# Run-To-Run Control of the Czochralski Process

Parsa Rahmanpour[a,*], Steinar Sælid[b], Morten Hovd[c]

[a]*Dept. of Engineering Cybernetics, Norwegian Univ. of Science and Technology (NTNU), NO-7491 Trondheim, Norway.*
[b]*Prediktor AS, Fredrikstad, Norway*
[c]*Dept. of Engineering Cybernetics, (NTNU)*

**Abstract**

Commercially, the Czochralski process plays a key role in production of monocrystalline silicon for semiconductor and solar cell applications. However, it is a highly complex batch process which requires careful control throughout the whole crystal production. In the present paper, an iterative method based on model predictive control (MPC) for calculating a new and improved trajectory from a growth run to the next is explored. The method uses the results of the previous growth run in combination with an underlying model which incorporates the complex dynamic effect of the heater temperature on the pulling rate. The motivation behind this choice of strategy is to enhance the quality of the fully grown ingot from one run to the next by applying the most recent estimates of the unknown parameters. The results show that combining MPC, estimation and Run-To-Run control has enabled simulation of effective control of the Czochralski crystallization process.

*Keywords:* Run-To-Run Control, Model Predictive Control, Czochralski process

## 1. INTRODUCTION

The Czochralski (Cz) process is a well established method for growing large scale single crystals of semiconductors, metals, salts and gemstones. A single crystal (*mono-crystalline*) is a material in which the crystal structure is near perfect, i.e., the arrangement of the atoms or molecules exhibit strict order. The most important use of the method is the growth of semiconductors, particularly that of mono-crystalline silicon, which cannot easily or inexpensively be grown using other methods. Silicon is a particularly important material because it has many desirable properties; it is abundant and cheap, strong and its semiconducting properties are excellent.

Today, solar cells made from crystalline silicon (Si) substrates have a share in excess of 90% of the global solar cell market, and they are likely to continue to dominate this market in many years to come. The efficiency of a solar cell is to a large extent dependant on the

*Corresponding author.
*Email address:* parsa.rahmanpour@itk.ntnu.no (Parsa Rahmanpour)

structure of the wafer, which, in turn depends on the crystallization of the grown crystal (ingot). The solidification process is therefore an important step in reducing PV energy costs and increasing cell efficiency.

In the Cz process, the solid silicon is placed in a crucible. Electrical heaters are used both to melt the silicon, and to maintain an appropriate temperature trajectory throughout the crystallization process. A single crystal seed is put in contact with the molten silicon. The seed is then slowly withdrawn from the melt, and surface tension causes the formation of a *meniscus* which connects the crystal to the melt. As the crystal is withdrawn, the melt solidifies along the top of the meniscus, causing the crystal to grow. The surface along which the material solidifies is referred to as the (crystal-melt) *interface*. In order to grow a crystal with acceptable shape and quality, the heater power as well as the pulling speed are manipulated appropriately during the growth process. After a brief period producing a thin crystal (to allow dislocations caused by the thermal shock when the seed crystal is dipped into the melt to propagate out of the crystal), the crystal diameter is increased quite quickly, whereas for most of the duration of the process it is desirable to keep the crystal diameter constant. In the case of solar cell production, the ingot is sliced into very thin wafers when the crystallization process is over. Each wafer is polished and cut into a specific shape, depending on the final application. A sketch of the Czochralski process is shown in Figure 1. During growth, the radius (diameter) of the crystal is typically measured

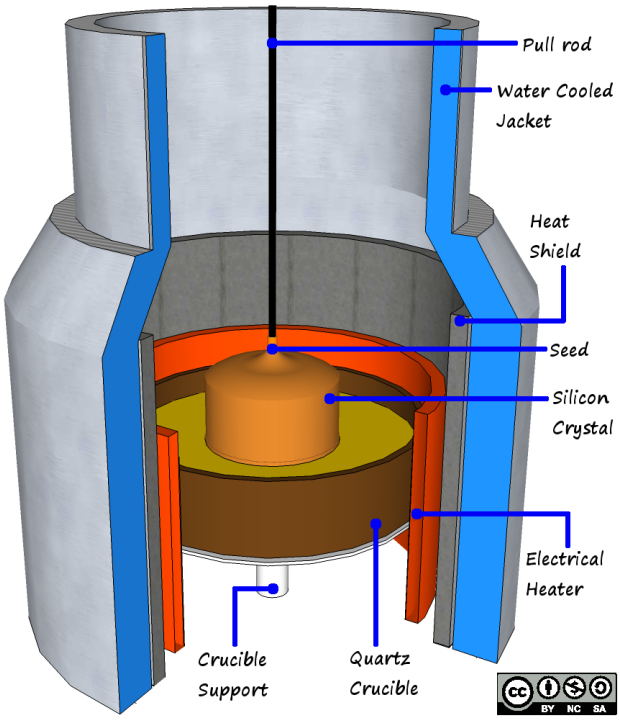


Figure 1: An illustration of the main parts of the Czochralski crystallization process (This Figure is licensed under a Creative-Commons BY-NC-SA license).

by a CCD camera aimed at the meniscus, which can be identified as a 'glowing ring' due to reflections from the glowing hot environment. Since the CCD camera is calibrated to be aimed at the meniscus, the melt level must be the same at all time. The other measurements required in this simulation study are two temperatures from the hot-zone of a Czochralski furnace. Pulling speed and heater power are the two major actuator inputs for the Cz crystal growth process, but they influence the material solidification process at the interface in different ways. The heater power affects the energy balance at the interface region, while the pulling rate acts on the crystal radius through manipulation of meniscus shape and growth angle. To keep the growth process operating uniformly in terms of constant crystal radius and growth rate, the pulling speed should match the growth rate for the applied heater power.

Feedback control was first introduced in the Cz process in order to reduce variations in crystal diameter. Since the 1970's, many different techniques have been applied in practice to meet the requirement for automatic diameter control, which have shown to work well for a variety of conditions. However, there are some side effects to the actions taken by automatic diameter control which have noticeable impact on material properties. The dopant distribution and electrically active defects are among such material properties that could be influenced, due to the coupled nature of the Cz process dynamics. On the other hand, real-time monitoring of crystal quality measures during crystal growth in the Cz process is usually not possible. Therefore, it is necessary to define a richer set of objectives for the closed-loop control. Growth rate and crystal shape are such alternative control objectives which have an impact on the crystal quality, as these are tightly coupled with crystal heat fluxes and melt conditions. The Czochralski process is a typical batch process where the operating conditions are time varying and where steady state is never reached. This mainly results from the fact that the whole system configuration changes during the growth, and in particular, the falling melt level leads to an ever-changing heat entry from the heaters into the system. Furthermore, the process is highly complex and requires careful control throughout the whole crystal production.

Commercial Czochralski pullers using the conventional control scheme depend to a large extent on the feed-forward melt temperature reference trajectory, in order to maintain the desired pulling rate. It is known that crystal quality depends strongly on the melt temperature trajectory. In many cases, the melt temperature reference trajectory is manually adjusted by empirical methods. Different operators and their different understanding of the process thereby increase the variability of the resulting crystal quality. Based on previous works by D.J. Hurle (Hurle, 1993), Y.A. Tatarchenko (Tatarchenko, 1993), M.A. Gevelber (Gevelber and Stephanopoulos, 1987) and Winkler et al. (Winkler et al., 2010), P. Rahmanpour has studied modeling, estimation and control of the Cz-process for crystallization of monocrystalline silicon (Rahmanpour et al., 2013, 2016).

In continuation of the previous study (Rahmanpour et al., 2016), the present paper explores an iterative method based on model predictive control (MPC) and extended Kalman filter (EKF) for calculating a new and improved trajectory from a growth run to the next. The method uses the results of the previous growth run in combination with an underlying model which incorporates the complex dynamic effect of the heater temperature on the

pulling rate. This is an advantage in several ways, for example because the hot-zone of a Czochralski furnace degrades over time, causing the process to change slightly between growth runs. The main motivation for this work is to propose a control methodology with the capability to achieve good control performance for the Cz process. In an actual plant, there may be hundreds of pullers, with subtle differences between each puller (due to different operational history, different ages of hot zone components, etc.). Thus, effective control will require the control methodology to adapt to both the individual differences between pullers, and to how the behavior of the pullers changes with time. Our MPC approach with run-to-run based model update fulfills this requirement, whereas this will be practically impossible with a PID-based control strategy. PID tunings will in practice be changed infrequently, and will be a compromise between different pullers.

Section 2.1 will give a brief introduction to the Czochralski process and present a simplified process model, section 2.2 will give a brief introduction to the introduced control strategy. Section 3 presents the simulation results of applying the designed controller strategy to the process model, whereas section 4 contains discussion and conclusions.

## 2. Modelling and control of the Cz process

When designing a control system, the first task is to characterize the dynamic relations between the manipulated and the controlled variables through process modeling. Based on these relationships, a control structure is chosen. A process model gives insight into the dynamical behavior of the process and enables use of the powerful methods provided by systems theory. Since the dynamics of the Cz process is mainly driven by complex nonlinear radiative, conductive and convective heat transport phenomena, it cannot be fully captured through a linear model (Gevelber and Stephanopoulos, 1987, Hurle et al., 1990).

During the last decades, many researchers in this field of study have worked on *model-based analysis* and *model-based control* of the Cz process for different materials. The proposed models have either been formulated as *distributed parameter systems* (infinite dimensional) or *lumped parameter systems* (finite dimensional).

For example, Bardsley et al. (1974a,b, 1972, 1977) presented the first model-based analyses of the general system behavior using lumped parameter models of selected process parts in the 1970's. Later on, Tatarchenko (1993) studied the modeling and stability of the Cz process in great detail. Furthermore, (Crowley, 1983, Gevelber and Stephanopoulos, 1987, Gevelber et al., 1987, Hurle et al., 1986, 1989, Satunkin, 1995, Satunkin and Rossolenko, 1986) explored important modeling aspects and considerations with respect to the dynamics of the process and the consequences for control system design. Unfortunately, none of the above mentioned studies have been backed up by experimental verification.

Although model-based studies can lead to improved insight into system behaviour, this type of approach also has its drawbacks. Typically, strict assumptions on the model accuracy must be fulfilled in order to make the study results applicable in real life control system. In addition, many parameters have unknown values or their values are not known with sufficient accuracy. Furthermore, actual parameter values may change considerably between different operating points.

One way of improving model accuracy would be to go from lumped parameter models to distributed parameter models. Distributed parameter models can potentially model the spatial dependencies of the system properties more accurately, and thereby describe the system behavior more accurately. The challenge arising from this choice is the necessity of solving complex partial differential equations numerically. As a trade-off to the gained model accuracy, there exist some limitations that affect the applicability of the numerical models in controller design. First of all, powerful solvers would require a great deal of computational performance, which is not possible on the limited computer hardware usually available for process control at the plant. Secondly, the information from the numerical models (and their inverse solutions required for control purposes) might not be available in real-time. Thirdly, much effort is needed to adapt the models to the actual plant design and to determine the physical parameters and boundary conditions that are required by the solver.

Bear in mind that from a controller design perspective, model properties such as observability and controllability are much more important than the 'stand-alone' model accuracy - as observability ensures that the model accuracy can be improved by feedback from on-line measurements, and controllability ensures that the system can be effectively controlled. With the limited number of measurements available for the Cz process, ensuring these properties might be quite difficult when using a detailed numerical model, specially when the numbers of states and parameters are high. Thus, as a control engineer, one should bear in mind that improved 'stand-alone' model accuracy does not necessarily lead to improved closed loop system performance.

Through application of advanced controller scheme and on-line state and parameter estimation, (Rahmanpour et al., 2016) compensated for the lack of model accuracy. One of the essential objectives for this paper is to show that the underlying simplified model can be further enhanced through Run-To-Run adaptation of the unknown estimated parameters.

### *2.1. A Crystal Formation Model for the Czochralski Process*

The geometrical and hydrodynamic relations in the process seem to be well agreed upon throughout the literature. However, there is less agreement in literature on modeling of the complex thermal part of the process. More specifically, the key issue is the description of how the heater power influences the rate of crystallization. Some researchers preferred numerical approaches using powerful finite element methods for solving the time dependent partial differential equations with free boundary conditions (den Bogaert and Dupret, 1997a,b, Dornberger et al., 1996). Others chose model-based control perspective applying lumped parameter models (Crowley, 1983, Gevelber and Stephanopoulos, 1987, Gevelber et al., 1987, Hurle et al., 1986, 1989, Satunkin, 1995, Satunkin and Rossolenko, 1986). Some have even avoided modeling the thermal dynamics only to successfully compensate for it by applying PID controllers (feedforward and feedback loops) as substitute for the missing thermal model (Winkler et al., 2010).

In (Rahmanpour et al., 2016), a combination of the discussed modeling techniques has been used. That work takes advantage from the existing well established geometrical and hydrodynamic relations in the process, which has traditionally been modeled using lumped parameters. The heat transfer from the heater elements to the melt is also modeled using

the same approach, while the heat flux from the meniscus into the crystal is approximated numerically using the result of multiple simulation runs on a high-fidelity simulator (that would be too complex for use in online control). Briefly described, it is necessary to model how the heat is transported from the electric heaters, through the isolation layers, the crucible wall and into the molten bath of silicon. Three control volumes have been used to model the heat transport. The first two control volumes describe the heat transport from the heater elements to the melt, while the last control volume represents the melt itself. In addition, heat loss from each of these control volumes has also been modeled.

The energy supplied by the heater elements to the first control volume is represented by $Q_{\mathrm{H}}$. The total thermal energy or heat flux that enters the melt (third control volume) is denoted as $Q_{\mathrm{in}}$. $Q_{\mathrm{loss1}}$, $Q_{\mathrm{loss2}}$ are used to denote heat losses from control volumes 1 and 2, which are estimated based on stationary numerical simulations, in order to express the heat loss through the argon gas circulation around the crystal and the surrounding cooling water-jacket which encapsulates the puller. $Q_{\mathrm{out}}$ represents the heat transferred away from the melt, and consists of the heat radiated away from the contact-free surface of the melt ($Q_{\mathrm{rad}}$), in addition to the heat transfer through the liquid-solid-interface ($\Phi_{\mathrm{l}}$). In this model the heat transfer dynamics from the heater to the melt is modeled by two intermediate temperatures as shown in Figure 2. The parameters of the proposed model could be adapted a priori from experiments. For instance, $T_h$ models the intermediate temperature which is usually measured by a pyrometer. In practice, the pyrometer measures the temperature on the graphite material, which surrounds the heater element. The purpose of using two additional control volumes in addition to the control volume for the melt itself, is to provide a simple approximate model for the time delay of heat transfer into the melt. During this study, the time delay from heater power to melt temperature is (as a rough approximation) equal to the sum of the time constants for control volumes 1 and 2.

The heat transfer from control volume 1 to control volume 2 is modeled using

$$Q_1 = \lambda_1(T_1 - T_h) \tag{1}$$

where $\lambda_1$ is a heat transfer coefficient. A conventional Czochralski puller has a low level of efficiency with regard to the overall thermal energy consumption. That means, most of the thermal energy from the heater elements will not reach the molten silicon inside the crucible.

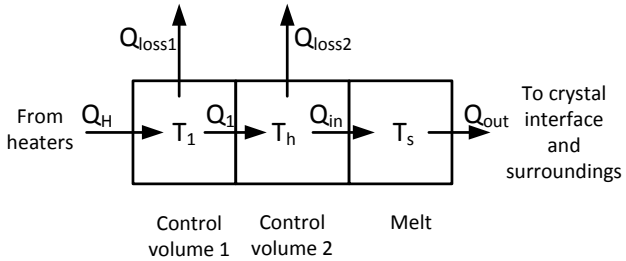


Figure 2: Illustration of a simplified model to express the heat transportation from the heater elements into the melt

The heat transfer from control volume 2 to the melt is therefore modeled as

$$Q_{in} = k_q \lambda_2 (T_h - T_s) \tag{2}$$

Here $k_q$ is a non-constant parameter which describes how much energy is actually transferred and absorbed by the melt. Its value varies with the increasing crystal length. The constantly dropping melt level reduces the contact surface between the melt and the crucible. Simultaneously, the crucible is lifted carefully in order to maintain a constant distance between the melt level and the radius measurement equipment. These factors affect the amount of heat absorbed by the melt and cause variations in $k_q$.

The heat radiation ($Q_{\mathrm{rad}}$) that is necessary to be modeled happens mostly between the melt surface and the surrounding heat shields and crucible walls. The so-called free melt surface is the area between the crucible wall and the crystal body (a donut shaped area). The area of this surface can be calculated as

$$A_{fs} = \pi (r_{cru}^2 - r_c^2) \tag{3}$$

By assuming that both the melt surface and the surface of the surrounding environment have the same emissivity, one can apply the Stefan-Boltzman relationship for $Q_{\mathrm{rad}}$:

$$Q_{\mathrm{rad}} = \sigma \epsilon A_{fs} (T_s^4 - T_{amb}^4) \tag{4}$$

In the equation above, $\sigma$ represents the Stefan-Boltzman constant, $\epsilon$ expresses the emissivity and $T_s$ and $T_{amb}$ are the melt temperature and ambient temperature, respectively. The main challenge with this definition is finding the proper value for the emissivity and the ambient temperature. If possible, one can use experimental data or simulations in order to find an approximation for the unknown parameters.

A heat balance about the crystal-melt interface shows that the heat of crystallization ($\Phi_{\mathrm{h}}$) must equal the difference between the heat flow from the interface to the crystal ($\Phi_{\mathrm{s}}$) and the heat flow from the melt to the interface ($\Phi_{\mathrm{l}}$). By expressing heat flows ($\Phi_{\mathrm{s}}$ and $\Phi_{\mathrm{l}}$) as heat fluxes ($\phi_{\mathrm{s}}$ and $\phi_{\mathrm{l}}$) with the unit [W/m$^2$], the crystal growth rate can be expressed as

$$v_{\mathrm{g}} = \frac{\Phi_{\mathrm{s}} - \Phi_{\mathrm{l}}}{A_{\mathrm{i}} \rho_{\mathrm{s}} \Delta H} = \frac{\phi_{\mathrm{s}} - \phi_{\mathrm{l}}}{\rho_{\mathrm{s}} \Delta H} \tag{5}$$

Here $\rho_{\mathrm{s}}$ is solid state density, $\Delta H$ is the latent heat of fusion per unit volume, and $A_{\mathrm{i}}$ is the interface or meniscus area. Crystal growth occurs as long as the term $\phi_{\mathrm{l}}$ is smaller than $\phi_{\mathrm{s}}$. Otherwise, the sign of $v_{\mathrm{g}}$ changes, i.e., growth is interrupted and the crystal starts to melt. The crystal growth rate has a direct influence on the properties of the growing crystal and correlates with the amount of imperfections.

Assuming a linear heat transfer between the melt and the interface[1], Rahmanpour et al. (2016) proposed the following expression for the melt-meniscus heat flux:

$$\phi_{\mathrm{l}} = k_m (T_{\mathrm{s}} - T_{\mathrm{M}}) \tag{6}$$

[1]And without considering the dependencies around the shape, size and geometry of the meniscus region

Here $T_{\mathrm{s}}$ is the temperature at the base of the meniscus, $T_{\mathrm{M}}$ is the melting temperature of Si and $k_m$ is an unknown constant affected by the convective flows in the melt and the thermal conductivity in meniscus region.

Accurate, but simple calculation of the heat flux from the interface to the crystal has been a very challenging task for many control engineers and researchers in this field. In conventional pullers, the heat flux $\phi_s$ is time varying, which influences the interface growth dynamics as a disturbance. The slow variations in the heat flux are mostly caused by the changes in the crystal radius, the crystal length and the radiation geometry. The continuous increase of the crystal length adds to the crystal's effective heat transfer area which results in a changing thermal environment experienced by the crystal. Other factors that could also cause changes in the thermal environment are the increase in exposure of the crystal to the hot crucible wall (as the melt level drops) and the surrounding cold environment above the hot zone.

In order to include such important dynamics without the need for solving complex PDE's, Rahmanpour et al. (2016) suggest an approximated $\phi_s$ based on simulation data which describes the relationship between the crystal lengths and the corresponding heat fluxes for a specific crystal diameter. This approximation is not constant and varies with the crystal length, which increases throughout the batch operation. Since the approximated $\phi_s$ will not be altered by the manipulated variables (assuming a reasonably well controlled crystal diameter), $\phi_l$ can only be affected by the heater power, through its effect on the temperature in the melt.

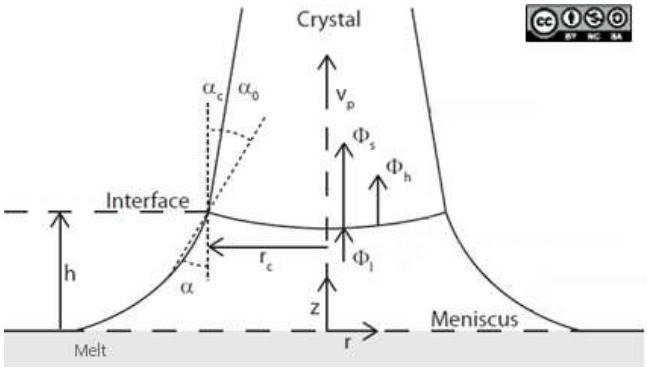


Figure 3: A sketch of the crystal and its contact with the molten metal(This Figure is licensed under a Creative-Commons BY-NC-SA license).

As illustrated in Figure 3, the growth angle $\alpha$ is the angle between the line tangent to the melt-crystal interface and the lateral surface of the growing crystal. $\alpha$ should not be confused with the wetting angle, which characterizes particular equilibrium relative to the liquid movement along a solid body and is not directly associated with crystallization (Tatarchenko, 1993). The growth angle can be defined as the sum of the contact angle $\alpha_0$ and the crystal slope angle $\alpha_{\mathrm{c}}$. In the earliest studies of the Cz technique, a crystal of constant cross - section was considered to grow in the case $\alpha = 0$. However, experimental

and theoretical investigations of the crystal growth process showed that the geometrical condition $\alpha = 0$ is not satisfied while growing crystals of constant cross - sections, and $\alpha$ is a physical characteristic of the melt-crystal system. This means that the slope angle $\alpha_c$ can be zero during crystal growth, but not $\alpha$, when $\alpha_0$ is assumed constant. The time-derivative of the slope angle can be obtained through a mass flow balance about the interface combined with the rate of change in meniscus height ($h$) (Winkler et al., 2010). Under steady state conditions, Johansen (1991) derived the following analytical approximation to describe the relationship between the crystal radius $r_c$, $\alpha$ and $h$

$$\frac{h}{a} = \sqrt{\frac{1 - sin(\alpha)}{1 + 0.6915(r_c/a)^{-1.1}}} \tag{7}$$

Here, the parameter $a$ is the capillary length, often called the *Laplace constant*.

| Notation | Unit | Description | Init. value |
|---|---|---|---|
| $r_c$ | $[mm/s]$ | crystal radius | 2,5 |
| $\alpha_c$ | $[rad]$ | crystal slope angle | 0,001 |
| $T_s$ | $[K]$ | melt temperature | 1714 |
| $T_1$ | $[K]$ | intermediate temperature | 1970 |
| $T_h$ | $[K]$ | intermediate temperature | 1914 |
| $M_s$ | $[kg]$ | melt mass | 108 |
| $L_c$ | $[mm]$ | crystal length | 0 |
| $v_p$ | $[mm/s]$ | crystal pulling velocity | 0,021 |
| $Q_H$ | $[kW]$ | heater power | 90 |
| $r_{CCD}$ | $[mm]$ | measured crystal radius | 2,5 |
| $v_g$ | $[mm/s]$ | growth rate | 0.02 |
| $r_{cru}$ | $[mm]$ | crucible radius | 245 |
| $\rho_l$ | $[kg/m^3]$ | liquid density of Si | 2570 |
| $\rho_s$ | $[kg/m^3]$ | solid density of Si | 2330 |
| $T_M$ | $[K]$ | melting temperature of Si | 1683 |
| $\Delta H$ | $[J/kg]$ | Latent heat of fusion | 1790000 |
| $C_p$ | $[J/(kg * K)]$ | heat capacity | 1000 |
| $Q_{in}$ | $[kW]$ | total thermal energy entering the melt | 7,7 |
| $Q_1$ | $[kW]$ | heat transfer between control volumes in Figure 2 | 28 |
| $Q_{loss_1}$ | $[kW]$ | heat losses from control volumes in Figure 2 | 30 |
| $Q_{loss_2}$ | $[kW]$ | heat losses from control volumes in Figure 2 | 20 |
| $Q_{out}$ | $[kW]$ | heat transferred away from the melt | 7,3 |
| $\phi_l$ | $[W/m^2]$ | melt-meniscus heat flux | 46000 |
| $\phi_s$ | $[W/m^2]$ | heat flux from the interface to the crystal | 130000 |
| $k_q$ | $[fraction]$ | coefficient for absorbed heat energy by the melt | 0.09 |

Table 1: Description of states and central parameters used in Eqs. (8a) and (8b)

A reduced order nonlinear model which incorporates the essential dynamics of the Cz process, while remaining appropriate for designing model-based controllers can be assembled from the relationships described above (Rahmanpour et al., 2016). The model assumes that the crystal is surrounded by a uniform thermal environment and is represented i matrix form, below:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{r}_c \\ \dot{\alpha}_c \\ \dot{T}_s \\ \dot{T}_1 \\ \dot{T}_h \\ \dot{M}_s \\ \dot{L}_c \end{bmatrix} = \begin{bmatrix} v_{\mathrm{g}} \tan \alpha_c \\ ((v_{\mathrm{p}} - v_{\mathrm{cru}} - C_{\alpha,\mathrm{z}} v_{\mathrm{g}})/(C_{\alpha,\mathrm{n}}) \\ (Q_{in} - Q_{out})/(V_s \rho_l C_p) \\ (Q_H - Q_1 - Q_{loss1})/(\tau) \\ (Q_1 - Q_{in} - Q_{loss2})/(\tau) \\ -\rho_s \pi r_c^2 v_{\mathrm{g}} \\ v_{\mathrm{g}} \end{bmatrix} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{8a}$$

$$\mathbf{y} = \begin{bmatrix} r_{\mathrm{CCD}} \\ T_{\mathrm{s}} \\ T_{\mathrm{h}} \end{bmatrix} = \mathbf{g}(\mathbf{x}), \qquad \mathbf{u} = \begin{bmatrix} v_{\mathrm{p}} \\ Q_{\mathrm{H}} \end{bmatrix} \tag{8b}$$

where $r_c$ is the radius of the crystal at the solid-melt interface and $\alpha_c$ is the slope angle. The crystal growth rate is represented by $v_g$ while $v_{cru}$ gives the crucible lift rate. The $u$ vector consists of the manipulators where the pulling velocity and the heater power are denoted by $v_p$ and $Q_H$, respectively. The simulated measurements in this study are the crystal radius measured by the camera ($r_{CCD}$), the melt temperature $T_s$ and the intermediate temperature $T_h$, which is typically measured by a pyrometer as previously explained.

Furthermore, in (8a) $\rho_l$ represents the liquid density, $V_s$ is the volume of the melt inside the crucible, $C_p$ is the heat capacity and $\tau[\frac{J}{K}]$ represents the parameter proportional to the time delay for the heat transfer from the heater to the melt. The equation for the remaining mass of melt ($M_s$) in the crucible uses the idea of mass conservation where the mass of the initial molten silicon is equal to the mass of the produced crystal. Therefore, the growth rate can be used as an indication for how long the crystal length ($L_c$) has increased. For a more detailed description of the parameters applied in (8a), such as $C_{\alpha,\mathrm{z}}$ and $C_{\alpha,\mathrm{n}}$, see (Rahmanpour et al., 2016).

### *2.2. Model-based Control*

#### *2.2.1. Conventional Control Scheme*

The Cz process has traditionally been controlled using nested-loop PID controllers, as the Figure 4 illustrates. The *automatic diameter controller* (ADC) affects the crystal radius by manipulating the pulling rate, and uses feedback from diameter measurement to follow a desired target diameter trajectory. The purpose of the *automatic temperature controller* (ATC), is to ensure that the temperature conditions in the melt are satisfactory. As described earlier, less heat is transferred from the heater to the melt as the melt level decreases during growth, and the temperature controller follows an empirically predetermined setpoint trajectory to compensate for these batch dynamics. The *automatic growth rate controller* (AGC) is placed in a cascade connection with the temperature controller, and adds an offset

to the temperature setpoint trajectory when the output (pulling rate) from the diameter controller differs from a pre-calculated pulling rate trajectory (Bones and Haugen, 2012). The trajectory generator must take physical properties of the system into account, thus

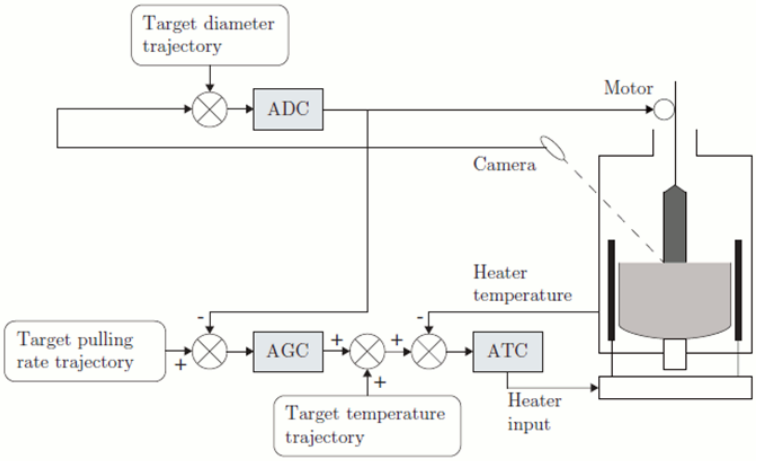


Figure 4: The process has traditionally been controlled using nested single-loop PID controllers (Bones and Haugen, 2012)

avoiding planning of unreachable or non-physical reference values. It should consider both the feasible values for the controlled variables and the physically achievable rates of change in the manipulated variables or states.

However, there are some disadvantages using a PID controller. The biggest drawback of the PID controller is that they are linear, thus the performance of PID controllers in nonlinear systems is variable. Another fundamental difficulty with PID control is that it is a feedback system, with constant parameters, and no direct knowledge of the process. This means that its parameterization is adjusted for a certain operating point of the process. If this operating point changes (as is the case in Cz growth) parameters may have to be scheduled. Another drawback is when the process contains time delays, which is also the case in the Cz process. The delay causes a decrease in phase margin which implies a lower damping ratio and a more oscillatory response for the closed loop system. These limitations have typically been crudely accounted for through empirical gain scheduling, which is both time consuming and difficult (Duffar, 2010).

### *2.2.2. Model Predictive Control*

The tightly coupled dynamics and the physical constraints on both the states and manipulators, motivate the application of a model-based control approach that can handle the nonlinearities, as well as the imposed operational constraints, in an optimal fashion. A model predictive control (MPC) scheme would seem to be an appropriate choice. MPC is an advanced method of process control that has been in use in the process industries since the 1970s. MPC controllers solve an optimization problem at each time instants. The main

point of this optimization problem is to compute a new control input vector to be fed to the system, and at the same time take process constraints into consideration. MPC designs have the ability to yield high performance control systems capable of operating without expert intervention for long periods of time. For more detailed description of model predictive control, see (J.M.Maciejowski, 2000).

Linear MPC can typically achieve acceptable control performance, provided the linearized process model simulates the plant behavior accurately. However, if the plant exhibits severe nonlinearities, the usefulness of predictive control based on a linearized model is limited, particularly if it is used to transfer the plant from one operating point to another one. The Czochralski process is a batch process, which unavoidably has significant changes in dynamics during each batch. For this reason a nonlinear MPC is chosen in the present study.

Given the present state, a prediction of the future manipulated variables (typically obtained from the previous execution of the MPC), and predicted values for future disturbances, the nonlinear model can be used to simulate the system in the future. The future disturbances and references are either provided by skilled process operators, predicted, estimated or simply by taking an extension of the present values into the future. This gives predicted future states that are consistent with the present state and the predicted future manipulated variables and disturbances. Thus, for each time-step in the future, the linearization is performed around the predicted state, manipulated variable and disturbance values. The resulting linearized model would then take the following general form:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}_k\mathbf{x}_k + \mathbf{B}_k\mathbf{u}_k + \mathbf{B}_{d,k}\mathbf{d}_k \\ \mathbf{y}_k &= \mathbf{C}_k\mathbf{x}_k + \mathbf{D}_k\mathbf{u}_k + \mathbf{F}_k\mathbf{d}_k \end{aligned} \tag{9}$$

Here $\mathbf{x}$ represents the state vector, $\mathbf{u}$ is the input vector, $\mathbf{y}$ contains the measured outputs of the model and $\mathbf{d}_k$ is the disturbance vector. By assuming that the cost function does not penalize particular values of the input vector, but only changes of the input vector, the general structure of the cost function can be written as:

$$\mathbf{J}_k = \sum_{i=1}^{H_p=H_u-1} ||\hat{\mathbf{y}}_{k+i|k} - \mathbf{y}^{ref}_{k+i}||^2_{\mathbf{Q}} + \sum_{i=0}^{H_u-1} ||\Delta\hat{\mathbf{u}}_{k+i|k}||^2_{\mathbf{R}} + \sum_{i=H_u}^{\infty} (||\hat{\mathbf{y}}_{k+i|k} - \mathbf{y}^{ref}_{k}||^2_{\mathbf{Q}}) \tag{10}$$

subject to the model in Eq. (9) and

$$\begin{aligned} \mathbf{y}_{min} \leq \mathbf{y}_k \leq \mathbf{y}_{max} \\ \Delta\mathbf{u}_{min} \leq \Delta\mathbf{u}_k \leq \Delta\mathbf{u}_{max} \end{aligned} \tag{11}$$

The introduced objective function minimizes the deviation from the target value ($\mathbf{y}^{ref}_k$) over the prediction horizon. Because of the difference between the predicted and closed-loop response, there is no guarantee that a receding horizon controller based on finite-horizon cost ($H_p$ has a finite value) will achieve the optimal predicted performance. In fact, the closed-loop system may even be unstable. This problem is avoided entirely if performance is evaluated over an infinite prediction horizon.

In practice, the process model is never perfect, thus, it is important to know how reliable the model is and what kind of limitations the controller needs to be aware of. The basic requirement is that the system remains stable under the influence of any unknown disturbances within some expected bounds.

For an open-loop stable linear system, the stability can be guaranteed by the MPC, without the need for terminal constraints. However, based on numerical linearization of the introduced nonlinear model along reference trajectories, it is found that the size of the inverse-eigenvalues (time constants) during the growth of the crystal body are several hours (up to 40 hrs) which are quite large. A large time constant means a close to integrating dynamics. Thus, in addition to the trivial integrating nature of the melt mass and the crystal length, the geometrical part (i.e. $\dot{r_c}$) of the nonlinear model introduced above seems to have a similar behavior as an integrating plant.

For unstable systems or integrating systems, the infinite horizon approach incorporates constrained stabilizability to bring unstable modes to the origin. That means, an effective way of dealing with such problem is to stabilize the predictions. What is involved is a kind of re-parametrization of the plant's input signal. The problem can be overcome by expressing predictions as perturbations on a stabilizing linear feedback law, see, e.g., (J.M.Maciejowski, 2000).

In order to successfully implement an MPC, some tuning parameters must be appropriately chosen. These include the *prediction horizon*, the *control horizon* and the *weight matrices* within the objective function. The prediction horizon is chosen in order to best satisfy a compromise between the stability of the closed-loop and the computational time required. Furthermore, the control horizon is often chosen equal to the prediction horizon, which is also the case in this study.

Applying these tuning rules to a nonlinear model with both fast and slow dynamics can be a tough task. For the fast dynamics, it is appropriate to choose a short prediction horizon combined with relatively small $\mathbf{R}(i)$s. This way, the computational load is kept acceptably low and the solution remains feasible. On the contrary, the slow dynamics require much longer prediction horizon before they reach steady state, while the weights $\mathbf{R}(i)$ should be chosen much larger relative to the $\mathbf{Q}(i)$s. Choosing a too large control horizon for fast dynamics in the presence of model mismatch and disturbances would most probably result in infeasibility. Similarly, a 'short-sighted' controller will have a degraded closed-loop behavior, when it has to deal with slow dynamics. One approach to simplify the tuning burden in these situations can be decoupling the model into parts with fast and slow dynamic responses, without loosing the connection between the states, outputs and inputs.

In the case of the Cz process, there exist two types of system dynamics. The fast dynamics (quick response to input changes) are represented by the geometrical properties of the process such as the radius and the meniscus shape. On the other hand, the bulk part of the process is responsible for the slow dynamics due to delays in the heat transportation. Using only one MPC for controlling such complex and coupled dynamics showed not so robust performance when introducing parameter errors in the prediction model. Therefore, dividing the control part into two separate controllers where the *puller MPC* handles the fast dynamics while the *heater MPC* deals with the slow dynamics, became a promising

choice of control structure. This makes it possible to choose different tuning parameters for each of the controllers based on their model dynamics. The objective functions for the puller MPC and the heater MPC would then become

$$J_{puller} = \sum_{i=1}^{H_{p_1}=H_{u_1}-1} \begin{bmatrix} \hat{r}_{c_{k+i|k}} - r^{ref}_{c_{k+i}} \\ \hat{\alpha}_{c_{k+i|k}} - \alpha^{ref}_{c_{k+i}} \end{bmatrix}^T \begin{bmatrix} Q_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{r}_{c_{k+i|k}} - r^{ref}_{c_{k+i}} \\ \hat{\alpha}_{c_{k+i|k}} - \alpha^{ref}_{c_{k+i}} \end{bmatrix} + \sum_{i=0}^{H_u-1} \Delta v_{p_{k+i|k}}[R_1]\Delta v_{p_{k+i|k}} \tag{12}$$

$$J_{heater} = \sum_{i=1}^{H_{p_2}=H_{u_2}-1} \begin{bmatrix} \hat{T}_{s_{k+i|k}} - T^{ref}_{s_{k+i}} \\ \hat{T}_{h_{k+i|k}} - T^{ref}_{h_{k+i}} \end{bmatrix}^T \begin{bmatrix} Q_2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{T}_{s_{k+i|k}} - T^{ref}_{s_{k+i}} \\ \hat{T}_{h_{k+i|k}} - T^{ref}_{h_{k+i}} \end{bmatrix} + \sum_{i=0}^{H_u-1} \Delta Q_{H_{k+i|k}}[R_2]\Delta Q_{H_{k+i|k}} \tag{13}$$

These objective functions were subjected to the linearized model in Eq. (9) and the appropriate constraints on the measurements, manipulated values and their rate of change. It is noteworthy that the particular MPC implementation used in this study is not capable of considering constraints on the states. Thus, $\hat{\alpha}_c$ is taken as a measurement in order for the MPC to be able to apply constraints on it. On the other hand, every measured state must have a reference trajectory in this formulation, even though none exists for the growth angle. This is compensated for by using zero as the respective weight in the Q-matrix for $\hat{\alpha}_c$, regardless of the chosen value for $\alpha^{ref}_c$.

The same tuning technique is used for the $T_h$, even though it is assumed to be measured. The reason is that it would in most cases not be possible for the MPC to follow two reference trajectories with only one available manipulated variable ($Q_H$). In this case, it is much more important to track the $T^{ref}_{s_{k+i}}$ as well as possible, rather than following both trajectories.

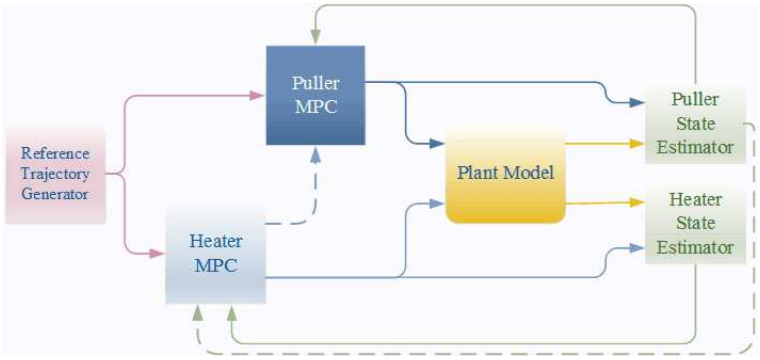


Figure 5: Control Strategy with two MPCs (Rahmanpour et al., 2016)

As Figure 5 illustrates, the data flow starts with the *reference trajectory generator*, which contains the pre-calculated radius and melt temperature reference trajectories for one batch (or run). In case the trajectories are functions of crystal length, the generator needs to convert them to functions of time, using the growth rate trajectory (made available by the

metallurgical engineers and researchers at production site). Then, the radius reference is sent to the Puller MPC while the reference for the melt temperatures follows the path to the Heater MPC.

Each of the MPCs will contain their appropriate parts of the plant model. Thus, the internal model of the Puller MPC is described by:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{r}_c \\ \dot{\alpha}_c \end{bmatrix} = \mathbf{f_1}(\mathbf{x}, \mathbf{u}, \mathbf{d}), \qquad \mathbf{d} = \begin{bmatrix} M_\mathrm{s} \\ \phi_\mathrm{s} \\ T_\mathrm{s} \end{bmatrix} \tag{14a}$$

$$\mathbf{y} = \begin{bmatrix} r_\mathrm{CCD} \end{bmatrix} = \mathbf{g_1}(\mathbf{x}), \qquad \mathbf{u} = \begin{bmatrix} v_\mathrm{p} \end{bmatrix} \tag{14b}$$

while the Heater MPC uses the remaining states in its internal model:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{T}_s \\ \dot{T}_1 \\ \dot{T}_h \end{bmatrix} = \mathbf{f_2}(\mathbf{x}, \mathbf{u}, \mathbf{d}), \qquad \mathbf{d} = \begin{bmatrix} M_\mathrm{s} \\ k_\mathrm{q} \\ r_\mathrm{c} \end{bmatrix} \tag{15a}$$

$$\mathbf{y} = \begin{bmatrix} T_\mathrm{s} \\ T_\mathrm{h} \end{bmatrix} = \mathbf{g_2}(\mathbf{x}), \qquad \mathbf{u} = \begin{bmatrix} Q_\mathrm{H} \end{bmatrix} \tag{15b}$$

The melt mass $M_s$, the model parameters $\phi_s$ and $k_q$, in addition to the estimated state $\hat{r}_c$ and the predicted state $\bar{T}_s$ are fed to their respective controllers as disturbances. In the illustrated design by Figure 5, the transfer of $T_s$ predictions from the heater MPC to the puller MPC shows the importance of internal communication between the two controllers, since $T_s$ is required for model calculations inside the puller MPC. In order to maintain the data flow and the component dependencies, the proposed structure is designed to run sequentially. Thus, the first component to run its calculations is the plant model. Subsequently, all the necessary states and parameters needed for model update are estimated and fed back through two extended Kalman filters, as explained in the next section. Then, the reference trajectory generator is next in line, before running the MPCs. In addition, the heater MPC should run before the Puller MPC, since all of its states and disturbances are available at this stage. At the end, the Puller MPC can do its calculations, using all available data.

As Figure 5 illustrates, the transfer of $T_s$ predictions from the heater MPC to the puller MPC shows the importance of internal communication between the two controllers, since $T_s$ is required for model calculations inside the puller MPC. Furthermore, all the necessary states and parameters needed for model update are estimated and fed back through two extended Kalman filters, as explained in the next subsection.

### *2.3. STATE ESTIMATION*

Using a state space model in model-based controller design, as in the previous section, requires estimation of the states unless all states are measured. The measurement signals for the radius, the melt temperature and the temperature on the graphite material, which surrounds the heater element are quite noisy. Thus, the actual radius and the melt temperature, the slope angle, the meniscus height, $T_1$ and $T_h$ are not available during crystal growth.

In order for these states to be used in the feedback loop, they need to be estimated from measurements. In addition to the explored estimation of the crystal radius in (Rahmanpour et al., 2014) and reconstruction of the temperature states in (Rahmanpour et al., 2016), this paper applies the same *Extended Kalman Filtering* (*EKF*) technique for estimations of the important parameters such as $\phi_s$ and $k_q$.

Some of the most successful applications of Kalman filtering have been in situations with nonlinear dynamics and/or nonlinear measurement relationships. Extended Kalman filters are a logical generalization of linear Kalman filters for the case where the system dynamics vary with operating and control points in nonlinear systems, such as the introduced Cz model. The EKF uses a nonlinear model to predict a new state estimate, and that the linear model used for updating the estimates is relinearized along the systems trajectories. In this case, the algorithm takes the following form (Brown and Hwang, 1997, pp. 343-347).

It is common to convert the continuous-time model to a discrete-time model in order to implement the model in a computer. Therefore, the estimator equations in this section are illustrated in their discrete-time version. The following notation is used in this section:

- $\hat{\mathbf{x}}$ is the state estimate
- $\hat{\mathbf{x}}^-$ is the predicted (a priori) state estimate
- $\hat{\mathbf{y}}$ is the measurement estimate
- $\mathbf{P}$ is the estimate covariance
- $\mathbf{Q}$ is the covariance of the process noise
- $\mathbf{R}$ is the covariance of the observation noise
- $\mathbf{K}_k$ is the Kalman gain
- **Step 1: Predict**

  Predict new state estimate

  $$\hat{\mathbf{x}}_{k+1}^- = \mathbf{F}(\hat{\mathbf{x}}_k, \mathbf{u}_k)$$

  Compute the error covariance of the new estimate before accounting for the new measurement

  $$\mathbf{P}_{k+1}^- = \mathbf{A}_{\mathrm{d},k}\mathbf{P}_k\mathbf{A}_{\mathrm{d},k}^\top + \mathbf{E}_{\mathrm{d},k}\mathbf{Q}_k\mathbf{E}_{\mathrm{d},k}^\top$$

- **Step 2: Update**

  Compute the Kalman gain

  $$\mathbf{K}_k = \mathbf{P}_k^-\mathbf{C}_{\mathrm{d},k}^\top\left(\mathbf{C}_{\mathrm{d},k}\mathbf{P}_k^-\mathbf{C}_{\mathrm{d}}^\top + \mathbf{R}_{\mathrm{d},k}\right)^{-1}$$

Update the state estimates

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k \left[\mathbf{y}_k - \mathbf{G}\left(\hat{\mathbf{x}}_k^-, \mathbf{u}_k, k\right)\right]$$

Compute the error covariance for the updated estimate

$$\mathbf{P}_k = \mathbf{E}\left[\mathbf{e}_k \mathbf{e}_k^\top\right] = \left(\mathbf{I} - \mathbf{K}_k \mathbf{C}_{\mathrm{d},k}\right) \mathbf{P}_k^- \left(\mathbf{I} - \mathbf{K}_k \mathbf{C}_{\mathrm{d}}\right)^\top + \mathbf{K}_k \mathbf{R}_{\mathrm{d},k} \mathbf{K}_k^\top$$

where the Jacobian matrices $\mathbf{A}_{\mathrm{d},k}$ and $\mathbf{C}_{\mathrm{d},k}$ are re-calculated along the systems trajectories according to the discretization of

$$\mathbf{A}_{\mathrm{d},k} = \left.\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_k, \mathbf{u}_k, \mathbf{k}} \quad \text{and} \quad \mathbf{C}_{\mathrm{d},k+1} = \left.\frac{\partial \mathbf{G}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_{k+1}^-, \mathbf{u}_k, \mathbf{k}}$$

Similar to the separation of the controllers (Sec. 2.2.2), two EKFs have been applied in this study for reconstruction of the noisy or unmeasured states. Each estimator has it's own specific sets of tasks that must be fulfilled. The *puller* EKF estimates the real radius, $\alpha_c$ and $\phi_s$ which is the parameter related to calculation of $v_g$, as described by Eq. (14a) and (14b). The second EKF estimates the actual $T_s$, $T_1$, $T_h$ and $k_q$ which is the parameter required for calculating $Q_{in}$ (Eq. (2)) as explained by Eq. (15a) and (15b).

In order to evaluate observability of each of the sub-systems used as models in the EKFs, the sub-systems were linearized at five different operation points consisting of crystal lengths $1mm$, $20mm$, $60mm$, $100mm$ and $200mm$. Based on the full rank of the calculated observability matrices, the augmented models (states and parameter) of both sub-systems are locally observable.

Since some of the states are generally measured either the first or the average of the first few measurements can be taken as the initial value $\mathbf{x_0}$ for the state. The initial value of the parameters, which are used as augmented states, can be set based on their respective reference trajectories. In principle, $\mathbf{P_0}$ can be set arbitrary. However, the necessity for proper tuning of $\mathbf{P_0}$ lies in the following. For instance, if $\mathbf{P_0}$ is set equal to zero then the filter learns nothing from the measurements. Moreover, choosing a very large $\mathbf{P_0}$ would force the filter to believe the measurements much more and ignore the state model values. The latter choice would result in quite large fluctuation in the state and parameter estimates, in addition to large final uncertainty. When it comes to $\mathbf{R}$, a good initial estimate can be obtained from the calibration of the measuring instrument, which in general, is assumed to be constant over the data length. Furthermore, the $\mathbf{Q}$ should reflect the uncertainty in the assumed state model. Its value should be small enough to retain the learning potential from the measurement, but not so large that the increased uncertainty results in useless filter estimates.

The performance of these estimators was dependent on the accurate characterization of the uncertainties in the state dynamics and in the measurement. The parameters of the noise densities associated with these uncertainties were, however, treated as tuning parameters and were adjusted in an ad hoc manner while carrying out state estimation.

*2.3.1. Smoothing*

The smoothing problem is of particular interest in post-experimental data analysis. The Kalman filter estimates the present state of a model based on measurements up to and including the present time, which is referred to as the a posteriori estimates. In contrast, smoothing refers to estimating a state vector at a time point intermediate to a span of measurements. Consequently, there is an essential element of non-causality in smoothing since some of the measurements occur at a later point in time than the time of the state(s) to be estimated. In this study, Discrete Fixed-Interval Smoothing has been chosen as the proper approach for post-experimental data analysis. Here, the time interval of measurements is fixed and the algorithm seeks optimal estimates at some, or perhaps all, interior points. Fixed-Interval smoothing is among the reasonably efficient recursive algorithms that has been extensively studied since the 1960's. The algorithm was published by Herbert E. Rauch, F. Tung, and Charlotte T. Striebel (RTS) in 1965 (Rauch et al., 1965) and is the fastest fixed-interval smoother (Park and Kailath, 1996). The implementation formulas have had some tweaking since then, however, the RTS algorithm has been used quite successfully for decades. The assumptions relative to the process and measurement models are the same as for the filter problem. The computational procedure for the RTS algorithm consists of a forward recursive sweep followed by a backward sweep. The first (forward) pass uses a Kalman filter but saves the intermediate a *priori* and *posteriori* state estimates ($\hat{\mathbf{x}}_{k|k-1}$ and $\hat{\mathbf{x}}_{k|k}$) and their associated covariance matrices ($\mathbf{P}_{k|k-1}$ and $\mathbf{P}_{k|k}$) at each measurement time $t_k$. These are needed for the backward sweep. The second pass runs backward in time in a sequence from the time $t_N$ of the last measurement, computing the smoothed state estimate from the intermediate results stored during the forward pass (Brown and Hwang, 1997).

The combination of a forward pass estimator (EKF) and a backward recursion is considered to have utilized all available information. Thus the scheme is capable of performing better estimations than the EKF alone. Equation 17 summarizes the backward recursion of the RTS smoother:

$$\mathbf{G}_k = \mathbf{P}_{k|k}\mathbf{A}_{k+1}^T\mathbf{P}_{k+1|k}^{-1} \tag{17a}$$

$$\hat{\mathbf{x}}_{k|N} = \hat{\mathbf{x}}_{k|k} + \mathbf{G}_k(\hat{\mathbf{x}}_{k+1|N} - \hat{\mathbf{x}}_{k+1|k}), \quad k = n-1, n-2, \cdots, 0 \tag{17b}$$

$$\mathbf{P}_{k|N} = \mathbf{P}_{k|k} + \mathbf{G}_k(\mathbf{P}_{k+1|N} - \mathbf{P}_{k+1|k})\mathbf{G}_k^T \tag{17c}$$

where

- $\mathbf{G}_k$ is the smoother gain matrix
- $N$ is the final time step
- $\mathbf{P}_{k|N}$ is the corresponding state error covariance matrix
- $\hat{\mathbf{x}}_{k|N}$ is the smoothed estimate of the state at time $k$, accounting for measurement information up until time $N$.

*2.4. Run-To-Run Control*

Run-To-Run (or run-by-run) control was first proposed by Sachs and his co-workers at MIT in the beginning of 1990s (Wang et al., 2009). The main motivation for development of Run-To-Run control is the lack of in situ measurements for the product qualities of interest. A typical example is semiconductor manufacturing, where the goal is to control qualities, which are difficult or impossible to measure in real-time. In semiconductor manufacturing processes, products are typically processed in the form of separate runs (batches). A run in the case of the Cz process is defined as a series of operations on the initial material (silicon) within an inseparable time interval, that would result in a fully grown ingot (crystal), *i.e.*, the production of each crystal represents a run. During processing, the resulting ingot from each batch has received identical treatments, given that the same reference trajectories (recipes) has been used in all runs.

For commercial Czochralski ingot pullers using the conventional control scheme, feedforward control is an essential part of the overall control system. For instance, the conventional pullers depend to a large extent on the feed-forward temperature reference trajectory to keep the desired pulling rate. A common method in practice is to determine the feedforward control by careful analysis of repeated growth runs, resulting in a trajectory for the control inputs which can then be used as part of the recipe. Although this method is widely accepted, it suffers from the fact that it is extremely time consuming and, thus, expensive. Besides that, it works well if the same conditions are repeated and there are no significant variations from run to run. However, any change in plant setup or in desired crystal diameter means repeating this procedure.

Moreover, the thermal environment of the growing crystal is of immense importance, since the hot-zone characteristics of the crystal-growing equipment affects the concentration and distribution of defects within the grown crystal ingot. However, the hot-zone of a Czochralski furnace degrades over time, causing the process to change slightly between growth runs. Among such factors one could mention gradual mechanical changes (wear and tear) in the crystal pulling mechanism or the susceptor, which lifts and rotates the crucible. In order to compensate for such process variations, the recipes and treatments applied to the ingots in different batches can be adjusted. In addition, when using a lumped-parameter model to express the Cz process, the system equations describe the time dependency of some system variables. The behavior of these quantities depends on initial values and control inputs. The desired inputs are calculated from given trajectories of the system quantities. Therefore, adaptation of such model quantities and trajectories to the real crystal-growing process is essential in terms of controller performance enhancement and ingot quality. Thus, R2R controllers can also be used to update the process model parameters in a recursive way when output from the previous run becomes available, and thereby help generate better recipes for future runs. Most traditional R2R controllers are designed to take real-time or delayed process output information and continuously generate updated recipes to compensate for process shifts or drifts. In general, in order to reduce failures caused by potential process drifts or shifts between different batches, research on Run-To-Run (R2R) process control has attracted extensive attention in recent years (Wang et al., 2009).

In this paper, unknown or uncertain model parameters are estimated during a run. The

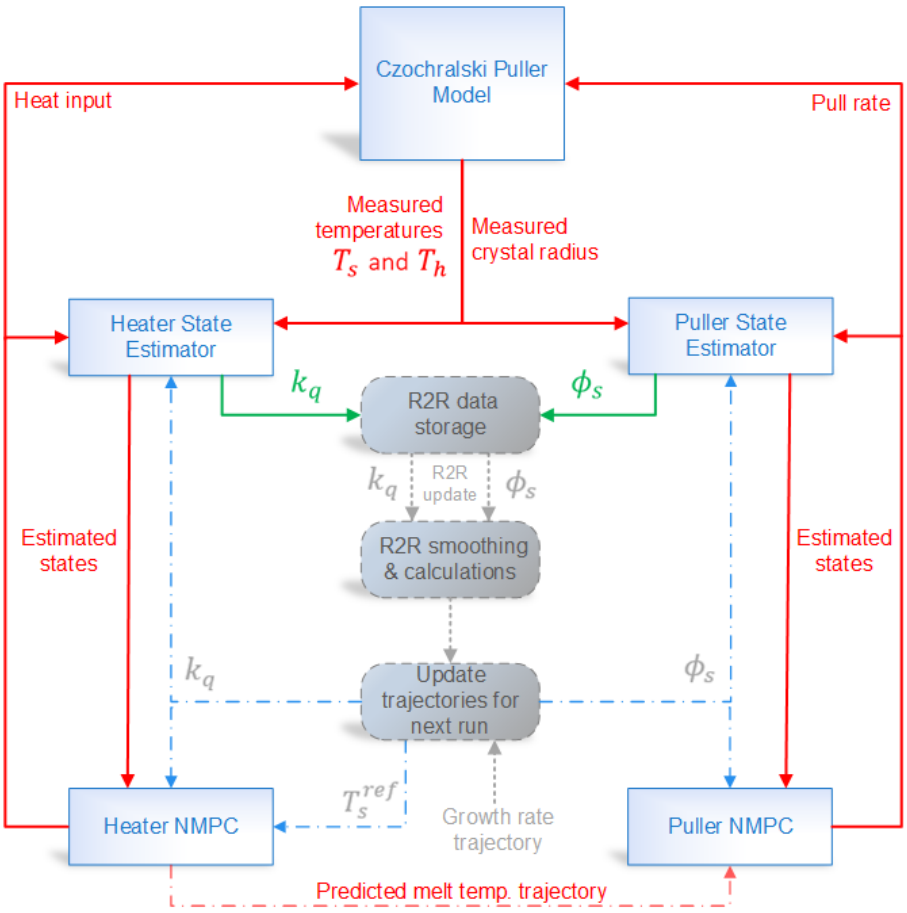


Figure 6: An illustration of the R2R control strategy applied during this study. The red solid and dashed arrows indicate basic online data transactions. The green arrows represent the online parameter estimations. The gray units and dashed arrows illustrate the offline calculations and transactions between two runs, while the blue dashed arrows show application of the most updated trajectories based on previous runs.

new estimates of the initial parameter values are expected to help increase the model adaptation to the real process. The motivation behind this choice of strategy is to enhance the quality of the fully grown ingot from the next run by applying the most recent estimates of the unknown parameters. The structure for a R2R control system applied here is illustrated in Figure 6.

To illustrate the effect of R2R control in this study, four runs have been simulated where the model, controllers and estimators are set-up similarly. During each run, both the states ($r_c$, $\alpha_c$, $T_s$, $T_1$ and $T_h$) and the parameters ($\phi_s$ and $k_q$) are estimated. However, $\phi_s$, $k_q$ and the reference trajectory for $T_s$ are the only parameters that are modified from one run to the next. For instance, using a simple initial guess as the trajectory for the $\phi_s$, the green dashed line in Figure 7 is fed to the puller MPC during the first run, while the plant model uses the correct trajectory, represented by the red dashed line. During the run, the *puller EKF* tries to estimate the true $\phi_s$. At the end of the run, the result is applied to the smoother and

then the smoothed estimate is saved to be used for the next run. Similar to the estimation of the $\phi_s$, the heat transfer coefficient $k_q$ is also estimated simultaneously using the *heater EKF*.

## 3. SIMULATION RESULTS

The model in section 2.1 describes the basic process of growing silicon crystals for solar wafer and cell production. Based on the suggested model-based control and estimation structure (Figure 5), Rahmanpour et al. (2016) show that estimation of a growth model is feasible.

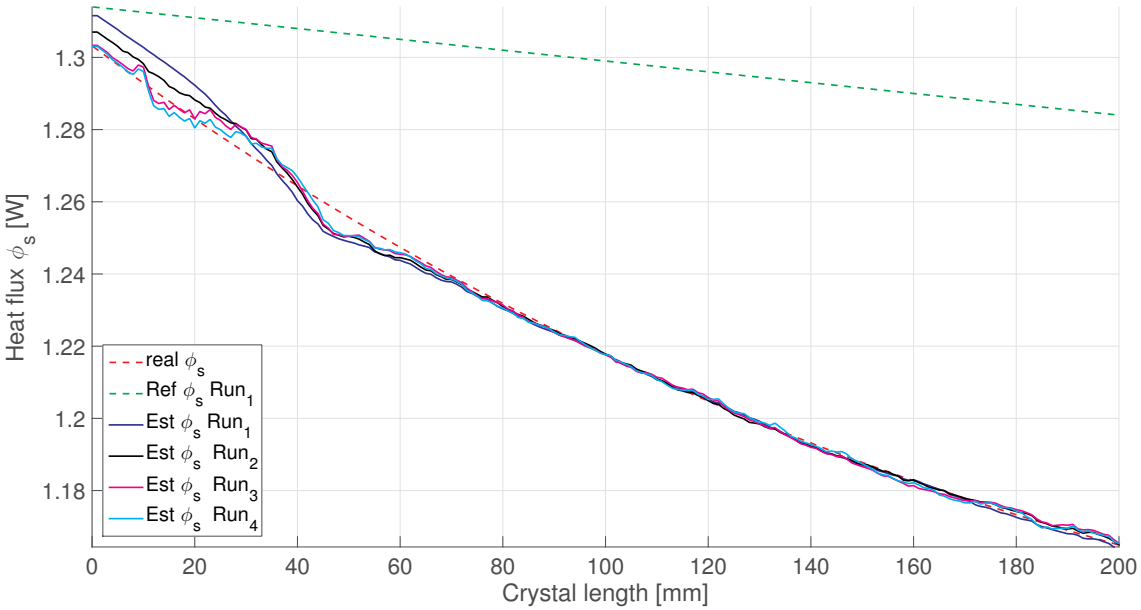


Figure 7: Estimated $\phi_s$ based on run 1 (dark blue) when applied initial $\phi_s$ was the simple graph in green. $\phi_s$ from run 2 (black) was estimated based on information from previous run. Run 3 (pink) and run 4 (cyan) were conducted in a similar fashion to show that the estimations will converge. The red graph shows the real calculated $\phi_s$ for this specific crystal growth.

Figures 7 and 8 show that, as expected, the estimated $\phi_s$ and $k_q$ converge more and more towards their respective true values after each run. Already after two runs, the model mismatch between the model in the MPC controllers and the plant is heavily reduced. Figure 9 describes the effect of the enhanced estimates of $\phi_s$ and $k_q$ on the calculated growth rate, compared to the original $v_g$ reference trajectory. As the parameter estimates converge, the calculated growth rate gets much closer to its true value. The reference trajectory for the melt temperature used by the heater MPC can be calculated based on the relationship between the $v_g$ and $\phi_s$ trajectories, as expressed in Eq. (5). After each completed run, the reference trajectory for the $T_s$ is recalculated using the latest smoothed estimate of the $\phi_s$

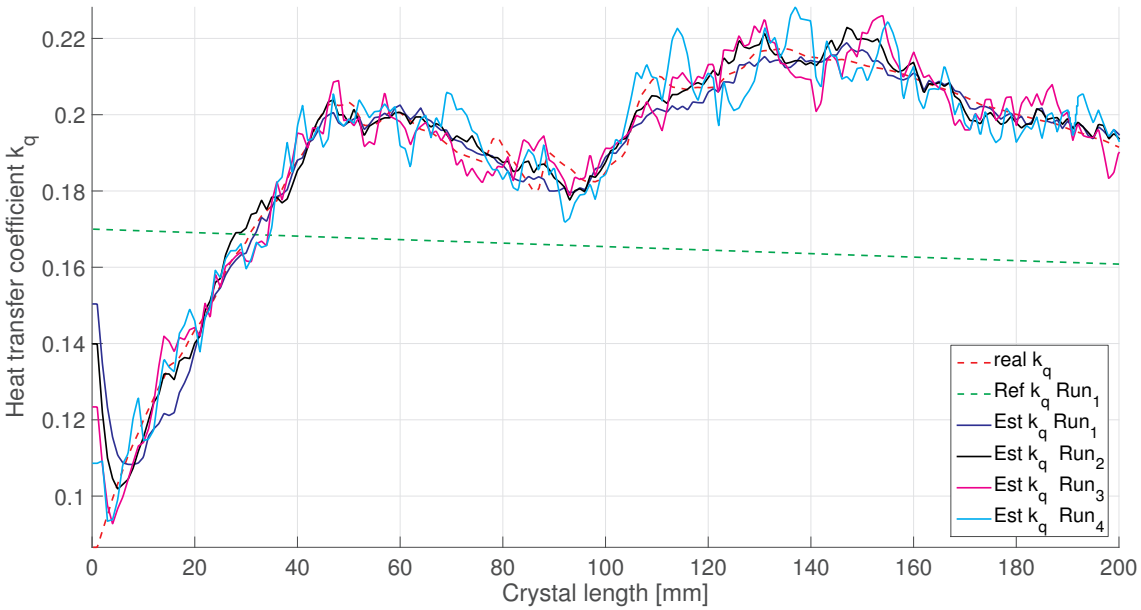


Figure 8: Estimated $k_q$ based on run 1 (dark blue) when applied initial $k_q$ was the simple graph in green. $k_q$ from run 2 (black) was estimated based on information from previous run. Run 3 (pink) and run 4 (cyan) were conducted in a similar to show that the estimations will converge. The red graph shows the real calculated $k_q$ for this specific crystal growth.

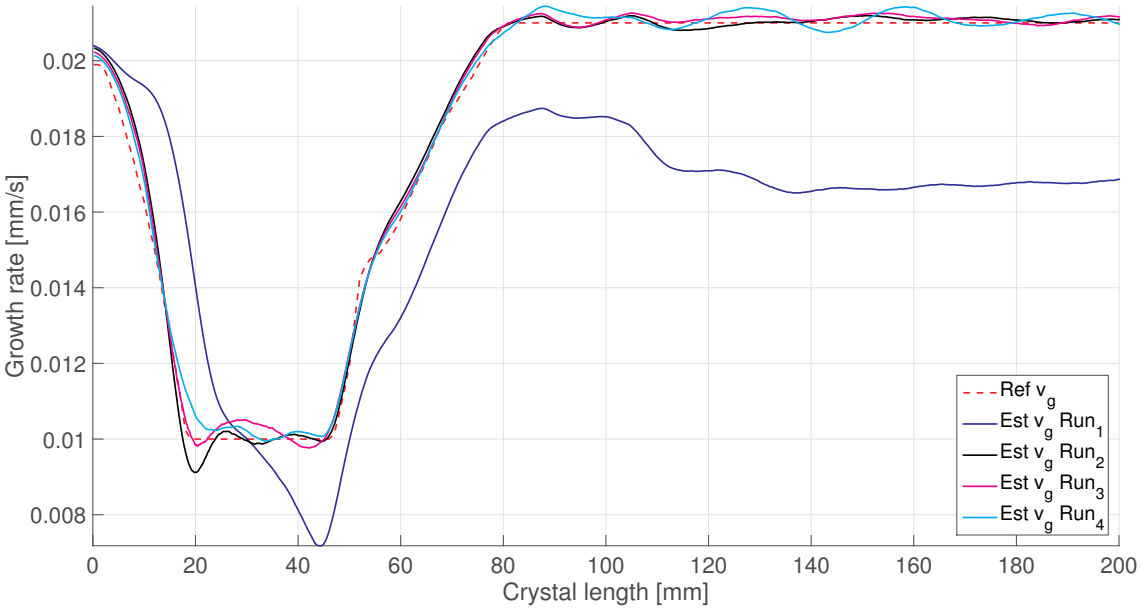


Figure 9: Estimation of growth rate during run 1 (dark blue), run 2 (black), run 3 (pink), and run 4 (cyan) compared against growth rate reference (red).

and by inserting Eq. (6) into Eq. (5) and reformulating it as

$$T_{\mathrm{s}}^{ref} = T_M + \frac{1}{k_m}(\hat{\phi}_{\mathrm{s}} - \rho_{\mathrm{s}}\Delta H v_{\mathrm{g}}^{ref}) \tag{18}$$

where $v_g^{ref}$ is the specified growth rate trajectory. Figure 10 illustrates what kind of melt temperature reference is appropriate for achieving better track of the growth rate reference. In addition, as the estimate of the $k_q$ gets closer to its correct value, it becomes easier for the heater MPC to follow the $T_s$ reference trajectory tightly. As the figures in the current

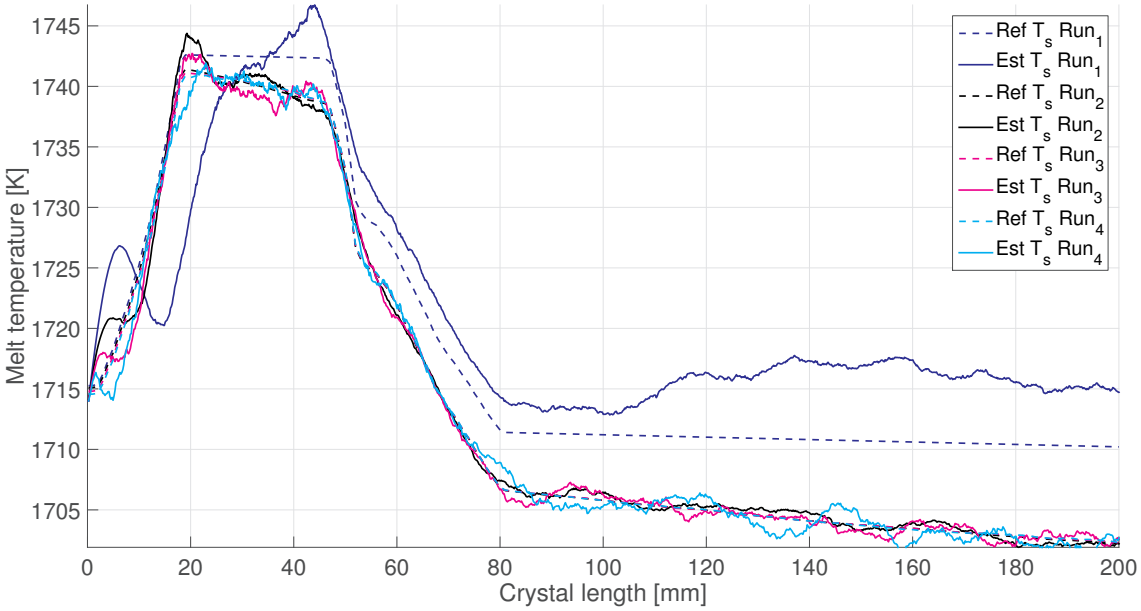


Figure 10: Estimated melt temperature during run 1 (dark blue), run 2 (black), run 3 (pink), and run 4 (cyan) with their respective reference trajectories.

section show, the MPC controllers can easily follow the predefined trajectories. Since the trajectories are designed by experienced metallurgists, an acceptable trajectory tracking should most probably result in crystals with less errors. The simulations also illustrate that the heat balance parameters can be estimated by the extended Kalman filter, and even further improved by a smoothing procedure for run to run control. By application of the suggested R2R control strategy, model adaptation is improved, which should enhance the controllers performance as well. This would lead to further reduction of the cut-offs from the produced crystals used in wafer production (see Figure 11). In addition, the improved trajectory tracking should also help maintaining the required crystal qualities for solar cell productions.

## 4. DISCUSSION AND CONCLUSIONS

The goal of this ongoing project has been to develop a method for control and monitoring of mono-crystalline Si-production, based on the Czochralski technique. As mentioned earlier,

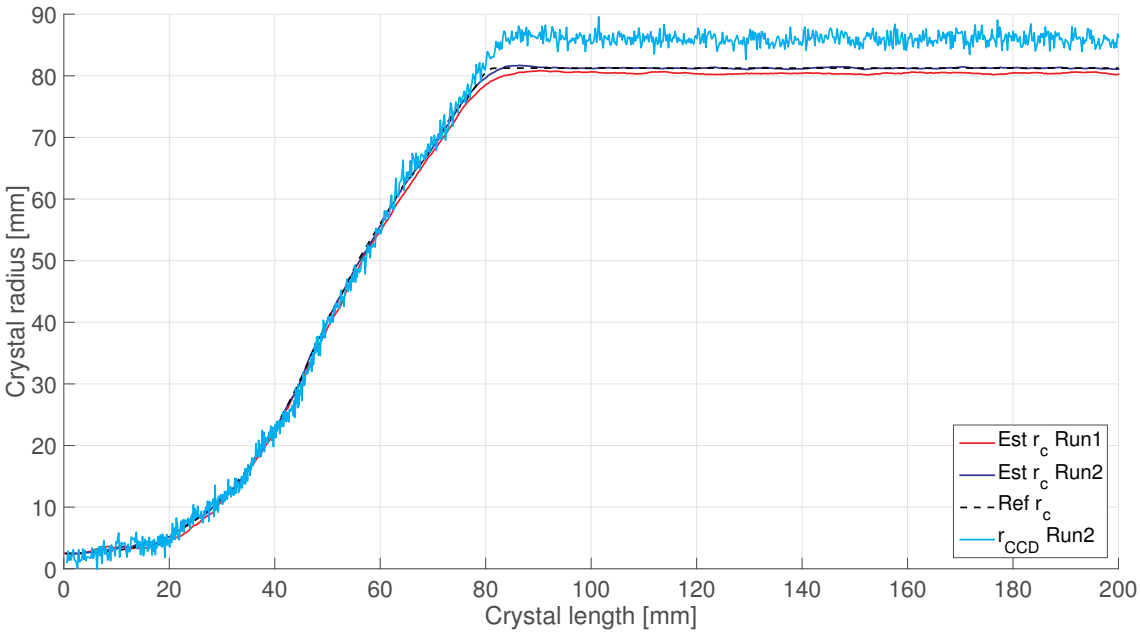


Figure 11: Simulation of grown crystal up to a length of 20cm where $r_{CCD}$ (cyan), $r_{estimated}$ from the first run (red), $r_{estimated}$ from the second run (blue) and $r_{reference}$ (black) are compared. Radius measurement with CCD camera introduces both bias and noise. With this knowledge, the estimator removes the bias and filters the noise.

the grown mono-crystals will later on be cut into wafers to be the most important part of the PV technology, which convert solar energy into electric energy. An acceptable approach in the Czochralski technique is essential to guarantee high quality product and relatively fast production rate. This means a Czochralski process producing monocrystals with specific quality and a profitable speed of fabrication. Pulling a single monocrystal Si using the Czochralski method can take up to 40 hours. Thus, there is significant economic gain in faster production with better or similar quality.

Control of the Czochralski-puller is challenging since the process dynamics are complicated and on-line measurements are scarce. This is due to having multi-variable dynamics where the operational parameters of the process change over time. In addition, the equipment used in the process can easily introduce noise to the system, affecting the quality of the crystal.

During this study, an advanced model-based control scheme has been proposed, using two Nonlinear Model Predictive Controllers (NMPC) coupled through the melt temperature predictions. This strategy has been chosen due to the existence of both fast (hydromechanial-geometrical part) and slow (bulk part) dynamics in the model. Thus, it becomes possible to choose different tuning parameters for each of the controllers based on their model dynamics, and to use different sampling rates and prediction horizons in the two MPCs. The advantage of the proposed set-up over the similar existing approaches, lies in the capability of each NMPC controllers' internal model to update itself on-line (during a run) and off-line (between

runs) based on measurements. For this purpose, the necessary states and parameters have been fed back to the controllers using nonlinear state and parameter estimation, through application of Extended Kalman filter. The combination of the advanced control structure and the nonlinear estimation technique has enhanced the model adaptation, thus reducing the model uncertainties. The results from the simulations illustrate that the Extended Kalman filter does a great job in filtering the measurement noise, while estimating the unmeasured states.

Commercial Czochralski pullers using the conventional control scheme depend to a large extent on the feed-forward melt temperature reference trajectory, in order to maintain the desired pulling rate. It is a common understanding among experts in this field that crystal quality depends strongly on the melt temperature trajectory. In many cases, the melt temperature reference trajectory is manually adjusted by empirical methods. Different operators and their different understanding of the process thereby increase the variability of the resulting crystal quality. Thus, an iterative method, based on the above control scheme, for calculating a new and improved trajectory from a growth run to the next has been explored. The method uses the results of the previous growth run in combination with an underlying model which incorporates the complex dynamic effect of the heater temperature on the pulling rate.

Combining MPC, estimation and Run-To-Run control has enabled robust, effective control of the Czochralski crystallization process. Using this approach, the traditional feed-forward control of the growth rate is replaced with a more direct control of the melt temperature. Consequently, for each completed run, the growth rate reference tracking becomes tighter, which results in faster and better control of the crystal diameter. Comparing the presented control structure to the traditional control strategies applied for the Czochralski process, the pulling velocity would not require a reference trajectory, which gives the puller MPC more freedom to focus only on the radius trajectory. In addition, the effect of the typical wear and tear that exists in every Czochralski puller, could be reduced through the Run-To-Run control methodology. The simulations also show that the model-based controllers applied here, can follow the predefined reference trajectories fairly well, while respecting the physical and practical constraints on the manipulators, as well as the states. This evaluation has been found to hold, even in cases where there has been considerable model mismatch between the model-based controllers (and the estimators), and the plant model. In an industrial application, it is also possible to use the conventional control approach in combination with the presented EKFs in a few runs until the parameter estimates have converged, before applying the suggested model-based control structure.

The next goal would be implementing and testing the Run-To-Run control strategy on a real application.

**Acknowledgement**

This work was supported by the Research Council of Norway and Prediktor AS.

## References


Bardsley, Cockayne, Green, Hurle, Joyce, Roslington, Tufton, Webber, 1974a. Developments in the weighing method of automatic crystal pulling. Journal of Crystal Growth 24-25, 369–373.

Bardsley, Frank, Green, Hurle, 1974b. The meniscus in czochralski growth. Journal of Crystal Growth 23, 341–344.

Bardsley, Green, Holliday, Hurle, 1972. Automatic control of czochralski crystal growth. Journal of Crystal Growth 16, 277–279.

Bardsley, Hurle, Joyce, 1977. The weighing method of automatic czochralski crystal growth: Basic theory. Journal of Crystal Growth 40, 13–20.

den Bogaert, V., Dupret, 1997a. Dynamical global simulation of the czochralski process, i. principles of the method. Journal of Crystal Growth 171, 65–76.

den Bogaert, V., Dupret, 1997b. Dynamical global simulation of the czochralski process, ii. analysis of the growth of a germanium crystal. Journal of Crystal Growth 171, 77–93.

Bones, J., Haugen, K., 2012. Model-based Control of the Czochralski Process. Master's thesis. Norwegian University of Science and Technology.

Brown, R., Hwang, P., 1997. Introduction to Random Signals and Applied Kalman Filtering. Third ed., John Wiley & Sons.

Crowley, A., 1983. Mathematical Modelling of Heat Flow in Czochralski Crystal Pulling. IMA Journal of Applied Mathematics 30, 173–189. doi:10.1093/imamat/30.2.173.

Dornberger, von Ammon, den Bogaert, V., Dupret, 1996. Transient computer simulation of a cz crystal growth process. Journal of Crystal Growth 166, 452–457.

Duffar, T., 2010. Crystal Growth Processes Based on Capillarity: Czochralski, floating zone, shaping and techniques. John Wiley & Sons, Chippenham, Wiltshire, GB.

Gevelber, M., Stephanopoulos, G., 1987. Dynamics and control of the Czochralski process. Journal of Crystal Growth 84, 647–668. doi:10.1016/0022-0248(87)90056-X.

Gevelber, M., Wargo, M., Stephanopoulos, G., 1987. Advanced control design considerations for the Czochralski process. Journal of Crystal Growth 85, 256–263. doi:10.1016/0022-0248(87)90232-6.

Hurle, Joyce, , Wilson, Ghassempoory, Morgan, 1986. A technique for experimentally determining the transfer function of a czochralski pulling process. Journal of Crystal Growth 74, 480–490.

Hurle, D., 1993. Crystal Pulling from the Melt. Springer-Verlag, Berlin, Germany.

Hurle, D., Joyce, G., Ghassempoory, M., Crowley, A., Stern, E., 1989. The dynamics of Czochralski Growth. Journal of Crystal Growth 100, 11–25.

Hurle, D., Joyce, G., Ghassempoory, M., Crowley, A., Stern, E., 1990. The dynamics of Czochralski growth. Journal of Crystal Growth 100, 11–25. doi:10.1016/0022-0248(90)90603-I.

J.M.Maciejowski, 2000. Predictive Control with Constraints. Prentice Hall.

Johansen, T., 1991. On the inherent stability of the czochralski crystal growth process. Journal of Crystal Growth 114, 27–30. doi:10.1016/0022-0248(91)90674-T.

Park, P., Kailath, T., 1996. New square-root smoothing algorithms. IEEE Transactions on Automatic Control 41, 727–732.

Rahmanpour, P., Bones, J.A., Hovd, M., Gravdahl, J.T., 2013. Linear and nonlinear state estimation in the Czochralski Process 10th IFAC Symposium on Dynamics and Control of Process Systems, Mumbai, India.

Rahmanpour, P., Hovd, M., Bones, J.A., 2014. Nonlinear state estimation in the Czochralski Process 19th IFAC World Congress, Cape Town, South Africa.

Rahmanpour, P., Sælid, S., Hovd, M., Grønning, O., Jomaa, M., 2016. Nonlinear model predictive control of the Czochralski Process 17th IFAC Symposium on Control, Optimization and Automation in Mining, Mineral and Metal Processing, Vienna, Austria.

Rauch, H., Tung, F., Striebel, C., 1965. Maximum likelihood estimates of linear dynamic systems. AIAA Journal 3, 1445–1465.

Satunkin, G., 1995. Mathematical modelling and control system design of czochralski and liquid encapsulated czochralski processes. Journal of Crystal Growth 154, 172–188.

Satunkin, G., Rossolenko, S., 1986. Analysis of the dynamics of the controlled crystallization process using the czochralski method. Crystal Research and Technology 21, 1125–1138.

Tatarchenko, Y., 1993. Shaped Crystal Growth. Kluwer Academic Publishers, Dordrecht, The Netherlands.

Wang, Y., Gao, F., Doyle, F., 2009. Survey on iterative learning control, repetitive control, and run-to-run control. Journal of Process Control 19, 1589–1600.

Winkler, J., Neubert, M., Rudolph, J., 2010. Nonlinear model-based control of the Czochralski process I: Motivation, modeling and feedback controller design. Journal of Crystal Growth 312, 1005–1018.