

This document contains a post-print version of the paper

Optimization-based estimator for the contour and movement of heavy plates in hot rolling

authored by **F. Schausberger, A. Steinboeck, and A. Kugi**

and published in *Journal of Process Control*.

The content of this post-print version is identical to the published paper but without the publisher's final layout or copy editing. Please, scroll down for the article.

Cite this article as:

F. Schausberger, A. Steinboeck, and A. Kugi, "Optimization-based estimator for the contour and movement of heavy plates in hot rolling", *Journal of Process Control*, vol. 29, pp. 23–32, 2015, ISSN: 0959-1524. DOI: [10.1016/j.jprocont.2015.03.006](https://doi.org/10.1016/j.jprocont.2015.03.006)

BibTex entry:

```
@article{Schausberger15a,  
  author = "F. Schausberger and A. Steinboeck and A. Kugi",  
  title = "Optimization-based estimator for the contour and movement of heavy plates in hot rolling ",  
  journal = "Journal of Process Control",  
  volume = "29",  
  pages = "23 - 32",  
  year = "2015",  
  issn = "0959-1524",  
  doi = "10.1016/j.jprocont.2015.03.006",  
  url = "http://www.sciencedirect.com/science/article/pii/S0959152415000529"  
}
```

Link to original paper:

<http://dx.doi.org/10.1016/j.jprocont.2015.03.006>
<http://www.sciencedirect.com/science/article/pii/S0959152415000529>

Read more ACIN papers or get this document:

<http://www.acin.tuwien.ac.at/literature>

Contact:

Automation and Control Institute (ACIN)
Vienna University of Technology
Gusshausstrasse 27-29/E376
1040 Vienna, Austria

Internet: www.acin.tuwien.ac.at
E-mail: office@acin.tuwien.ac.at
Phone: +43 1 58801 37601
Fax: +43 1 58801 37699

Copyright notice:

This is the authors' version of a work that was accepted for publication in *Journal of Process Control*. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in F. Schausberger, A. Steinboeck, and A. Kugi, "Optimization-based estimator for the contour and movement of heavy plates in hot rolling", *Journal of Process Control*, vol. 29, pp. 23–32, 2015, ISSN: 0959-1524. DOI: [10.1016/j.jprocont.2015.03.006](https://doi.org/10.1016/j.jprocont.2015.03.006)

Optimization-based estimator for the contour and movement of heavy plates in hot rolling

F. Schausberger*, A. Steinboeck, A. Kugi

Automation and Control Institute, Vienna University of Technology, Gußhausstraße 27–29, 1040 Vienna, Austria

Abstract

This paper deals with the estimation of the contour of heavy plates during the hot rolling process. Asymmetric rolling conditions lead to a non-rectangular contour. The reasons for this effect, e.g., temperature gradients or non-homogeneous input thickness profiles, are hard to predict in a real rolling mill. Hence, feedforward compensation of these disturbances is difficult, whereas feedback control could be a suitable means for improving the plate contour. Feedback control essentially requires the actual contour of the plate, which has to be measured or estimated in real-time. The method presented in this paper suggests to capture an image sequence of the plate by means of a thermographic measurement device. In the considered application, an image sequence is required because the whole plate contour cannot be captured by a single image. An optimization-based algorithm takes into account the image data and the restrictions of the plate movement in the rolling gap and uses this information to estimate the actual plate contour. In addition, the algorithm estimates the angular velocity and the speed of the plate. Measurement data from a heavy plate mill is used to validate the effectiveness of the proposed method.

Keywords: Plate contour measurement, Heavy plate mill, Unconstrained optimization, Moving horizon state estimation, Moving horizon parameter estimation, Model-based estimator, Image stitching, Snaking

1. Introduction

1.1. Measurement of the plate contour

In heavy plate mills, the thickness of the incoming steel slabs is successively reduced to the desired final plate thickness in several rolling passes at the mill stands. A major quality criterion of the final product is the shape of the resulting contour, i.e. the shape of the plate from the top view. Ideally, this contour is rectangular to maximize the usable area of the plate. A deviation from the desired contour with respect to the vertical axis of the plate is called camber (cf. Fig. 1). This may be caused by asymmetric rolling conditions, e.g., temperature gradients of the plate or non-homogeneous input thickness profiles. Minimizing the deviation from the desired plate contour is one main objective of the rolling process.

An important aspect in the design of measures to prevent the generation of shape defects is the detection of the actual plate contour. A precise knowledge of the contour (longitudinal boundaries and shape of the head and tail end) can be used to optimize the adjustment of the rolling mill to reduce the camber. Simply measuring the

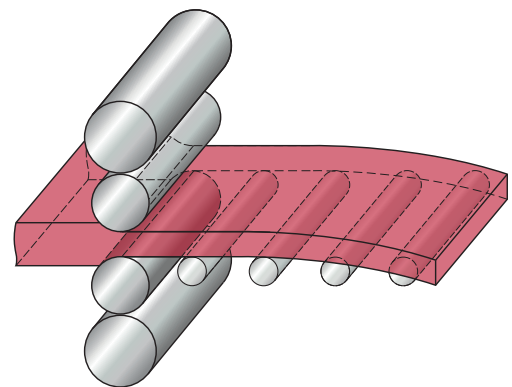


Figure 1: Sketched rolling mill with roller tables and rolled plate with camber.

whole plate contour at once, e.g., by means of edge detection of an image of the plate, is not possible in the considered application due to the following reasons:

- The contour of the plate is partly covered by plant components.
- The measurement should be performed during rolling in real-time.
- The length to width ratio of long plates is very different from the aspect ratio of common cameras.

*Corresponding author. Tel.: +43 1 58801 376231, fax: +43 1 58801 9376231.

Email addresses: schausberger@acin.tuwien.ac.at (F. Schausberger), andreas.steinboeck@tuwien.ac.at (A. Steinboeck), kugi@acin.tuwien.ac.at (A. Kugi)

Furthermore, the measurement of the contour should be carried out as near as possible to the rolling gap because:

- Short plates should also be captured.
- The time delay between the generation of a camber and its measurement should be kept to a minimum.

Therefore, the detection of camber must be conducted during the rolling pass itself and close to the rolling mill. At this position, the harsh environment may deteriorate the accuracy and robustness of measurements. Also the lateral and rotational movement of the plate, which is ignored in many published contour estimation procedures, makes contour detection difficult. In case of pure longitudinal movement, the contour could be obtained by simple integration of the plate velocity leading to the plate position and using measurement signals gathered at a spatially fixed position. Under real rolling conditions, this approach is infeasible because the plate is clamped in the rolling gap so that it may also rotate in addition to its main longitudinal motion (cf. Fig. 2). The possi-

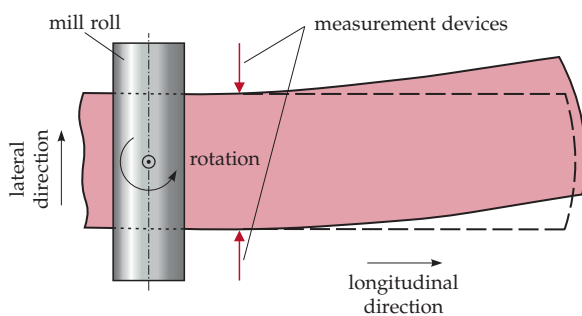


Figure 2: Measurement setup providing insufficient data for the estimation of the longitudinal boundaries of the plate. The solid line represents a rotating plate with camber. A rectangular plate with pure longitudinal movement is shown as dashed lines. Both contours may lead to the same signals of the measurement devices, which are located at a spatially fixed position.

ble rotation of the plate during the rolling process also complicates the detection of the contour.

1.2. Existing solutions

The use of three devices to measure the lateral position of the plate downstream the mill is discussed in [1]. Based on these measurements, a polynomial representation of the actual plate profile is estimated and used for feedback control to reduce the occurring camber by modifying the output thickness wedge.

Soaring computer performance enabled the usage of image processing techniques as proposed in [2], where three 2D-CCD cameras capture neighboring areas of the plate. In this camera configuration, the acute angle between the plate surface and the optical axis of the camera requires a precise calibration of the camera to accurately reconstruct the real image. After this preprocessing step,

a customized edge detection routine is employed to estimate the edge of the plate. The detected edges of neighboring images are joined based on the longitudinal speed of the plate and to ensure C_1 -continuity of the estimated plate edges. A very similar approach using just one camera to estimate the centerline of the plate is discussed in [3].

Also in strip rolling, 2D-cameras are used to track the lateral position of the strip during the rolling process. Carruthers-Watt et al. (cf. [4]) used measurements from several cameras between the mill stands to determine the lateral position of the strip in the finishing train of a hot strip mill. The edge is identified as maximum of the gradient of the intensity of the image in the lateral direction and parameterized using Bezier curves. A similar measurement setup and a mathematical model of the lateral position of the strip for steering control is discussed in [5]. In fact, an H_2 controller that is robust against heterogeneous properties of the different rolled products was designed using the tilts of several mill stands as control inputs.

An algorithm tailored to the stitching of several images of the plate was developed in [6], [7], and [8]. Common feature points are identified on two consecutive images to determine the displacement between the images. They are captured by a CCD camera. Ollikkala et al. (cf. [9]) used a very similar approach. However, in this solution the inclined viewing angle of the CCD-camera requires a perspective correction of the recorded images. After this image rectification step, an edge detection algorithm is used to extract the boundaries of the plate.

1.3. Motivation and objectives of this work

The existing solutions for the detection of the plate contour are mainly based on adequate image processing. In most published works in this field, neither lateral nor rotational movements of the plate are considered. The knowledge of the restrictions of the movement of the plate during the rolling process is also not taken into account, which may lead to reduced accuracy of the contour estimation.

Additionally, the angular velocity is linked with the lateral movement (snaking) of the plate in the rolling gap. This movement may lead to an eccentric position of the plate in the lateral direction. Because of the resulting asymmetric loading of the rolls, the knowledge of the evolution of the lateral position of the plate is also vital for the necessary adjustment of the rolling gap actuators.

Furthermore, the estimation of the longitudinal speed of the plate, which is required for the detection of the contour, is not covered in many works. Usually, the speed of the plate is calculated using a mathematical model (forward slip model) and measurements of the angular velocity of the rolls of the mill, see, e.g., [10]. This in general results in an error prone speed of the plate due to inaccuracies of the slip model and therefore in an additional error of the estimated contour. Hence, a

method to determine the plate velocity more accurately seems favorable in terms of the contour detection.

All these facts were the motivation to develop a new method to estimate the contour of heavy plates in hot rolling. The current work aims at:

- Accurate and robust estimation of the contour (longitudinal and lateral edges).
- Investigation of the influence and estimation of the movement of the plate (rotational and lateral motion).
- Precise estimation of the longitudinal speed of the plate.

The estimation of the plate contour has to respect several constrains:

- Harsh environment near the rolling gap.
- Real-time implementation of the contour detection algorithm.
- Small time delay between the generation and estimation of the plate contour.

The presented aspects and requirements make the contour detection a challenging task in terms of acquisition and processing of measurement data. This paper gives, in contrast to common image processing methods, an observer based approach utilizing the knowledge of the model for the estimation of the plate contour. The proposed method is tailored to the use in control algorithms.

1.4. Structure of the paper

The paper is organized as follows: Section 2 presents the mathematical model used for the description of the movement of the plate boundary. An optimization-based algorithm for the estimation of the contour of the plate is given in Section 3. The next Section, Section 4, contains some methods to improve the basic contour detection algorithm of Section 3. The recording and processing of image data are discussed in more detail in Section 5. The feasibility of the proposed approach is demonstrated in Section 6 by measurement results of a downstream contour measurement device in a heavy plate mill. Section 7 contains a short summary and gives an outlook on further research activities.

2. Mathematical model of the movement of the plate boundary

The exit velocity of the plate leaving the rolling gap may be non-uniform along the lateral direction. This is because the plate can experience rotations with respect to its vertical axis in addition to the main longitudinal motion, as observed in [11]. The measured boundary position is thus a superposition of the plate contour

and the plate movement. To analyze these effects, a mathematical model of the movement of the plate and the resulting measurement signal of the contour of the plate is deduced in this section. In the sequel, only one longitudinal edge of the plate is considered, however, the presented algorithm can be analogously applied to the second longitudinal edge.

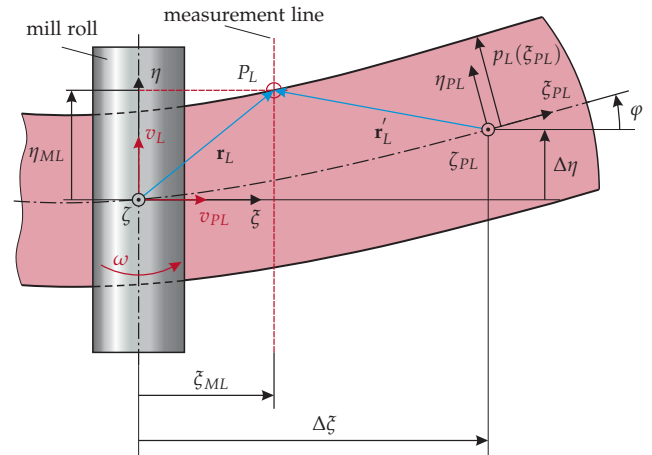


Figure 3: Top view of the rolling process with geometric parameterization of the plate contour.

A schematic representation of the mill roll and the rolled plate is shown in Fig. 3. The origin of a fixed global coordinate frame (ζ, η, ξ) with base vectors $\mathbf{e}_\zeta, \mathbf{e}_\eta$ and \mathbf{e}_ξ is located at the lateral center of the mill roll. At $\zeta = 0$, the plate moves out of the rolling gap with the velocity v_{PL} , which is assumed to be constant but unknown.

Moreover, the velocity in the direction η at $\zeta = 0$ is denoted by v_L . Although v_L is zero because the material is clamped in the rolling gap, it will be taken into account in the mathematical model. The reason to introduce this velocity is that misalignments of the camera as shown in Fig. 4 may be present. The plate is still clamped in the rolling gap and the velocity v of the plate is therefore perpendicular to the axis of the mill roll. Because of the misalignment of the camera and the rolls, this velocity induces a longitudinal and lateral velocity component v_{PL} and v_L in the (ζ, η, ξ) coordinate frame. The non-zero lateral velocity v_L results in a reduced accuracy of the estimated contour if not considered in the mathematical model. Due to the assembling situation of the camera, a small but constant misalignment has to be expected. Moreover, the rolls of the mill may exhibit a small time-dependent rotation about the ζ -axis, which also leads to an angular misalignment according to Fig. 4.

Furthermore, the plate contour is assumed to be constant after leaving the rolling gap. Let ω be the angular velocity of the plate about the axis ζ . A second local coordinate frame (index PL) that is fixed to the plate is used for parameterizing the longitudinal boundary by a

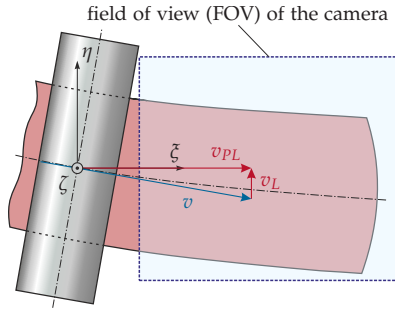


Figure 4: Angular misalignment of the camera and the rolls.

polynomial with degree N_L

$$p_L(\xi_{PL}) = \sum_{i=0}^{N_L} c_{L,i} \xi_{PL}^i, \quad (1)$$

with so far unknown coefficients $c_{L,i}$. The origin of the coordinate frame $(\xi_{PL}, \eta_{PL}, \zeta_{PL})$ is shifted by $(\Delta\xi, \Delta\eta, 0)$ and rotated by the angle φ about the axis ζ with respect to the coordinate frame (ξ, η, ζ) . Hence, the nonlinear dynamical model of the plate movement reads as

$$\frac{d}{dt} \begin{bmatrix} \Delta\xi \\ \Delta\eta \\ \varphi \end{bmatrix} = \begin{bmatrix} v_{PL} - \omega\Delta\eta \\ \omega\Delta\xi + v_L \\ \omega \end{bmatrix}, \quad (2)$$

with state vector $\mathbf{x} = [\Delta\xi \ \Delta\eta \ \varphi]^T$ and inputs v_{PL} , $v_L(t)$ and $\omega(t)$.

The current position of a point $P_L(\xi_L, \eta_L)$ on the longitudinal boundary in the local coordinate frame is written in vector representation as

$$\mathbf{r}'_L = [\xi_L(t) \ p_L(\xi_L(t))]^T, \quad (3)$$

starting from the origin of the plate-fixed coordinate frame. The same point can be described in the global coordinate frame with the vector

$$\mathbf{r}_L = [\xi_{ML} \ \eta_{ML}(t)]^T,$$

where $(\xi_{ML}, \eta_{ML}(t))$ refer to measurement results at the measurement line, see Fig. 3. These measurements are obtained from edge detection algorithms within bitmaps of the plate, which are captured by an infrared 2D-CCD camera mounted behind the rolling mill. A detailed description of the measurement setup is given in Section 5. It should be noted that ξ_{ML} does not vary with time in contrast to $\eta_{ML}(t)$. This is because the measurement device has a fixed position and orientation in the global coordinate frame. An alternative representation of (3) is given by

$$\mathbf{r}'_L = \begin{bmatrix} \xi_L(t) \\ \eta_L(t) \end{bmatrix} = \mathbf{A}_\zeta(\varphi(t)) \begin{bmatrix} \xi_{ML} - \Delta\xi(t) \\ \eta_{ML}(t) - \Delta\eta(t) \end{bmatrix}, \quad (4)$$

where the rotation matrix $\mathbf{A}_\zeta(\varphi(t))$ is defined in the form

$$\mathbf{A}_\zeta(\varphi(t)) = \begin{bmatrix} \cos(\varphi(t)) & \sin(\varphi(t)) \\ -\sin(\varphi(t)) & \cos(\varphi(t)) \end{bmatrix}.$$

Similar to (1), the head end of the plate is parameterized by the polynomial

$$p_H(\eta_{PL}) = \sum_{i=0}^{N_H} c_{H,i} \eta_{PL}^i, \quad (5)$$

with the degree N_H and the polynomial coefficients $c_{H,i}$, $i = 0, \dots, N_H$. Hence, a point $P_H(\xi_H, \eta_H)$ on the head end may be written in vector representation as

$$\mathbf{r}'_H = [p_H(\eta_H(t)) \ \eta_H(t)]^T$$

using (5) or as

$$\mathbf{r}'_H = \begin{bmatrix} \xi_H(t) \\ \eta_H(t) \end{bmatrix} = \mathbf{A}_\zeta(\varphi(t)) \begin{bmatrix} \xi_{MH}(t) - \Delta\xi(t) \\ \eta_{MH} - \Delta\eta(t) \end{bmatrix} \quad (6)$$

in the local coordinate frame using measurements $(\xi_{MH}(t), \eta_{MH})$ of the head end. In contrast to the measurements of the longitudinal boundaries, for the head end, the longitudinal coordinate $\xi_{MH}(t)$ varies with time and η_{MH} is constant. This is because of the spatially fixed measurement lines in the longitudinal direction used for the head end.

Note that (4) and (6) also depend on the states \mathbf{x} of (2).

3. Optimization-based contour detection

The challenging task of determining the boundary of the plate also includes the estimation of the states and inputs of the system (2). An Extended Kalman Filter (EKF), see, e.g., [12], may be used to estimate the states of the system. But the large number of measurements (more than 1000) obtained by the infrared camera makes the real-time usage in the considered application much more difficult due to the resulting extensive computational effort. Optimization-based state estimation serves as another option. An overview of this topic can be found in [13]. Because such methods are able to simultaneously estimate both parameters and states, an optimization-based approach is developed to determine the contour and the movement of the plate (angular velocity, lateral and longitudinal speed of the plate) based on the measurement signals.

3.1. Formulation of the optimization-based contour detection

The optimization-based detection of the contour of the plate can be divided into three parts. First, the coefficients of the polynomial (5) are calculated when the head of the plate is in the camera's field of view (FOV) for the first time. In the second part, the head end of the plate is still in the FOV, which enables the estimation of the longitudinal speed of the plate. When the head end of the plate is no longer in the FOV (third part), the plate speed is held constant for the remaining length of the plate.

3.1.1. Parameterization of the head end

The contour detection starts when the head end of the plate appears in the FOV for the first time and the time t is set to zero, i.e. $t = 0$. Then, the initial state of (2) is chosen as

$$\mathbf{x}(0) = \mathbf{x}_0 = [\xi_{ML,1} \quad 0 \quad 0]^T, \quad (7)$$

with $\xi_{ML,1}$ representing the longitudinal position of the left-most measurement line of the FOV in the global coordinate frame. At this time step, the coefficients of the polynomial parameterization of the head end (5) are calculated once by minimizing the longitudinal offset

$$e_H(\mathbf{x}(t), t; \mathbf{p}_H) = p_H(\eta_H(t)) - \xi_H(t) \quad (8)$$

between measured points at the head end ($\xi_{MH}(t), \eta_{MH}$) and their representation (6) in the least-squares sense

$$\min_{\mathbf{p}_H \in \mathbb{R}^{N_{H+1}}} \sum_{j=1}^{M_H} e_H^2(\mathbf{x}_0, 0; \mathbf{p}_H),$$

with the coefficient vector

$$\mathbf{p}_H = [c_{H,0} \quad c_{H,1} \quad \dots \quad c_{H,N_H}]^T$$

and M_H as the number of used rows of the camera.

3.1.2. Optimization problem with a head end in the FOV

Because the head end of the plate is in the FOV, v_{PL} can be estimated by minimizing the longitudinal offset (8) using the determined coefficient vector \mathbf{p}_H from Section 3.1.1.

A convenient method for determining the unknown polynomial coefficients $c_{L,i}$ of the longitudinal boundary is to minimize the lateral offset $e_L(\mathbf{x}(t), t; \mathbf{p}_L)$ between a measured boundary point ($\xi_{ML}, \eta_{ML}(t)$) and its representation (3) during a certain time period. At the time t , this error is defined as

$$e_L(\mathbf{x}(t), t; \mathbf{p}_L) = p_L(\xi_L(t)) - \eta_L(t), \quad (9)$$

with the coefficient vector

$$\mathbf{p}_L = [c_{L,0} \quad c_{L,1} \quad \dots \quad c_{L,N_L}]^T \quad (10)$$

of the boundary polynomial (1).

Remember that v_{PL} was assumed to be unknown but constant. To find the unknowns $\omega(t), v_L(t), \mathbf{p}_L$ and v_{PL} , a dynamic optimization problem has to be solved, which is formulated in the plate-fixed coordinate frame. In order to obtain a static optimization problem, (2) is discretized using a fixed sampling time T_s . Based on the assumptions $\omega(t) = \omega_k$ and $v_L(t) = v_{L,k}$ during a sampling interval $t_k \leq t < t_k + T_s$, the solution \mathbf{x} of (2) can be calculated analytically. This yields the discrete-time

system

$$\begin{aligned} \mathbf{x}_{k+1} &= \begin{bmatrix} \Delta \xi_k \cos(\omega_k T_s) - \Delta \eta_k \sin(\omega_k T_s) \\ \Delta \xi_k \sin(\omega_k T_s) + \Delta \eta_k \cos(\omega_k T_s) \\ \varphi_k + \omega_k T_s \end{bmatrix} \\ &+ \frac{v_{PL}}{\omega_k} \begin{bmatrix} \sin(\omega_k T_s) \\ 1 - \cos(\omega_k T_s) \\ 0 \end{bmatrix} + \frac{v_{L,k}}{\omega_k} \begin{bmatrix} \cos(\omega_k T_s) - 1 \\ \sin(\omega_k T_s) \\ 0 \end{bmatrix} \\ &= \mathbf{f}(\mathbf{x}_k, \omega_k, v_{PL}, v_{L,k}), \end{aligned} \quad (11)$$

with the state $\mathbf{x}_k = \mathbf{x}(kT_s) = [\Delta \xi_k \quad \Delta \eta_k \quad \varphi_k]^T, k \in \mathbb{N}_0$.

Hence, the static optimization problem can be formulated as

$$\min_{\substack{\boldsymbol{\omega} \in \mathbb{R}^N \\ \mathbf{v}_L \in \mathbb{R}^N \\ \mathbf{p}_L \in \mathbb{R}^{N_L+1} \\ v_{PL}}} \underbrace{\sum_{k=0}^N \sum_{j=1}^{M_L} e_{L,j,k}^2(\mathbf{x}_k; \mathbf{p}_L) + \sum_{k=0}^N \sum_{j=1}^{M_H} e_{H,j,k}^2(\mathbf{x}_k)}_J \quad (12a)$$

$$\text{subject to } \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \omega_k, v_{PL}, v_{L,k}) \quad (12b)$$

with the abbreviations

$$\begin{aligned} e_{L,j,k}(\mathbf{x}_k; \mathbf{p}_L) &= p_L \left(\underbrace{(\xi_{ML,j} - \Delta \xi_k) \cos(\varphi_k) + (\eta_{ML,j,k} - \Delta \eta_k) \sin(\varphi_k)}_{\xi_{L,j,k}} \right) \\ &- \left[\underbrace{(\eta_{ML,j,k} - \Delta \eta_k) \cos(\varphi_k) - (\xi_{ML,j} - \Delta \xi_k) \sin(\varphi_k)}_{\eta_{L,j,k}} \right]. \end{aligned}$$

and

$$\begin{aligned} e_{H,j,k}(\mathbf{x}_k) &= p_H \left(\underbrace{(\eta_{MH,j} - \Delta \eta_k) \cos(\varphi_k) - (\xi_{MH,j,k} - \Delta \xi_k) \sin(\varphi_k)}_{\eta_{H,j,k}} \right) \\ &- \left[\underbrace{(\xi_{MH,j,k} - \Delta \xi_k) \cos(\varphi_k) + (\eta_{MH,j} - \Delta \eta_k) \sin(\varphi_k)}_{\xi_{H,j,k}} \right]. \end{aligned}$$

The optimization variables are the vector of the angular velocities $\boldsymbol{\omega} = [\omega_0 \quad \omega_1 \quad \dots \quad \omega_{N-1}]^T$, the vector of the lateral velocities $\mathbf{v}_L = [v_{L,0} \quad v_{L,1} \quad \dots \quad v_{L,N-1}]^T$, the coefficient vector \mathbf{p}_L and the speed of the plate v_{PL} . M_L is the number of used columns of the measurement device. If no measurement ($\xi_{ML,j}, \eta_{ML,j,k}$) is available for a certain time step k and measurement line j , $e_{L,j,k}(\mathbf{x}_k; \mathbf{p}_L)$ is set to zero. Similarly, $e_{H,j,k}(\mathbf{x}_k)$ vanishes if no head end of the plate is in the FOV at $t = kT_s$. The considered time period ranges from $t = 0$ to $t = NT_s$, i.e., there are $N + 1$ sampling points and the optimization involves N values of ω_k and $v_{L,k}$.

Remark: Measuring the plate contour at just one fixed location ζ_{ML} provides insufficient information to identify the contour polynomial and the movement of the plate. To extract and separate this information, at least two measurement lines ($N_L \geq 2$) at significantly different positions $\zeta_{ML,j}$ must be provided. However, a larger number of measurements improves the robustness of the estimation as noise and fluctuating errors are suppressed by averaging.

3.1.3. Optimization problem without a head end in the FOV

In Section 4, a receding horizon approach for the presented optimization-based algorithm will be discussed. For this, measurement sets of the boundary of the plate without any head end may occur. Such measurement sets do not allow the estimation of the longitudinal speed of the plate with the presented approach. Hence, the estimated speed of the plate obtained from the last optimization with a head end in the FOV, see Section 3.1.2, is used by assuming $v_{PL} = \text{const.}$ for the remaining part of the plate. Furthermore, the optimization problem (12) simplifies to

$$\min_{\substack{\boldsymbol{\omega} \in \mathbb{R}^N \\ \mathbf{v}_L \in \mathbb{R}^N \\ \mathbf{p}_L \in \mathbb{R}^{N_L+1}}} \sum_{k=0}^N \sum_{j=1}^{M_L} e_{L,j,k}^2(\mathbf{x}_k; \mathbf{p}_L) \quad (13a)$$

$$\text{subject to } \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \omega_k, v_{PL}, v_{L,k}). \quad (13b)$$

3.2. Numerical solution of the optimization problem

Suitable algorithms for solving the static optimization problems (12) and (13) are, for instance, the steepest descent method [14], the conjugate gradient method [15], the quasi-Newton method [16], the Newton method [17], the trust-region method [14], and the Gauss-Newton method [14]. For the given problem, the quasi-Newton method proved useful due its superlinear convergence rate (cf. [14]) and the fact that it requires only the evaluation of the cost function J and its gradient in every iteration. The gradient may be calculated by numerical differentiation of the cost function. For the given optimization problem, the gradient is calculated analytically. Compared to the use of numerical differentiation, this leads to a faster convergence of the optimization algorithm as well as to a more accurate solution (cf. [18]).

The gradient of the cost function J with respect to $\boldsymbol{\omega}$, \mathbf{v}_L , \mathbf{p}_L , and v_{PL} is detailed in Appendix A. In the sequel, the numerical solution of the optimization problem (12) is discussed. However, the presented approach can be analogously applied to the optimization problem (13) by omitting the optimization variable v_{PL} . For a compact notation, all optimization variables are arranged in the

vector

$$\mathbf{w} = \begin{bmatrix} \boldsymbol{\omega} \\ \mathbf{v}_L \\ \mathbf{p}_L \\ v_{PL} \end{bmatrix} = [w_0 \ w_1 \ \dots \ w_{2N+N_L+1}]^T.$$

According to [16], the quasi-Newton method using the BFGS-formula to update the estimate of the inverse Hessian \mathbf{H}^{-1} proceeds as follows:

Step 0: Initialize the estimated inverse Hessian \mathbf{H}_0^{-1} and calculate the initial gradient $\mathbf{g}_0 = \nabla J(\mathbf{w}_0)$ for the initial guess \mathbf{w}_0 .

Step 1: Compute the search direction $\mathbf{d}_l = -\mathbf{H}_l^{-1} \mathbf{g}_l$ with $\mathbf{g}_l = \nabla J(\mathbf{w}_l)$.

Step 2: Perform a line search, i.e. solve $\min_{\alpha_l \geq 0} J(\mathbf{w}_l + \alpha_l \mathbf{d}_l)$ and compute the update $\mathbf{w}_{l+1} = \mathbf{w}_l + \alpha_l \mathbf{d}_l$.

Step 3: Check if any termination criterion (maximum number of iterations, convergence) is fulfilled. If yes, stop here.

Step 4: Update \mathbf{H}_l^{-1} in the form

$$\mathbf{H}_{l+1}^{-1} = \left(\mathbf{I} - \frac{\mathbf{d}_l \mathbf{q}_l^T}{\mathbf{q}_l^T \mathbf{d}_l} \right) \mathbf{H}_l^{-1} \left(\mathbf{I} - \frac{\mathbf{q}_l \mathbf{d}_l^T}{\mathbf{q}_l^T \mathbf{d}_l} \right) + \frac{\mathbf{d}_l \mathbf{d}_l^T}{\mathbf{q}_l^T \mathbf{d}_l} \alpha_l$$

with $\mathbf{q}_l = \mathbf{g}_{l+1} - \mathbf{g}_l$.

Step 5: Start again at Step 1.

A crucial point for the convergence rate of the quasi-Newton method is the choice of the initial guess \mathbf{H}_0^{-1} (cf. [14]). The used cost function can be written in the form $J = \mathbf{e}^T \mathbf{e}$. In the vector \mathbf{e} , the lateral offsets $e_{L,j,k}$, $j = 1, \dots, M_L$ and $k = 0, \dots, N$, and the longitudinal offsets $e_{H,j,k}$, $j = 1, \dots, M_H$ and $k = 0, \dots, N$, from (12a) are consecutively arranged. The specific order of these offsets in the vector \mathbf{e} is arbitrary. Let the Jacobian \mathbf{J} of \mathbf{e} with respect to \mathbf{w} be denoted as

$$\mathbf{J}(\mathbf{w}) = (\nabla \mathbf{e})^T.$$

Hence, the Hessian of $J(\mathbf{w})$ can be written as

$$\nabla^2 J(\mathbf{w}) = 2\mathbf{J}^T \mathbf{J} + 2\boldsymbol{\Gamma},$$

with $\boldsymbol{\Gamma} = [\Gamma_{q,n}]$ and

$$\Gamma_{q,n} = \sum_{k=0}^N \sum_{j=1}^{M_L} e_{L,j,k} \frac{\partial^2 e_{L,j,k}}{\partial w_q \partial w_n} + \sum_{k=0}^N \sum_{j=1}^{M_H} e_{H,j,k} \frac{\partial^2 e_{H,j,k}}{\partial w_q \partial w_n}$$

$$\forall q, n \in \{0, 1, \dots, 2N + N_L + 1\}.$$

This motivates the initial guess

$$\mathbf{H}_0^{-1} = \left(2\mathbf{J}^T(\mathbf{w}_0)\mathbf{J}(\mathbf{w}_0) \right)^{-1}. \quad (14)$$

\mathbf{H}_0^{-1} is a positive definite approximation of the inverse of the Hessian if Γ (second order derivatives) is negligible.

The expression $(2\mathbf{J}^T(\mathbf{w}_l)\mathbf{J}(\mathbf{w}_l))^{-1}$ could also be used as an approximation of the inverse Hessian in every iteration of the optimization problem. This choice would result in the Gauss-Newton method (cf. [14]). Despite the fact that only very few iterations are necessary for this method to converge for the given problem, the extensive computational effort of (14) leads to larger total optimization times than the proposed quasi-Newton method.

In step 2, a line search based on a quadratic interpolation of the cost function

$$J(\mathbf{w}_l + \alpha_l \mathbf{d}_l) \approx a_0 + a_1 \alpha_l + a_2 \alpha_l^2 \quad (15)$$

with coefficients a_i , $i = 1, 2, 3$, is performed. The polynomial coefficients a_0 , a_1 and a_2 can be computed in the form

$$a_0 = J_0, \quad a_1 = J'_0, \quad a_2 = J_1 - J_0 - J'_0,$$

where

$$J_0 = J(\mathbf{w}_l), \quad J_1 = J(\mathbf{w}_l + \mathbf{d}_l)$$

and

$$J'_0 = \left. \frac{dJ(\mathbf{w}_l + \alpha_l \mathbf{d}_l)}{d\alpha_l} \right|_{\alpha_l=0} = \mathbf{d}_l^T \mathbf{g}_l.$$

The optimal step length α_l^* that minimizes the right-hand-side of (15) therefore takes the form

$$\alpha_l^* = \frac{1}{2} \frac{J'_0}{J_0 + J'_0 - J_1}.$$

Three different termination criteria are used to check if the solution is acceptable:

- The gradient is sufficiently small, i.e., $\|\mathbf{g}_l\|_\infty < \gamma_g(1 + \|\mathbf{g}_0\|_\infty)$ with the tuning parameter $\gamma_g > 0$.
- The step size is sufficiently small, i.e., $|\mathbf{w}_{l+1} - \mathbf{w}_l| < \gamma_x \left([1 \ \dots \ 1]^T + |\mathbf{w}_{l+1}| \right)$ with the parameter $\gamma_x > 0$.
- The achieved decrease of the cost function value J along the current search direction is smaller than the constant $\gamma_J > 0$, i.e., $J(\mathbf{w}_{l+1}) - J(\mathbf{w}_l + \alpha_{l+1}^* \mathbf{d}_{l+1}) < \gamma_J$.

Properly chosen values for γ_g , γ_x and γ_J ensure both, a sufficiently accurate optimization result and a low number of iterations.

4. Receding horizon approach

With the proposed method, the contour is estimated based on the whole measurement set in one go after the plate has left the rolling gap. This global approach has two drawbacks:

1. There is a large number of optimization variables, which increases with the number of images used, leading to large computing times when solving (12).
2. The contour information is only available after the roll pass has finished. Thus, the contour information cannot be utilized for feedback control.

These problems are avoided if the optimization routine is applied to overlapping sections along the plate. Such a receding horizon approach reduces the number of optimization variables and provides almost real-time contour information.

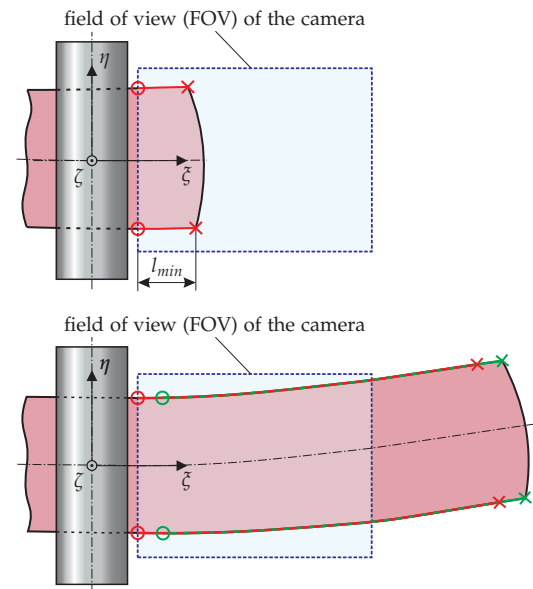


Figure 5: Receding horizon approach for the proposed contour detection method.

The beginning of the optimization procedure is outlined in Fig. 5, where the plate is shown at 2 different times. At any time, the field of view (FOV) of the 2D-camera remains the same. As indicated in the upper part of Fig. 5, the first optimization starts at $t = 0$ when the plate length in the FOV exceeds a lower bound l_{min} . The starting point of the optimization at the boundary of the plate is marked with a cross, while the end point is marked with a circle. As the plate leaves the rolling gap with the speed v_{PL} , the starting points (crosses) move through the FOV, i.e., they are fixed to the corner points of the plate. The end points (circles), however, are spatially fixed. Hence, they move along the boundary and the number of optimization variables increases. This continues until the maximum number N of images used in

one optimization horizon is reached. From this time onwards, the dimension of the optimization problem does not change anymore. Therefore, for every additional captured image, the oldest image in the used measurement set is discarded.

As indicated in the lower part of Fig. 5, the optimization regions are overlapping. In this case, the estimated boundary from the first optimization is shown in green. Without this spatial overlap, discontinuities at the junction of the optimization regions may occur. Nevertheless, a small but negligible deviation between the resulting contours of subsequent optimizations is present. It is defined that the overlapping part of the result from the most recent optimization overwrites the result from the previous run. In case of the example from Fig. 5, the resulting longitudinal boundaries after the first move of the optimization horizon consist of the part of the green line between the red and green cross and the red line.

The previous run is taken as initial guess for the actual one. A reasonable add-on of the algorithm is to estimate also the head- and tail-contour to obtain the whole plate boundary.

Remark: With increasing plate length, also $\Delta\zeta_k$ increases. Especially when using boundary polynomials p_L with a high degree N_L and for $\Delta\zeta_k \gg 1$, the Hessian $\nabla^2 J(\mathbf{w})$ and therefore the optimization problem become ill-conditioned. This property may be challenging in terms of the numerical solution of the optimization problem. An easy countermeasure is to regularly shift the plate-fixed local coordinate frame $(\zeta_{PL}, \eta_{PL}, \zeta_{PL})$ to a new position closer to the FOV and to reset the rotation of the coordinate frame to zero. Then, the estimated contour consists of different polynomials belonging to the respective optimization region. The parameters, which define the displacements and the rotations of the coordinate frame, have to be stored so that the whole contour can be assembled at the end of the rolling pass.

5. Recording and processing of image data

The proposed optimization algorithm requires measurement pairs $(\zeta_{ML,j}, \eta_{ML,j,k})$, $j = 1, \dots, M_L$ and $k = 0, \dots, N$, to estimate the plate contour. For the estimation of the speed of the plate, measurements $(\zeta_{MH,j,k}, \eta_{MH,j})$, $j = 1, \dots, M_H$ and $k = 0, \dots, N$, from the head end of the plate are required. One possible measurement principle is to extract the actual plate boundary from a 2D-image that is taken by an infrared CCD camera mounted above the plate. It is thus natural to select a sampling time T_s that equals the frame rate of the camera. The advantage of using a 2D-camera instead of a line scan device is that several measurement lines (depending on the camera resolution) are concurrently recorded. Furthermore, a single 2D-camera is cheaper than several line scan devices. Compared to a standard color 2D-CCD array, infrared

cameras are superior for the considered application because of the high thermal contrast between the plate and its environment. This is beneficial for the subsequent edge detection, because the plate may be surrounded by a cloud of steam resulting from cooling water sprayed onto the plate during the rolling process (cf. Fig. 6). The industrial camera used in this work captures 30 frames per second with an image resolution of 659×494 pixels.

In the sequel, a fixed mounting position is assumed for the 2D-camera. Many different algorithms are available for edge detection in a 2D bitmap. These algorithms differ in terms of accuracy and computational effort. A frequently used approach is the so called *Canny-algorithm* (cf. [19]), which proved useful for the given problem. This algorithm filters the image with a Gaussian filter to suppress noise in the subsequent calculation of image gradients along both Cartesian directions. Gaussian filtering is also beneficial because of the disturbances caused by the cooling water. After the edge detection, the edges are clustered into lateral and longitudinal edges according to their gradients. Large magnitudes of the gradient in ζ -direction indicate edges at the head and tail of the plate whereas large gradients in the η -direction are linked with longitudinal edges.

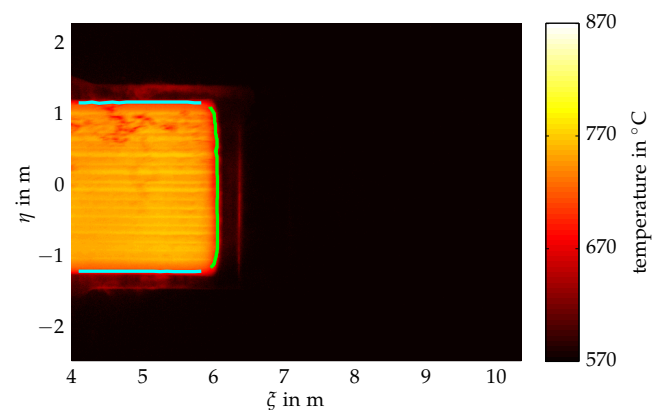


Figure 6: Pixel image of the temperature distribution of a plate rolled in the two-phase-region of austenite and ferrite and detected edges (blue: longitudinal edges, green: lateral edge).

6. Measurement results

In the following, results for the estimated contour of a plate rolled in an industrial rolling mill of the AG der Dillinger Hüttenwerke, Germany are given. Measurement data from a contour measurement device (CMD) located downstream (at the end of the production line) is used to validate the estimated contour. This contour measurement is performed by means of a laser line scanner and some image processing algorithms. The scanner is arranged across the roller table. It captures images (1D arrays) as the plate moves on the roller table along

a strictly straight path (no rotation of the plate). The image frames are joined by software to generate a full 2D picture of the plate contour.

The optimization problem (12) was implemented in C++. The used infrared 2D-CCD camera was installed at the finishing mill of Aktiengesellschaft der Dillinger Hüttenwerke. The camera is mounted 25 m above the pass-line level at the ceiling of the plant building. This isolates the camera from oscillations and harsh conditions (heat, dust, cooling water) near the rolling process. Using a 25 mm lens, a spatial resolution of 9.6 mm/pixel is achieved. By considering both longitudinal edges of the plate, the optimization variable \mathbf{w} reads as

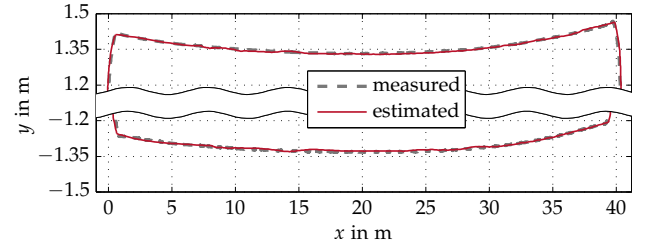
$$\mathbf{w} = [\omega^T \quad \mathbf{v}_L^T \quad \mathbf{p}_{L,left}^T \quad \mathbf{p}_{L,right}^T \quad v_{PL}]^T$$

with the coefficient vectors $\mathbf{p}_{L,left}$ and $\mathbf{p}_{L,right}$ (cf. (1) and (10)) of the boundary polynomials for the left and the right longitudinal boundary, respectively. The initial guess \mathbf{w}_0 is chosen as $\omega_0 = \mathbf{v}_{L,0} = \mathbf{0}$ and $v_{PL,0} = 3$ m/s (common rolling speed). The first entries of $\mathbf{p}_{L,left}$ and $\mathbf{p}_{L,right}$ (constant terms of the polynomials) are set to the mean values of the respective edge in the first detected image. All other elements of $\mathbf{p}_{L,left}$ and $\mathbf{p}_{L,right}$ are initially set to 0.

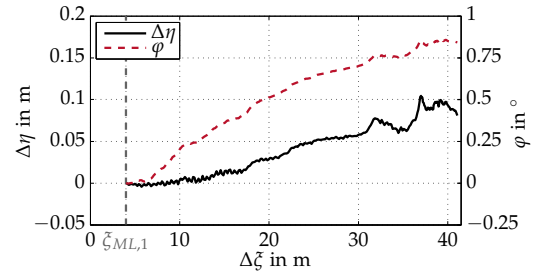
A crucial parameter for the estimation accuracy is the chosen length N of the optimization horizon. A larger value of N results in a smoother estimated contour due to averaging. Smaller horizons induce more noise in the detected contour. Clearly, N also controls the time needed for solving the optimization problem. The actual choice of N is therefore a tradeoff between a sufficiently smooth contour and a reasonable computing time. For the considered measurement configuration, $N = 10$ proved to be a good compromise. The remaining parameters used for the contour detection are shown in Tab. 1. With these parameters, it takes less than 25 ms (Standard PC with i7-2600 @ 3.4 GHz processor and 16 GB RAM) to solve the optimization problem (12) for one optimization horizon. This facilitates contour detection in real-time.

Table 1: Parameters used for the computations.

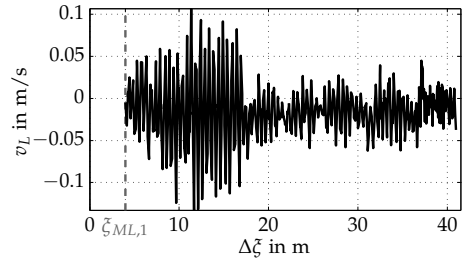
Parameter	Value	Unit
N_L	3	
N_H	4	
N	10	
M_L	659	
M_H	100	
T_s	1/30	s
l_{min}	3	m
$\xi_{ML,1}$	4	m
γ_g	10^{-10}	
γ_x	10^{-8}	
γ_j	10^{-3}	



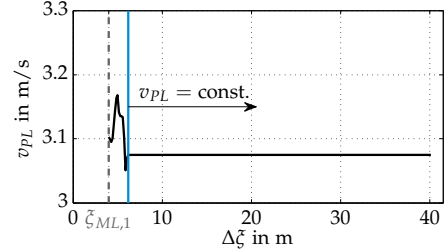
(a) Measured and estimated contour.



(b) Position and orientation of the plate-fixed coordinate frame.



(c) Lateral velocity v_L of the plate.



(d) Longitudinal velocity v_{PL} of the rolled plate.

Figure 7: Measurement and simulation results for a rolled plate.

Fig. 7a shows the contour of the plate both measured by the downstream CMD and estimated by the proposed algorithm. The estimated contour is rotated and shifted to render a comparison with the measured contour in the coordinate frame (x, y) of the CMD. The figure indicates a good accuracy of the contour detection in the range of one centimeter in lateral direction. Also the shapes of the head and tail edges as well as the length of the plate are accurately estimated.

Remark: In Fig. 7b - 7d, the longitudinal displacement $\Delta\xi$ of the plate-fixed coordinate frame starts at $\Delta\xi = \xi_{ML,1} = 4$ m because of the chosen initial condition (7)

and the parameters of the used measurement setup (cf. Tab. 1).

Fig. 7b shows the movement of the plate in form of the position and orientation of the plate-fixed coordinate frame. In Fig. 7c, the estimated velocity v_L in lateral direction is shown. Although there are oscillations present, they are not disturbing the estimation of the contour. This is because the lateral deviations resulting from these oscillations are very small, i.e. they are less than half a centimeter and therefore less than half of the width of a pixel. They are caused, e.g., by vibrations of the camera in lateral direction.

Furthermore, the estimated velocity v_{PL} of the plate is shown in Fig. 7d. As mentioned in Section 3.1, v_{PL} can only be estimated as long as the head of the plate is in the FOV of the camera. Therefore, a constant plate speed is assumed for the remaining part of the plate. The accurately estimated length of the plate indicates that the estimated velocity agrees well with its real (average) value.

Although only the contour of one representative plate is shown in this paper, similar results have been observed for randomly chosen plates rolled over a period of several weeks.

Additionally, the convergence properties for a single optimization horizon of the proposed optimization approach are analyzed. To prevent the optimization from premature termination, the termination criteria are temporarily set to $\gamma_g = \gamma_x = \gamma_f = 0$. In Fig. 8, the decrease of the cost function J in every iteration is shown. The cost value converges within only 5 iterations to the magnitude of the machine precision ($\approx 2.26 \cdot 10^{-16}$). When using the parameters from Tab. 1, the convergence criteria would have been already fulfilled after the second iteration. This shows that the convergence properties of this algorithm are quite good. An important prerequisite for this behavior is the initial choice of the Hessian according to (14).

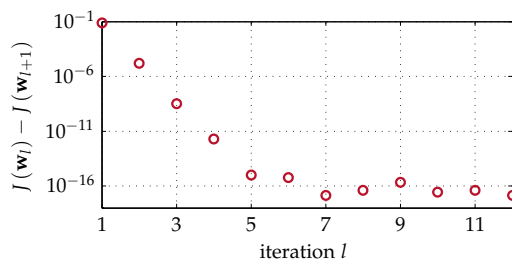


Figure 8: Decrease of $J(w_l)$ as a function of the iteration l .

7. Conclusions and outlook

In this work, an optimization-based algorithm for real-time estimation of the contour of hot rolled plates during the rolling pass was developed. It uses a mathematical model of the movement of the plate and top-view images

of the plate captured by an infrared camera. The model considers the restrictions of the lateral movement of the plate in the rolling gap. Additionally to the contour, the angular movement and the velocity of the plate are estimated. An approach to correct angular misalignments of the camera was also presented. The resulting unbounded optimization problem is solved numerically using the quasi-Newton method. Furthermore, a receding horizon approach is proposed which renders the method suitable for real-time applications.

The results of the proposed algorithm were compared with measurements from a contour measurement device located downstream at the finishing mill of the AG der Dillinger Hüttenwerke in Germany. The comparison indicates a good accuracy of the contour detection with deviations less than one centimeter. Even the velocity of the plate and the edges at the head and tail are accurately estimated.

It is planned to use the developed contour estimation system as a basis for the design of model predictive control algorithm that reduces the occurring camber. To this end, a model linking the deformation in the rolling gap with the resulting plate contour, see, e.g., [20], will be used to calculate the required control inputs to the rolling mill.

Acknowledgements

The authors gratefully acknowledge the financial support and the realization of the measurements by Aktiengesellschaft der Dillinger Hüttenwerke, Germany. The second author gratefully acknowledges financial support provided by the Austrian Academy of Sciences in the form of an APART-fellowship at the Automation and Control Institute of Vienna University of Technology.

Appendix A

The gradient of (12) with respect to the angular velocities ω_r , $r = 0, \dots, N - 1$ takes the form

$$\begin{aligned} \frac{d}{d\omega_r} J(\mathbf{w}) &= 2 \sum_{k=0}^N \sum_{j=1}^{M_L} e_{L,j,k} \frac{d}{d\omega_r} e_{L,j,k} + 2 \sum_{k=0}^N \sum_{j=1}^{M_H} e_{H,j,k} \frac{d}{d\omega_r} e_{H,j,k}, \end{aligned}$$

with

$$\frac{d}{d\omega_r} e_{L,j,k} = \frac{\partial p_L(\xi_{L,j,k})}{\partial \xi_{L,j,k}} \frac{d\xi_{L,j,k}}{d\omega_r} - \frac{d\eta_{L,j,k}}{d\omega_r} \quad (16)$$

and

$$\frac{d}{d\omega_r} e_{H,j,k} = \frac{\partial p_H(\eta_{H,j,k})}{\partial \eta_{H,j,k}} \frac{d\eta_{H,j,k}}{d\omega_r} - \frac{d\xi_{H,j,k}}{d\omega_r} \quad (17)$$

according to (9) and (8), respectively.

The additional derivatives used in (16) can be calculated as

$$\frac{\partial p_L(\xi_{L,j,k})}{\partial \xi_{L,j,k}} = c_{L,1} + 2c_{L,2}\xi_{L,j,k} + \dots + N_{LC_L,N_L}\xi_{L,j,k}^{N_L-1}, \quad (18a)$$

$$\begin{aligned} \frac{d\xi_{L,j,k}}{d\omega_r} &= -(\xi_{ML,j} - \Delta\xi_k) \sin(\varphi_k) \frac{d\varphi_k}{d\omega_r} \\ &\quad + (\eta_{ML,j,k} - \Delta\eta_k) \cos(\varphi_k) \frac{d\varphi_k}{d\omega_r} \\ &\quad - \cos(\varphi_k) \frac{d\Delta\xi_k}{d\omega_r} - \sin(\varphi_k) \frac{d\Delta\eta_k}{d\omega_r} \end{aligned} \quad (18b)$$

and

$$\begin{aligned} \frac{d\eta_{L,j,k}}{d\omega_r} &= -(\eta_{ML,j,k} - \Delta\eta_k) \sin(\varphi_k) \frac{d\varphi_k}{d\omega_r} \\ &\quad - (\xi_{ML,j} - \Delta\xi_k) \cos(\varphi_k) \frac{d\varphi_k}{d\omega_r} \\ &\quad - \cos(\varphi_k) \frac{d\Delta\eta_k}{d\omega_r} + \sin(\varphi_k) \frac{d\Delta\xi_k}{d\omega_r}. \end{aligned} \quad (18c)$$

The derivatives utilized in (17) may be obtained by exchanging p_L with p_H , N_L with N_H , $c_{L,i}$ with $c_{H,i}$ and $\xi_{L,j,k}$ with $\eta_{H,j,k}$ in (18a). Moreover, $\xi_{L,j,k}$ has to be replaced by $\xi_{H,j,k}$, $\eta_{L,j,k}$ by $\eta_{H,j,k}$, $\xi_{ML,j}$ by $\xi_{MH,j,k}$ and $\eta_{ML,j,k}$ by $\eta_{MH,j,k}$ in (18b) and (18c), respectively.

Based on (11), the derivatives $\frac{dx_k}{d\omega_r}$ can be recursively computed in the form

$$\frac{dx_k}{d\omega_r} = \begin{cases} 0 & \text{if } r \geq k \\ \frac{\partial f(x_r, \omega_r, v_{PL}, v_{L,r})}{\partial \omega_r} + \frac{\partial f(x_r, \omega_r, v_{PL}, v_{L,r})}{\partial x_r} \frac{dx_r}{d\omega_r} & \text{if } r = k - 1 \\ \frac{\partial f(x_{k-1}, \omega_{k-1}, v_{PL}, v_{L,k-1})}{\partial x_{k-1}} \frac{dx_{k-1}}{d\omega_r} & \text{if } r < k - 1, \end{cases}$$

with

$$\begin{aligned} \frac{\partial f(x_r, \omega_r, v_{PL}, v_{L,r})}{\partial \omega_r} &= T_s \begin{bmatrix} -\Delta\xi_r \sin(\bar{\omega}_r) - \Delta\eta_r \cos(\bar{\omega}_r) + \frac{v_{PL}}{\omega_r} \cos(\bar{\omega}_r) \\ -\frac{v_{PL}}{\omega_r^2 T_s} \sin(\bar{\omega}_r) - \frac{v_{L,r}}{\omega_r} \sin(\bar{\omega}_r) - \frac{v_{L,r}}{\omega_r^2 T_s} (\cos(\bar{\omega}_r) - 1) \\ \Delta\xi_r \cos(\bar{\omega}_r) - \Delta\eta_r \sin(\bar{\omega}_r) + \frac{v_{PL}}{\omega_r} \sin(\bar{\omega}_r) \\ -\frac{v_{PL}}{\omega_r^2 T_s} (1 - \cos(\bar{\omega}_r)) + \frac{v_{L,r}}{\omega_r} \cos(\bar{\omega}_r) - \frac{v_{L,r}}{\omega_r^2 T_s} \sin(\bar{\omega}_r) \\ 1 \end{bmatrix} \end{aligned}$$

and

$$\frac{\partial f(x_r, \omega_r, v_{PL}, v_{L,r})}{\partial x_r} = \begin{bmatrix} \cos(\bar{\omega}_r) & -\sin(\bar{\omega}_r) & 0 \\ \sin(\bar{\omega}_r) & \cos(\bar{\omega}_r) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (19)$$

where the abbreviation $\bar{\omega}_r = \omega_r T_s$ was used.

The gradient of (12) with respect to $v_{L,r}$ reads as

$$\begin{aligned} \frac{d}{dv_{L,r}} J(\mathbf{w}) &= 2 \sum_{k=0}^N \sum_{j=1}^{M_L} e_{L,j,k} \frac{d}{dv_{L,r}} e_{L,j,k} + 2 \sum_{k=0}^N \sum_{j=1}^{M_H} e_{H,j,k} \frac{d}{dv_{L,r}} e_{H,j,k} \end{aligned}$$

with

$$\frac{d}{dv_{L,r}} e_{L,j,k} = \frac{\partial p_L(\xi_{L,j,k})}{\partial \xi_{L,j,k}} \frac{d\xi_{L,j,k}}{dv_{L,r}} - \frac{d\eta_{L,j,k}}{dv_{L,r}}$$

and

$$\frac{d}{dv_{L,r}} e_{H,j,k} = \frac{\partial p_H(\eta_{H,j,k})}{\partial \eta_{H,j,k}} \frac{d\eta_{H,j,k}}{dv_{L,r}} - \frac{d\xi_{H,j,k}}{dv_{L,r}}.$$

Again the expressions

$$\frac{d\xi_{L,j,k}}{dv_{L,r}} = \frac{d\xi_{H,j,k}}{dv_{L,r}} = -\cos(\varphi_k) \frac{d\Delta\xi_k}{dv_{L,r}} - \sin(\varphi_k) \frac{d\Delta\eta_k}{dv_{L,r}}$$

and

$$\frac{d\eta_{L,j,k}}{dv_{L,r}} = \frac{d\eta_{H,j,k}}{dv_{L,r}} = -\cos(\varphi_k) \frac{d\Delta\eta_k}{dv_{L,r}} + \sin(\varphi_k) \frac{d\Delta\xi_k}{dv_{L,r}}$$

are recursively given by

$$\begin{aligned} \frac{dx_k}{dv_{L,r}} &= \begin{cases} 0 & \text{if } r \geq k \\ \frac{\partial f(x_r, \omega_r, v_{PL}, v_{L,r})}{\partial v_{L,r}} + \frac{\partial f(x_r, \omega_r, v_{PL}, v_{L,r})}{\partial x_r} \frac{dx_r}{dv_{L,r}} & \text{if } r = k - 1 \\ \frac{\partial f(x_{k-1}, \omega_{k-1}, v_{PL}, v_{L,k-1})}{\partial x_{k-1}} \frac{dx_{k-1}}{dv_{L,r}} & \text{if } r < k - 1 \end{cases} \end{aligned}$$

using (19) and

$$\frac{\partial f(x_r, \omega_r, v_{PL}, v_{L,r})}{\partial v_{L,r}} = \begin{bmatrix} \cos(\omega_r T_s) - 1 \\ \sin(\omega_r T_s) \\ \omega_r \\ 0 \end{bmatrix}.$$

The gradient of the cost function $J(\mathbf{w})$ with respect to the parameter vector \mathbf{p}_L yields

$$\begin{aligned} \frac{d}{d\mathbf{p}_L} J(\mathbf{w}) &= 2 \sum_{k=0}^N \sum_{j=1}^{M_L} e_{L,j,k} \begin{bmatrix} 1 & \xi_{L,j,k} & \xi_{L,j,k}^2 & \dots & \xi_{L,j,k}^{N_L} \end{bmatrix}. \end{aligned}$$

The gradient with respect to v_{PL} reads as

$$\begin{aligned} \frac{d}{dv_{PL}} J(\mathbf{w}) &= 2 \sum_{k=0}^N \sum_{j=1}^{M_L} e_{L,j,k} \frac{d}{dv_{PL}} e_{L,j,k} + 2 \sum_{k=0}^N \sum_{j=1}^{M_H} e_{H,j,k} \frac{d}{dv_{PL}} e_{H,j,k} \end{aligned}$$

with

$$\frac{d}{dv_{PL}} e_{L,j,k} = \frac{\partial p_L(\zeta_{L,j,k})}{\partial \zeta_{L,j,k}} \frac{d\zeta_{L,j,k}}{dv_{PL}} - \frac{d\eta_{L,j,k}}{dv_{PL}}$$

and

$$\frac{d}{dv_{PL}} e_{H,j,k} = \frac{\partial p_H(\eta_{H,j,k})}{\partial \eta_{H,j,k}} \frac{d\eta_{H,j,k}}{dv_{PL}} - \frac{d\zeta_{H,j,k}}{dv_{PL}}.$$

Furthermore, it is necessary to calculate

$$\frac{d\zeta_{L,j,k}}{dv_{PL}} = \frac{d\zeta_{H,j,k}}{dv_{PL}} = -\cos(\varphi_k) \frac{d\Delta\zeta_k}{dv_{PL}} - \sin(\varphi_k) \frac{d\Delta\eta_k}{dv_{PL}}$$

and

$$\frac{d\eta_{L,j,k}}{dv_{PL}} = \frac{d\eta_{H,j,k}}{dv_{PL}} = -\cos(\varphi_k) \frac{d\Delta\eta_k}{dv_{PL}} + \sin(\varphi_k) \frac{d\Delta\zeta_k}{dv_{PL}}$$

with recursively computing

$$\frac{dx_k}{dv_{PL}} = \begin{cases} 0 & \text{if } k = 0 \\ \left[\begin{array}{c} \frac{\sin(\omega_{k-1} T_s)}{\omega_{k-1}} \\ \frac{1 - \cos(\omega_{k-1} T_s)}{\omega_{k-1}} \\ 0 \end{array} \right] + \frac{\partial f(x_{k-1}, \omega_{k-1}, v_{PL}, v_{L,k-1})}{\partial x_{k-1}} \frac{dx_{k-1}}{dv_{PL}} & \text{if } k > 0 \end{cases}$$

using (19).

- [1] Y. Tanaka, K. Omori, T. Miyake, K. Nishizaki, M. Inoue, and S. Tezuka. Camber control techniques in plate rolling. Technical Report 16, Kawasaki Steel, June 1987.
- [2] R. C. González, R. Valdés, and J. A. Cancelas. Vision based measurement system to quantify straightness defect in steel sheets. 9th International Conference on Computer Analysis of Images and Patterns, pages 427–434, Warsaw, Poland, September 2001.
- [3] R. J. Montague, J. Watton, and K. J. Brown. A machine vision measurement of slab camber in hot strip rolling. *J. Mater. Process. Technol.*, 168:172–180, 2005.
- [4] B. N. Carruthers-Watt, Y. Xue, and A. J. Morris. A vision based system for strip tracking measurement in the finishing train of a hot strip mill. Proceedings of the 2010 IEEE ICMA, pages 1115–1120, Xi'an, China, August 2010.
- [5] I. Mallocci, J. Daafouz, C. Iung, R. Bonidal, and P. Szczepanski. Robust steering control of hot strip milling. *IEEE Trans. on Control Systems Technol.*, 18(4):908–917, 2010.
- [6] J. W. Yoo, N. W. Kong, J. Song, and P. G. Park. Camber detection algorithm using the image stitching technique in hot-rolling process. International Conference Robotics, pages 74–77, Phuket, Thailand, November 2010.
- [7] J. Lee, N. Kong, J. Yoo, and P. Park. A fast image stitching algorithm in the endless hot rolling process. 11th International Conference on Control, Automation and Systems, pages 1264–1268, Gyeonggi-do, Korea, October 2011.
- [8] N. W. Kong, J. W. Yoo, J. S. Lee, S. W. Yun, J. Bae, and P. G. Park. Vision-based camber measurement system in the endless hot rolling process. *Opt. Eng.*, 50(10):107202–1–107202–10, 2011.
- [9] A. Ollikkala, T. Kananen, A. Mäkynen, M. Holappa, E. Torppa, and T. Harvala. A single camera system for camber measurement in hot strip rolling. Rolling 2013 - 9th International Rolling Conference and the 6th European Rolling Conference, Venice, Italy, June 2013.

- [10] T. Kiefer and A. Kugi. An analytical approach for modelling asymmetrical hot rolling of heavy plates. *Math. and Comput. Model. of Dyn. Syst.*, 14(3):249–267, 2008.
- [11] T. Ishikawa, Y. Tozawa, and J. Nishizawa. Fundamental study on snaking in strip rolling. *Trans. Iron Steel Inst. Jpn.*, 28(6):485–490, 1988.
- [12] D. Dochain. State and parameter estimation in chemical and biochemical processes: A tutorial. *J. Process Control*, 13(8):801–818, 2003.
- [13] J. B. Rawlings and L. Ji. Optim.-based state estimation: Current status and some new results. *J. Process Control*, 22:1439–1444, 2012.
- [14] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer, New York, 2nd edition, 2006.
- [15] E. Polak. *Computational Methods in Optimization: A Unified Approach*. Academic Press, New York, 1971.
- [16] C. T. Kelley and E. W. Sachs. Quasi-newton methods and unconstrained optimal control problems. *SIAM J. Control Optim.*, 25:1503–1516, 1987.
- [17] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, Massachusetts, 2nd edition, 1999.
- [18] C. T. Kelley. *Iterative methods in optimization*. Frontiers in Applied Mathematics. SIAM, Philadelphia, 2nd edition, 1999.
- [19] J. Canny. A computational approach to edge detection. *IEEE Trans. on Pattern Anal. and Mach. Intell.*, 8:679–698, 1986.
- [20] F. Schausberger, A. Steinboeck, and A. Kugi. Mathematical modeling of the contour evolution of heavy plates in hot rolling. *Appl. Math. Modell.*, in Press, 2015.