Lecture 12

Motion Models with Uncertainty

CS 3630

Logistics Robots

Control Uncertainty

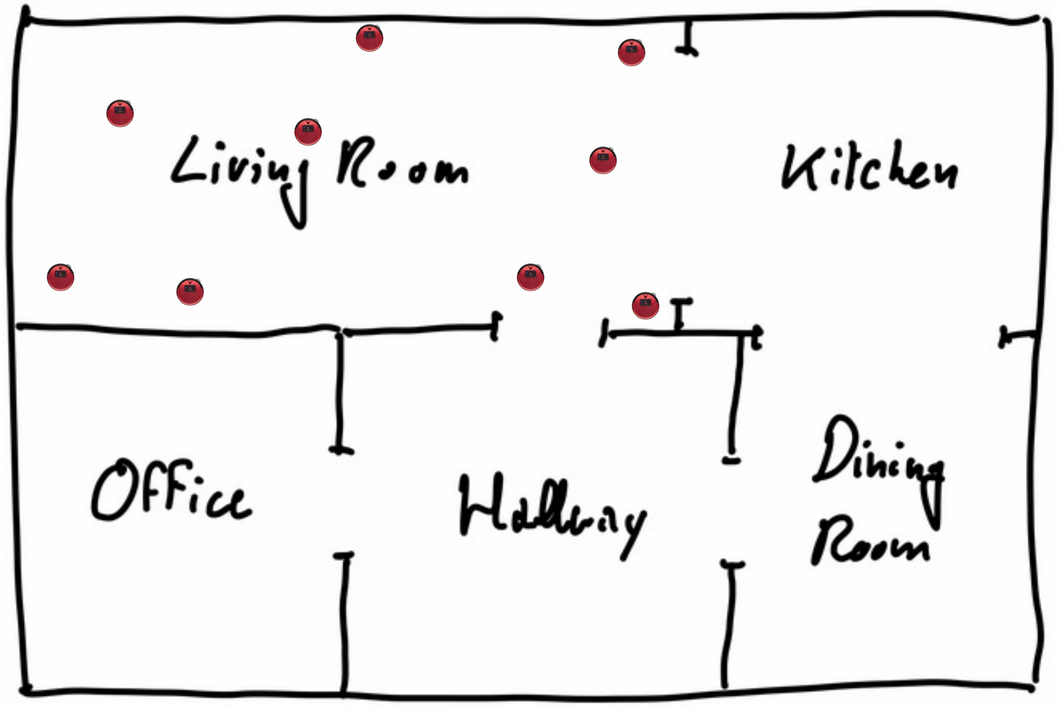
- Now that we have models for wheeled mobile robot kinematics, we can develop a model for uncertainty in the robot’s motion.
- We’ll start with a 1-D robot, and develop the necessary probability theory to model and propagate various types of uncertainty (uniform and Gaussian noise in the motion)
- Once we understand the basics, we’ll extend the results to the 2-D case (motion in the plane).
- We’ll use multivariate Gaussian random variables to model noise/disturbances in the motion model.

Modeling Uncertainty in Actions

- For our vacuum cleaning robot, we considered uncertainty when trying to move from one room to another.
- The uncertainties were given to us in a large table of conditional probabilities.
- There was no clear connection between these conditional probabilities and the geometry of the robot's motion.

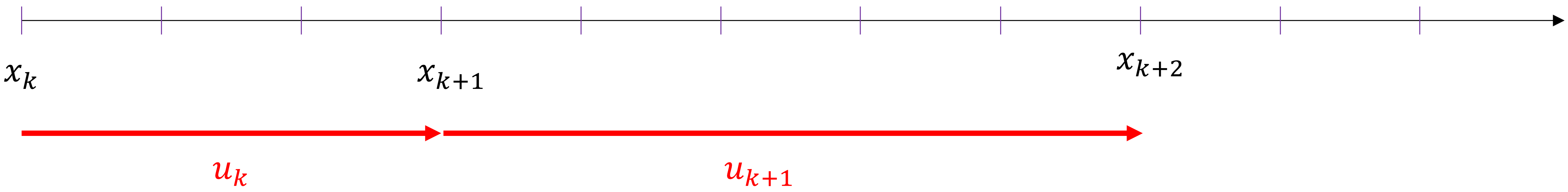
- For ***any*** starting location in in the living room, the probability of arriving to the kitchen by moving right is 0.8.
- For ***any*** starting location in in the living room, the probability of arriving to the hallway by moving down is 0.8.

➢ These probabilities don't seem to be based on the reality of navigating in this environment.



Motion Model – the 1-D Case.

- Wheeled mobile robot that is constrained to move along a single line (e.g., a robot on a track, or a robot following a magnetic guidewire in the floor).
- We will define the control input as $u_k = v\Delta T$, i.e., we command the robot to move along the track with velocity v for an amount of time ΔT.
- In the absence of uncertainty, the state equation is simple: $x_{k+1} = x_k + u_k$
- If we execute a sequence of actions, u_k, u_{k+1} we arrive to $x_{k+2} = x_k + u_k + u_{k+1}$



➢ If there's no uncertainty in the motion model, predicting future states is pretty easy.

Motion Model – the 1-D Case.

- Let's add noise to our motion model:

pronounced "eta" but looks like 'n'

$$x_{k+1} = x_k + u_k + \eta_k$$

- Here, η_k is a noise term, which could be the result of:
  - Variable friction on the floor (e.g., dusty floors are slippery)
  - Variable motor friction
  - Erratic battery discharge/uneven control voltages/currents to the motor
  - Worn brakes (variations in time required to stop moving)

<u>Common assumptions about η_k</u>

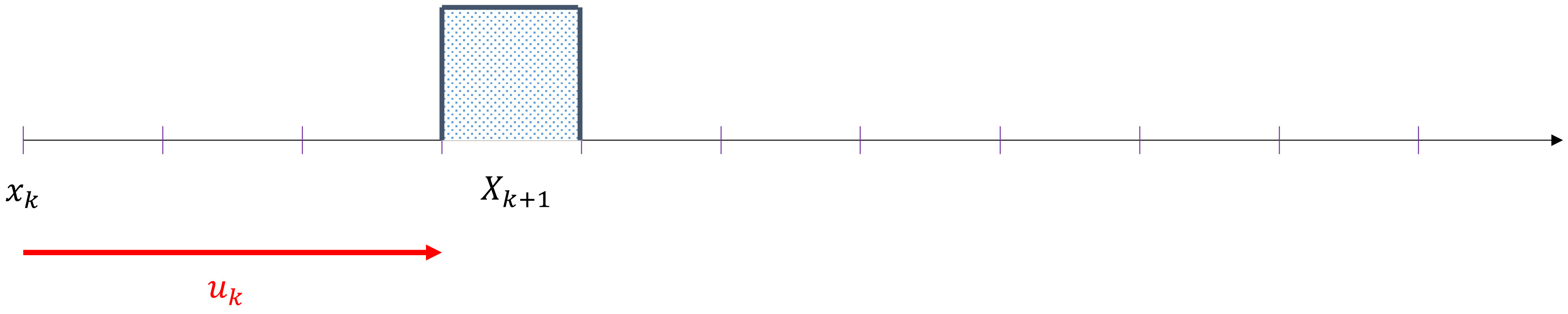
1. The noise η_k is independent of η_j for all $j \neq k$
2. All random disturbances/causes have the same probability distribution.

➢ Random variables that satisfy these two conditions are said to be *independent and identically distributed* (*i.i.d.*)

➢ *We typically assume i.i.d. noise for both motion and sensors, and it's almost always justified.*

Motion Model – the 1-D Case.

- Consider the motion model $x_{k+1} = x_k + u_k + \eta_k$, and let $\eta_k \sim U(0,1)$
- Suppose x_k is known.
- What can we say about x_{k+1}?



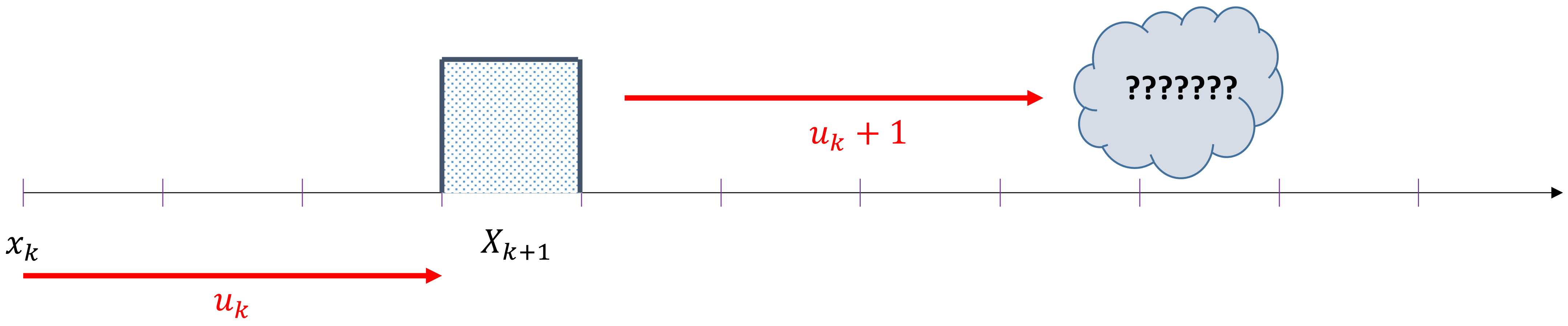
The next state is a random variable with uniform distribution

$$X_{k+1} \sim U(x_k + u_k, x_k + u_k + 1)$$

Motion Model – the 1-D Case.

- That was so simple!!
- What happens after two time steps?

$$x_{k+2} = x_k + u_k + \eta_k + u_{k+1} + \eta_{k+1} = (x_k + u_k + u_{k+1}) + (\eta_k + \eta_{k+1})$$



- The term $x_k + u_k + u_{k+1}$ is completely deterministic (and easy to compute).
- The term $\eta_k + \eta_{k+1}$ is completely stochastic, and somewhat mysterious.
- We need to determine the probability distribution of a sum of random variables.

Sum of Two Random Variables

- Let the random variable $\eta_{S2} = \eta_1 + \eta_2$ be the sum of two random variables η_1, η_2.

- The probability density function for η_{S2} is given by:

$$f_{\eta_{S2}}(\alpha) = \int_{-\infty}^{\infty} f_{\eta_1}(u) f_{\eta_2}(\alpha - u)\, \mathrm{du}$$

- This is a convolution integral (a useful tool for signal processing and control theory), sometimes written as

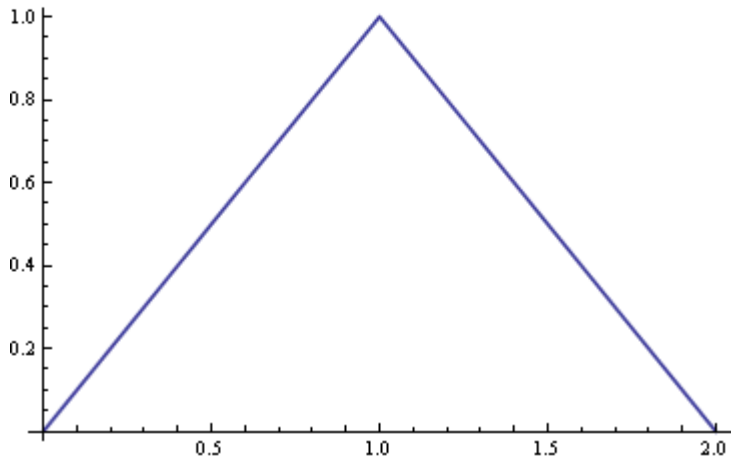
$$f_{\eta_{S2}} = f_{\eta_1} * f_{\eta_2}$$

- This is not a probability theory class, not a signal processing class, not a calculus class, so we won't worry about evaluating these integrals. We'll skip to the payoff…

Sum of Two Random Variables

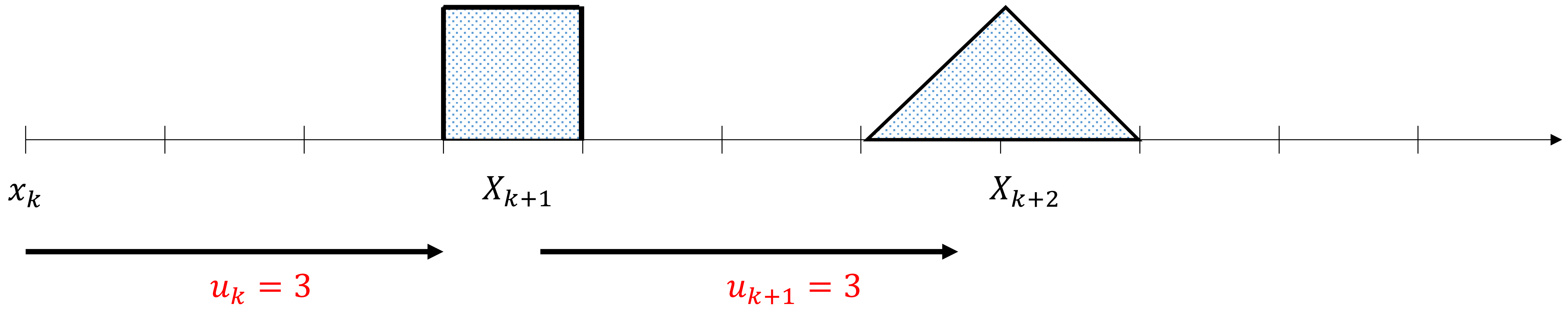
For $\eta_{S2} = \eta_1 + \eta_2$ if $\eta_k \sim U(0,1)$, the probability density function for η_{S2} is:

$$f_{\eta_{S2}}(\alpha) = \begin{cases} \alpha & 0 \leq \alpha \leq 1 \\ 2 - \alpha & 1 \leq \alpha \leq 2 \end{cases}$$



Motion Model – the 1-D Case.

After two time steps, $x_{k+2} = x_k + u_k + \eta_k + u_{k+1} + \eta_{k+1} = (x_k + u_k + u_{k+1}) + (\eta_k + \eta_{k+1})$



- Both of X_{k+1} and X_{k+2} are random variables.
- ***They do not have the same probability distribution!!!***

The Sum of n i.i.d. Uniform Random Variables

Let the random variable $\eta_{Sn} = \eta_1 + \ldots + \eta_n$ be the sum of n random variables.

The pdf for η_{Sn} is called the *Irwin-Hall distribution*.

The Irwin–Hall distribution is the continuous probability distribution for the sum of n independent and identically distributed $U(0, 1)$ random variables:

$$X = \sum_{k=1}^{n} U_k.$$

The probability density function (pdf) is given by

$$f_X(x; n) = \frac{1}{2(n-1)!} \sum_{k=0}^{n} (-1)^k \binom{n}{k} (x-k)^{n-1} \operatorname{sgn}(x-k)$$

where $\operatorname{sgn}(x - k)$ denotes the sign function:

$$\operatorname{sgn}(x-k) = \begin{cases} -1 & x < k \\ 0 & x = k \\ 1 & x > k. \end{cases}$$

[wikipedia]

The Sum of n i.i.d. Uniform Random Variables

This is a nice piece of trivia, but should we really care about this?

YES! As n becomes large, $f_{\eta_{Sn}}$ approaches a Gaussian distribution.

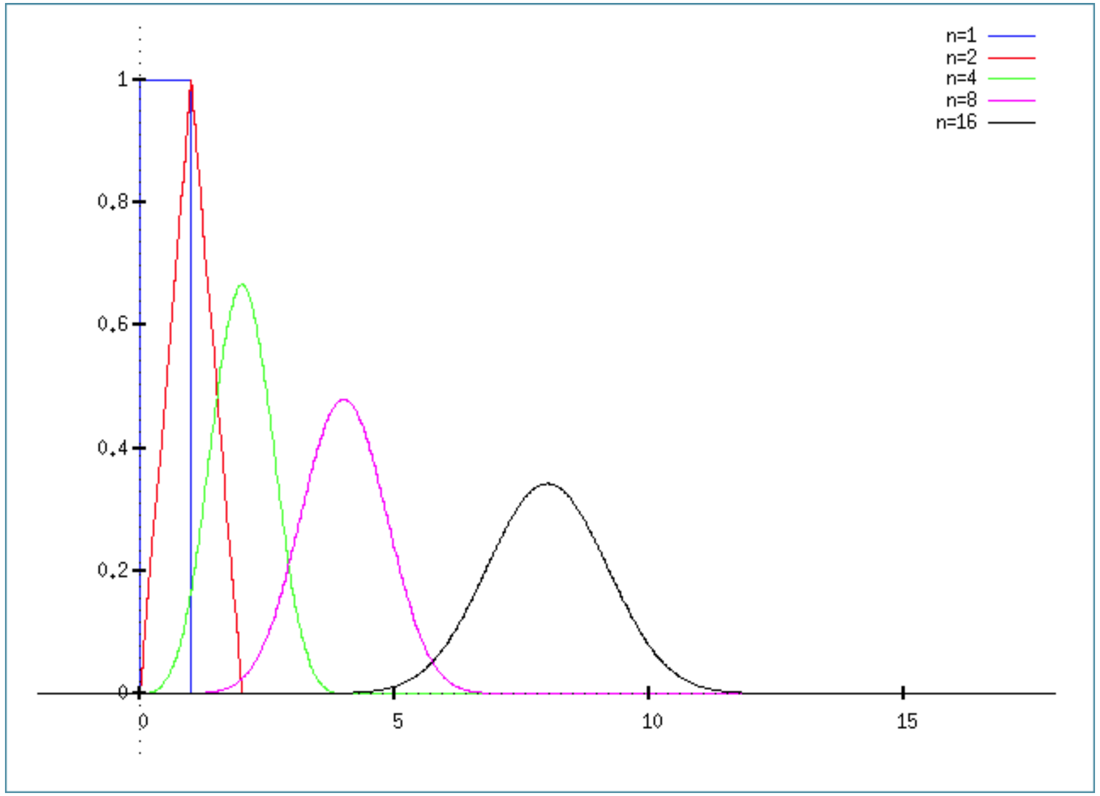
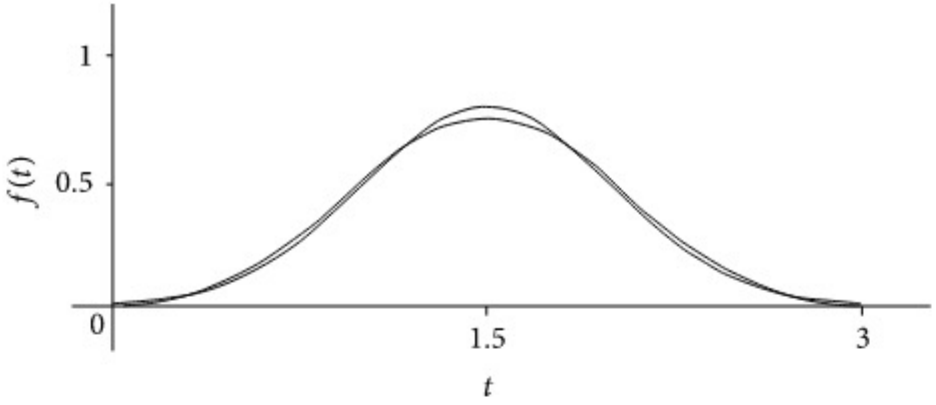


Figure 1

Irwin-Hall distribution with $n = 3$ and the matching normal distribution with mean 3/2 and variance 1/4.



Even for $n = 3$ we can start to see the similarity.

<u>In general, when we add together a bunch of i.i.d. random variables, things start to look Gaussian before long.</u>

Gaussian Noise

- The uniform distribution is great for teaching concepts, but typically it's not a very realistic model for noise in real-world systems.
- The Gaussian distribution is much more common, and much more realistic in most cases.

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- For stochastic noise, we often assume $\mu = 0$ (aka zero-mean Gaussian noise).
- For $\eta_k \sim N(0, \sigma^2)$, we have $E[\sum \eta_k] = \mu = 0$, which yields:

$$E[\sum \eta_k] = \sum E[\eta_k] = 0$$

➢ If we sum a bunch of i.i.d. zero-mean Gaussian random samples, on average the sum will be (approximately) zero.

Gaussian Noise

- One drawback to using Gaussian noise in our motion model is that $P(\alpha \leq \eta_k \leq \beta) > 0$ for any α, β with $\alpha < \beta$.
- Clearly this isn't realistic.
- Is there really a possibility that $10^5 \leq \eta_k \leq 10^6$ miles for our robot?
- Happily, most of the probability is concentrated near the mean:

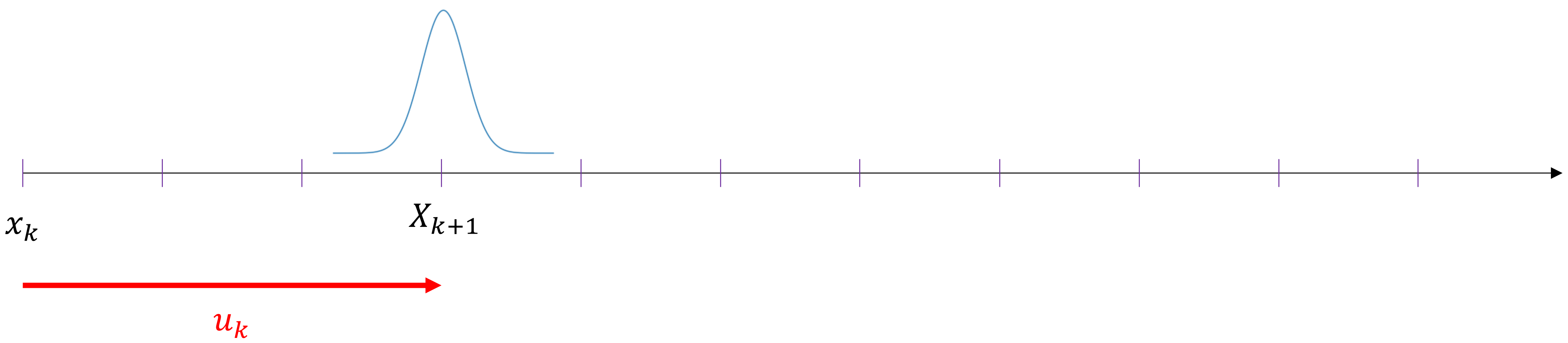
$$P(\mu - 2\sigma \leq \eta_k \leq \mu + 2\sigma) \approx 0.954$$

➢ The tails of the Gaussian don't really hurt that much. Gaussians are a good approximation to reality.

The variance σ^2 is a parameter of the model (either sensor or motion model), and can be estimated, as we've seen in previous lectures.

1D Motion Model with Gaussan Noise

- Consider again the motion model $x_{k+1} = x_k + u_k + \eta_k$, but now let $\eta_k \sim N(0, \sigma^2)$, with all η_k independent.
- Suppose x_k is known.
- What can we say about x_{k+1}?

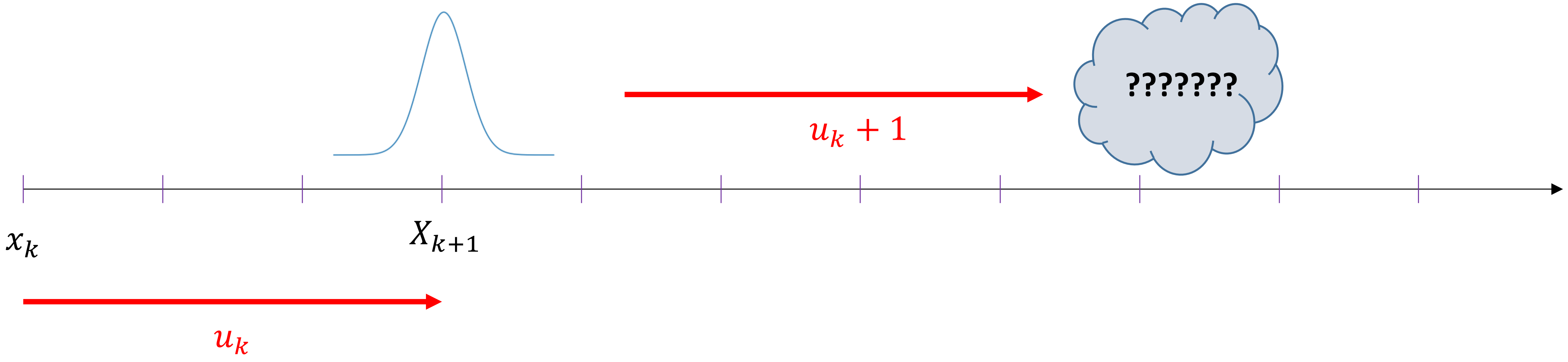


- The next state is a random variable with Gaussian distribution $X_{k+1} \sim N(x_k + u_k, \sigma^2)$.
- $E[X_{k+1}] = x_k + u_k$
- The variance of X_{k+1} is exactly the variance in the noise.

1D Motion Model with Gaussan Noise

- Not too difficult…
- What happens after two time steps?

$$x_{k+2} = x_k + u_k + \eta_k + u_{k+1} + \eta_{k+1} = (x_k + u_k + u_{k+1}) + (\eta_k + \eta_{k+1})$$



- The term $x_k + u_k + u_{k+1}$ is completely deterministic (and easy to compute).
- The term $\eta_k + \eta_{k+1}$ is completely stochastic, and somewhat mysterious.
- We need to determine the probability distribution of a sum of Gaussian random variables.

The Sum of i.i.d. Gaussian Random Variables

- Let the random variable $\eta_{S2} = \eta_1 + \eta_2$, with $\eta_1, \eta_2 \sim N(0, \sigma^2)$.

- The probability density function for η_{S_2} is given by the convolution integral:

$$f_{\eta_{S2}}(\alpha) = \int_{-\infty}^{\infty} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x)^2}{2\sigma^2}} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\alpha - x)^2}{2\sigma^2}} \mathrm{du}$$

- If you work this out, you'll discover that the sum η_{S2} is itself a Gaussian random variable:

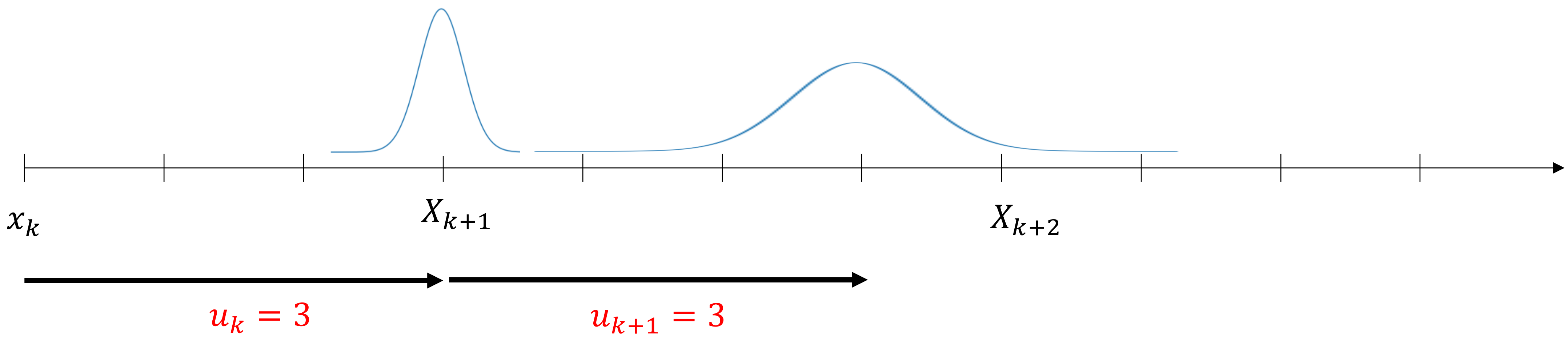
$$\eta_{S2} \sim N(0, 2\sigma^2).$$

- *In general, for $\eta_{S2} = \eta_1 + \eta_2$, with η_1, η_2 independent, and $\eta_1 \sim N(\mu_1, \sigma_1^2)$, $\eta_2 \sim N(\mu_2, \sigma_2^2)$, then the sum is a Gaussian random variable $\boldsymbol{\eta_{S2} \sim N(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)}$.*

1D Motion Model with Gaussan Noise

After two time steps, $x_{k+2} = x_k + u_k + \eta_k + u_{k+1} + \eta_{k+1} = (x_k + u_k + u_{k+1}) + (\eta_k + \eta_{k+1})$

- Both of X_{k+1} and X_{k+2} are Gaussian random variables.
- ***The do not have the same variance!!!***



The Sum of n i.i.d. Gaussian Random Variables

- *We can generalize (using induction) to the case of* $\eta_{Sn} = \eta_1 + \cdots + \eta_n$, *with* η_k *independent, and* $\eta_k \sim N(\mu_k, \sigma_k^2)$:

$$\eta_{Sn} \sim N(\sum \mu_k, \sum \sigma_k^2)$$

- *For the case of i.i.d. zero-mean Gaussian noise, if the initial state is* x_1, *and we execute the action sequence* $u_1, \ldots, u_n$, *the state* X_{n+1} *is a random variable with distribution*

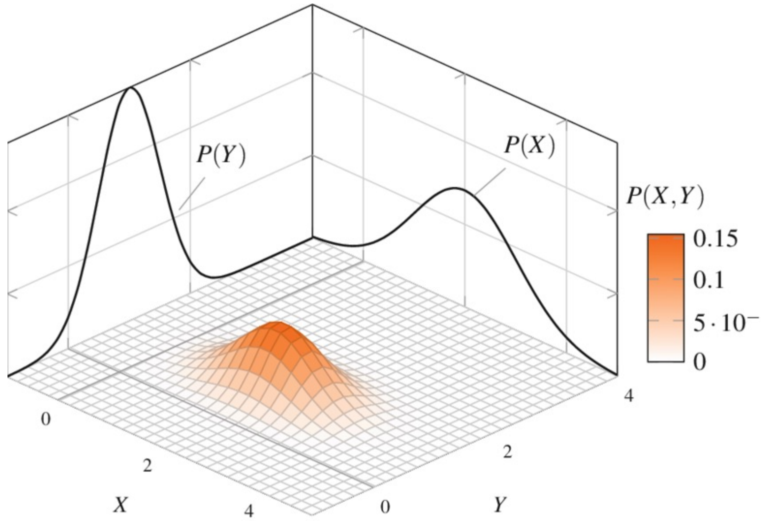
$$X_{n+1} \sim N\left(x_1 + \sum_k u_k\, , n\sigma^2\right)$$

- The good news: $E[X_{n+1}] = x_1 + \sum_k u_k$
- The bad news: $var(X_{n+1}) = n\sigma^2$ ---- ***the variance increases linearly with the number of steps***!
- More good news: we'll be able to use sensing to deal with this increasing uncertainty (not today, though).

Multivariate Gaussians

- Until now, we have considered Gaussian distributions for scalar random variables.
- For univariate Gaussians, η is a scalar, and it appears in the exponent:

$$f_{\mathrm{H}}(\eta) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\eta-\mu)^2}{2\sigma^2}}$$



- Note that H is the uppercase version of Greek letter η.
- For a multivariate Gaussian, the random variable is a vector:

$$\eta = \begin{bmatrix} \eta_x \\ \eta_y \end{bmatrix}$$

- How do we put a vector in an exponent??

Multivariate Gaussians

- Let’s take a look at the exponent in the Gaussian distribution:

  1. The term $|x - \mu|$ is the distance from x to the mean.
  2. The term $(x - \mu)^2$ is the squared distance to the mean.
  3. The term $\sigma^{-2}(x - \mu)^2$ is a ***scaled squared distance to the mean***.

➢ This idea – computing a scaled squared distance to the mean – is the key to extending Gaussians to the multivariate case.

➢ Instead of scalar scaling, we can actually apply scaling along different axes, e.g., we can treat motion in the direction of the x-axis as being more uncertain than motion in the direction of the y-axis.

Multivariate Gaussians

First let's define the relevant vectors:

$$\mu = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \qquad \vec{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$

NOTE:

- For the next few slides, we'll use $\vec{x}$ to denote a vector in $\mathbb{R}^2$.
- There's a possibility of confusion, because most of the time use x to denote a vector $x \in \mathbb{R}^2$.
- For the next derivations, we will use $\mathrm{x}, \mathrm{y}, \in \mathbb{R}$ to denote the scalar coordinates of the point $\vec{x}$.
- Don't lose track of this!

Quadratic Forms

- The squared distance between vectors $\vec{x}$ and μ can be conveniently written as:

$$(\vec{x}-\mu)^T(\vec{x}-\mu) = \begin{bmatrix} x-\mu_x & y-\mu_y \end{bmatrix} \begin{bmatrix} x-\mu_x \\ y-\mu_y \end{bmatrix} = (x-\mu_x)^2 + \left(y-\mu_y\right)^2$$

- Note that this term evaluates to a scalar value!
- The term $(x-\mu_x)^2$ gives the squared distance along the x-axis, and the term $\left(y-\mu_y\right)^2$ gives the squared distance along the y-axis.
- We can scale these simply by multiplying each by a scalar coefficients, say k_x and k_y:

$$k_x(x-\mu_x)^2 + k_y\left(y-\mu_y\right)^2$$

- We can incorporate these scaling values directly into a nice matrix equation:

$$\begin{bmatrix} x-\mu_x & y-\mu_y \end{bmatrix} \begin{bmatrix} k_x & 0 \\ 0 & k_y \end{bmatrix} \begin{bmatrix} x-\mu_x \\ y-\mu_y \end{bmatrix} = k_x(x-\mu_x)^2 + k_y\left(y-\mu_y\right)^2$$

➢ If you understand this, multivariate Gaussians are easy!

Quadratic Forms

- Let's generalize this just a bit

$$\|\vec{x} - \mu\|_{\Sigma^{-1}}^2 = (\vec{x} - \mu)^T \Sigma^{-1} (\vec{x} - \mu) = \begin{bmatrix} x - \mu_x & y - \mu_y \end{bmatrix} \begin{bmatrix} a & b \\ b & c \end{bmatrix} \begin{bmatrix} x - \mu_x \\ y - \mu_y \end{bmatrix}$$

- If you multiply this out (a bit tedious), you'll arrive to the general equation for an ellipse:
    - Center of the ellipse is at μ
    - The matrix Σ^{-1}encodes the major and minor axes (direction and length).
    - Check back to your old geometry books to refresh your memory.

Comments:

- We say that the matrix Σ is positive definite if $\vec{x}^T \Sigma \vec{x} > 0$ for all $\vec{x} \neq 0$.
- If a matrix Σ is positive definite, then Σ^{-1} exists, and $\vec{x}^{\mathrm{T}} \Sigma^{-1} \vec{x} = k$ defines an ellipse, for $k > 0$.

Multivariate Gaussians

- We can use this idea to build an n-dimensional Gaussian distribution:

$$f_{\vec{X}}(\vec{x}) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{-\frac{1}{2}\|\vec{x}-\mu\|^2_{\Sigma^{-1}}} = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{-\frac{1}{2}(\vec{x}-\mu)^T \Sigma^{-1} (\vec{x}-\mu)}$$

- As usual, the action is in the exponent; the constant $\sqrt{(2\pi)^n|\Sigma|}$ is merely to scale the pdf so that $\int f_{\vec{X}}(\vec{x}) d\vec{x} = 1$.

- ***The value of $f_{\vec{X}}(\vec{x})$ decreases exponentially with the square of the scaled distance $\|\vec{x} - \mu\|_{\Sigma^{-1}}$.***

- The matrix Σ is called the covariance matrix. In the two-dimensional case, it is defined as:

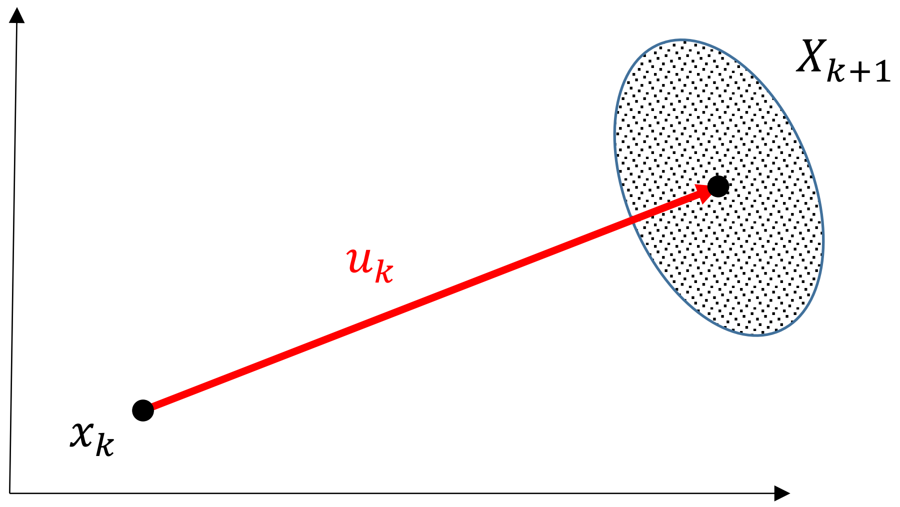
$$\Sigma = \begin{bmatrix} E[(X-\mu_x)^2] & E[(X-\mu_x)(Y-\mu_y)] \\ E[(X-\mu_x)(Y-\mu_y)] & E[(Y-\mu_y)^2] \end{bmatrix}$$

Bivariate Gaussians

For our motion model, we'll use

$$x_{k+1} = x_k + u_k + \eta_k$$

with $x_{k+1}, x_k, u_k, \eta_k \in \mathbb{R}^2$ and $\eta_k \sim N(0, \Sigma)$.



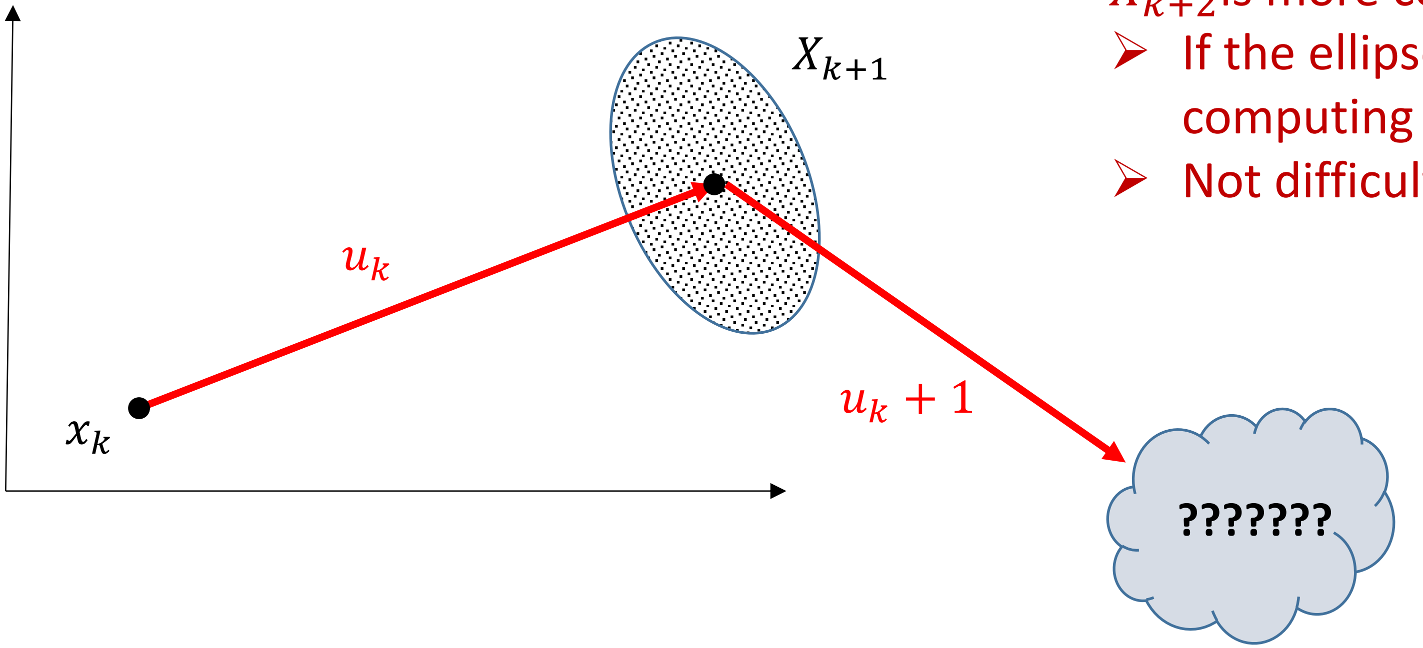
X_{k+1} is a bivariate Gaussian, $X_{k+1} \sim N(x_k + u_k, \Sigma)$

Bivariate Gaussians

What about two stages of execution?

$$x_{k+2} = x_k + u_k + \eta_k + u_{k+1} + \eta_{k+1} = (x_k + u_k + u_{k+1}) + (\eta_k + \eta_{k+1})$$

with $x_{k+2}, x_{k+1}, x_k, u_{k+1}, u_k, \eta_{k+1}, \eta_k \in \mathbb{R}^2$ and $\eta_{k+1}, \eta_k \sim N(0, \Sigma)$.



X_{k+2} is more complicated.

- If the ellipses for the two control inputs are not aligned, computing the covariance for x_{k+2} can be a bit messy.
- Not difficult if Σ is a diagonal matrix.

Multiple Time Steps

Conceptually, there's nothing new here.

➢ *Each time step adds a bit of Gaussian noise to the control input, introducing uncertainty that increases with the number of steps.*

Mathematically, things become a bit more difficult. We won't go into the details here.

Instead, we'll develop two numerical methods to propagate uncertainty, and both of these will be applicable to the case of Gaussian noise in our motion model:

- **Markov Localization**: Divide the world into a grid, and keep track of the probability mass that arrives to each grid cell as the robot moves.
- **Monte Carlo Localization**: Simulate lots of robots (generate samples from the noise distributions to simulate the motion model). The distribution of the simulated robots give insight to the probability distribution associated to the robot's location.

Next Time…

- Sensor models
- Markov Localization
- Monte Carlo Localization