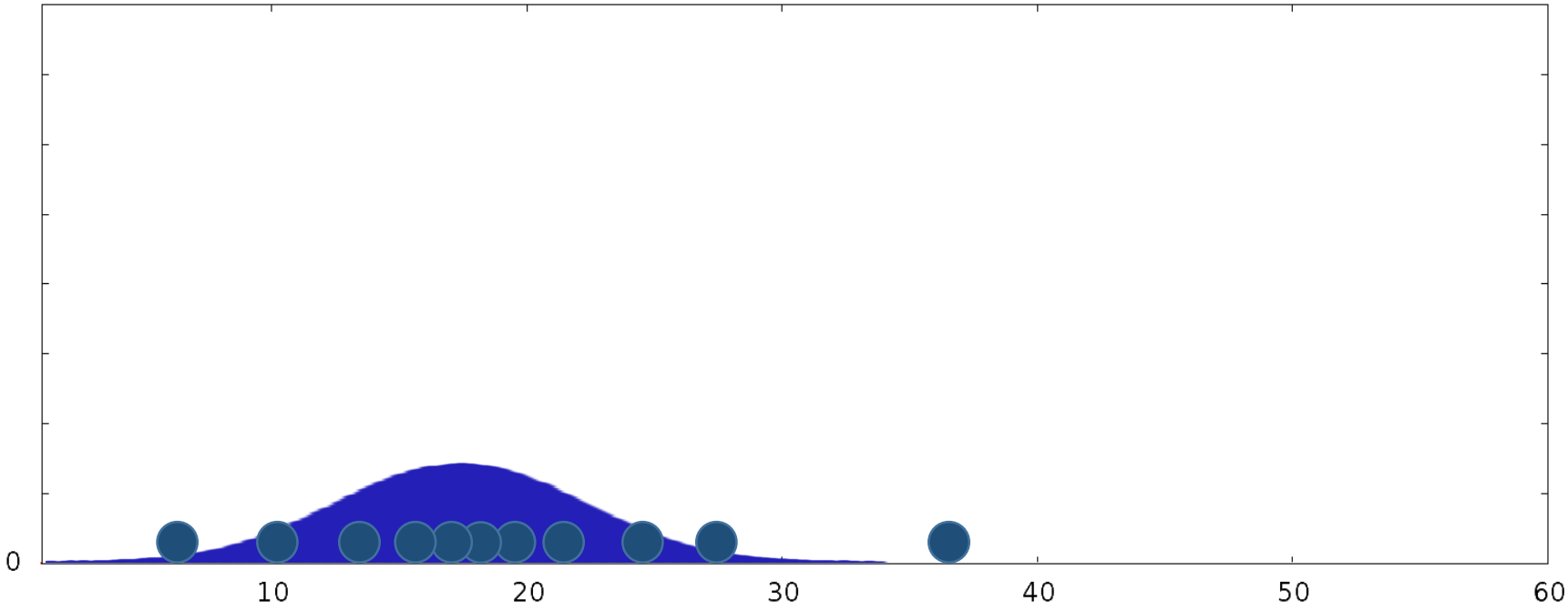
5. Go to 1.

The Bayes Filter / Convolute and Multiply

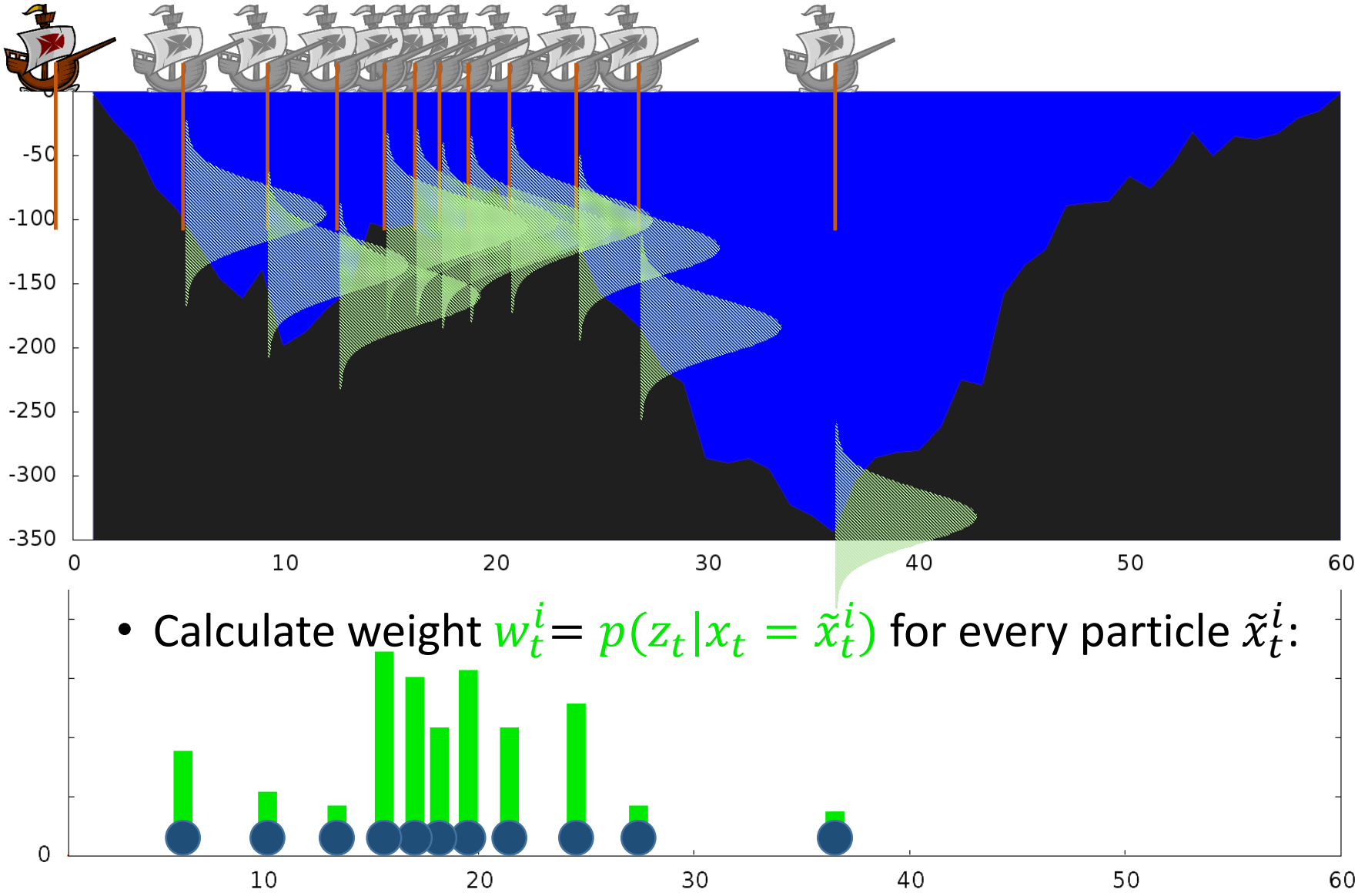
$t = 0$



- Represent continuous prior with particles $\tilde{x}_t^1, \ldots, \tilde{x}_t^n$
- Make measurement: z_t



Particle Filter / Resampling

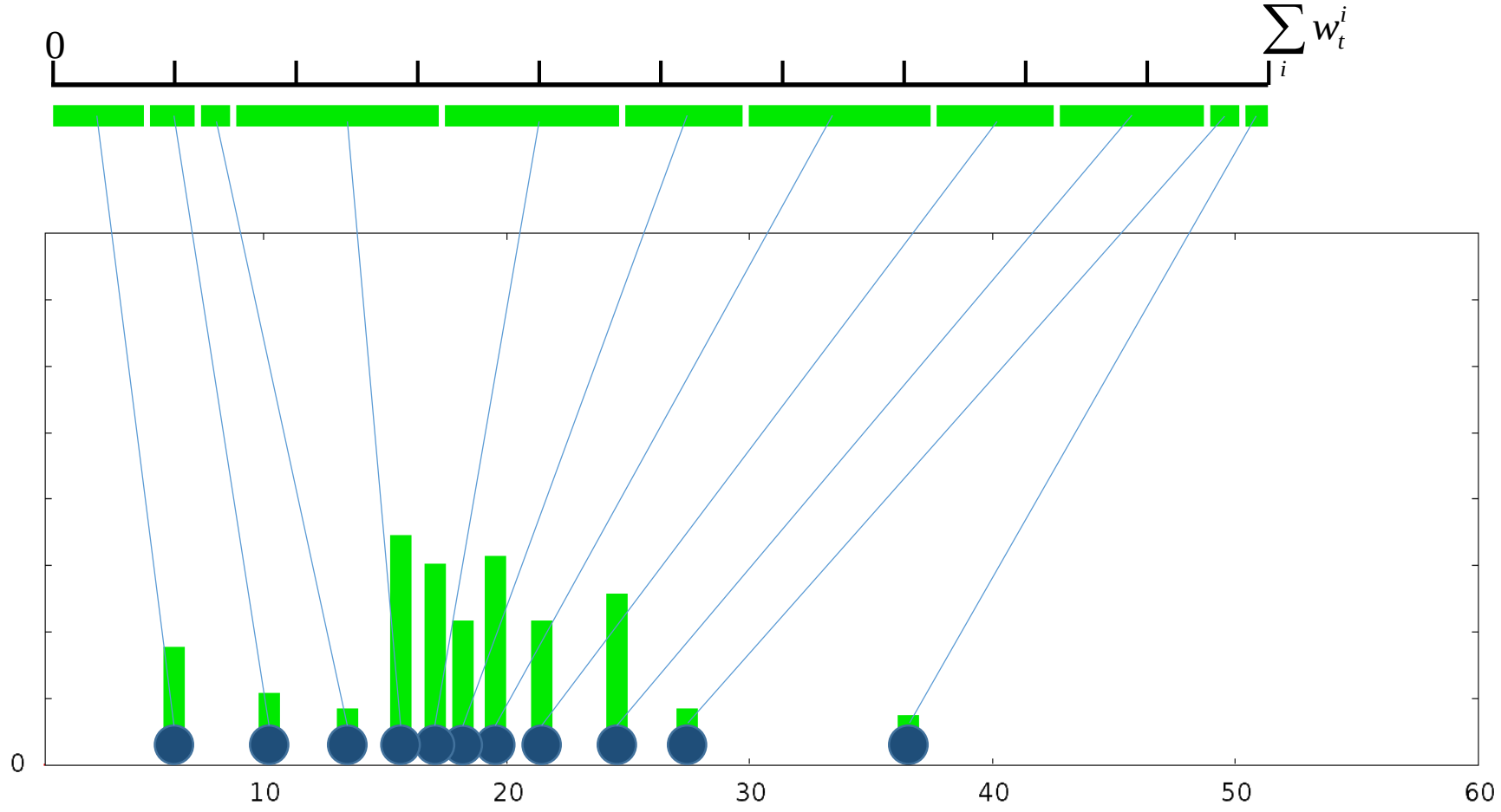


- Calculate weight $w_t^i = p(z_t | x_t = \tilde{x}_t^i)$ for every particle $\tilde{x}_t^i$:

Particle Filter / Resampling

$t = 0$

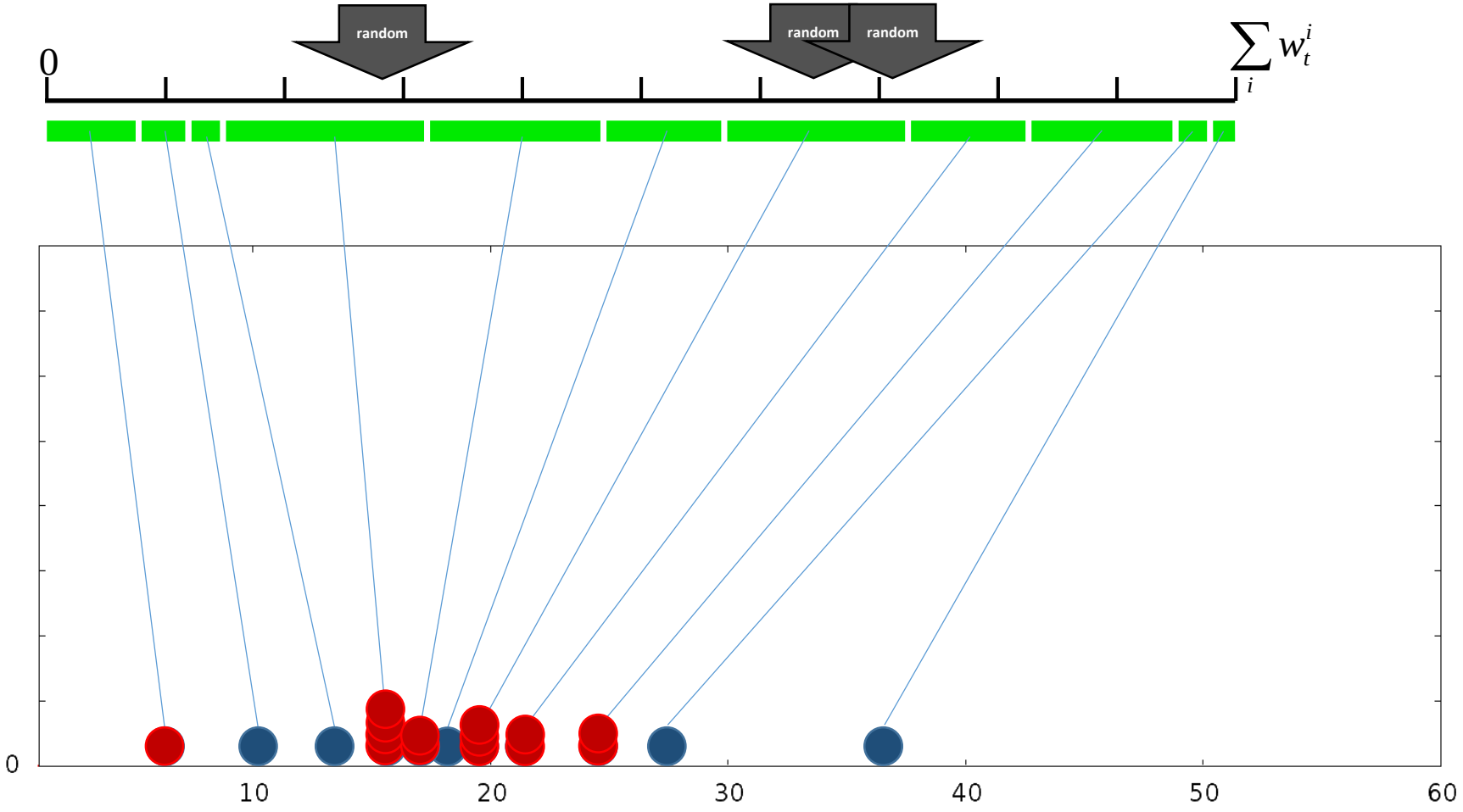
- Draw samples $x_t^1, \ldots, x_t^n$ from **posterior** by resampling from $\tilde{x}_t^1, \ldots, \tilde{x}_t^n$ using the weights w_t^i
- Reducing uncertainty



Particle Filter / Resampling

$t = 0$

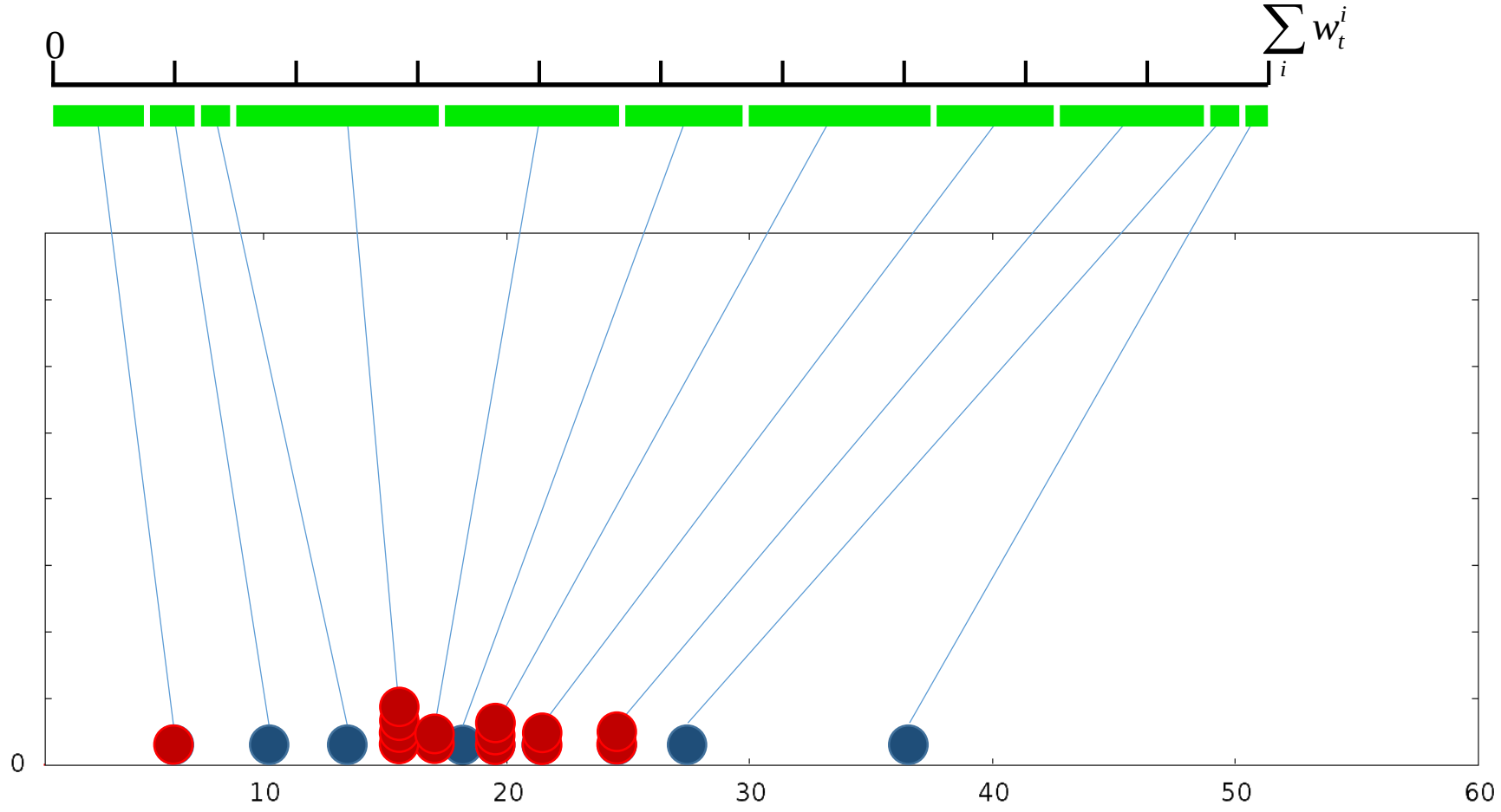
- Draw samples $x_t^1, \ldots, x_t^n$ from **posterior** by resampling from $\tilde{x}_t^1, \ldots, \tilde{x}_t^n$ using the weights w_t^i
- Reducing uncertainty



Particle Filter / Resampling

$t = 0$

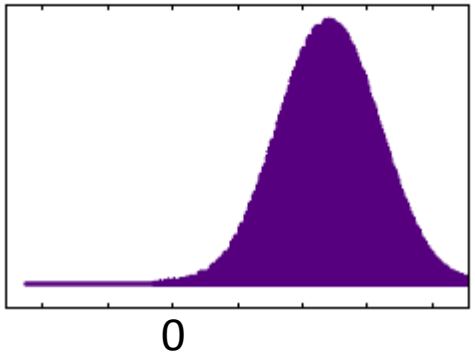
- Draw samples $x_t^1, \ldots, x_t^n$ from **posterior** by resampling from $\tilde{x}_t^1, \ldots, \tilde{x}_t^n$ using the weights w_t^i
- Reducing uncertainty

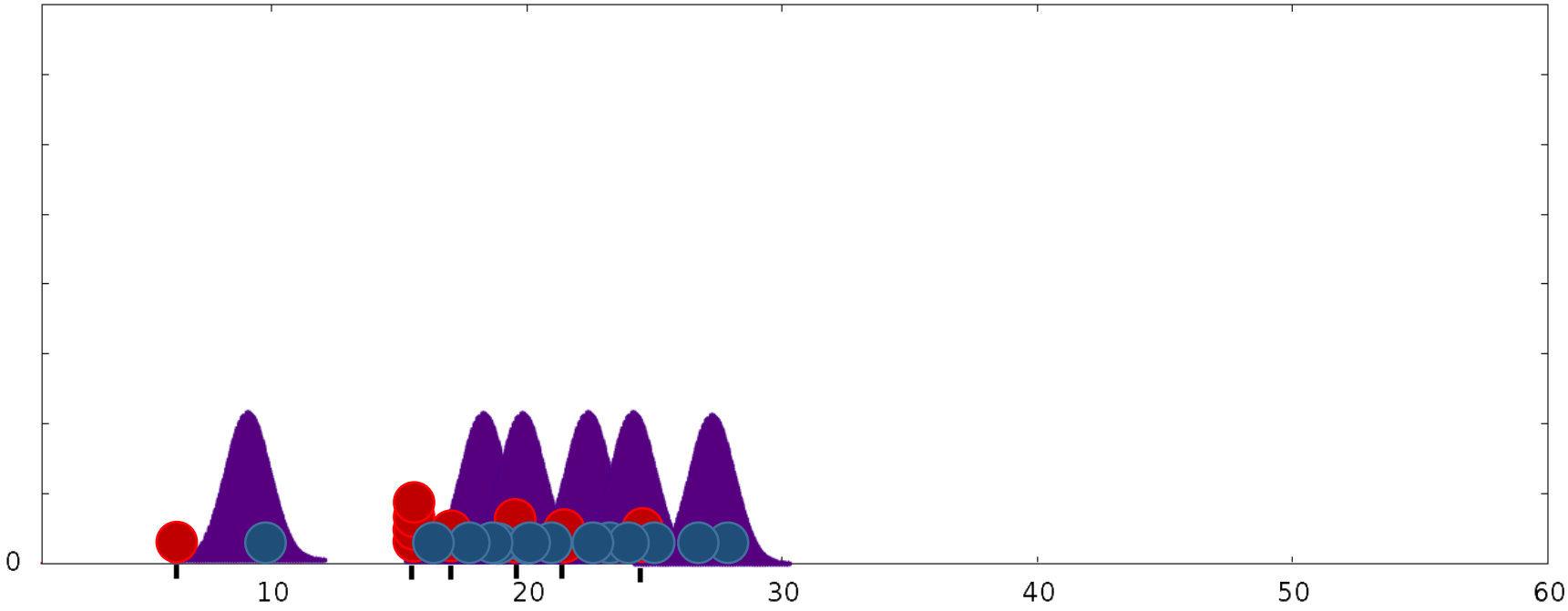


Particle Filter / Sampling

$t = 0$

- Obtain samples $\tilde{x}_{t+1}^1, \ldots, \tilde{x}_{t+1}^n$ from the **new prior** by moving each particle according to motion model
- Adding uncertainty





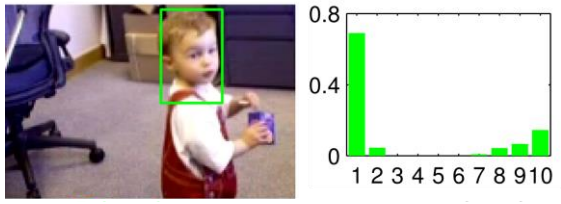
Particle Filter (Tracking application)

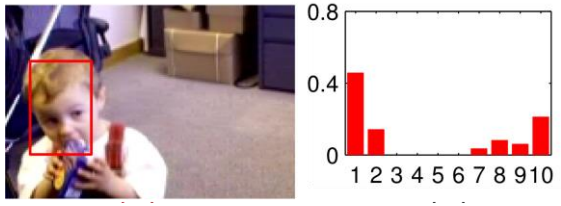
- Tracking an object in a video sequence [Perez at al. 2002]

- States:
  - 2D windows (location and size)

- Gaussian motion

- Observations:
  - Color histograms





Your Task

1. Draw the particles and the mean particle into the image (0p)
   - You still have to do it get other points
2. Implement the *MotionModel* class and move particles in the *processFrame*function:
   - Gaussian distribution around last position (1p)
   - Gaussian distribution around extrapolated position (1p)
     - (Based on previous motion **of each particle**)
3. Implement the *ObservationModel* class and calculate the weights and the mean particle (avg. x/y position and avg. window size) in the *processFrame* function. (1p)
4. Implement the *resampleParticles*function. There are different ways to do this:
   - Either naïve way (1p)
   - Or smarter way: binary search or *std::map* (2p)
5. (extra task) parallelize your code with openMP. Can you increase FPS? You might have to use multiple random number engines for multiple threads. (up to 1p)