Institutionen för systemteknik
Department of Electrical Engineering

Examensarbete

Evaluation of a statistical method to use prior information in the estimation of combustion parameters

Examensarbete utfört i Fordonssystem
vid Tekniska högskolan i Linköping
av

Patrick Rundin

LITH-ISY-EX–05/3721–SE

Linköping 2006

Department of Electrical Engineering
Linköpings universitet
SE-581 83 Linköping, Sweden

Linköpings tekniska högskola
Linköpings universitet
581 83 Linköping

Evaluation of a statistical method to use prior information in the estimation of combustion parameters

Examensarbete utfört i Fordonssystem
vid Tekniska högskolan i Linköping
av

Patrick Rundin

LITH-ISY-EX–05/3721–SE

Handledare: **Marcus Klein**
Linköpings Universitet
Jakob Ängeby
Mecel AB

Examinator: **Lars Eriksson**
Linköpings Universitet

Linköping, 28 March, 2006

	Avdelning, Institution Division, Department Vehicular Systems Department of Electrical Engineering Linköpings universitet S-581 83 Linköping, Sweden	**Datum** Date 2006-03-28

Språk Language ☐ Svenska/Swedish ☒ Engelska/English ☐ __________	**Rapporttyp** Report category ☐ Licentiatavhandling ☒ Examensarbete ☐ C-uppsats ☐ D-uppsats ☐ Övrig rapport ☐ __________	**ISBN** — **ISRN** LITH-ISY-EX–05/3721–SE **Serietitel och serienummer** Title of series, numbering **ISSN** —

URL för elektronisk version

www.liu.se/exjobb/isy/2006/3721

Titel Title

Utvärdering av en statistisk metod för att förbättra estimering av förbränningsparametrar med hjälp av förkunskap

Evaluation of a statistical method to use prior information in the estimation of combustion parameters

Författare Author

Patrick Rundin

Sammanfattning
Abstract

Ion current sensing, where information about the combustion process in an SI-engine is gained by applying a voltage over the spark gap, is currently used to detect and avoid knock and misfire. Several researchers have pointed out that information on peak pressure location and air/fuel ratio can be gained from the ion current and have suggested several ways to estimate these parameters.

Here a simplified Bayesian approach was taken to construct a lowpass-like filter or estimator that makes use of prior information to improve estimates in crucial areas. The algorithm is computationally light and could, if successful, improve estimates enough for production use.

The filter was implemented in several variants and evaluated in a number of simulated cases. It was found that the proposed filter requires a number of trade-offs between variance, bias, tracking speed and accuracy that are difficult to balance. For satisfactory estimates and trade-off balance the prior information must be more accurate than was available.

It was also found that similar a task, constructing a general Bayesian estimator, has already been tackled in the area of particle filtering and that there are promising and unexplored possibilities there. However, particle filters require computational power that will not be available to production engines for some years.

Nyckelord
Keywords

bayesian estimation, ion current, air/fuel ratio, peak pressure location, SI engine control

Abstract

Ion current sensing, where information about the combustion process in an SI-engine is gained by applying a voltage over the spark gap, is currently used to detect and avoid knock and misfire. Several researchers have pointed out that information on peak pressure location and air/fuel ratio can be gained from the ion current and have suggested several ways to estimate these parameters.

Here a simplified Bayesian approach was taken to construct a lowpass-like filter or estimator that makes use of prior information to improve estimates in crucial areas. The algorithm is computationally light and could, if successful, improve estimates enough for production use.

The filter was implemented in several variants and evaluated in a number of simulated cases. It was found that the proposed filter requires a number of trade-offs between variance, bias, tracking speed and accuracy that are difficult to balance. For satisfactory estimates and trade-off balance the prior information must be more accurate than was available.

It was also found that similar a task, constructing a general Bayesian estimator, has already been tackled in the area of particle filtering and that there are promising and unexplored possibilities there. However, particle filters require computational power that will not be available to production engines for some years.

Sammanfattning

Vid jonströmsmätning utvinns information om förbränningsprocessen i en bensinmotor genom att en spänning läggs över gnistgapet och den resulterande strömmen mäts. Jonströmsmätning används idag för knack- och feltändningsdetektion. Flera forskare har påpekat att det finns än mer information i jonströmmen, bl.a. om bränsleblandningen och cylindertrycket och har även föreslagit metoder för att utvinna och använda den informationen för skattning av dessa parametrar.

Här presenteras en förenklad Bayesisk metod i form av en lågpassfilter-liknande skattare som använder förkunskap till att förbättra estimat på relevanta områden. Algoritmen är beräkningsmässigt lätt och kan, om den är framgångsrik, leverera skattningar av förbränningsparametrar som är tillräckligt bra för att användas för sluten styrning av en bensinmotor.

Skattaren, eller filtret, implementerades i flera varianter och utvärderades i ett antal simulerade fall. Resultaten visade på att flera svåra avvägningar måste göras mellan förbättring i varians, avvikelse och följning eftersom förbättring i den ena ledde till försämring i de andra. För att göra dessa avvägningar och få goda skattningar krävs bättre förhandskunskap och mätdata än vad som var tillgängligt.

Bayesisk skattning är ett stort befintligt område inom statistik och signalbehandling och den mest generella skattaren är partikelfiltret som har många intressanta tillämpningar och möjligheter. De har hittills inte använts inom skattning av förbränningsparametrar och har således god potential för framtida utveckling. De är dock beräkningsmässigt tunga och kräver beräkningsresurser utöver vad som är tillgängliga i ett motorstyrsystem idag.

Acknowledgements

This master thesis was written for Mecel AB at the Department of Electrical Engineering at Linköping university under the supervision of Jakob Ängeby of Mecel and Marcus Klein of Linköping University.

I would especially like to give thanks to Jakob Ängeby for his enthusiasm and for taking me in at Mecel and to Marcus Klein for his patience, guidance and supervision. Many thanks also to all the people at Mecel who welcomed me openheartedly at my stays in Åmål.

Rickard Karlsson gave valuable help in introducing me to Particle Filtering and help sort out the mass of resulting questions.

Finally, thanks to all my friends in Linköping who have made my time here a joy and my parents for their support and loving care.

Linköping, March 2006
Patrick Rundin

Contents

Chapter 1

Introduction

The quest for refinement and perfection of the internal combustion engine has been going on for over a century now. Modern engine designs are vastly more effective, cleaner, lighter and more reliable then ever before. Yet with new materials, understanding and technology the limit for what is possible is pushed ever further. One crucial improvement over the last two decades has been the move to computer controlled engines. While often ingenious, mechanical systems for control of throttle, fuel injection and spark timing cannot match the accuracy and flexibility of computerized control. The advent and use of new measurement technologies offer a wealth of information about the operation of the engine. All available on the road, while the engine is running, every split second; information that can only be interpreted and processed by a computerized control system.

The piece of electronics that achieves this is the Engine Control Unit (ECU) and is present in every production car today. It is cram-packed with hard- and software that gathers information on everything from fuel injection to coolant temperature. The information consists of measurements from sensors in and outside the engine and inputs from the driver. This together with pre-determined lookup tables stored inside the ECU allows reliable and accurate control and operation of the engine, keeping efficiency high and emissions low.

Feedback, or closed-loop control, is comparable to what we do ourselves as drivers of the car. Essentially we are using measurements of reality, our vision and interpretation of the road to control the car by comparing and correcting it to the direction we wish to go. We can adapt to changes in the road, cars around us and other factors in our environment. In contrast, open-loop control would be like driving blind with only a step by step instruction to guide us.

Unfortunately, the cost of accurate and robust sensors may well be too high for production line use. A trade-off has to be made between the quality of the sensor used and its cost. The life span of the sensor also becomes an issue if it is considerably shorter than the product in which it is to be used.

In engine control in particular high-quality sensors are many times more expensive than off-the-shelf components. The harsh environment of the engine and its surroundings also quickly wear out sensitive equipment. Therefore closed-loop control is not viable in many cases and the engines instead rely on open-loop control, i.e. a huge database or lookup table, an engine map, which provides the settings of the engine for each particular driving condition and no sensors for feedback are used.

These tables are devised in a lab environment or test bench where one is less concerned with the problems of sensor cost and wear. A test engine can be fitted with all sorts of top-of-the-line equipment and if the recovered data extrapolates well to production engines the cost of the setup can be justified. This procedure is still quite costly however, since it significantly increases development time and tuning cost of the engine.

A combination then of less rigorous test bench data and a cheap but durable sensor is proposed as a way of overcoming the problems inherent with using one or the other.

The ion current sensor developed by Mecel AB is such a sensor. Already used in production engine control to detect abnormal combustion it makes use of the spark-plug as a sensor and is thus cheaply and easily integrated into existing engine designs.

> **This master thesis will explore an idea to make use of engine map data or similar prior information to improve normal combustion ion current measurements. If these improvements are good enough closed-loop engine control for each individual cylinder becomes possible.**

1.1 Goals and Motivation

Now to formalize the goals for this thesis, why these goals may be achievable and the motivations for setting them.

Goal: Algorithm implementation and evaluation

To in simulation implement and evaluate the estimator algorithm proposed by Jakob Ängeby of Mecel AB.

Motivation

- The proposed algorithm is a simple filter that could, backed by known results in Bayesian statistics, use measurements and prior information to achieve an improved estimator that has the potential of enabling closed-loop engine control at low cost.

- The algorithm is much simpler than existent Bayesian estimators and could potentially be software implemented in an ECU without requiring additional components.

- It has been proven in Bayesian theory that using correct prior information will improve the estimator [20]. Thus how prior information can be described and incorporated is an important part of the evaluation.

- It has also been proven that correlations between variables can be used by a Bayesian estimator to further improve the estimate [20]. Since the ion current contains information on a variety of parameters, many of them with correlations to each other, it would be desirable to make use of this correlation.

Goal: Comparison to current developments

To explore current literature and publications on Bayesian estimation and engine parameter estimation, to relate the algorithm to existing well understood estimators and to incorporate previous work in the area where possible and suggest paths and ideas for future work.

Motivation

- Study of current publications and papers in the area of engine parameter estimation is necessary to understand the potential, limitations and requirements of the proposed algorithm. Likewise in the area of Bayesian estimation to determine current developments and survey existing implementations.

- It is likely that similar estimation approaches have been used before, although not in an engine parameter context, and a survey of publications is necessary to determine the current knowledge in the field.

Goal: Ion current based estimates of engine parameters

To use previously implemented estimators in conjunction with engine test bench measurement data and the proposed algorithm to arrive at actual, improved estimates of the peak pressure location (PPL) and air/fuel ratio (AFR) engine parameters and evaluate the results against reference data.

Motivation

- The proposed algorithm is a link in a larger chain of prerequisites required for closed-loop engine control. Being able to form reliable estimates of important engine parameters leaves out only the last step, designing the controller, paving the way for the design of a complete closed-loop system.

- The proposed algorithm must be shown to handle the noise, irregularities and inconsistencies that actual measurements inevitably contain. Results from the implemented algorithm evaluated on real data would provide such proof-of-concept.

The long term goal

The previous thesis goals small steps toward a long term goal, namely: To formulate an estimation method based on ion current measurements that can deliver estimates of engine parameters that are accurate and reliable enough for closed-loop engine control of individual cylinders.

Motivation

- Current open-loop engine control schemes rely heavily on elaborate engine maps that cost much development and tuning time. Closed-loop schemes require accurate measurements but available sensors are expensive and susceptible to wear and breakdown. A successful ion current based estimator thus enables closed-loop control without the drawbacks of other sensors or the costly development time of extensive engine maps.

- Since ion current measurements are available from every spark plug each cylinder can be measured and controlled individually. This opens the possibility of for example cylinder individual AFR control and cylinder balancing.

1.2 Previous Work

A short overview of papers and publications used as background and reference in this thesis follows.

Bayesian estimation and filters

The Bayesian estimation theory that is used here can be found in any books on statistical estimation, for example Kay [20]. The Kalman filter and some of its extensions are described extensively in [1]. While the Kalman filters assume gaussian noise the Bootstrap filter can handle noise of arbitrary distributions as well as a non-linear system. It is the origin of most particle filters and was first presented in [15]. The overview [10] nicely sums up developments in sequential Bayesian filtering, (including Bootstrap and particle filters), up to 1998. For a more in depth study the book [26] is recommended.

Engine parameter estimation

Numerous attempts at estimating engine parameters, mainly the air/fuel ratio (AFR) and the peak pressure location (PPL) have been made. A multiple gaussian model was used by Klövmark [21] to estimate the PPL. In Reinmann [24] the AFR is estimated by using of a linear correlation between parts of the ion current and measured AFR. Somewhat better AFR estimates were found in Hellring [17] where neural networks were used as estimators. Neural networks were also used in Wickström [29] to successfully estimate PPL. Byttner uses neural networks for PPL as well as AFR estimates in [5].

Despite the partial success in the papers above the estimators were not reliable over varying external conditions. In particular with neural networks it is inherently difficult to make any statements about the networks performance or robustness outside the trained and tested situations. While the mentioned papers try to amend this problem no result was consistent and general enough for production implementation.

Chapter 2

Background

2.1 Overview

The practical context for the work in this thesis is the spark ignited internal combustion engine. Section 2.2 gives a brief overview of the here relevant parts of its operation. The proposed estimator algorithm, later presented in section 3.5, is intended to be used on ion current sensor data from a combustion engine. The ion current sensor, section 2.3, is a measurement device designed to collect measurements of the combustion process. For purposes of engine control certain engine parameters are of interest. The what and whys of these are described in section 2.4. Moving towards engine parameter estimation section 2.5 covers problems and possibilities as well as previous work. In section 2.6 fundamentals of the estimation theory used here, Bayesian estimation, is introduced. Existing implementations of Bayesian estimation are the Kalman and particle filters discussed in sections 2.7 and 2.8. The aim and potential of the estimator proposed in this thesis is set in relation to these existing and familiar estimators.

2.2 The Spark Ignited Combustion Engine

The algorithm under study in later parts of the thesis is intended to be applied in the engine powering most consumer cars today, the spark ignited combustion engine, or Otto-engine. The engine derives its power by burning gasoline fuel in a four stage process. Air and fuel is mixed prior to being injected into the cylinder and ignited by a spark from the spark plug. The combustion and resulting increased pressure drives a piston which transfers the work done to the crank axle. Figure 2.1 shows the four different strokes of the engine. Ignition occurs shortly before the piston reaches the top position, top dead center (TDC), at the end of the compression stroke.

The expansion stroke is the work producing stage and is thus naturally the most important part of the process. The ignition spark starts a small flame that propagates outward through the entire cylinder. The chemical reactions cause the

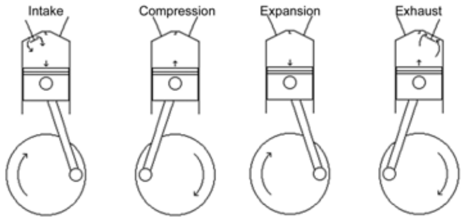


Figure 2.1. The four stages, or strokes, of the SI engine operation. Ignition occurs near the end of the compression stroke.

temperature to rise and the pressure to increase. The pressure development is crucial for the amount of work done. For low emissions and high fuel efficiency, a homogenous, well balanced air/fuel mixture is desired. Furthermore abnormal combustion, such as spontaneous ignition along the cylinder walls, should be avoided as it can damage the engine.

Balancing these factors is a difficult task and often leads to contradictory requirements. Elaborate testing and open loop control strategies allow modern engines to balance these benefits and trade-offs for increased efficiency under varying conditions. Yet there is still ample room for improvement if we were able to act on what actually is happening in the cylinder, particularly during the all-important combustion. One sensor for this purpose is the ion current sensor.

2.3 Ion Current Sensing

Ion current sensing is a measurement technology that makes use of the existing spark plug to retrieve information of the combustion process inside the cylinder. It is non-intrusive and does not need any additional equipment inside the cylinder as all components are part of the spark plug, ignition coil and engine control unit.

Ion current sensors are and have been in use in production engines since for over a decade now. Mainly as a cost-effective and reliable knock and misfire detector.

Ion current physics

The principal idea behind the ion current sensor is that the gases in the cylinder are ionized during combustion and that the amount of ionization is dependent on key factors such as flame propagation, pressure development and temperature. If exposed to an electric field the ions will conduct a current. A current that can be measured and from which the above factors and thereby parameters related to them may be inferred.

Complex multi-stage chemical reactions take place during combustion, converting the hydrocarbons into carbon dioxide and water. In the intermediate stages of these chain reactions ions are formed and recombined by competing processes.

All of which contribute to different parts and characteristics of the measured ion current [2].

The ion current contains a wealth of information but is difficult to interpret since no unambiguous and direct correlations exist between ion current and engine parameters. Neither is any accurate physical model known that can consistently and accurately describe the development of the ion current over an entire cycle. Consequently most of the correlations between engine parameters and the ion currents that have been useful in the past are based on empirical observations. These correlations have been compelling enough to motivate numerous studies and attempts at estimating engine parameters from ion currents, such as [24], [17], [5].

The ion current sensor

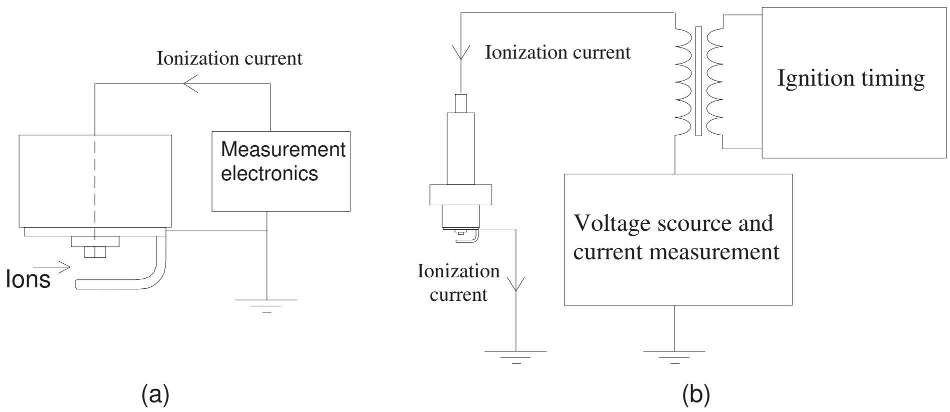


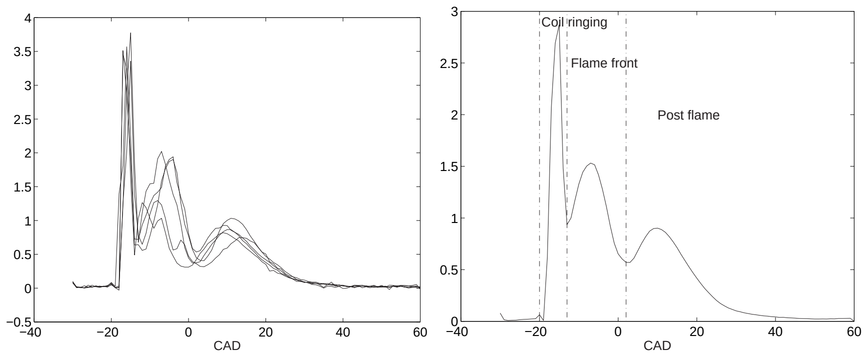
Figure 2.2. Measurement of the ionization current. (a) The spark plug-gap is used as a probe. (b) Measurement on the low voltage side. (Figure from Eriksson [11])

The ion current sensor used here has been developed by Mecel AB and is used in several production engines. The sensing circuit is placed on the low voltage side of the ignition coil, safely away from the harmful high-voltage spark, and receives its voltage directly from the spark current. The configuration can be seen in figure 2.2.

The spark current charges a capacitor which then supplies the voltage to the sensing circuit after the spark has been fired. The voltage is applied over the spark gap of the plug and the resulting current carried by the ions is measured over the remaining part of the cycle.

The measured ion current

The raw ion current is heavily influenced by noise and displays large cycle-to-cycle fluctuations, as can be seen in figure 2.3a. Cycle-to-cycle interpretations are thus difficult and are notoriously imprecise.



(a) (b)

Figure 2.3. (a) Five ionization current measurements from consecutive cycles. (b) An ionization current curve averaged over 100 cycles. The three characteristic phases are easily visible: Coil ringing, flame front and post flame.

A clearer picture emerges as the measurements are averaged over many cycles. In figure 2.3b an average over 100 cycles shows the typical characteristics. The ion curve is divided into three phases; coil ringing, flame front and post flame.

The first phase, the coil ringing, is caused by residual charge in the ignition coil and circuitry and contains no combustion information. This part of the measurement is most often removed in signal pre-processing by using the empirical relation for spark duration found in [21]. At higher engine speeds, say 3500 RPM and above, cycle duration becomes very short and the coil ringing overlaps part of the subsequent flame front phase interfering with measurements and rendering them useless.

Once the spark has ignited the fuel mixture it takes the flame some time to grow and spread. This is known as the ignition delay. Measurement in this area is often blocked by the coil ringing but the duration of the delay can be inferred from the rising slope or position of the second phase peak.

The second phase, from -15 to about -5 CAD, is caused by the flame front ionizing the mixture in the vicinity of the spark gap. The position and height of the resulting peak is known to be correlated with the AFR [24]. Unfortunately, ionization in the flame varies quickly as several processes rapidly release and absorb ions during the combustion. This, combined with the fact that the mixture is inhomogeneous and turbulent, gives rise to several brief peaks in cycle-to-cycle measurements. The presence of these multiple peaks complicates interpretation and any estimator based on information in the first peak must be robust to these errors.

Finally, the post flame phase of the ion current has a strong correlation with the pressure development in the cylinder. No theoretical model can as yet fully explain this second rise in ionization. Thermal ionization it thought to be a major factor but engine load, fuel additives and other conditions also affect its development. Often by diminishing it, sometimes equal to the point where no second peak can be distinguished at all. But as it clearly does contain important information when it is present, attempts to use it in parameter estimation are numerous.

This above summarizes some of the results found in [12, 24, 29, 21].

2.4 Engine Parameters

An engine parameter can be any physical quantity or otherwise derived value that describes and quantifies an aspect of the engine operation, in particular the combustion process. To be useful it must be closely linked to a physical quantity or phenomenon that we wish to measure or control. Several such parameters are used in engine control today. The ones most important to this paper are presented below.

Air-fuel ratio (AFR) and lambda (λ)

The air/fuel ratio is defined as the mass relationship between air and fuel in the mixture injected into the cylinder. The AFR influences nearly all other parameters, such as torque and peak pressure, as well as general properties like performance and emissions. It can be directly influenced by the fuel injection and throttle and is thus very useful for engine control.

For optimal combustion and efficiency the mixture should be at the stoichiometric ratio of 14.7 mass parts air to 1 part standard gasoline fuel. The exact numbers vary depending on the type of fuel. Normalizing the relation as in eq. (2.1) yields the more practical parameter lambda (λ) which is independent if the fuel type. A mixture with excess fuel, $\lambda < 1$, is called rich and with excess air, $\lambda > 1$, is called lean. To keep efficiency high and emissions low it is desirable to run the engine lean in most cases. Running rich gives more power, most engines have their peak power output at an AFR of 12.5 - 13.3:1, but leaves some fuel unburned resulting in high emissions and low efficiency.

$$\lambda \equiv \frac{\text{(parts air in mixture)}}{\text{(1 part fuel)} * \text{(stochiometric ratio)}} \tag{2.1}$$

In modern vehicles employing catalytic exhaust converters the averaged AFR over all cylinders must be kept very near stoichiometric for the converter to function. To achieve this AFR is usually measured directly by a lambda-probe in the exhaust manifold. The most common probe, the non-linear or integrating probe, distinguishes only between above or below stoichiometric and is commonly used to keep the total λ close to one to ensure good conversion efficiency in the catalyst. Another kind of probe, the linear, yields actual measurements of λ. These two probes are also called EGO- and UEGO-probes respectively. UEGO-probes are due to their high price, sensitivity to high temperatures and short lifespan not viable for use in individual cylinder measurements except in experimental setups.

Cylinder individual AFR measurements would make it possible to compensate for and eliminate cylinder imbalances. Due to the physical geometry of the intake manifold and imperfections and wear in the engine components, multi-cylinder engines are not perfectly balanced over all cylinders. That is, air/fuel ratio may vary between individual cylinders. Some cylinders may run lean, others rich and yet others might not be firing properly at all experiencing knock or misfire. While the catalyst will still work if these imbalances average out over the cylinders the engine efficiency and emissions will not be at optimal.

Peak pressure location (PPL)

The peak pressure location is the position of the crank axle, in degrees relative to top dead center, where the maximum pressure occurs during the combustion cycle. The pressure in the cylinder is closely linked to the combustion process and the work done on the crankshaft. Most of the pressure should develop after TDC for it to do maximum positive work. Controlling PPL has been shown to be an effective control strategy to keep the engine at maximum output [12]. The PPL can be controlled by adjusting ignition timing or AFR, among other ways. Cycle to cycle variations however are quite large - during normal operation the actual PPL varies by as much as 10 CAD.

Cylinder pressure can be measured by pressure probes drilled directly into the cylinder or as part of the spark plug. While these probes yield very reliable measurements, they are expensive and prone to wear. Better and cheaper probes are being developed by manufacturers but the ones currently available are to costly to be used in consumer car engines.

Other common engine parameters

There are several other engine parameters that can be of interest in engine control. Many are related to each other and their usefulness depends on the context. Particularly interesting are those that affect the ion current in some way, like knock, misfire, mass fraction burnt and combustion stability.

If knock and misfire can be detected the engine can also be controlled to avoid them while still operating on, or very near, their limit. Desirable since this has been shown to be near optimal. This was in fact the first application where ion current measurements were used [3].

Other parameters can also be used in the engine control system for purposes of feedback control or diagnostics. The usefulness of the parameter is always dependent on how well it reflects the process which we aim to control or diagnose.

2.5 Engine Parameter Estimation

There are numerous approaches to extract the engine parameter information contained within the ion current measurement. Estimation of the PPL and AFR parameters and improving these estimates have been the main focus in this work but the can be applied to the estimation of any parameter.

Interesting empirical correlations

The first step to estimation is to find correlation between the measured data and the desired parameter. We below present some correlations between ionization current and interesting parameters that can be made use of in estimation of AFR and PPL.

- The flame front peak amplitude, position, duration and rising edge derivative have all been found to be correlated with the AFR. The position is directly proportional to the ignition delay that is known to be dependent on the AFR. Position and duration are connected with flame propagation speed and reactability of the mixture. Using the rising edge derivative incorporates information from both amplitude and position into one variable and is thought to strengthen the correlation [24, 21].

- The post flame peak amplitude has been seen to correlate with the AFR although its nature is uncertain [24].

- The post flame phase is connected to the pressure development in the cylinder and its peak correlates well with the PPL. Thermal ionization is believed to be the cause and is supported by theoretical calculations. However the post flame ion current peak tends to occurs slightly before the pressure peak [25, 13, 29].

Estimating PPL

The peak pressure location is perhaps the most interesting engine parameter since it is closely connected to the amount of work done in a stroke and is directly influenced by the ignition timing.

Since the location of the post flame peak is known to be directly correlated with the PPL, the simplest estimation method is to locate the post flame peak of the ion curve and take that position, in CAD, as the estimate. If there are several peaks in the post flame phase one can pick a plausible candidate or average over all candidates within a predefined window. This is rather imprecise approach though and has not been used other than for initial guesses. In Klövmark [21] modelled the ion curve as two gaussian curves. The gaussians correspond to the bulbs seen in the flame front and post flame phases. The parameters of the curves were least square fitted to the ion curve and the position parameter of the second curve was taken as an estimate of the PPL. In another approach taken by Wickström [29] a neural network was used to estimate PPL from ion current data compressed using principal component analysis, PCA.

Wickström et al. compared results from the above estimators and achieved a performance of 2 - 4 CAD in RMSE with the neural network estimators and 6 - 13 CAD with the gaussian approach.

Malaczynski and Baker [22] implemented the same algorithm in a simulated real-time system and achieved a RMSE of 0.8 CAD.

Estimating AFR

Several attempts at estimating the AFR from the ion current have been made. In Reinmann [24] a linear function was fitted to the relation between flame peak rising edge derivative and AFR. In Klövmark [21] the ion current was modelled by two gaussian curves and the parameters of these curves were fed into a trained neural network. In Hellring [17] the ion current was normalized, then encoded using PCA and subsequently processed by a neural network. The performance of these estimators where in the range of approx 6% - 1% in RMSE.

It is worth to note that no estimator can get much better than that since the accuracy limit for any estimator is, due to system and measurement noise, at least 1-2% as can be seen in for example Johansson [4].

Problems inherent to ion current measurements

While the above is encouraging, a number of problems remain. Physical difficulties, such as the combustion chemistry, and mathematical difficulties, such as errors in modelling non-linear data, complicate the construction of a generally valid estimator. The chemistry of the combustion process differs significantly between lean and rich mixtures, [7], and it might be necessary to use respectively different estimators. Moreover the ion current measurement, (as opposed to factors directly affecting the parameter), has a number of inherent problems which are listed below

- The ion current is measured over a very small and very local volume. The spark gap represents only a small part of the cylinder volume and consequently any measurement based on it will only reflect the conditions in the gap. There is thus no guarantee that the ion current based estimate reflects the conditions for the entire cylinder even if it correctly estimates the conditions in the spark gap.

- Turbulence and non-uniformity of the mixture cause physical variations in the ion current. This accounts for a large part of the cycle-to-cycle variations in the ion current trace, typically multiple and erratic peaks in the flame phase. See figure 2.3a.

- Engine speed influences the characteristics and the quality of the ion current. For example, at high speeds each cycle is considerably shorter allowing less time for the current to develop. Some characteristics seen at lower speeds may entirely vanish or be drowned out by overlap or noise.

- Fuel composition and quality has a large impact on the ion current. Pollutants and fuel additives supply additional easily ionizable molecules and increase the magnitude of the ion current. While this improves the signal to noise ratio of the ion curve the estimator must be robust against such changes in amplitude.

- Exhaust gas recycling (EGR) is increasingly being used in production engines to reduce NO_x-emissions. Unfortunately EGR diminishes the ion current, in particular the post flame phase, to a large degree since the recycled exhaust gases reduce the concentration of available ions.

- Ambient conditions such as humidity, temperature and height over sea-level, (pressure), also affect combustion and the development of the ion current.

- Spark plug geometry, condition and type affect the measured current and specifically tested and approved spark plugs are required.

Even if the above problems are overcome the behavior of the ion current has been known to vary greatly between different engine types and brands. The reasons for this are not clear but indicate that there is no general out-of-the-box solution - each estimator needs to be built, tuned and tested for that specific line of engines.

Section conclusion

There are several proposed methods to estimate engine parameters from ion current data that have been implemented with some success. Problems with robustness and generality still remain however.

Regardless of how the above problems are handled this paper proposes that the use of prior knowledge any correlations between parameters can be made use of to additionally refine the estimates.

2.6 The Bayesian Approach to Parameter Estimation

In many applications additional information about the parameter we wish to estimate is available. Information that neither the measurement device nor the estimator takes into account. For example physical constraints, previous accurate measurements, additional sensors or other knowledge. It would naturally be desirable to use this knowledge to improve our estimator, since statistical estimation theory tells us that using such prior knowledge will always lead to a more accurate estimator [20].

The central equation in Bayesian estimation is Bayes theorem (2.2). It gives an expression for the conditional probability of a random variable θ after the observation of the measurement x.

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)} \tag{2.2}$$

$$p(x) = \int p(x|\theta)p(\theta)d\theta \tag{2.3}$$

where the denominator, (2.3) is a normalization factor to make $p(\theta|x)$ a valid PDF.

Making use of and combining different sources of information in the best possible way is a well known problem and is commonly solved by the Kalman filter. Its derivation follows from Bayesian statistics but it requires all noise to be gaussian. A requirement we wish to avoid. More on the Kalman filter in section 2.7.

Bayes equation and Bayesian statistical theory gives us ground to explore in our search for such an estimator. From existing literature we know that sequential estimators, crucial for an online application, can be constructed [20]. The concepts, equations and ideas that are starting points or key elements for this thesis presented and discussed further in section 3.

More background on Bayesian statistical theory can be found in Kay [20], chapters 10-12, and the reader interested in explicit derivations should turn there. For a more easily digested introduction the online *Engineering Statistics Handbook* [23] is recommended as well browsing papers in the area of Bayesian Filtering, for example [9, 10].

2.7 The Kalman Filter and Extensions

The Kalman filter is often used when combining different sources of information to estimate an unknown but dependent quantity. In a state-space system description this means reconstructing a non-measurable state from available states. Under gaussian conditions the Kalman filter is the optimal way of doing this. So the question arises, why aren't we using Kalman-filters?

The first reason lies in the limitations of the conditions. For the basic Kalman filter to be valid the system must be linear and all noise gaussian. Let's express this in a mathematical system description

$$\begin{aligned} x_k &= f(x_{k-1}, u_k) + v_1; \\ y_k &= g(x_k, u_k) + v_2; \end{aligned} \tag{2.4}$$

where x_k is the state vector and u_k a vector of known inputs at time k. The first line represents the system equation and the second is the measurement equation. For the Kalman filter the functions $f(\cdot)$ and $g(\cdot)$ are linear and the noise is additive and distributed as $v_1 \sim \mathcal{N}(\mu_1, \sigma_1)$ and $v_2 \sim \mathcal{N}(\mu_2, \sigma_2)$ and mutually independent.

Essentially the filter continuously updates the noise properties, μ_x and σ_x, by equation (2.2) and uses a ML- or MVU-estimator to estimate the new states x at time k.

Unfortunately real systems are seldom this friendly. Engine and vehicle systems in particular are notoriously non-linear and non-gaussian. There are several improved variants of the Kalman filter, the most common being the Extended Kalman Filter (EKF) which extends into non-linear problems, allowing non-linear functions $f(\cdot)$ and $g(\cdot)$, by linearizing the system at every time step [1]. The Unscented Kalman Filter (UKF) further improves handling of non-linearities and the noise propagation through them [28]. However system and measurement noise are still assumed to be gaussian.

The second reason that arises as soon as any of these methods are to be implemented is the computational load. While the Kalman filter is well researched and many "clever" solutions have been found to make it more effective it is not currently feasible for the applications under consideration here, i.e. in an existing real-time engine control system. Neither are its extensions as they require even more computational resources.

> **The approach that will be taken in this thesis is indeed related to Kalman filters but tries to do without the requirements of linearity and gaussian noise while keeping computational cost as low as possible.**

For more information on Kalman filters the reader is recommended to turn to the extensive literature in the area or textbooks in control theory [1, 27].

2.8 Particle Filtering and Sequential Monte Carlo Methods

While Kalman filters apply Bayes theorem to the mean and variance of the assumed gaussian, particle filters and related methods take the Bayesian estimation approach all the way. Bayes theorem is applied to the entire PDF and measurement data. Non-linearities and noise of any distribution can be handled but the computational cost is heavy.

The bootstrap filter

The bootstrap filter was first presented by Gordon, Salmond and Smith in [15] and is the origin of the whole particle filter family. The algorithm is presented in more detail in section 3.6.

The system model handled by the bootstrap filter allows for non-linear f and g and non-additive noise but the distribution of the system noise must be known or measurable.

At each step in the algorithm the discrete forms of (2.2) and (2.3) are evaluated over a set of points, samples, of the actual PDF. The resulting new samples form the PDF of the states given all available data. The major difficulty in this procedure is the evaluation of the integral in the normalization factor (2.3). The restrictions on the Kalman filter arise from the need to find analytical solutions to this integral. In particle filters this is overcome by using numerical methods to find an approximate solution.

The particle filter family springs from the several possible ways to choose the sample points and how to refining the choices over time as well as other improvements to the bootstrap filter [19, 6, 14].

Other particle filters and section conclusion

Methods in this area are also known as Sequential Monte Carlo, (SMC), methods since Monte Carlo simulation techniques are used in the numerical stages. From a mathematical viewpoint Markov processes are the equivalent of state space descriptions and hence the term Markov Chain Monte Carlo, (MCMC), is also used for related methods. An overview and chronology of the area up to 1998 can be found in Doucet [10]. An introduction to particle filters can be found in the textbook by Smith [26].

While improvements to the particle filter algorithms have appeared during the years they are by nature more complex than the Kalman filter. Their accuracy relies heavily on the Monte Carlo stage, the choice and propagation of samples of the PDF, which often requires thousands of sample points. The computational load is in the range of seconds per iteration even on a modern PC. Clearly infeasible for on-line use in engine control where processing power is but a fraction and the time available is in the order of milliseconds. Not to mention the strict requirements of a real-time system.

> **A particle filter is the most complete way to incorporate all knowledge by Bayesian means but not practically feasible in engine control today. However, the bootstrap filter implementation is used in this thesis as a comparison to the proposed method.**

A short note on Gaussian sum filters

A predecessor to the particle filters is the Gaussian sum filter which essentially represents the prior PDF by a weighted gaussian sum, much like a Fourier decomposition can represent a function as a sum of weighted sines. Each gaussian, described by its parameters, is treated separately and processed through a Kalman or Extended Kalman filter. The result is recombined to form the posterior PDF from which an estimated can be formed [1]. Difficulties may arise if one or several of the EKFs derail and computational cost quickly becomes heavy as the number of gaussians increase.

The Gaussian sum filter was not used in the thesis but some similarities between it and the proposed method were observed and are discussed in later sections.

2.9 Relationship to the Proposed Algorithm

In engine parameter estimation noise is often non-gaussian and computation resources are limited. Our goal here is to incorporate different information sources and correlations in an estimation method that is flexible and computationally efficient. An optimal estimator is not required as long as it is "good enough".

Bayesian theory suggests that this is possible since prior information can be used and sequential Bayesian estimators can be constructed.

Existing algorithms have drawbacks we wish to alleviate. The Kalman filter provides optimality but only under gaussian conditions and linearity restrictions on the system. The particle filter works with any noise distribution and non-linear system but requires too much computational power.

The approach in this thesis aims somewhere in between classical estimation, Kalman filters and particle filters. It will make use of prior information in the form of a PDF but be computationally very light. It need not be optimal and is expected to perform better than a classical estimator but worse than a particle filter or validly applied Kalman filter.

Chapter 3

Bayesian Theory, Prior Knowledge and the Algorithms

3.1 Overview

The theoretical groundwork needed is presented in this chapter beginning with a few results taken from Bayesian theory in section 3.2. How to represent prior information as a PDF and how to construct and interpret it is covered in section 3.3. Section 3.4 deals with the properties and choice of the weighting function. The estimator algorithm, the Static Prior Estimator or SPE, is presented in 3.5. Lastly two comparison algorithms, the LP and Bootstrap filters are described in section 3.6.

3.2 Important Results from Bayesian Theory

Bayesian theory has a number of basic ideas and results that were the starting point in the estimation approach taken in this thesis.

Prior and posterior PDF, Bayesian minimal estimator

Lets say we are interested in estimating or predicting values of a certain parameter θ. Representing for example the maximum pressure in the engine cylinder. Its value is stochastic, not constant and is thus modelled as a stochastic variable with a certain probability distribution.

Information about a parameter θ, or vector of parameters θ, is available from previous experience. All relevant aspects of this knowledge can be described by statistical properties, in particular by a probability density function, a PDF, of the parameter. Henceforth called *the prior PDF* or simply *the prior*.

As new measurements x are made of the parameter we wish to combine the new knowledge with the old. The new measurements, represented by a likelihood function $p(x)$, are combined with the old by Bayes theorem (2.2) yielding a *posterior PDF*, $p(\theta|x)$, containing all known information of θ after the data x has been observed.

The prior PDF naturally describes statistical knowledge about the parameter but other things may be modelled in this way as well. A uniform PDF as in figure 3.1a for example describes a physical constraint or boundary condition or the figure 3.1b which is the PDF of a variable with biased distribution and a lower bound. Or even as in the last figure where the underlying variable flips between two distinct values. It can thus be said that the PDF is a very versatile tool to describe knowledge of the behavior of a parameter but that it requires careful consideration of the problem at hand.

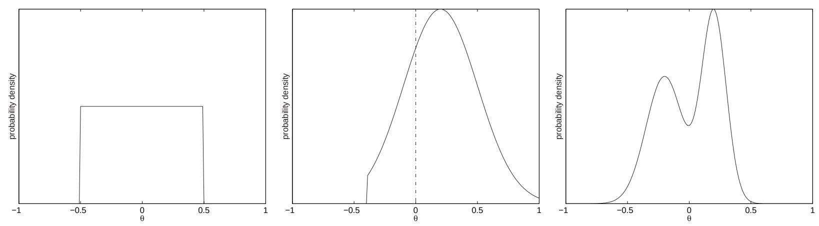


Figure 3.1. (a) Uniform PDF, (b) Uniform and biased gaussian PDF with lower bound. The line marks the true value of θ. (c) A multimodal PDF.

It is important to choose a valid prior PDF for the estimation. If the prior PDF is incorrect, if the mean is biased for example, estimates will also be biased and increase the error. Unless there is good reason to choose a particular PDF it is better to use classical estimation or use methods that start with a non-informative prior and adapt it as new measurements arrive. (See section 2.8 on Particle Filters and Sequential Monte Carlo methods.)

With the posterior PDF several interesting estimators can be formed. The most important one being the Bayesian minimal estimator which is the mean of the posterior PDF. This is the estimator that will minimize the mean square error. Other estimators can be found by taking for example the median or mode of the posterior PDF.

The Bayesian mean square error that is minimized by the mean estimator is defined as

$$\mathrm{BMSE}(\hat{\theta}) = E[(\theta - \hat{\theta})^2] = \iint (\theta - \hat{\theta})^2 p(x, \theta) dx d\theta \tag{3.1}$$

The BMSE is different to the classical mean square error as the BMSE is taken over the joint PDF $p(x, \theta)$. The BMSE is the MSE averaged over all realizations of θ and x. Thus the two types of estimators cannot be directly compared! [20]

A Bayesian estimator that is optimal in the sense that it minimizes the BMSE is called a minimum mean square error estimator or MMSE.

The gaussian prior PDF

A natural prior PDF to use is the gaussian. It is often the best approximation of the actual statistics and has a number of "friendly" properties. A Bayesian estimator can, unlike a classical estimator, always be found although it might not be possible to obtain a closed analytical expression. But if $p(\theta)$ and $p(x)$ are both gaussian the joint PDF $p(\theta|x)$ will also be gaussian and the Bayesian estimator, being the mean of the joint PDF, is straight forward to evaluate and will yield

$$\hat{\theta} = \frac{\sigma_\theta^2}{\sigma_\theta^2 + \sigma^2/N} x + \frac{\sigma^2/N}{\sigma_\theta^2 + \sigma^2/N} \mu_\theta \tag{3.2}$$

where σ^2 and σ_θ^2 are the variances of the measurements and prior respectively, μ_θ is the mean of the prior and N the number of measurement samples. See equation 10.11 and theorem 10.2 in [20].

Now let the x and y be gaussian random variables in a vector $\theta = [xy]^T$ with PDF

$$p(x,y) = \frac{1}{2\pi \det^{1/2}(\mathbf{C})} \exp\left(\frac{1}{2} \begin{bmatrix} x - E(x) \\ y - E(y) \end{bmatrix}^T \mathbf{C^{-1}} \begin{bmatrix} x - E(x) \\ y - E(y) \end{bmatrix} \right) \tag{3.3}$$

where

$$\mathbf{C} = \begin{bmatrix} \mathrm{var}(x) & \mathrm{cov}(x,y) \\ \mathrm{cov}(y,x) & \mathrm{var}(y) \end{bmatrix}$$

This is a bivariate gaussian PDF which has the contour plot in figure (3.2). From theorem 10.1 [20] we have that the conditional PDF $p(y|x)$ is also gaussian and has

$$E(y|x) = E(y) + \frac{\mathrm{cov}(x,y)}{\mathrm{var}(x)}(x - E(x)) \tag{3.4}$$

$$\mathrm{var}(x|y) = \mathrm{var}(y) - \frac{\mathrm{cov}^2(x,y)}{\mathrm{var}(x)} \tag{3.5}$$

additionally if x and y are not independent, i.e. $\mathrm{cov}(x,y) \neq 0$, the posterior variance of y becomes

$$\mathrm{var}(x|y) = \mathrm{var}(y)(1 - \rho^2) \tag{3.6}$$

where ρ is the correlation coefficient and satisfies $|\rho| < 1$

$$\rho = \frac{\mathrm{cov}(x,y)}{\sqrt{\mathrm{var}(x)\mathrm{var}(y)}} \tag{3.7}$$

As can be seen in equation (3.6) the variance of the posterior PDF will reduce with increasing correlation. The posterior mean, equation (3.4), then is the estimator $\hat{y}$ that after observing x also makes use of the correlation between the variables to yield an estimate of y.

The above also extends to cases when x and y are parameter vectors, $\mathbf{x}$ and $\mathbf{y}$.

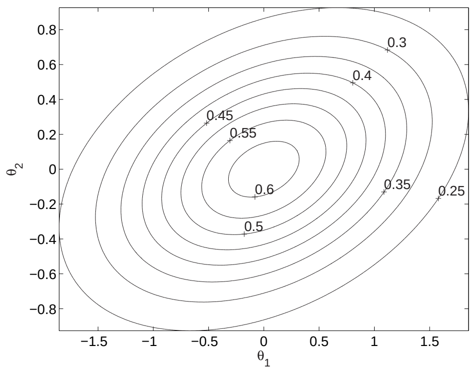


Figure 3.2. Contour plot of a gaussian 2d-prior. The contours represent constant probability density with the highest density in the middle. The inclination of the major axis represents the correlation between x and y.

Sequential Estimator

In a practical application, such as an engine measurement and control system, data arrives one sample at a time. When starting the measurement, or moving to a new operating point, we start with only the initial values and wish to improve the estimate as more data becomes available. A sequential estimator accomplishes this and is well suited for implementation as an iterative algorithm.

Deriving sequential Bayesian MMSE estimators that are valid in general is possible but rather complex [20]. Content with the fact that it can be done we proceed here with the special gaussian case only. Assume a random variable $\theta \sim \mathcal{N}(\mu_\theta, \sigma_\theta)$ and a discrete data model of the form

$$x[n] = \theta + w[n] \qquad n = 0, 1, ..., N-1 \tag{3.8}$$

where x is the measured sample at time n and w represents white gaussian noise with zero bias and standard deviation σ. N is the total number of samples. This is very similar to the classic case where θ is a fixed, deterministic parameter and the stochastics are due to measurements noise. Here θ is not constant but has some randomness in itself namely the system noise. Deriving the MMSE estimator $\hat{\theta} = E(\theta|x)$, after some algebra, results in equations (3.9) and (3.11). (See Kay [20], example 10.1 and 10.2 for details.)

$$\hat{\theta} \;=\; E(\theta|x) \;=\; \mu_{\hat{\theta}|x} \;=\; \alpha x + (1-\alpha)\mu_\theta \;=\; \mu_\theta + \alpha(x - \mu_\theta) \tag{3.9}$$

where the weighting factor α is defined as

$$\alpha \equiv \frac{\sigma_\theta^2}{\sigma_\theta^2 + \frac{\sigma^2}{N}} \Rightarrow 0 < \alpha < 1 \tag{3.10}$$

The posterior variance of $\hat{\theta}$ will then be

$$\mathrm{var}(\theta|x) = \sigma^2_{\theta|x} = \frac{1}{\frac{N}{\sigma^2} + \frac{1}{\sigma^2_\theta}} \tag{3.11}$$

The estimator in (3.9) reminds of a sequential algorithm since the new estimate is the previous μ_θ corrected by the error between the data and the previous estimate. The gain or weighting factor α reflects the relative confidence of the new measurement vs. our current knowledge. Indeed a slight change in notation results in the sequential Bayesian estimator below. (Although proving this is in fact more complicated and not discussed here. Again see Kay, chapter 12 [20].)

$$\hat{\theta}[n] = \hat{\theta}[n-1] + a(x[n] - b\hat{\theta}[n-1)] \tag{3.12}$$

The above is, with some slight modification, also valid if we take the viewpoint of the bivariate model.

Weighting functions

Taking one step back - even before we have formulated any kind of estimator we know that a new measurement will be affected by noise and we do not want to abandon our previous estimate completely when the new measurement arrives. Rather we wish to take a step in the direction of the new measurement. The size of the step depending on the relative confidence between new data and previous estimate.

The weighting function (3.10) used above is an excellent example of this. If we have gaussian conditions. As this is rarely the case we work with approximations at best. For this reason other weighting functions were tested. Their validity judged mainly by empirical means, i.e. testing and simulation. Specific weighting functions that have been tested are presented in section 3.4.

Although similar the weighting functions should not be confused with the so called *cost* and *Bayes risk functions* encountered in Bayesian statistics. The latter two are used to weigh the error when deriving a Bayesian estimator while the weighting function assigns a weight to relative probabilities.

3.3 The Prior PDF

As was mentioned previously using prior knowledge can improve the measurements from a sensor. The main purpose of using a prior PDF in estimation is to reduce the variance of the final estimates. The effect can be imagined as a concentration or pulling together of the estimates towards more probable values, in many cases similar to the behavior of an LP-filter, see figure 3.3. We will now focus on how to represent prior knowledge and how it can be obtained.

The use of a prior PDF and Bayesian estimation introduces the possibility to combine prior measurements with online measurements.

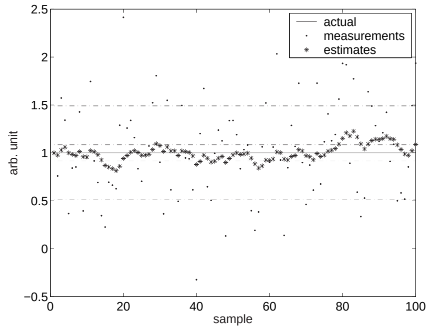


Figure 3.3. Example showing the concentration/reduced variance of the estimates when using an LP-filter. Dash-dotted lines represent the variance of the measurements and estimates respectively.

Constructing the Prior PDF

If we are to improve estimation using the Bayesian approach it is vital that the prior knowledge, the prior PDF, is accurate and reliable. So how do we construct or obtain this PDF in the first place? There are three distinct sources of knowledge to start from: measurements, physical understanding and experience.

The first and most common path is to obtain accurate measurements from a test setup. Here we can measure, test, prod and poke to our hearts desire with equipment that for economic or practical reasons will not be available outside the lab. If we conduct our tests well, anticipate and cover all or most scenarios that will be encountered in the real application we can construct an accurate map of our engine.

This map or set of measurements is prior knowledge which can easily be expressed in a PDF. A procedure for doing so is

1. Gather measurement data over, for example, varying ignition timing.
2. Collate the data in a histogram,see figure 3.4
3. Smooth and, if necessary normalize the histogram into a PDF.
4. Store the PDF in its histogram form, as a parameterized function or represented by statistical properties of a known distribution.
5. Repeat from 1. for all relevant operating points.

Leaving measurements, we may instead have knowledge of the physics governing the process. Or of relations between variables, for example spark timing and peak pressure location in an SI engine, or known limits to a parameter. Maybe even a working mathematical model. A relation between variables combined with knowledge of the system and measurement noise yields a statistical description of the relation that can represented by a PDF.

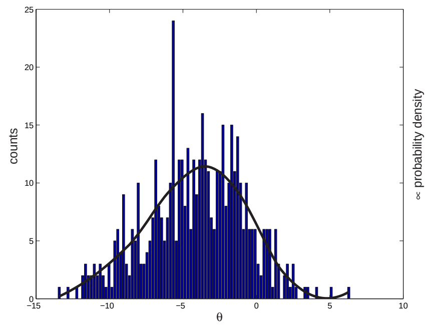


Figure 3.4. A histogram formed directly from measurements and the corresponding smoothed prior.

Lastly, professionals build up hands-on experience over the years. Experience that is often invaluable for tuning or extrapolation but difficult to incorporate due to its “fuzzy” nature. By manually constructing or adjusting the prior PDF the approach taken in this report provides a powerful tool to include this knowledge into the mathematical description of the system.

The shape of the prior PDF

The above mentioned ways to construct the prior can be combined by joining and manipulating the PDFs. Which brings us to discuss the possibilities of shaping the PDF to obtain certain effects and the errors that a misshaped PDF may give rise to.

Let us first define a few terms that will be used in describing the PDF.

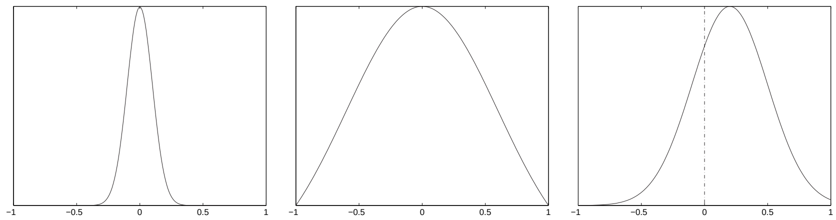
A sharp PDF has a small variance. It is narrow and sharply concentrated around one or several points. Reflects strong confidence in the prior knowledge. Figure 3.5a.

A smooth PDF has a large variance, is wide and smoothly spread out. Reflects weak confidence in the prior knowledge. Figure 3.5b.

A biased PDF has its mean at a value different from the actual. Figure 3.5c.

A PDF-like histogram can be used instead of an actual PDF. It serves the same purpose but is represented by a set of values corresponding to choice points on the PDF. Such a histogram can be made in several dimensions and can have any shape. See figure 3.4.

Shaping the prior is now a consideration of the following issues. A sharp prior will cause the estimator to rely more on measurements near the peak of the prior. In other words to rely heavily on the prior information. If the prior mean is the true mean then the bias of the estimates will be reduced. If not the true mean estimates



(a) sharp or thin prior (b) smooth or wide prior (c) biased prior

Figure 3.5. Three distinct shapes of a one dimensional prior

will be biased toward the prior mean. Variance reduction will be considerable in both cases. This is illustrated in figure 3.6.

A smooth prior in effect contains less prior information and the estimator will rely more on measurements. Estimates will follow measurements more closely and an incorrect mean will lead to less bias. Variance reduction however will be less than for the sharp prior, as is also seen in figure 3.6.

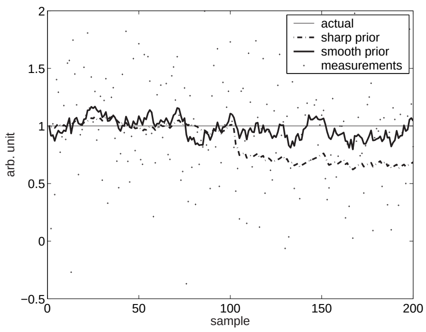


Figure 3.6. Stationary tracking with sharp prior vs. smooth prior. Prior biased after sample 100.

This trade-off, that variance reduction comes at the cost of tracking performance and bias is common to many estimation methods. Consider for example the simple moving average or low-pass filter. The benefit in the approach in section 3.5 is that the estimator can be adjusted to have good properties in ranges considered more probable at the cost of performance in improbable ranges.

If the prior has multiple peaks, like the bimodal histogram in figure 3.1c, the more probable values will be preferred. Tracking of a step from one peak to the other will be faster than from one peak to a value in between.

Depending on the implementation of the estimation algorithm, in particular the weighting function, areas may also be completely forbidden by setting the probability in that area to zero.

3.4 Choosing the Weighting Function

The weighting function tells us how much to trust the new measurement compared to the previous estimate and defines the general behavior of the estimator.

A closer look on the how and why follows. The prior contains information on how probable different values of a parameter are according to previous experience. Upon receiving a new measurement we judge its validity, based on our prior information, compared to the current estimate. This is achieved by taking the quotient of the prior values for the new measurement and the previous estimate.
Let

$$q_k = \frac{\mathit{Prior}(\text{new measurement})}{\mathit{Prior}(\text{previous estimate})} = \frac{P(y_k)}{P(\hat{\theta}_{k-1})} \tag{3.13}$$

This quotient gives a rough idea how much to trust the new measurement when forming the new estimate. This is similar to the interpretation of the weighting function α in section 3.3. However, a sequential update algorithm like (3.12) also requires that the weighting function α

- is dependent only of the quotient (3.13) and the shaping parameter σ.
- is bounded to $0 < \alpha < 1$.
- is near 1 for small quotients and approaches 0 for large quotients.

Since consecutive measurements are relatively equal, the main characteristics of the filter are determined by the behavior of the weighting function near quotients equal to 1. For which it should return values in the range $0.85 < \alpha < 0.98$ since this strikes a good balance between tracking and noise reduction. Compare with an LP-filter with a forgetting factor in that same range. Actually the choice depends upon the intended application but the above values can serve as an initial guideline.

Several choices of weighting function fulfilling the requirements above are available. Some possibilities are presented in figures 3.7 and 3.8. They are defined as

$$\begin{aligned}
\alpha(q_k, \sigma) & \quad = \\
\text{linear} & \quad \begin{cases} 1 - \frac{q_k}{\sigma} & q_k < c \\ 0 & \text{else} \end{cases} && (3.14) \\
\text{inverse} & \quad \frac{1}{\sigma q_k + 1} && (3.15) \\
\text{inverse quadratic} & \quad \frac{1}{\sigma q_k^2 + 1} && (3.16) \\
\text{negative exponential} & \quad e^{-\frac{q_k}{\sigma}} && (3.17) \\
\text{tanh} & \quad 1 - \tanh(\sigma q_k) && (3.18) \\
\text{square root} & \quad 1 - \sqrt{q_k/\sigma} && (3.19) \\
\text{hit or miss} & \quad \begin{cases} k & q_k < c, \quad 0 < k < 1 \\ 0 & \text{else} \end{cases} && (3.20)
\end{aligned}$$

The parameter σ is to be seen as a design parameter that can be adjusted according to the desired behavior of the estimator.

The above functions are examples. Any function fulfilling the requirements can be used, as implicated by the last plot in figure 3.8.

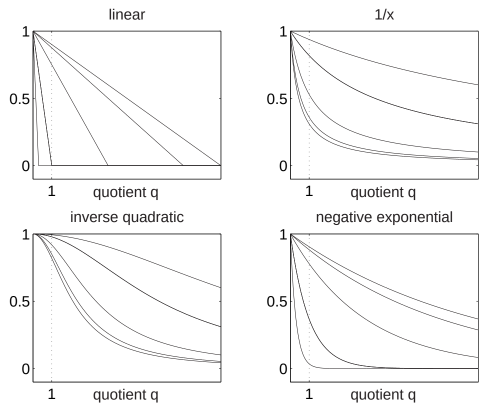


Figure 3.7. Four examples of sets of weighting functions. In each set the functions are evaluated for six increasing values of the parameter σ.*top left:* linear *top right:* exponential *bottom left:* square-root *bottom right:* square

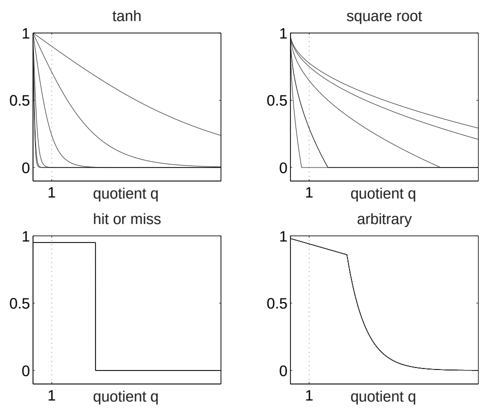


Figure 3.8. Four more weighting functions.*top left:* tanh *top right:* square root *bottom left:* hit or miss *bottom right:* arbitrary

The linear function is the easiest to implement but is not analytical, the negative exponential displays an appealing shape but can only be numerically approximated and the hit-or-miss function results in an interesting accept/reject special case. The specific choice is thus dependent on application and desired behavior and requires a bit of trial and error. Some of the listed functions will be tested on simulated data in the next chapter but for the most part the negative exponential will be used.

For the choice of σ it can be said that small values result in a fast filter and large values in a slow filter. For the negative exponential $\sigma = 8$ is a good starting point.

Lastly it should be noted that since both the weighting function and the prior affect the behavior of the estimator adjustments can be made in both. As a guideline the general behavior of the estimator, such as sensitivity or tracking speed, should be adjusted with the weighting function while specifics, such as behavior around a certain value, with the prior.

3.5 The Static Prior Estimator Algorithm

With the groundwork from the previous sections the algorithm implementation is now very straight forward. Consider again the sequential estimator in (3.12) now written as

$$\hat{\theta}_k = \alpha\hat{\theta}_{k-1} + (1-\alpha)y_k \tag{3.21}$$

where α is one of the weighting functions. This estimator now takes into account new and prior information and is extremely easy to implement. Its' behavior is similar to an LP-filter but with the advantages and drawbacks mentioned in section 3.3.

Algorithm 1 The SPE algorithm

0. The prior $P(\cdot)$ is known or assumed

1. Compute the quotient of relative probability (3.13):

$$q_k = \frac{P(y_k)}{P(\hat{\theta}_{k-1})}$$

2. Compute the weighting function

$$\alpha = w(q_k, \sigma)$$

   where σ is a tuning parameter.

3. Update the estimate

$$\hat{\theta}_k = \alpha\hat{\theta}_{k-1} + (1-\alpha)y_k$$

It is assumed that y_k are direct measurements of the parameter we wish to estimate. No system model is assumed. Noise properties are described by the prior. The SPE algorithm will always return a value that is between the previous estimate and the new measurement.

If the SPE is shown to deliver improved estimates it will be applied to the results Klövmarks nonlinear gaussian-fit PPL-estimator to *improve* those estimates. Thus y_k will be an indirect measurement in that it is first estimated by some other estimator.

Multiple Dimensions

The SPE algorithm can also be expanded into several dimensions. That is, modified to estimate several parameters at once. The prior is then the joint PDF of all parameters, $\mathbf{y}$ is a vector of measurements and $\hat{\theta}$ a vector of estimates. The quotient q remains a scalar.

In the two dimensional case the prior PDF can be presented as a contour plot like figure 3.2. Looking back to equation (3.6) we then hope to make use of the correlation between the two variables to improve the estimate of one variable with information from the other.

3.6 Comparison Algorithms

In evaluating the performance and properties of the algorithm described, answering questions like "What is an improvement?", "Better than what?" and "What is good or bad, I mean really... when you get down to it?" gets more difficult than one would initially expect. Therefore two existing algorithms will be presented here. The simple low-pass, or LP, filter will serve as a lower bound and the bootstrap filter as an upper bound in our evaluation. Our algorithm is expected to perform, in terms of mean square error, noise suppression and tracking, somewhere between the two.

The low-pass filter

The filter used here is a 1:st order LP-filter. It is described by the same difference equation that represents the estimator in (3.21) where α is a constant and represents the forgetting factor of the filter. To avoid confusion we rewrite (3.21) to describe the LP-filter as

$$\hat{\theta}_k = a\hat{\theta}_{k-1} + (1-a)y_k \tag{3.22}$$

A discrete time Z-transform of the above difference equation yields the pole-zero form

$$U(z) = \frac{z(1-a)}{z-a} \tag{3.23}$$

The step response of the LP filter with forgetting factors between $0.5 \leq a \leq 0.98$ shown in figure 3.6. As a increases the response is slowed, a behavior we will expect to see in the SPE as well.

This LP-filter is equivalent to a sliding mean where the forgetting factor corresponds to the length of the window.

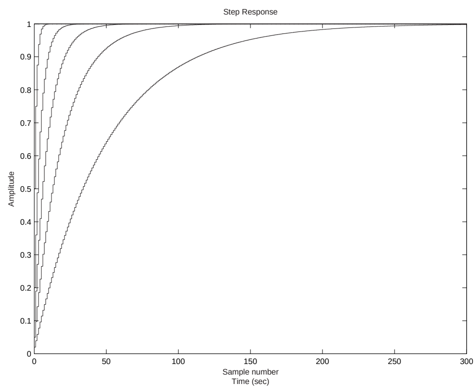


Figure 3.9. The step responses of the LP filter in (3.23). Forgetting factors a={0.5 0.8 0.9 0.95 0.98}.

The bootstrap filter

The background to particle filters was given in section 2.8. These filters are considerably more complex than the LP filter as well as the SPE and are consequently expected to perform better. The implementation of the bootstrap filter according to [15] is presented here.

The algorithm consists of two stages, prediction and update, and is initiated with an assumed PDF for the states, $p(\theta_1)$. The prediction stage takes a set of random samples of the PDF at time $k-1$ and feeds them through the system model (3.24) to form a prediction of the current state θ_k:

$$\hat{\theta}_k(i) = f_{k-1}(\theta_{k-1}(i), w_{k-1}(i)) \tag{3.24}$$

where samples $i = 1, \dots, N$ are from $p(\theta_{k-1})$ and $p(w_k)$ is known.

Next the predictions together with the new measurements are combined by Bayes rule (2.2) in the update stage. This is achieved by obtaining a normalized weight for each sample by the equation

$$q_i = \frac{p(y_k|\hat{\theta}_k(i))}{\sum_{j=1}^{N} p(y_k|\hat{\theta}_k(j))} \tag{3.25}$$

which approximates the new PDF by the discrete set of probability masses q_i. These new samples are resampled so that they are more scattered around important areas, i.e. areas of high probability. To do this a random sample u_i from the uniform distribution [0,1] is drawn. The value $\hat{\theta}_k(M)$ corresponding to

$$\sum_{j=0}^{M} q_j < u_i \sim U(0,1) \leq \sum_{j=0}^{M} q_j \tag{3.26}$$

is selected as a sample for the posterior. This is repeated for all i.

From the discrete distribution represented by the weights q an estimate of the current state can now be formed by normal statistical methods, such as taking the mean of the PDF.

Algorithm 2 The Bootstrap/SIR filter, as presented in [15]

0. An initial sample $\hat{\theta}_1^{(i)}$ of $i = 1, ..., N$ particles is drawn from $P(\theta_1)$ which is known or assumed.

1. Prediction: Each sample is passed through the system model (3.24) to obtain samples at k:
$$\hat{\theta}_k(i) = f_{k-1}(\theta_{k-1}(i), w_{k-1}(i))$$

2. Update: On receipt of the measurements y_k, evaluate the the likelihood of each sample and obtain a normalized weight for each sample by equation (3.25):
$$q_i = \frac{p(y_k|\hat{\theta}_k(i))}{\sum_{j=1}^{N} p(y_k|\hat{\theta}_k(j))}$$
and let these define a discrete distribution with probability mass q_i $i = 1, ..., N$.

3. Resampling: Resample N times from the discrete distribution to generate samples $\theta_k(i) : i = 1, ..., N$ so that for any j, Prob$(\theta_k(j) = \hat{\theta}_k(i) = q_j)$

4. Form an estimate of the state(s) from the resulting (marginal) posterior by, for example, ML- or MAP-methods.

5. Repeat from 2 for the next time step $k = k + 1$ using the samples θ_k.

The implementation used for this thesis was derived from the Matlab implementation by Nando de Freitas [8].

The bootstrap algorithm is an adaptive filter that can handle non-linear time varying systems with non-gaussian noise by relatively straight forward computations. Its accuracy however is heavily dependent on the number of samples and the quality of the resampling. For state vectors of larger dimensions the number of necessary samples *in each step* grows very quickly, slowing the calculation down [15].

Chapter 4

Algorithm Evaluation on Simulation Data

4.1 Overview

The evaluation of the SPE algorithm, by the criteria listed in 4.2, is contained in section 4.3. In sections 4.4 and 4.5 an adaptive and a two dimensional version of the algorithm is evaluated. Another extension of the SPE is discussed in section 4.6.

4.2 Evaluation Criteria

To be able to judge merit of any results we first define a number of criteria by which they are judged.

Tracking performance. The ability of the estimator to follow, or track, the true value over steps and ramps. A quick response and lack of bias is good.

Noise suppression / variance reduction. The estimators ability to smooth out and disregard noise in the signal. The effective opposite of sensitivity. Noise suppression is good if the estimator has a small variance despite large noise variance. There is always a trade-off between tracking and noise suppression where improvement in one comes at the cost of the other.

Comparison to LP-filter. The LP-filter from section 3.6 is used as comparison. If the forgetting factor a is chosen so that the noise suppression matches that of our estimator the tracking performance can be compared. Similarly the forgetting factor can be chosen so that the LP-filter matches the estimator in tracking performance, (in ranges of interest), after which the noise suppression is compared.

Comparison to bootstrap-filter. This more complex filter puts an upper limit to what we expect from our algorithm. Model noise variances are set according to what is expected. The number of state samples, particles, in each time step was set to 500. Less than 100 would not approximate the PDF well and more than 1000 would lead to cumbersome calculation times.

Root Mean Square Error, RMSE. Is used as a numerical measure of the estimator performance and is defined as $\text{RMSE}(\hat{\theta}) = \sqrt{Var[\hat{\theta}] + (E[\hat{\theta}] - \theta)^2}$

4.3 Single Dimension Static Prior Estimator

The first case that will be considered is the same as was used for discussion in section 3.3, a one dimensional single gaussian prior. The gaussian is chosen for convenience as it can be easily implemented. Arbitrary distributions can be used in the SPE. No significant benefit is expected in the gaussian case since it can be proven that the mean value is the most effective steady state estimator. This section aims more to understand the general behavior of the SPE and the effect of different weighting functions.

The two extremes

In the extreme the gaussian prior, or with some generalization any prior, can take two extreme shapes: completely uniform or completely collapsed as seen in figure 4.1a. In theory the first case would mean that the estimate would be completely independent of the prior and in the latter independent of the measurements. Figure 4.1b shows that this is indeed the case; the collapsed prior forces the estimate to stay at 15 and the wide, nearly uniform prior reverts the SPE to an LP-filter. The parameter to be estimated is θ.

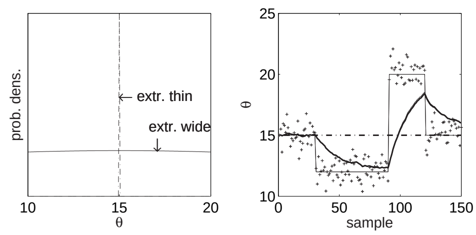


Figure 4.1. (a)The two extreme priors. (b) The resulting step tracking plot. The dash-dotted track is with the thin prior and thick solid with the wide prior. The setpoint is here and henceforth represented by a thin solid line. An exponential weighting function with $\alpha = 0.95$ was used.

Although it can not be discerned in the plot a LP-filter with forgetting factor 0.95 was plotted as well. It is completely overlapped by the SPE.

Simulations with different weighting functions

The choice of weighting function influences the behavior of the estimator. To discern how we tested six weighting functions on simulated data. The functions used were the exponential , linear, inverse linear, inverse quadratic, tanh and hit-or-miss from eq. (3.14-3.20). The parameter σ was chosen so that the value of the weighting function $\alpha(q, \sigma)$ was equal to a factor $\alpha = 0.9$ at $q = 1$. The prior was a gaussian with center at 15, $\sigma_P = 1$ and minimum level 0.1. Plot of the weighting functions and the tracking plot can be seen in figures 4.2 and 4.3.

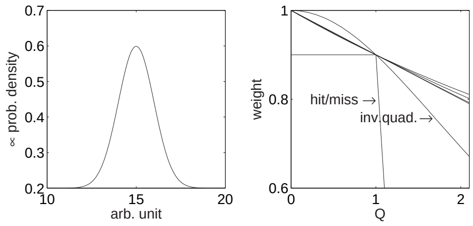


Figure 4.2. (a) The prior used. (b) The weighting functions. Q on the x-axis. Note their similarity in the important area $0 < Q < 1$. Also see the larger plot of the weighting functions, figures 3.7 and 3.8

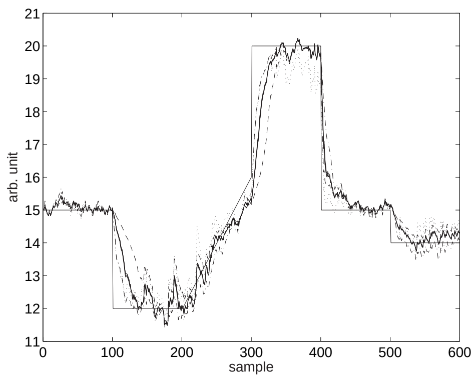


Figure 4.3. Tracking plot with the various weighting functions. LP-filter (dashed-dotted) and hit-or-miss (dotted) are fastest and inverse quadratic (dashed) slowest tracking while all others, (solid), are in between.

Several other configurations were also tested with similar results. The quotient Q is between 0 and 1 most of the time, notably so in steady state, and it follows that this is also the most important region of the weighting function.

Most weighting functions led to very similar tracking behavior, see figure 4.3, which is expected from their similarity in the $0 - 1$ range, see figure 4.2b. The two exceptions are the hit-or-miss, which displays erratic behavior particularly in the ramp, and the inverse quadratic.

The inverse quadratic is slower than the other in steps away from the prior and faster in steps towards. This follows directly from its shape, figure 4.2b. It is larger than the others for $q < 1$ and smaller for $q > 1$.

Consequently, the weighting functions' influence on the estimator was found to be mainly determined by its values in the range $0 < Q < 1$. All functions except the inverse quadratic and the hit-or-miss are nearly identical in that range and consequently so are the resulting estimators. Because of this the comparison LP-filter should have a forgetting factor a equal to the value of $\alpha(q = 1)$.

Other weighting functions may be feasible, and possibly lead to better estimators, but for this study only the tested six were considered. Of these only three discernible differences were noted. Namely the heavy prior dependance of the inverse quadratic, the erratic but LP-like of the hit-or-miss and the typical of the remaining. The exponential was chosen to represent the last four very similar functions and is the one used in all further evaluation.

Tracking comparison

The exponential was chosen as weighting function and the resulting estimator compared to the LP and bootstrap filters. The tracks were chosen to incorporate all or some of the elements: stationary tracking, step away from prior, step towards prior, small step away and a ramp. From theory it is expected that the error of the estimator will increase if the track contains many setpoints outside the prior since the estimator will be biased. Therefore stationary tracking and small steps, i.e. still within the more probable part of the prior, as well as large were evaluated.

Table 4.1 shows statistics for comparison of stationary tracking while figures 4.4 and 4.5 show tracking over small and large steps. The same prior as in figure 4.2a was used and the LP-factor was 0.90.

	SPE		**LP**		**Bootstrap**	
	RMSE	Var	RMSE	Var	RMSE	Var
On prior	0.1374	0.0158	0.2326	0.0494	0.1943	0.0356
Near prior	0.4272	0.0385	0.2900	0.0834	0.2545	0.0648
Far from prior	0.2772	0.0768	0.2408	0.0578	0.2231	0.0496

Table 4.1. Estimator comparison in three stationary cases. LP-filter forgetting factor was 0.90

The SPE is superior for stationary tracking on, or very near, the peak of the prior which is evident by the reduced variance at samples 0-100 in both figures and in the table 4.1. The bias, apparent at sample 300-400 in figure 4.4, is greatest near the prior and contributes to a large error. Far away from the prior the SPE returns to near LP behavior as the weighting function saturates.

The step response of the SPE is faster for steps towards the prior than away from it, as seen at samples 400 vs. 100 in figure 4.5.

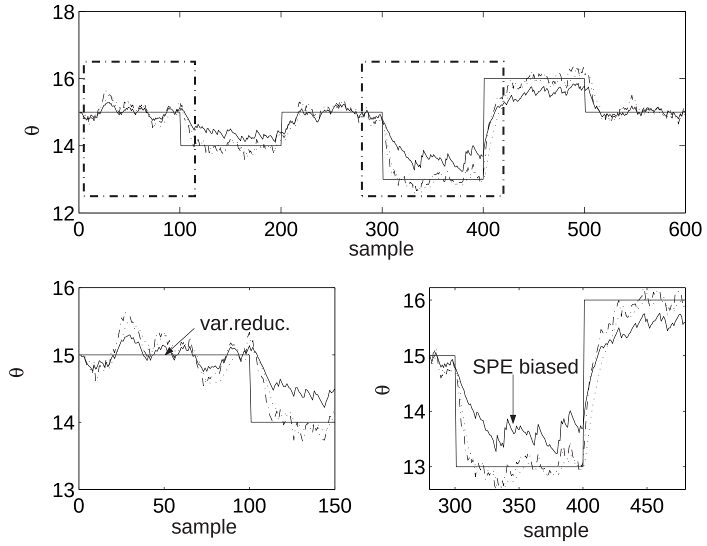


Figure 4.4. Tracking over small steps. SPE solid, LP estimate dash-dotted, bootstrap estimate dotted.
Settings: $\mu_p = 15$, $\sigma_p = 1$, noise $\sigma_\theta = 1$, LP-factor= 0.90, min(Prior)= 0.1

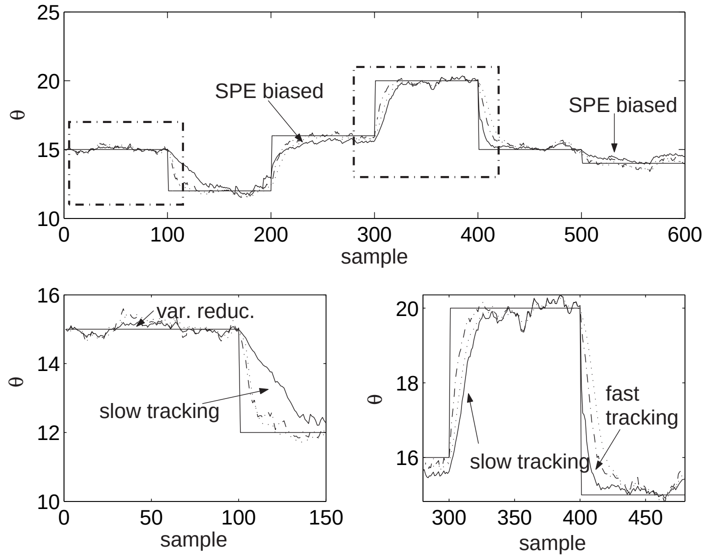


Figure 4.5. Tracking over large steps. SPE solid, LP estimate dash-dotted, bootstrap estimate dotted.
Settings: $\mu_p = 15$, $\sigma_p = 1$, noise $\sigma_\theta = 1$, LP-factor= 0.90, min(Prior)= 0.1

A few simulations with an asymmetric prior were also made with the result that this leads to less bias and more LP-like behavior on the heavy side of the prior.

Overall, it was found that the SPE is strongly biased toward the prior. This is the price for the reduced variance seen when estimating on the prior. A possible benefit is that steps toward the prior are followed faster by the SPE than the other estimators but that comes at the cost of being slower on steps away and that these properties can be strengthened and weakened by the choice of weighting function and prior.

Tracking with a bimodal prior

To investigate the possibility of using multimodal and other non-standard distributions as prior a bimodal prior of two gaussians was tested. The prior had peaks at 15 and 18 of standard deviation 0.7 and 0.5 respectively. A typical resulting tracking plot is presented in figure 4.6.

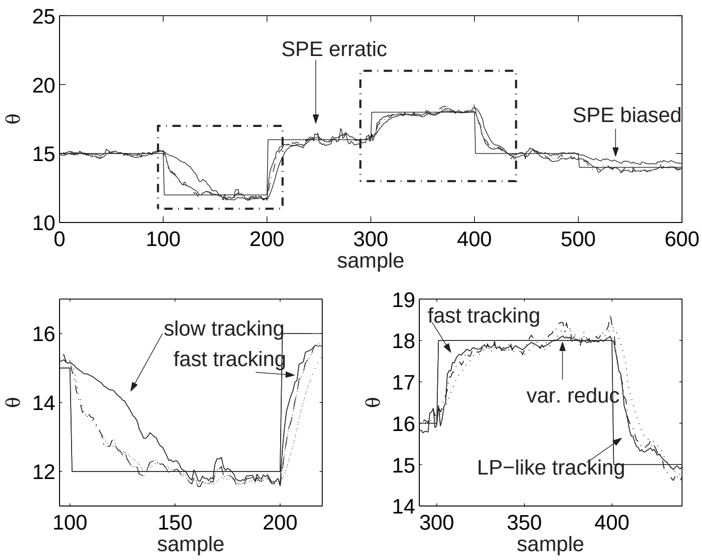


Figure 4.6. Tracking when using a bimodal prior. SPE solid and biased, LP estimate dashed, bootstrap estimate dotted.
Settings: Prior peaks = $\{15, 18\}$, Peaks $\sigma = \{0.7, 0.5\}$, noise $\sigma_\theta = 1$, LP-factor= 0.90, min(Prior)= 0.1

All features previously encountered with the gaussian prior SPE are present here. The slow tracking when stepping away from the prior, fast tracking when stepping toward the prior, persistent bias when off but near the prior and the variance reduction when on the prior.

The two new features are in the behavior between the two peaks in the prior, in this case at $\theta = 16$, and when stepping between the two peaks at sample 400. Between the peaks the SPE displays an erratic but unbiased behavior since measurements both above and below the setpoint are deemed quite probable. When stepping from one prior peak to another both old and new measurements are highly probable, resulting in a quotient q near one and consequently LP-filter behavior.

We conclude that a multimodal prior merely extends the behavior of the unimodal prior SPE into the regions where the prior is large (has high probability). No additional benefits were seen. On the contrary, behavior in between peaks in the prior was often erratic.

Section conclusion

The SPE is a biased estimator that can be used to reduce variance over a desired range. The SPE will be biased towards the probable area of of the prior and steps toward these areas will be faster and steps away from will be slower than a comparison LP-filter.

The bias can be made greater or lesser in different regions depending on the shape of the prior but a decrease in one region will inevitably lead to an increase in another.

The SPE is, as yet, not very useful compared to simply using an LP-filter since tracking and variance reduction can only be achieved in certain areas and the associated bias can be rather severe.

One common situation where the SPE does have an advantage and can be quite useful is if we wish to remain sensitive to measurements close to a known value yet insensitive to outliers far away. Severe outliers can totaly disrupt normal mean-value estimators but would have little or no effect on the SPE. The prior then serves as a "fuzzy" bound on the estimates which can be preferable to setting strict limits or if-then bounds.

Now looking ahead we wish to make the prior adaptive or use multidimensional priors which can make use of correlations between variables to improve estimates of one with help of the other.

4.4 Single Dimension Adaptive Prior Estimator

A simple modification to the SPE algorithm yields an adaptive version henceforth referred to as the Adaptive Prior Estimator or APE. Starting with a gaussian prior as in the previous section the center and variance are updated and the prior recalculated in each step. In this type of parametrization we describe the prior as a function, a gaussian in this case, and we loose the possibility of using arbitrary histogram distributions. If we wish to use histograms, parameters describing transforms may be used instead. For example the histogram prior may be translated, skewed or stretched. Other possibilities are also available. Which parametrization to choose would depend on the problem at hand.

The adaptive algorithm

The algorithm was expanded so that in each iteration

$$\begin{aligned}
\texttt{center} \quad \mu_P &= \hat{\theta}[k-1] = \texttt{previous estimate} \\
\texttt{standard deviation} \quad \sigma_P &= \texttt{std dev of n previous estimates} \\
\texttt{Prior} \quad P &= \mathcal{N}(\mu_P, \sigma_P)
\end{aligned}$$

where n was chosen between 1 and 20. To avoid collapse of the prior a lower limit of 0.5 was chosen for σ_P. Fixing σ_P and letting only μ_P vary was also tried.

The simulations

Variance reduction and tracking effects of this estimator were evaluated by a number of step simulations. A few typical results are presented in figures 4.7 through 4.9. In all except the last case the LP-factor was 0.90. A SPE with a gaussian prior with $\mu_P = 15$ and $\sigma_P = 1$ was used for comparison.

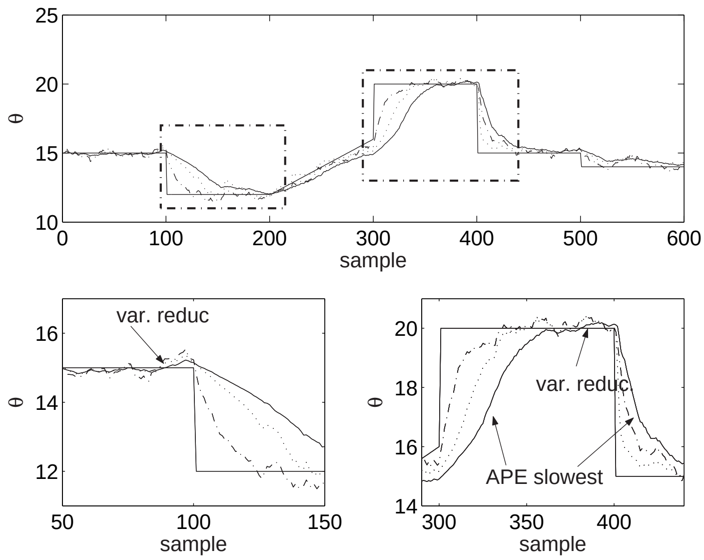


Figure 4.7. APE with adaptive center and sigma. The APE estimate has lower variance but is notably slower response time than other estimators. APE estimates solid, SPE dotted and LP-filter dash-dotted.

As can be seen in all the plots the adaptive estimator has lower variance and bias but is slower than the SPE and the LP-filter. An adaptive sigma should result in reduced variance of the estimator in steady state, but looking at the first 100 samples over several trails showed that the APE sometimes performs better than

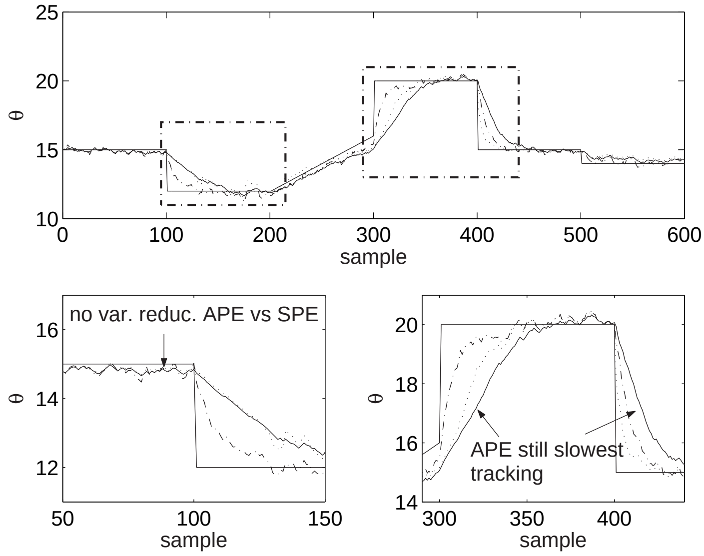


Figure 4.8. Estimator with adaptive center and static sigma. APE shows little or no variance reduction vs. SPE but still slower response time. APE estimates solid, SPE dotted and LP-filter dash-dotted.

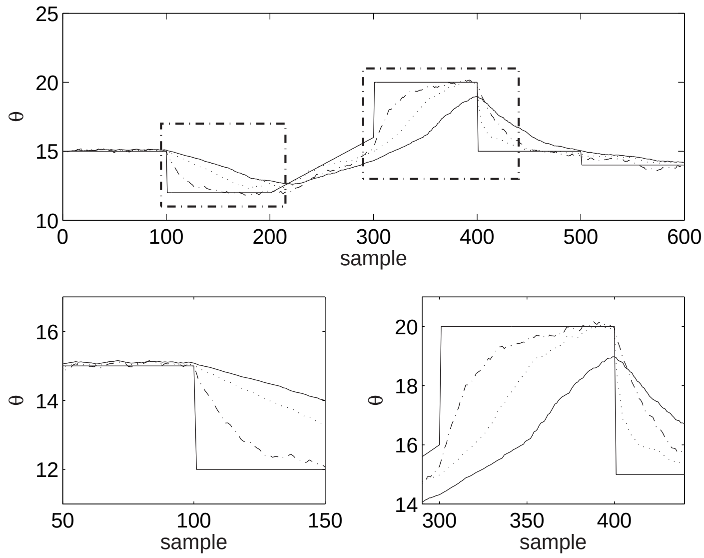


Figure 4.9. Estimator with adaptive center, adaptive sigma and LP-factor 0.95. Increasing the LP-factor further slows the APE estimate down. APE estimates solid, SPE dotted and LP-filter dash-dotted.

the SPE, (lower variance of the estimate), and sometimes worse, averaging out almost equal to the SPE.

Furthermore figure 4.9 shows that increasing the LP-factor slows the APE even more.

The APE with adaptive sigma, figure 4.7, can achieve a very sharp prior if allowed and thus potentially better stationary tracking than the static sigma APE. However, trying to improve the estimator by allowing very small sigma led to unstable and erratic behavior. Somewhat improved variance reduction can be seen in figure 4.7 vs. 4.8 due to allowing a slightly lower sigma in the first case.

To keep some level of visibility the bootstrap estimates were not plotted. Its results however were consistently better or even to the other estimators.

Section conclusion

It was seen from the simulations that an adaptive prior estimator scheme, the APE algorithm, can achieve marginally improved estimates compared to the SPE. In steady state variance was in some cases reduced and not reduced in other. As before, estimates were bias-free as when the prior was correct. The APE had a slower response time than the SPE estimators.

It should be noted that other ways of adapting the prior have not been explored. One idea would be to use an approximate prior consisting of discrete weights, the numerical equivalent of the histogram, and recalculate the weights for each arriving measurement so that a new PDF incorporating the new knowledge is formed at every time step. This would allow for arbitrarily formed PDFs and a flexible APE. However, this line of reasoning seems to lead to the particle filter and this is currently not our aim.

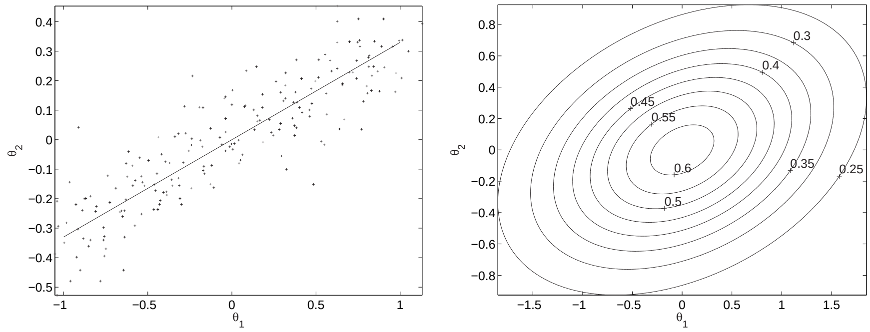
> **The APE has potential usefulness when it is more important that the estimates are stable despite severe noise and when the underlying parameter or state changes only slowly. The risk is that the APEs response is too slow for the intended application and that a bias is introduced if the prior is wrong. Overall the benefits of using an APE instead of an SPE or LP-filter are not convincing.**

Recalculating the prior in each step also increases the calculation time significantly. While not of any consequence in simulation this could pose a problem in an on-line implementation. Furthermore there is no way of guaranteing that the parametrization of the prior is sensible, or even relevant, to the problem at hand. Other algorithms, such as the Kalman-filter or a particle filter seem more appropriate.

4.5 Two Dimensional Static Prior Estimator

Apparently the benefits gained by using the SPE in one dimension were not overwhelming. So let us expand the algorithm into two dimensions and try to make use of the correlation between parameters to improve the estimates.

Suppose we now wish to estimate two correlated parameters, θ_1 and θ_2, with known distributions. From experiments we have the plot in figure 4.10a in which we can see a clear linear correlation. Say that a statistical analysis has given the correlation coefficient $\rho = 0.90$.



(a) (b)

Figure 4.10. (a) Correlation of two variables from a hypothetical experiment. The line shows the actual relationship between the variables. (b) A contour plot of a 2D PDF of two variables with linear correlation. Contours marked with corresponding probability density.

An estimator would benefit from making use of this linear correlation as is known from Bayesian theory and equation (3.6). If θ_1 and θ_2 have gaussian distributions their joint PDF will look like figure 4.10b. The correlation is seen as the inclination of the major axis of the ellipses.

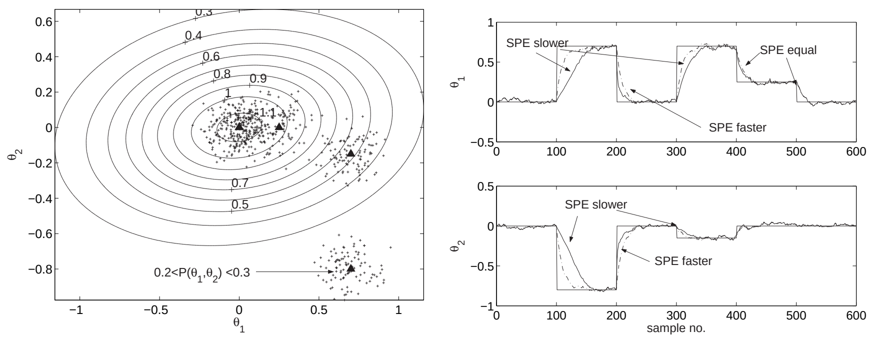
The 2D Static Prior Estimator

The SPE in two dimensions, or any number of dimensions, is not very different than in one dimension. In fact the only change is in the evaluation of the prior which now is a function of two variables, corresponding to two measurements. The quotient q and the rest of the algorithm is then evaluated in the same manner as before. The weighting function still yields a single value that is used in updating both estimates by equation (3.21).

The two dimensional SPE makes use of the correlation found in the prior simply by its tendency to be biased towards areas of high probability, i.e. the ridge formed along the major axis in figure 4.10b. It is the same property that biases the 1D estimator in section 4.3.

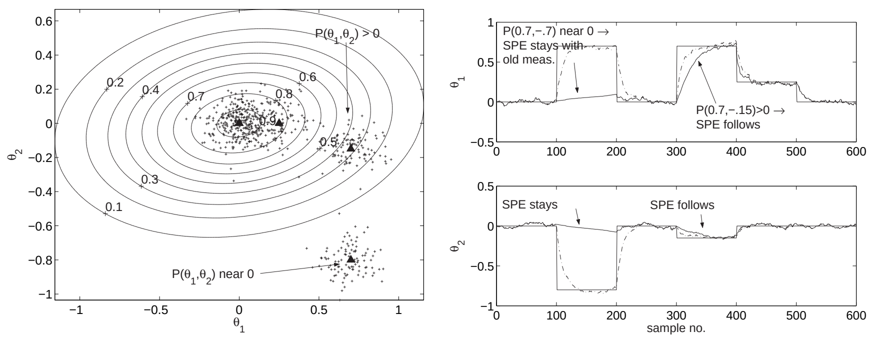
A number of simulations were performed with a two dimensional gaussian prior and correlated variables. The correlation was chosen to be very strong, $\rho = 0.99$. Figures 4.11 through 4.15 show typical outcomes. The tracks for θ_1 and θ_2 in the first two figures were chosen to include a setpoint at, a step away from and step a towards a highly probable area. Triangles in the prior denote the setpoints. The setpoint, rectangular solid track, was chosen to emphasize the differences in stepping between probable areas and improbable areas and from one to the other. Figures 4.11 and 4.12 have the same setpoints and show the effect of setting a lower bound to the prior.

In figure 4.11 it can be seen that the tracking properties of the 2D SPE are very similar to those seen in the one dimensional case. The SPE is faster in steps toward probable areas from improbable areas and slower in steps away. The important difference here being that the prior is a function of two variables, seen



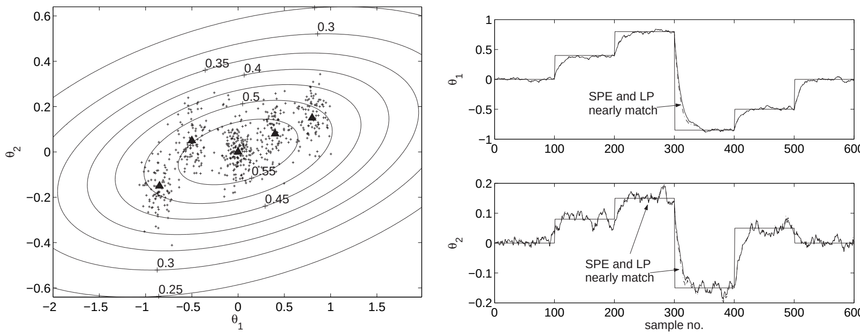
(a) Prior, measurements dotted (b) Tracking

Figure 4.11. Note the reluctance to move away from a probable area at 100-200, the fast step towards a probable area at 200 and the difference in response at 300 and 100. Triangles in (a) correspond to the setpoints in (b).
Settings: Noise levels (σ_1, σ_2) = (0.1,0.08), prior σ = (0.3,0.1), LP-factor: 0.9, min(prior) = 0.2



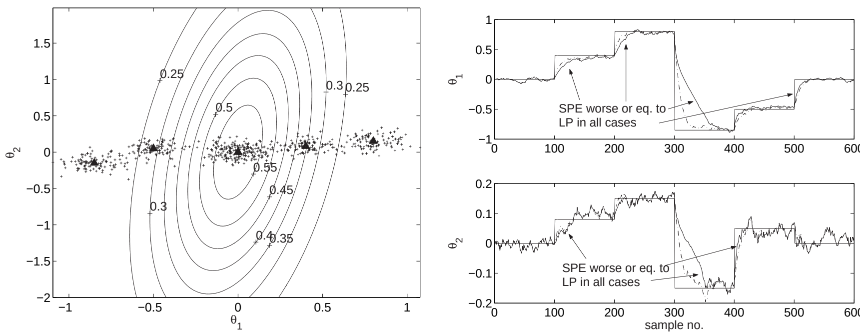
(a) Prior, measurements dotted (b) Tracking, SPE solid, LP dash-dotted

Figure 4.12. Note that the SPE estimates in (b) more or less refuse to move away from zero at 100-200 since measurements are deemed very improbable as seen in the lower right area of the prior in (a)
Settings: Noise levels $(\sigma_{\theta_1}, \sigma_{\theta_2})$ = (0.1,0.08), prior σ = (0.3,0.1), LP-factor: 0.9, min(prior) = 0.0



(a) Prior, measurements dotted (b) Tracking, SPE solid, LP dash-dotted

Figure 4.13. This track stays within probable area of the prior. Note the near match between the SPE and the LP-filter.
Settings: Noise levels $(\sigma_{\theta_1}, \sigma_{\theta_2}) = (0.1{,}0.08)$, prior $\sigma = (2{,}0.1)$, LP-factor: 0.9, min(prior) $= 0.2$



(a) Prior, measurements dotted (b) Tracking, SPE solid, LP dash-dotted

Figure 4.14. This track moves mainly in the improbable areas of the prior. Note that the LP filter performs better everywhere except at the step back to (0,0) at 500.
Settings: Noise levels $(\sigma_{\theta_1}, \sigma_{\theta_2}) = (0.1{,}0.08)$, prior $\sigma = (0.1{,}2)$, LP-factor: 0.9, min(prior) $= 0.2$

as coordinates in the prior, so the setpoint step to $(0.7, -0.1)$ at 300 leads to a different behavior than the step to $(0.7, -0.8)$ at 100.

In figure 4.12 the prior was allowed to approach zero in the outer areas. (In fact this is the only case in which the prior is a true PDF and integrates to one). As a consequence the SPE will not track steps toward these highly improbable areas, as seen in the interval 100-200.

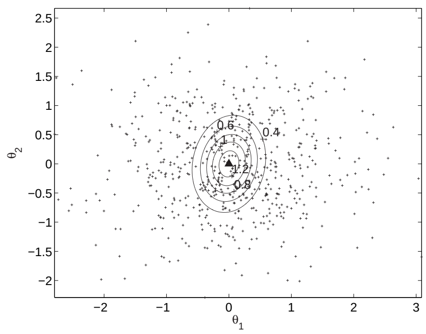
The setpoints in figure 4.13 where chosen to be within the highly probable areas and the resulting tracking plot shows that the SPE is nearly indistinguishable from the LP-filter, a distinctly different behavior than in 4.11. This follows directly from the fact that for consecutive, nearly equal values of the prior the quotient q will be near one where the weighting function is tuned to match the LP filter.

Conversely, figure 4.14 shows tracking between setpoints in improbable areas. As expected the SPE performs equal to or worse than the LP-filter.

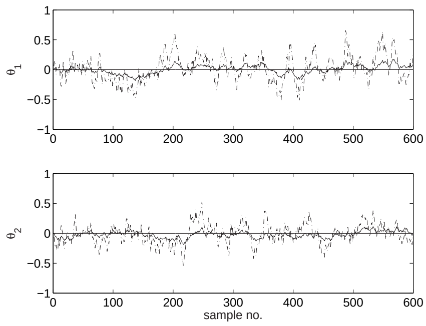
Finally, the variance reduction effect of the SPE is shown in figure 4.15. As in one dimension there is a trade-off between variance reduction and tracking.

Table 4.2 shows a comparison of stationary tracking performance for 2D and 1D SPE. The variance reduction effect of the prior is also compared by reducing the width σ_P of the prior by a factor 10 in both cases. This table shows a number of interesting things: the tenfold reduction of the prior variance has a greater effect on the 1D than the 2D SPE, the variance reduction of the estimates is less than tenfold and that there is a clear crossover effect between θ_1 and θ_2 in the 2D case - note that RMSE and variance of θ_1 in the 2D, first row, is lower than for the 1D SPE and vice versa for θ_2. This means that in the 2D SPE some of the accuracy in θ_2 has "spilled over" to θ_1 at the cost of accuracy in θ_2. In the second row however, that is with a tenfold reduced prior, the 1D SPE performs better in all cases.

A significant improvement in the estimates gained from the correlation of the



(a) Prior, measurements dotted



(b) Steady state variance reduction, SPE solid, LP dash-dotted

Figure 4.15. Variance reduction at a fixed setpoint. The prior is significantly smaller than the measurements noise, pulling estimates toward the more probable center at (0,0). Settings: Noise levels $(\sigma_{\theta_1}, \sigma_{\theta_2})$ = (1,0.8), prior σ = (0.1,2), LP-factor: 0.9, min(prior) = 0.2

		θ_1		θ_2	
		nRMSE	nVar	nRMSE	nVar
2D SPE	$\sigma_P = (0.3, 0.1)$	0.8483	0.7196	0.8010	0.6415
	$\sigma_P = (0.03, 0.01)$	0.3066	0.0937	0.2791	0.0779
1D SPE	$\sigma_P = 0.3, 0.1$	0.8942	0.7997	0.1743	0.0304
	$\sigma_P = 0.03, 0.01$	0.1684	0.0283	0.0106	0.0001

Table 4.2. Stationary estimator comparison. It is clear that a sharper prior reduced the error and variance and that it is more so for the 1D SPE.
Note: For ease of comparison all values were normalized by RMSE(LP-estimate) or var(LP-estimate). For example nvar(θ_{1SPE}) = var(θ_{1SPE})/var(θ_{1LP}). The same measurement data was used for all estimations. The numerics are representative of results from many (20+) runs. In the 1D case the estimates for θ_1 and θ_2 were calculated independently with the respective σ_P.
Settings: Measurement noise $(\sigma_1, \sigma_2) = (0.3, 0.1)$, LP-factor=0.90, min(P)=0.2

variables could not be seen in the plots. The results in table 4.2 suggest that we can benefit from using the 2D SPE but also that there is a trade-off between the involved variables - the accuracy in one is lost in benefit of improvement in the other.

> **Overall it was seen that the 2D SPE shows the same principal behavior as the 1D SPE. All properties shown in section 4.3 carry over to the 2D case. Because only the prior is changed with higher dimensions this is also likely to be true for an n-dimensional SPE, although this has not been proven.**
> **A significant improvement of the estimates of strongly correlated variables was not observed and we conclude that under the tested conditions this effect is not strong enough to motivate the use of a 2D SPE but that closer study may be required.**

Two Dimensional Adaptive Prior

Since the results of the 2D static estimator showed no significant improvement over the 1D and since it was shown that its properties were the same as the 1D SPE no attempt was made to construct a 2D APE. It is very likely that such an estimator would show the same properties as in the 1D case.

4.6 Dynamic Prior Estimator

An essential step in estimating the time evolution of a non-stationary state is the system model, equation (2.4). Using the system model, a fair prediction of the state at time k can be made from the known state at time $k-1$ in what is commonly referred to as the prediction stage of the filter. This procedure is used in the Kalman and bootstrap filters for example.

The known noise properties need to be propagated through the model together with the states. In the Kalman filter the mean and variance of the gaussian represent all knowledge of the noise and in the bootstrap filter the approximated PDF of the noise is propagated by means of discrete weighted samples.

The bootstrap and Kalman filters thus take the time-evolution of the system into account.

A dynamic prior estimator?

With the above in mind it would be preferable to make the prior used in previous sections dynamic in the same manner. So far we have not assumed any system model and the SPE and APE algorithms merely perform tracking with no knowledge of the time evolution of the states. This is a major drawback which we now try to amend.

It is clear that the prior must in some way be propagated through the system model, thus letting the prior evolve over time along with the system. The new prior could then be used as before, together with the prior at the previous time step, to form the quotient q after which we could proceed as in the SPE.

This raises the question of what to propagate. The process model is an equation and equations crunch numbers. The priors in the simulations above have, for convenience, been gaussian or gaussian sums and are thus represented by their respective mean and variances. Obviously this is what we want to input into the equations. This however is exactly what the Kalman filter was designed for! Or, in the case of multiple gaussians, the Gaussian sum filter [1].

With the prior described by a histogram, as proposed in section 3.3, an arbitrarily shaped prior can be used. This graphical histogram could be numerically described by a uniformly spaced weighted sum. The weights would then be propagated through the system model in some way and result in a new prior representing the probability of the states in the next time step. This is a direct parallel to the reasoning behind the bootstrap filter!

Furthermore, the heavier, more probable parts of the prior are more important than the tails and a non-uniform spacing should be used to so that more weights are in the probable areas. But the prior must not collapse if the probable area is very small and represented by few weights. This is, loosely, the reason for the resampling stage of the bootstrap and particle filters!

So, in trying reason as to how to construct a dynamic SPE, we encounter ideas and problems that have already been explored and solved in the area of particle filters. Thus further attempts in this direction were abandoned in favor of exploring existing solutions.

Section conclusion

While sensible, expanding the SPE into a dynamic algorithm with a prediction and an update stage leads to already existing algorithms such as the Kalman, gaussian sum or particle filter.

The Kalman filter is already proven to be optimal under gaussian conditions so we could only hope to be almost as good while requiring less computation. Much the same is true for the other algorithms.

It may be argued that the update stage of the proposed algorithm would differ from existing filters and be computationally lighter. Unfortunately most of the computational load arises from the prediction stage and thus we consider it very unlikely that any significant improvement can be made. While this possibility should not be completely abandoned without further research we will leave it at present.

Chapter 5

Results

The results from the previous chapters are now presented in view of the goals of section 1.1.

Algorithm implementation and evaluation

The proposed estimator, the SPE as presented in section 3.5, was successfully implemented as a Matlab script and run on simulation data.

Simulations presented in section 4.3 show that the 1D SPE can perform better than a first order LP-filter in certain cases. These cases are when moving from an improbable to a probable area of the prior. However, the static estimator is then biased toward the more probable areas and the increased tracking performance invariably comes at the cost of more bias. Furthermore, there is a trade-off between tracking performance and variance reduction that cannot be avoided.

The 2D SPE in section 4.5 has the same essential properties as the 1D SPE regarding tracking, bias and variance reduction. The benefit of moving to two dimensions; being able to make use of correlations between variables to improve the estimate, was found to show up as a trade-off between improving one variable at the cost of the other.

Comparison to current developments

Estimation problems similar to those in this thesis have been tackled before in various branches of estimation theory and its applications. The results from Bayesian theory that the proposed SPE algorithm builds upon, section 3.2, are essentially the same that are used in Kalman and particle filters.

Kalman filters are well known and preferable if the involved distributions are gaussian, gaussian sum filters if the distributions can be approximated by a sum of gaussians and particle filters if the distributions are arbitrary. All of these algorithms require considerable computation power. The SPE is far simpler than these algorithms, both to implement and compute, but it was found in section 4.3 and 4.5 that the successful use of an SPE requires trade-offs between several factors and that these must be judged in each specific case. To make these judgements accurate prior information is needed.

Incorporating particle filtering into an ion current based engine parameter estimator was not attempted as it lies outside the scope of this thesis.

For future work, developments in particle filters used for positioning and tracking applications are of particular interest, see section 6.

Since the prior information available was not accurate enough to make balanced trade-offs and good prior PDFs, improved and reliable engine PPL and AFR estimates could not be obtained with the SPE. Existing results from ion current based estimation methods mentioned in section 1.2 still stand. None of these use or make any mention of particle filtering however.

Ion current based estimates of engine parameters

The performance of the proposed algorithm, the SPE and the extension the APE, was in sections 4.3 to 4.5 shown to be a matter of trade-offs and quality of the underlying prior information. In section 4.6 that extending the SPE into a dynamic estimator, i.e. capable of modelling the time-evolution of the system, led to problems whose solution essentially are the Kalman or particle filters and was thus not further explored. Since a prior good enough for satisfying estimates was not available and could not be obtained this goal was abandoned.

Chapter 6

Future Work

The important argument in favor of the 2D SPE, that correlations between variables can be used to improve the estimate, was shown to be a matter of balancing an accuracy trade-off. Another trade-off between accuracy, tracking and bias must also be made both for the 1D and 2D SPE and so makes use of the SPE difficult as it stands. However, in cases where a good balance in these trade-offs can be achieved the SPE can be useful. Thus future work using the SPE should first concentrate on how to achieve this balance.

While the APE in section 4.4 did not improve over the SPE there are other ways of implementing the adaptiveness which are as yet unexplored.

There are also several developments in the area of particle filtering that have promising potential. Some of the ideas and alternatives that have cropped up in the course of this work are presented here. These are possible starting points for future work only and may or may not stand up to scrutiny.

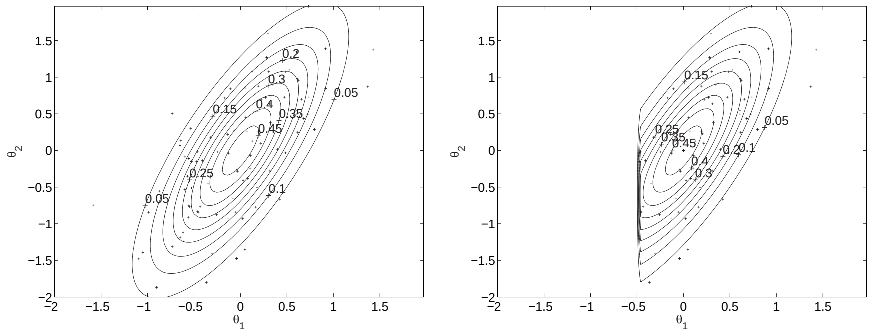
Bootstrap filter with constraints on the particles

The bootstrap, or SIR, filter was presented in section 3.6. The particles $\hat{\theta}^{(i)}$, (where $i = 1, \ldots$,number of particles), in the algorithm 2 are unrestricted, i.e. can take on any values. A possible way to incorporate knowledge of physical bounds or other limitations, such as strict correlations between variables, would be to restrict the particles generated by steps 1 and 4 to lie within a bounded set

$$\hat{\theta}^{(i)} \quad \epsilon \quad \Omega \quad \forall i. \tag{6.1}$$

Since the particles $\hat{\theta}^{(i)}$ represent a discrete approximation of the PDF of true state θ this is equivalent to a "cut-off" bound on the PDF, as illustrated by figure 6.1b.

The information about this restriction thus becomes incorporated into the estimator. One can also imagine the set to be dependent on some other variable to reflect changing restrictions with changes in the environment.



(a) $x \epsilon R^2$ (b) $x \epsilon \Omega, \quad \Omega : 0.5 < \theta_1 < \infty, \theta_2 \epsilon R$

Figure 6.1. (a) The particles and thus the prior are unbounded. (b) A bounded set Ω, where θ_1 has been restricted to reflect a lower bound.

Particle filters in navigation

Particle filtering features frequently in image processing publications and its applications. Often the aim is to combine visual images from a camera with knowledge of an objects velocity and direction, to track the object against a noisy background.

In Karlsson [19] such an approach was developed to aid submarine navigation. Particle filtering was used to combine sonar data with a depth map to yield accurate estimates of the submarines position.

The parallel to our case is straight-forward. The depth map is the prior, the appropriate engine map for example, and the sonar data is replaced by the ion current measurement. Instead of position we are trying to estimate engine parameters. The approach seems feasible if they can be expected to behave nicely the way a physical object might, i.e. the position of the sub does not jump from one end of the map to another from one measurement to the next.

Further development in this direction is of interest, preferably with collaboration with researchers in the mentioned field, and may well yield an strong ion current based engine parameter estimator. However, the computational cost for any such solution will be considerable and much greater than currently available in an ECU.

A recommended starting point

In Doucet [10] it is mentioned that the importance function of the SIR or SIS filter in some cases has been chosen in a way that is very similar to the static prior in the SPE. The conclusion there is also similar, that the results are rather poor as the dynamic of the model nor the observations are taken into account.

The paper by Doucet [10], containing a comprehensive overview of sequential Bayesian estimation methods up to 1998, is an excellent starting point for any future work in this area.

The publications and work of Karlsson [19], Hendeby [18] and Gustafsson [16] cover navigation problems, fundamental limits and many other aspects of interest.

Chapter 7

Conclusions

In this thesis an LP-like filter algorithm for ion current based estimation of engine parameters was implemented, evaluated in simulation and compared to existing Bayesian estimators and a first order LP-filter. The results showed that as improvements were a matter of a trade-offs - variance reduction at the cost of bias and a slowed tracking performance, accuracy improvement in one variable at the cost of accuracy of the other - a balance must be found in each specific application and the prior information must be accurate for the approach to be worthwhile.

A side benefit of the SPE is that the algorithm is well suited as a filter against outliers as these can be handled nicely, affecting but not disrupting the estimate, without resorting to if-then checks or strict limits.

Regarding the APE, there may well be other unexplored ways of implementing an adaptive algorithm which can yield better estimates.

Extending the original non-dynamic algorithm into a dynamic one was seen to lead in the direction of already existing estimators like the particle filter.

While the proposed algorithm came somewhat short of expectations, new and exiting possibilities were encountered on the way. The re-emergent area of particle filters shows much promise as similar problems have been solved to tackle robot navigation, image recognition and wireless signal processing. Of special interest are the uses of particle filters in map-aided navigation since it is straight forward to imagine the engine map as a terrain map and the ion current measurement as radar information in the virtual parameter landscape. Computational cost is heavy by current standards but not overwhelming. It can be expected that enough processing power for particle filters is available in the ECU within five years or that dedicated electronics can be feasible if the benefits are great enough.

Abbreviations

AFR	Air/Fuel Ratio
APE	Adaptive Prior Estimator
BMSE	Bayesian Mean Square Error
CAD	Crank Axle Degrees
ECU	Engine Control Unit
EGR	Exhaust Gas Recycling
EKF	Extended Kalman Filter
LP	Low Pass
MAP	Maximum A Posteriori
MCMC	Markov Chain Monte Carlo
ML	Maximum Likelihood
MSE	Mean Square Error
MMSE	(Bayesian) Minimum Mean Square Error
MVU	Mean Value Unbiased
nRMSE	Normalized Root Mean Square Error
PCA	Principal Component Analysis
PDF	Probability Density Function
PF	Particle Filter
PPL	Peak Pressure Location
RMSE	Root Mean Square Error
RPM	Revolutions Per Minute
SMC	Sequential Monte Carlo
SPE	Static Prior Estimator
TDC	Top Dead Center
UKF	Unscented Kalman Filter

Bibliography

[1] B.D.O. Andersson and J.B. Moore. *Optimal filtering*. Prentice-Hall, Englewood Cliffs, NJ, 1979. ISBN 0-13-345711-7.

[2] I. Andersson. Cylinder pressure and ionization current modeling for spark ignited engines. Technical report, Linköpings Universitet, 2002. LiU-TEK-LIC-2002:35, Thesis No. 962.

[3] J. Auzins, H. Johansson, and J. Nytomt. Ion-gap sense in misfire detection, knock and engine control. *SAE 950004*, 1995.

[4] M. Alden B. Johansson, H. Neij and G.Juhlin. Investigations of the influence of mixture preparation on cyclic variations in a si-engine, using laser induced flourescence. *SAE 952463*, 1995.

[5] S. Byttner. Algorithms for ion current based sensing of combustion variability and preassure peak position. Technical report, Chalmers University of Technology, 2003. No. 482L.

[6] Y. Chen and T.L. Lai. Fast particle filters and their applications to recursive identification and adaptive control of nonlinear stochastic systems.

[7] R.M. Clements and P.R. Smy. The variation of ionization with air/fuel ratio for a spark-ignition engine. *J.Applied Physics*, 47, 1976.

[8] N. de Freitas. Sequential sampling-importance resampling (sir) software. URL: `http://www.cs.ubc.ca/~nando/software/demo1.tar`, sep 2005.

[9] P.M. Djuric, J.H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M.F. Bugallo, and J. Miguez. Particle filtering. *Signal Processing Magazine, IEEE*, 20:19–38, 2003. ISSN: 1053-5888.

[10] A. Doucet. On sequential simulation-based methods for bayesian filtering. Technical report, Cambridge University Department of Engineering, 1998.

[11] L. Eriksson. *Spark Advance Modeling and Control*. PhD thesis, Linköping University, May 1999. ISBN 91-7219-479-0, ISSN 0345-7524.

[12] L. Eriksson and L. Nielsen. Ignition control by ionization current interpretation. *SAE 960045*, 1996.

[13] A. Franke et al. The effect of in cylinder gas flow on the interpretation of the ionization sensor signal. *SAE 031120*, 2003.

[14] J. Geweke and H. Tanizaki. On markov chain monte carlo methods for non-linear and non-gaussian state-space models. *Communications in Statistics, Simulation and Computation*, pages 867–894, 1999.

[15] N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing, IEEE Proceedings F*, 140:107–113, 1993.

[16] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssel, J. Jansson, R. Karlsson, and P.J. Nordlund. Particle filters for positioning, navigation, and tracking. *Transactions on Signal Processing, IEEE*, 50, 2002.

[17] M. Hellring, T. Munther, T. Rögnvaldsson, N. Wickström, C. Carlsson, M. Larsson, and J. Nytomt. Robust afr estimation using the ion current and neural networks. *SAE 991161*, 1999.

[18] G. Hendeby. Fundamental estimation and detection limits in linear non-gaussian systems. Technical Report Licentiate Thesis no. 1199, Department of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden, Nov 2005.

[19] R. Karlsson. *Particle Filtering for Positioning and Tracking Applications*. PhD thesis, Linköping University, Sweden, Mar 2005.

[20] S.M. Kay. *Fundamentals of Statistical Signal Processing*. Prentice-Hall, 1993. ISBN 0-13-345711-7.

[21] H. Klövmark. Estimating air/fuel ratio from gaussian parametrizations of the ionization currents in internal combustion si engines. Master's thesis, Chalmers University of Technology, 1998. Thesis no. EX 065/98.

[22] G.W. Malaczynski and M.E. Baker. Real-time digital signal processing of ionization current for engine diagnostic and control. *SAE 031119*, 2003.

[23] NIST/SEMATECH. E-handbook of statistical methods. URL: `http://www.itl.nist.gov./div898/handbook/`, sep 2005.

[24] R. Reinmann, A. Saitzkoff, and F. Mauss. Local air-fuel ratio measurements using the spark plug as an ionization sensor. *SAE 970856*, 1997.

[25] A. Saitzkoff, R. Reinmann, F. Mauss, and M. Glavmo. In-cylinder pressure measurements using the spark plug as an ionization sensor. *SAE 970857*, 1997.

[26] A. Smith, A. Doucet, and N. de Freitas. *Sequential Monte Carlo Methods in Practice*. Springer, 2001. ISBN 0-38-795146-6.

[27] L. Ljung T. Glad. *Reglerteori*. Studentlitteratur, 1 edition, 1997. ISBN 91-44-00472-9.

[28] R. van der Merwe and E.A. Wan. The square-root unscented kalman filter for state and parameter-estimation. Technical report, Oregon Graduate Institute of Science and Technology, 2001.

[29] N. Wickström, M. Taveniku, A. Linde, M. Larsson, and B. Svensson. Estimating peak pressure position and air-fuel ratio using the ionization current and artificial neural networks. *Proceedings of IEEE Conference on Intelligent Transportation Systems*, nov 1997.