Open Peer Review on Qeios

RESEARCH ARTICLE

Self-Driving Development of Perfusion Processes for Monoclonal Antibody Production

Claudio Müller[1], Thomas Vuillemin[2], Chethana Janardhana Gadiyar[2], Jean-Marc Bielser[2], Jonathan Souquet[2], Alessandro Fagnani[1], Michael Sokolov[1], Moritz von Stosch[1], Fabian Feidl[1], Alessandro Butté[1], Mariano Nicolas Cruz Bournazou[1]

1 DataHow AG, Zürich, Switzerland

2 Biotech Development Center, Ares Trading SA, Merck Group, Darmstadt, Germany

Funding: No specific funding was received for this work.

Potential competing interests: No potential competing interests to declare.

Abstract

It is essential to increase the number of autonomous agents bioprocess development for biopharma innovation to shorten time and resource utilization in the path from product to process. While robotics and machine learning have significantly accelerated drug discovery and initial screening, the later stages of development have seen improvement only in the experimental automation but lack advanced computational tools for experimental planning and execution. For instance, during development of new monoclonal antibodies, the search for optimal upstream conditions (feeding strategy, pH, temperature, media composition, etc.) is often performed in highly advanced high-throughput (HT) mini-bioreactor systems. However, the integration of machine learning tools for experiment design and operation in these systems remains underdeveloped. In this study, we introduce an integrated framework composed by a Bayesian experimental design algorithm, a cognitive digital twin of the cultivation system, and an advanced 24 parallel mini-bioreactor perfusion experimental setup. The result is an autonomous experimental machine capable of 1. embedding existing process knowledge, 2. learning during experimentation, 3. Using information from similar processes, 4. Notifying events in the near future, and 5. Autonomously operating the parallel cultivation setup to reach challenging objectives. As a proof of concept, we present experimental results of 27 days long cultivations operated by an autonomous software agent reaching challenging goals as are increasing the VCV and maximizing the viability of the cultivation up to its end.

Claudio Müller[1], **Thomas Vuillemin**[2], **Chethana Janardhana Gadiyar**[2], **Jean-Marc Bielser**[2], **Jonathan Souquet**[2], **Alessandro Fagnani**[1], **Michael Sokolov**[1], **Moritz von Stosch**[1], **Fabian Feidl**[1], **Alessandro Butté**[1], and **Mariano Nicolas Cruz Bournazou**[1,3,*]

[1] *DataHow AG, Zürich, Switzerland*

[2] *Biotech Development Center, Ares Trading SA (an affiliate of Merck KGaA, Darmstadt, Germany), Fenil- sur-Corsier, Switzerland*

[3] *Technische Universität Berlin, Institute of Biotechnology, Chair of Bioprocess Engineering, Berlin, Germany*

*Correspondence: mariano.n.cruzbournazou@tu-berlin.de

1. Introduction

Competition in the biopharmaceutical industry has driven many advances in process development and clinical manufacturing for recombinant proteins. Today, R&D teams are challenged in early development phases to deliver a product within very short timelines. To achieve this, High Throughput (HT) devices are key to accelerate the development process of new molecules in the biopharmaceutical industry pipeline. Robotic platforms are used to increase experimental throughput and deliver the best production strategy through Quality by Design (QbD) approaches[1][2]. Robocolumns are for example used for downstream process (DSP) condition screening[3]. Microwell plates are used to cultivate cells during cell line development and upstream process (USP) development (Rouiller et al., 2016). Other advanced robotic systems enable to isolate cells in a single pen of 1.7 nL on a chip containing 1750 pens, the Beacon from Berkley Lights[4]. Also, devices that run perfusion cell cultures at a scale of only 2 mL with fully automated fluid controls and on-line measurements (Mobius Breez, MilliporeSigma) are commercially available[5].

While in drug discovery[6][7] and at the initial screening stages, different machine learning tools are used for micro-well plates experiments[8] the design of advanced cultivation strategies that include optimal media composition and optimal profiles of feeding, pH and temperature among others, require larger vessels as well as advanced monitoring, and control systems[9][10]. Additionally, with an increasing complexity on modern production processes as is continuous production[11], experimental design and operation face new challenges. As stated in the thorough review by[12], the current literature lacks a comprehensive solution for the autonomous operation of parallel cultivation systems with advanced controls. Furthermore, experimental setups in high throughput that properly mimic industrial conditions are key to maximize speed and robustness during scaleup[13] and promote development following the QbD principles. In the field of cell culture, this has driven the development of highly sophisticated experimental systems, such as the commercially available family ambr® robots[14] (Figure 1). This stage represents a significant bottleneck in development, as even with the presence of robotic systems, the design and execution of experiments still heavily rely on human intervention.

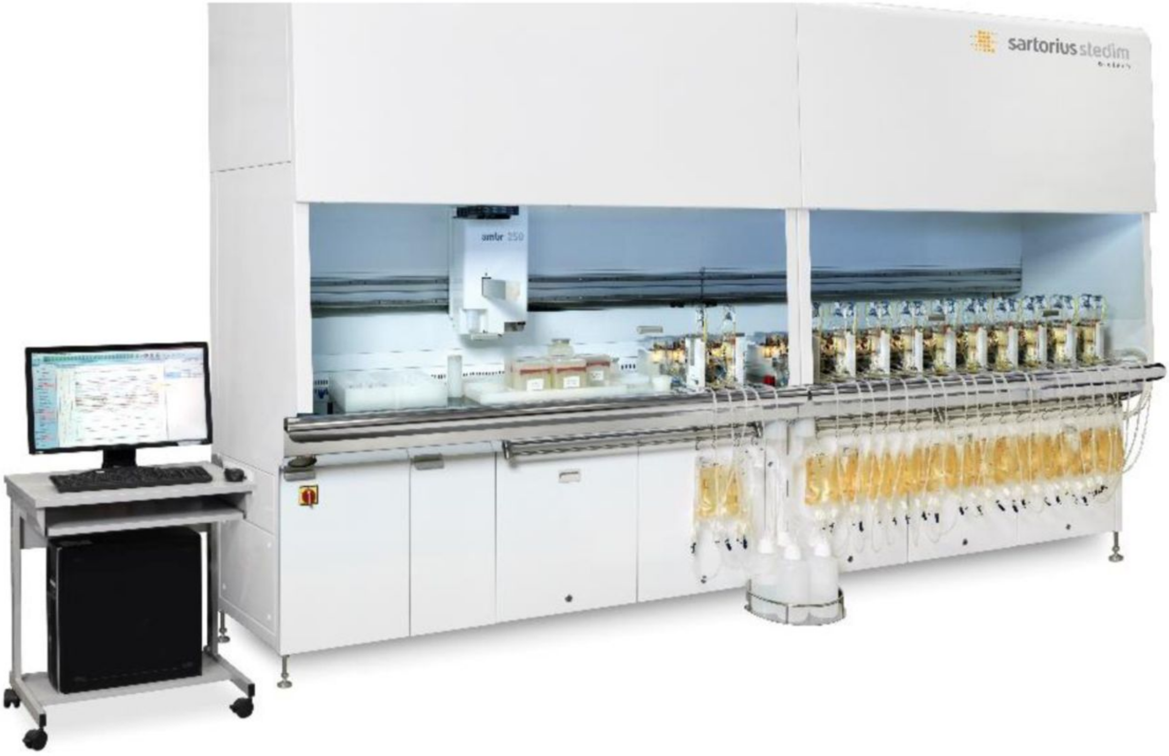


Figure 1. ambr®250 perfusion system

At this stage, emulating industrial process conditions is key to accelerate scaleup and minimize the risk of failure. Nevertheless, performing proper scale-down experiments for cell cultures poses significant challenges[15]. The cultivation demands a tight control over several critical factors such as agitation, aeration, feeding, pH and temperature among others The complexity of the system is further increased with the incorporation of long-term membrane perfusion cultivations. This setup requires simultaneous control of additional media inlet flow, continuous harvest of liquid bulk to maintain constant working volume, cell retention within the reactor, and a bleed output for process purging. Despite the high level of automation in these systems, they still heavily rely on manual monitoring and operation. While low level controls (feedings, pH, temperature) are well implemented, computational tools that make important high-level decisions during operation are currently missing. Data acquisition also presents challenges, with parameters like pH, dissolved oxygen, and temperature being measured online, while metrics such as viable cell density, viability, and glucose concentration are assessed at-line. Furthermore, certain quality attributes may take weeks to quantify[16]. All these aspects make it difficult for scientists to optimize the experiments while they are running. Also, the necessary computational tools to support operators and perform autonomous experiments are missing. This is in contrast with both, the ever-increasing number of computational tools developed and applied in biopharma in manufacturing and other development stages[17], and the technologies already deployed in other fields, for example in material science and chemistry[18].

Self-driving research is pushing the boundaries of digital discovery by giving full autonomy to “robot scientists”[19]. These systems can perform experiments with minimum human intervention, learn from their experimental results, and decide how to continue experimentation[20]. While self-driving clearly suggests an association with mobile systems[21], the use cases of self-driving laboratories rarely deal with dynamic systems and the challenges hereby involved[22]. This has evidently been no limitation to achieving incredible results in chemistry, catalysis, material science, and biology[23] but process dynamics become increasingly relevant as we approach industrial scale production[24][25]. To create self-driving machines for effective bioprocess development and scale-up, it is necessary to integrate process dynamics and control into the framework.[26]. We need hence a digital twin (DT) of the dynamic cultivation process to ensure an optimal design of the experimental campaign and its proper operation by automated systems[27]. Furthermore, due to system uncertainty and discovery related nature of experiments in R&D, the DT requires important cognitive properties[28]. The capability to use existing knowledge and learn from the experiment that is being operated[29] is essential to ensure the fastest development possible and efficient process control of well-designed experiments[30]. Lastly, designing experimental campaigns involving multiple parallel runs, extended cultivation durations, and numerous input variables to optimize poses a significant challenge for Optimal Experimental Design (OED). OED and optimal experimental operation problems have been tackled for many decades by the Process Systems Engineering (PSE) community[31] and has also rapidly gained popularity in the machine learning community[32]. There are also several applications in parallel systems[33] and specifically tailored for bioprocess cultivations[34][22][35].

Nevertheless, there are important drawbacks in existing methods. First, most applications work either offline or decouple the learning phase from the experimental plan. A Cognitive Digital Twin, with the ability to learn on the fly during experimentation is needed to ensure an efficient experimental campaign of new clones and/or processes. The standard methods for adaptive control, for example, still rely on mechanistic models and reliable estimates for the initial parameter values[36].

Second, the transfer of learnings from previous processes to new ones is essential to ensure the use of all available knowledge and a rapid development of new products[37]. To tackle this, frameworks that are capable of both learning form experiments and transferring previous learnings are needed. Unfortunately, the tools that have been developed up to now for biopharma are very specific and require expertise in modelling, optimization, and programming to be adapted to new scenarios[27]. Today, R&D in biopharma is missing the necessary toolset to fully exploit the potential of parallel mini-bioreactor systems and speed up the generation of knowledge to reach truly accelerated innovation. These solutions need, furthermore, to be user friendly to enable its use by experts in experimentation of mammalian processes.

Furthermore, the implementation of robust tailor-made computational pipelines and development of corresponding computational workflows that can automatically handle and store all the data and metadata generated before, during, and after conclusion of an experiment, play an essential role[38][39]. Nevertheless, speaking of miniaturized bioreactor cultivations, while the experimental tasks have been automated to great extend and the throughput increased significantly in the last decade, less advances have been achieved in embedding automated computational workflows for data management and decision making. As a result, while the robots can perform very complex tasks automatically for long periods of time, experts' intervention is still regularly required to analyze the data and to operate the parallel cultivations. This has to be considered in the context of the large experimental time spans (up to months of cultivation) and the high costs of the experiments reaching 50 k€ in some cases.

Finally, the high degree of automation of the experimental setup necessitates an integrated decision- making agent, to minimize the human involvement in these lengthy and time-consuming experiments. Algorithms that make the best decisions to operate several cultivation processes in parallel maximizing the knowledge derived from such experiments, are pivotal for accelerating development.

In an effort to tackle these issues and significantly reduce time and costs in the development of monoclonal antibody manufacturing strategies, we present a user-friendly software solution that enables the autonomous operation of the parallel bioreactor system with perfusion membrane modules. The key features are:

1. Hybrid modelling formulation for flexible and robust process
2. Online model re-training to enable real-time learning
3. A toolset to transfer learnings from past projects
4. Predictive process monitoring and notifications autonomous feed-back experimental operation minimizing the human in the loop

The results demonstrate the capabilities of step-wise Gaussian process models (SW-GP) to learn from the data, to transfer learnings between different cell lines, and to support optimal experimental design with Bayesian methods. They also show that the optimal operation can robustly fulfill challenging tasks throughout longer periods of time and maintain critical conditions as is cell viability within specification.

2. Computational Methodology

In this section, we first discuss the communication with the experimental system using a Structured Query Language (SQL) database. We then shortly discuss the computational framework based on hybrid modelling and Bayesian optimization to finally give a detailed description of the experimental setup.

2.1. Gateway connection and data storage

As more laboratories adopt HT experimental technologies and the amount of data generated per time increases, systems for HT data treatment and storage are imminent to assure the traceability and provenance of the experiments as well as to keep up with the data generation speed.

The communication between the ambr250 system and the software is based on a ClientServer Architecture, over TCP/IP standards developed together with Prof. A. Papaemmanouil and T. Prudhomme from Institutsleiter Elektrotechnik, Hochschule Luzern T&A.

The bioreactor system interfaces with an OPC UA Server (KEPServerEX), from which data transmissions to an OPC client over TCP/IP, to a MySQL Database, allowing to collect and store the data from the experimental system. The software interfaces with the SQL client, allowing to query the process data stored in the MySQL Database. Most importantly, the software is also able to send experimental set- points to the OPC client, which are then enacted in the bioreactors with low level control in real time.

Additionally, all metadata is stored in an Excel macros file (.xlsm) and the actions performed by the robots is recorded in (.csv) files and stored to ensure full traceability of the robotic actions during experimentation[39]. The communication architecture is shown schematically in Figure 2.

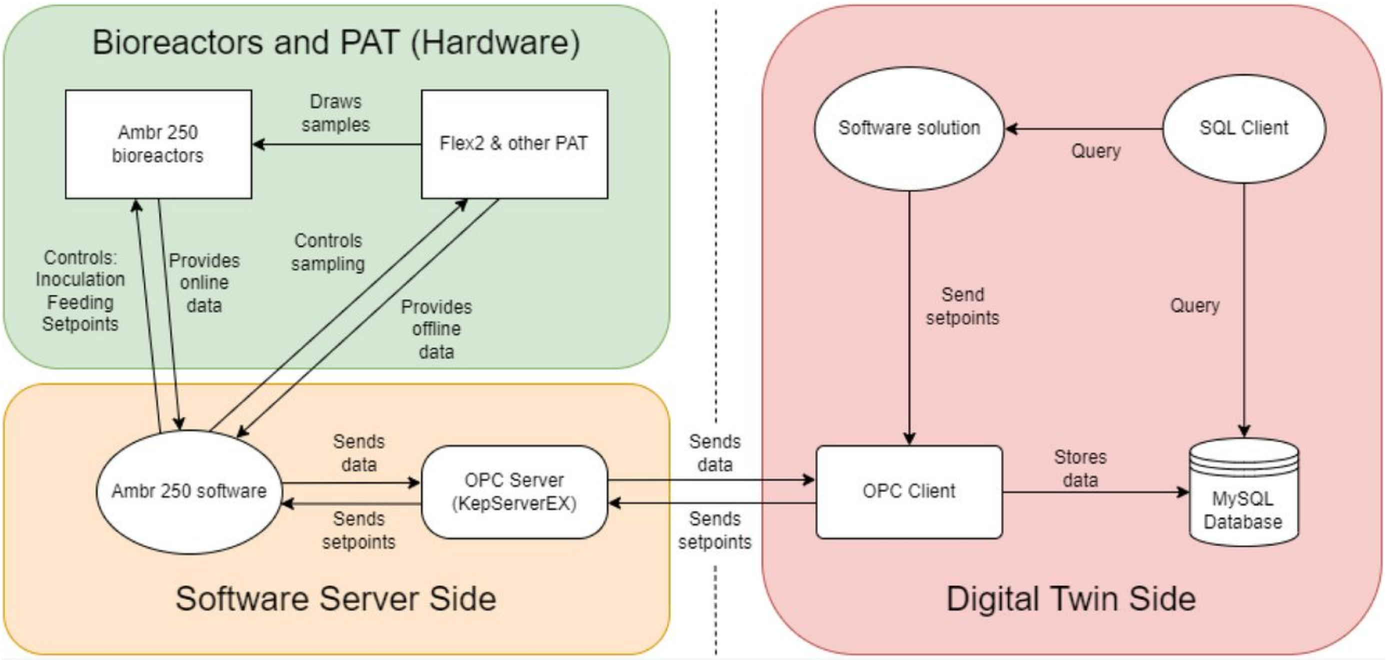


Figure 2. Overview of the interconnectivity of the DT software solution, the OPC and SQL client, as well as the ambr®250 server.

2.2. Hybrid gaussian process model

The mathematical model used to describe the evolution of the bioreactors during experimentation needs to consider a system with a vast number of cellular biochemical reactions that take place in bioprocess cultivations[40]. In a bioreactor, the chain of reactions that take place in the cells for metabolite secretion and/or antibody production might be unknown. The existing models might not capture all the factors affecting the reactions, for ex. pH and temperature dependency on shear rates. The presence of impurities in the media can also affect cellular reactions and are very difficult to detect. Due to these reasons, purely mechanistic model might not be able to accurately predict process behavior in a bioreactor[41]. Hybrid modelling is an alternative approach, which combines the advantages of mechanistic and data-driven models to better describe complex systems[42].

Hybrid models are widely applied for biological processes since added to this large complexity we are confronted with limited and low-quality observations. From the thousands of biochemical reactions taking place intracellularly, we can at most observe a few metabolites and enzymes. Most importantly, the desired product that undergoes a very strict quality control (a large number of Critical Quality Attributes based mostly on glycans and impurities) require very complex analytical methods, strongly limiting the number of possible samples per experiment.

For our current purposes, we use a hybrid model formulation based on basic mass balance equations represented by a system of ordinary differential equations (ODE) combined with Gaussian Process (GP) models as the data-driven counterpart of the hybrid model[43][44]. GPs use a measure of similarity (kernel function) between points in the training data to predict the distribution of a value for an unseen datapoint. With the kernel function being usually non-linear, such an algorithm is capable to reproduce highly non-linear and complex behavior. The major advantage being the estimation of both, the predicted value and its associated uncertainty[45].

In literature, Gaussian Process State Spaces Models (GPSSM)[46], that aim to describe the state space with GPs, have shown to also represent mammalian cultivations well[47]. This formulation allows to use GPs to describe nonlinear dynamic processes with limited knowledge of the phenotype and growth dynamics of the cultivation[37]. The advantages of GPSSMs have been documented in several applications with small to mid-size data sets with noise that is close to Gaussian[48].

Given the slow dynamics of mammalian processes, results show that discretion with a time step of up to 24 hours is sufficient to properly mimic the evolution of the process over time. Based on this assumption, we can significantly reduce the computation burden implementing a SW-GP as described here.

When using ODE systems, material balances can be generalized as follows:

$$\frac{dc \cdot V}{dt} = R(s) \cdot V + u_f - u_b - u_p = r(s) \cdot VCD \cdot V + u_f - u_b - u_p \qquad \text{(Eq.1)}$$

where c is a vector of concentrations (e.g., Viable Cell Density (VCD), glutamine, glucose, lactate, ammonium, titer);s a vector of process states (i.e., all time dependent variables, including the concentrations c); V is the culture volume; t is a vector with time stamps t_i; u_f, u_b, and u_p vectors of mass feed rates (nonzero only for compounds that are fed), mass bleeding rates, and mass perfusion rates, respectively. The term $R(s)$ is generally representing the rate of production/consumption of a species as function of the state vector s. Since we are mostly interested in the specific performance per cell, it is possible to further expand $R(s)$ as $R(s) = r(s) \cdot VCD$, where $r(s)$ is the cell specific rate of production/consumption.

The ODE system is discretized as follows:

$$\frac{dc \cdot V}{dt} = V \cdot \frac{dc}{dt} + c \cdot \frac{dV}{dt} \cong V \cdot \frac{c\left(t_{i+1}\right) - c\left(t_i\right)}{t_{i+1} - t_i} + c \cdot \frac{dV}{dt} \qquad \text{(Eq.2)}$$

$$V \cdot \frac{c\left(t_{i+1}\right) - c\left(t_i\right)}{t_{i+1} - t_i} \cong R(s) \cdot V + u_f - u_b - u_p - c \cdot \frac{dV}{dt} \qquad \text{(Eq.3)}$$

where $c(t_i)$ is the concentrations measured at the step$i$, and $c(t_{i+1})$ the concentration measured at the following step,$i + 1$. For the sake of simplicity, in Eq. 3 all quantities on the right-hand side are evaluated at time t_i. Note that the change in bioreactor volume over time, dV/dt can be computed explicitly. Using the discrete hybrid model, called SW-GP from now on, it is possible to calculate the term $R(s(t_i))$ at every time step t_i, as all other terms are measured, by rearranging Eq. 2:

$$R(s) = \frac{c\left(t_{i+1}\right) - c\left(t_i\right)}{t_{i+1} - t_i} - \frac{1}{V} \cdot \left[u_f - u_b - u_p - c \cdot \frac{dV}{dt}\right] \qquad \text{(Eq.4)}$$

We now can explicitly compute the discrete rate of production for every species at the step i, and relate such term to the state of the process measured at the step i, *i.e.*, $s(t_i)$. In the SW-GP formulation, this is done using a GP model, *i.e.*: $R(s) \approx GP(s)$. The inputs to the GP are the process states s measured at every time t_i, while the outputs are the corresponding discrete rates of production, R.

The formulation of the discrete hybrid model of Eq. 2 allows to learn the process evolution in time for each variable in a stepwise fashion. Once the initial condition $s(t_0)$ are defined, where $t_0 = 0$, it is possible to compute the rate of production for each state variable in the vector $c(t_i)$ corresponding to the step $(t_{i+1} - t_i)$ and, using the discretized mass balance above, compute e.g. the value $c(t_1)$ at time t_1. At this point, the state of the process at the new time, $s(t_1)$, is defined and the procedure can be repeated for all steps.

It is worth noting that the discrete hybrid model of Eq. 2 can be further simplified as follows:

$$\frac{c\left(t_{i+1}\right) - c\left(t_i\right)}{t_{i+1} - t_i} \cong R(s) + \frac{1}{V} \cdot \left[u_f - u_b - u_p - c \cdot \frac{dV}{dt}\right] = \tilde{R}\left(s, u_f, u_b, u_p, V\right) \qquad \text{(Eq.5)}$$

In this formulation, the contributions of the different feeds or outlets to the material balances are lumped into a single effective rate of production, $\tilde{R}$. Such model is still able to infer the effect of different feed strategies on the evolution of the different concentrations, even though such contributions are not made fully explicit. This formulation of the material balances can return better models when feeds are not fully or precisely characterized, thus letting the machine learning part to compensate for such lack of knowledge.

The SW-GP model is trained as an ensemble of smaller Gaussian Process models. The model consists of many sub-models, which are individually trained on randomly sampled subsets of the full training data. For each prediction, the sub-models individually predict the process dynamics, and these predictions are used to calculate confidence intervals. Specifically, the 10th to 90th percentiles of the aggregated predictions provide the 80% confidence interval, allowing for uncertainty estimates. The median prediction is given by the 50th percentile.

2.3. Optimization framework

The optimizer feature in the software solution uses a Bayesian optimization algorithm, which computationally is sufficiently fast as the mammalian process dynamics are slow. It searches the parameter space of set points (in particular, temperature, agitation rate, vessel volume per day (VVD) and pyruvate additions) with a Bayesian surrogate using the GP formulation described, by utilizing the models to guide its acquisition function (the expected improvement acquisition function was used here) evaluation to find the maximum.

The objective function is constructed by defining a target value for the variable to optimize and by considering only the model predictions of the next 3 daily timesteps. The predictions utilized to evaluate the performance of the process at the given parameter values aiming to find the optimal inputs to bring the process closest to the defined target.

Mathematically, this is described as,

$$\mathrm{OFV} = \left(\hat{y}_{t+1} - y_{\mathrm{tar}}\right)^2 + \left(\hat{y}_{t+2} - y_{\mathrm{tar}}\right)^2 + \left(\hat{y}_{t+3} - y_{\mathrm{tar}}\right)^2 \qquad \text{(Eq. 6)}$$

where *OFV* is the objective function value to be minimized, $\hat{y}_{t+j}$ is the predicted value of the process variable *y* at *j* days in the future and y_{tar} is the desired target value of *y*. The inputs that were chosen to be varied were the set-points of temperature, agitation rate, VVD (perfusion rate) and pyruvate addition.

2.4. Model error calculation

The relative root mean squared error (rRMSE) is used to evaluate the model error. Compared to the absolute RMSE it is normalized by the standard deviation of the respective variable over the entire data set, making the resulting values comparable between variables. In this case, the model was evaluated considering its purpose within the control framework, namely in its capability to predict just the next three daily timesteps, since for the real-time optimization this is the time horizon that was considered. So, 3-day model predictions are done iterating over the entire process duration, from day 0 to the day before the last.

Therefore, the overall error of the model for a variable *x* is given by,

$$rRMSE_x = \frac{1}{\sigma_x}\sqrt{\frac{\sum_{t=0}^{t_f-1}\sum_{j\le t_f-t}^{h}\left(\hat{x}_{t+j,t} - x_{t+j}\right)^2}{n_{\mathrm{points}}}} \qquad \text{(Eq.7)}$$

where *tf* is the process duration in days, *t* the current process day from which the model predicts, *h* the prediction horizon of interest (here three time steps), $x\hat{}_{t+j,t}$ the prediction of variable *x* from day *t* to day *t+j*, x_{t+j} the future observed value on day *t+j*, $\sigma_x$the standard deviation of the respective X variable and *n*p the total number of points to be averaged over, given by,

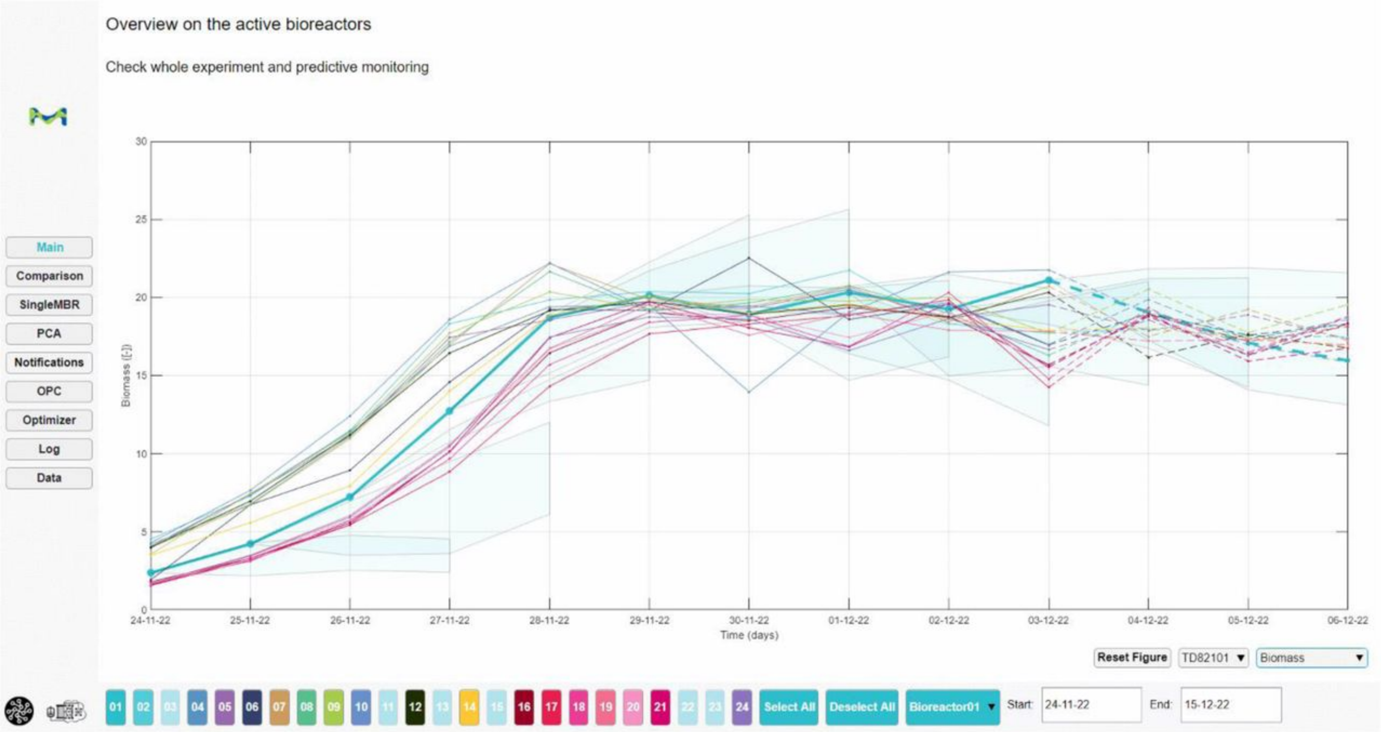
$$n_p = h \cdot t_f - \sum_{j=1}^{h-1} j \qquad \text{(Eq.8)}$$

2.5. User interface

Finally, in order to enable a human centric digitalization of the development process in compliance with bioprocessing 5.0[49], a user interface was developed together with expert users and technical operators.

The software solution was equipped with multiple features, which are briefly described below, where some are supplemented with a screenshot of the respective interface:

- **Main Overview (**Figure 3**):** Visualization of all measured data for all ongoing bioreactor runs, as well as past experiments. The user may choose any process variable, highlight individual runs or remove them from the visualization.
- **Comparison:** Multiple univariate data visualizations in one interface, allowing fast comparison across variables and ongoing runs.
- **Model evaluation (**Figure 4: Overview of all past model predictions for a single run in focus. Includes observed vs. predicted for a given process day as well as rRMSE metrics.
- **PCA (**Figure 5**):** Multivariate data analysis of all ongoing runs, allowing to identify key correlations between variables, as well as detecting outliers.
- **Notifications:** The user may set up alarms regarding violation of critical limits of important process variables. The model predicts if constraints will be violated in the future and informs the user on a daily basis if this will be the case.
- **OPC:** In this section the current connection status is shown and can be queried.
- **Optimizer (**Figure 6**):** Here the objective functions for optimization are defined on an individual basis for each bioreactor. The control parameters (such as temperature, perfusion rate, agitation rate and pyruvate addition) are chosen, as well as the lower and upper bounds for search space definition. For the target variables a target value is defined, or they are simply maximized or minimized.
- **Log:** Record of all actions taken by the software, such as data queries, written set-points and connectivity status.
- **Data:** Tabular visualization of all process data of ongoing or past runs.



Overview on the active bioreactors

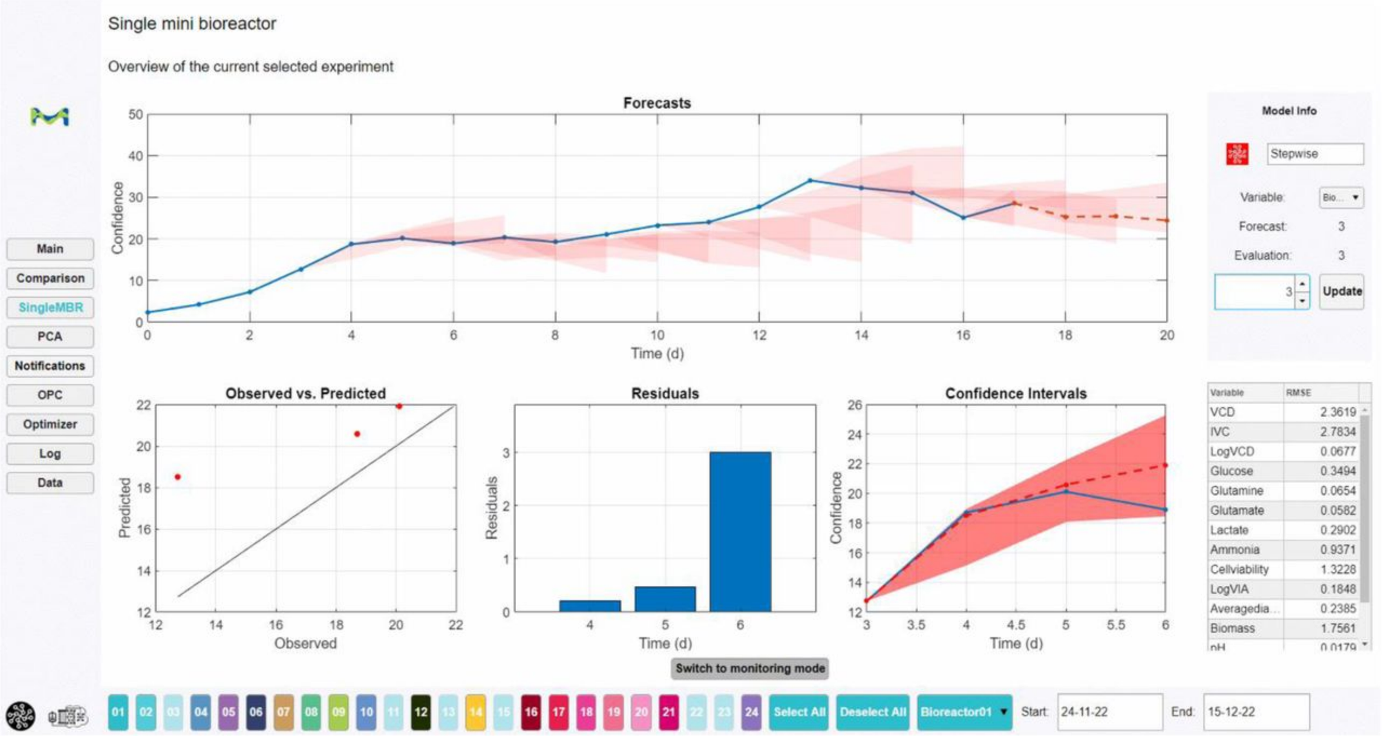
Check whole experiment and predictive monitoring

Start: 24-11-22

End: 15-12-22

Figure 3. Main overview of active bioreactors. All process variables can be visualized. The solid line shows the measured process data, whereas

the dashed lines that start from the current process day represent the model's predictions three days into the future. The shaded polygons are the past and present prediction intervals of the highlighted run.



Stepwise

Variable: Bio...

Forecast: 3

Evaluation: 3

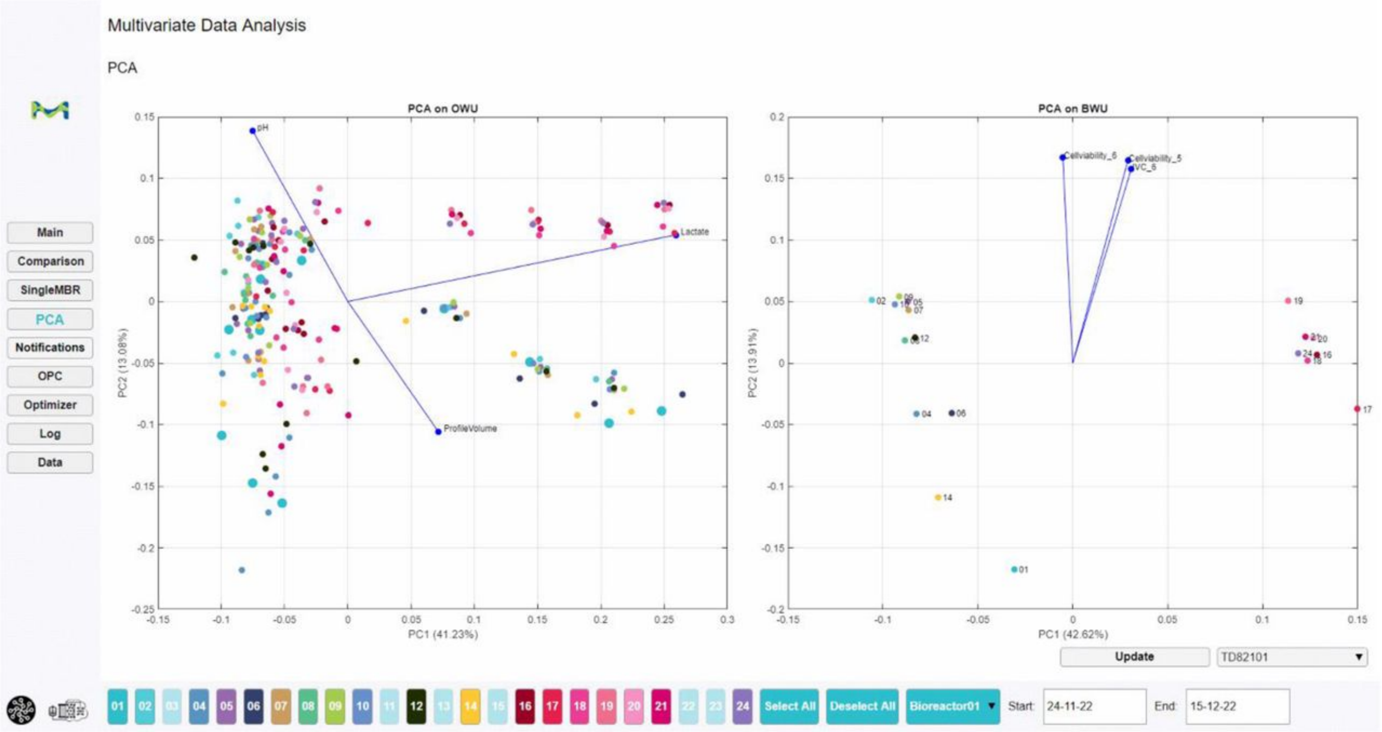
3

Variable	RMSE
VCD	2.3619
IVC	2.7834
LogVCD	0.0677
Glucose	0.3494
Glutamine	0.0654
Glutamate	0.0582
Lactate	0.2902
Ammonia	0.9371
Cellviability	1.3228
LogVIA	0.1848
Averagedia...	0.2385
Biomass	1.7561
pH	0.0179

Start: 24-11-22

End: 15-12-22

Figure 4. Model evaluation for a single run. Past forecast's prediction intervals are shown in the top plot, as well as current predictions. The visualizations on the bottom show evaluations for a selected process day (here day 3), such as observed vs. predicted, absolute residuals and the prediction from that day onward with the prediction interval.



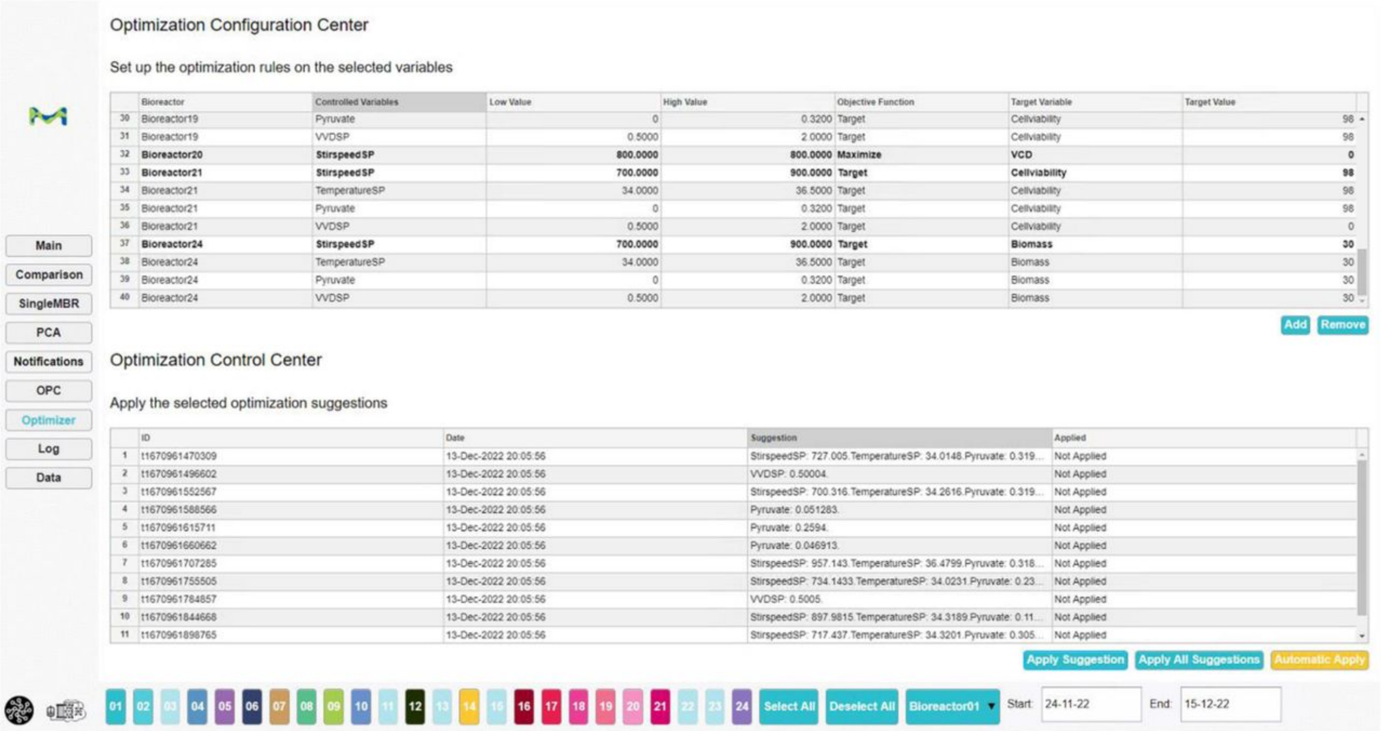
TD82101

Start: 24-11-22

End: 15-12-22

Figure 5. PCA plots of the variable- or observation wise unfolded (OWU) matrix on the left and batch-wise unfolded (BWU) matrix on the right.

Variance explained of the first two principal components are shown, the three loadings with the largest magnitude are highlighted as blue vectors.



Optimization Configuration Center

Set up the optimization rules on the selected variables

	Bioreactor	Controlled Variables	Low Value	High Value	Objective Function	Target Variable	Target Value
30	Bioreactor19	Pyruvate	0	0.3200	Target	Cellviability	98
31	Bioreactor19	VVDSP	0.5000	2.0000	Target	Cellviability	98
32	**Bioreactor20**	**StirspeedSP**	**800.0000**	**800.0000**	**Maximize**	**VCD**	**0**
33	**Bioreactor21**	**StirspeedSP**	**700.0000**	**900.0000**	**Target**	**Cellviability**	**98**
34	Bioreactor21	TemperatureSP	34.0000	36.5000	Target	Cellviability	98
35	Bioreactor21	Pyruvate	0	0.3200	Target	Cellviability	98
36	Bioreactor21	VVDSP	0.5000	2.0000	Target	Cellviability	0
37	**Bioreactor24**	**StirspeedSP**	**700.0000**	**900.0000**	**Target**	**Biomass**	**30**
38	Bioreactor24	TemperatureSP	34.0000	36.5000	Target	Biomass	30
39	Bioreactor24	Pyruvate	0	0.3200	Target	Biomass	30
40	Bioreactor24	VVDSP	0.5000	2.0000	Target	Biomass	30

Optimization Control Center

Apply the selected optimization suggestions

	ID	Date	Suggestion	Applied
1	t1670961470309	13-Dec-2022 20:05:56	StirspeedSP: 727.005.TemperatureSP: 34.0148.Pyruvate: 0.319...	Not Applied
2	t1670961496602	13-Dec-2022 20:05:56	VVDSP: 0.50004.	Not Applied
3	t1670961552567	13-Dec-2022 20:05:56	StirspeedSP: 700.316.TemperatureSP: 34.2616.Pyruvate: 0.319...	Not Applied
4	t1670961588566	13-Dec-2022 20:05:56	Pyruvate: 0.051283.	Not Applied
5	t1670961615711	13-Dec-2022 20:05:56	Pyruvate: 0.2594.	Not Applied
6	t1670961660662	13-Dec-2022 20:05:56	Pyruvate: 0.046913.	Not Applied
7	t1670961707285	13-Dec-2022 20:05:56	StirspeedSP: 957.143.TemperatureSP: 36.4799.Pyruvate: 0.318...	Not Applied
8	t1670961755505	13-Dec-2022 20:05:56	StirspeedSP: 734.1433.TemperatureSP: 34.0231.Pyruvate: 0.23...	Not Applied
9	t1670961784857	13-Dec-2022 20:05:56	VVDSP: 0.5005.	Not Applied
10	t1670961844668	13-Dec-2022 20:05:56	StirspeedSP: 897.9815.TemperatureSP: 34.3189.Pyruvate: 0.11...	Not Applied
11	t1670961898765	13-Dec-2022 20:05:56	StirspeedSP: 717.437.TemperatureSP: 34.3201.Pyruvate: 0.305...	Not Applied

Start: 24-11-22

End: 15-12-22

Figure 6. Optimization configuration center. For each bioreactor, the control variables are configured in terms of lower and upper bounds to define the optimization search space. The target variables can be maximized or minimized, or a specific target value can be chosen. Each day, optimizations are performed upon querying the current data. The software supports manual applying of suggested conditions or an automatic mode.

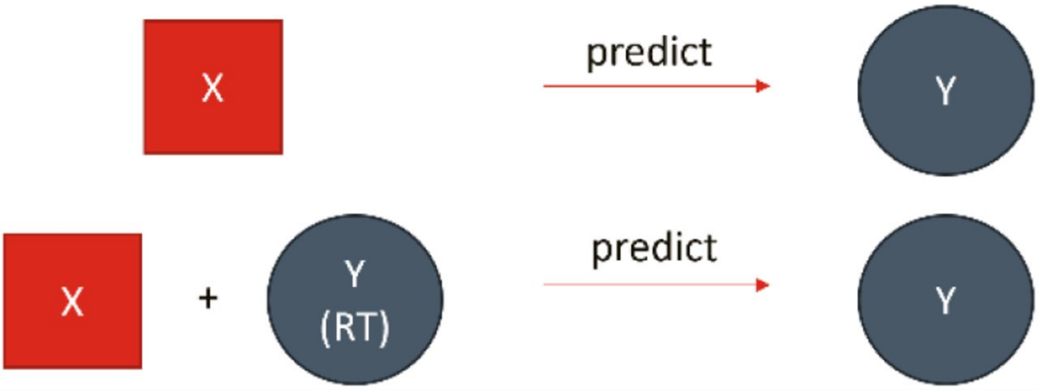


Figure 7. Schematic representation of model comparison approach. In the first case (top row), the model is only trained on clone A and used directly to predict runs of clone B. In the second approach, the model is initially trained on clone X, but then continuously retrained with current available clone Y data during the process (RT).

3. Material and Methods

3.1. Cell expansion

Cells were thawed and diluted at 0.30 10^6 cells/mL in 50 mL SpinTube (TPP, Trasadingen, Switzerland). Cells were diluted every 2 or 3 days to 0.30 10^6 cells/mL or 0.20 10^6 cells/mL respectively and using proprietary expansion medium. Cell culture volumes were increased from SpinTubes to 2L wavebags (Sartorius, Goettingen, Germany). After two weeks of expansion, cells were transferred to inoculate the perfusion cell cultures on the AMBR250 (Sartorius, Goettingen, Germany).

3.2. Perfusion process

ambr®250 perfusion bioreactors were inoculated with cells at the maximum concentration available in 213 mL of final volume. Cells were cultivated in a proprietary chemical defined medium. During the growth phase, the perfusion rate was increased from 0 to 1.3 d-1 until reaching the target viable cell volume (VCV) which was calculated according to the following equation,

$$\mathrm{VCV}(\%) = \frac{4}{3} * \pi * \left(\frac{d}{2}\right)^3 * VCD * 100 \qquad \text{(Eq.9)}$$

where d is the average cell diameter and VCD the viable cell density (10^6 cells/mL).

The perfusion volume was maintained constant at 213 mL throughout the duration of the run. Once the VCV target was reached, the culture was considered to be in a state of control and the given setpoints for the input parameters were followed (see Table 1 and Table 2). The bleeding was triggered if the value of VCV was higher than the setpoint. Sample analysis of the bioreactors was performed daily using a Flex2 (Nova Biomedical, Waltham, MA, USA) to count the cells, measure cell viability (Via), average cell diameter (Diam), quantify glucose (Glc), glutamine (Gln), glutamate (Glu), lactate (Lac), ammonium (Amm) concentration, pH, pO_2, pCO_2 and osmolality. Samples for off-line analysis were taken to quantify monoclonal antibody (mAb) titer, amino acid concentrations and quality attributes (glycans, oxidation and deamidation) of the mAb produced. A glucose bolus was performed if the glucose concentration was lower than 2.00 g/L after the sampling. The feed volume was calculated to replenish glucose to a concentration of 5.00 g/L.

Table 1: Steady state experimental plan to build a design space where the digital twin will build and update basal models

Parameter	-1	0	1
Perfusion rate (Vessel Volume per Day (VVD, d^{-1}))	0.5	1.25	2
Temperature shift after reaching VCV target (°C)	34	35.25	36.5
Stir speed (rpm)	700	1050	1400
Pyruvate feed flowrate ($g.L^{-1}.d^{-1}$)	0	1	2

Table 2. Clone tested, control variables and output variables for each bioreactor.

Condition	Clone	Variable to be controlled	What to optimize
1	Clone A	VVD, Temperature, stir speed, pyruvate flow	Target 30% VCV
2	Clone A	Pyruvate flow	Target 1mM NH_4^+
3	Clone B	Pyruvate flow	Target 1mM NH_4^+
4	Clone B	VVD, Temperature, stir speed, pyruvate flow	Target 98% viability
5	Clone B	VVD, Temperature, stir speed, pyruvate flow	Target 98% viability
6	Clone B	Pyruvate flow	Target 1mM NH_4^+

Variables to be controlled could change in the tested ranges of Table 1 except for stir speed which could evolve between 700 to 1050 rpm as higher rpm was deleterious for cell viability.

3.3. Training experiment

A perfusion experiment consisting of 24 runs with a fixed experimental design according to Table 1 was performed with one cell line CHO-K1 Clone A. The purpose of this run was to gather data with sufficient variability for the initial model training using standard process conditions. This experiment is referred to as the Training experiment.

Once the VCV target was reached, all parameters gathered in Table 1 were changed according to the design of experiment.

3.4. Use case experiment

The second perfusion experiment (Use case experiment) was performed with two cell lines: Clone A (same clone as Training experiment) and Clone B (new clone). Once the state of control was reached, inputs suggested as setpoint by the model aiming to comply with the constraints and reach the objectives given to each bioreactor (Table 2). All other set-points were kept as in the previous experiment.

4. Results

4.1. Online retraining of hybrid models (example with in-silico data)

The main challenges related to the design and operation of informative experiments in R&D can be strongly alleviated by 1. the transfer of existing knowledge from similar processes and 2. a frequent re- calibration of the models as data is being recorded, to maximize the efficiency of long experimental campaigns (several weeks) of experimentation. To show the added benefits of retraining a model during an ongoing experiment, in-silico datasets for two different clones (clone X and clone Y), corresponding to a dataset size with 24 run each, were generated using experts' knowledge of the real process. The data was generated using a mechanistic model that describes fed-batch cultivation of mammalian processes with lactate consumption (clone X) and without lactate consumption (clone Y). A fed-batch process over 14 days of cultivation with similar conditions to the use case (see M&M) was considered to generate the insilico data. The state variables are VCD, titer, glucose (Glc), glutamine (Gln), ammonium (Amm), lactate (Lac), dead cell density (DCD) and lysed cells (Lysed). The input variables that were controlled are the stirring rate set-point, the DO set-point, initial conditions of glutamine, glucose and VCD, and the feeding volumes. The hybrid model was initially trained using only data from clone X with 24 runs. The experimental design was simulated with each insilico data point treated as a daily measurement. On each timestep, the model is re-trained using the existing insilico data, prediction calculated for the next three days. We show two different strategies to demonstrate the added value of re-training and the effect of the transfer learning properties of the framework. In the first, the initial model of clone A is used, without adaptations. In the second, the new process data of the clone B runs is added to the training data and the model is retrained.

Prediction errors were then computed for a 3-day horizon considering the operation scenario. The comparison between the two approaches is shown in **Error! Reference source not found.** using the rRMSE as a metric. The results of the above comparison are shown in Figure 8, where the rRMSE was determined as described in section 2.4. The prediction error evolutions with time for a set of variables are depicted in Figure 9.

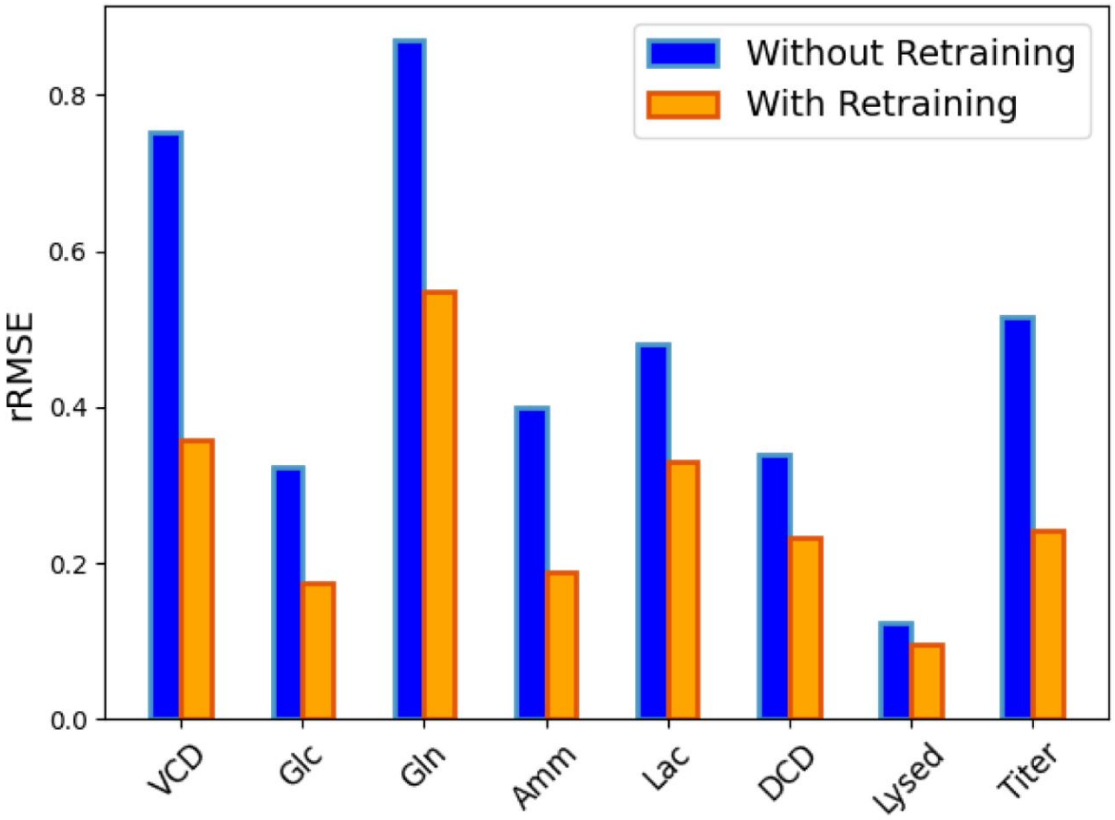


Figure 8. Comparison of relative RMSE results when the model is not being retrained (blue) and when it is retrained (orange). Predictions are evaluated over a 3-day horizon. The model errors drop substantially when ongoing retraining using daily new measurements is applied.

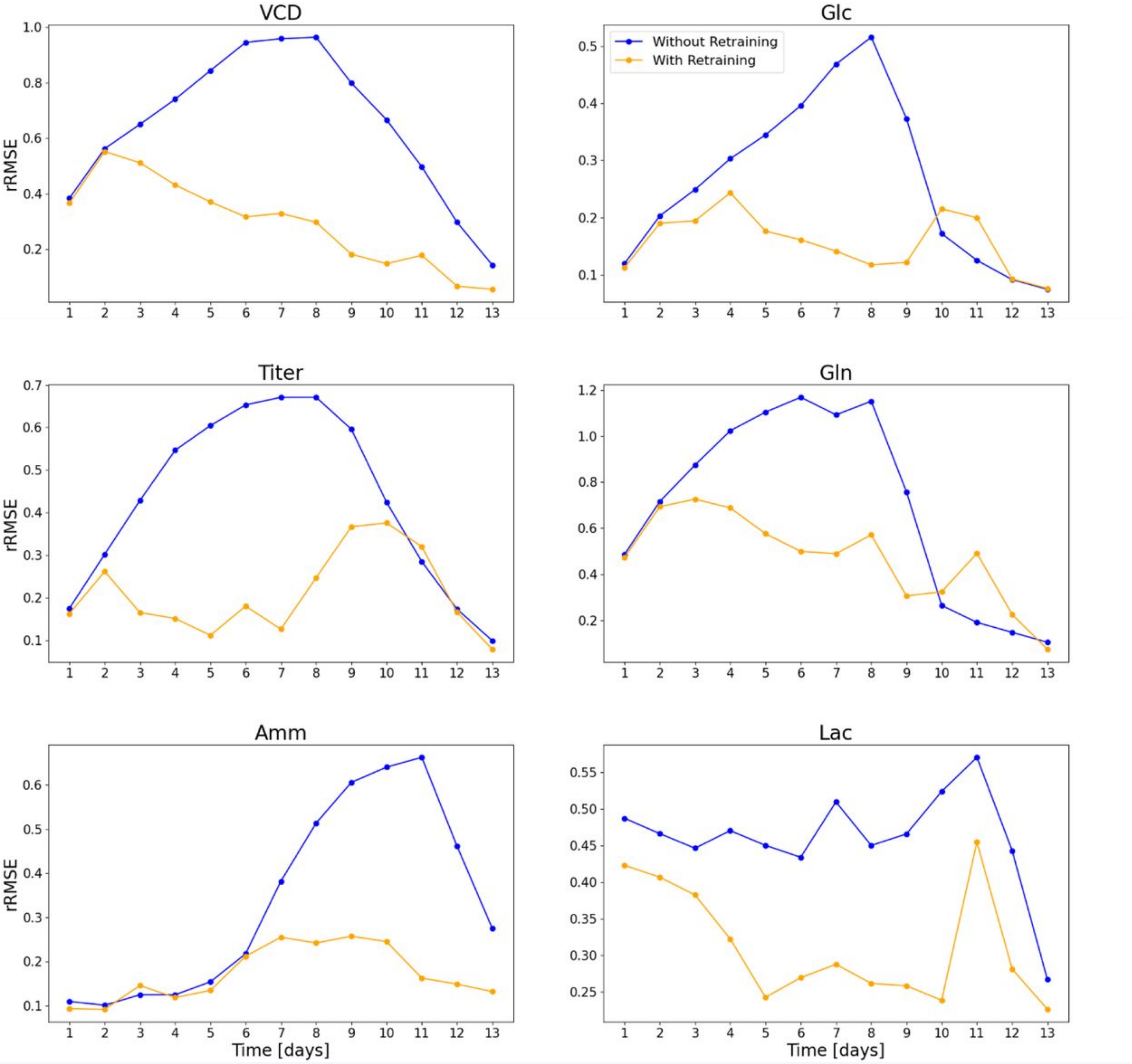


Figure 9. Relative RMSE as a function of the process day the prediction was made from. Model errors were averaged over all 24 runs in the simulated validation campaign. For the error calculation, only 3-day ahead predictions were considered until day 11, for day 12 the model predicted two more days ahead, and for day 13 the model predicted only one day ahead, as the total process duration was 14 days. The results are shown for VCD, Titer, glucose, glutamine, ammonium, and lactate. The model error for the retrained model are decreasing over time for most variables, while at the same time outperforming the model that is not retrained.

In Figure 8, when the model is not retrained (blue), the errors for some variables, most importantly VCD, are large (above 0.5). The cell behavior of clone B is different to clone X of the simulated Excitation Experiment in some respects, which the model does not understand. If the model is continuously retrained using new daily measurements, the overall error over the entire process improves drastically for each variable. This shows that the model can learn from the ongoing measured data and adapt its parameters to model the new clone. This is consequential for the model's ability to be used in optimization of the ongoing process, as without retraining the clone's sensitivity towards certain parameters might not be accurate enough.

Similar to the above conclusion, the error of the “With retraining model” (displayed in orange in Figure 9) is lower than that of “Without retraining model” (displayed in blue in Figure 9) throughout the process, and the divergence usually occurs in 2-6 days. For some variables, notably VCD, the error also consistently decreases with more data being gathered, demonstrating even further the power of retraining the model. For some variables the model error at the end of the process is briefly higher for the retrained model. This coincides with the stopping of the feed on day 10. A possible explanation is that the model with retraining has gathered a lot of data where the feed is active, temporarily becoming worse at predicting the process when there is no feeding. However, as can be seen for glucose, titer, glutamine and lactate, with retraining the model recovers and again predicts better than the model without retraining. The error of the model without retraining also gets lower towards the end of the process. This is expected, as predictions horizons shorten and the process either has stagnated or the cells have died, which is easier to predict for any model. Nevertheless, the model with retraining clearly outperforms the model without retraining throughout most of the process duration, especially midway through, where the most critical actions are taken for process control.

To compare the predictions of the model without retraining to the model with retraining, results of a representative run are shown in Figure 10. In Figure 10, the observed data is shown in green; the predictions for the 3 following days calculated on each day of operation (termed as “3-day model prediction”) without retraining are in blue and the 3-day model predictions with retraining are in orange. Consider the case of VCD on the top left. The model without retraining predicts an increase of VCD throughout the process duration, as this is the expected behavior of the clone that it was trained on. On the other hand, the model with training shows better predictions in the 3-day horizon. For VCD it soon learns that this clone grows less than the initial clone, given the current process conditions. Consistently for all variables, while the predictions of the two approaches are similar in the beginning of the process, as not a lot of new data is available yet, towards the end of the process they become more and more different.

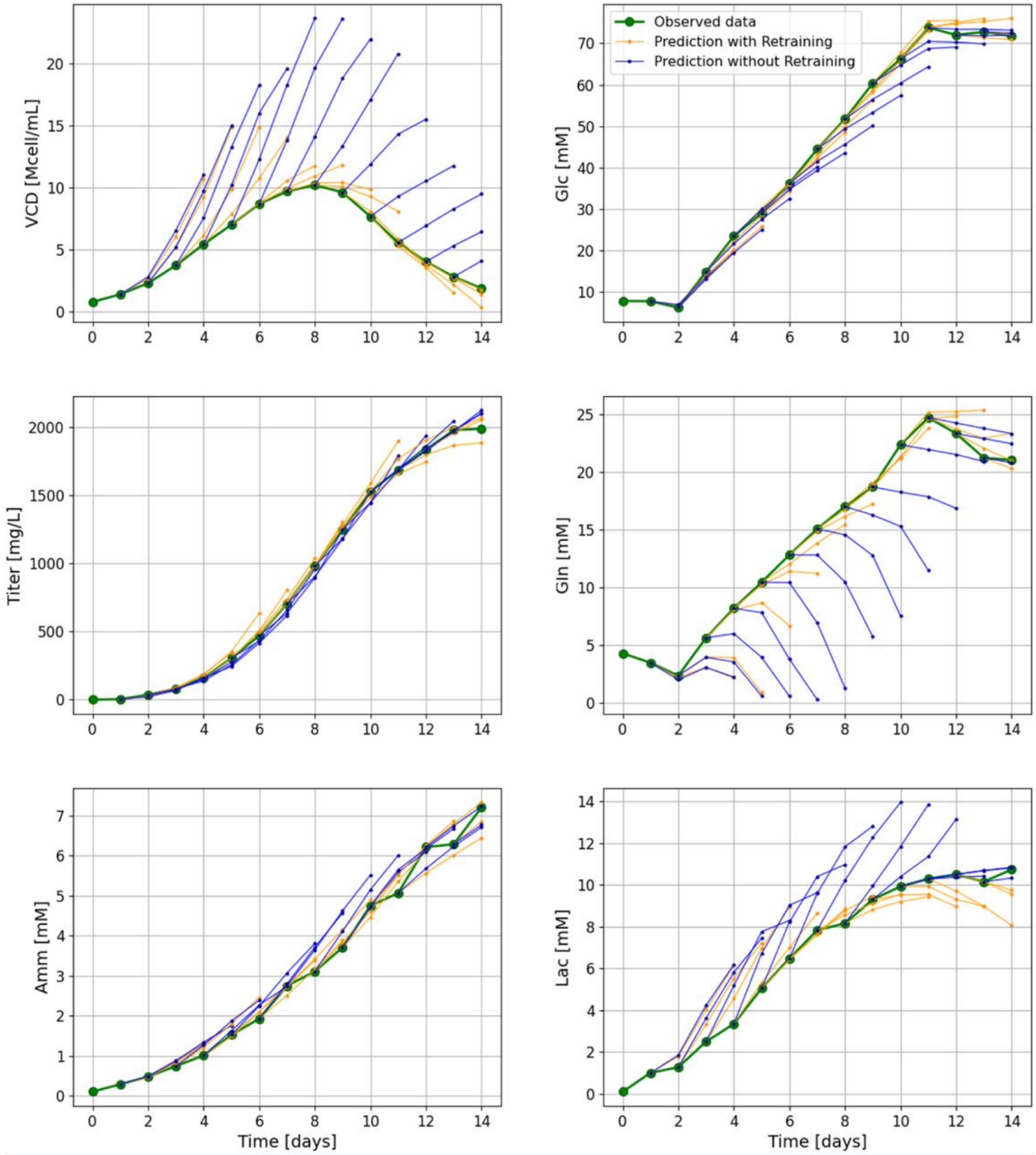


Figure 10. Observed vs. predicted time evolution profiles for selected process variables, comparing the predictions of the model with retraining (green) and the predictions of the model without retraining (blue) to the observed data (green). On each day, the model predicts three days ahead. As can be seen, the model that is continuously retrained learns the behavior of the new clone.

Looking at the data from average rRMSE for all variables (Figure 8) as well as the individual trends of rRMSE with time for all variables (Figure 9), we conclude that the model with retraining predicts better than model without retraining for predicting behavior of new clones. This trend is indeed exemplified for processes which run for long durations, which is typically the use case and purpose in perfusion processes.

4.2. Training experiment

In order to demonstrate the retraining capability, this concept was applied to actual laboratory data. Specific experiments were tailored to collect data, as described in Section 3.3 and were used to build a hybrid model as explained in Section 2.2. The model capability was assessed by training models in leave-one run out validation. The model errors were determined as described in Section 2.5.

Figure 11 shows the rRMSE obtained for the trained model against the validation data sets. The left bar plot shows the average rRMSE over all runs in testing over a 3-day horizon. For all variables the predictions have a rRMSE below 0.5 in the 3-day prediction horizon considered, making it suitable for control. On the right-hand side the rRMSE is displayed at the resolution of the runs for each variable. There are no runs (rows), where all variables are predicted with significantly higher rRMSE than the overall results, as can be seen by the intensity of the blue shadings. Therefore the heat map is indicating that the model can generalize well on the entire design space.

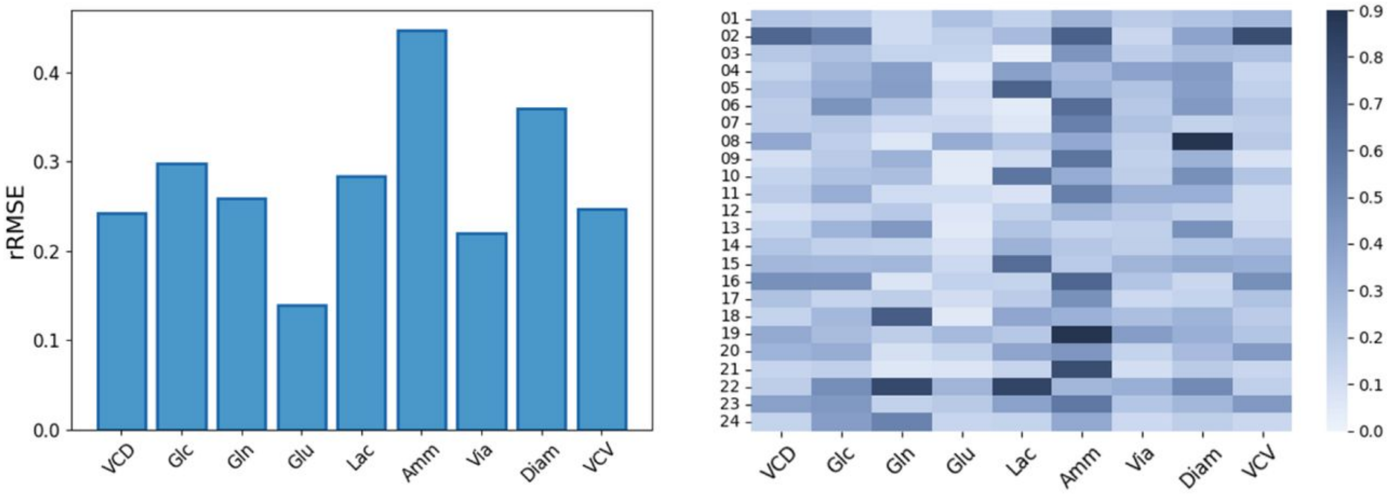
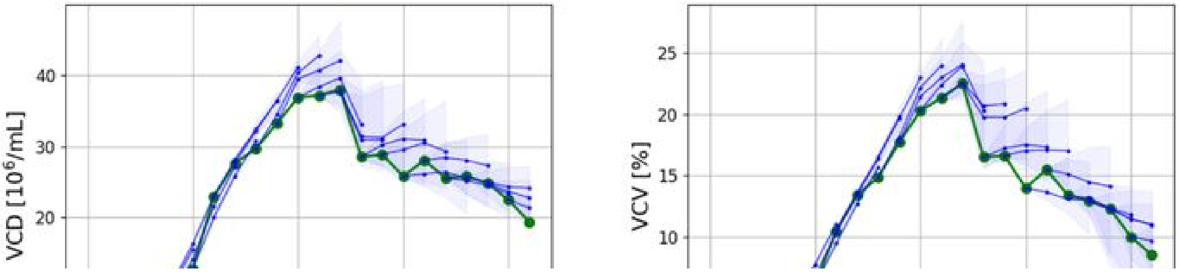
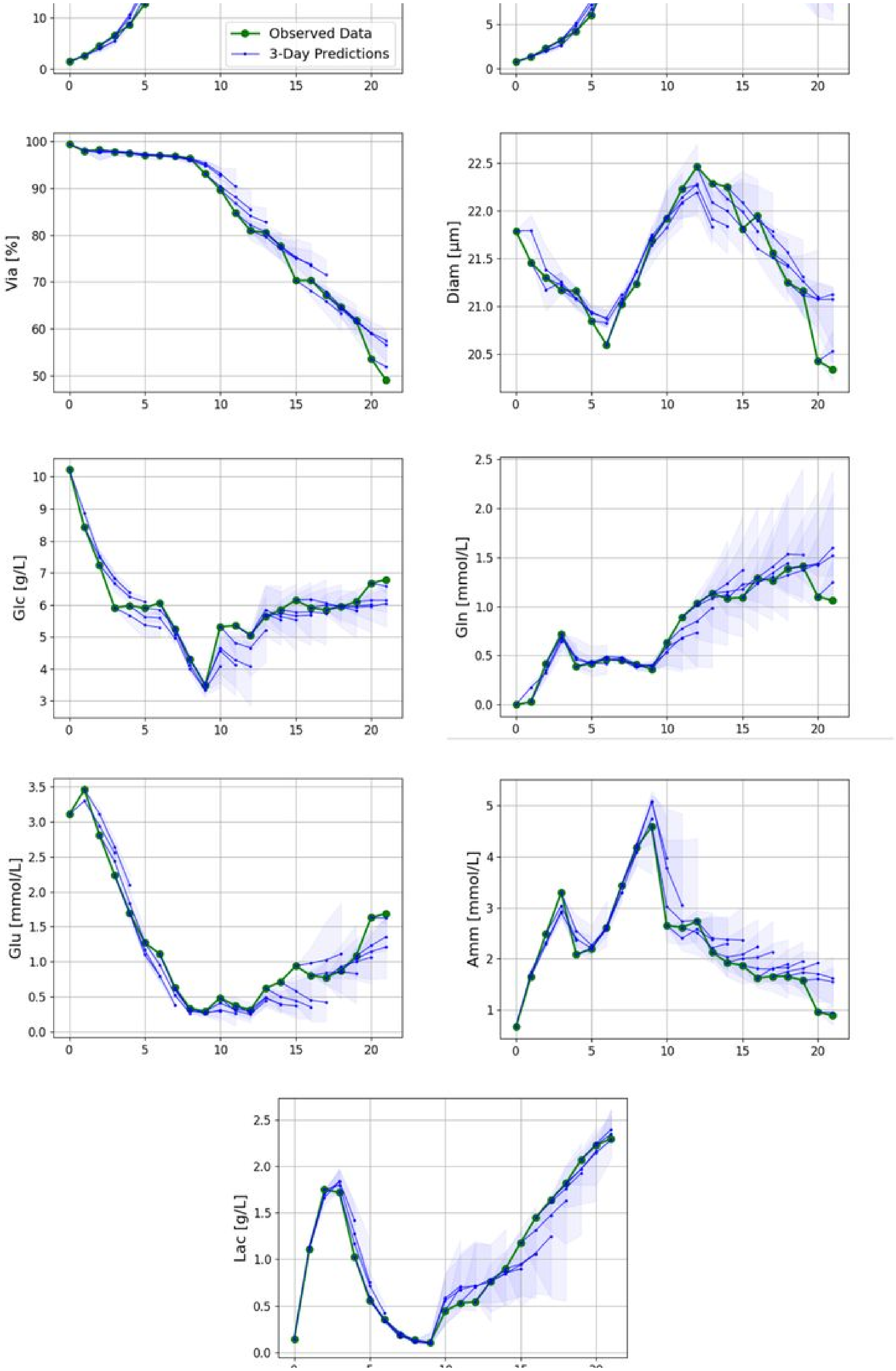


Figure 11. Left: rRMSE results averaged over all runs in testing (leave-one out) over a 3-day prediction horizon. The relative RMSE is well below 0.5 for all variables indicating good model performance. Right: rRMSE heat map showing the errors for each run (row) and variable (column).

In order to better visualize the model prediction capability, a representative run in terms of model performance (labeled as “01” in Figure 11, right-hand side) was picked for the purpose of illustration. Figure 12 shows the median model prediction and the prediction interval for the following 3 days on each day of operation. We can clearly see that the model predicts the trends of the process accurately and in general the observed process data is within the prediction interval of previous predictions.




Observed Data
3-Day Predictions
Via [%]
Diam [µm]
Glc [g/L]
Gln [mmol/L]
Glu [mmol/L]
Amm [mmol/L]
Lac [g/L]
0
5
10
15
20

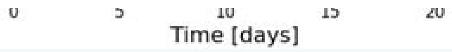


Figure 12. Predictions for a run in the Training experiment in testing. The model was trained on the remaining 23 runs. The green line shows the measured experimental data, the solid blue lines are the 3-day median predictions of the model, starting on each respective day. The shaded polygons represent the prediction intervals. In the evaluated 3-day horizon, the model predicts the process trends with sufficient accuracy for software operation.

The above results demonstrate that the initial model can learn the process behavior and can make meaningful predictions across different input process conditions.

4.3. Use case experiment

In order to leverage the online control of the bioreactors by software solution as well as it's optimization and retraining capability, three different use cases were selected to target process indicators such as VCV, Viability and ammonium. The parameters which were varied in the training experiments such as perfusion rate, temperature shift, agitation rate and pyruvate addition flowrate were selected as control variables. The software solution allows the user to select one target variable and any number of control variables.

Each day, the software solution would read the observed data from the robotic platform and perform an optimization to calculate the best set of conditions to reach the defined target. Following this step, the software solution would apply the optimized set points of the control variables to the robotic platform. The frequency at which the new measurements as well as optimization was performed can be defined by the user. For our use case, we selected the frequency to be once per day. The software solution can also be used in open loop control where the calculated set points need to be approved by a human before application to the robotic platform. In our use case experiment, the hybrid model developed is used as a prescriptive tool, where control actions are taken each day by the optimizer based on model predictions of the target.

4.3.1. VCV use case

The aim of this use case was to increase VCV to a higher value which was defined as a target value = 30%. To achieve this, all four control parameters were varied simultaneously every day as shown in Table 2, Condition 1.

Figure 13 shows the VCV evolution during the process for the bioreactor with software operation (green), 3 bioreactors of the same clone without software operation (blue). Perfusion rate (Perfusion) is the control variable that has the biggest impact on the target and is shown in light red in the large plot. The smaller subfigures at the top of the figure show the set points suggested by the optimizer for the other control variables, namely temperature shift (Temp), agitation rate (Stir) and pyruvate addition flowrate (Pyr) respectively.

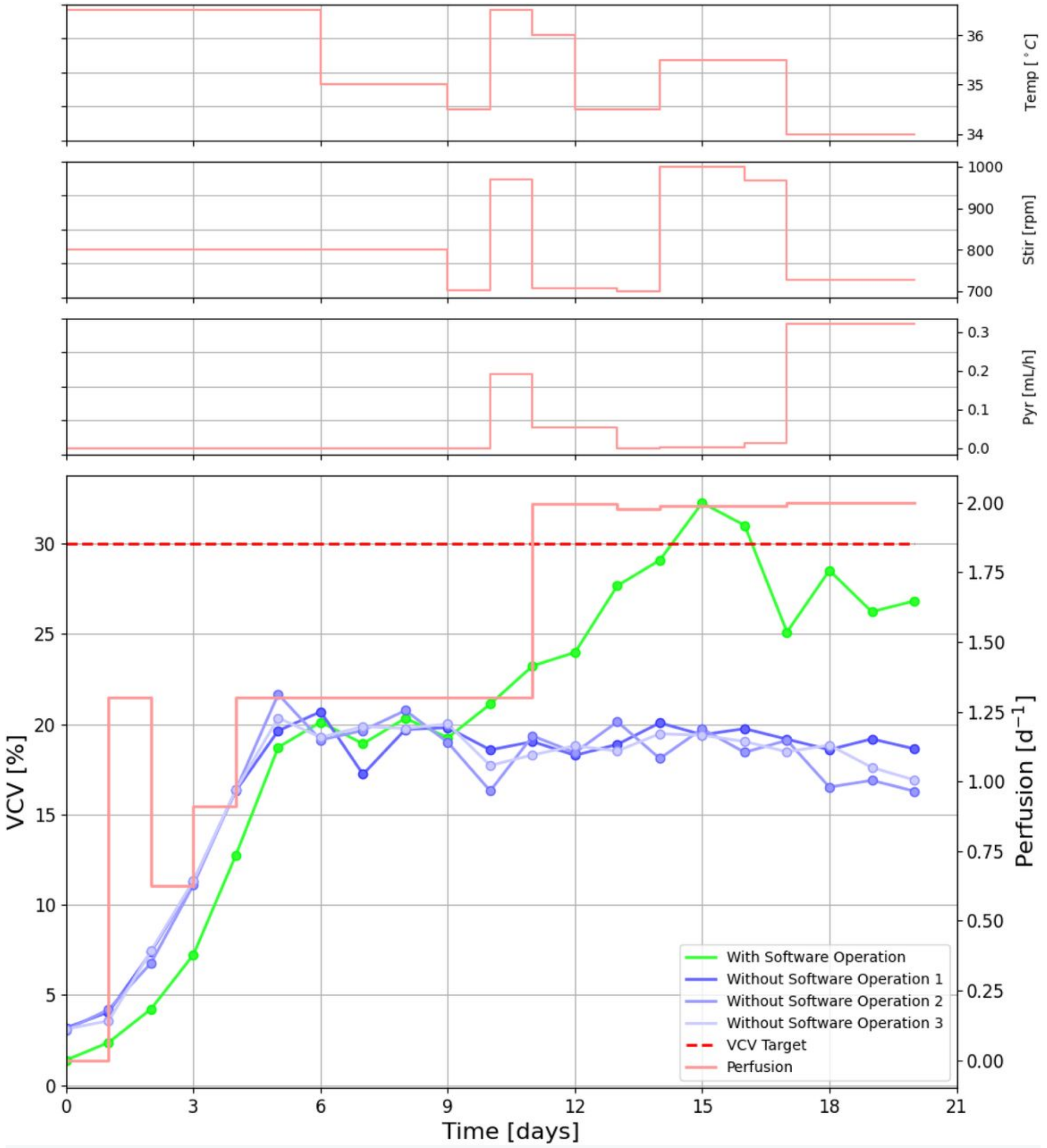


Figure 13. Condition 1 results: VCV use case with VCV Target = 30% (red dashed line). The observed VCV values of the bioreactor with software operation are shown in green, the observed values of three bioreactors without software operation are shown in blue. The optimizer suggested perfusion rate is plotted on the secondary axis. All control parameter profiles are shown in light red.

The optimizer was making suggestions from day 9 onwards. For this use case the VCV target value of 30% is significantly higher than the value of VCV in standard experiments (average VCV after state of control < 20%). The software was able to suggest conditions that significantly increased VCV reaching the target in contrast to the benchmark bioreactors.

The software solution identified perfusion rate as the most significant factor for controlling VCV. Higher perfusion rate will lead to faster replenishment of the perfusion media, thereby allowing faster cell growth. This is a well-known phenomenon among the process scientists, which was inherently learned by the hybrid model using the training experiment data as well as the constant retraining of the model.

4.3.2. Viability use case

The aim of this use case was to increase the cell viability by varying all the control parameters as shown in Table 2, Condition 4. Hence a high target value for cell viability = 98% was selected. This use case was tested on clone B, which was not part of the Training experiment. Figure 14 shows the cell viability evolution during the process for the bioreactor operated by the software (green), 3 bioreactors of the same clone without software operation (blue). In this use case, temperature is identified as the control variable which had the highest impact on the target and the set points suggested by the software solution are shown in red in the large plot. The daily optimised set points for the other control variables are plotted at the top of Figure 14.

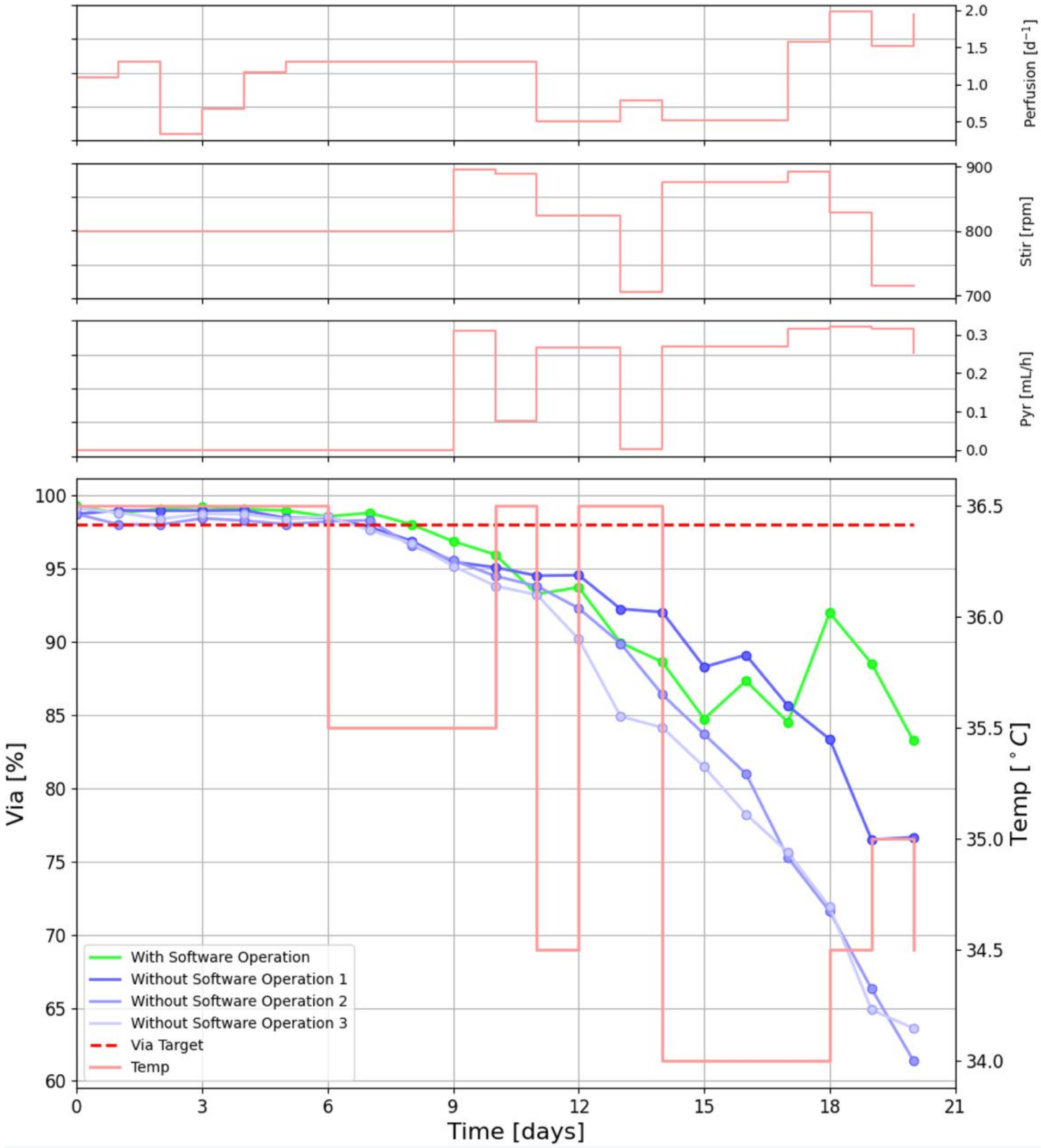


Figure 14. Condition 4 results: viability (Via) use case with Via Target = 98% (red dashed line). The observed viability values of the bioreactor with software operation are shown in green, three bioreactors without software operation are shown in blue. The optimizer suggested temperature rate is plotted on the secondary axis. All control parameter profiles are shown in light red.

Similar to the previous use case, the online control of the software solution was started after day 9. The variations in the optimised set points, especially temperature and VVD were quite large until day 14 in comparison to the VCV use case. However, the response of the bioreactor to these changes were used to train the model. The retraining of the model with the new observed data led to an improvement in prediction of cell viability which in turn aids the optimisation of control variables. This can be inferred from the increase in cell viability observed after day 15. The software solution suggested lower temperature set points as well as higher perfusion rate towards the end of the process. These conditions led to higher cell viability in the bioreactor in comparison to the bioreactors without software control.

The better performance of the bioreactor with software operation in comparison to those without towards the end of the process could be attributed to two reasons. The first is that there might be little room to enhance viability in the beginning of the perfusion operation compared to the end, as there could be intrinsic process limitations. The second reason could be that the training experimental data as well as the measured data until day 8 were not sufficient to accurately predict the cell viability. We can see a similar lag in prediction accuracy when the hybrid model trained on one clone is applied to a new clone in Section 4.1. However, the prediction accuracy improved towards the end of the run by retraining the model with more number of measured data points leading to an increase in cell viability as seen in Figure 14.

The fact that the target bioreactor only started to outperform the control bioreactors at the end of the process may have two reasons. The first is that there might be little room to enhance viability in the beginning of the perfusion operation compared to the end, as there could be intrinsic process limitations. The second is that the ability of the model to understand the impact of the process parameters on viability for the new clone at day 9 might not have been sufficient. Only after retraining with the data of the first half of the validation run, the software solution was able to learn these relationships and re-optimize the set points accordingly.

4.3.3. Ammonium use case

It has been suggested in literature that sodium pyruvate addition in a perfusion cell culture can be used to stabilize ammonium concentration[50][51]. Pyruvate is the metabolic entry point for the Krebs cycle. If the gylcolysis pathway is saturated and pyruvate becomes limiting, cell will use other sources (amino acids) to feed this cycle and this can result in ammonium accumulation. Hence, the aim of this use case is to try and control ammonium concentration to a value = 1.0 mM by using pyruvate addition as the only control parameter. This is illustrated in Table 2 (Condition 2).

Figure 15 shows the ammonium evolution during the process for the bioreactor with software operation (green), 3 bioreactors of same clone without software operation (blue) and model suggested pyruvate addition set points (red). We can see an increase in the ammonium concentration up to 4 mM until day 9 when there is no addition of pyruvate feed. The optimization results suggested the addition of pyruvate to reach the ammonium target, which led to a reduction in the ammonium value until day 13.

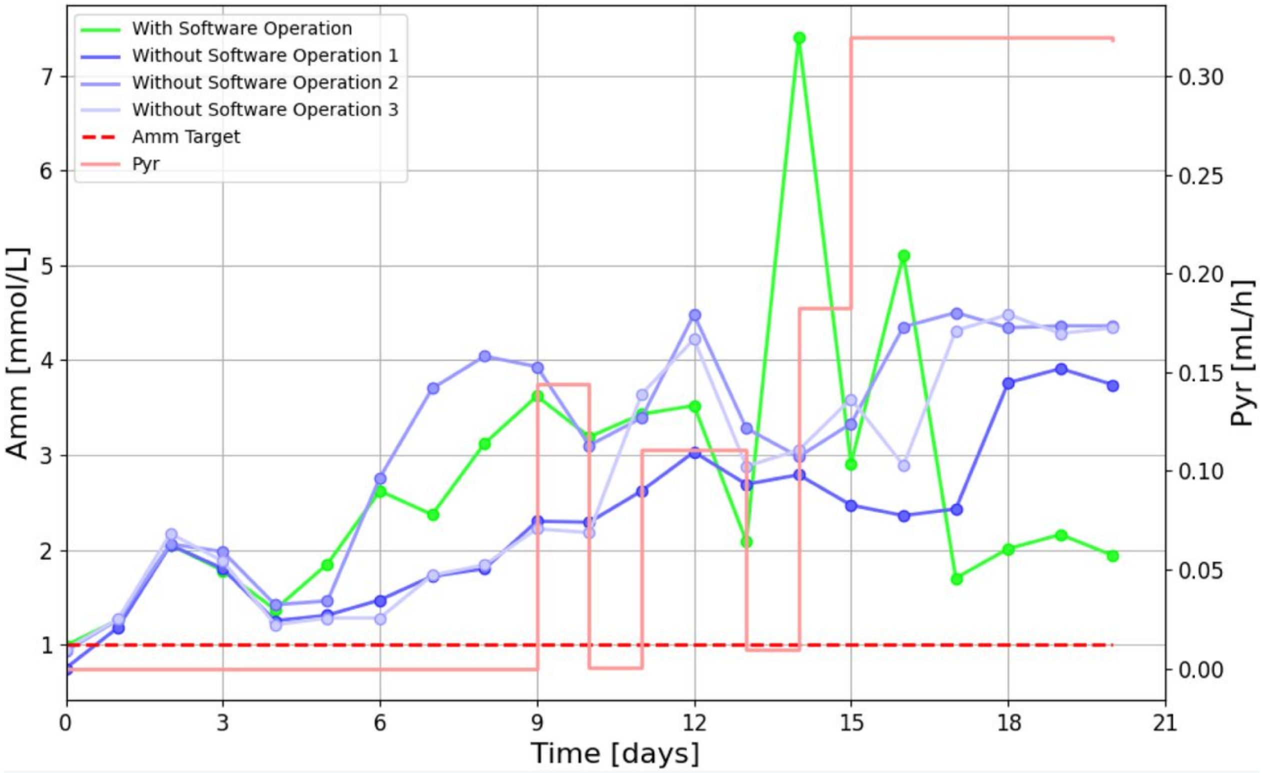


Figure 15. Condition 2: ammonium (Amm) use case with Amm target= 1 mM (red dashed line). The observed ammonium values of the bioreactor with software operation are shown in green, three bioreactors without software operation are shown in blue. The optimizer suggested pyruvate additions are plotted on secondary axis and is shown in light red.

Manual injection of ammonium was performed on day 13 to confirm the response of the model to high values of ammonium in bioreactor, leading to the observed peak. Once the concentration of ammonium increased above 7 mM, the software responded by increasing the pyruvate addition set point, as it had learned that the addition of pyruvate decreases ammonium.

These different use cases demonstrate that a training experiment which covers the expected variability for given process parameters will help the hybrid model to learn how control variables impact process indicators. This knowledge from the model can then be used to operate the process once the desired state of control is reached. Deviations from the state of control can be predicted and avoided with a correct response from the system. This strategy is applicable for process development but could also be useful in manufacturing. Debates on how to validate such models for GMP applications should help the biomanufacturing community to exploit such capabilities to their full potential.

Furthermore, the retraining of the model is an important feature which can be applied to new clones as well as new molecules in the industry pipeline. There is a lag in finding the optimum conditions for new clones as the initial data measured is used to learn the relationships between control variable and process indicator. This capability of retraining will be crucial in the initial process development stage of a new molecule. The final optimized process parameter from this run can be used to lock the set points of the process parameters.

5. Discussion

Novel robotic systems for HT experimentation at every lab scale has drastically changed biotechnology laboratories. The large number of runs that can operate in parallel, the highly complex tasks that robots can perform, the associated process analytical tools and the addition of mobile assistants that overcome spatial limitations, are just some of the many advantages. On the other hand, this significantly increases the number of high-tech devices that rely on skilled personnel and on a robust digital infrastructure to enable communication and control of all devices. To operate these robotic platforms, several new tools entered the biotech lab environment. Some examples are data management tools, IT systems, optimization algorithms (for design or scheduling), monitoring and control tools.

Still, the synergy between novel computational and robotic tools can only be fully exploited based on a tight integration of methods, devices, and expertise. Bioprocess development lies behind in terms of development and availability of specific tools for this purpose. The strategic integration of appropriate tools has a large potential to revolutionize biotechnology laboratories. As demonstrated by the findings presented in this study, the combination of computational tools with existing robotic experimental setups can drastically increase the throughput and efficiency of laboratories. This has already been demonstrated in other fields[52], overtaking tasks with ever increasing complexity gives researchers freedom to engage in challenging and creative activities[38][21]. In the above given example, automated experimental design tackles a major bottleneck, as is the preparation and design of experimental runs based on existing data, a resource and personnel intensive task. Moreover, the ability to continuously learn and adapt the experimental design during cultivation allows for real-time improvements as insights into the process are acquired. Especially for development of perfusion experiments that are expected to run for months, a repeated improvement of the experimental design is pivotal to maximize efficient use of time and resources.

The use case experiment presented in this work shows that the system developed has five capabilities required in self-driving laboratories for bioprocess development, namely:

1. **Hybrid modelling formulation for flexible and robust process:**A robust model backbone that allows a proper description of the time evolution of performance even for new clones. The implemented hybrid model based on SW-GP could describe the process properly as reported in literature[44].
2. **Online model re-training to enable real-time learning:**As described in detail with insilico data, re-training is essential to empower the algorithm with “learning” capabilities. This feature exploits the fact that in parallel experiments the information generated by neighboring bioreactors is very useful for the running system.
3. **A toolset to transfer learnings from past projects:**The software was able to describe perfusion runs with a new clone (Clone B), for which no previous data was available. This is possible since the GP Kernel can extract the existing similarities between the new clone and the known one. By this the existing information can be used to drastically reduce the number of experiments required in development.
4. **Predictive process monitoring and notifications:**Even with models that show a poor description power (say due to radically new processes or unknown issues in the system), the forecast and notification of probable future events is a major advantage for a robust operation of the experimental setup. Despite the significant autonomy of the robotic

system, auxiliary resources and external tasks (liquid containers, at-line analytics, sample handling for off-line analysis) still need to be performed by operators. In addition, unexpected malfunction in the bioreactors or liquid handlers need to be tackled by experts. Notifications that inform of an increasing probability that undesired events will take place (low dissolved oxygen, low glucose concentration, deviation on the pH set-point) allow acting on these events before they occur, enabling planning of resources and personnel.

5. **Autonomous feed-back experimental operation minimizing the human in the loop:**One of the key advantages of robotics systems is operation 24/7. Processes that run in tightly defined conditions use process control to ensure robust operation, yet in development, neither a good process understanding (typically in formulated as a mechanistic mathematical model), nor a defined operating regime (as in manufacturing) are available. The software agent in charge of the operation of the parallel cultivation system is confronted with the challenging tasks to learn the new process behavior and simultaneously take actions that drive it to the desired targets[53]. Furthermore, some targets set in the use case (e.g. cell viability) do not have a known input-output relationship, which must be learned during experimentation. It is hence interesting to confirm that the system was indeed able to find the process conditions that drive the process to the desired objective in all cases.

The above described capabilities make this software solution a viable asset for process development, where high throughput systems are used for optimization of process conditions. However, the application of this technology can be extended to technology transfer for a large scale bioreactor. As proof of concept, the prototype software solution developed for laboratory scale in this study shows the potential of the autonomously operated bioreactor to reach and maintain the defined target by learning from the small scale data as well as historical large scale data used to build the model. In the context of process development, steps to define set points for the optimized process based on the software solution suggestions during the experiment will be further studied.

An open challenge remains modelling and feedback operation considering product quality. Currently, the main CQAs can only be measured offline at the specific department for analytics, leading to a significant time gap between the sampling and data availability. Process Analytical Technology (PAT) tools that enable online or atline quantification of glycoforms, HCPs, aggregates, fragments will disrupt bioprocess development and operation, since beyond productivity and yield, stable quality attributes are paramount. Being able to predict how they vary using model-based tools could also one day lead to an active control over these CQAs and truly optimal operation.

6. Conclusion

The introduction of autonomous experimental facilities (self-driving laboratories) in the pipeline of biopharma development is an essential step to accelerate the long and uncertain path for product to patient. The software solution presented here tackles important challenges by exploiting the potential of advanced parallel mini-bioreactor experimental systems. We demonstrate the capabilities of the developed software with three use cases, showing the added value of the implementation of machine learning tools in modern experimental systems. Furthermore, the user friendliness of the software and support offered to operators and scientists make this software an important step towards human centric industry 5.0.

We expect the results to motivate further development in this area and a larger acceptance in the industry. The current gap between robotic capabilities and the autonomy of the devices needs to be urgently addressed in bioprocess development for the biotechnological and biopharmaceutical industries to match the high expectations set to digital biotechnology and bioindustry 5.0.

Appendix

1. Use case experiment case studies

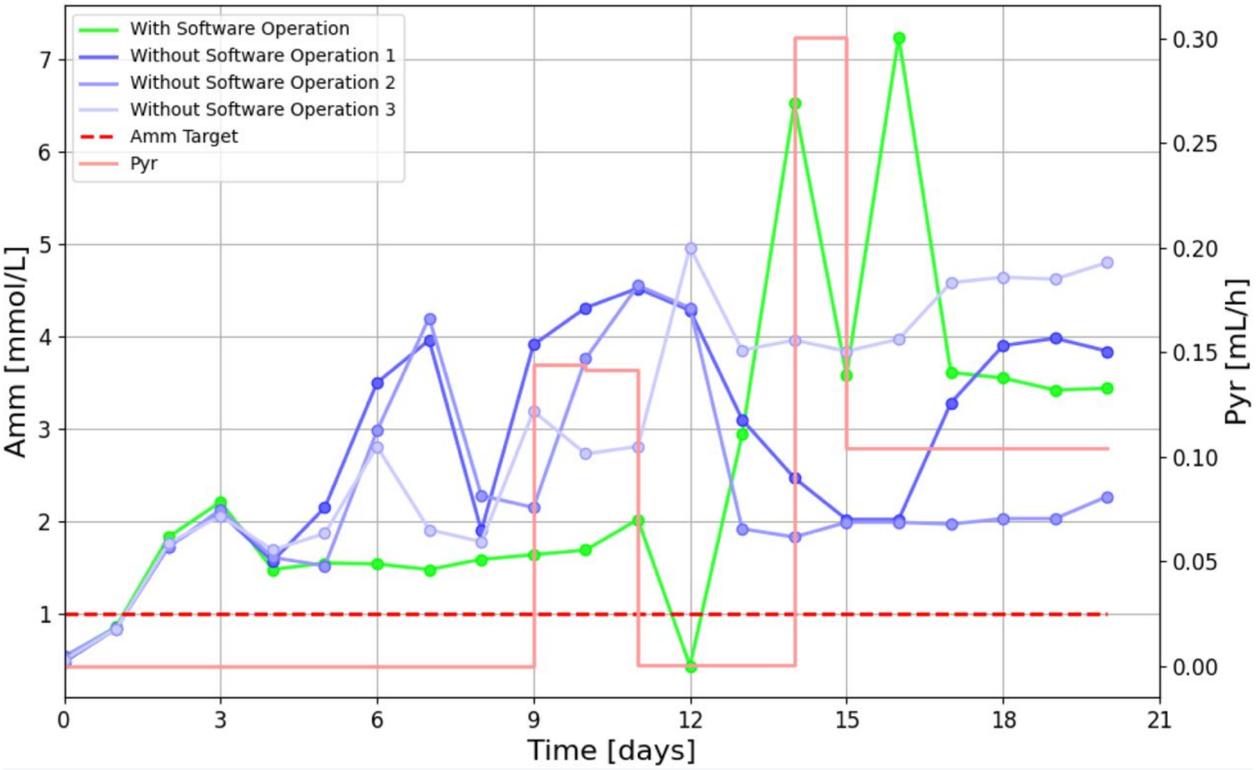


Figure 16. Condition 3 results: ammonium (Amm) use case with Amm target= 1 mM (red dashed line). The observed ammonium values of the bioreactor with software operation are shown in green, three bioreactors without software operation are shown in blue. The optimizer suggested pyruvate additions are plotted on secondary axis and is shown in light red. The model was originally not trained on data of this clone.

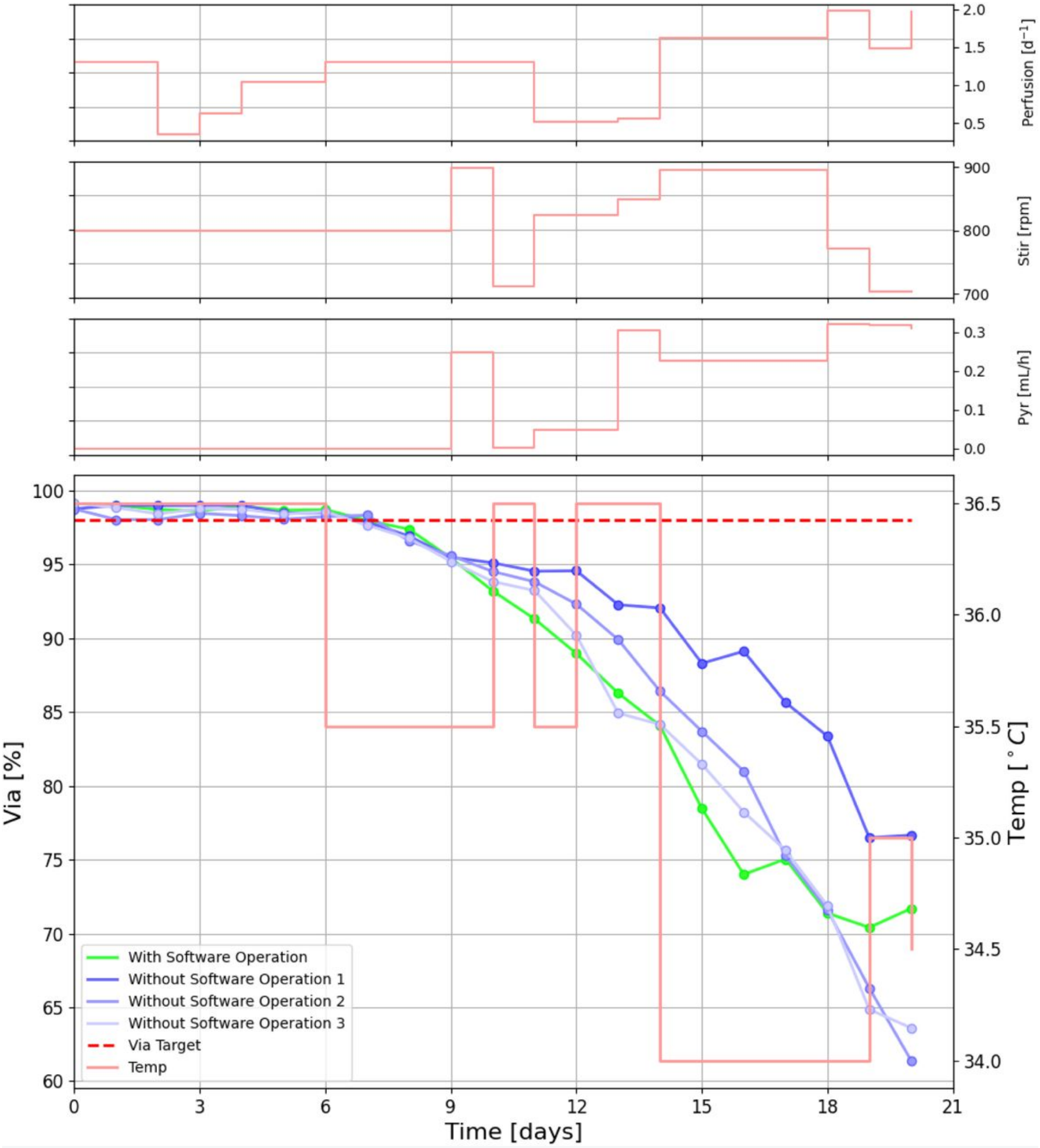


Figure 17. Condition 5 results: viability (Via) use case with Via Target = 98% (red dashed line). The observed viability values of the bioreactor with software operation are shown in green, three bioreactors without software operation are shown in blue. The optimizer suggested temperature rate is plotted on the secondary axis. All control parameter profiles are shown in light red.

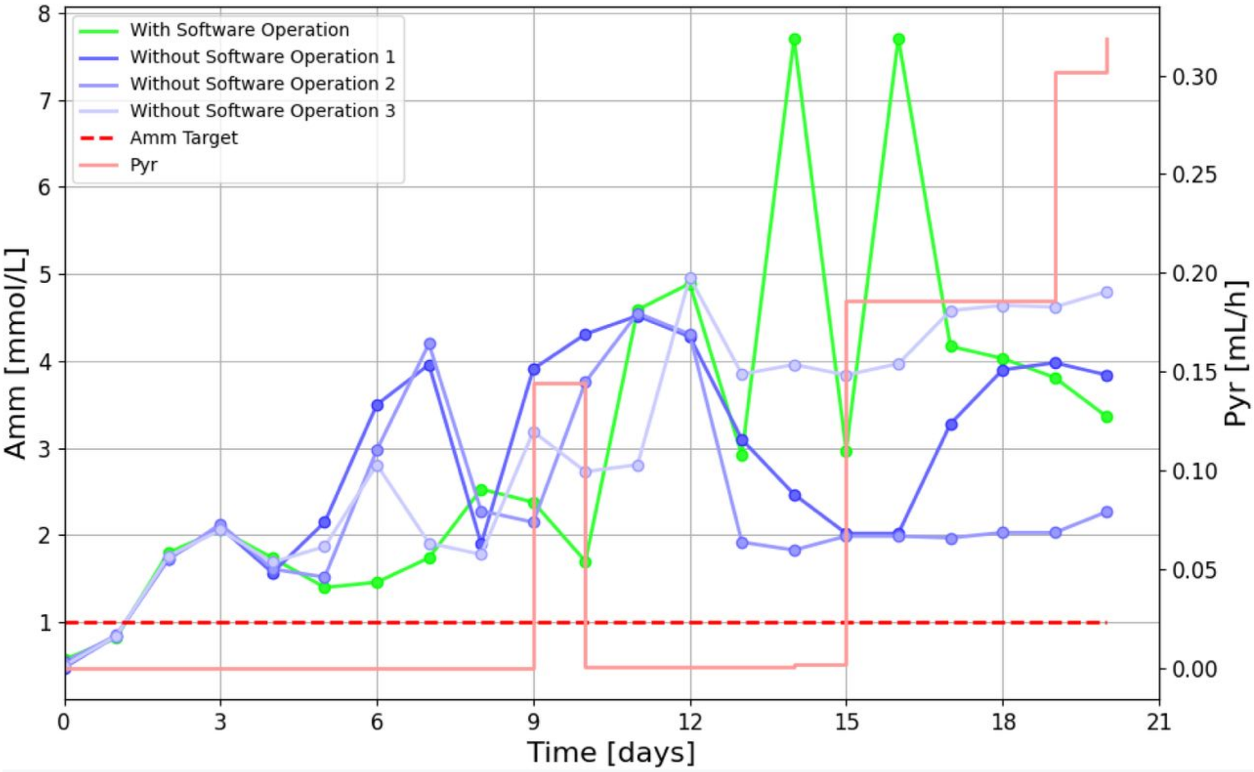


Figure 18. Condition 6 results: ammonium (Amm) use case with Amm target= 1 mM (red dashed line). The observed ammonium values of the bioreactor with software operation are shown in green, three bioreactors without software operation are shown in blue. The optimizer suggested pyruvate additions are plotted on secondary axis and is shown in light red. The model was originally not trained on data of this clone.

Acknowledgments

We would like to thank the entire USP process design team (Biotech Development Center, Switzerland) for the support with the design and execution of the experiments in the lab. Specific thanks go to Raphaël Guillot and Anaïs Roulet for the feedbacks and interactions with the Datahow team. We also want to thank the Digital team, especially Alain Gillieron and Alexandre Gilet for their continuous support and agility. In addition, we do not forget our former colleagues Nicolas-Julian Hilbold, Stefania Caso and David Brühlmann who largely contributed to set the stage and kick-off this collaboration.

We also want to thank Daniel Stoeckel and Sonja Hatz (Merck KGaA, Darmstadt, Germany) for support with strategic decision and project management and the Merck 350 Research Grants funding for financing this innovation effort.

The discussion with the Digital Twin Team at Datahow significantly improved the software solution, specifically front end. We want to highlight Giulia Cantini for the implementation of the user interface.

We also thank Antonios Papaemmanouil and Thierry Prud’homme from Hochschule Luzern – Technik & Architektur for their support in the interface communication and control with the ambr software.

The writing of this manuscript was assisted by grammar checker and language correction AI tools specifically Trinka, Claude 3 Haiku, and GPT-3.5 Turbo. All authors have read, corrected and verified all information presented in this

manuscript and Supplementary Information.

References

1. ^*Rouiller Y, Solacroup T, Deparis V, Barbafieri M, Gleixner R, Broly H, Eon-Duval A. 2012. "Application of Quality by Design to the characterization of the cell culture process of an Fc-Fusion protein". European Journal of Pharmaceutics and Biopharmaceutics 81:426–437.*
2. ^*Saleh D, Wang G, Rischawy F, Kluters S, Studts J, Hubbuch J. 2021. "In silico process characterization for biopharmaceutical development following the quality by design concept". Biotechnology Progress 37:e3196.*
3. ^*Baumann P, Hahn T, Hubbuch J (2015). "High-throughput micro-scale cultivations and chromatography modeling: Powerful tools for integrated process development". Biotechnology and Bioengineering. 112: 2123–2133.*
4. ^*Rienzo M, Lin K-C, C. Mobilia K, K. Sackmann E, Kurz V, H. Navidi A, King J, M. Onorato R, K. Chao L, Wu T, Jiang H, K. Valley J, A. Lionberger T, D. Leavell M. 2021. "High-throughput optofluidic screening for improved microbial cell factories via real-time micron-scale productivity monitoring". Lab on a Chip 21:2901–2912.*
5. ^*Schwarz H, Lee K, Castan A, Chotteau V. 2023. "Optimization of medium with perfusion microbioreactors for high density CHO cell cultures at very low renewal rate aided by design of experiments". Biotechnology and Bioengineering 120:2523–2541.*
6. ^*Carracedo-Reboredo P, Liñares-Blanco J, Rodríguez-Fernández N, Cedrón F, Novoa FJ, Carballal A, Maojo V, Pazos A, Fernandez-Lozano C (2021). "A review on machine learning approaches and trends in drug discovery". Computational and Structural Biotechnology Journal. 19: 4538–4558.*
7. ^*Dara S, Dhamercherla S, Jadav SS, Babu CM, Ahsan MJ (2022). "Machine Learning in Drug Discovery: A Review". Artif Intell Rev. 55: 1947–1999.*
8. ^*Vamathevan J, Clark D, Czodrowski P, Dunham I, Ferran E, Lee G, Li B, Madabhushi A, Shah P, Spitzer M, Zhao S. 2019. "Applications of machine learning in drug discovery and development". Nat Rev Drug Discov 18:463–477.*
9. ^*O'Flaherty R, Bergin A, Flampouri E, Mota LM, Obaidi I, Quigley A, Xie Y, Butler M. 2020. "Mammalian cell culture for production of recombinant proteins: A review of the critical steps in their biomanufacturing". Biotechnology Advances 43:107552.*
10. ^*Pogodaev A, Hernández Rodríguez T, Li M, García Münzer DG. 2024. "Modeling of bioprocess pre-stages for optimization of perfusion profiles and increased process understanding". Biotech & Bioengineering 121:228–237.*
11. ^*Fisher AC, Kamga M-H, Agarabi C, Brorson K, Lee SL, Yoon S (2019). "The Current Scientific and Regulatory Landscape in Advancing Integrated Continuous Biopharmaceutical Manufacturing". Trends in Biotechnology. 37: 253–267.*
12. ^*Khuat TT, Bassett R, Otte E, Grevis-James A, Gabrys B (2024). "Applications of machine learning in antibody discovery, process development, manufacturing and formulation: Current trends, challenges, and opportunities". Computers & Chemical Engineering. 182: 108585.*
13. ^*Anane E, Haby B, Hans S, Glauche F, Neubauer P, Cruz Bournazou MN (2018). "Scaling down further: Model-based scale-down studies in minibioreactors". In: New Biotechnology. Elsevier, Vol. 44, pp. S60–S61.*

14. ^*Sandner V, Pybus LP, McCreath G, Glassey J. 2019. "Scale-Down Model Development in ambr systems: An Industrial Perspective". Biotechnology Journal 14:1700766.*
15. ^*Karst DJ, Steinebach F, Morbidelli M (2018). "Continuous integrated manufacturing of therapeutic proteins". Current Opinion in Biotechnology. 53. Chemical Biotechnology Pharmaceutical Biotechnology: 76–84.*
16. ^*Jang K-S, Kim Y-G, Gil G-C, Park S-H, Kim B-G (2009). "Mass spectrometric quantification of neutral and sialylated N-glycans from a recombinant therapeutic glycoprotein produced in the two Chinese hamster ovary cell lines". Analytical Biochemistry. 386: 228–236.*
17. ^*Harrer S, Menard J, Rivers M, Green DVS, Karpiak J, Jeliazkov JR, Shapovalov MV, del Alamo D, Sternke MC (2024). "Chapter 40 - Artificial intelligence drives the digital transformation of pharma". In: Krittanawong, C, editor. Artificial Intelligence in Clinical Practice. Academic Press, pp. 345–372. https://www.sciencedirect.com/science/article/pii/B9780443156885000498.*
18. ^*Aspuru-Guzik A (2022). "A forward view for Digital Discovery: the scientific challenges of the twenty-first century require accelerated discovery approaches". Digital Discovery. 1: 6–7.*
19. ^*Kramer S, Cerrato M, Džeroski S, King RD. 2023. "Automated Scientific Discovery: From Equation Discovery to Autonomous Discovery Systems".*
20. ^*Abolhasani M, Kumacheva E (2023). "The rise of self-driving labs in chemical and materials sciences". Nat. Synth. 2: 483–492.*
21. a, b*Burger B, Maffettone PM, Gusev VV, Aitchison CM, Bai Y, Wang X, Li X, Alston BM, Li B, Clowes R, Rankin N, Harris B, Sprick RS, Cooper AI (2020). "A mobile robotic chemist". Nature. 583: 237–241.*
22. a, b*Duong-Trung N, Born S, Kim JW, Schermeyer M-T, Paulick K, Borisyak M, Cruz-Bournazou MN, Werner T, Scholz R, Schmidt-Thieme L, Neubauer P, Martinez E (2023). "When bioprocess engineering meets machine learning: A survey from the perspective of automated bioprocess development". Biochemical Engineering Journal. 190: 108764.*
23. ^*Boiko DA, MacKnight R, Kline B, Gomes G (2023). "Autonomous chemical research with large language models". Nature. 624: 570–578.*
24. ^*Haringa C, Tang W, Wang G, Deshmukh AT, Winden WA van, Chu J, Gulik WM van, Heijnen JJ, Mudde RF, Noorman HJ (2018). "Computational fluid dynamics simulation of an industrial P. chrysogenum fermentation with a coupled 9-pool metabolic model: Towards rational scale-down and design optimization". Chemical Engineering Science. 175: 12–24.*
25. ^*Villiger TK, Neunstoecklin B, Karst DJ, Lucas E, Stettler M, Broly H, Morbidelli M, Soos M. 2018. "Experimental and CFD physical characterization of animal cell bioreactors: From micro- to production scale". Biochemical Engineering Journal 131:84–94.*
26. ^*Chopda V, Gyorgypal A, Yang O, Singh R, Ramachandran R, Zhang H, Tsilomelekis G, Chundawat SPS, Ierapetritou MG (2022). "Recent advances in integrated process analytical techniques, modeling, and control strategies to enable continuous biomanufacturing of monoclonal antibodies". Journal of Chemical Technology & Biotechnology. 97: 2317–2335.*
27. a, b*Narayanan H, Luna MF, von Stosch M, Cruz Bournazou MN, Polotti G, Morbidelli M, Butté A, Sokolov M. 2020. "Bioprocessing in the Digital Age: The Role of Process Models". Biotechnology Journal 15:1900172.*

28. ^Lu J, Yang Z, Zheng X, Wang J, Kiritsis D. 2022. "Exploring the Concept of Cognitive Digital Twins from Model-Based Systems Engineering Perspective". Preprint. In Review. https://www.researchsquare.com/article/rs-1431416/v1.
29. ^Cai C, Wang S, Xu Y, Zhang W, Tang K, Ouyang Q, Lai L, Pei J (2020). "Transfer Learning for Drug Discovery". J. Med. Chem.. 63: 8683–8694.
30. ^Kim JW, Krausch N, Aizpuru J, Barz T, Lucia S, Neubauer P, Bournazou MNC (2023). "Model predictive control and moving horizon estimation for adaptive optimal bolus feeding in high-throughput cultivation of E. coli". Computers & Chemical Engineering. 172: 108158.
31. ^Franceschini G, Macchietto S (2008). "Model-based design of experiments for parameter precision: State of the art". Chemical Engineering Science. 63: 4846–4872.
32. ^Rainforth T, Foster A, Ivanova DR, Smith FB. 2023. "Modern Bayesian Experimental Design". arXiv. http://arxiv.org/abs/2302.14545.
33. ^González LD, Zavala VM (2022). "New Paradigms for Exploiting Parallel Experiments in Bayesian Optimization". arXiv. http://arxiv.org/abs/2210.01071.
34. ^Cruz Bournazou MN, Barz T, Nickel DB, Lopez Cárdenas DC, Glauche F, Knepper A, Neubauer P (2017). "Online optimal experimental re-design in robotic parallel fed-batch cultivation facilities". Biotechnology and Bioengineering. 114: 610–619.
35. ^Helleckes LM, Hemmerich J, Wiechert W, Von Lieres E, Grünberger A (2023). "Machine learning in bioprocess development: from promise to practice". Trends in Biotechnology. 41: 817–835.
36. ^Krausch N, Kim JW, Barz T, Lucia S, Groß S, Huber MC, Schiller SM, Neubauer P, Cruz Bournazou MN. 2022. "High-throughput screening of optimal process conditions using model predictive control". Biotechnology and Bioengineering 119:3584–3595.
37. a, bHutter C, Stosch M von, Bournazou MNC, Butté A (2021). "Knowledge transfer across cell lines using hybrid Gaussian process models with entity embedding vectors". Biotechnology and Bioengineering. 118: 4389–4401.
38. a, bBai J, Cao L, Mosbach S, Akroyd J, Lapkin AA, Kraft M (2022). "From Platform to Knowledge Graph: Evolution of Laboratory Automation". JACS Au. 2: 292–309.
39. a, bMione FM, Kaspersetz L, Luna MF, Aizpuru J, Scholz R, Borisyak M, Kemmer A, Schermeyer MT, Martinez EC, Neubauer P, Cruz Bournazou MN. 2024. "A workflow management system for reproducible and interoperable high-throughput self-driving experiments". Computers & Chemical Engineering:108720.
40. ^González-Hernández Y, Perré P (2024). "Building blocks needed for mechanistic modeling of bioprocesses: A critical review based on protein production by CHO cells". Metabolic Engineering Communications. 18: e00232.
41. ^Cardillo AG, Castellanos MM, Desailly B, Dessoy S, Mariti M, Portela RMC, Scutella B, Stosch M von, Tomba E, Varsakelis C (2021). "Towards in silico Process Modeling for Vaccines". Trends in Biotechnology. 39: 1120–1130.
42. ^Narayanan H, Sokolov M, Morbidelli M, Butté A. 2019. "A new generation of predictive models–the added value of hybrid models for manufacturing processes of therapeutic proteins". Biotechnology and Bioengineering.
43. ^Azevedo CR, Díaz VG, Prado-Rubio OA, Willis MJ, Préat V, Oliveira R, Stosch M (2019). "Hybrid Semiparametric Modeling: A Modular Process Systems Engineering Approach for the Integration of Available Knowledge Sources". In: Systems Engineering in the Fourth Industrial Revolution. Wiley, pp. 345–373.

44. a, b *Mahanty B. 2023. "Hybrid modeling in bioprocess dynamics: Structural variabilities, implementation strategies, and practical challenges". Biotech & Bioengineering 120:2072–2091.*
45. ^ *Kocijan J, Murray-Smith R, Rasmussen CE, Girard A. 2004. "Gaussian process model based predictive control". Proceedings of the 2004 American Control Conference. Boston, MA, USA: IEEE, pp. 2214–2219 vol.3. https://ieeexplore.ieee.org/document/1383790/.*
46. ^ *Umlauft J, Lederer A, Hirche S. 2017. "Learning stable Gaussian process state space models". In:. 2017 American Control Conference (ACC). Seattle, WA, USA: IEEE, pp. 1499–1504. https://ieeexplore.ieee.org/document/7963165/.*
47. ^ *Cruz Bournazou MN, Narayanan H, Fagnani A, Butté A (2022). "Hybrid Gaussian Process Models for continuous time series in bolus fed-batch cultures". IFAC-PapersOnLine. 55. 13th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems DYCOPS 2022: 204–209.*
48. ^ *Kocijan J, Girard A, Banko B, Murray-Smith R. 2005. "Dynamic systems identification with Gaussian processes". Mathematical and Computer Modelling of Dynamical Systems. https://www.tandfonline.com/doi/abs/10.1080/13873950500068567.*
49. ^ *Xu X, Lu Y, Vogel-Heuser B, Wang L. 2021. "Industry 4.0 and Industry 5.0—Inception, conception and perception". Journal of Manufacturing Systems 61:530–535.*
50. ^ *Caso S, Aeby M, Jordan M, Guillot R, Bielser J-M (2022). "Effects of pyruvate on primary metabolism and product quality for a high-density perfusion process". Biotechnology and Bioengineering. 119: 1053–1061.*
51. ^ *Romann P, Schneider S, Tobler D, Jordan M, Perilleux A, Souquet J, Herwig C, Bielser J-M, Villiger TK. 2024. "Raman-controlled pyruvate feeding to control metabolic activity and product quality in continuous biomanufacturing". Biotechnology Journal 19:2300318.*
52. ^ *King RD, Rowland J, Oliver SG, Young M, Aubrey W, Byrne E, Liakata M, Markham M, Pir P, Soldatova LN (2009). "The automation of science". Science. 324: 85–89.Abstract/FREE Full Text*
53. ^ *Nair SH, Govindarajan V, Lin T, Wang Y, Tseng EH, Borrelli F. 2022. "Stochastic MPC with Dual Control for Autonomous Driving with Multi-Modal Interaction-Aware Predictions". arXiv. http://arxiv.org/abs/2208.03525.*