

Robust optimization with optimal experiment design - with application to continuous biopharmaceutical production

Yu Wang^{*,***} Mirko Pasquini^{*,***} Veronique Chotteau^{**,***}
Håkan Hjalmarsson^{*,***} Elling W. Jacobsen^{*,***}

^{*} *Division of Decision and Control Systems, EECS, KTH Royal Institute of Technology, Stockholm 10044, Sweden (e-mails: wang3@kth.se, pasqu@kth.se, hjalmars@kth.se, jacobsen@kth.se).*

^{**} *Division of Industrial Biotechnology, CBH, KTH Royal Institute of Technology, Stockholm 10044, Sweden (e-mail: chotteau@kth.se).*

^{***} *Competence Centre for Advanced Bioproduction by Continuous Processing, AdBIOPRO, Sweden.*

Abstract: Model-based optimization typically obtains the optimum based on a nominal identified model. However, in the presence of uncertainty, the nominal optimum leads to suboptimal operating conditions that furthermore can be highly sensitive to uncertainties. Hence, uncertainty should be considered in the optimization and, furthermore, experiments should be designed to reduce the uncertainty most important for optimization. Herein, we propose a general framework that combines model-based robust optimization with optimal experiment design. The proposed framework can take advantage of prior knowledge in the form of a mechanistic model structure, and the importance of this is demonstrated by comparing to more standard black-box models typically employed in learning. Through optimal experiment design, we repeatedly reduce uncertainties in the region where the likelihood of improvement on the worst-case performance is maximized. This makes the proposed method an efficient model-based robust optimization framework, especially with limited experiment resources. The effectiveness of the method is illustrated by a cell culture development example in continuous biopharmaceutical production.

Keywords: Robust optimization, experiment design, identification for optimization, bioproduction

1. INTRODUCTION

Optimization problems in biology, e.g., cell culture development for biopharmaceutical production [Hu (2020)], are often approached based on trial and error. Such approaches are time-consuming and furthermore lead to suboptimal conditions. To obtain the optimum in a systematic fashion, model-based optimization methods, in which first a nominal model is identified and then the optimization problem is solved based on it, can be employed.

However, due to the existence of various uncertainties, the nominal optimum also leads to suboptimal conditions. Indeed, in bioprocess applications, the available models are typically highly uncertain; due to process complexity and limited experiment resources, such model-plant mismatch is inevitable. There may also exist measurement uncertainty and implementation errors when implementing the optimal solution. In many cases the objective function will be highly sensitive to uncertainty and the system hence operated at suboptimal conditions, far from the optimal ones. Hence, it is important to take uncertainties into account in the optimization.

Robust optimization methods aim at obtaining a robust optimum that has the best worst-case performance within the considered uncertainty set. Although it might appear conser-

vative at first glance, it can guarantee the performance of e.g., production rate of products of interest such as therapeutic antibodies by mammalian cells in the presence of uncertainties, while there are no such guarantees using nominal optimization. Existing model-based optimization methods that seek robust optima usually leverage one-shot open-loop optimization methods, ignoring the possibility to perform additional experiments to reduce uncertainties (see e.g., Bertsimas et al. (2018); Bertsimas and Nohadani (2010); Gabrel et al. (2014); Bertsimas et al. (2011, 2010a,b); Ben-Tal and Nemirovski (2002)). However, in bioprocess applications, the experiments are costly and usually conducted in sequence. Hence, it is important to design the next experiments based on all available information, such that the robust optimum can be sought efficiently in terms of experiment resources.

In order to combine optimization with optimal experiment design, a number of black-box optimization methods have been proposed [Shields et al. (2021); Sabug et al. (2020); Frazier (2018)]. These methods are essentially based on similar ideas as Bayesian optimization: build a surrogate model of the unknown black-box objective function, quantify the corresponding model uncertainty, and then decide the location where the next experiments are performed so as to reduce the uncertainty in the most promising regions. The key advantage of such methods is the efficient search for the global optimum of an objective function that is expensive to evaluate experimentally.

* This work has been supported by the VINNOVA Competence Centre Ad-BIOPRO, contract 2016-05181.

The black-box approaches described above are powerful for applications that lack prior knowledge about the system. However, in many cases, e.g., in most bioprocess applications, valuable prior knowledge is available, e.g., in the form of mechanistic model structures. As we demonstrate in this paper, in cases where black-box models contain insufficient information for optimization, mechanistic models can determine the robust optimum in relatively few experiments. Prior knowledge can in principle be combined with black-box models also, see e.g., Perrone et al. (2019), but such methods are rather involved and may not fully utilize the prior knowledge available.

We here propose a general framework that employs mechanistic model structures directly in the robust optimization method, and combines this with optimal experiment design. This enables a better exploitation of prior knowledge, which is useful especially when the mechanistic model contains significant structural information and only few experiments are allowed. We remark that the model identification and experiment design considered here are for the sole purpose of optimization. By applying the proposed framework, we demonstrate that

- (1) the robust optimum can be distinctly different from the nominal optimum; the robust optimum should be preferred when a performance guarantee is important.
- (2) the prior knowledge available in a mechanistic model can be instrumental in determining the optimum for cases where black-box models are insufficient.
- (3) the proposed framework is efficient in terms of required experiment resources:
 - (a) at each experiment step, the robust optimum with the best worst-case performance can be obtained efficiently based on all available information;
 - (b) the proposed optimal experiment design reduces the uncertainty in the region where the likelihood of the improvement on the worst-case performance is maximized.

We remark that the proposed framework is general and not limited by specific choices of modeling, robust optimization or experiment design methods. Moreover, although our main motivation here is optimization in bioprocess applications, the proposed framework is completely general and can be applied directly to other applications, and is particularly relevant to those for which experimental data are expensive.

The rest of the paper is organized as follows. We first provide a motivating example in Section 2, then formulate the considered problem in Section 3. In Section 4, we introduce the proposed framework. In Section 5, we illustrate the effectiveness of the method on a cell culture development example in continuous biopharmaceutical production. Finally, we provide some conclusions and discussion in Section 6.

2. MOTIVATING EXAMPLE

We first consider a simple numerical example to illustrate the impact of uncertainty on the optimization problem and the importance of exploiting prior knowledge. Consider the problem of minimizing the objective function

$$f(u) = [(au - b)^2 + c] \cdot [(du + e)^2 + g], \quad (1)$$

within the input interval $u \in [-3, 3]$, where the real parameters, to be estimated from measurements, are $a = 1, b = 1.5, c = 0, d = 1, e = 2.5, g = 1$. We assume Gaussian measurement noise at each measurement of the function values,

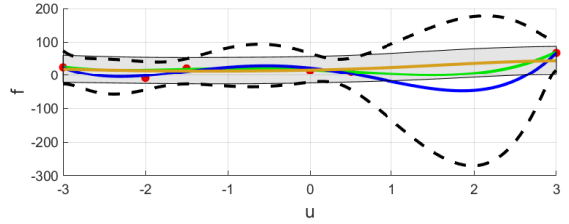


Fig. 1. Comparison between robust and nominal optimum; between model-based and black-box optimization. The meaning of the curves are described in Section 2. In Table 1 the true, nominal and robust optima, with the corresponding worst-case and nominal objective function values, are shown.

$$\hat{f}(u) = f(u) + v, \quad v \sim N(0, \sigma^2), \quad (2)$$

where $\sigma = 10$ and $f(u)$ is the noise-free function value. Initially, we have 6 noisy measurements represented by the red dots in Fig. 1. The true function is represented by the green curve.

To illustrate the difference between the robust and nominal optimum, we apply nonlinear least square with the given function structure to identify a nominal function (blue curve in Fig. 1), and obtain the nominal optimum. We then obtain the upper and lower bounds of the identified function within the 95% confidence ellipsoidal parametric uncertainty set (dash curves in Fig. 1). The robust optimum is the input corresponding to the minimum of the upper bound of the function. The optima are summarized in Table 1. As shown in Fig. 1 and Table 1, in the presence of uncertainty, the function value at the nominal optimum can be the worst within the considered input interval. This implies that the nominal optimum might have a poor performance in the presence of uncertainty. On the other hand, the robust optimum has the minimum worst-case function value and is less sensitive to uncertainty. To guarantee performance in the presence of uncertainty, the robust optimum is the preferred choice.

To illustrate the advantage of utilizing prior model structure knowledge over black-box optimization, we obtain a Gaussian process (GP) model using the noisy measurements and Gaussian kernel [Frazier (2018)]. Note that no prior knowledge is incorporated in the GP model. Both the mean and the 95% confidence interval of the fitted GP model (yellow curve and grey area in Fig. 1) show small variation over the considered input interval. As we demonstrate in Section 5.1, when additional experiments are performed, the variation becomes even smaller and therefore the GP fails to provide useful information for optimization.

Table 1. True, nominal and robust optimization using 6 initial experiments in motivating example.

	True	Nominal	Robust
optimum location	1.50	1.88	-1.85
worst-case function value	144.6	171.9	39.9
nominal function value	-42.1	-48.1	1.2

We next propose a general framework to seek the robust optimum and design further experiments to maximally reduce uncertainty in the most promising region for optimization.

3. PROBLEM FORMULATION

In the presence of uncertainty, we consider obtaining a robust optimum that provides the best worst-case performance guarantee, by solving the, possibly nonconvex, robust optimization problem

$$\begin{aligned} \min_{u \in \mathcal{U}} \max_{\delta \in \Delta} f(u, x, \theta + \delta) \\ \text{s.t. } g_j(u, x, \theta + \delta) \leq 0, \forall \delta \in \Delta, j \in \{1, \dots, I\}, \\ h(u, x, \theta + \delta) = 0, \end{aligned} \quad (3)$$

where $u \in \mathbb{R}^m$ and $\theta \in \mathbb{R}^p$ are the inputs and nominal (identified) parameter values of the system, respectively; $\mathcal{U} \subseteq \mathbb{R}^m$ is the set of feasible inputs; $x \in \mathbb{R}^n$ represents the steady state of the system; $\delta \in \mathbb{R}^p$ represents parametric uncertainties on θ that belong to the uncertainty set Δ ; $f(u, x, \theta + \delta)$ is the objective function to be minimized; $h(u, x, \theta + \delta) = 0$ represents the steady-state equation of the system, which corresponds to the nominal identified one if $\delta = 0$; $g_j(u, x, \theta + \delta) \leq 0, j \in \{1, \dots, I\}$ represents a set of inequality constraints to be satisfied robustly, i.e., for any possible realizations within the uncertainty set; I is the number of considered inequality constraints. We remark that the true parameter values θ are unknown and can be any possible realization within the uncertainty set, i.e., $\theta = \theta + \delta$ for some $\delta \in \Delta$. Note that, in the presence of such uncertainty, nominal optimization cannot provide any performance guarantee.

We consider solving the robust optimization problem (3) under the following mild assumption.

Assumption 1. We assume that

- (1) the parameters θ and the uncertainty set Δ can be uniquely estimated;
- (2) the robust optimization has at least a local optimum.

Solving the robust optimization problem (3) is known to be challenging in general. However, many methods have been developed to seek the robust optimum in tractable ways by using e.g., heuristics [Bertsimas and Nohadani (2010); Bertsimas et al. (2010a,b)] or convexification [Chen et al. (2017)]. We remark again that existing model-based methods solve (3) in an open-loop manner, i.e., further experiments are not designed using the obtained information. We here propose a general model-based framework, which incorporates optimal experiment design, to solve (3) efficiently and reduce uncertainties by sufficiently exploiting available information within a limited number of experiments.

4. MODEL-BASED FRAMEWORK FOR ROBUST OPTIMIZATION

The proposed model-based framework for robust optimization consists of modeling, robust optimization and optimal experiment design, as illustrated in Fig. 2. Given available data $\{u_{seq}, x_{seq}\}$, we first obtain the nominal parameters θ , to which we associate the nominal steady-state model $h(u, x, \theta) = 0$, and uncertainty set Δ , using system identification methods. Based on the identified model and uncertainty set, we then obtain the robust optimum u^* and corresponding best worst-case objective function f^* by solving the robust optimization problem (3). Then, we evaluate the likelihood of the improvement on the worst-case performance, design and conduct a batch of experiments in the region where this likelihood is maximized, and obtain new data $\{u_{new}, x_{new}\}$. We then use all the available data

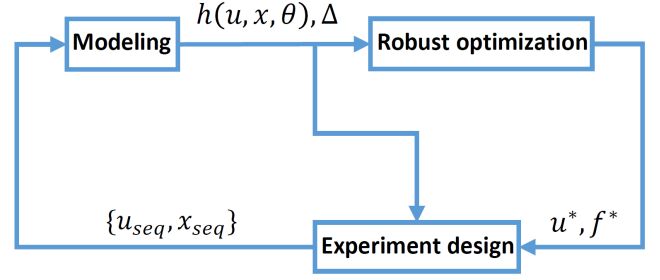


Fig. 2. Model-based framework for robust optimization.

to update the model as well as the uncertainty set. The procedure is repeated until no further improvement on the worst-case performance is possible, or the maximum number of allowed experiments is reached.

4.1 Modeling

We first introduce the modeling module used to identify and update the nominal model and the corresponding uncertainty set. To identify the steady-state model and the uncertainty set using all available data from previous experiments $\{u_{seq}, x_{seq}\}$, we solve a nonlinear least squares problem

$$\theta = \arg \min_{\hat{\theta}} \sum_{(u_i, x_i)} \|h(u_i, x_i, \hat{\theta})\|_2^2, \quad (4)$$

using all the available data $(u_i, x_i) \in \{u_{seq}, x_{seq}\}$. To avoid conservativeness of the uncertainty set, the correlations between the parametric uncertainties should be considered. Hence, we consider the 95% confidence ellipsoidal parametric uncertainty set

$$\Delta = \{\delta \in \mathbb{R}^m \mid \delta^T \Sigma^{-1} \delta \leq \chi^{-2}(\alpha, m)\}, \quad (5)$$

where Σ is the covariance matrix of the vector of parameters θ that can be directly obtained by solving the nonlinear least squares (4); m is the number of parameters; $\chi^{-2}(\alpha, m)$ is the inverse of the chi-square cumulative distribution with m degrees of freedom and significance level $\alpha = 0.05$. We here consider (5) as the deterministic uncertainty set for robust optimization. It is remarked that the above update method can be replaced by other suitable methods, such as set membership identification [Milanese and Taragna (2005)], in the proposed framework.

4.2 Robust optimization

After obtaining the identified model and the corresponding uncertainty set, the robust optimum with the worst-case performance guarantee can be determined by solving (3).

As mentioned in Section 3, solving (3) is challenging in general, but many methods can obtain the robust optimum in tractable ways. As shown in Bertsimas and Nohadani (2010), the global robust optimum for nonconvex problems can be searched efficiently. First note that (3) is equivalent to

$$\begin{aligned} \min_{u \in \mathcal{U}} \max_{\delta \in \Delta} f(u, x, \theta + \delta) \\ \text{s.t. } \max_{\delta \in \Delta} g_j(u, x, \theta + \delta) \leq 0, \forall j \in \{1, \dots, I\}, \\ h(u, x, \theta + \delta) = 0. \end{aligned} \quad (6)$$

We here adopt an approach based on a local search of the robust optimum, mostly drawn from Bertsimas et al. (2010a,b), from multiple starting points in the input space, in which we also consider steady-state equality constraints. We here only

provide an overview of the robust optimization algorithm, for the sake of clarity and ease of explanation. The interested reader is referred to Bertsimas et al. (2010a,b) for additional details. We remark that the particular robust optimization method can be replaced by other methods in the proposed framework.

The robust optimization algorithm can be summarized as follows: starting from different initial inputs, we perform a local search of robust optima, for which in each step we move in a direction that points away from infeasible regions and inputs that would worsen the worst-case objective function value. We stop when there are no feasible directions towards which to move. After obtaining different local robust optima, starting from different initial inputs, we select the one corresponding to the best worst-case objective function value.

We next detail the different parts of the above described algorithm.

Initialization Let u^1 be an initial point within the input space.

Perturbations on current input To determine the direction that reduces the worst-case objective function value $\max_{\delta \in \Delta} f(u, x, \theta + \delta)$ and avoids potential constraint violation, we explore the neighborhood around the input $u^k \in \mathbb{R}^m$, at step k , by generating inputs around u^k . These inputs can be generated with different methods, such as component-wise perturbation, random perturbation, specific region exploration technique etc. We here generically consider \mathcal{U}_p^k as a set of perturbed inputs around u^k .

Neighborhood exploration To ensure constraint satisfaction, for each perturbed input $u_p^k \in \mathcal{U}_p^k$, we obtain

$$\begin{aligned} \max_{\delta \in \Delta} g_j(u_p^k, x_p^k, \theta + \delta), \\ \text{s.t. } h(u_p^k, x_p^k, \theta + \delta) = 0. \end{aligned} \quad (7)$$

where $j = 1, \dots, I$ and x_p^k is the corresponding state obtained by solving the steady-state equation at the particular parameter realization $\theta + \delta$, when the input u_p^k is applied. This can be obtained by using e.g., multiple gradient ascents from different initial parameter values within Δ . If there exists potential constraint violation within the uncertainty set, i.e.,

$$\exists j \in \{1, \dots, I\} \text{ s.t. } \max_{\delta \in \Delta} g_j(u_p^k, x_p^k, \theta + \delta) > 0, \quad (8)$$

we record the corresponding perturbed input in a set \mathcal{M}^k . To reduce the worst-case objective function value, we obtain the worst-case objective function value for each perturbed input u_p^k as

$$\begin{aligned} \max_{\delta \in \Delta} f(u_p^k, x_p^k, \theta + \delta), \\ \text{s.t. } h(u_p^k, x_p^k, \theta + \delta) = 0, \end{aligned}$$

where x_p^k is obtained as mentioned above. If the worst-case objective function value of u_p^k is not smaller than that of the current input u^k , i.e.,

$$\max_{\delta \in \Delta} f(u^k, x^k, \theta + \delta) \leq \max_{\delta \in \Delta} f(u_p^k, x_p^k, \theta + \delta), \quad (9)$$

we record the point $(u_p^k, \max_{\delta \in \Delta} f(u_p^k, x_p^k, \theta + \delta))$ in a set \mathcal{Y}^k .

Move towards the next input To reduce the worst-case objective function value and ensure feasibility, the direction towards the next input u^{k+1} is selected such that it points away from

the points in \mathcal{M}^k and \mathcal{Y}^k , which is obtained by solving the following second-order cone program (SOCP) [Bertsimas et al. (2010a)]

$$\begin{aligned} \min_{d_k, \beta} \beta, \\ \text{s.t. } \|d_k\|_2 \leq 1, \beta \leq -\varepsilon, \\ d_k' \left(\frac{u_p^k - u^k}{\|u_p^k - u^k\|_2} \right) \leq \beta, \forall u_p^k \in \{\mathcal{M}^k, \mathcal{Y}^k\}, \end{aligned} \quad (10)$$

where ε is a small positive scalar. The interpretation of the optimal d_k^* in (10) is the direction that forms the largest possible angle with the points in \mathcal{M}^k and \mathcal{Y}^k . After obtaining d_k^* , the next input u^{k+1} is then given by

$$u^{k+1} = \text{Proj}_{\mathcal{U}} \{u^k + a d_k^*\}, \quad (11)$$

where a is a predefined fixed step size chosen by the user to be coherent with the set of explored perturbed inputs \mathcal{U}_p^k , and $\text{Proj}_{\mathcal{U}} \{v\}$ represents the projection of v inside the feasible inputs set \mathcal{U} . If (10) is infeasible, the seeking of robust optimum should stop, since we cannot determine a viable direction to update the input. A similar algorithm has been used in Bertsimas et al. (2010a) to seek local robust optima.

Multiple local search We limit ourselves to local optima here, but the approach can be extended to global optimization by employing, e.g., simulated annealing based optimization [Bertsimas and Nohadani (2010)]. In our case, we repeat the search of a local robust optimum starting from different initial points in the input space. The local robust optimum corresponding to the minimum worst-case objective function value f^* is then selected.

4.3 Optimal experiment design

After obtaining the robust optimum based on the current available information, we design the next experiments to efficiently reduce uncertainties through model and uncertainty set update.

Given limited experiment resources, we would like to reduce uncertainties in a data efficient fashion, that is, instead of reducing uncertainties globally, the experiments should be focused on regions where the objective function has a small worst-case value and where the uncertainty is large. This is called exploitation and exploration, respectively, in contexts such as Bayesian optimization [Frazier (2018)]. It is important to design further experiments such that there is a tradeoff between exploitation and exploration. We assume, in this context, that the objective function can take any value within the uncertainty set, with equal probability.

We here design the next experiments in the region where the likelihood of the expected improvement of the best worst-case performance is maximized, to achieve the exploitation and exploration tradeoff automatically. Note that although the experiment design method is similar to Bayesian optimization [Frazier (2018)], we here consider a deterministic model-based approach to obtaining robust optimum, rather than using a black-box model that may not contain sufficient information for optimization, as we will demonstrate in Section 5.

We first need to obtain the upper and lower bounds within the uncertainty set Δ of the function $f(u, x, \theta + \delta)$, for each feasible input. In general, this can be computationally demanding, so in order to reduce such computational burden, we sample the input space by obtaining the bounds only for the feasible inputs

u_p^k already explored in Section 4.2. The upper bound corresponding to u_p^k has already been obtained during neighborhood exploration in robust optimization, while the lower bounds can be obtained by solving

$$\begin{aligned} \min_{\delta \in \Delta} f(u_p^k, x_p^k, \theta + \delta), \\ \text{s.t. } h(u_p^k, x_p^k, \theta + \delta) = 0. \end{aligned} \quad (12)$$

Let the upper and lower bounds corresponding to u_p^k be \bar{f}_p^k and \underline{f}_p^k respectively. Since for each given input u_p^k , we assume that the probability of any realization of the objective function within the uncertainty set is equal, the expected value of the objective function with the input $u = u_p^k$ is given by

$$\mathbb{E}(f_p^k) = \frac{\bar{f}_p^k + \underline{f}_p^k}{2}, \quad (13)$$

while the corresponding uncertainty interval of the objective function value is given by

$$\Delta_{f_p^k} = \bar{f}_p^k - \underline{f}_p^k. \quad (14)$$

Similar to the expected improvement acquisition function in Bayesian optimization [Frazier (2018)], we select the center of the next experiment region by solving

$$u_{exp} = \arg \max_{u_p^k} \max\{0, f^* - \mathbb{E}(f_p^k)\} \Delta_{f_p^k}. \quad (15)$$

We recall that the inputs u_p^k , over which the maximization in (15) is done, are the ones already explored in the robust optimization step, hence they already belong to \mathcal{U} . The method favors locations where the expected best worst-case objective function value is small compared to f^* , i.e., the objective function value corresponding to the current robust optimum, and where the uncertainty interval of the possible worst-case objective function value is large. Therefore, the tradeoff between exploitation and exploration is achieved automatically.

If $\max\{0, f^* - \mathbb{E}(f_p^k)\} = 0$, $\forall u_p^k \in \mathcal{U}_p^k$, it means that there is little further improvement to be obtained on the current robust optimum by conducting new experiments. Otherwise, we design and conduct a batch of steady-state experiments in a neighborhood around u_{exp} , i.e., a sequence of input $\{u_{new}\}$ and the corresponding steady-state measurements $\{x_{new}\}$, to further reduce uncertainties in the region.

After collecting the new data $\{u_{new}, x_{new}\}$, the uncertainties in the region can be reduced by re-identifying the steady-state model using all available data including those from previous experiments.

Note that the acquisition function can be replaced by other function forms that are more suitable for particular applications, for example by trying to achieve a different trade-off between exploitation and exploration.

4.4 Algorithm summary

The algorithm of the proposed model-based framework for robust optimization is summarized as follows:

- (1) with the initially available data $\{u_{seq}, x_{seq}\}$, select the fixed step size a in (11) and the maximum allowed number of experiments N ;
- (2) identify nominal parameters θ and steady-state model $h(u, x, \theta) = 0$ by solving (4), and construct the 95% confidence ellipsoidal parametric uncertainty set Δ using (5);

- (3) obtain the robust optimum u^* and the corresponding worst-case objective function f^* based on the available information, using the algorithm described in Section 4.2;
- (4) design and conduct a new batch of steady-state experiments in a neighborhood around u_{exp} to further reduce uncertainties in the region, where u_{exp} is determined as in (15). These new data $\{u_{new}, x_{new}\}$ are added to the already available data $\{u_{seq}, x_{seq}\}$;
- (5) repeat 2) - 4) until $\max\{0, f^* - \mathbb{E}(f_p^k)\} = 0$, $\forall u_p^k \in \mathcal{U}_p^k$ or the number of experiments exceeds N ;
- (6) perform steps 2) and 3), based on all the available data $\{u_{seq}, x_{seq}\}$, to obtain the best robust optimum given the available experiment resources.

Remark 2. As already mentioned previously, the proposed framework is not limited by the specific choices of modeling, robust optimization, or experiment design methods made here. Indeed, each of them can be replaced by methods better suited to a specific application.

5. EXAMPLES

We first revisit the motivating example of Section 2 and then consider a more realistic example in cell culture medium development for continuous biopharmaceutical production.

5.1 Motivating example revisited

As shown in Section 2, based on the initially available information, the nominal optimum has a small function value, but also potentially poor performance due to large uncertainty around the nominal optimum. It is therefore logical to attempt to reduce the uncertainty around the nominal optimum in the next experiment. We here apply the proposed method to efficiently design the next experiments in order to repeatedly reduce the uncertainty in the most promising region for optimization.

We assume that the available experiment resources allow us to perform $N = 2$ additional batches of experiments, with one single experiment in each batch. Due to the limited number of experiments, it is of utmost importance to efficiently design the next experiment to reduce the uncertainty in the region where the likelihood of the expected improvement of the best worst-case performance is maximized. By maximizing the acquisition function in (15), the location of the next experiment is determined at $u = 1.99$ which is close to the nominal optimum, as illustrated in Fig. 3 (a). As shown in Fig. 3 (b), after conducting the new experiment in batch 1, the uncertainty around the region of interest is indeed greatly reduced.

As illustrated in Fig. 3, the proposed method repeatedly computes the robust optimum and designs new experiments to reduce the most relevant uncertainty to optimization. The robust optimum and the next experiment at each batch of experiments are summarized in Table 2, where we compare the results from the robust and nominal optimization, by comparing the worst-case of the objective function value when the inputs obtained from the robust (Worst-case w. robust below) or the nominal (Worst-case w. nominal below) optimization are applied. After 2 batches of experiments, the robust optimum is $u^* = 1.62$ which is close to the true optimum $u = 1.5$. The lower figure in Fig. 3 (c) shows that if more experimental resources are available, we should conduct experiments at $u = 1.78$, to further reduce uncertainties and get more information around the region of the true optimum. We note that although the nominal

optima are close to the true optimum, the performance cannot be guaranteed based on the available information. On the other hand, the robust optimum at each batch is always with the best worst-case performance and converges to a neighborhood around the true optimum as more experiments are conducted.

Note that the GP model identified at each batch fails to provide useful information for optimization and reduce uncertainties, since both the corresponding mean (yellow curve) and confidence interval (grey region) are essentially constant. This is mainly due to the fact that the employed kernel of the GP, which here corresponds to the standard choice, is not able to fully utilize the information available in the data. However, the proposed model-based approach is able to identify the robust optimum and design new experiments to reduce the most important uncertainty. Thus, prior knowledge in the form of a model structure as considered here can prove crucial to locate the optimum and design informative experiments for optimization.

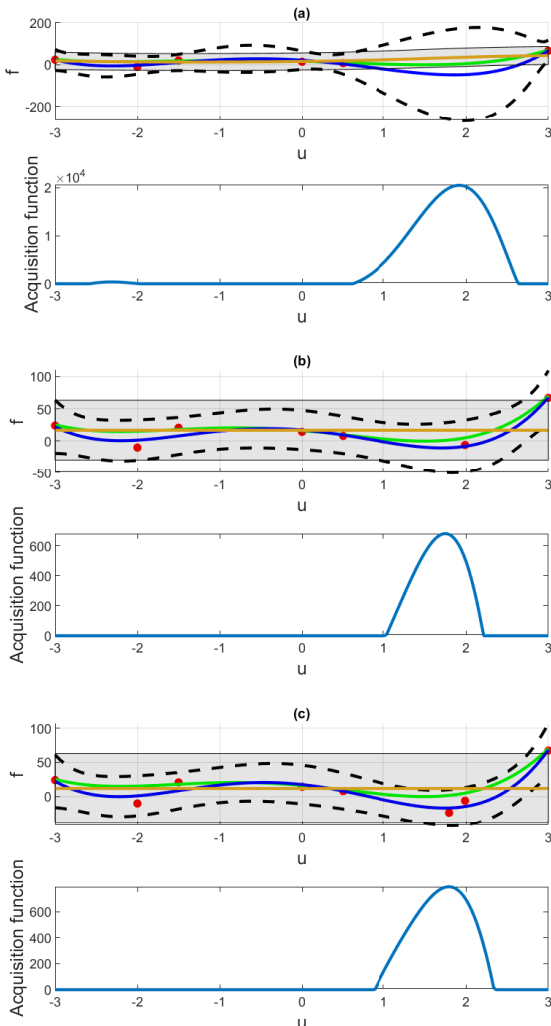


Fig. 3. Results of the proposed method in Example 5.1. Fig. (a), (b), (c) are the results of initial experiments, batch 1 and batch 2 of experiments respectively. The meaning of the curves are described in Section 2.

Table 2. Results with robust and nominal optimization and experiment design in Example 5.1.

	Init. exp.	Batch 1	Batch 2
Robust optimum u^*	-1.85	1.38	1.62
Worst-case w. robust	39.9	26.2	9.2
Nominal optimum	1.88	1.70	1.73
Worst-case w. nominal	171.9	28.0	9.5
Next experiment location u_{exp}	1.99	1.74	1.78

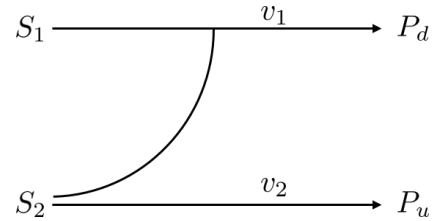
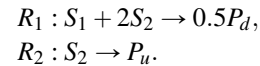


Fig. 4. Two substrates and two products metabolic network.

5.2 Application: robust perfusion culture medium optimization in biopharmaceutical production

In biomanufacturing using mammalian cells for the production of biopharmaceuticals, a particular case of continuous production operation mode, perfusion process, is becoming popular due to its potential to maintain high productivity for an extended period of time. One of the key problems in this context is the determination of the optimal cell culture medium composition [Zhang et al. (2021)]. The culture medium determines the state of the biochemical reaction network of the cells, and therefore the rates at which products of interest are produced. Existing methods to solve this problem are typically time-consuming and lead to suboptimal operating conditions due to the fact that available models typically are highly uncertain and applied optimization methods mainly are based on trial and error [Hu (2020)]. Furthermore, in a bio-pharmaceutical context, it is important to have guarantees on the yield of products of interest, something which can be obtained using a robust optimization approach. Moreover, due to resource limitations, only a limited number of experiments are typically allowed for the purpose of optimization. Thus, it is important to take both uncertainty and limited resources into account in the medium optimization. To tackle the problem, we here apply the framework proposed in Section 4.

Consider the simple example of the metabolic network illustrated in Fig. 4 with two substrates S_1 and S_2 , two products P_d and P_u and the two reactions:



We use the uppercase (lowercase) letters to represent the substrates and the products (their concentrations). The objective is to obtain the optimal culture medium composition, consisting of the feed of the substrates S_1 and S_2 , that maximizes the desired product concentration p_d while minimizing the undesired product concentration p_u at the steady state of the metabolic network. The corresponding objective function, to be minimized, can be formulated as:

$$f(p_d, p_u) = -\lambda p_d + (1 - \lambda)p_u,$$

where $\lambda \in [0, 1]$ is a weight on the importance of maximizing p_d relative to minimizing p_u . The steady-state equations for the metabolic network are:

$$\begin{aligned}
-v_1(x, \theta) \cdot b - F s_1 + F u_1 &= 0, \\
-2v_1(x, \theta) \cdot b - v_2(x, \theta) \cdot b - F s_2 + F u_2 &= 0, \\
\frac{1}{2}v_1(x, \theta) \cdot b - F p_d &= 0, \\
v_2(x, \theta) \cdot b - F p_u &= 0,
\end{aligned} \tag{16}$$

where b is the biomass concentration at steady state; v_1 and v_2 are the rates of the reactions R_1 and R_2 ; F is the perfusion flow rate; $x = [s_1 \ s_2 \ p_1 \ p_2]^\top$ is the vector of the steady-state substrates and products concentrations; $u = [u_1 \ u_2]^\top$ is the feeding culture medium which we aim to optimize. We assume that the rates v_1 and v_2 have the following structures:

$$\begin{aligned}
v_1(x, \theta) &= v_{1,\max} \frac{s_1}{(s_1 + \theta_1) \cdot (1 + s_1 \theta_2)} \cdot \frac{s_2}{(s_2 + \theta_3) \cdot (1 + s_2 \theta_4)} \cdot \frac{1}{(1 + \theta_5 p_d)}, \\
v_2(x, \theta) &= v_{2,\max} \cdot \frac{s_2}{(s_2 + \theta_6) \cdot (1 + s_2 \theta_7)} \cdot \frac{1}{(1 + \theta_8 p_u)},
\end{aligned}$$

where the parameters θ are the nominal ones, which are estimated and hence inherently subject to uncertainties. Due to the existence of uncertainties, we seek the robustly optimal feeding culture medium u within the available experiment resources.

Let \underline{x}, \bar{x} be the imposed lower and upper bounds on the steady-state concentrations, respectively, and \underline{u}, \bar{u} be the imposed lower and upper bounds on the feeding medium concentrations, respectively. The robust optimization problem is then formulated as

$$\begin{aligned}
\min_u \max_{\delta \in \Delta} & -\lambda p_d + (1 - \lambda) p_u \\
\text{s.t.} \quad & \underline{u} \leq u \leq \bar{u} \\
& \max_{\delta \in \Delta} (x - \bar{x}) \leq 0 \\
& \max_{\delta \in \Delta} (x - \underline{x}) \leq 0 \\
& -v_1(x, \theta + \delta) \cdot b - F s_1 + F u_1 = 0 \\
& -2v_1(x, \theta + \delta) \cdot b - v_2(x, \theta + \delta) \cdot b - F s_2 + F u_2 = 0 \\
& \frac{1}{2}v_1(x, \theta + \delta) \cdot b - F p_d = 0 \\
& v_2(x, \theta + \delta) \cdot b - F p_u = 0
\end{aligned} \tag{17}$$

where the uncertainty set Δ is defined as in (5). Note that x indirectly depends on the uncertainty δ , since it is obtained by solving the steady-state equations that depend on the particular uncertainty realization. We next solve (17) using the framework described in Section 4. The nominal parameter values and the setup of the framework are summarized in Appendix A.

We assume there exist 25 initial experiments, and the available experiment resources allow us to perform $N = 3$ batches of experiments, with 5 experiments for each batch. The experiments are with 10% uniformly distributed relative measurement noise. To illustrate the results, following each batch of experiments we compare the obtained robust optimum with the nominal one obtained from a nominal optimization using the estimated parameters. As shown in Table 3, the robust and nominal optimum correspond to widely different feeding culture medium compositions. In the presence of uncertainty, the worst-case performance of the input obtained through nominal optimization is significantly worse than the one obtained through robust optimization. Thus, to guarantee a certain yield of the product of interest under uncertainty, the medium composition corresponding to the robust optimum should be selected. In Fig. 5, the worst-case objective function values obtained in the two cases are compared. It is noted that the worst-case objective function value decreases with new experiments, i.e., the worst-

case performance is improved, as new information is taken into account by the robust optimization. This also implies that uncertainties are repeatedly reduced by the proposed optimal experiment design. This demonstrates that the proposed framework is an efficient and attractive alternative for bioproduction applications.

Table 3. Results of robust and nominal optimization, in Example 5.2 in different experiment batches.

	Init. exp.	Batch 1	Batch 2	Batch 3
Input w. robust opt.	$\begin{bmatrix} 0.55 \\ 0.50 \end{bmatrix}$	$\begin{bmatrix} 0.89 \\ 1.77 \end{bmatrix}$	$\begin{bmatrix} 0.79 \\ 1.43 \end{bmatrix}$	$\begin{bmatrix} 0.67 \\ 1.46 \end{bmatrix}$
Worst-case w. robust	0.0405	0.0154	0.0115	0.0111
Input w. nominal opt.	$\begin{bmatrix} 0.94 \\ 3 \end{bmatrix}$	$\begin{bmatrix} 1.21 \\ 3 \end{bmatrix}$	$\begin{bmatrix} 1.14 \\ 3 \end{bmatrix}$	$\begin{bmatrix} 0.85 \\ 2.83 \end{bmatrix}$
Worst-case w. nominal	0.1041	0.0557	0.0601	0.0522

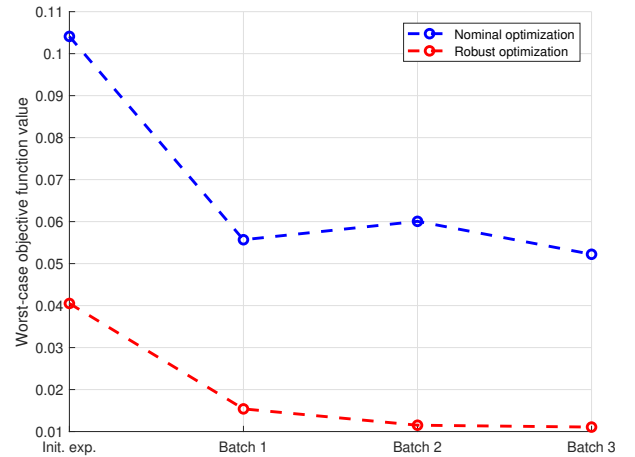


Fig. 5. Comparison between the results of the nominal optimization (blue) and the robust optimization (red), in Example 5.2. It is noted that, for each batch, the robust optimization leads to a lower worst-case of the objective function.

6. CONCLUSION AND DISCUSSION

In this work, we proposed a general model-based framework combining robust optimization with optimal experiment design. We demonstrated that by employing a mechanistic model structure in the robust optimization we could better utilize the available information and also design more informative experiments for the purpose of optimization, as compared to standard black-box approaches. We stress that uncertainty should explicitly be taken into account both in the optimization and in the design of experiments. The proposed method determines a robust optimum that provides worst-case performance guarantee in the presence of uncertainty, and furthermore efficiently design further experiments to reduce the uncertainty that is most relevant to optimization. The framework is general and not limited by specific modeling, robust optimization or experiment design methods, which can all be changed in a modular fashion. We have illustrated the effectiveness of the proposed framework by

studying a feeding cell culture medium optimization example relevant to continuous biopharmaceutical production.

In the context of black-box optimization, robust variants have been proposed previously, e.g., [Bogunovic et al. (2018)]. However, as demonstrated here, adding prior knowledge in the form of mechanistic model structures can be crucial to determine both the robust optimum and also the most informative experiments for optimization.

Although we only considered parametric uncertainties in this work, the proposed framework can be directly extended to other types of uncertainties, including uncertainty that cannot be reduced by obtaining more data, e.g., input uncertainties. This, however, would require more involved methods for the model and uncertainty set update, since the identification problem would become of error-in-variables type. We leave this for future work.

Regarding the computational complexity of the framework, it is remarked that this strictly depends on the particular choices for the different modules of model identification, robust optimization and experiment design. In the case described in Section 4, the main contribution to the computational complexity is the robust optimization scheme. In fact, at step k of the local search, up to $M \cdot I$ maximizations are performed, where M is the size of the perturbed inputs set and I is the number of inequality constraints, to explore the neighborhood of the current input and check for potential constraint violations. Following this, another set of up to M maximizations are performed, to obtain the worst-case objective function values, corresponding to feasible perturbed inputs. Then the SOCP problem (10) is solved, to determine the update direction towards which the input is moved. This is repeated for N different initial conditions, and for each of these K possible steps of the local search algorithm are performed (a stopping criterion can be met before). This implies that to solve the global robust optimization problem, we may need to solve up to $M \cdot (I + 1) \cdot K \cdot N$ maximization problems. Despite being computationally demanding, we note that the complexity increases only linearly with the number of constraints and strictly depends on the number M of perturbed inputs.

In future work, we will consider possible extensions of the framework to enable more general uncertainties. With regards to biopharmaceutical production, we plan to apply the method experimentally, involving mammalian cell lines.

REFERENCES

- Ben-Tal, A. and Nemirovski, A. (2002). Robust optimization—methodology and applications. *Mathematical Programming*, 92(3), 453–480.
- Bertsimas, D., Brown, D.B., and Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM Review*, 53(3), 464–501.
- Bertsimas, D., Gupta, V., and Kallus, N. (2018). Data-driven robust optimization. *Mathematical Programming*, 167(2), 235–292.
- Bertsimas, D. and Nohadani, O. (2010). Robust optimization with simulated annealing. *Journal of Global Optimization*, 48(2), 323–334.
- Bertsimas, D., Nohadani, O., and Teo, K.M. (2010a). Non-convex robust optimization for problems with constraints. *INFORMS Journal on Computing*, 22(1), 44–58.
- Bertsimas, D., Nohadani, O., and Teo, K.M. (2010b). Robust optimization for unconstrained simulation-based problems. *Operations Research*, 58(1), 161–178.
- Bogunovic, I., Scarlett, J., Jegelka, S., and Cevher, V. (2018). Adversarially robust optimization with Gaussian processes. *arXiv preprint arXiv:1810.10775*.
- Chen, R., Lucier, B., Singer, Y., and Syrgkanis, V. (2017). Robust optimization for non-convex objectives. *arXiv preprint arXiv:1707.01047*.
- Frazier, P.I. (2018). A tutorial on Bayesian optimization. *arXiv preprint arXiv:1807.02811*.
- Gabrel, V., Murat, C., and Thiele, A. (2014). Recent advances in robust optimization: An overview. *European Journal of Operational Research*, 235(3), 471–483.
- Hu, W.S. (2020). Medium design for cell culture processing. In *Cell Culture Bioprocess Engineering*, 239–277. CRC Press.
- Milanese, M. and Taragna, M. (2005). H_∞ set membership identification: A survey. *Automatica*, 41(12), 2019–2032.
- Perrone, V., Shen, H., Seeger, M., Archambeau, C., and Jenatton, R. (2019). Learning search spaces for Bayesian optimization: Another view of hyperparameter transfer learning. *arXiv preprint arXiv:1909.12552*.
- Sabug, L., Ruiz, F., and Fagiano, L. (2020). On the use of set membership theory for global optimization of black-box functions. In *2020 59th IEEE Conference on Decision and Control (CDC)*, 3586–3591. IEEE.
- Shields, B.J., Stevens, J., Li, J., Parasram, M., Damani, F., Alvarado, J.I.M., Janey, J.M., Adams, R.P., and Doyle, A.G. (2021). Bayesian reaction optimization as a tool for chemical synthesis. *Nature*, 590(7844), 89–96.
- Zhang, L., Schwarz, H., Wang, M., Castan, A., Hjalmarsson, H., and Chotteau, V. (2021). Control of IgG glycosylation in CHO cell perfusion cultures by GReBA mathematical model supported by a novel targeted feed, TAFE. *Metabolic Engineering*, 65, 135–145.

Appendix A. SIMULATION SETUP AND PARAMETERS IN EXAMPLE 5.2

The true parameters of the metabolic network are $b = 1.6$, $\theta = [0.5 \ 1 \ 3 \ 1 \ 0.75 \ 3 \ 2.75 \ 2 \ 0.25]^\top$, $v_{1,max} = 3$, $v_{2,max} = 3.5$, $F = 0.25$. In the optimization problem, the weight in objective function is $\lambda = 0.9$; the input bounds are $\underline{u} = [0.5 \ 0.5]^\top$ and $\bar{u} = [3 \ 3]^\top$; the concentration bounds are given by $\underline{x} = [0 \ 0 \ 0 \ 0]^\top$ and $\bar{x} = [25 \ 25 \ \infty \ \infty]^\top$. In the proposed framework, after each batch of experiments, we seek a local optimum starting from 4 different initial points $[1 \ 1]^\top$, $[1 \ 2]^\top$, $[2 \ 1]^\top$, $[2 \ 2]^\top$, using 40 steps with constant step size $a = 0.15$; for experiment design, we design 5 new experiments for each batch, which are generated by

$$\text{Proj}_{\mathcal{U}}(u_{exp} + \sigma v),$$

where $\text{Proj}_{\mathcal{X}}(x)$ is the projection of the point x in the set \mathcal{X} ; $\mathcal{U} = \{u \in \mathbb{R}^2 \mid \underline{u} \leq u \leq \bar{u}\}$; $\sigma = 0.25$ and $v \in [-1, 1]^2$.