MASTER

Online fuel-efficiency optimization of an RCCI engine using constrained extremum-seeking control

van de Wauw, Tim P.M.

*Award date:*
2022

*Report of the Graduation Project*

*Thesis*

# Online fuel-efficiency optimization of an RCCI engine using constrained extremum-seeking control

Student:

ing. T.P.M. van de Wauw
Student ID: 0976076

Supervisors:

prof.dr.ir. F.T.P. Willems (TU/e & TNO)
dr.ir. L.M.T. Somers (TU/e)
ir. M.G. Vlaswinkel (TU/e)

Eindhoven, June 2022

# Abstract

The overall goal of all internal combustion engines, including reactivity controlled compression ignition (RCCI) engines, is to deliver the desired torque and maximize fuel efficiency while meeting emission and safety constraints. To reach this goal, advanced combustion concepts, such as RCCI, require advanced combustion control, because of their relatively unstable combustion and present disturbances, such as cycle-to-cycle variations. A disadvantage of current control strategies is their reliance on controlling combustion parameters (e.g. IMEP, CA50, intake pressure) to achieve optimal efficiency and cannot control the efficiency directly. This work presents an approach for online, model-free fuel-efficiency optimization for steady-state RCCI operation, using constrained extremum-seeking control. It directly controls and optimizes the gross indicated efficiency (GIE) using two fuel path parameters, namely the blend ratio (BR) and the start-of-injection of the direct injected fuel ($SOI_{DI}$). It does this for a constant total fuel energy and engine speed. A key element of this approach is a cost function based on GIE, in which the safety and input constraints are incorporated as soft constraints. The safety constraints are the maximum pressure and peak pressure rise rate and the input constraints are the upper and lower bound of the inputs BR and $SOI_{DI}$. Simulations using a data-based RCCI combustion model are used to test the algorithm on the following performance criteria: reaching the highest GIE, meeting the constraints and achieving a convergence speed at least fast enough for effective use on highways, based on the Common Artemis Driving Cycles ($\leq 5$ seconds). These simulations show robustness with respect to disturbance of initial fuel path conditions and cycle-to-cycle variation, combined with an average convergence speed of approximately 1 second, while meeting constraints and reaching the highest GIE (47.77%). Finally, two steps in intake pressure (increase and decrease) are simulated to investigate the algorithm's ability to deal with changing cost functions. Even though the algorithm is not intended to deal with transient operation, it only needs roughly 2 seconds to converge to the new global minimum for both steps. This shows that there is potential for this method in handling transient operation.

# Contents

# List of Figures

# List of Tables

# 1 Introduction

## 1.1 Background

One of the main challenges the automotive industry is still focusing on, is improving the internal combustion engine (ICE) to make it cleaner in terms of pollutant emissions (e.g. $NO_x$, soot) and minimize operational costs by minimizing fuel consumption. The increasingly strict emission legislation causes traditional ICEs to be insufficient which leads to a rise of advanced combustion concepts.

## 1.2 RCCI

Reactivity controlled compression ignition (RCCI) is one of those advanced combustion concepts that promises to meet the emission legislations. RCCI is a low-temperature combustion concept (other examples are homogeneous (HCCI) and premixed (PCCI) charge compression ignition) that has shown great potential in reducing the $NO_x$ and soot emission in combination with high thermal efficiency [1]. It is a premixed ignition concept that blends a low reactivity (e.g. gasoline or natural gas) early in the process in order to mix fuel and a high reactivity (e.g. diesel) fuel later in the process in order to auto-ignite the air-fuel mixture. Because RCCI relies on auto-ignition in combination with a relatively high ignition delay, controlling the combustion is much more complex than for a regular diesel engine.

The overall goal of all ICEs, including RCCI, is to deliver the desired torque and maximize fuel efficiency while meeting emission and safety constraints. To reach this goal, advanced combustion concepts, such as RCCI, require advanced combustion control, because of present disturbances (in e.g. ambient conditions, cylinder-to-cylinder differences for lambda or EGR), cycle-to-cycle variation (consequence of stochastic process due to controlled auto-ignition) and to guarantee stable combustion [2]. This means that, for example, a simple open-loop controller based on lookup tables is insufficient, because it is unable to handle disturbances and changing ambient conditions.

## 1.3 RCCI control challenges

Traditionally, combustion parameters derived from the in-cylinder pressure profile are used to control ICEs by controlling combustion parameters, for example the indicated mean effective pressure (IMEP) or the crank angle at which 50% of the heat of combustion is released (CA50). To achieve the goal of higher efficiency and less pollutant emissions while dealing with disturbances and cycle-to-cycle variations, more and more manipulated variables are applied, which leads to a significant increase in system complexity [2]. To achieve this and deal with disturbances and cycle-to-cycle variations, next-cycle control strategies are developed. Examples of such next-cycle control strategies for RCCI are PID control of IMEP, CA50 and temperature of the intake manifold $T_{im}$ [3, 4]. A disadvantage of these control strategies is that they rely on controlling combustion parameters to achieve optimal fuel efficiency and cannot control the efficiency directly. During engine calibration, a function is created from driver input to a reference for a combustion parameter, for optimal efficiency. However, the reference may not be optimal for different ambient conditions. By controlling and optimizing efficiency directly, this can be resolved.

Another way to achieve optimal fuel efficiency are optimization methods to find the inputs to achieve optimal efficiency in RCCI engines under constraints, such as safety limits. Xia et al. [5] presents an offline optimization method, using a combination of multiple regression analysis and particle swarm optimization. Using this method the optimal operating conditions (exhaust gas recirculation (EGR) rate, start-of-injection of direct injection ($SOI_{DI}$) and blend ratio (BR)) for optimal gross indicated efficiency (GIE) were determined under emission and safety constraints. In a following paper of Xia et al. [6], the method is extended to realize an optimal feedforward control that can account for various uncertain conditions such as deviated fuel injections and intake pressure. The latter is realized by considering the input uncertainties in the robust optimization. However, these

methods are offline optimization methods, which create a mapping from optimal operating inputs to optimal GIE for multiple operating points in a lab setting, for it later to be used on the road. This means it cannot deal with changing ambient conditions or changes caused by hardware wear, because different inputs might be needed to achieve optimal GIE.

An online optimization method can be used to directly optimize and control GIE and overcome the aforementioned issue that come with classic feedback control methods and offline optimization methods. Extremum-seeking (ES) could be a method which can achieve this. ES is a model-free-based adaptive control method used to determine and maintain the extremum value of a (nonlinear) function and optimize the input-output characteristic when knowledge of the input-output is limited [7, 8]. Van der Weijst et al. [9] provides an example of constrained, multivariable ES for online fuel-efficiency optimization of a diesel engine. In this case, an optimization problem, solved using ES, is integrated into a tracking control system that tracks IMEP, CA50, $NO_x$ and pumping mean effective pressure (PMEP). The online ES controller adjusts the CA50 and PMEP references to minimize the cost function. Additionally, Peter [10] presents a constrained time-varing extremum-seeking control (ESC) approach for a class of discrete-time nonlinear systems, which has the ability to guarantee feasibility of the system's trajectories during optimization.

## 1.4    Objective

The goal of this study is to develop a control system that achieves online optimization of GIE using the fuel path while meeting safety constraints in an RCCI engine. In contrast to previously mentioned sources, the control system should be able to correct for changing ambient conditions or wear of parts. This is done by using constrained, online ESC. During this study, a data-driven combustion model of an RCCI engine is used and it will be assumed that the total energy input of the engine is constant. As the model is not able to calculate emission levels, emissions constraints are omitted from the optimization problem.

## 1.5    Outline of report

The outline of this report is as follows. First, the used combustion model and adaptive control problem, including the optimization problem, are introduced. Next, the theory behind the optimization algorithm and its tuning guidelines are presented. The next chapter presents and discusses the results regarding the performance and robustness of the algorithm. Finally, based on these results, the main findings are summarized in the conclusion and directions for future research are suggested.

# 2 Problem and setup description

## 2.1 System description

Figure 2.1 shows the six-cylinder PACCAR MX13 engine that is used to acquire data, which is used during this study. As shown in the figure, only one cylinder is actively fueled. To enable RCCI operations, the engine uses a direct injection (DI) system to inject diesel into the cylinder and a port fuel injection (PFI) system to inject E85 (85% ethanol, 15% gasoline) into the intake port. Additionally, EGR can be used to recirculate exhaust gas into the intake manifold. Boosted dry air is provided by an external compressor and the intake manifold pressure can be adjusted using a regulator. The engine is kept running at a constant speed by an electric machine that generates the required torque. The in-cylinder pressure and intake and exhaust pressure are measured using pressure sensors. Emissions (e.g. $CO_2$, $NO_x$, total hydrocarbon content (THC)) are measured after the back pressure valve using specialised hardware. Fuel consumption is measured by mass flow sensors. See Willems et al. [11], for more details about the setup.

## 2.2 Combustion model

As the test engine setup is not available, a data-based combustion model, presented in detail by Vlaswinkel et al. [12], is used to simulate the RCCI combustion process. The engine and fuel specifications and operating conditions from the experimental data used for this model are listed in Table 2.1. The constant total fuel energy is based on a medium load point and the chosen intake pressure and temperature and EGR ratio are in the middle of their validation range, to minimize the chance of extrapolation.

The total fuel energy is described as follows:

$$Q_{\text{Total}} = m_{\text{PFI}}\text{LHV}_{\text{PFI}} + m_{\text{DI}}\text{LHV}_{\text{DI}},$$

where $m_{\text{PFI}}$ and $m_{\text{DI}}$ are the injected mass of E85 and diesel and $\text{LHV}_{\text{PFI}}$ and $\text{LHV}_{\text{DI}}$ are the lower heating values of E85 and diesel, respectively. The inputs to the combustion model are the following in-cylinder conditions at intake valve closing and fuel settings, which are summarized in $\hat{s}_{\text{IVC}}$:

- Pressure at the intake manifold $p_{\text{im}}$;
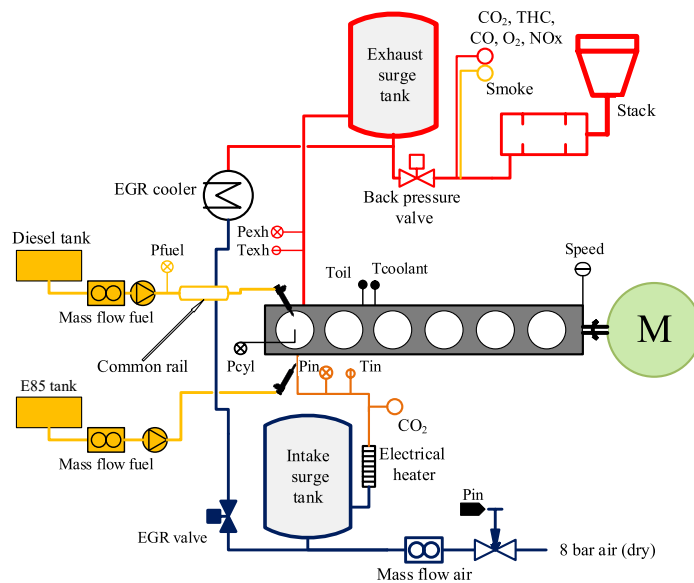


Figure 2.1: Schematic representation of the RCCI engine. [11]

Table 2.1: Engine and fuel specifications and nominal operating conditions followed by the operating range for which the model is validated.

| Parameters | Value |
|---|---|
| Displaced volume [cc] | 2097 |
| Bore x Stroke [mm] | 130 x 158 |
| Compression ratio [-] | 15.85 |
| Intake valve closure [CAD aTDC] | -153 |
| Exhaust valve opening [CAD aTDC] | 128 |
| Direct injected fuel | Diesel (EN590) |
| Lower heating value of Diesel [MJ/kg] | 42.6 |
| Stoichiometric air-to-fuel ratio of Diesel [-] | 14.5:1 |
| Port injected fuel | Ethanol (E85) |
| Lower heating value of Ethanol [MJ/kg] | 29.1 |
| Stoichiometric air-to-fuel ratio of Ethanol [-] | 9.7:1 |
| Engine speed [RPM] | 1205 |
| Total fuel energy $Q_{\text{total}}$ [kJ] | 3.94 |
| Intake pressure [kPa] | 228 |
| Intake temperature [°C] | 34 |
| EGR ratio [%] | 9.0 |
| **Inputs** | **Range** |
| BR [%] | 69.75 to 99.85 |
| $\text{SOI}_{\text{DI}}$ [CAD aTDC] | -70 to -35 |

- Temperature at the intake manifold $T_{\text{im}}$;

- Air-to-fuel ratio $\lambda$

$$\lambda = \frac{\dot{m}_{\text{air}}}{\dot{m}_{\text{PFI}}\text{AFR}_{\text{PFI,st}} + \dot{m}_{\text{DI}}\text{AFR}_{\text{DI,st}}}$$

  with $\text{AFR}_{\text{PFI,st}}$ and $\text{AFR}_{\text{DI,st}}$ the stoichiometric air-to-fuel ratio for the port injected and direct injected fuel, respectively.

- EGR ratio

$$X_{\text{EGR}} = \frac{\text{CO}_{2,\text{in}}}{\text{CO}_{2,\text{out}}}$$

  with $\text{CO}_{2,\text{in}}$ and $\text{CO}_{2,\text{out}}$ the $\text{CO}_2$ concentration at the intake and exhaust, respectively.;

- Energy-based blend ratio

$$\text{BR} = \frac{m_{\text{PFI}}\text{LHV}_{\text{PFI}}}{m_{\text{DI}}\text{LHV}_{\text{DI}} + m_{\text{PFI}}\text{LHV}_{\text{PFI}}};$$

- Start-of-injection of the direct injected fuel $\text{SOI}_{\text{DI}}$.

The model predicts the in-cylinder pressure trace during the combustion stroke, using principle components (PC) analysis. This can then be used to determine combustion parameters of interest, e.g. GIE, maximum pressure ($p_{\text{max}}$) and pressure rise rate (PRR). The following equation is used for the in-cylinder pressure using PCs:

$$p\left(\theta, \hat{\boldsymbol{s}}_{\text{IVC}}\right) \approx \tilde{p}\left(\theta, \hat{\boldsymbol{s}}_{\text{IVC}}\right) = \tilde{\boldsymbol{w}}^{\top}\left(\boldsymbol{s}_{\text{IVC}}\right)\tilde{\boldsymbol{f}}(\theta) + f_{\mu}(\theta) \tag{2.1}$$

with crank angle $\theta$, estimated in-cylinder conditions at intake valve closing (IVC) and fuel settings $\hat{\boldsymbol{s}}_{\text{IVC}}$ (as listed above), weights $\tilde{\boldsymbol{w}}\left(\boldsymbol{s}_{\text{IVC}}\right) := \begin{bmatrix} w_1\left(\boldsymbol{s}_{\text{IVC}}\right) & w_2\left(\boldsymbol{s}_{\text{IVC}}\right) \cdots & w_7\left(\boldsymbol{s}_{\text{IVC}}\right) \end{bmatrix}^{\text{T}}$, the reduced set of seven PCs $\tilde{\boldsymbol{f}}(\theta) := \begin{bmatrix} f_1(\theta) & f_2(\theta) & \cdots & f_7(\theta) \end{bmatrix}^{\text{T}}$ and the mean pressure $f_{\mu}(\theta)$. In practice,
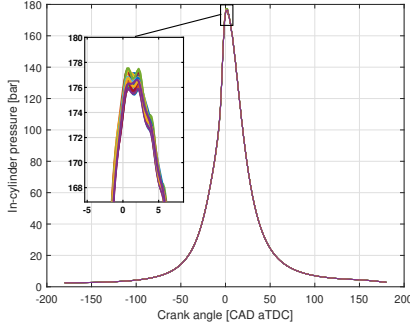
Figure 2.2: Variation between 25 in-cylinder pressure traces, caused by cycle-to-cycle variation.

Table 2.2: Average relevant combustion parameters with corresponding standard deviations (STD) and relative STD (rSTD).

| Parameter | Average | STD | rSTD [%] |
|---|---|---|---|
| GIE [%] | 45.86 | 0.085 | 0.2 |
| CA50 [CAD aTDC] | -3.1 | 0.04 | 1.2 |

the same $\hat{\boldsymbol{s}}_{\mathrm{IVC}}$ result in different $\tilde{\boldsymbol{w}}$, because RCCI is driven by physics that are very sensitive to initial charge composition and/or temperature. For this reason, $\tilde{\boldsymbol{w}}(\hat{\boldsymbol{s}}_{\mathrm{IVC}})$ is described as a stochastic process, such that

$$\tilde{\boldsymbol{w}}(\hat{\boldsymbol{s}}_{\mathrm{IVC}}) := \mathcal{N}(\hat{\boldsymbol{w}}(\hat{\boldsymbol{s}}_{\mathrm{IVC}}), W(\hat{\boldsymbol{s}}_{\mathrm{IVC}})) \tag{2.2}$$

with $\hat{\boldsymbol{w}}(\hat{\boldsymbol{s}}_{\mathrm{IVC}}) := \mathbb{E}[\boldsymbol{w}(\hat{\boldsymbol{s}}_{\mathrm{IVC}})]$ and $W(\hat{\boldsymbol{s}}_{\mathrm{IVC}}) := \mathbb{E}[(\tilde{\boldsymbol{w}}(\hat{\boldsymbol{s}}_{\mathrm{IVC}}) - \hat{\boldsymbol{w}}(\hat{\boldsymbol{s}}_{\mathrm{IVC}}))(\tilde{\boldsymbol{w}}(\hat{\boldsymbol{s}}_{\mathrm{IVC}}) - \hat{\boldsymbol{w}}(\hat{\boldsymbol{s}}_{\mathrm{IVC}}))^{\mathrm{T}}]$. This makes it possible to describe the in-cylinder pressure as a stochastic process with a mean and covariance. The latter is used to simulate cycle-to-cycle variation of the in-cylinder pressure. Figure 2.2 shows the variation between 25 in-cylinder pressure traces for exactly the same $\boldsymbol{s}_{\mathrm{IVC}}$, caused by the cycle-to-cycle variation. To give an idea of its impact on relevant combustion parameters, the relative standard deviations of GIE and CA50 are given in Table 2.2. The variation in GIE seems small, however, considering that the measured GIE roughly varies between 45% and 48%, the relative impact of the STD is higher.

It is worth mentioning that the model is unable to predict emissions, which is why emissions are omitted from the optimization problem. The operating range for which the model is validated, is also shown in Table 2.1.

## 2.3 Online optimization problem

Figure 2.3 shows the closed-loop control scheme for this study, including the RCCI engine with separated combustion block and air path in the dashed rectangle. It is assumed that the air path is controlled in order to keep the air path parameters ($X_{\mathrm{EGR}}$, $\lambda$ and $p_{\mathrm{IM}}$) constant. Therefore, the focus in the remainder of this study is only on fuel path control. The combustion block represents the combustion model, as discussed in the previous section. A distinction is made between fast and slow disturbances. The fast disturbance is the cycle-to-cycle variations in the in-cylinder pressure. The slow disturbances are wear of parts and changing ambient conditions ($s_{\mathrm{amb}}$), e.g. ambient pressure, temperature and humidity.

The control objective is to maximize GIE by changing BR and $\mathrm{SOI}_{\mathrm{DI}}$, while not violating any constraints and being able to deal with all aforementioned disturbances. The maximization is done for a constant total fuel energy $Q_{\mathrm{total}}$ and engine speed. The constant $Q_{\mathrm{total}}$ means that the IMEP changes with changing GIE and is not constant. In this case, the total fuel energy can be seen as the driver demand. A variable BR in combination with a constant $Q_{\mathrm{total}}$ is chosen instead of the two inputs direct injected and port injected fuel mass (allowing a varying $Q_{\mathrm{total}}$) to limit the optimization problem to two inputs in total (including $\mathrm{SOI}_{\mathrm{DI}}$). Three total inputs would significantly increase complexity of visualizing the cost function as a function the inputs. For two inputs, a 3D plot of the cost function as a function of the inputs is possible. Visualizing this can help with understanding
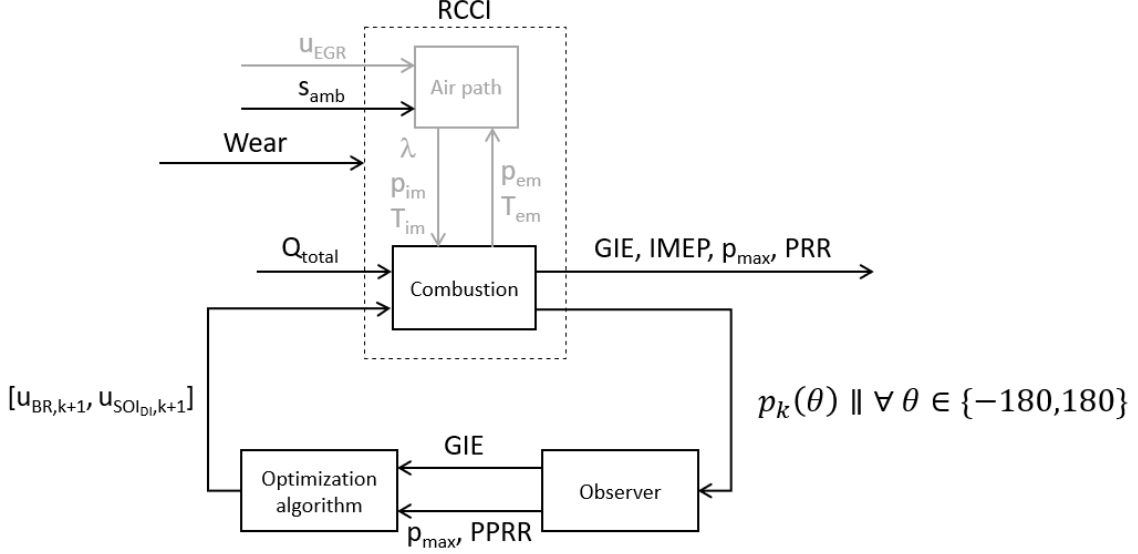
Figure 2.3: Block diagram of the control loop for RCCI with $k$ the current iteration with a duration of one combustion cycle, $p_k(\theta)$ the in-cylinder pressure as a function of crank angle (CA) $\theta$, $[u_{\mathrm{BR},k+1}, u_{\mathrm{SOI_{DI}},k+1}]$ the next-cycle inputs, $u_{\mathrm{EGR}}$ the EGR valve position, $\lambda$ the equivalence air-fuel ratio and $p_{\mathrm{im}}, T_{\mathrm{im}}, p_{\mathrm{em}}$ and $T_{\mathrm{im}}$ the pressure and temperature at the intake and exhaust manifold, respectively. The maximum pressure $p_{\mathrm{max}}$ and peak pressure rise rate (PPRR) are the safety limits. Present disturbances are wear of parts and ambient conditions $s_{\mathrm{amb}}$.

certain behaviors and the tuning process of the used optimization method, which is introduced in the next chapter.

Using the GIE and constraints, the optimization algorithm's aim is to solve the following optimization problem, formulated as a general maximization problem with hard constraints:

$$
\begin{aligned}
\max_{\mathrm{BR},\mathrm{SOI_{DI}}} \quad & \mathrm{GIE} \\
\text{s.t.} \quad & Q_{\mathrm{total}} = \text{constant} \\
& p_{\mathrm{max}} \le p_{\mathrm{max,UB}} \\
& \mathrm{PPRR} \le \mathrm{PPRR_{UB}} \\
& \mathrm{BR_{LB}} \le \mathrm{BR} \le \mathrm{BR_{UB}} \\
& \mathrm{SOI_{DI,LB}} \le \mathrm{SOI_{DI}} \le \mathrm{SOI_{DI,UB}}
\end{aligned}
\tag{2.3}
$$

where $p_{\mathrm{max}}$ is the maximum pressure, PPRR is the peak pressure rise rate and UB and LB stand for upper and lower bound, respectively. The output of the combustion block is the in-cylinder pressure trace, which can be translated to the GIE and constraints, via an observer. The GIE is calculated as follows:

$$
GIE = \frac{\int_{\theta=-180}^{\theta=180} p(\theta)\, dV(\theta)}{Q_{\mathrm{total}}},
\tag{2.4}
$$

where $\theta$ is the crank angle, $p(\theta)$ is the in-cylinder pressure trace, $V(\theta)$ is the combustion chamber volume and $Q_{\mathrm{Total}}$ is the total fuel energy. The constraints are the safety limits (peak pressure $p_{\mathrm{max}}$ and peak pressure rise rate (PPRR)) and the input ranges. The two inputs are the energy-based blend ratio BR and $\mathrm{SOI_{DI}}$. The maximum values for the constraints are listed in Table 2.3.

Table 2.3: Upper (UB) and lower (LB) bounds for constraints.

| Constraints | Values |
|---|---|
| $p_{\mathrm{max,UB}}$ [bar] | 200 |
| $\mathrm{PPRR_{UB}}$ [bar/CAD] | 20 |
| $\mathrm{BR_{UB}}$ [%] | 99.85 |
| $\mathrm{BR_{LB}}$ [%] | 69.75 |
| $\mathrm{SOI_{DI,UB}}$ [CAD aTDC] | -35 |
| $\mathrm{SOI_{DI,LB}}$ [CAD aTDC] | -70 |

The maximum GIE and corresponding optimal inputs for an ideal situation without any measurement noise ($\mathrm{GIE_{max,ideal}}$), are determined offline using the interior-point method. The calculated maximum GIE is 47.65% with a BR and $\mathrm{SOI_{DI}}$ of 69.75% and -55.88 CAD aTDC, respectively.

# 3   Constrained extremum-seeking control used for fuel-efficiency optimization

A possible solution to the adaptive control problem described in the previous chapter, could be constrained extremum-seeking control (ESC). In this chapter, the theory behind the algorithm is discussed and a guide for tuning the algorithm is provided.

## 3.1   Theory

The theory of the algorithm is based on Samuel [10]. This section starts with the introduction of the steady-state optimization problem, written in the form of a general, nonlinear discrete-time system for readability purposes. Thereafter, the barrier function is introduced, which is used to incorporate the constraints into the cost function. This is followed by the explanation of the gradient-descent control law. Finally, the Kalman filter that is used for gradient estimation is explained. Figure 3.1 shows the control scheme with all parts of the algorithm highlighted.

### 3.1.1   Problem description for general nonlinear, discrete-time system

The optimization is performed for a constant working point with a constant $Q_{\text{total}}$ and constant air path parameters, and variable BR and $\text{SOI}_{\text{DI}}$. Considering that the optimization is performed for a constant working point, the optimization problem becomes a steady-state optimization problem as the driver demand stays constant. The cost function and constraint function at steady-state are $\ell(p_k, u_k)$ and $\gamma(p_k, u_k)$, respectively, with $k$ being the current time step equal to the duration of a combustion cycle, $p_k$ being the in-cylinder pressure trace, $u_k$ being the inputs BR and $\text{SOI}_{\text{DI}}$ at time step $k$, $\ell(p_k, u_k) = -\text{GIE}$ (negative, because maximization of GIE is desired) and $\gamma(p_k, u_k)$ containing the constraints. During this study, the constraints are the safety limits of maximum pressure and PPRR and the input ranges, as shown in Equation (2.3).

Now, the optimization problem, given in Equation (2.3), can be reformulated as:

$$
\begin{aligned}
\min_{\text{BR},\text{SOI}_{\text{DI}}} \quad & \ell(p, u) \\
\text{s.t.} \quad & \gamma(p, u) \geq 0.
\end{aligned}
\tag{3.1}
$$
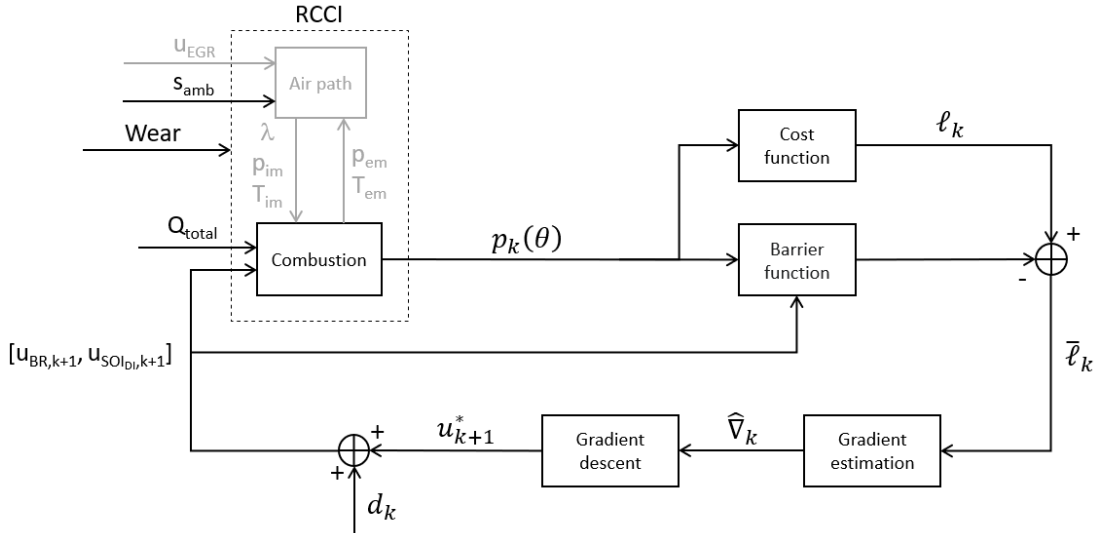


Figure 3.1: Constrained ESC scheme for steady-state process.

8

All constraints are normalized to equalize the impact of each constraints and be able to fit in the form $\gamma(p, u) \geq 0$. The normalization for the constraints is done as follows:

$$\gamma(p, u) = \begin{bmatrix} \frac{p_{\max,\text{UB}}}{p_{\max}} - 1 \\ \frac{\text{PPRR}_{\text{UB}}}{\text{PPRR}} - 1 \\ \frac{\text{BR}_{\text{UB}}}{\text{BR}} - 1 \\ \frac{\text{BR}}{\text{BR}_{\text{LB}}} - 1 \\ \frac{\text{SOI}_{\text{DI,UB}}}{\text{SOI}_{\text{DI}}} - 1 \\ \frac{\text{SOI}_{\text{DI}}}{\text{SOI}_{\text{DI,LB}}} - 1 \end{bmatrix}. \tag{3.2}$$

Similarly, the input $\text{SOI}_{\text{DI}}$ is divided by 100 to equalize the order of magnitude of both inputs. This is important, because only one scalar can be chosen for the learning gain $k_{\text{g}}$, which will be properly introduced later in this chapter. Knowing that the $\text{SOI}_{\text{DI}}$ ranges from 35 to 70 CAD bTDC, dividing it by 100 will bring them to the same order of magnitude as the input BR, which ranges between 0.7 and 1. Only when entered into the combustion model, it is brought back to the original order of magnitude.

The following assumptions for $\ell(p, u)$ and $\gamma(p, u)$ are made, to guarantee the existence of a unique extremum:

- the cost function $\ell(p, u)$ is a strictly convex function,

- the constraint functions $\gamma_i(p, u)$ are strictly concave.

Figure 3.2 shows that $\ell(p, u)$ is not strictly convex as it has multiple local minima. As this gradient-descent based algorithm only works for local optimization, reaching the global minimum is not evident. However, the additional assumption is made that the approximate location of the global minimum is known. Only a saddle point in the region close to the global minimum can still prevent strict convexity. However, the chances of a saddle point (gradient is zero where the local maximum of one input exactly matches location of local minimum of the other) are expected to be low, thus there is assumed that no saddle points are present in that region. After these additional assumptions, there can be stated that, locally, the cost function is strictly convex. A similar situation holds for the constraint function of the PPRR. This is solved with the same additional assumptions. Figure 3.2 also shows that the cost function $\ell(p, u)$ is a smooth function, which also holds for the constraint functions $\gamma_i(u)$.



Figure 3.2: Cost function $\ell(p, u)$ as a function of inputs BR and $\text{SOI}_{\text{DI}}$.

### 3.1.2 Incorporating soft constraints with a barrier function

The current optimization problem shown in Equation (3.1), is an optimization problem with hard constraints. For the chosen ESC algorithm to work, this needs to be transformed to an optimization problem where the constraints are incorp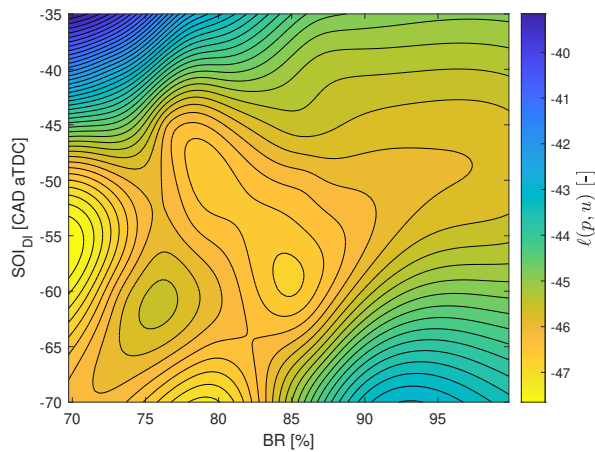orated into the cost function as soft constraints. This is done by augmenting the measured cost using a barrier function to yield the following augmented cost function:

$$\bar{\ell}(p_k, u_k) = \ell(p_k, u_k) - \mu \sum_{i=1}^{n_c} \psi(\gamma_i(p_k, u_k)), \tag{3.3}$$

where $\mu$ is a positive constant called the barrier parameter, $n_c$ is the number of constraints and $\psi$ is the barrier function. The barrier function $\psi$ should be a continuous function that acts as a barrier when constraints are violated, but this barrier quickly fades out further away from where the constraints are met. A natural logarithm is able to achieve this, until a certain point. As during this study, the gradient of the augmented cost with respect to $u$ must be estimated. This means that strict feasibility of the trajectories of $u$ may not be enforced by the control. This poses a problem with a logarithmic barrier function as even a small violation in the constraints ($\gamma(p_k, u_k)$ becomes zero or negative) can cause the barrier function to result in unreal values, as $\ln(0) = -\infty$ and the ln of a negative value results in a complex number. To avoid such situations, the so-called modified barrier method is used. The modified barrier method switches to a quadratic function when the optimization problem approaches infeasibility.

Let $b(a) = \ln(a)$ and let $q(a)$ be the quadratic function with $a$ representing the argument of the barrier function (in this case $\gamma_i(p_k, u_k) \ \forall \ i \in \{1, n_c\}$). Additionally, let $\epsilon$ be a small, positive number that represents the distance between the argument $a$ and the constraint. The quadratic function $q(a)$ is defined such that its gradient and Hessian matches that of $b(a)$ for $a = \frac{\epsilon}{2}$, as this is the switching point between $b(a)$ and $q(a)$. This ensures that $\psi$ is a continuous function. The quadratic function is as follows:

$$q(a) = l + m \left(a - \frac{\epsilon}{2}\right) + n \left(a - \frac{\epsilon}{2}\right)^2, \tag{3.4}$$

where $l = b\left(\frac{\epsilon}{2}\right) = \ln\left(\frac{\epsilon}{2}\right)$, $m = \frac{db}{da}|_{a=\frac{\epsilon}{2}} = \frac{2}{\epsilon}$ and $n = \frac{d^2 b}{da^2}|_{a=\frac{\epsilon}{2}} = \frac{-4}{\epsilon^2}$. The modified barrier function is then:

$$\psi(a) = \begin{cases} b(a) \text{ if } a > \frac{\epsilon}{2} \\ q(a) \text{ if } a \leq \frac{\epsilon}{2} \end{cases} . \tag{3.5}$$
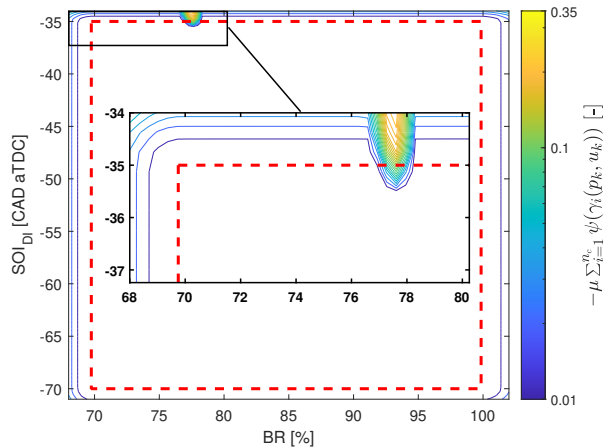


Figure 3.3: Visualization of working of barrier function as a function of inputs BR and $\text{SOI}_{\text{DI}}$. The red dotted lines represent the input ranges and the colorbar is on a logarithmic scale.

Now the barrier function $\psi$ is well defined, the constraints are included in the augmented cost function using Equation (3.3). In Figure 3.3, the term $-\mu \sum_{i=1}^{n_c} \psi(\gamma_i(p_k, u_k))$ (sum of the barrier function) is visualized as using a contour plot for $\mu = 10^{-5}$ and $\epsilon = 10^{-3}$ as a function of the inputs BR and SOI$_{\text{DI}}$. The contour lines are caused by the input and safety constraints. First of all, it shows the effect of the input constraints. At the edges of the input ranges, a clear upwards ramp is visible, which means the cost is increased. It also shows a clear peak for SOI$_{\text{DI}} = -34$ CAD aTDC and BR $= 77.62\%$, which is caused by a violation of the PPRR constraint.

The hard-constrained optimization problem defined in Equation (3.1) can be reformulated as a soft-constrained optimization problem. This reformulation is done as follows:

$$\min_{\text{BR,SOI}_{\text{DI}}} \bar{\ell}(p_k, u_k), \tag{3.6}$$

where $\bar{\ell}(p_k, u_k)$ is the augmented cost function. The shape of the full augmented cost function is unknown.

### 3.1.3 Gradient descent

Now, consider the incremental change in augmented cost:

$$\Delta\bar{\ell}_k = \bar{\ell}(p_{k+1}, u_{k+1}) - \bar{\ell}(p_k, u_k) = \bar{\ell}_{k+1} - \bar{\ell}_k. \tag{3.7}$$

Since it has been assumed that $\bar{\ell}(p, u)$ is continuously differentiable, $\Delta\bar{\ell}_k$ can be locally linearized as follows:

$$\Delta\bar{\ell}_k = \nabla_k^{\text{T}} \Delta u_k, \tag{3.8}$$

where $\Delta u_k = u_{k+1} - u_k$ and $\nabla_k$ is the gradient of $\bar{\ell}(p, u)$ as a function of the two inputs BR and SOI$_{\text{DI}}$.

The proposed ESC used to drive $\bar{\ell}(p_k, u_k)$ (Equation (3.6)) to its minimum, is based on gradient descent. The gradient descent control law is as follows:

$$u_{k+1}^* = u_k^* - k_{\text{g}} \nabla_k, \tag{3.9}$$

where $u^*$ is the input towards the minimum and $k_{\text{g}}$ is the learning gain. In a perfect situation without any noise, the gradient $\nabla_k$ can be determined using Equation (3.8). However, the cycle-to-cycle variation of the pressure signal causes noise to be present in the measurements. To find the gradient while dealing with noise, the gradient is estimated using the Kalman filter. For the estimation of $\nabla_k$, two points on the cost function are needed, which means $\Delta u_k$ cannot be equal to zero. To realize this, a dither function is added to the control law, given by Equation (3.9), as follows:

$$u_{k+1} = u_{k+1}^* + d_k, \tag{3.10}$$

where $d_k = [d_{k,\text{BR}}, d_{k,\text{SOI}_{\text{DI}}}]$ is the bounded dither function and $u$ is the input, with added dither function, actually going into the system. The dither function perturbs the inputs BR and SOI$_{\text{DI}}$, which is needed for the estimation of the gradient $\nabla_k$. It is defined as:

$$d_{k_i} = A_{\text{d}_i} \cos(\omega_{\text{d}_i} k), \tag{3.11}$$

with $A_{\text{d}_i}, \omega_{\text{d}_i} \in \mathbb{R}_+$, the dither amplitude and dither frequency, respectively with $i = 1, 2$ for the inputs BR and SOI$_{\text{DI}}$, respectively. Since only one output is available for the estimation of the two gradients in $\nabla_k$, a separation of the two dither signals is needed. This is done by choosing two different dither frequencies, which is further explained in Section 3.2.

### 3.1.4  Gradient estimation using the Kalman filter

Let the estimation of the linearized $\Delta\bar{\ell}_k$, given by Equation (3.8), be written as:

$$\Delta\hat{\bar{\ell}}_k = \hat{\nabla}_k^{\mathrm{T}}\Delta u_k, \tag{3.12}$$

where $\hat{\nabla}_k$ is the estimated gradient. Using Equation (3.8) and Equation (3.12), the output prediction error is defined as:

$$e_k = \Delta\bar{\ell}_k - \Delta\hat{\bar{\ell}}_k. \tag{3.13}$$

Equation (3.12) can be reformulated in a general state-space notation:

$$\begin{aligned} x_{k+1} &= Ax_k + w_k \\ y_k &= C_k x_k + v_k \end{aligned} \tag{3.14}$$

where $x_k$ is the estimated gradient $\hat{\nabla}_k$, $A$ is the identity matrix, $w_k$ is process noise with process covariance $Q$, $y_k$ is $\Delta\hat{\bar{\ell}}_k$, $C_k$ is $\Delta u_k^{\mathrm{T}}$ and $v_k$ is the cycle-to-cycle variation (with cycle-to-cycle covariance $R$). The cycle-to-cycle variation is only present in $y_k$, as the GIE and safety constraints are dependent on the in-cylinder pressure signal. The process noise $w_k$ acts as a correction for the assumed local linearity of $\bar{\ell}(p, u)$. Notice that $C_k$ is not constant, since it is dependent on $u_k$.

Now, the gradient estimation update approach can be defined based on Equation (3.14). The Kalman filter can be used for this, because of the local linearization of $\Delta\hat{\bar{\ell}}_k$ in Equation (3.12). The update law for for the error covariance $\Sigma^{-1}$ and estimated gradient $\hat{\nabla}$, split into prediction and update step, is defined as:

**Prediction**

$$\hat{\nabla}_{k|k-1} = \hat{\nabla}_{k-1|k-1} \tag{3.15}$$

$$\Sigma_{k|k-1}^{-1} = \Sigma_{k-1|k-1}^{-1} + Q \tag{3.16}$$

**Update**

$$K_k = \Sigma_{k|k-1}^{-1}\Delta u_k^{\mathrm{T}}\left(R + \Delta u_k^{\mathrm{T}}\Sigma_{k|k-1}^{-1}\Delta u_k\right)^{-1} \tag{3.17}$$

$$\hat{\nabla}_{k|k} = \hat{\nabla}_{k|k-1} + K_k(\Delta\bar{\ell}_k - \hat{\nabla}_{k|k-1}\Delta u_k^{\mathrm{T}}) \tag{3.18}$$

$$\Sigma_{k|k}^{-1} = (I - K_k\Delta u_k)\Sigma_{k|k-1}^{-1} \tag{3.19}$$

where $K_k$ is the Kalman gain, $Q \in \mathbb{R}^{2x2}$ and $R \in \mathbb{R}_+$. As shown, the prediction step is just the previous step and in case of the error covariance, $Q$ is added. The update step is of the same form as the general Kalman filter. The initial condition $\hat{\nabla}_0$ is defined as $\underline{0}$, as no knowledge about the gradient of the initial inputs is available. By choosing something other than zero, the initial gradient might be the wrong way. The initial condition $\Sigma_0^{-1}$ is a tuning parameter and will be discussed in the Section 3.2.

## 3.2  Tuning guideline

The final step in developing this optimization algorithm is tuning. In this section, a structural guideline is given on how to tune the relevant tuning parameters such that certain control design criteria are met. The guideline is given using just initial input, namely BR = 74% and $\mathrm{SOI_{DI}} = -50.2$ CAD aTDC. This input is chosen as it converges to the global minimum and is relatively far from the global minimum. It is assumed that if the design criteria for this initial input are met, all initial

inputs closer to the global minimum will also meet the criteria.

There are three control design criteria defined. First, the global minimum (i.e. the highest GIE) must be reached. The highest GIE (for hard constraints) is 47.65% with corresponding BR and SOI$_{\text{DI}}$ of 69.75% and -55.88 CAD aTDC, respectively.

The second design criteria is meeting the constraints. The safety constraints may not be violated and the input constraints may be violated by maximum 2% of the lower or upper bound. In this case, a small violation (chosen for 2%) of the input constraints has no significant consequences, as the input range represents the range for which the model is validated and does not represent any physical limits. An exception to this is that the BR cannot go over 100%. However, in practice one must take certain limits into consideration. For example, the SOI$_{\text{DI}}$ cannot take place before IVC or after TDC.

The third design criteria is that the convergence speed of GIE must be as fast as possible. At least fast enough for effective use on highways according to the Common Artemis Driving Cycles (CADC) [13]. The convergence speed is determined by the first iteration for which:

$$\text{GIE}_{\text{movmean}} > \text{GIE}_{\text{mean}} - 0.05 \, (\text{GIE}_{\text{mean}} - \text{GIE}_0), \tag{3.20}$$

where GIE$_{\text{movmean}}$ is the moving average GIE over 5 iterations, GIE$_{mean}$ is the average GIE of the last 50 iterations (after full convergence) and GIE$_0$ is the initial GIE. Based on the CADC, the duration at constant speed ($\pm$ 5 km/h) are roughly 5 to 20 seconds for an urban environment and over 50 seconds for a highway environment. For the urban environment, the worst case scenario is assumed and thus 5 seconds are used as target. Let the convergence speed of the algorithm be maximum 10% of that duration. This means that the maximum convergence speed for an urban environment is 0.5 seconds (i.e. 5 iterations) and for the highway environment 5 seconds (i.e. 50 iterations) (in a worst case scenario).

In summary, the control design criteria are:

- Reaching global minimum, i.e. the highest GIE;

- Meeting constraints: safety constraints may not be violated and input constraints may be violated by maximum 2%;

- Convergence speed as fast as possible. At least under 5 seconds (50 iterations) for effective use on highways based on the Common Artemis Driving Cycles (CADC) [13].

The parameters that need to be tuned are the barrier parameters $\mu$ and $\epsilon$, dither amplitude $A_{\text{d}}$, dither frequency $\omega_{\text{d}}$, cycle-to-cycle covariance $R$, process covariance $Q$, initial error covariance $\Sigma_0^{-1}$ and the learning gain $k_{\text{g}}$.

### 3.2.1  Barrier parameters

The parameters $\mu$ and $\epsilon$ need to be chosen, such that the effect of the barrier function is visible in the augmented cost function $\bar{\ell}(p, u)$ and the global minimum has moved towards BR$_{\text{LB}}$, but without considerably compromising the original shape of $\ell(p, u)$ within the input constraints. In this case, the initial global minimum is located at BR = 68.32% ($<$ BR$_{\text{LB}}$) and SOI$_{\text{DI}}$ = -56.42 CAD aTDC. Figure 3.4a shows the augmented cost function with correctly tuned $\mu$ and $\epsilon$, as the effect of the barrier function is visible and the original shape remains recognizable. Additionally, the global minimum is moved towards BR$_{\text{LB}}$. Figures 3.4b and 3.4c show the augmented cost function for too high and low values of $\mu$, respectively. It shows that when $\mu$ is too low, the effects of the barrier function are not visible and the minimum still lies outside the input range and when $\mu$ is too high the original shape is compromised. A similar, but reversed, effect holds for $\epsilon$. The barrier parameters are chosen to be $\mu = 10^{-5}$ and $\epsilon = 10^{-3}$ and kept constant during tuning of the remaining parameters.
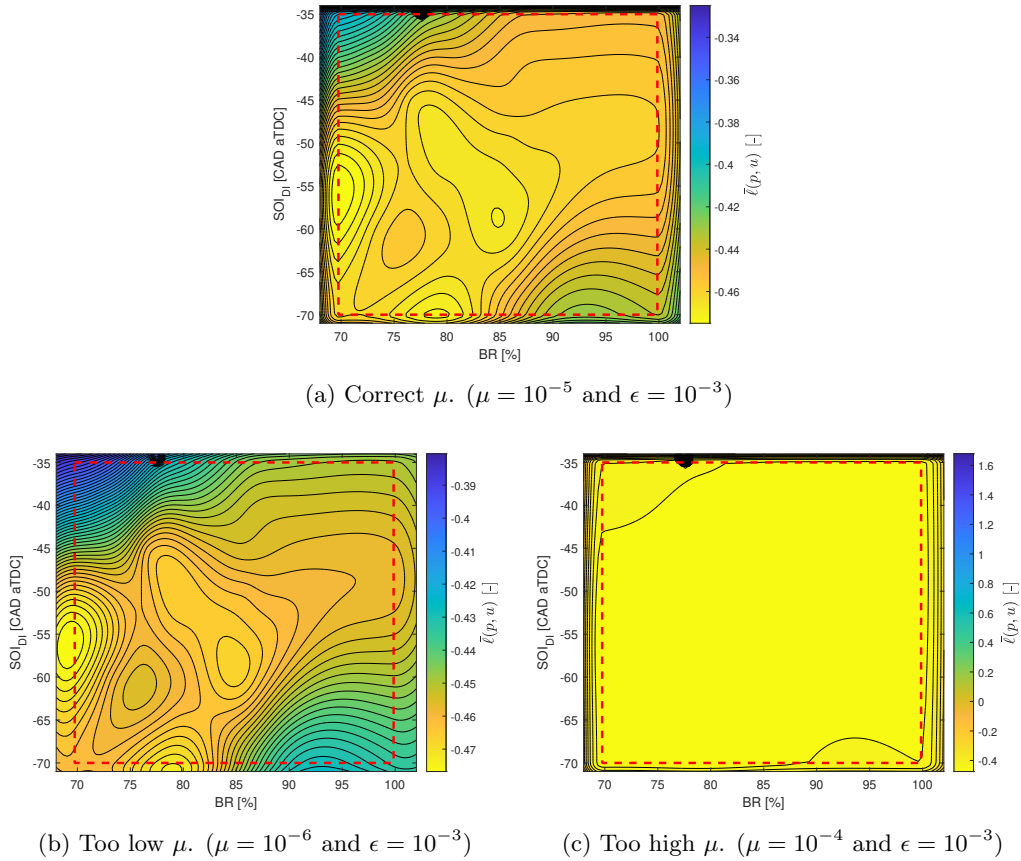
13

(a) Correct $\mu$. ($\mu = 10^{-5}$ and $\epsilon = 10^{-3}$)



(b) Too low $\mu$. ($\mu = 10^{-6}$ and $\epsilon = 10^{-3}$)

(c) Too high $\mu$. ($\mu = 10^{-4}$ and $\epsilon = 10^{-3}$)

Figure 3.4: Visualization of the augmented cost function $\bar{\ell}(p, u)$ as a function of inputs BR and $\mathrm{SOI_{DI}}$ for various values of $\mu$. The red dotted lines represent the input ranges.

Table 3.1: Offline determined ideal maximum GIE with the corresponding inputs for without barrier function (Hard-constrained) and with barrier function (Soft-constrained).

|  | **Hard-costrained** | **Soft-constrained** |
|---|---|---|
| $\mathrm{GIE_{max,ideal}}$ [%] | 47.65 | 47.77 |
| BR [%] | 69.75 | 68.50 |
| $\mathrm{SOI_{DI}}$ [CA aTDC] | -55.88 | -56.47 |

Now that the constraints are incorporated as soft constraints in $\bar{\ell}(p, u)$, the maximum GIE and corresponding optimal inputs for an ideal situation without noise will be slightly different to those for hard constraints ($\mathrm{GIE_{max,ideal}}$), as mentioned in Section 2.3. The maximum, GIE for the soft-constrained optimization problem without noise ($\mathrm{GIE_{max,ideal,SC}}$) and its corresponding inputs are compared to those for the hard-constrained optimization problem in Table 3.1. There can be noticed that for the soft-constrained optimization problem, the optimal BR slightly exceeds its lower bound, which is expected as the barrier at the input constraint is not perfectly vertical. However, it remains within the allowed 2% violation of the input constraints. The slightly exceeded BR also explains why the maximum GIE is slightly higher for the optimization problem with soft constraints.

### 3.2.2 Dither amplitude

In order to tune the dither amplitude $A_{\mathrm{d}}$, learning is turned off ($k_{\mathrm{g}} = 0$), as the dither amplitude is used for the gradient estimation. Certain limits of the dither amplitude can be determined, before
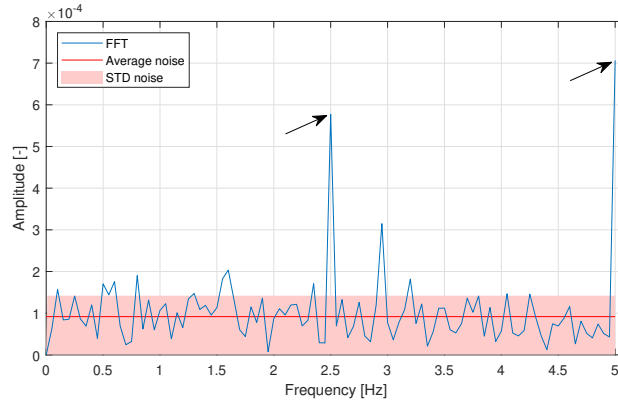
Figure 3.5: FFT of a point in the cost function with the average noise level and one standard deviation above and everything below the average noise for a dither amplitude of 0.005 for both inputs. The dither frequencies are indicated by the arrows.

Table 3.2: Average convergence speed (over 100 iterations) and average GIE with its peak-to-peak value for two different dither amplitudes. (Average convergence speed is determined after tuning the remaining parameters.)

|  | $\mathbf{A_d = 0.005}$ | $\mathbf{A_d = 0.01}$ |
| --- | --- | --- |
| Average convergence speed [iteration] | 27.7 | 29.2 |
| $\text{GIE}_{\text{average}}$ [%] | 47.60 | 47.50 |
| $\text{GIE}_{\text{peak-to-peak}}$ [%] | 0.30 | 0.50 |

starting the tuning process. First of all, one has to take into account the limits of the input actuators (in this case, the injectors). The actuators' resolution is not infinitely small and thus determines a minimum value for the possible dither amplitude. Contrastingly, a dither amplitude that is too high, can negate the assumed local linearization of the augmented cost function. Taking these limitations into account, one can start tuning the dither amplitude. It is important that the dither amplitude is higher than the amplitude of the noise (over the full cost function), to ensure that the Kalman filter uses the dither signal for the parameter estimation, not the noise. A fast Fourier transform (FFT) can be used to differentiate the dither signal from the noise and to determine if the dither signal is above the noise floor. Figure 3.5 shows the FFT of a point in the cost function for which the dither amplitude is higher than the mean amplitude of the noise plus one standard deviation of the noise. There can be seen that the dither amplitude at 2.5 and 5 Hz, is higher than the noise and one standard deviation. As a rule of thumb, if this holds for every point in the cost function, the dither amplitude is high enough to overcome the noise. The lowest dither amplitude for which this holds, is 0.005 for both inputs. Additionally, in theory, a larger dither amplitude should result in faster convergence (as long as local linearity is maintained). However, it also results in a larger oscillation away from the optimum ($\text{GIE}_{\text{max,ideal,SC}} = 47.77\%$), once the optimum is reached. This means that when the optimum (i.e. maximum GIE) is reached, a higher dither amplitude results in a lower average GIE. This trade-off between convergence speed and oscillation away from the optimum mostly comes down to preference.

After tuning the remaining parameters, using $A_d = 0.005$ for both inputs, and turning learning on, the differences between $A_d = 0.005$ and $A_d = 0.01$ are investigated for the design criteria of convergence speed and reaching highest GIE. As listed in Table 3.2, the difference in average convergence speed between $A_d = 0.005$ and $A_d = 0.01$ for both inputs is not even 2 iterations (0.2 seconds), which is not significant. However, the oscillation away from the optimum after convergence (calculated without cycle-to-cycle variation) for $A_d = 0.005$ is smaller. For $A_d = 0.005$, the average GIE is 0.17% lower than $\text{GIE}_{\text{max,ideal,SC}}$, compared to 0.27% for $A_d = 0.01$. However, adding half

of $GIE_{peak-to-peak}$ to $GIE_{average}$ shows that $GIE_{max,ideal,SC}$ is reached for both dither amplitudes. In this case, there is chosen for a dither amplitude of 0.005 for both inputs, because of the smaller oscillation away from the optimum.

### 3.2.3　Dither frequency

Like the amplitude, the dither frequency $\omega_d$ can be tuned with learning turned off. The goal of the dither frequency is to perturb the inputs BR and $SOI_{DI}$, in order to estimate the gradient of $\bar{\ell}(p, u)$. The dither frequency is limited by the combustion frequency. In this case the combustion frequency is 10 Hz, which means that the dither frequency cannot be chosen higher than 5 Hz (Nyquist frequency of this engine speed). In order to suppress the effect of the cycle-to-cycle variation in the cost function $\Delta\bar{y}$ on the estimated gradient $\hat{\nabla}$ as well as possible, $\Delta u$ should be as high as possible for each iteration. This can be achieved for both inputs by choosing 5 and 2.5 Hz as the dither frequencies. The dither amplitude is not used for this, as it is desired to keep $A_d$ as low as possible to minimize oscillation away from the optimum. To show this, Figure 3.6 shows two dither signals with 2.5 and 5 Hz, compared to a signal with a frequency of 3 Hz. There can be seen that a frequency of 3 Hz, has a varying $\Delta d_k$, which means $\Delta d_k$ is not as high as possible for each iteration. In theory, the better the cycle-to-cycle variation suppression, the more accurate the gradient estimation and the higher the convergence speed. Table 3.3 shows that for 1.5 and 3 Hz, the convergence speed is approximately 0.5 seconds slower. The remaining question is: which frequency should be linked to which input? In the case of dynamic systems, this can significantly influence the convergence speed. However, as this is a static system, it is insignificant. It was decided that the BR is dithered at 5 Hz and $SOI_{DI}$ at 2.5 Hz.

### 3.2.4　Cycle-to-cycle covariance

The cycle-to-cycle covariance $R$ can be determined based on the noise in $\bar{\ell}(p, u)$ caused by the cycle-to-cycle variation in the in-cylinder pressure signal. Therefore, learning and dithering is turned off. The cycle-to-cycle covariance is calculated as follows:

$$R = \sigma_{noise}^2, \tag{3.21}$$

where $\sigma_{noise}$ is standard deviation of the noise in $\bar{\ell}(p, u)$. From this, it is determined that approximately $R = 5 \cdot 10^{-7}$, which is kept constant throughout the tuning process as the variance of the cycle-to-cycle variation remains the same.



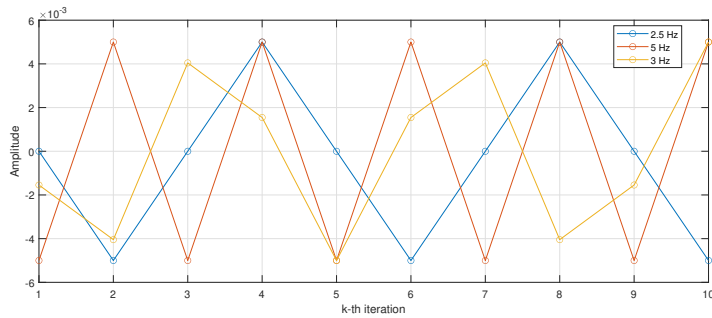Figure 3.6: Three dither signals with different dither frequencies.

Table 3.3: Average convergence speed (over 100 iterations) for dither frequency of 2.5 and 5 Hz versus 1.5 and 3 Hz. (Average convergence speed is determined after tuning the remaining parameters.)

| | $\omega_d = [\mathbf{2.5, 5}]$ [Hz] | $\omega_d = [\mathbf{1.5, 3}]$ [Hz] |
|---|---|---|
| Average convergence speed [iteration] | 27.7 | 32.2 |

### 3.2.5　Learning gain, process covariance and initial error covariance

Finally, learning can be turned on and the learning gain $k_{\mathrm{g}}$ can be slowly increased. Consequently, with increasing $k_{\mathrm{g}}$, the prediction in Equation (3.15) becomes more inaccurate, as increasing $k_{\mathrm{g}}$ means increasing $\Delta u_k$ and thus increasing step in $\hat{\nabla}_k$ (because of the assumed local linearization in Equation (3.8)). As the prediction of $\hat{\nabla}_k$ becomes more inaccurate, a higher value for $\Sigma_0^{-1}$ and $Q$ is desired, to increase the contribution of the measurement data in the calculation of $\hat{\nabla}_k$. The subtle, but important, difference between increasing $\Sigma_0^{-1}$ and increasing $Q$ is that the effect of $\Sigma_0^{-1}$ fades out after a certain amount of iterations, but the effect of $Q$ remains constant over all iterations. This is the case, because $\Sigma_0^{-1}$ is the error covariance of the initial estimated gradient $\hat{\nabla}_0 = \underline{0}$, which it is known to be wrong. However, $Q$ is the process covariance and is constant over the entire process. For this reason, $\Sigma_0^{-1}$ is mainly used to directly increase convergence speed and $Q$ is used to regulate the suppression of the cycle-to-cycle variation in $\hat{\nabla}_k$. The latter is possible as only the ratio between $R$ and $Q$ is what influences the algorithm. $R$ is a fixed value based on the cycle-to-cycle variation, but $Q$ can still be used to regulate the suppression of the cycle-to-cycle variation by changing the ratio between $R$ and $Q$.
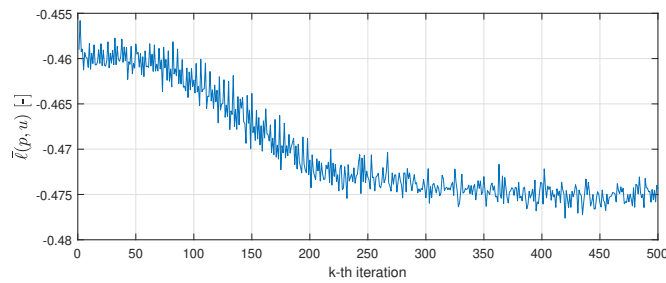


(a) Low $k_{\mathrm{g}}$. $(k_{\mathrm{g}} = 0.001)$
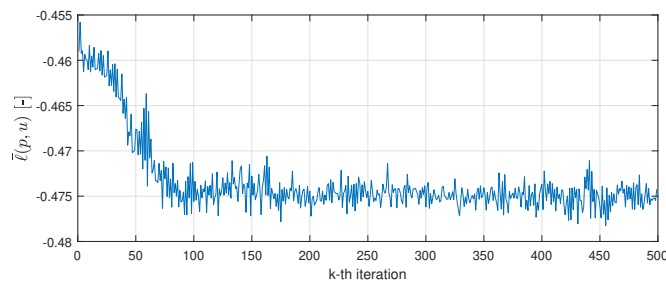
(b) Correct $k_{\mathrm{g}}$. $(k_{\mathrm{g}} = 0.01)$
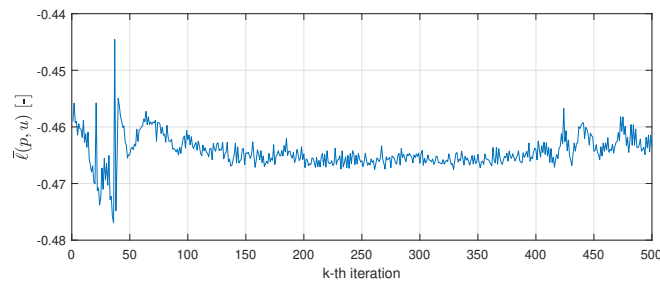
(c) Too high $k_{\mathrm{g}}$. $(k_{\mathrm{g}} = 0.1)$

Figure 3.7: Plot of the augmented cost function $\bar{\ell}(p, u)$ for various values of $k_{\mathrm{g}}$, with $\Sigma_0^{-1} = 10^{-5}I$ and $Q = 10^{-5}I$.

First, $k_g$ is increased (starting from 0.001) for $\Sigma_0^{-1} = 10^{-5}I$ and $Q = 10^{-5}I$, which, in this case, are values that result in slow convergence. $k_g$ is increased until the point that $\bar{\ell}(p,u)$ (and the inputs) do not converge anymore. Figure 3.7 shows $\bar{\ell}(p,u)$ for three different values of $k_g$ for the initial inputs BR = 74.00% and $SOI_{DI} = -50.20$ CAD aTDC. Figures 3.7b and 3.7c show convergence, with fastest convergence for $k_g = 0.01$ of roughly 100 iterations versus 250 iterations for $k_g = 0.001$. However, both convergence speeds are over the design criteria of lower than 50 iterations. Figure 3.7a does not show convergence as $\bar{\ell}(p,u)$ does not gradually decrease to its minimum at around -0.475, which means that $k_g = 0.1$ is too high. After further fine-tuning, $k_g = 0.01$ is deemed the best value for the learning gain.

Next, $\Sigma_0^{-1}$ is increased, to increase convergence speed. Figure 3.8 shows the difference in convergence for a low and correct value of $\Sigma_0^{-1}$. It shows that a higher $\Sigma_0^{-1}$, results in faster convergence for approximately the first 50 iterations, after which the effect of $\Sigma_0^{-1}$ fades out. At a certain point, further increasing $\Sigma_0^{-1}$ does not affect the convergence anymore. In this case, that point is $\Sigma_0^{-1} = 0.1I$, which is chosen as the best value for $\Sigma_0^{-1}$. For $\Sigma_0^{-1} = 0.1I$, the design criteria for convergence speed is met, as the convergence speed is roughly 30 iterations, which is lower than the determined maximum of 50 iterations.

Finally, the process covariance $Q \in \mathbb{R}^{2x2}$ can be used to find the balance between suppressing the cycle-to-cycle variation (lower $Q$) and limit the amount of violation of the input constraints (higher $Q$). This is explained by the fact that $Q$ influences the amount of correction needed for the assumption of local linearity. At locations where the barrier function is dominant over GIE, the cost function is even less linear. Figure 3.9 shows that higher values of $Q$ smooth out the input signals and result in a BR closer to its lower bound and a smaller violation of $BR_{LB}$. The maximum violation of $BR_{LB}$ for Figure 3.9a is 1.7% with respect to $BR_{LB}$ compared to 1.2% for Figure 3.9c. However, it also results in less suppression of the cycle-to-cycle variation, which can be demonstrated by the difference in $SOI_{DI}$ in Figures 3.9a and 3.9c. Since the diagonals of $Q$ can be tuned separately, a fitting value for each input can be chosen. Based on the different input signals in Figure 3.9, the process covariance is chosen as $Q = \begin{bmatrix} 10^{-3} & 0 \\ 0 & 10^{-4} \end{bmatrix}$.



(a) Low $\Sigma_0^{-1}$. ($\Sigma_0^{-1} = 10^{-5}I$)                (b) Correct $\Sigma_0^{-1}$. ($\Sigma_0^{-1} = 0.1I$)

Figure 3.8: Plot of the augmented cost function $\bar{\ell}(p,u)$ and the inputs BR and $SOI_{DI}$ for two different values of $\Sigma_0^{-1}$, with $k_g = 0.01$ and $Q = 10^{-5}I$.

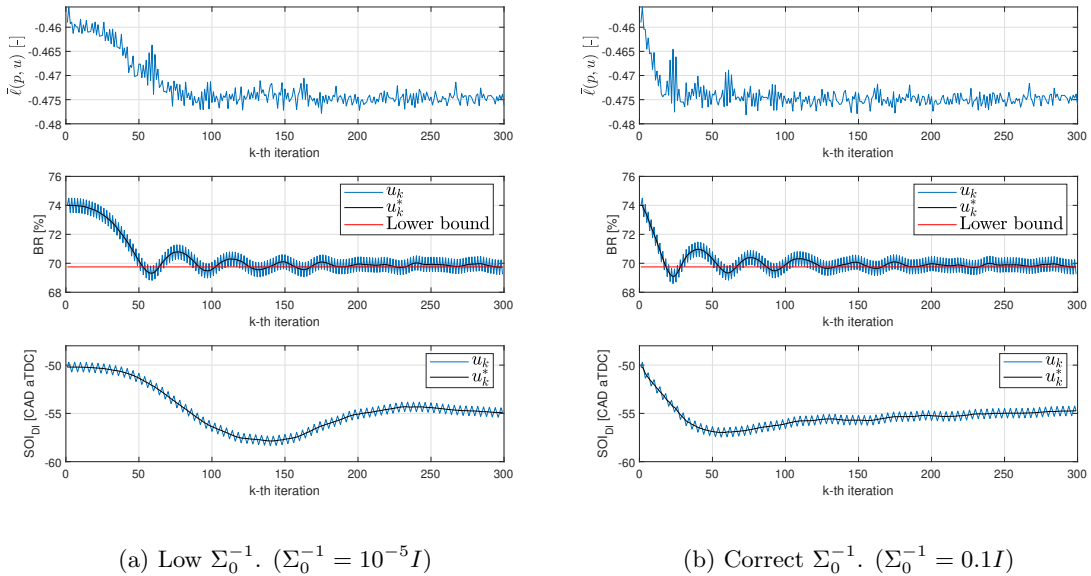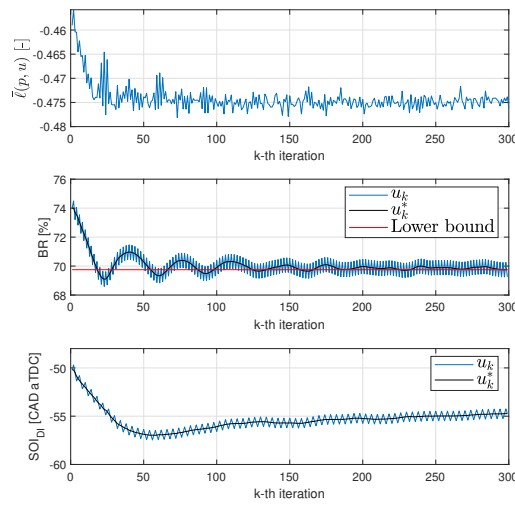(a) $Q = 10^{-5}I$
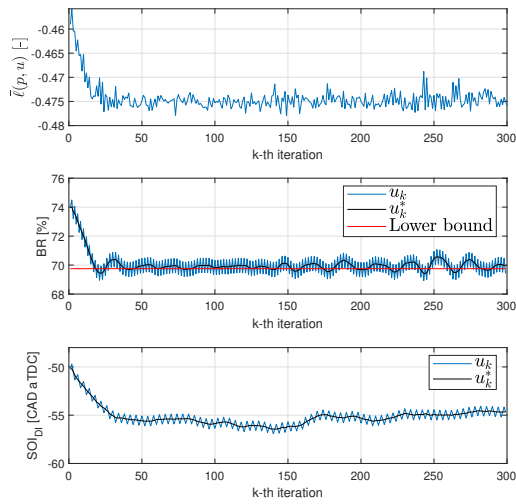


(b) $Q = 10^{-4}I$
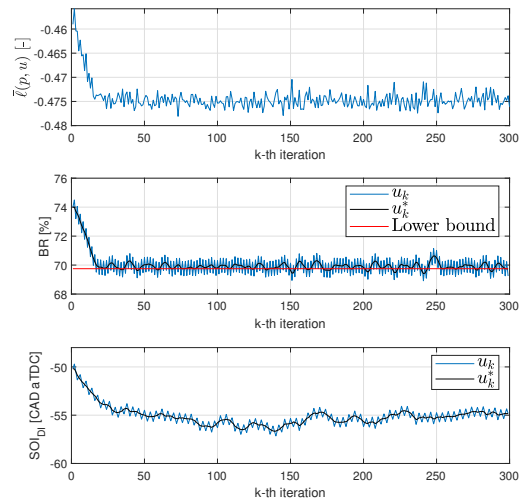
(c) $Q = 10^{-3}I$

Figure 3.9: Plot of the augmented cost function $\bar{\ell}(p, u)$ and the inputs BR and $\mathrm{SOI_{DI}}$ for various values of $Q$, with $k_{\mathrm{g}} = 0.01$ and $\Sigma_0^{-1} = 0.1I$.

# 4   Simulation results

In this chapter, the results of the tuned algorithm are presented. The final values for all tuning parameters are summarized in Table 4.1. The algorithm is tested on different forms of robustness (with respect to disturbance of initial fuel path conditions, cycle-to-cycle variation and air path disturbance) in combination with the control design criteria, listed in Section 3.2, based on the following case study: Based on base calibrations in the lab, the inputs corresponding to the approximate global minimum are known (for a constant engine speed and $Q_{\mathrm{total}}$ as mentioned in Table 2.1). However, in practice, this global minimum shifts because of changing ambient conditions ($s_{\mathrm{amb}}$) or wear of parts. The objective of the algorithm is to converge to the known global minimum as fast as possible, while meeting the constraints, as mentioned in the control design criteria.

## 4.1   Convergence speed and disturbance of initial fuel path conditions

The average convergence speed and robustness with respect to initial fuel path disturbances are investigated by means of a Monte Carlo simulation. Meaning that the same global minimum has to be found for 500 different initial inputs, which represent fuelling variations caused by wear of injector parts. Because of the latter, the actual injections vary from the intended injection settings. These initial inputs are generated around the known global minimum of the hard-constrained optimization problem, using an exponential and normal distribution for the BR and $SOI_{\mathrm{DI}}$, respectively. The distribution for BR is exponential, because the optimal BR is located on its lower bound. The standard deviations for the distribution of BR and $SOI_{\mathrm{DI}}$ are 1.5% (translates to 1 mg of DI fuel) and 2.0 CAD, respectively. These standard deviations are assumed to be realistic values for variations in fuel injectors.

In this case, robustness with respect to the preset fueling variations is dictated by the amount of initial points that converge to the global minimum compared to those that converge to a local minimum. Performance is dictated by the average and maximum convergence speed of GIE of all 500 initial inputs. For the 494 different initial inputs that converged to the global minimum, displayed in Figure 4.1a, the average convergence speed is 10.2 iterations (i.e. combustion cycles), which corresponds to approximately 1 second, while the design criteria for the constraints are met. The distribution of the convergence speeds are shown in Figure 4.1b. The highest convergence speed is 47 iterations (4.7 seconds). This means that, according to our rough estimations based on the CADC, the algorithm is fast enough for effective use on a highway. However, it is roughly 0.5 seconds too slow for driving in an urban environment based on the average convergence speed. The blue dots in Figure 4.1a show 103 out of the 494 initial inputs for which the convergence speed is lower than 0.5 seconds and thus fast enough for effective use in an urban environment. As can be expected, the blue dots are located closer to the global minimum compared to the black dots with a standard

Table 4.1: Final values for tuning parameters for this study.

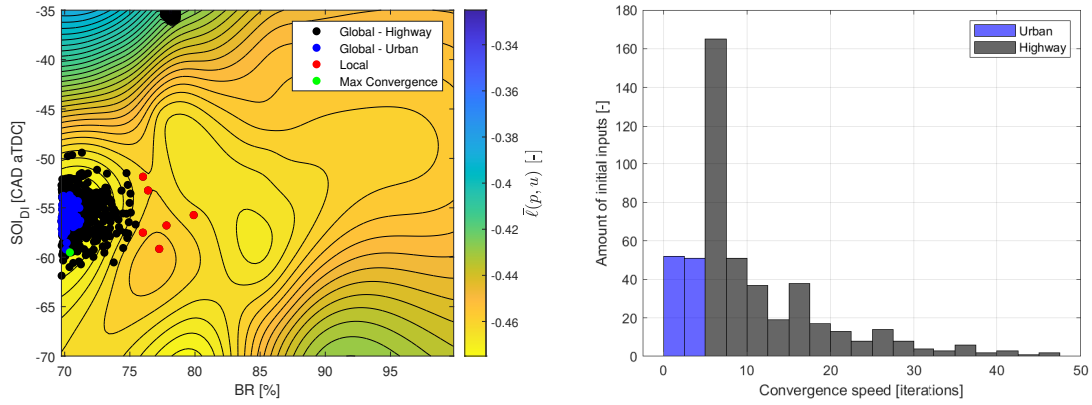| Parameters | Values |
|---|---|
| $\mu$ [-] | $10^{-5}$ |
| $\epsilon$ [-] | 0.001 |
| $\omega_{\mathrm{d,BR}}$ [Hz] | 5.0 |
| $\omega_{\mathrm{d,SOI_{DI}}}$ [Hz] | 2.5 |
| $A_{\mathrm{d,BR}}$ [-] | 0.005 |
| $A_{\mathrm{d,SOI_{DI}}}$ [CAD/100] | 0.005 |
| $R$ [-] | $5 \cdot 10^{-7}$ |
| $k_{\mathrm{g}}$ [-] | 0.01 |
| $\Sigma_0^{-1}$ [-] | $0.1I$ |
| $Q$ [-] | $\begin{bmatrix} 10^{-3} & 0 \\ 0 & 10^{-4} \end{bmatrix}$ |

(a) Visualization of the locations of the initial inputs plotted on the augmented cost function. The red dots represent initial inputs that converged to a local minimum and the green dot is the initial input with the highest convergence speed.

(b) Distribution of the initial inputs that converge to a global minimum.

Figure 4.1: Visualisation of the initial inputs and the convergence speeds.

deviation for BR and $SOI_{DI}$ of 1.0% and 1.0 CAD aTDC, respectively.

In terms of robustness, 6 out of 500 initial inputs converge to a local minimum, which translates to 1.3% of the tested initial inputs. This percentage is relatively low and acceptable, especially since all six points are located furthest away from the known global minimum ($> 5\%$ difference in BR). These variations in BR are exceptionally high and might be considered unrealistic. Figure 4.2 shows the difference in taken path between an initial input that converges to the global minimum and one of the six initial inputs that converges to a local minimum.
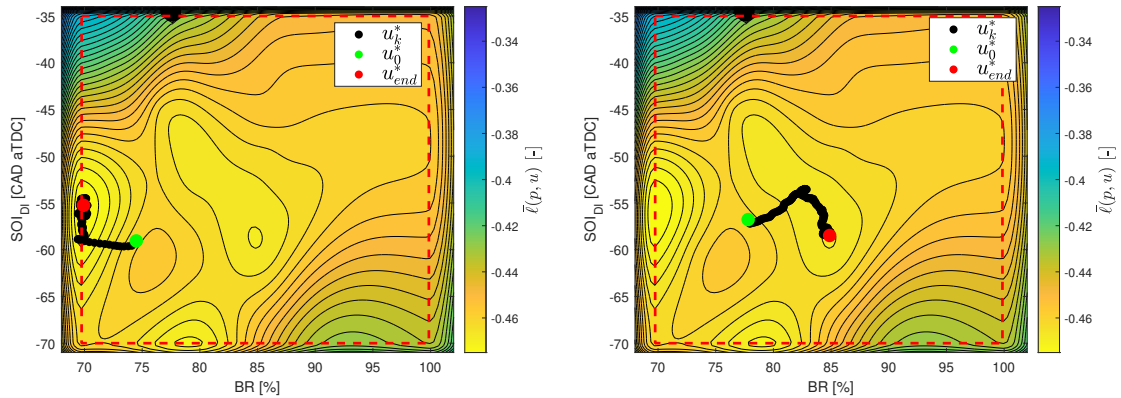


(a) Initial input that converges to the global minimum.

(b) One of the six initial input that converges to a local minimum.

Figure 4.2: Visualisation of taken paths of two different initial inputs plotted on the augmented cost function.
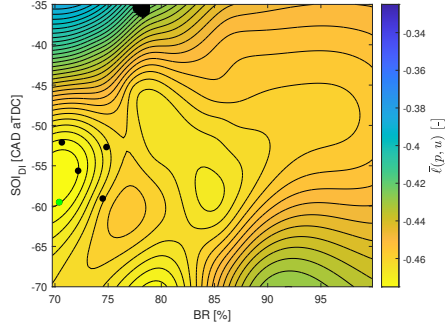
Figure 4.3: Location of the initial inputs listed in Table 4.2. The green point corresponds to the green point in Figure 4.1a.

Table 4.2: Initial inputs with the corresponding average convergence speed (CS) and its relative standard deviation (rSTD$_{CS}$).

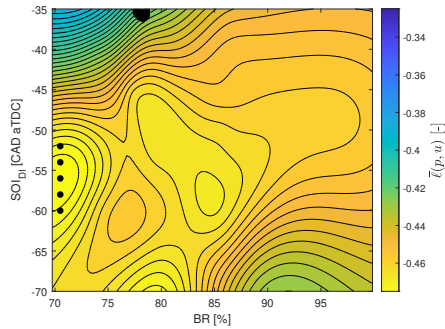| BR [%] | SOI$_{DI}$ [CAD aTDC] | Average CS [iterations] | rSTD$_{CS}$ [%] |
|---|---|---|---|
| 70.40 | -59.51 | 18.0 | 41 |
| 70.65 | -52.10 | 16.5 | 43 |
| 72.18 | -55.62 | 7.6 | 26 |
| 74.49 | -59.06 | 16.5 | 19 |
| 74.84 | -52.68 | 26.5 | 28 |



Figure 4.4: Location of the initial inputs listed in Table 4.3.

Table 4.3: Initial inputs with the corresponding average convergence speed (CS) and its relative standard deviation (rSTD$_{CS}$).

| BR [%] | SOI$_{DI}$ [CAD aTDC] | Average CS [iterations] | rSTD$_{CS}$ [%] |
|---|---|---|---|
| 70.50 | -60.0 | 20.0 | 38 |
| 70.50 | -58.0 | 9.9 | 69 |
| 70.50 | -56.0 | 4.1 | 74 |
| 70.50 | -54.0 | 9.9 | 105 |
| 70.50 | -52.0 | 16.6 | 42 |

## 4.2    Robustness with respect to cycle-to-cycle variation

The highest convergence speed of 47 iterations stands out compared to the average convergence speed, as the initial input is relatively close to the global minimum. Therefore, this initial input is optimized 25 times, to investigate the influence of the cycle-to-cycle variation on the convergence speed. This is repeated for 4 other initial inputs in different areas, as shown in Figure 4.3. The corresponding average convergence speed and its relative standard deviation are listed in Table 4.2. It shows a high relative standard deviation for initial inputs close to the optimal BR, including the green point. This means that the convergence speed is mainly determined by SOI$_{DI}$, which indicates that the influence of the cycle-to-cycle variation compared to the cost function, is higher for SOI$_{DI}$. To test this hypothesis, the same investigation is done for five different initial inputs with a constant BR close to the optimal BR, as shown in Figure 4.4. Table 4.3 shows high relative standard deviations for convergence speed, which confirms the hypothesis of higher influence of the cycle-to-cycle variation for SOI$_{DI}$.

## 4.3    Robustness with respect to air path disturbance

Finally, a different kind of robustness is investigated. An experiment is performed to test if the algorithm is robust towards changing air path settings, which changes the shape of the cost function, the location of the global minimum and the GIE$_{max,ideal,SC}$. Therefore, starting from the inputs corresponding to the hard-constrained global minimum and the original intake pressure $p_{im}$ of 2.28 bar, the intake pressure is changed after 100 iterations to 1.88 and changed back to 2.28 bar after 240 iterations. These changes are done to see if the algorithm corrects for the change in location of the global minimum. It is worth noticing that by changing the the air path setting during operation

of the algorithm, the assumption of a smooth cost function becomes invalid.

Figure 4.5 shows that, after both steps, the GIE converges to the new $GIE_{max,ideal,SC}$ (determined offline). Convergence after both steps takes roughly 20 iterations, i.e. 2 seconds. There can be seen that for some iterations, GIE is higher than $GIE_{max,ideal,SC}$. This is caused by the cycle-to-cycle variation on the in-cylinder pressure trace. As for the BR, it shows that the it fluctuates significantly more for $p_{im} = 1.88$ bar, than for $p_{im} = 2.28$ bar. A possible explanation for this is that the influence of the cycle-to-cycle variation on the augmented cost function for $p_{im} = 1.88$ bar is higher than for $p_{im} = 2.28$ bar. This is confirmed by the fact that the variation in GIE is roughly the same for both intake pressures. Additionally, Figure 4.5 shows a steep change in BR just after both steps, which is caused by the sudden change of augmented cost function. This causes an unusually high $\Delta \bar{\ell}_k$, for a relatively small $\Delta u_k$. Consequently, the gradient is overestimated. Thereafter, $\Delta \bar{\ell}_k$ becomes normal again and the algorithm corrects the overestimation. This is especially visible for the BR at the second step. First the gradient is overestimated in the negative way, caused by the step and pushing the BR below its lower bound. Then, the gradient is overestimated caused by the barrier function, pushing the BR too high. Thereafter, the BR converges to the global minimum. It is important to notice that all constraints are met, with the exception of one iteration just after the second step, where the input constraint is violated by 3.4% with respect to $BR_{LB}$, which is more than the maximum of 2% as stated in the design criteria. As this algorithm is not designed to deal with these steps in intake pressure, an input constraint violation for just one iteration is tolerable. Finally, one can see that the $SOI_{DI}$ is slightly lower on average for $p_{im} = 1.88$, but mostly fluctuates as the $SOI_{DI}$ has a small influence on the augmented cost function.

Even though this algorithm is not intended to be used for transient operation, the convergence speeds of 2 seconds for each step in the intake pressure is fast enough for effective use on highways. This shows there is potential for this method in transient operation.
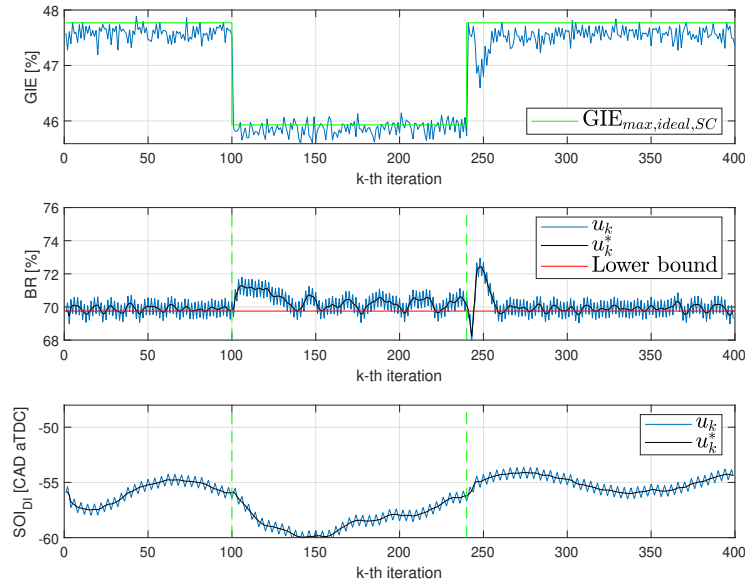


Figure 4.5: Plot of the GIE and the inputs BR and $SOI_{DI}$ with a step in intake pressure at 100 and 240 iterations.

# 5  Conclusion and future research

During this study, an online, model-free optimization algorithm is presented that directly optimizes and controls the GIE of an RCCI engine by using the fuel path and while meeting safety constraints. It does this for steady-state operation of an RCCI engine, with a constant total fuel energy and engine speed. The algorithm used to achieve this is online, constrained extremum-seeking control using the BR and $SOI_{DI}$ as inputs. It uses a modified barrier function to enforce the constraints. Also included in this study, is a tuning guideline to help gaining insight in how to tune the algorithm. The extremum-seeking controller is tested using a data-based combustion model, which simulates the RCCI combustion process and is able to include cycle-to-cycle variation of the in-cylinder pressure trace. The optimization algorithm is able to deal with these cycle-to-cycle variations.

The algorithm is tested on different forms of robustness (with respect to disturbance of initial fuel path conditions, cycle-to-cycle variation and air path disturbance) in combination with convergence speed. First, the average convergence speed is determined for different initial fuel path conditions by means of a Monte Carlo simulation. Meaning that the same global minimum has to be found for 500 different initial inputs (representing disturbance of the initial fuel path conditions), which are normally distributed around the known global minimum. The average convergence speed is 10.2 iterations (i.e. combustion cycles), which corresponds to approximately 1 second. The highest convergence speed is 4.7 seconds. According to rough estimations based on the Common Artemis Driving Cycles [13], this means that the algorithm is fast enough for effective use on a highway (<5 seconds), but, on average, too slow for driving in an urban environment (<0.5 seconds). The algorithm can be considered robust with respect to disturbance of initial fuel path conditions, as only 1.3% converge to a local minimum. Next, there is shown that the algorithm can deal with the cycle-to-cycle variation. It was shown that for $SOI_{DI}$ the influence of the cycle-to-cycle variation is relatively high compared to BR. However, that influence does not significantly affect the average convergence speed of GIE. Finally, another part of robustness was investigated by testing if the algorithm performed for a different air path setting, as this changes the location of the global minimum. To test this, starting from the hard-constrained global minimum, the intake pressure is decreased and, after convergence, increased back to its original value, which results in a sudden decrease and increase in GIE. Even though this algorithm is not intended to be used for such transient operations, there was shown that, for both steps in intake pressure, the convergence speed was roughly 2 seconds, which shows the potential of this method in transient operation.

The future research of this study can be split into short-term and long-term goals. A first limitation of the current algorithm is that it only works for steady-state operation, in this case a constant total fuel energy and constant engine speed. A first short-term goal would be to expand the current algorithm such that it works for transient operation and can account for changing driver demands and air path conditions at IVC. This could possibly be realised by implementing time-scale separation between the dynamical system (fast time-scale) and the steady-state optimization routine (slow time-scale), as presented by Guay [14]. Thereafter, to further increase transient performance, an integral action can be added to the control law, as presented by Guay and Burns [15]. However, this would be more of a long-term project, as the gradient estimation for this approach is more extensive than for the current algorithm. Secondly, as the current (constant) driver demand is the total fuel energy, there is no control over the desired IMEP. A long-term goal would be to change the driver demand to the desired IMEP. This could be realised by adding a PID controller, which uses $m_{PFI}$ to control the IMEP in parallel with the current ESC. As $m_{PFI}$ is known, the BR can be replaced by $m_{DI}$ as input for the ESC. A third limitation of the current approach, is that emissions are excluded from the constraints, as the current combustion model is not able to predict emissions. Another long-term goal would be to add emissions (e.g. $NO_x$, soot) to the optimization problem as constraints. This could be realised by adding emission prediction to the RCCI model or by using real-time emission data, when testing on an engine test setup. The latter leads to the final recommendation, which is to test the algorithm on an actual engine test setup. This can be done to see if the current algorithm works and if the provided tuning guidelines are sufficient for application on an actual engine.

# References

[1]  R. Reitz and G. Duraisamy, "Review of high efficiency and clean reactivity controlled compression ignition (rcci) combustion in internal combustion engines," *Progress in Energy and Combustion Science*, vol. 46, pp. 12–71, 2015. DOI: 10.1016/j.pecs.2014.05.003.

[2]  F. Willems, "Is cylinder pressure-based control required to meet future hd legislation," *IFAC PapersOnLine*, vol. 51, no. 31, pp. 111–118, 2018. DOI: 10.1016/j.ifacol.2018.10.021.

[3]  A. Indrajuana, C. Bekdemir, X. Luo, and F. Willems, "Robust multivariable feedback control of natural gas-diesel rcci combustion," *IFAC-PapersOnLine*, vol. 49, pp. 217–222, 11 2016. DOI: 10.1016/j.ifacol.2016.08.033.

[4]  F. Willems, F. Kupper, S. Ramesh, A. Indrajuana, and E. Doosje, "Coordinated air-fuel path control in a diesel-e85 rcci engine," *SAE Technical Paper*, 2019. DOI: 10.4271/2019-01-1175.

[5]  L. Xai, R. Willems, B. de Jager, and F. Willems, "Constrained optimization of fuel efficiency for rcci engines," *IFAC-PapersOnLine*, vol. 52, pp. 648–653, 5 2019. DOI: 10.1016/j.ifacol.2019.09.103.

[6]  L. Xai, B. de Jager, T. Donkers, and F. Willems, "Robust constrained optimization for rcci engines using nested penalized particle swarm," *Control Engineering Practice*, vol. 99, Jun. 2020. DOI: 10.1016/j.conengprac.2020.104411.

[7]  Y. Tan, W. Moase, C. Manzie, D. Nešić, and I. Mareels, "Extremum seeking from 1922 to 2010," in *Proceedings of the 29th Chinese Control Conference*, 2010, pp. 14–26.

[8]  K. Ariyur and M. Krstic, *Real-Time Optimization by Extremum-Seeking Control*. John Wiley & Sons, Inc., 2003, ISBN: 9780471468592. DOI: 10.1002/0471669784.

[9]  R. van der Weijst, T. van Keulen, and F. Willems, "Constrained multivariable extremum-seeking for online fuel-efficiency optimization of diesel engines," *Control Engineering Practice*, vol. 87, pp. 133–144, 2019. DOI: 10.1016/j.conengprac.2019.03.008.

[10]  S. Peter, "Constrained extremum-seeking control in discrete-time with application to liquefier unit power minimization," M.S. thesis, Queen's University, 2016.

[11]  R. Willems, F. Willems, N. Deen, and B. Somers, "Heat release rate shaping for optimal gross indicated efficiency in a heavy-duty rcci engine fueled with e85 and diesel," *Fuel*, vol. 288, Mar. 2021. DOI: 10.1016/j.fuel.2020.119656..

[12]  M. Vlaswinkel, B. de Jager, and F. Willems, "Data-based in-cylinder pressure model including cyclic variations of an rcci engine," 2022.

[13]  M. Andre, "The artemis european driving cycles for measuring car pollutant emissions," *Science of the Total Environment*, vol. 334-335, pp. 73–84, 2005. DOI: 10.1016/j.scitotenv.2004.04.070.

[14]  M. Guay, "A time-varying extremum-seeking control approach for discrete-time systems," *Process Control*, vol. 24, pp. 98–112, 3 2014. DOI: 10.1016/j.jprocont.2013.11.014.

[15]  M. Guay and D. Burns, "A proportional integral extremum-seeking control approach for discrete-time nonlinear systems," *International Journal of Control*, vol. 90, 8 2016. DOI: 10.1080/00207179.2016.1213425.