Universität Stuttgart
IPVS – Institute for Parallel and Distributed Systems

Analytic Computing

Advanced Topics in Machine Learning 4 Representation: Dynamic Bayesian Networks

Prof. Dr. Steffen Staab
Dr. Rafika Boutalbi
Zihao Wang
https://www.ipvs.uni-stuttgart.de/departments/ac/

Learning Objectives

- How to model time in Bayesian Networks?
  - State space models
    - Discrete: Hidden Markov Models
    - Continuous: Linear Gaussian state space models
      - Kalman Filter
    - Mixed:
      - Conditional Linear Gaussian state space models

Disclaimer

Figures and examples not marked otherwise are taken from the book by Koller & Friedman

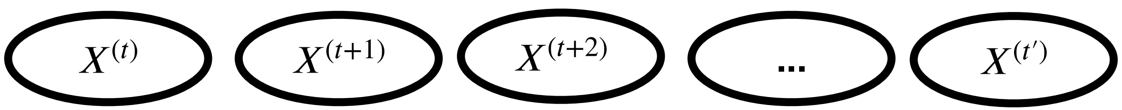
1 Dynamic Bayesian Networks

Modelling sequential data

- Sequential data is everywhere, e.g.,
  - Sequence data (offline): Biosequence analysis, text processing, ...
  - Temporal data (online): Speech recognition, visual tracking, financial forecasting, ...
- Problems: classification, segmentation, state estimation, fault diagnosis, prediction, ...
- Solution: build/learn generative models, then compute P(quantity of interest|evidence) using Dynamic Bayesian Network.

Distributions over discrete time

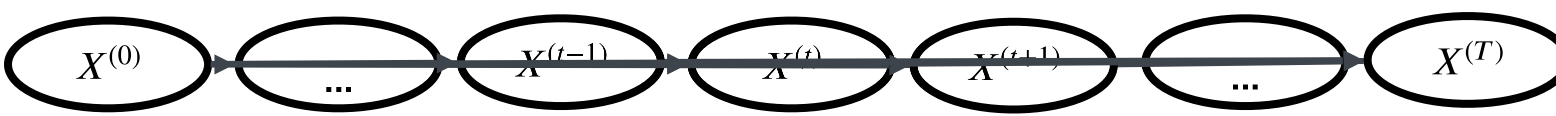
- Time granularity Δ
- variable X at time Δt: $X^{(t)}$
- $X^{(t:t')} = \{X^{(t)}, \ldots, X^{(t')}\},\ t \leq t'$



Represent joint probability $\mathrm{P}(X^{(t:t')})$

General Dynamic Bayesian Models

$$\mathrm{P}\left(X^{(0:T)}\right) = \mathrm{P}\left(X^{(0)}\right)\prod_{t=0}^{T-1}\mathrm{P}\left(X^{(t+1)}\,\middle|\,X^{(0:t)}\right)$$



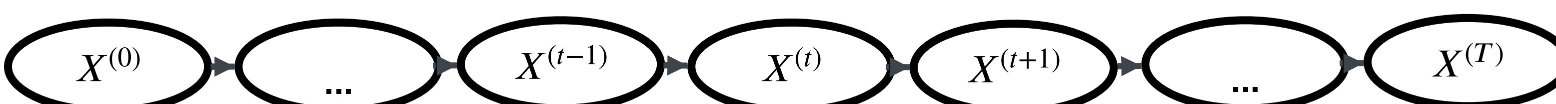
Markov Assumption in Dynamic Bayesian Models

$$\mathrm{P}\left(X^{(0:T)}\right) = \mathrm{P}\left(X^{(0)}\right) \prod_{t=0}^{T-1} \mathrm{P}\left(X^{(t+1)} \,\middle|\, X^{(0:t)}\right)$$

$$\left(X^{(t+1)} \perp X^{(0:t-1)} \,\middle|\, X^{(t)}\right)$$

Therefore:

$$\mathrm{P}\left(X^{(0:T)}\right) = \mathrm{P}\left(X^{(0)}\right) \prod_{t=0}^{T-1} \mathrm{P}\left(X^{(t+1)} \,\middle|\, X^{(t)}\right)$$



Further Assumption: Time Invariance

- Template probability model $P(X' \mid X)$
- For all t:

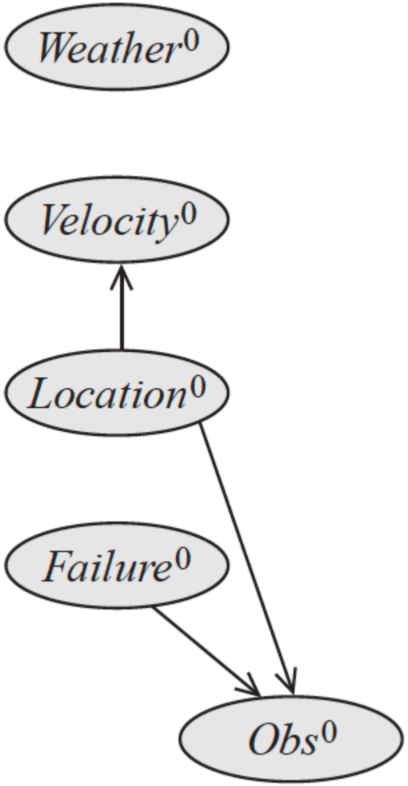
$$\mathrm{P}\left(X^{(t+1)} \,\middle|\, X^{(t)}\right) = P(X' \mid X)$$

Do these assumptions hold?

- Markov assumption: forget the past
  - counterexample: X is robot location, then velocity affects probability of next location
  - example: X models robot location and velocity.
    - Yes!
      - Unless different types of friction must be considered
- Time invariance:
  - rush hour may change traffic patterns over time

Applicability depends on what you model and how you model!

Initial State Distribution



Time slice 0

$$P\left(W^{(0)}, V^{(0)}, L^{(0)}, F^{(0)}, O^{(0)}\right)$$

$$=$$

$$P\left(W^{(0)}\right)$$

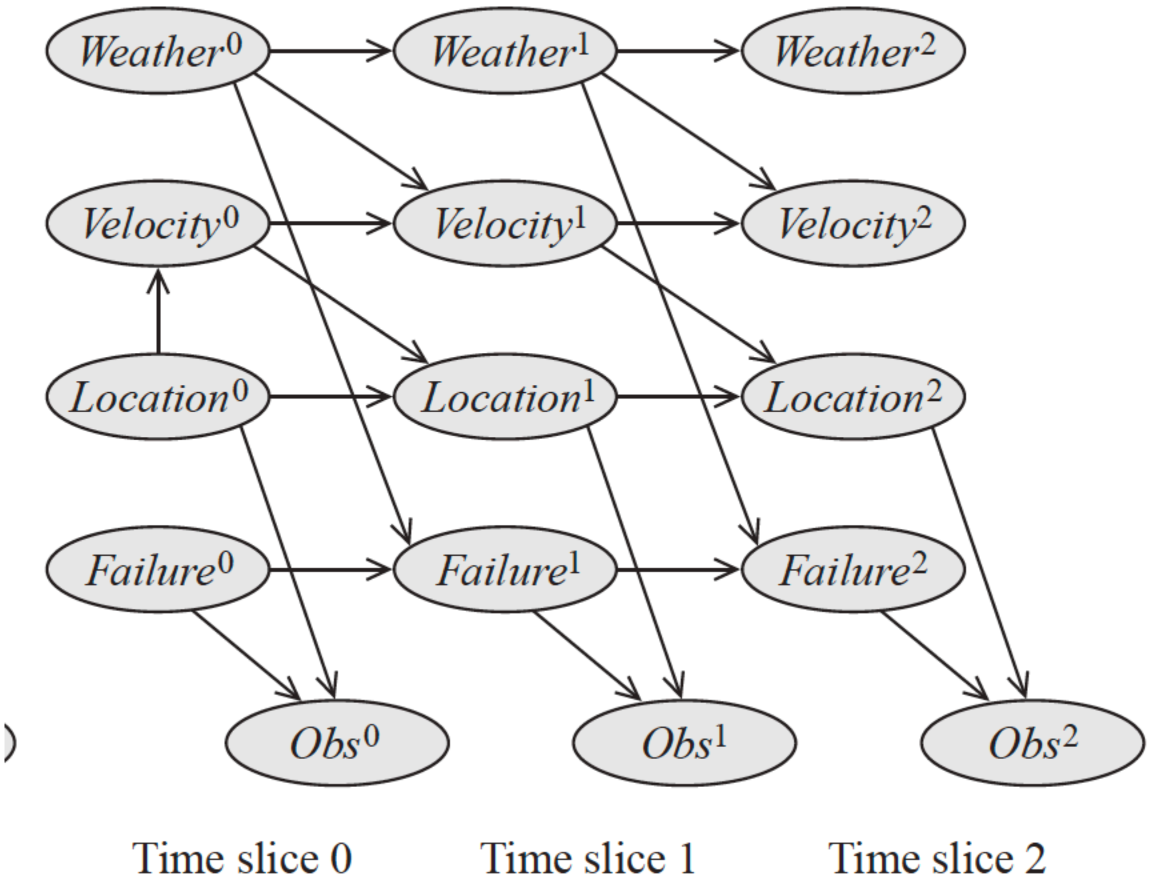
$$P\left(V^{(0)} \middle| L^{(0)}\right)$$

$$P\left(L^{(0)}\right)$$

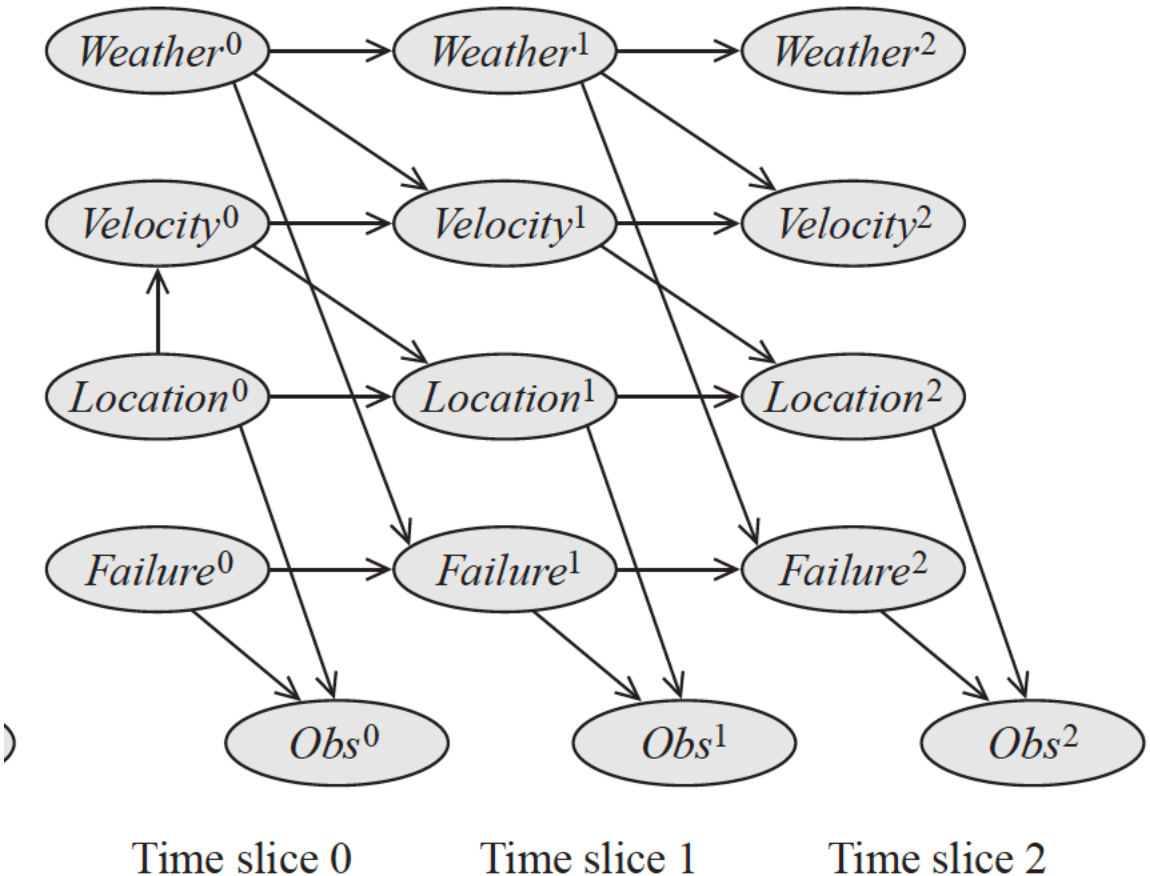
$$P\left(F^{(0)}\right)$$

$$P\left(O^{(0)} \middle| F^{(0)}, L^{(0)}\right)$$

3 Steps of the Dynamic Bayesian Network



Ground Bayesian Network



Grounding is the projection of first-order formulas onto variable-less formulas:

First-order:

$$\forall x, y: \left(x > y\right) \rightarrow \left(y < x\right)$$

Example groundings:

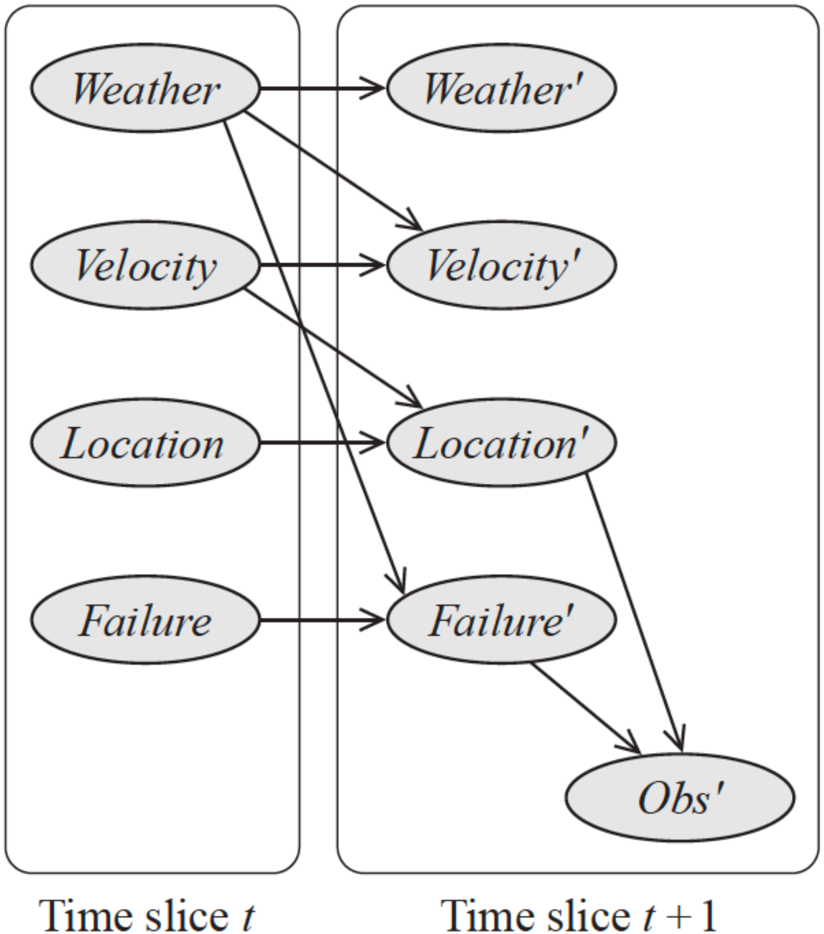
$$\left(5 > 3\right) \rightarrow \left(3 < 5\right)$$

$(3 > 4) \rightarrow (4 < 3)$

$(tom > eva) \rightarrow (eva < tom)$

Lifting is the inverse

Template Transition Model



Structure of a state-observation model

1. system evolves: transition model
2. observation/sensing is modeled separately

   observations are modeled only as O' (not as O)

   observation model

$$P(W', V', L', F', O' \mid W, V, L, F)$$

$$=$$

$$P(W' \mid W)$$

$$P(V' \mid W, V)$$

$$P(L' \mid L, V)$$

$$P(F' \mid F, W)$$

2-time-slice Bayesian Network

- A transition model over template $\boldsymbol{X} = \{X_1, \ldots X_n\}$ is specified as a Bayesian Network fragment such that:
  - The nodes include all $X'_1, \ldots X'_n$ and a subset of $X_1, \ldots X_n$
  - Only the nodes $X'_1, \ldots X'_n$ have parents and a conditional probability table
  - The 2TBN defines a conditional distribution

$$P(\boldsymbol{X}' \mid \boldsymbol{X}) = \prod_{i=1}^{n} \underbrace{P\left(X'_i \,\middle|\, \mathrm{Parents}(X'_i)\right)}_{\text{Template factor}}$$

Dynamic Bayesian Network

Definition: A dynamic Bayesian network (DBN) is a pair $\langle \mathscr{B}_0, \mathscr{B}_\rightarrow \rangle$, where $\mathscr{B}_0$ is a Bayesian network over $\mathscr{X}^{(0)}$, representing the initial distribution over states, and $\mathscr{B}_\rightarrow$ is a 2-time-slice Bayesian Network for the process.

For any desired time span $T \geq 0$, the distribution over $\mathscr{X}^{(0:T)}$ is defined as a unrolled Bayesian network, where, for any $i = 1, \ldots, n$:

- the structure and $CPDs$ of $X_i^{(0)}$ are the same as those for X_i in $\mathscr{B}_0$,
- the structure and CPD of $X_i^{(t)}$ for $t > 0$ are the same as those for X_i' in $\mathscr{B}_\rightarrow$.

Inference Tasks

- Prediction:

$$P(X^{(T+k)} \mid e^{(0:T)})$$

- Most likely explanation:

$$\text{argmax}_{x^{(0:T)}} P(x^{(0:T)} \mid e^{(0:T)})$$

- Filtering:

$$P(X^{(T)} \mid e^{(0:T)})$$

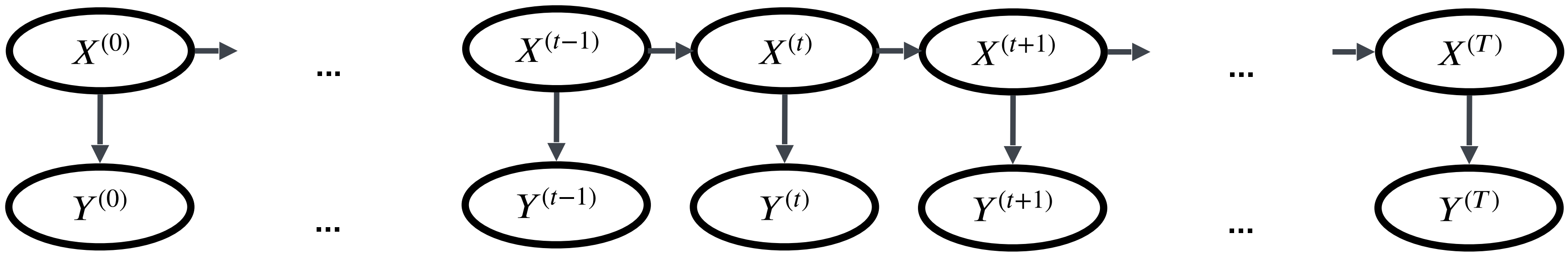
- Smoothing

$$P(X^{(t)} \mid e^{(0:T)})$$

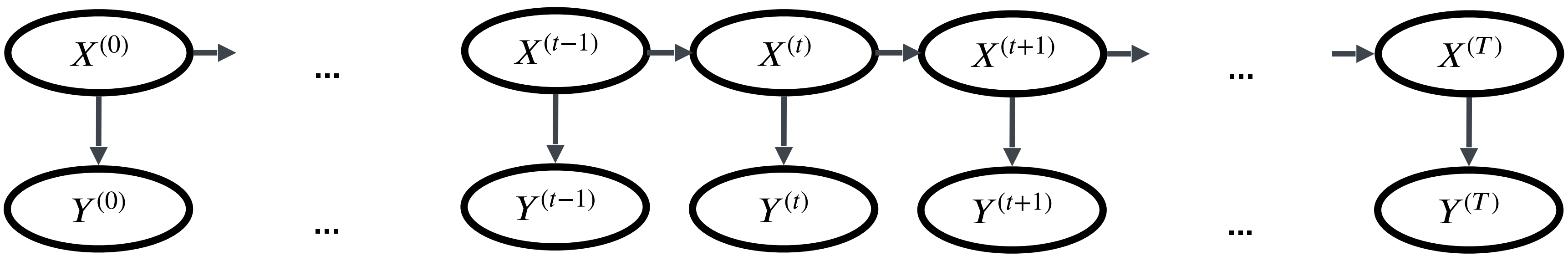
Question 1: Temporal Bayesian Networks

2 Hidden Markov Model (HMM)

State-space Model



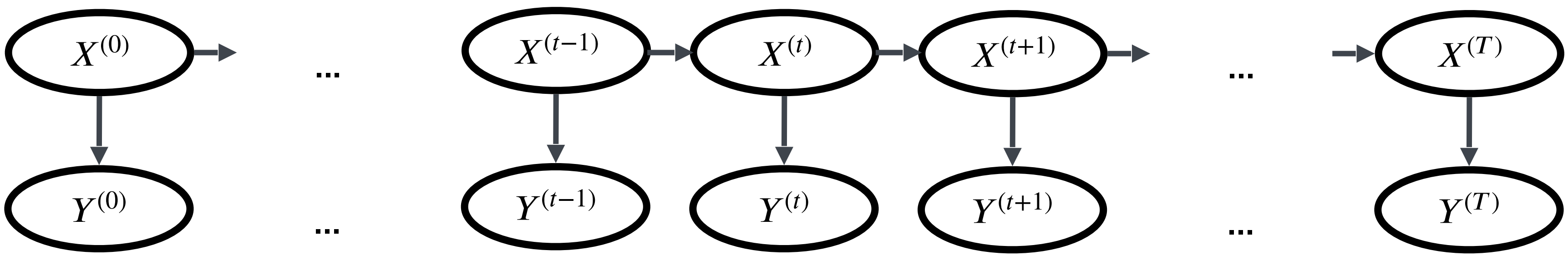
Hidden Markov Model



Model:

- $\boldsymbol{X} = \{X^{(1)}, \ldots X^{(n)}\}$ are latent variables with values from **discrete set of states** $\{1, \ldots. K\}$
- Probability of transition from state $i \in \{1 \ldots K\}$ to state $j \in \{1 \ldots K\}$ is specified in time-invariant, probabilistic transition matrix A of shape $K \times K$
- $\boldsymbol{Y} = \{Y^{(1)}, \ldots Y^{(n)}\}$ are observable variables with values from a set of size L
- Probability of emitting symbol $j \in \{1 \ldots L\}$ is specified in $K \times L$ observation matrix C

Hidden Markov Model



Model:

$$P\left(X^{(0:T)}, Y^{(0:T)}\right) = P\left(X^{(0)}\right) P(Y^{(0)} \mid X^{(0)}) \prod_{t=1}^{T} P\left(X^{(t)} \middle| X^{(t-1)}\right) P(Y^{(t)} \mid X^{(t)})$$

- $\boldsymbol{X} = \{X^{(1)}, \ldots X^{(n)}\}$ are latent variables with values from a **discrete set of states**
- $\boldsymbol{Y} = \{Y^{(1)}, \ldots Y^{(n)}\}$ are observable variables with values from a set of size L

Example

- $X^{(t)}$ is the ideal phonem
- $Y^{(t)}$ is the uttered phon
- There is Gaussian noise between what is meant and what is uttered

arxiv.org/abs/2011.04640

Apps UniStuttgart LEO English-Germ... Google Suche Google Ilias Soton UniKO Begutachtung Privat Projects Reisen Suchen Watch Last week t...

Cornell University

W
the Simon

arXiv.org > cs > arXiv:2011.04640

Search...

Help | Advance

Computer Science > Computation and Language

[Submitted on 9 Nov 2020]

Scaling Hidden Markov Language Models

Justin T. Chiu, Alexander M. Rush

The hidden Markov model (HMM) is a fundamental tool for sequence modeling that cleanly separates the hidden state from the emission structure. However, this separation makes it difficult to fit HMMs to large datasets in modern NLP, and they have fallen out of use due to very poor performance compared to fully observed models. This work revisits the challenge of scaling HMMs to language modeling datasets, taking ideas from recent approaches to neural modeling. We propose methods for scaling HMMs to massive state spaces while maintaining efficient exact inference, a compact parameterization, and effective regularization. Experiments show that this approach leads to models that are more accurate than previous HMM and n-gram-based methods, making progress towards the performance of state-of-the-art neural models.

Comments:	9 pages, accepted as a short paper at EMNLP 2020
Subjects:	**Computation and Language (cs.CL)**; Machine Learning (cs.LG)
Journal reference:	EMNLP 2020
Cite as:	arXiv:2011.04640 **[cs.CL]**
	(or arXiv:2011.04640v1 **[cs.CL]** for this version)

Question 2: HMM

3 Continuous Variables in Bayesian Networks

Linear Gaussian Model

- How to model a continuous dependency $P(Y \mid X)$?

- Linear function of X:

$$P(Y \mid x) = \mathcal{N}(\beta_0 + \beta_1\, x; 1)$$

does not depend on parents

- Several parents:

$$P\left(Y \,\middle|\, x_1, \ldots, x_k\right) = \mathcal{N}(\beta_0 + \beta_1\, x + \ldots + \beta_k x_k; \sigma^2)$$

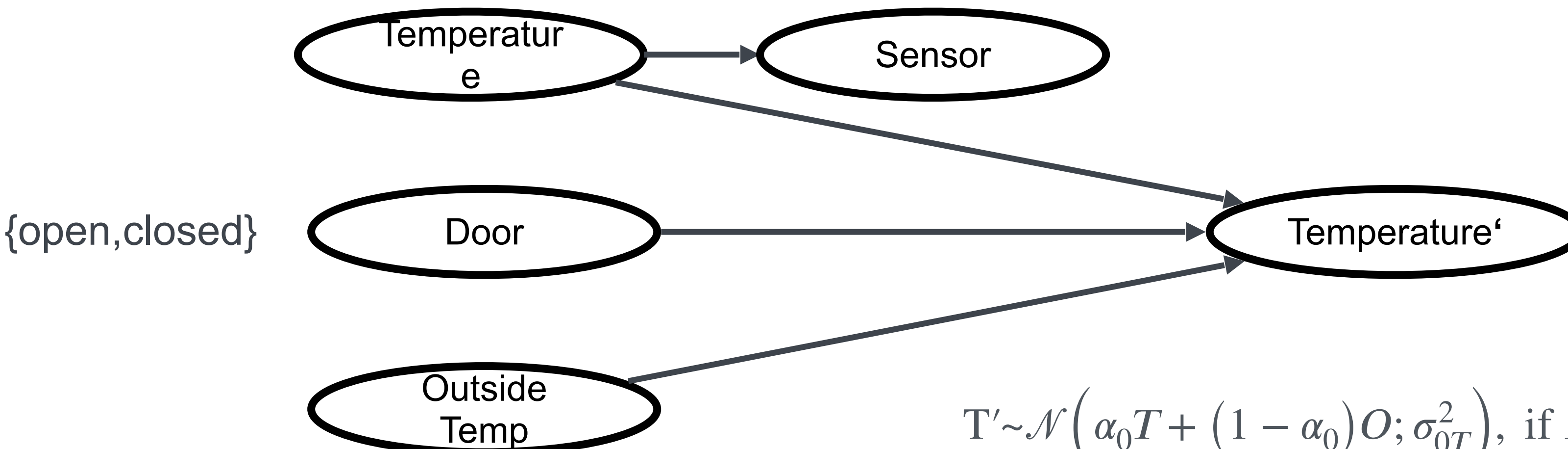
- In vector notation:

$$P(Y \mid x) = \mathcal{N}(\beta_0 + \beta^T x; \sigma^2)$$

Limited in expressivity, but still rich enough for many applications.

Conditional Linear Gaussian

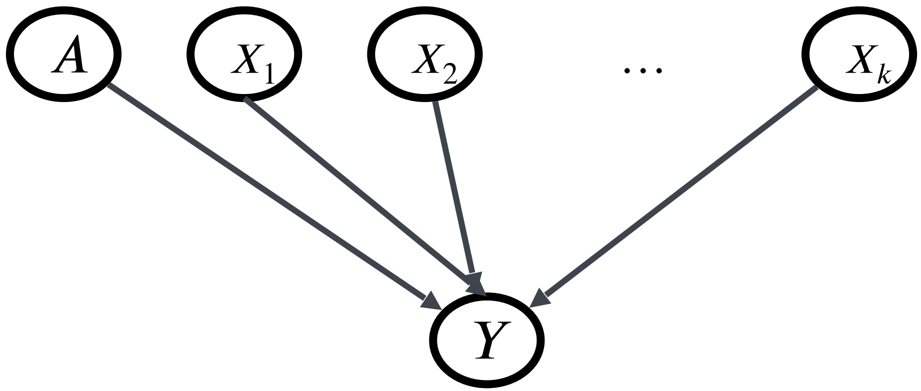
$S \sim \mathcal{N}(T; \sigma_S^2)$



$\mathrm{T}' \sim \mathcal{N}\left(\alpha_0 T + (1 - \alpha_0) O; \sigma_{0T}^2\right),\ \text{if } D = \text{„closed“}$

$\mathrm{T}' \sim \mathcal{N}\left(\alpha_1 T + (1 - \alpha_1) O; \sigma_{0T}^2\right),\ \text{if } D = \text{„open“}$

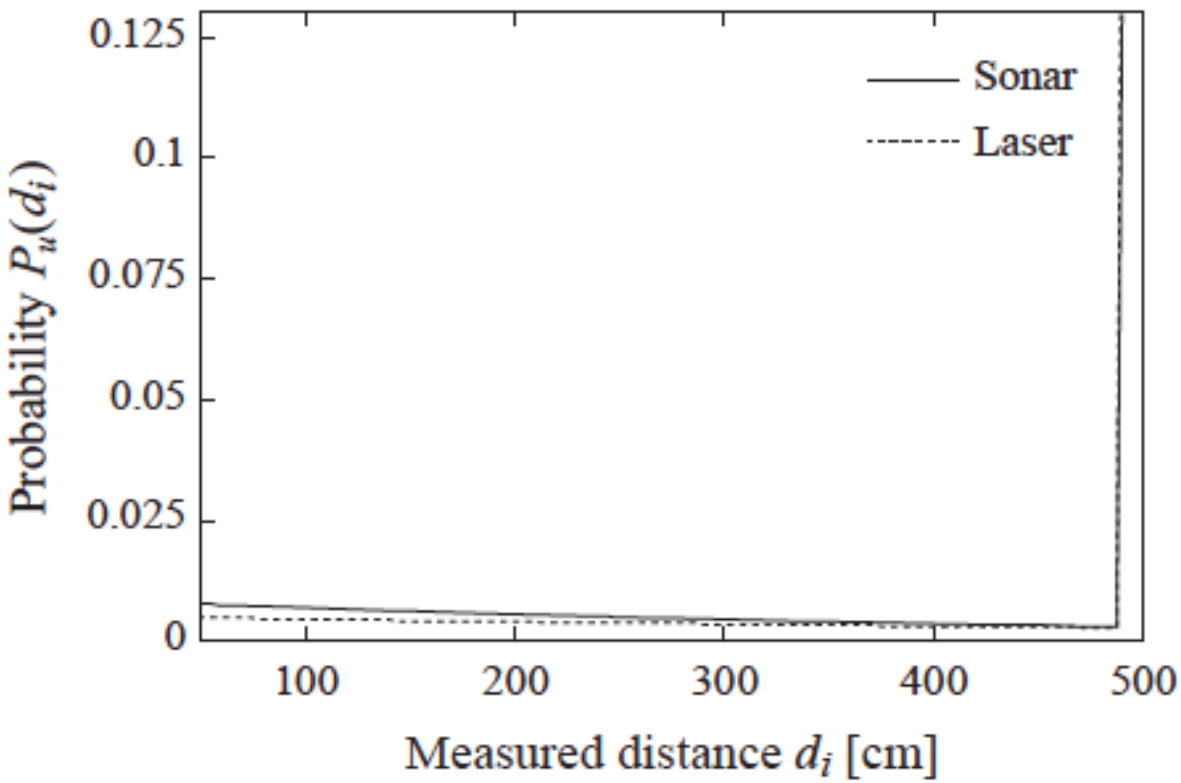
Conditional



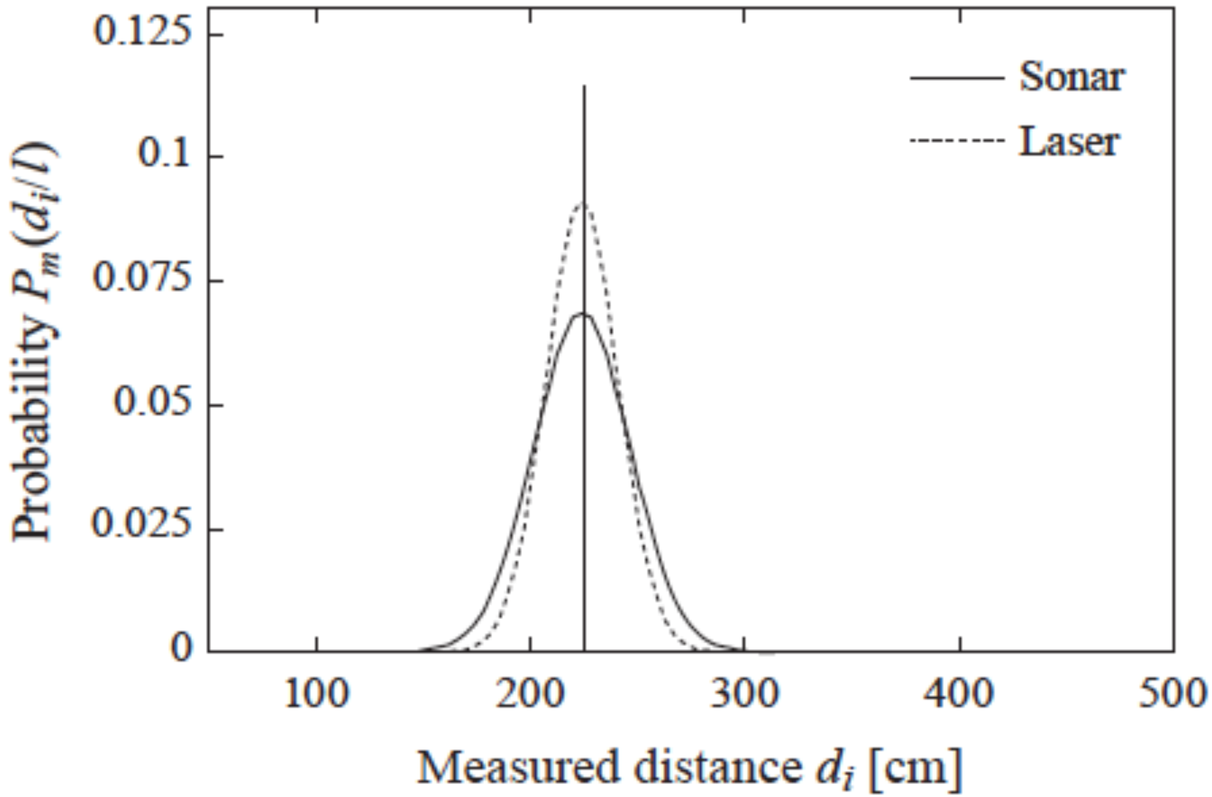
$$Y \sim \mathcal{N}\left(w_{a,0} + \sum_{i \in 1..k} w_{a,i} X_i; \sigma_a^2\right)$$

Robot Localization: Laser Scan and Map

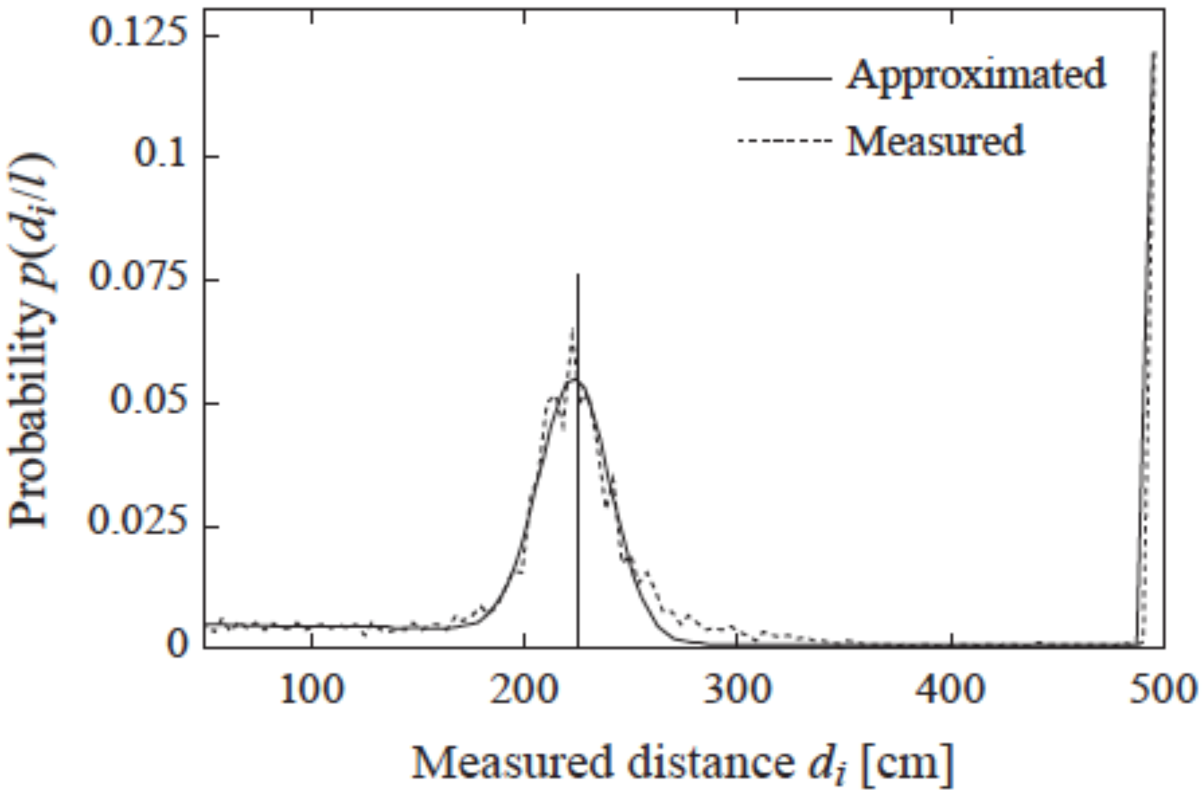
Probability of observing a distance for sonar/laser sensor



Sensor characteristics given distance to object is 2.3m



Overall model given distance to object is 2.3m



Pose Estimation

S. Thrun et al. / Artificial Intelligence 128 (2001) 99–141 105

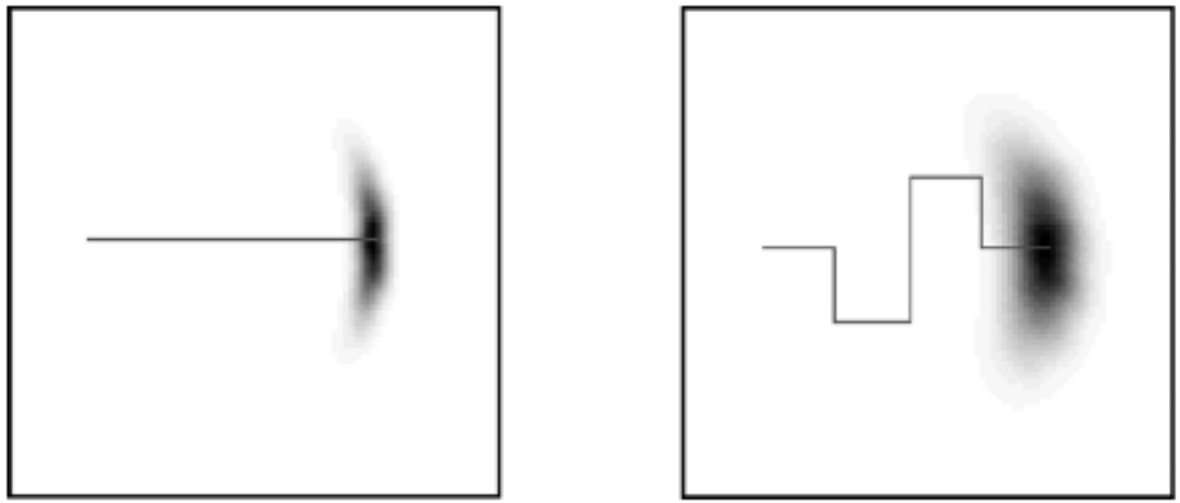


Fig. 1. The density $p(x' \mid x, a)$ after moving 40 meter (left diagram) and 80 meter (right diagram). The darker a pose, the more likely it is.

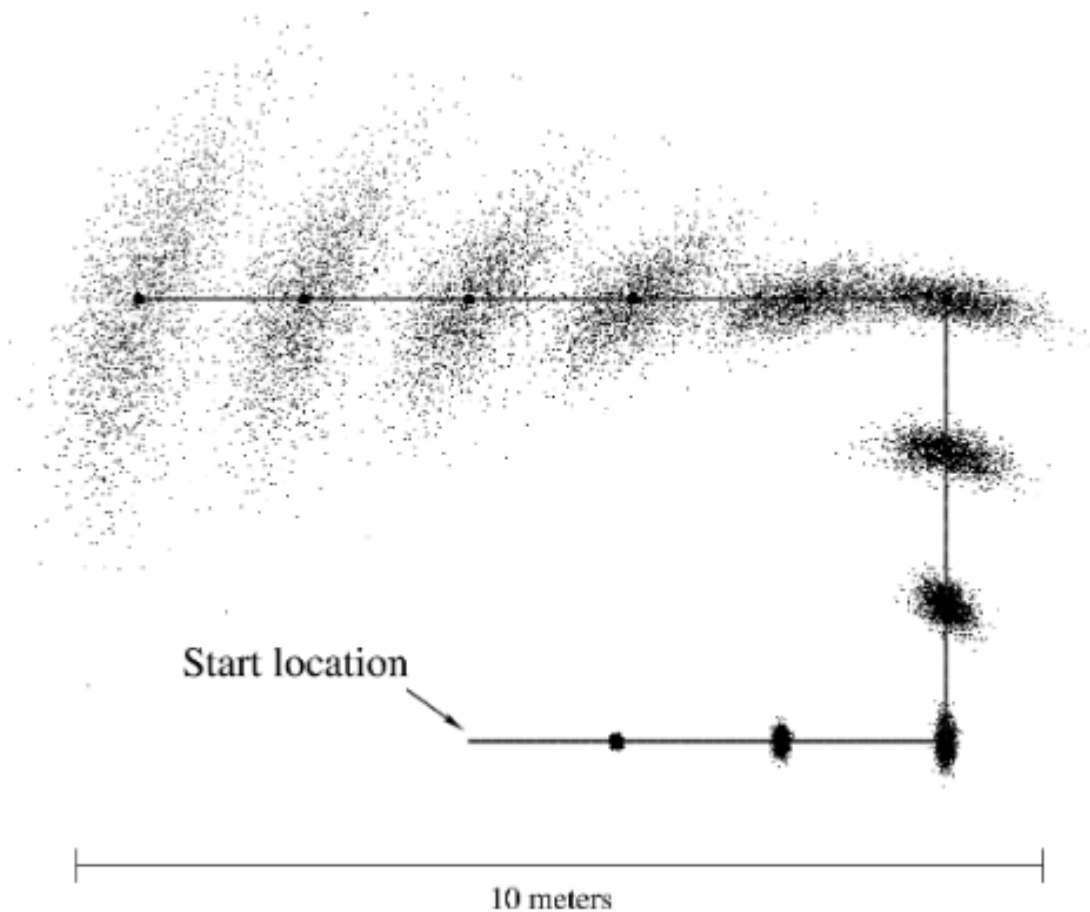
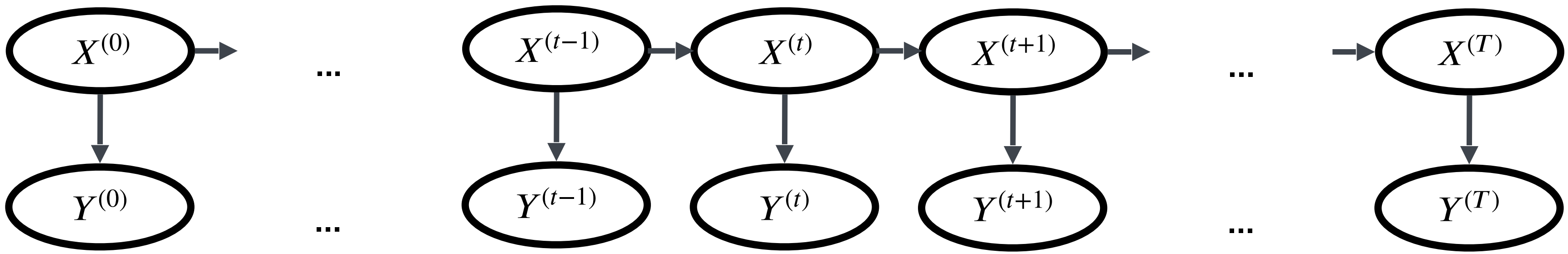


Fig. 2. Sampling-based approximation of the position belief for a robot that only measures odometry. The solid line displays the actions, and the samples represent the robot's belief at different points in time.

Question 3:
Continuous Variables

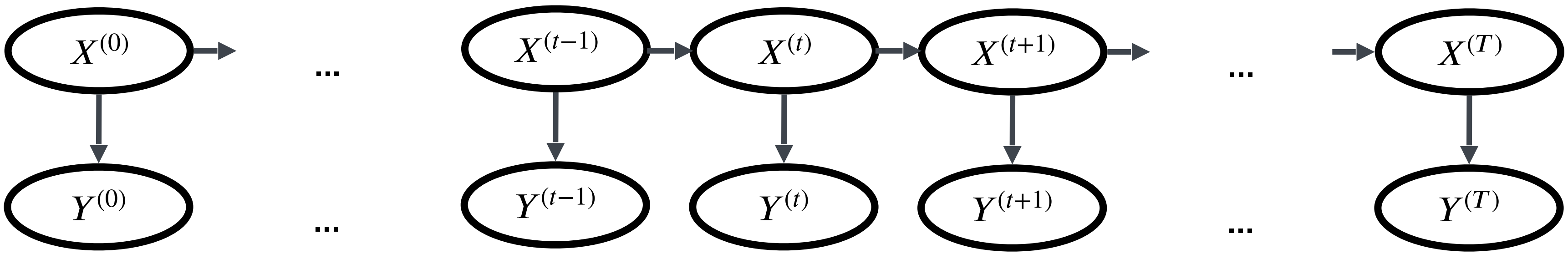
4 Kalman Filter

State-space Model



Kalman Filter

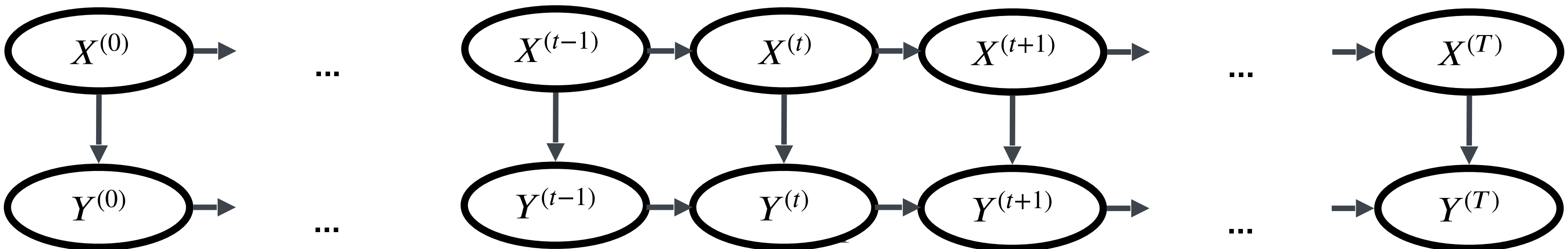
- Linear-Gaussian state-space model



- Assumption:
  - $\boldsymbol{X} = \{X^{(1)}, \dots X^{(n)}\}$ are latent variables with values from a **continuous set of states**
  - $\boldsymbol{Y} = \{Y^{(1)}, \dots Y^{(n)}\}$ are observable variables

Kalman Filter

- Linear-Gaussian state-space model



$$P\left(X^{(0:T)}, Y^{(0:T)}\right) = P\left(X^{(0)}\right) P(Y^{(0)} \mid X^{(0)}) \prod_{t=1} P\left(X^{(t)} \middle| X^{(t-1)}\right) P(Y^{(t)} \mid X^{(t)})$$

Decompose into: $X^{(t)} = f^{(t)}\left(X^{(t-1)}\right) + w^{(t)}$ and $Y^{(t)} = g^{(t)}\left(X^{(t)}\right) + v^{(t)}$

where

- $f^{(t)}, \ g^{(t)}$ are deterministic functions and
- $w^{(t)}, v^{(t)}$ are zero-mean random noise vectors

Linearity and time invariance in state-space model

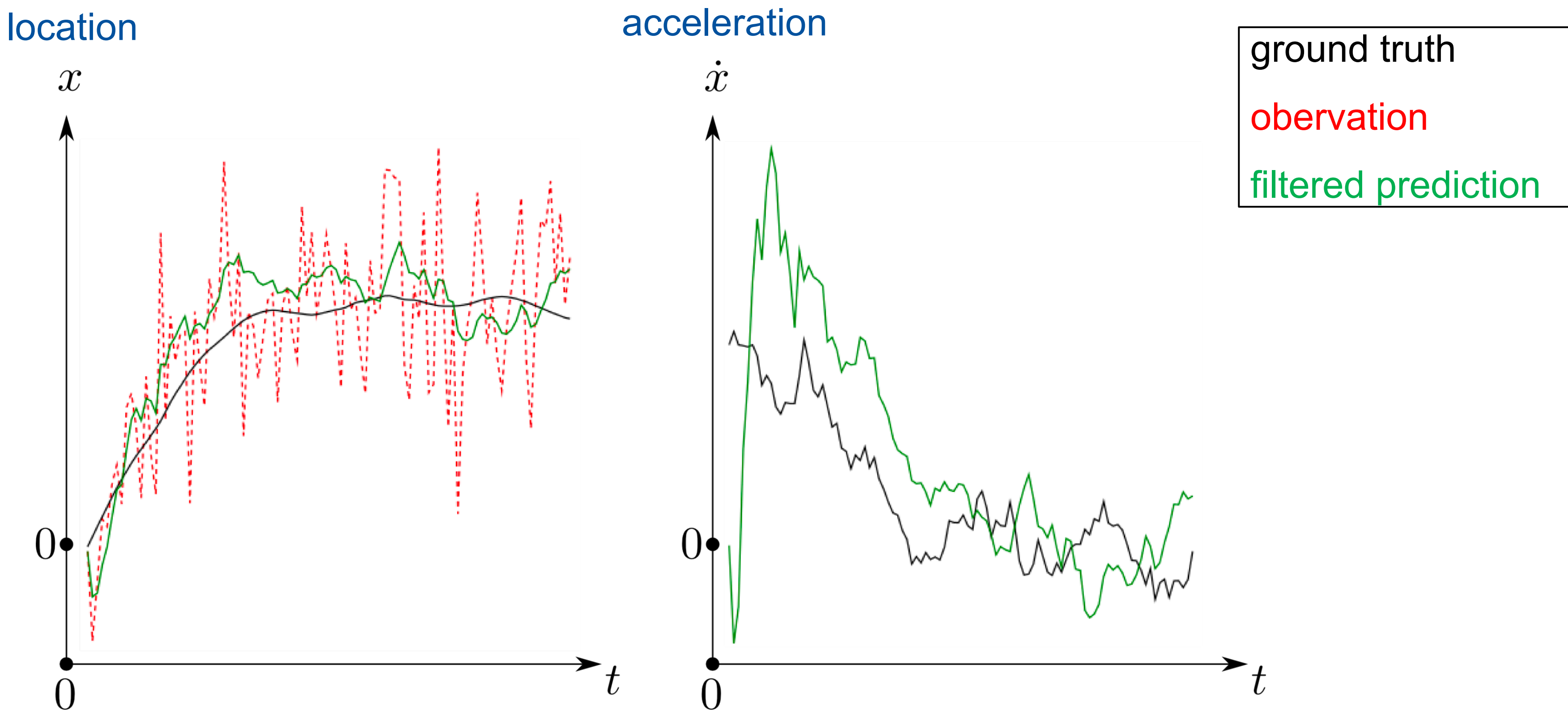
- Assumption:
  - Transition functions $f^{(t)}$ are linear and time-invariant
  - Output functions $g^{(t)}$ are linear and time-invariant
  - Noise variables $w^{(t)}, v^{(t)}$ are Gaussian
- Linear-Gaussian state-space model

$$X^{(t)} = A\, X^{(t-1)} + w^{(t)}$$

$$Y^{(t)} = C\, X^{(t)} + v^{(t)}$$

  - State transition matrix: A
  - Observation matrix: C

Observing and predicting one dimensional movement



https://en.wikipedia.org/wiki/Kalman_filter#/media/File:Kalman.png

Linear state-space model with control

$$X^{(t)} = A\, X^{(t-1)} + B\, U^{(t)} + w^{(t)}$$

$$Y^{(t)} = C\, X^{(t)} + v^{(t)}$$

- Input observation vector: $U^{(t)}$
- State transition matrix: A
- Input matrix: B
- Observation matrix: C

Kalman filter works well if all dynamics are modeled.

Model diverges if dynamics are missing.

Easy to mistake dynamics for noise.
$\Longrightarrow$ Robust Control

Question 4:
Kalman Filter

Universität Stuttgart
IPVS

Thank you!

Steffen Staab

E-Mail Steffen.staab@ipvs.uni-stuttgart.de
Telefon +49 (0) 711 685- To be defined
www. ipvs.uni-stuttgart.de/departments/ac/

Universität Stuttgart
Analytic Computing, IPVS
Universitätsstraße 32, 50569 Stuttgart