Stochastic Optimal Control of an Optical Sorter with Material Recirculation

Markus Walker, Marcel Reith-Braun, Albert Bauer, Florian Pfaff *Senior Member, IEEE*, Georg Maier, Robin Gruna, Thomas Längle, Jürgen Beyerer *Member, IEEE*, Harald Kruggel-Emden, and Uwe D. Hanebeck *Fellow, IEEE*

***Abstract*—Optical bulk material sorting is a key technology on our way toward a circular economy and efficient recycling. However, controlling the sorting accuracy has so far been severely limited, as the achievable accuracy of conventional sorters is strongly determined by the mass flow and the mixing ratio of the incoming particle stream. To enable closed-loop control, in previous work, we introduced a modification to the sorter design in which controlled fractions of the already sorted mass flows are returned to the inlet of the sorter. In this article, we now propose two open-loop and two closed-loop feedback stochastic model predictive controllers for the control of sorting systems with recirculation operating under dynamically changing conditions. In addition, we propose to integrate a desired minimum accuracy as a chance constraint into our controllers' stochastic formulation. Our evaluations using a coupled discrete element–computational fluid dynamics simulation show that our controllers considerably improve on the system without recirculation and outperform the previously known controllers. Furthermore, we found that they are able to maintain a predefined minimum quality even in highly dynamic scenarios, making the approach highly valuable for tasks where achieving a certain quality at any point in time is crucial.**

***Index Terms*—Stochastic optimal control, optical sorting, finite-horizon, open-loop feedback, closed-loop feedback, global approaches, trajectory optimization.**

I. Introduction

OPTICAL SORTING combines machine vision and motion forecasting to physically separate bulk material of different classes, usually by a binary division into desirable and undesirable classes. A typical optical sorter consists of a transport unit, such as a belt or a chute, a camera that analyzes the particle flow on the transport unit, and a separation unit. Usually, the latter consists of a bar equipped with compressed air nozzles, which is mounted behind the transport unit and ejects particles of undesired classes with bursts of compressed air (see Fig. 1 and Fig. 2) [1]. Optical sorters are widely applied in the mineral [2], [3], food [4], and recycling industries [5]. In particular, the recycling sector is registering a substantial and growing demand. This is mainly due to global waste pollution being considered as one of the most pressing problems of our time. It is indeed expected that the demand will increase in the future due to new environmental legislation, such as the European Union's recycling directive that stipulates that recycling rates of 70 % for packaging waste by 2030 and

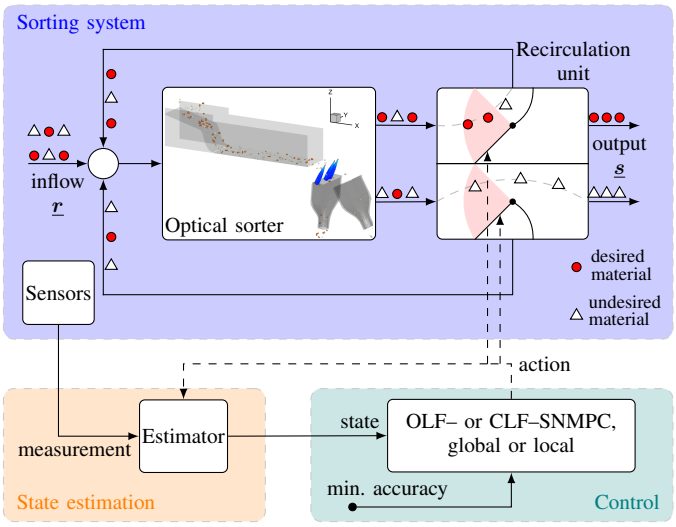


Fig. 1. Structure of the sorting system with recirculation and the stochastic nonlinear model predictive controller (SNMPC) used for closed-loop control. The sorting system consists of an optical sorter, which separates a particle stream into two classes, and a recirculation unit (illustrated by the flaps), which returns controlled fractions of each of the two sorted mass flows to the inlet of the sorter. As controllers, we consider global and local stochastic model predictive controllers with either open-loop feedback (OLF) or closed-loop feedback (CLF). The goal of the controller is to maximize the overall sorting accuracy, while additional chance constraints specifying a minimum of falsely sorted particles in either of the two output mass flows can be specified. For our study, we consider a coupled discrete element–computational fluid dynamics simulation model of the sorter (as illustrated in the cutout) and the overall sorting model (as illustrated in Fig. 2).

The IGF project 20354 N of the research association Forschungs-Gesellschaft Verfahrens-Technik e.V. (GVT) was supported via the AiF in a program to promote the Industrial Community Research and Development (IGF) by the Federal Ministry for Economic Affairs and Climate Action on the basis of a resolution of the German Bundestag. *(Markus Walker and Marcel Reith-Braun contributed equally to this work.) (Corresponding author: Marcel Reith-Braun.)*

Markus Walker, Marcel Reith-Braun, and Uwe D. Hanebeck are with the Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany (e-mail: markus.walker@kit.edu; marcel.reith-braun@kit.edu; uwe.hanebeck@kit.edu).

Florian Pfaff is with the Institute of Industrial Automation and Software Engineering, University of Stuttgart, 70569 Stuttgart, Germany (e-mail: pfaff@ias.uni-stuttgart.de).

Albert Bauer and Harald Kruggel-Emden are with the Chair of Mechanical Process Engineering and Solids Processing (MVTA), Technische Universität Berlin, 10587 Berlin, Germany (e-mail: a.bauer@tu-berlin.de; kruggel-emden@tu-berlin.de).

Georg Maier, Robin Gruna, Thomas Längle, and Jürgen Beyerer are with the Fraunhofer Institute of Optronics, System Technologies and Image Exploitation (IOSB), 76131 Karlsruhe, (e-mail: georg.maier@iosb.fraunhofer.de; robin.gruna@iosb.fraunhofer.de; thomas.laengle@iosb.fraunhofer.de; juergen.beyerer@iosb.fraunhofer.de).

Manuscript received February 13, 2024; revised July 26, 2024; accepted September 24, 2024.

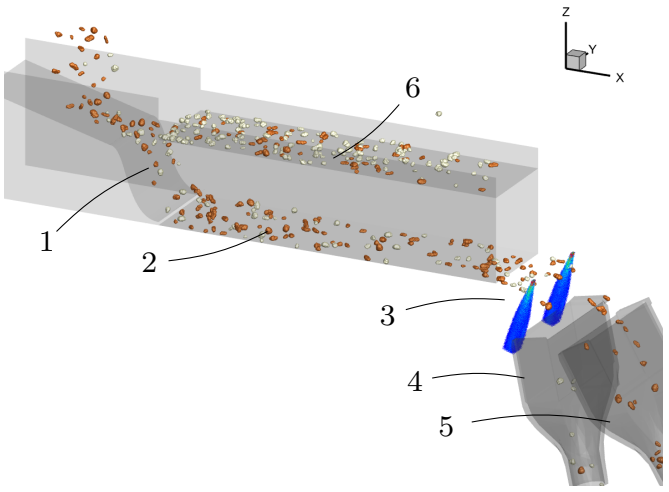


Fig. 2. DEM–CFD model used for our simulations of the sorting system with controlled material recirculation. The input mass flow is fed onto the feeding chute (1) and then transported by the lower of the two conveyor belts (2) to the nozzles (3). Here, particles of undesired classes are ejected into the reject hopper (4) by bursts of compressed air, while particles not ejected fall into the accept hopper (5). The material recirculation is modeled by a second conveyor belt (6) that introduces a time delay. For this, a controlled fraction of the sorted particles are moved instantaneously from the hoppers (4), (5) to the belt (6), which returns them to the chute (1).

$65\,\%$ for municipal waste by 2035 are to be achieved [6]. To this end, optical sorting is viewed as a key technology in the development of a sustainable circular economy [5], [6].

A key challenge in optical sorting is achieving a desired sorting accuracy while still maintaining a generally high throughput rate. The sorting accuracy can be measured in terms of the true negative rate $\mathrm{TNR} = \mathrm{TN}/(\mathrm{FP} + \mathrm{TN})$ and the true positive rate $\mathrm{TPR} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$, where we refer to positive particles as those that should not or are not ejected, and $\mathrm{TP}, \mathrm{TN}, \mathrm{FP}$, and FN denote true positives, true negatives, false positives, and false negatives particles after sorting (note that we consider binary sorting tasks throughout this article). A high TNR is typically requested when removing contaminants is crucial, such as in food processing, while achieving a high TPR is of importance when positive particles are of high value, e.g., in diamond sorting [3]. Besides sufficiently accurate image processing, successful separation depends on the spatial and temporal distances between the particles at the separation unit. Thus, TNR and TPR are mainly affected by the type of particles, the mass flow processed, and the mixing ratio between the classes [7]. Apart from that, a higher TNR can be achieved by trading it for a lower TPR and vice versa, which is accomplished by manually altering system parameters, such as the deflection pattern, before the actual sorting process [8]. Besides being costly and not guaranteeing the desired accuracy during deployment, the impact of manual adjustments is limited. When a desired accuracy cannot be achieved, one is therefore usually forced to reduce mass flow or use several stages of sorting, e.g., by cascading sorters [3].

A second challenge arises in dynamic scenarios since changes in mass flow and material composition automatically affect TNR and TPR. Therefore, consistent quality over time cannot be ensured. In particular, this is a severe problem when sorters are integrated into continuous bulk material processing plants rather than operate in batch mode, i.e., when the sorted mass flows are immediately processed further instead of first being collected in separate bins before being fed to the next processing step. In situations where certain minimum sorting accuracies are required, manual re-sorting of already sorted material is usually the only option remaining.

In our previous work [9], we proposed to meet the first key challenge using a closed-loop control approach. For this, a controlled fraction of both the accepted and rejected mass flow is recirculated to the inlet of the sorter. For controlling the recycled fractions, a deterministic model predictive controller (MPC) is employed that is fed with measurements providing information about the compositions of the mass flow at the transport unit and after the separation. The goal of the MPC is to increase a weighted sum of TNR and TPR over a finite control horizon. Results of our approach show that we were able to improve either TNR or TPR of the closed-loop system, depending on which fraction one desires to improve.

In this article, we extend the aforementioned approach (see Fig. 1 for an illustration) to also address the problems arising from the second key challenge. For this purpose, we propose a stochastic model of the sorting system, in combination with a stochastic nonlinear model predictive controller (SNMPC) for closed-loop control of the sorting system. The use of stochastic modeling and stochastic optimal control (SOC) instead of a deterministic controller offers two major benefits:

- *stochastic nature of the sorting system captured*: an optical sorter is a highly complex system. Capturing all influences on particles requires a highly complex and computationally demanding model that is unsuitable for an MPC. A simple, but uncertainty-aware model accounts for the incomplete information, disturbances, and model and discretization errors that are naturally introduced in any kind of model. It thus prevents control decisions based on overconfident predictions while still being deployable in an MPC.
- *possibility to specify chance constraints for a minimum TNR or TPR*: Using a stochastic model and controller naturally allows specifying chance constraints for a minimum TNR or TPR that should be respected with a desired level of confidence. Note that similar specifications cannot be incorporated into a purely deterministic, but erroneous model.

We evaluate the controllers in static and dynamically changing scenarios by means of an extensive simulation study, including scenarios with changing mass flow as well as fluctuating mixing ratios. For this, we build a coupled discrete element–computational fluid dynamics (DEM–CFD) simulation model of a sorting system with recirculation (see Fig. 2) and compare the results to a DEM–CFD model of an optical sorter without recirculation.

Contributions: First, we develop a stochastic model of the sorting system with recirculation and propose four variants of an SNPMC for closed-loop control. In this regard, we propose a novel, fitness proportional discretization method for the control inputs of global SNMPCs. Second, we demonstrate superior sorting accuracies on static and dynamically changing scenarios compared with both the sorter without recirculation and the

previously known deterministic MPC through an extensive DEM–CFD simulation study. Third, we show how a minimum TNR or TPR that the sorting system should exceed at any time can be incorporated into the SNMPCs as a chance constraint. Finally, we experimentally demonstrate that this lower bound is respected with the desired confidence even in highly dynamically changing scenarios.

Notation: Throughout this article, vectors will be indicated by underlined letters, e.g., $\underline{x}$, boldface letters, such as $\boldsymbol{x}$, will represent random variables, and boldface capital letters, e.g., $\mathbf{A}$, will indicate matrices. As a shorthand notation, we write $\boldsymbol{x} \in \mathcal{A}$ if we mean that the set $\mathcal{A}$ is the sample space of $\boldsymbol{x}$. To label a measurement, i.e., a realization, of a random variable $\boldsymbol{x}$, we write $\hat{x}$. The $\mathrm{diag}\{\underline{x}\}$ operator denotes a square diagonal matrix with the elements of the vector $\underline{x}$ on the main diagonal and all other elements equal to zero. To denote a sequence over n time steps starting at time step k, we will use the index $k, 0{:}n$. For instance, $x_{k,0:n}$ denotes the sequence $x_k, \ldots, x_{k+n}$. Similarly, the index k, n refers to time step $k+n$.

Article Overview: The remainder of the article is organized as follows. In Sec. II, we state the general problem of optical sorting with material recirculation and introduce the generic SOC framework, into which we will transfer the sorting problem in the subsequent sections. We then explicitly describe the challenges in solving the problem in Sec. III. To keep our previous work [9] separate from the contributions first published in this article, in Sec. IV, we first summarize the related work regarding both SOC and optical sorting, including the previous work on controlled material recirculation. In Sec. V, we then address the probabilistic modeling and identification of the sorting system, which is original to this work. Afterward, we introduce our SNMPC approaches in Sec. VI. Finally, we present and discuss our results in Sec. VII and Sec. VIII, and conclude in Sec. IX.

II. Problem Statement

A. Generic SOC Problem

We formulate the control task as a discrete-time, finite-horizon SOC problem with imperfect state feedback. We consider the general nonlinear stochastic system and measurement equations

$$\underline{\boldsymbol{x}}_{k+1} = \underline{a}_k(\underline{\boldsymbol{x}}_k, \underline{u}_k, \underline{\boldsymbol{w}}_k) \ , \tag{1}$$

$$\underline{\boldsymbol{z}}_k = \underline{h}_k(\underline{\boldsymbol{x}}_k, \underline{\boldsymbol{v}}_k) \ , \tag{2}$$

where $\underline{\boldsymbol{x}}_k \in \mathbb{R}^{n_{\underline{\boldsymbol{x}}}}$ and $\underline{\boldsymbol{z}}_k \in \mathbb{R}^{n_{\underline{\boldsymbol{z}}}}$ are random vectors denoting the system's state and measurements, respectively, at time step k, $\underline{u}_k \in \mathcal{A}$ is the control input, or action, and $\underline{\boldsymbol{w}}_k \in \mathbb{R}^{n_{\underline{\boldsymbol{w}}}} \sim f_k^{\underline{\boldsymbol{w}}}(\underline{w}_k)$ and $\underline{\boldsymbol{v}}_k \in \mathbb{R}^{n_{\underline{\boldsymbol{z}}}} \sim f_k^{\underline{\boldsymbol{v}}}(\underline{v}_k)$ are random disturbances. The state $\underline{\boldsymbol{x}}_k$ is not directly accessible, i.e., we can only gain information about the system state via measurements $\hat{\underline{z}}_k$ of $\underline{\boldsymbol{z}}_k$, and by assuming that the initial state is distributed according to $\underline{\boldsymbol{x}}_0 \sim f_0^{\underline{\boldsymbol{x}}}$. Decision problems involving systems described by (1) and (2) are often referred to as *partially observable Markov decision process* (POMDP), in contrast to a *Markov decision process* (MDP), i.e., decision problems in which the state is fully observable, and therefore, no measurement equation needs to be considered.

Let us introduce the information set $\mathcal{I}_k$ defined as $\mathcal{I}_k = \left\{ f_0^{\underline{\boldsymbol{x}}}, \hat{\underline{z}}_{0:k}, \underline{u}_{0:k-1} \right\}$ for $k > 0$, and $\mathcal{I}_0 = \left\{ f_0^{\underline{\boldsymbol{x}}}, \hat{\underline{z}}_0 \right\}$. The goal of SOC is to find a sequence of control laws, or policies, $\underline{\mu}_{k,0:N-1}^{\mathrm{CLF}}$, with $\underline{\mu}_{k,n}^{\mathrm{CLF}}(\mathcal{I}_{k,n}) \in \mathcal{A}$, that map $\mathcal{I}_{k,n}$ to actions $\underline{u}_{k,n} = \underline{\mu}_{k,n}^{\mathrm{CLF}}(\mathcal{I}_{k,n})$ so that the *closed-loop feedback* (CLF) objective function

$$J_k^{\mathrm{CLF}} = \mathrm{E}\Bigg\{ g_N(\underline{\boldsymbol{x}}_{k,N}) + \sum_{n=0}^{N-1} g_n\left(\underline{\boldsymbol{x}}_{k,n}, \underline{\mu}_{k,n}^{\mathrm{CLF}}(\mathcal{I}_{k,n})\right) \Bigg| \mathcal{I}_k \Bigg\} \tag{3}$$

subject to (1) and (2) is minimized. Here, $N \in \mathbb{N}$ denotes the finite control horizon, $g_n : \mathbb{R}^{n_{\underline{\boldsymbol{x}}}} \times \mathcal{A} \to \mathbb{R}$ denotes the one-step objective function as a function of a state–action pair, and $g_N : \mathbb{R}^{n_{\underline{\boldsymbol{x}}}} \to \mathbb{R}$ is the terminal objective function. Alternatively, we seek a sequence of *open-loop feedback* (OLF) policies $\underline{\mu}_{k,0:N-1}^{\mathrm{OLF}}$, with $\underline{\mu}_{k,n}^{\mathrm{OLF}}\left(\tilde{\mathcal{I}}_{k,n}\right) \in \mathcal{A}$, where $\tilde{\mathcal{I}}_{k,n} = \left\{ f_0^{\underline{\boldsymbol{x}}}, \underline{u}_{0:k+n-1} \right\}$ for $n > 0$ and $\tilde{\mathcal{I}}_{k,0} = \mathcal{I}_k$, such that the OLF objective

$$J_k^{\mathrm{OLF}} = \mathrm{E}\Bigg\{ g_N(\underline{\boldsymbol{x}}_{k,N}) + \sum_{n=0}^{N-1} g_n\left(\underline{\boldsymbol{x}}_{k,n}, \underline{\mu}_{k,n}^{\mathrm{OLF}}\left(\tilde{\mathcal{I}}_{k,n}\right)\right) \Bigg| \mathcal{I}_k \Bigg\} \tag{4}$$

subject to (1) and (2) is minimized. In MPC, also called receding horizon control, the optimal control found for the current time step $\underline{u}_{k,0}^*$ is then applied to the system, and the procedure is repeated in the next time step $k+1$. Note that controllers minimizing J_k^{CLF} are called CLF SNMPCs since they take into account that future control decisions $\underline{u}_{k,n}, n > 0$ will affect the future measurements $\underline{\boldsymbol{z}}_{k,n+1}, \ldots, \underline{\boldsymbol{z}}_{k,N}$ and therefore the information $\mathcal{I}_{k,n+1}, \ldots, \mathcal{I}_{k,N}$ available in the following time steps [10]. In contrast, OLF neglects the dependency of future measurements on the control decisions and only considers the measurement information available until time k [10], [11]. Thus, it is sufficient to predict the future system states using the system equation (1), effectively treating the POMDP as an MDP within the control horizon. While OLF achieves a significant simplification to the control problem, the imposed assumption leads to conservative approximations since the uncertainties in a pure prediction approach are commonly higher than in a combined prediction–estimation setting [12].

B. Sorting with Recirculation

We now describe some general properties and boundary conditions that hold for our sorting problem with recirculation. The inputs $\underline{u}_k$ control the proportion of the sorted mass flows of each fraction to be returned. Thus, $\mathcal{A} = [0,1]^2$ is two-dimensional and bounded. We want the controller to either a) increase the system's TNR and/or TPR in a desired way (the definition of what is desired is problem-specific) or b) exceed a desired minimum TNR or TPR in addition to a). Note that the TNR and TPR here refer to the accuracies measured at the sorting system's outlet, not at the sorter's outlet. A particularity of the system is that the system noise $\underline{\boldsymbol{w}}_k$ is composed of two

classes $\underline{d}_k$ and $\underline{\boldsymbol{w}}_k^{\boldsymbol{x}}$: systematic disturbances that are imposed by the system's environment $\underline{d}_k$, and noise effects that occur due to the randomness in the system $\underline{\boldsymbol{w}}_k^{\boldsymbol{x}}$. Whereas the latter is assumed to follow a discrete-time white noise process, the former models the expected mass flow of the input particle stream to the sorting system and is considered a second, non-controllable but known input. If $\underline{d}_k$ is not known, we may assume that we can measure $\underline{d}_k$ with a negative time delay, i.e., that we can measure the future input particle stream to a certain extent. This assumption may easily be satisfied by measurement devices placed at some distance in front of the sorting system's inlet. Note that it is the controller's goal to optimize the cost function given a particular sequence $\underline{d}_{k,0:M}$. Therefore, using a finite-horizon cost function with a control horizon similar to the sequence length M is more natural than a formulation of the SOC problem as an infinite-horizon $N \to \infty$ problem, which would require additional assumptions on $\underline{d}_{k,0:N}$. The measurements consist of noisy observations of the sorted mass flows after the separation unit in the form of camera images. The measurement noise $\underline{\boldsymbol{v}}_k$ is assumed to follow a white noise process that is mutually independent and independent of the system noise. The problem then decomposes into modeling the functions $\underline{a}_k$, $\underline{h}_k$, g_n, g_N, and all quantities involved as well as solving the resulting finite-horizon decision problem.

III. Challenges

A. Challenges in Modeling

A suitable model should allow for sufficiently fast computations to frequently process new measurements (usually arriving at a rate of a few ms) and to determine new control sequences within a time range of at most a few seconds. On the other hand, the model should be sufficiently accurate to capture all prevailing effects. Since we are faced with various complex technical components, our solution needs to handle the nonlinearities, time delays, and uncertainties arising from model abstraction and additional random perturbations. For example, whether a particle is ejected or not depends on factors including particle class, proximity to neighboring particles, and the quality of the data processing. Likewise, the recycled fraction will not have exactly the expected composition as not all intended particles may be returned, and some particles may be returned faster than others. In addition, the model's functional form should preferably be general enough to apply to a wide class of sorting systems, covering different choices for a transport unit, the optical sensor, or the recirculation.

B. Challenges in Stochastic Optimal Control

The standard approach for solving optimization problems with cumulative cost functions, such as J_k^{CLF} and J_k^{OLF}, is *dynamic programming* (DP) (see [11]). DP solves the optimization problem recursively by applying *Bellman's principle of optimality*. This yields nested optimization problems and can be solved backward in time. The DP algorithm for the aforementioned CLF optimization problem is

$$\begin{aligned} J_{k,n}(\mathcal{I}_{k,n}) = \min_{\underline{u}_{k,n}} \Bigg[\mathop{\mathrm{E}}_{\underline{\boldsymbol{x}}_{k,n}, \underline{\boldsymbol{w}}_{k,n}, \underline{\boldsymbol{z}}_{k,n+1}} \Big\{ g_n\big(\underline{\boldsymbol{x}}_{k,n}, \underline{u}_{k,n}\big) \\ + J_{k,n+1}(\mathcal{I}_{k,n+1}) \Big| \mathcal{I}_{k,n}, \underline{u}_{k,n} \Big\} \Bigg], \quad 0 \leq n < N-1 \ , \end{aligned} \tag{5}$$

$$\begin{aligned} J_{k,N-1}(\mathcal{I}_{k,N-1}) = \min_{\underline{u}_{k,N-1}} \Bigg[\mathop{\mathrm{E}}_{\underline{\boldsymbol{x}}_{k,N-1}, \underline{\boldsymbol{w}}_{k,N-1}} \Big\{ g_N\big(\underline{\boldsymbol{x}}_{k,N}\big) \\ + g_{N-1}\big(\underline{\boldsymbol{x}}_{k,N-1}, \underline{u}_{k,N-1}\big) \Big| \mathcal{I}_{k,N-1}, \underline{u}_{k,N-1} \Big\} \Bigg] \ , \end{aligned} \tag{6}$$

where $J_{k,n}(\mathcal{I}_{k,n})$ is called the *cost-to-go* or *value* of $\mathcal{I}_{k,n}$ and $J_{k,n}$ is referred to as the *value function* at stage n. If the OLF formulation is used, $\mathcal{I}_{k,n}$ is replaced with $\tilde{\mathcal{I}}_{k,n}$ and the expectation w.r.t. $\underline{\boldsymbol{z}}_{k,n+1}$ in (5) vanishes accordingly. However, the DP algorithm described above introduces two major challenges: the challenge of *state estimation* and the *support for continuous action spaces*.

Since the dimension of $\mathcal{I}_k$ increases with k, one is usually searching for a sufficient statistic with a constant dimension. Since (1) and (2) describe a Markov system, it can be shown that the conditional state distribution $f_k^{\underline{\boldsymbol{x}}_k|\mathcal{I}_k}(\underline{x}_k^{\mathrm{e}})$ of the estimated state $\underline{\boldsymbol{x}}_k^{\mathrm{e}} = \underline{\boldsymbol{x}}_k|\mathcal{I}_k$ is a sufficient statistic for $\mathcal{I}_k$ [11]. We thus require an estimator for $\underline{\boldsymbol{x}}_k^{\mathrm{e}}$, which is typically realized as a recursive Bayesian state estimator. (Note that OLF control also requires a state estimator at stage $n = 0$.) This is a particular challenge for nonlinear systems or measurement equations since closed-form formulations are not available and one has to resort to approximations.

Solving the DP equations requires considering the objective function values $J_{k,n}$ for all possible state–action pairs. Unfortunately, this is only computationally feasible in a few special cases, such as for control of systems with a finite set of states that need to be considered within the control horizon. The latter requires that the space of possible inputs $\mathcal{A}$ and—for CLF problems, also the space of measurements—is finite [13]. For continuous state and action spaces, the DP algorithm involves evaluating an infinite set of states during optimization. However, except for some rare special cases for which analytic solutions are known, such as linear-quadratic Gaussian (LQG) control, the optimal solution is computationally intractable [11]. Thus, we require a suitable approximation method to address our continuous state and action space problem.

IV. Related Work

A. State of the Art in Stochastic Model Predictive Control

Here, we recap relevant literature in the field of SNMPC with a focus on finite-horizon, continuous state and action space SOC problems as encountered in our problem setting. Roughly, the methods of this class can be divided into *global* and *local*. Global approaches typically try to reconstruct the value functions along a subset or the entire continuous state space, either by analytical methods or by fitting regression or interpolation architectures to state–value pairs provided by some method (subsumed under the term *approximate dynamic programming* (ADP)). Local *trajectory optimization* (TO)

approaches on the contrary perform an optimization along a single state–action reference trajectory starting at the current information set $\mathcal{I}_k$.

1) Global Approaches: For finite horizon problems with continuous actions, a natural approach is to discretize the action space and consider only a finite set of actions $\mathcal{A}'$. These actions are then used to build a tree structure that is evaluated backward in time using standard DP. The discretization approach is guaranteed to converge to the true DP solution under weak conditions if the discretization grid becomes finer and finer [14]. It was pursued by [15] in the context of OLF control and later extended to CLF policies and continuous measurement spaces by [12]. To incorporate measurement information into the tree, the latter considers a finite set of future *virtual measurements*, drawn from $f\Big(\underline{\boldsymbol{y}}_{k+1}|\mathcal{I}_k, \underline{u}_k\Big)$. This can be viewed as Monte Carlo integration and renders the problem feasible at the cost of approximating exact CLF control. Since the computational complexity depends strongly on the number of virtual measurements, only a small number or a single sample, e.g., the expected measurement $\mathrm{E}\{\underline{h}_k(\underline{\boldsymbol{x}}_k, \underline{\boldsymbol{v}}_k)\}$, is used. An adaptive discretization scheme is proposed by a special variant of the Perseus algorithm [16], a *point-based value iteration* (PBVI) [17] method originally developed for continuous action, finite state space POMDPs in the infinite horizon setting. The set $\mathcal{A}' = \left\{\mathcal{A}^{\mathcal{U}}, \mathcal{A}_b^{\mathcal{N}}, \mathcal{A}_b^{\mathrm{old}}\right\}$ is rebuild at each time step and composed of subsets of randomly chosen ($\mathcal{A}^{\mathcal{U}}$), previously applied ($\mathcal{A}_b^{\mathrm{old}}$), and augmented versions of the best previously applied actions ($\mathcal{A}_b^{\mathcal{N}}$). More precisely, $\mathcal{A}^{\mathcal{U}}$ is a finite set of uniform samples from $\mathcal{A}$, $\mathcal{A}_b^{\mathrm{old}}$ is composed of the previously applied actions with the smallest values, and $\mathcal{A}_b^{\mathcal{N}}$ contains samples from a normal distribution with covariance $\mathbf{C}^{\mathcal{A}}$ that is centered at the current best-known action.

There are only a few finite-horizon ADP approaches, presumably because one is generally forced to find approximations of the value function for all time steps $n = 1, \ldots, N-1$, known as *sequential backward approximation* [18]. This is in contrast to infinite horizon problems, where one typically can work with stationary value functions. Formally, one can prove that sequential backward approximation closely approximates the optimal cost-to-go and policy if one is able to fit an architecture in each stage with sufficient accuracy [18]. For example, [19] used sequential backward approximation based on Gaussian processes defined over probability distributions for a continuous state and observation POMDP with a finite action space.

2) Local Approaches: Two popular classes of TO approaches based on DP are differential dynamic programming (DDP) [20] and iterative LQG (iLQG) [21]. Both start with a given nominal state–action trajectory and then expand the model and the cost function in a Taylor series at the reference trajectory. While DDP uses a second-order Taylor series, iLQG linearizes the model and quadratizes the cost function. The DP algorithm is then solved backward on the simplified system, the calculated inputs are used to generate a new state–action reference trajectory, and the process is iteratively repeated until convergence. Notably, DDP exhibits second-order convergence to a local optimum [22], whereas iLQG has the convergence properties of the Levenberg–Marquardt algorithm [21]. Both DDP and iLQG have been successfully applied to OLF and CLF control, see, e.g., [22], [23].

A TO method avoiding iterative linearization of the system involves homotopy continuation, as proposed by [24]. The method transfers a linearization of the system back to the original system by a continuous transformation, represented mathematically as a homotopy. The homotopy allows following the LQG solution for the simplified system to a solution of the original system by tracking the minimum as the system model is gradually transferred back to its original nonlinear version.

Another class of TO algorithms directly aims to minimize the original cumulative cost function J_k^{CLF} or J_k^{OLF} considering the system and measurement equation as constraints, using methods for constrained nonlinear optimization, such as sequential quadratic programming (SQP), the augmented Lagrangian method, or interior-point methods (see, e.g., [25]–[27]). The SQP approach exhibits the characteristics of a Newton-type method, and converges to a local solution of the SOC problems, as demonstrated in [25].

Recently, another class of TO algorithm for MDP control, subsumed under the term *approximate input inference* has gained renewed interest (its origins date back to the time of Kalman, see [28]). They view the finite-horizon SOC problem as the task of inferring inputs for a latent state sequence of length N, given the desired outcomes. A state-of-the-art representative is the input inference for control (I2C) algorithm [29], which employs expectation maximization (EM) to find the unknown controls. Therefore, if designed appropriately, convergence to a zero-gradient point of the underlying likelihood function is guaranteed, with the rate of convergence being determined by the EM algorithm employed. I2C was extended in [30] to learn the parameters of a parameterized control law given as a linear combination of state-dependent basis functions.

For the special class of MDPs where the control inputs act in the same domain as the system noise, such as control-affine systems, and particular choices of the step costs (e.g., zero or quadratic control cost), a path integral formulation can be found for the cost-to-go. Approaches belonging to this class are referred to as *path integral* (PI) control [31]. In PI control, the expectation over the state space is replaced by an expectation over future trajectories of the uncontrolled system and evaluated by sampling. A version for continuous state–action nonlinear stochastic MDP was developed by [32] under the name information theoretic MPC. Recent applications of this version include, e.g., [33].

B. State of the Art in Optical Sorting

A current major trend in optical sorting is to allow for sorting increasingly smaller particles [34]. Today, particles of sizes around $100\,\mathrm{nm}$ can be detected and sorted, e.g., in gold [35] and aerosol sorting [36]. As data acquisition and processing capacities have vastly increased, recent developments move toward data analysis [37] and machine learning for system improvement, in particular for particle detection [38]. Improvements in sorting accuracies based on precise forecasting of particle motion and subsequent nozzle activation can be achieved by tracking the particles on the transport unit using a

multitarget tracking approach, as proposed by [8], [39]. Here, the gained motion information is used to precisely predict the particles' arrival time and location at the separation unit. This approach can be extended to incorporate particle-specific nozzle activations that account for the uncertainties in the prediction of the particles' arrival time, as recently suggested by [40]. In [41], we proposed to combine learned neural network models with the previously used first-principle models to describe the particle motion using a mixture-of-experts approach and showed that thereby, very precise as well as robust prediction can be achieved. Recently, we proposed an image forecasting approach using a convolutional long short-term memory-based neural network and applied it to the task of nozzle activation prediction [42]. As this approach can be trained in an unsupervised fashion, it allows sorting with a minimum of manual setup and supervision. An extensive numerical and experimental investigation of the application of neural networks in a sorting task can be found in [43]. On the experimental side, e.g., [2], [7] investigated the influence of mass flow or occupation density and mixing ratio on the sorting accuracies.

C. DEM–CFD Models for Optical Sorting

DEM–CFD was used to simulate optical sorters in [44]–[47] and showed good alignment with real experiments (see, e.g., [45], [47]). The goal of DEM–CFD simulations is to improve the design of optical sorters by numerical simulations as well as to quickly test algorithms for optical sorting (see, e.g., [46] for a numerical optimization method for the sorter design and the nozzle activation pattern). DEM–CFD resolves contacts of particles with other particles, other components of the sorter, and the fluid field surrounding the particles during ejection in detail and high temporal resolution. The DEM is able to handle systems with large numbers of particles and is thus especially suitable for modeling optical sorters. The discrete element method relies on the fundamental laws of classical mechanics, i.e., Newton's law of motion and Euler's equation to determine the motion of particles, the applied forces, and torques. Concurrently, in computational fluid dynamics, the Navier–Stokes equations are solved numerically to account for fluid forces imposed by relative velocities between the surrounding air and the particles. Here, drag models are commonly used to estimate the fluid forces on the particles.

D. Optical Sorters with Closed-Loop Control of Material Recirculation

In [9], we proposed a closed-loop control approach for TNR and TPR by adding a material recirculation to the optical sorter that allows recirculating specified fractions of the sorted mass flows to the inlet of the sorter. Moreover, we employed a deterministic MPC for closed-loop control. In the following paragraphs, we briefly describe the approach used in [9]. All considered quantities are at the level of mass flows rather than at the level of individual particles (unlike DEM–CFD, for example).

1) Modeling: Let us denote the input mass flow at the discrete time step k by $\underline{d}_k \in \mathbb{R}^2_{\geq 0}$ and the mass flow on the transport unit (in the center of the camera field of view) by $\underline{q}_k \in \mathbb{R}^2_{\geq 0}$ (see Fig. 3). Furthermore, let $\underline{y}_k, \underline{t}_k, \underline{s}_k \in \mathbb{R}^4_{\geq 0}$ be the mass flow after the separation unit, the returned partial mass flow to the inlet of the sorter, and the output mass flow of the sorting system with recirculation, respectively. The vectors describing mass flows before the sorting, $\underline{d}_k, \underline{q}_k$, consist of the mass flow of the accept and the reject particles (indicated by upper indices P and N), i.e., $\underline{q}_k = \begin{bmatrix} q_k^{\mathrm{P}} & q_k^{\mathrm{N}} \end{bmatrix}^\top$ and $\underline{d}_k = \begin{bmatrix} d_k^{\mathrm{P}} & d_k^{\mathrm{N}} \end{bmatrix}^\top$. The quantities concerning mass flows after the sorting $\underline{y}_k, \underline{t}_k, \underline{s}_k$, on the other hand, can be subdivided into the mass flows of TPs, FPs, FNs, and TNs, i.e., $\underline{y}_k = \begin{bmatrix} y_k^{\mathrm{TP}} & y_k^{\mathrm{FP}} & y_k^{\mathrm{FN}} & y_k^{\mathrm{TN}} \end{bmatrix}^\top$ and analogously for $\underline{t}_k, \underline{s}_k$. The control inputs are given by the vector $\underline{u}_k = \begin{bmatrix} u_k^{\mathrm{P}} & u_k^{\mathrm{N}} \end{bmatrix}^\top \in [0,1]^2$, where u_k^{P} denotes the fraction of particles to be returned of the accept mass flow and u_k^{N} the fraction of particles to be returned of the reject mass flow. We define $\underline{q}_k$ as the state of the system. We derive the system model by the principle of conservation of mass and by modeling the separation unit as a nonlinear function.

With the previous definitions, the mass flow on the belt is

$$\begin{aligned} \underline{q}_{k+k_{\mathrm{A}}} &= \mathbf{N}\underline{t}_k + \underline{d}_{k+k_{\mathrm{TR}}} \ , \\ \mathbf{N} &= \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \ , \end{aligned} \tag{7}$$

with the time delay $k_{\mathrm{A}} = k_{\mathrm{TR}} + k_{\mathrm{RQ}}$, k_{TR} being the estimated travel time from the recirculation unit to the mixing point with the input mass flow $\underline{d}_k$ (which is assumed to be known exactly), and k_{RQ} being the estimated travel time from this point to the center of the camera field of view.

Under the assumption that y^{TP} and y^{FP} are equally affected by u_k^{P} and, analogously, that u_k^{N} affects y^{FN} and y^{TN} equally, the returned partial mass flow $\underline{t}_k$ is given by

$$\begin{aligned} \underline{t}_k &= \mathbf{U}(\underline{u}_k)\underline{y}_{k-k_{\mathrm{YT}}} \ , \\ \mathbf{U}(\underline{u}_k) &= \mathrm{diag}\big\{\mathbf{M}^\top \underline{u}_k\big\} \ , \quad \mathbf{M} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \ , \end{aligned} \tag{8}$$

with k_{YT} describing the estimated time it takes for a particle to move from the separation unit to the recirculation unit.

For modeling the separation, we assume that the separation unit itself is a static system, i.e., its behavior only depends on the current and not on previous $\underline{q}$. The mass flow after the separation unit is then modeled by a static function $\mathbb{R}^2_{\geq 0} \to \mathbb{R}^4_{\geq 0}$ mapping $\underline{q}$ to $\underline{y}$ according to

$$\begin{aligned} \underline{y}_{k+k_{\mathrm{QY}}} &= \mathbf{\Gamma}(\underline{q}_k)\underline{q}_k \ , \\ \mathbf{\Gamma}(\underline{q}_k) &= \begin{bmatrix} \epsilon(\underline{q}_k) & 0 \\ 0 & 1-\zeta(\underline{q}_k) \\ 1-\epsilon(\underline{q}_k) & 0 \\ 0 & \zeta(\underline{q}_k) \end{bmatrix} \ , \end{aligned} \tag{9}$$

with k_{QY} being the estimated travel time from the center of the camera field of view to the separation unit. Here, the potentially nonlinear functions $\epsilon\colon \mathbb{R}^2_{\geq 0} \to (0,1)$ and $\zeta\colon \mathbb{R}^2_{\geq 0} \to (0,1)$ can be understood as the TPR and the TNR of the separation unit,

respectively. In [9], we modeled $\epsilon(\underline{q}_k)$ and $\zeta(\underline{q}_k)$ by quadratic polynomials

$$\begin{aligned}\epsilon(\underline{q}_k) &= \alpha_{00} + \alpha_{10} q_k^{\mathrm{P}} + \alpha_{01} q_k^{\mathrm{N}} + \alpha_{20} (q_k^{\mathrm{P}})^2 \\ &\quad + \alpha_{11} q_k^{\mathrm{P}} q_k^{\mathrm{N}} + \alpha_{02} (q_k^{\mathrm{N}})^2 \ , \\ \zeta(\underline{q}_k) &= \beta_{00} + \beta_{10} q_k^{\mathrm{P}} + \beta_{01} q_k^{\mathrm{N}} + \beta_{20} (q_k^{\mathrm{P}})^2 \\ &\quad + \beta_{11} q_k^{\mathrm{P}} q_k^{\mathrm{N}} + \beta_{02} (q_k^{\mathrm{N}})^2 \ ,\end{aligned} \tag{10}$$

whose parameters were identified with the help of preceding DEM–CFD simulations for the material under consideration.

Finally, the output mass flow of the sorting system with recirculation is given by

$$\underline{s}_k = \underline{y}_{k-k_{\mathrm{YT}}} - \underline{t}_k \ . \tag{11}$$

In summary, the previous steps yield the system and output equation

$$\underline{q}_{k+k_{\mathrm{A}}} = \mathbf{N}\mathbf{U}(\underline{u}_k)\mathbf{\Gamma}\Big(\underline{q}_{k-k_{\mathrm{B}}}\Big)\underline{q}_{k-k_{\mathrm{B}}} + \underline{d}_{k+k_{\mathrm{TR}}} \ , \tag{12}$$

$$\underline{s}_k = (\mathbf{I} - \mathbf{U}(\underline{u}_k))\,\mathbf{\Gamma}\Big(\underline{q}_{k-k_{\mathrm{B}}}\Big)\underline{q}_{k-k_{\mathrm{B}}} \ , \tag{13}$$

where $k_{\mathrm{B}} = k_{\mathrm{QS}} + k_{\mathrm{SY}} + k_{\mathrm{YT}}$. In addition to the measurements of the sorted mass flows after the separation unit $\hat{\underline{y}}_k$, the controller uses measurements of the mass flow on the transport unit (these are captured by the optical sorter), resulting in the measurement equations

$$\underline{z}_k^1 = \underline{q}_k \ , \tag{14}$$

$$\underline{z}_k^2 = \mathbf{\Gamma}\Big(\underline{q}_{k-k_{\mathrm{QY}}}\Big)\underline{q}_{k-k_{\mathrm{QY}}} \ . \tag{15}$$

Note that the system, output, and measurement equations are not in standard form (cf. (1) and (2)). To obtain it, one can use state augmentation, as described in [11].

2) System Analysis and Control: In [9], we analyzed the steady-state behavior of the system. We showed analytically (under some simplifying assumptions) and numerically that using material recirculation, it is almost always possible to improve either TNR or TPR of the system compared with a sorter without recirculation. However, increasing the TNR generally comes with the cost of decreasing the TPR and vice versa. Still, as increase and decrease have different magnitudes, it is possible to achieve a better result in total as measured by, e.g., a weighted sum of TNR and TPR.

In [9], a deterministic MPC based on the derived model is applied to the problem of closed-loop control and tested with the help of a DEM–CFD model of the sorting system with recirculation. Here, the goal of the MPC was to improve a combined objective function $J_k = c^{(\underline{s})} J_k^{(\underline{s})} + c^{(\underline{w})} J_k^{(\underline{q})}$ with weights $c^{(\underline{s})}, c^{(\underline{q})}$ that consists of a part $J_k^{(\underline{s})}$ being a weighted sum of TNR and TPR in the system's output mass flow $\underline{s}$ and a part $J_k^{(\underline{q})}$ being a weighted sum of TNR and TPR in the mass flow after the separation unit $\underline{q}$. The results show improvements in the desired quantity (either TNR or TPR) of approximately 1 percent point (p.p.) compared with a sorter without recirculation. This can be considered a significant improvement, as the sorting results of the sorter without recirculation were already exceeding $95\,\%$.

V. Stochastic Modeling and Identification

Due to the stochastic nature of optical sorters and particle motion as well as simplifications in modeling, the deterministic model described in Sec. IV-D deviates from the behavior of a real sorter for various reasons. For example, local clusters of particles lead to different sorting results for the same mass flows, and fluctuations in the sorted mass flows result in different recirculated fractions than intended. Furthermore, discretization, model identification, and measurement errors occur. To account for these deviations, we build a stochastic system and measurement model for our sorting system with recirculation, which we describe in the following paragraphs.

A. Key Ideas

For the probabilistic modeling of the sorting system, we extend our previous model from [9], which has been shown to strike an appropriate balance between computational requirements and predictive capabilities. While there are much more sophisticated modeling principles, such as the ones used in DEM–CFD, a lightweight model can run at a higher frequency and process measurement data more often, which in turn may lead to better control results. Specifically, we model the quantities $\underline{\boldsymbol{q}}, \underline{\boldsymbol{y}}, \underline{\boldsymbol{t}}, \underline{\boldsymbol{s}}$, and the measurements $\underline{\boldsymbol{z}}$ as normally distributed random variables and assume that the noise effects are additive, zero-mean, and independently and identically distributed across all time steps. In addition, we assume that the mass flows of accepted and rejected particles are uncorrelated.

To identify the model parameters, we again conduct several experiments using a DEM–CFD model of the sorter without recirculation, providing us with mass flow-sorting accuracy pairs. We then apply constrained least-squares regression to fit our model to the collected data.

B. Modeling

We begin with the input mass flow, which is now given by $\underline{\boldsymbol{r}}_k = \underline{d}_k + \underline{\boldsymbol{w}}_k^{\boldsymbol{r}}$ with the noise term $\underline{\boldsymbol{w}}_k^{\boldsymbol{r}} \in \mathbb{R}^2$ taking into account fluctuations in the input mass flow. The formula for the sorted mass flow $\underline{\boldsymbol{y}}$ changes from (9) to

$$\underline{\boldsymbol{y}}_{k+k_{\mathrm{QY}}} = \mathbf{\Gamma}(\underline{\boldsymbol{q}}_k)\underline{\boldsymbol{q}}_k + \underline{\boldsymbol{w}}_k^{\boldsymbol{y}} \ , \tag{16}$$

incorporating the noise term $\underline{\boldsymbol{w}}_k^{\boldsymbol{y}} \in \mathbb{R}^4$ that accounts for the uncertainties in the separation unit. To model the uncertainty in the recirculated mass flow, the additive noise term $\underline{\boldsymbol{w}}_k^{\boldsymbol{t}} \in \mathbb{R}^2$ is added to (7). Hence, the new mass flow on the belt is

$$\begin{aligned}\underline{\boldsymbol{q}}_{k+k_{\mathrm{A}}} &= \mathbf{N}\underline{\boldsymbol{t}}_k + \underline{\boldsymbol{w}}_k^{\boldsymbol{t}} + \underline{\boldsymbol{r}}_{k+k_{\mathrm{TR}}} \ , \\ \underline{\boldsymbol{w}}_k^{\boldsymbol{t}} &\sim \mathcal{N}\Big(\underline{w}_k^{\boldsymbol{t}}; \underline{0}, \mathbf{C}_{\underline{w}}^{\boldsymbol{t}}(\underline{u}_k)\Big) \ ,\end{aligned} \tag{17}$$

where we additionally assume that the covariance matrix

$$\mathbf{C}_{\underline{w}}^{\boldsymbol{t}}(\underline{u}_k) = \begin{bmatrix} \sigma_{t_{\mathrm{P}}}^2(u_k^{\mathrm{P}}) & 0 \\ 0 & \sigma_{t_{\mathrm{N}}}^2(u_k^{\mathrm{N}}) \end{bmatrix} \tag{18}$$

of the returned mass flow depends on the applied actions $\underline{u}_k$. We approximate $\sigma_{t_{\mathrm{P}}}^2(u_k^{\mathrm{P}})$, $\sigma_{t_{\mathrm{N}}}^2(u_k^{\mathrm{N}})$ using cubic functions of $\underline{u}_k$ (which are known with certainty). This way, it can be taken into account that, for example, if $\underline{u}_k = \underline{0}$, i.e., when no material

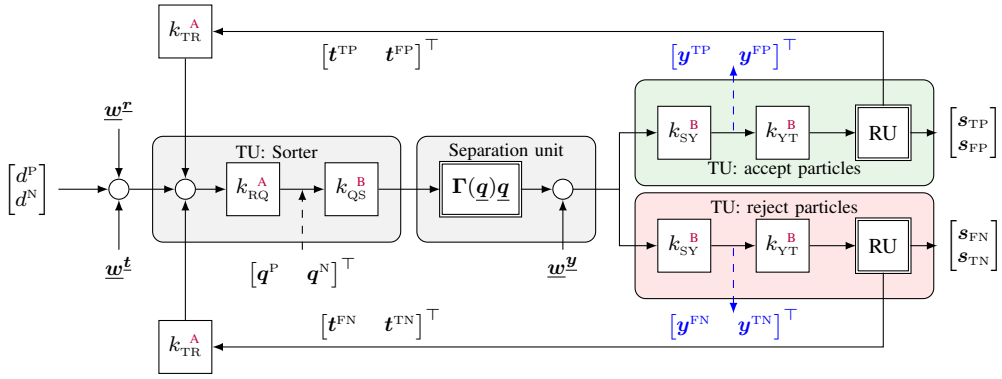


Fig. 3. General structure of our probabilistic model for a sorting system with recirculation subject to system noise. Discrete-time delays are indicated by $k_a, a \in \{\mathrm{RQ}, \mathrm{QS}, \mathrm{SY}, \mathrm{YT}, \mathrm{TR}\}$. Time delays with an upper index A or B are part of the cumulative delays k_A or k_B. RU stands for the nonlinear recirculation unit described by (8). TU stands for the transport unit. The measurement vectors are highlighted in blue. Time indices are omitted for simplicity.

TABLE I
PARAMETERS OF ϵ AND ζ FROM (10) IDENTIFIED BY THE CONSTRAINED LEAST SQUARES ACTIVE SET METHOD

Parameter	$(\cdot)_{00}$	$(\cdot)_{10}$	$(\cdot)_{01}$	$(\cdot)_{20}$	$(\cdot)_{11}$	$(\cdot)_{02}$
α	$9.9245\cdot10^{-1}$	$-1.3064\cdot10^{-4}\,\mathrm{s\,g^{-1}}$	$4.2201\cdot10^{-5}\,\mathrm{s\,g^{-1}}$	$2.9146\cdot10^{-7}\,\mathrm{s^2\,g^{-2}}$	$-5.1492\cdot10^{-6}\,\mathrm{s^2\,g^{-2}}$	$-5.8975\cdot10^{-8}\,\mathrm{s^2\,g^{-2}}$
β	$9.8788\cdot10^{-1}$	$8.2687\cdot10^{-6}\,\mathrm{s\,g^{-1}}$	$-1.9234\cdot10^{-5}\,\mathrm{s\,g^{-1}}$	$3.1924\cdot10^{-8}\,\mathrm{s^2\,g^{-2}}$	$-5.4149\cdot10^{-6}\,\mathrm{s^2\,g^{-2}}$	$8.6928\cdot10^{-8}\,\mathrm{s^2\,g^{-2}}$

is returned, the recirculation does not introduce additional uncertainty. In summary, the stochastic system shown in Fig. 3, taking into account noise effects, is described by

$$\begin{aligned}\underline{\boldsymbol{q}}_{k+k_\mathrm{A}} &= \mathbf{N}\left(\mathbf{U}(\underline{u}_k)\left(\boldsymbol{\Gamma}\left(\underline{\boldsymbol{q}}_{k-k_\mathrm{B}}\right)\underline{\boldsymbol{q}}_{k-k_\mathrm{B}} + \underline{\boldsymbol{w}}^{\boldsymbol{y}}_{k-k_\mathrm{B}}\right)\right) \\ &\quad + \underline{\boldsymbol{w}}^{\boldsymbol{t}}_k + \underline{\boldsymbol{r}}_{k+k_\mathrm{TR}}\ , \end{aligned} \tag{19}$$

$$\underline{\boldsymbol{s}}_k = (\mathbf{I} - \mathbf{U}(\underline{u}_k))\left(\boldsymbol{\Gamma}\left(\underline{\boldsymbol{q}}_{k-k_\mathrm{B}}\right) + \underline{\boldsymbol{w}}^{\boldsymbol{y}}_{k-k_\mathrm{B}}\right)\ . \tag{20}$$

To consider the measurement uncertainty of the detected particles, the measurement equation (15) is extended by the additive noise term $\underline{\boldsymbol{v}}_k \in \mathbb{R}^4$ to

$$\underline{\boldsymbol{z}}_k = \boldsymbol{\Gamma}\left(\underline{\boldsymbol{q}}_{k-k_\mathrm{QY}}\right)\underline{\boldsymbol{q}}_{k-k_\mathrm{QY}} + \underline{\boldsymbol{v}}_k\ . \tag{21}$$

Note that we incorporate only measurements of $\underline{\boldsymbol{y}}$, as opposed to the deterministic MPC of [9], in which measurements of both $\underline{\boldsymbol{y}}$ and $\underline{\boldsymbol{q}}$ were used.

C. Identification

The model parameters consist of the coefficients of the functions $\epsilon(\underline{\boldsymbol{q}}_k)$ and $\zeta(\underline{\boldsymbol{q}}_k)$ from (9), which we again model as quadratic polynomials, and the parameters of the covariance matrices of the noise effects. As in [9], we use DEM–CFD simulations for identification with a sampling time equal to the time difference between the control time steps, in our case $62.5\,\mathrm{ms}$.

1) Function Coefficients: To ensure that the values of the functions ϵ and ζ fall within the statistically valid range of $[0, 1]$, we use constrained least squares for parameter identification. For this, we employ the active set method using the condition that ϵ and ζ should be in $[0, 1]$ within the prevailing mass flow interval $\left[\underline{0}, \underline{q}_\mathrm{max}\right]$, $\underline{q}_\mathrm{max} = \begin{bmatrix} q^\mathrm{P}_\mathrm{max} & q^\mathrm{N}_\mathrm{max}\end{bmatrix}^\top$ (in our simulation model, $q^\mathrm{P}, q^\mathrm{N} \in [0\,\mathrm{g\,s^{-1}}, 400\,\mathrm{g\,s^{-1}}]$). The identified parameters are shown in Tab. I.

2) System and Measurement Noise: To identify the noise covariances of $\underline{\boldsymbol{w}}^{\boldsymbol{r}}_k$ and $\underline{\boldsymbol{w}}^{\boldsymbol{y}}_k$, the deviations of the particle mass flows from the DEM–CFD simulation and the deterministic model are considered. In our case, this yields the empirical covariance matrices

$$\mathbf{C}_{\underline{w}^{\boldsymbol{r}}} = \begin{bmatrix} 57.76 & 0 \\ 0 & 8.50 \end{bmatrix} \mathrm{g^2\,s^{-2}}\ , \tag{22}$$

$$\mathbf{C}_{\underline{w}^{\boldsymbol{y}}} = \begin{bmatrix} 2.20 & 0.00 & -2.21 & 0 \\ 0.00 & 0.39 & 0 & -0.39 \\ -2.21 & 0 & 2.22 & 0 \\ 0 & -0.39 & 0 & 0.40 \end{bmatrix} \mathrm{g^2\,s^{-2}}\ . \tag{23}$$

Note that according to the aforementioned assumptions, the mass flows of accepted and rejected particles are uncorrelated.

The function parameters of $\sigma^2_{t_\mathrm{P}}(u^\mathrm{P}_k)$, $\sigma^2_{t_\mathrm{N}}(u^\mathrm{N}_k)$ are obtained by the least squares method using data from DEM–CFD simulations with constant actions. This yields

$$\begin{aligned}\sigma^2_{t_\mathrm{P}}(u^\mathrm{P}_k) &= 2669\,\mathrm{g^2\,s^{-2}}\cdot(u^\mathrm{P}_k)^3 - 2250\,\mathrm{g^2\,s^{-2}}\cdot(u^\mathrm{P}_k)^2 \\ &\quad + 500.8\,\mathrm{g^2\,s^{-2}}\cdot u^\mathrm{P}_k + 0.01\,\mathrm{g^2\,s^{-2}}\ , \end{aligned} \tag{24}$$

$$\begin{aligned}\sigma^2_{t_\mathrm{N}}(u^\mathrm{N}_k) &= 858.7\,\mathrm{g^2\,s^{-2}}\cdot(u^\mathrm{N}_k)^3 - 552.9\,\mathrm{g^2\,s^{-2}}\cdot(u^\mathrm{N}_k)^2 \\ &\quad + 149.1\,\mathrm{g^2\,s^{-2}}\cdot u^\mathrm{N}_k + 0.01\,\mathrm{g^2\,s^{-2}}\ . \end{aligned} \tag{25}$$

Since all measured values are exactly available in the DEM–CFD simulation, the measurement noise $\underline{\boldsymbol{v}}_k$ is added artificially in the later evaluations.

VI. SNMPC

Based on the identified stochastic model, we now present the four proposed SNMPCs. In what follows, we first explain our key ideas before describing the prediction and filter steps, the objective function, and the four controllers in detail.

TABLE II
OVERVIEW OF OUR PROPOSED SNMPCS FOR THE SORTING SYSTEM

Approach	OLF	CLF
global	*OLF–PBVI*: DP with action space discretization	*CLF–PBVI*: DP with action space discretization
local	*OLF–TO*: TO using SQP and zero returned mass flow as reference trajectory	*CLF–TO*: TO using SQP and zero returned mass flow as reference trajectory

A. Key Ideas

For SOC, we investigate two OLF and two CLF SNMPCs, one each from the class of local or global approaches, as displayed in Tab. II. Our motivation for considering both local and global controllers is that while most state-of-the-art approaches are local, they may end up in a poor local optimum, i.e., they may not sufficiently explore the search space. On the other hand, for our low-dimensional system, also global action-discretization approaches are applicable, for which it is at least known that the deviation of computed and true values is bounded and decreases with a finer grid [14]. However, they may not be as accurate as local approaches due to their intrinsic limitation to a discretized action set. Considering both approaches thus offers a good chance to see which effect is more pronounced, therefore confirming the suitability of the individual controllers by empirical comparison.

Our global approach is based on an appropriate discretization of the input space using an adaptive method based on the one in Perseus [16]. We extend this method by incorporating a fitness proportional selection of discrete control inputs that worked well in previous time steps, thus incorporating a learning mechanism that learns from previous runs of the algorithm. Our local approach uses an SQP method similar to the one employed in [25]. For the initial solution, we choose the open-loop sequence, i.e., no mass flow is returned. This is because one of our major goals is to improve upon this solution. To deal with the nonlinearities of our system during state estimation, we assume all disturbances to be additive Gaussian and deploy the Smart Sampling Kalman Filter (S^2KF) [48], [49] to the problem of nonlinear state prediction and filtering.

B. Prediction and Filter Steps

For filtering and prediction, the system's state representation, system (19), and output equation (20) are reformulated so that they cover only the necessary quantities and time steps. This is based on the observation that starting from $\underline{\boldsymbol{q}}_{k-k_\mathrm{B}}$, the action $\underline{u}_k$ only affects $\underline{\boldsymbol{q}}$ and $\underline{\boldsymbol{s}}$ at times $k+k_\mathrm{A}$ and k, respectively, as well as their $\tilde{k} = k_\mathrm{A} + k_\mathrm{B}$ time multiples. Here, $\tilde{k}$ corresponds to the time for one circulation through the system. Thus, iterations of the prediction and filter steps can be performed w.r.t. multiples of $\tilde{k}$ instead of w.r.t. multiples of k. We therefore define the augmented state vector

$$\underline{\boldsymbol{x}}_{k,n} = \begin{bmatrix} \underline{\boldsymbol{q}}_{k,n} \\ \underline{\boldsymbol{s}}_{k,n} \end{bmatrix} = \begin{bmatrix} \underline{\boldsymbol{q}}_{k+n\cdot\tilde{k}-k_\mathrm{B}} \\ \underline{\boldsymbol{s}}_{k+(n-1)\cdot\tilde{k}} \end{bmatrix} \quad , \tag{26}$$

where the second index $n \in \mathbb{N}_0$ of the state $\underline{\boldsymbol{x}}_{k,n}$ specifies the future n multiples of $\tilde{k}$ starting from time k, i.e., $\underline{\boldsymbol{x}}_{k,n} = \underline{\boldsymbol{x}}_{k+\tilde{k}n}$. This transformation can be viewed as treating $\hat{k}$ independent states in parallel, one for each time step k in $\tilde{k}$. It eliminates evaluating intermediate time steps during the optimization within the control horizon, on which the targeted control variable $\underline{u}^*_{k,0}$ has no influence. Substituting (26) into the system and measurement equations (19), (20) and (21) leads to the augmented system, output, and measurement equations

$$\begin{aligned} \underline{\boldsymbol{q}}_{k,n+1} = \mathbf{N}\Big(\mathbf{U}(\underline{u}_{k,n})\Big(\boldsymbol{\Gamma}\Big(\underline{\boldsymbol{q}}_{k,n}\Big)\underline{\boldsymbol{q}}_{k,n} + \underline{\boldsymbol{w}}^{\boldsymbol{y}}_{k,n}\Big)\Big) \\ + \underline{\boldsymbol{w}}^{\boldsymbol{t}}_{k,n} + \underline{\boldsymbol{r}}_{k,n} \ , \end{aligned} \tag{27}$$

$$\underline{\boldsymbol{s}}_{k,n+1} = \big(\mathbf{I} - \mathbf{U}(\underline{u}_{k,n})\big)\Big(\boldsymbol{\Gamma}\Big(\underline{\boldsymbol{q}}_{k,n}\Big) + \underline{\boldsymbol{w}}^{\boldsymbol{y}}_{k,n}\Big) \ , \tag{28}$$

$$\underline{\boldsymbol{z}}_{k,n} = \boldsymbol{\Gamma}(\underline{\boldsymbol{q}}_{k,n})\underline{\boldsymbol{q}}_{k,n} + \underline{\boldsymbol{v}}_{k,n} \tag{29}$$

used in the prediction and filter steps.

While OLF SNMPCs use the measurement equation (29) only for state estimation before the optimization within the control horizon, i.e., at stage $n = 0$, the CLF SNMPCs also evaluate (29) within the optimization. Therefore, we incorporate future virtual measurements $\hat{\underline{z}}_{k,n} = \mathrm{E}\{\underline{\boldsymbol{z}}_{k,n}\}$ in the filter steps of the CLF optimization problem.

For both OLF and CLF approaches, the S^2KF [48], [49] is employed to the task of state prediction and filtering. The S^2KF, similar to the Unscented Kalman Filter (UKF) [50], uses deterministic samples to represent the uncertainty. Thus, all densities involved are represented by a set of samples and all operations are implemented using samples. Since the mass flows in the model description are restricted to non-negative real values, invalid samples can occur due to the unrestricted deterministic sampling that is used in the S^2KF. In particular, this concerns the functions ϵ, ζ, which are not defined for negative values. To avoid processing invalid samples, we shift them to the closest point (given by the Euclidean distance) on the edge of the valid range using the method of [51]. For example, if $\begin{bmatrix}-1 & -1\end{bmatrix}^\top$ is a sample of $\underline{\boldsymbol{q}}$, it is shifted to the $\begin{bmatrix}0 & 0\end{bmatrix}^\top$ in order to stay within the valid range $\mathbb{R}^2_{\geq 0}$. It should be noted that shifting invalid samples also causes a shift in the mean and higher moments. Since in preliminary results, we could not observe an accuracy gain when additionally inserting a correction step for the first two moments, we refrain from correcting the moments after shifting in favor of a lower computational complexity of the SNMPC.

C. Objective Function

The objective function to be minimized is in the form

$$J_k = \mathrm{E}\left\{ \sum_{n=1}^{N} \big(g_n(\underline{\boldsymbol{x}}_{k,n}) + p_n(\underline{\boldsymbol{x}}_{k,n})\big) \middle| \mathcal{I}_k \right\} \ , \tag{30}$$

with $p_n(\underline{\boldsymbol{x}}_{k,n})$ being an additional penalty term and zero terminal costs. The one-step objective function

$$g_n(\underline{\boldsymbol{x}}_{k,n}) = c^{\underline{q}} g^{\boldsymbol{q}}_n(\underline{\boldsymbol{q}}_{k,n}) + c^{\underline{s}} g^{\boldsymbol{s}}_n(\underline{\boldsymbol{s}}_{k,n}) \tag{31}$$

builds upon the deterministic formulation of [9]. It evaluates the sorting results after the separation unit via $g^{\boldsymbol{q}}_n(\underline{\boldsymbol{q}}_{k,n})$ and at the sorting system's output via $g^{\boldsymbol{s}}_n(\underline{\boldsymbol{s}}_{k,n})$ using a weighted sum of both with constants $c^{\underline{q}}, c^{\underline{s}} \geq 0$. Similar to the deterministic

MPC, $g_n^{\boldsymbol{s}}(\underline{\boldsymbol{s}}_{k,n})$ and $g_n^{\boldsymbol{q}}(\underline{\boldsymbol{q}}_{k,n})$ are given by the weighted sum of TNR and TPR

$$g_n^{\boldsymbol{s}}(\underline{\boldsymbol{s}}_{k,n}) = c^{\mathrm{P}} \frac{\boldsymbol{s}_{k,n}^{\mathrm{TP}}}{\boldsymbol{s}_{k,n}^{\mathrm{TP}} + \boldsymbol{s}_{k,n}^{\mathrm{FN}}} + c^{\mathrm{N}} \frac{\boldsymbol{s}_{k,n}^{\mathrm{TN}}}{\boldsymbol{s}_{k,n}^{\mathrm{TN}} + \boldsymbol{s}_{k,n}^{\mathrm{FP}}} \ , \tag{32}$$

$$g_n^{\boldsymbol{q}}(\underline{\boldsymbol{q}}_{k,n}) = c^{\epsilon}\epsilon(\underline{\boldsymbol{q}}_{k,n}) + c^{\zeta}\zeta(\underline{\boldsymbol{q}}_{k,n}) \tag{33}$$

with weights $c^{\mathrm{P}}, c^{\mathrm{N}} \leq 0$ and $c^{\epsilon}, c^{\zeta} \leq 0$. Note that $g_n^{\boldsymbol{q}}(\underline{\boldsymbol{q}}_{k,n})$ evaluates the TNR and TPR after the separation unit, thus attempting to directly improve the separation and allowing for shorter control horizons. The penalty term

$$p_n(\underline{\boldsymbol{x}}_{k,n}) = p_n^{\boldsymbol{q}}(\underline{\boldsymbol{q}}_{k,n}) + p_n^{\boldsymbol{s}}(\underline{\boldsymbol{s}}_{k,n}) \tag{34}$$

includes the range constraint on the mass flows $\underline{\boldsymbol{q}}_{k,n}$ and chance constraints for minimum accuracies. The range limitation

$$\begin{aligned} p_n^{\boldsymbol{q}}(\underline{\boldsymbol{q}}_{k,n}) = {} & c_{\mathrm{Q}}^{\mathrm{P}} \left(\boldsymbol{q}_{k,n}^{\mathrm{P}} - q_{\max}^{\mathrm{P}}\right) \mathbb{1}_{\mathbb{R}_{>0}}\left(\boldsymbol{q}_{k,n}^{\mathrm{P}} - q_{\max}^{\mathrm{P}}\right) \\ & + c_{\mathrm{Q}}^{\mathrm{N}} \left(\boldsymbol{q}_{k,n}^{\mathrm{N}} - q_{\max}^{\mathrm{N}}\right) \mathbb{1}_{\mathbb{R}_{>0}}\left(\boldsymbol{q}_{k,n}^{\mathrm{N}} - q_{\max}^{\mathrm{N}}\right) \ , \end{aligned} \tag{35}$$

with the indicator function $\mathbb{1}_{(\cdot)}(\cdot)$ and appropriately large chosen weights $c_{\mathrm{Q}}^{\mathrm{P}}, c_{\mathrm{Q}}^{\mathrm{N}} \geq 0$ is designed to limit the operating point $\underline{\boldsymbol{q}}_{k,n}$ of the separation unit to $\left[\underline{0}, \underline{q}_{\max}\right]$, i.e., to the identified domain of ϵ and ζ. In addition, $p_n^{\boldsymbol{q}}(\underline{\boldsymbol{q}}_{k,n})$ penalizes the accumulation of mass flow on the belt.

The chance constraint $p_n^{\boldsymbol{s}}(\underline{\boldsymbol{s}}_{k,n})$ allows specifying a minimum accuracy in the form of a minimum TNR and TPR, $\mathrm{TNR}_{\min}$, $\mathrm{TPR}_{\min}$, both in $[0, 1)$, which should be respected with confidence η^{TNR} and η^{TPR}, respectively. For this, we use

$$p_n^{\boldsymbol{s}}(\underline{\boldsymbol{s}}_{k,n}) = \sum_{a \in \{\mathrm{TPR}, \mathrm{TNR}\}} c^{\mathrm{a}} w(\mathrm{a}) \mathbb{1}_{\mathbb{R}_{\geq 0}}(w(\mathrm{a})) \ , \tag{36}$$

$$w(\mathrm{a}) = \eta^{\mathrm{a}} - \Pr\left(\mathrm{a}(\underline{\boldsymbol{s}}_{k,n}) \geq \mathrm{a}_{\min}\right) \ , \tag{37}$$

with weights $c^{\mathrm{a}} \geq 0$. Here, the penalty term includes the deviation of η^{a} and $\Pr\left(\mathrm{a}(\underline{\boldsymbol{s}}_{k,n}) \geq \mathrm{a}_{\min}\right)$, which allows for the calculations of gradients during the optimization. Note that it is reported that violating constraints within acceptable probabilistic limits can lead to better overall results [52], which is an additional advantage of chance constraints compared with strict optimization constraints. Again, the chance constraint and the expectations in the objective function are evaluated based on samples, using the state samples from the S^{2}KF.

D. Global Control Using Discretized Actions

For global OLF and CLF control, we use a discretization of the control input space and an approach similar to [15] and [12] for creating tree structures. The tree consists of edges, each corresponding to an action or virtual measurement, and nodes corresponding to predicted or filtered states, respectively. The root node represents the current state $\underline{\boldsymbol{x}}_{k,0}$, and the objective function (30) is then minimized recursively w.r.t. the action sequence $\underline{u}_{k,0:N-1}$ from the leaf nodes to the root node by the DP algorithm. For computing the expectations involved in DP, the samples of the S^{2}KF are used again.

Since creating tree structures requires control inputs from a discrete set $\mathcal{A}'$, we discretize $\mathcal{A} = [0, 1]^2$ using an extended version of the continuous actions mapping method of [16] (see Fig. 4 for an illustration). The extended approach is

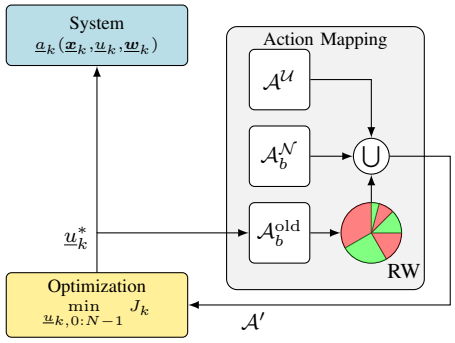


Fig. 4. Global control scheme with recursive action mapping for discretizing a continuous action space $\mathcal{A}$ into a finite set $\mathcal{A}'$. The discrete actions in $\mathcal{A}'$ are used in the optimization and the best action $\underline{u}_k^*$ is applied to the system and stored in $\mathcal{A}_b^{\mathrm{old}}$. The set $\mathcal{A}'$ consists of a subset $\mathcal{A}^{\mathcal{U}}$ containing samples uniformly drawn from $\mathcal{A}$, a subset $\mathcal{A}_b^{\mathcal{N}}$ with samples drawn from a Gaussian distribution centered at the best-known action, and a set of stored, previously applied action samples $\mathcal{A}_b^{\mathrm{old}}$. Fitness proportional selection [53] (denoted by the roulette wheel RW) is used to draw multiple action samples from $\mathcal{A}_b^{\mathrm{old}}$. For simplicity, measurements and state estimates are not shown.

based on fitness proportional selection, as applied in evolutionary algorithms [53]. Therefore, $\mathcal{A}_b^{\mathrm{old}}$ is expanded to $\{\underline{u}_m, J_m\}_{m=1}^M$ so that it contains the M best applied actions $\underline{u}_m$ of the previous optimization steps and their objective function values J_m. Actions from $\mathcal{A}_b^{\mathrm{old}}$ are assigned a probability $\Pr(u_m) = \tilde{J}_m / \sum_{i=1}^M \tilde{J}_i$ depending on their objective function value. Since J_m can also be positive in rare cases by violating constraints, $\tilde{J}_m = J_m - \max(\{J_1, \ldots, J_M\})$ guarantees that all values are less than or equal to zero and thus all $\Pr(u_m)$ correspond to valid probabilities. Based on the assigned probabilities, $M' \in [1, \ldots, M]$ action samples are drawn from $\mathcal{A}_b^{\mathrm{old}}$ and become elements of $\mathcal{A}'$. In the next iteration, the optimization result $\underline{u}_k^* \in \mathcal{A}'$ replaces the $\underline{u}_m$ with the highest objective function value in $\mathcal{A}_b^{\mathrm{old}}$ and the algorithm is repeated. Due to the similarities to the PBVI method [16] regarding the action sampling, we refer to the two global controllers as OLF–PBVI and CLF–PBVI.

E. Trajectory Optimization

For OLF–TO and CLF–TO control, we combine the method of [24] for evaluating OLF and CLF cost functions given a fixed control sequence with the SQP scheme for constrained optimization of [25]. Therefore, the first step of the cost function evaluation is to construct a state–action trajectory from a given action sequence starting from the current state $\underline{\boldsymbol{x}}_{k,0}$, by stepping forward in time. In the case of OLF control, the sequence of states is generated by repeatedly applying the prediction step of the S^{2}KF. In CLF control, additional S^{2}KF filter steps are performed each time a virtual measurement should be incorporated. In the second step, the trajectory's cumulated cost J_k is calculated by going backward in time using the samples of the S^{2}KF for evaluating all involved expectations. The optimization problem for SQP then reads

$$\min_{\underline{u}_{k,0:N-1}} J_k \ , \quad \text{s.t.} \quad \underline{\mathbf{0}} \leq \underline{u}_{k,0:N-1} \leq \underline{\mathbf{1}} \ . \tag{38}$$

Note that SQP is an iterative algorithm that calls the cost function several times during optimization, each time with a

different action sequence. As an initial reference trajectory for the optimization, we propose to use the open-loop action sequence $\underline{u}_{k,0:N-1} = \mathbf{0}$. This is because improving upon the open-loop sequence is one of our major goals, and the computational effort is reduced.

VII. Simulation

We compare the two proposed CLF and OLF SNMPCs with the deterministic MPC of [9] (referred to as NMPC) and a sorter without recirculation (referred to as open-loop) on a static and two dynamic scenarios. In the following, we describe the experimental settings and then present our results, whereby we distinguish between batch-mode and continuous-mode results.

A. DEM–CFD Model Parameters

We use the DEM–CFD model of the sorter with recirculation (see Fig. 2), the model parameters, and the particle models of [9]. Thus, the time delays k_{RQ}, k_{QS}, k_{QS}, k_{YT}, and k_{VR} correspond to $445\,\mathrm{ms}$, $125\,\mathrm{ms}$, $0\,\mathrm{s}$, $0\,\mathrm{s}$, and $3.7425\,\mathrm{s}$. As we use a sampling time of $62.5\,\mathrm{ms}$, the values of k_{A}, k_{B}, and $\tilde{k}$ equal 67, 2, and 69 time steps, respectively. As particle models, we use models of brick and sand-lime brick, where brick is considered as material to be ejected. Both particle models have diameters from $4\,\mathrm{mm}$ to $8\,\mathrm{mm}$.

B. Parameters of the Controller

For preprocessing the raw measurements of $\underline{y}$ from the DEM–CFD simulation, we apply a moving average filter covering a range of ten time steps that smooths high-frequency fluctuations (similar to [9]). While the deterministic NMPC directly uses the smoothed values, we corrupt them with artificial measurement noise for the SNMPCs. The covariance of the measurement noise $\underline{\boldsymbol{v}}_k$ is assumed to be diagonal and time-invariant, i.e., $\mathbf{C}^{\underline{\boldsymbol{v}}}_k = \mathrm{diag}\big(\begin{bmatrix}\sigma^{\mathrm{TP}} & \sigma^{\mathrm{FP}} & \sigma^{\mathrm{FN}} & \sigma^{\mathrm{TN}}\end{bmatrix}^2\big)$ with $\sigma = \sigma^{\mathrm{TP}} = \sigma^{\mathrm{FP}} = \sigma^{\mathrm{FN}} = \sigma^{\mathrm{TN}}$. Its parameter σ is selected according to the typical deviations in the position measurements that can be observed after image processing in optical sorters. Since these are usually in the range of $1\,\%$ of the image width, we choose σ so that 3σ corresponds to a deviation of $1\,\%$. We further assume that $\underline{d}$ is known to the controllers, i.e., that we can measure the input mass flow up to time step $k+L$ with the sequence length $L = (N-1)\cdot\tilde{k} + k_{\mathrm{VR}}$. The SNMPC control horizon is set to $N = 2$, and the NMPC horizon is set accordingly so that they cover the same period. The weights of (31) that balance the costs evaluated at the separation unit and the sorter's output are set to $c^{\underline{q}} = c^{\underline{s}} = 0.5$. Furthermore, we set $c^{\mathrm{P}} = c^{\epsilon}$ and $c^{\mathrm{N}} = c^{\zeta}$. Violations of the range limits from (35) are penalized by $c^{\mathrm{P}}_{\mathrm{Q}}, c^{\mathrm{N}}_{\mathrm{Q}} = -\frac{1}{100}\cdot\mathrm{E}\big\{g_n(\underline{\boldsymbol{x}}_{k,n})\big\}$. For the global methods, we use a covariance $\mathbf{C}^{\mathcal{A}} = 0.1\cdot\mathbf{I}$ to generate the normally distributed part of the action mapping. The number of mapped actions per subset of $\mathcal{A}'$ is set to 3, i.e., $|\mathcal{A}^{\mathcal{U}}| = |\mathcal{A}^{\mathcal{N}}_b| = M' = 3$. The number of applied actions stored in $\mathcal{A}^{\mathrm{old}}_b$ is $M = 10$. The initialization is done with an equidistant grid of $u^{\mathrm{P}}, u^{\mathrm{N}} \in [0,1]$. All controllers apply only the next action $\underline{u}^*_k$ to the system.

TABLE III
Scenarios considered in the simulations

Scenario	$\lvert\underline{d}(t)\rvert$ in $\mathrm{g\,s^{-1}}$	$d^{\mathrm{P}}(t)/\lvert\underline{d}(t)\rvert$	$c^{\mathrm{N}}/c^{\mathrm{P}}$	$\mathrm{TPR}_{\min}$ (η_{TPR})
S1	100	0.9	$1/10$	0
S2	$80 + 30\sin(2\pi t/10\,\mathrm{s})$	0.9	10	0
S3	$80 + 30\sin(2\pi t/10\,\mathrm{s})$	0.9	10	0.94 (0.7)
S4	80	$0.9 + 0.05\sin(2\pi t/5\,\mathrm{s})$	10	0
S5	80	$0.9 + 0.05\sin(2\pi t/5\,\mathrm{s})$	10	0.94 (0.7)

TABLE IV
Batch-mode TNR, TPR, and Rate (32) for the Scenarios S1–S5 After 60 s of Sorting (all Displayed Quantities are in %)

		Open-loop	NMPC	OLF–PBVI	OLF–TO	CLF–PBVI	CLF–TO
S1	TNR	98.38	93.35	97.28	97.68	97.82	**98.41**
	TPR	95.46	**96.33**	95.98	95.91	95.97	96.31
	Rate	95.72	96.06	96.10	96.07	96.14	**96.50**
S2	TNR	98.80	99.68	99.62	99.75	99.39	**99.88**
	TPR	**96.35**	88.65	93.03	90.03	91.43	90.36
	Rate	98.57	98.68	**99.02**	98.87	98.67	99.01
S3	TNR	98.80	99.68	98.98	99.70	99.17	**99.76**
	TPR	**96.35**	88.65	94.31	92.48	94.11	92.15
	Rate	98.57	98.68	98.55	99.04	98.71	**99.07**
S4	TNR	98.70	99.40	99.25	99.56	99.41	**99.74**
	TPR	**96.51**	90.40	92.56	91.86	92.36	91.42
	Rate	98.50	98.58	98.64	98.86	98.77	**98.99**
S5	TNR	98.70	99.40	99.23	**99.62**	99.28	99.30
	TPR	**96.51**	90.40	93.77	93.53	93.23	93.57
	Rate	98.50	98.58	98.73	**99.07**	98.73	98.78

C. Scenarios

We examine five distinct sorting scenarios, each spanning a duration of $60\,\mathrm{s}$. These scenarios are characterized based on three specific criteria: a constant mixing ratio $d^{\mathrm{P}}(t)/\lvert\underline{d}\rvert(t)$, a constant total mass flow $\lvert\underline{d}\rvert$, or a combination of both these factors (see Tab. III). In all scenarios, $c^{\mathrm{P}} = c^{\epsilon}$, $c^{\mathrm{N}} = c^{\zeta}$, and $c^{\mathrm{P}} + c^{\mathrm{N}} = -1$ apply. Scenario S1 has a constant input mass flow of $\lvert\underline{d}\rvert = 100\,\mathrm{g\,s^{-1}}$ and a constant mixing ratio of $d^{\mathrm{P}}(t)/\lvert\underline{d}(t)\rvert = 0.9$ and thus can be regarded as static. The scenarios S2–S5 consider dynamic situations. Furthermore, whereas in S1, the objective is to improve the TPR, in S2–S5, we aim to improve the TNR. Note that S1 precisely corresponds to a scenario from our previous work [9]. For both S2 and S3, we consider an input mass flow that evolves according to $80\,\mathrm{g\,s^{-1}} + 30\,\mathrm{g\,s^{-1}}\sin(2\pi t/10\,\mathrm{s})$, but a constant mixing ratio of 0.9. Thus, d^{P} is ranging from $45\,\mathrm{g\,s^{-1}}$ to $99\,\mathrm{g\,s^{-1}}$. In contrast, S4 and S5 have a constant input mass flow of $80\,\mathrm{g\,s^{-1}}$, whereas the mixing ratio varies according to $0.9 + 0.05\sin(2\pi t/5\,\mathrm{s})$. In S3 and S5, we additionally consider a minimum TPR, $\mathrm{TPR}_{\min} = 0.94$, that should not be surpassed by the TPR of the sorting system with confidence $\eta^{\mathrm{TPR}} = 0.7$.

D. Batch-Mode Evaluation

The results of the controller application to a DEM–CFD simulation model of the optical sorting system with recirculation for the scenarios S1–S5 are displayed in Tab. IV. Here, we

TABLE V
CONTINUOUS-MODE (TIME-RELATED) STATISTICS FOR THE COURSE OF MEDIAN AND INTERQUANTILE RANGE OF TNR AND TPR FOR S2 AND S3

		S2						S3			
		Open-loop	NMPC	OLF–PBVI	OLF–TO	CLF–PBVI	CLF–TO	OLF–PBVI	OLF–TO	CLF–PBVI	CLF–TO
$\mu^{\mathrm{TNR}}_{\mathrm{Median}}$	(in %)	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
$\mu^{\mathrm{TNR}}_{\mathrm{IQR}}$	(in p.p.)	7.10	**0.00**	**0.00**	**0.00**	0.16	**0.00**	4.31	**0.00**	1.63	**0.00**
$\sigma^{\mathrm{TNR}}_{\mathrm{IQR}}$	(in p.p.)	15.25	**0.00**	**0.00**	**0.00**	1.14	**0.00**	8.58	**0.00**	7.79	**0.00**
$\mu^{\mathrm{TPR}}_{\mathrm{Median}}$	(in %)	**99.55**	92.48	96.70	91.19	93.87	93.61	97.85	95.29	97.08	93.89
$\mu^{\mathrm{TPR}}_{\mathrm{IQR}}$	(in p.p.)	**14.62**	31.92	32.77	45.58	35.75	46.33	20.27	50.08	23.09	42.69
$\sigma^{\mathrm{TPR}}_{\mathrm{IQR}}$	(in p.p.)	**2.54**	6.34	19.87	23.68	15.71	4.23	3.26	21.86	6.68	19.66

TABLE VI
CONTINUOUS-MODE (TIME-RELATED) STATISTICS FOR THE COURSE OF MEDIAN AND INTERQUANTILE RANGE OF TNR AND TPR FOR S4 AND S5

		S4						S5			
		Open-loop	NMPC	OLF–PBVI	OLF–TO	CLF–PBVI	CLF–TO	OLF–PBVI	OLF–TO	CLF–PBVI	CLF–TO
$\mu^{\mathrm{TNR}}_{\mathrm{Median}}$	(in %)	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
$\mu^{\mathrm{TNR}}_{\mathrm{IQR}}$	(in p.p.)	2.90	0.36	0.03	0.04	0.33	**0.00**	0.04	8.18	3.16	0.27
$\sigma^{\mathrm{TNR}}_{\mathrm{IQR}}$	(in p.p.)	7.53	1.95	0.57	0.54	2.34	**0.00**	0.48	27.28	12.02	1.68
$\mu^{\mathrm{TPR}}_{\mathrm{Median}}$	(in %)	**100.00**	93.82	95.71	94.70	95.12	94.85	96.33	96.39	95.40	97.64
$\mu^{\mathrm{TPR}}_{\mathrm{IQR}}$	(in p.p.)	**15.57**	34.07	34.54	48.49	32.68	54.48	23.63	39.03	23.98	53.42
$\sigma^{\mathrm{TPR}}_{\mathrm{IQR}}$	(in p.p.)	**2.86**	8.22	12.54	22.01	18.11	25.86	4.15	14.39	4.61	22.18

evaluate the TNR, TPR, and the combined objective (referred to as rate) according to (32) after $60\,\mathrm{s}$ of sorting, i.e., after the sorting task was completed. This applies to the case where the optical sorter is used in batch mode, i.e., one is only interested in the purity of the accumulated sorted masses. Note that for both the open-loop system and the NMPC, the scenarios S2 and S4 are equivalent to S3 and S5, respectively, as neither allows the specification of a chance constraint.

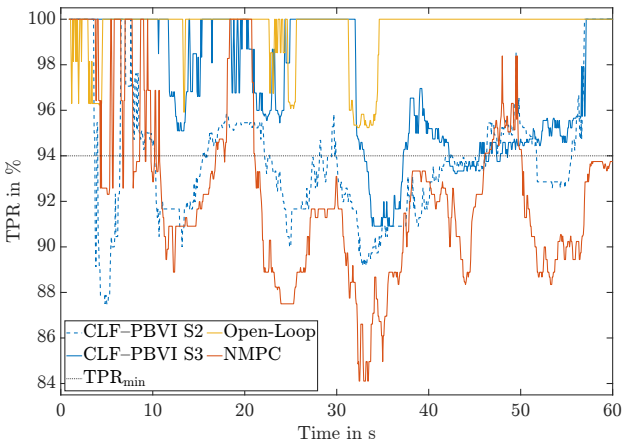
All considered controllers succeed in improving the TPR (for S1) or TNR (for S2–S5), compared with the sorter without recirculation. Thus, all controllers reach their goal to either improve TNR or TPR as determined by the chosen weights. Comparing the rate, all stochastic controllers outperform the previously known deterministic NMPC in all scenarios. For example, considering S1, the best SNMPC, the CLF–TO, achieves a TPR of $96.31\,\%$ or an improvement of approximately $0.85\,$p.p. compared with the open-loop system and a rate of $96.5\,\%$, which is $0.44\,$p.p. above the deterministic NMPC. Interestingly, the CLF–TO was able to also improve its TNR compared with the open-loop system, whereas the deterministic NMPC suffers from a decrease in TNR. Looking at S2, once again, the SNMPC outperforms the deterministic NMPC and the open-loop system. The CLF–TO controller achieves the highest TNR of $99.88\,\%$ with an improvement of $1.08\,$p.p. compared to the open-loop system.

Introducing the chance constraint for the TPR, as in S3 and S5, generally slightly decreases the achieved TNRs compared with S2 and S4. Simultaneously, the sharp decrease in TPR, as can be observed in S2 and S4, is mitigated by the additional constraint. For example, comparing S2 and S3, the TNR of CLF–TO decreases by $0.12\,$p.p., while the TPR increases by $1.79\,$p.p., thereby enhancing the rate by $0.06\,$p.p. when applying the constraint. However, in batch mode, introducing the additional chance constraint does not necessarily lead to an overall TPR higher than the desired constraint. To investigate the influence of the constraint in more detail, we examine the temporal evolution of the TNR and TPR in the following subsections.

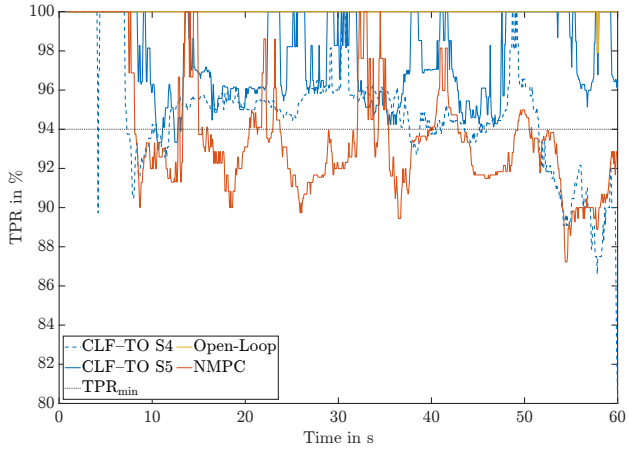
E. Continuous-Mode Evaluation

Due to the dynamic nature of S2–S5, we additionally consider the TNR and TPR of the sorted mass flows over time for evaluation rather than the accumulated sorted masses. Note that this kind of evaluation is suitable for sorters that are applied in continuous sorting tasks (and not in batch mode), where high (or stable) and less fluctuating accuracy is of interest at all times.

1) Evaluation Metrics: For S2–S5, Tab. V and Tab. VI present statistics of TNR and TPR evaluated using a sliding window of length $5\,\mathrm{s}$. To assess the average accuracy within the windows, the mean of the windows' median TNR and



(a) TPR for CLF–PBVI, NMPC, and the open loop system in S2 and S3.



(b) TPR for CLF–TO, NMPC, and the open loop system in S4 and S5.

Fig. 5. Illustration of the impact of the chance constraints. The graphs show the median within a sliding window of length 5 s, centered on the current evaluation point. S3 and S5 apply chance constraints while S2 and S4 do not, but otherwise share the same settings with S3 and S5, respectively (see Tab. III).

TPR, $\mu^{\text{TNR}}_{\text{Median}}$ and $\mu^{\text{TPR}}_{\text{Median}}$, is used. A high $\mu^{\text{TNR}}_{\text{Median}}$, $\mu^{\text{TPR}}_{\text{Median}}$ indicates a high degree of purity. To account for the magnitude of deviations within each window, we calculate the mean of the windows' interquantile ranges w.r.t. the 5 % and 95 % quantile range, $\mu^{\text{TNR}}_{\text{IQR}}$ and $\mu^{\text{TPR}}_{\text{IQR}}$ (the smaller, the better). To compare the deviations across the windows, we use the standard deviation of the interquartile ranges, $\sigma^{\text{TNR}}_{\text{Median}}$ and $\sigma^{\text{TPR}}_{\text{Median}}$ (the smaller, the better). Whereas the mean of the interquantile ranges thus represents the average overall noise level, their standard deviation is a measure of the deviations in the noise w.r.t. different points in time. For example, a high standard deviation indicates that noise levels in the windows strongly differ between different times during the evaluation scenario.

2) Evaluation of Medians: As can be seen from Tab. V and Tab. VI, as a consequence of optimizing the TNR in S2–S5, the averaged median TNR for all controllers within the sliding window remains consistently at 100 %. Compared with the open-loop system, the averaged medians of the TPR decrease for all controllers. For example, the average median TPR of the CLF–PBVI in S2 decreases by 5.68 p.p. to 93.87 % compared with the open-loop system. Note that besides the averaged median TNR, the averaged median TPR for the open-loop case in S4 is also at 100 %, which is the best result, although the cumulative open-loop rate (see Tab. IV) is lower than the resulting rates obtained using the controllers. This can be explained by the fact that the evaluation through the sliding windows does not take into account the absolute amount of sorted mass flow within each window, as both TNR and TPR are relative measures. For qualitative comparison, the medians of the TPR within each sliding window are displayed in Fig. 5a for the example of the CLF–PBVI, NMPC, and the open-loop system. Similarly, Fig. 5b shows the TPR medians for CLF–TO, NMPC, and the open-loop system in S4 and S5. The curves suggest that the deviations in TPR are much more pronounced when improving the TNR using controlled recirculation.

3) Evaluation of Interquantile Ranges: The hypothesis that the deviations in TPR are increased when the TNR is improving is confirmed by comparing the interquantile ranges of the TNR with the open-loop system in Tab. V and Tab. VI.

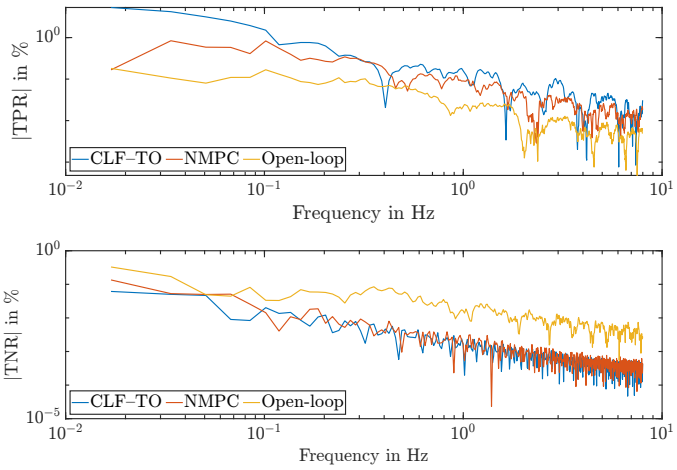


Fig. 6. Amplitude of the frequency response of CLF–TO, NMPC, and the open-loop system for the TNR and TPR in S2. The curves show that the fluctuations in the TNR decrease compared with the open-loop system, while the fluctuations in the TPR increase.

Simultaneously, regarding the TNR in S2–S4, it can be seen that all controllers reduce both the mean interquantile range and their standard deviation compared with the open-loop system. Only in S5, OLF–TO and CLF–PBVI have greater $\mu^{\text{TNR}}_{\text{IQR}}$ and $\sigma^{\text{TNR}}_{\text{IQR}}$ than the open-loop approach. In S3, all controllers except CLF–PBVI reduce $\mu^{\text{TNR}}_{\text{IQR}}$ and $\sigma^{\text{TNR}}_{\text{IQR}}$ to 0 p.p., i.e., eliminate almost all fluctuations in the TNR. This effect is more conveniently illustrated in Fig. 6, where we plotted the amplitudes of the frequency responses of the TNR and TPR for the CLF–TO, NMPC, and the open-loop system. Note that this representation does not use sliding windows and is thus independent of the choice of a window length. As can be seen, all controllers have lower fluctuations in the TNR than the open-loop system, while the fluctuations in the TPR increase.

4) Impact of Chance Constraints: Comparing S2 and S4 with S3 and S5, respectively, it can be seen that the additional chance constraint on the TNR leads to average median TPRs close to or above the desired threshold. For a qualitative comparison, see Fig. 5. Additionally, except for OLF–TO in

S3, the constraint leads to decreasing mean interquantile ranges of the TPR $\mu_{\mathrm{IQR}}^{\mathrm{TPR}}$ (cf. Tab. V and Tab. VI). Thus, incorporating the chance constraint raises the TPR to the desired level while simultaneously reducing the noise in the TPR.

VIII. Summary of Results

Our results show that the proposed controllers improve on the open-loop case and outperform the previously known deterministic MPC in both static and dynamic situations. For example, in batch-mode, they achieve TNRs up to $99.88\,\%$ or improvements in TNR up to 1.08 p.p. compared with the open-loop system and 0.2 p.p. compared with the deterministic NMPC (see S2 in Tab. IV). In other words, in this scenario, the error given by $1-\mathrm{TNR}$, which by definition is proportional to the number of FPs, was effectively reduced by a factor of 10 compared with the open loop system and by a factor of 2.7 compared with the NMPC. For dynamic situations, the controllers not only improve either the TPR or TNR but also reduce its deviation to almost zero. Again, the SNMPCs in most cases perform better than the deterministic NMPC. However, in general, improving TNR and reducing its variance comes at the cost of decreasing TPR and increasing its variance and vice versa.

Incorporating chance constraints for a minimum TPR generally slightly decreases the TNR, but increases the TPR to a substantial amount (e.g., trading a 0.12 p.p. decrease in TNR with an 1.79 p.p. increase in TPR for CLF–TO in S2 and S3). In dynamic scenarios, the chance constraint raises the TPR to the desired level in good agreement with the specified confidence level while simultaneously reducing the noise in the TPR. In general, incorporating chance constraints can therefore be considered a valuable strategy.

When comparing the scenarios, fortunately, no significant differences in behavior between static and dynamic mass flows with either fluctuating total mass flow or mixing ratio can be observed. In all cases, improvements are in a similar range, indicating that the controllers are able to cope with different kinds of input disturbances. Comparing the SNMPCs, the CLF–TO in most cases achieves the highest accuracies and low deviations. Therefore, it proves to be particularly suitable for our task.

IX. Conclusion

We proposed two CLF and two OLF SNMPCs for closed-loop control of an optical sorting system with material recirculation. The stochastic formulation of the controllers allows specifying a minimal TNR or TPR that should be maintained in any plant condition with a predefined level of confidence. We showed that the proposed controllers achieved the desired sorting accuracy in static and highly dynamic scenarios, with sorting accuracies generally higher than those achieved with a sorter without recirculation. The results thereby demonstrate that controlled recirculation can reduce the number of falsely undeflected particles by a factor of up to 10. Furthermore, fluctuations in either TNR or TPR are effectively reduced.

Sorting applications can greatly benefit from our approach. Since the sorting accuracy is in general higher than for a sorter without recirculation, our method allows for either smaller or fewer sorters (e.g., by eliminating the need for a second sorting stage) or higher mass flows to be sorted, thus enhancing the profitability of the plant. In particular, it eliminates the need for laborious and costly hardware or software adjustments, which are often required prior to a sorting task when the desired accuracy cannot be achieved ad hoc. It allows adjusting the trade-off between TNR and TPR, i.e., the purity of the two sorted fractions, at runtime, thus enabling more flexible sorting. In particular, since a desired minimum accuracy for either TNR or TPR can be specified, the focus can be moved to improving the other quantity, thereby reducing losses by minimizing FNs or FPs, respectively. Fluctuations in either TNR or TPR are effectively reduced, making the approach valuable for continuously operating plants where optical sorters are part of a process chain, and achieving a certain quality at any point in time is crucial.

References

[1] G. Maier *et al.*, "A Survey of the State of the Art in Sensor-Based Sorting Technology and Research," *IEEE Access*, vol. 12, pp. 6473–6493, 2024.

[2] E. Gülcan and Ö. Y. Gülsoy, "Performance Evaluation of Optical Sorting in Mineral Processing – A Case Study with Quartz, Magnesite, Hematite, Lignite, Copper and Gold Ores," *International Journal of Mineral Processing*, vol. 169, pp. 129–141, 2017.

[3] C. Robben and H. Wotruba, "Sensor-Based Ore Sorting Technology in Mining—Past, Present and Future," *Minerals*, vol. 9, no. 9, p. 523, 2019.

[4] R. C. Bruce *et al.*, "The Impact of Optical Berry Sorting on Red Wine Composition and Sensory Properties," *Foods*, vol. 10, no. 2, p. 402, 2021.

[5] S. P. Gundupalli *et al.*, "A Review on Automated Sorting of Source-Separated Municipal Solid Waste for Recycling," *Waste Management*, vol. 60, pp. 56–74, 2017.

[6] K. Friedrich, "Sensor-Based and Robot Sorting Processes and their Role in Achieving European Recycling Goals - A Review," *Academic Journal of Polymer Science*, vol. 5, no. 4, 2022.

[7] B. Küppers *et al.*, "Influence of Material Alterations and Machine Impairment on Throughput Related Sensor-Based Sorting Performance," *Waste Management & Research: The Journal for a Sustainable Circular Economy*, vol. 39, no. 1, pp. 122–129, 2021.

[8] G. Maier *et al.*, "Experimental Evaluation of a Novel Sensor-Based Sorting Approach Featuring Predictive Real-Time Multiobject Tracking," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 2, pp. 1548–1559, 2021.

[9] J. Vieth *et al.*, "Improving Accuracy of Optical Sorters Using Closed-Loop Control of Material Recirculation," in *2023 American Control Conference (ACC)*, 2023, pp. 3257–3263.

[10] Y. Bar-Shalom and E. Tse, "Dual Effect, Certainty Equivalence, and Separation in Stochastic Control," *IEEE Transactions on Automatic Control*, vol. 19, no. 5, pp. 494–500, 1974.

[11] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 4th ed. Belmont, Massachusetts: Athena Scientific, 2017, vol. 1.

[12] F. Weissel *et al.*, "Stochastic Model Predictive Control of Time-Variant Nonlinear Systems with Imperfect State Information," in *Proceedings of the 2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2008)*, Seoul, Republic of Korea, Aug. 2008, pp. 40–46.

[13] R. D. Smallwood and E. J. Sondik, "The Optimal Control of Partially Observable Markov Processes Over a Finite Horizon," *Operations Research*, vol. 21, no. 5, pp. 1071–1088, 1973.

[14] D. Bertsekas, "Convergence of Discretization Procedures in Dynamic Programming," *IEEE Transactions on Automatic Control*, vol. 20, no. 3, pp. 415–419, 1975.

[15] F. Weissel *et al.*, "A Nonlinear Model Predictive Control Framework Approximating Noise Corrupted Systems with Hybrid Transition Densities," in *Proceedings of the 2007 IEEE Conference on Decision and Control (CDC 2007)*, New Orleans, Louisiana, USA, Dec. 2007, pp. 3661–3666.

[16] M. Spaan and N. Vlassis, "Perseus: Randomized Point-based Value Iteration for POMDPs," *Journal of Artificial Intelligence Research (JAIR)*, vol. 24, pp. 195–220, 01 2005.

[17] J. Pineau, G. Gordon, and S. Thrun, "Point-Based Value Iteration: An Anytime Algorithm for POMDPs," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, ser. IJCAI'03, San Francisco, CA, USA, 2003, p. 1025–1030.

[18] D. Bertsekas and J. N. Tsitsiklis, *Neuro-dynamic Programming*. Athena Scientific, 1996.

[19] J. Mayer *et al.*, "Stochastic Optimal Control Using Gaussian Process Regression over Probability Distributions," in *2019 American Control Conference (ACC)*, 2019, pp. 4847–4853.

[20] D. Mayne, "A Second-order Gradient Method for Determining Optimal Trajectories of Non-linear Discrete-time Systems," *International Journal of Control*, vol. 3, no. 1, pp. 85–95, 1966.

[21] W. Li and E. Todorov, "Iterative Linearization Methods for Approximately Optimal Control and Estimation of Non-linear Stochastic System," *International Journal of Control*, vol. 80, no. 9, pp. 1439–1453, 2007.

[22] J. Van Den Berg *et al.*, "Motion Planning Under Uncertainty Using Differential Dynamic Programming in Belief Space," in *Robotics research: The 15th international symposium ISRR*. Springer, 2017, pp. 473–490.

[23] G. Chen *et al.*, "Locally Optimal Estimation and Control of Cable Driven Parallel Robots using Time Varying Linear Quadratic Gaussian Control," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 7367–7374.

[24] C. Chlebek and U. D. Hanebeck, "Approximation of Stochastic Nonlinear Closed-Loop Feedback Control with Application to Miniature Walking Robots," in *Proceedings of the 2016 European Control Conference (ECC 2016)*, Aalborg, Denmark, Jun. 2016.

[25] X. Feng *et al.*, "Inexact Adjoint-based SQP Algorithm for Real-Time Stochastic Nonlinear MPC," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 6529–6535, 2020.

[26] F. Lin, M. Fardad, and M. R. Jovanovic, "Augmented Lagrangian Approach to Design of Structured Optimal State Feedback Gains," *IEEE Transactions on Automatic Control*, vol. 56, no. 12, pp. 2923–2929, 2011.

[27] J. Zhang and T. Ohtsuka, "A Recursive Riccati Interior-point Method for Chance-constrained Stochastic Model Predictive Control," *SICE Journal of Control, Measurement, and System Integration*, vol. 16, no. 1, pp. 273–285, 2023.

[28] E. Todorov, "General Duality between Optimal Control and Estimation," in *2008 47th IEEE Conference on Decision and Control*, 2008, pp. 4286–4292.

[29] J. Watson, H. Abdulsamad, and J. Peters, "Stochastic Optimal Control as Approximate Input Inference," in *Proceedings of the Conference on Robot Learning*, 2020, pp. 697–716.

[30] S. P. Q. Syed and H. Bai, "Parameterized Input Inference for Approximate Stochastic Optimal Control," in *2023 American Control Conference (ACC)*, 2023, pp. 2574–2579.

[31] H. J. Kappen, "Linear Theory for Control of Nonlinear Stochastic Systems," *Physical Review Letters*, vol. 95, no. 20, p. 200201, 2005.

[32] G. Williams *et al.*, "Information Theoretic MPC for Model-based Reinforcement Learning," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 1714–1721.

[33] H. Homburger, S. Wirtensohn, and J. Reuter, "Swinging Up and Stabilization Control of the Furuta Pendulum using Model Predictive Path Integral Control," in *2022 30th Mediterranean Conference on Control and Automation (MED)*, 2022, pp. 7–12.

[34] S. C. Das *et al.*, "An Optical Cluster: Near-Field Optical Sorting and Separation of Plasmonic, Dielectric, and Chiral Nanoparticle," *Annalen der Physik*, vol. 533, no. 10, p. 2100299, 2021.

[35] M. Ploschner *et al.*, "Bidirectional Optical Sorting of Gold Nanoparticles," *Nano Letters*, vol. 12, no. 4, pp. 1923–1927, 2012.

[36] J. J. Curry and Z. H. Levine, "Continuous-Feed Optical Sorting of Aerosol Particles," *Optics Express*, vol. 24, no. 13, p. 14100, 2016.

[37] K. Friedrich *et al.*, "Assessment of Technological Developments in Data Analytics for Sensor-Based and Robot Sorting Plants Based on Maturity Levels to Improve Austrian Waste Sorting Plants," *Sustainability*, vol. 13, no. 16, p. 9472, 2021.

[38] Z. Luan *et al.*, "Sunflower Seed Sorting Based on Convolutional Neural Network," in *Eleventh International Conference on Graphics and Image Processing (ICGIP 2019)*, vol. 11373. SPIE, 2020, pp. 428–434.

[39] F. Pfaff, "Multitarget Tracking Using Orientation Estimation for Optical Belt Sorting," Ph.D. dissertation, Karlsruher Institut für Technologie (KIT), 2019.

[40] M. Reith-Braun *et al.*, "Approximate First-Passage Time Distributions for Gaussian Motion and Transportation Models," in *2023 26th International Conference on Information Fusion (FUSION)*, 2023, pp. 1–8.

[41] J. Thumm *et al.*, "Mixture of Experts of Neural Networks and Kalman Filters for Optical Belt Sorting," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 6, pp. 3724–3733, 2022.

[42] M. Reith-Braun *et al.*, "GridSort: Image-based Optical Bulk Material Sorting Using Convolutional LSTMs," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 4620–4626, 2023, 22nd IFAC World Congress.

[43] G. Maier *et al.*, "Simulation Study and Experimental Validation of a Neural Network-based Predictive Tracking System for Sensor-based Sorting," *tm - Technisches Messen*, vol. 90, no. 7-8, pp. 489–499, 2023.

[44] R. S. Fitzpatrick *et al.*, "CFD–DEM Modelling of Particle Ejection by a Sensor-Based Automated Sorter," *Minerals Engineering*, vol. 79, pp. 176–184, 2015.

[45] C. Pieper *et al.*, "Numerical Modelling of an Optical Belt Sorter Using a DEM–CFD Approach Coupled with Particle Tracking and Comparison with Experiments," *Powder Technology*, vol. 340, pp. 181–193, 2018.

[46] A. Bauer *et al.*, "Towards a Feed Material Adaptive Optical Belt Sorter: A Simulation Study Utilizing a DEM-CFD Approach," *Powder Technology*, vol. 411, p. 117917, 2022.

[47] ——, "Benchmarking a DEM-CFD Model of an Optical Belt Sorter by Experimental Comparison," *Chemie Ingenieur Technik*, vol. 95, no. 1, pp. 256–265, 2023.

[48] J. Steinbring and U. D. Hanebeck, "LRKF Revisited: The Smart Sampling Kalman Filter (S2KF)," *Journal of Advances in Information Fusion*, vol. 9, no. 2, pp. 106–123, Dec. 2014.

[49] J. Steinbring *et al.*, "The Smart Sampling Kalman Filter with Symmetric Samples," *Journal of Advances in Information Fusion*, vol. 11, no. 1, pp. 71–90, Jun. 2016.

[50] S. J. Julier and J. K. Uhlmann, "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[51] R. Kandepu *et al.*, "Constrained State Estimation Using the Unscented Kalman Filter," in *2008 16th Mediterranean Conference on Control and Automation*, 2008, pp. 1453–1458.

[52] M. Cannon *et al.*, "Stochastic Tube MPC with State Estimation," in *19th international symposium on mathematical theory of networks and systems*, Budapest, Ungarn, 2010, pp. 536–541.

[53] T. Blickle and L. Thiele, "A Comparison of Selection Schemes Used in Evolutionary Algorithms," *Evolutionary Computation*, vol. 4, no. 4, pp. 361–394, 1996.