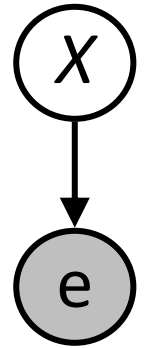


Warm-up as you log in

When sampling with likelihood weighting, what distribution do we have when we multiply fraction of counts times the weight?



Announcements

Assignments (everything left for the semester):

- HW11 (online) due Tue 4/21
- P5 due Thu 4/30
- HW12 (written) out Tue 4/21, due Tue 4/28

Participation points

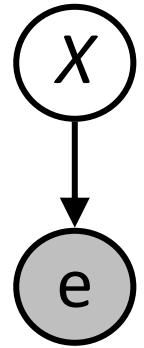
- Starting now, we're capping the denominator (63 polls) in the participation points calculation

Final Exam:

- 5/4 1-4pm (let us know by today 4/20 if you need it rescheduled)

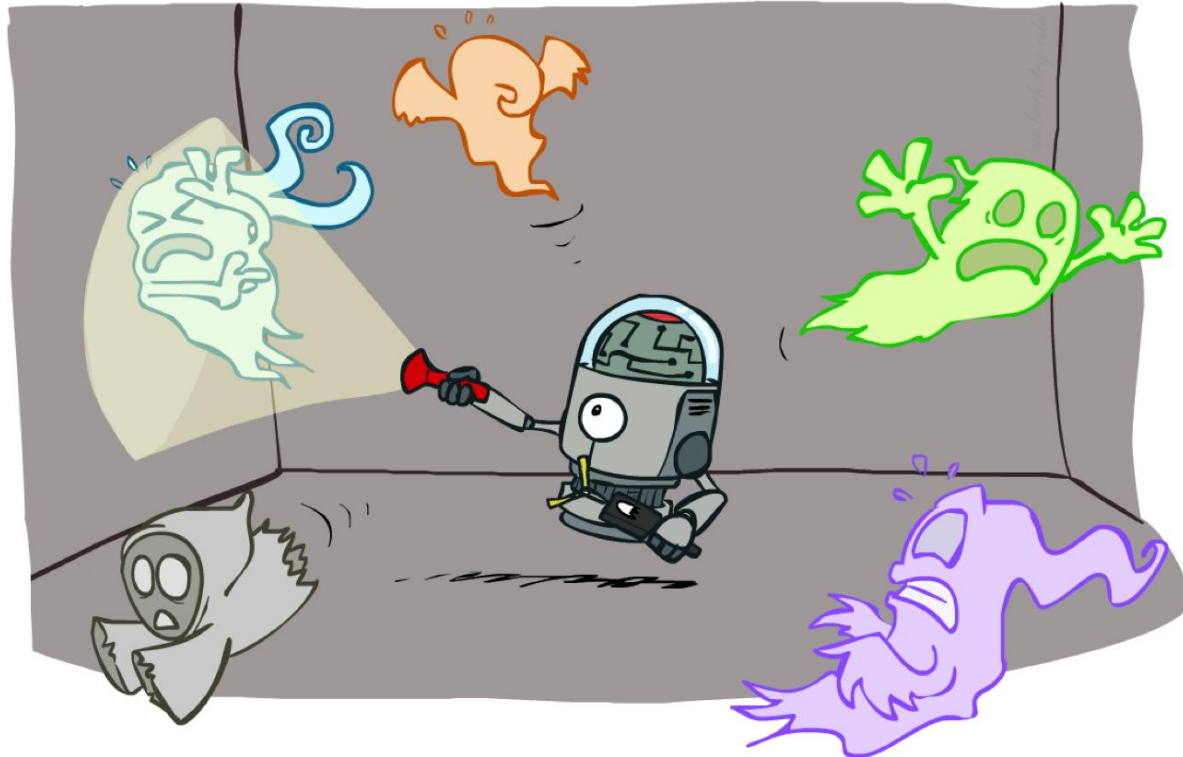
Warm-up as you log in

When sampling with likelihood weighting, what distribution do we have when we multiply fraction of counts times the weight?



AI: Representation and Problem Solving

HMMs and Particle Filtering

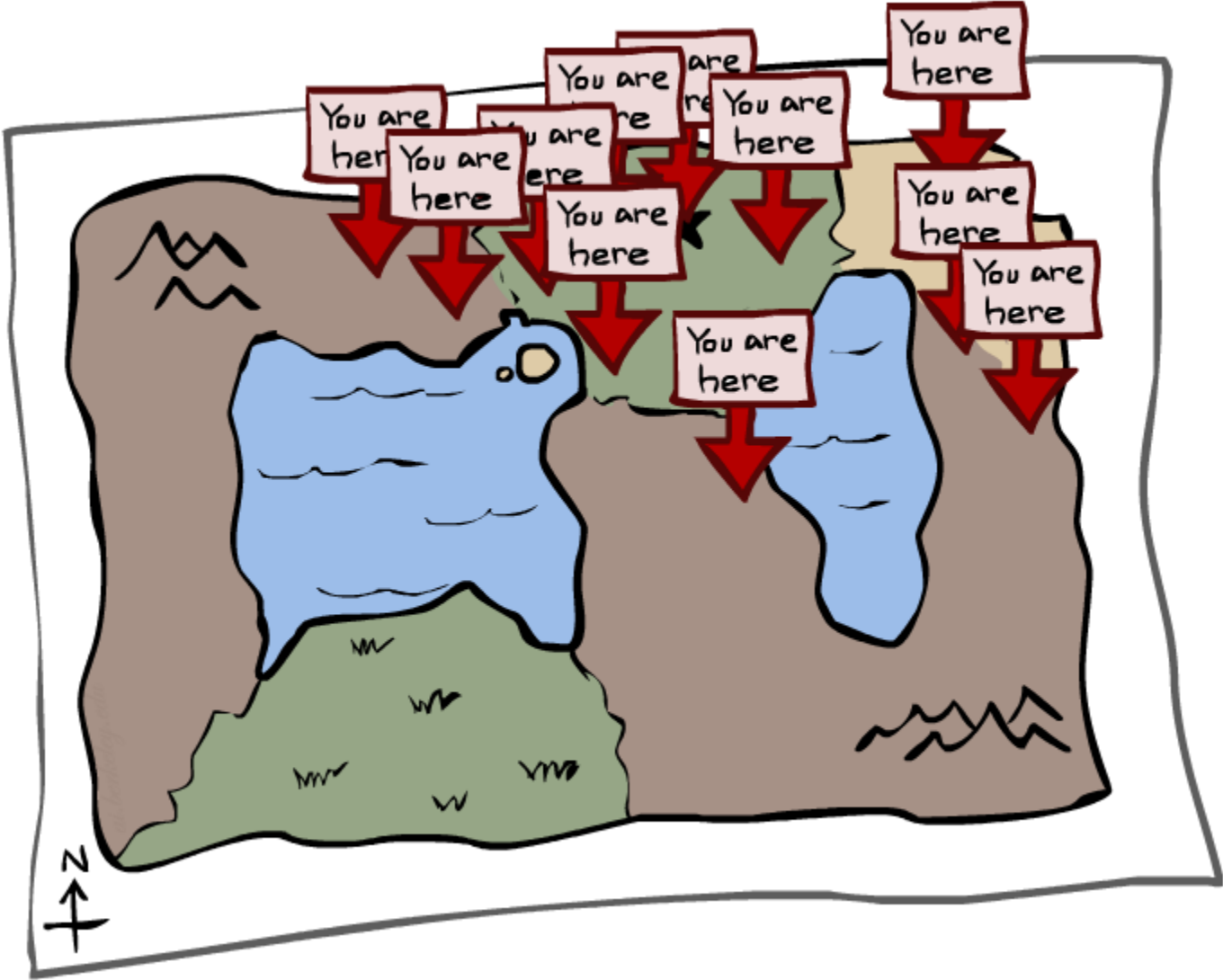


Instructors: Pat Virtue & Stephanie Rosenthal

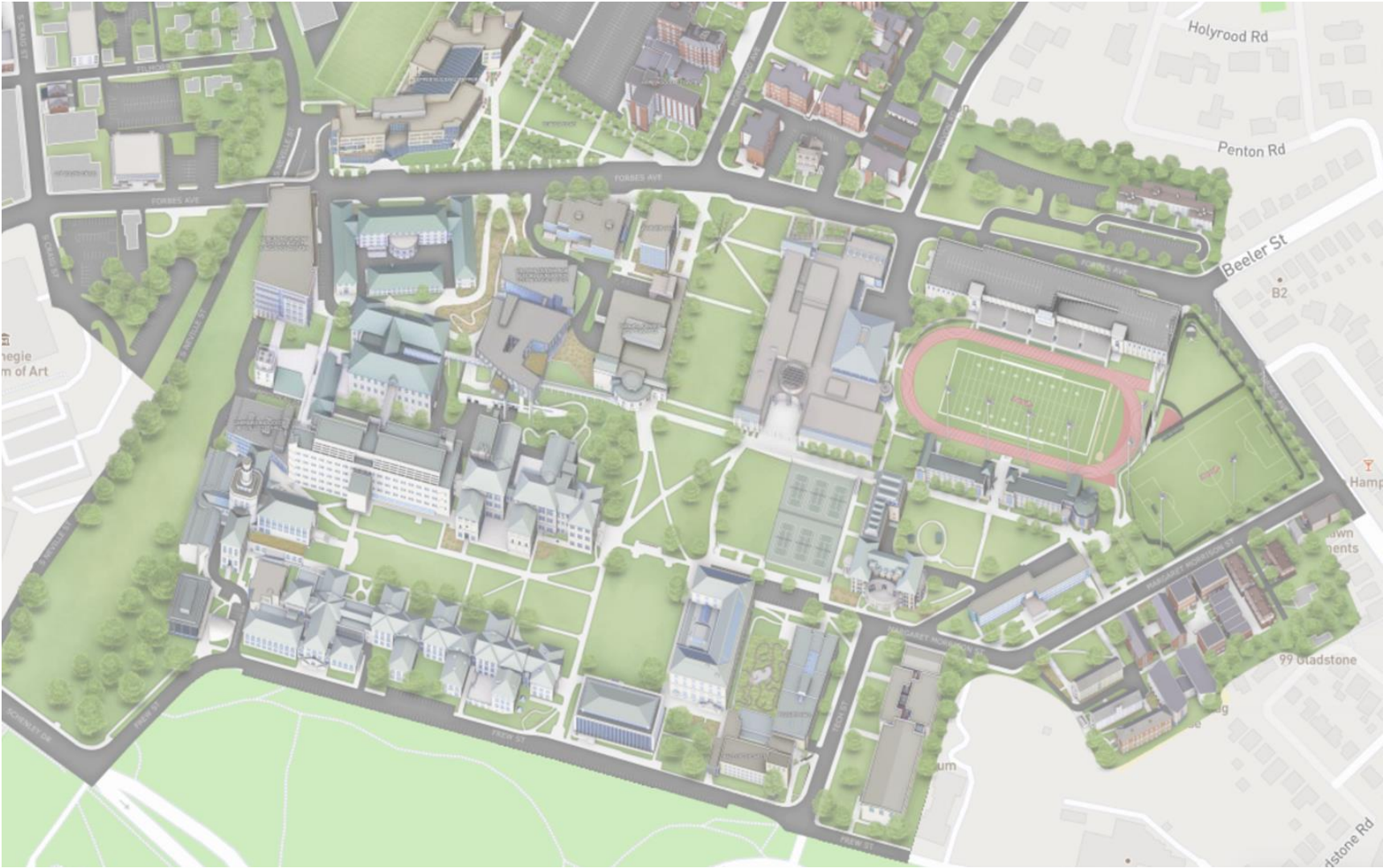
Slide credits: CMU AI and <http://ai.berkeley.edu>

Demo: Pacman Ghostbusters

Particle Filtering



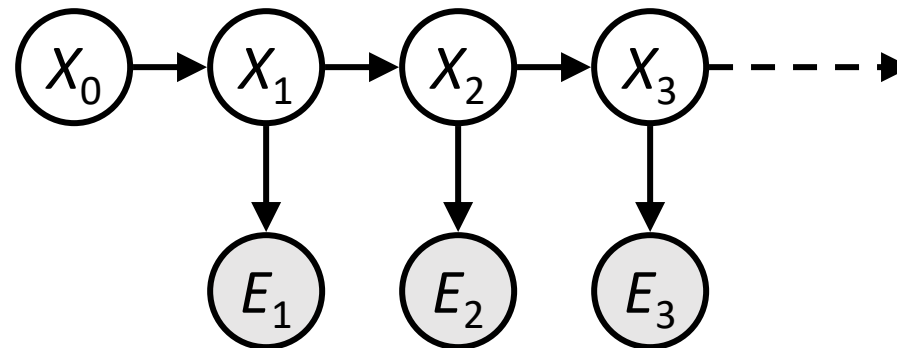
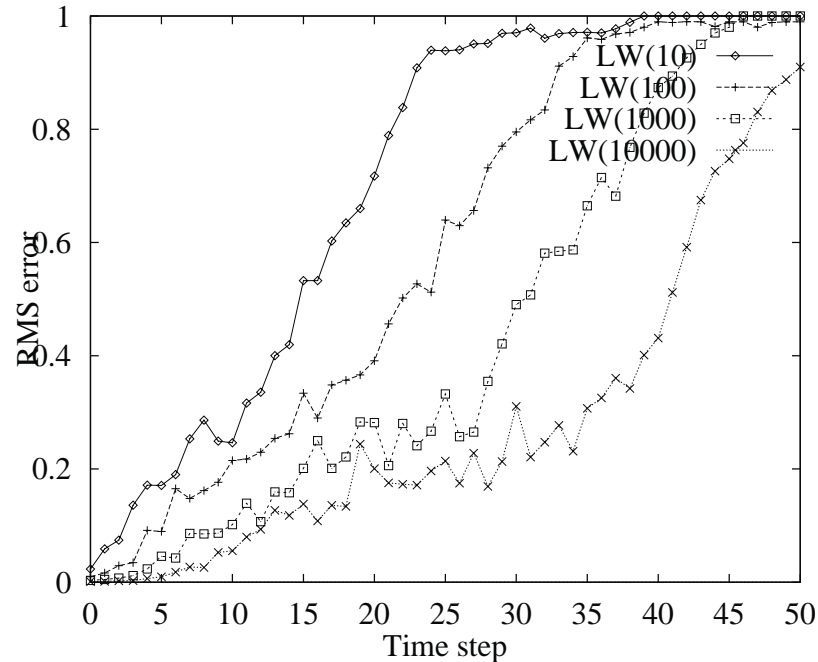
Particle Filtering



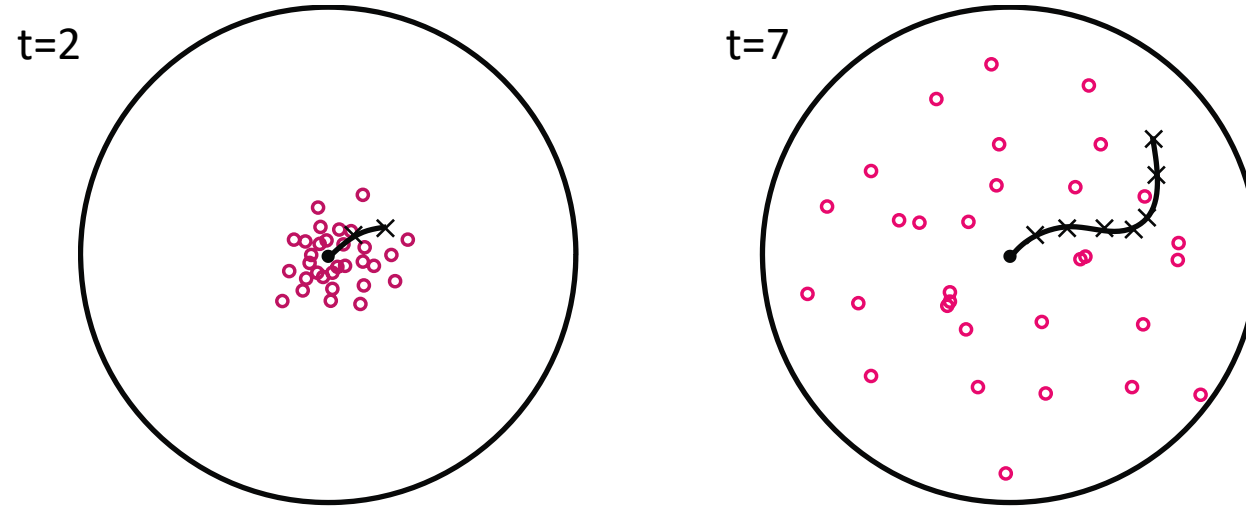
We need a new algorithm!

When $|X|$ is more than 10^6 or so (e.g., 3 ghosts in a 10x20 world), exact inference becomes infeasible

Likelihood weighting fails completely – number of samples needed grows *exponentially* with T



We need a new idea!



The problem: sample state trajectories go off into low-probability regions, ignoring the evidence; too few “reasonable” samples

Solution: kill the bad ones, make more of the good ones

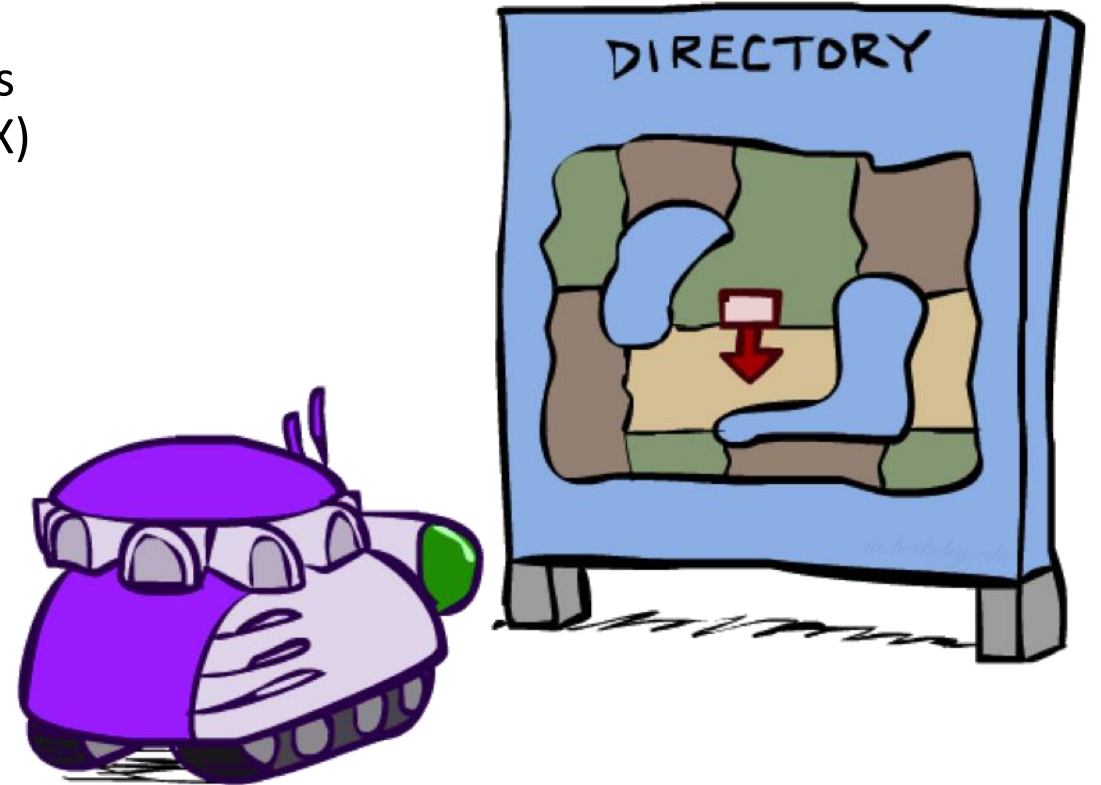
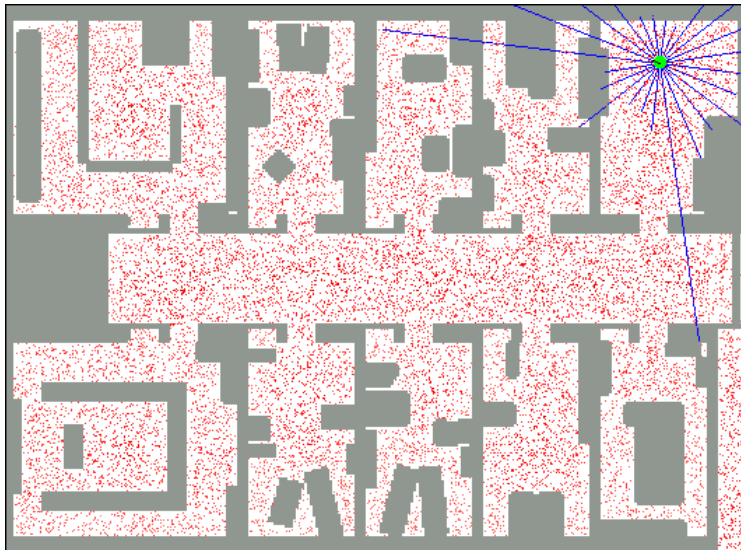
This way the population of samples stays in the high-probability region

This is called **resampling** or survival of the fittest

Robot Localization

In robot localization:

- We know the map, but not the robot's position
- Observations may be vectors of range finder readings
- State space and readings are typically continuous (works basically like a very fine grid) and so we cannot store $B(X)$
- Particle filtering is a main technique



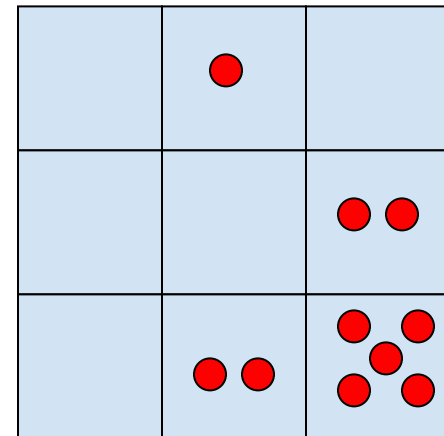
Particle Filter Localization (Sonar)



Particle Filtering

- Represent belief state by a set of samples
 - Samples are called *particles*
 - Time per step is linear in the number of samples
 - But: number needed may be large
- This is how robot localization works in practice

0.0	0.1	0.0
0.0	0.0	0.2
0.0	0.2	0.5



Representation: Particles

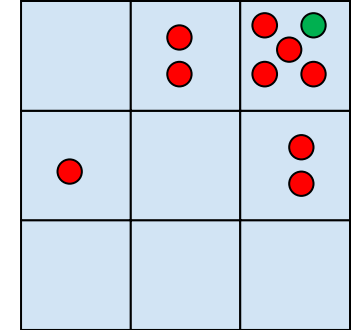
Our representation of $P(X)$ is now a list of N particles (samples)

- Generally, $N \ll |X|$
- Storing map from X to counts would defeat the point

$P(x)$ approximated by number of particles with value x

- So, many x may have $P(x) = 0$!
- More particles, more accuracy
- Usually we want a low-dimensional marginal
 - E.g., “Where is ghost 1?” rather than “Are ghosts 1,2,3 in {2,6}, [5,6], and [8,11]?”

For now, all particles have a weight of 1



Particles:

(3,3)

(2,3)

(3,3)

(3,2)

(3,3)

(3,2)

(1,2)

(3,3)

(3,3)

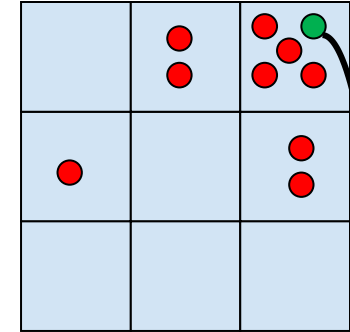
(2,3)

Particle Filtering: Propagate forward

- A particle in state x_t is moved by sampling its next position directly from the transition model:
 - $x_{t+1} \sim P(X_{t+1} | x_t)$
 - Here, most samples move clockwise, but some move in another direction or stay in place
- This captures the passage of time
 - If enough samples, close to exact values before and after (consistent)

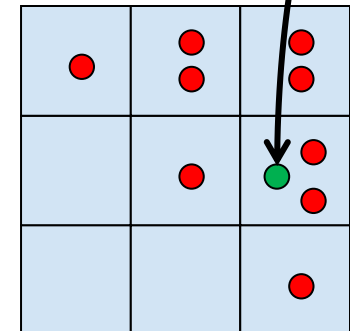
Particles:

(3,3)
(2,3)
(3,3)
(3,2)
(3,3)
(3,2)
(1,2)
(3,3)
(3,3)
(2,3)



Particles:

(3,2)
(2,3)
(3,2)
(3,1)
(3,3)
(3,2)
(1,3)
(2,3)
(3,2)
(2,2)

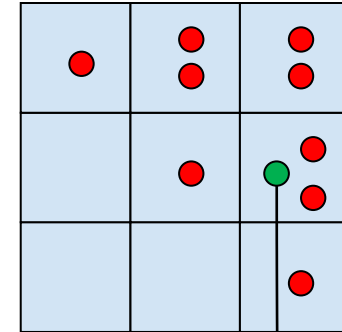


Particle Filtering: Observe

- Slightly trickier:
 - Don't sample observation, fix it
 - Similar to likelihood weighting, weight samples based on the evidence
 - $W = P(e_t | x_t)$
 - Normalize the weights: particles that fit the data better get higher weights, others get lower weights

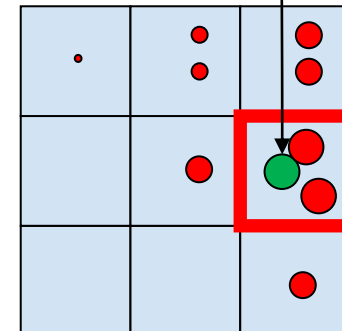
Particles:

(3,2)
(2,3)
(3,2)
(3,1)
(3,3)
(3,2)
(1,3)
(2,3)
(3,2)
(2,2)



Particles:

(3,2) w=.9
(2,3) w=.2
(3,2) w=.9
(3,1) w=.4
(3,3) w=.4
(3,2) w=.9
(1,3) w=.1
(2,3) w=.2
(3,2) w=.9
(2,2) w=.4



Particle Filtering: Resample

Rather than tracking weighted samples, we *resample*

We have an updated belief distribution based on the weighted particles

We sample N new particles from the weighted belief distributions

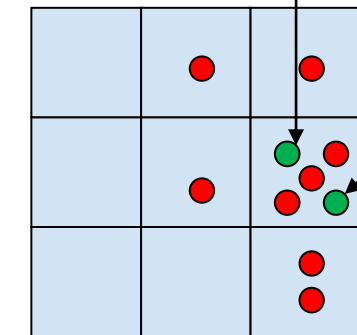
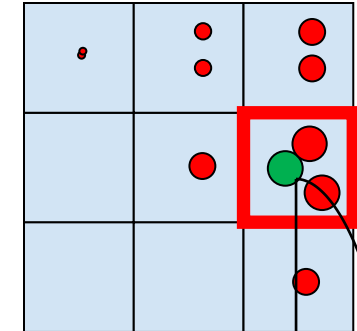
Now the update is complete for this time step, continue with the next one

Particles:

- (3,2) $w=.9$
- (2,3) $w=.2$
- (3,2) $w=.9$
- (3,1) $w=.4$
- (3,3) $w=.4$
- (3,2) $w=.9$
- (1,3) $w=.1$
- (2,3) $w=.2$
- (3,2) $w=.9$
- (2,2) $w=.4$

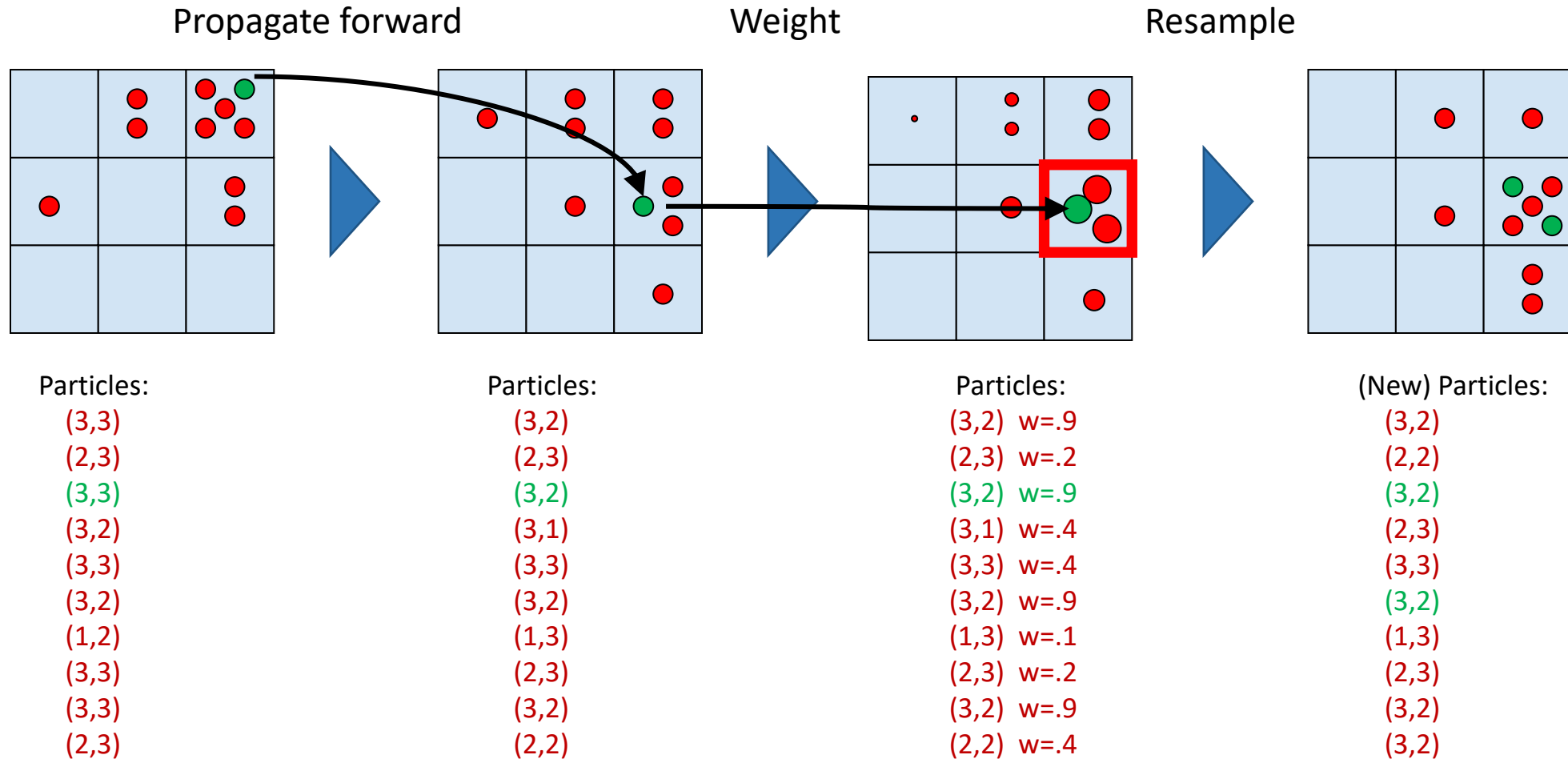
(New) Particles:

- (3,2)
- (2,2)
- (3,2)
- (2,3)
- (3,3)
- (3,2)
- (1,3)
- (2,3)
- (3,2)
- (3,2)



Summary: Particle Filtering

Particles: track samples of states rather than an explicit distribution



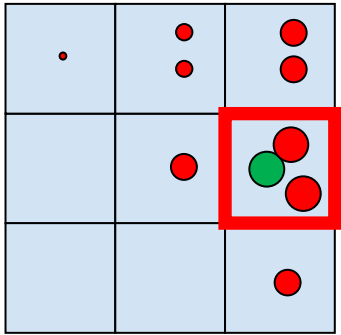
Consistency: see proof in AIMA Ch. 15

[Demos: ghostbusters particle filtering (L15D3,4,5)]

Weighting and Resampling

How to compute a belief distribution given weighted particles

Weight

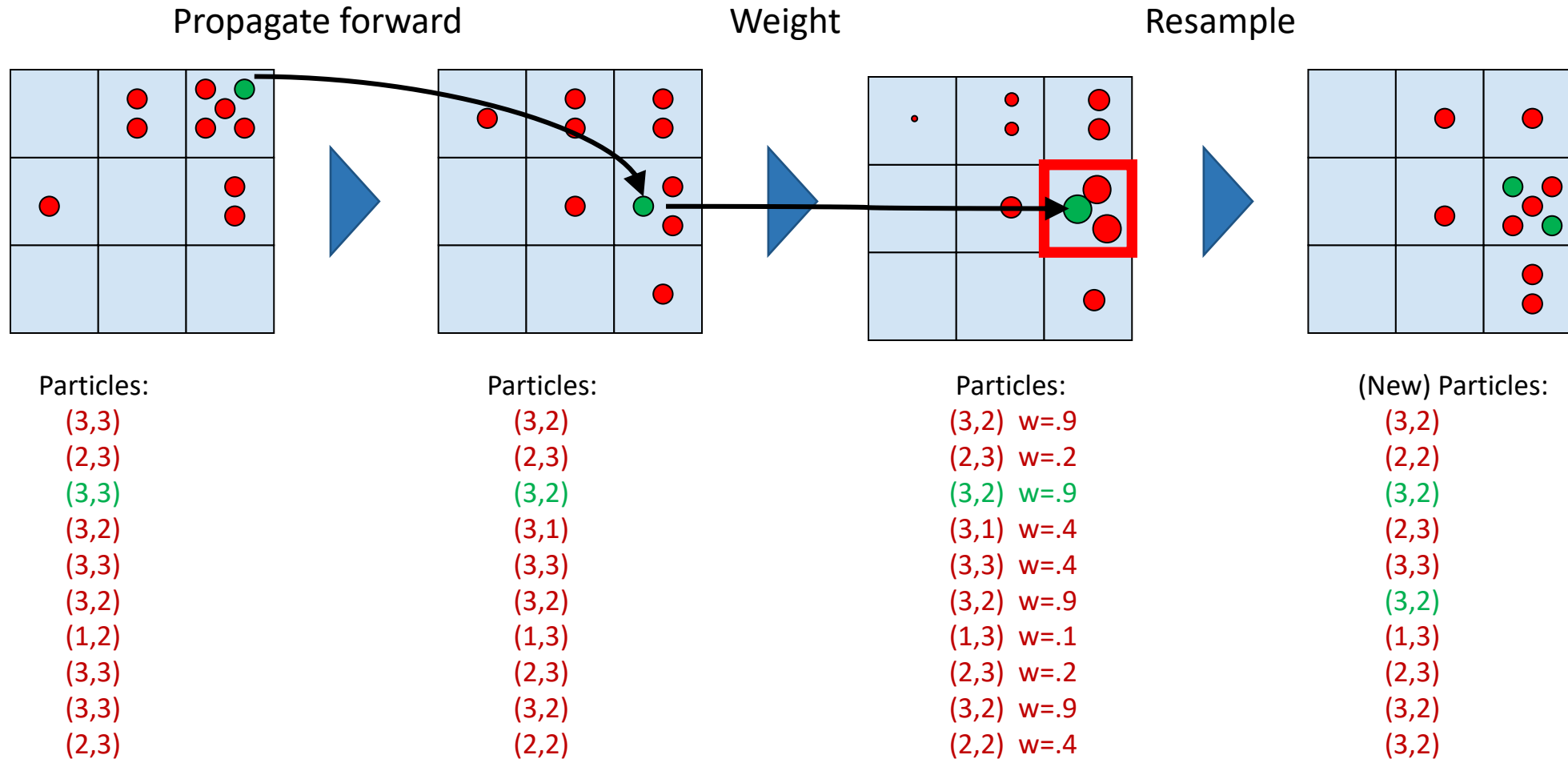


Particles:

- (3,2) $w=.9$
- (2,3) $w=.2$
- (3,2) $w=.9$
- (3,1) $w=.4$
- (3,3) $w=.4$
- (3,2) $w=.9$
- (1,3) $w=.1$
- (2,3) $w=.2$
- (3,2) $w=.9$
- (2,2) $w=.4$

Summary: Particle Filtering

Particles: track samples of states rather than an explicit distribution

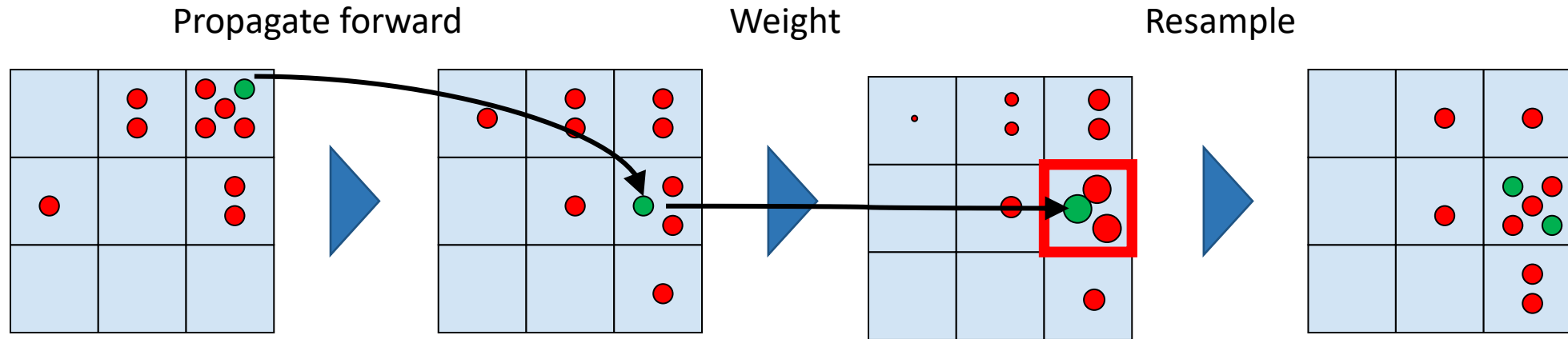


Consistency: see proof in AIMA Ch. 15

[Demos: ghostbusters particle filtering (L15D3,4,5)]

Piazza Poll 1

If we only have one particle which of these steps are unnecessary?

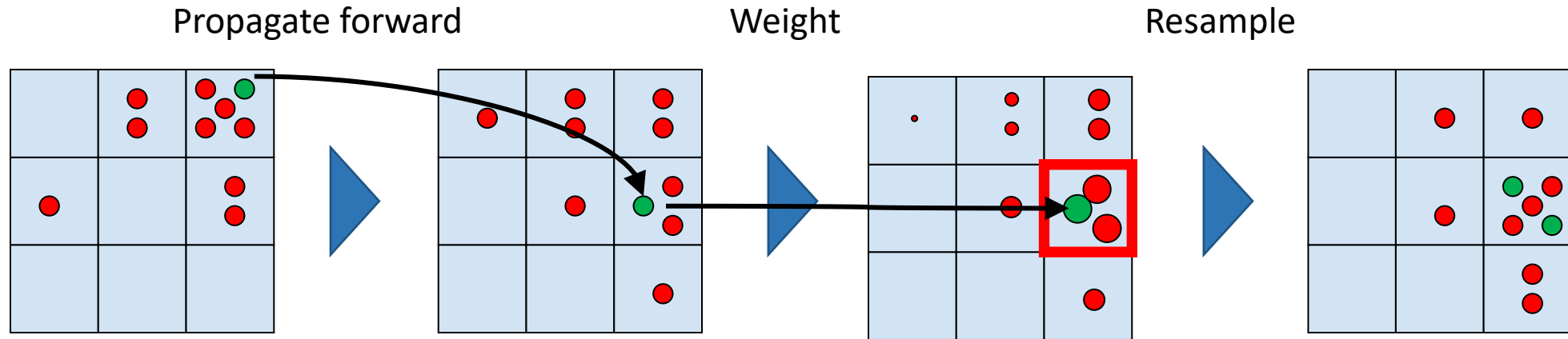


Select all that are unnecessary.

- A. Propagate forward
- B. Weight
- C. Resample
- D. None of the above

Piazza Poll 1

If we only have one particle which of these steps are unnecessary?



Select all that are unnecessary.

A. Propagate forward

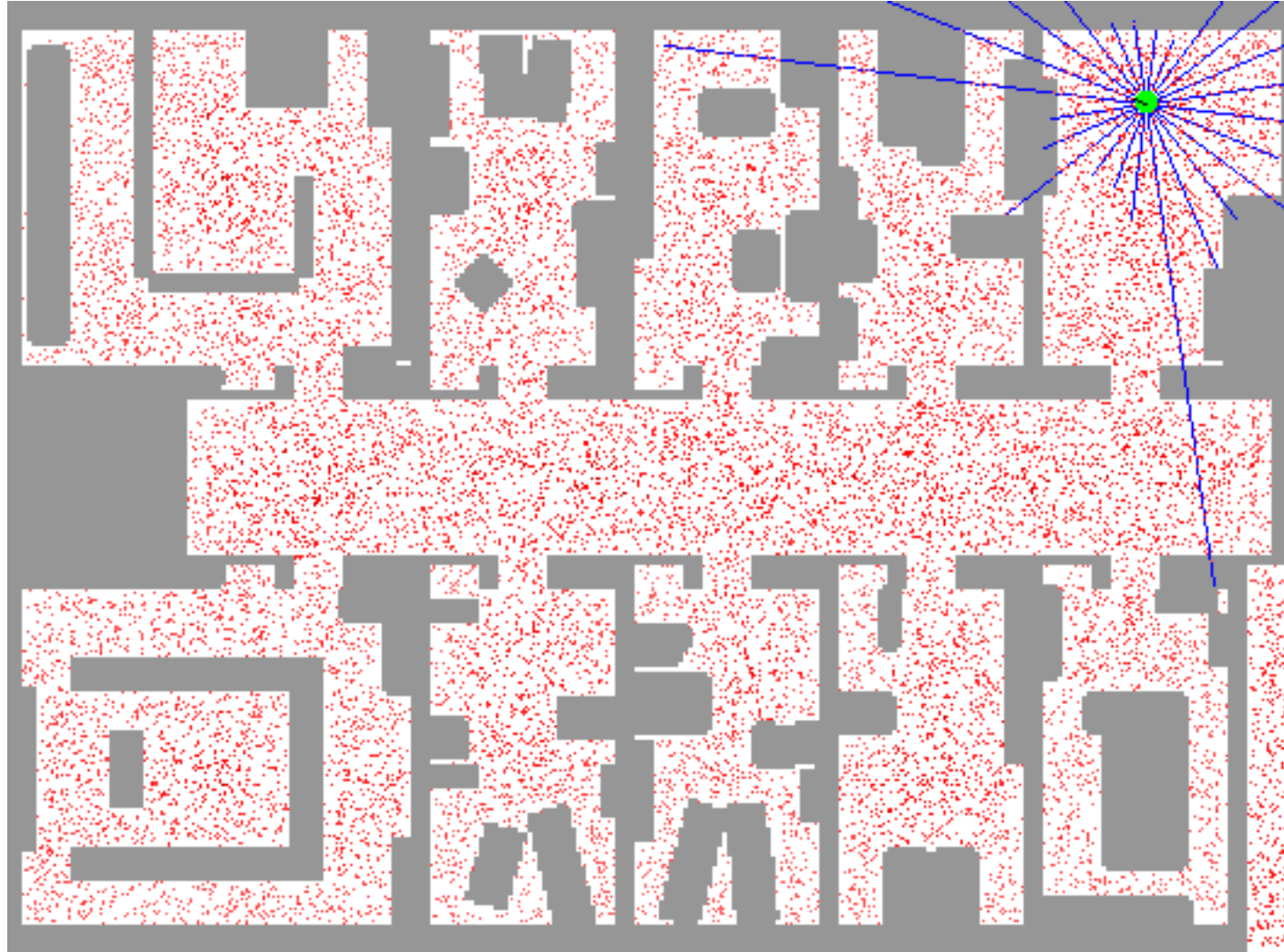
B. Weight Unless the weight is zero, in which case, you'll

C. Resample want to resample from the beginning 😞

D. None of the above

Demo: Pacman Particle Filtering

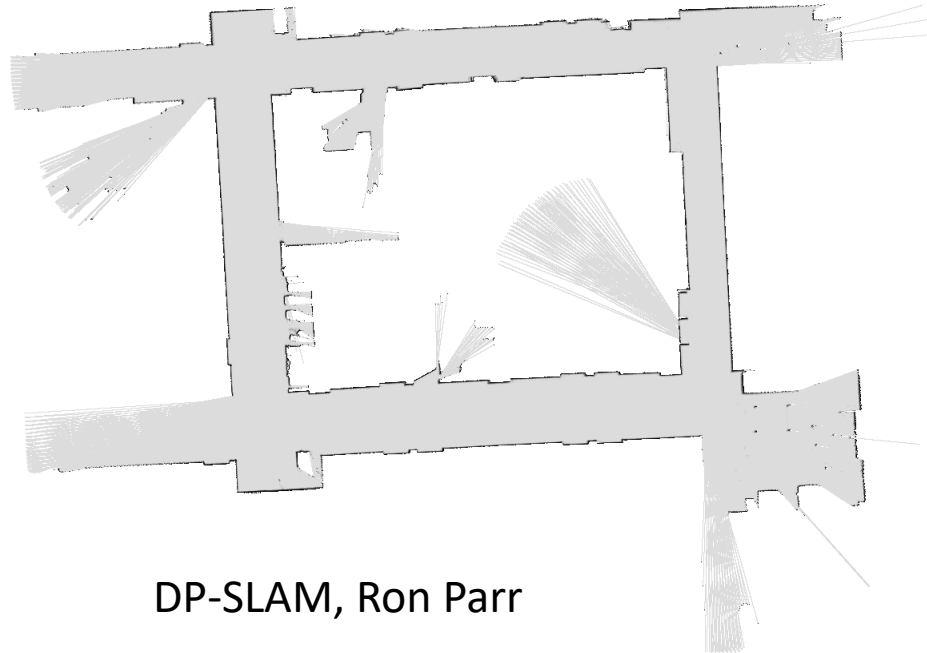
Particle Filter Localization (Laser)



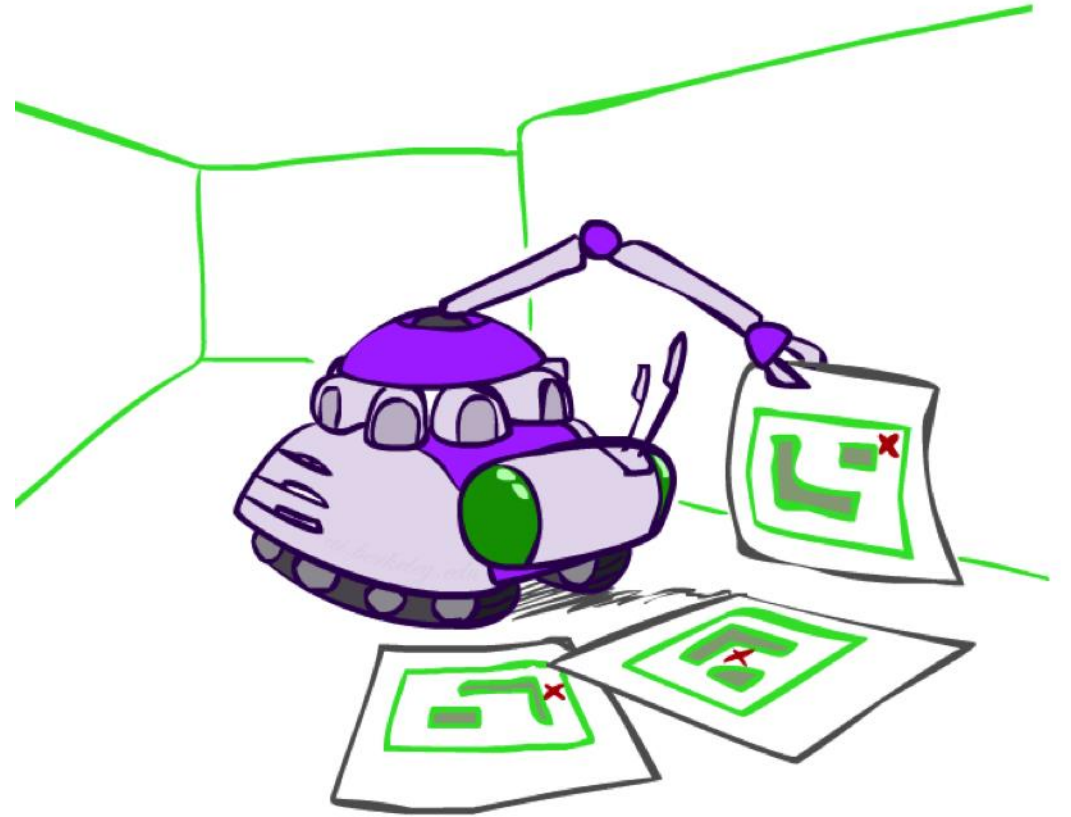
Robot Mapping

SLAM: Simultaneous Localization And Mapping

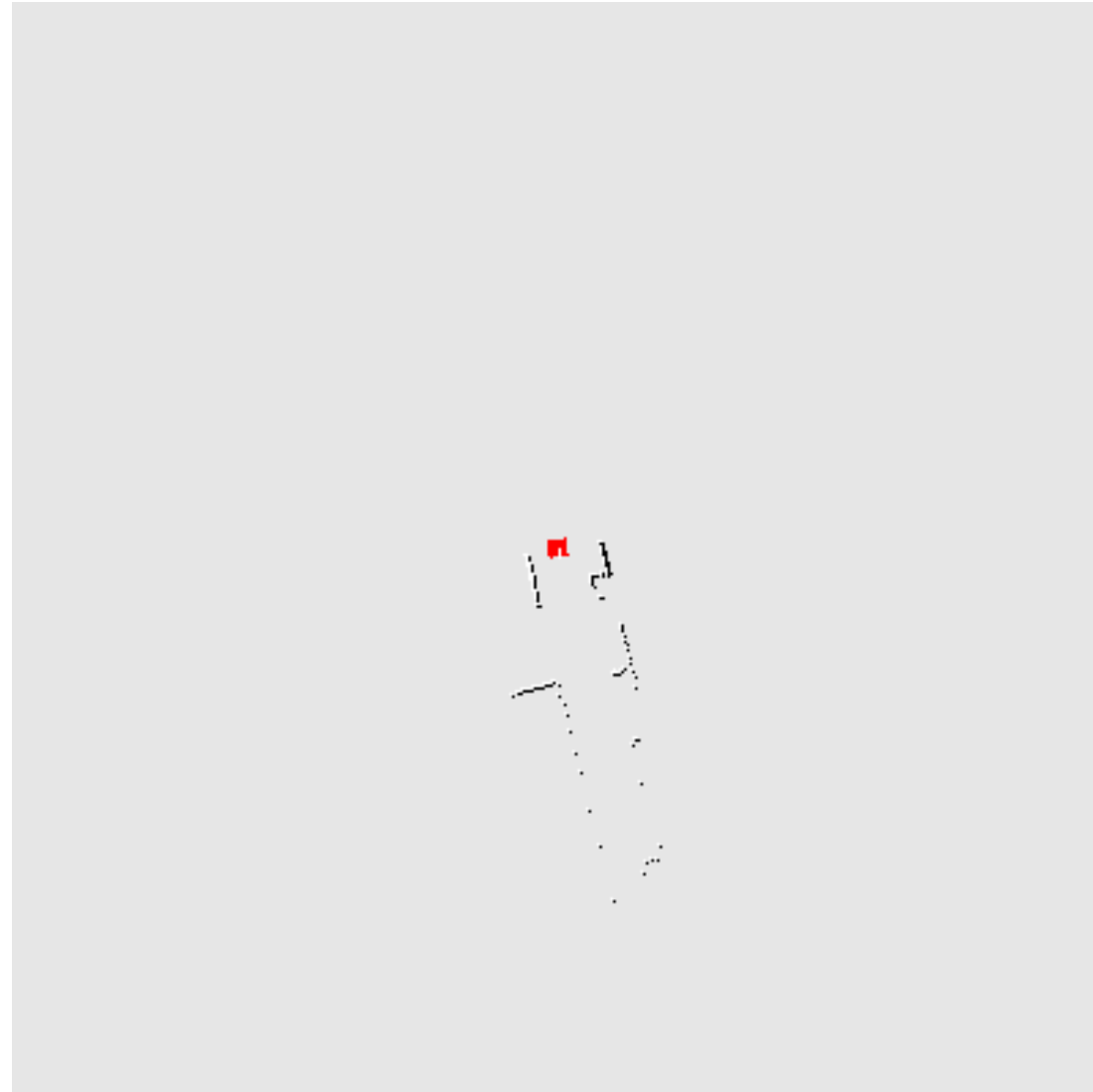
- We do not know the map or our location
- State consists of position AND map!
- Main techniques: Kalman filtering (Gaussian HMMs) and particle methods



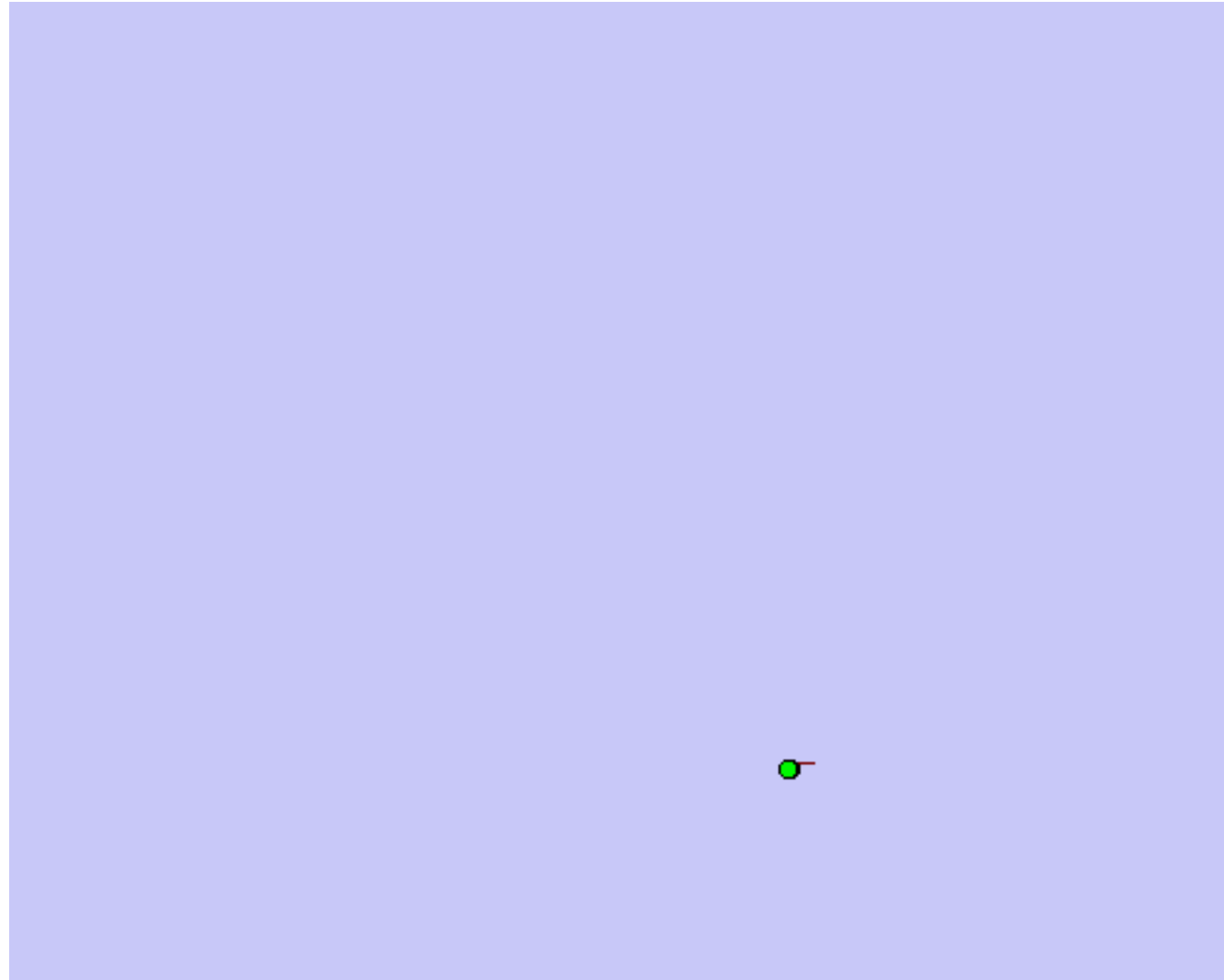
DP-SLAM, Ron Parr



Particle Filter SLAM – Video 1



Particle Filter SLAM – Video 2



SLAM

