Blessing of Dimensionality: How Many-Objective Search Supports Inverse Modeling

Abhishek Gupta
Singapore Institute of Manufacturing Technology, A*STAR, Singapore.
abhishek_gupta@simtech.a-star.edu.sg

Yew-Soon Ong
School of Computer Science and Engineering, Nanyang Technological University, Singapore.
asysong@ntu.edu.sg

Mojtaba Shakeri
Singapore Institute of Manufacturing Technology, A*STAR, Singapore.
mojtaba_shakeri@simtech.a-star.edu.sg

Xu Chi
Singapore Institute of Manufacturing Technology, A*STAR, Singapore.
cxu@simtech.a-star.edu.sg

Allan Zhang NengSheng
Singapore Institute of Manufacturing Technology, A*STAR, Singapore.
nzhang@simtech.a-star.edu.sg

***Abstract*—Sample-based evolutionary algorithms (EAs) are widely used for optimizing problems with *multi* (greater than one but less than four) or even *many* (greater than or equal to four) objectives of interest. In general, the difficulty of a problem exponentially increases with the number of objectives, serving as a clear example of the *curse of dimensionality*. The exploratory approach an EA takes in these cases has led to it being thought of as a big data generator, progressively sampling and evaluating solutions in high performing regions of a decision space to guide the search towards optimal solutions. Notably, in both multi- and many-objective EAs, the data samples can be further utilized for building *inverse models*, mapping points in objective space back to solutions in the decision space. *Such models offer immense flexibility to a decision maker in generating new target solutions on the fly, thereby facilitating real-time a posteriori preference incorporation*. In this paper, we show that the data generated from a many-objective optimization search is in fact more conducive to inverse modeling than its multi-objective counterpart (where the output space is compressed via synthetic objective aggregation) – highlighting a rare *blessing of dimensionality* that has yet to be explored in the context of optimization. We first present simple theoretical arguments supporting this statement. Thereafter, experimental studies of Gaussian process-based inverse modeling for a pedagogical and a practical composite manufacturing example are carried out to further confirm the theory.**

***Keywords*—Blessing of dimensionality, inverse modeling, many-objective optimization, Gaussian processes**

I. Introduction

Over the years, the distinctive features of *natural selection*-inspired *evolutionary algorithms* (EAs) have come to be greatly valued for their efficacy in handling *multi-objective optimization problems* (MOPs) [1, 2]. Within the evolutionary computation research community, the term MOP is customarily used for problems comprising two or three conflicting objectives to be optimized at the same time. This setting gives rise to multiple optimal solutions – cumulatively referred to as the *Pareto set* – instead of a single optimum that is typically encountered in standard single-objective problems. EAs, by virtue of the *implicit parallelism* of their population-based search strategy [3], efficiently explore the space of possible solutions to facilitate simultaneous convergence to the Pareto set – a feature unavailable in other conventional optimization engines.

Lately, it has been demonstrated that with some design enhancements, the unique facets of EAs for MOPs (or MOEAs for short) can be scaled to objective spaces of even higher dimensionality. The resultant *many-objective EAs* (or MaOEAs) are being increasingly deployed for so-called *many-objective optimization problems* (MaOPs), characterized by the presence of more than three conflicting objectives to be optimized simultaneously [4]. It is worth noting that preceding the development of effective MaOEAs, a practical approach to dealing with MaOPs was to synthetically reduce the number of objectives – for example, by aggregating them based on *a priori* known preference relationships – bringing the problem to the ambit of MOEAs. One of the reasons why vanilla MOEAs have proven unsatisfactory for directly tackling MaOPs is that with an increasing number of conflicting objectives, the induced *evolutionary selection pressure* begins to weaken (as will be explained in more detail in Section II), thereby severely deteriorating the convergence property of the algorithm [5]. This phenomenon serves as a clear manifestation of the *curse of dimensionality* in optimization, and has thus attracted significant research attention in recent years – leading to the development of a plethora of novel EAs tailored for solving MaOPs [6, 7, 8].

In this paper, we consider MOPs and MaOPs from a different perspective. We recognize that much research effort has been dedicated to improving the convergence rate of algorithms for MaOPs; however, we also note that in practice the likelihood of finding a solution, mapping to a high dimensional objective space, that closely satisfies a decision maker's (DM's) preferences is low. Given the finite (albeit big) solution sets sampled and evaluated by EAs, the likelihood is even lower in the case of continuous optimization problems where the number of Pareto optimal solutions may be infinite. What is more, the DM's preferences may not be limited to just the Pareto set. It is indeed possible that solutions that are optimal at one point in time (given the ready availability of resources) become infeasible at another (due to resource exhaustion); thus,

stressing the utility of finding (and keeping track of) sub-optimal solutions in any given region of the DM's preference [9].

A promising approach to overcome the difficulty of EAs in guaranteeing DM satisfaction is to leverage the large amounts of data collected during the optimization for *inverse modeling* [10] – i.e., building models that take points from objective space as input and generate corresponding solutions in decision space. In particular, if the formulation of an MOP or MaOP is based on the forward mapping $F: \mathcal{X} \rightarrow \mathcal{Y}$, where $\mathcal{X}$ is an n-dimensional decision space and $\mathcal{Y}$ is an m-dimensional objective space, then the inverse model is aimed at learning the following backward mapping: $F^{-1}: \mathcal{Y} \rightarrow \mathcal{X}$. With a sufficiently accurate model F^{-1} at hand, the DM can (in principle) freely focus the sampling of solutions within any arbitrary region of interest, *facilitating real-time preference incorporation in a posteriori preference articulation scenarios* (as is typically the case for MOPs).

Following from the above, consider $F_{multi}^{-1}: \mathcal{Y} \rightarrow \mathcal{X}$ and $F_{many}^{-1}: \mathcal{Y} \rightarrow \mathcal{X}$ to represent inverse maps learned from multi-objective search data (D_{multi}) and many-objective search data (D_{many}), respectively; the target problem in both cases is essentially the same, but with the key distinction that multi-objective evolution includes an objective aggregation step prior to solution selection. Note that even though the input features and output labels of the two datasets are identical, their underlying probability distributions, denoted as $P(D_{multi})$ and $P(D_{many})$, are likely to differ (i.e., $P(D_{multi}) \neq P(D_{many})$); this is due to the distinctive search evolution patterns and final outcomes produced under either a multi- or many-objective problem formulation. *Given this starting point, the main thesis of the present paper is that the distribution of dataset D_{many} is in fact better suited for building accurate inverse generative models as opposed to the distribution of D_{multi}.* Our conjecture suggests a rare *blessing of dimensionality* [11] in the context of search in high dimensional objective spaces that is yet to be exploited in the optimization literature. Accordingly, we first establish a conceptual basis for our claim under the assumption of Gaussian process-based inverse models. Thereafter, numerical tests are carried out, encompassing a real-world case study in advanced composite manufacturing, to provide further empirical verification to our claim.

The remainder of the paper is organized as follows. In Section II, we present basic concepts and formalisms of MOPs and MaOPs, and illustrate how a linear objective aggregation can be applied to transform the latter into the former. Section III briefly introduces the Gaussian process (GP) model that is used throughout the paper for inverse modeling. Next, Section IV lays out the theoretical arguments supporting the conceived blessing of dimensionality of many-objective search; we show that as a result of the resultant search behavior, an inverse GP model conditioned on D_{many} is expected to generalize more effectively than one conditioned on D_{multi}. Section V presents the experimental studies, and Section VI concludes the paper with a brief summary of the main ideas.

II. Multi- and Many-Objective Optiimzation

In this section, we start with a general mathematical statement of optimization problems where several objectives of interest are to be considered simultaneously; MOPs and MaOPs have come to be informally viewed as separate classes of this general problem type. For simplicity, the presence of constraints is ignored. Some of the key concepts in solving such problems are presented thereafter, together with a simple illustration of how (and why) MaOPs may be turned into MOPs.

Consider the following minimization problem,

$$\min F(\boldsymbol{x}) = [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \dots, f_m(\boldsymbol{x})]^T,$$
$$\text{such that, } \boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^n, \quad (1)$$

where $f_i(\boldsymbol{x})$ represents the ith objective function, and $x = (x_1, x_2, \dots, x_n)$ is a vector of n decision variables in the decision space $\mathcal{X}$. As per convention, if $1 < m \leq 3$ then the problem is classified as an MOP, otherwise if $m \geq 4$ then it is regarded as an MaOP [4].

Given the setup in Eq. (1), a candidate solution $\boldsymbol{x}_a$ is said to *dominate* (i.e., be better than) another solution $\boldsymbol{x}_b$, denoted as $\boldsymbol{x}_a \succ \boldsymbol{x}_b$, if and only if $\forall i \in \{1,2,\dots,m\}: f_i(\boldsymbol{x}_a) \leq f_i(\boldsymbol{x}_b)$ and $\exists j: f_j(\boldsymbol{x}_a) < f_j(\boldsymbol{x}_b)$. We thus label a solution $\boldsymbol{x}_*$ as being optimal (or more precisely: *Pareto optimal*) for Eq. (1) if $\nexists \boldsymbol{x} \in \mathcal{X}$ such that $\boldsymbol{x} \succ \boldsymbol{x}_*$ [9]. The set of all such *non-dominated* solutions constitutes the possibly infinitely large *Pareto set* (PS). The mapping of each element of PS to the objective space $\mathcal{Y}$ forms what is commonly referred to as the *Pareto frontier* (PF); i.e., $PF \equiv \{F(\boldsymbol{x}_*), \forall \boldsymbol{x}_* \in PS \subseteq \mathcal{X}\}$.

Originating from the definitions above, the notion of *non-domination rank* is one that has been widely used in the design of MOEAs [12]. Briefly, if solution $\boldsymbol{x}_a$ dominates $\boldsymbol{x}_b$, then $\boldsymbol{x}_a$ is guaranteed to have a better non-domination ranking than $\boldsymbol{x}_b$. Accordingly, by preferentially preserving (i.e., ensuring survival of) solutions with a better non-domination rank in the population of an MOEA, the algorithm is able to induce an evolutionary selection pressure that guides the search towards the PF. This selection pressure needs to be strong for the algorithm to converge fast. However, notice that if we randomly pick two points in an m-dimensional objective space, the probability that a dominance relation will exist between them is $2^{-(m-1)}$. In other words, as m increases, the probability that a definite preference relationship can be established between solutions tends to reduce at an exponential rate – at least in the initial stages of an MOEA when the solutions are randomly initialized. The lack of dominance feedback implies that the evolutionary selection pressure weakens, leading to a significant slowdown in the convergence rates of MOEAs for large m.

With the goal of scaling EAs for MaOPs, a plethora of sampling-based algorithms have been proposed in recent years that attempt to enhance the selection pressure by modifying the definition of dominance [5] (or alternatively, by using novel selection criteria that are distinct from the notion of dominance). Nevertheless, before the general effectiveness of such methods is established, a practical approach to dealing with objective spaces of high dimensionality is to reduce the number of objectives via aggregation. As an example, assuming without loss of generality that the first l objectives in F have some *a priori* known preference relationship that can be captured by an l-dimensional weight vector $\boldsymbol{w} = [w_1, w_2, \dots, w_l]$, where $l < m$, Eq. (1) may be reformulated as,

$$\min F_{agg}(\boldsymbol{x}) = [f_{agg}(\boldsymbol{x}), f_{l+1}(\boldsymbol{x}), \dots, f_m(\boldsymbol{x})]^T,$$
$$\text{s.t., } f_{agg}(\boldsymbol{x}) = [w_1, w_2, \dots, w_l] \cdot [f_1(\boldsymbol{x}), \dots, f_l(\boldsymbol{x})]^T, \quad (2)$$
$$\text{where, } \textstyle\sum_{i=1}^{l} w_i = 1;\ w_i > 0\ \forall i \in \{1,2,\dots,l\};\ \boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^n.$$

In Eq. (2), the linear aggregation $f_{agg}(\boldsymbol{x})$ under strictly positive weights implies that none of the initial objective functions is ignored. More importantly, if we have $m - l = 1$ or 2, then the resultant definition of F_{agg} can be treated as an MOP, that is efficiently solvable by well-known MOEAs.

III. Gaussian Processes for Inverse Modeling

In this paper, we focus on inverse modeling as a way to focus the generation of solutions in decision space on the fly, such that they may correspond to any arbitrary region of the DM's preference in the objective space. The use of probabilistic non-parametric GPs has previously been proposed in a related context in the evolutionary computation literature [13].

Given a dataset $D = \{X, Y\} = \{(\boldsymbol{x}_s, \boldsymbol{y}_s)\}_{s=1}^{S}$ of S *unique* sample evaluations obtained from an evolutionary optimization run, where $\boldsymbol{y}_s = F(\boldsymbol{x}_s)$, a GP serves to learn the inverse mapping $F^{-1}: \mathcal{Y} \rightarrow \mathcal{X}$ taking points from the objective space $\mathcal{Y}$ as input to generate corresponding solutions in decision space $\mathcal{X}$. Specifically, we intend to infer a set of latent functions $\{g_d, \forall d \in \{1,2,\dots,n\}\}$ such that $F_d^{-1}(\boldsymbol{y}) = x_d = g_d(\boldsymbol{y}) + \varepsilon_d$. Here, ε_d represents an additive noise term that models the difference between observed values x_d and the latent function value g_d, and is assumed to follow an *i.i.d* Gaussian distribution with zero mean and variance $\sigma_{d,\varepsilon}^2$; i.e.,

$$\varepsilon_d \sim \mathcal{N}(0, \sigma_{d,\varepsilon}^2). \quad (3)$$

We approach the inverse modeling problem by placing a *joint* GP prior over the latent functions $\{g_d\}$. We further assume that the GPs have zero mean, and that the joint covariance function is defined as [14],

$$\langle g_d(\boldsymbol{y}), g_e(\boldsymbol{y}') \rangle = K_{de}^f \cdot k(\boldsymbol{y}, \boldsymbol{y}'). \quad (4)$$

In the above, K^f is a positive semi-definite matrix the allows the GP to capture relationships between latent functions, and $k(\cdot,\cdot)$ is any valid kernel over GP inputs.

For a more detailed discussion on the properties of the aforementioned joint GP model, the reader is referred to [15]. In all subsequent analyses and experiments, we limit the matrix K^f to be diagonal, i.e., $K_{de}^f = 0$ if $d \neq e$; this not only simplifies model training, but also allows the different latent functions to be analyzed independently from each other.

A. Hyperparameter Learning

Considering a single latent function $g_d(\boldsymbol{y})$ at a time, the hyperparameters $\boldsymbol{\theta}_d$ of the GP covariance function together with the noise variance $\sigma_{d,\varepsilon}^2$ may be learned by maximizing the following log-marginal likelihood,

$$\log p\left(\{x_{s,d}\}_{s=1}^{S} | Y, \boldsymbol{\theta}_d, \sigma_d^2\right) = -\frac{1}{2}\left(\boldsymbol{\psi}^T \left(K + \sigma_{d,\varepsilon}^2 I_S\right)^{-1} \boldsymbol{\psi} + \log\left|K + \sigma_{d,\varepsilon}^2 I_S\right|\right). \quad (5)$$

Here, $x_{s,d}$ is the dth decision variable of $\boldsymbol{x}_s$ in X, $\boldsymbol{\psi} = \{x_{s,d}\}_{s=1}^{S}$ expressed as a column vector, and $K = K_{dd}^f \cdot k(Y, Y)$ is the symmetric kernel matrix.

As is well-known, the training of a GP scales as $\mathcal{O}(S^3)$, which practically limits them to datasets of size $\mathcal{O}(10^4)$ [16]. With this in mind, it is worth noting that the datasets generated from standard applications of EAs are typically quite large. For instance, even using a small population size of 100 solutions (this number is deemed small for searches in high dimensional objective space) that are evolved over a modest period of 200 iterations (which is common), we already have $S \approx 2 \times 10^4$. Thus, for the big datasets that are likely to be encountered in the proposed inverse modeling, fast approximations to the full GP model must be sought. To this end, a simple approach is the *product-of-GP-experts* (POE), where, instead of training a single GP model on the entire dataset, Q separate GP experts are trained on data subsets $D^{(q)}$ such that $\bigcup_{q=1}^{Q} D^{(q)} = D$ [17]; the data-to-model assignment may be random or ascertained by clustering methods (in the present paper, the former is applied in favor of speed). In either case, given non-overlapping data subsets of similar size, the training complexity of each expert reduces to $\mathcal{O}(S^3/Q^3)$ in time. In addition, the method of POE allows for straightforward parallelization.

B. Inverse Model Predictions

Given optimized hyperparameters $\boldsymbol{\theta}_d$, noise variance $\sigma_{d,\varepsilon}^2$, and training data subset $\boldsymbol{\psi}^{(q)}$ and $Y^{(q)}$, the qth GP expert's posterior predictive distribution of the corresponding latent function value $g_d(\boldsymbol{y}_*)$, at some (preferred) objective vector value $\boldsymbol{y}_* \in \mathcal{Y}$ supplied by the DM, is,

$$\mathbb{E}\left[g_d(\boldsymbol{y}_*)|D^{(q)}\right] = \mu_d^{(q)}(\boldsymbol{y}_*) = k_*^T (K + \sigma_{d,\varepsilon}^2 I)^{-1} \boldsymbol{\psi}^{(q)}, \quad (6)$$
$$\text{var}[g_d(\boldsymbol{y}_*)|D^{(q)}] = \sigma_d^{2(q)}(\boldsymbol{y}_*) = k_{**} - k_*^T \left(K + \sigma_{d,\varepsilon}^2 I\right)^{-1} k_*. \quad (7)$$

In the above, $k_* = k(Y^{(q)}, \boldsymbol{y}_*)$ and $k_{**} = k(\boldsymbol{y}_*, \boldsymbol{y}_*)$. With this, the POE predicts the latent function value at $\boldsymbol{y}_*$ by combining the predictions of the Q GP experts as [17],

$$\mathbb{E}[g_d(\boldsymbol{y}_*)|D] = \mu_d(\boldsymbol{y}_*) = \sigma_d^2(\boldsymbol{y}_*) \sum_{q=1}^{Q} \sigma_d^{-2(q)}(\boldsymbol{y}_*) \cdot \mu_d^{(q)}(\boldsymbol{y}_*), \quad (8)$$

$$\text{var}[g_d(\boldsymbol{y}_*)|D] = \sigma_d^2(\boldsymbol{y}_*) = \left[\sum_{q=1}^{Q} \sigma_d^{-2(q)}(\boldsymbol{y}_*)\right]^{-1}. \quad (9)$$

Thus, the POE leads to a linear combination of the individual GP experts to construct an approximation of the full GP model prediction. The efficacy of the precise form of Eq. (8) can be explained by the following known result.

Theorem 1 *(Perrone and Cooper [18]). Suppose that the errors between the Q expert predictions and the target solutions are mutually uncorrelated and have zero mean. Then, the optimal linear combination* $\sum_q \alpha_q \cdot \mu_d^{(q)}$ *that minimizes the mean squared predictive error, under the constraint* $\sum_q \alpha_q = 1$*, is* $\alpha_q = \sigma_d^{-2(q)} / \sum_q \sigma_d^{-2(q)}$.

IV. Analysing the Blessing of Dimensionality

When m in Eq. (1) is large, the selection pressure induced by well-known dominance-based MOEAs may not suffice to ensure fast convergence to the PF. For such MOEAs to still be applicable, an MaOP must therefore be transformed into a relatively low dimensional search (i.e., an MOP); for example, via objective aggregation as shown in Eq. (2). While recent advances in MaOEAs have shown promise in enhancing evolutionary selection pressure for the purpose of many objectives, due to the inherent difficulties of performance visualization in high dimensions, much remains to be achieved in terms of their rigorous verification.

In this section, we present theoretical support to the claim that despite the curse of dimensionality, the data accumulated during many-objective evolutionary search is in fact better suited for inverse modeling as opposed to its low dimensional counterpart. As reasoned in the introduction, *given that inverse modeling may potentially be a necessary step in guaranteeing DM satisfaction*, the effectiveness of many-objective search to this end marks a rare blessing of dimensionality that is yet to be highlighted and explored in evolutionary computation.

A. Assumptions

For assisting our argumentation, the following simplifying assumptions are made.

Assumption 1. The initial noisy samples in datasets $D_{multi} = \{X_{multi}, Y_{multi}\}$ and $D_{many} = \{X_{many}, Y_{many}\}$ – caused by the random population initialization of EAs – are assumed to be excluded. *The datasets are thus deemed to (mainly) consist of solution samples that map to and across the PFs of the multi- and many-objective problem formulations, respectively*.

Assumption 2. Decision space $\mathcal{X}$ and the high dimensional objective space $\mathcal{Y}$ are considered to be discretized with a *one-to-one mapping* between them.

Assumption 3. The POE-based inverse model predictions, as detailed in Section III-B, accurately replicate a full GP model with correctly determined prior. We note that although the validity of this assumption begins to breakdown with increasing number of GP experts – as the vanishing variance in Eq. (9) leads to overconfident predictions [17] – it allows us to leverage well-established theories of the GP as long as Q is small.

B. Theoretical Results

The first result below compares the PF of a many-objective problem formulation as in Eq. (1) with that of its multi-objective counterpart in Eq. (2). We denote the former as PF_{many} and the latter as PF_{multi}.

Lemma 1. *Given that the l-dimensional weight vector $\boldsymbol{w}$ in Eq. (2) satisfies $w_i > 0\ \forall i \in \{1,2,\dots,l\}$, the MOP formulation with aggregated objectives F_{agg} leads to the relationship: $PF_{multi} \subseteq PF_{many}$ for all feasible $\boldsymbol{w}$'s.*

Proof. We put forward a proof by contradiction. Let $F^*_{agg} = \left[f^*_{agg}, f^*_{l+1}, \dots, f^*_m\right]^T \in PF_{multi}$, and let the same point expressed in the original high dimensional objective space be $F^* = [f^*_1, \dots, f^*_{l+1}, \dots, f^*_m]^T$. We start by considering,

$$F^* \notin PF_{many}. \tag{10}$$

Accordingly, $\exists F' = [f'_1, \dots, f'_{l+1}, \dots, f'_m]^T$ such that $f'_i \leq f^*_i\ \forall i \in \{1,2,\dots,m\}$, with at least one strict inequality. Given the positivity of the aggregation weights, we then have $f'_{agg} \leq f^*_{agg} \wedge f'_i \leq f^*_i\ \forall i \in \{l+1, l+2, \dots, m\}$ with at least one strict inequality. This implies that $F^*_{agg} \notin PF_{multi}$. In other words, Eq. (10) contradicts our initial assertion, implying that,

$$F^*_{agg} \in PF_{multi} \Rightarrow F^* \in PF_{many}. \tag{11}$$

Further to the above, it is trivial to find counterexamples affirming $F^* \in PF_{many} \nRightarrow F^*_{agg} \in PF_{multi}$. Therefore, the key statement of the lemma follows from Eq. (11). ■

On the basis of Lemma 1 and Assumption 1, it is possible to approximate $supp[P(Y_{multi})] \subseteq supp\left[P\left(Y_{many}\right)\right]$, where $supp[\cdot]$ represents the support of the underlying probability distribution. With this approximation, we can now show that for big evolutionary optimization data, the expected prediction error of an inverse GP model conditioned on D_{many} is bounded from above by the expected error of an equivalent model conditioned on D_{multi}. Similar to [19], we explicitly account for the big data assumption by setting $S \to \infty$.

Theorem 2. *Given $supp[P(Y_{multi})] \subseteq supp\left[P\left(Y_{many}\right)\right]$ and Assumptions 2 & 3, we have,*

$$\lim_{S\to\infty}\left(\mathbb{E}\left[\left(x_d - \mathbb{E}\left[g_d(\boldsymbol{y})|D_{many}\right]\right)^2\right]\right) \leq \lim_{S\to\infty}(\mathbb{E}[(x_d - \mathbb{E}[g_d(\boldsymbol{y})|D_{multi}])^2]),$$

where x_d is the target value at an arbitrary test point $\boldsymbol{y}$.

Proof. For a GP with correctly specified prior, the expected generalization error at any test point $\boldsymbol{y}$ is identical to the predictive variance of the model at that point [20]. Thus, in order to prove the theorem, it suffices to show that $\lim_{S\to\infty}\mathrm{var}\left[g_d(\boldsymbol{y})|D_{many}\right] \leq \lim_{S\to\infty}\mathrm{var}[g_d(\boldsymbol{y})|D_{multi}]$. We shall arrive at this inequality from an information theoretic perspective. Specifically, since our model predictions are Gaussian distributions, we may also write the inequality as $\lim_{S\to\infty}\mathrm{H}(g_d(\boldsymbol{y})|D_{many}) \leq \lim_{S\to\infty}\mathrm{H}(g_d(\boldsymbol{y})|D_{multi})$, where $\mathrm{H}(\cdot)$ represents differential entropy.

From the Glivenko-Cantelli theorem [21], it follows that for $S \to \infty$ the empirical distributions of Y_{multi} and Y_{many} converge to the true underlying distributions $P(Y_{multi})$ and $P\left(Y_{many}\right)$. Since $supp[P(Y_{multi})] \subseteq supp\left[P\left(Y_{many}\right)\right]$, it can then be said that $Y_{multi} \subseteq Y_{many}$, which, together with Assumption 2, implies that $D_{multi} \subseteq D_{many}$.

Based on the above, let $Y_{many} \backslash Y_{multi} = Y_{diff}$; accordingly, $X_{many} \backslash X_{multi} = X_{diff}$. Note that $Y_{multi} \backslash Y_{many} = \emptyset$. Now, consider the following conditional mutual information defined for the posterior multivariate Gaussian distribution of an inverse GP model with known prior,

$$\mathrm{I}(g_d(\boldsymbol{y}); Z|D_{multi}) = \mathrm{H}(g_d(\boldsymbol{y})|D_{multi}) - \mathrm{H}(g_d(\boldsymbol{y})|Z, D_{multi}), \tag{12}$$

where $Z = \{(\boldsymbol{x}_{s,d}, \boldsymbol{y}_s)\colon\ \forall \boldsymbol{y}_s \in Y_{diff}\}$. Since,

$$\mathrm{H}(g_d(\boldsymbol{y})|Z, D_{multi}) = \mathrm{H}(g_d(\boldsymbol{y})|D_{many}), \qquad (13)$$

and given the non-negativity of mutual information, we have,

$$\mathrm{H}(g_d(\boldsymbol{y})|D_{multi}) - \mathrm{H}(g_d(\boldsymbol{y})|D_{many}) \geq 0. \qquad (14)$$

The statement of the theorem is implied. ■

Theorem 2 serves as the crux of the conceived blessing of dimensionality. Recall that as the input features and output labels of datasets D_{multi} and D_{many} are identical, the expected performance gain is primarily a consequence of the differing data distributions induced during either a multi- or many-objective search. Still, the result has been derived under certain assumptions that may not necessarily hold in practice. For instance, while training the inverse GP models, it is advisable to utilize all the data made available during the evolutionary optimization runs, without any exclusion. Further, conditions such as the one-to-one mapping between the decision and objective spaces are not always satisfied. Nevertheless, in the next section, we show that despite the simplifying assumptions and approximations that have been made, the derived theoretical inferences are borne out experimentally as well.

V. Empirical Verification

Here, we present experimental results for a toy example as well as a real-world case study in composite manufacturing. Problem descriptions are provided first, followed by comparisons of inverse modeling accuracy based on multi- and many-objective search data.

A. Problem Descriptions

1) Convex variant of a DTLZ test function

The scalable $DTLZ$ benchmark functions are a widely used test suite for analyzing the optimization performance of MaOEAs [22]. Notably, the majority of test functions in the suite are characterized by concave PFs. In this regard, as a linear objective aggregation of the type shown in Eq. (2) is known to be incompatible for concave PFs (since not every Pareto optimal solution to the original problem formulation can be obtained by varying $\boldsymbol{w}$), we consider a convex variant of $DTLZ$. As proposed in [23], we convexify the DTLZ2 function by multiplying its objective values by -1. The resultant test problem is denoted as $DTLZ2^{-1}$.

The statement of the basic $DTLZ2$ minimization problem for arbitrary m is as follows,

$$\begin{gathered}
\min f_1(\boldsymbol{x}) = (1+\varphi(\boldsymbol{x}_m))\cos\left(\frac{x_1\pi}{2}\right)\cos\left(\frac{x_2\pi}{2}\right)\ldots\cos\left(\frac{x_{m-2}\pi}{2}\right)\cos\left(\frac{x_{m-1}\pi}{2}\right)\\
\min f_2(\boldsymbol{x}) = (1+\varphi(\boldsymbol{x}_m))\cos\left(\frac{x_1\pi}{2}\right)\cos\left(\frac{x_2\pi}{2}\right)\ldots\cos\left(\frac{x_{m-2}\pi}{2}\right)\sin\left(\frac{x_{m-1}\pi}{2}\right)\\
\min f_3(\boldsymbol{x}) = (1+\varphi(\boldsymbol{x}_m))\cos\left(\frac{x_1\pi}{2}\right)\cos\left(\frac{x_2\pi}{2}\right)\ldots\sin\left(\frac{x_{m-2}\pi}{2}\right)\\
\vdots\\
\min f_{m-1}(\boldsymbol{x}) = (1+\varphi(\boldsymbol{x}_m))\cos\left(\frac{x_1\pi}{2}\right)\sin\left(\frac{x_2\pi}{2}\right)\\
\min f_m(\boldsymbol{x}) = (1+\varphi(\boldsymbol{x}_m))\sin\left(\frac{x_1\pi}{2}\right)\\
0 \leq x_i \leq 1, \forall i \in \{1,2,\ldots,n\}, \text{and } \varphi(\boldsymbol{x}_m) = \textstyle\sum_{i=m}^{n} {x_i}^2. \qquad (15)
\end{gathered}$$

By multiplying $f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x})$ by -1, Eq. (15) is turned into a maximization problem with a convex PF.

In our experiments, we set $m = 6$ and $n = 8$ as the primary many-objective formulation of $DTLZ2^{-1}$. We denote this setting as $DTLZ2^{-1}_{6,6}$. In addition, we consider compressed reformulations of the same problem, constructed via linear objective aggregations. Specifically, a synthetic reformulation, denoted as $DTLZ2^{-1}_{6,r}$ for $r < 6$, is constructed by retaining the first $r-1$ objectives in $DTLZ2^{-1}_{6,6}$ and scalarizing the remaining $7-r$ objectives by assigning them equal weightage. Notice that for the many-objective maximization problem $DTLZ2^{-1}_{6,6}$ and all its compressed reformulations, the convex PF is always characterized by $\boldsymbol{x}_m = \mathbf{1}$ in Eq. (15).

2) Composite manufacturing case study

Compression resin transfer molding (or CRTM for short) is a popular method for high volume production of fiber-reinforced polymer composite parts. In order to simultaneously improve the cost-effectiveness, throughput, and quality of the CRTM cycle, a number of process parameters must be appropriately tuned keeping in view associated objective functions. For any given configuration of process parameters, the corresponding performance is evaluated using a numerical resin flow simulation engine [24]. We employ low-fidelity simulations in this paper in order to speedup evaluations.

In this paper, we consider the manufacture of a simple circular composite plate made of glass-fiber reinforced epoxy resin. The exact part dimensions and material properties required for the numerical flow simulations can be found in [25]; these details are omitted herein for the sake of brevity. The overall CRTM process introduces two separate manufacturing phases, namely, the *filling phase* followed by the *curing phase*. During filling, the *time required to inject the liquid resin* through the densely packed fibrous reinforcement placed inside a metallic mold (which has been precisely machined according to the geometry of the part) is to be minimized. At the same time, the *internal forces* generated in the mold due to fiber compression together with the pressure of resin injection must be kept at a minimum – as the maximum force determines the size and cost of peripheral equipment needed. During curing, the *time required for the resin to cure* (i.e., to turn from liquid to solid) is to be minimized by controlling the thermal conditions inside the mold; further, the *thermal and cure gradients* induced within the part must be kept at minimum for ensuring good part quality. Accordingly, for the manufacturing cycle as a whole, we have the following five objective functions of interest,

$$\min\left[time_{filling}, force, time_{curing}, grad_{thermal}, grad_{cure}\right]. \qquad (16)$$

There are eight decision variables to be optimized (i.e., $n = 8$), spanning the thermal control of the resin and the metallic mold, the liquid injection pressure, and the rate at which the peripheral equipment is to be operated. Each decision variable has been scaled to the range $[0,1]$, which makes the mean squared Euclidean distance in decision space a viable error measure for the inverse model predictions.

As $m \geq 4$, Eq. (16) is categorized as an MaOP which is hard to solve using vanilla MOEAs. A common approach to

simplify the formulation and to aid performance visualization is to reduce the objective space dimensionality via the following aggregations,

$$time_{cycle} = time_{filling} + time_{curing}, \quad (17)$$

$$gradient_{agg} = w_1 \cdot grad_{thermal} + w_2 \cdot grad_{cure}. \quad (18)$$

This turns Eq. (16) into a three-objective optimization problem that falls within the ambit of MOEAs.

B. Settings for the Experimental Tests

All the MOEA / MaOEA runs are conducted using the popular NSGA-III solver [6], provided by the PlatEMO toolbox [26]. No changes are made to the default parameter settings of the toolbox. A population of 200 solution samples is deployed for the test $DTLZ2_{6,r}^{-1}$ functions, and 50 samples are used in the real-world case study. The populations are evolved for a period of 250 iterations in all experiments, ensuring that the multi- and many-objective search data are of same size. In the face of the large datasets generated, the computational tractability of GP-based inverse modeling is maintained by applying the method of POEs (as discussed in Section III-B) with different numbers of GP experts. GPs with the squared exponential kernel have been used throughout.

In order to examine the efficacy of the trained inverse models, their performance generalization is verified on separate test sets. In the case of the $DTLZ2_{6,r}^{-1}$ benchmark, a test set of 2500 data samples is used. Keeping in mind the goal of accurately capturing even sub-optimal solutions within a region of the DM's preference (as reasoned in the introduction), the test samples have been obtained via Latent hypercube sampling (LHS) in the vicinity of the known PS. This is achieved by constraining LHS to the range $[\mathbf{0.8}, \mathbf{1}]$ with respect to $\boldsymbol{x}_m$; all other decision variables are sampled via unbiased LHS. Constraining the test samples in the aforementioned manner can be thought of as representing a real scenario in which achieving the optimal value of $\boldsymbol{x}_m$ (i.e., $\boldsymbol{x}_m = \mathbf{1}$) becomes infeasible due to sudden resource exhaustion. An inverse model that is accurate over such a test set thus offers a DM the flexibility to reconfigure her decisions in real-time, bypassing the need to rerun the optimization engine.

For the composite manufacturing case study, since no prior knowledge about the location of the PS is available given the black box nature of the problem, a test dataset consisting of 1000 samples has been created via entirely unbiased LHS.

C. Numerical Results

1) The pedagogical DTLZ example

A bar chart summarizing the accuracy of the inverse GP modeling is depicted in Fig. 1. Notice that across both sets of experiments, comprising 10 and 25 GP experts, respectively, the general inference drawn in Section IV is borne out. As was theoretically anticipated, the inverse prediction accuracy conditioned on the many-objective $DTLZ2_{6,6}^{-1}$ data is visibly better than that of its compressed optimization counterparts. Moreover, in Fig. 1, we see that as the extent of compression (based on objective aggregation) increases, the mean squared error measure monotonically deteriorates.

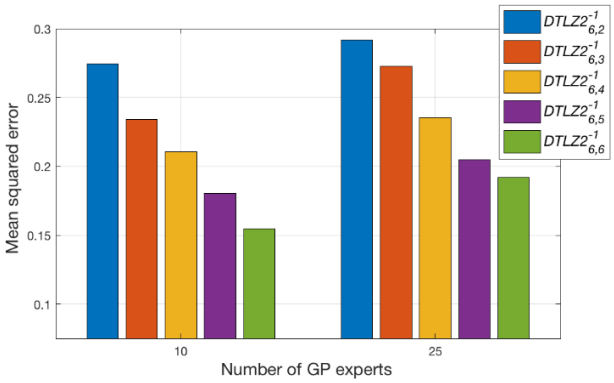

Fig. 1. Comparison of mean squared error of inverse GP modeling under multi- and many-objective formulations of the $DTLZ2_{6,r}^{-1}$ function. As the extent of objective aggregation decreases (i.e., as we transition from multi-to many-objective search), the performance steadily improves.

The afore-stated observation is in fact a direct corollary of Theorem 2, as the formulation of $DTLZ2_{6,r}^{-1}$, for any r, can be derived through a linear objective aggregation of its immediate predecessor $DTLZ2_{6,r+1}^{-1}$. It therefore follows that the expected prediction error of an inverse model trained on $DTLZ2_{6,r+1}^{-1}$ is bounded from above by the error of an equivalent model trained on $DTLZ2_{6,r}^{-1}$; this is verified in Fig. 1.

As an aside, we also draw attention to the outcome that the overall accuracy of the POE diminishes by ~20% as we increase the number of GP experts from $Q = 10$ to $Q = 25$. Thus, in practice, simultaneously ensuring real-time as well as precise inverse generative modeling requires an informed choice of Q. Although the prediction complexity scales down with Q to $\mathcal{O}(S^2/Q^2)$ in time (per expert), there is a cost to be paid in terms of the achievable predictive performance that must also be kept in mind.

2) The CRTM process engineering example

The real-world study gives rise to qualitatively similar results. A summary is depicted in Fig. 2 (top panel). Three sets of experiments were carried out with 10, 50, and 100 GP experts (in the POE), respectively. The outcome of each of the experimental settings shows that many-objective search data from a general five-objective formulation of CRTM leads to significantly lower mean squared error of the corresponding inverse model. In particular, an accuracy improvement of ~28.5% is achieved on average across the three settings; individual performance gains are reported in Fig. 2 (bottom panel). Notice that the performance gain tends to diminish with increasing number of GP experts in the POE, reinforcing our previously suggested need to select Q with care.

Inverse model accuracy in applications of this type provides a strong motivation for many-objective search as a form of effective decision space exploration. In engineering design, for example, a DM may first apply an MaOEA to obtain a crude approximation of the PF in high dimensions; following this, the learned inverse model can be utilized to rapidly focus the search within any given region of preference. In addition, if any change in the problem setting occurs, the same model may be deployed to generate new candidate solutions in real-time.

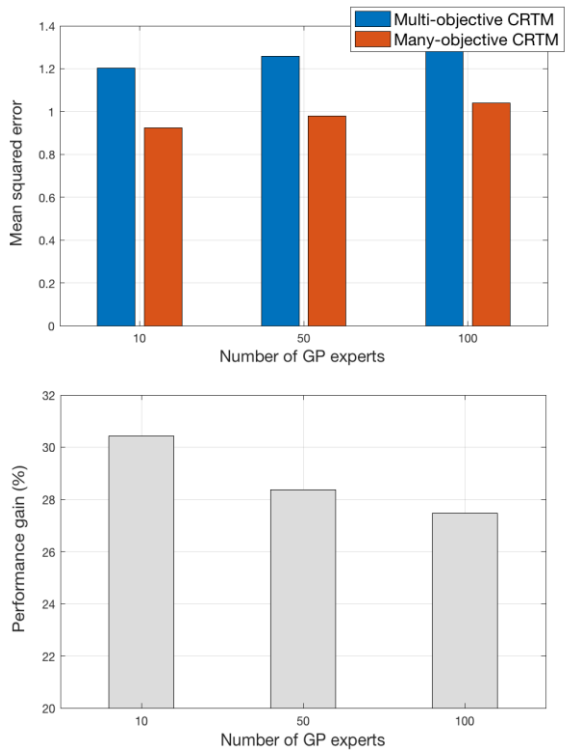

Fig. 2. Comparison of mean squared error of inverse GP modeling under the multi- and many-objective formulations of the CRTM problem (top panel). As the number of GP experts in the POE increases, the inverse modeling accuracy gain achieved as a consequence of many-objective search data tends to gradually diminish (bottom panel).

VI. Conclusions

In this paper, we unveiled and analyzed a manifestation of the blessing of dimensionality that is yet to highlighted in the optimization literature. *In particular, we focused on GP-based inverse models as tools for DMs to rapidly focus search within any region of interest, enabling them to configure preferred solutions on the fly in a posteriori preference articulation scenarios*. Through simple theoretical and experimental results, we showed that inverse models conditioned on many-objective search data are indeed more accurate than those obtained after searching in synthetically aggregated objective spaces. This observation provides strong motivation for performing many-objective searches in practice, a domain that has previously been encumbered by the curse of dimensionality.

Acknowledgment

This work was supported in part by the Data Science and Artificial Intelligence Research (DSAIR) Center at Nanyang Technological University, Singapore.

References

[1] Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms* (Vol. 16). John Wiley & Sons.

[2] Trivedi, A., Srinivasan, D., Sanyal, K., & Ghosh, A. (2016). A survey of multiobjective evolutionary algorithms based on decomposition. *IEEE Transactions on Evolutionary Computation*, *21*(3), 440-462.

[3] Holland, J. H. (1992). Genetic algorithms. *Scientific American*, *267*(1), 66-73.

[4] Li, B., Li, J., Tang, K., & Yao, X. (2015). Many-objective evolutionary algorithms: A survey. *ACM Computing Surveys (CSUR)*, *48*(1), 13.

[5] Ishibuchi, H., Tsukamoto, N., & Nojima, Y. (2008, June). Evolutionary many-objective optimization: A short review. In *2008 IEEE Congress on Evolutionary Computation* (pp. 2419-2426). IEEE.

[6] Deb, K., & Jain, H. (2013). An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. *IEEE Trans. Evol. Comput.*, *18*(4), 577-601.

[7] Cheng, R., Jin, Y., Olhofer, M., & Sendhoff, B. (2016). A reference vector guided evolutionary algorithm for many-objective optimization. *Trans. Evol. Comput.*, *20*(5), 773-791.

[8] Cai, X., Yang, Z., Fan, Z., & Zhang, Q. (2016). Decomposition-based-sorting and angle-based-selection for evolutionary multiobjective and many-objective optimization. *IEEE Transactions on cybernetics*, *47*(9), 2824-2837.

[9] Gupta, A., & Ong, Y. S. (2019). Back to the roots: Multi-*X* evolutionary computation. *Cognitive Computation*, *11*(1), 1-17.

[10] Giagkiozis, I., & Fleming, P. J. (2014). Pareto front estimation for decision making. *Evolutionary computation*, *22*(4), 651-678.

[11] Donoho, D. L. (2000). High-dimensional data analysis: The curses and blessings of dimensionality. *AMS math challenges lecture*, *1*(2000), 32.

[12] Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000, September). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *International conference on parallel problem solving from nature* (pp. 849-858). Springer, Berlin, Heidelberg.

[13] Cheng, R., Jin, Y., Narukawa, K., & Sendhoff, B. (2015). A multi-objective evolutionary algorithm using Gaussian process-based inverse modeling. *IEEE Trans. Evol. Comput.*, *19*(6), 838-856.

[14] Bonilla, E. V., Chai, K. M., & Williams, C. (2008). Multi-task Gaussian process prediction. In *Advances in neural information processing systems* (pp. 153-160).

[15] Chai, K. M. (2009). Generalization errors and learning curves for regression with multi-task Gaussian processes. In *Advances in neural information processing systems* (pp. 279-287).

[16] Da, B., Ong, Y. S., Gupta, A., Feng, L., & Liu, H. (2019). Fast transfer Gaussian process regression with large-scale sources. *Knowledge-Based Systems*, *165*, 208-218.

[17] Deisenroth, M. P., & Ng, J. W. (2015, July). Distributed Gaussian processes. In *Proceedings of the 32nd International Conference on Machine Learning-Volume 37* (pp. 1481-1490).

[18] Perrone, M. P., & Cooper, L. N. (1995). When networks disagree: Ensemble methods for hybrid neural networks. In *How We Learn; How We Remember: Toward an Understanding of Brain and Neural Systems: Selected Papers of Leon N Cooper* (pp. 342-358).

[19] Liu, H., Cai, J., Wang, Y., & Ong, Y. S. (2018, July). Generalized Robust Bayesian Committee Machine for Large-scale Gaussian Process Regression. In *International Conference on Machine Learning* (pp. 3137-3146).

[20] Williams, C. K., & Vivarelli, F. (2000). Upper and lower bounds on the learning curve for Gaussian processes. *Machine Learning*, *40*, 77-102.

[21] Tucker, H. G. (1959). A generalization of the Glivenko-Cantelli theorem. *The Annals of Mathematical Statistics*, *30*(3), 828-830.

[22] Deb, K., Thiele, L., Laumanns, M., & Zitzler, E. (2005). Scalable test problems for evolutionary multiobjective optimization. In *Evolutionary multiobjective optimization* (pp. 105-145). Springer, London.

[23] Ishibuchi, H., Setoguchi, Y., Masuda, H., & Nojima, Y. (2016). Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes. *IEEE Trans. Evol. Comput.*, *21*(2), 169-190.

[24] Gupta, A., Kelly, P. A., Bickerton, S., & Walbran, W. A. (2012). Simulating the effect of temperature elevation on clamping force requirements during rigid-tool liquid composite moulding processes. *Composites Part A: Applied Science and Manufacturing*, *43*(12), 2221-2229.

[25] Gupta, A., Mańdziuk, J., & Ong, Y. S. (2015). Evolutionary multitasking in bi-level optimization. *Complex & Intelligent Systems*, *1*(1-4), 83-95.

[26] Tian, Y., Cheng, R., Zhang, X., & Jin, Y. (2017). PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum]. *IEEE Computational Intelligence Magazine*, *12*(4), 73-87.