Proceedings of the 14th International Symposium on Process Systems Engineering – PSE 2021+
June 19–23, 2022, Kyoto, Japan
http://dx.doi.org/10.1016/B978-0-323-85159-6.50286-4

Decision-Focused Surrogate Modeling with Feasibility Guarantee

Rishabh Gupta[a], Qi Zhang[a*]

[a]*Department of Chemical Engineering and Materials Science, University of Minnesota, Minneapolis, MN 55455, USA*

qizh@umn.edu

Abstract

Surrogate models are commonly used to reduce the computational complexity of solving difficult optimization problems. In this work, we consider decision-focused surrogate modeling, which focuses on minimizing decision error, which we define as the difference between the optimal solutions to the original model and those obtained from solving the surrogate optimization model. We extend our previously developed inverse optimization framework to include a mechanism that ensures feasibility (or minimizes potential infeasibility) over a given input space. The proposed method gives rise to a robust optimization problem that we solve using a tailored cutting-plane algorithm. In our computational case study, we demonstrate that the proposed approach can correctly identify sources of infeasibility and efficiently update the surrogate model to eliminate the found infeasibility.

Keywords: surrogate modeling, learning for optimization, inverse optimization, feasibility guarantee.

1. Introduction

A common strategy for solving difficult optimization problems, especially in real-time applications, is to develop surrogate models of reduced computational complexity. In particular, data-driven surrogate modeling methods have become very popular with the opportunity to leverage recent advances in machine learning. Here, one uses the original model to generate data, which are used to fit the surrogate model that can then be embedded in the optimization problem. A key challenge in surrogate modeling is the balance between model accuracy and computational efficiency. As a result, much of the research effort in this area has focused on developing surrogate models that have simple functional forms or specific structures such that the optimization problems are easier to solve using the surrogate models [Cozad et al., 2014, Zhang et al., 2016].

The vast majority of existing surrogate modeling methods construct models that are given as systems of equations, which represent all or part of the equality constraints of the original optimization model [Bhosekar and Ierapetritou, 2018]. The goal of these surrogate modeling algorithms is to minimize the prediction error with respect to the original systems of equations. However, as we found in our recent (not yet published) work, a low prediction error in this kind of surrogate models does not necessarily lead to a low *decision error*, which is defined as the difference between the optimal solutions of the original and the surrogate optimization models. Yet arguably, decision accuracy is what the user

primarily cares about once the optimization model is deployed as a decision-making tool. We developed a data-driven inverse optimization approach to construct surrogate models that take the form of simpler optimization models and directly minimize the decision error; hence, we refer to it as *decision-focused surrogate modeling*.

Decision-focused surrogate modeling focuses on the set of optimal solutions rather than the larger set of feasible solutions. As such, it is prone to generating surrogate models that violate constraints in the original model. In this work, we address this issue by extending our inverse optimization framework to construct surrogate models with feasibility guarantees. We propose a robust optimization approach where we treat the set of possible inputs as an uncertainty set, and we develop a tailored cutting-plane algorithm to solve the resulting extended inverse optimization problem. Results from our computational case study show that using the proposed approach, we can construct surrogate optimization models with feasibility guarantees without substantial sacrifice of decision accuracy.

2. Mathematical Formulation

We consider an original optimization problem of the following general form:

$$\begin{aligned} \underset{x\in R^n}{\text{minimize}} \quad & f(x,u) \\ \text{subject to} \quad & g(x,u) \le 0, \end{aligned} \tag{1}$$

which is a, possibly nonconvex, nonlinear program (NLP). Here, x and u denote the decision variables and model input parameters, respectively. We assume that solving problem (1) requires more time than what is allowed in our desired online application; however, we can solve it offline to generate data in the form of (u_i, x_i)-pairs, where x_i is the optimal solution to problem (1) given the input u_i.

Given a set of data points $\mathcal{I}$, the goal is to generate a surrogate optimization model that is easier to solve but achieves the same or almost the same optimal solutions as the original model. We postulate a surrogate optimization model of the following form:

$$\begin{aligned} \underset{x\in R^n}{\text{minimize}} \quad & \hat{f}(x,u;\theta) \\ \text{subject to} \quad & \hat{g}(x,u;\omega) \le 0, \end{aligned} \tag{2}$$

where $\hat{f}$ and $\hat{g}$ are parameterized by θ and ω, respectively, and are constructed to be convex in x, which renders problem (2) a convex NLP.

The decision-focused surrogate modeling problems attempts to directly learn an optimization model from data that are assumed to be optimal solutions to this model. As such, it gives rise to a data-driven inverse optimization problem (IOP) [Gupta and Zhang, 2021], which can be formulated as follows:

$$\underset{\theta\in\Theta,\,\omega\in\Omega,\,\hat{x}}{\text{minimize}} \quad \sum_{i\in\mathcal{I}} \|x_i - \hat{x}_i\| \tag{3a}$$

$$\text{subject to} \quad \hat{x}_i \in \underset{\tilde{x}\in R^n}{\arg\min} \left\{ \hat{f}(\tilde{x},u_i;\theta) : \hat{g}(\tilde{x},u_i;\omega) \le 0 \right\} \quad \forall\, i \in \mathcal{I}, \tag{3b}$$

where $\hat{x}_i$ denotes the solution predicted by the surrogate model. The objective is to determine the surrogate model parameters θ and ω that minimize the decision error defined in (3a) as the difference between the optimal solution to the original problem x_i and $\hat{x}_i$ across the given data set. Constraints (3b) state that for each $i \in \mathcal{I}$, $\hat{x}_i$ is the optimal solution to the surrogate optimization model with input u_i.

One potential issue with the IOP formulation (3) is that a predicted solution $\hat{x}_i$ is not guaranteed to be feasible in the original model (1). In addition, assuming that the input u can be chosen from a set $\mathcal{U}$, the optimal solution to the surrogate model is not guaranteed to be feasible in (1) for all $u \in \mathcal{U}$ even if $\hat{x}_i$ is feasible in (1) for all $i \in \mathcal{I}$. Hence, to ensure feasibility, we add the following constraints to problem (3):

$$\left.\begin{array}{c} \bar{x} \in \underset{\tilde{x} \in R^n}{\arg\min} \left\{ \hat{f}(\tilde{x}, u; \theta) : \hat{g}(\tilde{x}, u; \omega) \leq 0 \right\} \\ g(\bar{x}, u) \leq 0 \end{array}\right\} \quad \forall\, u \in \mathcal{U}, \tag{4}$$

which state that given a surrogate model defined by θ and ω, the optimal solution to the surrogate model for any $u \in \mathcal{U}$, $\bar{x}$, also has to satisfy the original constraints $g(\bar{x}, u) \leq 0$.

3. Solution Strategy

The extended IOP is a bilevel semi-infinite program. To solve this problem, we propose a cutting-plane algorithm that iterates between a master problem and a cut-generating separation problem. The master problem is formulated as follows:

$$\begin{array}{ll} \underset{\theta \in \Theta,\, \omega \in \Omega,\, \hat{x},\, \bar{x}}{\text{minimize}} & \displaystyle\sum_{i \in \mathcal{I}} \|x_i - \hat{x}_i\| \\ \text{subject to} & \hat{x}_i \in \underset{\tilde{x} \in R^n}{\arg\min} \left\{ \hat{f}(\tilde{x}, u_i; \theta) : \hat{g}(\tilde{x}, u_i; \omega) \leq 0 \right\} \quad \forall\, i \in \mathcal{I} \\ & \bar{x}_j \in \underset{\tilde{x} \in R^n}{\arg\min} \left\{ \hat{f}(\tilde{x}, u_j; \theta) : \hat{g}(\tilde{x}, u_j; \omega) \leq 0 \right\} \quad \forall\, j \in \mathcal{J} \\ & g(\bar{x}_j, u_j) \leq 0 \quad \forall\, j \in \mathcal{J}, \end{array} \tag{5}$$

which is a relaxation of the extended IOP since the semi-infinite constraints (4) have been replaced by a finite number of constraints defined over a discrete set $\mathcal{J}$. For each $j \in \mathcal{J}$, we have a specific input u_j and the corresponding predicted solution $\bar{x}_j$. If the optimal solution to (5) satisfies constraints (4), then it is also optimal for the extended IOP. Otherwise, we solve the following separation problem for each constraint function g_k to identify inputs for which the solutions of the surrogate model violate the original constraints:

$$\begin{array}{ll} \underset{u \in \mathcal{U},\, \bar{x}}{\text{maximize}} & g_k(\bar{x}, u) \\ \text{subject to} & \bar{x} \in \underset{\tilde{x} \in R^n}{\arg\min} \left\{ \hat{f}(\tilde{x}, u; \theta) : \hat{g}(\tilde{x}, u; \omega) \leq 0 \right\}. \end{array} \tag{6}$$

If the optimal value of (6) is greater than zero (or some defined feasibility threshold ϵ), we add the corresponding input u to the set $\mathcal{J}$ and re-solve problem (5). By doing so, we iterate between the master and the separation problems until no more constraint violations can be found, which indicates that we have solved the extended IOP.

Both problems (5) and (6) are bilevel optimization problems. To solve them, we first reformulate them into single-level problems by replacing the lower-level problems with their KKT conditions, which is possible since the surrogate optimization model is designed to be convex. The resulting formulations generally do not satisfy common regularity conditions, which makes their direct solution using standard NLP solvers difficult. Instead, we solve an exact penalty reformulation, which we do not describe here in detail due to space limitations. Note that while a local solution to problem (5) is usually enough to provide good results, problem (6) has to be solved to global optimality to guarantee feasibility.

4. Computational Case Study

In our case study, we consider the heat exchanger network shown in Figure 1, which is adopted from Biegler et al. [1997]. Here, the inlet temperature of stream H2, T_5, has a nominal value of 583 K but is subject to random disturbances. Whenever there is a change in T_5, we optimize the operation of the heat exchanger network by solving the following NLP in which we can adjust the cooling duty Q_c and the heat capacity flowrate F_{H2}:

$$\begin{aligned}
\underset{Q_c,\,F_{H2}}{\text{minimize}} \quad & 10^{-2}\,Q_c + 4\,(F_{H2} - 1.7)^2 && \text{(7a)}\\
\text{subject to} \quad & 0.5\,Q_c + 165 \geq 0 && \text{(7b)}\\
& -10 - Q_c + (T_5 - 558 + 0.5\,Q_c)\;F_{H2} \geq 0 && \text{(7c)}\\
& -10 - Q_c + (T_5 - 393)\;F_{H2} \geq 0 && \text{(7d)}\\
& -250 - Q_c + (T_5 - 313)\;F_{H2} \geq 0 && \text{(7e)}\\
& -250 - Q_c + (T_5 - 323)\;F_{H2} \leq 0 && \text{(7f)}\\
& Q_c \geq 0,\; F_{H2} \geq 0, && \text{(7g)}
\end{aligned}$$

which is nonconvex due to the bilinear term in constraint (7c).

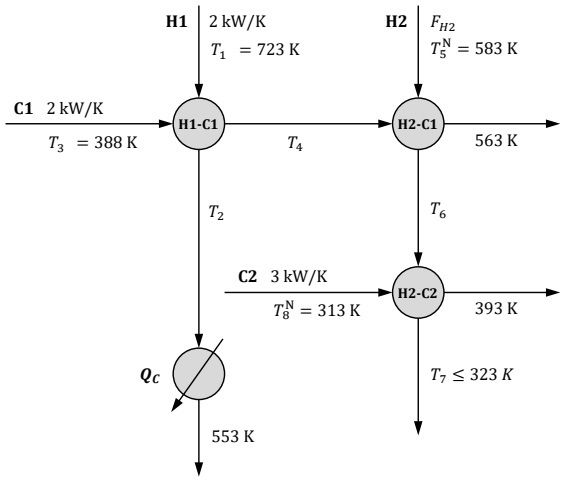


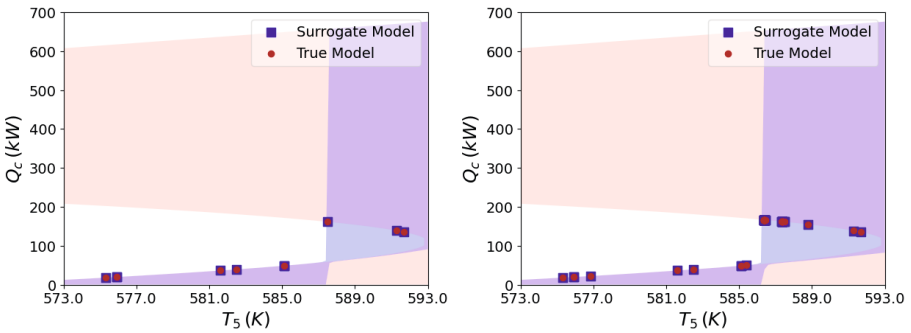
Figure 1: Given heat exchanger network.

We employ the proposed decision-focused surrogate modeling approach to replace the bilinear term $Q_c\, F_{H2}$ in constraint (7c) with the following approximation:

$$Q_c\, F_{H2} \rightarrow a(T_5)\, Q_c + b(T_5)\, F_{H2}, \tag{8}$$

where a and b are some functions of the input parameter T_5. This change, together with estimating the objective function $\hat{f}$ as a convex quadratic function and keeping all linear constraints, results in a surrogate convex QP for problem (7) that is much easier to solve.

We obtain the initial surrogate model by randomly sampling ten values of T_5 in the range [573 K, 593 K] and solving problem (3) with the corresponding global optimal solutions of (7). Here, we assume a and b in (8) to be cubic polynomials in T_5. The result is depicted in Figure 2a, which shows, for each chosen T_5, the true optimal Q_c and the Q_c obtained from solving the surrogate optimization model. In addition, it shows the sets of feasible Q_c for the original (red area) and surrogate (blue area) models. One can observe that while the feasible regions are quite different, there is very good agreement in the true and predicted optimal solutions, which can be attributed to the decision-focused nature of our approach.



(a) Iteration 1 (b) Iteration 25

Figure 2: Comparison between the original model and the surrogate optimization model.

Next, we solve the extended IOP to minimize the violation of constraint (7c) at the optimal solutions of the surrogate model. We perform 25 iterations of the proposed cutting-plane algorithm. Figure 3 shows the maximum constraint violation, which is the optimal value of problem (6) solved for constraint (7c), and the corresponding violated input temperature T_5 that is then added to set $\mathcal{J}$ in problem (5) at each iteration. One can see that as the algorithm progresses, violations across the entire input range are detected until from iteration 13 onward, the algorithm only detects constraint violation in the region around $T_5 = 586.3$ K. This can be explained by Figure 2b, which shows all training data points accumulated over the 25 iterations and the feasible regions of the true and surrogate models. We see that for $T_5 \geq 586.3$ K, part of the feasible region of the surrogate model is infeasible in the true model. While the surrogate model achieves a very good fit for almost all optimal solutions in this region, there seems to be always some point at $T_5 \approx 586.3$ K that is infeasible, which is where we see a "transition" in the feasible region of the surrogate model. This indicates that the proposed cubic approximation of constraint (7c) is not sufficient to achieve feasibility across the entire input range, resulting in the algorithm focusing

on minimizing infeasibility by repeatedly sampling the area around 586.3 K. However, our algorithm correctly identifies the main source of infeasibility. In this particular case, the result instructs a simple remedy of the problem, which is to create two surrogate models, one for $T_5 < 586.3$ K and one for $T_5 \geq 586.3$ K. Then, with the same training data points, solving the corresponding IOPs directly returns two surrogate optimization models whose optimal solutions are feasible for the entire input space.

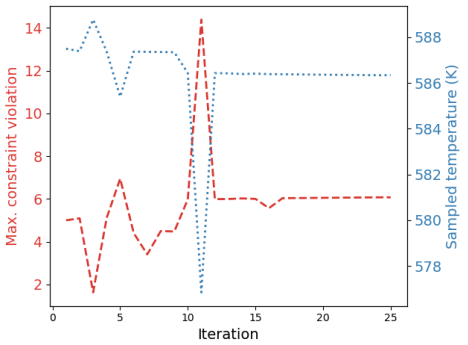


Figure 3: Progression of the cutting-plane algorithm.

5. Conclusions

In this work, we developed a decision-focused surrogate modeling approach that generates surrogate optimization models with feasibility guarantees. This is achieved by combining concepts from inverse optimization and robust optimization, and solving the resulting problem using a tailored cutting-plane algorithm. A computational case study considering a heat exchanger network example demonstrates the ability of the proposed approach to effectively identify and eliminate sources of infeasibility.

References

A. Bhosekar and M. Ierapetritou, 2018. Advances in surrogate based modeling, feasibility analysis, and optimization: A review. *Computers and Chemical Engineering*, 108:250–267.

L. T. Biegler, I. E. Grossmann, and A. W. Westerberg, 1997. Systematic methods for chemical process design.

A. Cozad, N. V. Sahinidis, and D. C. Miller, 2014. Learning Surrogate Models for Simulation-Based Optimization. *AIChE Journal*, 60(6):2211–2227.

R. Gupta and Q. Zhang, 2021. Decomposition and adaptive sampling for data-driven inverse linear optimization. *INFORMS Journal on Computing*, forthcoming.

Q. Zhang, I. E. Grossmann, A. Sundaramoorthy, and J. M. Pinto, 2016. Data-driven construction of Convex Region Surrogate models. *Optimization and Engineering*, 17(2): 289–332.