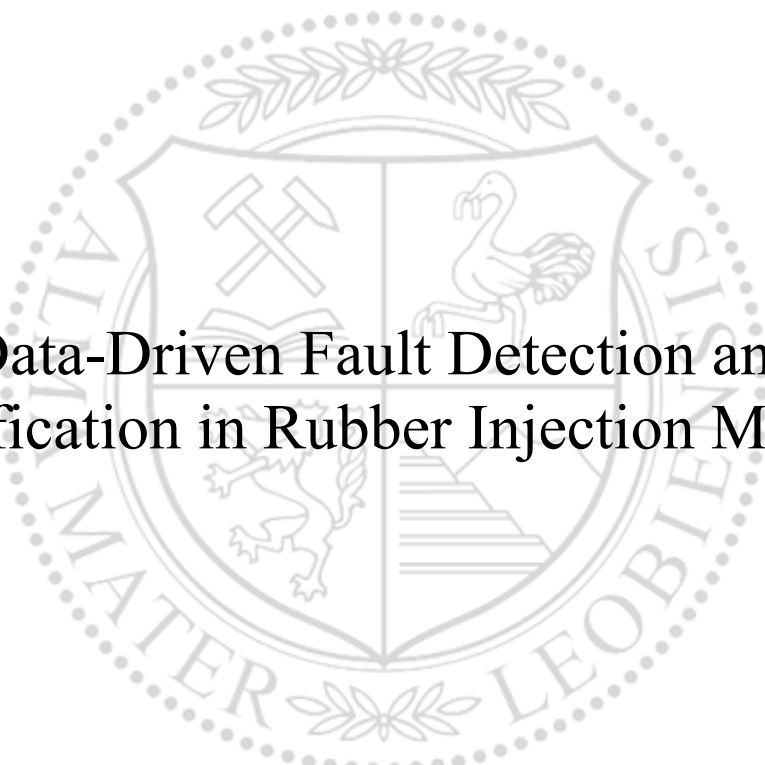Chair of Injection Moulding of Polymers

Doctoral Thesis

Data-Driven Fault Detection and Identification in Rubber Injection Molding

Dipl.-Ing. Thomas Hutterer, BSc

September 2020

Data-Driven Fault Detection and Identification in Rubber Injection Molding

PhD Thesis

by

DI Thomas Hutterer

at

Montanuniversitaet Leoben

Supervision: Univ.-Prof. Dipl.-Ing. Dr.mont. Walter Friesenbichler

Mentoring: O. Univ.-Prof. Dipl.-Ing. Dr.techn. Paul O'Leary

Leoben, September 2020

In a dark place
we find ourselves,
and a little more knowledge
lights our way.

YODA

Acknowledgement

Then, I want to thank my supervisor Univ. Prof. Dipl.-Ing. Dr.mont Walter Friesenbichler for enabling this thesis and always supporting me from the start to the finishing touches. Also, I'd like to thank O. Univ.-Prof. Dipl.-Ing. Dr.techn. Paul O'Leary for mentoring me during my time as a PhD student and pushing me in the right direction. I also want to thank Assoz.Prof. Dipl.-Ing. Dr.mont. Gerald R. Berger-Weber for always being open for discussion and helping me improve the things I was working on.

Furthermore, I want to mention DI Dr.mont. Roman Kerschbaumer and Eduard Leitner, on whose expertise I could always rely on. Without it some things would not have worked out the way they did in the end.

I also want to thank my parents for never stopping to support me and who enabled me being in this place now in the first place. Additionally, I want to thank my brother Andreas and his wife Katie, who drastically improved the writing of this work.

Finally I want to thank Anja for always being there.

Abstract

I N injection molding, the properties of rubber compounds, and the manu-
facturing conditions themselves are subject to fluctuations, which impact
the quality of the finished rubber parts negatively. To detect such fluctua-
tions and to keep part quality constant, process monitoring systems are used
in a wide range of industrial applications. For this purpose, a data-driven
statistical monitoring system for rubber injection molding needs to be devel-
oped. By employing multivariate statistics, faulty parts can be detected and,
once a fault database is present, fault types can be identified. To develop the
system, first a dynamic compression testing methodology is presented, which
is able to determine the relevant dynamic quality parameters of rubber parts
in a way that is fast enough for on-line implementation and $100\,\%$ quality
control. Second, the temperature of the rubber, the most important factor
influencing the final part quality, is determined at every stage of the rubber
injection molding process. By employing ultrasound and thermography, the
rubber temperature resulting from the processing conditions can be modeled.
On this base, the data-driven process monitoring system is built.

This process monitoring system detects fluctuations causing faults by
using a Principal Component Analysis (PCA) based approach. As a result,
all available process signals can be evaluated simultaneously, even linear
dependent ones. Additionally, by Fisher Discriminant Analysis (FDA), the
type of fault can be automatically identified. Compared to other methods
of process monitoring available for rubber injection molding, the presented
data-driven system eliminates the need for any preliminary material tests,
modelling or data selection. It is therefore much more straightforward and
cost-efficient in implementing in smart manufacturing facilities.

Kurzfassung

BEIM Kautschukspritzgießen können Schwankungen der Verarbeitungseigenschaften und -bedingungen auftreten, welche die Qualität der hergestellten Bauteile negativ beeinflussen. Um solche Fluktuationen zu erkennen, und Gegenmaßahmen treffen zu können, werden in vielen industriellen Anwendungen Prozessmonitoringsysteme eingesetzt. Deswegen wird in dieser Arbeit ein datenbasiertes System zur statistischen Prozessüberwachung des Kautschukspritzgießprozesses entwickelt. Durch den Einsatz multivariater Statistik können fehlerhafte Bauteile detektiert, und die Art des Fehlers kann automatisch identifiziert werden. Zur Entwicklung des Systems wird zuerst eine dynamische Testmethode vorgestellt, welche die relevanten dynamischen Parameter eines Elastomerbauteils schnell genug bestimmen kann, um für den on-line Einsatz geeignet zu sein. Weiters wird die Temperatur des Kautschuks im Verarbeitungsprozesses bestimmt, da diese den wichtigste Faktor für die finale Bauteilqualität darstellt. Durch den Einsatz von Ultraschall und Thermographie konnten Modelle für die Abhängigkeit der Kautschuktemperatur von den Verarbeitungsbedingungen erstellt werden. Diese Messungen bilden die Basis für die Entwicklung des datenbasierten Monitoringsystems. Dieses erkennt Fluktuationen, welche Fehler verursachen, durch den Einsatz von Hauptkomponentenanalyse (PCA). Dadurch können alle verfügbaren Prozesssignale gemeinsam zur Überwachung verwendet werden. Zusätzlich kann das Monitoringsystem mit Hilfe von Fisher Discriminant Analysis (FDA) Fehlerarten automatisch identifizieren. Das in dieser Arbeit entwickelte Prozessmonitoringsystem verlangt keine Vorabuntersuchungen, Modellierung oder Datenverarbeitung. Es ist dadurch einfacher und kosteneffizienter in Industrie 4.0 Fertigungsanlagen zu integrieren als andere für das Kautschukspritzgießen verfügbare Systeme.

Work published by the author

Thomas Hutterer et al. "Rubber injection molding: Applying PCA and FDA to identify quality issues solely from process signals". In: *Polymer Engineering and Science (submitted)* (2021).

Thomas Hutterer et al. "Online detection of storage-induced changes of rubber properties in the injection molding process". In: *Proceedings of the 34th Conference of the Polymer Processing Society.*

Thomas Hutterer, Gerald Roman Berger-Weber, and Walter Friesenbichler. "Multivariate statistical process monitoring in rubber injection molding by employing principal component analysis". In: *Proceedings of the IRC 2019.*

Thomas Hutterer, Gerald Roman Berger–Weber, and Walter Friesenbichler. "Determination of the temperature after dosing an industrial rubber compound. A new ultrasonic-based method". In: *Proceedings of DKT 2018.*

Thomas Hutterer et al. "Simulative and experimental investigation of rapid heat cycle molding for rubbers". In: *AIP Proceedings* (2018). DOI: 10. 1063/1.5084877.

Thomas Hutterer, Gerald Roman Berger-Weber, and Walter Friesenbichler. "Fault detection in rubber injection molding with multivariate statistics". In: *Rubber World* (2019), pp. 40–44.

Contents

1 Introduction and current state of technology

I N this work, a data-driven statistical process monitoring system for rubber injection molding will be developed. It will enable automatic fault detection and identification based on processing data readily available on modern rubber injection molding equipment. Thus, abnormal process fluctuations are detected on-line and machine operators are advised on the possible causes of the deviation. While a large body of work can be found on data-driven approaches for monitoring the thermoplastic injection molding process, and some methods are already implemented into the operating systems of modern injection molding machines, no statistical process monitoring approach has been reported on rubber injection molding in literature outside of what has been done in this work.

Rubber goods are found in a wide range of consumer and industrial applications, whenever excellent damping, sealing, and friction capabilities are needed for a system's intended operation. A large portion of those are manufactured by injection molding, as this technology is most economically sensible when manufacturing lots exceed 10000 parts [1]. Due to global competition, market saturation effects and ever-increasing performance requirements, part manufacturers are forced to increase the complexity of their products by -for example- optimizing part weight or increasing function integration [2–4]. Consequentially, the complexity of the manufacturing processes has to rise as well, which can only be done by increasing automation. In conjunction, costs need to be cut, and to do so, reducing the scrap rate is a very effective measure, because it brings ecological advantages with it as well.

To meet these customer demands, manufacturing equipment suppliers constantly increased the capabilities of their injection molding machines to

levels, where repetition accuracy of temperatures and machine movements is not an issue any longer. However, even perfectly operating injection molding equipment can not prevent process disturbances from outside sources, especially property variations of the raw material. Such variations become ever more critical as the limits of the processes and parts are pushed further. Since modern injection molding equipment is able to collect massive amounts of processing data, and fitting additional sensors to the mold has become more common, process monitoring systems are of major interest both to academia and industry [5–10]. The prime goal of all monitoring systems is to reliably detect process fluctuations and adapt the process accordingly without human intervention to keep the part quality within specification boundaries at all times. While there have been significant successes for this approach in thermoplastic injection molding (which will be discussed in section 1.3), little has been done for controlling or even monitoring the rubber injection molding process [11, 12]. This is surprising, as rubber compounds are complex mixtures of physically and chemically active ingredients, which only increases their susceptibility to exhibit variations of their properties batch-to-batch or even within-batch.

Machine learning methods will be applied in this work to develop a process monitoring system for rubber injection molding, which detects critical material property variations without the need for any prior testing. Furthermore, classification methods will be applied to predict the type of fault, when certain types occur repeatedly.

To start, after revisiting the basic principles of process monitoring and methods for generating knowledge from data, a current state of technology on process monitoring in injection molding both for thermoplastic materials and rubber is given.

1.1 Principles of process control

Modern devices are able to capture far more data about their current state than can ever be translated into information. Suppliers of manufacturing

equipment have implemented full electronic closed-loop control of every feature of their machines to meet the ever-increasing demands of their customers in process and part quality [13]. To enable closed-loop control, every modern tempering unit, handling robot, or injection molding machine has built-in sensors providing real-time actual values to the operating system about every actuator position, cooler temperature, or motor power consumption. The aim of this work is to gather relevant information from these data streams and establish a process control, monitoring and fault identification system for rubber injection molding that is able to recognize faulty rubber parts before they exit the mold and provide insight into the type of fault that occured. Many of the basic principles of quality control systems that accompany modern manufacturing processes have been laid down by Shewhart in his book "Economic control of quality of manufactured product", which was originally published in 1931 and has been re-issued ever since [14]. There, he defines the aspect of quality in a broader sense than is common in more recent literature, where quality is often considered the goodness of a thing [15]. To Shewhart, quality is a set of measurable attributes which make up an object and also clearly distinguish it from other, similar objects. When objects (products) are made repeatedly, not one of these objects is completely identical to any other, thus the quality of a process output (product) is subject to variability. Variability is inherent to every process, natural or artificial, because a process is always the results of many contributing factors which do not possess a singular state but rather follow some probability density function. Thus, the magnitude of quality attributes of a product is also a statistical distribution and not a fixed value [16]. The aim of any process control system is to be able to measure the current value of the statistical contributors and, based on past observations, infer the quality of the product manufactured under the measured conditions. To design process monitoring systems effectively, two types of variablity need to be identified [14, 15]:

1. **Variablity from Assignable Causes** are disturbances introduced by sources which can be named, and thus measures to counter vari-

ability caused by them have to be set. Such assignable causes may be non-stationary process temperatures, wear, or -specific to rubber injection molding- batch-to-batch variations. In best practice, possible assignable causes should be considered from the earliest conception phase of the product and its related processes, as design flaws lead to not being able to eliminate these causes of variability in the production phase. In the second half of the 20th century, the mindset of a more complete approach to quality engineering was made popular by Taguchi [17]. He also popularized the application of Design of Experiments (DoE) outside of academia, even though this methodology was already around since the beginning of the century. By DoE methodology, the response of the product quality to changes of the process can be systematically investigated. By evaluating a DoE, a robust process can be set up, of which variation lies well within specification limits and inherent variability does not cause the product to be outside specification immediately.

2. **Inherent Variability** incorporates the variations when all adjustable factors are set to their desired values, equipment is working as expected, and product fluctuations are well within confidence limits. Hence, the process is in control. The true sources of Inherent Variability can not be determined easily, and variations of the measurement systems used to control the process contribute to perceived product quality fluctuations. It is in this scenario, where process monitoring and control come into effect. By Statistical Process Monitoring (SPM), trends and shifts which affect the process capability negatively are detected before the process is outside the specification limits. With the fault detection and identification methods used in this work, such events can be analysed and assignable causes can be determined. By taking countermeasures, the process capability can then be increased again. Inherent variability specific to rubber injection molding is for example, small differences in storage maturation or control-based mold

temperature fluctuations.

With ever-increasing complexity, harsher application conditions and environmental regulations, quality standards of technical parts are also steadily increasing. In conjunction with rising requirements for part performance, manufacturing equipment was fitted with more sophisticated mechanical components but also with far more capable electronic control systems [7, 15, 18]. These systems also drastically increased the amount of data available for monitoring in modern manufacturing equipment, considering digital controllers need measurements of actual values of every variable that has to be controlled [7, 19, 20]. For about two decades, much work has been done to use injection molding machine and sensor data to transfer quality fluctuations from Inherent Variability into the realm of Variability from Assignable Causes. When additional sources of error can be eliminated and the range of process variability can be reduced further, more complex parts can be manufactured and higher part performance can be expected from those parts. [7, 8, 18, 19, 21].

1.2 Gathering knowledge from data

Artificial intelligence (AI) is the megatrend of the 21$^{\text{st}}$ century, so it has found applications in almost every aspect of modern life such as finance, security, entertainment or manufacturing [19, 22, 23]. For process control purposes, it has to be understood that AI in itself is only one method that can be used in the process of Knowledge Discovery in Databases (KDD). According to Fayyad, KDD is a "nontrivial process of indentifying valid, novel, potentially useful and ultimately understandable patterns in data" [19, 21, 24].

As shown in Figure 1.1, first a database has to be created, for example by data collection from external sources, or as in this work, by conducting extensive experiments. In the experiments, factor level settings should cover a setpoint range as wide as possible, to ensure data are generated in which possible patterns are represented fully. Only if this is true, they can

later be identified by the KDD. Once the database exists, data which are promising in their information content need to be selected. For industrial applications, the first two steps already require high level knowledge of the experimental process, since experiments need to be conducted in a way that creates useful data. When the experiments are designed improperly, they may obscure patterns, and the KDD has to be restarted [24]. Furthermore, data have to be selected with consideration of the investigated phenomena, and ideally, only data from independent variables with high information content are retained. The selected data then need to be preprocessed, which is facilitated by extensive computer libraries that offer automated routines for many important preprocessing steps, such as capturing outliers, cleaning missing data, and centering and scaling [25].
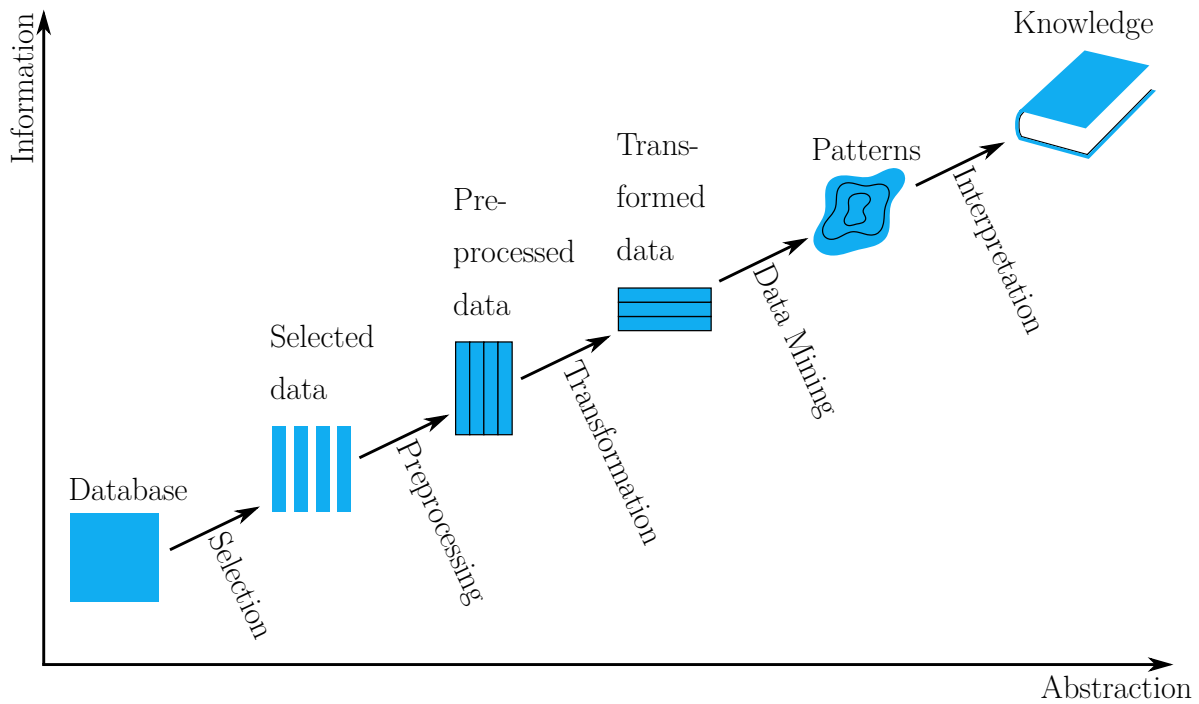
Figure 1.1: The Knowledge Discovery in Databases (KDD) process is a multi step procedure aimed to generating an understanding of underlying patterns of data in large data bases [19]

Literature on the KDD is consistent in treating the subsequent steps of transformation, and data mining as separate steps [19, 21, 24]. However,

transformation as performed by some auto-encoder methods, for example PCA, can already extract the basic patterns in the data, not necessitating additional data mining methods [26]. Today, from the preprocessed data onward, much of the KDD process is performed by artificial intelligence methods, and, especially considering deep learning methods, patterns may remain inconceivable to humans [22]. If knowledge is even discovered by humans in such situations is up to debate. Whenever the KDD process should be employed in industrial environments, deep learning can be non-viable to many manufacturers, because emerging patterns and their interpretation is not accessible to them. Thus, they are not able to deduct process improvement steps when faults occur, even worse, have any direct control on the decision logic of the process monitoring system.

1.3 Process control in injection molding

For industrial applications, specific knowledge is of great importance to draw correct conclusions from the KDD process and thus, machine learning methods other than AI are commonly more popular. These methods can present emerging patterns in a way humans can more easily understand and check for their validity [27–34]. To model any given process, three basic approaches are commonly mentioned in literature [7]:

1. First principle model based methods

2. Knowledge-based methods

3. Data-driven methods

The main difference between first principle model/knowledge-based methods and data-driven methods are the in- and outputs of the systems. In first principle model based methods, rules about the process' behavior are established upon fundamental physical principles of the process components. The most prominent application of such methods in injection molding is injection molding simulation software [35–37]. There, the laws of thermodynamics

Rules

Data

First principle model/
knowledge-based
monitoring method

Answers

Answers

Data

Data-driven monitoring
method

Rules
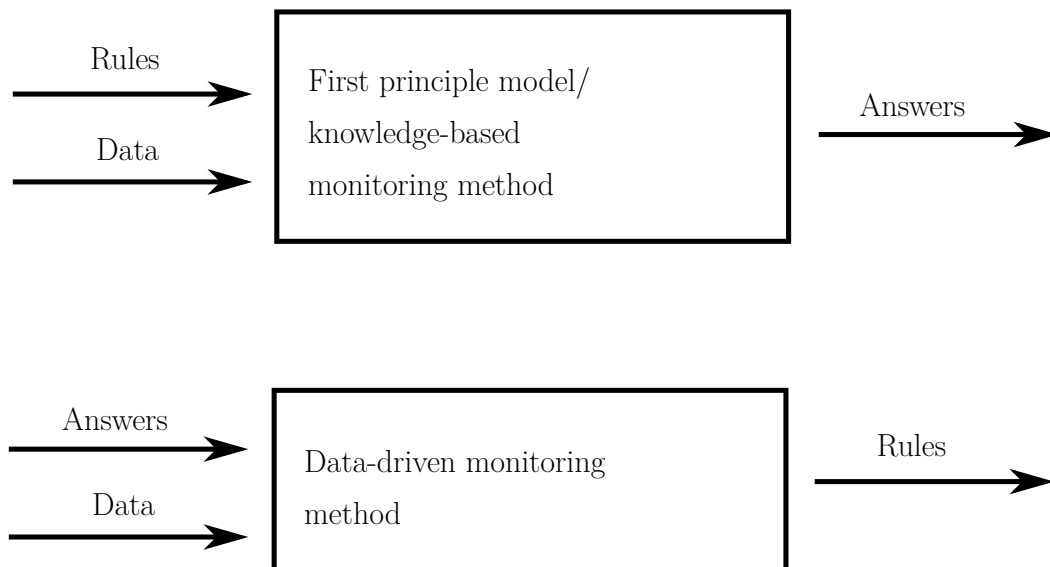
Figure 1.2: Model/knowledge-based monitoring methods differ from data-driven ones mainly on what their inputs and outputs are [22]

and motion (chapter 2) are used to predict, for example the filling pattern, energy consumption, and reaction kinetics during injection molding of plastic and rubber parts. To work properly, injection molding simulation needs precise and accurate data on the thermodynamic and rheologic behavior of the plastic or rubber of which filling should be modeled. Much effort has been made by academia and industry to measure the true material behavior in experiments, and success is still not guaranteed, especially for industrial rubber compounds [38–42].

Monitoring the injection molding process of rubber based on first-principal models is done very rarely, and only the works of Berkemeier et al., can be found [11, 12]. This research culminated in the commercially available Jidoka PM101 III (CAS GmbH & Co KG, Reinbek, Germany) process

monitoring system. With this system, Berkemeier et al. calculate the full energy input into a volume of rubber as it enters the injection molding machine until it is molded into a part. Their calculations are of course not dissimilar to the methods of injection molding software, although they also try to incorporate the dosing process and dwell time of the rubber [43].

Similar to injection molding simulation software, issues arise with the determination of correct material parameters. To adjust for batch-to-batch variations and differences in storage maturations, rheological and chemical parameters need to be measured for every batch, which can be a factor of cost and time that is able to outweigh any benefits of a quality-focus process control system. Furthermore, measurement devices of material parameters in themselves rely on samples taken from the batches, reducing the effectiveness of the approach to adapt to within-batch variations of material properties.

Knowledge-based approaches conceptually work in the same way as model-based ones, but differ in the origin of the rules they are based on. Knowledge-based approaches empirically derive rules from operator experiments and experimental setups, which employ industrial manufacturing equipment. Of course, these empirical rules are rooted in the true physical behavior of the manufacturing components and raw materials, but describe the phenomena they evoke in the respective application. Such models are most prominently used in process monitoring and control of injection molding, and especially pressure demand during filling has been investigated extensively. The following methods also rely on training data to monitor the injection molding process, but the exact models - while not disclosed publicly - seem to rely on empirical relationships found in extensive experiments by the respective suppliers, thus they are considered knowledge-based [7, 20].

As a primary supplier of in-mold pressure sensors, Kistler Group (Winterthur, Switzerland) offers process control and monitoring software for injection molding in conjunction with their sensors. Of interest for this work is their ComoNeoPREDICT package. The software assists operators in creating the training data by proposing an appropriate DoE. After measuring the quality-specific parameters of the part, the software can be used

to link injection pressure curves and quality. After this modelling step is completed, live in-mold pressure curves can be analysed on-line to predict the part quality and monitor the injection molding process [44].

ENGEL Austria GmbH (Schwertberg, Austria) offers a wide range of software tools to be used in connected manufacturing environment. The software solution iQ weight control, for example, measures the pressure signal at the mount of the screw during injection [45]. For iQ weight control to work as intendend, training data need to be generated, but, in contrast to Kistler's solution, they do not need to be generated by a DoE. Rather, the part weight of twenty shots during production has to be measured for model training. Subsequently, the system can not only predict the part quality based on the actual injection signal, but also take actions to keep part weight constant as it detects deviations in the early stages of the injection phase. The system distinguishes between three fault types: (1) A time shift of the actual injection pressure signal compared to the training data, (2) relative variations in slope of the injection pressure and (3) shape deviations of the injection pressure signal. When cases (1) or (2) are detected, the switch point between volumetric and pressure controlled filling is adapted in a way that ensures a constant filling of the cavity at the switch. If (3) is detected, a warning is sent to the operator that a fault of unspecified causes occurred [20]. The iQ clamp control system is another process monitoring system available for ENGEL injection molding machines [46]. It is able to measure how much the mold, which is compressed by the clamping force, relaxes during injection due to the buoyancy of the part. This phenomenon is called mold breathing and is a function of the strength of the mold, the part geometry, and most importantly the pressure in the mold. While by design, the iQ clamp control system is intended for process optimization by assisting in setting the optimal clamping force, it can also be used for process monitoring since it also provides information on the pressure in the cavity. In section 6.3, the iQ clamp control system will be incorporated into the multivariate statistical process monitoring system.

Besides these, there are many commercially availabe solutions for moni-

toring the injection molding process of thermoplastic parts based on process signals[2, 9, 47]. However, most solutions not specifically mentioned here do not provide any built-in modelling or fault detection besides a reference curve for the monitored signal. Thus, their monitoring capabilities are limited to the knowledge and experience of the personnel operating the machine or overseeing the plant. Furthermore, all commercially available systems only provide univariate monitoring. While the information in the signals these systems track for monitoring (most of them track the injection pressure) is also high, the probability of missing faults by omitting other information is high. This is especially surprising given that every major manufacturer of injection molding machines offers database solutions for storing process data cycle-by-cycle from every component added to the manufacturing cell.

In academic research, data-driven approaches have risen to the spotlight. A very high number of papers are published on the possibilities of such monitoring methods for a wide range of manufacturing processes [48–52]. Due to the peculiarities of the injection molding process mentioned in section 2.2, however, many of them cannot be directly transferred. A short overview of approaches specific to injection molding is given instead.

Chen et al., propose a system of three pressure transducers mounted at the injection nozzle, the runner and the cavity to calculate relative viscosity changes during molding experiments. Placing multiple sensors downstream in the flow path augments monitoring by enabling a pressure drop calculation. Thus, their setup resembles a high-pressure capillary viscosimeter. From the pressure curves, they calculate features such as the integral or the peak of each sensors signal, and many of the features correlate well with changes in the barrel or mold temperature. However, they do not apply methods of multivariate statistics in their approach, even though linear independence of pressures measured along a flow path is highly unlikely [28].

Gordon et al. present a multivariate sensor, which is capable of measuring the pressure, temperature and velocity of thermoplastic melt passing by. They mount the sensor in the cavity alongside commercially available temperature and pressure sensors. From the sensor's signals, they also

extract manually defined characteristic features such as slopes, maxima and characteristic times. By applying multivariate methods, they are able to increase the sensor's signal correlation with quality parameters including mass and even tensile strength [53].

Zeaiter et al. also use multivariate modelling to predict the part quality from pressure and temperature data gathered in a injection molding DoE. They employ partial least squares, which enables them to model multiple linearly dependent quality indicators in their multivariate model. Similar to some process signals (e.g. injection and in-mold pressure), quality features can also show interdependencies. For example, sink marks are related to holding pressure efficacy, which also determines the part weight [29].

To sum up, these works show how additional signals and multivariate methods are powerful tools for process monitoring and quality prediction in injection molding. However, they do not apply their methods to monitor manufacturing lots where machine setpoints stay constant but only perform testing with data from the DoEs used for model building. To develop process monitoring systems, DoEs are necessary tools for model building, but they often vary factors which drastically change variable relationships, for instance altering the injection volume flow rate. Furthermore, the spacing between factor steps is commonly much higher than it is when fluctuations occur in continuous manufacturing. Control algorithms for closed-loop setpoint control of modern injection molding machines, such as model predictive controllers, are extremely powerful in controlling machine movements even when disturbances occur [7]. As a result, for the objectives of this work, which include primarily catching small-scale process fluctuations, it is not optimal to train a monitoring model by changing setpoints by high amounts. For instance, varying injection velocity would be counterproductive, because the machine will most likely be able to keep it constant even when the viscosity of the rubber is subject to variations. Also, whenever the material viscosity changes to a magnitude at which the machine motors are no longer able to reach their set points, complex monitoring systems are not needed for fault detection [53].

12

Still there are reports where the focus was set on monitoring process fluctuations with advanced methods: Woll et al., trained two artificial neural nets differing in design (feed-forward and back-propagating respectively) to detect if too much reground Polyamide, which can cause faulty parts, is fed into the injection unit. Most notably, they compare model performance of two different inputs: In one training run, they used discrete data features including, but not limited to maxima and integrals as input for the networks, and in another training run, they feed complete time-domain trajectories of the pressure during injection to the networks. It turned out that the networks trained on trajectories are far more capable of monitoring the injection molding process [30, 54].

The research group of Gao started in the early 2000s to publish work on injection molding process control and monitoring. At first, they presented a capacitor, which can be monunted in injection molds to monitor the filling process [55]. Furthermore, univariate data from this sensor type can be used to predict various quality parameters [7, 10]. Later, by employing multivariate methods, they published significant improvements in statistical process monitoring of the injection molding process. By employing recurrent neural nets and discrete feature data from standard machine sensor, the flow length could be estimated. Consecutively, they developed a method to apply PCA-based monitoring to injection molding [7, 32]. This method does not rely on features, which have to be calculated in advance and can lead to information loss, it is sensitive to the process characteristics of injection molding and needs few training data compared to most neural nets. Thus, in this work, the PCA monitoring approach presented in the book by Yang et al. will be transferred to rubber injection molding. However, while it is very capable for fault detection, fault identification is challenging with PCA based methods. To counter this, Fisher Discriminant Analysis (FDA) will be added to discern between different types of faults that can occur in the rubber injection molding process [7, 13].

1.4 Quality of injection molded rubber parts

Except the works of Berkemeier et al., none of the systems and methods discussed so far were developed for rubber injection molding, but rather for injection molding of thermoplastic parts. However, while the equipment needed for each of these classes of material is similar, the operating points -especially the thermal conditions- are very dissimilar. This is rooted in a basic and drastic difference between the two materials: The quality of thermoplastic parts is determined by physical processes, but the quality of rubber parts is determined by chemical reactions. It was shown in numerous works that the degree of cure (x_c) is by a large margin the most important factor of a rubber part's mechanical, physical and chemical properties [11, 40, 56, 57]. Thus, to keep the quality of injection molded rubber parts constant, it is paramount to keep the x_c constant for every manufactured part as it is removed from the mold. To reach a specific degree of conversion in endothermic chemical reactions, a specific amount of energy has to be brought into the reactive system. In rubber injection molding, the primary source of energy is thermal energy, which is transferred into the rubber either by dissipation or thermal conduction [1, 56, 58–60]. The x_c of an isothermal curing reaction is determined by a first order differential equation:

$$\frac{\mathrm{d}x_c}{\mathrm{d}t} = k_c(T)f(x_c), \tag{1.1}$$

where the kinetic constant of curing (k_c) is a function of temperature (T) and also specific to the compound ingredient fractions, and $f(x_c)$ is a compound specific function of the current degree of cure. For sulphur cured rubbers, an n-th order reaction function is common and the temperature dependence of the kinetic constant of curing is determined by an Arrhenius approach [56, 61, 62]:

$$f(x_c) = (1 - x_c)^{n_c}, \tag{1.2}$$
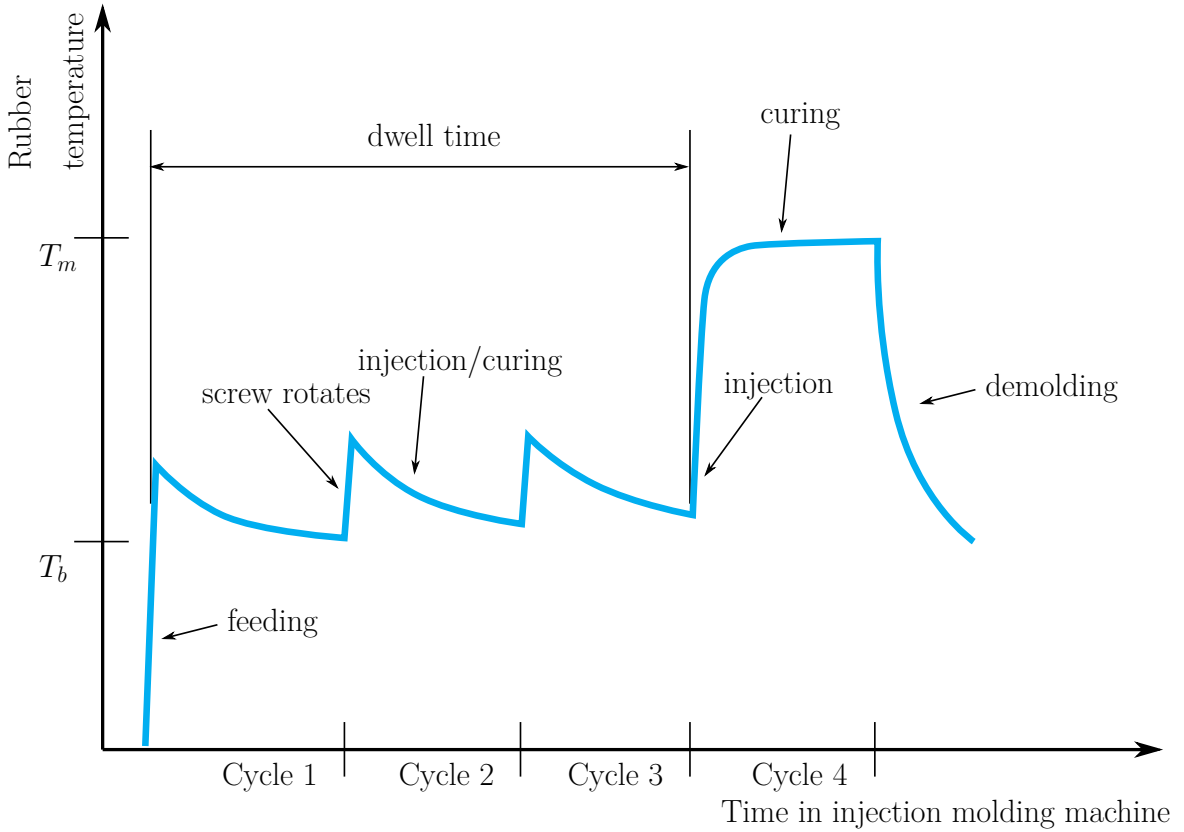
Figure 1.3: From entering the screw to demolding, thermal energy is put into rubber during the injection molding process [11].

$$k_c(T) = k_c(T_0)\mathrm{e}^{\frac{E_A}{RT}}. \tag{1.3}$$

In these equations, n_c is the curing reaction exponent, T_0 is the reference temperature, E_A is the activation energy, and R is the universal gas constant. Thus, the differential equation describing the sulphur curing reaction is

$$\frac{\mathrm{d}x_c}{\mathrm{d}t} = k_c(T_0)\mathrm{e}^{\frac{E_A}{RT}}\left(1 - x_c\right)^{n_c}. \tag{1.4}$$

In rubber injection molding, temperature conditions are strongly non-isothermal, as is outlined in Figure 1.3. There, the temperature of a volume element of rubber is plotted over time, from entering the barrel of the injection molding machine until being demolded within a finished part. Four

15

cycles of dwell time are chosen, because this is a common value, although this is entirely dependent on the manufacturing equipment. It is evident, that the rubber spends two cycles in the barrel until it reaches the screw antechamber and subsequently is injected into the mold. When the screw rotates to convey rubber into the screw antechamber, the rubber temperature rises sharply, and in addition to thermal conduction, dissipational heating occurs. Kerschbaumer investigated the temperature increase of rubber compounds during injection molding extensively and could calculate the temperature increase when the pressure loss is known [63]. For the remaining time of each cycle, the screw does not spin, allowing the rubber temperature to approach equilibrium with the barrel temperature (T_b). In cycle three, the sample rubber volume has reached the screw antechamber and is thus injected into the mold in cycle 4. The high shear rates present in the runner system and gate cause immense shear dissipation, heating the rubber much faster than would be possible just by thermal conduction across the mold/rubber interface. For the remaining cure time, the rubber reaches thermal equilibrium with the mold temperature (T_m) solely by thermal conduction. In an ideal setup, in which machine setpoints are matched exactly all the time and the rubber is a perfect compound, the energy input in this process remains constant, and thus the degree of cure of the finished part remains constant. When considering Equation 1.4, the energy input (E) of such a temperature trajectory can be determined as [64]:

$$E = k_c(T_0) \int_0^{t_d} e^{\frac{E_A}{R}\left(\frac{1}{T(t)} - \frac{1}{T_0}\right)} \mathrm{d}t, \qquad (1.5)$$

where continuous time (t) is the curing time, t_d is the time of demolding, and $T(t)$ is the curing time dependent temperature in non-isothermal conditions. It is clearly visible from both Figure 1.3 and Equation 1.5, that the higher temperatures present in the mold make up the majority of the energy input of the rubber during injection molding, and are thus much more critical to process consistency.

In real world settings, a high number of factors can be named which
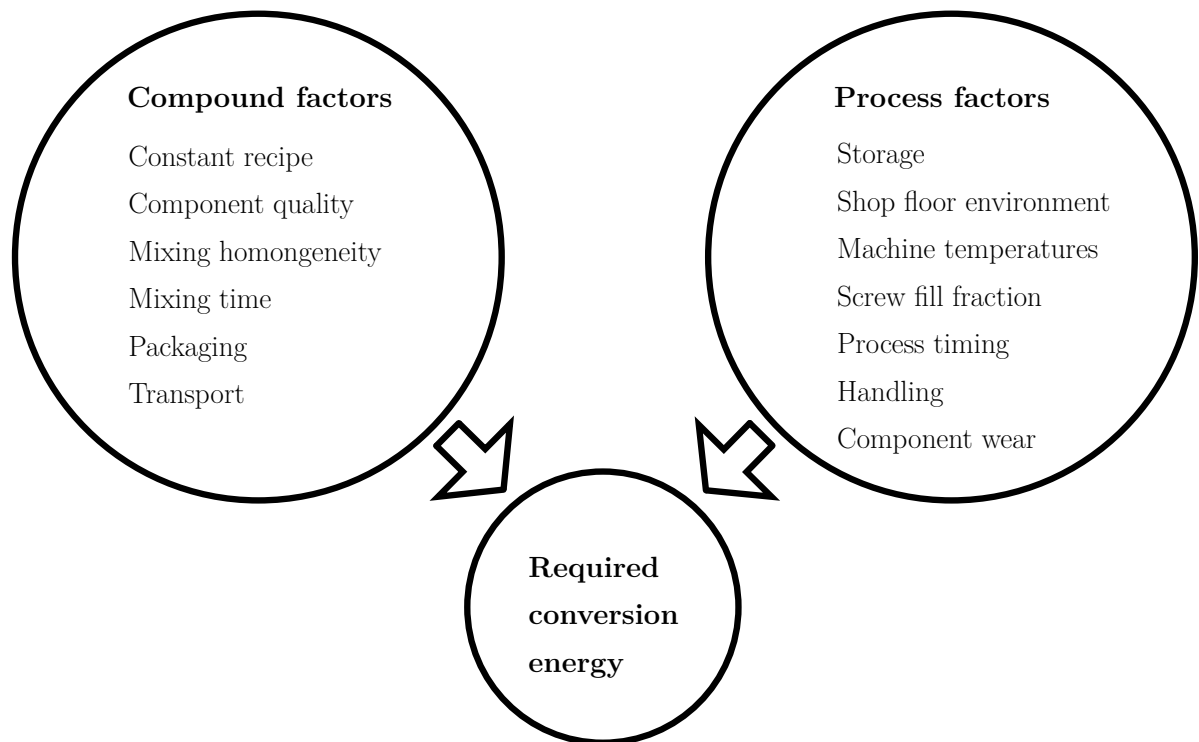
Figure 1.4: Numerous factors both from the compound and the process influence the degree of cure of a rubber part upon demolding [56, 57, 65].

alter both the energy input or k_c, causing the actual x_c to deviate from the required x_c, even though the injection molding equipment's operating point stayed constant. Figure 1.4 shows the most important sources of fluctuations for rubber injection molding purposes. On the compound side, variations in recipe, activity of fillers, the reactivity of the curing system, and mixing homogeneity show a direct impact on the amount of energy that is needed to reach the desired degree of cure. For example, if there are less inhibitors in a batch, or their reactivity is reduced, the activation energy (E_A) for the curing reaction will be increased. Also, if variations alter the flow behaviour of the compound, the energy input by dissipation in the injection molding machine will be different. The mixing time, packaging (extruded strips, bulk) and transport conditions will add to the total amount of energy input

17

into the rubber compound, which is not accounted for in the injection molding process. Further sources of variability can be introduced by the manufacturing process: During storage, physical and chemical processes take place which can change the rubber's flow behaviour and reduce scorch time. The rate at which these take place is strongly dependent on storage climate [65–67]. Handling, process timing, machine temperatures, and shop floor environment again contribute to the total energy input into the rubber compound. In addition, component wear, for instance degradation of the non-return valve, has severe impacts on process stability, as the injected volume can not be kept constant and holding pressure can not be applied properly.

1.5 A data-driven process control approach

To manufacture rubber parts by injection molding, which are cured exactly to the required amount and thus meet the set quality requirements, variations from all these influences need to be minimized. To 100 % ensure this, great efforts have to be taken by part manufacturers. Thus, manufacturers wish for the injection molding machine to detect deviations in the state of the material and adapt its process settings accordingly. As the approaches discussed in section 1.3 are either developed for thermoplastic injection molding or require extensive amounts of testing prior to injection molding, a data driven-monitoring and fault identification system for rubber injection molding will be developed in this work. When parts are manufactured from thermoplastic materials, the plastic itself is commonly developed by the material supplier. OEMs and component suppliers then search the market for a grade that suits their mechanical, chemical, or other requirements. As a result, while there are many grades on offer, their variety is limited to a certain degree. Rubber compounds, on the other hand, are almost always developed in-house by the part suppliers or part manufacturers, and rubber compound recipes tailor the materials properties exactly to the application requirements. Therefore, there is an almost unlimited amount of rubber

compounds in use, which all differ in their processing properties such as flow behavior, curing characteristics or storage maturation. One compound, for example, could show an increase in viscosity and accelerated curing when stored, while another compound's viscosity and curing rate decreases during storage [65, 68].

First principle and knowledge-based monitoring methods require the knowledge of rules (models for the material behavior) before they can work successfully, which is shown in Figure 1.2(page 8), As there is so much variation in rubber compounds, manufacturing equipment suppliers wouldn't be able to supply a satisfying amount of models with the operating system of their equipment. Moreover, rubber goods manufacturers are also unwilling to establish those models for their own compounds, as this would require a significant amount of resources [11]. In this work, a data-driven approach to process monitoring was taken, since it does not require any a-priori knowledge of the material behavior to be implemented [7]. Rather, as long as a sufficient amount of training data can be provided, no preliminary tests need to be performed with the rubber compound.

To develop this approach, first the rubber temperature will be determined throughout the injection molding process with new methods, which increase the reliability of the gathered temperature data. Common sources of error such as temperature differences or variations in storage history of the same Acrylonitrile-butadiene rubber (NBR) rubber compound will be investigated and their impact on the temperature of the rubber will be quantified (chapter 5). Furthermore, a data-driven PCA-based monitoring approach will be implemented. This monitoring approach takes into account process signals readily available on commercial machines to automatically detect changes in the material state without any need for prior testing. Finally, a fault identification approach is presented, which is able to discern between the types of faults that are introduced to the process.

2 Background

MANUFACTURING high quality rubber parts requires considerations in multiple engineering areas. First, the material properties of rubber compounds and the chemical and physical structure that determines them need to be understood. Second, the process needs to be designed and set up in a way that ensures technical and economic optimality. Third, process monitoring needs to be established to ensure variations in the boundary conditions (batch to batch variations, environment, machine wear), which may lead to defective parts being detected automatically so the operating point of the process can be adapted to the new conditions. In this chapter, an engineering background on these subjects shall be given to (1) back up assumptions made about the flow behavior of rubber compounds with known material laws and (2) give insights on the methods used to detect faults in the process and determine the type of these faults.

2.1 Flow behavior of highly filled rubber compounds

When an incompressible fluid (its density does not change with pressure) flows through a constant control volume, which has no sources or sinks, the conservation of mass can be written in its simplest form:

$$\nabla \mathbf{v} = 0, \tag{2.1}$$

where the velocity vector ($\mathbf{v}$) describes the velocity components in each dimension. Also, the conservation of momentum applies, which can be written as [43, 69, 70]. When steady state and isothermal flow are assumend

and gravitational forces are omitted, it reads as

$$-\nabla \mathbf{p} + \nabla \underline{\underline{\tau}} = 0. \tag{2.2}$$

In this expression, $\mathbf{p}$ is the vector of the pressure present in every dimension, and $\underline{\underline{\tau}} \in \mathbb{R}^{3\times3}$ is the stress tensor. However, for the flow of rubber through cylindrical flow channels (e.g. the runner of injection molds), Equation 2.1 and Equation 2.2 can be reduced to a two dimensional problem, thus:

$$\frac{\mathrm{d}v}{\mathrm{d}x} = 0, \tag{2.3}$$

and

$$\frac{\mathrm{d}p}{\mathrm{d}x} = \frac{\tau}{r}\frac{\mathrm{d}\tau}{\mathrm{d}r}, \tag{2.4}$$

where $p = \mathbf{p}_1$, $\tau = \underline{\underline{\tau}}_{1,2} = \underline{\underline{\tau}}_{2,1}$, and r is the radius of the flow channel. Integration along the direction of flow (x) for the entire length (L) of the channel yields:

$$\int \mathrm{d}p = \int_0^L \left(\frac{\tau}{r}\frac{\mathrm{d}\tau}{\mathrm{d}r} \right) \mathrm{d}x, \tag{2.5}$$

and its definite integral is:

$$\frac{\Delta p}{L} = \frac{\tau}{r} + \frac{\mathrm{d}\tau}{\mathrm{d}r}. \tag{2.6}$$

Solving the differential equation (Equation 2.5) and using it to express τ leads to

$$\tau = \frac{\Delta p}{L}\frac{r}{2}. \tag{2.7}$$

This shows that in the case of circular flow channels, the shear stress is a linear function of the radius (r). To obtain expressions for the radial distributions of the shear rate ($\dot{\gamma}$), and the velocity (v), τ of Equation 2.7 is to be replaced with empirical laws mapping shear rate to shear stresses for different fluid types (e.g.: Newton's law, Power law or Carreau-WLF) [43, 69, 71, 72]. However, rubber compounds are suspension liquids with potentially very high amounts of filler loadings (commonly exceeding 100 parts per hundred rubber (phr)), which allows the fillers to form continuous
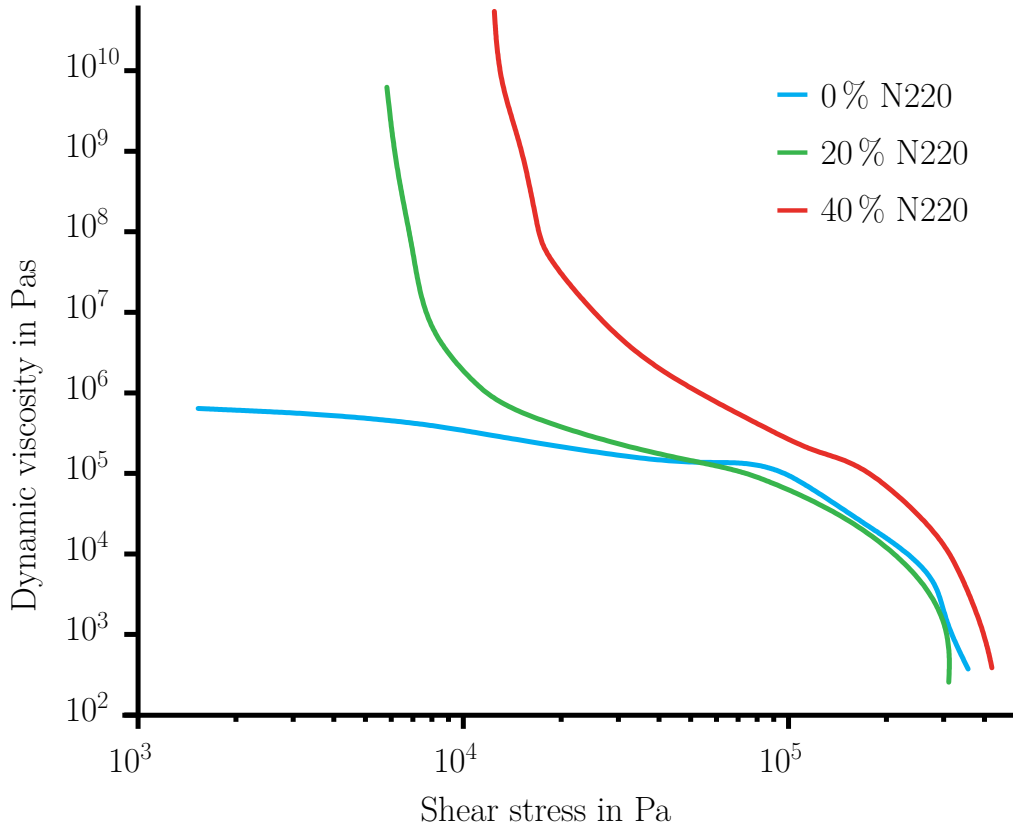
Figure 2.1: The dynamic viscosity of EPDM filled with N220 carbon black increases sharply when shear stress is low. Modeled after [79].

networks when no or little shear is applied.

Furthermore, many rubber compounds - including grades in this work (section 3.1) - use carbon black as a reinforcing filler, which is widely used for its very strong interactions with the polymer molecules [73–79].

As a consequence, rubbers compounds show characteristics of Bingham fluids, where τ needs to surpass the yield stress (τ_0) for the compound to start flowing. To model the radial $\dot{\gamma}$ and v profiles of such suspensions, a more general flow model, introducing a term for τ_0 has to be used [43, 80]:

$$\dot{\gamma} = -\frac{\mathrm{d}v}{\mathrm{d}r} = \left(\frac{\tau - \tau_0}{k}\right)^{\frac{1}{n}}. \tag{2.8}$$

In this equation, n is the flow exponent, characterizing the amount of
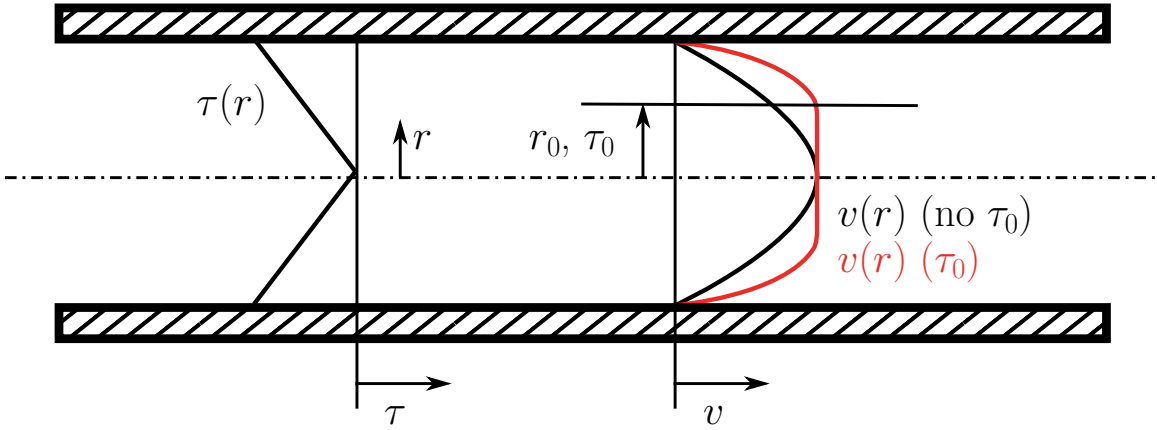
Figure 2.2: When the shear stress is lower than τ_0, the velocity profile of carbon black filled rubbers forms a plug with radius r_0 [43].

shear thinning of the flowing medium and k is the consistency index. As a result, when the shear stress is lower than τ_0 towards the center of the flow channel, the rubber develops a kind of plug flow, illustrated in Figure 2.2. This limits the level of temperature control that can be applied by varying the volume flow rate during injection, as shear heating will be large where $r > r_0$, but does not exist at $r < r_0$. Experimental evidence for this shall be discussed in section 5.2.

2.2 Statistical Process Monitoring

Many industrial processes are considered to be steady state processes, where the relationship between the process input matrix (X) and process output matrix (Y) is given by a linear relationship [7, 8, 81]:

$$Y = AX + E, \tag{2.9}$$

where the $Y \in \mathbb{R}^{I \times M}$ contains all M output features determined for all I batches, $X \in \mathbb{R}^{I \times J}$ contains measurements of all J process variables (signals). $A \in \mathbb{R}^{J \times M}$ is the process parmeter matrix mapping X onto Y, and $E \in \mathbb{R}^{I \times J}$ represents normally distributed noise with zero mean, which is

independent of the X. For example, Y could be the collection of part quality attributes such as physical and mechanical characteristics. The X matrix would then consist of features from process variables such as maximum injection pressure (p_i), the maximum in-mold pressure (p_m) etc. Thus, X can be written as a matrix containing all variable vectors:

$$X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_J]^T , \qquad (2.10)$$

where each process signal vector ($\mathbf{x}$) contains the measurements of process signal J at each batch. Commonly, each $\mathbf{x}_j$ is treated as being normal distributed with a mean value μ_j and a variance σ_j^2 [8]:

$$\mathbf{x}_j \sim \mathcal{N}\left(\mu_j, \sigma_j^2\right). \qquad (2.11)$$

2.2.1 Exponentially Weighted Moving Average

As already discussed in chapter 1, univariate control charts are widely used in industrial production to monitor the quality of processes, detect fluctuations or process drifts, which negatively influence the quality of the product. The most important methods are the Shewhart-chart, Cumulative Sum (CUSUM) charts and the Exponentially Weighted Moving Average (EWMA) [14, 82, 83]. As they are used for model comparison in section 6.2, the construction of EWMA charts shall be discussed briefly. It is important to note that one EWMA chart has to be constructed for each $\mathbf{x}_j$ in Equation 2.10. One key difference between EWMA and other control charts it is consideration of past measurements when calculating the present monitoring statistic. The EWMA monitoring statistic (z) of a signal j is calculated by:

$$z_{j,i} = \lambda \sum_{n=0}^{i-1} (1-\lambda)^n x_{j,i-n} + (1-\lambda)^i z_{j,0} \qquad (2.12)$$

where $i = 1, 2, \ldots, I$ is the number of the batch, and λ ($0 \leq \lambda \leq 1$) is the memory factor and $z_{j,0}$ is the control statistic of signal j at the beginning of monitoring. λ has to be chosen empirically, as its value determines the

impact of past values on the statistic of the latest measurement. The test statistic at the beginning of the monitoring, $z_{j,0}$ is, in this work, chosen to be the sample mean of the training set $\mu(\mathbf{x}_{j,train})$. As always, which data belong to the training set is chosen during model building. To perform process monitoring, control limits for each z_j need to be calculated. When $\text{EWMA}_{j,i}$ is outside of these limits, the process is considered to be out-of-control. The upper control limit of the EWMA chart (UCL) is calculated by [82]:

$$\text{UCL}_j = z_{j,0} + 3\sigma\sqrt{\frac{\lambda}{2-\lambda}(1-(1-\lambda)^{2i}},} \tag{2.13}$$

where σ is the standard deviation of the train data $\mathbf{x}_{j,train}$. In analogy, the lower control limit of the EWMA chart (LCL) is calculated as:

$$\text{LCL}_j = z_{j,0} - 3\sigma\sqrt{\frac{\lambda}{2-\lambda}(1-(1-\lambda)^{2i}}.} \tag{2.14}$$

To visualize the effect of different values for memory factor of EWMA (λ), $I = 50$ instances of a randomly generated variable X are monitored with EWMA charts. The variable was sampled from the following distributions:

$$\begin{cases} i < 30 \implies X \sim \mathcal{N}\left(\mu = 0, \sigma = 1\right) \\ i > 30 \implies X \sim \mathcal{N}\left(\mu = 3, \sigma = 1\right). \end{cases}$$

It is clearly visible, that in Figure 2.3 a), where $\lambda = 0.2$, z values are smoothed and the shift in distribution mean is detected with a significant delay. However, once the shift is detected, all samples are recognized as out-of-control. The exact opposite can be seen in Figure 2.3 b). Thus, the value of λ needs to be chosen with the requirements of the monitoring system and the expected fault behavior of the process in mind.
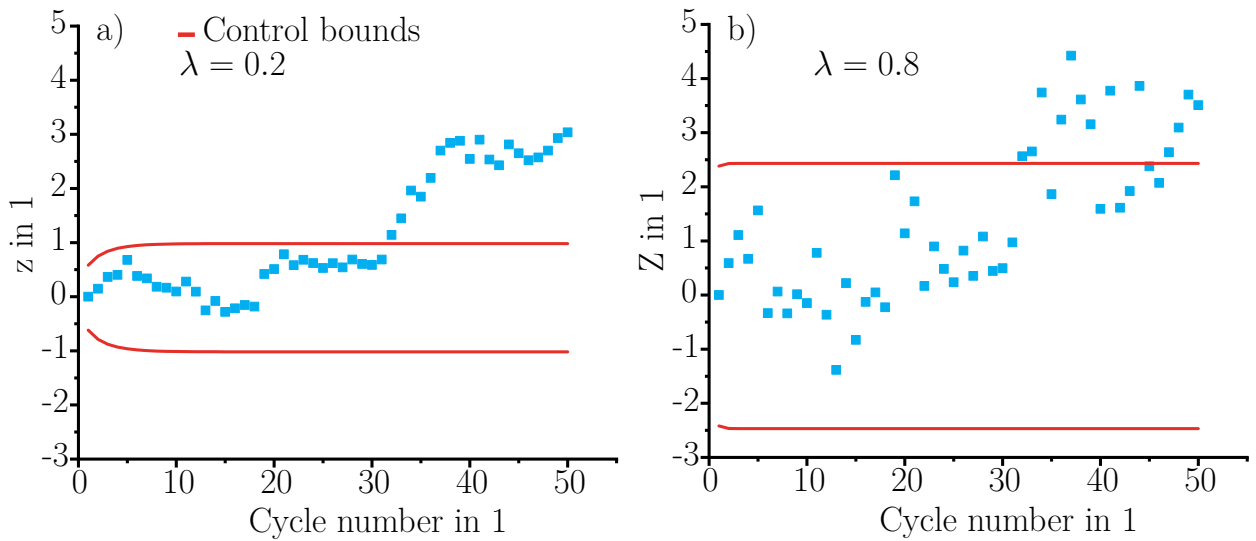
Figure 2.3: A change of sample distribution mean from 0 to 3 at cycle 30 is detected differently by plotting the EWMA with a) $\lambda = 0.2$ and b) $\lambda = 0.8$

2.2.2 Principal Component Analysis

Univariate monitoring (monitoring each $\mathbf{x}_j$ independently) is only effective when the $\mathbf{x}$ are linearly independent of each other, which is almost never true for industrial processes.

In injection molding for example, p_i and p_m can not be independent of each other, as the pressure of the same fluid is measured, just at different locations in its flow path. Figure 2.4 shows the limits of monitoring dependent variables. All 3 faults (marked red) can not be detected by calculating the control bounds at 3σ from the mean, but are clearly outside the elliptical control bound obtained by multivariate methods.

Monitoring two or more dependent variables needs additional methods, and a number of them do exist [8, 13, 29, 53, 85, 86]. In this work, for fault detection purposes, a PCA-based monitoring approach is implemented, which takes into account (1) the linear dependent relationship of process variables of the injection molding process, and (2) process properties unique

27

to injection molding [7].

PCA can be thought of as an autoencoding method being able to represent high dimensional data in a lower dimensional subspace without losing information. As most modern manufacturing equipment provides a high number of monitorable signals, it can be challenging to separate signals with high information density from signals with low information density. PCA accomplishes this automatically. It decomposes the data matrix X into a set of matrices with special properties.

$$X = TP^T = \sum_{j=1}^{J} \mathbf{t}_j \mathbf{p}_j^T = \mathbf{t}_1 \mathbf{p}_1^T + \mathbf{t}_2 \mathbf{p}_2^T + \cdots + \mathbf{t}_J \mathbf{p}_J^T, \qquad (2.15)$$

where $\mathbf{t}_j \in \mathbb{R}^{I \times 1}$ is the score vector which contains the coordinate of each batch in the j-th principal component dimension, and $\mathbf{p}_j \in \mathbb{R}^{J \times 1}$ is the loading vector, which transforms the original data into the principal component space. $T \in \mathbb{R}^{I \times J}$ and $P \in \mathbb{R}^{J \times J}$ are the corresponding score and loading matrices [7, 8, 13]. The most straightforward way to compute P is by singular value decomposition (SVD) of the covariance matrix (Σ) of X:

$$\Sigma_X = \frac{1}{I-1} X^T X, \qquad (2.16)$$

$$\Sigma_X = P \Lambda P^T. \qquad (2.17)$$

Because Σ_X is symmetric and positive semi-definite (if none of the signals $\mathbf{x}_j$ are exactly collinear or zero), SVD directly computes the eigenvalues (λ_j) in $\Lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_J)$, where $\lambda_1 > \lambda_2 > \cdots > \lambda_J$. X can now be decomposed as:

$$T = XP. \qquad (2.18)$$

Dimensionality reduction in PCA is achieved by only selecting the first L dimensions of which the corresponding eigenvalues satisfy

$$\frac{\sum_{j=1}^{L} \lambda_j}{\sum_{j=1}^{J} \lambda_j} \geq 0.8, \qquad (2.19)$$

which means reducing P to $P_{pc} \in \mathbb{R}^{J \times L}$ by keeping only the first L columns and is equal to retaining $80\,\%$ of data variance. This value is empirically chosen as it presents a good balance between data reduction and information loss [7, 13].

2.2.3 PCA-based process monitoring

When PCA modelling is performed on I batches which are considered to be in-control (the training set), it can be used for monitoring future batches of which the values of the process variables are stored in a vector $\mathbf{x}_{new} = [x_1, x_2, \ldots, x_J]$. In analogy to Equation 2.18

$$\mathbf{t}_{new} = \mathbf{x}_{new} P_{pc},\tag{2.20}$$

the score $\mathbf{t}_{new}$ can be determined. It is important to note that calculating the score of a new batch by a loading matrix that was determined without $\mathbf{x}_{new}$ existing is only viable when the underlying variable relationships did not change and P_{pc} still is a relatively good estimate for the real loading matrix of $\mathbf{x}_{new}$ [7, 8, 13]. Almost certainly, a small error will be made in this calculation, and this error proves to be very useful for process monitoring purposes. If Equation 2.20 is possible, then:

$$\hat{\mathbf{x}}_{new} = \mathbf{t}_{new} P_{pc}{}^{T},\tag{2.21}$$

$$\mathbf{e}_{new} = \mathbf{x}_{new} - \hat{\mathbf{x}}_{new},\tag{2.22}$$

where $\hat{\mathbf{x}}_{new}$ is the vector of process variable values estimated by the PCA model, and $\mathbf{e}_{new}$ is the residual vector which captures the prediction error of the PCA model. Two control statistics, which work in tandem to determine whether $\mathbf{x}_{new}$ belongs to a batch which can be considered in-control or out-of-control. First, Hotelling's T^2 statistic (T^2) can be calculated as [7, 87]

$$T^2{}_{new} = \mathbf{t}_{new} \Lambda_{pc}^{-1} \mathbf{t}_{new}^{T},\tag{2.23}$$

where Λ_{pc} is Λ reduced in the same way as P_{pc} to feature only the M largest eigenvalues. It can be seen that the T^2 statistic is closely related to the squared generalized distance of the multivariate normal distribution, as it measures the distance from the center of the training data's T matrix to $\mathbf{t}_{new}$ [88, 89]. It does not indicate changes in the process signals correlation, therefore the Squared Predictive Error (SPE), also known as the Q-statistic is also calculated [7, 8, 13]:

$$\text{SPE}_{new} = \mathbf{e}_{new}\mathbf{e}_{new}^T.$$
(2.24)

As the SPE is more sensitive to changes in varying signal relationships, it should always be used in conjunction with T^2. To detect faults with either of these statistics, the confidence bounds T^2_t and SPE_t respectively need to be calculated at a specified significance level (α).

$$T^2_t = \frac{M(I^2 - 1)}{I(I - M)} \text{F}_{\alpha,M,I-M}$$
(2.25)

$\text{F}_{\alpha,M,I-M}$ is the $(1 - \alpha) \cdot 100\,\%$ percentile of the F distribution (F) with the degrees of freedom L and I-L [13]. The critical value SPE_t can be determined by

$$\text{SPE}_t = \theta_1 \left(\frac{c_\alpha \sqrt{\theta_2 h_0^2}}{\theta_1} + 1 + \frac{\theta_2 h_0 (h_0 - 1)}{\theta_1^2} \right)^{\frac{1}{h_0}},$$
(2.26)

with

$$\theta_i = \sum_{j=A+1}^{M} \lambda_j^i \; (i = 1, 2, 3),$$
(2.27)

and

$$h_0 = 1 - \frac{2\theta_1 \theta_3}{3\theta_2^2}.$$
(2.28)

In Equation 2.26, c_α is the deviation of the upper $(1 - \alpha) \cdot 100\%$ percentile of the gaussian distribution.
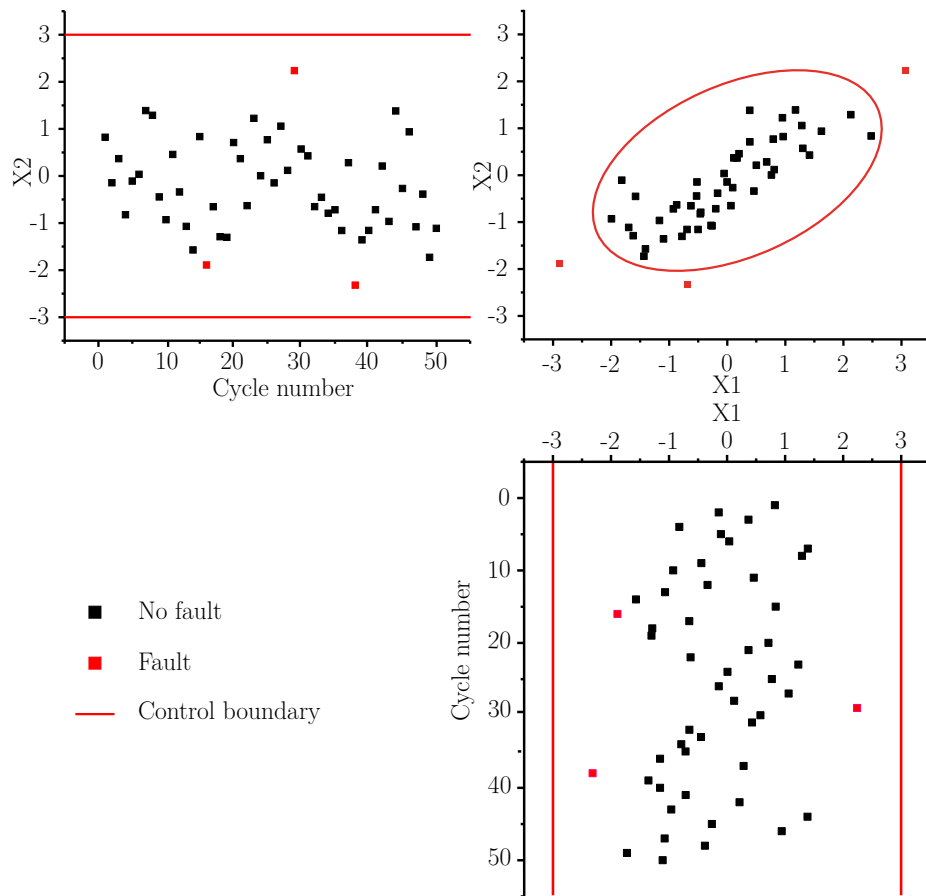
Figure 2.4: Univariate monitoring of the linearly dependent variables X1 and X2 misses many faults, while multivariate control can increase fault detection rate significantly [84].

2.2.4 Challenges of monitoring the injection molding process

At the beginning of section 2.2, it was stated that X may contain process signal features. However, in injection molding and many manufacturing processes, process signals are sampled at a number of instances during one batch, and feature extraction reduces these time series to a single value. Consider the injection pressure p_i. When only its maximum value is used for monitoring, faults that present themselves more clearly in other features, such as kurtosis or skewness, can easily be missed. To establish more reliable process monitoring methods for injection molding, the nature of the injection molding process has to be pictured as in Figure 2.5. A single cycle (batch) of the injection molding process presents itself can be considered as a non-stationary dynamic process, where each process signal is a discrete time series vector. Each batch can be imagined as a 2-dimensional matrix of size $J \times K$, containing all J process signals and all K samples captured of each cycle by the monitoring system. These matrices can then be stacked batch-wise for all I batches, to form a tensor of shape $I \times J \times K$ as it is done in Figure 2.5 a) [7]:
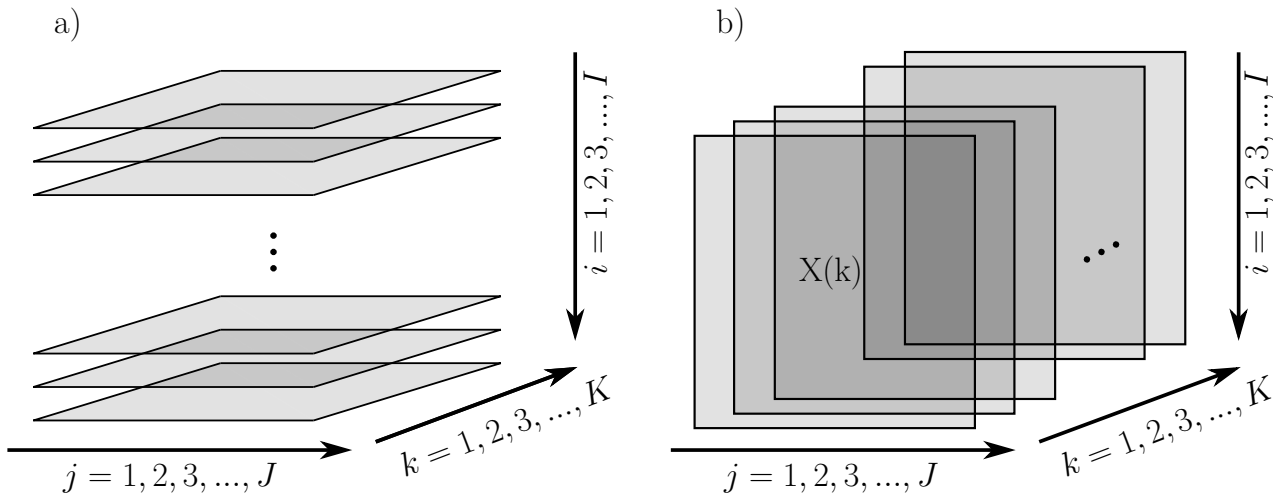
Figure 2.5: An injection molding process can either be viewed as a) a sequence of dynamic instationary process batches or b) stationary single point values containing one measurement of each process signal taken at sample k [7].

The process can then be also viewed at each sample k (sample-wise), as a discontiuous process, which -after startup- operates in a steady state. Equation 2.18 can be applied to each $X(k)$ of the training set. When new batches occur, monitoring as in Equation 2.20 to Equation 2.22 can be performed for every $\mathbf{x}_{new}(k)$. Programmatically, cycling through every k (sampling rate is commonly between $100\,\mathrm{Hz}$ and $1000\,\mathrm{Hz}$ and an injection molding cycle can take up to one minute without curing) can also be replaced by stacking the $X(k)$ in 2 dimensions. Details on this can be found in [7, 26].

More important are injection molding's phase characteristics. A typical injection molding process consists of a number of different phases, with the most important to monitoring being injection, packing, curing, and dosing. Within each of these phases, the variable correlations are similar. For example, during injection, both p_i and p_m rise, while $\dot{V}_i >> 0$. During packing however, p_i stays constant, while p_m varies depending on the process and $\dot{V}_i$ is very small. These phase-dependent correlations are depicted in Figure 2.6 with normalized process data from experiments discussed in chapter 6.

The major implication for monitoring a multi-phase process with PCA is that for all $X(k)$ belonging to a specific phase $c = 1, 2, \ldots, C$, the Σ should be similar. Thus, instead of using separate loading matrix (P) matrices to calculate statistics from $\mathbf{x}_{new}(k)$, one can determine a representative phase loading matrix (P_c) as:

$$P_c = \frac{1}{I_c} \sum_{k}^{K_c} P(k), \tag{2.29}$$

where I_c and K_c are the number of batches and samples in the training data set for the specific phase and $P(k)$ is the loading matrix of $X(k)$ [7]. Performing PCA monitoring with separate $P(k)$ for every k leads to a high False Alarm Rate (FAR) as hyper local deviations are detected. Using P_c improves the monitoring performance, as it provides smoothing to the statistics calculated from the process signal vector ($\mathbf{x}_{new}$). Process
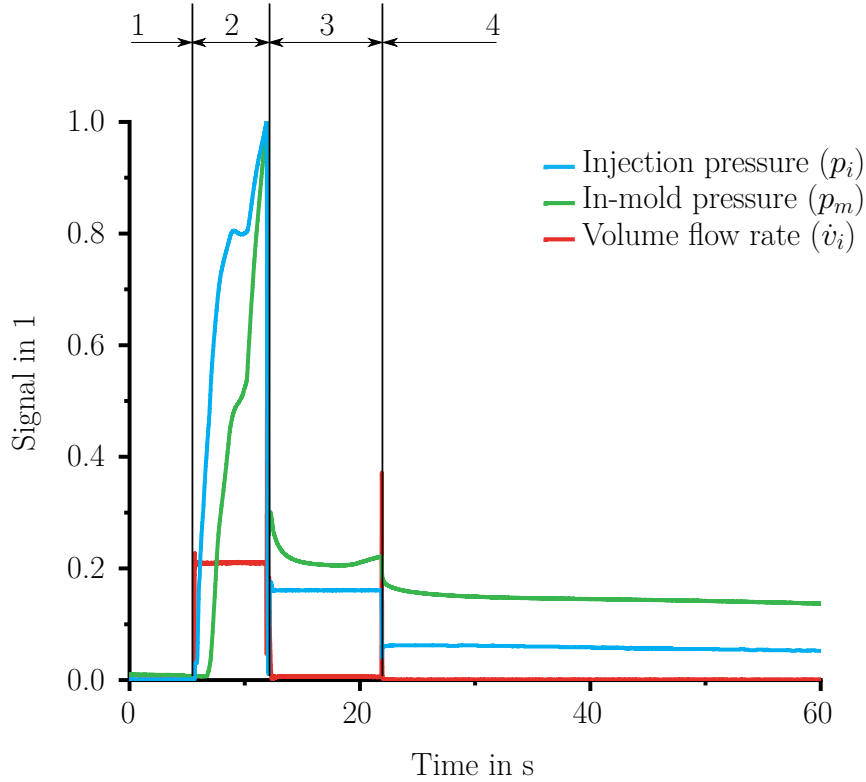
Figure 2.6: Different phases in the rubber injection molding process exhibit different variable correlations. 1: mold closing, 2: injection phase, 3: packing phase, 4: curing.

monitoring is thus performed by calculating:

$$T^2{}_{new}(k) = \mathbf{t}_{new}(k)\Lambda_c^{-1}\mathbf{t}_{new}^T(k), \tag{2.30}$$

and

$$\mathrm{SPE}_{new}(k) = \mathbf{e}_{new}(k)\mathbf{e}_{new}^T(k), \tag{2.31}$$

which means that Equation 2.23 and Equation 2.24 are calculated at each sample point k. Λ_c is defined in analogy to Equation 2.29 as $\Lambda_c = \frac{1}{I_c}\sum_{k=1}^{K_c}\Lambda_{pc}(k)$.

Lastly, the control bounds from Equation 2.25 and Equation 2.26 have to

be adapted [7]:

$$T^2_c = \frac{MI(K_c - 1)}{I(K_c - 1) - M} F_{\alpha, M, I(K_c - 1) - A},$$ (2.32)

and

$$\mathrm{SPE}_c(k) \sim g(k) \mathcal{X}^2_{h(k), \alpha}.$$ (2.33)

where $g(k) = \frac{v(k)}{2m(k)}$ and $h(k) = \frac{2m(k)^2}{v(k)}$. $m(k)$ is the mean SPE value at sample k and $v(k)$ is the variance of SPE a the same k. While the form of T^2_c stays similar to standard PCA based monitoring, SPE_c changes and becomes a trajectory with k elements. Yang et. al. apply a very strict logic, where a batch is marked out-of-control when at any k, $\mathrm{SPE}_c(k) < \mathrm{SPE}_{new}(k)$ [7]. However, this has proven to be too strict for the processes monitored in this work, yielding an unacceptable FAR. A low false alarm rate was achieved by a more lenient approach, which averages the detections for each phase:

$$\Delta_{SPE} = \frac{1}{K_c} \sum_{k=1}^{K_c} \mathrm{SPE}_{new}(k) - \mathrm{SPE}_t(k).$$ (2.34)

Monitoring based on the SPE statistic can now be done by a simple decision logic for each process phase:

$$\begin{cases} \Delta_{SPE} > 0 \implies fault \ (out{-}of{-}control) \\ \Delta_{SPE} < 0 \implies no \ fault \ (in{-}control). \end{cases}$$ (2.35)

As the threshold for the T^2 statistic, T^2_t the decision logic is:

$$\begin{cases} T^2_t < any(T^2_{new}(k)) \implies fault \ (out{-}of{-}control) \\ T^2_t > all(T^2_{new}(k)) \implies no \ fault(in{-}control). \end{cases}$$ (2.36)

2.3 Fault identification with Fisher Discriminant Analysis

PCA based multivariate process monitoring is able to reliably detect fluctuations and faults in industrial processes with many linearly dependent process

variables, but on its own, it does not provide insight on the causes of the fault or the type of the fault. A simple method to outline the signals that are most responsible for causing an out-of-control state ($\Delta_{SPE} > 0$) in the SPE statistic is to calculate the mean value of $\mathbf{e}_{new}(k)$ for each phase c. This is called the error contribution of signal j [7]. For shop floor personnel, this can be valuable information, as they possess a high amount of knowledge on which faults manifest themselves in which signals. There are also methods for calculating contributions for $\mathbf{t}$, however, as values are still transformed in principal component space, their real-world use is limited [13, 90].

In injection molding manufacturing setups different types of faults will occur repeatedly over time. When these are detected by process monitoring, it is possible to create databases with the process variables labeled by the fault type. When such databases are present, FDA is a powerful method to automatically identify the fault type as it occurs, granting maintenance personnel advice on how to get the process back to an in-control state quickly [8, 13]. FDA is, just as PCA, an autoencoder. However, unlike PCA its learning step has to be supervised and a labelled dataset has to be used as the training data. Given the data matrix X from section 2.2, when the data are labelled, each row $\mathbf{x}_i$ is linked to one of W classes. FDA then projects the train data into a new subspace, where the distances between the centroids of individual classes is maximized, while the distance between the measurements of class w, $\mathbf{x}_{i,w}$ are minimized. To do this, metrics quantifying the scatter within a class, between different classes and the total scattering of the train data need to be calculated [13, 91, 92]. The total scatter matrix (S_t) is given by

$$S_t = \sum_{i=1}^{I}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T, \tag{2.37}$$

where $\overline{\mathbf{x}}$ is the total mean vector

$$\overline{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{I}\mathbf{x}_i. \tag{2.38}$$

For each subset of vectors $\mathbf{x}_i \in w$, the within-class scatter matrix (S_w) has to be calculated as

$$S_w = \sum_{w=1}^{W} \sum_{\mathbf{x}_i \in w} (\mathbf{x}_i - \overline{\mathbf{x}}_w)(\mathbf{x}_i - \overline{\mathbf{x}}_w)^T, \qquad (2.39)$$

where $\overline{\mathbf{x}}_w$ is the mean vector of each class w. Finally, the between-class scatter matrix (S_b) is determined with

$$S_b = \sum_{w=1}^{W} n_w(\overline{x}_w - \overline{x})(\overline{x}_w - \overline{x})^T. \qquad (2.40)$$

The optimization of distances is then carried out by calculating

$$\max_{\mathbf{u} \neq 0} \frac{\mathbf{u}S_b\mathbf{u}}{\mathbf{u}S_w\mathbf{u}}, \qquad (2.41)$$

with $\mathbf{u} \in \mathrm{R}^J$ acting as the transformation vectors. It was shown that this maximization step can be done by solving the generalized eigenvalue problem [91]:

$$S_b\Omega = S_w\Omega\Lambda. \qquad (2.42)$$

Similar to Equation 2.17, Λ is a diagonal matrix containing the eigenvalues λ_j, and the FDA transformation matrix (Ω) contains the FDA vectors ω_j as $\Omega = [\omega_1, \omega_2, \ldots, \omega_J]$. The projected values ζ_i of the data vectors $\mathbf{x}_i$ can then be determined by

$$\zeta_i = \Omega^T \mathbf{x}_i. \qquad (2.43)$$

As FDA is an autoencoding method, dimensionality reduction should be performed as a subsequent step. In FDA, the optimal number of remaining dimensions L is best determined by cross-validation, reducing $\Omega \in \mathrm{R}^{I \times J}$ to $\Omega \in \mathrm{R}^{I \times L}$ [22]. When transformation is successful, classification is performed

by calculating the discriminant function for each class w:

$$
\begin{aligned}
g_w(\mathbf{x}) = &-\frac{1}{2}(\mathbf{x} - \overline{\mathbf{x}}_j)^T \Omega_L \left(\frac{1}{n_w} \Omega_L^T S_{ww} \Omega_L \right)^{-1} \Omega_L^T (\mathbf{x} - \overline{\mathbf{x}}) + ln(n_w) \\
&-\frac{1}{2} ln \left[det \left(\frac{1}{n_w} \Omega_L^T S_{ww} \Omega_L \right) \right],
\end{aligned}
\tag{2.44}
$$

where Ω_L is the reduced FDA transformation matrix and S_{ww} is the first sum in Equation 2.39: $\sum_{x_i \in \omega}(\mathbf{x_i} - \overline{\mathbf{x}}_w)(\mathbf{x_i} - \overline{\mathbf{x}}_w)^T$. When Equation 2.44 has been calculated for every class w, the values can be summarized in a vector of scores $\mathbf{g}_i = [g_1, g_2, \ldots, g_W]$, from which class probabilities are obtained by applying the softmax function [22]:

$$
s(g_w) = \frac{\mathrm{e}^{g_w}}{\sum_W \mathrm{e}^{g_w}}
\tag{2.45}
$$

2.4 Method testing

While PCA and FDA are standard multivariate data analysis methods, they are very rarely used for injection molding, especially rubber injection molding [7, 31, 32, 85]. Thus, the expected monitoring and classification performance is estimated by applying these methods on simulated data. The necessary data were generated by CADMOULD injection molding simulation software (simcon kunststofftechnische Software GmbH, Germany) in conjunction with custom Python routines. More specifically, injection molding simulation of the mold and part detailed in chapter 3 was done by taking the material property data of the NBR used in the experiments of chapter 6, and boundary conditions in simulation listed in Table 2.1 are also chosen to be similar to chapter 6. The process signals were generated by placing sensor nodes on the part's mesh on positions equivalent to the measurement points of the real-world setup. Thus, a pressure signal was recorded at the nozzle entry (similar to p_i), with another sensor at the exact location of the in-mold sensor, where p_m and T_p are recorded.

To generate fault-free and faulty data, the following procedure was imple-

Table 2.1: Boundary conditions of the injection molding simulation runs for method testing

Simulation parameter	Set point
Cure time (t_c) in s	254
Mold tempperature (T_m)	160
Initial rubber temperature in °C	80
Injection volume flow rate ($\dot{V}_i$) in $\mathrm{cm^3\,s^{-1}}$	15
Packing pressure (p_h) in bar	150

mented:

1. Set up the simulation and use real material properties of NBR.

2. Extract pressure and temperature calculations trajectories at positions equivalent to the machine setup.

3. Repeat 1 and 2 with a 5 % increased viscosity of NBR.

4. With Python routines, apply randomized noise, offset and sensor drift similar to real world phenomena to create fault-free and faulty datasets.

5. Perform monitoring and classification with generated datasets.

It is challenging for injection molding simulation software to correctly estimate the pressure loss of flowing rubber [41], thus simulated pressure and temperature values at the sensor location are incorrect. Also, their information content is greatly reduced, as they follow the pressure value at the nozzle entry exactly. However, for model performance evaluation simulated data are valuable, as the models do not assume specific variable trajectories or values to work. Variable relationships are learned entirely from training data, which in this case were 15 randomly selected samples of the fault-free data for PCA monitoring. For FDA analysis the entire dataset was taken into account. For reference, the maximum value and

Table 2.2: Mean and standard deviation of process signal maximum values from simulated fault-free and faulty batches

	mean		standard deviation	
	fault-free	fault	fault-free	fault
Maximum of p_i in bar	956	723	57	22
Maximum of p_m in bar	647	488	39	17
Maximum of T_p in °C	103	9	100	9

scattering of the maximum of each signal is detailed in Table 2.2. Pressures show stark differences between faulty and fault-free data, while scattering of temperature peak height (T_p) is higher than the distance of the respective means. Thus, the monitoring methods should be capable of discerning between the two data classes.
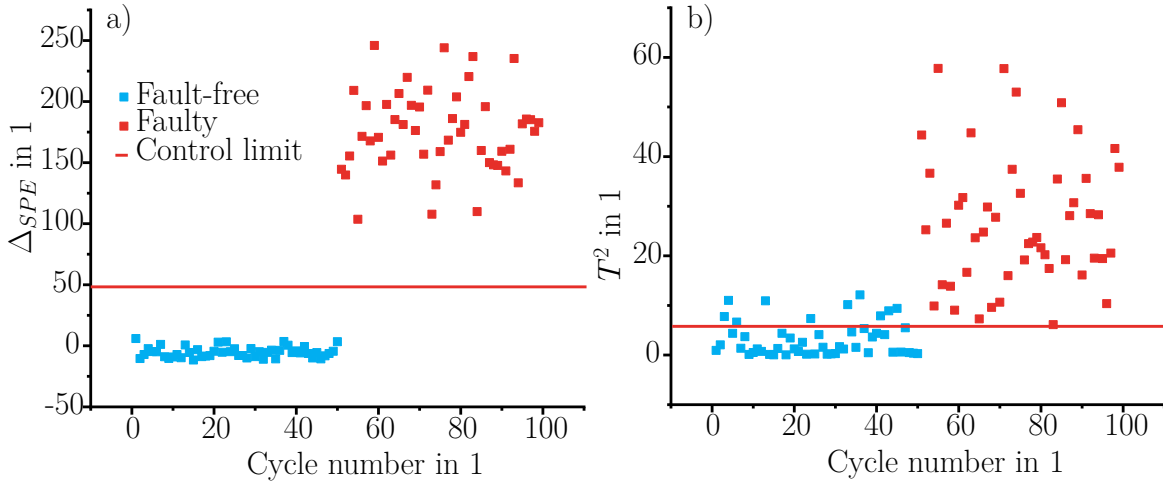
Figure 2.7: Δ_{SPE} and T^2 statistics are capable of detecting faulty simulated process data.

Fault-free and faulty cycles were stacked to follow each other block wise, and the data structure was analysed as if they were recorded in a temporal succession. Figure 2.7 shows the Δ_{SPE} and the T^2 statistic for the simulated
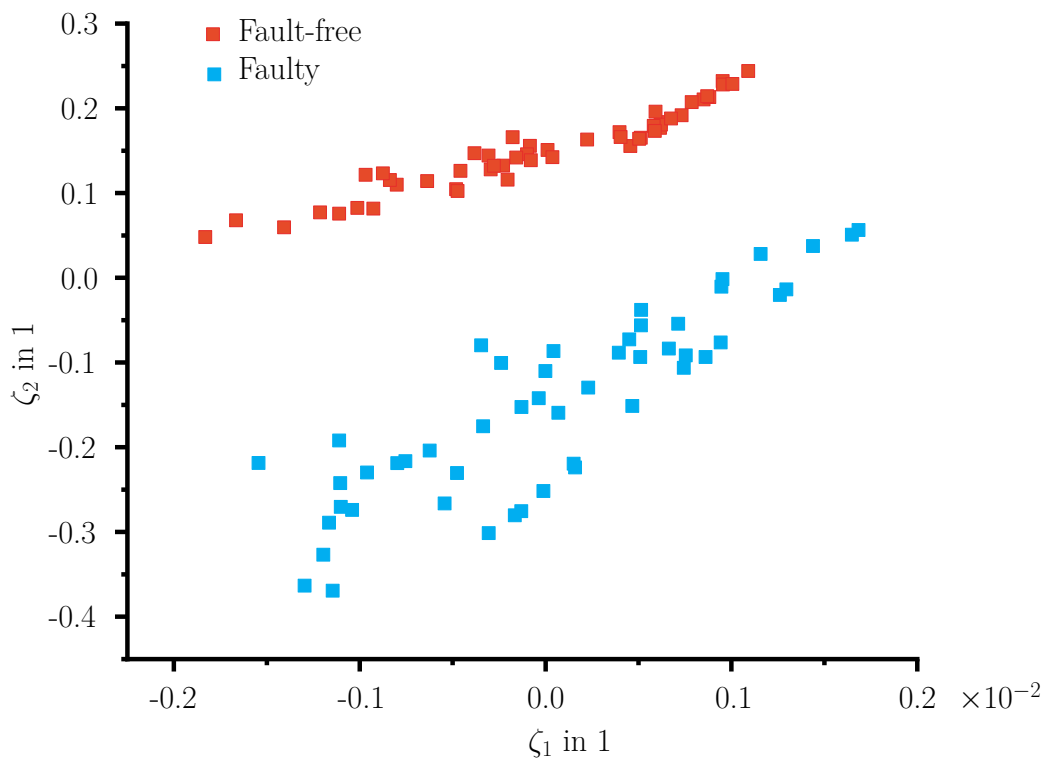
Figure 2.8: In FDA space, fault-free and faulty simulated data can be distinguished clearly.

process data at a level of significance of 95 %. It can be seen that both statistics perform well for detecting the increased viscosity of the NBR. For fault classification, FDA modelling was used to separate the two classes efficiently. Figure 2.8 shows a clear separation between fault-free and faulty batches in FDA space. From these results, it can be said that the performance of both methods is high, and they should be able to detect changes in data from real-world specimen manufacturing runs. Thus, they should be able to be applied to data from real-world injection molding experiments, where the Δ_{SPE} and T^2 statistic can be used to detect faults and the FDA method can detect the type of fault once a fault-type database is present.

3 Materials and equipment

In this work, the injection molding process is investigated with regards to the transient material state, its stability cycle to cycle, and the part quality resulting from specific manufacturing conditions. Thus, diverse equipment is needed, of which short overviews shall be given.

3.1 Rubber compound

Every injection molded part investigated in this work is manufactured from an NBR based rubber compound. In cooparation with SKF sealing solutions Austria (Judenburg, Austria), a compound which does not cause large process fluctuations in industrial manufacturing environments was chosen. Thus, the effects of specific kinds of fluctuations and faults are more likely to be separable by process monitoring systems and less likely to be overpowered by randomly caused disturbances. While disturbances should be detected by process monitoring systems in application, for development, a controlled way of introducing them should be possible. The acrylonitrile-butadiene base polymer for this grade is compounded with carbon black as an active filler and silica as a passive one. It is vulcanized by sulphur based vulcanization agents. For the investigations of this work, the most important material properties of the unvulcanized NBR compound are its flow and curing characteristics.

3.1.1 Curing characteristics of Nitrile-Butadiene Rubber

In this work, degree of cure (x_c) was determined by oscillatory shear experiments according to DIN 53529-1 and DIN 53529-2 for multiple temperatures which are relevant to the experiments [93, 94]. The Rubber Process An-

alyzer (RPA) used for testing in this work was of type D-MDR 3000 by Montech Werkstoffprüfmaschinen GmbH (Buchen,Germany). The test was conducted with an angular frequency (ω) of $10.5\,\text{rad}\,\text{s}^{-1}$ and an amplitude of 5°. The strong temperature dependence of k_c can be seen prominently in the distance between isothermal plots, as k_c commonly doubles with every increase of the curing temperature (T_c) by around $10\,\text{K}$ [56]. Figure 3.1 shows plots of the x_c obtained by RPA measurements at two different isothermal curing temperatures. The test was done for NBR stored for one month at $5\,°\text{C}$ in a refrigerator, and at room temperature (RT). It can be seen that for this specific rubber compound, storing at room temperature does not significantly alter the isothermal reaction kinetics, although for other compounds such behavior could be shown [95] For industrial purposes, a single value for describing the curing kinetics is the time until $90\,\%$ cured (t_{c90}), which is marked for $T_c = 160\,°\text{C}$ in Figure 3.1. It is the time needed to reach an x_c of 0.9, which is the the target x_c after manufacturing for many technical rubber parts.

In the experiments reported in chapter 5, and chapter 6, the impact of storing NBR for 1 month at RT on the material condition as well as the process stability and output of the rubber injection molding process was investigated. During storage of industrial rubber compounds containing active fillers and curing agents, storage maturation can occur. The term storage maturation sums up a number of physical and chemical phenomena which change the flow properties due to physical processes and also, as mentioned in section 1.4, environmental conditions also contribute to the total energy input of rubber. [65, 66, 68, 96].

3.1.2 Flow properties of NBR

The oscillatory shear flow properties of the NBR compound were determined partially in accordance with ASTM 6204:2007. Both the conditioning step and the measurement step were carried out at $100\,°\text{C}$ required for this compound. For other compounds, optimal measurement temperatures may differ significantly [42]. Additionally, oscillatory shear flow properties were
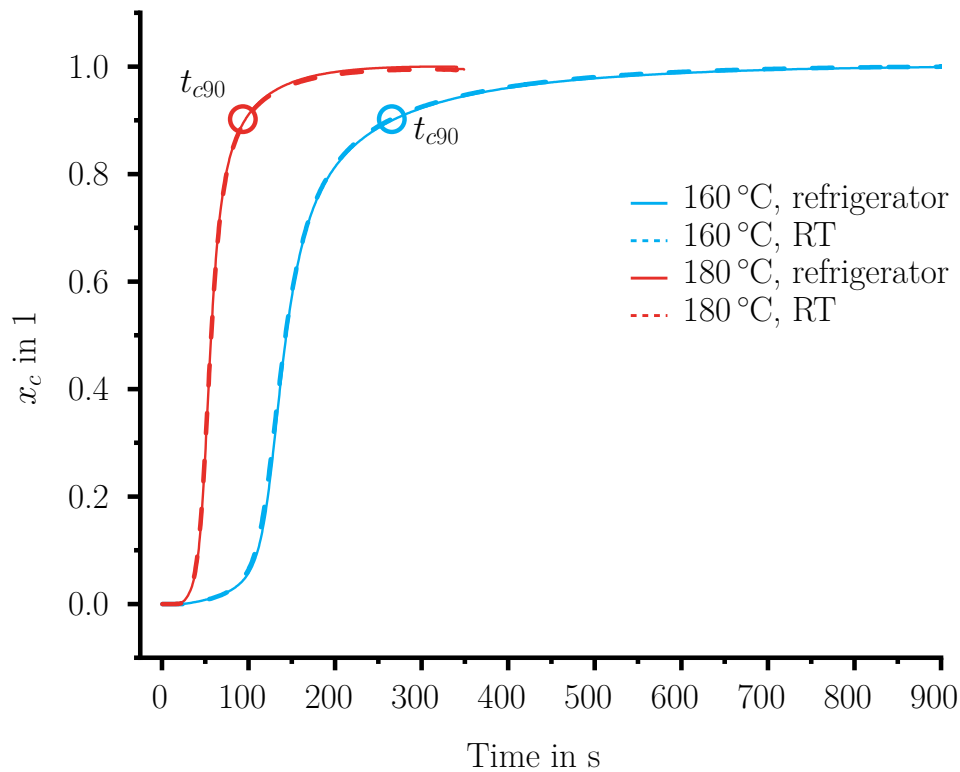
Figure 3.1: The t_{c90} of vulcanization of sulphur curing NBR is highly dependent on the temperature.

measured at 80 °C do show temperature dependencies. The conditioning step breaks down filler networks, which develop due to filler-filler interactions when rubber compounds rest [74, 97]. The ω range of the measurement step is augmented to measure at 20 ω settings from $0.5 \, \mathrm{rad \, s^{-1}}$ to $300 \, \mathrm{rad \, s^{-1}}$. Figure 3.2 shows the complex viscosity (η^*) of the NBR compound measured at 80 °C and 100 °C. When the rubber is stored at RT instead of in a refrigerator at 5 °C for 1 month, its viscosity increases by around 10 %. Such effects of storage maturation for this specific NBR compound have also been reported by Fasching et. al, where they also briefly explore the consequences of increased viscosity on the material energy input during injection [98]. In chapter 5, and chapter 6, the rubber condition and process stability of RT stored NBR will be investigated in greater detail.
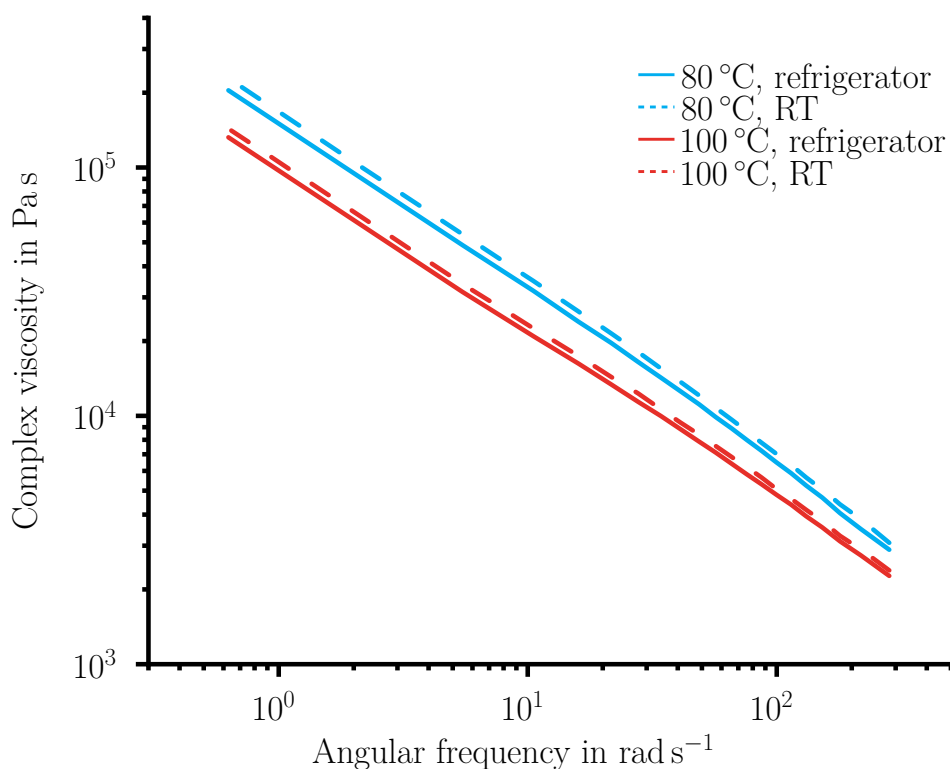
Figure 3.2: Storage maturation increases η^* of the NBR compound slightly both at 80 °C and 100 °C measurement temperature.

3.2 Mechanical testing

The degree of cure of rubber parts is the most important factor on the mechanical properties besides the compound formulation. As the degree of cure is actively influenced by the process operating point, hitting the desired degree of cure every cycle is critical for stable part quality across long production runs. To determine the degree of cure in manufacturing environments, multiple standardized mechanical testing routines are common with manufacturers. In this work, two methods are employed for experiment analysis: The Compression Set (CS) measurement according to DIN ISO 815-1:2016-09 and a custom dynamic testing setup in reference to Kerschbaumer
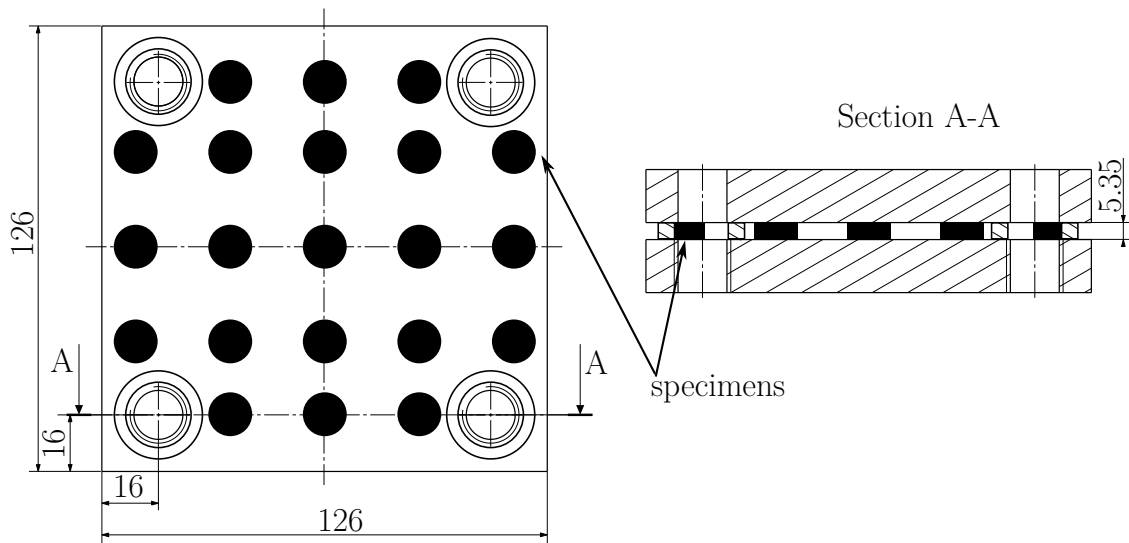
Figure 3.3: The clamping plates are designed to fit 21 CS specimens at once.

[42, 99]. All CS specimens were cut from the location indicated in Figure 3.14 (page 63) and dynamic testing was performed on the same part location, with no need for cutting out specimens.

3.2.1 Standardized Compression Set tests

CS measurements were conducted strictly according to DIN ISO 815-1:2016-09 and the chosen test conditions can be seen in Table 3.1. 21 specimens were clamped in each testing rig outlined in Figure 3.3 and subsequently placed in a lab oven for the required amount of time. The specimen thickness before and after clamping and heating was measured with a digital gauge with an accuracy of 0.01 mm (Käfer Messuhrenfabrik GmbH & CO. KG, Villingen-Schwenningen, Germany).

3.2.2 Testing of dynamic mechanical properties

To test the local quasi-static and dynamic properties of injection molded rubber parts, an Instron E3000 (Illinois Tool Works, IL, USA) testing machine with electrical actuation was used. It was fitted with a custom

Table 3.1: Testing conditions for CS tests [99]

Specimen Type	A
Specimen preparation	rotational cutting
Test temperature in °C	100 ± 1
Test duration in h	$24 \, {}^{0}_{-2}$
Compression in %	15 ± 2

sample holder and a cylindrical test probe shown in Figure 3.4. In this setup, entire parts can be fitted, eliminating the need for extracting specimens at the desired location. In the method building experiments reported in chapter 4, the dynamic properties of compression molded plates shown in Figure 3.5 were measured. To determine the influence of the sample shape, dynamic tests were done using the entire plate (outline marked green), a cuboid specimen (outline marked orange), and a CS style specimen (outline marked blue).

In the experiments discussed in section 6.3, the mold depicted in Figure 3.15 on page 64 was used. The dynamic properties of the parts manufactured in these experiments were measured by applying load on the part area marked "CS specimen" in Figure 3.15. The testing routine is designed to measure the quasi-static and dynamic material parameters static spring stiffness (C_s), relaxation force (F_r), dynamic spring stiffness (C_d), and the loss angle (δ) respectively the loss factor ($\tan(\delta)$). These parameters correlate well with the degree of cure of rubber parts, as Kerschbaumer et al. showed in a similar setup [42, 100]. The displacement (ϵ) and force (F) signals were recorded and analysis was performed in custom Python software, which also incorporated raw data pretreatment such as offset correction and filtering. In this setup, ϵ is actively closed-loop controlled by the testing machine and F is the response signal troughout the routine. Prior to a measurement run, the PID controller of the machine needs to be tuned to a rough estimate of the stiffness of the part to ensure ϵ setpoints are accurately met at all times. To illustrate the measurement routine, sample ϵ and F response trajectories
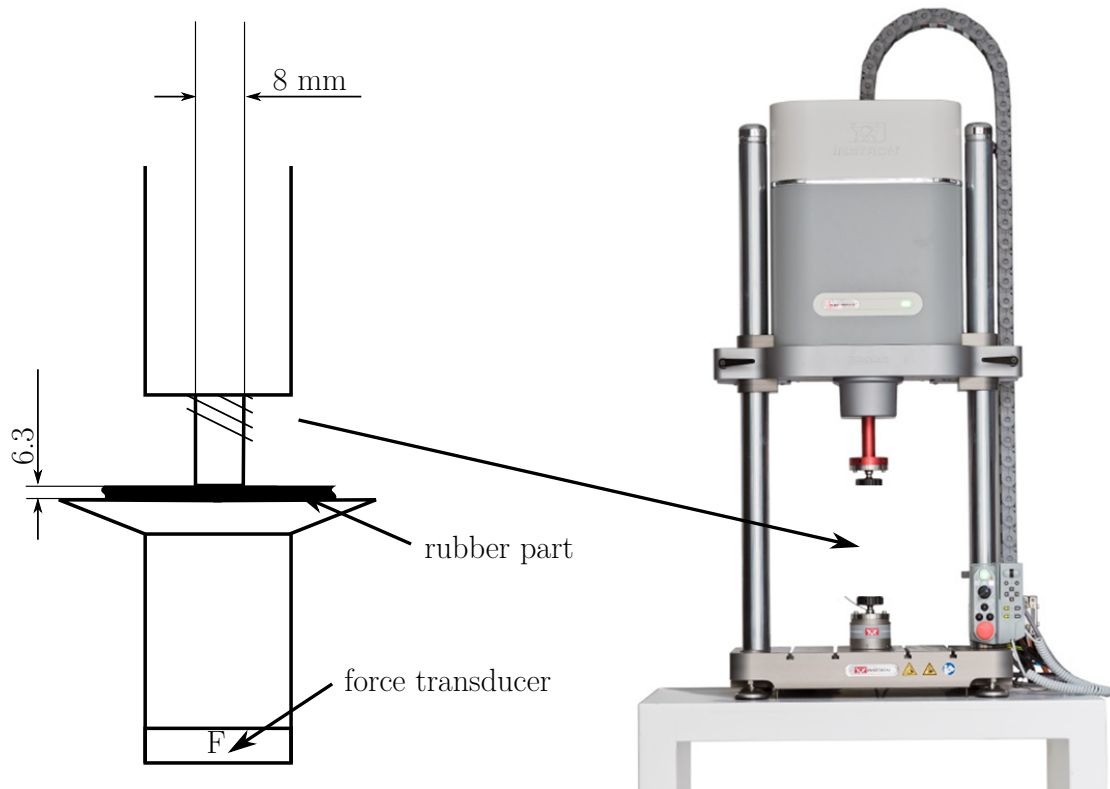
Figure 3.4: The Instron E3000 testing machine was fitted with a custom sample holder and test stamp to perform dynamic testing

are depicted in Figure 3.6. There, five different displacement control phases are marked:

1. To zero the displacement on the surface, the testing probe is pushed against the part with a force of $3\,\mathrm{N}$ for $5\,\mathrm{s}$.

2. The probe compresses the part by 20 % at $1\,\mathrm{mm\,s^{-1}}$.

3. The compression is held constant for $120\,\mathrm{s}$.

4. A dynamic displacement amplitude of $60\,\mathrm{\mu m}$ is applied for 100 cycles at $frequency(f) = 1\,\mathrm{Hz}$.

5. The probe retracts away from the part to end the test.
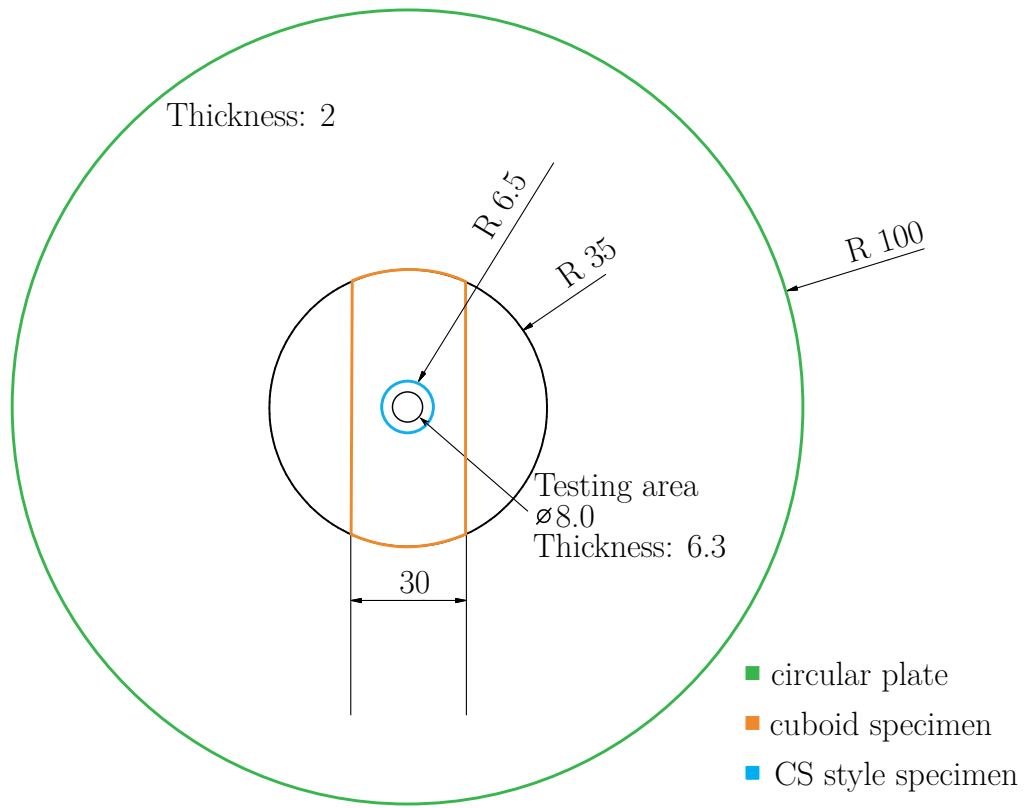
Figure 3.5: The centermost area of compression molded plates was used for dynamic property characterization

In phase 2, C_s is calculated by

$$C_s = \frac{\mathrm{d}F_1}{\mathrm{d}\epsilon}, \tag{3.1}$$

where $F_i, i = 1, 2, \ldots, 5$ is the force signal of the respective displacement control phase. C_s at small displacements corresponds with the compression modulus, which is evaluated in standardized compression tests and common in rubber quality control [56, 101].

The relaxation force (F_r) is another important parameter in rubber testing, used in compound development, material comparisons and quality control

[56, 102–104]. In this setup it is calculated as

$$F_r = max(F_3) - min(F_3). \tag{3.2}$$
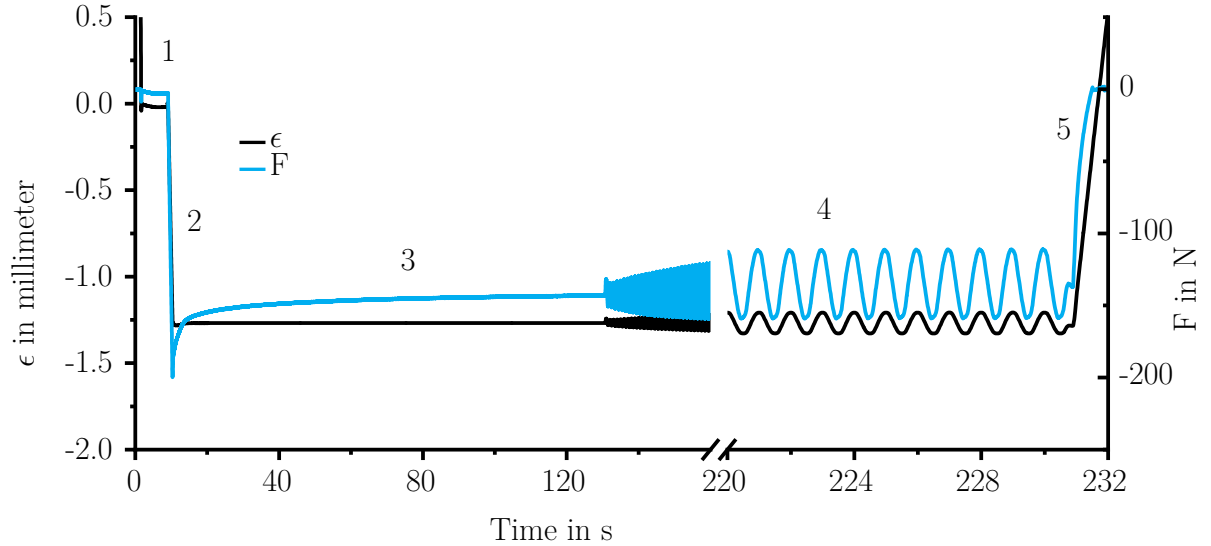
Figure 3.6: The testing routine is a five phase (marked 1 to 5) process designed to measure quasi-static and dynamic material parameters. In phase 1, the displacement is zeroed at the part surface. In phase 2, the part is compressed 20 %, and the compression is held for relaxation testing in phase 3. In phase 4, dynamic testing is conducted and phase 5 is the unloading phase.

In phase 4, the dynamic property parameters of the part are calculated. The evaluation follows the most widely used procedures for analysis of forced oscillation dynamic tests. There, the F is plotted against ϵ instead of time as in Figure 3.7. For ideal elastic materials following Hooke's law, this plot would be a line at an angle of 45° [56]. Rubber can not be considered an ideal-elastic material but rather a visco-elastic one. The molecular level background on this can be found in [43, 56, 69, 70, 104] and many more. For visco-elastic materials, the F-ϵ plots form ellipses, as F is not in phase with the forced ϵ. If ϵ is applied as a sinusoidal function

$$\epsilon = \epsilon_0 \sin(\omega t), \tag{3.3}$$

51

where t is the time, $\omega = 2\pi f$ and ϵ_0 is the displacement at zero angle, the force follows with

$$F = F_0 sin(\omega t + \delta). \tag{3.4}$$

Thus, δ is the angle at which the F response is out of tune with ϵ. When the force response is treated as a complex number

$$F^* = F' + iF'', \tag{3.5}$$

where F' is the real part with no phase shift and F'' is the complex part at 90° shift, δ results as

$$tan(\delta) = \frac{F''}{F'}. \tag{3.6}$$

In material testing, $\tan(\delta)$ is often the preferred dynamic mechanical parameter, as it directly tells which deformation phenomenon -elastic behavior indicated by F', or dissipation loss indicated by F''- dominates within the material.

In Figure 3.7, the hysteresis ellipsis present in phase 5 of the measurement routine is plotted schematically. The features of this hysteresis curve, the maximum displacement (ϵ_0), the maximum force (F_m), the force at maximum displacement (F_{ϵ_0}), and the force at zero displacement (F_l) are the basis for calculating the dynamic material parameters evaluated in this work: C_d follows as [104]:

$$C_d = \frac{F_m}{\epsilon_0}, \tag{3.7}$$

and $\tan(\delta)$ as:

$$\tan(\delta) = \frac{F_l}{F_{\epsilon 0}}. \tag{3.8}$$
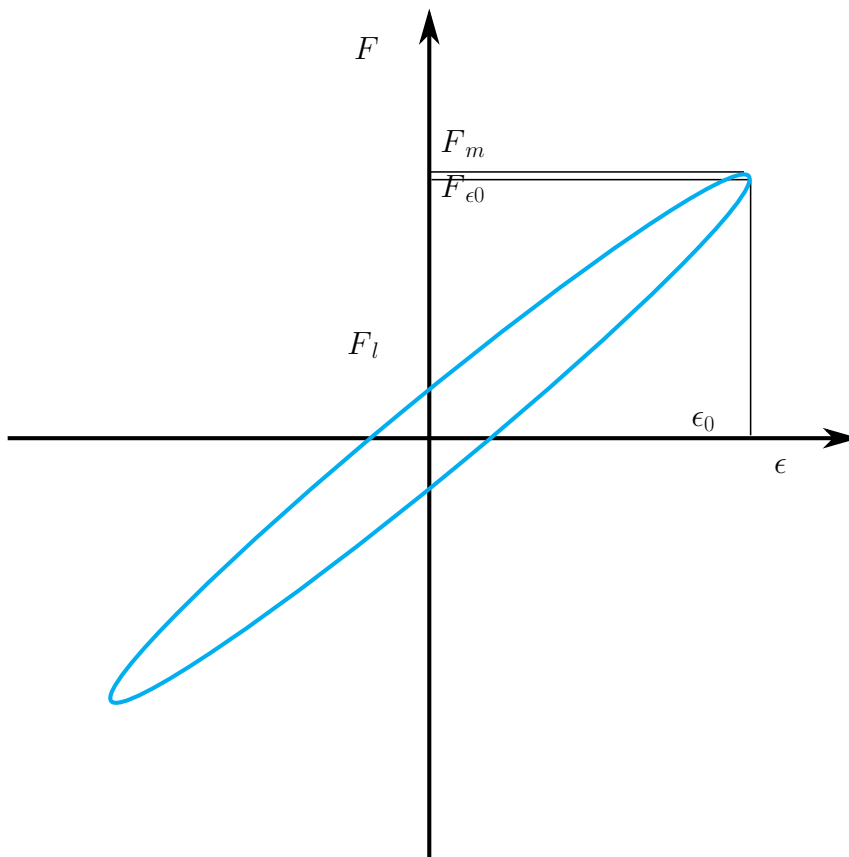
Figure 3.7: All features necessary to calculate dynamic material parameters can be determined from a F-ϵ plot [104].

3.3 On-line rubber temperature measurement

Determining accurate temperature values for the rubber during injection molding is a challenging but necessary task when the process should be optimized further. The mechanical properties of rubber parts -the dominating aspect of their quality- is mainly influenced by their degree of cure, with the curing time being primarly determined by the temperature of the material [56, 105–107]. Thus, there are many setups proposed for measuring the tempature during injection molding and even determining spatial and trainsient temperature distributions [42, 108–110]. In this work, two approaches were used to accurately measure the temperature of the rubber

in each processing stage and investigate the impact of process setpoints on the relative temperature changes.

3.3.1 Ultrasound based temperature measurement setup

The amount of heating by dissipation during the dosing phase of NBR rubber was measured by a setup based on ultrasound. This setup is adapted from Praher et. al., who used it to measure the tempeature of thermoplastic materials in the barrel of an injection molding machine and it is outlined in Figure 3.8 [108, 111]. Its core principle is the measurement of variances in the time of travel (t_t) of sound through a body of rubber, caused by different temperatures of the rubber. t_t is given by

$$t_t = \int_s \frac{\mathrm{d}s}{c + v_\perp},\tag{3.9}$$

where c denotes the velocity of sound in the medium it travels through and $v_\perp$ is the velocity of the medium flowing perpendicular to the distance s between the emitter and the reciever. c itself can be determined by

$$c = \left[\frac{1}{\rho\kappa}\right]^{\frac{1}{2}},\tag{3.10}$$

where ρ denotes the density of the medium and κ its adiabatic compressibility. $\rho = f(T,p)$ and $\kappa = f(T,p)$, thus if t_t,$v_\perp$, and p are known, the temperature T of the medium can be calculated.

While pvT data are available for the NBR compound (and thus κ is known, enabling calculations according to Equation 3.9 and Equation 3.10), calibration steps as shown in Figure 3.8 were done prior to the experiments to reduce errors and facilitate the temperature analysis.

3.3.2 Thermal imaging based measurement setup

The temperature of the rubber after passing through the nozzle was measured with a thermal imaging camera equipped with a macro lens. In the

a)

b)

c)

emitter

reciever
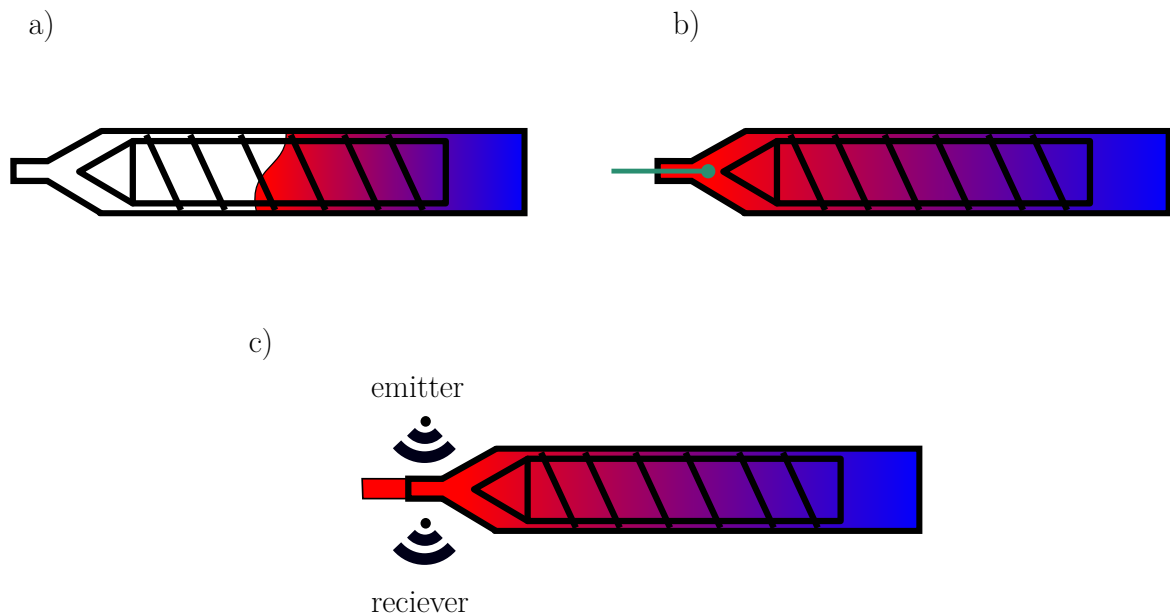
Figure 3.8: An ultrasound based setup determines the temperature of rubber after dosing. a) dosing with settings from the DoE (Table 5.1) b) calibration with thermal probe (marked green) c) ejection under constant pressure and measurement of sound travel time

experiments, two nozzles differing only in orifice diameter and angle of the reduction cone, as shown in Figure 3.9, were used.

The rubber strand was cut off after ejection into air and the temperature of the new cut surface at the orifice was measured by the thermal imaging system. The mean cross-section temperature was evaluated by processing the images with a custom made Matlab®(The MathWorks Inc., Natick MA, USA) script which also allows batch processing. The thermal imaging camera was a handheld system of the type RevealPRO (seek thermal, Santa Barbara CA, USA). The camera was fitted with a macro lens setup built according to Prutchi [112], as shown in Figure 3.10. Its main component is a plano-convex Germanium lens, which decreases the minimum focusing distance from about 31 cm to ≤ 1 cm. The entire setup was tested for its accuracy and precision with regards to common sources of error found in literature [113]. In particular, the temeperature of camera calibration tape
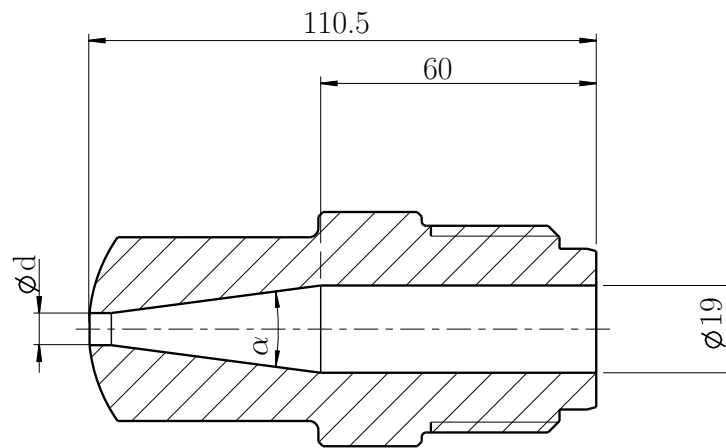
d in mm	3.5	7.0
α in °	15.0	19.5

Figure 3.9: The nozzles used to compare dissipation during injection only
differ in orifice diameter and angle of the reduction cone.

mounted on the calibration block of a thermocouple calibration unit Profi
Cal II (Gräff GmbH, Troisdorf, Germany) was determined with and without
the macro lens fitted at different angles to the surface.

Figure 3.10: A macro lens was fitted to a thermal imaging camera to allow the measurement of the cross section temperature of rubber strands 3.5 mm in diameter.

3.4 Injection molding equipment

In this section, a brief descriptions of the injection molding machines, the monitoring equipment and molds used for part and specimen manufacturing is given.

3.4.1 Injection molding machines

Injection molding experiments have been conducted unsing two different injection molding machines, which feature differing specifications. They are given in Table 3.2. Both machines are supplied by ENGEL Austria GmbH (Schwertberg, Austria) and feature hydraulically actuated clamping units and electrically actuated injection units. The injection units of both machines were reciprocating-screw types, where the screw performes a backwards motion during dosing and acting as a piston during injection [56]. Machine 1 is used for experiments in section 5.2, section 5.3, and section 6.2. All other injection molding experiments were conducted with machine 2.

Table 3.2: Two different injection molding machines were used for experiments

Machine number	Type	Clamping force in kN	maximum p_i in bar	Barrel diameter in mm
1	VC 940/130	130	2400	50
2	e-victory 740/220	220	2400	45

3.4.2 Monitoring equipment

The machine control software of both machines puts out cycle-to-cycle statistics of machine signals and signal time series data, the latter being used for process monitoring with PCA and FDA. A summary of all signals used in this work together with their source and machine availability is given in Table 3.3. Software of both injection molding machines allows the output of any variable of the entire system, however, for monitoring purposes, only non-controlled-material related variables are of use. For data structuring and pretreatment however, it is also preferential to output certain variables controlled by the closed-loop machine control.

Besides built-in capture methods, machine 1 was equipped with eight analog ports, which can be configured to output any process variable in a range from 0 V to 10 V. To enable online process monitoring, signals were fed to a PC via LucidControl (deciphe it GmbH, Kaufbeuren,Germany) USB IO modules acting as analog/digital converters. By means of a programming interface, signal values can be read from the modules by the monitoring system, which is implemented in the Python programming language.

The mold detailed in Figure 3.13 also features a pressure-temperature (p-T) sensor (shown in Figure 3.11) manufactured by Kistler Group (Winterthur, Switzerland). This sensor measures the in-mold pressure via a piezo electric crystal, while temperature readings are obtained with a thermocouple of type K. Both signals provided by this sensor are used as input signals for the process monitoring system when this mold is used on machine one. Machine
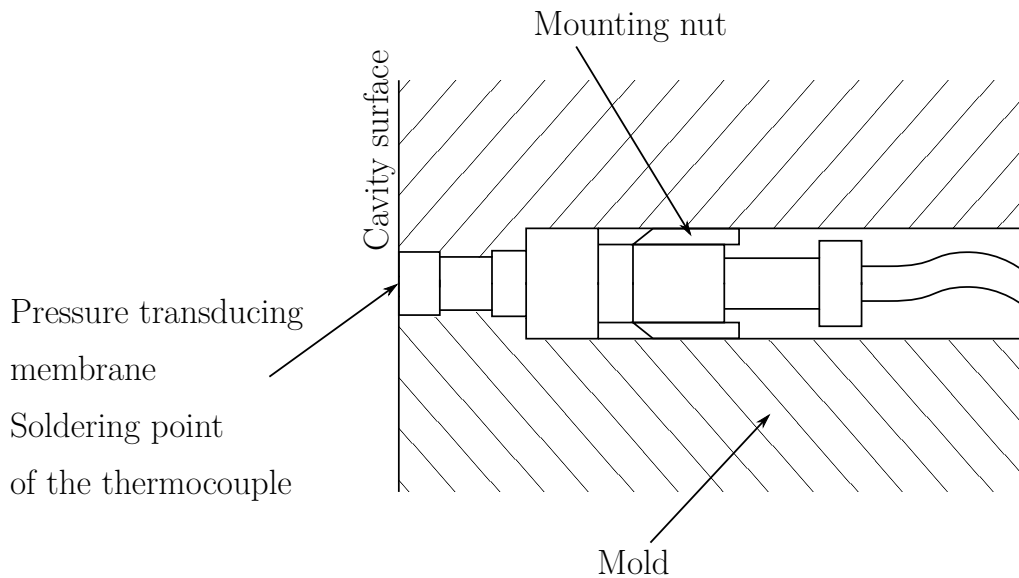
Figure 3.11: The Kistler pressure-temperature sensor mounted in a way that ensures direct contact with the rubber in the cavity. Pressure and temperatures are measured at the surface of the sensor. Modeled after [114]

2 did not feature analog output ports, allowing only network storage of process variable files. Thus, experiments conducted with machine 2 were only analyzed after they were finished. However, when process monitoring systems are implemented into machine control software, online monitoring would also be possible on machine 2. This injection molding machine is also equipped with ENGEL iQ weight control and iQ clamp control monitoring systems, which are part of the inject 4.0 suite of smart manufacturing systems [45] in addition to the signals available from the machine control of machine 1. iQ weight control is designed to keep part weights constant by detecting deviations in dosed volume and viscosity, consecutively adjusting the machine setpoints accordingly. It does so by comparing the injection pressure signal of the current cycle to a reference signal and calculating error values from this comparison [20]. The error values can then be used as inputs to the closed-loop control of the screw. iQ clamp control in its design is meant to work as a process optimization feature rather than being used for

59

Table 3.3: For monitoring and pretreatment, a number of process signals are essential

Signal	Usage	Source	Availability
Injection pressure (p_i)	monitoring	machine	machine 1 and 2
Torque (D)	monitoring	machine	machine 1 and 2
In-mold pressure (p_m)	monitoring	mold sensor	machine 1
Mold temperature (T_m)	monitoring	mold sensor	machine 1
Injection volume flow rate $(\dot{V}_i)$	pretreatment	machine	machine 1 and 2
Injection trigger signal (D_i)	pretreatment	machine	machine 1 and 2
iQ weight control	monitoring	machine	machine 2
iQ clamp control	monitoring	machine	machine 2

process monitoring. But, as was mentioned in section 1.3, in this work it is used for monitoring purposes, as it measures mold buoyancy. Buoyancy (F_b) is strongly dependent on the pressure in the cavity (p_m) and the projected area of the cavitys perpendicular to the opening axis (A_p) [46, 115].

Thus, the ability of the iQ clamp control signal trajectory to replace an in-mold pressure sensor for process monitoring is investigated in section 6.3.

3.4.3 Molds

All injection molded parts in this work have been manufactured with molds each used exclusively on machine 1 or 2 respectively. An overview of the main parts of the mold used on machine 1 is given in Figure 3.12, also the location of the cross-section detailed in Figure 3.13 is marked. A special feature of this mold is its movable jaw, which changes the direction of the opening motion to a vertical one. As most rubber injection molding setups feature vertical injection molding machines for easier handling, this mold emulates this type of part handling. The thermal control setup of this mold features both channels for temperature control with liquids and electrical heater pads made from ceramic.

Fixed jaw

cut for Figure 3.13

Movable platen

Part

Fixed platen

Movable jaw

Jaw guides

Feed bushing
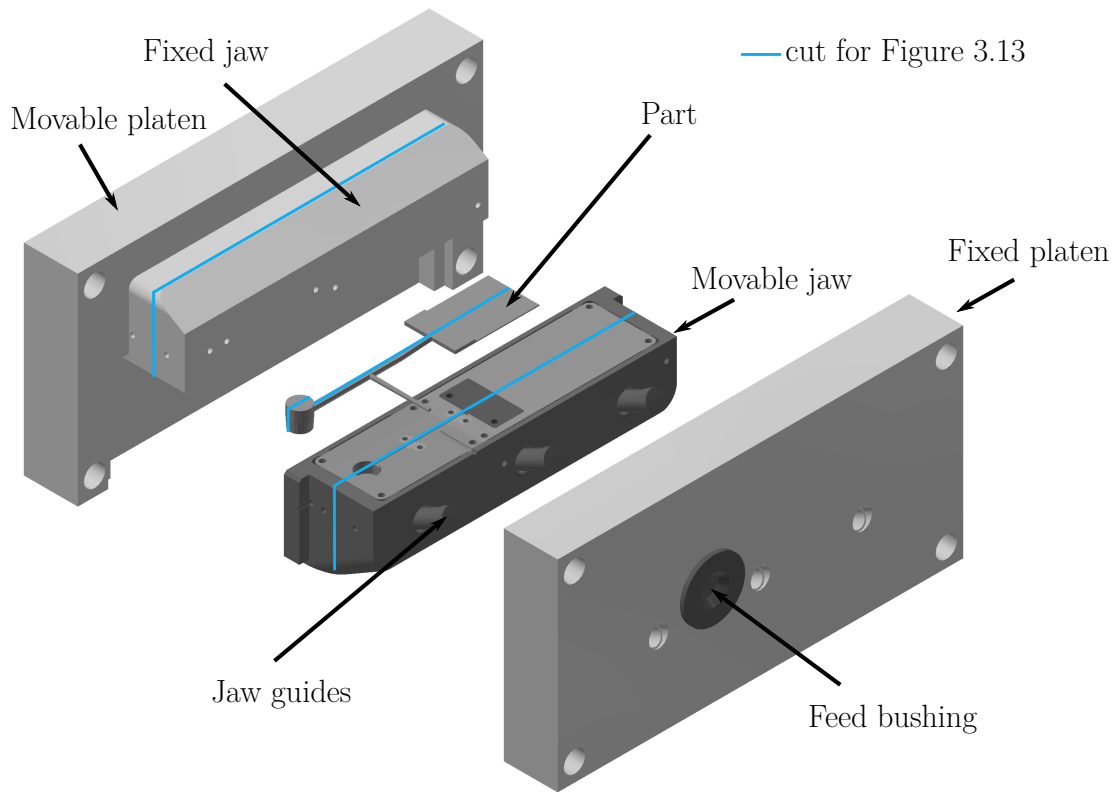
Figure 3.12: The mold used for experiments on machine 1 features a movable jaw which changes the opening direction to vertical, more in line with the majority of rubber injection molding setups

Figure 3.13 shows a cut through the closed jaws to outline the cooling channel and heater plate arrangement inside the mold block. With this setup, the mold is able to be dynamically temperature controlled: During filling, the heater pads are not active to prevent a premature start of the curing reaction and once the cavity is filled, the heater pad temperature can be set to temperatures higher than in conventional rubber injection molding, as no filling has to take place during curing. Each heater pad is equipped with a thermocouple for individual closed-loop control. All heaters are connected to a custom designed central control unit built by NET-automation GmbH (Zeltweg, Austria) utilizing an m-tron T control unit
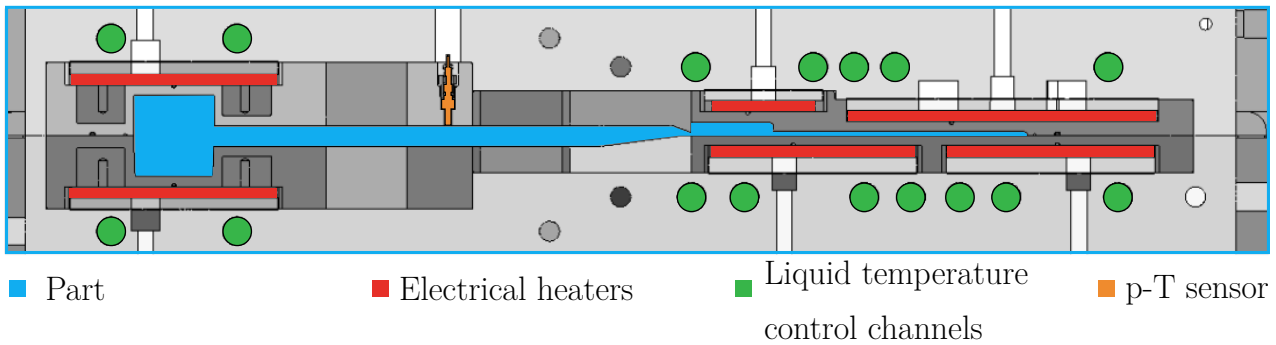
Figure 3.13: Electrical heaters and cooling channels are located close to the cavity to improve temperature control reliability

(JUMO Mess- und Regelgeräte GmbH, Wien, Austria). In the experiments in this work, the temperature of the ceramic heaters was kept constant at all times for method development. The possibilities and challenges of dynamic mold temperature control in rubber injection molding have been reported in [110]. The part which can be manufactured with this mold is detailed in Figure 3.14. A T-shaped runner leads to two cavities which can be blocked off individually, and for better control of the flow, the two different cavities are used mutually exclusive. For the experiments in this work, flat parts are manufactured, as they are better suited for monitoring and mechanical testing. Their advantage for monitoring experiments are the filling patterns, which do not exhibit jetting, causing random jetting related faults to overlap process monitoring results. They are also better suited for mechanical testing, as CS specimens can be easily extracted at the location marked in Figure 3.14 by cutting and the entire part can be mounted on the dynamic testing setup subsection 3.2.2.
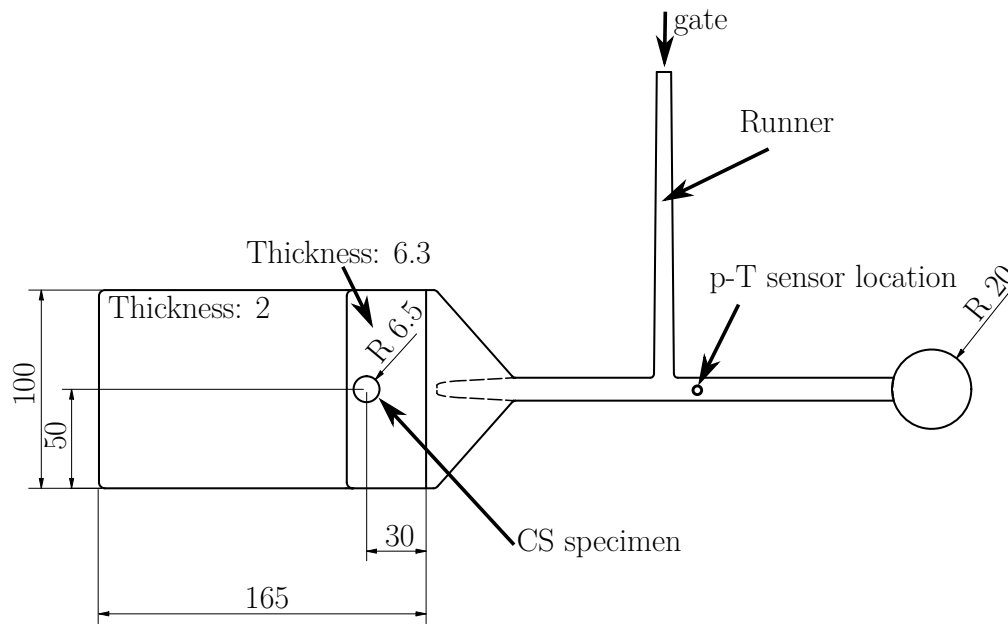
Figure 3.14: Parts manufactured with the dynamically temperature controlled mold are designed to enable easy specimen extraction

The wall thicknesses of the flat part are chosen to be 6.3 mm in order to be in line with the requirements of the standard for CS testing [99], and 2 mm to be in line with the standard for tensile testing [116]. Thus, no further cutting along the thickness of specimens needs to be done, reducing a common source of error.

The mold used for manufacturing parts with machine 2 is of much simpler design and it is outlined in Figure 3.15 a). Most notable, its parting plane and larges part dimension is parallel to the mounting platens of the machine. Thus, the clamping force and buoyancy act in the same direction, which is needed for the iQ clamp control sensors to measure correctly. Further increasing the projected part area of this mold is its double-cavity design, with two identical cavitys for manufacturing flat parts similar to the parts manufactured with the dynamically temperature controlled mold. The dimensions of the parts are given in Figure 3.15 b). While the footprint of the part was decreased in comparison to the part design of the dynamically

a)

b)
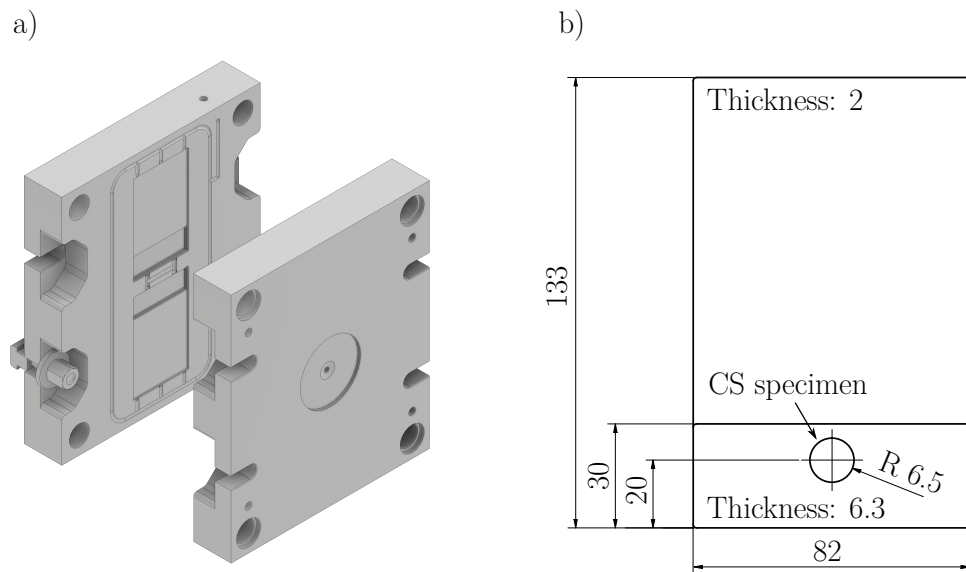
Figure 3.15: The cavities of the mold are arranged in a way to maximize the projected area of the parts to increase the resolution of the iQ clamp control signal

temperature controlled mold, the thicknesses were kept to allow for equally straightforward testing, and the spot for dynamic testing and CS specimen extraction retained its relative position on the part. Lastly, this mold does not feature active temperature control elements but rather is designed to be mounted on generic electrical heater plates which are mounted on machine 2 prior to the mold being installed. Due to its uniform shape and relatively narrow construction, good temperature control and uniformity is still expected from this setup.

4 Non-destructive dynamic part testing for part quality monitoring

THE dynamic testing setup detailed in subsection 3.2.2 offers an alternative to CS tests for quality control of injection molded rubber parts. It offers two key advantages over commonly used methods to determine the dynamic properties of cured rubber parts such as CS testing, Dynamic mechanical analysis (DMA), or Differential Scanning Calorimetry (DSC). First, it eliminates the need for extracting specimens from parts, facilitating measurement preparation and startup. By being destruction-free it could also be incorporated into quality control routines in serial production. Second, the testing time is only around 240 s compared to the multiple hours that are needed for CS testing, or around one hour, as is common in DMA testing. This allows, although only with further optimization, to implement a 100 % testing strategy for suitable injection molded rubber parts. As of now, the methods most prominent disadvantage is that it is not a standardized testing setup. Yet, similar tests have been performed by Kerschbaumer et al., which showed good correlation of the measured parameters with the degree of cure [100] and also CS values, which are the most wide spread means of quality control in rubber injection molding. Furthermore, isothermal cyclic dynamic tests have been reported to be a viable mean of determining dynamic rubber part properties [56, 104, 117].

4.1 Method evaluation

In the experiments of this section, the dynamic testing method is validated and its ability to provide material quality data for use with multivariate statistical process monitoring is evaluated.

Table 4.1: Sample parts for dynamic tests were cured at 160 °C at nine different t_c.

t_c in s	120	150	180	210	240	270	300	600	900

For method validation, sample parts were manufactured from NBR by compression molding. Compression molding was chosen instead of injection molding for verifying the dynamic testing method, since it minimizes the probability of manufacturing-related effects of part performance such as molecular orientations or mold temperature fluctuations. The geometry of the compression molded plate-shaped parts is outlined in Figure 3.5. The temperature of the mold was kept constant at 160 °C, and the t_c was set in nine steps from 120 s to 900 s, shown in Table 4.1. All samples were quenched in ice water after demolding to minimize post curing, which would increase the degree of cure of the samples in an uncontrolled way, possible reducing the response of the tests. Comparing the set cure times with the curing kinetics of this NBR grade (section 3.1, page 43), it can be seen that the range should span from a degree of cure which is just enough for demolding to almost fully cured. At each setpoint, 6 sample parts were manufactured, three of which to be tested by the dynamic setup and three to be CS tested. CS values were used for verification, as its correlation with the degree of cure is well-established [100, 118–120].

The first question to be raised is how many parameters are needed to capture the curing process in the parts sufficiently. In Figure 4.1 a), every parameter calculated from the dynamic testing setup is normalized from 0 to 1 and plotted over the cure time. It is clear that every parameter responds in a nonlinear way to the increase in cure time. This is to be expected, as the curing reaction is not linear in time domain. Also, all parameters related to the elastic behavior of the rubber are increasing with cure time, while $\tan(\delta)$, which is the ratio of viscous to elastic behavior (Equation 3.8, autopagerefeq:tand), decreases. Also, Figure 4.1 a) shows that the measurement scattering is higher at lower cure times. The reason
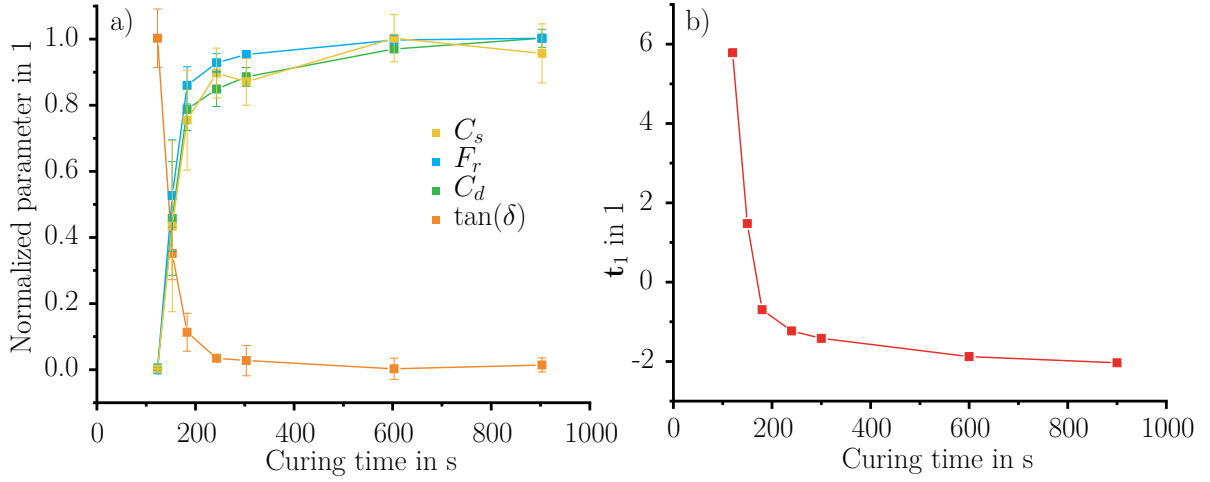
Figure 4.1: The degree of cure of the parts can be detected a) by every parameter calculated from the dynamic tests b) by PCA.

for this can be found in the curing kinetics of sulphur cured rubbers. At the lower curing time and thus low reaction conversion rate, the reaction rate is much higher, amplifying any timing inconsistencies in sample handling or temperature variation during quenching. As all parameters show similar response shapes, PCA was performed to reduce the dimensions of this multivariate problem.

As shown in Equation 2.15 to Equation 2.18, PCA takes a high number of input dimensions and encodes them in a new space, which is optimal in the information captured in each dimension. Then, by evaluating the magnitude of the eigenvalue (λ) in Equation 2.19, the number of dimensions absolutely needed to avoid losing information can be determined. In this case, retaining only one dimension proved to be sufficient, which is equivalent to stating that there is no need for a combination of parameters to measure differences in the degree of cure of the sample. This is underlined by plotting the first score vector $\mathbf{t}_1$ over time in Figure 4.1 b), which shows the same kind of response as all other parameters. It is preferable to use a parameter originally determined from the tests, instead of the more abstract PCA
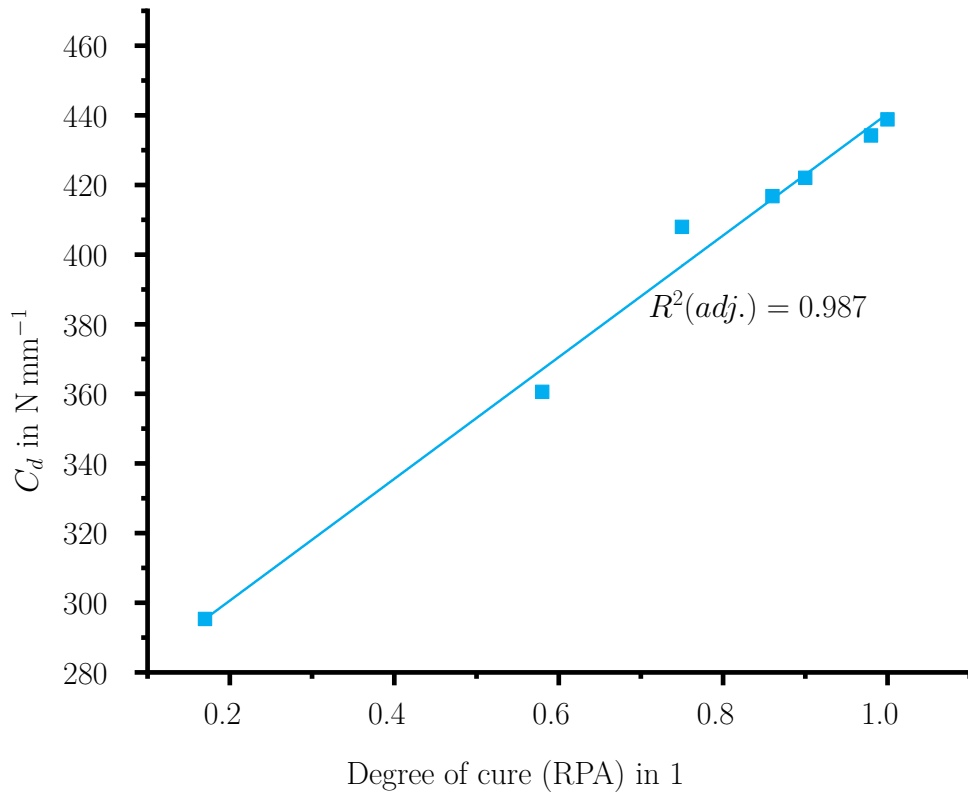
Figure 4.2: C_d linearly fits the degree of cure obtained from RPA measurements with a high coefficient of determination.

score value $\mathbf{t}_1$. Ideally, the parameter which correlates best with the degree of cure is chosen. To find out which material parameter best correlates linearly with the degree of cure of the part, mechanical parameter values are compared with the degree of cure obtained from RPA measurements (section 3.1) at corresponding times. To do so, the adjusted Coefficient of determination (R^2) of all material parameters evaluated in this work is given in Table 4.2. It can be seen that among the obtained parameters, C_d obtains the highest score, and it is plotted against the degree of cure obtained from RPA measurements in Figure 4.2. As a result, it will be used in further experiments as the sole reported mechanical material parameter.

In further experiments, the part geometry will differ strongly from the compression molded samples used for method evaluation. Especially, while

Table 4.2: The dynamic spring stiffness linearly correlates best with the degree of cure from RPA measurements compared to other material parameters

Parameter	$R^2(adj.)$
C_s	0.979
F_r	0.971
C_d	0.987
$\tan(\delta)$	0.941
CS	0.902

the compression molded plates are circular and their borders are concentric to the setup's test probe, the outline of parts manufactured with the dynamic temperature controlled mold (Figure 3.12, page 61) is rectangular. Thus, waves propagated from the tested area to the edges are not reflected evenly. As shown in Figure 3.5, specimens were cut from the center of the compression molded plates, one with rectangular edges to match the thicker area of the injection molded parts as close as possible. Another type of specimen was cut with the equipment used for CS measurements. The C_d values measured from each of the specimen geometries are depicted in Figure 4.3. It can be seen that C_d of cuboid specimens is equal to the circular plates. This hints that there is no noticable measurement disturbance by waves reflected at the specimen borders, and part dimensions can be seen as infinite compared to the measured area. The dynamic stiffness of much smaller compression set style specimens is much lower and scattering is higher, while the shape of the response curve remains similar. As no surrounding rubber blocks the measured area of the latter specimens, the compressed material is able to bulge outward, reducing the resistance to compression. As the dynamic tests are only used for relative comparisons to a self-established baseline, specimen geometry is not a primary concern for method applicability.
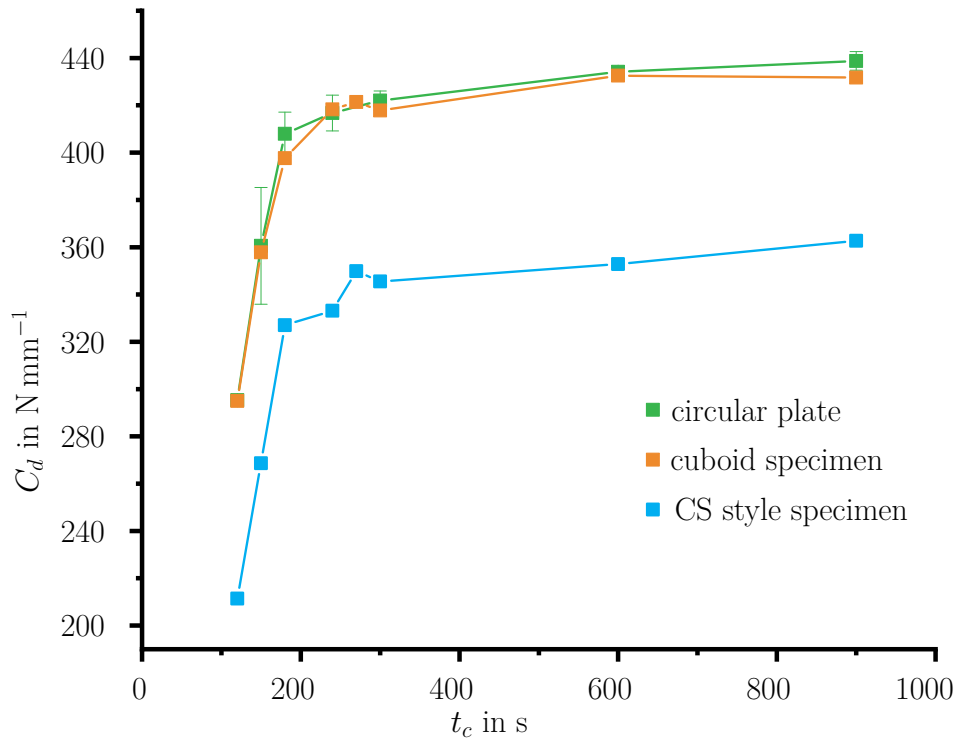
Figure 4.3: C_d is almost equal between complete compression molded plates and cuboids cut from them. As no surrounding rubber supports CS style specimens, their C_d is lower.

4.2 Conclusions to dynamic method building

The investigations detailed in chapter 4 aimed to develop a setup, that is suitable for measuring dynamic properties of injection molded rubber parts at a pace similar to the manufacturing process. Such a test setup is needed to obtain readings of part quality during serial production to assist the development of a fault detection system. The results presented in this chapter indicate that, in a controlled environment, the custom setup is able to measure the relative degree of cure accurately and with high repeatability. This setup is also able to be incorporated into production lines, as it does not require specimen extraction. Furthermore, it operates at room temperature, which eliminates the need to inserting parts into closed-off tempering cells in contrast to DSC and DMA, where specimen

temperature control is an important source of operator error, increasing the ease of handling dramatically [56, 104]. Therefore, this setup should be well suited for quality control of process monitoring experiments, as it is able to resolve the dynamic part behavior precise enough to not oversee any effects of process fluctuations induced by scattering from the testing method itself. Furthermore, compared to the setup presented by Kerschbaumer, no specialized testing device is needed, and the method should be able to be transferred to other general-purpose dynamic testing machines in a straightforward fashion [100].

5 Measuring dissipation heating of rubber in all injection molding stages

DURING dosing, when flowing through the nozzle and the runner of the mold, the rubber temperature increases not only due to thermal conduction, but also due to dissipation. As was detailed in chapter 1, the time and temperature of each injection molding phase contributes to the cumulative energy input into the rubber during injection molding (see Figure 1.3). Hence, variations in the energy input throughout the rubber injection molding process could cause critical deviations of the product's actual degree of cure to the required degree of cure. This has led to efforts being made to measure and control the amount of energy input in these processing stages [11, 60, 63, 121, 122]. However, it is surprisingly challenging to remove sources of error such as the temperature of the surrounding steel from the temperature readings. Furthermore, when thermocouples are used, of which the thermal capacity is not neglectable and measurement times are short, the contact temperature is determined rather than the temperature of surface which is of relevance to the investigations.

In this chapter, the focus is set on determining the energy input into the rubber prior to curing in a more exact way than previously done. This shall be achieved by applying a number of methods designed to deliver accurate readings of the actual rubber temperature in every stage of the injection molding process. The aim is creating a better understanding of how process setpoints and fluctuations impact the energy input into the rubber, aiding the operator in setting up more stable processes. Also, the development of process monitoring methods in chapter 6 is facilitated, as the effect of artificially introduced process faults on the rubber temperature can be determined with increased accuracy. Thus, the magnitude of the

monitoring system's response to changes in the state of the rubber can be judged more accurately.

5.1 Rubber temperature measurements by ultrasound

During the dosing phase, in addition to the rubber being heated by thermal conduction through the barrel walls, the rotating screw causes shear dissipation in the rubber as it is conveyed into the screw antechamber [115]. Obtaining the correct mean temperature of the rubber after dosing, but before injection, continues to be challenging due to the enclosed nature of the screw antechamber. The ultrasound-based method used in the present experiments mitigates limitations of the more common method of inserting thermocouple probes radially or axially. It does not influence the rubber temperature by thermal conduction along the probe and it provides mean cross-sectional temperature values instead of punctual radial measurements. Thus, the impact of storage maturation and dosing conditions on the heating of the rubber caused by dissipation could be investigated with an accuracy that has not been reported outside of what has been published in conjunction with this work [110].

The experiments follow a DoE that was designed to investigate three factor levels of v_s and two for the conditions, in which the NBR compund was stored prior to the investigations. Values for the most important fixed experimental process settings are shown in Table 5.1. Figure 5.1 a) and b) show that for the NBR compound, increasing v_s during dosing also increases the mean temperature of the rubber. The ultrasound based method is also able to show that rubber temperature is not homogenous along the longitudinal axis of the screw antechamber. Instead a characteristic temperature profile occurs. Such a temperature profile is intrinsic to the operation principle of reciprocating screw injection units (section 3.4): From close to the nozzle (at 75 mm stroke) to about 30 mm the number of turns of the screw the rubber has to go through increases, but as the screw moves back during dosing, the effective screw length (number of turns between the feed and
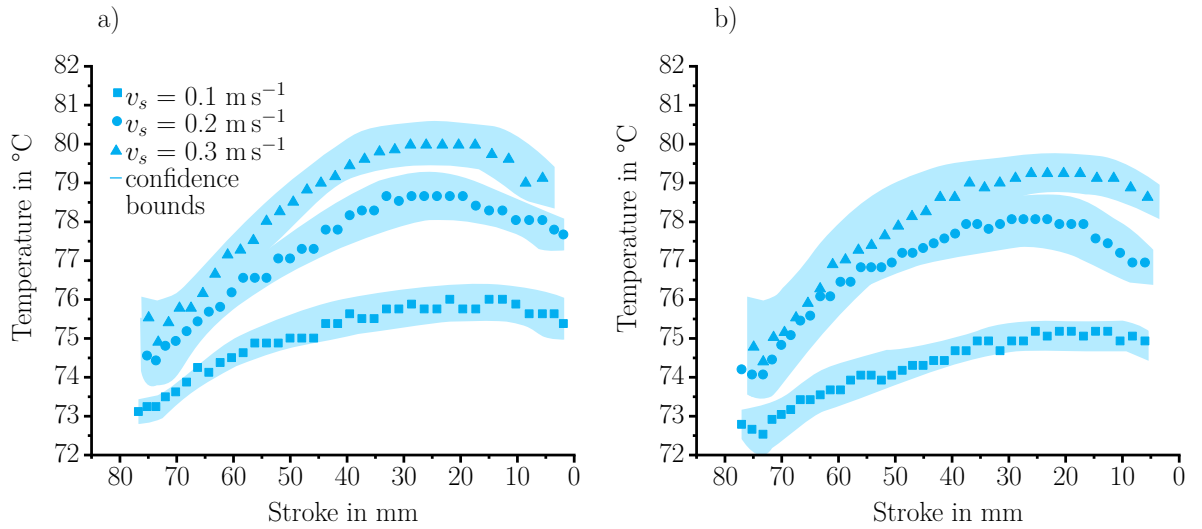
Figure 5.1: A higher v_s increases the mean temperature of NBR in the screw antechamber and leads to a higher curtosis of the axial temperature profile both for a) NBR stored at 5 °C and b) NBR stored at RT [123].

the screw tip) decreases, causing falling temperatures between 30 mm and 0 mm at the nozzle tip. However, the amplitude of the temperature profiles in Figure 5.1 a) and b) remains in the range of under 6 K, and no effect of the axial temperature profile on process stability could be proven in successive experiments chapter 6.

The mean temperature of the NBR stored at RT (Figure 5.1 b)) is lower at the same settings than the temperature of the NBR stored at 5 °C. This may seem contrary to the increase in viscosity, that is detected by RPA measurements in rubber stored at RT, which are given in section 3.1. However, an increased viscosity also causes less pressure induced backflow during dosing at the same back pressure settings, and thus reduces dosing time and dissipation heating, as shown in Figure 5.2.

Table 5.1: DoE for determining the rubber temperature with ultrasound

Rubber compound	NBR		
Barrel inner diameter in mm	35		
Barrel temperature in °C	80		
Back pressure in bar	100		
Ejection pressure in bar	600		
Cycle time in s	600		
Circumferential screw speed in $\mathrm{m\,s^{-1}}$	0.1	0.2	0.3
Storage condition	5 °C	1 month RT	
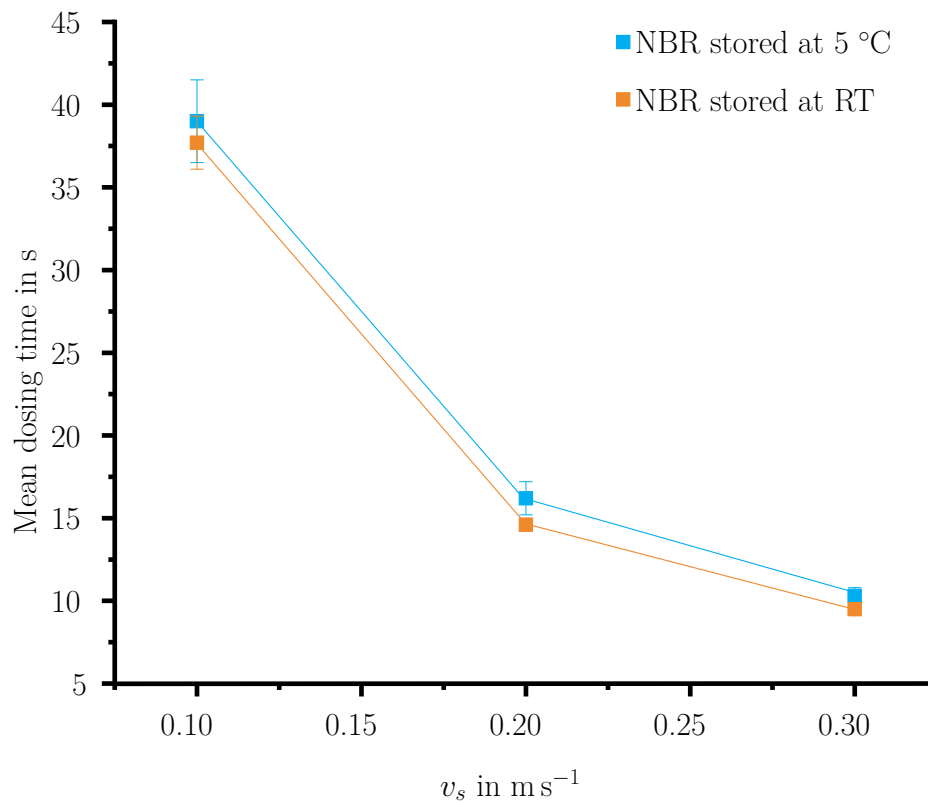
Figure 5.2: Dosing time decreases at every setpoint of v_s when NBR is stored at RT compared to NBR stored at 5 °C

5.2 Rubber temperature measurements by thermal imaging

After the dosing phase, the rubber is injected into the mold, where it passes through the nozzle of the injection unit. Most industrial nozzles feature a contraction zone, where the flow diameter is reduced, so the rubber is not only subject to shear but also elongation and compression forces. As a result, the work input of the axial screw movement is not only translated into motion, but also causes material heating due to dissipation [41, 59, 121, 122, 124, 125] Using the thermographic setup presented in subsection 3.3.2, the temperature of NBR at the orifice was determined. Again, processing parameters were set to different factor levels, shown in Table 5.2, to highlight the impact of the processing conditions on the heating of the rubber. Lastly, effects were analyzed for two levels of storage maturation of the NBR in use.

Figure 5.3 shows the temperature of NBR at the orifice which is 7 mm in diameter and it is clear that the barrel temperature has by far the most impact on the measured temperature. Especially notable is that while at $T_b = 60$ °C, temperatures range from 66 °C to around 70 °C, which is higher than the barrel temperature. At $T_b = 80$ °C, measured temperature values scatter close to the setpoint and at $T_b = 100$ °C, temperatures are consistently lower than 100 °C. As already shown in Figure 5.1, the temperature of NBR in the screw antechamber is lower than the T_b for cycle times of 600 s (which was set in both experiments).

When conditions are transferable, it can be seen that the effect of increasing the temperature of NBR by injection is reduced with increasing T_b. Furthermore, for every setting of T_b, Figure 5.3 shows a plateau behavior of the temperature when increasing the injection volume flow beyond 20 cm³ s⁻¹. This is in good accordance with general flow theories for non-newtonian fluids and highly filled fluids (section 2.1) and measurements found in literature [41, 43, 69, 73, 74, 79, 80, 126]. Here, increasing injection volume flow rate ($\dot{V}_i$) mainly affects shear dissipation in the outer layers of the rubber , while the plug section shown in Figure 2.2 is not affected

Table 5.2: DoE settings for determining the rubber temperature with thermal imaging

Rubber compound	NBR			
Screw speed in $\mathrm{m\,s^{-1}}$	0.1			
Cycle time in s	600			
Orifice diameter in mm	3.5	7		
Barrel temperature in °C	60	80	100	
Volume flow in $\mathrm{cm^3\,s^{-1}}$	5	12.5	20	40
storage maturation	refrigerator	1 month RT		
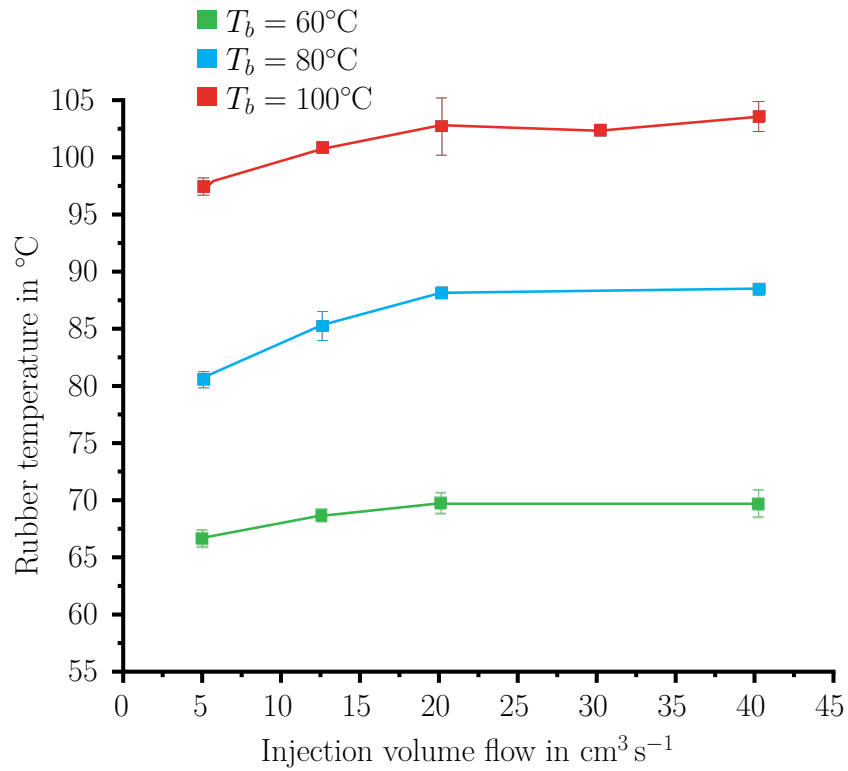
Figure 5.3: The temperature of NBR after ejection can be increased by increasing both the barrel temperature (T_b) and the injection volume flow.
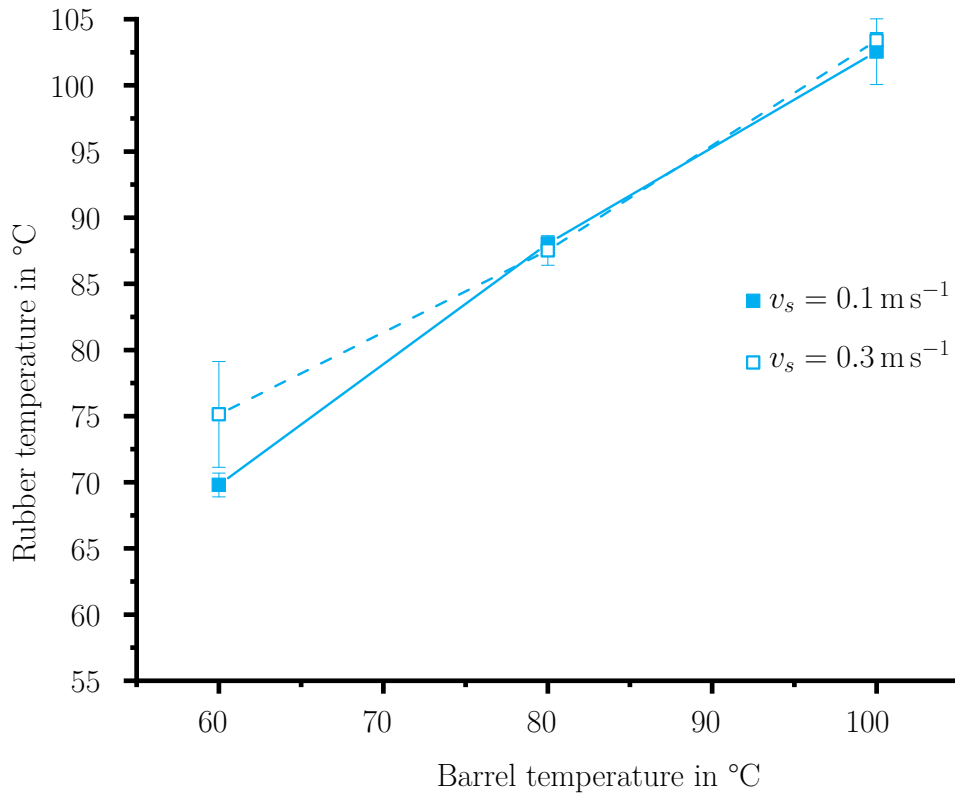
Figure 5.4: A change in v_s in the dosing phase does not impact the temperature of the rubber at the nozzle orifice.

by shear. A temperature increase in the plug section is only possible by dissipation through elongation flow.

In Figure 5.1, the impact of changing the setting of v_s on the temperature in the screw antechamber can clearly be seen. If this temperature difference can be carried into the mold, it could prove valuable for controlling fluctuations. In Figure 5.4, the impact of v_s on the temperature at the orifice was investigated for an orifice diameter of $7\,\mathrm{mm}$ and two $\dot{V_i}$ setpoints. One can clearly see that for $T_b > 80\,°\mathrm{C}$, any temperature increase caused by increasing v_s has vanished. Flowing through the nozzle has a smoothing effect on the rubber temperature, as warmer rubber is subject to less dissipation due to its lower viscosity and the opposite is true for cooler rubber. Unfortunately, this means that smaller temperature differences introduced during dosing

will not be present any more as the rubber reaches the cavity, as they are overpowered by dissipation occuring in the nozzle and the runner.

Further evaluation of the factors controlling the NBR temperature at the nozzle orifice was done via three-way Analysis of Variance (ANOVA). Means were compared pairwise by Tukey's test at a significance level of 95 %. Figure 5.5 shows the effect of each factor set in the DoE. It can be seen that at a T_b of 100 °C, each effect is not statistically significant. First, this means that adjusting the injection volume flow to counter quality fluctuations is only viable at lower T_b settings. However, influences such as different levels of storage maturation also do not alter the temperature of the rubber greatly, which can facilitate setting up more stable processes.

The impact of storage maturation on scorch behavior was not investigated here, but it was found that sulphur curing NBR can show reduced scorch time when stored at RT for longer periods of time [6]. Thus, while higher T_b settings can help stabilize the rubber injection molding process, it also limits the meaningful possibilities to (automatically or manually) adapt the process setpoints to counter unwanted quality fluctuations.
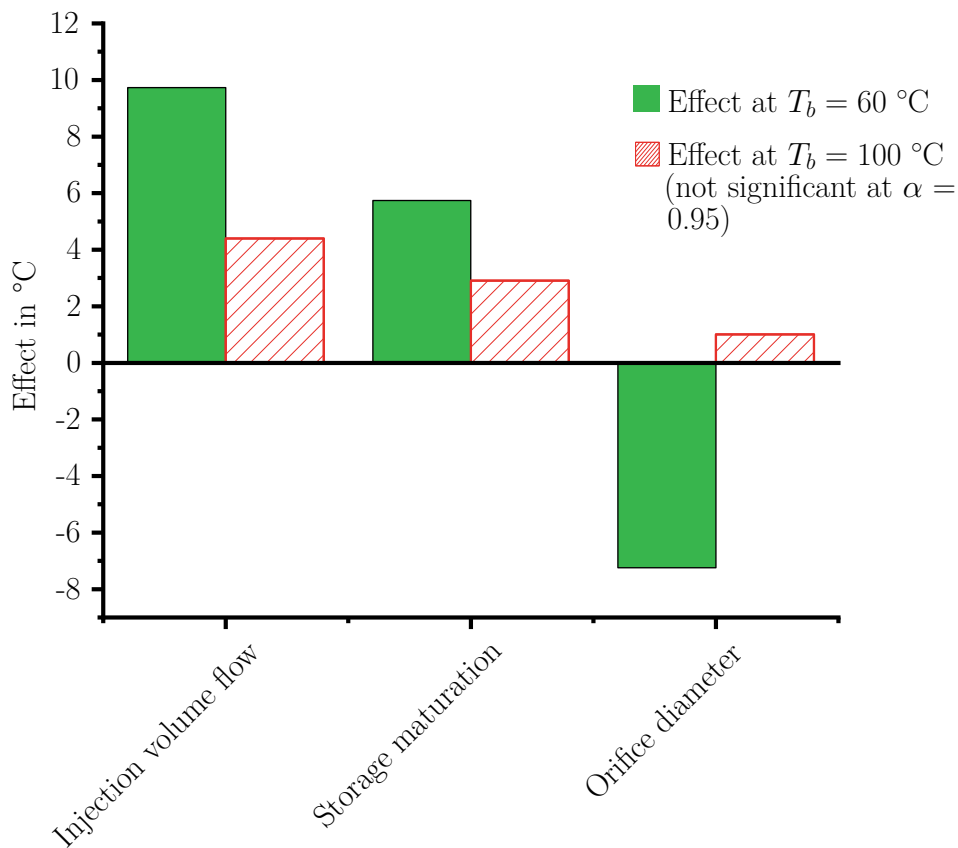
Figure 5.5: Injection volume flow, storage maturation and orifice diameter significantly impact the temperature after ejection only at $T_b = 60$ °C.

5.3 Rubber temperature estimation in the mold

When being injected, the rubber flows through the runner system which directs it into the cavity or cavities. As mentioned in chapter 3, the mold used in all experiments is equipped with a combined pressure and temperature sensor in the runner. Thus, it is possible to set the results of subsection 3.3.1 and section 5.2 in context with the temperatures measured at the in-mold sensor. First, the shape of the sensor signal and features derived from it need to be discussed. Figure 5.6 shows a temperature signal recorded during an injection molding cycle with NBR rubber.

The graph shows that the rubber touches the sensor at about 60 s, and
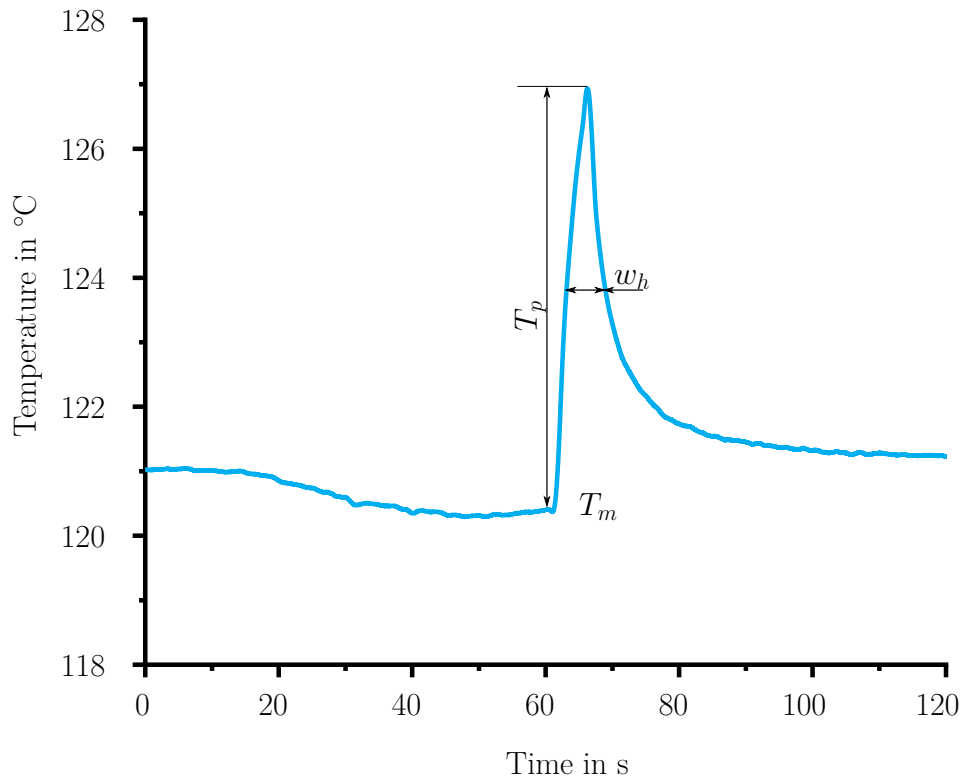
Figure 5.6: The temperature peak height (T_p) -with regards to the mold temperature (T_m)- and the width at half height (w_h) can be determined from the in-mold temperature sensor signal of each cycle. The data shown is gathered by injecting NBR with $\dot{V}_i$ set to 20 cm^3 s^{-1}, T_b to 100 °C, and the temperature of the fluid to 150 °C

before, the temperature measured is that of the steel surrounding the sensor. As the sensor is located in a dead end (Figure 3.14), no further rubber flows past the sensor, causing the surrounding volume of the sensor(rubber and steel) to cool down to T_m over time. Two characteristic features of the peak, the T_p and w_h can thus be evaluated. T_p, in this case is closely related to the contact temperature and to the temperature of the outer layers of the rubber while w_h is a better measure for the thermal energy of the rubber volume and thus more in line with its mean temperature.

This statement is substantiated by the plots in Figure 5.7. Plot a) shows
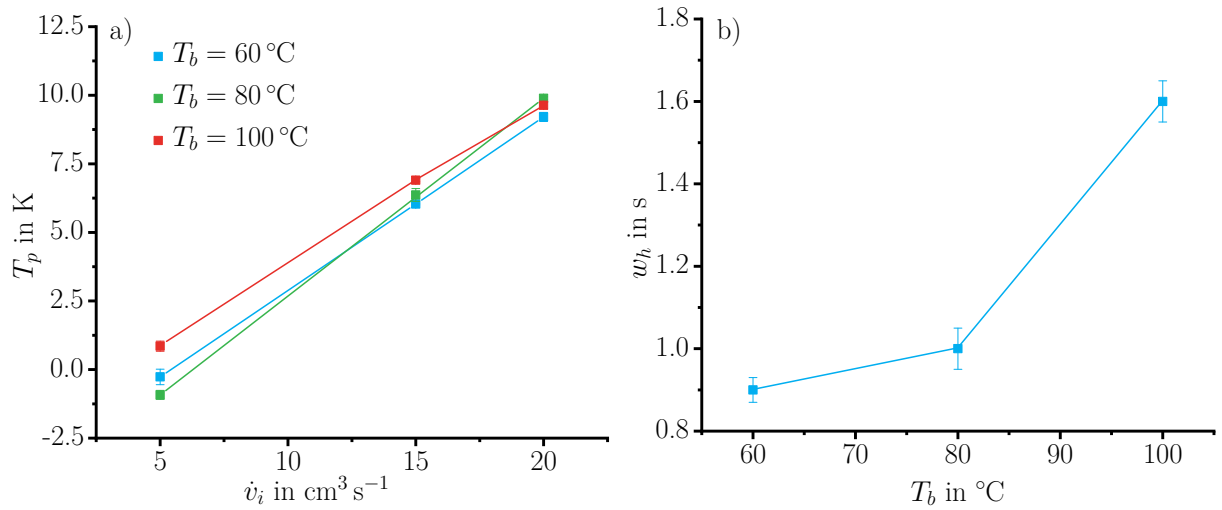
Figure 5.7: a) While T_p is increasing with increasing $\dot{V}_i$, the T_b does not greatly affect T_p. b) Increasing the T_b broadens w_h greatly.

that T_p is highly dependent on $\dot{V}_i$ but not on T_b. In other words, it is a function of shear dissipation but not the mean temperature of the rubber. In contrast, plot b) shows w_h is not a function of $\dot{V}_i$, but only of T_b, which is the most important influence on the mean temperature of the rubber in injection molding (Figure 5.3).

It is notable, that in Figure 5.3 no measured temperature is higher than the mold temperature (T_m), but T_p can be as high as $10\,\mathrm{K}$. Thus some effects must have occured, which raise the contact temperature well above T_m.

5.4 Conclusions

As mentioned at the beginning of this chapter, being able to track the temperature evolution of rubber as it is injection molded can provide valuable insights into the possibilities and limits of automatic process control. Such a process monitoring system could use data collected from these experiments to adapt process setpoints to ensure constant part quality even in a changing

process environments. However, as shown in Figure 5.1, and Figure 5.4, temperature differences lower than $10\,\mathrm{K}$ achieved by altering the dosing speed are not carried through the nozzle. In other words, actively controlling the rubber temperature by altering the setpoint of v_s may not be the best lever to control part quality.

Furthermore, Figure 5.6 shows that in the mold, the effect of T_b on the bulk temperature of NBR is still present, while in the outer layers, $\dot{V}_i$ is the only temperature controlling factor. This renders it extremely challenging to balance detected fluctuations in the process with controlling the temperature by injection unit settings. Perko showed, that by using runners with conical or hyperbolic flow path crosssections, the temperature of the rubber could be increased significantly across the entire diameter [59]. By incorporating such runners in mold designs, the lever of injection settings on part quality can be greatly increased.

At this point, there is no known way to reliably measure the temperature of a rubber part just when filling is completed, let alone resolve the cross-sectional temperature profile. Furthermore, only set-point changes which show quick response are suitable for adaptive control, which is not true for T_b, as will be discussed in chapter 6. To conclude, options to control the rubber temperature with fast responding process setpoint changes are very limited. Thus, being able to monitor the process, detect fluctuations and automatically detect the type of fault is the best case scenario that can be achieved for now.

6 Monitoring the injection molding process

EXPERIMENTS presented in this work up to this point aimed at closely examining the effect of manufacturing conditions and property fluctuations of the uncured rubber on the quality of vulcanized rubber parts. Knowledge of these effects is important to achieve the primary goal of this work, which is to develop a process monitoring system for rubber injection molding capable of detecting such fluctuations during manufacturing lots spanning extended periods of time. A number of approaches to perform process monitoring of thermoplastic injection molding or other industrial processes have been discussed in section 1.3. Among those, data-driven approaches present the greatest potential for industrial applications, as they do not require any testing or model tuning prior to the start of production. However, for rubber injection molding, no data-driven SPM approaches have been developed as of yet.

In chapter 2, an approach relying on PCA and FDA to perform multivariate statistical process monitoring has been presented. To test this method on its response to fluctuations peculiar to rubber injection molding, in this chapter, extensive injection molding experiments were conducted. These experiments were set up similar to what is typical on the shop floor of rubber part manufacturers, and artificial sources of fluctuations were introduced to the experiment runs, such as changing the mold temperature (T_m), the barrel temperature (T_b), or feeding a rubber compound with differing storage maturation.

6.1 SPM method schematics

The flow chart of Figure 6.1 displays the routine followed in all monitoring experiments: When the manufacturing process is started, process signals (variables), for instance the injection pressure (p_i) or dosing torque torque (D) are captured as trajectories with K sampling points, and the cycle number they belong to is used as a tag. When the process has reached stable operation and the quality of the parts is deemed to be OK by visual inspection, the cycle number is marked to start the training. Multiple cycles, 15 cycles in these experiments, need to be recorded to form a full training set. When the training set is completed, the data are structured by PCA and parameters essential for process monitoring are calculated. More specifically, the parameters are:

1. The representative phase loading matrix (P_c) of each process phase is calculated according to Equation 2.29.

2. The control bound of the T^2 statistic T^2_t as in Equation 2.32.

3. The control bound of the SPE statistic $\text{SPE}_c(k)$ for every sampling point k according to Equation 2.33.

With these parameters, the subsequent cycles can be monitored on-line. For the full trajectories of all captured signals, the statistics SPE_{new} and T^2_{new} are calculated. They are then compared to the respective control bounds by applying the decision logic of Equation 2.35 and Equation 2.36 to detect critical deviations of the process stability from the training data. Also, at this stage, FDA can be performed to identify the type of deviation or fault that occured. However, since FDA is a supervised learning method, the fault types need to be labelled before training this method. Due to constraints of the lab setup used for model development, labelling could not be done during monitoring and FDA classification was calculated off-line.
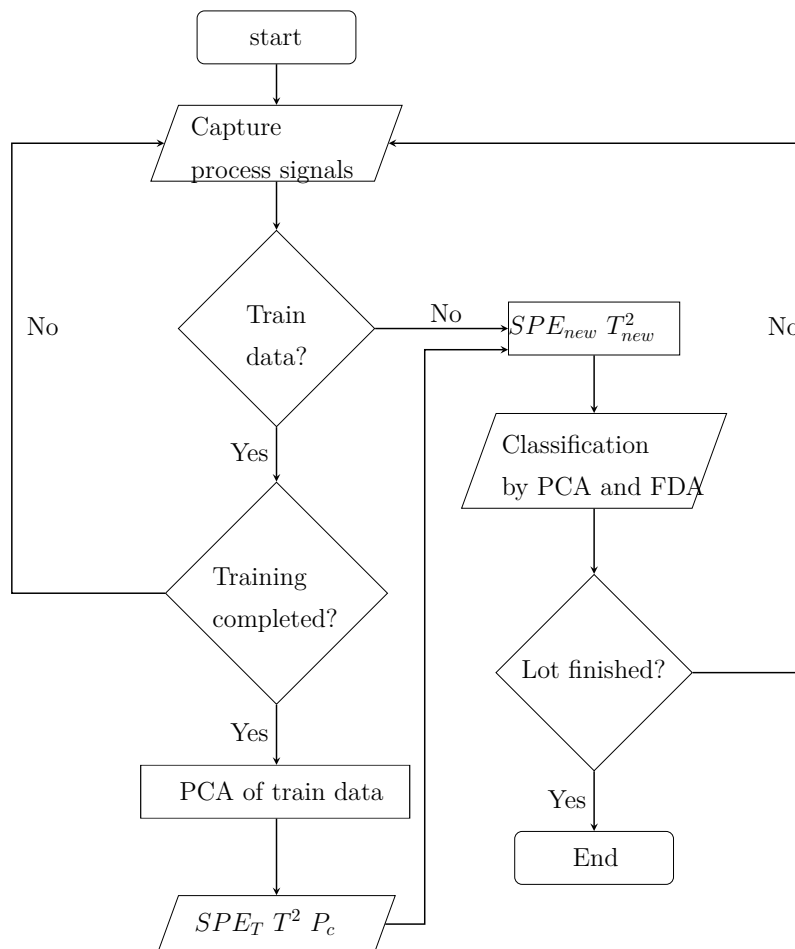
Figure 6.1: The process monitoring approach follows a straight forward decision logic.

6.2 Baseline experiments for PCA-based process monitoring

To create an understanding how the multivariate statistical monitoring methods perform when they are used to detect faults in a rubber injection molding setup, baseline experiments need to be performed. There, reliable training data have to be generated by manufacturing NBR parts which are deemed to be of target quality regarding their state of cure, their dimensions and mechanical properties. To do this, an operating point for the process

Table 6.1: Operating point of the training data generated in first experiments to test PCA based process monitoring

Process parameter	Set point
Curing time (t_c) in s	254
Mold temperature (T_m) in °C	160
Barrel temperature (T_b) in °C	80
Cirumferential screw speed (v_s) in $\mathrm{m\,s^{-1}}$	0.1
Back pressure (p_b) in bar	50
Injection volume flow rage ($\dot{V}_i$) in $\mathrm{cm^3\,s^{-1}}$	15
Packing pressure (p_h) in bar	150

was determined in preliminary trial runs, and the resulting parameter set-points are given in Table 6.1. With these settings, the part is cured to around 80 % in the mold, with the remainder of the curing reaction taking place after the part is demolded. However, reliable methods (chapter 3) for determining the part properties are not able to be applied in-line, thus, on-line part quality assessment was done visually. In first experiments, 23 parts were manufactured in this way and the process data were stored (8 cycles for startup and 15 as a training set). From this training set P_c, $\mathrm{SPE}_t(k)$, and $T^2{}_t$ are calculated. One base assumption taken in subsection 2.2.4 for calculating P_c is that variable correlation is similiar within a process phase and dissimilar between phases. Whether this holds true for monitoring the rubber injection molding process is a powerful indicator of the PCA monitoring system working as intended. To visualize the similarities and dissimilarities of different process phases, the first three entries of each loading matrix $P(k)$ of the training data are plotted in Figure 6.2. Even visually, a clear distinction between different clusters can be made, proving the assumption of P capturing within-phase similarities to be true. Yang et al. [7] use a k-means algorithm to find the clusters and distinguish each phase. However, molding equipment used in this work provides digital
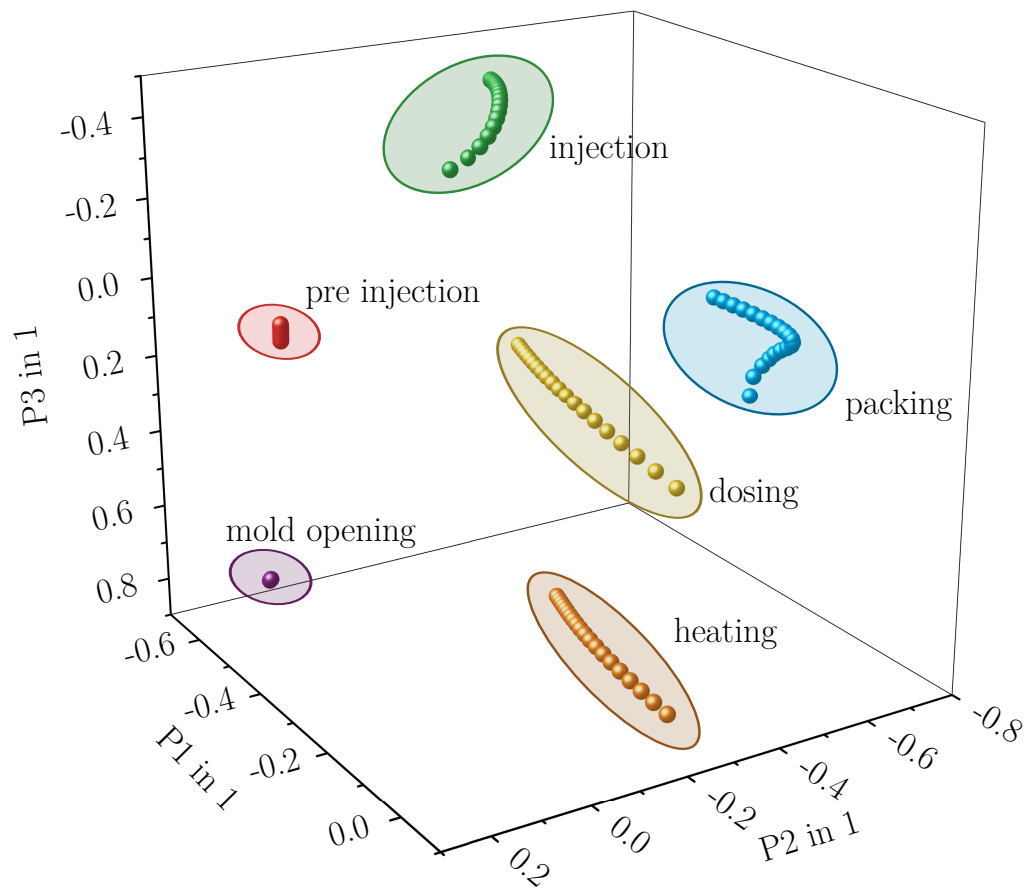
Figure 6.2: The *P* matrices of different process phases are spatially clustered.

signals inidicating which phase the machine control has initiated. It is notable that clustering differs from machine control in phase recognition. For example, clustering recognizes a pre-injection phase which covers the amount of time until the rubber has reached the inmold sensor, while this is grouped with the main injection phase by the machine control. Nevertheless, it is clearly visible that the distances between individual clusters is high, which underlines the need for treating every process phase separately. Once the phase loading matrices P_c are known, the control bounds $\text{SPE}_t(k)$, and $T^2{}_t$ (Equation 2.33, and Equation 2.32 respectively) are determined. In this work, we set the significance level to 95 % ($\alpha = 0.95$), which is common in many engineering applications of statistics [7, 8].

After completing the training data set, we introduced artificial process deviations to test the monitoring system. These changes are shown in Table 6.2. Between those different types of deviations, the process parameters were set back to the main operating point for 10 cycles to let the process return to an in-control state. This is done to verify if the monitoring system responds to the artificial change and not recognize every cycle that is not part of the training set as a fault. The artificial changes were each selected to be introduced for specific reasons.

First, by altering the rotation speed of the screw during dosing v_s, the total work of dosing changes. Consequentially, as shown in Figure 5.1 (page 75), the temperature of the rubber prior to injection is different. This has two important effects: First, η changes, affecting the pressure needed for filling the cavity and second, the curing kinetics change, altering the optimal cure time and thus part properties.

Second, variations in the storage maturation of NBR are common sources of error in industrial setups. The impact of the storage conditions on kinetic constant of curing (k_c) has been shown to be minimal in RPA testing conditions (section 3.1). Yet, its impact in application has not been investigated broadly [97]. Furthermore, as was shown in Figure 3.2, storage at RT increases the η of the NBR used in the experimexents.

Third, changing the setpoint of the barrel temperature T_b mimics fluctuations of the temperature control in industrial setups. Although T_b is treated like it was the true temperature of the injection units' barrel, it is the setpoint of the temperature control unit of the water flowing through the tempering channels. This simplification is also true for most other temperatures recorded in injection molding. The change of seasons brings variations of air humidity and temperature of the environment, which affects the efficiency of central cooling systems commonly found in production plants.

All artificial changes introduced in the experiments do not directly affect monitored signals, which of course would be trivial for the monitoring system to detect. Rather, all changes affect processing-relevant properties of the

Table 6.2: Three types of artificial deviations were introduced in the first experiments

Setpoint changed	Magnitude	Cycles
v_s	$\times 3$	23-30
Rubber	Store 1 month at RT	43-52
T_b	$-20\,\mathrm{K}$	62-end

NBR which are then picked up by machine and mold sensors.

The PCA monitoring approach in this work differs from other reported work in its fault detection logic (subsection 2.2.4 [7, 8, 13]). It detects faults only if the mean difference between the SPE statistic and the control bound SPE_t is positive, instead of detecting a fault if at any sample k the SPE is above its control bound. Thus, the detection logic in this work is more lenient than reported approaches. To illustrate the necessity for this more lenient appraoch, Figure 6.3 visualizes SPE trajectories of cycles representative of their class for the first $60\,\mathrm{s}$ of each batch of the experiments. Every sampling point of each curve is calculated as in Equation 2.33 and Equation 2.31. Process parameters in cycle 58 were set to their normal operating conditions, thus the cycle should be classified as in-control. It also is not part of the training set, and thus assumes the role of belonging to a control group. The SPE values of cycle 32, where v_s was tripled, are also not entirely above SPE_t, but both investigated cycles are correctly identified as in- and out-of control respectively. Furthermore, it can be seen that when looking at Figure 2.6, SPE values posess information across all process phases, and are not limited to a single one, for instance the injection phase or the dosing phase. This again solidifies the applicability of multivariate statistics to monitor rubber injection molding processes.

Consecutively, Δ_{SPE} was calculated for all cycles and is plotted in Figure 6.4 a). The T^2 statistic is plotted in b). Every artificially introduced fault can be detected in most of the cycles and it is detected by both statistics. When conditions are brought back to the normal operating point, both

statistics mostly show a return to in-control operation as well. Some cycles, however, are either false positives or false negatives.

These misclassifications occur consistenly when process parameters are changed and highlight the lagged nature of some fault types. When v_s is tripled before dosing of cycle 23, the rubber that gets injected in this cycle is already subject to increased dissipation, being immediatly detected as a fault by the process monitoring system. The same is true for cycle 30, where v_s is set back to normal operation. On the other hand, a change in
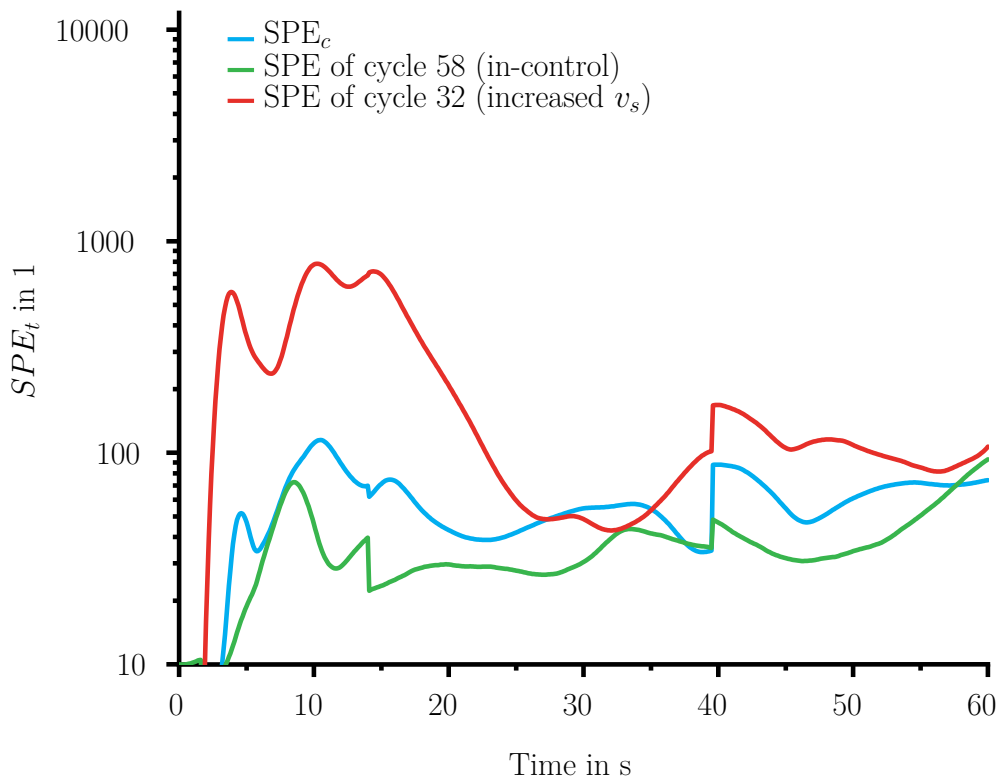
Figure 6.3: In the first 60 s of the injection molding process, in-control and out-of-control cycles can be discerned easily by their SPE values.

storage maturation of the rubber can only be detected with a time delay of about three to four cycles, which approximately matches the dwell time of the injection unit in use: NBR stored at RT is fed in cycle 43, but a volume of around four cycles of rubber is contained within the injection unit. Thus, process variables are effected by the changed rubber properties only when
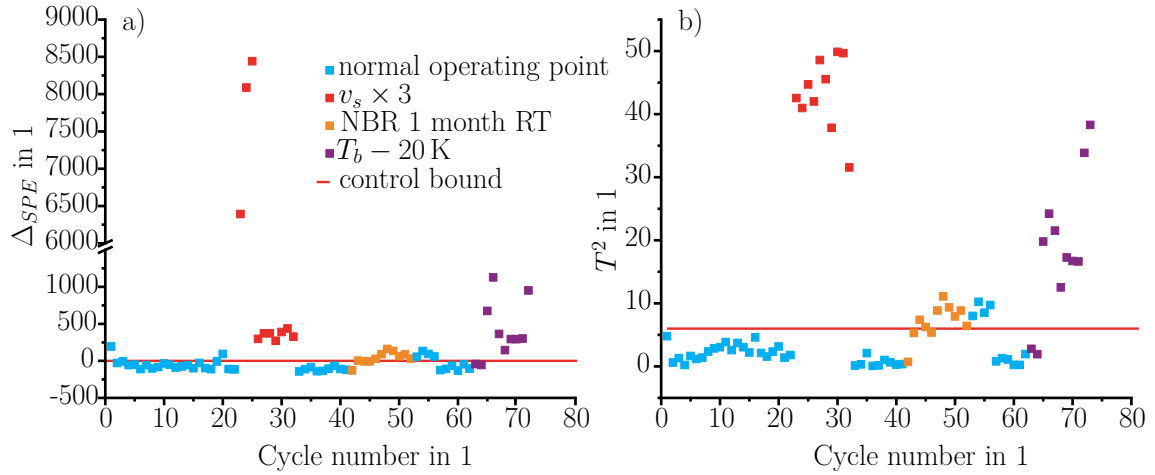
Figure 6.4: Artificially introduced faults are detected a) by the Δ_{SPE} statistic and b) by the T^2 statistic as reported in [84].

the amount of NBR stored at RT surpasses a critical amount in the screw. Similarly, when NBR stored at 5 °C is fed again (cycle 53), out-of-control states are still detected for 3 cycles after the event.

A delayed response is also present when T_b is lowered to 60 °C. Thus, increasing Δ_{SPE} and T^2 values are recorded while the rubber in the barrel cools gradually with passing time. These delayed responses pose challenges to the correct performance evaluation of monitoring results. For example, when feeding an incorrectly stored NBR, only very little of this rubber portion reaches the mold in the first cycles after feeding it, not affecting part quality. In the subsequent cycles, D may already change and a fault is detected even if still very little of the new rubber portion reaches the screw antechamber. Fortunately, every time such transitions appeared, PCA monitoring gives a conservative estimate, resulting in false alarms rather than undetected faults, which is preferable in industrial application. However, precision, recall, and false alarm rate may still be reduced by those uncertainties and, as will be discussed in subsection 6.2.2, FDA training can also be influenced negatively. As multivariate monitoring is not commonly used by rubber part manufacturers to monitor their processes, the experiments of section 6.2

should also be monitored by more widespread univariate methods.

6.2.1 Comparison to EWMA

For method comparison, data from the same experiments are also evaluated by plotting EWMA charts from the data. Before z-values (Equation 2.12) can be calculated, a feature which contains the most information on the process state has to be determined for every process variable (as mentioned in subsection 2.2.4). Based on common knowledge, the maximum value of each cycle was calculated for p_i, p_m and D [56, 127]. Furthermore, λ needs to be chosen empirically and was in this case set to $\lambda = 0.8$, as lower values tend to smooth results too much. The confidence bounds were calculated as in Equation 2.13 and Equation 2.14.

When comparing Figure 6.5 to Figure 6.4, it is visible that the monitoring capabilities of EWMA charts are far inferior to multivariate monitoring by the PCA-based approach, and precision and recall values in Table 6.3 for each method underline this intuition. EWMA charts also feature two-sided control bounds, giving information if signals belonging to faults were comparatively lower or higher than those beloging to in-control cycles. In this special application, this proves to be a disadvantage to fault detection, as drifts from one fault to another (visible in Figure 6.5 a), cycle 35 to 45) may cross the confidence region, thus generating false "in-control" cycles. However, when EWMA charts are evaluated off-line by personnel which understands the correlation of the specific manufacturing setup, these double sided confidence bounds provide more information without performing additional analysis. Still, when multivariate fault detection is coupled with multivariate fault identification (e.g. by performing FDA) in the present investigations, it outperforms EWMA in every aspect.

6.2.2 Application of FDA

In the current experiments, all three artificially introduced faults can be detected with PCA-based methods. However, the Δ_{SPE} and T^2 statistics do
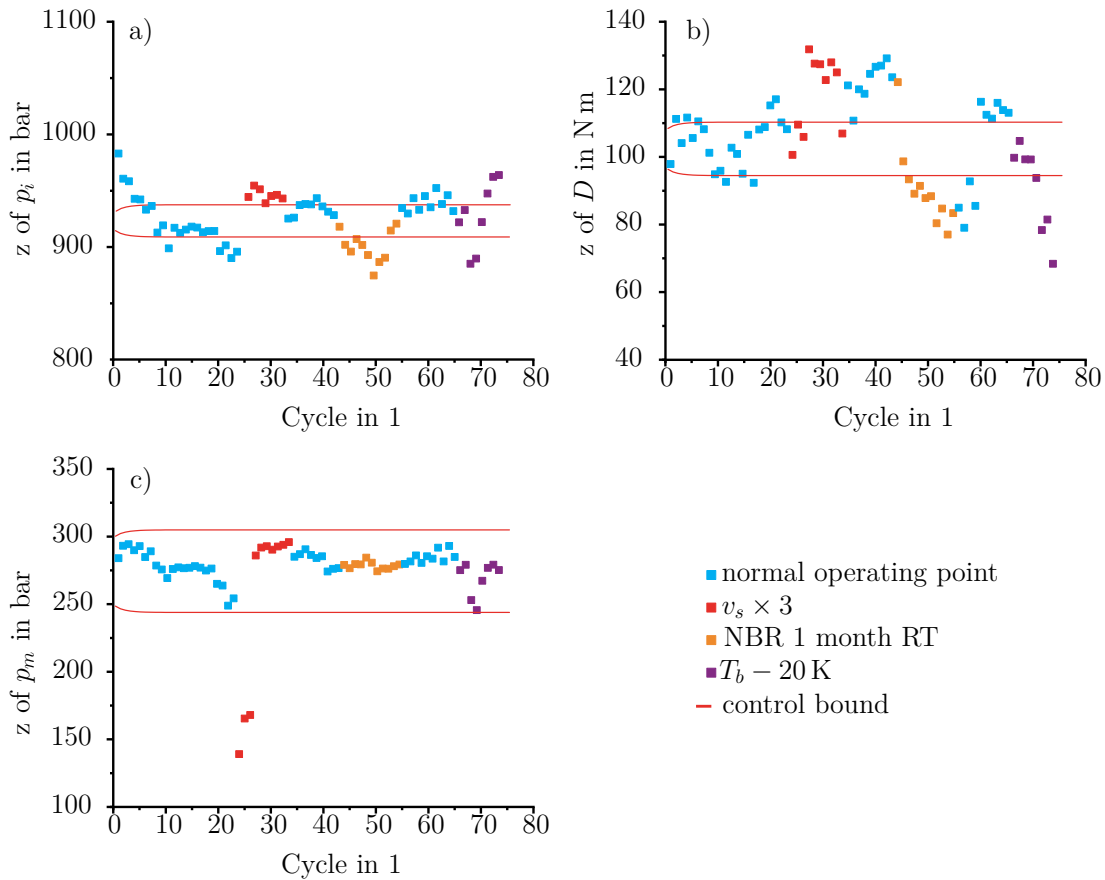
Figure 6.5: Fault detection accuracy is greatly reduced when a) p_i, b) D or c) p_m are monitored seperately.

not provide information on what kind of fault has occured. To differentiate between the different types of faults which cause the out-of-control process states, FDA is applied. As already discussed in section 2.3, FDA is a supervised learning method, and data from every class that should be identified needs to be present in the training data set. Thus, data from each cycle of the manufacturing run needs to be labelled with their respective fault class (no fault also being one class). This is only possible after the experiments have passed (off-line), because when PCA training finishes, no fault has yet occurred. Only when FDA training has been completed with

Table 6.3: Precision and recall values for univariate EWMA charts and PCA based process monitoring

	Precision	Recall
EWMA of p_i	0.51	0.78
EWMA of D	0.42	0.72
EWMA of p_m	1.00	0.18
PCA	0.81	0.89

the most common fault types presented to the optimization algorithm, it can be used for detecting and classifying new faulty cycles.

Furthermore, in this work, FDA is not performed with every $X(k)$ but rather a matrix $X \in \mathrm{R}^{I \times J}$, containing process feature vectors $\mathbf{x}_j$ for each signal j. Thus, each batch is represented by a feature vector $\mathbf{x}_i$ rather than by a matrix of signal trajectories. In Figure 6.6 a), FDA transformed data from the first experiments are plotted. They are colorized to represent the real class strictly taken from Table 6.2. In Figure 6.6 b), the color represents the predicted class by FDA classification.

Visually, it is obvious that the accuracy of classification is high, and this is underlined by a precision score of 0.74 and a recall of 0.80. The labelling of the data was done by not taking into account lagged effects discussed in section 6.2. The challenge to correctly labelling cycles is also apparent in FDA space, as borders between fault types are very close together. The true class of samples in these border areas are also not determined definitely, so strict labelling always introduces errors. Especially distinguishing between the fault $v_s \times 3$ and normal operation causes a high percentage of the total confusion of the FDA model.
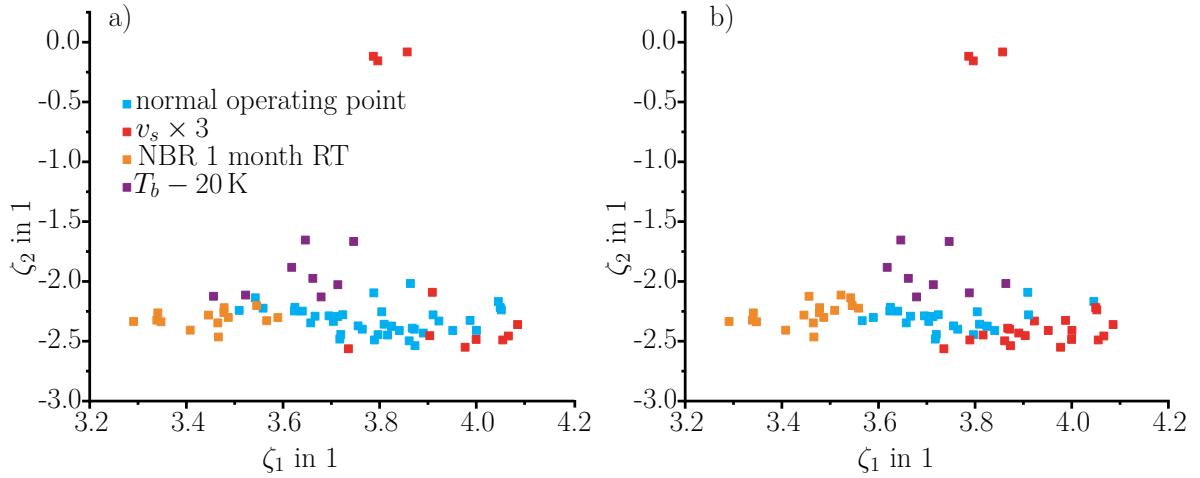
Figure 6.6: Transformed into FDA subspace, regions with different fault types can be distinguished. a) color represents the real class. b) Colors represent predicted class

6.2.3 Eliminating mold sensors

Suppliers of injection molded parts rarely tie one single mold to a single machine, but rather use various molds on different compatible injection molding machines. This is even more common when technical rubber parts are injection molded, as lot sizes tend to be smaller than with commodities. Setting up process monitoring systems which rely on specific kinds of mold sensors or simply on mold sensors being present at all is not optimal, as different molds may be equipped with different sensors or molds do not feature sensors at all. Even when the correct sensors are mounted on different molds, reconnecting them with the setup after mold swaps costs time and introduces additional complexity. Thus it is preferable to perform process monitoring only with signals available from the injection machine control itself. The mayor drawback of doing so is of course the loss of information, such as the pressure gradient during injection and temperature increase due to dissipation. In this current injection molding setup, the only available not-controlled signals are p_i and D. The range of available signals is expanded in
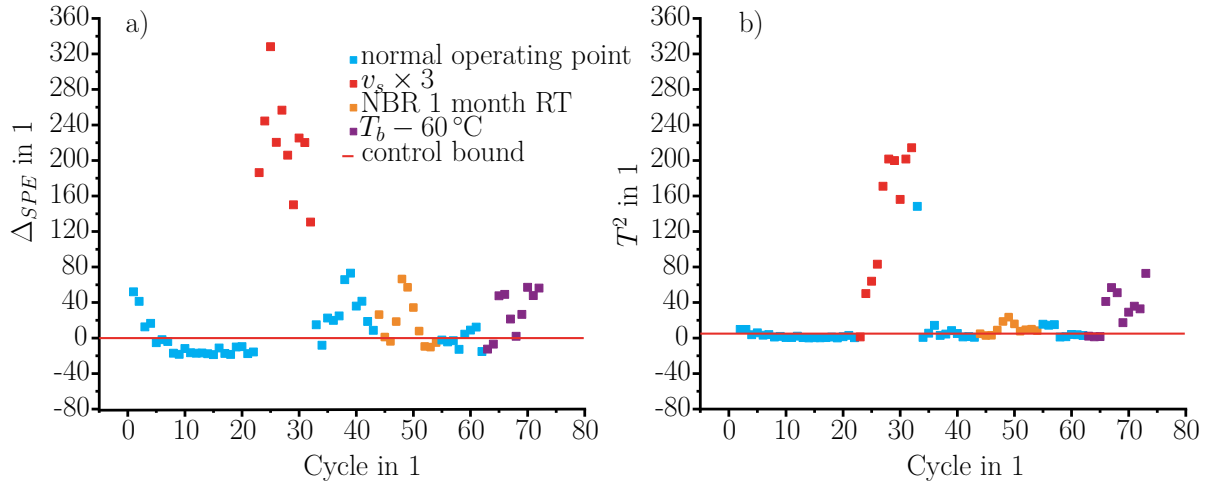
Figure 6.7: Monitoring the experiments without sensors installed in the mold reduces the reliability of the a) Δ_{SPE} and b) T^2 statistics.

section 6.3 by experiments done with an injection molding machine, which features additional monitoring systems. Figure 6.7 shows Δ_{SPE} and T^2 statistics with only p_i and D as variables. Multivariate monitoring is still applied even when univariate monitoring is easily feasible with only two signals, as both signals cannot be considered to be linearly independent. It is apparent that the artificially introduced faults still cause their respective monitoring systems to be above the control bounds. Surprisingly, while faults can still be detected, the majority of control cycles where setpoints are returned to normal operation are detected as out-of control. This reduces the precision and recall of the system to 0.66 for both measures and shows that more signals can not only increase a multivariate monitoring system's response to faults, but also its tolerance to minor fluctuations and reduce its false alarm rate.
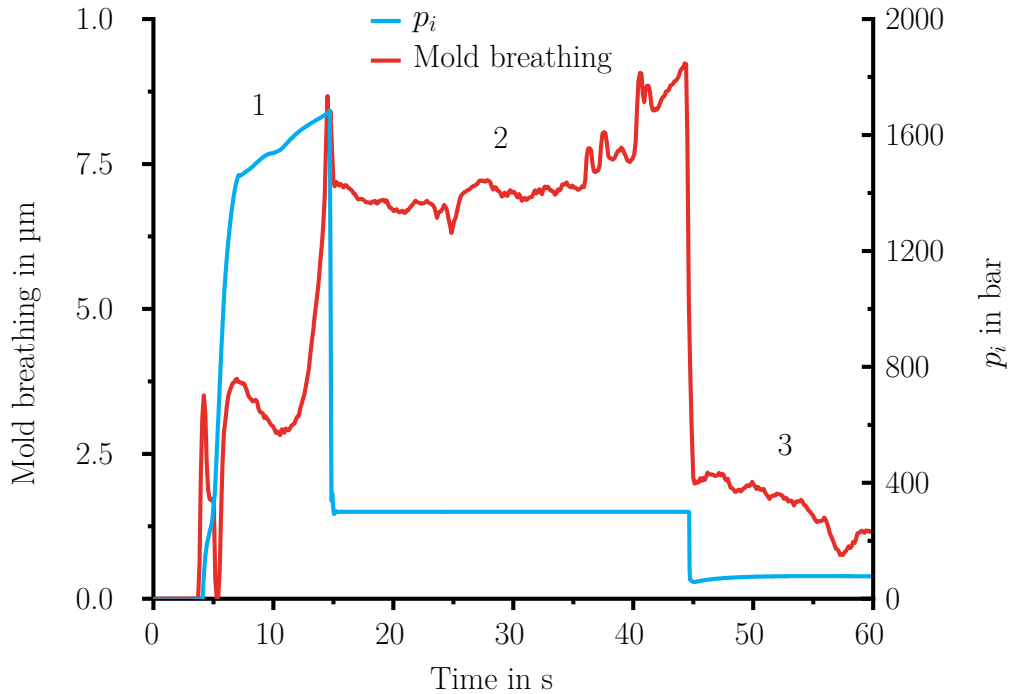
Figure 6.8: The mold breathing captured by the iQ clamp control system shows distinctive trajectories in the injection phase (1), the packing phase (2) and the curing phase (3).

6.3 Expanding the monitoring system with advanced machine systems

The rubber injection molding machine ENGEL e-victory 740/220 detailed in section 3.4, which is equipped with the iQ clamp control system for measuring mold breathing (section 1.3) was used for the experiments in this section. The iQ clamp control signal can be incorporated into the PCA-based monitoring approach without need for large-scale adaptions, as the system is not sensitive to the source or type of information used as input.

Figure 6.8 puts the trajectory of the mold breathing captured by iQ clamp

Table 6.4: Operating point of the training data generated in first experiments to test PCA based process monitoring

Process parameter	Set point
t_c in s	254
T_m in °C	160
T_b in °C	80
v_s in $\mathrm{m\,s^{-1}}$	0.16
p_b in bar	50
$\dot{V}_i$ in $\mathrm{cm^3\,s^{-1}}$	7.5
p_h in bar	300

control in context with the trajectory of p_i for the first 60 s of a sample cycle of the experiments. In the injection phase (1), the packing phase (2) and the curing phase (3), each signal shows distinct trajectories. In the injection phase the signal rises (the side peak could be caused by an imbalance in filling of the two cavities) as the pressure in the cavities builds. In the packing phase, a slight rise in the mold breathing signal can be caused by temperatures of the rubber and the mold creeping back to balance. In the curing phase, all external forces (from the screw) are removed, and the rubber in the cavity can relax to atmosphere pressure.

The experiments discussed in this section are chosen to be as similar as possible to section 6.2, in order to allow a certain degree of comparability, even though machine and mold are dissimilar. First, train data were generated by manufacturing parts at an operating point (Table 6.4), where stable conditions can be achieved and the part quality is optimal.

After enough training data were generated, artificial changes of the same type as in section 6.2 were introduced and the cycles at which the changes to the operating conditions were made are shown in Table 6.5. On this setup, the range of signals available from the injection molding machine is expanded from only p_i and D to also incorporate the iQ clamp control

Table 6.5: Two types of artificial deviations were introduced in the advanced
monitoring experiments

Setpoint changed	Magnitude	Cycles
Rubber	Store 1 month at RT	20-35
T_b	$-20\,\mathrm{K}$	54-end
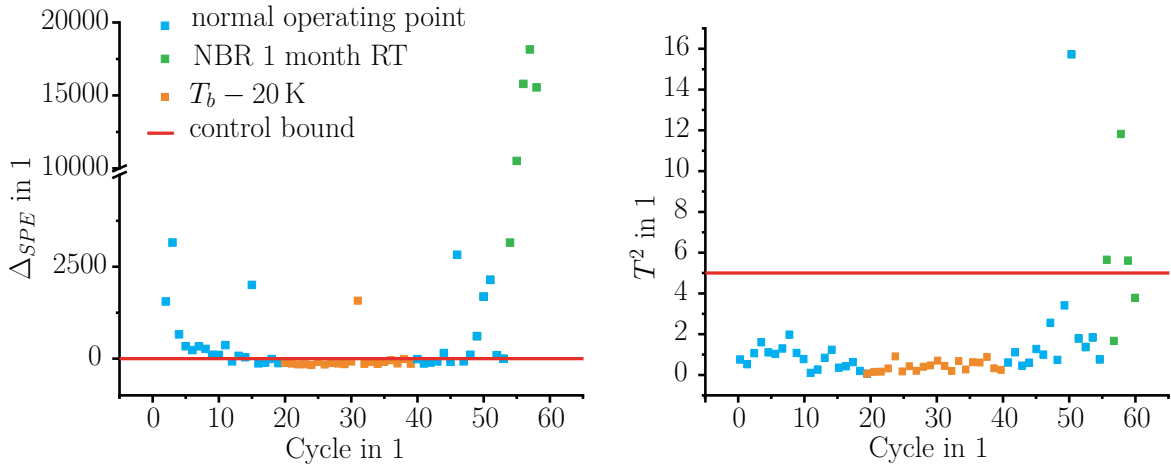
Figure 6.9: When advanced monitoring signals are used for PCA based
monitoring, the Δ_{SPE} statistic performs better for fault detection
than the T^2 statistic.

value. While this is still a limited amount of signals, the mold breathing
represented by the iQ clamp control value shows good correlation with the
in-mold pressure [46].

Figure 6.9 shows the Δ_{SPE} and T^2 results of all cycles recorded during
the experiments. It is notable that the Δ_{SPE} statistic clearly detects the
warm-up phase as well as the reduction of the T_b setpoint. The change in
storage maturation of the NBR compound, however can not be detected. As
already shown in section 2.4 and section 6.2, all monitoring methods operate
well with simulated and real world data. In comparison to PCA, FDA also
has access to a posteriori knowledge about data classes and its separation
accuracy is generally higher [13]. In Figure 6.10, cycles where NBR was
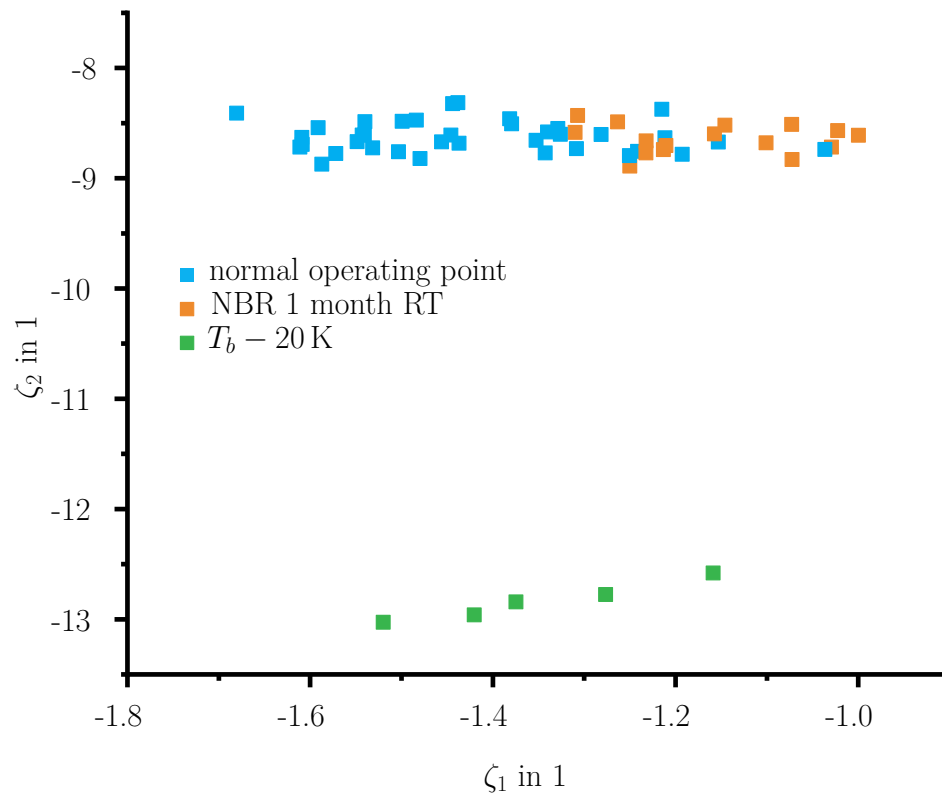
Figure 6.10: FDA can not completely separate cycles of the normal operating point from cycles where NBR storage maturation was changed.

stored at RT for one month are spatially grouped together, which is an improvement over PCA monitoring. However, there is no sharp distinction to cycles conducted in normal operation and as a result, misclassifications occur. When both monitoring methods are not able to fully seperate these two fault types, it can be concluded that this specific change in storage maturation did not impact available signals enough to be detected as out-of-control states.

6.3.1 Part quality control by mechanical testing

The purpose of process monitoring is of course to detect changes that are detrimental to the quality of the resulting products. For technical rubber parts, mechanical performance is the most important measure of part

quality. To characterize the impact of the artificially introduced changes on part quality, the CS and C_d of the parts were determined as described in section 3.2. Test results of both parameters are plotted in Figure 6.11 a) and b) respectively. Data points are again color coded to match the fault class that should be present at the current cycle, and the control bounds were calculated as the standard deviation of cycles deemed in control by the Δ_{SPE} statistic to illustrate overall scattering and shifts from artificial disturbances. It can be seen that both parameters do not respond to any artificial change in a consistant way. Rather, in a wide interval of cycles from the process start to around cycle 17, values of both parameters are outside the control bounds. This can be accounted to the mold still heating up during manufacturing, as both the CS and C_d indicate a rising degree of cure as manufacturing moves on. This mold heat up is also detected by the Δ_{SPE} values shown in Figure 6.9. Also, as the PCA based online monitoring and FDA fault identification are able to detect the artificial change to $T_b = 60\,°\mathrm{C}$, there is evidence that the degree of cure of injection molded parts is robust against fluctuations which manifest themselves during dosing and injection, and the process monitoring system is more cautious than it needs to be. It is, however sensitive to changes in the mold temperature. This indicates that for the NBR used in the investigations, the mold temperature is the most important factor determining the degree of cure of the parts. This could also be found by Fasching et al. [128].
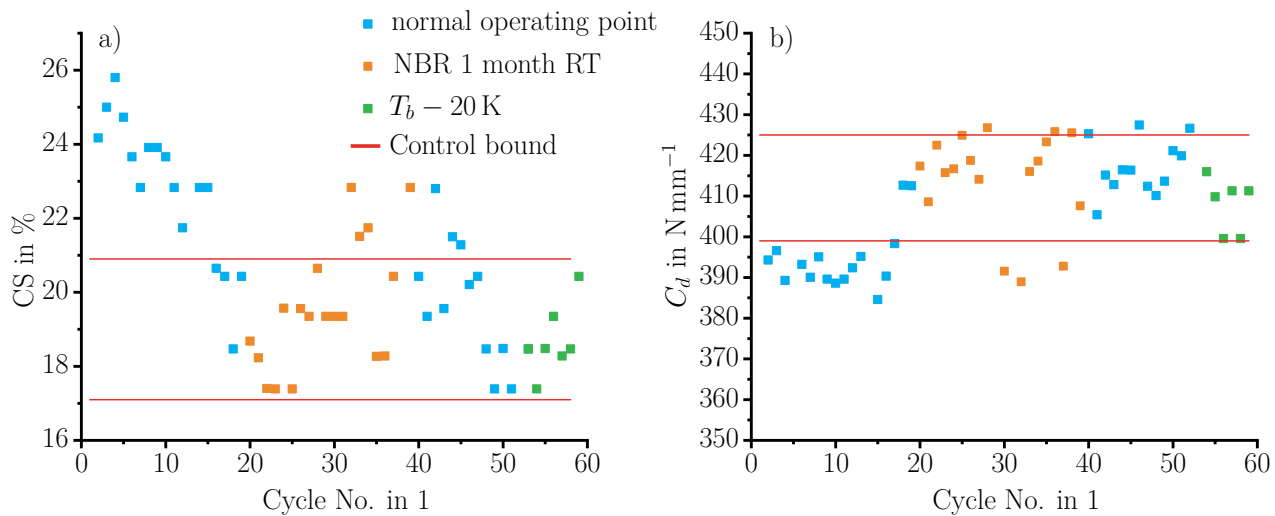
Figure 6.11: Mechanical parameters of rubber parts are not affected by artificial process disturbances.

6.4 Conclusions to process monitoring experiments

In this chapter, methods of multivariate statistics were successfully applied to rubber injection molding. When fluctuations were introduced to the process, the PCA-based online monitoring system strongly outperformed univariate monitoring methods such as EWMA charts in fault detection. In rubber injection molding, many injection molding process signals are not linear independent, as most of them measure phenomena which are all rooted in the rubber's intrinsic physical and chemical properties. This linear dependence can be used by multivariate methods both to reduce the false alarm rate and increase the detection accuracy of true faults. For example, by calculating a one-sided confidence bound instead of an upper and a lower control limit, no traversing of the in-control signal region occurs when the fault type changes. With FDA, the different types of faults could be identified automatically, which is of great benefit for the operator as it can play a guiding role in trouble shooting. However, limitations of this approach are still evident:

104

6.4.1 Limits of SPM in rubber injection molding

In the experiments presented in this work, no significant quality problems resulted from the artificially introduced fluctuations, as the temperature fluctuations at low levels during dosing and injection proved to be overpowered by the energy input during curing. This was evident before investigations started, and counteractions were set from the beginning. To maximize the contrast between in-control and out-of-control cycles, the artificially introduced faults of the monitoring system building experiments were carefully chosen, as they clearly altered the temperature of the NBR in preliminary investigations. Yet, as is shown in Figure 6.11, these temperature fluctuations do not manifest in the mechanical properties of the parts. An estimation of the energy contributions of the phases of the injection molding phases according to Equation 1.5 can provide insights. To perform this estimation, a dwell time of the rubber in the barrel of three dosing cycles is assumed. In Table 6.6, temperatures and dwell times for each phase of the rubber injection molding process to estimate the relative energy input are listed. Also in this table, the relative energy is stated. The contribution of the curing phase overpowering the contribution of other phases is clearly visible, it is around 40 times greater than the relative energy input of the 900 s (3 cycles), when the rubber is in the barrel. As a result, temperature fluctuations caused by variations of the rubber viscosity can be measured with the measurement methods presented in chapter 5, but their effect on the degree of cure after demolding is very limited.

To correctly predict the curing reaction, the system would need to be expanded with sensors specifically measuring the chemical conversion. While this has been proven to be possible for thermosets, little work has been found on methods for inline rubber cure measurements. Curing of thermosets can be measured in-line by optical sensors, as many resins are transparent in their uncured state, or by capacitive setups measuring the change in dielectric properties [129–131]. Because many industrial rubber compounds are highly filled with carbon black, these methods can not be transferred,

Table 6.6: Time, temperature and estimated energy contribution of the injection molding phases of the experiments discussed in subsection 6.3.1

	Temperature	Dwell time	Relative energy input
	°C	s	1
dosing	80	300	3.8
injection	100	14	0.4
curing	160	254	480.4

as the carbon black amount and type is the determining factor for the optical and electrical characteristics of the rubber compound [132]. More promising approaches for monitoring the curing reaction of rubber prove to be dynamic mechanical analysis and ultrasound based methods. While oscillatory dynamic measurements with the RPA device are already state of the art, Jaunich et al. successfully determined the degree of cure by an ultrasound setup [133]. To transfer both of these approaches into injection molds, a high amount of additional research will be necessary, as the respective measurement setups would need to be minituarized and tested for their applicability in highly non-isothermal conditions. Even then, fitting molds with such systems would increase mold design and manufacturing costs dramatically and also would require far more frequent maintenance. Thus, in-line cure measurements for quality prediction will most likely only be economically reasonable for the highest-end technical products. Still, when looking at other works, it remains unclear how measuring the actual degree of cure would improve results. In these papers, possibilities to alter the temperature of the rubber before it enters the mold were shown, but no conclusive results on the effects on the degree of cure after curing could be proven. Only curing time and mold temperature remain as significant factors for influencing the final degree of cure of injection molded rubber parts [40, 60, 97]. It can be concluded that, the only viable option to counteract when quality problems do occur during manufacturing, is to adapt curing time or

temperature. But for many industrial applications, adjusting the time or temperature of the curing phase is not a viable mean of automatic quality control. For one, adjusting the process timing can reduce productivity and downstream manufacturing steps can become out of tune. Second, the time constant of temperature changes of injection molds commonly far exceeds a cycle, thus any change would not come into effect the cycle it is needed.

6.4.2 Strenghts of SPM in rubber injection molding

The multivariate statistical process monitoring system presented in this work is capable, in all experiments, to reliably discern artificially introduced fluctuations from normal operation. As it is a data-driven system, one of its key advantages is its adaptability to different manufacturing equipment setups, process operation points and material setups. This adaptability is especially important for rubber injection molding, as industrial rubber compounds are almost always tailored to the product that is made from them. For example, while the NBR compound used in this work tends to show increased viscosity and accelerated curing rates when stored at increased temperatures (section 3.1), other compounds may show exactly opposing effects [96, 134]. The SPM system developed in this does not require any a-priori input of curing characteristics, flow behavior or other material parameters to function. As such, it can be much more readily applied than other, first principle based monitoring systems for rubber injection molding, which require extensive material testing prior to being used for monitoring manufacturing lots [11, 12].

Like all data-driven monitoring systems, the PCA based monitoring approach requires training data to learn the relationships between process variable values. In this work, mainly for development purposes, training was conducted off-line, but on-line training functions are possible to be implemented in the system, further increasing its real-time capabilities. During training, the fault detection system does not require to be fed data from out-of-control injection molding cycles to learn, and can be thus set up fully at the beginning of the manufacturing lot. When the parameters of

the process are changed, simple retraining can be performed without the need for extensive off-line hyperparameter adaptions.

As the transition to smart manufacturing facilities is made by rubber injection molded part manufacturers, online process monitoring systems become a necessity to be implemented into new manufacturing setups. According to Hermann et. al., a smart manufacturing facility should be designed to follow the principles of interoperability, virtualization, decentralization, and real-time capability [135]. The PCA-based process monitoring system can increase device interoperability. Besides being not limited to specific kinds of rubber compounds, it is not bound to specific kinds of data, as shown in subsection 6.2.3. There, a signal from an in-mold pressure sensor could be swapped for a mold breathing signal provided by the ENGEL iQ clamp control technology. Eliminating the need for sensors in the injection mold greatly reduces complexity for part manufacturers, as sensor placement is increasingly difficult when moving mold elements are present or the part features visible surfaces in application. Thus, the presented SPM system can be preferable to systems strongly relying on specific data obtained by additional sensor equipment. Furthermore, fault detection is not strictly limited to the injection molding equipment itself. In future applications, for example, process variables from peripheral injection molding equipment such as temperature control and handling devices can be readily included. If peripheral equipment is included, FDA fault detection can also be expanded to indicate malfunction of up- or downstream devices. Also, if faults are detected, the system could instruct handling devices to dispose parts automatically.

7 Summary and outlook

THIS work aimed at developing a statistical process monitoring system for rubber injection molding, which can detect unwanted process fluctuations detrimental to the product quality, on-line and without human interaction. Also, the causes of the fluctuations should be automatically identified by the system to facilitate process adaptions and allow the manufacturer to return to optimal quality production faster and easier.

To do so, first, a fast and robust method for testing the dynamic properties of rubber parts as a measure of their quality was developed. The measurement procedure as well as an analysis of the method's capabilities is presented in chapter 4. This method successfully provides a fast and reliable solution to determine various mechanical quality parameters, especially the dynamic spring stiffness and loss factor without any need for extracting specimens from the injection molded parts.

Second, fluctuations that are critical to the quality of the rubber parts were singled out. Such fluctuations commonly manifest themselves in variations of the rubber temperature while process setpoints are kept constant. In chapter 5, the transient material temperature throughout the entire injection molding process was measured with high accuracy by employing inline temperature measurement technologies such as ultrasound transmission and thermography. It was shown that for example, storage maturation is a source of process fluctuations that alter the temperature after dosing. Also, various dosing and injection set points were tested for their ability to significantly impact the quality of rubber parts.

Lastly, a data-driven process monitoring system was developed. This monitoring system detects faults automatically by using a Principal Component Analysis (PCA) based method, which takes into account all available

process signals at every sample of each process cycle. Therefore, as was shown in chapter 6, it offers superior fault detection performance compared to conventional statistical process monitoring methods such as Exponentially Weighted Moving Average (EWMA) charts. Furthermore, the process monitoring system can identify the type of disturbance that occured, once enough instances of a specific fault have occured and were labelled correctly by the manufacturing related personnel. By Fisher Discriminant Analysis (FDA), this fault type classification operates in a similar manner to PCA and increases the information the system is able to provide to the manufacturer significantly. In chapter 6, the system is tested on two different rubber injection molding setups, and it is able to correctly detect and identify the artificially introduced fluctuations in the experiments.

Consecutively, my data-driven fault detection and identification system for rubber injection molding processes provides a viable way of implementing process monitoring in smart rubber part manufacturing facilities. It is readily adaptable to any type of input signal, and does not require any a-priori assumptions on the process. Thus, it does not require large scale adaptions when the setup is changed, or rely on any specific type of equipment fitted to the injection mold. The tradeoff for its flexible design is its inabilty to predict the product quality as a continuous variable. Yet, it provides binary decisions between in-control and out-of control process states and a fault type prediction for operator guidance.

In future work, a data-driven Statistical Process Monitoring (SPM) system that is able to predict the part quality could be developed based on the results of this work. For one, manufacturers may use rubber compounds which exhibit a different curing behavior, where process fluctuations and storage maturation do alter the degree of cure of the final part more significantly. Also, for other applications, the most important aspects of quality, the ones which shall be predicted by the monitoring system, may be strongly dependent on storage maturation or batch-to-batch variations. An example of such is the strength of weld lines, which are more influenced by premature scorch than overall degree of cure. As soon as one can gather quality data

that show significant responses to process fluctuations, and enough data are present, training non-linear models, whether they are built by Artificial intelligence (AI) systems or regression will be feasible. Thus, quality aspects may be predicted from PCA score values, enabling the process control system to adapt the process settings accordingly.

However, implementing a quality-based closed-loop control necessitates determining the controller parameters specific to the machine setup and rubber compound in use. These can only be found, when quality functions can be retrieved, either by first-principle modelling or learning. Regardless, the system's complexity increases, demanding more resources and knowledge from part manufacturers. This again negates the key advantage of the multivariate SPM approach presented in this work, which requires minimal preparation and can be trained when parts are already manufactured. It remains questionable if such systems would find widespread application in the industry in the near future, as for injection molding of thermoplasts, they already existed for around 10 years and adoption rates stay well below expectations.

In the shift to build smart manufacturing facilities, the straightforward nature of the PCA and FDA-based SPM approach (for injection molding of thermoplastics and rubbers alike), combined with its fault detection and identification capabilities, can offer manufacturers considerable economical benefits. Especially since setting up such systems in smart manufacturing facilities does not require a great amount of resources, they are much more likely to find widespread adoption than their model- or knowledge based counterparts.

Acronyms

A   Area

A process parameter matrix

AI   Artificial intelligence

ANOVA Analysis of Variance

α significance level

c normal distribution

c velocity of sound

C_d dynamic spring stiffness

CS   Compression Set

C_s static spring stiffness

CUSUM Cumulative Sum

$\mathcal{X}^2$ Chi squared distribution

D torque

D_i Injection trigger signal

DMA   Dynamic mechanical analysis

DoE   Design of Experiments

DSC   Differential Scanning Calorimetry

Δ_{SPE} delta SPE value

δ loss angle

E process error matrix

$\mathbf{e}$ prediction error vector

E_A activation energy

EWMA Exponentially Weighted Moving Average

ϵ displacement

η dynamic viscosity

$\mathbf{F}$ F distribution

FAR False Alarm Rate

FDA Fisher Discriminant Analysis

F force

f frequency

F_r relaxation force

$\dot{\gamma}$ shear rate

k consistency

k_c kinetic constant of curing

KDD Knowledge Discovery in Databases

κ adiabatic compressibility

L length

LCL lower control limit of the EWMA chart

Λ matrix of eigenvalues

λ eigenvalue

λ memory factor of EWMA

μ mean

n flow index

NBR   Acrylonitrile-butadiene rubber

n_c curing reaction exponent

Ω FDA transformation matrix

ω FDA vector

ω angular frequency

P loading matrix

p scalar pressure value

p   pressure vector

p   loading vector

PCA   Principal Component Analysis

P_c representative phase loading matrix

phr   parts per hundred rubber

p_i injection pressure

p_m in-mold pressure

P_{pc} loading matrix containing the first m principal components

pvT   pressure-specific volume-temperature

R	universal gas constant
r	radius
r_0	radius of plug flow region
R^2	Coefficient of determination
RPA	Rubber Process Analyzer
RT	room temperature
ρ	density
S_b	between-class scatter matrix
SPE	Squared Predictive Error
SPM	Statistical Process Monitoring
S_t	total scatter matrix
SVD	singular value decomposition
S_w	within-class scatter matrix
Σ	covariance matrix
σ	variance
T	score matrix
T	temperature
t	continuous time
t	score vector
T_0	reference temperature
T^2	T^2 statistic

$\tan(\delta)$ loss factor

T_b barrel temperature

T_c curing temperature

t_c cure time

t_{c90} time until $90\,\%$ cured

T_m mold temperature

T_p temperature peak height

t_t time of travel

τ scalar shear stress value

$\underline{\underline{\tau}}$ stress tensor

τ_0 yield stress

$\mathbf{u}$ transformation vectors for FDA analysis

$\mathbf{UCL}$ upper control limit of the EWMA chart

v velocity

$\mathbf{v}$ velocity vector

$\dot{V}_i$ injection volume flow rate

$v_\perp$ velocity of perpendicular flow

v_s circumferential screw speed

w_h width at half height

Y process output matrix

X process input matrix

x   process signal vector

x_c degree of cure

z EWMA monitoring statistic

ζ FDA transformed coordinate vector

Bibliography

[1]   John A. Lindsay. *Practical guide to rubber injection moulding.* Shawbury, Shrewsbury, Shropshire, UK: Smithers Rapra, 2012. ISBN: 978-1-84735-707-6.

[2]   Matthias Zeppenfeld, Rupert Hirn, and Maximilian Wardenberg. "Drahtlose Prozessüberwachung im Spritzgussbauteil". In: *Kunststoffe* 4 (2019), pp. 24–29.

[3]   Thorsten Hickmann and Eric Klemp. "Das Potenzial von Redesign und Bauteilkonsolidierung". In: *Kunststoffe* 6 (2020), pp. 43–46.

[4]   Ulrike Luetzow. "Schutz und attraktives Design". In: *Kunststoffe* 5 (2020), pp. 54–57.

[5]   Reinhard Schiffers, Stefan Kruppa, and Stefan Moser. "Für jeden Schuss der richtige Umschaltpunkt". In: *Kunststoffe* 2014.11 (2014), pp. 58–63.

[6]   Michael Fasching et al. "Advanced Mold Concepts to Improve Part Quality and Productivity in Rubber Injection Molding". In: *Proceedings of the Regional Conference of the Polymer Processing Society.* Graz, Austria, 2015, p. 49.

[7]   Yi Yang, Xi Chen, and Ningyun Lu. *Injection molding process control, monitoring, and optimization.* Progress in polymer processing series. München: Hanser, 2016. ISBN: 978-1-56990-592-0.

[8]   Zhiwen Chen. *Data-Driven Fault Detection for Industrial Processes: Canonical Correlation Analysis and Projection Based Methods.* Wiesbaden: Springer Fachmedien, 2017. ISBN: 978-3-658-16755-4. DOI: 10.1007/978-3-658-16756-1. URL: http://dx.doi.org/10.1007/978-3-658-16756-1.

[9] Christopherus Bader and Erwin König. "Wir regeln das schon". In: *Kunststoffe* 6 (2012), pp. 2–6.

[10] Xi Chen, Furong Gao, and Guohua Chen. "A soft-sensor development for melt-flow-length measurement during injection mold filling". In: *Materials Science and Engineering: A* 384.1-2 (2004), pp. 245–254. ISSN: 09215093. DOI: 10.1016/j.msea.2004.06.039.

[11] D. Berkemeier et al. "Produktadaptive Prozeßregelung für das Kautschuk-Spritzgießen Teil 2". In: *Kautschuk Gummi Kunststoffe* 43.8 (1990), pp. 705–716.

[12] D. Berkemeier et al. "Produktadaptive Prozeßregelung für das Kautschuk-Spritzgießen Teil 1". In: *Kautschuk Gummi Kunststoffe* 43.7 (1990), pp. 610–625.

[13] Evan L. Russell, Leo H. Chiang, and Richard D. Braatz. *Data-driven Methods for Fault Detection and Diagnosis in Chemical Processes.* Advances in Industrial Control. London: Springer, 2000. ISBN: 978-1-4471-1133-7. DOI: 10.1007/978-1-4471-0409-4. URL: http://dx.doi.org/10.1007/978-1-4471-0409-4.

[14] Walter A. Shewhart. *Economic control of quality of manufactured product.* Mansfield Centre, CT: Martino Publishing, 2015. ISBN: 9781614278115.

[15] John S. Oakland. *Statistical process control.* 7th edition. Abingdon, Oxon and New York, NY: Routledge, 2019. ISBN: 9781315160511.

[16] Ellis R. Ott, Edward G. Schilling, and Dean V. Neubauer. *Process quality control: Troubleshooting and interpretation of data.* 4. ed. Milwaukee, Wis: ASQ Quality Pr, 2005. ISBN: 0-87389-655-6.

[17] Ranjit K. Roy. *Design of experiments using the Taguchi approach: 16 steps to product and process improvement.* New York: Wiley, 2001. ISBN: 0-471-36101-1.

[18] Zhongbao Chen and Lih-Sheng Turng. "A review of current developments in process and quality control for injection molding". In: *Advances in Polymer Technology* 24.3 (2005), pp. 165–182. DOI: `10.1002/adv.20046`.

[19] Xue Z. Wang. *Data mining and knowledge discovery for process monitoring and control*. Advances in Industrial Control. London and New York: Springer, 1999. ISBN: 9781447111375.

[20] Georg Pillwein, Thomas Gradl, and Josef Gießauf. "Umschalten auf konstante Qualität: Prozessschwankungen in der Silikonverarbeitung online ausgleichen". In: *Kunststoffe* 09 (2012), pp. 31–35.

[21] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. "From Data Mining to Knowledge Discovery in Databases". In: *AI Magazine* 17.3 (1996), p. 37. ISSN: 2371-9621. DOI: `10.1609/aimag.v17i3.1230`. URL: `https://wvvw.aaai.org/ojs/index.php/aimagazine/article/view/1230`.

[22] François Chollet. *Deep learning with Python*. Safari Tech Books Online. Shelter Island, NY: Manning, 2018. ISBN: 9781617294433. URL: `http://proquest.safaribooksonline.com/9781617294433`.

[23] Aurélien Géron. *Hands-on machine learning with Scikit-Learn and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. First edition, eleventh release. Beijing et al.: O'Reilly, 2017. ISBN: 978-1-491-96229-9.

[24] Krzysztof J. Cios et al. *Data Mining: A Knowledge Discovery Approach*. Boston, MA: Springer Science+Business Media LLC, 2007. ISBN: 9780387367958. DOI: `10.1007/978-0-387-36795-8`. URL: `http://dx.doi.org/10.1007/978-0-387-36795-8`.

[25] Wes McKinney. *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. Second edition. Sebastopol, CA: O'Reilly Media, 2017. ISBN: 9781491957660. URL: `http://proquest.tech.safaribooksonline.de/9781491957653`.

[26] Svante Wold et al. "Multi-Way Principal Components and PLS-Analysis". In: *Journal of Chemometrics* 1 (1987), pp. 41–56.

[27] Yi Yang and Furong Gao. "Cycle-to-cycle and within-cycle adaptive control of nozzle pressure during packing-holding for thermoplastic injection molding". In: *Polymer Engineering & Science* 39.10 (1999), pp. 2042–2063. ISSN: 00323888. DOI: 10.1002/pen.11597.

[28] Jian-Yu Chen, Kai-Jie Yang, and Ming-Shyan Huang. "Online quality monitoring of molten resin in injection molding". In: *International Journal of Heat and Mass Transfer* 122 (2018), pp. 681–693. ISSN: 00179310. DOI: 10.1016/j.ijheatmasstransfer.2018.02.019.

[29] Magida Zeaiter, Wendy Knight, and Simon Holland. "Multivariate regression modeling for monitoring quality of injection moulding components using cavity sensor technology: Application to the manufacturing of pharmaceutical device components". In: *Journal of Process Control* 21.1 (2011), pp. 137–150. DOI: 10.1016/j.jprocont.2010.10.018.

[30] Suzanne L. B. Woll and Douglas J. Cooper. "Pattern-based closed-loop quality control for the injection molding process". In: *Polymer Engineering & Science* 37.5 (1997), pp. 801–812. ISSN: 00323888. DOI: 10.1002/pen.11723.

[31] Songtao Zhang, Rickey Dubay, and Meaghan Charest. "A principal component analysis model-based predictive controller for controlling part warpage in plastic injection molding". In: *Expert Systems with Applications* 42.6 (2015), pp. 2919–2927. DOI: 10.1016/j.eswa.2014.11.030.

[32] Yi Yang and Furong Gao. "Injection molding product weight: Online prediction and control based on a nonlinear principal component regression model". In: *Polymer Engineering & Science* 46.4 (2006), pp. 540–548. ISSN: 0032-3888. DOI: 10.1002/pen.20522.

[33] Hwaseop Lee, Kwangyeol Ryu, and Youngju Cho. "A Framework of a Smart Injection Molding System Based on Real-time Data". In: *Procedia Manufacturing* 11 (2017), pp. 1004–1011. ISSN: 23519789. DOI: `10.1016/j.promfg.2017.07.206`.

[34] Peng Zhao et al. "A nondestructive online method for monitoring the injection molding process by collecting and analyzing machine running data". In: *The International Journal of Advanced Manufacturing Technology* 72.5-8 (2014), pp. 765–777. ISSN: 0268-3768. DOI: `10.1007/s00170-014-5711-0`.

[35] Michal Stanek, David Manas, and Jakub Javorik. "Simulation of Injection Molding Process by Cadmould Rubber". In: *International Journal of Mathematics And Computers in Simulation* 5.5 (2011), pp. 422–429.

[36] Thomas Droste. "Simulation und Praxis: Analyse von Differenzen zwischen rheologisch berechneten und praktischen Formteilfüllungen". In: *Kunststoffe* 92.3 (2002), pp. 46–49.

[37] Rajganesh Jegadeesan. "Measurement of polymer melts' viscosity at increased pressure levels with the consideration of the compression heat". Master Thesis. Leoben: Montanuniversität Leoben, 2005.

[38] Walter Friesenbichler, Michael Fasching, and Leonhard Perko. *Shear and Extensional Viscosity- Measurement for Rubber Compounds by Means of Capillary Rheometry.* Cleveland, Ohio, 8.-12. Juni 2014.

[39] Stefan Schuschnigg. "Rheologische Untersuchungen bei hohen Schergeschwindigkeiten mit Hilfe eines Mikorheologie-Schlitzdüsen Messsystems". Master Thesis. Leoben: Montanuniversität Leoben, 2004.

[40] Michael Fasching et al. "Robust process control for rubber injection moulding with use of systematic simulations and improved material data". In: *International Polymer Sicence and Technology* 41.11 (2014), pp. 1–5.

[41] Sebastian Stieger et al. "Contraction and capillary flow of a carbon black filled rubber compound". In: *Polymer Engineering & Science* 60.1 (2020), pp. 32–43. ISSN: 00323888. DOI: `10.1002/pen.25256`.

[42] Roman Christopher Kerschbaumer et al. "Comparison of steady-state and transient thermal conductivity testing methods using different industrial rubber compounds". In: *Polymer Testing* 80 (2019), p. 106121. ISSN: 01429418. DOI: `10.1016/j.polymertesting.2019.106121`.

[43] Manfred Pahl, Wolfgang Gleissle, and Hans-Martin Laun. *Praktische Rheologie der Kunststoffe und Elastomere*. 4., überarb. Aufl. Kunststofftechnik. Düsseldorf, Germany: VDI-Verlag, 1995. ISBN: 9783182341925.

[44] *Prozessüberwachung und -regelung: Transparenz in der vernetzten Spritzgießproduktion*. 2020. URL: `https://www.kistler.com/?type=669&fid=110724&model=document&callee=frontend` (visited on).

[45] ENGEL Global, ed. *inject 4.0: Solutions for the smart factory*. engelglobal.com, 2020.

[46] Georg Pillwein, Johannes Lettner, and Markus Lehr. "Atmungssignale für die Prozessoptimierung nutzen". In: *Kunststoffe* 1 (2020), pp. 44–49.

[47] Robert Vaculik, Thomas Mann, and Willi Steinko. "Vernetzte Transparenz in Echtzeit". In: *Kunststoffe* 6 (2018), pp. 30–35.

[48] Yongyong Hui and Xiaoqiang Zhao. "Multi-phase batch process monitoring based on multiway weighted global neighborhood preserving embedding method". In: *Journal of Process Control* 69 (2018), pp. 44–57. DOI: `10.1016/j.jprocont.2018.06.012`.

[49] Q. Peter He and Jin Wang. "Statistical process monitoring as a big data analytics tool for smart manufacturing". In: *Journal of Process Control* 67 (2018), pp. 35–43. DOI: `10.1016/j.jprocont.2017.06.012`.

[50] Q. Peter He and Jin Wang. "Statistical process monitoring as a big data analytics tool for smart manufacturing". In: *Journal of Process Control* 67.67 (2018), pp. 35–43. DOI: `10.1016/j.jprocont.2017.06.012`.

[51] Yining Dong and S. Joe Qin. "A novel dynamic PCA algorithm for dynamic data modeling and process monitoring". In: *Journal of Process Control* 67 (2018), pp. 1–11. DOI: `10.1016/j.jprocont.2017.05.002`.

[52] Ray Wang et al. "A geometric method for batch data visualization, process monitoring and fault detection". In: *Journal of Process Control* 67 (2018), pp. 197–205. DOI: `10.1016/j.jprocont.2017.05.011`.

[53] Guthrie Gordon et al. "Quality control using a multivariate injection molding sensor". In: *The International Journal of Advanced Manufacturing Technology* 78.9-12 (2015), pp. 1381–1391. ISSN: 0268-3768. DOI: `10.1007/s00170-014-6706-6`.

[54] Suzanne L. B. Woll, Douglas J. Cooper, and Blair V. Souder. "Online pattern-based part quality monitoring of the injection molding process". In: *Polymer Engineering & Science* 36.11 (1996), pp. 1477–1488. ISSN: 00323888. DOI: `10.1002/pen.10542`.

[55] Xi Chen, Guohua Chen, and Furong Gao. "Capacitive transducer for in-mold monitoring of injection molding". In: *Polymer Engineering & Science* 44.8 (2004), pp. 1571–1578. ISSN: 00323888. DOI: `10.1002/pen.20154`.

[56] Fritz Röthemeyer and Franz Sommer, eds. *Kautschuk-Technologie: Werkstoffe - Verarbeitung - Produkte*. 3., neu bearb. und erw. Aufl. Hanser eLibrary. Munich, Germany: Hanser, 2013. ISBN: 978-3-446-43776-0.

[57] Michael Fasching, Walter Friesenbichler, and Gerald Roman Berger. "Validation of rubber injection molding simulation by correlation of simulated curing degree to measureable part properties". In: *Proceedings of the 30th Annual Conference of the Polymer Processing Society.* Cleveland (OH), USA, 2014.

[58] Wolfgang Fidi. "Vorlesung Elastomertechnologie II". In: (2011).

[59] Leonhard Perko. "Heizzeitverkürzung im Kautschukspritzguss unter Ausnutzung der Scher-, Dehn- und Kompressionserwärmung". Doctoral Thesis. Leoben: Montanuniversität Leoben, 2014.

[60] Leonhard Perko, Michael Fasching, and Walter Friesenbichler. "Model for the prediction of bulk temperature changes and pressure losses in rubber compounds flowing through conical dies". In: *Polymer Engineering & Science* 55.3 (2015), pp. 701–709. ISSN: 00323888. DOI: 10.1002/pen.23931.

[61] Miloslav Nič et al., eds. *IUPAC Compendium of Chemical Terminology.* Research Triagle Park, NC: IUPAC, 2009. ISBN: 0-9678550-9-8. DOI: 10.1351/goldbook.

[62] A. Sadr-Bazaz, Roberg Granger, and Jean Maurice Vergnaud. "Calculation of profiles of temperature and state of cure developed within the rubber mass in injection molding process". In: *Journal of Applied Polymer Science* 29.3 (1984), pp. 955–963. ISSN: 00218995. DOI: 10.1002/app.1984.070290323.

[63] Roman Christopher Kerschbaumer. "Prozessmodell zur Beschreibung des Dosiervorgangs von Kautschukspritzgießmaschinen". PhD Thesis. Leoben: Montanuniversitaet Leoben, 2019.

[64] Horst W. Stumpf. *Handbuch der Reifentechnik.* Vienna and s.l.: Springer Vienna, 1997. ISBN: 978-3-211-82941-7. DOI: 10.1007/978-3-7091-6519-5. URL: http://dx.doi.org/10.1007/978-3-7091-6519-5.

[65] Jean L. Leblanc and Alex Staelraeve. "Studying the Storage Maturation of Freshly Mixed Rubber Compounds and Its Effects on Processing Properties". In: *Journal of Applied Polymer Science* 53 (1994), pp. 1025–1035.

[66] Sung-Seen Choi. "Influence of storage time and temperature and silane coupling agent on bound rubber formation in filled styrene–butadiene rubber compounds". In: *Polymer Testing* 21.2 (2002), pp. 201–208. ISSN: 01429418. DOI: 10.1016/S0142-9418(01)00071-X.

[67] M. Fasching, W. Friesenbichler, and G. Berger. "Virtual modelling of the rubber injection molding process including verification in industrial scale experiments". In: *Proceedings of the IRC and DKT 2015*. Nuremberg, Germany, 2015.

[68] G. Knobloch. "Einfluß der Lagerungsbedingungen und der Stabilisatoren von Rohkautschuk auf sein Verarbeitungsverhalten und auf die Eigenschaften von Rohmischungen und Vulkanisaten". In: *Kautschuk Gummi Kunststoffe* 45.4 (1992), pp. 288–294.

[69] J. Murali Krishnan, Abhijit P. Deshpande, and P. B. Sunil Kumar. *Rheology of Complex Fluids*. New York, NY: Springer Science+Business Media LLC, 2010. ISBN: 9781441964939. DOI: 10.1007/978-1-4419-6494-6. URL: http://dx.doi.org/10.1007/978-1-4419-6494-6.

[70] Aroon V. Shenoy. *Rheology of Filled Polymer Systems*. Dordrecht, Netherlands: Springer Science+Business Media, 1999. ISBN: ISBN 978-90-481-4029-9.

[71] Ivica Duretek. "Vorlesungsunterlagen zu Rheologie der Kunststoffe". PhD thesis. Leoben: Montanuniversität Leoben, 2009.

[72] Robert Byron Bird, Warren E. Stewart, and Edwin N. Lightfoot. *Transport phenomena*. 2. ed. New York NY u.a.: Wiley, 2002. ISBN: 0-471-41077-2.

[73] Leonard Mullins. "The Thixotropic Behavior of Carbon Black in Rubber". In: *The Journal of Physical and Colloid Chemistry* 54.2 (1950), pp. 239–251. ISSN: 0092-7023. DOI: `10.1021/j150476a006`.

[74] A. R. Payne. "The dynamic properties of carbon black-loaded natural rubber vulcanizates. Part I". In: *Journal of Applied Polymer Science* 6.19 (1962), pp. 57–63. ISSN: 00218995. DOI: `10.1002/app.1962.070061906`.

[75] Peter Hornsby. "Rheology, Compounding and Processing of Filled Thermoplastics". In: *Advances in Polymer Science* 139 (1999), pp. 156–217.

[76] Jean L. Leblanc. "Elastomer-filler interactions and the rheology of filled rubber compounds". In: *Journal of Applied Polymer Science* 78.8 (2000), pp. 1541–1550. ISSN: 00218995. DOI: `10.1002/1097-4628(20001121)78:8{\textless}1541::AID-APP110{\textgreater}3.0.CO;2-1`.

[77] Gert Heinrich and Manfred Klüppel. "Recent Advances in the Theory of Filler Networking in Elastomers". In: *Advances in Polymer Science* 160 (2002), pp. 1–44.

[78] Uwe Sgodzaj. "Einfluß der Polymer-Struktur auf die Rheologie und das Dispersionsverhalten von BR- und SBR-Systemen". Doctoral Thesis. Karlsruhe: Universität Fridericiana, 2001.

[79] Gabriel J. Osanaiye, Arkadii I. Leonov, and James L. White. "Investigations of the rheological behavior of rubber-carbon black compounds over a wide range of stresses including very low stresses". In: *Journal of Non-Newtonian Fluid Mechanics* 49.1 (1993), pp. 87–101. ISSN: 03770257. DOI: `10.1016/0377-0257(93)85024-5`.

[80] Winslow H. Herschel and Ronald Bulkley. "Konsistenzmessungen von Gummi-Benzollösungen". In: *Kolloid-Zeitschrift* 39.4 (1926), pp. 291–300. ISSN: 0303-402X. DOI: `10.1007/BF01432034`.

[81] Steven X. Ding. *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms and Tools.* 2nd ed. 2013. Advances in Industrial Control. London: Springer, 2013. ISBN: 9781447147985. DOI: `10.1007/978-1-4471-4799-2`. URL: `http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10641776`.

[82] S. W. Roberts. "Control Chart Tests Based on Geometric Moving Averages". In: *Technometrics* 1.3 (1959), pp. 239–250. ISSN: 0040-1706. DOI: `10.1080/00401706.1959.10489860`.

[83] Douglas M. Hawkins and David H. Olwell. *Cumulative Sum Charts and Charting for Quality Improvement.* Statistics for Engineering and Physical Science. New York, NY: Springer, 1998. ISBN: 9781461272458. DOI: `10.1007/978-1-4612-1686-5`.

[84] Thomas Hutterer, Gerald Roman Berger-Weber, and Walter Friesenbichler. "Fault detection in rubber injection molding with multivariate statistics". In: *Rubber World* (2019), pp. 40–44.

[85] Olga Ogorodnyk and Kristian Martinsen. "Monitoring and Control for Thermoplastics Injection Molding A Review". In: *Procedia CIRP* 67 (2018), pp. 380–385. ISSN: 22128271. DOI: `10.1016/j.procir.2017.12.229`.

[86] Regina Y. Liu. "Control Charts for Multivariate Processes". In: *Journal of the American Statistical Association* 90.432 (1995), p. 1380. ISSN: 01621459. DOI: `10.2307/2291529`.

[87] Harold Hotelling. "The Generalization of Student's Ratio". In: *The Annals of Mathematical Statistics* 2.3 (1931), pp. 360–378. ISSN: 0003-4851. DOI: `10.1214/aoms/1177732979`.

[88] Prasanta Chandra Mahalanobis. "On the Generalized Distance in Statistics". In: *Journal of the Asiatic Society of Bengal* 26 (1930), pp. 541–588.

[89]   Alvin C. Rencher. *Methods of multivariate analysis*. 2. ed. Vol. 1. Wiley series in probability and statistics. Hoboken, NJ: Wiley-Interscience, 2002. ISBN: 0-471-41889-7.

[90]   Pekka Teppola et al. "Principal component analysis, contribution plots and feature weights in the monitoring of sequential process data from a paper machine's wet end". In: *Chemometrics and Intelligent Laboratory Systems* 44 (1998), pp. 307–317. ISSN: 01697439.

[91]   Richard O. Duda and Peter E. Hart. *Pattern classification and scene analysis*. [28. Dr.] A Wiley-Interscience publication. New York: Wiley, 1976. ISBN: 0471223611.

[92]   Ronald Aylmer Fisher. "The use of multiple measurements in taxonimic problems". In: *Annals of Eugenics* 7.2 (1936), pp. 179–188. ISSN: 2050-1439. DOI: `10.1111/j.1469-1809.1936.tb02137.x`. URL: `https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1469-1809.1936.tb02137.x`.

[93]   DIN 53529-1. *Testing of rubber and elastomers; measurement of vulcanization characteristics (curometry); general working principles*. 1983.

[94]   DIN 53529-2. "Testing of rubber and elastomers; measurement of vulcanization characteristics (curometry); evaluation of cross-linking isotherms in terms of reaction kinetics". In: (1983).

[95]   *Rubber meets Science-Tagung*.

[96]   Ping-Yue Wang et al. "Characterization of the aging behavior of raw epoxidized natural rubber with a rubber processing analyzer". In: *Journal of Applied Polymer Science* 100.2 (2006), pp. 1277–1281. ISSN: 00218995. DOI: `10.1002/app.23062`.

[97]   Michael Fasching, Gerald Roman Berger, and Walter Friesenbichler. "Inline Detection of Material Storage Effects on Processing Behavior of Rubber Compounds". In: *Proceedings of the Antec 2015*. Orlando (FL), USA, 2015. ISBN: 978-0-9850112-7-7.

[98]   Michael Fasching, Walter Friesenbichler, and Gerald Roman Berger. "Change of Processing Behavior of Rubbers in Injection Molding caused by Material Storage". In: *Proceedings of the Regional Conference of the Polymer Processing Society*. Graz, Austria, 2015, p. 103.

[99]   DIN ISO 815-1. *Elastomere oder theroplastische Elastomere - Bestimmung des Druckverformungsrestes*. 2010.

[100]   Roman Kerschbaumer, Bernhard Lechner, and Walter Friesenbichler, eds. *Advanced Part Quality Testing - Compression Set vs. Non-Destructive and Fast Measurement Device JIDOKA S101 II*. 2018.

[101]   DIN ISO. *Elastomere oder thermoplastische Elastomere- Bestimmung des Druckverformungs-Verhaltens*. Berlin, 2016-08. DOI: 10.31030/2512357.

[102]   Frank Fuchs. "Anisotherme Spannungsrelaxationsmessung TSSR". In: *KGK Kautschuk Gummi Kunststoffe* 6 (2006), pp. 302–303.

[103]   Caiuã Caldeira de Melo et al. "A novel mechanical test for the stress relaxation analysis of polymers". In: *Polymer Testing* 73 (2019), pp. 276–283. ISSN: 01429418. DOI: 10.1016/j.polymertesting.2018.11.027.

[104]   Roger Brown. *Physical testing of rubber*. 4. ed. New York, NY: Springer, 2006. ISBN: 9780387282862. DOI: 10.1007/0-387-29012-5. URL: http://dx.doi.org/10.1007/0-387-29012-5.

[105]   S. Raa Khimi and Kim Pickering. "A new method to predict optimum cure time of rubber compound using dynamic mechanical analysis". In: *Journal of Applied Polymer Science* 131.6 (2014), n/a–n/a. ISSN: 00218995. DOI: 10.1002/app.40008.

[106]   Kuno A.J. Dijkhuis, Jacques W.M. Noordermeer, and Wilma K. Dierkes. "The relationship between crosslink system, network structure and material properties of carbon black reinforced EPDM". In: *European Polymer Journal* 45.11 (2009), pp. 3302–3312. ISSN: 00143057. DOI: 10.1016/j.eurpolymj.2009.06.029.

[107] Diana Zaimova, Emin Bayraktar, and Nikolay Dishovsky. "State of cure evaluation by different experimental methods in thick rubber parts". In: *Journal of Achievements in Materials and Manufacturing Engineering* 44.2 (2011), pp. 161–167.

[108] Bernhard Praher, Klaus Straka, and Georg Steinbichler. "An ultrasound-based system for temperature distribution measurements in injection moulding: System design, simulations and off-line test measurements in water". In: *Measurement Science and Technology* 24.8 (2013), p. 084004. ISSN: 0957-0233. DOI: 10.1088/0957-0233/24/8/084004.

[109] Youssef Farouq et al. "Temperature measurements in the depth and at the surface of injected thermoplastic parts". In: *Measurement* 38.1 (2005), pp. 1–14. ISSN: 02632241. DOI: 10.1016/j.measurement.2005.04.002.

[110] Thomas Hutterer et al. "Simulative and experimental investigation of rapid heat cycle molding for rubbers". In: *AIP Proceedings* (2018). DOI: 10.1063/1.5084877.

[111] Bernhard Praher and Georg Steinbichler. "Ultrasound-based measurement of liquid-layer thickness: A novel time-domain approach". In: *Mechanical Systems and Signal Processing* 82 (2017), pp. 166–177. ISSN: 08883270. DOI: 10.1016/j.ymssp.2016.05.016.

[112] David Prutchi. *diy "Macro" and "Telephoto Converter Lenses for Thermal Cameras.* 2018.

[113] Waldemar Minkina and Sebastian Dudzik. *Infrared thermography: Errors and uncertainties.* Chichester, West Sussex, U.K and Hoboken, NJ: J. Wiley, 2009. ISBN: 978-0470-74718-6. DOI: 10.1002/9780470682234. URL: http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10344060.

[114] Kistler Group. *p-T-Sensor.* Ed. by Kistler Group. URL: https://www.kistler.com/files/document/000-680e.pdf?callee=frontend.

[115] Friedrich Johannaber and Walter Michaeli. *Handbuch Spritzgießen.* Carl Hanser Verlag GmbH & Company KG, 2014. ISBN: 9783446440982. URL: https://books.google.at/books?id=XcHOAwAAQBAJ.

[116] DIN 53504. *Testing of rubber - Determination of tensile strength at break, tensile stress at yield, elongation at break and stress values in a tensile test.* 2015.

[117] Giorgio Ramorino et al. "Developments in dynamic testing of rubber compounds: assessment of non-linear effects". In: *Polymer Testing* 22.6 (2003), pp. 681–687. ISSN: 01429418. DOI: 10.1016/S0142-9418(02)00176-9.

[118] Sudhir Bafna. "Factors Influencing Hardness and Compression Set Measurements on O-rings". In: *Polymer-Plastics Technology and Engineering* 52.11 (2013), pp. 1069–1073. ISSN: 0360-2559. DOI: 10.1080/03602559.2013.779710.

[119] Ejolle E. Ehabe and Saeed A. Farid. "Chemical kinetics of vulcanisation and compression set". In: *European Polymer Journal* 37.2 (2001), pp. 329–334. ISSN: 00143057. DOI: 10.1016/S0014-3057(00)00112-9.

[120] Saeed Ostad Movahed, Ali Ansarifar, and Farnaz Mirzaie. "Effect of various efficient vulcanization cure systems on the compression set of a nitrile rubber filled with different fillers". In: *Journal of Applied Polymer Science* 132.8 (2015), n/a–n/a. ISSN: 00218995. DOI: 10.1002/app.41512.

[121] Leonhard Perko, Walter Friesenbichler, and Evan Mitsoulis. "Capillary Flow Behavior of a Rubber Compound". In: *Proceedings of the Regional Conference of the Polymer Processing Society.* Tel Aviv, Israel, 2014.

[122] Evan Mitsoulis et al. "Flow behaviour of rubber in capillary and injection moulding dies". In: *Plastics, Rubber and Composites* 46.3

(2017), pp. 110–118. ISSN: 1465-8011. DOI: `10.1080/14658011.2017.1298207`.

[123] Thomas Hutterer, Gerald Roman Berger–Weber, and Walter Friesenbichler. "Determination of the temperature after dosing an industrial rubber compound. A new ultrasonic-based method". In: *Proceedings of DKT 2018*.

[124] Howard A. Barnes. "A review of the rheology of filled viscoelastic systems". In: *Rheology Reviews* (2003), pp. 1–36.

[125] Peter Pelz. *Rheologie der Kautschukmischungen: Elastomer Werkstoffe, 2001*. 2001.

[126] Kwang Jea Kim and James L. White. "Rheological investigations of suspensions of talc, calcium carbonate, and their mixtures in a polystyrene melt". In: *Polymer Engineering & Science* 39.11 (1999), pp. 2189–2198. ISSN: 00323888. DOI: `10.1002/pen.11608`.

[127] Christoph Jaroschek. *Spritzgießen für Praktiker*. 2. Aufl. München: Hanser, 2008. ISBN: 978-3-446-40577-6.

[128] Michael Fasching et al. "Robuste Prozessführung beim Kautschukspritzgießen unter Nutzung systematischer Simulation und verbesserter Materialdaten". In: *Gummi Fasern Kunststoffe* 67.10 (2014), pp. 640–644.

[129] Vincenza Antonucci et al. "Real time monitoring of cure and gelification of a thermoset matrix". In: *Composites Science and Technology* 66.16 (2006), pp. 3273–3280. ISSN: 02663538. DOI: `10.1016/j.compscitech.2005.07.009`.

[130] David D. Shepard and Benjamin Twombly. "Simultaneous dynamic mechanical analysis and dielectric analysis of polymers". In: *Thermochimica Acta* 272 (1996), pp. 125–129. ISSN: 00406031. DOI: `10.1016/0040-6031(95)02622-3`.

[131] Clay Olaf Ruud and Robert E. Green, eds. *Nondestructive Methods for Material Property Determination.* Boston, MA: Springer US, 1984. ISBN: 978-1-4684-4769-9.

[132] Thomas Lanzl. "Charkterisierung von Ruß-Kautschuk-Mischungen mittels dielektrischer Spektroskopie". Dissertation. Regensburg: Universität Regensburg, 2001.

[133] Matthias Jaunich, Wolfgang Stark, and Bernhard Hoster. "Monitoring the vulcanization of elastomers: Comparison of curemeter and ultrasonic online control". In: *Polymer Testing* 28.1 (2009), pp. 84–88. ISSN: 01429418. DOI: 10.1016/j.polymertesting.2008.11.005.

[134] Krishnan Sasidharan et al. "Effect of the vulcanization time and storage on the stability and physical properties of sulfur-prevulcanized natural rubber latex". In: *Journal of Applied Polymer Science* 97.5 (2005), pp. 1804–1811. ISSN: 00218995. DOI: 10.1002/app.21918.

[135] Mario Hermann, Tobias Pentek, and Boris Otto. *Design Principles for Industrie 4.0 Scenarios: A Literature Review.* 2015. DOI: 10.13140/RG.2.2.29269.22248.

Appendices

Source code

Evaluation script of dynamic tests

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy import signal
import os


def Maschinenberechnungen(loadpath, file, versuch):
    #This function extracts the parameter
        calculations done by the testing machine
    rohdaten = pd.read_csv(loadpath + versuch + "\\
        " + versuch + file, delimiter=';', header=0,
                            decimal=',')

    k_stern = rohdaten['DMA-Berechnung:k* (N/mm)'
        ][-5:].mean()
    tand = rohdaten['DMA-Berechnung:Tan Delta'
        ][-5:].mean()

    out = k_stern, tand

    return out
```

```python
def Hysterese_eigen(loadpath,file,versuch, ploton=
    False):
    #This function evaluates the dynamic parameters
        from the hysteresis loop
    rohdaten = pd.read_csv(loadpath + versuch + "\\
        " + versuch + file, delimiter=';', header=0,
         decimal=',')
    segment = rohdaten['Schritt'].values
    zeit = rohdaten['Gesamtzeit (s)'].values
    kraft = rohdaten['Kraft(Linear:Kraft) (N)'].
        values
    weg = rohdaten['Verschiebung(Linear:Digitale
        Position) (mm)'].values
    zyklenzahl = rohdaten['Gesamtzyklenzahl(Linear
        Wellenform)'].values

    zeit_dynamisch = zeit[segment == 6]   # ab
        nummer 2000, weil die ersten zwanzig Zyklen
        die Auslenkung nicht erreicht wurde
    zeit_dynamisch = zeit_dynamisch -
        zeit_dynamisch[0]
    kraft_dynamisch = kraft[segment == 6]
    weg_dynamisch = weg[segment == 6]
    zyklenzahl_dynamisch = zyklenzahl[segment == 6]

    Y = abs(np.fft.fft(kraft_dynamisch))
    N = int(len(Y) / 2 + 1)
    fvec = np.linspace(0, 50, N)

    b, a = signal.butter(2, 2 / 50)
    kraft_smoothed = signal.filtfilt(b, a,
        kraft_dynamisch)
```

```python
# Nur drei runden fuer die Auswertung heran
    ziehen
C_dyn = []
tand = []
for i in range(95,100):
    auswertezyklus = (zyklenzahl_dynamisch >= i
        ) * (zyklenzahl_dynamisch < i+1)
    kraft_auswertung = kraft_smoothed[
        auswertezyklus]
    weg_auswertung = weg_dynamisch[
        auswertezyklus]
    C_dyn.append(
        (np.max(kraft_auswertung) - np.min(
            kraft_auswertung)) / (np.max(
            weg_auswertung) - np.min(
            weg_auswertung)))
    # C_dyn.append(np.max(kraft_auswertung/np.
        max(weg_auswertung)))
    kraftmax = np.argmax(kraft_auswertung)
    kraftmin = np.argmin(kraft_auswertung)
    wegmax = np.argmax(weg_auswertung)
    wegmin = np.argmin(weg_auswertung)

    delta = 2 * np.pi * (wegmax - kraftmax) /
        100
    tand.append(np.tan(delta))

    if ploton:
        plt.plot(weg_auswertung,
            kraft_auswertung)
```

```python
            plt.plot(weg_auswertung[wegmax],
                kraft_auswertung[kraftmax], 'o')
            plt.plot(weg_auswertung[wegmin],
                kraft_auswertung[kraftmin], 'o')

            plt.show()

    C_final = np.mean(C_dyn)
    tand_final = np.mean(tand)

    out = C_final, tand_final
    return out


def Relaxation(loadpath,file,versuch,ploton=False):
    #This function evaluates the relaxation part of
        the test routine
    rohdaten = pd.read_csv(loadpath + versuch + "\\
        " + versuch + file, delimiter=';', header=0,
                            decimal=',')
    segment = rohdaten['Schritt'].values
    zeit = rohdaten['Gesamtzeit (s)'].values
    kraft = rohdaten['Kraft(Linear:Kraft) (N)'].
        values
    weg = rohdaten['Verschiebung(Linear:Digitale
        Position) (mm)'].values

    zeit_relaxation = zeit[segment == 5]
    zeit_relaxation = zeit_relaxation -
        zeit_relaxation[0]
    kraft_relaxation = kraft[segment == 5]
    weg_relaxation = weg[segment == 5]
```

```python
    max_kraft = -(np.min(kraft_relaxation))
    relax_kraft = -(kraft_relaxation[-1])

    if ploton:
        plt.plot(zeit_relaxation,kraft_relaxation)
        plt.show()

    out = max_kraft, relax_kraft

    return out


def Statisch(loadpath,file,versuch, ploton = False)
    :
    #This function evaluates the quasi-static
        portion of the testing routine
    rohdaten = pd.read_csv(loadpath + versuch + "\\
        " + versuch + '.steps.tracking.csv',
        delimiter=';', header=0,
                            decimal=',')
    segment = rohdaten['Schritt'].values
    zeit = rohdaten['Gesamtzeit (s)'].values
    kraft = rohdaten['Kraft(Linear:Kraft) (N)'].
        values
    weg = rohdaten['Verschiebung(Linear:Digitale
        Position) (mm)'].values

    kraft_indent = kraft[segment == 4]
    weg_indent = weg[segment == 4]

    start = np.argmax(kraft_indent)
    kraft_start = np.max(kraft_indent)
    ende = np.argmin(kraft_indent)
```

```python
        kraft_ende = np.min(kraft_indent)
        weg_start = weg_indent[start]
        weg_ende = weg_indent[ende]

        if ploton:
            plt.plot(weg_indent,kraft_indent)
            plt.plot(weg_start,kraft_ende, 'o')
            plt.plot(weg_ende,kraft_ende,'o')

        C_stat = (kraft_ende - kraft_start) / (weg_ende
            - weg_start)

        return C_stat


#File names
maschinen_file = '.steps.trends.csv'
relax_file = '.steps.tracking.csv'
hysterese_file = '.stop.csv'
projekt = """20191218_8mm20pro_T1"""


#Paths for loading files
loadpath = ("""P:\\Rohdaten\\AG Hutterer\\
    Praktische_Experimente\\20190826
    _ClampControlEngel\\Werkstoffpruefung\\Jidoka\\
    """
                + projekt + """\\""") #Das ist der
                    Pfad an dem die Messfile Ordner
                    liegen muessen, wichtig: \\ fuer
                    ebenen trennen und am ende \\
savepath = ("""P:\\Rohdaten\\AG Hutterer\\
    Praktische_Experimente\\20190826
    _ClampControlEngel\\verarbeitete_Daten\\Jidoka\\
```

```python
    """) # Das ist der Pfad wo du deine csvs hin
    haben willst
dateien = os.listdir(loadpath)


Ausgabedaten = []


#Main loop over all files calling all evaluation
    functions
for item in dateien:
    print(item)

    kstern, tand = Maschinenberechnungen(loadpath,
        maschinen_file,item)
    max_kraft, relax_kraft = Relaxation(loadpath,
        relax_file,item)
    keig,tandeig = Hysterese_eigen(loadpath,
        hysterese_file, item)
    kstat = Statisch(loadpath,relax_file, item)

    Ausgabedaten.append([item, kstern, tand,
        max_kraft, relax_kraft, keig, tandeig, kstat
        ])


#Cleanup and save outfile
cols=("Name", "Cdyn_inst", "tand_inst", "Max_Force"
    , "Relax_Force", "Cdyn_eig", "tand_eig", 'Cstat'
    )
out = pd.DataFrame.from_records(Ausgabedaten,
    columns = cols)


out.to_csv(savepath +projekt + "_results.csv", sep=
    '\t')
```

Functions shared by other routines

```python
import numpy as np
import scipy.signal as scs


class dataprocessing(object):

    def interp_messdaten(self,fsample,xdaten,
        ydaten):
        #Data collected with Lucid IO devices need
            to be brought to even sampling rates
        a = int(np.floor(xdaten[-1]))
        x = np.linspace(0,a,a*fsample+1)
        sampled_data = np.interp(x,xdaten,ydaten)
        return [x,sampled_data]

    def filt_messdaten(self,daten,par1,par2):
        #Data collected with Lucid IO devices need
            to be filtered
        filtdata = scs.symiirorder1(daten,par1,par2
            )
        return filtdata

    def timeslice_pca(self,daten,batchrange):
        #This function takes a numpy array with
            required shape: axis0 = batches
        #Axis1 = process variables
        #Axis2 = samples
        #Returned arrays are eigenvalues, and P and
            T trajectories
```

```python
        eigenwerte = []
        loadings =[]
        w_loadings = []
        for ii in range(daten[0,0,:].shape[0]):
            X = daten[batchrange[0]:batchrange
                [1],:,ii] #Take a timeslice
            S = 1 / float(X.shape[0] - 1) * np.
                matmul(X.T , X)   # Calculate its
                covaraiance
            V, D, Vt = np.linalg.svd(S)  # SVD
            Dnorm = np.diag(D / np.sum(D))  #
                Reshape of D to be diag and 0,1
                scaled
            Pweighted = np.matmul(V,Dnorm)  #
                Calculate the weighted loadings
            eigenwerte.append(D)
            loadings.append(Vt)
            w_loadings.append(Pweighted)

        eigenwerte = np.asarray(eigenwerte)
        eigenwerte = np.swapaxes(eigenwerte,0,1)
        loadings = np.asarray(loadings)
        loadings = np.swapaxes(loadings,0,2)
        w_loadings = np.asarray(w_loadings)
        w_loadings = np.swapaxes(w_loadings,0,2)

        return eigenwerte, loadings, w_loadings


    def normalize(self, gesamtdaten, starttrain,
        endtrain, batches, signals, samples):
        #This function normalizes the process
            variable array batch-wise
```

```
batchmw = np.mean(gesamtdaten[starttrain:
    endtrain, :, :], axis=0)
batchmw = np.broadcast_to(batchmw, (batches
    ,
                                    signals
                                    ,
                                    samples
                                    ))
                                    #
                                    Broadcasting

batchnorm = gesamtdaten - batchmw  # Zero
    mean
batchstd = np.std(np.std(batchnorm[
    starttrain:endtrain,:,:], axis=2),axis =
     0) #Standard deviation is calculated
    from the trajectory
batchstd = np.broadcast_to(batchstd, (
    samples, batches, signals))  #
    Broadcasting
batchstd = np.swapaxes(batchstd, 0, 1)
batchstd = np.swapaxes(batchstd, 1, 2)
batchnorm = batchnorm / batchstd  #
    Normalize to unit variance

return batchnorm,batchmw,batchstd
```

Phase detection

```
import numpy as np
import data_processing
```

```python
from os import listdir
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt


#Preparation and loading
dpc = data_processing.dataprocessing()
#Number of expected samples
anz_datenpunkte = 35000
#Sampling frequency
fsample = 100
'Loading'
loadpath = 'P:\\Rohdaten\\AG Hutterer\\
   Praktische_Experimente\\20190826
   _ClampControlEngel\\verarbeitete_Daten\\
   Prozesssignale\\Dienstag\\'
savepath = 'P:\\Rohdaten\\AG Hutterer\\
   Praktische_Experimente\\20190826
   _ClampControlEngel\\verarbeitete_Daten\\PCA\\'


#Loop preparations
datenliste = listdir(loadpath)
moddaten = []
gesamtdaten = []
n = 0

#Looping over raw data
for daten in datenliste:

    #Reading data
    n+= 1
    rohdaten = np.genfromtxt((loadpath + daten),
```

```
        delimiter = '\t')
    rohdaten = rohdaten[0:anz_datenpunkte ,:]
    moddaten = []#alle 2d matrizzen zu 3d matrix
        machen axis0 = batches axis1 = datenpunkte
        axis2 = variablen

    #Filtering and Interpolation
    for ii in range(1,rohdaten.shape[1]):

        zeit, yinterp = dpc.interp_messdaten(
            fsample ,rohdaten[:,0],rohdaten[:,ii]) #
            Interpolieren auf 100 Hz
        yinterp = dpc.filt_messdaten(yinterp
            ,0.01,0.9) #Filtern
        moddaten.append(yinterp[zeit <110]) #in
            liste speichern: listeneintraege sind
            datenstream

    interpdaten = np.asarray(moddaten).T #alle
        daten in 2d matrix speichern. axis0 =
        datenpunkte axis1 = datenstrems
    dpinj = np.gradient(interpdaten[:,4])
    start = np.nonzero(dpinj > 0.25)
    interpdaten2 = interpdaten[start[0][0]:start
        [0][0] + 9000, :]
    gesamtdaten.append(interpdaten2)

#Combine data into array
gesamtdaten = np.asarray(gesamtdaten)
gesamtdaten = np.swapaxes(gesamtdaten ,0,1)

#Normalize data time-wise
```

```
timemw = gesamtdaten -np.mean(gesamtdaten,axis=0) #
    abziehen der Trajectory mittelwerte
timenorm = timemw / np.std(gesamtdaten, axis = 0)
timenorm = np.nan_to_num(timenorm)
timenorm = np.swapaxes(timenorm,0,1)
timenorm = np.swapaxes(timenorm,1,2)


#PCA
eigenwerte, loadings, w_loadings = dpc.
    timeslice_pca(timenorm,(30,40))


#Clustering
clusters = 4
loadings_unfold =w_loadings.reshape(loadings.shape
    [0]*loadings.shape[1],loadings.shape[2]).T #
    loadings aufklappen zu Zeilenvektor
kmeans = KMeans(n_clusters = clusters, random_state
    =0).fit(loadings_unfold)
phasen = kmeans.labels_


#Output preparation
gesamtdaten = np.swapaxes(gesamtdaten,0,1)
gesamtdaten = np.swapaxes(gesamtdaten,1,2)


#Saving
np.save((savepath +'gesamtdaten.npy'), gesamtdaten)
np.save((savepath+'eigenwerte.npy'), eigenwerte)
np.save((savepath+'loadings.npy'), loadings)
np.save((savepath+'w_loadings.npy'), w_loadings)
np.save((savepath+'phasen.npy'), phasen)
np.save((savepath+'timenorm.npy'), timenorm)
```

151

Off line process monitoring routine for development

```
import numpy as np
import data_processing
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
from scipy.stats import f
from scipy.stats import chi2




path = 'P:\\Rohdaten\\AG Hutterer\\
   Praktische_Experimente\\20190826
   _ClampControlEngel\\verarbeitete_Daten\\PCA\\'
dpc = data_processing.dataprocessing()
#Read data from phase separation
gesamtdaten = np.load(path+'gesamtdaten.npy')
phasen = np.load(path+'phasen.npy')
timenorm = np.load(path+'timenorm.npy')

#Reduce the number of signals
gesamtdaten = gesamtdaten[:,0:5,:]

ges_batches = 58#Number of batches
start_train = 0 #Begin of training
end_train = 15 #Stop training
no_samples = 9000 #Number of samples
no_signals = 5 #Number of signals
numberofPCs = 2 #PCs to be kept
untersuchtePhase = 3 #Phase detected by
   phaseseparation, which is of interest
```

```python
# Empty lists for later use
Prepreds = [None]*4
eigenwerterepreds = [None]*4
eigenwertereprests = [None]*4
evrepredinvs = [None]*4
SPE_batches = []


#Normalization
batchnorm, trainmw,trainstd = dpc.normalize(
    gesamtdaten,start_train,end_train,ges_batches,
    no_signals, no_samples)


#Put the phases in timewise order
phasen[-2] = 100 #To detect final phase
log = phasen[0:-2] != phasen[1:-1] #Detect phase
    change
log = np.nonzero(log)
phasenabfolge = phasen[np.subtract(log,1)]


for ii in range(0,phasenabfolge.shape[1]):
    actphase = phasenabfolge[0,ii] #Call only
        samples from specific phase
    normphase = batchnorm[:,:,phasen==actphase]

    phase_eigenwerte,phase_loadings,phase_wloadings
         = dpc.timeslice_pca(normphase,(start_train,
        end_train)) #PCA

    Prep = np.mean(phase_loadings,axis = 2) #Phase
        representative loadings
    eigenwerterep = np.mean(phase_eigenwerte,axis
        =1).reshape(1,-1) #Phase representative
```

```
        eigenvalues

    Prepred = Prep [: ,: numberofPCs]  #Dimensionality
        reduction
    Prepreds [actphase] = Prepred  #Put
        representative  loadings  on  their  phase's
        respective  list  slot
    eigenwerterepred = np.diag(eigenwerterep.
        reshape(-1)[:numberofPCs]) #Reshape  to  diag
    eigenwertereprest = eigenwerterep.reshape(-1)[
        numberofPCs :] #Dimensionality  reduction
    # Match  list  spot  of  respective  phase
    eigenwerterepreds [actphase] = eigenwerterepred
    eigenwertereprests [actphase] =
        eigenwertereprest
    evrepredinv = np.linalg.inv(eigenwerterepred)
    evrepredinvs [actphase] = evrepredinv

#Calculate  the  95%  control  bound  of  the  Tsquared
    statistic
n_batches = end_train - start_train
len_phase = np.asarray(np.asarray(np.nonzero(phasen
    ==untersuchtePhase)).shape[1])
Tc = (numberofPCs*n_batches*(len_phase -1))/(
    n_batches*(len_phase -1)-numberofPCs)*f.ppf(0.95,
    numberofPCs ,n_batches*(len_phase -1)-numberofPCs)


#Set  up  variables
Xpred_gesamt = []
T_gesamt = []
T2_gesamt = []
```

```python
SPE_gesamt = []
E_gesamt = []
for ii in range(0,ges_batches): #All batches are
   now evaluated separately
    #Set up more variables
    Xkpred_trajectory = []
    T_trajectory = []
    T2_trajectory = []
    E_trajectory = []
    SPE_trajectory = []
    batchdaten = batchnorm[ii,:,:] #Grab normalized
        timeslices
    print(ii+1)
    for nn in range(0,batchdaten.shape[1]-2): #time
       -slice wise calculation
        Xk = batchdaten[:,nn] #grab single
            timeslice
        Tred = np.matmul(Xk, Prepreds[phasen[nn]])
            #Calculate score
        Xkpred = np.matmul(Tred, Prepreds[phasen[nn
            ]].T) #Calculate predicted signal values
        Xkpred_trajectory.append(Xkpred) #Form
            predicted trajectory

        Tred = Tred.reshape(-1, 1)
        E = Xk - Xkpred  #Calculate the prediction
            error
        SPE = np.matmul(E.T, E) #Calculate SPE
        T2= np.matmul(Tred.T,np.matmul(evrepredinvs
            [phasen[nn]],Tred)) #Calculate the
            Tsquared statistic
        #Form statistics trajectories
```

```
        T_trajectory.append(np.squeeze(Tred))
        T2_trajectory.append(np.squeeze(T2))
        E_trajectory.append(E)
        SPE_trajectory.append(SPE)


    #Array formation
    Xkpred_trajectory = np.asarray(
        Xkpred_trajectory)
    T_trajectory = np.asarray(T_trajectory)
    SPE_trajectory = np.asarray(SPE_trajectory)
    T2_trajectory =np.asarray(T2_trajectory)
    SPE_gesamt.append(SPE_trajectory)
    E_trajectory = np.asarray(E_trajectory)
    E_gesamt.append(E_trajectory)
    T_gesamt.append(T_trajectory)
    Xpred_gesamt.append(Xkpred_trajectory)
    T2_gesamt.append(T2_trajectory)

Xpred_gesamt = np.asarray(Xpred_gesamt)
T_gesamt = np.asarray(T_gesamt)
E_gesamt = np.asarray(E_gesamt)
SPE_gesamt = np.asarray(SPE_gesamt)
T2_gesamt = np.asarray(T2_gesamt)



#SPE control limit calculation at 95%
SPE_gesamt = np.asarray(SPE_gesamt)
mean_SPE = np.mean(SPE_gesamt[start_train:end_train
    , :], axis = 0)
std_SPE = np.var(SPE_gesamt[start_train:end_train,
    :],axis = 0)
gk = std_SPE/(2*mean_SPE)
```

```
hk = 2*( mean_SPE )**2/ std_SPE
SPE_c = gk*chi2.ppf (0.95 , hk )


#Variable  setup
badbatch = np.zeros (( ges_batches ,1))
deltaspe = np.zeros (( ges_batches ,1))
t_goodness = np.zeros (( ges_batches ,1))
contribution = []
phasen = phasen [: -2]


#Calculate  Delta  SPE,  contributions  and  T2  values
   of  each  batch
for ww in range (0 , ges_batches ):
    print ( ww )
    deltaspe [ ww ] = np.mean ( SPE_gesamt [ww , phasen ==
         untersuchtePhase ][: -100] - SPE_c [ phasen ==
        untersuchtePhase ][: -100])
    fi = np.mean ( E_gesamt [ww , phasen ==
        untersuchtePhase ,:] , axis = 0)

    contribution.append (fi )
    ti = ( np.max ( T2_gesamt [ww ,phasen ==
        untersuchtePhase ]))
    t_goodness [ ww ] = ti


contribution = np.asarray ( contribution )


#Saving  data
np.save ( path + 'Xp_gesamt.npy ', Xpred_gesamt )
np.save ( path + 'T_gesamt.npy ',T_gesamt )
np.save ( path + 'E_gesamt.npy ', E_gesamt )
np.save ( path + 'SPE_gesamt.npy ', SPE_gesamt )
```

```
np.save(path + 'deltaspe.npy', deltaspe)
np.save(path + 'contribution.npy', contribution)
np.save(path + 't_goodness.npy', t_goodness)
```

FDA analysis

```python
import numpy as np
import matplotlib.pyplot as plt
import data_processing
from sklearn.model_selection import
    train_test_split

dpc = data_processing.dataprocessing()


#Import the data
Xdata = np.load('Einspritzdaten20190319.npy')
ydata= np.genfromtxt('Klassen201903.txt')

#Split the data set in Test and train data
Xtrain, Xtest, ytrain, ytest = train_test_split(
    Xdata, ydata, test_size=0.1)

#Normalize
Xtrain_norm, Xtrain_mw, Xtrain_std = dpc.normalize(
    Xtrain,0,Xtrain.shape[0],64,4,600)

#Loop across multiple timeslices for stability
farbe = ['b', 'r', 'k', 'g']
startslice = 300
w_trajectory = []
```

```python
for k in range(0,Xtrain_norm.shape[2]):
    #Grab timeslice
    xtrain = Xtrain_norm[:,:,k]

    #Calculate class mean
    mu_c = []
    for i, c in enumerate(np.unique(ytrain)):
        l = ytrain==c
        mu_c.append(np.mean(xtrain[l],axis = 0))


    mu_c = np.array(mu_c)


    single_SW = []
    Nc = []

    #Calculate the scatter matrices
    for i, c in enumerate(np.unique(ytrain)):
        l = ytrain == c
        a = xtrain[l]-mu_c[i]
        single_SW.append(a.T@a)
        Nc.append(np.sum(l == True))

    SW = np.sum(single_SW, axis = 0)
    SB = Nc * mu_c.T @ mu_c

    #Eigenvectors
    eigval, eigvec = np.linalg.eig(np.linalg.inv(SW
        ) @ SB)
    #Find eigenvector-eigenvalue pairs
    eigen_pairs = [[np.abs(eigval[i]), eigvec[:, i
        ]] for i in range(len(eigval))]
```

```
    eigen_pairs = sorted(eigen_pairs, key=lambda k:
        k[0], reverse=True)
    w = np.hstack((eigen_pairs[0][1][:, np.newaxis
        ].real, eigen_pairs[1][1][:, np.newaxis].
        real)) # Select the 2 largest
    w_trajectory.append(w)



#Calculate the mean transformation matrix
w_trajectory = np.asarray(w_trajectory)
w_mu = np.mean(w_trajectory, axis = 0)

#Transform the data
Y = Xtrain_norm[:,:,startslice] @ w_mu


m = ['x', 's', 'o', '1', 'p']
for i in range(0, Y.shape[0]):
    plt.scatter(Y[i, 0], Y[i, 1], marker=m[int(
        ytrain[i])], color= 'k')



#Testing
Itest = Xtest.shape[0]
Xtest_norm = (Xtest-Xtrain_mw[:Itest,:,:])/
    Xtrain_std[:Itest,:,:]
Ytest = Xtest_norm[:,:,startslice] @ w_mu
plt.scatter(Ytest[0,0],Ytest[0,1], marker = m[int(
    ytest[0])], color = 'orange')
plt.show()

#saving
ausgabe1 = np.hstack((Y,ytrain.reshape(-1,1)))
```

```
ausgabe2 = np.hstack((Ytest,ytest.reshape(-1,1)))
ausgabe = np.vstack((ausgabe1,ausgabe2))
np.save('FDA_transformed.npy',ausgabe)
```