Think Global and Act Local: Bayesian Optimisation over High-Dimensional Categorical and Mixed Search Spaces

Xingchen Wan [1] **Vu Nguyen** [2] **Huong Ha** [3] **Binxin Ru** [1] **Cong Lu** [1] **Michael A. Osborne** [1]

Abstract

High-dimensional black-box optimisation remains an important yet notoriously challenging problem. Despite the success of Bayesian optimisation methods on continuous domains, domains that are categorical, or that mix continuous and categorical variables, remain challenging. We propose a novel solution – we combine local optimisation with a tailored kernel design, effectively handling high-dimensional categorical and mixed search spaces, whilst retaining sample efficiency. We further derive convergence guarantee for the proposed approach. Finally, we demonstrate empirically that our method outperforms the current baselines on a variety of synthetic and real-world tasks in terms of performance, computational costs, or both.

1. Introduction

Bayesian Optimisation (BO) (Jones et al., 1998; Brochu et al., 2010; Shahriari et al., 2016), which features expressive surrogate model(s) and sample efficiency, has found many applications in black-box optimisation, particularly when each evaluation is expensive. Such applications include but not limited to selection of chemical compounds (Hernández-Lobato et al., 2017), reinforcement learning (Parker-Holder et al., 2020), hyperparameter optimisation of machine learning algorithms (Snoek et al., 2012), and neural architecture search (Kandasamy et al., 2018; Nguyen et al., 2021; Ru et al., 2021)

Despite its impressive performance, various challenges still remain for BO. The popular surrogate choice, vanilla Gaussian Process (GP) models, is limited to problems of modest dimensionality defined in a continuous space. However, real-world optimisation problems are often neither low-dimensional nor continuous: many large-scale practical problems exhibit complex interactions among high-dimensional input variables, and are often *categorical* in nature or involve a mixture of both continuous and categorical input variables. An example of the former is the maximum satisfiability problem, whose exact solution is NP-hard (Creignou et al., 2001), and an example for the latter is the hyperparameter tuning for a deep neural network: the optimisation scope comprise both continuous hyperparameters, e.g., learning rate and momentum, and categorical ones, e.g., optimiser type {SGD, Adam, ...} and learning rate scheduler type {step decay, cosine annealing}.

These problems are challenging for a number of reasons: first, categorical variables do not have a natural ordering similar to continuous ones for which GPs are well-suited. Second, the search space grows exponentially with the dimension and the mixed spaces are usually high-dimensional, making the objective function highly multimodal, often heterogeneous, and thus difficult to be modelled by a good, global surrogate (Rana et al., 2017; Eriksson et al., 2019). Partially due to these difficulties, only very few prior works (Hutter et al., 2011; Gopakumar et al., 2018; Nguyen et al., 2020; Ru et al., 2020a) have focused on developing BO strategies for such problems, and, to the best of our knowledge, achieving promising performance, easy applicability for high-dimensional inputs and reasonable computing costs simultaneously is still an open question.

To tackle these challenging yet important problems, we propose a novel yet conceptually simple method. It not only fully preserves the merits of GP-based BO approaches, such as expressiveness and sample efficiency, but also demonstrates state-of-the-art performance in high-dimensional optimisation problems, involving categorical or mixed search spaces. Specifically, we make the following contributions:

- Propose a new GP-based BO approach which designs tailored GP kernels and harnesses the concept of local trust region to effectively handle high-dimensional optimisation over categorical and mixed search spaces.
- Derive convergence analysis to show that our proposed method converges to the global maximum of the objective function in both categorical and mixed space settings, under some assumptions.

[1]Machine Learning Research Group, University of Oxford, Oxford, UK [2]Amazon, Adelaide, Australia [3]RMIT University, Melbourne, Australia. Correspondence to: Xingchen Wan <xwan@robots.ox.ac.uk>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

- Empirically show that our method achieves superior performance, better sample efficiency, or both, over the existing approaches for a wide variety of tasks. The code implementation of our method is available at https://github.com/xingchenwan/Casmopolitan.

2. Related work

BO for high-dimensional problems A popular class of high-dimensional BO methods (Kandasamy et al., 2015; Rolland et al., 2018; Wang et al., 2017; 2018; Mutnỳ and Krause, 2019) decompose the search space into multiple overlapping or disjoint low-dimensional subspaces and use an additive surrogate (e.g. additive GPs). However, accurately inferring the decomposition is often very expensive. Another group of BO methods (Binois et al., 2015; Wang et al., 2016; Binois et al., 2020) assume the objective function is mainly influenced by a small subset of effective dimensions and aims to learn such low-dimensional effective embedding (Wang et al., 2016; Nayebi et al., 2019; Letham et al., 2020). However, its effectiveness is conditional on the extent the assumption holds. A recent state-of-the-art approach is Trust-region Bayesian Optimisation (TURBO) (Eriksson et al., 2019), which constrains BO on local Trust Region (TR) centered around the best inputs so far. This circumvents the aforementioned issues such as the need for finding an accurate global surrogate and over-exploration due to large regions of high posterior variance. However, its convergence properties are not analysed, and it only works in continuous spaces.

BO for categorical search spaces The basic approach is to one-hot transform the categorical variables into continuous (Rasmussen, 2006; GPyOpt, 2016; Snoek et al., 2012). While simple in implementation, the drawbacks are equally obvious: first, for a d_h-dimensional problems with $\{n_1, ..., n_{d_h}\}$ choices per input, the one-hot-transformed problem has $\sum_{i=1}^{d_h} n_i$ dimensions, further aggravating the curse of dimensionality. Second, categorical spaces differ fundamentally with the continuous in, for e.g., differentiability and continuity, with function values only defined in finite locations. These lead to difficulties in using gradient-based methods in acquisition function optimisation of the transformed problems.

To ameliorate these drawbacks, BOCS (Baptista and Poloczek, 2018) first tailors BO in categorical spaces: it uses a sparse monomial representation up to the second order and Bayesian linear regression as the surrogate, and is primarily used for boolean optimisation. Inevitably, its expressiveness is constrained by the quadratic model, while scaling beyond the second order and/or to high dimensionality is usually intractable due to the exponentially-increasing number of parameters that need be learnt explicitly. Combinatorial Bayesian Optimisation (COMBO) (Oh et al., 2019) is a state-of-the-art method that instead uses a GP surrogate (which is capable of learning interactions of an arbitrary order), and is capable of dealing with multi-categorical problems via a combinatorial graph over all possible joint assignments of the variables and a diffusion graph kernel to model the interactions. Nonetheless, both methods deal with categorical optimisation only, which is an important problem in its own right, but does not extend to our setting of mixed-variable problems. They also suffer from poor scalability (e.g. to avoid overfitting COMBO approximately marginalises the posterior via Monte Carlo sampling instead of cheaper optimisation, and it needs to pre-compute the combinatorial graph beforehand). Other methods, such as COMEX and its inspired works (Dadkhahi et al., 2020; 2021) take a non-Bayesian black-box optimisation approach to improve computing efficiency, but they are typically less sample-efficient with respect to the number of function queries and are less suitable for problems where querying the objective functions is expensive. Finally, several recent works aim to improve BO on combinatorial structures by improving the effectiveness (Deshwal et al., 2020) or reducing the expenses (Swersky et al., 2020) of the *acquisition function*; these are largely orthogonal to our method, and we defer a thorough investigation on whether there are additional benefits by combining with these methods to a future work.

BO for mixed input types BO in mixed categorical-continuous search spaces is still rather under-explored, despite attempts in modelling less complicated spaces, such as mixed continuous-integer problems (Daxberger et al., 2019; Garrido-Merchán and Hernández-Lobato, 2020). In our specific setting, Categorical and Continuous Bayesian Optimisation (COCABO) (Ru et al., 2020a) first explicitly handles multiple categorical and continuous variables: it alternates between selecting the categorical inputs with a Multi-Armed Bandit (MAB) and the continuous inputs with GP-BO, and uses a tailored kernel to connect the two. However, COCABO requires optimising a MAB over a non-stationary reward (since the values of continuous variables improves over BO iterations and hence so does the function value). Furthermore, MAB requires pulling each arm at least once, and hence it is difficult to scale COCABO to high-dimensional problems, where the total number of possible arm combinations explode exponentially. Lastly, while the two sub-components are provably convergent, COCABO as a whole is not. Related works along this direction also include Gopakumar et al. (2018) and Nguyen et al. (2020), but the continuous inputs are constrained to be *specific* to the categorical choice, and being MAB-based, it also suffers from aforementioned limitations. Separately, Bliek et al. (2020) recently propose Mixed-Variable ReLU-based Surrogate Modelling (MVRSM), which the authors claim to be suitable for mixed-variable, high-dimensional problems. However,

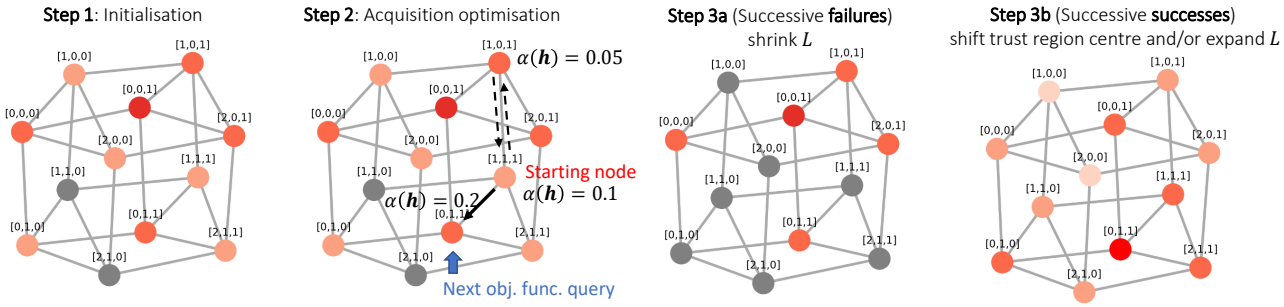

Figure 1. Illustration of CASMOPOLITAN in categorical space. Suppose we optimise over a 3-dimensional problem with $\{3, 2, 2\}$ choices for each input respectively. Initially (**Step 1**), the best location so far $\mathbf{h}_T^* = \arg\max_{\mathbf{h}} \{y_j\}_{j=1}^T$ (marked in red) is $[0, 0, 1]$ with TR radius $L = 2$ (the orange nodes, with different shades denoting their Hamming distances to $\mathbf{h}_T^*$. The gray nodes are outside the current TR). In optimisation of the acquisition function (**Step 2**), we conduct local search within the TR, moving to a neighbour only if it has a higher acquisition function value $\alpha(\cdot)$ and is still within the TR. In case of successive failures (**Step 3a**) in increasing $\mathbf{h}_T^*$, we shrink the TR down to length $L_{\min}^h$, below which we restart the optimisation, or in case of successive successes (**Step 3b**), we shift the TR centre to the new $\mathbf{h}_T^*$ and/or expand TR up to length $L_{\max}^h$. Note that the combinatorial graph is shown here for illustration; it does not need to be computed explicitly or otherwise.

in trading for efficiency, the expressiveness is limited by the ReLU formulation and we compare against it in Sec. 4.

In addition to these more recent works explicitly handling the mixed spaces, earlier attempts such as SMAC with Random Forest (RF) (Breiman, 2001) surrogates (Hutter et al., 2011) are also compatible. However, the predictive distribution of the RF used to select new evaluation is less accurate due to reliance on randomness from bootstrap samples and the randomly chosen subset of variables to be tested at each node to split the data. Moreover, RFs easily suffer from overfitting and require careful hyperparameter choice.

3. CASMOPOLITAN: BO for Categorical and Mixed Search Spaces

Problem Statement We consider the problem of optimising an expensive black-box function, defined over a categorical domain or one with mixed continuous and categorical inputs. Formally, we consider a function in the mixed domain for generality: $f : [\mathcal{H}, \mathcal{X}] \to \mathbb{R}$ where $\mathcal{H}$ and $\mathcal{X} \subset \mathbb{R}^{d_x}$ denote the categorical and continuous search spaces, respectively (for problems over categorical domains, we simply have $f : \mathcal{H} \to \mathbb{R}$ and the goal is to find $\mathbf{h}^* = \arg\max f(\mathbf{h})$).We further denote $\mathbf{z} = [\mathbf{h}, \mathbf{x}]$ to be an input in the mixed space where $\mathbf{h}$ and $\mathbf{x}$ are the categorical and continuous parts, d_h to be the number of categorical variables, i.e. $\mathbf{h} = [h_1, h_2, ..., h_{d_h}]$, and the number of possible, distinct value that the j-th categorical variable may take to be n_j. Given f, at time t we observe the noisy perturbation of the form $y_t = f(\mathbf{z}_t) + \epsilon_t$ where $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$ and σ^2 is a noise variance which can be learned by maximizing the log-marginal likelihood (Rasmussen, 2006). We sequentially select inputs $\mathbf{z}_t \; \forall t = 1, ..., T$ (or simply $\mathbf{h}_t$ if the problem is purely categorical) to query f with the goal of finding the maximiser the objective $\mathbf{z}^* = \arg\max f(\mathbf{z})$ with the fewest numbers of iterations. We further include a primer on GP and BO in App. A.

Algorithm 1 CASMOPOLITAN.

Input: #init (the number of random initialing points at initialisation or restarts), #iter T, initial TR size for categorical $L_0^h \in \mathbb{Z}^+$, and continuous variables $L_0^x \in \mathbb{R}^+$.
Output: The best recommendation $\mathbf{z}_T$
`restart` = `True` // *Set restart to True initially*
for $t = 1, \ldots, T$ **do**
  if `restart` **then**
    Reset TR $L^h = L_0^h$ and $L^x = L_0^x$ and reset GP. Randomly select #init points in the search space as $\mathbf{z}_t$ *(if at initialisation)*, or set the TR center as the point determined by Eq. (3) and randomly select #init points within the newly constructed TR as $\mathbf{z}_t$ *(if at subsequent restarts)*.
  else
    Construct a TR $\text{TR}_{\mathbf{h}}(\mathbf{h}_t^*)$ around the categorical dimensions of the best point $\mathbf{h}_t^*$ using Eq. (2).
    Construct a hyper-rectangular TR of length L^x, $\text{TR}_{\mathbf{x}}(\mathbf{x}_t^*)$ for the continuous variables.
    Select next query pt(s) *within the* TRs $\mathbf{z}_t = \text{argmax}_{\mathbf{z}} \alpha(\mathbf{z})$ s.t. $\mathbf{x} \in \text{TR}_x(\mathbf{x}_t^*), \mathbf{h} \in \text{TR}_h(\mathbf{h}_t^*)$.
  end if
  Query at $\mathbf{z}_t$ to obtain y_t; fit/update the surrogate $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup (\mathbf{z}_t, y_t)$ and optimise GP hyperparameters.
  Update the TRs and decide whether to `restart`.
end for

3.1. Categorical Search Space

Our first contribution is to propose a conceptually-simple yet effective BO strategy that preserves all of the advantages of GP modelling, but is specifically designed for the categorical search space (later extended to the mixed space in Sec. 3.2). We present an illustration in Fig. 1 and the pseudocode in Algorithm 1. We name our algorithm CASMOPOLITAN

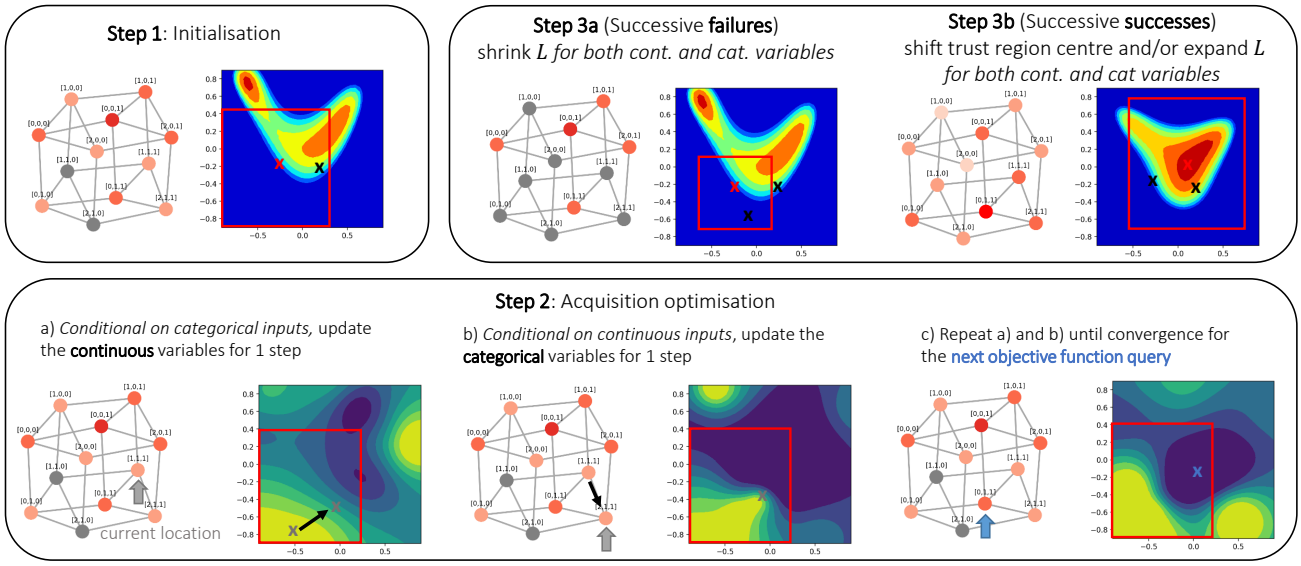

Figure 2. Illustration of CASMOPOLITAN in mixed space. Note that in **Steps 1** & **3** we show the GP *posterior* on $\mathcal{X}$ conditioned on the incumbent $\mathbf{h}_\mathbf{T}^*$, and in **Step 2** we show the *acquisition function* on $\mathcal{X}$ conditioned on $\mathbf{h}$ at various optimisation steps. Suppose we optimise over a 5-dimensional mixed problem with the categorical dimensions identical to that in Fig. 1 and 2 additional continuous dimensions. Initially (**Step 1**), the best location so far $\mathbf{z}_T^* = \arg\max_{\mathbf{z}}\{y_j\}_{z=1}^T = [\mathbf{h}_\mathbf{T}^*, \mathbf{x}_\mathbf{T}^*]$ (with the continuous TR and $\mathbf{x}_T^*$ in red box and cross). In optimisation of acquisition function (**Step 2**), we interleave the local search on $\mathcal{H}$ described in Sec. 3.1 with gradient-based optimisation on $\mathcal{X}$ until convergence. In **Steps 3a/3b**, we adjust both the continuous and categorical TRs correspondingly and restart if/when either shrinks below its minimum length.

(CAtegorical Spaces, or Mixed, OPtimisatiOn with Local-trust-regIons & TAilored Non-parametric), and we highlight the key design features in this section.

Kernel design In Line 12 of Algorithm 1, we impose GP on the categorical variables with a kernel defined directly on them (note that it does not increase the dimensions like one-hot transform). Specifically, we modify the overlap (or Hamming) kernel $k(\mathbf{h}, \mathbf{h}') = \frac{\sigma}{d_h}\sum_{i=1}^{d_h}\delta(h_i, h_i')$, in Ru et al. (2020a) and Kondor and Lafferty (2002):

$$k_h(\mathbf{h}, \mathbf{h}') = \exp\left(\frac{1}{d_h}\sum_{i=1}^{d_h}\ell_i\delta(h_i, h_i')\right), \qquad (1)$$

where $\{\ell_i\}_i^{d_h}$ are the lengthscale(s)[1], and $\delta(\cdot,\cdot)$ is the Kronecker delta function. The modification affords additional expressiveness in modelling more complicated functions: for e.g., the kernel in Eq. (1) can discern the dimensions to which the objective function value is more sensitive via learning different lengthscales but the original categorical overlap kernel treats all dimensions equally. We empirically validate the performance gain of the exponentiated kernel in Sec. 4.4, and we prove this kernel is positive semi-definite (p.s.d) in App. D.1.

[1]The lengthscales will be different for each dimension if we enable automatic relevance determination (ARD).

Trust region One key difficulty in applying GP-BO in high-dimensional problems is that the surrogate, by default, attempts to model the entire function landscape and over-explores. Optimisation over the categorical search space also suffer this problem. To effectively scale up the dimensions, we adapt the TR approach from Eriksson et al. (2019) in categorical search space (Line 8 in Algorithm 1). However, the challenge is that the Euclidean distance-based TR is no longer applicable; instead, we define TRs in terms of *Hamming distance*, i.e. a TR of radius L^h from the best location, $\mathbf{h}^*$, observed at iteration T includes all points that are up to L^h variables different from $\mathbf{h}^*$:

$$\mathrm{TR}_h(\mathbf{h}^*)_{L^h} = \left\{\mathbf{h} \mid \sum_{i=1}^{d_h}\delta(h_i, h_i^*) \leq L^h\right\}. \qquad (2)$$

The TR radius is adjusted dynamically during optimisation, expanding on successive successes (if best function value f_T^* improves) and shrinking otherwise. Since Hamming distance is integer-valued bounded in $[0, d_h]$, we also set these two values as the minimum and maximum TR lengths $L^h_{\min}$ and $L^h_{\max}$.

TRs in local optimisation are typically biased toward the starting points. Therefore, most local optimisation approaches rely on a restarting strategy to attain good performance (Shylo et al., 2011; Kim and Fessler, 2018). In our case, we restart the optimisation when the TR length L^h reaches the smallest possible value (Line 13 in Algorithm 1).

Rather than restarting randomly as in Eriksson et al. (2019), we propose to restart our method using GP-UCB principle (Srinivas et al., 2010), which as we will show in Section 3.3 is crucial for theoretical guarantee. Specifically, we introduce an auxiliary global GP model to achieve this. Suppose we are restarting the i-th time, we first fit the global GP model on a subset of data $D^*_{i-1} = \{\mathbf{h}^*_j, y^*_j\}_{j=1}^{i-1}$, where $\mathbf{h}^*_j$ is the local maxima found after the j-th restart. Alternative, a random data point, if the found local maxima after the j-th restart is same as one of previous restart. Let us also denote $\mu_{gl}(\mathbf{h}; D^*_{i-1})$ and $\sigma^2_{gl}(\mathbf{h}; D^*_{i-1})$ as the posterior mean and variance of the global GP learned from D^*_{i-1}. Then, at the i-th restart, we select the following location $\mathbf{h}^{(0)}_i$ as the initial centre of the new TR:

$$\mathbf{h}^{(0)}_i = \arg\max_{\mathbf{h}\in\mathcal{H}} \mu_{gl}(\mathbf{h}; D^*_{i-1}) + \sqrt{\beta_i}\sigma_{gl}(\mathbf{h}; D^*_{i-1}), \quad (3)$$

where β_i is the trade-off parameter. As formally shown in Sec. 3.3, this strategy is optimal in deciding the next TR by balancing exploration against exploitation (Srinivas et al., 2010). Finally, while the use of UCB-restart is primarily theoretically driven, we show that it could offer empirical benefits over random restarts, and the readers are referred to App. B for details.

Optimisation of the acquisition function Since we preserve the discrete nature of the variables in our method, we cannot optimise the acquisition function via gradient-based methods. Instead, we use the simple strategy of local search within the TRs defined previously: at each BO iteration, we randomly sample an initial configuration $\mathbf{h}_0 \in \mathrm{TR}_h(\mathbf{h}^*)$. We then randomly select a neighbour point of Hamming distance 1 to $\mathbf{h}_0$, evaluate its acquisition function $\alpha(\cdot)$, and move from $\mathbf{h}_0$ if the neighbour has a higher acquisition function value and is still within the TR. We repeat this process until a pre-set budget of queries is exhausted and dispatch the best configurations for objective function evaluation (Line 10 in Algorithm 1).

3.2. Extension to Mixed Search Spaces

In addition to the purely categorical problems, our method naturally generalises to mixed, and potentially high-dimensional, categorical-continuous spaces, a setting frequently encountered in real life but hitherto under-explored in BO literature. To handle such an input $\mathbf{z} = [\mathbf{h}, \mathbf{x}]$ where $\mathbf{x}$ is the continuous inputs, we first modify the GP kernel to the one proposed in Ru et al. (2020a):

$$k(\mathbf{z}, \mathbf{z}') = \lambda\Big(k_x(\mathbf{x}, \mathbf{x}')k_h(\mathbf{h}, \mathbf{h}')\Big) + (1-\lambda)\Big(k_h(\mathbf{h}, \mathbf{h}') + k_x(\mathbf{x}, \mathbf{x}')\Big), \quad (4)$$

where $\lambda \in [0, 1]$ is a trade-off parameter, k_h is defined in Eq. (1) and k_x is a kernel over continuous variables (we use the Matérn 5/2 kernel). While we use the same kernel as Ru et al. (2020a), we emphasise and formally show in Sec. 3.3 that, unlike COCABO, CASMOPOLITAN retains convergence guarantee even in the mixed space.

This formulation therefore allows us to use tailored kernels that are most appropriate for the different input types while still flexibly capturing the possible additive and multiplicative interactions between them. For the continuous inputs, we use a standard TURBO surrogate (Eriksson et al., 2019) by maintaining, and adjusting where necessary, separate standard hyper-rectangular TR(s) for them $\mathrm{TR}_x(\mathbf{x}^*)_L = \{\mathbf{x} |\ \mathbf{x} \in \mathcal{X}$ and within the box centered around $\mathbf{x}^*\}$. We include an illustration in Fig. 2. We restart the continuous TR TR_x in similar manner as described in Eq. (3), if and when the either TR_h or TR_x length L reaches the smallest possible value.

Interleaved acquisition optimisation In Ru et al. (2020a), the categorical $\mathbf{h}$ and continuous $\mathbf{x}$ of the proposed points $\mathbf{z} = [\mathbf{h}, \mathbf{x}]$ are optimised separately similar to a *single* EM-style iteration: the categorical parts are first proposed by the multi-armed bandit; *conditioned on these*, the continuous parts are then suggested by optimising the acquisition function. In our approach, since both the categorical and continuous inputs are handled by a single, unified GP, we may propose points and optimise acquisition functions more naturally and effectively: at each optimisation step (Line 10 of Algorithm 1), we simply do one step of local search defined in Sec. 3.1 on the categorical variables, followed by one step of gradient-based optimisation of the acquisition function on the continuous variables. However, instead of doing this alternation once, we repeat until convergence or when a maximum number of steps is reached.

Other types of discrete input While we mainly focus on categorical-continuous problems, our method can be easily generalised to more complex settings by virtue of its highly flexible sub-components. For instance, we often encounter combinatorial variables with ordinal relations: for these, we treat them as categorical, but instead of using Kronecker delta function in Eq. (1) we encode the problem-specific distances. We defer a full investigation to a future work, but we include some preliminary studies in App. B.4.

3.3. Theoretical Analysis

We first provide upper bounds on the maximum information gains of our proposed categorical kernel in Eq. (1) and mixed kernel in Eq. (4) (Theorem 3.1). We then prove that after a restart, under Assumptions 3.1 and 3.2, CASMOPOLITAN converges to a local maxima after a finite number of iterations or converges to the global maximum (Theorem 3.2). Finally, we prove that with our UCB-restart strategy, under Assumptions 3.1, 3.2 and some assumptions described

in Srinivas et al. (2010), CASMOPOLITAN converges to the global maximum with a sublinear rate over the number of restarts in both categorical (Theorem 3.3) and mixed space settings (Theorem 3.4). We refer readers to App. D for the detailed proofs.

Theorem 3.1. *Let us define* $\gamma(T; k; V) := \max_{A \subseteq V, |A| \leq T} \frac{1}{2} \log |I + \sigma^{-2}[k(\mathbf{v}, \mathbf{v}')]_{\mathbf{v},\mathbf{v}' \in A}|$ *as the maximum information gain achieved by sampling* T *points in a* GP *defined over a set* V *with a kernel* k*. Let us define* $\tilde{N} := \prod_{j=1}^{d_h} n_j$*, then we have,*

1. *For the categorical kernel* k_h, $\gamma(T; k_h; \mathcal{H}) = \mathcal{O}(\tilde{N} \log T)$*;*
2. *For the mixed kernel* k, $\gamma(T; k; [\mathcal{H}, \mathcal{X}]) \leq \mathcal{O}\big((\lambda\tilde{N} + 1 - \lambda)\gamma(T; k_x; \mathcal{X}) + (\tilde{N} + 2 - 2\lambda) \log T\big)$.

Using Theorem 3.1, the maximum information gain of the mixed kernel k can be upper bounded for some common continuous kernels k_x. For instance, when k_x is the Matérn kernel, the maximum information gain $\gamma(T; k; [\mathcal{H}, \mathcal{X}])$ of the mixed kernel is upper bounded by $\mathcal{O}\big((\lambda\tilde{N} + 1 - \lambda)T^{d_x(d_x+1)/(2v+d_x(d_x+1))}(\log T) + (\tilde{N} + 2 - 2\lambda) \log T\big)$ as $\gamma(T; k_{Mt}; \mathcal{X}) = \mathcal{O}(T^{d_x(d_x+1)/(2v+d_x(d_x+1))}(\log T))$ (Srinivas et al., 2010). Similar bounds can be established when k_x is the squared exponential or the linear kernel.

To analyse the convergence property of CASMOPOLITAN, similar to any TR-based algorithm (Yuan, 2000), we assume that (i) f is bounded in $[\mathcal{H}, \mathcal{X}]$ (Assumption 3.1), and (ii), given a small enough region, the surrogate model (i.e. GP) accurately approximates f with any data point belonging to this region (Assumption 3.2). We note that Assumption 3.1 is common as it is generally assumed in BO that f is Lipschitz continuous (Brochu et al., 2010), thus f is bounded given the search space is bounded. Assumption 3.2 considers the minimum TR lengths $L^x_{\min}, L^h_{\min}$ are set to be small enough so that GP approximates f accurately in TRs specified in Assumption 3.2. We note that in practice, this assumption is only possible asymptotically, i.e. when the number of observed data in these TRs goes to infinity. In our implementation (see App. C), these TRs are always set to be very small so that Assumption 3.2 can be close to true.

Assumption 3.1. *The objective function* $f(\mathbf{z})$ *is bounded in* $[\mathcal{H}, \mathcal{X}]$*, i.e.* $\exists F_l, F_u \in \mathbb{R} : \forall \mathbf{z} \in [\mathcal{H}, \mathcal{X}]$, $F_l \leq f(\mathbf{z}) \leq F_u$.

Assumption 3.2. *Let us denote* $L^h_{\min}$, $L^x_{\min}$ *and* L^h_0, L^x_0 *be the minimum and initial* TR *lengths for the categorical and continuous variables, respectively. Let us also denote* α_s *as the shrinking rate of the* TRs*. In the categorical setting, for any* TR *with length* $\leq \lceil (L^h_{\min} + 1)/\alpha_s \rceil - 1$,[2] *the corresponding local* GP *approximates* f *accurately. That is, the* GP *posterior mean approximates* f *accurately whilst the* GP *posterior variance is negligible within this* TR*. In the mixed space setting, the local* GP *approximates* f *accurately within any* TR *with length* $L^x \leq \max\big(L^x_{\min}/\alpha_s, L^x_0(\lceil (L^h_{\min} + 1)/\alpha_s \rceil - 1)/L^h_0\big)$ *and* $L^h \leq \max\big(\lceil (L^h_{\min} + 1)/\alpha_s \rceil - 1, \lceil L^h_0 L^x_{\min}/(\alpha_s L^x_0) \rceil\big)$.

Theorem 3.2. *Given Assumptions 3.1 & 3.2, after a restart,* CASMOPOLITAN *converges to a local maxima after a finite number of iterations or converges to the global maximum.*

Finally, we define the cumulative regret after I restarts, R_I, to be $\sum_{j=1}^{I}(f(\mathbf{z}^*) - f(\mathbf{z}^*_j))$ with $\mathbf{z}^*_j$ being the local maxima found at the j-th restart and $\mathbf{z}^*$ being the global maximum of f. We then provide the regret bounds of CASMOPOLITAN in both categorical (Theorem 3.3) and mixed space setting (Theorem 3.4). With these regret bounds, it can be seen that CASMOPOLITAN converges to the global maximum with a sublinear rate over the number of restarts (i.e. $R_I/I \xrightarrow{I \to \infty} 0$) in both categorical and mixed space settings.

Theorem 3.3. *Let us consider the categorical setting,* $f : \mathcal{H} \to \mathbb{R}$*. Let* $\zeta \in (0, 1)$ *and* $\beta_i = 2\log(|\mathcal{H}|i^2\pi^2/6\zeta)$ *at the* i*-th restart. Suppose the objective function* f *satisfies that: there exists a class of functions which pass through all the local maxima of* f*,*[3] *share the same global maximum with* f*, and is sampled from the auxiliary global* GP $GP(0, k_h)$*. Then given Assumptions 3.1 & 3.2,* CASMOPOLITAN *obtains a regret bound of* $\mathcal{O}^*\big(\sqrt{I\gamma(I; k_h, \mathcal{H}) \log |\mathcal{H}|}\big)$ *w.h.p. Formally,*

$$Pr\Big\{R_I \leq \sqrt{C_1 I \beta_I \gamma(I; k_h, \mathcal{H})} \quad \forall I \geq 1\Big\} \geq 1 - \zeta,$$

with $C_1 = 8/\log(1 + \sigma^{-2})$, $\gamma(I; k_h, \mathcal{H}) = \mathcal{O}(\tilde{N} \log(\tilde{N}) \log(I))$ *and* $\tilde{N} = \prod_{j=1}^{d_h} n_j$.

Theorem 3.4. *Let us consider the mixed space setting,* $f : [\mathcal{H}, \mathcal{X}] \to \mathbb{R}$*. Let* $\zeta \in (0, 1)$*. Suppose the objective function* f *satisfies that: there exists a class of functions* g *which pass through all the local maximas of* f*, share the same global maximum with* f *and lies in the RKHS* $\mathcal{G}_k([\mathcal{H}, \mathcal{X}])$ *corresponding to the kernel* k *of the auxiliary global* GP *model. Suppose that the noise* ϵ_i *has zero mean conditioned on the history and is bounded by* σ *almost surely. Assume* $\|g\|^2_k \leq B$*, and let* $\beta_i = 2B + 300\gamma_i \log(i/\zeta)^3$*, then given Assumptions 3.1 & 3.2,* CASMOPOLITAN *obtains a regret bound of* $\mathcal{O}^*\big(\sqrt{I\gamma(I; k, [\mathcal{H}, \mathcal{X}])\beta_I}\big)$ *w.h.p. Specifically,*

$$Pr\Big\{R_I \leq \sqrt{C_1 I \beta_I \gamma(I; k; [\mathcal{H}, \mathcal{X}])} \quad \forall I \geq 1\Big\} \geq 1 - \zeta,$$

with $C_1 = 8/\log(1 + \sigma^{-2})$, $\gamma(I; k; [\mathcal{H}, \mathcal{X}]) = \mathcal{O}\big((\lambda\tilde{N} + 1 - \lambda)\gamma(T; k_x; \mathcal{X}) + (\tilde{N} + 2 - 2\lambda) \log T\big)$ *and* $\tilde{N} = \prod_{j=1}^{d_h} n_j$.

Discussion We show in Theorem 3.2 that our TR-based algorithm with BO converges to a local maxima or global maximum after a restart. We note that similar convergence can

[2] The operator $\lceil . \rceil$ denotes the ceiling function.

[3] This means for every function g belonging to this class of functions, $g(\mathbf{h}^*_j) = f(\mathbf{h}^*_j)$ where $\mathbf{h}^*_j$ is a local maxima of f.

be found in the original TR-based algorithms using gradient-descent (Yuan, 2000). However, our proof technique is very different from Yuan (2000). In addition, in Theorems 3.3 & 3.4, the fact that CASMOPOLITAN converges to the global maximum with a sublinear rate over the number of restarts - not over the number of iterations as in Srinivas et al. (2010) - can be considered as the price paid for a more relaxed assumption. In particular, Srinivas et al. (2010) assume that it is possible to model the objective function f with a GP with kernel k on the whole search space. On the other hand, we relax this assumption in Theorems 3.3 & 3.4 by assuming that there is a class of functions, which pass through the local maxima and share the same global maximum with f, that we can model with a GP with kernel k. Further details on this class of functions can be found in Apps. D.4 & D.5.

Despite the aforementioned strengths, there are some limitations with our theoretical analysis. First, the maximum information gains $\gamma(T; k_h; \mathcal{H})$ and $\gamma(T; k; [\mathcal{H}, \mathcal{X}])$ derived in Theorem 3.1 increase exponentially with the dimension of the categorical input (d_h). Thus, these terms can be large when the categorical dimension is high. As we are solving a noisy NP-hard combinatorial problem, it might not be possible to get away these exponential terms without a strict assumption. Second, as briefly discussed above, Assumption 3.2 is true asymptotically, resulting Theorems 3.2, 3.3 and 3.4 to hold asymptotically. One way to eliminate this assumption is to instead prove CASMOPOLITAN achieves ϵ - *accuracy*, that is, CASMOPOLITAN can find a point whose function value is within ϵ of the objective function global maximum, where ϵ is a small positive value depending on the minimum TR lengths $L^x_{\min}, L^h_{\min}$. We consider these directions for future work.

4. Experiments

4.1. Categorical Problems

We first evaluate our proposed method on a number of optimisation problems in the categorical search space against a number of competitive baselines, including TPE (Bergstra et al., 2011), SMAC (Hutter et al., 2011), BOCS (Baptista and Poloczek, 2018)[4] and COMBO (Oh et al., 2019) which claims the state-of-the-art performance amongst comparable algorithms. We also include two additional baselines: BO, which performs the naïve BO approach after converting the categorical variables into one-hot representations, and TuRBO, which is identical to BO except that we additionally incorporate the TR approach in Eriksson et al. (2019). We experiment on following real-life problems (for detailed implementation and descriptions for the setup of these problems and those in Sec. 4.2, see App. C).

- Contamination control over 25 binary variables (3.35×10^7 configurations). This problem and the Pest control problem below simulate the dynamics of real-life problems whose evaluations are extremely expensive (Hu et al., 2010).
- Pest control over 25 variables, with 5 possible options for each (2.98×10^{17} configurations) (Oh et al., 2019).
- Weighted maximum satisfiability (MAXSAT) problem over 60 binary variables (1.15×10^{18} configurations).

In all experiments in this section and Sec. 4.2, we report the sequential version (denoted as CASMOPOLITAN-1 as batch size $b = 1$) of our method as all baselines we consider are also sequential. We investigate the parallel version of varying batch sizes of our method in Sec. 4.3.

The results are shown in Fig. 3: our method achieves the best convergence speed and sample efficiency in general, and in terms of the performance at termination, our method again outperforms the rest except in Contamination and MAXSAT where it performs on par with COMBO. However, it is worth noting that in terms of wall-clock speed, our method is $2-3$ times faster than COMBO in the problems considered (See App. B).

4.2. Mixed Problems

We then consider the optimisation problems involving a mix of continuous and categorical input variables. In these experiments, in addition to SMAC, TPE, BO and TURBO described in Section 4.1, we also include a number of recent advancements in this setup including COCABO (Ru et al., 2020a) and MVRSM (Bliek et al., 2020). Additionally, we run a small comparison against several other high-dimensional BO methods such as ALEBO (Letham et al., 2020) and REMBO (Wang et al., 2016), and the readers are referred to details in App. B. Note that we do not compare against BOCS and COMBO since they are suitable for purely categorical spaces only. Under this setup, we consider the following synthetic and real-life problems of increasing dimensionality and complexity:

- Func2C with $d_h = 2$ and $d_x = 2$, and Func3C with $d_h = 3$ and $d_x = 3$, respectively (Ru et al., 2020a).
- Hyperparameter tuning of the XGBoost model (Chen and Guestrin, 2016) on the MNIST dataset (LeCun, 1998), with $d_x = 5$ and $d_h = 3$ with 2 choices for each.
- 53-dimensional Ackley function (Ackley-53) (Bliek et al., 2020) with $d_h = 50$ where $\mathbf{h} \in \{0, 1\}^{50}$ and $d_x = 3$ where $\mathbf{x} \in [-1, 1]^3$.
- Black-box adversarial attack on a CNN trained on CIFAR-10 inspired by Ru et al. (2020b), but with adapted *sparse* setups where we perturb a small number of pixels only. The task is an optimisation problem with $d_h = 43$ (42 pixel locations being attacked with $n_{1:42} = 14$ choices each and the image upsampling technique which has $n_{43} = 3$

[4] BOCS is only run in Contamination, as it by default does not support multi-categorical optimisation and on Weighted Maximum Satisfiability (MAXSAT), a single trial takes more than 100 hours, rendering comparison infeasible within our computing constraints.

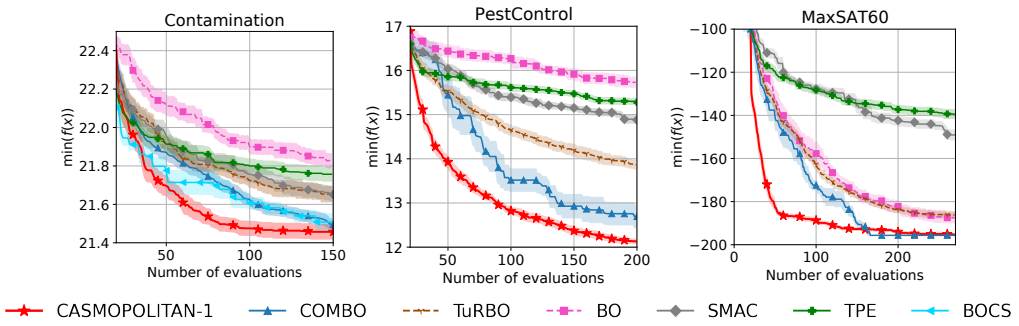

Figure 3. Results on various categorical optimisation problems. Lines and shaded area denote mean $\pm$ 1 standard error.

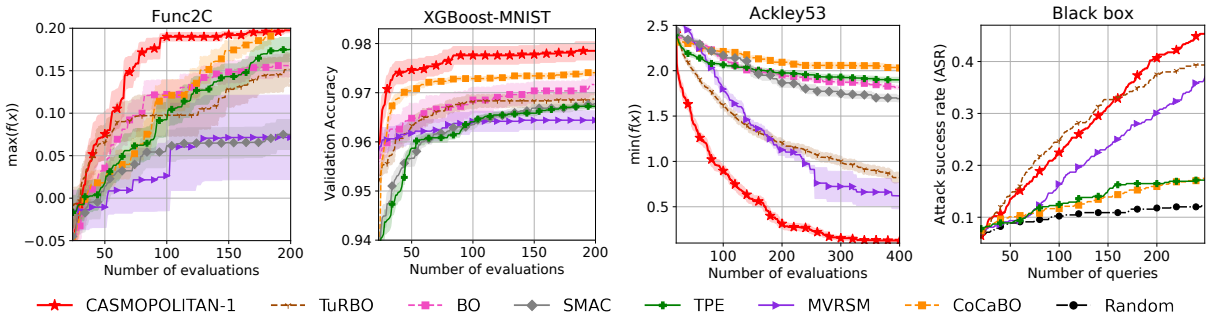

Figure 4. Results on various mixed optimisation problems. Lines and shaded area denote mean $\pm$ 1 standard error (except for Black-box where we show the ASR against number of queries). Additional experiment results in App. B.

choices) and $d_x = 42$ for continuous perturbation added to each pixel under attack. We perform a total of 450 targetted attack instances and limit the maximum budget to be 250 queries for each attack to simulate a highly constrained attack setup.

We report the results on the objective function values in Fig. 4 except for the black-box attack, where we instead report the attack success rate ASR against the number of queries following Ru et al. (2020b) (Additional attack results are shown in App. B). In this problem we also compare against random search, as it has been shown to be a strong baseline both in adversarial attack (Croce et al., 2020) and high-dimensional black-box optimisation (Rana et al., 2017) literature. Overall, it is evident that CASMOPOLITAN performs the best, but it is also interesting to observe that in lower dimensions (the first 2 problems), COCABO featuring tailored categorical kernels performs well, while MVRSM and categorical variable-agnostic TURBO, both focusing on high dimensions, under-perform. However, in high-dimensional problems (last 2 problems), the relative performance switches completely, suggesting that the focus on dimensionality now outweighs the importance of treating different input types differently. Nonetheless, with *both* tailored kernels and focus on scaling to high dimensions, CASMOPOLITAN consistently out-performs by a comfortable margin, further demonstrating its versatility.

4.3. Parallel Setting

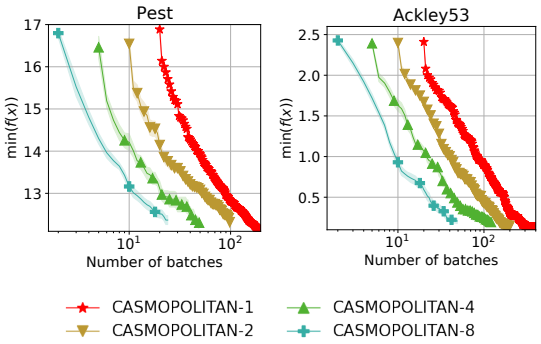

Figure 5. Parallel CASMOPOLITAN on representative categorical and mixed problems by *number of batches* excluding the initially randomly-sampled batches. Note the x-axis is in log-scale for better presentation. We show the comparison by *number of function queries* in App. B.

We would often like to exploit parallelism in computing where we dispatch different queries to the black-box objective function for independent evaluations. This setting necessitates the development of *batch* methods to propose a batch of b points for simultaneous evaluation at each BO iteration. However, this often involves trade-off between wall-clock time efficiency against performance, because surrogates in batch methods are updated only once per b

objective function evaluations. Here we investigate the performance of CASMOPOLITAN under different batch sizes where $b = 1$ (sequential setting) $2, 4$ & 8 in Pest Control and Ackley-53 problems previously considered; where $b > 1$, we use the Kriging believer strategy (Ginsbourger et al., 2010) during acquisition optimisation to deliver b proposals simultaneously. In both experiments, we keep the budget of the objective function queries to be identical to that in Sec. 4.1 & 4.2 but scale the number of batches accordingly, and the results are shown in Fig. 5: it is evident that larger batch sizes, while leading to almost linear reduction in wall-clock time, do not lead to significant performance deterioration, except some minor under-performance at the end which seems to scale with b. However, in both problems, CASMOPOLITAN even with the largest batch size investigated still outperforms *sequential* baselines in Figs. 3 & 4.

4.4. Ablation Studies

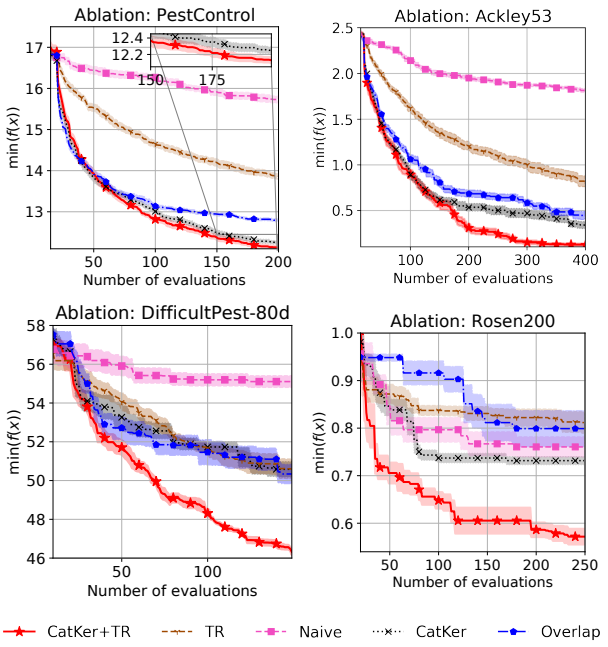

Figure 6. Ablation studies of our method in categorical (left) and mixed (right) optimisation problems. First row: Pest (left), Ackley-53 (right); Second row: DifficultPest (left), Rosenbrock-200 (right).

Our method introduces a number of modifications over the naïve BO approach. To understand the benefits of these, we conduct ablation studies in both the categorical and mixed problems. Specifically, we include the following setups.

- The naïve BO approach with global GP surrogate and one-hot transformation on the categorical variables (`Naive`);
- One-hot transformed BO, but with *local* TRs i.e. TURBO (`TR`);
- GP with *global* surrogates, but with the categorical overlapping kernel in Ru et al. (2020a) where applicable (`Overlap`);
- BO with *global* GP surrogate, but with the kernel defined in Eq. (1) or (4), where appropriate (`CatKer`);
- Our approach that incorporates both local modelling and the kernel in Eq. (1) or (4) (`CatKer+TR`).

We firstly include Pest Control and the Ackley-53 problems as representative problems for the categorical and mixed setups for the ablation studies. To further understand the relative importance of the various features of CASMOPOLITAN especially as the dimensionality of the problems changes, we also include two even higher-dimensional problems, namely 1) Pest control with number of stages expanded to 80, which we term DifficultPest (the number of possible configurations is more than 8.27×10^{55}), and 2) 200-d Rosenbrock with 100 binary dimensions and 100 continuous dimensions (detailed in App. C).

We show the results in Fig. 6: in most problems, the usage of the categorical kernel leads to improvements over baselines, with kernels used in our method generally outperforming the overlap kernel. Unsurprisingly, the additional benefits of local optimisation and the use of trust regions increase with increasing dimensionality and complexity of the problems, with largest benefits coming from the two high-dimensional problems of the second row. Nonetheless, it is worth noting that even in the relatively modestly-dimensioned Pest Control problem where the difference between `CatKer+TR` and `CatKer` seems small, the outperformance is still statistically significant (Two-sample Student's t-test yields $p = 0.043 < 0.05$ at the final iteration). Finally, our method, similar to TURBO, introduces a number of additional hyperparameters related to the TR; we examine the sensitivity of performance towards these extra hyperparameters in App. B.

5. Conclusion and Future Work

We propose CASMOPOLITAN, a novel GP-BO approach using ideas of tailored kernels and trust regions to tackle the challenging high-dimensional optimisation problem over categorical and mixed search spaces. We both analyse our method theoretically and empirically demonstrate its effectiveness over a wide range of problems. Possible future directions may extend our model to even more diverse search spaces, such as problems on graphs, trees, and/or in conditional spaces.

Acknowledgements

The authors would like to thank the Oxford-Man Institute of Quantitative Finance for providing computing resources in this project. The authors also thank the anonymous ICML reviewers and the area chair for the constructive feedback which helped to improve the paper.

References

Moustafa Alzantot, Yash Sharma, Supriyo Chakraborty, Huan Zhang, Cho-Jui Hsieh, and Mani B Srivastava. Genattack: Practical black-box attacks with gradient-free optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1111–1119, 2019. (Cited on page 19)

Ricardo Baptista and Matthias Poloczek. Bayesian optimization of combinatorial structures. In *International Conference on Machine Learning*, pages 462–471. PMLR, 2018. (Cited on pages 2, 7, 13, and 21)

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems*, pages 2546–2554, 2011. (Cited on page 7)

Felix Berkenkamp, Angela P. Schoellig, and Andreas Krause. No-regret bayesian optimization with unknown hyperparameters. *Journal of Machine Learning Research*, 20(50):1–24, 2019. (Cited on page 20)

John Bibby. Axiomatisations of the average and a further generalisation of monotonic sequences. *Glasgow Mathematical Journal*, 15(1):63–65, 1974. (Cited on page 23)

Mickaël Binois, David Ginsbourger, and Olivier Roustant. A warped kernel improving robustness in Bayesian optimization via random embeddings. In *International Conference on Learning and Intelligent Optimization*, pages 281–286. Springer, 2015. (Cited on page 2)

Mickaël Binois, David Ginsbourger, and Olivier Roustant. On the choice of the low-dimensional domain for global optimization via random embeddings. *Journal of global optimization*, 76(1):69–90, 2020. (Cited on page 2)

Laurens Bliek, Sicco Verwer, and Mathijs de Weerdt. Black-box mixed-variable optimisation using a surrogate model that satisfies integer constraints. *arXiv preprint arXiv:2006.04508*, 2020. (Cited on pages 2, 7, 19, 21, and 26)

Leo Breiman. Random forests. *Machine learning*, 45(1): 5–32, 2001. (Cited on page 3)

Eric Brochu, Vlad M Cora, and Nando De Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010. (Cited on pages 1, 6, and 13)

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SigKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM, 2016. (Cited on page 7)

Nadia Creignou, Sanjeev Khanna, and Madhu Sudan. *Complexity classifications of boolean constraint satisfaction problems*. SIAM, 2001. (Cited on page 1)

Francesco Croce, Maksym Andriushchenko, Naman D Singh, Nicolas Flammarion, and Matthias Hein. Sparse-rs: a versatile framework for query-efficient sparse black-box adversarial attacks. *arXiv preprint arXiv:2006.12834*, 2020. (Cited on page 8)

Hamid Dadkhahi, Karthikeyan Shanmugam, Jesus Rios, Payel Das, Samuel C Hoffman, Troy David Loeffler, and Subramanian Sankaranarayanan. Combinatorial black-box optimization with expert advice. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1918–1927, 2020. (Cited on page 2)

Hamid Dadkhahi, Jesus Rios, Karthikeyan Shanmugam, and Payel Das. Fourier representations for black-box optimization over categorical variables. 2021. (Cited on page 2)

Erik Daxberger, Anastasia Makarova, Matteo Turchetta, and Andreas Krause. Mixed-variable Bayesian optimization. *arXiv preprint arXiv:1907.01329*, 2019. (Cited on page 2)

Aryan Deshwal, Syrine Belakaria, Janardhan Rao Doppa, and Alan Fern. Optimizing discrete spaces via expensive evaluations: A learning to search framework. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34 (04):3773–3780, Apr. 2020. (Cited on page 2)

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. Scalable global optimization via local Bayesian optimization. In *Advances in Neural Information Processing Systems*, pages 5496–5507, 2019. (Cited on pages 1, 2, 4, 5, 7, 20, and 21)

Peter I Frazier. A tutorial on Bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018. (Cited on page 13)

Jacob R Gardner, Geoff Pleiss, David Bindel, Kilian Q Weinberger, and Andrew Gordon Wilson. Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. *arXiv preprint arXiv:1809.11165*, 2018. (Cited on page 13)

Eduardo C Garrido-Merchán and Daniel Hernández-Lobato. Dealing with categorical and integer-valued variables in Bayesian optimization with Gaussian processes. *Neurocomputing*, 380:20–35, 2020. (Cited on page 2)

David Ginsbourger, Rodolphe Le Riche, and Laurent Carraro. Kriging is well-suited to parallelize optimization. In *Computational intelligence in expensive optimization problems*, pages 131–162. Springer, 2010. (Cited on pages 9 and 20)

Shivapratap Gopakumar, Sunil Gupta, Santu Rana, Vu Nguyen, and Svetha Venkatesh. Algorithmic assurance: An active approach to algorithmic testing using Bayesian optimisation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5465–5473, 2018. (Cited on pages 1 and 2)

GPyOpt. GPyOpt: A Bayesian optimization framework in python. http://github.com/SheffieldML/GPyOpt, 2016. (Cited on page 2)

José Miguel Hernández-Lobato, James Requeima, Edward O Pyzer-Knapp, and Alán Aspuru-Guzik. Parallel and distributed Thompson sampling for large-scale accelerated exploration of chemical space. *In International Conference on Machine Learning*, pages 1470–1479, 2017. (Cited on page 1)

Yingjie Hu, JianQiang Hu, Yifan Xu, Fengchun Wang, and Rong Zeng Cao. Contamination control in food supply chain. In *Proceedings of the 2010 Winter Simulation Conference*, pages 2678–2681. IEEE, 2010. (Cited on pages 7, 17, and 26)

Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *Learning and Intelligent Optimization*, pages 507–523. Springer, 2011. (Cited on pages 1, 3, and 7)

Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998. (Cited on page 1)

Kirthevasan Kandasamy, Jeff Schneider, and Barnabás Póczos. High dimensional Bayesian optimisation and bandits via additive models. In *International Conference on Machine Learning*, pages 295–304, 2015. (Cited on page 2)

Kirthevasan Kandasamy, Willie Neiswanger, Jeff Schneider, Barnabas Poczos, and Eric P Xing. Neural architecture search with bayesian optimisation and optimal transport. In *Advances in Neural Information Processing Systems*, pages 2016–2025, 2018. (Cited on page 1)

Donghwan Kim and Jeffrey A Fessler. Adaptive restart of the optimized gradient method for convex optimization. *Journal of Optimization Theory and Applications*, 178(1): 240–263, 2018. (Cited on page 4)

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015. (Cited on page 20)

Risi Kondor and John D. Lafferty. Diffusion kernels on graphs and other discrete input spaces. In *International Conference on Machine Learning*, pages 315–322, 2002. (Cited on page 4)

Andreas Krause and Cheng S Ong. Contextual Gaussian process bandit optimization. In *Advances in Neural Information Processing Systems*, pages 2447–2455, 2011. (Cited on page 23)

Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998. (Cited on page 7)

Ben Letham, Roberto Calandra, Akshara Rai, and Eytan Bakshy. Re-examining linear embeddings for high-dimensional bayesian optimization. *Advances in Neural Information Processing Systems*, 33, 2020. (Cited on pages 2, 7, and 16)

Vladimir Mazya and Tatyana Shaposhnikova. *Jacques Hadamard: A Universal Mathematician*. 1st edition, 1999. (Cited on page 23)

Mojmír Mutnỳ and Andreas Krause. Efficient high dimensional Bayesian optimization with additivity and quadrature fourier features. *Advances in Neural Information Processing Systems 31*, pages 9005–9016, 2019. (Cited on page 2)

Amin Nayebi, Alexander Munteanu, and Matthias Poloczek. A framework for Bayesian optimization in embedded subspaces. In *International Conference on Machine Learning*, pages 4752–4761. PMLR, 2019. (Cited on page 2)

Dang Nguyen, Sunil Gupta, Santu Rana, Alistair Shilton, and Svetha Venkatesh. Bayesian optimization for categorical and category-specific continuous inputs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5256–5263, 2020. (Cited on pages 1 and 2)

Vu Nguyen, Tam Le, Makoto Yamada, and Michael A Osborne. Optimal transport kernels for sequential and parallel neural architecture search. In *International Conference on Machine Learning*, 2021. (Cited on page 1)

Changyong Oh, Jakub Tomczak, Efstratios Gavves, and Max Welling. Combinatorial Bayesian optimization using the graph cartesian product. In *Advances in Neural Information Processing Systems*, pages 2914–2924, 2019. (Cited on pages 2, 7, 13, 15, 18, 21, and 26)

Jack Parker-Holder, Vu Nguyen, and Stephen J Roberts. Provably efficient online hyperparameter optimization with population-based bandits. *Advances in Neural Information Processing Systems*, 33, 2020. (Cited on page 1)

Santu Rana, Cheng Li, Sunil Gupta, Vu Nguyen, and Svetha Venkatesh. High dimensional Bayesian optimization with elastic gaussian process. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 2883–2891, 2017. (Cited on pages 1 and 8)

Carl Edward Rasmussen. Gaussian processes for machine learning. 2006. (Cited on pages 2, 3, and 13)

Paul Rolland, Jonathan Scarlett, Ilija Bogunovic, and Volkan Cevher. High-dimensional Bayesian optimization via additive models with overlapping groups. In *International conference on artificial intelligence and statistics*, pages 298–307. PMLR, 2018. (Cited on page 2)

Binxin Ru, Ahsan Alvi, Vu Nguyen, Michael A Osborne, and Stephen Roberts. Bayesian optimisation over multiple continuous and categorical inputs. In *International Conference on Machine Learning*, pages 8276–8285. PMLR, 2020a. (Cited on pages 1, 2, 4, 5, 7, 9, 18, 19, 20, 21, and 26)

Binxin Ru, Adam Cobb, Arno Blaas, and Yarin Gal. Bayesopt adversarial attack. In *International Conference on Learning Representations*, 2020b. (Cited on pages 7, 8, and 19)

Binxin Ru, Xingchen Wan, Xiaowen Dong, and Michael Osborne. Interpretable neural architecture search via Bayesian optimisation with weisfeiler-lehman kernels. *International Conference on Learning Representations*, 2021. (Cited on page 1)

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando de Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016. (Cited on pages 1 and 13)

Oleg V Shylo, Timothy Middelkoop, and Panos M Pardalos. Restart strategies in optimization: parallel and serial cases. *Parallel Computing*, 37(1):60–68, 2011. (Cited on page 4)

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, pages 2951–2959, 2012. (Cited on pages 1 and 2)

Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on Machine Learning*, pages 1015–1022, 2010. (Cited on pages 5, 6, 7, 24, and 25)

Kevin Swersky, Yulia Rubanova, David Dohan, and Kevin Murphy. Amortized bayesian optimization over discrete spaces. In *Conference on Uncertainty in Artificial Intelligence*, pages 769–778. PMLR, 2020. (Cited on page 2)

James Joseph Sylvester. Xxxvii. on the relation between the minor determinants of linearly equivalent quadratic functions. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 1(4):295–305, 1851. doi: 10.1080/14786445108646735. (Cited on page 22)

Chun-Chen Tu, Paishun Ting, Pin-Yu Chen, Sijia Liu, Huan Zhang, Jinfeng Yi, Cho-Jui Hsieh, and Shin-Ming Cheng. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 742–749, 2019. (Cited on page 19)

Zi Wang, Chengtao Li, Stefanie Jegelka, and Pushmeet Kohli. Batched high-dimensional Bayesian optimization via structural kernel learning. In *International Conference on Machine Learning*, pages 3656–3664. PMLR, 2017. (Cited on page 2)

Zi Wang, Clement Gehring, Pushmeet Kohli, and Stefanie Jegelka. Batched large-scale Bayesian optimization in high-dimensional spaces. In *International Conference on Artificial Intelligence and Statistics*, pages 745–754. PMLR, 2018. (Cited on page 2)

Ziyu Wang, Masrour Zoghi, Frank Hutter, David Matheson, N Freitas, et al. Bayesian optimization in high dimensions via random embeddings. AAAI Press/International Joint Conferences on Artificial Intelligence, 2013. (Cited on page 16)

Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Feitas. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, 55:361–387, 2016. (Cited on pages 2 and 7)

Ya-xiang Yuan. A review of trust region algorithms for optimization. In *Iciam*, volume 99, pages 271–282. Citeseer, 2000. (Cited on pages 6 and 7)