Numerical integration for probabilisitc reasoning skripsie

by

Jacobus Martin Louw

Bachelor of Engineering (Electrical and Electronic)

Final Report

Study leader: Dr CE van Daalen

2018

Declaration

By submitting this report electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: . . . .

Contents

List of Figures

List of Acronyms

RV - random variable
PGM - probabilistic graphical model
PDF - probability density function
CPD - conditional probability distribution
KF - Kalman Filter
EKF - Extended Kalman Filter
UKF - Unscented Kalman Filter

Abstract

Text in default language ...

Chapter 1

Introduction

For modern mobile vehicles and robots it is important for them to have the ability to navigate their environment. It is critical for these devices to avoid collisions or dangerous environments. Some robots must move very precisely and therefore should have an accurate reading of their location.

Localisation is essential for robot navigation, but is used in various other applications such missile tracking. For applications like this, it is crucial to have accurate and instantaneous information of the missile's location. Measurements of any object's location will always have some noise, the measured location is thus never 100% accurate. Therefore, one should rather approach the localisation problem in a probabilistic manner and the object's location can be described in terms of a PDF.

For systems with continuous random variables, most of the operations used in probabilistic reasoning use integration. These integrals can be solved analytically in the case of a problem with linear movement. Most systems are nonlinear. The integrals in nonlinear systems cannot be computed analytically and one has to use numerical methods. Commonly-used techniques such as the extended or unscented Kalman filters use primitive numerical integration that are very inaccurate, especially when measurements are also nonlinear. There are several numerical techniques available that are more accurate.

The end goal of this project is to investigate different numerical techniques to solve the nonlinear localisation problem. To reach the end goal, one should have a good understanding of Gaussian random variables, motion and measurement models, and traditional techniques such as the extended - and unscented kalman filters. Also, modeling the problem with a Probabilistic Graphical Model has advantages and is therefore investigated.

A relevant problem has been simulated in Python. Different techniques were implemented and compared in terms of accuracy and efficiency.

Chapter 2

Gaussian Random Variables

The Gaussian or normal distribution is commonly used in probability theory, as it is easy to work with and are remarkably close to real-world distributions. Gaussians make severe assumptions such as exponential decay of the distribution away from its mean and linear interactions between variables. Gaussians are in many cases a good estimation for real world distributions, such as noise, even though they make invalid assumptions. The Gaussian random variable (RV) is a very important concept in this paper as all probability distributions are approximated as Gaussian distributions. Key concepts and features of the Gaussian RV are discussed in this chapter. The canonical form and conditional distributions are very important concepts and will be used in following chapters. The following chapter is based on the work of Peebles and Shi [1] and Koller and Friedman [2].

2.1 Covariance form

2.1.1 Univariate Gaussians

An univariate Gaussian random variable X with mean μ and variance σ^2 is described by

$$p(x) = \mathcal{N}(\mu; \sigma) = \eta \exp \left[\frac{-(x - \mu)^2}{2\sigma^2} \right] \tag{2.1}$$

with normalisation coefficient

$$\eta = \frac{1}{\sqrt{2\pi\sigma^2}} \tag{2.2}$$

The mean parameter μ describes the location of the peak of the distribution and the variance parameter σ^2 describes the tempo which the distribution decays. The standard deviation is denoted by σ. The probability of x decays exponentially as x moves farther from the mean. Figure 2.1 indicates the mean μ and standard deviation σ of an univariate Gaussian PDF.
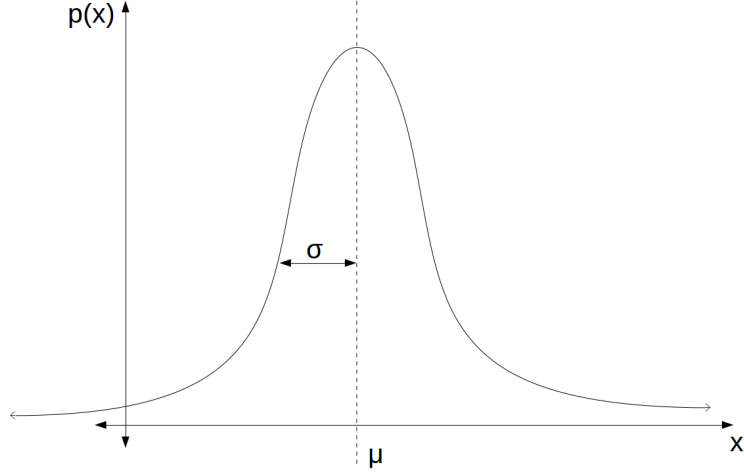
Figure 2.1: Univariate Gaussian PDF

2.1.2 Multivariate Gaussians

The multivariate Gaussian distribution is described by an n-dimensional mean vector $\boldsymbol{\mu}$ and an $n \times n$ covariance matrix Σ. The multivariate Gaussian distribution is commonly described by the following function

$$p(x) = \mathcal{N}(\boldsymbol{\mu}; \Sigma) = \eta \exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \Sigma^{-1}(\boldsymbol{x} - \boldsymbol{\mu})\right] \tag{2.3}$$

with normalisation coefficient

$$\eta = \frac{1}{(2\pi^{n/2)}|\Sigma|^{\frac{1}{2}}} \tag{2.4}$$

$|\Sigma|$ is the the determinant of Σ It is important that the determinant of Σ is nonzero for the equation above to be valid. It is a requirement that Σ is positive definite and therefore also non-singular, which guarantees a determinant that is nonzero. In this paper we focus on positive definite covariance matrices, as it is invertible, consequentially one can use the alternative canonical or information parameterization. The canonical form is discussed in the following section.
The mean and covariance can sequentially be found by calculating the first and second moments of the Gaussian distribution.
The mean is equal to the first moment:

$$\boldsymbol{\mu} = \boldsymbol{E}\left[\boldsymbol{X}\right] \tag{2.5}$$

The covariance is equal to the second moment:

$$\Sigma = \boldsymbol{E}[\boldsymbol{X}\boldsymbol{X}]^T - \boldsymbol{E}[\boldsymbol{X}]\boldsymbol{E}[\boldsymbol{X}]^T \tag{2.6}$$

2.1.3 Error ellipse

Multivariate Gaussian distributions can be visualized as a series of ellipsoidal contours around the mean vector $\boldsymbol{\mu}$. The contours are parallel to each other and the distance between contours suggests the steepness of the density function. The steepness of the density function is determined by the covariance matrix Σ.

In the following chapters it is important to show the uncertainty of PDFs, a plot of an error ellipses is an effective way to illustrate this.

Eigenvalue decomposition

The following section is based on an article from H Abdi [3].

In order to plot an error ellipse, the directions the covariance varies the most and the variances in those directions are needed.

We can use eigenvalue decomposition to factorize the covariance matrix Σ in terms of its eigenvalues Λ and its eigenvectors Q.

$$\Sigma = Q\Lambda Q^{-1} \tag{2.7}$$

Each column of the eigenvector matrix Q contains an eigenvector $\boldsymbol{v}$. The eigenvectors indicate the direction which the covariance varies the most. Each eigenvector has a corresponding eigenvalue λ which specifies the variance in the direction of the eigenvector.

Plotting a 2D error ellipse

The following section is based on a webpage from "Computer vision for dummies" [4].

Error ellipsis are specified in terms of the confidence or probability that a random data point will fall inside the ellipse.

To plot an error ellipse, the major - and minor axes' lengths are sequentially specified as $2\sqrt{s\lambda_1}$ and $2\sqrt{s\lambda_2}$, where λ_1 and λ_2 represent the eigenvalues of the covariance matrix ($\lambda_1 \geq \lambda_2$). For a 95% confident ellipse s is set to 5.991. If we are interested in other confidence intervals, the Chi-Sqaure distribution can be used to find an s that less than or equal to a specific value which correlates with the probability of the confidence interval.

To obtain the orientation α of the error ellipse, we calculate the angle of the largest eigenvector relative to the x-axis.

$$\alpha = \arctan\left(\frac{\boldsymbol{v_1}(y)}{\boldsymbol{v_1}(x)}\right) \tag{2.8}$$

In the case of an uncorrelated covariance, the eigenvalues are equal to the variances of the covariance matrix and the eigenvectors are in the same direction as the x-axis and y-axis. Figure 2.2 shows α and indicates the major and minor axes of the error ellipse.
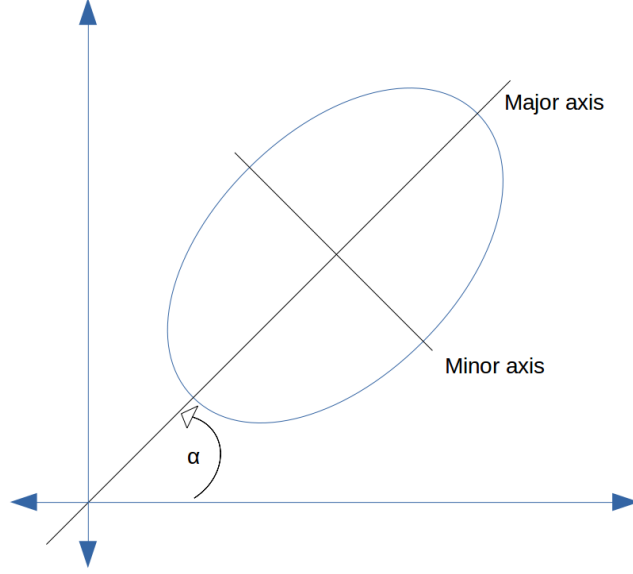
Figure 2.2: Error Ellipse

2.2   Canonical form

Linear Gaussian CPDs (conditional probability densities) are generally not Gaussians. It will be handy to find general representation that accommodates both Gaussian distributions and linear Gaussian models. The canonical or information form is a viable option. It is also much easier to perform certain operations in the canonical form. The following section is based on the work of Koller and Friedman [2] and JC Schoeman [5].

$$\mathcal{C}(\boldsymbol{X}; K, \boldsymbol{h}, g) = \exp\left(-\frac{1}{2}\boldsymbol{X}^T K \boldsymbol{X} + \boldsymbol{h}^T \boldsymbol{X} + g\right) \tag{2.9}$$

It is possible to represent every Gaussian as a canonical form. Equation 2.3 can be rewritten:

$$\eta \exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \Sigma^{-1}(\boldsymbol{x} - \boldsymbol{\mu})\right]$$
$$= \exp\left(-\frac{1}{2}\boldsymbol{x}^T \Sigma^{-1}\boldsymbol{x} + \boldsymbol{\mu}^T \Sigma^{-1}\boldsymbol{x} - \frac{1}{2}\boldsymbol{\mu}^T \Sigma^{-1}\boldsymbol{\mu} + \ln \eta\right) \tag{2.10}$$

$\mathcal{N}(\boldsymbol{\mu}; \Sigma) = \mathcal{C}(K, \boldsymbol{h}, g)$ by comparing 2.10 with 2.9

$$K = \Sigma^{-1} \tag{2.11}$$

$$\boldsymbol{h} = \Sigma^{-1}\boldsymbol{\mu} \tag{2.12}$$

$$g = -\frac{1}{2}\boldsymbol{\mu}^T \Sigma^{-1}\boldsymbol{\mu} + \ln \eta \tag{2.13}$$

$\boldsymbol{h}$ is called the potential vector and K is called the information matrix.

The covariance parameters can again be recovered

$$\Sigma = K^{-1} \tag{2.14}$$

$$\boldsymbol{\mu} = \Sigma\boldsymbol{h} \tag{2.15}$$

The covariance form is not defined when K is not invertible, even though the distribution can still be represented in the canonical form. The canonical form is only a valid Gaussian density if and only if K (information matrix) is symmetric and positive definite. The canonical form is therefore more general than the covariance form. The canonical form is very useful to present linear Gaussian CPDs. From 2.13 it can be seen that only K and $\boldsymbol{h}$ are necessary to calculate g. Thus, g can be omitted when working in the canonical form.

2.2.1 Operations using the canonical form

The main advantage of using the canonical form is that it is very easy to perform various operations on distributions. The variable g is included in the following operations for completeness sake, but can be omitted as stated above. Extending and rearranging scopes of canonical forms may be necessary before doing operations, as it is important that scopes of canonical forms are identical when doing operations.

Extending the scope of a canonical form:

If additional variables are needed, the scope of a canonical form can be extended by adding zero entries to K, $\boldsymbol{h}$ and g.

$$\mathcal{C}\left(\begin{bmatrix} x \\ y \end{bmatrix} : \begin{bmatrix} k_{xx} & k_{xy} \\ k_{yx} & k_{yy} \end{bmatrix}, \begin{bmatrix} h_x \\ h_y \end{bmatrix}, \begin{bmatrix} g_x \\ g_y \end{bmatrix} \right)$$

$$= \mathcal{C}\left(\begin{bmatrix} x \\ y \\ z \end{bmatrix} : \begin{bmatrix} k_{xx} & k_{xy} & 0 \\ k_{yx} & k_{yy} & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} h_x \\ h_y \\ 0 \end{bmatrix}, \begin{bmatrix} g_x \\ g_y \\ 0 \end{bmatrix} \right) \tag{2.16}$$

Rearranging the scope of a canonical form:

The order of the scope can be rearranged by rearranging the rows and columns of K and rearranging the entries of $\boldsymbol{h}$ and g.

$$
\mathcal{C}\left(\begin{bmatrix} x \\ y \\ z \end{bmatrix} : \begin{bmatrix} k_{xx} & k_{xy} & k_{xz} \\ k_{yx} & k_{yy} & k_{yz} \\ k_{zx} & k_{zy} & k_{zz} \end{bmatrix}, \begin{bmatrix} h_x \\ h_y \\ h_z \end{bmatrix}, \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix}\right)
$$
$$
= \mathcal{C}\left(\begin{bmatrix} y \\ x \\ z \end{bmatrix} : \begin{bmatrix} k_{yy} & k_{yx} & k_{yz} \\ k_{xy} & k_{xx} & k_{xz} \\ k_{zy} & k_{zx} & k_{zz} \end{bmatrix}, \begin{bmatrix} h_y \\ h_x \\ h_z \end{bmatrix}, \begin{bmatrix} g_y \\ g_x \\ g_z \end{bmatrix}\right) \quad (2.17)
$$

Multiplication of canonical forms:

When multiplying two canonical forms, it is important that their scopes are identical. Multiplying canonical forms with the same scope is simply:

$$
\mathcal{C}(K_1, \boldsymbol{h_1}, g_1) \times \mathcal{C}(K_2, \boldsymbol{h_2}, g_2) = \mathcal{C}(K_1 + K_2, \boldsymbol{h_1} + \boldsymbol{h_2}, g_1 + g_2) \quad (2.18)
$$

Division of canonical forms:

Again, it important that the scopes of the distributions are identical.

$$
\frac{\mathcal{C}(K_1, \boldsymbol{h_1}, g_1)}{\mathcal{C}(K_2, \boldsymbol{h_2}, g_2)} = \mathcal{C}(K_1 - K_2, \boldsymbol{h_1} - \boldsymbol{h_2}, g_1 - g_2) \quad (2.19)
$$

Marginalization of a canonical form:

A marginal distribution can be found by integrating over a subset of variables. For example the marginal distribution over $\boldsymbol{X}$ can be found by integrating over $\boldsymbol{Y}$.

Let $\mathcal{C}(\boldsymbol{X}, \boldsymbol{Y}; K, \boldsymbol{h}, g)$ be a canonical form with subsets $\boldsymbol{X}$ and $\boldsymbol{Y}$ where

$$
K = \begin{bmatrix} K_{\boldsymbol{XX}} & K_{\boldsymbol{XY}} \\ K_{\boldsymbol{XY}} & K_{\boldsymbol{YY}} \end{bmatrix} ; h = \begin{pmatrix} h_{\boldsymbol{X}} \\ h_{\boldsymbol{Y}} \end{pmatrix} \quad (2.20)
$$

To obtain the marginal distribution over $\boldsymbol{X}$, we have to find the integral over $\boldsymbol{Y}$. Therefore

$$
\int \mathcal{C}(\boldsymbol{X}, \boldsymbol{Y}; K, \boldsymbol{h}, g) d\boldsymbol{Y} = \mathcal{C}(\boldsymbol{X}; K', \boldsymbol{h}', g') \quad (2.21)
$$

where

$$
K' = K_{\boldsymbol{XX}} - K_{\boldsymbol{XY}} K_{\boldsymbol{XY}}^{-1} K_{\boldsymbol{YX}} \quad (2.22)
$$

$$
h' = \boldsymbol{h_X} - K_{\boldsymbol{XX}} K_{\boldsymbol{YY}}^{-1} \boldsymbol{h_Y} \quad (2.23)
$$

$$
g' = g - \frac{1}{2}\left(\ln|2\pi K_{\boldsymbol{YY}}^{-1}| + \boldsymbol{h_Y}^T K_{\boldsymbol{YY}}^{-1} \boldsymbol{h_Y}\right) \quad (2.24)
$$

It is important to that $K_{\boldsymbol{YY}}$ is positive-definite for the result to be finite.

Reduction with evidence:

Let $\mathcal{C}(\boldsymbol{X}, \boldsymbol{Y}; K, \boldsymbol{h}, g)$ be a canonical form with subsets $\boldsymbol{X}$ and $\boldsymbol{Y}$. If subset $\boldsymbol{Y}$ is known ($\boldsymbol{Y} = \boldsymbol{y}$), then the canonical form can be reduced to $\mathcal{C}(\boldsymbol{X}; K', \boldsymbol{h}', g')$, where

$$K' = K_{\boldsymbol{X}\boldsymbol{X}} \tag{2.25}$$

$$h' = \boldsymbol{h}_{\boldsymbol{X}} - K_{\boldsymbol{X}\boldsymbol{Y}}\boldsymbol{y} \tag{2.26}$$

$$g' = g + \boldsymbol{h}_{\boldsymbol{Y}}^T \boldsymbol{y} - \frac{1}{2}\boldsymbol{y}^T K_{\boldsymbol{Y}\boldsymbol{Y}}\boldsymbol{y} \tag{2.27}$$

2.2.2 Using the canonical form the present conditional distributions (linear transformations)

The following section is based on a skripsie from JC Schoeman [5] and notes from Dr CE van Daalen. One of the advantages of the canonical form is that it is possible to present conditional probability distributions (CPDs).
If a CPD such as $p(\boldsymbol{y}|\boldsymbol{x})$ can be described as a linear transform:

$$\boldsymbol{y} = F\boldsymbol{x} + \boldsymbol{g} + \boldsymbol{n} \tag{2.28}$$

where

$$n = \mathcal{N}(\boldsymbol{0}, \Sigma_n) \tag{2.29}$$

$$
\begin{aligned}
p(\boldsymbol{y}|\boldsymbol{x}) &= \mathcal{N}(F\boldsymbol{x} + \boldsymbol{g}, \Sigma) \\
&= \eta \exp\left[-\frac{1}{2}(\boldsymbol{y} - (F\boldsymbol{x} + \boldsymbol{g}))^T \Sigma^{-1}(\boldsymbol{y} - (F\boldsymbol{x} + \boldsymbol{g}))\right]
\end{aligned} \tag{2.30}
$$

Rewriting

$$
\begin{aligned}
\boldsymbol{y} - (F\boldsymbol{x} + \boldsymbol{g}) &= \begin{bmatrix} I & -F & -I \end{bmatrix} \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \\ \boldsymbol{g} \end{bmatrix} \\
&= \begin{bmatrix} I \\ -F^T \\ -I \end{bmatrix}^T \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \\ \boldsymbol{g} \end{bmatrix}
\end{aligned} \tag{2.31}
$$

Equation 2.30 becomes

$$
p(\boldsymbol{y}|\boldsymbol{x}) = \eta \exp\left[-\frac{1}{2}\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \\ \boldsymbol{g} \end{bmatrix}^T \begin{bmatrix} I \\ -F^T \\ -I \end{bmatrix} \Sigma^{-1} \begin{bmatrix} I \\ -F^T \\ -I \end{bmatrix}^T \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \\ \boldsymbol{g} \end{bmatrix}\right] \tag{2.32}
$$

Defining a combined random vector

$$\boldsymbol{w} = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \\ \boldsymbol{g} \end{bmatrix} \tag{2.33}$$

and information matrix

$$K' = \begin{bmatrix} I \\ -F^T \\ -I \end{bmatrix} \Sigma^{-1} \begin{bmatrix} I \\ -F^T \\ -I \end{bmatrix}^T \tag{2.34}$$

Equation 2.32 can be rewritten

$$p(\boldsymbol{y}|\boldsymbol{x}) = \eta \exp\left[-\frac{1}{2}\boldsymbol{w}^T K' \boldsymbol{w} \right] \tag{2.35}$$

This closely relates with the canonical form in equation 2.9

$$
\begin{aligned}
p(y|x) &= \mathcal{C}(\boldsymbol{w} : K', \boldsymbol{0}) \\
&= \mathcal{C}\left(\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \\ \boldsymbol{g} \end{bmatrix} ; \begin{bmatrix} I \\ -F^T \\ -I \end{bmatrix} \Sigma_n^{-1} \begin{bmatrix} I & -F & -I \end{bmatrix}, \boldsymbol{0}, - \right) \\
&= \mathcal{C}\left(\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \end{bmatrix} ; \begin{bmatrix} I \\ -F^T \end{bmatrix} \Sigma_n^{-1} \begin{bmatrix} I & -F \end{bmatrix}, \begin{bmatrix} I \\ -F^T \end{bmatrix} \Sigma_n^{-1}\boldsymbol{g}, - \right) \\
&= \mathcal{C}\left(\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \end{bmatrix} ; \begin{bmatrix} \Sigma_n^{-1} & -\Sigma_n^{-1}F \\ -F^T\Sigma_n^{-1} & F^T\Sigma_n^{-1}F \end{bmatrix}, \begin{bmatrix} -\Sigma_n^{-1}\boldsymbol{g} \\ -F^T\Sigma_n^{-1}\boldsymbol{g} \end{bmatrix}, - \right)
\end{aligned}
\tag{2.36}
$$

Chapter 3

Motion models

Before implementing a localisation algorithm to localise an object, it is important to have a model which desctibes the movement of the object.

Chapter 4

Localisation using Kalman Filters

This chapter will focus on localisation using traditional techniques such as the Kalman filter, extended Kalman filter and Unscented Kalman filter. This is important to investigate and understand as there is a strong correspondance to the PGM technique that will be discussed in a following section. The motion an measurement models will also be discussed

4.1   Motion models

4.2   Measurement models

4.3   Kalman Filter

4.4   Extended Kalman Filter

4.5   Unscented Kalman Filter

Chapter 5

Probabilistic Graphical Models

This chapter gives a brief introduction to Probabilistic Graphical Models (PGMs). PGMs are a very vast field which is beyond the scope of this document, therefore only the theory relevant to this paper will be covered. This chapter is based on the work of Koller and Friedman [2], Barber [6] and Schoeman [5].

5.1 Overview

Reasoning about systems with uncertainty can become very complex and sometimes it is completely unmanageable. The PGM is a graphical technique which allows to construct probabilistic systems in a logical and compact manner which makes problems tractable. PGMs are thus used to describe the relationship between RVs and allows to reason about them. This action of querying a system or reasoning about a variable is called inference. The relationships in a PGM are usually in the form of conditional probability densities (CPDs). It is a very popular technique, as it is intuitive, transparent and easy to manipulate. Hence, PGMs have countless applications from making medical diagnosis to localising robots. Although it is a powerful technique, it has its limitations such as image recognition, where neural networks thrive.

PGMs can be divided in to two categories. The first is Markov networks, which are used for non-causal systems and uses directed graphs. The second is Bayesian networks, which are used for causal systems and uses undirected graphs. As the localisation problem is in most cases causal, the focus of this chapter is on Bayesian networks.

The theory is also discussed in terms of continuous RVs, because problems addressed in following chapters make use of continuous RVs. Note that all of the theory can also be applied to discrete RVs.

5.2 Basic concepts of graphical models

A graph consists of nodes and edges. Edges are the link between nodes and can be undirected or directed. If all edges are undirected then the graph is classified as undirected. If all edges are directed, then the graph is classified as directed.

In the case of directed graphs, nodes can be classified as ancestors, parents, children and descendants. If there exists a directed path from $X_1 \rightarrow X_k$, then X_1 is an ancestor of X_k, and X_k is a descendant of X_1. In Figure 5.1(b), A is an ancestor of C and C is a descendant of A. A parent is an ancestor with only one edge between the ancestor and descendant. A child is a descendant with only one edge between the descendant and the ancestor. In Figure 5.1 (b), then A is a parent of B and B is a child of A. [6]
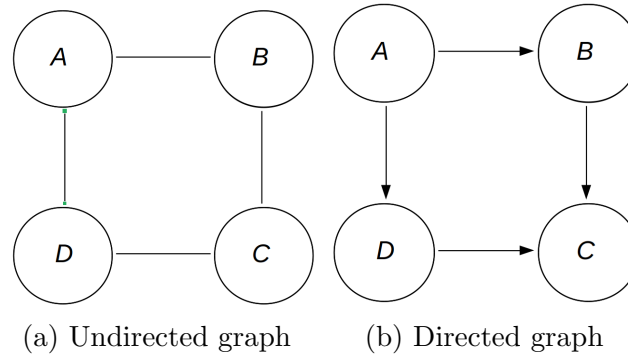

(a) Undirected graph   (b) Directed graph

Figure 5.1

5.3 Bayesian PGMs

Bayesian networks consists out of a set of nodes and directed edges. Nodes represent RVs($X_1, ..., X_n$).

Each node is associated with a conditional probability distribution.

$$X_i \sim p(X_i | Par(X_i)) \tag{5.1}$$

where $Par(X_i)$ indicates the parent of node X_i.

CPDs can be written as factors

$$\phi_i(X_i, Par(X_i)) = p(X_i | Par(X_i)) \tag{5.2}$$

Where the scope of a factor is

$$Scope\{\phi_i\} = \{X_i, Par(X_i)\} \tag{5.3}$$

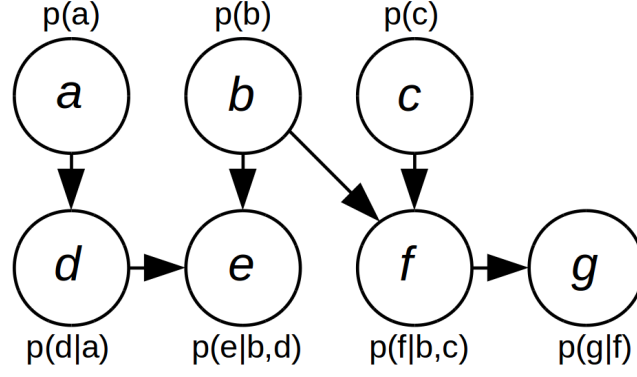Figure 5.2 is a simple example of a Bayesian network.

13

Figure 5.2: Bayesian PGM

Chain Rule for Bayesian Networks

All of the CPDs or factors of a Bayesian network can be multiplied to find the joint distribution of the network. (Koller and Friedman [2])

$$p(X_1, ..., X_n) = \prod_i p(X_i | Par(X_i)) \tag{5.4}$$

Applying the Chain Rule to the Bayesian network in Figure 5.2

$$
\begin{aligned}
p(a, b, c, d, e, f, g) &= p(a)p(b)p(c)p(d|a)p(e|b, d)p(f|b, c)p(g|f) \\
&= \phi_a(a)\phi_b(b)\phi_c(c)\phi_d(d, a)\phi_e(e, b, d)\phi_f(f, b, c)\phi_g(g, f)
\end{aligned} \tag{5.5}
$$

A marginal PDF can be found by integrating over a subset of variables.

$$p(a, b, c) = \int \int \int \int p(a, b, c, d, e, f, g)dd \; de \; df \; dg \tag{5.6}$$

5.4 Cluster graphs

Cluster graphs can be generated from Bayesian networks and are used to do inference. The cluster graph consists of clusters and undirected edges. Clusters are subsets of variables and can be formed by grouping factors together.

$$\boldsymbol{C}_i \subseteq \{X_1, ..., X_n\} \tag{5.7}$$

The undirected edges between clusters are called sepsets and are responsible for passing messages between clusters. A sepset between two clusters contains information about variables that are common to both clusters.

$$\boldsymbol{S}_{i,j} \subseteq \boldsymbol{C}_i \cap \boldsymbol{C}_j \tag{5.8}$$

There are multiple ways to construct clusters, but it should adhere to two requirements specified Koller and Friedman [2].

1. Family Preservation
Every factor ϕ_k in a set of factors Φ should be accommodated by a cluster.

$$Scope[\phi_k] \subseteq \boldsymbol{C}_i \tag{5.9}$$

2. Running Intersection Property
There exists an unique path connecting a pair of clusters containing the same variable X, and every cluster and sepset along the path also contain X. This path allows clusters to share their beliefs of X. In other words, for any variable X, the set of sepsets and clusters containing X form a tree [2]. This prevents feedback loops and thus counters the phenomena where cluster's reinforce their own beliefs of variables.

Figure 5.2 is updated in Figure 5.3 where CPDs is replaced with factors and potential Cluster boundaries are indicated with dashed rectangles. Note that there are other ways to configure the cluster graph.
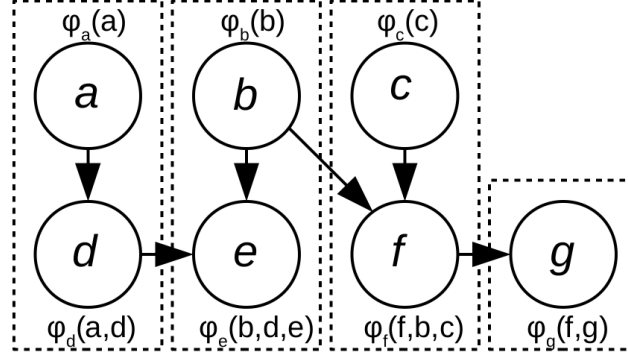

Figure 5.3: Bayesian PGM with cluster boundaries

The initial belief of an cluster can be calculated by finding the product of all the factors inside the cluster. [2]

$$\psi_i(C_i) = \prod_{k:\alpha(k)=i} \phi_k \tag{5.10}$$

Figure 5.3 is updated in Figure 5.4. Clusters are indicated by ellipsis. Initial beliefs are calculated and shown inside the ellipsis. Sepsets are indicated by rectangles.
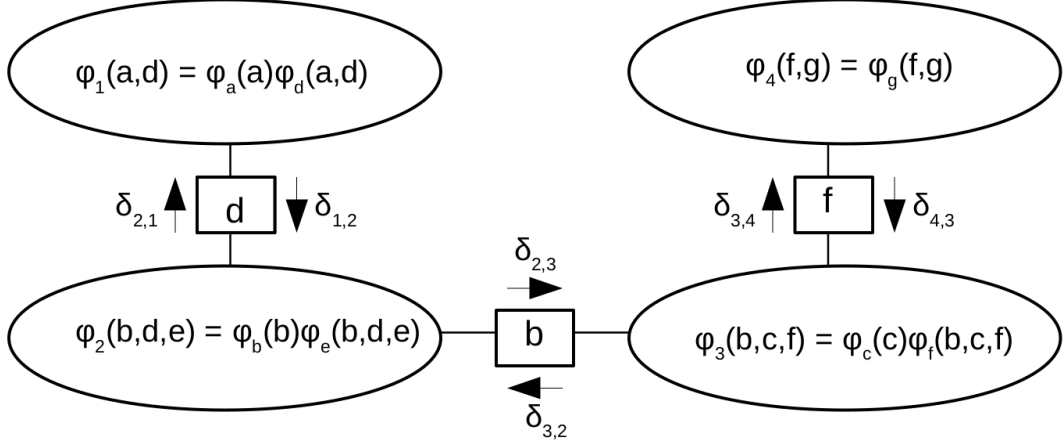
Figure 5.4: Cluster graph

5.5 Message Passing

Clusters share beliefs of common variables with messages passing. Messages are associated with sepsets. An outgoing message of a cluster can be calculated by multiplying the cluster with all the incoming messages and then integrating over the variables that are not in the sepset.

Thus, messages can be calculated accordingly [2]

$$\delta_{i \to j}(\boldsymbol{S}_{i,j}) = \int_{\boldsymbol{C}_i - \boldsymbol{S}_{i,j}} \psi_i \times \prod_{k \in (\mathcal{N}_i - \{j\})} \delta_{k \to i} \tag{5.11}$$

Applying to the cluster graph in Figure 5.4

$$\delta_{1,2}(d) = \int_a \psi_1(a, d) da \tag{5.12}$$

$$\delta_{2,3}(b) = \int_d \int_e \delta_{1,2}(d) \psi_2(b, d, e) dd \ de \tag{5.13}$$

$$\delta_{3,4}(f) = \int_b \int_c \delta_{2,3}(b) \psi_3(b, c, f) db \ dc \tag{5.14}$$

$$\delta_{4,3}(f) = \int_g \psi_4(f, g) dg \tag{5.15}$$

$$\delta_{3,2}(b) = \int_c \int_f \delta_{4,3}(f) \psi_3(b, c, f) dc \ df \tag{5.16}$$

$$\delta_{2,1}(d) = \int_b \int_e \delta_{3,2}(b) \psi_2(b, d, e) db \ de \tag{5.17}$$

5.6 Inference: Variable Elimination

There are many different algorithms for inferencing graphical models. A popular one is that of variable elimination. Consider Figure 5.2 again. The belief $p(f, g)$ can be obtained by finding the marginal distribution of f and g from the joint distribution of the Bayesian network.

$$p(f, g) = \int_a \int_b \int_c \int_d \int_e p(a, b, c, d, e, f, g) \, da \, db \, dc \, dd \, de \qquad (5.18)$$

With the chain rule, the joint distribution can be rewritten as the product of clusters defined in Figure 5.4.

$$p(f, g) = \int_b \int_c \int_d \int_e \int_a \phi_a(a)\phi_b(b)\phi_c(c)\phi_d(d, a)\phi_e(e, b, d)\phi_f(f, b, c)\phi_g(g, f) \, da \, db \, dc \, dd \, de$$
$$= \int_b \int_c \int_d \int_e \int_a \psi_1(a, d)\psi_2(b, d, e)\psi_3(b, c, f)\psi_4(f, g) \, da \, db \, dc \, dd \, de$$
$$(5.19)$$

Factors with variables that are not in the scope of an integration can be moved out of the integration. Thus, in equation 5.20 ψ_2, ψ_3 and ψ_4 can be moved out the integral over a, as they do not contain a.

$$p(f, g) = \int_b \int_c \int_d \int_e \left[\int_a \psi_1(a, d) \, da \right] \psi_2(b, d, e)\psi_3(b, c, f)\psi_4(f, g) \, db \, dc \, dd \, de$$
$$(5.20)$$

The result in equation 5.12 can be used to make a substitution

$$p(f, g) = \int_b \int_c \int_d \int_e \delta_{1,2}(d)\psi_2(b, d, e)\psi_3(b, c, f)\psi_4(f, g) \, db \, dc \, dd \, de \qquad (5.21)$$

Steps in equation 5.20 and 5.21 are repeated in equation 5.22 to 5.25.

$$p(f, g) = \int_b \int_c \left[\int_d \int_e \delta_{1,2}(d)\psi_2(b, d, e) \, dd \, de \right] \psi_3(b, c, f)\psi_4(f, g) \, db \, dc \qquad (5.22)$$

$$p(f, g) = \int_b \int_c \delta_{2,3}(b)\psi_3(b, c, f)\psi_4(f, g) \, db \, dc \qquad (5.23)$$

$$p(f, g) = \left[\int_b \int_c \delta_{2,3}(b)\psi_3(b, c, f) \, db \, dc \right] \psi_4(f, g) \qquad (5.24)$$

$$p(f, g) = \delta_{3,4}(f)\psi_4(f, g) \qquad (5.25)$$

The belief of a cluster can be calculated by multliplying the cluster with all of its' incoming messages. Koller and Friedman [2] specifies this as

$$\beta_i(\boldsymbol{C}_i) = \psi_i \times \prod_{k \in \mathcal{N}_i} \delta_{k \to i} \qquad (5.26)$$

5.7   Summary

Bayesian networks consists of factors and directed edges.
Cluster graphs can be constructed by grouping factors.
Clusters in a cluster graph are connected by sepsets.
Adjacent clusters pass information to each other about variables in sepset.
Outgoing messages can be calculated with equation 4.11.

Chapter 6

Localisation using Probabilistic Graphical Models

6.1   Linear localisation

6.2   Non-linear localisation

6.2.1   Taylor series expansion

6.2.2   Unscented transform

6.2.3   Monte Carlo integration

List of References

[1] Peebles, P.Z.: *Probability, random variables, and random signal principles*, vol. 3. McGraw-Hill New York, NY, USA:, 2001.

[2] Koller, D., Friedman, N. and Bach, F.: *Probabilistic graphical models: principles and techniques.* MIT press, 2009.

[3] Abdi, H.: The eigen-decomposition: Eigenvalues and eigenvectors. *Encyclopedia of measurement and statistics*, pp. 304–308, 2007.

[4] Visiondummy, how to draw a covariance error ellipse. `http://www.visiondummy.com/2014/04/draw-error-ellipse-representing-covariance-matrix/`. Accessed: 2018-10-01.

[5] Schoeman, J.: *Simultaneous Localisation and Mapping of a Robotic Vehicle using a Probabilistic Graphical Model.* Stellenbosch University, 2016.

[6] Barber, D.: *Bayesian reasoning and machine learning.* Cambridge University Press, 2012.