VMG - Mobile Application

Velocity Made Good - supporting sailors to find best driving angles

Master Thesis

Jan-Jasper Wagner
Matricle number: 868948

Tutor Prof. Dr. Amy Siu

Contents

List of Symbols

Allgemeine Symbole

Symbol	Bedeutung
a	der Skalar a
$\vec{x}$	der Vektor $\vec{x}$
$\mathbf{A}$	die Matrix $\mathbf{A}$

1 Introduction

The aim of the thesis is to provide a tool for sailors to find the best driving angle compared to the wind direction in order to sail in the boat in the most efficient way. A sailing boat is not able to sail against the wind. So there must be an angle between the wind direction and the propagation direction of the boat. A different angle to the wind result in a different boat speed. It can only sail against towards wind up to a certain angle. This is possible for both sides of the boat. It doesn't make a difference if the wind come form the left (port tack) or from the right (starboard tack) the best angle towards the wind will remain the same. There is a dependency between boat speed and driving angle. This dependency is called velocity made good and the goal of the thesis is to develop an Android Mobile Application that's helps sailors to find the best VMG. The VMG can be calculated if a boat sails upwinds (Against the wind) or downwind (down the wind). Further details will follow in the upcoming chapters.Nowadays the technical state of smartphone has come to an level which allows mobile developers to create valuable, high functional and reliable mobile applications in a convenient way at low prices. Almost everyone owns a mobile device. So there is no additional and expensive equipment necessary to benefit from great products.The main purpose of the thesis will be collecting positioning data in order to calculate the current velocity and heading of the boat in the most accurate and highest possible frequency. The most common data source and in most cases most accurate is GPS data source.The challenge will be to provide trustful results from relatively inaccurate data streams. Just calculating the distance between two positions won't lead to satisfying results.The accuracy of GPS points can be influenced by buildings, big trees, various weather conditions, and the configuration of the satellites, also different GPS-receivers produce different errors. [2]In order to achieve reasonable results research has shown that the most common solution for this issue of correct positioning and noise is the application of several filters on the raw GPS data. The main role will take the Kalman Filter. With the aid of those filters it is the aim to come close to the true value. In this thesis the task will be evaluating which filters in which combination will provide the best results.

2 Sailing basics

As mentioned already in the introduction for a sailing boat it is not possible to sail against the wind. There is a limit angle-wise towrds the wind until the boat is still going forward. If the boat is exceeding this angle the boat will start sailing backwards. Actually sailors are not playing around this crucial point where the boat would start sailing backwards or slightly forward. The aim of every sailor is to point the sailboat as close into the wind as possible while still keeping the winds blowing across the sails in a manner that provides aerodynamic lift which then propels the boat. The average angle is around 45°. By definition the boat is sailing then on a upwind course.The sailor can turn slightly away from the wind to create more forward wind pressure on the sails, which allows it to move with greater speed, but less directly toward the wind (or mark). So there is strong dependency between driving angle, boat speed and wind speed as shown in figure 1. So it is some kind of extremum problem to find the best compromise between driving angle and speed.It is important to point out that different boat types have different upwind angles to the wind. The wind strength also plays a role in this context but will not be further distinguishedas it is not important for the processing of the data.

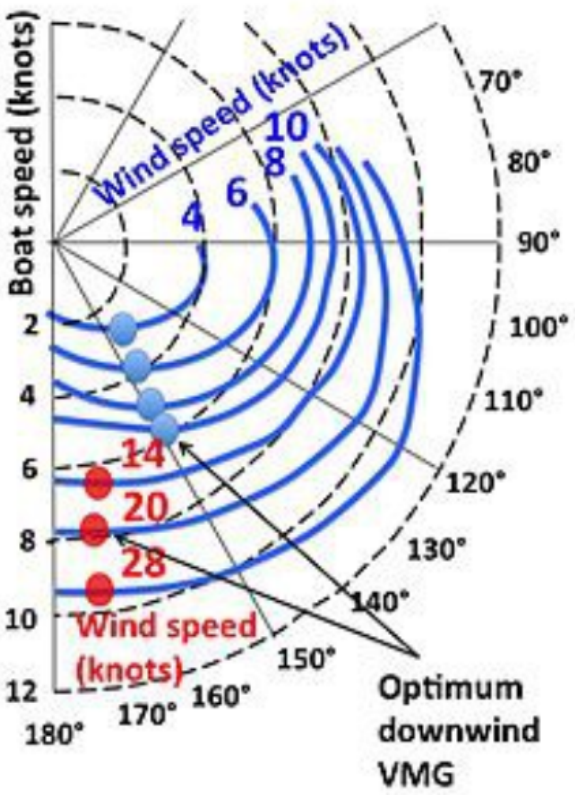

Figure 2.1: Polar diagram of a sailing boat. The diagram shows the boat speed for different angles and windspeed

3 Definition of VMG

At optimum VMG, steering closer to the direction of the wind will reduce boat speed, while steering further away from the direction of the wind might give a higher boat speed, but at the cost of a larger deviation in heading, so less progress towards a mark.As mentioned already in the previous chapter the sailor can slightly turn away from the wind to gather more speed. But the downside of creating more speed is the increasing angle to the wind. So there is a trade-off between speed and progress towards the wind which is definded by the vmg which is dereived by basic trigonometry as follows:

$$VMG = v * \cos a \tag{3.1}$$

with:
v: current speed of the boat
a: current angle towards the wind

4 State of the scientic knowledge

It's a great time for location applications because technology hardware standards and Android application programming interfaces (APIs) are all evolving simultaneously to enable an improved location accuracy that has not previously been possible when using smartphones.The propriety of VMG is strongly dependent on the amicability of the GPS reading.By default, the Android SDK offers the Location API. However, this API is not really optimized to save battery life. So, Google has created the Fused Location Provider API integrated in the Google Play Services.

4.1 Fused location provider Api

The Fused location provider API is a simple and battery-efficient location API for Android with improved location updates with higher accuracy. As the name reveals it is fusing location data from more sources like sensors, wifi, context,and history and evalutes by default which source provides the better accuracy. But there isalso a downside of the new fused location provider api. It dosen't give you information about the source's origin, wether it comes from GPS or network. That can lead to jumps in the received location data but it turned out it is very rare that the origin of the data is jumping between the two different sources

4.2 Sensor fusion

Modern smartphones have a bunch of sensors such as an accelerometer, magnetometer, and gyroscope. It means that theoretically, we have one more source of position data, which will play an important role for applying a kalman filter and lead to much better results if you would depend only on one source. For the project two data sources will be used, GPS and accelerometer.

4.2.1 GPS data

Global Positioning System receivers calculate their locations by analyzing signals that theyreceive from satellites. These signals don't pass through solid objects. A GPS in a vehicle may have an external antenna, or it may pick up enough of bounced signal out of the air tooperate. If signals in a tunnel are too weak, the GPS may still function, depending on its quality and features.GPS coordinates are not very accurate, but on the other hand each of them doesn't depend on previous values. So, there is no accumulation error in this case. [Dev20]

GPS Accuracy

You might have noticed that your phone seems to be more accurate when you're inside shopping malls and office blocks than it was a few years ago, you're not imagining it. With each release of the fused location provider, we have had steady improvement of the Android algorithms and machine learning for Wi-Fi locations.If you're outside and can see the open sky, the GPS accuracy from your phone is about five meters, and that's been constant for a while. But with raw GNSS measurements from the phones, this can now improve, and with changes in satellite and receiver hardware, theimprovements can be dramatic. Unfortunately within the application context it is not possible to fall back using GNSS measurements as this is only possible with a network connection which have RTT-capable access points. [FVDW18]

GPS Provider-Receiver communication - Carrier-phase precision

Carrier-phase precision has been in commercial GPS receivers since the 1980s. What is new is the availability of these carrier-phase measurements from phones and dual-frequency measurements in phones. Right now, all smart phones have GPS or GNSS on one frequency band only. It's known as L1. But there'sa new frequency in town called L5, and it's supported by all these GNSS systems: GPS, Galileo, BeiDou QZSS and IRNSS. The availability of a second frequency means that you get much faster convergence to carrier-phase accuracy. [FVDW18]

4.2.2 Accelorometer

On the other hand, the accelerometer has very accurate GPS readings, but it accumulates error related to noise and integration error. Therefore, it is necessary to fuse these two sources. [Dev20]

4.3 Kalman Filter

The Application and set up of an Kalman Filter will play the role for this project. Kalman filters are ideal for systems which are continuously changing. Kalman filtering, also known as linear quadratic estimation (LQE), is an algorithm that uses a series of measurements observed over time, containing statistical noise and other inaccuracies, and produces estimates of unknown variables that tend to be more accurate than those based on a single measurement alone, by estimating a joint probability distribution over the variables for each time frame. [Dev20] Kalman filters can be used in any place where you have uncertain information about some dynamic system, and you can make an educated guess about what the system is going to do next. That could be any kind of data like the amount of fluid in a tank, the temperature of a car engine, the position of a user's finger on a touchpad, or any number of things you need to keep track of. Even if messy reality comes along and interferes with a clean motion, the Kalman Filter will often do a very good job of figuring out what actually happened. And it can take advantage of correlations between unexpected phenomena which seem impossible to exploit of. The ‚magic' a Kalman Filter ist hat it can have several inputs from different sensors. The GPS sensor tells us something about the state, but only indirectly, and with some uncertainty or inaccuracy. Our prediction tells us something about how the object is moving, but only indirectly, and with some uncertainty or inaccuracy. If all parameters are set up well, with the combination of all inputs, the filter will be able to give extraordinary results close to true values.

4.3.1 Kalman Filter parameter and definitions

In order to understand the functionality of the Kalman Filter, it is obligatory getting familiar with some fundamental mathematical definitions.

Variance

The variance σ^2 measures the spreading of the data set, so we get the deviation from its mean μ.

$$\sigma^2 = \frac{1}{N} * \sum_{i=1}^{N} (x_n - \mu)^2 \tag{4.1}$$

Standard deviation

The standard deveation is definded as the square root of the variance.

$$\sigma = \sqrt{\frac{1}{N} * \sum_{i=1}^{N} (x_n - \mu)^2} \tag{4.2}$$

Covariance

In the context of the thesis, there will be not only one parameter with variance so it is neccessary to take the covariance into count. The covariance of two random variables is important as an indicator of the relationship between them. In this case how are postion error x and velocity error v correlated.

$$Cov(x, v) = \frac{\sum (x_i - \mu_x)(v_i - \mu_v)}{N} \tag{4.3}$$

Estimate

Estimate is about evaluating the hidden state of the system. Every measured or computed parameter is an estimate.

Accuracy

Accuracy indicates how close the measurement is to the true value. The accuracy which is needed for the kalman filter has two dimensions. There is position and velocity accuracy.

Precision

Precision describes how much variability there is in a number of measurements of the same parameter. A high precision doesn't automatically come along with a high accuracy as the following figure shows.

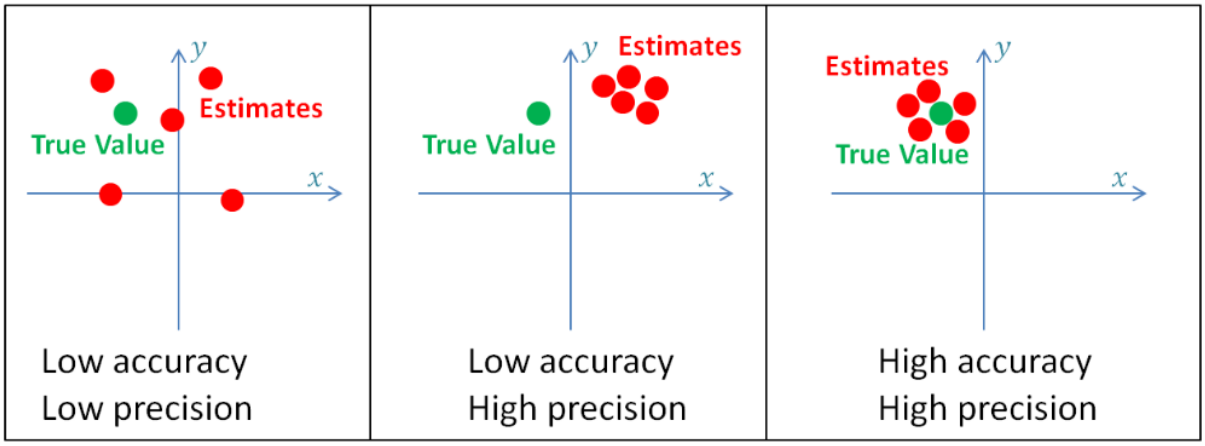

Figure 4.1: An graphical representation of precison and accuracy of a magnitude

4.3.2 One-dimensional kalman filter

After understanding the basic mathematical terms which come into place for kalman filter parameters, an easy example can be shown. An easy and applicitory example is evaluating an airplanes velocity by evaluating measurements given by a radar. For simplicity it is assumed that the aircraft is moving only in one diemension (e.x x direction). To evaluate the airplane's velocity two positions will be tracked of the aircraft at time n-1 and time n with a time difference of Δt. Assuming constant velocity the system's dynamic model can be described by two equations of motion:

$$x_{k+1} = x_k + \Delta t * x_k \tag{4.4}$$

$$x_{k+1} = x_k \tag{4.5}$$

The first equation is called state extra polation equation (or transition equation or prediction equation) and is one of the five Kalman filter equations. (one-dimensional) This system of equation extrapolates the current state to the next state which ist the predicted state

done by the kalman filter. For the new predicted state the previous state will be take into account together with a new measured data point, if t is the first prediction – the previous state will be the initial state otherwise it will be the previous predicted state. The 'magic' behind the Kalman filtering is that the so called Kalman Gain KG. It determines the weight of the measured new value for the predicted state. So it either puts the importance in the estimate or the measurement. That means on the other hand that the value of Kalman Gain always lies between 0 and 1 and is define as:

$$KG = \frac{E_{est}}{E_{est} + E_{mea}} \tag{4.6}$$

The Kalman Gain will decrease if the readings (measurements) match the predicted system state. As the Kalman filter is an iterative system, also the Kalman Gain will be estimated after every new data input again. After examining the Kalman Gain the formula of the the new estimated state EST_t at time t can be defined after every iteration by:

$$EST_t = EST_{t-1} + KG * \left[MEA - EST_{t-1}\right] \tag{4.7}$$

Ideally after some iterations the estimate EST_t at time t will converge to the true values as figure 2 shows

4.3.3 two-dimensional Kalman Filter

Settig up a two-dimesional Kalman Filter is a little more complex process. The procedure mentioned with multiplying or adding the mean values and variances thus only works in the one-dimensional case. In multi-dimensional problems, mean and the variance have to be represented by matrices on which all the operations are performed. That is, when the state you want to measure can be fully described with just one variable. For a complete iteration seven steps are necessary to predict a new state. This results in a Kalman Filter with the following state variables.

$$x = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix} \tag{4.8}$$

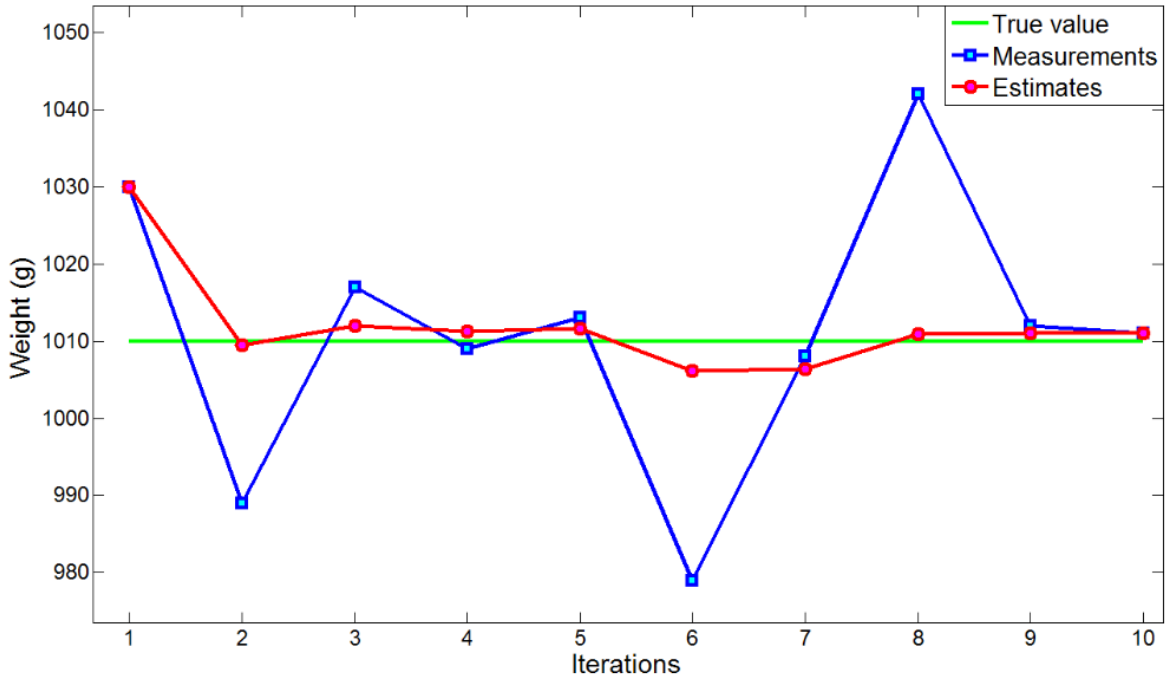

Figure 4.2: Progress of a Klaman filtering process. If the Kamlan Filter parameters are set correctly the Estimate by the Kalman Filter ideally will get closer and closer to the true value.

System state X

At the beginning we will have to set the initial state. In the one dimensional case the state was a vector. If nothing is known about postion or velocitiy, all entries can be set to zero. If some boundary conditions are already known, they can be communicated to the filter. The choice of the following covariance matrix controls how fast the filter converges to the correct (measured) values.

Covariance process matrix P

An uncertainty must be given for the initial state (P_{k-1}). The covariance matrix consists of uncertainty in the position and the velocity in the x and y coordinates. The Matrix will be initialized on basis of the sensor accuracy. Most likely the values will change during every iteration and are changed in both the predict and correct steps. For initial set up, if the sensor is very accurate, small values should be chosen. If the sensor delivers relatively inaccurate measurements, large values should be used to allow the filter to converge relatively quickly. If there are no values available for the inital set it's

recommended to start with a relativley high value like 10. In context of the app development x and y don't have a particluar dependency which means cross-terms can be set to 0.

$$P = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix} \tag{4.9}$$

Dynamic matrix A

The dynamic matrix A comes into place at kind of the core of the filter. From it, the predicted state of each step is derived. As mentioned in the introduction, in the case of tracking a sailing boat's speed, we assume that the boat propagates with smooth motion and constant speed, as the speed differences are particular small. For the state matrix shown above, the dynamics in matrix notation is as follows:

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.10}$$

with:
v: current speed of the boat
a: current angle towards the wind

This states where the state vector moves from one calculation step to the next. The new state (prediction step) is derived from the previous state multiplicated with the dynamics matrix A.

$$\begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix}_{t+1} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix}_{t} \tag{4.11}$$

prediction step for the State matrix.

Control Matrix B and control input $\vec{u_k}$

There might be some changes that aren't related to the state itself. The outside world could be affecting the system. For example, in case of sailing, a big influences might be caused by waves, which have a force effect on the boat. Or the speed of the sailboat changes caused by small driving mistakes by the sailor or adjusting the sails not in a proper way, which all lead to small accelerations. Those iritations of the state transition will be put into the control vector $\vec{u_k}$, and add it to our prediction as a correction. For very simple systems with no external influence, those influences can be omitted. The controll Matrix B is needed to contribute the changes to the filter as follows:

$$B * \vec{u_k} \tag{4.12}$$

$$\begin{bmatrix} \frac{1}{2}\Delta t^2 & 0 \\ 0 & \frac{1}{2}\Delta t^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} * \begin{bmatrix} a_x \\ a_y \end{bmatrix} \tag{4.13}$$

Process noise covariance matrix Q

Everything is fine if the state evolves based on its own properties. Everything is still fine if the state evolves based on external forces, so long as we know what those external forces are. But what about forces that we don't know about? We can't keep track of these things, and if any of this happens, our prediction could be off because we didn't account for those extra forces. We can model the uncertainty associated with the "world" by adding some new uncertainty after every prediction step. So Q contributes to the overall uncertainty of the system caused by any accelerations force from outside. If an acceleration now affects the system state, then the physical dependence for it is Q. When Q is large, the Kalman Filter tracks large changes in the data more closely than for smaller Q. The matrix is a co-variance matrix containing the following elements:

$$Q = \begin{bmatrix} \sigma_x^2 & \sigma_x\sigma_y & \sigma_x\sigma_{\dot{x}} & \sigma_x\sigma_{\dot{y}} \\ \sigma_y\sigma_x & \sigma_{y^2} & \sigma_y\sigma_{\dot{x}} & \sigma_y\sigma_{\dot{y}} \\ \sigma_{\dot{x}}\sigma_x & \sigma_{\dot{x}}\sigma_y & \sigma_{\dot{x^2}} & \sigma_{\dot{x}}\sigma_{\dot{y}} \\ \sigma_{\dot{y}}\sigma_x & \sigma_{\dot{y}}\sigma_y & \sigma_{\dot{y}}\sigma_{\dot{x}} & \sigma_{\dot{y^2}} \end{bmatrix} \tag{4.14}$$

In most cases this matrix becomes to a diagonal matrix as in most cases there is not dependency between x and y.

Measuring matrix H

The filter must also be told what is measured and how it relates to the state vector. In the example of the vehicle, the car enters a tunnel with only position measured at first point, only the speed is measured! The values can be measured directly with the factor 1.0 (i.e. the velocity is measured directly in the correct unit), which is why only 1.0 is set to the appropriate position.

$$H = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.15}$$

If the sensors measure in a different unit or the size by detours, the relationships in the measuring matrix must be mapped in a formula.

Measurement noise covariance matrix R

The measurement uncertainty indicates how much trust can be put into the measured values of the sensors. Since the position and the velocity is measured, R is a 2×2 matrix. If the sensor is very accurate, small values should be placed here. If the sensor is relatively inaccurate, large values should be applied.

$$R = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix} \tag{4.16}$$

Unit matrix I

A unit matrix is necessary, for simplifyng the Kalman equations.

Updating and prediction steps

Now as all parameters which needs to be set for applying a Kalman Filter have been explained, the focus now lies on how to update each magnitude for every iteration and

how the magnitudes are linked together, to get an unterstanding how the prediction and updating of the new state works. The co-variance must be recalculated. Uncertainty about the state of the system increases in the predict step, as we have seen in the one dimension case. In the multidimensional case, the measurement uncertainty is added, so the uncertainty becomes larger and larger.

1. Step - new predicted state

$$x_{kp} = A * x * B * \vec{u_k} \tag{4.17}$$

2. Updating covarinace matrix
if initial covariance matrix is already set

$$P_k = A * P_{k-1} * A^T + Q \tag{4.18}$$

3. Computing the Kalman Gain

$$KG = \frac{P_k * H^T}{H * P_k * H^T + R} \tag{4.19}$$

The Kalman Gain will decrease if the readings match the predicted system state. If the measured values are off compare the predicted values, the elements of matrix KG become larger.

4. Update the estimate y_k via new readings

$$y_k = C * y_{km} * H * x_{k-1} \tag{4.20}$$

with: C: transform matrix,
y_{km}: actual observations,
x_{k-1}: previous predicted state

5. Updating current state

$$x_k = x_{k-1} + (KG * y) \tag{4.21}$$

which we got to know laready in one-dimensional case Equation 4.7

6. Update process covariance matrix

$$P_k = (I - KG * H) * P_{k-1} \tag{4.22}$$

From here the iteration or loop will start again.The filter runs permanently as long as measured values come in. It can also be an open loop, so only the prediction step will be executed if no measurements are available. Then the uncertainty gets bigger and bigger. Graphically it looks like this:

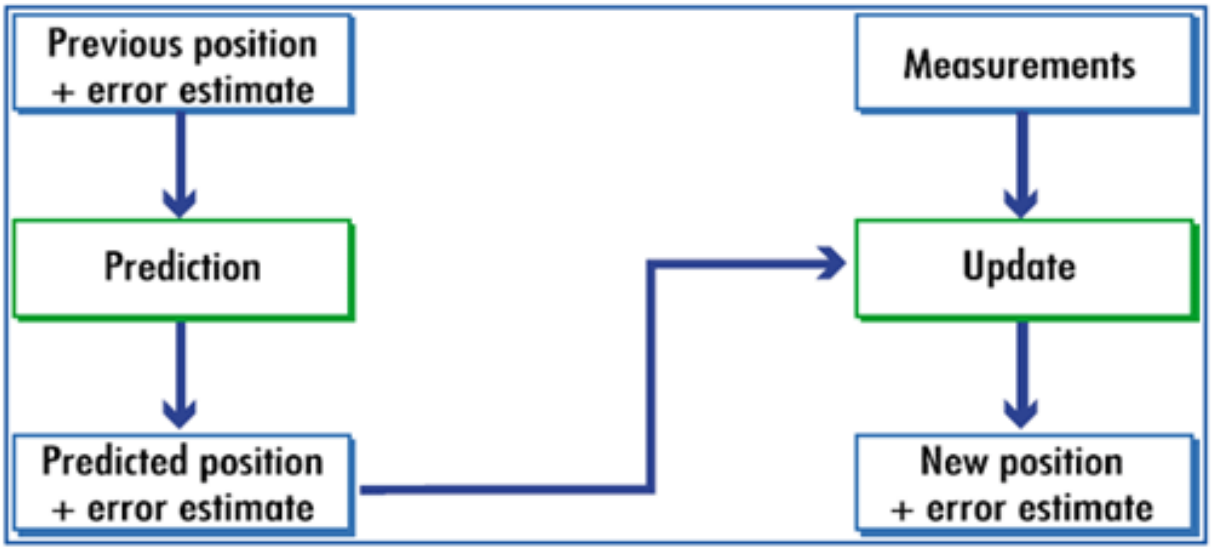

Figure 4.3: Visual interpretation of one Kalman Filter iteration

5 Conclusion

The kalman filter is a very powerfull tool and mathematical concept,not a very complicated construct. The crucial part of applying a Kalman Filter is figuring out which parameter correspond to which which phenomena (uncertainities, forces from outside, process uncertainities etc.) and the challenge will be tuning the those parameter in a proper way. A lot of test data will be needed and avery possible different condition have to be covered to see the effects of how the parameters are effecting the result, so that the system can adopt to changes accordingly and constantly. A particular choice might perfom well in one situation, but very poorly in another.

Bibliography

[Dev20] Devs, Mad: *MAD Location Manager*. https://github.com/maddevsio/mad-location-manager/wiki/Theory, 2020. – last access: 25.06.2020

[FVDW18] Frank Van Diggelen, Roy W.; Wang, Wei: *How to achieve 1-meter accuracy in Android*. https://www.gpsworld.com/how-to-achieve-1-meter-accuracy-in-android/, 2018. – last access: 23.06.2020