Adaptive Real Time Exploration and Optimization for Safety-Critical Systems

Buse Sibel Korkmaz
Imperial College London

Mehmet Mercangöz
Imperial College London

Marta Zagórowska
ETH Zürich

Abstract

We consider the problem of decision-making under uncertainty in an environment with safety constraints. Many business and industrial applications rely on real-time optimization with changing inputs to improve key performance indicators. In the case of unknown environmental characteristics, real-time optimization becomes challenging, particularly for the satisfaction of safety constraints. We propose the ARTEO algorithm, where we cast multi-armed bandits as a mathematical programming problem subject to safety constraints and learn the environmental characteristics through changes in optimization inputs and through exploration. We quantify the uncertainty in unknown characteristics by using Gaussian processes and incorporate it into the utility function as a contribution which drives exploration. We adaptively control the size of this contribution using a heuristic in accordance with the requirements of the environment. We guarantee the safety of our algorithm with a high probability through confidence bounds constructed under the regularity assumptions of Gaussian processes. Compared to existing safe-learning approaches, our algorithm does not require an exclusive exploration phase and follows the optimization goals even in the explored points, which makes it suitable for safety-critical systems. We demonstrate the safety and efficiency of our approach with two experiments: an industrial process and an online bid optimization benchmark problem.

1 INTRODUCTION

In most sequential decision-making problems under uncertainty, there exists an unknown function with noisy feedback that the decision-maker algorithm needs to sequentially estimate and optimize to reveal the decisions leading to the highest reward. Each decision incurs an instantaneous reward with an initially unknown distribution in a stochastic optimization setting. In this setting, initial decisions are made based on some heuristics, and incurred reward is memorized to exploit for the next decisions. Therefore, the uncertainty due to unknown characteristics decreases while new decisions are made based on previous reward observations. Even though more exploration can help to optimize the decisions by revealing more information in each iteration, it could be very expensive to evaluate the unknown function for many applications. Therefore, there is a need to balance exploring unknown decision points and exploiting previous experiences. The trade-off between exploration and exploitation is extensively studied in the literature, and the multi-armed bandit (MAB) approaches with confidence bounds are suggested to solve this problem by optimally balancing exploration and exploitation (Bubeck et al., 2012).

Related work. The idea of using confidence bounds to balance the exploration-exploitation trade-off through the optimism principle first appeared in Lai and Robbins (1985) with the utilization of the upper confidence bound (UCB). Since then, it led to the development of UCB algorithms for stochastic bandits with many arms (Lattimore and Szepesvári, 2020). Many efficient algorithms build for bandit problems having a cost or reward function under certain regularity conditions (Dani et al., 2008; Bubeck et al., 2009). In Srinivas et al. (2010), the authors divided the stochastic optimization problem into two objectives: (1) unknown function estimation from noisy observations, and (2) optimization of the estimated function over the decision set. They used kernel methods and Gaussian processes (GPs) to model the reward function since these methods encode the regularity assumptions through kernels (Rasmussen, 2004). Even though these methods are very successful to model and optimize unknown reward functions, they are not applicable to the specific setting that we are interested in this paper since they do not consider safety constraints.

In safety-critical systems, it is not possible to do exploration in some parts of the decision space due to safety
concerns, which are modelled as safety constraints in optimization. Many industrial applications fall under the safety-critical systems due to their risk of danger to human life, leading to substantial economic loss, or causing severe environmental damage. For example, we can consider a chemical process plant as a safety-critical operation since we need to satisfy the constraints of chemical reactions and surrounding processes to not cause a hazardous event for the environment and human operators. Therefore, if we want to approximate the unknown characteristics and optimize a process in this plant, we need to utilize safe exploration algorithms which allow only the exploration of feasible decision points by enforcing safety constraints. In this work, we define a feasible decision point in the given decision set as satisfying the safety and other constraints of the given problem.

The safe exploration problem has been studied by formalizing it as both bandit and Markov Decision Processes (MDPs) problems [Schreiter et al. 2015; Turchetta et al. 2019; Wachi et al. 2018; Turchetta et al. 2016]. In Sun et al. (2015), the authors have introduced the SafeOpt algorithm and showed it is possible that safely optimize a function with an unknown functional form by creating and expanding a safe decision set through safe exploration under certain assumptions. They also provided the safety guarantee of SafeOpt by using the confidence bounds construction method of Srinivas et al. (2010). These safe exploration algorithms are applied to many control and reinforcement learning problems and proved their success in those domains Berkenkamp et al. 2016; Kabzan et al. 2019. However, these algorithms either require an exploration phase or apply a trade-off strategy between optimal decisions and exploration. There are some real-world applications such as industrial processes which cannot afford an exclusive exploration phase or require following optimization goals of the system even in the explored points. For example, any deviation from target satisfaction in an industrial process may cause high costs to the plant or damage the reputation of the responsible party. Thus, there is a need to consider the exploration in an adaptive manner in accordance with the requirements of the environment and this aspect is not covered by existing safe exploration algorithms.

Our contributions. In this paper, we propose a novel safe exploration algorithm, ARTEO, where we cast multi-armed bandits as a mathematical programming problem subject to safety constraints for the optimization of safety-critical systems. Our contribution is posing a further constraint to the safe exploration by instructing the algorithm to avoid decisions not satisfying the optimization goals of the safety application. This constraint discourages exploration which makes it harder to learn unknown rewards in the MAB setting. To compensate for that, we incorporate the exploration as a contribution to the utility function in an adaptive manner by encouraging explicitly to take decisions at points with high uncertainty which is quantified by the covariance of GP models. We adaptively control the size of this contribution using a heuristic in accordance with the requirements of the environment. ARTEO uses GPs to model how the unknown parameters of an application change based on external factors such as inputs and optimization goals, make new decisions when a change in the optimization goals or in the given input is occurred and explores new decision points with either a change in the optimization goals or based on uncertainty. We establish the safety of ARTEO by constructing the confidence bounds as Srinivas et al. (2010) and showing certain assumptions hold in our setting. We demonstrate the safety and efficiency of our algorithm using two real-world examples: the power management problem of the food processing plant refrigeration system and the high-dimensional online bid optimization problem. In the first experiment, the results show that we are able to successfully control three key performance indicators of the process by applying ARTEO. In the second experiment, our approach satisfies the constraints while yielding more profitable outcomes.

2 PROBLEM STATEMENT AND BACKGROUND

We want to find a sequence of decisions, \( x_1, x_2, \ldots, x_T \), such that a certain utility function \( f \) is minimized. At every iteration \( t, t = 1, \ldots, T \), the cost function depends on the decision \( x_t \in \mathcal{D} \) and on the system characteristics \( p_t \in \mathcal{P} \). The characteristics \( p_t \) depends on the decision \( x_t \), i.e. \( p_t = h(x_t) \) where \( p : \mathcal{D}^N \rightarrow \mathcal{P}^N \) with \( N \) arms. After making a decision \( x_t \), we obtain a noisy measurement, \( y_t = f(x_t, p(x_t)) + \epsilon \) where \( f : \mathcal{D}^N \times \mathcal{P}^N \rightarrow \mathbb{R} \) and \( \epsilon \) is \( R \)-sub-Gaussian noise for a fixed constant \( R \geq 0 \) (Agrawal and Goyal 2013). We assume that we know the functional form of \( f(\cdot) \), but the functional form of \( p(\cdot) \) is unknown. Furthermore, at every iteration, the decision \( x_t \) must satisfy the constraints \( g_k(x_t, p(x_t)) + h_k \leq 0 \), where \( k = 1, 2, \ldots, K \) with \( K \) denoting the number of constraints. The value of \( h_k \geq 0 \) is called a safety threshold. Thus, we can formalise our optimisation problem at time \( t \) as:

\[
X_t = \text{argmin}_x f(X_t, p(X_t)) \quad s. t. \quad g_k(X_t, p(X_t)) + h_k \leq 0, \forall k
\]

(1)

where \( X = [x_1, \ldots, x_N] \) for arms \( i = 1, \ldots, N \). If we know \( p(x_t) \), we can solve the problem as an optimization problem with noise. For instance, we could use the concept of real-time optimization (RTO) from the process control domain [Nayshmith and Douglas 2008] and use the approach proposed by Petsagkourakis et al. (2021). However, the characteristics \( p(\cdot) \) are unknown and need to be estimated. Thus, at every iteration \( t \), we need to first solve an estimation problem to find \( p \), then solve the optimization problem.
Equation (1). Solving an optimization problem by combining estimation then optimization is a common approach (Zhang et al., 2022; Fu and Levine, 2021). However, few approaches quantify the uncertainty inherent in the estimation of $p$. In the current paper, we propose to estimate $p$ using Gaussian processes and use their properties to ensure safety.

**Gaussian processes.** Gaussian processes are non-parametric models which can be used for regression. GPs are fully specified by a mean function $\mu(x)$ and a kernel $k(x, x')$ which is a covariance function and decides the shape of prior and posterior in GP regression (Rasmussen, 2004). The goal is to predict the value of the utility function $f$ including unknown components at the decision point $x^*$ by using GPs. Assuming having a zero mean prior, the posterior $p(f(x^*)|x^*, D)$ follows $N(\mu(x^*), \sigma^2(x^*))$ that satisfy,

$$
\begin{align*}
\mu(x^*) &= K_{N,N}^T(K_{N,N} + \sigma^2 I)^{-1}(y_1, \ldots, y_N)^T \\
\sigma^2(x^*) &= k(x^*, x^*) - K_{N,N}^T(K_{N,N} + \sigma^2 I)^{-1}
\end{align*}
$$

where $i, j \in \{1, \ldots, N\}$ and denotes the index of observations, $[K_{N,N}]_{i,j} = k(x_i, x_j)$, $K_{N,N}$ is the $N \times N$ positive definite kernel matrix with $[K_{N,N}]_{i,j} = k(x_i, x_j)$ and $y_i$ is the noisy feedback of $i$th observation.

**Regularity assumptions.** We do not have any prior knowledge of how unknown parameters in $f$ change based on external factors such as the optimization goals or inputs, and to provide safety with high probability at decision points and in exploration we need to make some assumptions, which are common in the literature (Srinivas et al., 2010; Su et al., 2015; 2018; Berkenkamp et al., 2016). We assume the decision set $D$ is compact (and finite in our case) as being a closed and bounded subset of Euclidean space (Hanche-Olsen and Holden, 2010). Furthermore, we assume the utility function $f$ is a continuous function on the compact set $D$ such that it is possible to attain the minimum and maximum values for $f$ due to its smoothness (Ruskonk and Kerr-Lawson, 2005). The smoothness assumption holds for utility function $f$ since we sample it from a GP with a positive definite kernel (Rasmussen, 2004).

Gaussian processes are related to reproducing kernel Hilbert space (RKHS) notion through its positive semidefinite kernel functions (Sriperumbudur et al., 2011), which help us to construct confidence bounds in a safe manner later. The RKHS which is denoted by $\mathcal{H}_k(D)$ is formed by “nice functions” in a complete subspace of $L_2(D)$ and the inner product $\langle \cdot, \cdot \rangle_k$ of functions in RKHS follows the reproducing property: $\langle f, k(x, \cdot) \rangle_k = f(x)$ for all $f \in \mathcal{H}_k(D)$. The smoothness of a function in RKHS with respect to kernel function $k$ is measured by its RKHS norm $\|f\|_k = \sqrt{\langle f, f \rangle_k}$ and for all functions in $\mathcal{H}_k(D)$ $\|f\|_k < \infty$ (Scholkopf and Smola, 2001). Thus, we assume a known bound $B$ for the RKHS norm of the unknown function $f$: $\|f\|_k < B$. We use this bound $B$ to control the width of the confidence interval later in Equation (4).

**Confidence bounds.** In ARTEO, we give safety constraints to the solver as hard constraints. The solver uses the bounds of the confidence interval which is constructed by using the standard deviation of conditioned GP on previous observations to decide the feasibility of a chosen point $x_t$. Hence, the correct classification of decision points in $D$ relies on the confidence-bound choice. Under the regularity assumptions stated before, Theorem 3 of Srinivas et al. (2010) and Theorem 2 of Ray Chowdhury and Gopalan (2017) proved that it is possible to construct confidence bounds which include the true function with probability at least $1 - \delta$ where $\delta \in (0, 1)$ on a kernelized multi-armed bandit problem setting with no constraints. Moreover, as shown by Su et al. (2015) in Theorem 1, this theorem is applicable to multi-armed bandit problems with safety constraints. Hence, we can state that the probability of the true value of safety function $g$ at the decision point $x$ is included by the confidence bounds in iteration $t$:

$$P[|g(x_t) - \mu_{t-1}(x_t)| \leq \sqrt{\beta_t \sigma_{t-1}(x_t)}] \leq 1 - \delta, \ t \geq 1 \tag{3}$$

where $\mu_{t-1}(x)$ and $\sigma_{t-1}(x)$ denote the mean and the standard deviation of sampled functions at $x_t$ from a GP at iteration $t$, which is conditioned on previous $t-1$ observations to obtain the posterior. $\delta$ is a parameter that represents the failure probability in Equation (3) and $\beta_t$ controls the width of the confidence interval and satisfies Equation (3) when:

$$\sqrt{\beta_t} = B + R\sqrt{2(\gamma_{t-1} + 1 + \ln(1/\delta))} \tag{4}$$

where the noise in observations is $R$-sub-Gaussian and $\gamma_{t-1}$ represents the maximum information gain after $t - 1$ iterations and it is formulated as:

$$\gamma_{t-1} = \max_{|A| \leq t-1} I(f; y_A) \tag{5}$$

**3 ARTEO ALGORITHM**

We develop the ARTEO algorithm for safety-critical environments with high exploration costs. Therefore, the ARTEO algorithm is triggered by a change in the optimization goal or by a new event. At each iteration, the algorithm updates the posterior distributions of GPs with previous noisy observations as in Equation (2) and provides an optimized solution for the desired outcome based on how GPs model the unknown components. It does not require a separate training phase, instead, it learns during normal operation. The details of the safe learning and optimization are given next.
3.1 Safe Learning

In a setup with multi-arms, the decision set \( D_i \) is defined for each arm \( i \) as satisfying the introduced assumptions in Section 2. For each arm, a GP prior and initial “safe seed” set is introduced to the algorithm. The safe seed set \( S_0 \) includes at least one safe decision point with the true value of the safety function at that point. As in many published safe learning algorithms (Sui et al., 2015, 2018; Turchetta et al., 2019), without a safe seed set, an accurate assessment of the feasibility of any points is not possible. Each iteration of the algorithm could be triggered by time or an event. After receiving the trigger, the algorithm utilizes the past noisy observations to obtain the GP posterior of each arm to use in the optimization of the utility function, which includes the cost of decision and uncertainty. For the first iteration, past observations are given as safe seed sets.

The uncertainty in the utility function \( f_i \) is quantified as:

\[
U_t(x_i, p(x_i)) = \sum_{i} \sigma(x_{it})
\]

where \( \sigma(x_{it}) \) is the standard deviation of \( GP_i \) at the point \( x_{it} \) for the arm \( i \) in the iteration \( t \). It is incorporated into the utility function \( f \) by multiplying by an adjustable parameter \( z \):

\[
f_t(x_i, p(x_i)) = C_t(x_i, p(X_t)) + zU_t(x_i, p(X_t))
\]

where \( C_t(x_i, p(X_t)) \) represents the cost of decision at the evaluated points. In our experiments, the priority of the algorithm is optimizing the cost of decisions under given constraints. Hence, the uncertainty weight remains zero until the environment becomes available for exploration. Until that time, the algorithm follows optimization goals and learns through changes in the optimization goals such as operating in a different decision point to satisfy a new demand or a change in the given input such as starting a new advertising campaign in our second experiment.

The exploration is controlled by the \( z \) hyperparameter. It is possible to create different heuristic rules for changing the \( z \) parameter during the execution to fine-tune it based on the needs of the simulated environment. This change in the utility function encourages the real-time optimizer to take decisions on unexplored points to decrease uncertainty as a part of the optimization. The safety constraints and the aim of satisfying the expectation with a minimum cost are still under consideration by the RTO during this phase, so, the algorithm does not violate safety constraints with a \( 1 - \delta \) probability as explained in the previous section.

Since we apply the ARTEO algorithm to a dynamical system simulation in our first experiment, we build a steady-state detector and evaluate costs and constraints in a steady-state. For the steady-state detection algorithm, we use a statistical test-based detector as common in many real-time optimization tools (Câmara et al., 2016). The ARTEO algorithm is given in Algorithm 1.

### Algorithm 1 ARTEO

**Input:** Decision set \( D_i \) for each arm \( i \in \{1, ..., n\} \)
- GP priors for each \( GP_i \)
- Safe seed set for each GP as \( S_{i,0} \)
- Uncertainty weight as \( \zeta \)
- Utility function \( f \)
- Safety function \( g \)
- Safety threshold \( h \)
- Confidence interval width parameter \( \beta_t \)

**for** \( t = 1, ..., T_0 \) **do**

**if** Steady-State = True **then**

**if** the exploration heuristic rule holds **then**

\[ z \leftarrow \zeta \]

**else**

\[ z \leftarrow 0 \]

**end if**

**for** \( i = 1, ..., n \) **do**

Update \( GP_i \) by conditioning on \( S_{i,t-1} \)

**end for**

\[ x^*_t \in \{1, ..., n\}, t \leftarrow \arg\min_{x \in D} f(s, t, g) \]

**for** \( i = 1, ..., n \) **do**

\[ y^f_{it} \leftarrow f(x^*_it, p(x^*_it)) + \epsilon^f_t \]

\[ y^g_{it} \leftarrow g(x^*_it, p(x^*_it)) + \epsilon^g_t \]

\[ S_{i,t} \leftarrow S_{i,t-1} \cup y_i, t \]

**end for**

**end if**

**end for**

**Discussion.** We want to discuss the significance and alternative implementations of parameter \( z \) for the exploration. In a fixed environment with a fixed \( z \), the contribution of the exploration could decrease as the environment is explored. As an extension to our current implementation, adding a forgetting factor can be considered to recover the excitement for exploration. For an environment with dynamically changing reward distributions of arms over time, the exploration can be driven by an adaptive version of \( z \) where the distance between the prediction of GPs and the noisy feedback from the system is monitored as a metric for adapting (or adjusting) the exploration effort.

3.2 Optimization

The RTO incorporates the posterior of the Gaussian process into decision-making by modelling the unknown parameters for each arm by using the mean and standard deviation of GPs. The utility function is the objective function in the RTO formulation and the safety thresholds are constraints. In the utility function, the mean of GP posterior of each arm is used to evaluate the cost of decision and the standard deviation of GPs is used to measure uncertainty as in Equation (6). In the safety function, the standard deviation of the GP posterior of each arm is used to construct con-
confidence bounds and then these bounds are used to assess the feasibility of evaluated points. The optimizer solves the minimization problem under safety constraints within the defined decision set of each arm. Any optimization algorithm that could solve the given problem can be used in this phase. We have constrained nonlinear problems in the experiments section, and we choose interior-point and sequential-least square programming (SLSQP) algorithms to solve our first and second problems, respectively.

3.3 Complexity

In each iteration, the ARTEO updates GP models by computing marginal likelihoods, and finds a feasible solution within constraints. The overall time complexity of each run of the algorithm is the number of iterations $t$ times the time complexity of each iteration. The first computationally demanding step in the ARTEO is fitting GPs on safe sets. The time complexity of training a full GP, i.e. exact inference, is $O(n^3)$ due to the matrix inversion where $n$ is the number of observations (Hensman et al., 2013). It is possible to reduce it further by using low-rank approximations which is not in the scope of our work (Chen et al. 2013, Liu et al. 2020). We introduce an individual GP for each arm, so the total complexity of GP calculations is $O(kn^3)$, where $k$ is the number of arms.

The next demanding step is optimization. The computational demand of RTO depends on chosen optimization algorithm and the required $m$ number of steps to converge. The most computationally expensive step for all iterations in both mentioned optimization algorithms is the LDL factorization of a matrix with a $O(n^3)$ complexity (Potra and Wright 2000; Schittkowski 1986). Hence, the complexity of RTO becomes $O(mn^3)$. In our implementation, $m \gg k$, so, the time complexity of each iteration in ARTEO scales with the RTO complexity. Therefore, the overall time complexity of one iteration of ARTEO is $O(mn^3)$. The memory complexity of the algorithm is $O(kn^2)$, which is dominated by matrix storage in GPs and optimization.

4 EXPERIMENTS

In this section, we evaluate our approach on two real-life applications: a fish-freezing process plant refrigeration system and online bid optimization. The former case study is introduced first in Widell and Eikevik (2010) and the latter one in Zhang et al. (2014). We conduct all experiments on an M1 Pro chip with 16 GB memory. The first case study is developed with MATLAB/Simulink and the latter with Python. The code is also available.

---

1 https://github.com/buseskorkmaz/ARTEO
Adaptive Real Time Exploration and Optimization for Safety-Critical Systems

Figure 3: The first iteration of the algorithm. The safe seed set includes three points for each compressor within their respective operating intervals. The blue-shaded area represents the uncertainty which is high due to unknown regions.

Figure 4: The last iteration in the simulation. GP predicted performance lines are converged to the actual performance curves of each compressor. Purple-coloured sample points are chosen by exploration when the environment becomes available. The algorithm encourages exploration specifically in the bounds of operating intervals since these points have high uncertainty as seen in Figure 3.

For the dynamics of the screw compressors, 200 seconds is considered to be an average time to move to a new steady-state based on the dynamic experiments from Fu et al. (2003). Therefore, ARTEO is triggered by 200 seconds time intervals. For the performance of screw compressors, the data is obtained from Manske et al. (2000). The system consists of one small-sized, one medium-sized and three large-sized compressors. The minimum and maximum cooling capacities for each compressor type are given in Table 1. The coefficient of performance for all compressors is assumed to be 1.6 (Widell and Eikevik, 2010).

Table 1: Operating Intervals of Screw Compressors

| Compressor Size | Min. Capacity (kW) | Max. Cooling Capacity (kW) |
|-----------------|--------------------|---------------------------|
| Small           | 56                 | 220                       |
| Medium          | 237                | 537                       |
| Large           | 194                | 795                       |

We proceed as next to put this problem into our framework. We first model the plant in MATLAB/Simulink with the plant’s dynamic characteristics. We choose the Squared Exponential kernel, which is a positive definite kernel function, as the GP prior for each compressor based on the suggestion of the GP hyperparameter optimization tool in Matlab (Rasmussen 2004). The minimum and maximum cooling capacities in Table 1 are introduced as decision-set boundaries to the optimization as in Equation (8).

\[ x_{i \min} \leq x_{it} \leq x_{i \max} \quad \forall i, t \]  

Then, we create the safe seed set including three safe decision points for each compressor to obtain the posterior of GPs before making any decision. We define the utility function \( f_t \) as follows:

\[ f_t(x) = \left[ \sum_{i=1}^{5} \mu_{P_i(x_{it})} \right]^2 + \left[ E_t - \sum_{i=1}^{5} x_{it} \right]^2 + z \sum_{i=1}^{5} \sigma_{P_i(x_{it})} \]  

where \( \mu_{P_i(x_{it})} \) represents the mean prediction of power consumption at production load \( x \) and \( \sigma_{P_i(x_{it})} \) represents the standard deviation of prediction to represent uncertainty in predictions at load \( x_{it} \). \( E_t \) denotes the desired production load at time \( t \) based on the introduced cooling scenario. We develop the following two conditions and assign a positive value \( \zeta = 1000 \) to \( z \) when these conditions hold: (1) the cooling demand is the same with two immediate previous states, (2) there is at least one feasible decision point that satisfies the cooling demand with a tolerance \( \alpha = 10 \) that is explored in two immediate previous states. The safety threshold for power consumption is set as 80% of the maximum power consumption for the sum of all compressors’
power consumptions. The safety function $g$ is defined next as:

$$g_t(x) = \sum_{i=1}^{5} \left[ \mu P_i(x_{it}) + \sqrt{\beta_t \sigma P_i(x_{it})} \right]$$  \hspace{1cm} (10)$$

where $\beta_t$ is chosen as explained in Equation (4).

The experiment is simulated with the given cooling load trajectory in Figure 2. Figure 3 demonstrates the first iteration of the algorithm. In the first iteration, the uncertainty is high and it can be recognized from the wide confidence bounds. At each iteration, GPs are updated with new observations, and the algorithm takes new decisions by the updated GPs. Figure 4 shows the final results for performance curve estimations at the end of the simulated cooling load scenario. The ARTEO algorithm is able to successfully approximate the power consumption curves as close to their real shape.

Figure 5 illustrates the expected and achieved production loads. The expected production load is satisfied unless: (1) it is higher than what compressors can achieve without crossing the safety threshold, (2) it requires compressors to operate lower than their minimum operating points. Figure 6 shows that when the total power consumption is on the limit of the maximum power threshold, expected load could not be satisfied. At those production loads, our approach finds safe operating points which minimize the stated objective function.

### 4.2 Online Bid Optimization

In the second experiment, we investigate the implementation of ARTEO in a multi-dimensional problem of online bid optimization from the advertiser perspective. In bid optimization, the advertiser sets bid values with the aim of achieving high volumes by maximizing the number of shown advertisements and high profitability by maximizing the return-on-investment (ROI) ratio. In most agreements, unsatisfied ROI causes financial losses for advertisers. It becomes more challenging to sustain high ROIs when the number of advertisements increases. Constraining the ROI to remain above a certain threshold is a common approach, however, this method does not guarantee to satisfy the ROI constraint with zero violation (Castiglioni et al., 2022). ROI is measured by the revenues and costs, which are unknown to the bidding algorithms and brings uncertainty to the online bid optimization problem. Therefore, safe optimization algorithms could be useful to set bid values under the uncertainty of the revenues.

We apply the ARTEO algorithm to the iPinYou dataset (Zhang et al., 2014). This dataset has been released by a leading DSP (Demand-Side Platform) in China and consists of relevant information for personalized ads such as creative metadata, interests of users, and advertisement slot properties which include slot width, height, price and visibility. We simulate our approach by creating different campaign subsets from the original data, and for each campaign $t$, we minimize the following utility function:

$$f_t(x) = \sum_{j=1}^{m} c_{x_{tj}} \mu_{C_{tj}} + \sum_{j=1}^{m} \sqrt{(x_{tj} - \mu_{B_{tj}})^2} - \sum_{j=1}^{m} \left( \sigma_{C_{tj}} + \sigma_{B_{tj}} \right)$$  \hspace{1cm} (11)$$

where $j$ denotes the ad number in the campaign, $\mu_{C_{tj}}$ is the mean prediction of GP for the $j$th advertisement to get a click with set bid values $x$, and $\mu_{B_{tj}}$ is the mean prediction of GP for the bid price of $j$th ad in campaign $t$ based on previous similar observations. The fixed budget constraint for $m$ number of ads in campaign $t$ is formulated as:

$$\sum_{j=1}^{m} x_{tj} \leq 180m$$  \hspace{1cm} (12)$$

The safe ROI constraint is constructed for the threshold $h_t$ for the campaign $t$ as follows:

$$\frac{\sum_{j=1}^{m} \mu_{C_{tj}} - \sqrt{\beta_t \sigma_{C_{tj}}}}{\sum_{j=1}^{m} x_{tj}} \geq h_t$$  \hspace{1cm} (13)$$
Lastly, the bid values $x_{tj}$ are bounded with non-negativity for all campaigns $t$ and for all advertisements $j$.

As opposed to the first experiment, where the changes in optimization goals were driven by changes in cooling demand, an increase in $t$ in the online bid optimization example is driven by the new input as starting a new campaign. We construct two GPs in this experiment, the first GP learns the bid prices from past observations, and the second one models the impressions, which are represented in binary for clicks. The impressions are traditionally predicted by classifiers due to their binary representation. However, it is possible to cast it as a regression problem where we decide the binary representations after thresholding. Since we use covariance functions of GPs to model uncertainty, we cast it as a regression and guide our RTO with continuous values. The optimization algorithm bids an ad comparatively high when its value is higher than others. Different feature sets in the dataset are used to compute posteriors based on relevance to the predictions. The GP of the bid price is initialized with the Matern kernel with $\nu = 1.5$ and is trained over 143 features whereas the impression GP has the Squared Exponential kernel and 69 features. The safe seeds start with 30 samples, which is higher than our first experiment since this is a higher dimensional problem. As a minimum ROI threshold, 90% of the given benchmark data ROI is set due to having a strict budget and ROI requirements in our setup. We partition the selected subset of the dataset into 25 campaigns. Each campaign has its ROI threshold and budget, which are calculated as Equation (12) and Equation (13).

The algorithm starts with a safe seed set to compute the posteriors of GPs, and then for each campaign, it utilizes the mean and standard deviation of GP posteriors to measure ROI and click probability. During the RTO phase, the higher bid prices for higher estimated click values are encouraged within a fixed budget, and the difference between the predicted bid price by GP and the proposed bid price by RTO for each advertisement is accumulated and introduced as a penalty in the objective function. Thus, the algorithm does not put the entire budget into the highest-valued ad within the campaign. At the end of each campaign, true bid prices and clicks with additive Gaussian noise are used to update the posteriors of GPs. The feedback is given only for ads in the campaign with a non-negative optimized bid price which leads to high standard deviations for non-bid similar ads. The environment becomes available for exploration after spending less than the sum of predicted bid prices and satisfying minimum thresholds in two consecutive campaigns. Hence, $z$ is set to $\zeta = 100$ to excite the RTO to take decisions at points that could reduce uncertainty in predictions. The results of the simulation are given in Figure 7 and Figure 8. Our approach remains above the safety threshold while proposing lower bid prices compared to the benchmark dataset.

5 CONCLUSIONS

In this work, we study real-time optimization of sequential-decision making in a safety-critical system under uncertainty due to unknown parameters. We propose a safe exploration and optimization algorithm which follows the optimization goals and does exploration without violating the constraints. We adaptively control the contribution of exploration via the uncertainty term in the utility function. We demonstrate the applicability of our approach to real-world datasets with industrial process optimization and online bid optimization. Our algorithm is able to model the unknown parameters in both applications and satisfy the optimization goals and safety constraints. We believe our study provides important examples to show how uncertainty in a safety-critical system can be modelled and guide the exploration.

References

Agrawal, S. and Goyal, N. (2013). Thompson sampling for contextual bandits with linear payoffs. In Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28, ICML’13, page III–1220–III–1228. JMLR.org.
Berkenkamp, F., Schoellig, A. P., and Krause, A. (2016). Safe controller optimization for quadrotors with gaussian processes. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 491–496.

Bubeck, S., Cesa-Bianchi, N., et al. (2012). Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122.

Bubeck, S., Munos, R., Stoltz, G., and Szepesvári, C. (2009). Online optimization in x-armed bandits. *Advances in Neural Information Processing Systems 21 - Proceedings of the 2008 Conference*, pages 201–208.

Castiglioni, M., Nuara, A., Romano, G., Spadaro, G., Trovò, F., and Gatti, N. (2022). Safe online bid optimization with return-on-investment and budget constraints subject to uncertainty. *ArXiv*, abs/2201.07139.

Chen, J., Cao, N., Low, K. H., Ouyang, R., Tan, C., and Jaillet, P. (2013). Parallel gaussian process regression with low-rank covariance matrix approximations. *Uncertainty in Artificial Intelligence - Proceedings of the 29th Conference, UAI 2013*.

Câmara, M. M., Quelhas, A. D., and Pinto, J. C. (2016). Performance evaluation of real industrial rtos systems. *Processes*, 4(4).

Dani, V., Hayes, T., and Kakade, S. (2008). Stochastic linear optimization under bandit feedback. pages 355–366.

Fu, J. and Levine, S. (2021). Offline model-based optimization via normalized maximum likelihood estimation. *ArXiv*, abs/2102.07970.

Fu, L., Ding, G., and Zhang, C. (2003). Dynamic simulation of air-to-water dual-mode heat pump with screw compressor. *Applied Thermal Engineering*, 23(13):1629–1645.

Hanche-Olsen, H. and Holden, H. (2010). The kolmogorov–riesz compactness theorem. *Expositiones Mathematicae*, 28(4):385–394.

Hensman, J., Fusi, N., and Lawrence, N. D. (2013). Gaussian processes for big data. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, UAI’13, page 282–290, Arlington, Virginia, USA, AUAI Press.

Kabzan, J., Hewing, L., Liniger, A., and Zeilinger, M. N. (2019). Learning-based model predictive control for autonomous racing. *IEEE Robotics and Automation Letters*, 4(4):3363–3370.

Lai, T. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22.

Lattimore, T. and Szepesvári, C. (2020). *Stochastic Bandits with Finitely Many Arms*, page 73–74. Cambridge University Press.

Liu, H., Ong, Y., Shen, X., and Cai, J. (2020). When gaussian process meets big data: A review of scalable gps. *IEEE Transactions on Neural Networks and Learning Systems*, PP:1–19.

Manske, K. A., Reindl, D., and Klein, S. (2000). Load sharing strategies in multiple compressor refrigeration systems.

Naysmith, M. R. and Douglas, P. L. (2008). Review of real time optimization in the chemical process industries. *Developments in Chemical Engineering and Mineral Processing*, 3:67–87.

Petsagkourakis, P., Chachuat, B., and Antonio del Rio-Chanona, E. (2021). Safe real-time optimization using multi-fidelity gaussian processes. In *2021 60th IEEE Conference on Decision and Control (CDC)*, page 6734–6741. IEEE Press.

Potra, F. A. and Wright, S. J. (2000). Interior-point methods. *Journal of Computational and Applied Mathematics*, 124(1):281–302. Numerical Analysis 2000. Vol. IV: Optimization and Nonlinear Equations.

Rasmussen, C. E. (2004). *Gaussian Processes in Machine Learning*, pages 63–71. Springer Berlin Heidelberg, Berlin, Heidelberg.

Ray Chowdhury, S. and Gopalan, A. (2017). On kernelized multi-armed bandits.

Rusnock, P. and Kerr-Lawson, A. (2005). Bolzano and uniform continuity. *Historia Mathematica*, 32(3):303–311.

Schittkowski, K. (1986). Nlpql: A fortran subroutine solving constrained nonlinear programming problems. *Annals of Operations Research*, 5:485–500.

Scholkopf, B. and Smola, A. J. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA.

Schreiter, J., Nguyen-Tuong, D., Eberts, M., Bischoff, B., Markert, H., and Toussaint, M. (2015). Safe exploration for active learning with gaussian processes. pages 133–149.

Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. W. (2010). Gaussian process optimization in the bandit setting: No regret and experimental design. In *ICML*.

Sriperumbudur, B. K., Fukumizu, K., and Lanckriet, G. R. (2011). Universality, characteristic kernels and rkhs embedding of measures. *Journal of Machine Learning Research*, 12(70):2389–2410.

Sui, Y., Gotovos, A., Burdick, J. W., and Krause, A. (2015). Safe exploration for optimization with gaussian processes. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML’15, page 997–1005. JMLR.org.
Sui, Y., Zhuang, V., Burdick, J., and Yue, Y. (2018). Stage-wise safe bayesian optimization with gaussian processes. In 35th International Conference on Machine Learning, pages 4781–4789. PMLR.

Turchetta, M., Berkenkamp, F., and Krause, A. (2016). Safe exploration in finite markov decision processes with gaussian processes. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R., editors, Advances in Neural Information Processing Systems, volume 29. Curran Associates, Inc.

Turchetta, M., Berkenkamp, F., and Krause, A. (2019). Safe Exploration for Interactive Machine Learning. Curran Associates Inc., Red Hook, NY, USA.

Wachi, A., Sui, Y., Yue, Y., and Ono, M. (2018). Safe exploration and optimization of constrained mdps using gaussian processes. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI’18/IAAI’18/EAAI’18. AAAI Press.

Widell, K. and Eikevik, T. (2010). Reducing power consumption in multi-compressor refrigeration systems. International Journal of Refrigeration, 33(1):88–94.

Zhang, D., Wang, K., Xu, Z., Tula, A. K., Shao, Z., Zhang, Z., and Biegler, L. T. (2022). Generalized parameter estimation method for model-based real-time optimization. Chemical Engineering Science, 258:117754.

Zhang, W., Yuan, S., and Wang, J. (2014). Real-time bidding benchmarking with ipinyou dataset.