Introduction to Kalman-Filter, SDE, and LFM

Shikai Fang
2021/10 for group present

Material form PRML ch13, books/paper/poster from Simo Sarkka/Aron Solin

Skeleton

SDE
(stochastic differential equation)

Brown Motion

Ito integration

State Space
Gaussian Process

Latent Force Model
(LFM)

Markov Models

State Space Model &
Linear Dynamic Sys(LDS)

Kalman Filter &
Smoother

Non-linear & Cont time
Kalman

Skeleton

We start here!

Markov Models

State Space Model & Linear Dynamic Sys(LDS)

Kalman Filter & Smoother

Non-linear & Cont time Kalman

SDE (stochastic differential equation)

Brown Motion

Ito integration

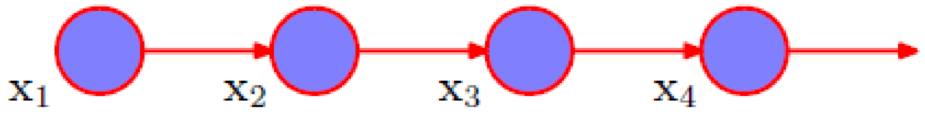
State Space Gaussian Process

Latent Force Model (LFM)

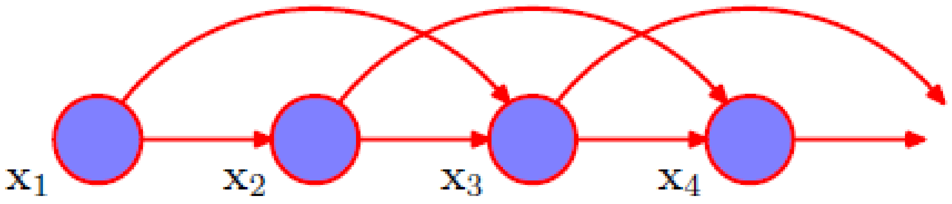
Markov Models :

model the relation of seq data

$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \prod_{n=1}^{N} p(\mathbf{x}_n | \mathbf{x}_1, \ldots, \mathbf{x}_{n-1}).$$



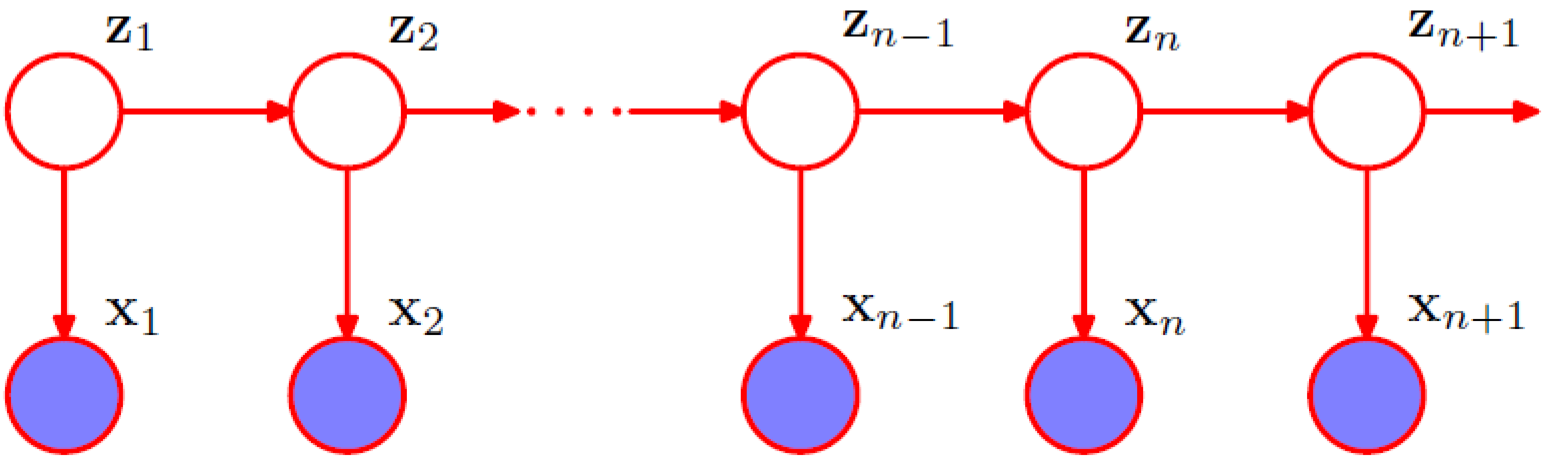
$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N) = p(\mathbf{x}_1) \prod_{n=2}^{N} p(\mathbf{x}_n | \mathbf{x}_{n-1}).$$



$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N) = p(\mathbf{x}_1) p(\mathbf{x}_2 | \mathbf{x}_1) \prod_{n=3}^{N} p(\mathbf{x}_n | \mathbf{x}_{n-1}, \mathbf{x}_{n-2}).$$

Markov chains

- First order
- Second order

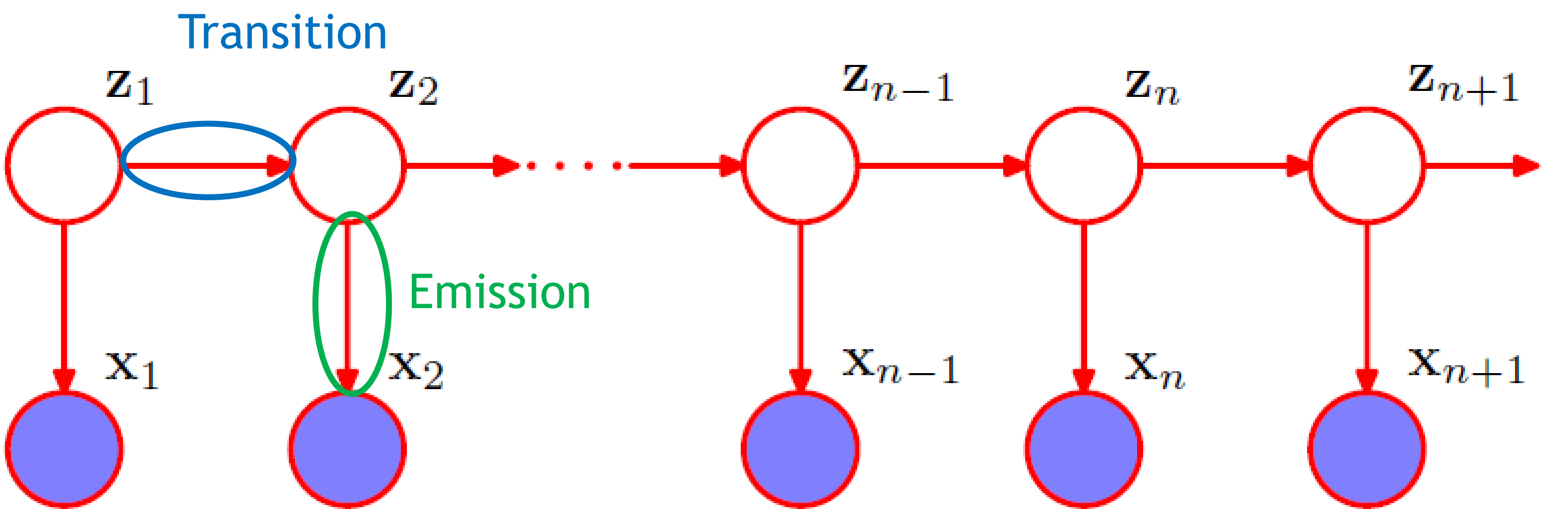


$$\mathbf{z}_{n+1} \perp\!\!\!\perp \mathbf{z}_{n-1} \mid \mathbf{z}_n.$$

$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N, \mathbf{z}_1, \ldots, \mathbf{z}_N) = p(\mathbf{z}_1) \left[\prod_{n=2}^{N} p(\mathbf{z}_n | \mathbf{z}_{n-1}) \right] \prod_{n=1}^{N} p(\mathbf{x}_n | \mathbf{z}_n).$$

State Space Model :

Markov
+ latent factors



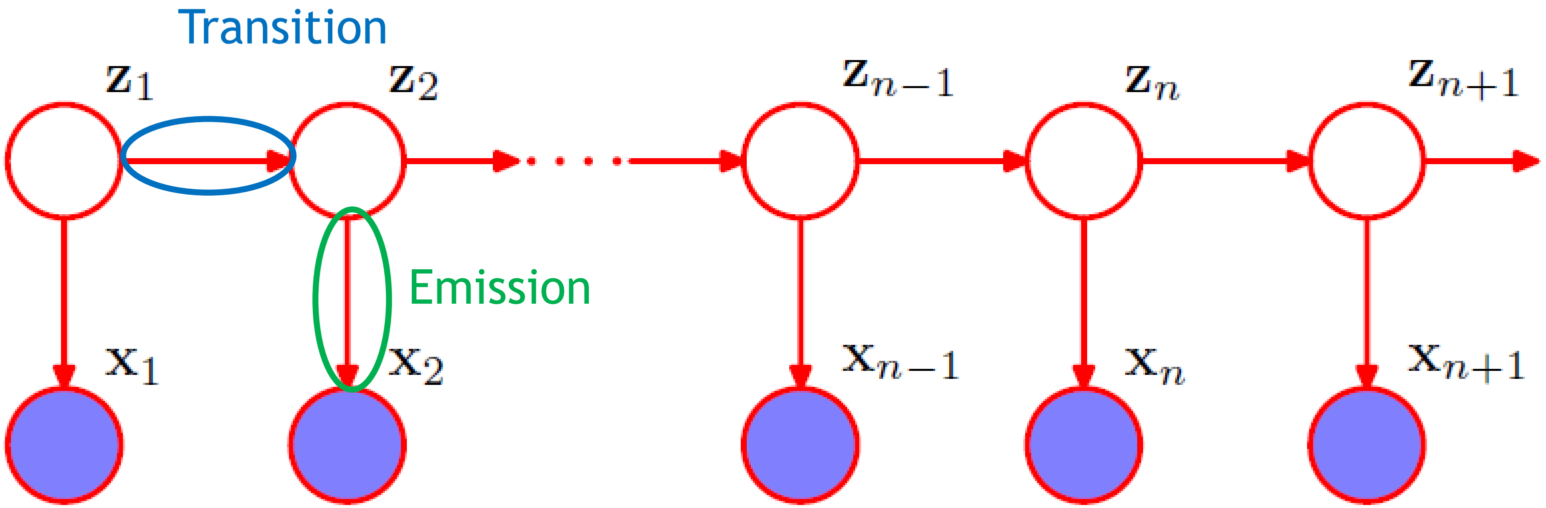
If Z is discrete – Hidden Markov Model(HMM)

If Z, X are Gaussian, with linear transition and emission

- **Linear Dynamic System(LDA)**

State Space Model：

HMM & LDA



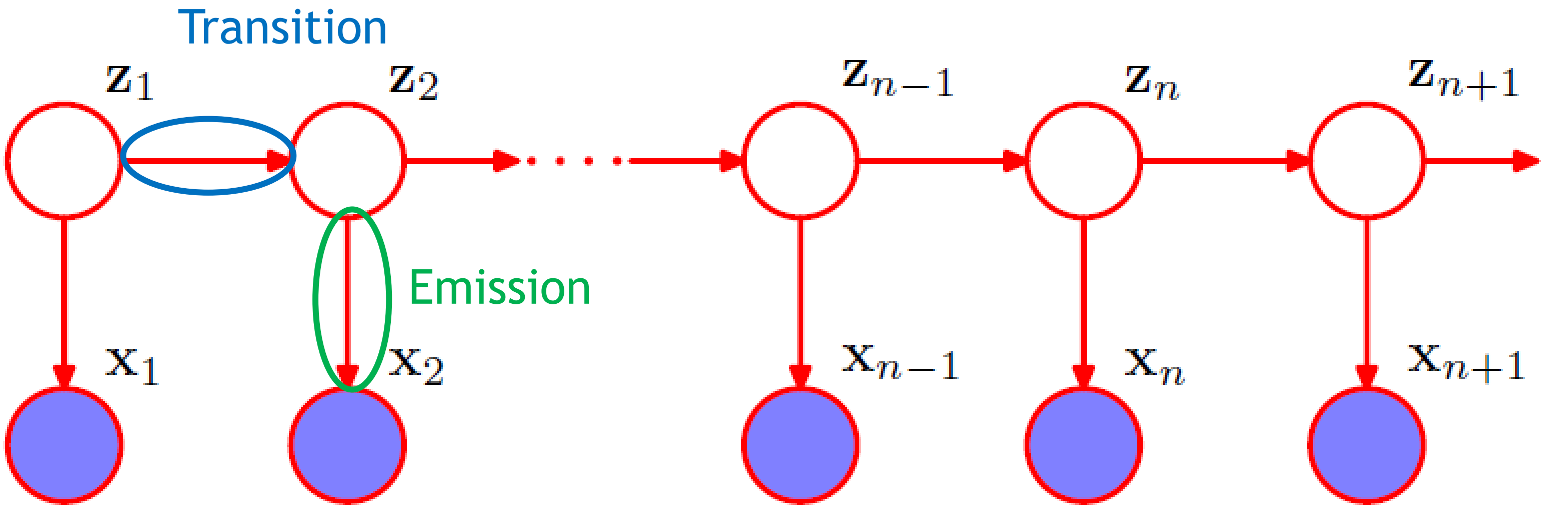
$$p(\mathbf{z}_1) = \mathcal{N}(\mathbf{z}_1 | \boldsymbol{\mu}_0, \mathbf{V}_0)$$

Transition $p(\mathbf{z}_n|\mathbf{z}_{n-1}) = \mathcal{N}(\mathbf{z}_n|\mathbf{A}\mathbf{z}_{n-1}, \boldsymbol{\Gamma})$

Emission $p(\mathbf{x}_n|\mathbf{z}_n) = \mathcal{N}(\mathbf{x}_n|\mathbf{C}\mathbf{z}_n, \boldsymbol{\Sigma}).$

LDS:

Linear Transition & Emission of Gaussian



Alternative form

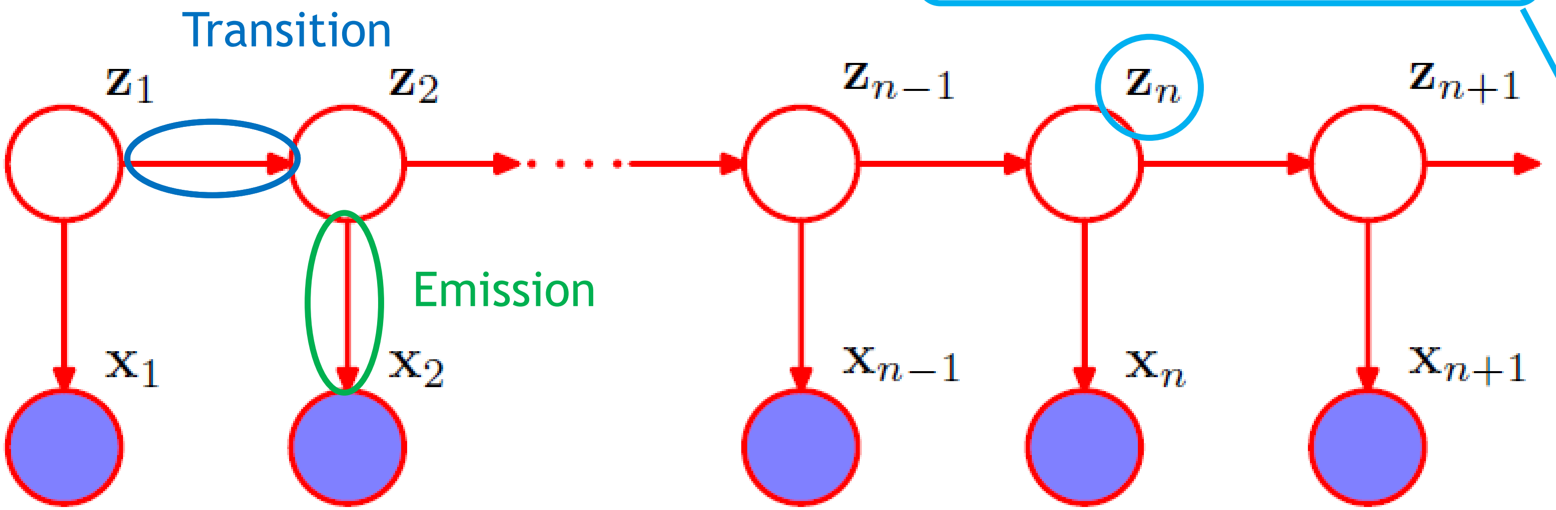
$$
\begin{aligned}
&\text{Transition} \quad & \mathbf{z}_n &= \mathbf{A}\mathbf{z}_{n-1} + \mathbf{w}_n \\
&\text{Emission} \quad & \mathbf{x}_n &= \mathbf{C}\mathbf{z}_n + \mathbf{v}_n \\
& & \mathbf{z}_1 &= \boldsymbol{\mu}_0 + \mathbf{u}
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{w} &\sim \mathcal{N}(\mathbf{w}|\mathbf{0}, \boldsymbol{\Gamma}) \\
\mathbf{v} &\sim \mathcal{N}(\mathbf{v}|\mathbf{0}, \boldsymbol{\Sigma}) \\
\mathbf{u} &\sim \mathcal{N}(\mathbf{u}|\mathbf{0}, \mathbf{V}_0).
\end{aligned}
$$

LDS:

Linear Transition & Emission of Gaussian

$$\widehat{\alpha}(\mathbf{z}_n) = \mathcal{N}(\mathbf{z}_n|\boldsymbol{\mu}_n, \mathbf{V}_n).$$



$$c_n\widehat{\alpha}(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n)\int \widehat{\alpha}(\mathbf{z}_{n-1})p(\mathbf{z}_n|\mathbf{z}_{n-1})\,\mathrm{d}\mathbf{z}_{n-1}.$$

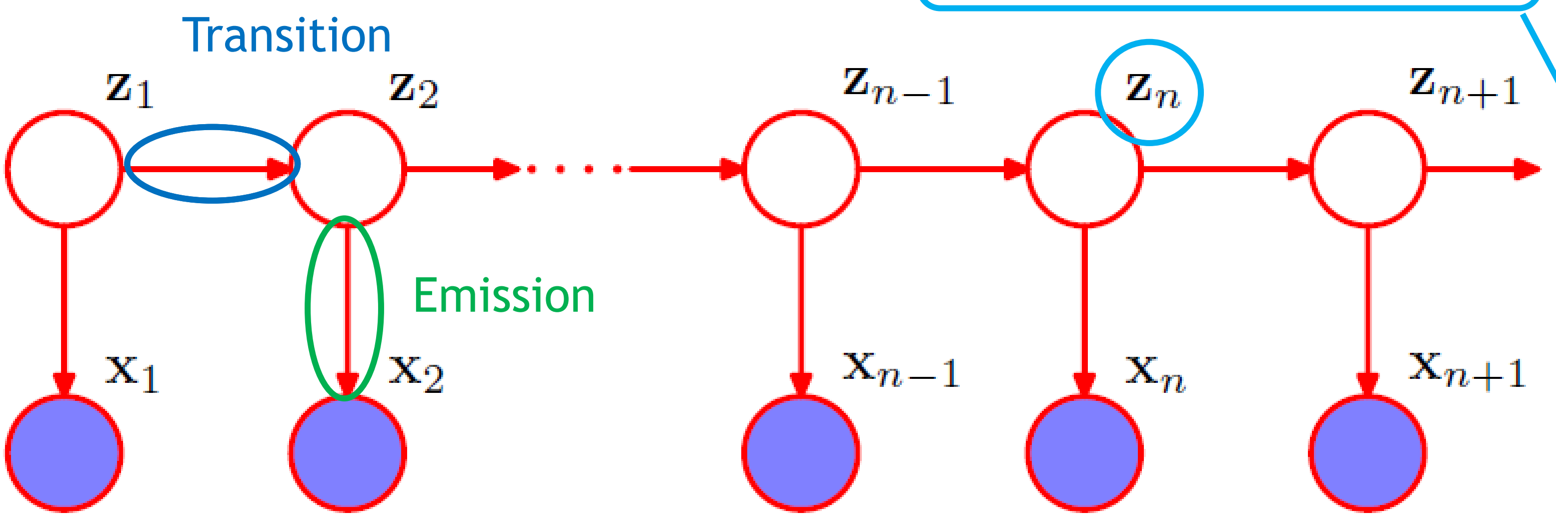
$$c_n\mathcal{N}(\mathbf{z}_n|\boldsymbol{\mu}_n, \mathbf{V}_n) = \mathcal{N}(\mathbf{x}_n|\mathbf{C}\mathbf{z}_n, \boldsymbol{\Sigma})$$
$$\int \mathcal{N}(\mathbf{z}_n|\mathbf{A}\mathbf{z}_{n-1}, \boldsymbol{\Gamma})\mathcal{N}(\mathbf{z}_{n-1}|\boldsymbol{\mu}_{n-1}, \mathbf{V}_{n-1})\,\mathrm{d}\mathbf{z}_{n-1}.$$

Goal of "Prediction"

Compute posterior:

P(Z_n|Xn,X_(n-1)..X_1

$$\widehat{\alpha}(\mathbf{z}_n) = \mathcal{N}(\mathbf{z}_n|\boldsymbol{\mu}_n, \mathbf{V}_n).$$



$$c_n\widehat{\alpha}(\mathbf{z}_n) = p(\mathbf{x}_n|\mathbf{z}_n)\int \widehat{\alpha}(\mathbf{z}_{n-1})p(\mathbf{z}_n|\mathbf{z}_{n-1})\,\mathrm{d}\mathbf{z}_{n-1}.$$

$$c_n\mathcal{N}(\mathbf{z}_n|\boldsymbol{\mu}_n, \mathbf{V}_n) = \boxed{\mathcal{N}(\mathbf{x}_n|\mathbf{C}\mathbf{z}_n, \boldsymbol{\Sigma})}$$
$$\boxed{\int \mathcal{N}(\mathbf{z}_n|\mathbf{A}\mathbf{z}_{n-1}, \boldsymbol{\Gamma})\mathcal{N}(\mathbf{z}_{n-1}|\boldsymbol{\mu}_{n-1}, \mathbf{V}_{n-1})\,\mathrm{d}\mathbf{z}_{n-1}.}$$

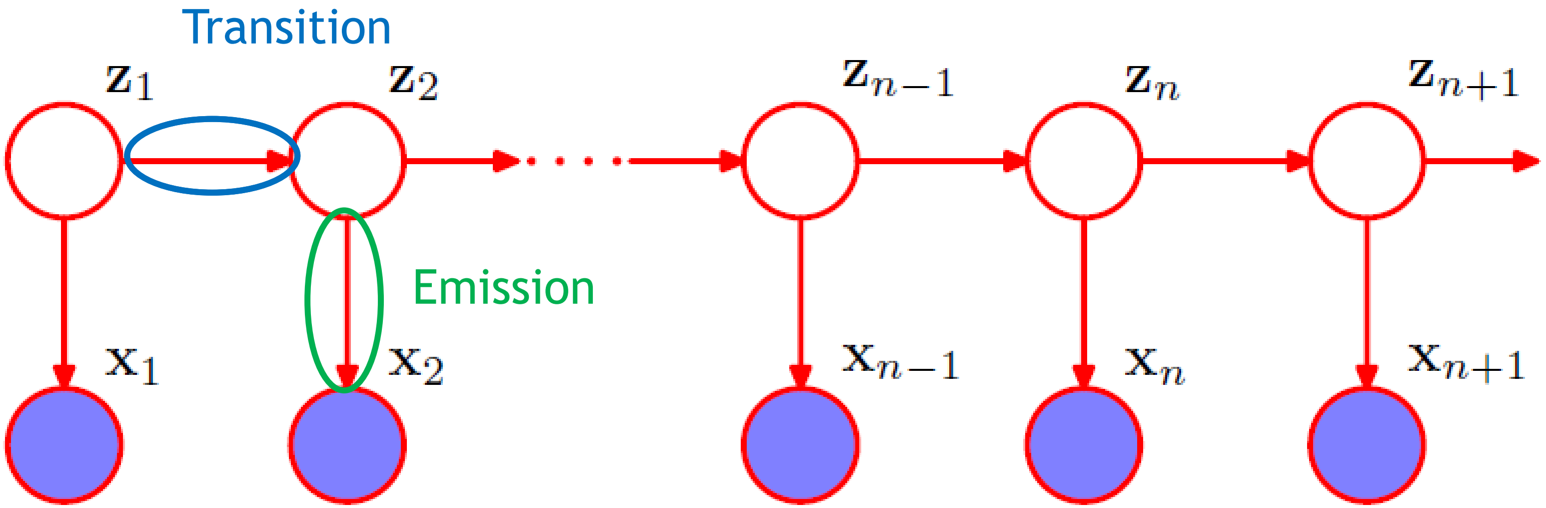
Everything is Gaussian here..!

Goal of "Prediction"

Compute posterior:

P(Z_n|Xn,X_(n-1)..X_1

$$\widehat{\alpha}(\mathbf{z}_n) = \mathcal{N}(\mathbf{z}_n|\boldsymbol{\mu}_n, \mathbf{V}_n).$$



$$\begin{aligned}
\boldsymbol{\mu}_n &= \mathbf{A}\boldsymbol{\mu}_{n-1} + \mathbf{K}_n(\mathbf{x}_n - \mathbf{C}\mathbf{A}\boldsymbol{\mu}_{n-1}) \\
\mathbf{V}_n &= (\mathbf{I} - \mathbf{K}_n\mathbf{C})\mathbf{P}_{n-1} \\
c_n &= \mathcal{N}(\mathbf{x}_n|\mathbf{C}\mathbf{A}\boldsymbol{\mu}_{n-1}, \mathbf{C}\mathbf{P}_{n-1}\mathbf{C}^{\mathrm{T}} + \boldsymbol{\Sigma}).
\end{aligned}$$

$$\boxed{\mathbf{K}_n} = \mathbf{P}_{n-1}\mathbf{C}^{\mathrm{T}}\left(\mathbf{C}\mathbf{P}_{n-1}\mathbf{C}^{\mathrm{T}} + \boldsymbol{\Sigma}\right)^{-1}.$$

Kalman gain matrix

Goal of "Prediction"

Compute posterior:

P(Z_n|Xn,X_(n-1)..X_1

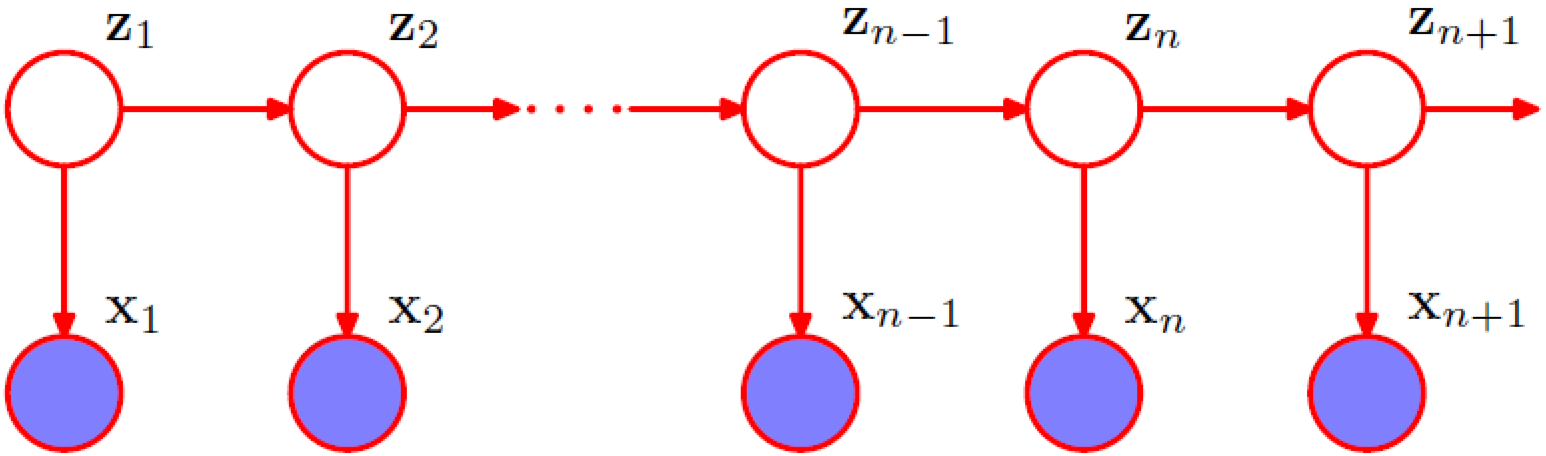
Close-form + Recursive

update posterior

---We got Kalman Filter

Forward passing of msg

$$\gamma(\mathbf{z}_n) = \boxed{\widehat{\alpha}(\mathbf{z}_n)}\boxed{\widehat{\beta}(\mathbf{z}_n)} = \mathcal{N}(\mathbf{z}_n|\widehat{\boldsymbol{\mu}}_n, \widehat{\mathbf{V}}_n).$$



$$c_{n+1}\widehat{\beta}(\mathbf{z}_n) = \int \widehat{\beta}(\mathbf{z}_{n+1})p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1})p(\mathbf{z}_{n+1}|\mathbf{z}_n)\,\mathrm{d}\mathbf{z}_{n+1}.$$

$$\begin{aligned} \widehat{\boldsymbol{\mu}}_n &= \boldsymbol{\mu}_n + \mathbf{J}_n\left(\widehat{\boldsymbol{\mu}}_{n+1} - \mathbf{A}\boldsymbol{\mu}_N\right) \\ \widehat{\mathbf{V}}_n &= \mathbf{V}_n + \mathbf{J}_n\left(\widehat{\mathbf{V}}_{n+1} - \mathbf{P}_n\right)\mathbf{J}_n^{\mathrm{T}} \end{aligned}$$

where we have defined

$$\mathbf{J}_n = \mathbf{V}_n\mathbf{A}^{\mathrm{T}}\left(\mathbf{P}_n\right)^{-1}$$

How about "backward"

Compute posterior:

P(Z_n|X1,X2..X_N)

Similar things:

Close-form + Recursive update posterior

---Kalman Smoother

Skeleton

Now we are here

SDE
(stochastic differential equation)

Brown Motion

Ito integration

Done!

Markov Models

State Space Model &
Linear Dynamic Sys(LDS)

Kalman Filter &
Smoother

State Space
Gaussian Process

Latent Force Model
(LFM)

Non-linear & Cont-time
Kalman

What is a stochastic differential equation (SDE)?

- At first, we have an ordinary differential equation (ODE):

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t).$$

- Then we add white noise to the right hand side:

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t) + \mathbf{w}(t).$$

- Generalize a bit by adding a multiplier matrix on the right:

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t) + \mathbf{L}(\mathbf{x}, t)\,\mathbf{w}(t).$$

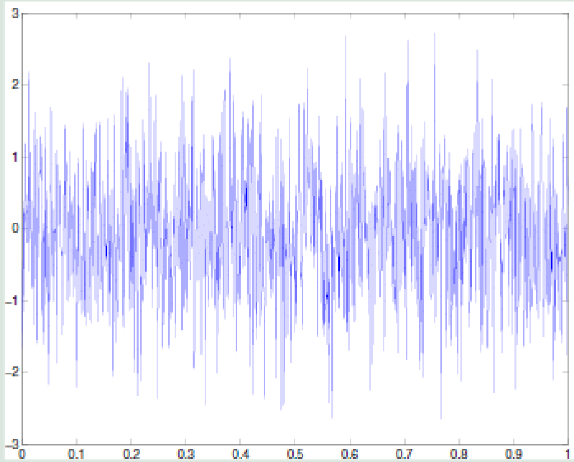
- Now we have a stochastic differential equation (SDE).
- $\mathbf{f}(\mathbf{x}, t)$ is the drift function and $\mathbf{L}(\mathbf{x}, t)$ is the dispersion matrix.

White noise

White noise

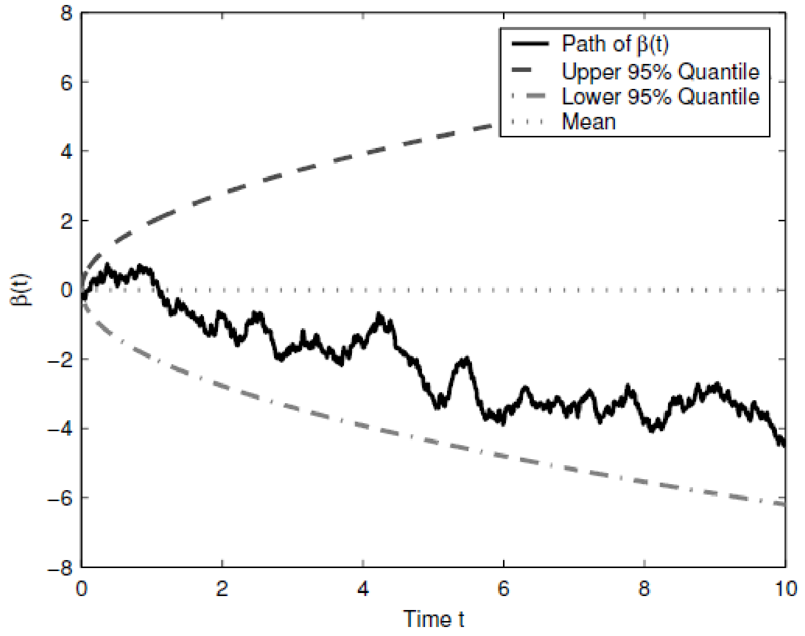
1. $\mathbf{w}(t_1)$ and $\mathbf{w}(t_2)$ are independent if $t_1 \neq t_2$.
2. $t \mapsto \mathbf{w}(t)$ is a Gaussian process with the mean and covariance:

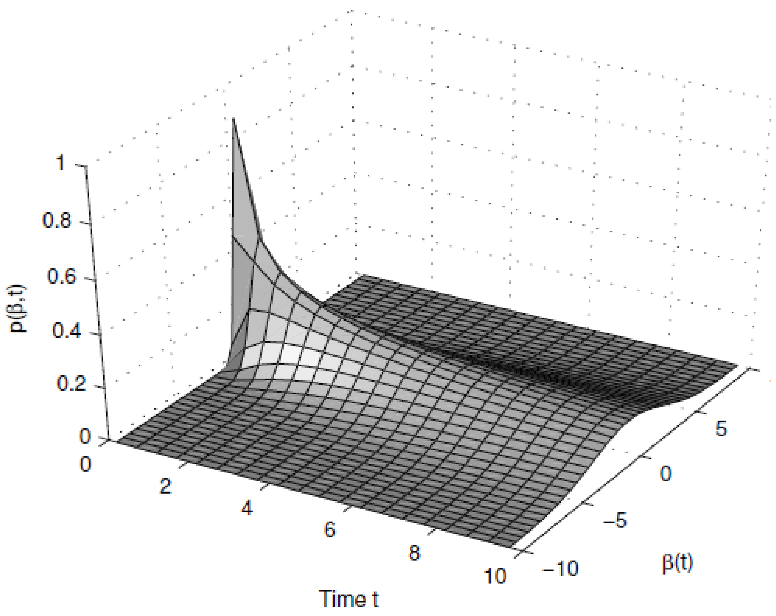
$$\mathrm{E}[\mathbf{w}(t)] = \mathbf{0}$$
$$\mathrm{E}[\mathbf{w}(t)\,\mathbf{w}^{\mathsf{T}}(s)] = \delta(t-s)\,\mathbf{Q}.$$



- $\mathbf{Q}$ is the spectral density of the process.
- The sample path $t \mapsto \mathbf{w}(t)$ is discontinuous almost everywhere.
- White noise is unbounded and it takes arbitrarily large positive and negative values at any finite interval.

What does a solution of SDE look like?





$$B_{t+\Delta t} = B_t + N(0, \Delta t)$$

$$B_t - B_s \sim N(0, t - s)$$

- *Left*: Path of a Brownian motion which is solution to stochastic differential equation

$$\frac{dx}{dt} = w(t)$$

- *Right*: Evolution of probability density of Brownian motion.

Mathematical Meaning and Notation of SDE [1/2]

- What is Itô stochastic calculus then?
- Let's take a look at the scalar equation

$$\frac{dx(t)}{dt} = f(x(t)) + L(x(t))\, w(t).$$

- Integrating from s to t gives

$$x(t) - x(s) = \int_s^t f(x(t))\, dt + \boxed{\int_s^t L(x(t))\, w(t)\, dt.}$$

- White noise is unbounded and discontinuous almost everywhere – the second integral cannot be defined as Riemann, Stieltjes, or Lebesgue integral!

Mathematical Meaning and Notation of SDE [2/2]

- Itô's idea: define $d\beta(t) = w(t)\,dt$, where $\beta(t)$ is the Wiener/Brownian process:

$$x(t) - x(s) = \int_s^t f(x(t))\,dt + \int_s^t L(x(t))\,d\beta(t).$$

- Commonly used shorthand notation for the above:

$$dx(t) = f(x(t))\,dt + L(x(t))\,d\beta(t).$$

- In stochastics literature you see this in form:

$$dX_t(\omega) = f(X_t(\omega))\,dt + L(X_t(\omega))\,d\beta_t(\omega).$$

Itô Integral

$$E(\int_T g(X_s, s)dB_s) = 0$$

- The Itô integral is defined as the limit of the expression

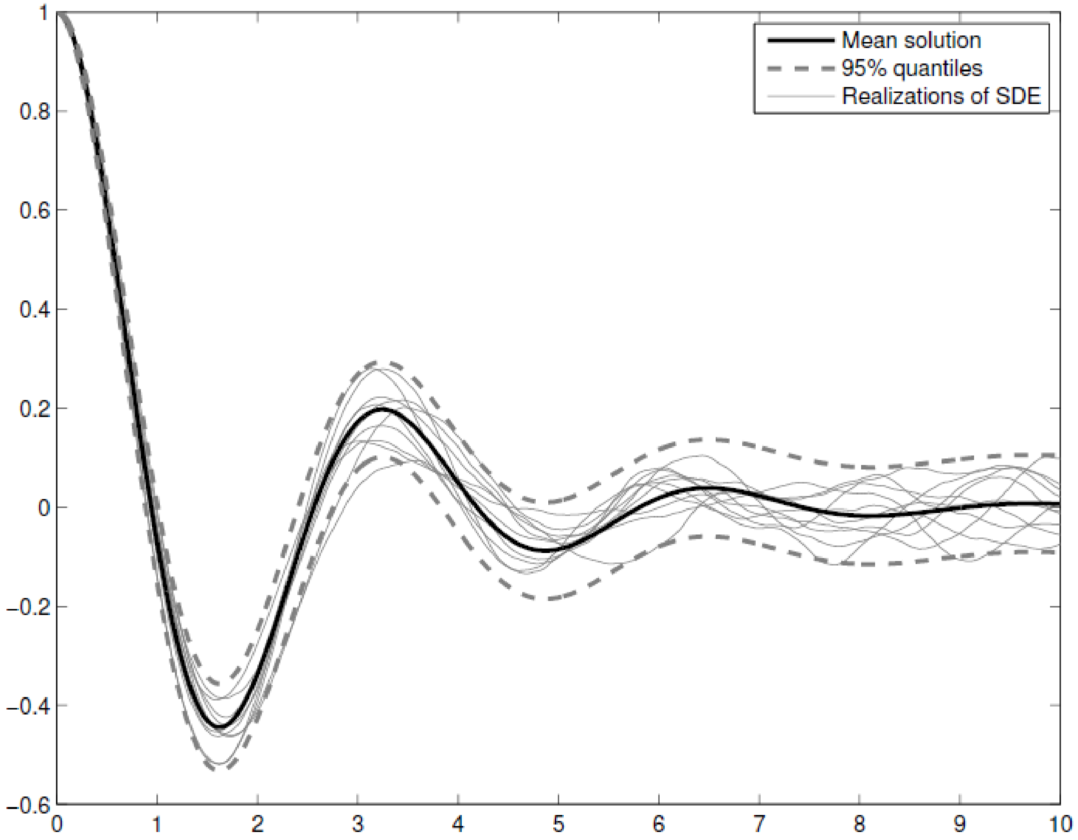
$$\begin{aligned}\int_s^t L(x(t))\, d\beta(t) = & \; L(x(t_1))\, [\beta(t_2) - \beta(t_1)] \\ & + L(x(t_2))\, [\beta(t_3) - \beta(t_2)] \\ & + \dots \\ & + L(x(t_{n-1}))\, [\beta(t_n) - \beta(t_{n-1})]\end{aligned}$$

$$E(\int_T f(X_s, s)dB_s \cdot \int_T g(X_s, s)dB_s) = \int_T E[f \cdot g]dt$$

$$(dB_t)^2 = dt$$

- The key issue is that b is evaluated at the beginning of interval, that is, we have $L(x(t_1))\, [\beta(t_2) - \beta(t_1)]$ instead of, say, $L(x(t_2))\, [\beta(t_2) - \beta(t_1)]$.
- In Riemann, Stieltjes, or Lebesgue integral the result should be independent of the evaluation point.
- The resulting calculus is called Itô calculus or the stochastic calculus

What does a solution of SDE look like? (cont.)



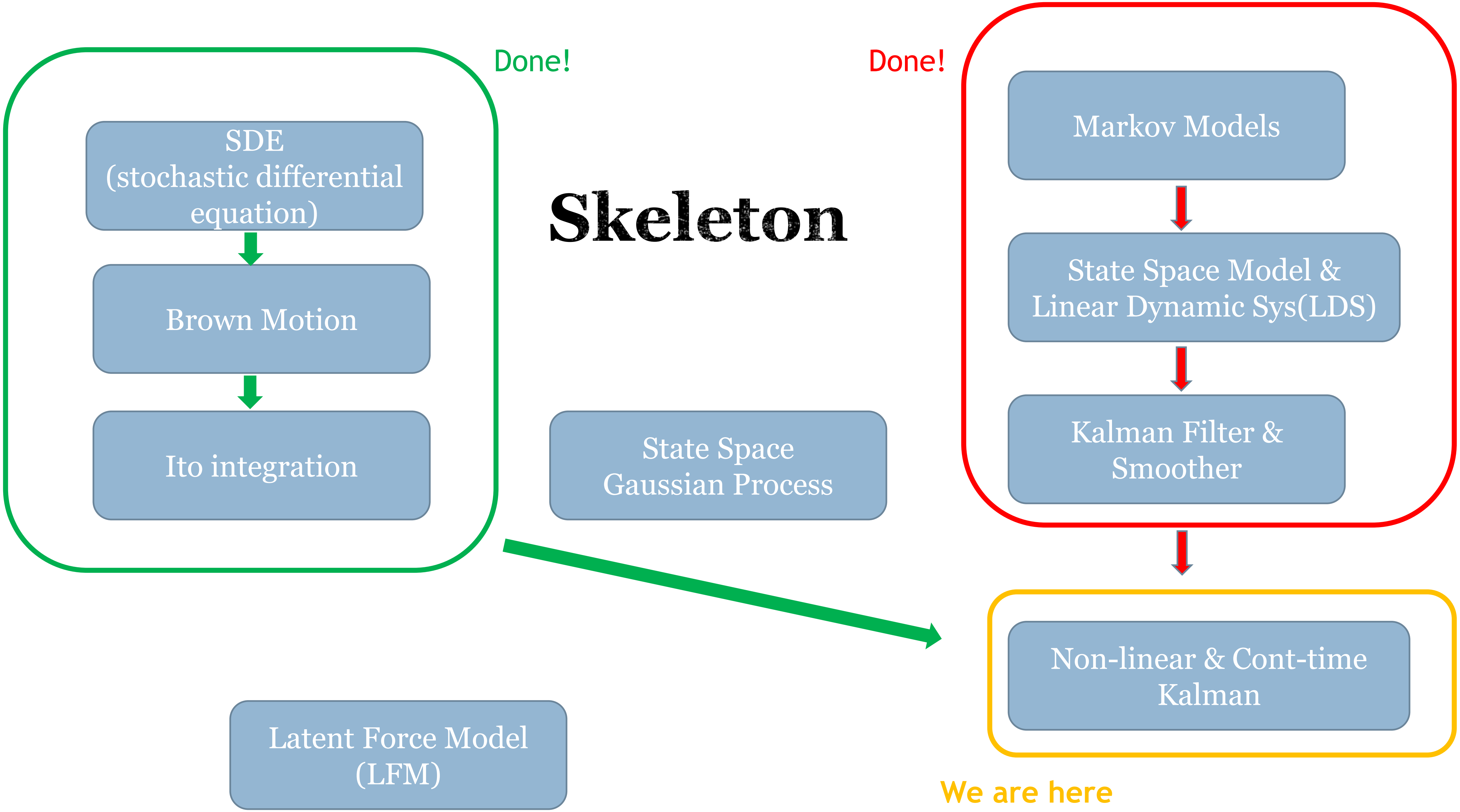
Paths of stochastic spring model

$$\frac{d^2x(t)}{dt^2} + \gamma\,\frac{dx(t)}{dt} + \nu^2\,x(t) = w(t).$$

What kind of solutions do SDEs have?

- Path of solution: Draw random path $\mathbf{w}(t)$ (or $\beta(t)$) and solve the equation using it as the input.
    - Monte Carlo simulation of SDE solutions.
    - Used in particle filtering and smoothing methods.
- Distribution of solution: Given many random $\mathbf{w}(t)$'s, what is the distribution of the state $p(\mathbf{x}(t))$?
    - Solution is given by the Fokker-Planck-Kolmogorov PDE.
    - Used in grid based and basis function methods (FEM, BEM).
- Moments: What are the mean and covariance of $\mathbf{x}(t)$?
    - Ordinary differential equations for the mean and covariance.
    - Used in non-linear Kalman (Gaussian) filters and smoothers.

Skeleton


Done!
SDE
(stochastic differential equation)
Brown Motion
Ito integration
Done!
Markov Models
State Space Model &
Linear Dynamic Sys(LDS)
Kalman Filter &
Smoother
State Space
Gaussian Process
Non-linear & Cont-time
Kalman
We are here
Latent Force Model
(LFM)

Mathematical Problem Formulation

Classical Kalman:
Linear + discrete

- Mathematical model is (the special case considered here):

Nonlinear +
Continue (trans) +
Discrete (emission)

$$\boxed{\begin{aligned} d\mathbf{x} &= \mathbf{f}(\mathbf{x})\, dt + \mathbf{L}\, d\beta(t) \\ \mathbf{y}_k &= \mathbf{H}\, \mathbf{x}(t_k) + \mathbf{r}_k . \end{aligned}}$$

Transition
Emission

$$\boxed{\begin{aligned} \mathbf{z}_n &= \mathbf{A}\mathbf{z}_{n-1} + \mathbf{w}_n \\ \mathbf{x}_n &= \mathbf{C}\mathbf{z}_n + \mathbf{v}_n \end{aligned}}$$

- The dynamics of state $\mathbf{x}(t) \in \mathbb{R}^n$ are modeled as Itô-type stochastic differential equations (SDE, Itô diffusion).
- $\beta(t) \in \mathbb{R}^s$ is a vector of Brownian motions (Wiener processes) with diffusion matrix $\mathbf{Q}$ and dimension $s \leq n$.
- $\mathbf{r}_k \in \mathbb{R}^d$ is a Gaussian random variable $\mathbf{r}_k \sim \mathrm{N}(0, \mathbf{R})$.
- We can think SDE as white noise driven differential equation

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}) + \mathbf{L}\, \mathbf{w}(t),$$

where the white noise is defined as $\mathbf{w}(t) = d\beta(t)/dt$.

Bayesian Filtering and Smoothing Solution

- We don't aim to compute the full (infinite-dimensional) posterior of the state, but instead only its time-marginals.
- Filtering/prediction solutions: Compute the posterior distribution(s)

$$p(\mathbf{x}(t) \mid \mathbf{y}_1, \ldots, \mathbf{y}_k), \qquad t \in [t_k, t_{k+1}).$$

- Smoothing solution: Compute the posterior distribution(s)

$$p(\mathbf{x}(t) \mid \mathbf{y}_1, \ldots, \mathbf{y}_T), \qquad t \in [t_0, t_T].$$

- If we could solve the transition density $p(\mathbf{x}(t_k) \mid \mathbf{x}(t_{k-1}))$, the model would reduce to a discrete-time model:

$$\begin{aligned}\mathbf{x}(t_k) &\sim p(\mathbf{x}(t_k) \mid \mathbf{x}(t_{k-1})) \\ \mathbf{y}_k &\sim p(\mathbf{y}_k \mid \mathbf{x}(t_k)).\end{aligned}$$

Continuous-Discrete Non-Linear Kalman Filtering [1/2]

- The current special case of the model is:

$$
\begin{aligned}
d\mathbf{x} &= \mathbf{f}(\mathbf{x})\,dt + \mathbf{L}\,d\beta(t) \\
\mathbf{y}_k &= \mathbf{H}\,\mathbf{x}(t_k) + \mathbf{r}_k\,.
\end{aligned}
$$

- We can now apply Gaussian (process) approximation to the posteric of the process $\mathbf{x}(t)$ – when combined with approximate Bayesian filter, leads to non-linear Kalman filters.
- Note that we can easily generalize to non-linear measurement mode $\mathbf{H}\,\mathbf{x}(t_k) \to \mathbf{h}(\mathbf{x}(t_k))$.
- The resulting approximation is of the form

$$
p(\mathbf{x}(t) \mid \mathbf{y}_{1:k}) \approx \mathrm{N}(\mathbf{x}(t) \mid \mathbf{m}(t), \mathbf{P}(t)), \quad t \in [t_k, t_{k+1}),
$$

  where $\mathbf{m}(t)$ and $\mathbf{P}(t)$ are computed by the non-linear Kalman filter.
- Different brands: EKF, UKF, CKF, GHKF, etc.

Continuous-Discrete Non-Linear Kalman Filtering [2/2]

Continuous-Discrete Non-Linear Kalman Filter

1. Prediction step: Integrate the following time t_{k-1} to t_k^-:

$$\frac{d\mathbf{m}}{dt} = \mathrm{E}[\mathbf{f}(\mathbf{x})]$$
$$\frac{d\mathbf{P}}{dt} = \mathrm{E}[(\mathbf{x} - \mathbf{m}_k)\, \mathbf{f}^T(\mathbf{x})] + \mathrm{E}[\mathbf{f}(\mathbf{x})\,(\mathbf{x} - \mathbf{m})^T] + \mathrm{E}[\mathbf{L}(\mathbf{x})\, \mathbf{Q}\, \mathbf{L}^T(\mathbf{x})].$$

2. Update step: Update step is the linear Kalman filter update:

$$\begin{aligned}
\mathbf{S}_k &= \mathbf{H}\, \mathbf{P}(t_k^-)\, \mathbf{H}^T + \mathbf{R}_k \\
\mathbf{K}_k &= \mathbf{P}(t_k^-)\, \mathbf{H}_k^T\, \mathbf{S}_k^{-1} \\
\mathbf{m}(t_k) &= \mathbf{m}(t_k^-) + \mathbf{K}_k\, [\mathbf{y}_k - \mathbf{H}\, \mathbf{m}(t_k^-)] \\
\mathbf{P}(t_k) &= \mathbf{P}(t_k^-) - \mathbf{K}_k\, \mathbf{S}_k\, \mathbf{K}_k^T.
\end{aligned}$$

Continuous-time extended Kalman filter [edit]

Model

$$\boxed{\begin{aligned}\dot{\mathbf{x}}(t) &= f\bigl(\mathbf{x}(t), \mathbf{u}(t)\bigr) + \mathbf{w}(t) & \mathbf{w}(t) &\sim \mathcal{N}\bigl(\mathbf{0}, \mathbf{Q}(t)\bigr)\\ \mathbf{z}(t) &= h\bigl(\mathbf{x}(t)\bigr) + \mathbf{v}(t) & \mathbf{v}(t) &\sim \mathcal{N}\bigl(\mathbf{0}, \mathbf{R}(t)\bigr)\end{aligned}}$$

State transition become a First-order SDE

Initialize

$$\hat{\mathbf{x}}(t_0) = E\bigl[\mathbf{x}(t_0)\bigr], \mathbf{P}(t_0) = Var\bigl[\mathbf{x}(t_0)\bigr]$$

Predict-Update

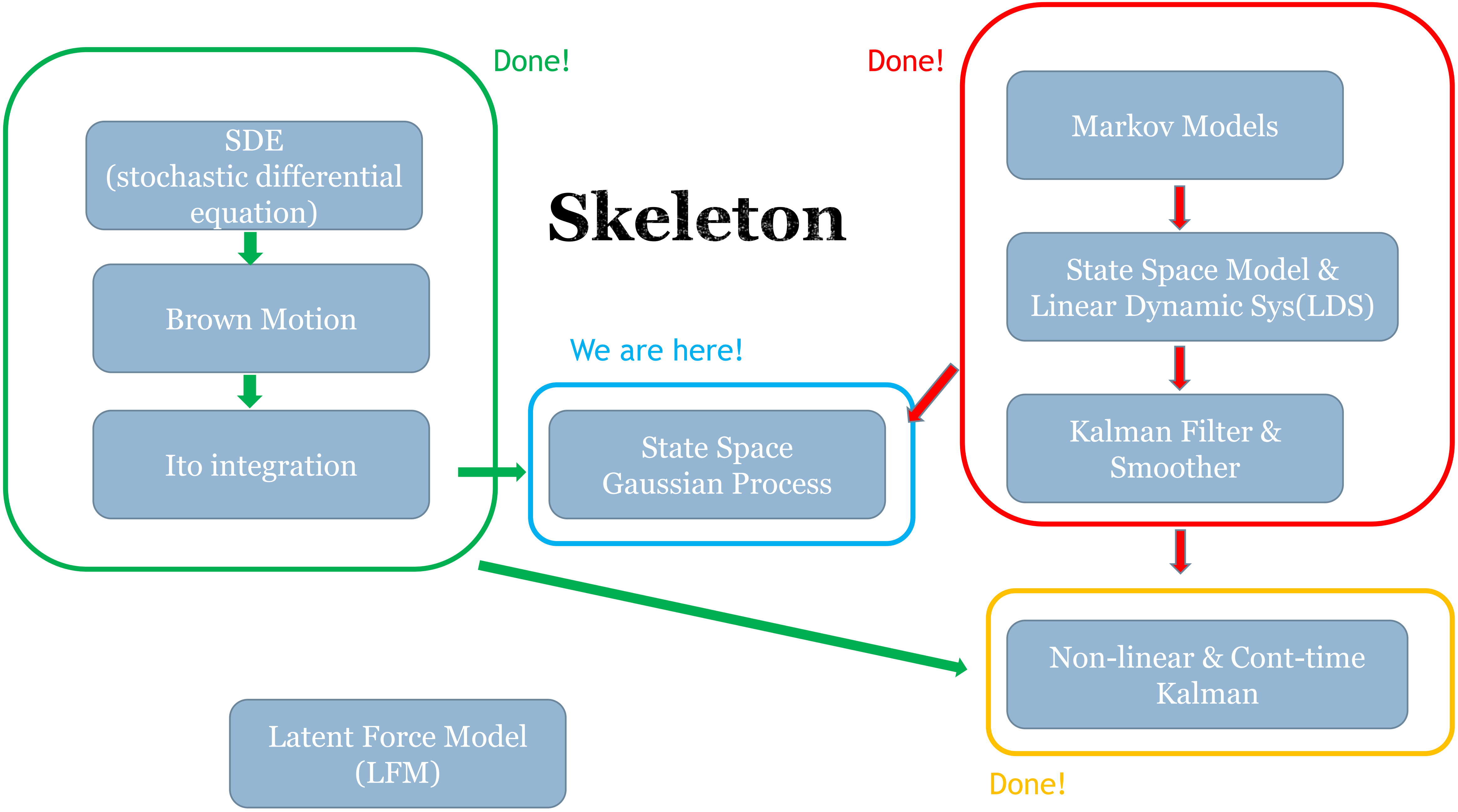
$$\boxed{\begin{aligned}\dot{\hat{\mathbf{x}}}(t) &= f\bigl(\hat{\mathbf{x}}(t), \mathbf{u}(t)\bigr) + \mathbf{K}(t)\Bigl(\mathbf{z}(t) - h\bigl(\hat{\mathbf{x}}(t)\bigr)\Bigr)\\ \dot{\mathbf{P}}(t) &= \mathbf{F}(t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{F}(t)^{\top} - \mathbf{K}(t)\mathbf{H}(t)\mathbf{P}(t) + \mathbf{Q}(t)\end{aligned}}$$

$$\mathbf{K}(t) = \mathbf{P}(t)\mathbf{H}(t)^{\top}\mathbf{R}(t)^{-1}$$

$$\mathbf{F}(t) = \left.\frac{\partial f}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}(t), \mathbf{u}(t)}$$

$$\mathbf{H}(t) = \left.\frac{\partial h}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}(t)}$$

Moment of state can be computed by solving ODE


Skeleton
Done!
SDE
(stochastic differential equation)
Brown Motion
Ito integration
We are here!
State Space
Gaussian Process
Done!
Markov Models
State Space Model &
Linear Dynamic Sys(LDS)
Kalman Filter &
Smoother
Non-linear & Cont-time
Kalman
Done!
Done!
Latent Force Model
(LFM)

State Space GP

- GPs with certain stationary covariance functions (e.g.Matern) can be represented as state space models.

- Consider a Gaussian process regression problem

$$f(x) \sim \mathrm{GP}(0, \sigma^2 \exp(-\lambda|x - x'|))$$
$$y_k = f(x_k) + \varepsilon_k$$

- This is equivalent to the state-space model

$$\frac{df(t)}{dt} = -\lambda f(t) + w(t)$$
$$y_k = f(t_k) + \varepsilon_k$$

that is, with $f_k = f(t_k)$ we have a Gauss-Markov model

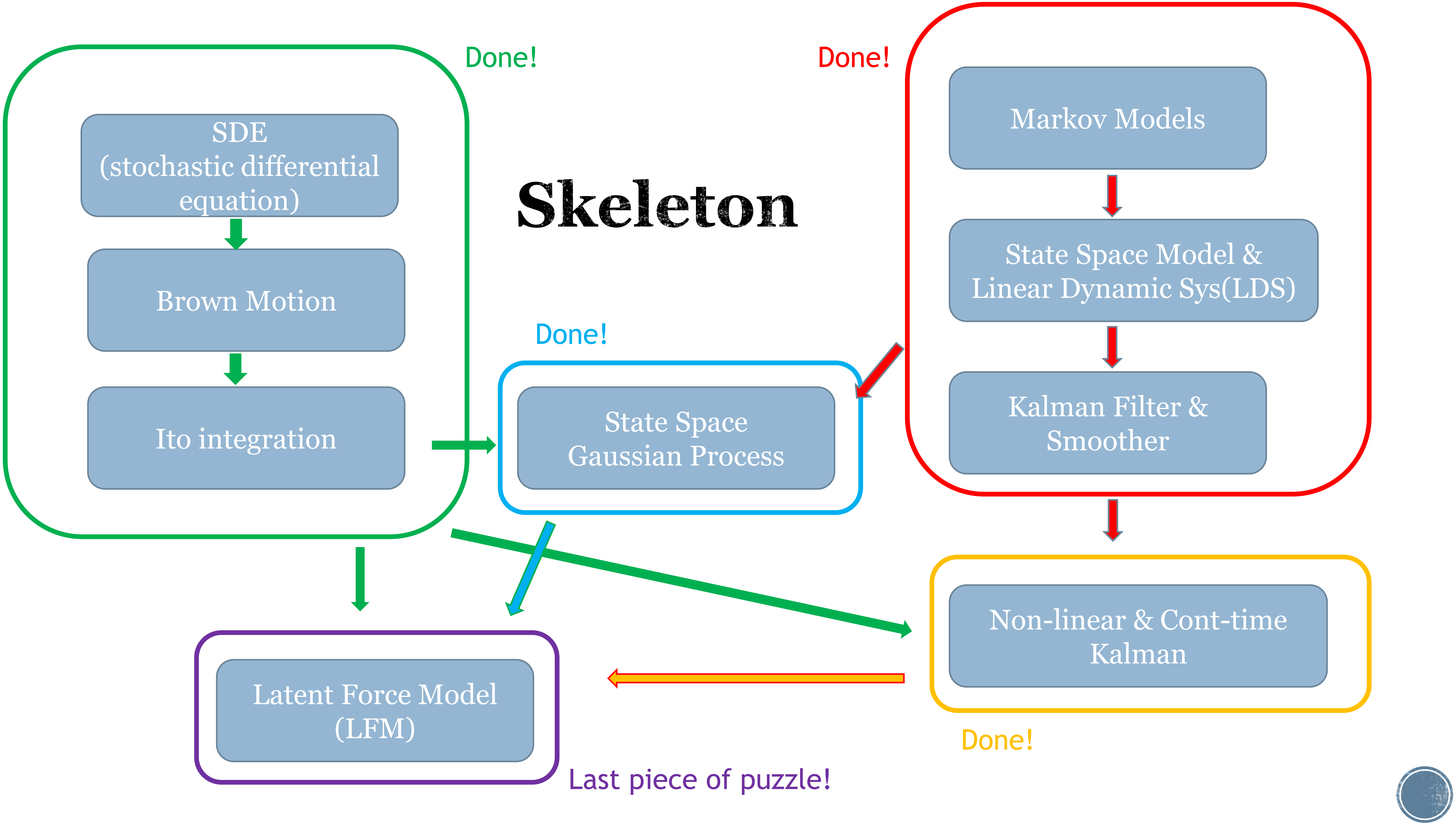
$$f_{k+1} \sim p(f_{k+1} \mid f_k)$$
$$y_k \sim p(y_k \mid f_k)$$

- Solvable in $O(n)$ time using Kalman filter/smoother.

State Space GP

GPs with certain stationary covariance functions (e.g.Matern) can be represented as state space models.

GP model $\mathbf{x} \in \mathbb{R}^d, t \in \mathbb{R}$	Equivalent S(P)DE model
Spatial $k(\mathbf{x}, \mathbf{x}')$	SPDE model ($\mathcal{L}$ is an operator) $$\mathcal{L}\, f(\mathbf{x}) = w(\mathbf{x})$$
Temporal $k(t, t')$	State-space/SDE model $$\frac{d\mathbf{f}(t)}{dt} = \mathbf{A}\,\mathbf{f}(t) + \mathbf{L}\,w(t)$$
Spatio-temporal $k(\mathbf{x}, t; \mathbf{x}', t')$	Stochastic evolution equation $$\frac{\partial}{\partial t}\mathbf{f}(\mathbf{x}, t) = \mathcal{A}_x\,\mathbf{f}(\mathbf{x}, t) + \mathbf{L}\,w(\mathbf{x}, t)$$



Skeleton
Done!
SDE
(stochastic differential equation)
Brown Motion
Ito integration
Done!
Markov Models
State Space Model &
Linear Dynamic Sys(LDS)
Kalman Filter &
Smoother
Done!
State Space
Gaussian Process
Non-linear & Cont-time
Kalman
Done!
Latent Force Model
(LFM)
Last piece of puzzle!

The Basic Idea of State-Space Representation

- Assume that our latent force model is of the form

$$\boxed{\frac{dx_f(t)}{dt} = g(x_f(t)) + u(t),}$$

where $u(t)$ is the latent force.

- We measure the system at discrete instants of time:

$$y_k = x_f(t_k) + r_k$$

- Let's now model $u(t)$ as a Gaussian process of Matern type

$$C(\tau) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\,\frac{\tau}{l}\right)^{\nu} K_{\nu}\left(\sqrt{2\nu}\,\frac{\tau}{l}\right)$$

- Recall that if, for example, $\nu = 1/2$ then the GP can be expressed as the solution of the stochastic differential equation (SDE)

$$\boxed{\frac{du(t)}{dt} = -\lambda\, u(t) + w(t)}$$

The Basic Idea of State-Space Representation (cont.)

- If we define $\mathbf{x} = (x_f, u)$, we get a two-dimensional SDE

$$\frac{d\mathbf{x}}{dt} = \underbrace{\begin{pmatrix} g(x_1(t)) + x_2(t) \\ -\lambda\, x_2(t) \end{pmatrix}}_{\mathbf{f}(\mathbf{x})} + \underbrace{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}_{\mathbf{L}} w(t)$$

- We can now rewrite the measurement model as

$$y_k = \underbrace{\begin{pmatrix} 1 & 0 \end{pmatrix}}_{\mathbf{H}} \mathbf{x}(t_k) + r_k$$

- Thus the result is a model of the generic form

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}) + \mathbf{L}\,\mathbf{w}(t)$$
$$\mathbf{y}_k = \mathbf{H}\,\mathbf{x}(t_k) + \mathbf{r}_k.$$

- This model can now be efficiently tackled with non-linear Kalman filtering and smoothing.

SDE View of Latent Force Models [1/4]

- Let's now take a look at the non-linear state-space LFM methodology presented in Hartikainen and Särkkä (2012).
- Consider the latent force model (Lawrence et al., 2006)

$$\frac{dx_j(t)}{dt} = B_j + \sum_{r=1}^{R} S_{j,r} g_j(u_r(t)) - D_j x_j(t), \quad j = 1, \ldots, N$$

- We can now use independent Gaussian process (GP) priors

$$u_r(t) \sim \mathrm{GP}(m(t), k_{u_r}(t, t')), \quad r = 1, \ldots, R$$

where $m(t)$ and $k_{u_r}(t, t')$ were suitably chosen mean and covariance functions.

- That is, we can formulate the GP priors on the components of $\mathbf{u}(t) = (u_1(t) \ldots u_R(t))^T$ as multivariate space space models (SDEs) of form

$$d\mathbf{z}_r(t) = \boxed{\mathbf{F}_{z,r}} \mathbf{z}_r(t)\, dt + \boxed{\mathbf{L}_{z,r}} d\beta_{z,r}(t)$$

Companion form matrix

where

$$\mathbf{z}_r(t) = \begin{pmatrix} u_r(t) & \frac{du_r(t)}{dt} & \cdots & \frac{d^{d_r-1}u_r(t)}{dt^{d_r-1}} \end{pmatrix}^T$$

and

$$\mathbf{F}_{z,r} = \begin{pmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -a_r^0 & \cdots & -a_r^{p_r-2} & -a_r^{p_r-1} \end{pmatrix}, \quad \mathbf{L}_{z,r} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ q_r \end{pmatrix}.$$

General LFM in SDE view

Summary

- Non-linear LFMs can be converted into state-space form by:
  1. Converting the latent GPs into state-space form.
  2. Forming an augmented state space model.
- Bayesian filtering and smoothing, in principle, provide the full solution to the problem.
- In practice, formal solution is intractable – involves, e.g., solutions to particle differential equations.
- Approximate inference in non-linear LFMs can be implemented with non-linear Kalman filters and smoothers.

LFM in SDE view:

One trivial application

Single-latent-force + learnable transition & emission

q(f,\theta) can be further inferenced by SVI, sampling..

Black-Box Inference for Non-Linear Latent Force Models, AISTAT 2020

$$\alpha_0(\boldsymbol{x}, u, t; \boldsymbol{\theta})\boldsymbol{x}(t) + \sum_{i=1}^{n} \alpha_i(\boldsymbol{x}, u, t; \boldsymbol{\theta}) \frac{\mathrm{d}^i}{\mathrm{d}t^i}\boldsymbol{x}(t) = u(t),$$

$$\boldsymbol{y}_j \sim \pi(\boldsymbol{h}(\boldsymbol{x}(\tau_j); \boldsymbol{\theta}))$$

where $u(t) \sim \mathcal{GP}(0, k(t, t'))$

define a joint state vector $\boldsymbol{f}(t) = [\boldsymbol{x}(t), \mathrm{d}\boldsymbol{x}/\mathrm{d}t, \ldots, u(t_k), \mathrm{d}u/\mathrm{d}t, \ldots]^\top$

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{f}(t) = \boxed{\boldsymbol{D}}(\boldsymbol{f}, t; \boldsymbol{\theta}) + \boxed{\mathbf{L}}w(t),$$

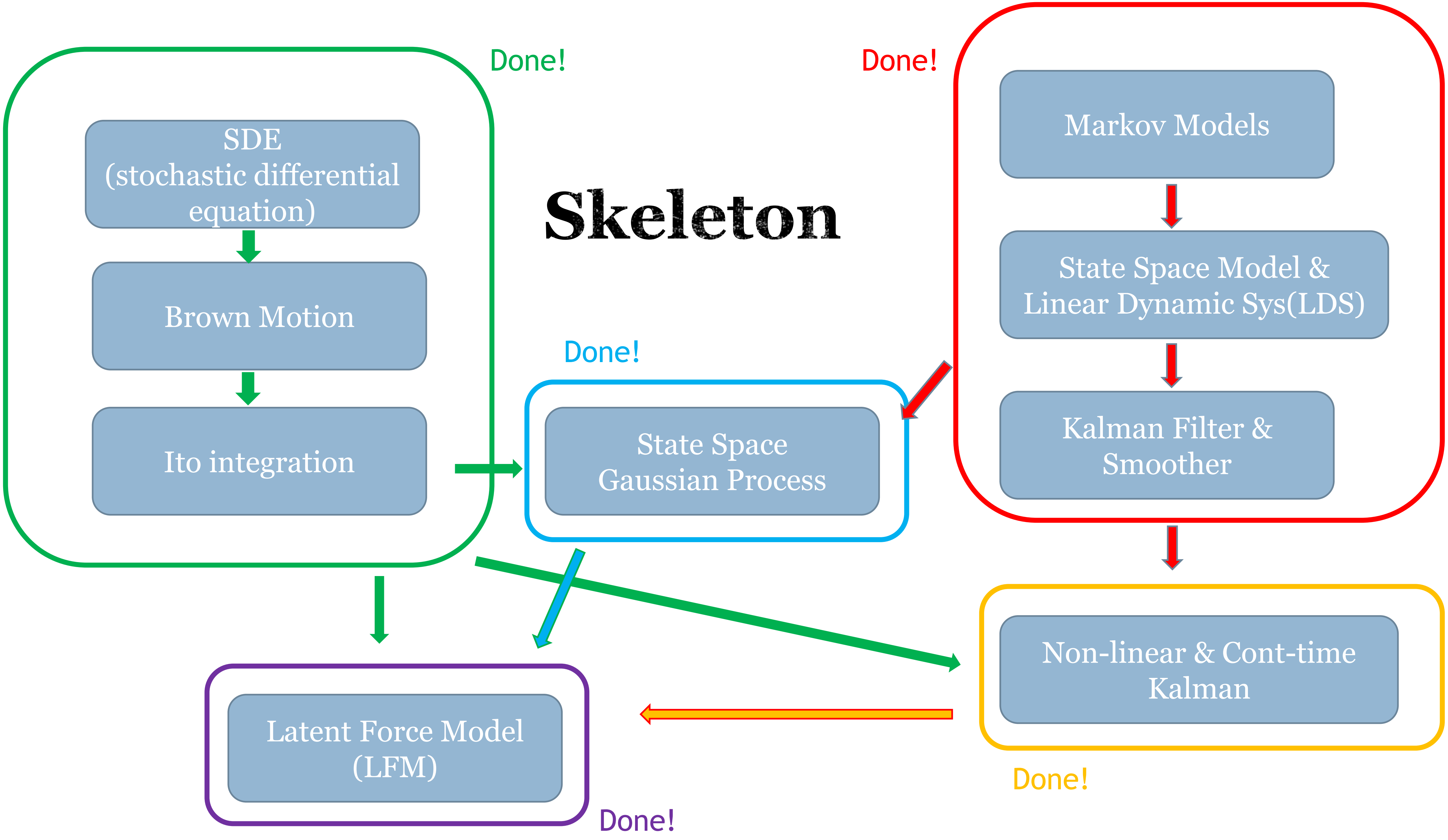
Companion form sys

$$p(\boldsymbol{x}_{0:T}, u_{0:T}, \boldsymbol{\theta} \mid \boldsymbol{y}) \propto$$

$$p(\boldsymbol{\theta})p(\boldsymbol{f}_0 \mid \boldsymbol{\theta}) \prod_{k=0}^{T-1} \boxed{p(\boldsymbol{f}_{k+1} \mid \boldsymbol{f}_k, \boldsymbol{\theta})} \prod_{j=1}^{N} \boxed{p(\boldsymbol{y}_j \mid \boldsymbol{f}(\tau_j), \boldsymbol{\theta})},$$

Transition density from non-linear Kalman

Observed/llk function


Skeleton
Done!
SDE
(stochastic differential
equation)
Brown Motion
Ito integration
Done!
State Space
Gaussian Process
Done!
Markov Models
State Space Model &
Linear Dynamic Sys(LDS)
Kalman Filter &
Smoother
Non-linear & Cont-time
Kalman
Done!
Latent Force Model
(LFM)
Done!

Simo Särkkä

FOLLOW

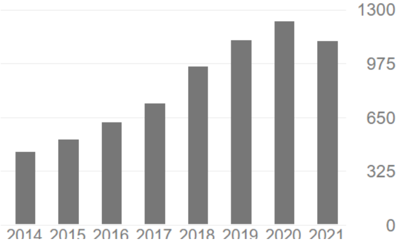
Associate Professor, Aalto University
Verified email at aalto.fi - Homepage

multi-sensor data fusion   Bayesian filtering and smoo…   machine learning   medical technology   AI for health

TITLE	CITED BY	YEAR
Bayesian filtering and smoothing S Särkkä Cambridge University Press	1795	2013
On unscented Kalman filtering for state estimation of continuous-time nonlinear systems S Sarkka IEEE Transactions on automatic control 52 (9), 1631-1641	601	2007
Recursive noise adaptive Kalman filtering by variational Bayesian approximations S Sarkka, A Nummenmaa IEEE Transactions on Automatic control 54 (3), 596-600		
Rao-Blackwellized particle filter for multiple target tracking S Sarkka, A Vehtari, J Lampinen Information Fusion 8 (1), 2-15		
Unscented Rauch--Tung--Striebel Smoother S Särkkä IEEE transactions on automatic control 53 (3), 845-849		
Kalman filtering and smoothing solutions to temporal Gaussian process regression models J Hartikainen, S Särkkä 2010 IEEE international workshop on machine learning for signal processing …		
Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing: A look a Gaussian process regression through Kalman filtering		

Cited by   VIEW ALL

	All	Since 2016
Citations	7920	5795
h-index	38	34
i10-index	91	84



Arno Solin

FOLLOW

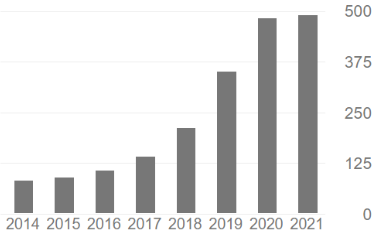
Assistant Professor in Machine Learning, Aalto University
Verified email at aalto.fi - Homepage

Machine learning   Bayesian nonparametrics   Signal processing

TITLE	CITED BY	YEAR
Interpolation consistency training for semi-supervised learning V Verma, K Kawaguchi, A Lamb, J Kannala, A Solin, Y Bengio, … Neural Networks	241	2021
Spatio-Temporal Learning via Infinite-Dimensional Bayesian Filtering and Smoothing S Särkkä, A Solin, J Hartikainen Signal Processing Magazine 30 (4), 51-61	182 *	2013
Applied Stochastic Differential Equations S Särkkä, A Solin Cambridge University Press	161	2019
Optimal filtering with Kalman filters and smoothers–a Manual for Matlab toolbox EKF/UKF J Hartikainen, A Solin, S Särkkä Biomedical Engineering, 1-57	154	2008
Optimal filtering with Kalman filters and smoothers J Hartikainen, A Solin, S Särkkä Department of Biomedical Engineering and Computational Science, Aalto …	136 *	2011
Variational Fourier Features for Gaussian Processes J Hensman, N Durrande, A Solin Journal of Machine Learning Research 18 (151), 1-52	134	2018
Hilbert space methods for reduced-rank Gaussian process regression A Solin, S Särkkä Statistics and Computing 30 (2), 419-446	123	2020

Cited by   VIEW ALL

	All	Since 2016
Citations	2113	1797
h-index	23	22
i10-index	33	32



Public access   VIEW ALL

0 articles   38 articles

not available   available

Based on funding mandates

Co-authors   VIEW ALL

Simo Särkkä
Associate Professor, Aalto Unive…

Aalto University, Espoo, Finland