Computers and Electronics in Agriculture 224 (2024) 109173

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

Feed formulation using multi-objective Bayesian optimization

Gabriel D. Uribe-Guerra [a,*], Danny A. Múnera-Ramírez [a], Julián D. Arias-Londoño [b]

[a] Intelligent Information Systems Lab, Department of Systems Engineering, Universidad de Antioquia, Calle 67 No. 53 - 108, 050010, Medellín, Colombia
[b] Department of Signals, Systems and Radiocommunications, ETSI Telecomunicación, Universidad Politécnica de Madrid, Av. Complutense, 30, 28040, Madrid, Spain

ARTICLE INFO

Dataset link: https://github.com/GabrielUribe29/MOBO-feed

Keywords:
Multi-objective Bayesian optimization
Food production
Precision agriculture
Swine diet design

ABSTRACT

Animal diet design has been addressed mainly by optimizing analytical functions that describe digestible energy and essential nutrients, along with a set of restrictions regarding minimum nutritional content in the feed formulation. This approach results in limitations since theoretical models are not flexible enough to incorporate variables related to environmental or zootechnical conditions that affect production efficiency or to include multiple objectives regarding current challenges associated with the adaptability to new environmental contexts and the reduction of ecological footprint. Unlike analytical methods, heuristic approaches can deal with variables from multiple sources using surrogate data-driven models of the objectives functions but commonly require thousands of evaluations of the target function, which is unfeasible in the context of animal diet formulation. This work proposes the use of Bayesian Optimization as an alternative solution to address the animal diet design problem since it is intended to optimize costly-to-evaluate target functions and is able to deal with noisy sampling, which is helpful in handling the intrinsic variability in the nutrient content of raw materials. A multi-objective swine diet design problem is used to evaluate the suitability of Bayesian optimization to optimize three target functions: digestible energy, lysine, and cost, and the solutions are compared with a fractional stochastic programming approach. The analytical formulation of the problem is not considered by the Bayesian optimization approach, but target functions are modeled through surrogate Bayesian models, where only input and output responses are used to drive the optimization process. Results show that a multi-objective Bayesian optimization process is able to find better solutions than previously proposed methods, improving in 10.71%, 14.77%, and 3.79% the three objectives defined. Experiments using batches of query samples per iteration show that the optimization process can also be accelerated by sampling the objective functions simultaneously.

1. Introduction

The livestock industry confronts challenges due to the persistent growth witnessed in global food demand and climate change (Van der Poel et al., 2020). Projections indicate that the world population is poised to expand by 34% over the next two decades, thereby engendering a corresponding surge in the demand for food by approximately 70% (Sharma et al., 2020). This surge is exacerbated by the rapid pace of urbanization, which, in turn, generates a decline in the availability of arable land for agricultural purposes. Additionally, to maintain the cost-effectiveness of food production, farmers must now grapple with additional hurdles, such as preserving production efficiency, adapting their production to different environmental contexts, ensuring the adequacy of nutrients in livestock rations, and trying to reduce the industry's ecological footprint (Pomar et al., 2021). All these factors make the feed formulation an increasingly complex problem.

Specifically regarding the adequacy of diets for pigs, several factors must be considered to determine the efficiency with which animals metabolize and convert nutrients. On the one hand, the characteristics of each animal, marked by inherent variability and complexities of individual metabolic processes, acquire fundamental importance (Pomar et al., 2021). On the other hand, due to differences in the production of raw materials and sources of agricultural by-products, farmers have to deal with an intrinsic variation in the nutrient content of the food to be supplied, meaning that although the formulation is the same, the amount of nutrients in it may differ. Such variability, in turn, changes the quality and composition depending on the suppliers' origin and cultivation conditions. Added to this are other factors, such as climatic and agronomic considerations, that challenge feed producers to balance ingredients to obtain the minimum amounts of required nutrients to

* Corresponding author.
E-mail address: gdario.uribe@udea.edu.co (G.D. Uribe-Guerra).

https://doi.org/10.1016/j.compag.2024.109173
Received 14 December 2023; Received in revised form 19 April 2024; Accepted 11 June 2024

ensure the quality of their products and to guarantee the efficiency of the farms (Amit et al., 2017).

Since the mid-20th century, there has been a concerted effort among farmers and livestock feed producers to optimize the feed efficiency of animals. The goal is to ascertain the optimal combination of available ingredients that, when blended, fulfill the minimum essential nutrient requirements, resulting in economic advantages (Babić and Perić, 2011). Initially, methods involved the compilation of tables integrating ingredient-nutrient data and ingredient prices. Numerous numerical optimization techniques were applied to derive feed mix designs that align with specified nutritional requirements. However, despite their theoretical soundness, these formulations often encountered practical constraints in real-world applications (Pratiksha et al., 2011).

An early exploration of this approach was conducted by Waugh (1951), who employed Linear Programming (LP) to identify the most cost-effective combination of feeds meeting the minimum requirements for dairy cows. This pioneering work set the groundwork for subsequent studies utilizing LP. For example, Pesti and Miller (1993) employed LP to devise cost-effective feeds for poultry, sheep, and cattle, incorporating feed substitution techniques. LP was also utilized by Nyhodo et al. (2014) to establish the least-cost feed mixtures, considering fluctuations in ingredient prices for poultry formulations. Beyond cost reduction, LP objectives and constraints have undergone extensive examination, encompassing diet design for various breeds (Chappell, 1974), ensuring weight gain at different animal life stages (Glen, 1980), and adapting diets to non-traditional foods such as legumes and forages. This methodology empowers small-scale farmers to determine daily diets for their herds based on locally available products (Sebastian et al., 2008).

Notwithstanding its impact, the application of LP to diet design for biological systems reveals significant limitations. One major limitation lies in LP's exclusive focus on optimizing a single objective, which is insufficient for capturing biological systems' intrinsic complexity during diet design. Another highlighted limitation relates to the constraints expressed solely through linear equations or inequalities. This linear approach may lead to suboptimal solutions due to an inability to effectively identify nutrient imbalances and interactions. This deficiency can result in adverse consequences, such as unnecessary nutrient addition leading to extra costs, undetected deficiencies causing underfeeding, or excessive excretions in animals due to overconsumption of nutrients (Ghosh et al., 2014). Moreover, the previously pointed out nutrient variations in some ingredients pose an additional challenge that, combined with inherent LP limitations, can result in significant economic losses in the industry, as indicated in Peña et al. (2009).

To overcome these limitations, adjustments to LP have been explored, particularly focusing on constraints. An emerging alternative is Stochastic Programming (SP), which accounts for variations in nutrient composition in food. Studies indicate that SP enhances precision and reliability in formulating livestock feed mixtures (D'Alfonso et al., 1992). Besides, recent research (Patil et al., 2022) incorporates the probability of variation in key nutrients like crude protein, calcium, and phosphorus to minimize costs. In Peña et al. (2009), authors also integrated digestible energy and lysine cost functions into a fractional SP-based optimization analysis and were able to find successful formulations in comparison to an LP-based method, although with a strong manual intervention through the iterations of the optimization process, which limits its applicability.

In addition to SP, the ability to optimize various objectives has also been studied. The considered objectives include energy density, weight maximization, reduction of excretions, greenhouse gas reduction, variation in nutrient reduction, conversion minimization, and productivity maximization, among others (D'Alfonso et al., 1992; Ghosh et al., 2014), and some of them have been addressed simultaneously. Multi-Objective Optimization (MOO) showcases several advantages over other approaches since considering multiple objectives may lead to formulating economically sound, sustainable, and environmentally friendly feed compositions. Furthermore, MOO can contribute to cost reduction in formulations without compromising animal growth and health (Uyeh et al., 2019), even though it imposes new challenges to find feasible mathematical methods able to deal with the entire set of problem constrictions.

Aiming to deal with an increasingly complex mathematical and numerical problem formulation, in recent years, several works have focused on employing heuristic techniques as an alternative to address the intricacies associated with diet design (Innocent et al., 2023). These algorithms explore the search space by sampling solutions, evaluating their fitness, and biasing the search in the direction of promising solutions (Naharro et al., 2022). The recourse to heuristics is driven by the exigencies of handling complex dietary composition challenges and the imperative to derive efficient solutions within a reasonable timeframe, rendering them particularly well-suited for real-life applications. Moreover, these heuristic methods exhibit an inherent flexibility and adaptability that renders them amenable to diverse instances of dietary formulation problems, thereby facilitating the efficient exploration of the search space (Lisitsyn et al., 2023). An additional salient advantage is their capacity to accommodate large-scale problems, thereby enabling facile adaptation to the considerable variability inherent in raw materials and nutritional variables, as well as the consideration of non-dietary factors (Wang and Liao, 2023).

A noteworthy heuristic approach employed in diet design is the use of genetic algorithms (GA). In particular, GAs have been leveraged to optimize feed mixtures (Şahman et al., 2009), and successful implementations have been recorded in the context of poultry and livestock nutrition. This application has also been evaluated through comparative analyses with conventional LP approaches, underscoring the effectiveness of GA in addressing diet design challenges and adapting to changing parameters. Another heuristic method tested in the context of feed formulation is particle swarm optimization, which has also been employed in optimizing mixed feeding in cattle, sheep, and rabbits and has shown more stability than the LP-based solutions (Altun and Şahman, 2013).

An additional advantage of heuristic methods is their flexibility to carry out optimization processes on complex analytic/non-analytic nonlinear problems or to combine variables from different sources through the use of surrogate data-driven models (Naharro et al., 2022). In the context of diet design, this would allow the algorithm to break the limitations of classical approximations based on theoretical models of animal metabolisms and potentially include environmental and zootechnical variables that also influence animal performance. However, building an accurate surrogate model typically requires extensive sampling of the cost function under analysis, which is unfeasible for animal diet design. An alternative is to use an approach that allows the optimization process to be carried out simultaneously with the training of the substitute model in such a way that the objective function only has to be sampled in promising positions according to the current model, thus reducing the number of necessary evaluations and at the same time allowing the optimization problem to be solved. A technique that has these characteristics is Bayesian optimization (BO) (Frazier, 2018a).

BO is a class of machine-learning-based optimization methods suitable for objective functions that take a long time to evaluate (Frazier, 2018a). It uses a Bayesian surrogate model that provides predictions and uncertainty measures over the predictions to build an alternative objective function (called acquisition function), which can be optimized to get the next candidate for sampling the cost function. In some sense, BO has relationships with the concept of active learning, but instead of accurately reconstructing a target function f, its aim is to get the optimum value of f, sampling as few times as possible. Another advantage is that BO tolerates stochastic noise in function evaluations and can deal with complex constraints, which makes it suitable for modeling the intrinsic variability of nutrient content in the context of feed formulation. Moreover, BO has shown flexibility in handling

variable types, accommodating continuous and discrete variables, allowing simultaneous optimization of multiple objectives, and allowing the specification of diverse optimization constraints (Garnett, 2023). Recently, BO has been used to address problems in complex agricultural tasks, such as seed germination (Nikitin et al., 2019) and crop model calibration (Moon et al., 2023). However, to the best of our knowledge, no previous research has been conducted to analyze the adaptability of BO to feed formulation problems.

Bearing all this in mind, this work focuses on the evaluation of a BO approach to optimize a previously formulated multi-objective problem of swine diet design, consisting of 17 input variables associated with the proportion of ingredients in the formulation, 28 constraints compromising minimum nutritional requirements or zootechnical decisions, and three objectives: energy content, lysine, and cost. The results are compared with a previously proposed solution based on fractional SP (Peña et al., 2009). A detailed analysis of relevant hyperparameters, such as the number of initial samples and required iterations to get stable solutions, is carried out to establish the proposed solution's feasibility. Two acquisition functions were compared: the expected Hypervolume improvement and its noisy version (Daulton et al., 2021). The set of non-dominate solutions provided by the BO algorithm is evaluated in terms of hypervolume, cardinality, uniformity, and spread. Besides, a complete evaluation of the quality of the BO-based solutions regarding those proposed in Peña et al. (2009) is also presented. While the addressed task pertains to optimizing an analytical rendition of the swine diet design issue, it serves as the initial stride in verifying the suitability of BO for resolving a broader scenario where the cost function sampling involves testing a specific diet design to nourish a pig herd, and variables from multiple sources are combined to model the pig performance. Thus, the analytical formulation of the problem is not considered by the BO approach at all, but target functions are modeled through surrogate Bayesian models, where only input and output responses are used to drive the optimization process. The rest of the paper is organized as follows: Section 2 presents the mathematical formulation of the objective problem and describes in more detail the solution proposed in Peña et al. (2009); Section 3 introduces BO, its core components, and the Multi-objective variant; Section 4 describes the set of experiments, the evaluation metrics and presents the results obtained. Lastly, Section 5 provides some conclusions derived from the results.

2. Problem statement

In Peña et al. (2009), the authors introduced a multi-objective optimization problem where two key variables regarding the nutritional content and the cost were selected as objectives. The purpose is to maintain the minimum amount of required nutrients in diets for pigs, guaranteeing economic sustainability. Lysine and energy were identified as integral components in ensuring the nutritional adequacy of the diet. Lysine is an indispensable amino acid vital for food metabolism and protein synthesis, it plays a vital role in facilitating animal growth. Simultaneously, energy is essential for sustaining fundamental bodily functions. The intricate interdependence of these two nutrients is highlighted by research indicating that an imbalanced diet with inadequate lysine and energy content can constrain the performance of pigs (Cho et al., 2012).

Hence, the optimization problem endeavors to identify the optimal combination of ingredients, specifically their proportions within the diet, with the objective of maximizing the availability of both lysine and energy while minimizing costs. This task involves imposing additional constraints on various nutritional prerequisites, including but not limited to crude fiber, phosphorus, calcium, and crude protein, as outlined in Peña et al. (2009). The set of ingredients, along with their nutritional content and costs, was defined based on the specifications provided by the Spanish Foundation for the Development of Animal Nutrition (FEDNA for its Spanish acronym) in 1999, and can be consulted in Peña et al. (2009). For the sake of comparison, the same tables of nutrient content and cost were used in this work.

Formally, the problem can be formulated as

$$\begin{aligned} \min \quad & f_c(\mathbf{x}) = \mathbf{c}^T\mathbf{x} \\ \max \quad & f_l(\mathbf{x}) = \mathbf{l}^T\mathbf{x} \\ \max \quad & f_e(\mathbf{x}) = \mathbf{e}^T\mathbf{x} \\ \text{s.t} \quad & \mathbf{1}^T\mathbf{x} = 1 \\ & \underline{\mathbf{b}} \le \mathbf{A}^T\mathbf{x} \le \overline{\mathbf{b}} \\ & 0 \le \mathbf{x} \le \mathbf{s} \end{aligned} \tag{1}$$

where $\mathbf{x} = [x_1, x_2, \dots, x_i, \dots, x_d]$ represents a solution vector containing the proportions of each ingredient in a diet composed of d ingredients, $\mathbf{A}$ is a matrix that describes the nutrient content of each ingredient regarding a set of a nourishments, $\mathbf{c}$ is an d-dimensional vector denoting the costs associated with each ingredient, $\mathbf{l}$ and $\mathbf{e}$ represent vectors containing the lysine and energy content of each ingredient, respectively. Additionally, $\underline{\mathbf{b}}$ and $\overline{\mathbf{b}}$ are vectors specifying the lower and upper bounds for each nutrient, while $\mathbf{s}$ is a vector that represents the upper bounds for each raw material in the diet formulation process. As previously mentioned, $\mathbf{A}$, $\mathbf{l}$ and $\mathbf{c}$ are defined in accordance with FEDNA tables (Peña et al., 2009). Following the original formulation by Noblet (Noblet and Van Milgen, 2004), the model of digestible energy f_e uses the equation linking the nutritional content in the diet with the energy, which defines the vector $\mathbf{e}$. The complete list of ingredients and nutrients described in Peña et al. (2009) is included in Appendix.

Peña and her coworkers proposed to address the former optimization formulation as a Multi-Objective Fractional Problem (MFP) (Peña et al., 2009), which focuses on the development of economically viable animal diets, mitigating nutrient variability within the raw materials. MFP incorporates considerations of both ratio costs and the probabilities associated with meeting the animal's nutrient requirements. The probability of attaining the desired lysine content in a food ratio can be defined as $P(f_l(\mathbf{x}) \ge \gamma_l)$, with γ_l representing the lysine requirement in the feed. Similarly, the probability of achieving the desired energy content is $P(f_e(\mathbf{x}) \ge \gamma_e)$, where γ_e denotes the minimum energy requirement.

For solving the MFP, unlike a standard SP approach, the authors proposed an interactive approach involving a decision-maker who articulates preferences without requiring prior knowledge of the minimum probabilities to satisfy the nutritional requirements. This method, called Interactive Multi-Objective Goal Programming (IMGP), is detailed in Spronk and Spronk (1981).

In the IMGP approach, the three objectives are initially optimized independently, establishing optimal and least favorable values for each. These values set the range for defining objective goals in subsequent iterations. The decision-maker then selects an objective for improvement, and the objectives are optimized again, considering constraints from the initial problem and new ones based on the defined limits. The decision-maker assesses whether improvements in the selected objective justify potential modifications in others; adjustments are made accordingly. Constraints evolve in each iteration, progressively narrowing the set of feasible solutions until the lower and upper limits of objectives converge, signifying an efficient solution where no alternative improves one objective without worsening another.

3. Methods

This section presents all the methods and techniques used to perform the optimization of the swine diet formulation problem presented in the previous section using BO. Fig. 1 shows a schematic of the proposed methodology to help the reader identify each component and how it is used throughout the whole process.

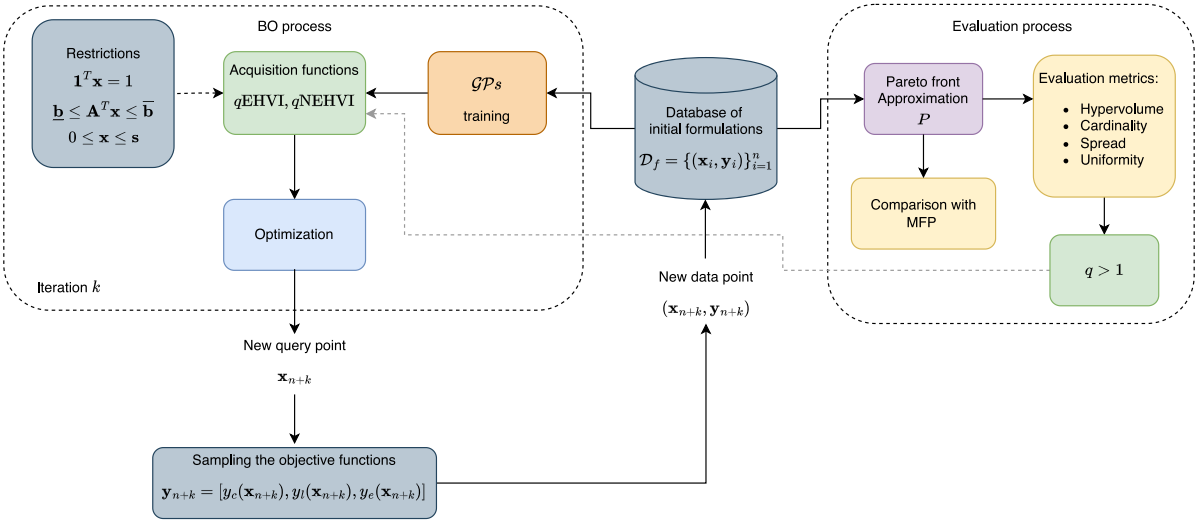

Fig. 1. General scheme of the proposed methodology.

3.1. Bayesian optimization

BO is a machine learning-based optimization method specifically designed for scenarios in which the objective function $f(\mathbf{x})$ is continuous, expensive to evaluate, lacks an analytic formulation, and where access to first and second-order derivatives of f is not feasible (Frazier, 2018b). BO consists of two main components: a Bayesian surrogate model for modeling the objective function and an acquisition function (AF) for deciding where to sample next. The general idea behind BO algorithms is to use the posterior distribution of the Bayesian model to explore the search space $\mathcal{X} \subset \mathbb{R}^d$ and select input values $\mathbf{x}$ that will most probably maximize the target function f in the objective space $\mathcal{Y}$.

Generally, in BO, the surrogates models are based on Gaussian Processes (GPs) because they are flexible in terms of kernel design and their ability to provide a principled and tractable quantification of uncertainty (Williams and Rasmussen, 2006). To start the process, an initial set of observations $\mathcal{D}_f = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$, which are assumed to be corrupted with additive Gaussian noise, $y_i = f(\mathbf{x}_i) + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ must be available. The initial set is used to train the surrogate GP model. Then, the AF determines what areas in the search space are worth exploiting and what areas are worth exploring based on the GP's current posterior distribution over f. Accordingly, areas where $f(\mathbf{x})$ is optimal or unexplored areas with the potential to improve the current best solution get a high AF value. By optimizing AF, a next point $\mathbf{x}_{n+1}$ is identified, and once sampled from the target function $f(\mathbf{x}_{n+1})$, it can be added to the history of observations $\mathcal{D}_f = \mathcal{D}_f \cup (\mathbf{x}_{n+1}, y_{n+1})$. The posterior distribution is updated each time a new data point is observed, and the whole process is repeated until the optimization budget is exhausted.

3.1.1. Gaussian processes

A GP is a collection of random variables, which, for some finite subsets, have a joint Gaussian distribution (Williams and Rasmussen, 2006). Therefore, a GP represents a distribution over functions $f(\cdot) \sim \mathcal{GP}(\mu(\cdot), k(\cdot,\cdot))$, parametrized for the mean $\mu(\cdot)$ and the kernel $k(\cdot,\cdot)$, which is defined for each pair of points $\mathbf{x}, \mathbf{x}' \in \mathbf{R}^d$ and represents the covariance between them. Thus:

$$\mu(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$$
$$k(\mathbf{x}, \mathbf{x}') = \mathbb{E}\left[(f(\mathbf{x}) - \mu(\mathbf{x}))(f(\mathbf{x}') - \mu(\mathbf{x}'))\right]$$

Typically, the hyperparameters of the kernel function are adjusted by maximizing the marginal likelihood over the training set, although full Bayesian approaches are also possible (Lalchand and Rasmussen, 2020).

Given a set of input samples $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ and their corresponding noisy output values $Y = \{y_1, \ldots, y_n\}$; the posterior probability of a new point $\hat{\mathbf{x}}$, can be estimated from the joint Gaussian distribution:

$$\begin{bmatrix} Y \\ \hat{y} \end{bmatrix} = \mathcal{N}\left(\begin{bmatrix} \mu(X) \\ \mu(\hat{\mathbf{x}}) \end{bmatrix}, \begin{bmatrix} k(X,X) + \sigma^2 I & k(X,\hat{\mathbf{x}}) \\ k(\hat{\mathbf{x}},X) & k(\hat{\mathbf{x}},\hat{\mathbf{x}}) \end{bmatrix}\right)$$

Then the posterior distribution of $f(\cdot)$ at the point $\hat{\mathbf{x}}$, which is denoted $f(\hat{\mathbf{x}}) = p(y|\hat{\mathbf{x}}, X, Y)$, can be estimated using the standard conditioning rules for Gaussian random variables, resulting in $f(\hat{\mathbf{x}}) \sim \mathcal{N}\left(\mu_f(\hat{\mathbf{x}}), \sigma_f(\hat{\mathbf{x}})\right)$, where

$$\mu_f(\hat{\mathbf{x}}) = \mu(\hat{\mathbf{x}}) + k(\hat{\mathbf{x}}, X)\left[k(X,X) + \sigma^2 I\right]^{-1}(Y - \mu(X))$$
$$\sigma_f(\hat{\mathbf{x}}) = k(\hat{\mathbf{x}}, \hat{\mathbf{x}}) - k(\hat{\mathbf{x}}, X)\left[k(X,X) + \sigma^2 I\right]^{-1} k(X, \hat{\mathbf{x}})$$

A more complete treatment of GPs can be found in Williams and Rasmussen (2006). In the context of this work, the effect of the initial set's size n, which corresponds to previously known diet formulations, will be studied during the experimental phase.

3.1.2. Acquisition functions

The second component to consider in BO is the selection of the AF. These functions utilize the posterior mean and variance at each point in the function to calculate a value that indicates the desirability of sampling at that position in the next iteration. An effective acquisition function should strike a balance between exploration and exploitation.

There are several AF proposed in the literature, among the most used are the upper confidence bound (UCB) (Srinivas et al., 2010), the Probability of improvement (PI) (Kushner, 1964), and the Expected Improvement (EI) (Močkus, 1975). UCB balances exploitation and exploration through the simple sum of the mean and variance of the posterior distribution, while PI selects the next query sample such that it maximizes the probability of obtaining an objective value greater than the current optimum. The way most used AF is EI, which not only considers the probability of improvement but the amount of improvement and evaluates f at the point that, in expectation, improves upon f the most. Formally, EI estimates:

$$\alpha_{\text{EI}}(\mathbf{x}) = (\mu_f(\mathbf{x}) - y^*)\Phi(\mathbf{z}) + \sigma_f(\mathbf{x})\phi(\mathbf{z})$$

where y^* is the incumbent, $\phi(\cdot)$ is the probability density function, $\Phi(\cdot)$ is the cumulative distribution function, $\mathbf{z} = \dfrac{\mu_f(\mathbf{x}) - y^* - \xi}{\sigma_f(\mathbf{x})}$, and ξ is a constant that balances exploration and exploitation.

The standard BO formulation provides a single new query point for every iteration; however, several approaches extending BO to provide multiple candidate solutions have been proposed to speed up

the process. In the case of EI, the most extended approach estimates the improvement over the joint probability distribution of q points and selects the set of points that maximize the multi-points expected improvement (Ginsbourger et al., 2010). This variant is called parallel EI (q-EI).

3.2. Multi-objective Bayesian optimization

Using EI, BO can be straightforwardly extended to vector-valued functions by defining and improving a performance metric over sets, which can be used to guide the search process across multiple objectives. However, this increases the problem's difficulty because there are many directions in which the objectives can be improved (Shu et al., 2020). As in any multi-objective approach, the ultimate goal of Multi-objective BO (MOBO) is to identify a collection of points that describes the best trade-offs among m different conflicting objectives, which is typically called the Pareto set $X_p = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_t\}$, $\mathbf{x}_i \in \mathcal{X}$, along with its corresponding Pareto front ($\mathcal{P}$), which is the set of solutions $\mathcal{P} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_t\}$ in the objective space, for $\mathbf{y}_j = [y_{j1}, y_{j2}, \dots, y_{jm}]^T = \mathbf{f}(\mathbf{x}_j) + \boldsymbol{\epsilon}$, such that $\nexists\ \mathbf{y}_l | \mathbf{y}_l \succ \mathbf{y}_j, l \neq j,\ \forall \mathbf{y}_l \in \mathcal{Y} \subset \mathbb{R}^m$ (Galuzio et al., 2020). The symbol $\succ$ represents dominance and, for the previous definition, would imply that objective values of $\mathbf{y}_l$ are no worse than those of $\mathbf{y}_j$, and objective values of $\mathbf{y}_l$ are strictly better than at least one of those of $\mathbf{y}_j$. In MOBO, $\mathcal{P}$ is approximated by a set P of all the non-dominated solutions in $\mathcal{D}_f$.

For the sake of simplicity, in this work, every objective function is modeled by independent GP priors. Each vector of solutions $\mathbf{y}_i$ corresponds to the noisy sampling of each of the three objective functions $\mathbf{y}_i = [y_e(\mathbf{x}_i), y_l(\mathbf{x}_i), y_c(\mathbf{x}_i)]$, where $y_e(\mathbf{x}_i) = f_e(\mathbf{x}_i) + \varepsilon_e$ and similarly for y_l and y_c.

3.2.1. Acquisition functions for MOBO

The most used AF in MOBO aims to estimate the expected improvement of the area under $\mathcal{P}$ given by a new point $\mathbf{x}$ and its corresponding posterior distribution, which is estimated based on the hypervolume (HV) indicator. HV was introduced in Zitzler and Thiele (1999) and stands as one of the fundamental unary indicators for assessing the quality of a Pareto front approximation set. Notably, this indicator possesses a distinct advantage because it does not necessitate prior knowledge of the Pareto front. Maximizing the HV can yield a Pareto front approximation set that is both highly qualified and diverse (Yang et al., 2019a). The function of the HV indicator is to measure the size of the subspace dominated by $\mathcal{P}$, which is bounded below by a reference point $\mathbf{r}$ and defined as

$$HV(\mathcal{P}) = \lambda_m \left(\cup_{\mathbf{y} \in \mathcal{P}} \left[\mathbf{r}, \mathbf{y} \right] \right)$$

where λ_m is the Lebesgue measure of a m-dimensional subspace bounded by $[\mathbf{r}, \mathbf{y}]$. As suggested in Yang et al. (2019a), the reference point can be selected so that it is dominated by all the elements in a Pareto front approximation set.

Accordingly, given an approximation of the Pareto front P, the improvement in HV due to the incorporation of a new vector of solutions $\mathbf{y}$ is given by $HVI(\mathbf{y}, P) = HV(P \cup \mathbf{y}) - HV(P)$. Therefore, the Expected hypervolume improvement (EHVI) extends the concept of EI to Multi-objective optimization settings by looking for the solution $\mathbf{y}$ that expands the volume of the subspace dominated by P the most (Yang et al., 2019a). Formally, EHVI is given by (Emmerich, 2005):

$$\alpha_{\text{EHVI}}(\mu, \sigma, P, r) = \int_{\mathbb{R}^m} HVI(\mathbf{y}, P)\, \zeta_{\mu,\sigma}(\mathbf{y})\, d\mathbf{y} \tag{2}$$

where $\zeta_{\mu,\sigma}(\cdot)$ is a multivariate independent normal distribution with the mean values $\mu \in \mathbb{R}^m$ and the standard deviations $\sigma \in \mathbb{R}^m_+$. EHVI is the most widely used AF in MOBO due to its advantages of high convergence and success in providing solutions close to the real Pareto front (Li and Yao, 2019). Moreover, similar to the single solution EI, EHVI supports parallel candidate generation (qEHVI) and gradient-based acquisition optimization (Daulton et al., 2021).

Still, the standard EHVI suffers from some limitations, including the assumption that observations are noise-free and the exponential scaling of its batch variant, qEHVI, which precludes large-batch optimization (Daulton et al., 2021). When there are noisy observations, there is a variant of EHVI called Noisy expected Hypervolume improvement (NEHVI), which integrates over the uncertainty in the function values at the observed points and can also be extended to parallel settings (qNEHVI) (Daulton et al., 2021). The estimation of Eq. (2) requires solving multiple multi-variate integrals, and even though there are several exact and approximate proposals in the literature, due to their complexity for large m, the most widely extended approach is to use numerical methods based on Monte Carlo (Yang et al., 2019b). Therefore, both qEHVI and qNEHVI are required to set up the number of Monte Carlo (MC) samples to estimate the AF correctly during its optimization process.

3.3. Optimality evaluation

According to Li and Yao (2019), an effective representation of the Pareto front is one that elucidates key properties of the problem, including its shape, dimensionality, scale, inflection points, and the interplay between objectives. This enables the decision-maker to gain a more comprehensive understanding of the problem. Therefore, the quality of a solution set is generally evaluated based on a sound representation of the Pareto front, assessed through four key aspects: convergence, indicating the proximity of the solution set to the Pareto front; spread, reflecting the coverage of the solution set; uniformity, gauging the evenness of the set's distribution; and cardinality, indicating the number of elements in the solution set.

While the HV indicator is linked to the cardinality and spread of the solution set, as sets with more non-dominated elements can inherently cover a larger space, the HV indicator alone does not ensure the uniform distribution of the solution set. Therefore, in conjunction with HV, the assessment of the Pareto front approximation provided by MOBO will incorporate spread and uniformity indicators in addition to the cardinality value. The specific definitions of these metrics are presented below.

3.3.1. Cardinality indicator

Given the set of observations $\mathcal{D}_f$ at an iteration k, Cardinality ($C_{\mathcal{D}_f}$) is derived from the number of elements within the approximate Pareto front set P containing the t non-dominated solutions in $\mathcal{D}_f$.

3.3.2. Spread indicator

The spread indicator measures the spatial coverage encapsulated by the approximation of the solution set. A set exhibiting significant dispersion should encompass solutions across every segment of P, striving to encompass each region comprehensively. In this study, we specifically focus on the Maximum Spread (MS) (Zitzler and Thiele, 1999), a widely used dispersion indicator that measures the range of a set of solutions considering the maximum range of each objective. MS is defined as

$$\text{MS}(P) = \sqrt{\sum_{j=1}^{m} \max_{\mathbf{y}, \mathbf{y}' \in P} (y_j - y'_j)^2}$$

The higher the MS is, the better P covers the Pareto front.

3.3.3. Uniformity indicator

Numerous indicators capable of measuring the variation in distances between elements of the solution set are available in the literature, as documented in Li and Yao (2019). Given its prevalence and recognition, in this study, the Spacing Schott indicator (SS) is used as a uniformity metric for evaluation purposes. SS is given by:

$$\mathrm{SS}(P) = \sqrt{\frac{1}{t-1}\sum_{i=1}^{t}\left(\overline{d} - d_1(\mathbf{y}_i, P_{/\mathbf{y}_i})\right)^2}$$

where $P_{/\mathbf{y}_i}$ is a set containing all the non-dominated solutions in P, but $\mathbf{y}_i$; $\overline{d}$ is the mean of all $d_1(\mathbf{y}_i, P_{/\mathbf{y}_i})$ for $i = [1, 2, \dots t]$ and $d_1(\mathbf{y}_i, P_{/\mathbf{y}_i})$ is estimated as the L_1 norm distance of $\mathbf{y}_i$ to set $P_{/\mathbf{y}_i}$, given by:

$$d_1(\mathbf{y}_i, P_{/\mathbf{y}_i}) = \min_{\mathbf{y}_u \in P_{/\mathbf{y}_i}} \sum_{j=1}^{m} |y_{ij} - y_{uj}|$$

In other words, $d_1(\mathbf{y}_i, P_{/\mathbf{y}_i})$ is the L_1 distance between $\mathbf{y}_i$ and the closest vector of solutions in $P_{/\mathbf{y}_i}$. The lower the value of SS, the better the uniformity. A SS value of zero indicates that all members of the solution set are spaced equidistantly based on the Manhattan distance (Li and Yao, 2019). In the literature, this indicator is widely used in conjunction with MS to describe the diversity quality of the set P.

4. Experiments and results

4.1. Experimental setup

The experiments are organized into three phases: (*I*) hyperparameter selection, (*II*) assessment of the solution's quality of MOBO in comparison to the best solution found in Peña et al. (2009), which is used as a benchmark, and (*III*) evaluation of the effects of the batch configuration of MOBO to speed up the optimization process. During the whole experimental phase, the results obtained using qEHVI and qNEHVI are evaluated in parallel to determine which AF provides the best results. For all the experiments, the GP models used an isotropic Matérn kernel, whose hyperparameters are estimated by maximizing the log-likelihood with respect to the data. Matérn kernel is a generalization of the exponentiated quadratic that adds an additional parameter controlling the smoothness of the resulting function, and it is preferred for high dimensional problems (Williams and Rasmussen, 2006).

The configuration of the whole optimization process involves the selection of three main hyperparameters: the number of MC samples to estimate the EHVI, the number of samples n used to train the initial GP and start the optimization process, and the number of iterations required for the model to converge. Therefore, during phase *I*, the number of MC samples is evaluated in the set $\{32, 64, 128, 256, 512, 1024, 2048\}$ in order to identify the minimum value of samples required to get stable estimations of the AFs. The grid search for selecting the number of initial samples is defined as $\{n = 10v, 1 \leq v \leq 12\}$. These samples are drawn randomly from the feasible space, ensuring each adheres to all problem constraints. This analysis aims to discern how varying levels of initial information impact the optimization process. Lastly, the number of iterations is evaluated in the range $[10, 50]$ with increments of 10. This is a critical hyperparameter since it is directly proportional to the objective function, a factor that should ideally be minimized in a nonsimulated context. The assessment of all the hyperparameters is centered on their influence on the HV, which is the primary metric optimized by the AFs used. Given the stochastic nature of the proposed method, each experiment is repeated 30 times for every hyperparameter value, utilizing different seeds. This approach allows for statistical analysis of the results, thereby facilitating the assessment of the stability of the optimization process.

In the case of MC samples, the primary objective is to establish the minimum sample quantity required for a consistent estimation of the AF, irrespective of its initialization.

Once the minimum values for the critical hyperparameters are set, the phase *II* focuses on evaluating the quality of the Pareto front approximation provided by MOBO. The evaluation is performed in terms of metrics previously defined, namely HV, MS, SS, and $C_{\mathcal{D}_f}$, with particular emphasis on their dependency on the number of iterations. The exploration/exploitation tradeoff of the MOBO process is also evaluated by estimating the Euclidean distance between consecutive solutions across all the iterations. This comparison is performed in the solution space (ingredients), after converting the diet formulation to its equivalent in nutrient content and in the objective space.

Considering that the methodology delineated in Peña et al. (2009) results in a singular solution (MFP) rather than a Pareto front approximation, which necessitates manual intervention from a decision-maker, the evaluation of MOBO solutions concerning MFP centers on determining the percentage of solutions within the Pareto set approximation that surpasses the objective values achieved by MFP.

To make the comparison between solutions coming from MOBO with respect to that of the MFP method, each objective was evaluated independently using the metric

$$d_{y_j} = \frac{y_j - \mathrm{MFP}_j}{max_j - min_j} \text{ with } j \in \{1, 2, \dots m\} \tag{3}$$

where y_j and MFP_j are the values obtained for jth objective by a MOBO and MFP solutions, respectively. Besides, max_j and min_j are the maximum and minimum values observed during simulations for the objective j. In other words, the difference between the two objective values is normalized with respect to the observed range of that objective in feasible solutions.

Lastly, phase *III* involves appraising the Pareto front approximation for values of $q > 1$. The purpose, in this case, is to gauge the extent to which a parallel BO strategy can expedite the optimization process in the context of diet design without compromising the quality of the solutions. In this set of experiments, the number of times the objective function is sampled remains constant while the number of iterations is adjusted accordingly. As emphasized earlier, sampling objective functions is a pivotal factor constraining the applicability of the MOBO approach in nonsimulated contexts. The maximum number of samplings to the objective functions is fixed according to the results obtained for $q = 1$ in the previous phase regarding the number of iterations required for the algorithm to converge.

4.2. Results

Fig. 2 shows the effect of the number of MC samples in the estimation of HV during one iteration of the BO process applied to the problem defined by Eq. (1). We present results for the two AF described in Section 3.2.1, qEHVI and qNEHVI, both for $q = 1$. From Fig. 2, it is possible to observe that the HV estimation provided by qNEHVI is more sensitive to the number of MC samples than qEHVI. qNEHVI is also computationally more expensive than qEHVI, with estimation time increasing exponentially with respect to the number of MC samples. For 256 MC samples, qNEHVI consumes 10x more time than qEHVI, and this value increases to more than 30x for 2048 samples. According to the results, for the problem addressed in this work, HV estimation for qEHVI stabilizes for MC samples greater than 64, while qNEHVI requires 256 MC samples to achieve a similar behavior. For the sake of comparison, all the following experiments will use 256 MC samples for both AFs.

Fig. 3 shows HV evolution during the MOBO process for 10 (column A), 50 (column B), and 90 (column C) initial samples and during 50 iterations. As pointed out before, every experiment was run 30 times using a different set of initial samples drawn randomly but guaranteeing that all of them satisfied the set of problem restrictions; thus, plots in the last row show summary statistics of the performance displayed during the experiment. From Fig. 3, it is possible to observe that, almost in all repetitions during the first ten iterations, the MOBO

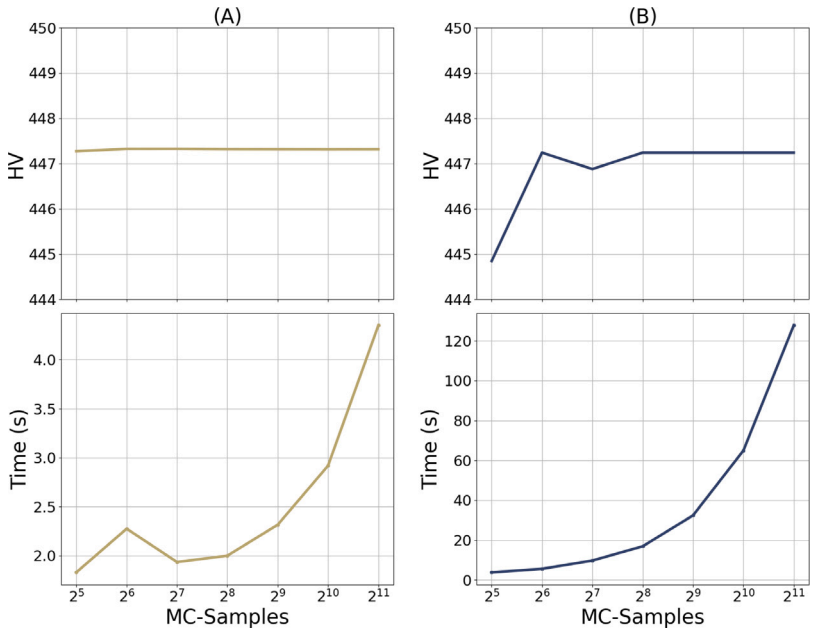

Fig. 2. Effect of the number of MC samples in the estimation of HV. Graphs in column (A) show the HV values (above) and the time taken to estimate it (below) when qEHVI is used. Graphs in column (B) show the HV value and consumed time, in the same order, for qNEHVI.

process is able to improve the Pareto Front approximation strongly and stabilizes after 30. However, such stabilization happens before as the number of initial samples increases. In general terms, there is an inverse relationship between the number of initial samples and the number of iterations required to get high values of HV, and that behavior is more evident for qNEHVI, which means that, as expected, qNEHVI favors exploration more than qEHVI. For an initial set with 90 samples, results in Fig. 3 show that is less likely for the MOBO process to get stuck in regions where the solution set does not change, as for 10 and 50.

Interestingly, there is also a direct relationship between the number of initial samples with which the optimization process is started and the final value of HV achieved. According to the results, for an initial sample of size 10, even though the MOBO process is able to improve the Pareto Front approximation substantially, the algorithm converges to a local optimum of the HV. However, as the number of initial samples increases, the effect of a poor initial sample set can be compensated by a large number of iterations. This trend is confirmed in the bar diagrams of Fig. 4, where the dependency of final HV on the number of initial samples is analyzed in more detail. From Figs. 3 and 4, it is also possible to observe that, on average, qNEHVI achieves solution sets with better HV than those selected by qEHVI, which can be explained due to better exploration traits of the first one.

The variability of the Pareto front HV during multiple repetitions and for different initial samples can be observed in Fig. 5. The boxplots show the inter-quartile ranges of HV values obtained for both AFs during 50 iterations. Concerning the number of initial samples, it is noteworthy that boxplots representing runs with fewer than 50 initial samples exhibit larger dispersion. Conversely, boxplots with initial samples equal to or greater than 50 showcase narrower boxes, and their medians align consistently. This suggests that the HV distribution tends to converge toward similar values, which allows us to consider 50 as a feasible lower bound for the number of initial samples, although larger values are always desirable because that would imply more informed surrogate models and better uncertainty estimations that, in turn, will promote better choices of exploratory regions in the search space.

Regarding the differences between the two AFs evaluated, except for the experiment with ten initial samples, on average, qNEHVI achieved higher HV values than qEHVI. Furthermore, the distribution of HV values for qNEHVI shows less dispersion and greater concentration around the median value than those obtained with qEHVI. As pointed out before, qNEHVI promotes exploration more than qEHVI, which means that the higher computational cost is rewarded in achieving better local optima.

According to the previous results, to evaluate the quality of the Pareto front approximation and compare the solutions with that proposed in Peña et al. (2009), the number of initial samples was set to 50. Even though better results can be obtained for larger values, this is the minimum value from which MOBO shows stabler results regarding HV convergence, so it was defined as a minimum requirement to move forward and analyze the quality of the solutions in the Pareto front approximation. In this sense, Fig. 6 shows the evolution of the four quality indicators defined in Section 3.3 across 50 iterations of a MOBO process that uses qNEHVI and 50 initial samples. Fig. 6(A) shows HV vs. C_{D_f}, while Fig. 6(B) shows MS vs. SS. It is possible to observe that the C_{D_f} grows monotonically throughout the entire experiment, even though the HV value practically stops growing after 30 iterations. Similar to HV, MS and SS practically stop improving and stabilize after 40 iterations. These results suggest that the ability to obtain more solutions across all iterations enhances the diversity of non-dominated solutions, but after the first 20 iterations, such diversity does not result in a significant improvement of the quality indicators. Nevertheless, the SS indicator shows an enhancement in the uniform distribution along the approximation of the Pareto front, enabling a more balanced representation of the objective space. Besides, it is important to highlight that from a decision-maker standpoint, the more elements the solution set has, the more possible solutions can be analyzed by the expert to choose the best solution to be implemented.

The evolution of the quality metrics in Fig. 5 also suggests that the MOBO process mainly explores during the first 20 iterations and then focuses on the exploitation around the best-known solution. To confirm this observation, Fig. 7 shows the Euclidian difference between consecutive solutions throughout the optimization process. Since the solution to a diet design problem can be described in three different spaces: input (ingredients), restrictions (nutrients), and outputs (objectives), the figure shows the same analysis carried out in each of these three spaces. According to the results, the most pronounced exploration ends after the first ten iterations, but fluctuations in standard deviations suggest the presence of moments of greater and lesser variability in the exploration across all the iterations. It could indicate that the MOBO process predominantly explores a relatively confined

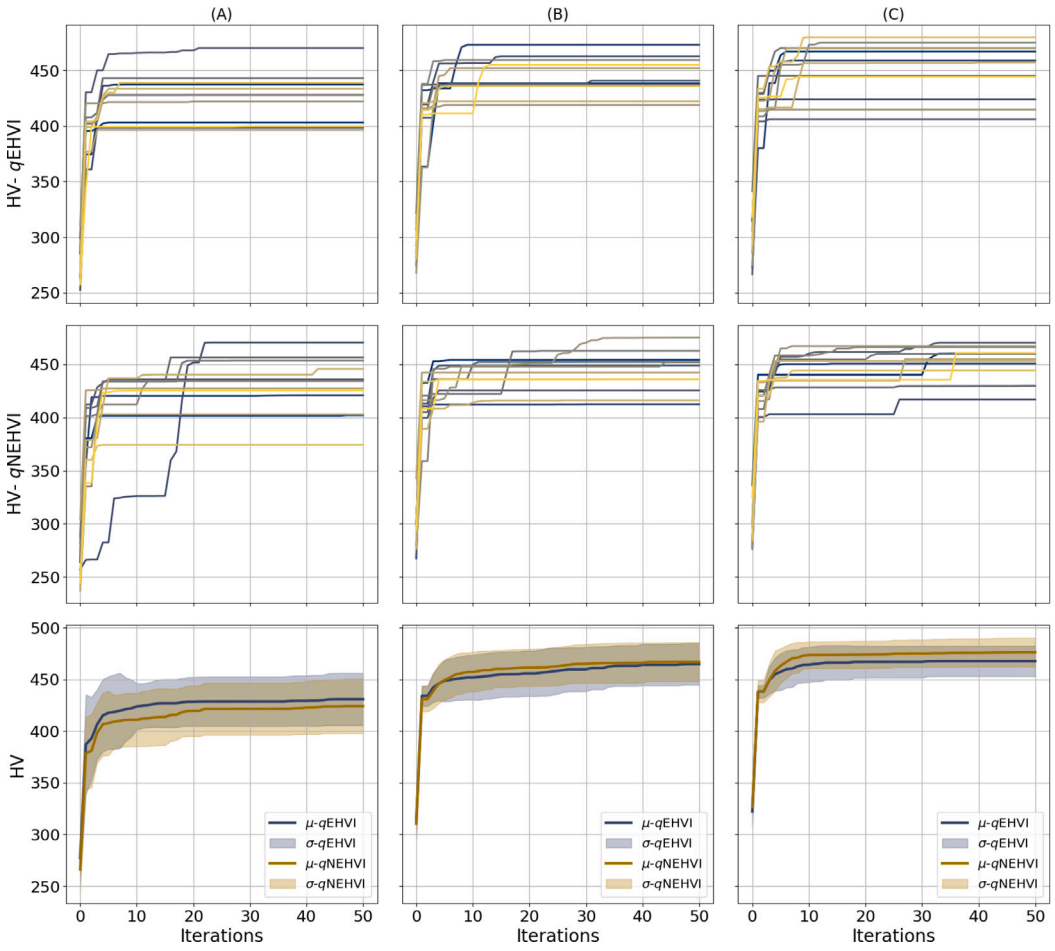

Fig. 3. Progression of HV over iterations in multiple repetitions of the MOBO process. In Column (A), HV values are depicted when employing ten initial samples for both qEHVI (upper row) and qNEHVI (middle row) AFs. Column (B) displays HV evolution with 50 initial samples for both AFs, while Column (C) presents HV evolution with 90 initial samples for both AFs. The lower row provides the average HV and standard deviation of the runs corresponding to each AF in the respective column.

region within the broader search space due to the effectiveness of the optimization method in quickly identifying a promising region of the search space and focusing its efforts there. In contrast, the observation is different regarding the spaces of nutrients and ingredients. These spaces are relatively narrow, and the optimization process appears to explore a more comprehensive range within these spaces. This could be attributed to the restricted nature of these domains, where a smaller region is explored due to their inherent constraints.

These observations are supported by the distance values shown in Fig. 5, which suggest that the region covered by the exploration process is larger in the objective space than in the ingredients and nutrients spaces. Especially considering that for this work's particular case, the objective space has dimension 3, while ingredients and nutrients have dimensions 17 and 10, respectively. Interestingly, it is also possible to observe that small changes in the objective space result in bigger changes in the ingredients and nutrient spaces since variability in those spaces is more evident across all the iterations.

After the characterization of the optimization process and its corresponding evolution of the Pareto front approximation, an important outcome is that MOBO was able to automatically find a better solution to the diet design problem than that of the MFP method proposed in Peña et al. (2009). Table 1 shows the objective values obtained by the best solutions found by MOBO and MFP. As it is possible to observe, MOBO was able to find three non-dominated solutions that, in turn, dominate the solution provided by MFP. The corresponding formulation in terms of ingredients and nutrient content of each of the solutions found by MOBO can be consulted in Appendix.

Table 1
Objective values of the best solution found by MFP and the three best non-dominated solutions found by the MOBO process.

Method	Cost	Lysine	Energy
MFP (Peña et al., 2009)	151.4	1.02	14.31
MOBO	150.35	1.15	14.35
	150.58	1.11	14.36
	150.53	1.06	14.51

It is important to note that MOBO identified some of the best solutions within the initial ten iterations of the optimization process, indicating its ability to swiftly pinpoint a promising region in the search space and concentrate efforts on exploitation. For a more systematic comparison of solutions within the Pareto front approximation generated by MOBO, Table 2 illustrates how these solutions compare to the MFP solution from Peña et al. (2009). The comparison is conducted across various iterations to assess whether an increase in iterations leads to, on average, a greater number of superior solutions. To achieve this, the percentage of non-dominated solutions outperforming MFP is calculated, considering simultaneously one, two, or three objectives. Consequently, Table 2 presents the percentage of non-dominated solutions for each objective: cost (C), Lysine (L), energy (E), pairs of objectives: Cost-Lysine (CL), Cost-Energy (CE), Lysine-Energy (LE), and finally, all three objectives combined, Cost-Lysine-Energy (CLE).

From Table 2, it is possible to observe that, after ten iterations, 93% of the solutions in the Pareto front approximation are non-dominated regarding that of the MFP method. Interestingly, after that same number of iterations, more than 66% of solutions outperform the Lysine

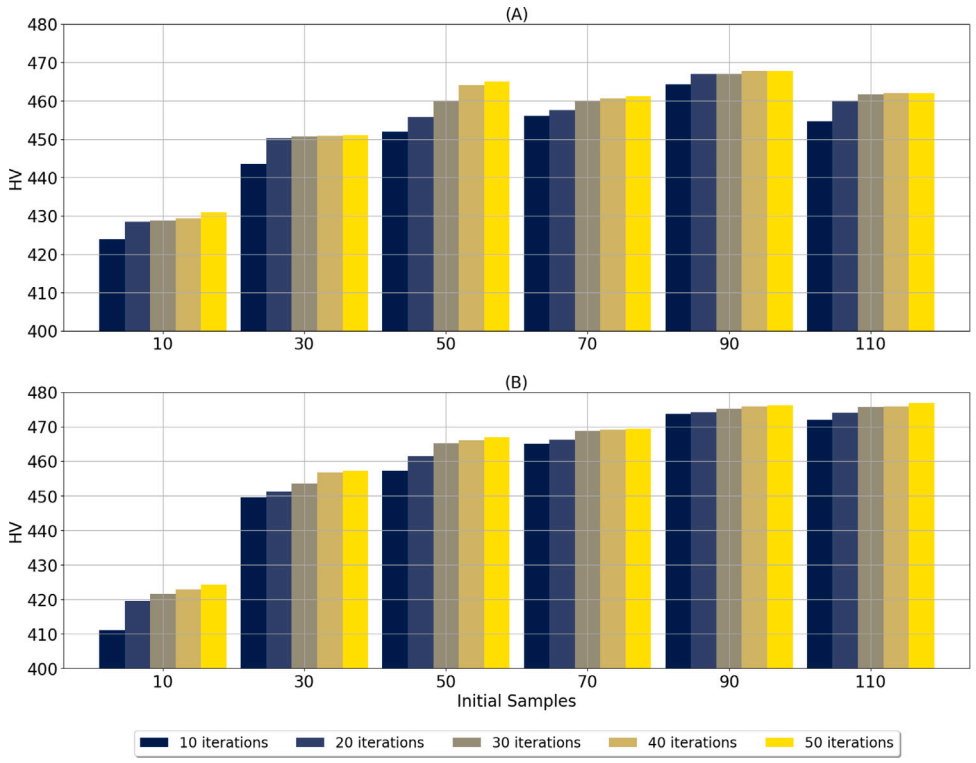

Fig. 4. Average HV values for MOBO using 10, 20, 30, 40, 50 iterations and 10, 30, 50, 70, 90, 110 initial samples. Figure (A) shows results for qEHVI and Figure (B) for qNEHVI.

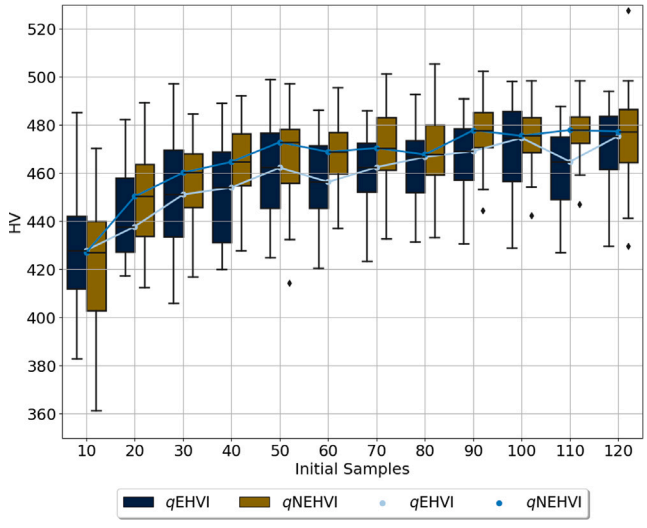

Fig. 5. Boxplot of HV obtained for different amounts of initial samples during 50 iterations. Figure shows results for the two AFs evaluated.

Table 2
Percentages of MOBO solutions improving 0, 1, 2, or 3 objectives with respect to MFP (Peña et al., 2009).

MOBO			MFP							
k	HV	C_{D_f}	Ø (%)	C (%)	L (%)	E (%)	CL (%)	CE (%)	LE (%)	CLE (%)
10	457.34	21.3	0.7	29.2	74.4	31.2	5.1	18.7	11.9	0.2
20	461.56	28.4	0.6	36.6	68.4	37.2	7.3	19.8	16.0	0.4
30	465.28	34.6	0.5	41.4	67.1	38.9	11.2	19.9	17.3	0.4
40	466.14	40.2	0.5	44.0	67.4	38.5	14.2	19.3	17.2	0.4
50	466.93	44.7	0.4	45.4	67.7	38.0	16.0	18.9	16.5	0.4

value achieved by the MFP solution. Besides, applying set theory, it is easy to estimate that after 20 iterations, on average, the percentage of solutions achieving better values in two out of three objectives exceeds 40% (CL+CE+LE-2CLE). In other words, the MOBO process can quickly provide an important percentage of solutions that achieve a strong trade-off between at least two objectives.

Based on Eq. (3), Table 3 shows the amount of improvement, per objective, achieved by those solutions outperforming that of the MFP method. Once again, only solutions in the Pareto front approximation are considered for this analysis. The results are shown in terms of percentage since the metric d_y is normalized with respect to the observed dynamic range of each objective. The analysis is performed for the same objective combinations presented in Table 2 and across 50 iterations. Positive values in Table 3 indicate that MOBO solutions achieve, on average, an improvement regarding the MFP solution. Negative values, instead, represent the cases where MFP achieved better results. Thus, as the MOBO solutions are split into non-disjoint subsets depending on whether they overcome one or several objectives regarding MFP, from Table 3, it is possible to observe the gains and losses of each particular combination of objectives and iterations. e.g., if only the MOBO solutions that overcome MFP for Lysine and Energy are considered, it is possible to observe that for 20 iterations, the gain in Lysine and Energy achieve 21.16% and 31.02%, respectively, while the loss in Cost is of 10.21%, which is considerably less than the benefit obtained summing up the other two objectives.

From Tables 2 and 3, it is possible to observe to what extent the quality of the Pareto front approximation improves as the number of iterations increases. The gain of the solutions outperforming all the objectives (CLE) is more concentrated on the Energy for the first ten interactions than for 20, where a better balance among the three objectives is achieved. Interestingly, if the set of solutions improving Energy is considered, the other two objectives are also improved. That behavior is consistent across all the iterations. Considering MOBO solutions where two out of the three objectives are improved in comparison to MFP, CE, and LE present gains that compensate for the loss in the

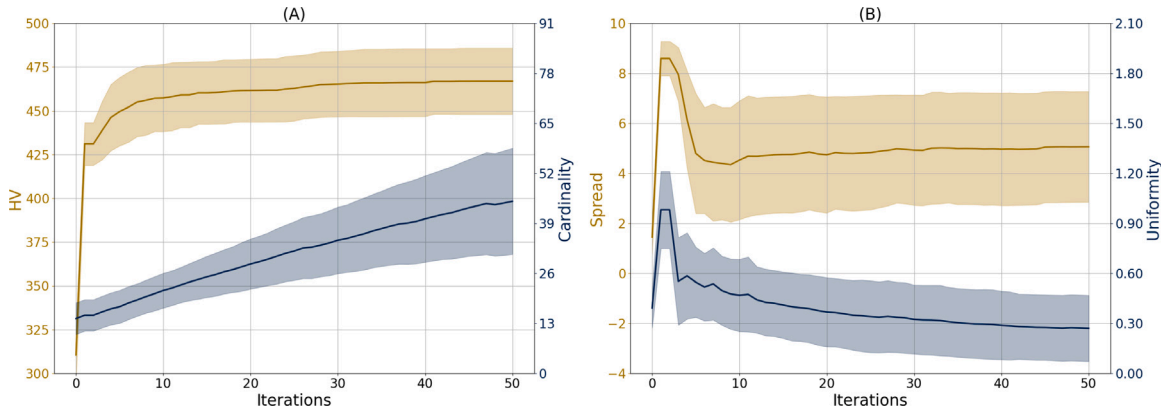

Fig. 6. Figure (A) shows the evolution of HV vs. C_{D_f} across iterations. Figure (B) shows the MS vs. SS across iterations. Solid lines correspond to mean values and shadow regions to the standard deviation. All the experiments used 50 initial samples and qNEHVI as AF.

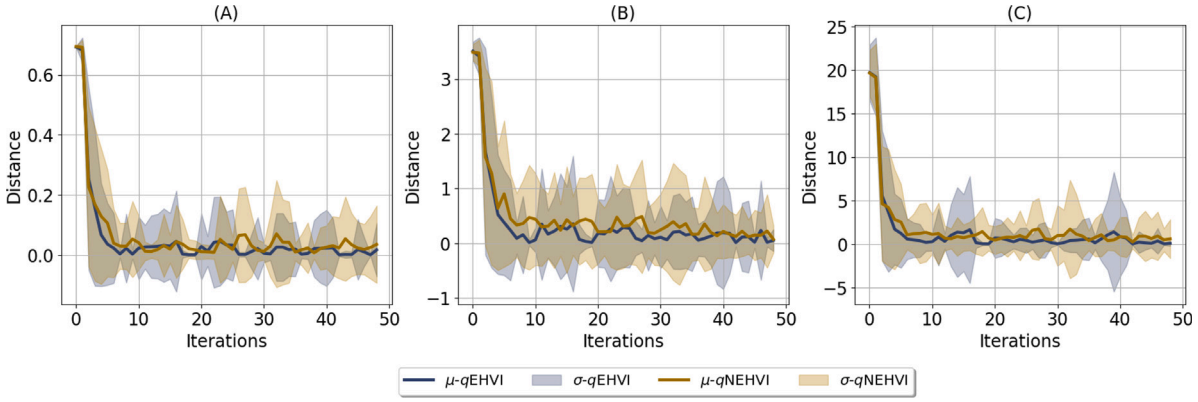

Fig. 7. Figure (A) shows the average distances between consecutive solutions in the ingredient vector space. Figure (B) shows the average distances between consecutive solutions estimated in the nutrient vector space, and plot (C) shows the average distances between consecutive solutions in the objective space. The distances are estimated across 50 iterations for all figures and for the two AFs evaluated previously.

Table 3
Average percentage of improvement of MOBO solutions in comparison to MFP (Peña et al., 2009).

Objective	10			20			30			40			50		
	Cost (%)	Lys (%)	Ene (%)	Cost (%)	Lys (%)	Ene (%)	Cost (%)	Lys (%)	Ene (%)	Cost (%)	Lys (%)	Ene (%)	Cost (%)	Lys (%)	Ene (%)
C	17.75	−5.01	−0.63	17.00	−4.45	−4.87	15.53	−2.28	−8.28	14.07	0.00	−11.2	13.15	1.30	−12.61
L	−22.4	37.93	−11.92	−18.67	35.12	−7.59	−15.77	32.91	−7.16	−13.81	31.73	−8.59	−12.74	31.31	−10.44
E	3.30	3.98	21.56	2.91	3.88	24.79	2.72	3.44	26.12	2.53	2.95	26.85	2.71	2.42	26.20
CL	5.44	21.89	−36.87	5.65	21.34	−36.83	6.14	20.77	−37.96	5.34	21.63	−38.77	4.88	22.23	−39.11
CE	17.20	−10.46	18.25	15.26	−10.18	18.59	14.07	−10.13	18.82	13.36	−10.19	18.96	12.85	−10.24	19.07
LE	−15.45	24.99	26.01	−10.21	21.16	31.02	−8.93	20.07	33.05	−8.57	19.45	34.50	−8.27	19.56	33.37
CLE	3.58	6.06	25.97	3.65	13.13	12.55	3.79	14.77	10.71	3.79	14.77	10.71	3.79	14.77	10.71

third objective (in case all objectives are considered equally important), while CL shows a relevant percentage drop in Energy.

Lastly, Fig. 8 and Table 4 show the quality indicators obtained for $q = 2$ and $q = 3$, in comparison to the sequential MOBO process ($q = 1$). In order to make the comparison fair, the analysis focuses on the evolution of the HV under the condition of an equal number of new query samples. Since for $q = 2$, the MOBO process obtains two new samples from the objective functions instead of only one, it requires 25 iterations to get the same number of new samples in the solution set as when $q = 1$ is used. Take into account that the increment in the number of parallel samples comes at the cost of sampling the objective functions twice per iteration. A similar analysis can be made for $q = 3$ to conclude that, in this case, 17 iterations are required to achieve 50 new query samples.

Fig. 8 presents HV's mean and standard deviations across iterations for 30 runs. As it is possible to observe, the parallel process is able to achieve very competitive values of HV, although the performance drops as q increases. The values of C_{D_f} and SS in Table 4 confirm that behavior. In particular, C_{D_f} shows that the Pareto front approximation contains a smaller number of solutions, which has a negative impact on SS. However, similar to 6, there is not a direct relationship between C_{D_f} and MS, which means that a large number of solutions in the Pareto front approximation does not imply a better covering of the Pareto front. Notably, for $q > 1$, MOBO was also able to find non-dominated solutions in comparison to MFP, even during the first ten iterations, which means that a successful acceleration would be possible.

5. Conclusions

This work evaluates the use of BO as an alternative to the diet design in the context of animal production. A multi-objective optimization problem including Lysine, Digestible Energy, and Cost was addressed, aiming to compare the capacity of BO to provide competitive solutions regarding stochastic programming methods able to deal with the intrinsic variability of the nutrient content in raw materials.

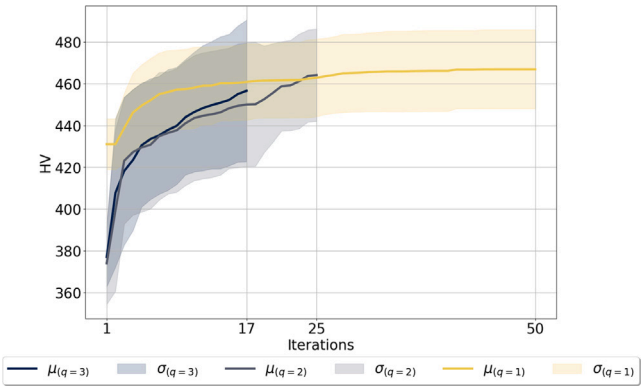

Fig. 8. Means and standard deviations of HV values for $q = 1, 2, 3$.

Table 4
Quality indicators for $q = 1, 2, 3$. k corresponds to the number of iterations required to achieve 50 candidate solutions per each q value.

	$q = 1$	$q = 2$	$q = 3$
Indicator	k		
	50	25	17
HV	466.92	444.77	434.78
C_{D_j}	44.73	34.23	35.83
MS	5.06	4.54	5.83
SS	0.27	0.34	0.42

The MOBO process implemented was able to provide better solutions than that achieved by a previously proposed method (Peña et al., 2009), overcoming it in all the three objectives analyzed. For 50 initial samples, some solutions found by MOBO yield improvements of 10.71% for energy, 14.77% for Lysine, and 3.79% for Costs. Better results could even be obtained if more initial samples are considered. Moreover, since BO is a data-driven strategy, the proposed methodology could easily incorporate additional zootechnical and environmental variables, which can influence the performance of animals in a real context, while methods depending entirely on analytical formulations of the objective functions are limited to variables strictly defined by such formulations. This makes MOBO a more adaptable methodology to changes in the environmental conditions of production farms or to incorporate zootechnical decisions due to particular conditions of a pharm, such as the number of animals in a herd. Moreover, unlike MFP, the MOBO process is automatic and does not require the manual intervention of a decision-maker.

Regarding the set of hyperparameters, the experiments show that a number of 256 MC samples are enough to estimate any of the two evaluated AFs reliably based on the improvement of HV. Moreover, qNEHVI yields consistently better results than qEHVI, mainly because qNEHVI promotes more exploration than its standard version.

For values of initial samples greater than 50, the MOBO process started to show reduced dispersion and more stability in the HV values. In the context of animal production, such a number of initial samples does not represent a big challenge, considering that typical production pharms rise dozens of swine herds simultaneously every month, and historical data is perfectly valid to initialize surrogate GP models.

Predictably, the quality of the Pareto front approximation increases as more interactions are considered, but remarkably, the MOBO process was able to find better solutions than that of the MFP method during the first ten iterations. Moreover, by analyzing the solutions where at least two objectives overcome that of the MFP, it was found that CE and LE provide a good balance regarding the third objective, so a decision-maker could take them as a reference to pick one single solution. In other words, the solutions demonstrating enhancements in Lysine and Energy are more likely also to exhibit favorable impacts on costs. Furthermore, the utilization of parallel candidate generation in the AF (i.e., using $q > 1$) enables the user to accelerate the optimization process without significantly compromising the quality of solutions in the Pareto front approximation.

An important drawback of the AFs based on HV is the computational complexity of its estimation. In fact, estimating AFs for more than three objectives is an open problem, so the incorporation of additional objectives, such as the reduction of greenhouse gases, would require the analysis of alternative AFs, such as the one proposed in Tu et al. (2022). Besides, due to the observed limitations in the exploration characteristics of the HVI-based AFs, strategies based on the partition of the search spaces are also worth exploring (Daulton et al., 2022). In order to implement real solutions to the swine diet design problem based on MOBO, additional issues must be considered, such as the incorporation of contextual (non-controllable) variables (Krause and Ong, 2011) and dealing with data coming from different farms and fidelity levels (Belakaria et al., 2021).

Diet design using optimization methods is a common problem that has been addressed for different livestock (Nyhodo et al., 2014; Sebastian et al., 2008; Ghosh et al., 2014). Certainly, they all face similar restrictions in terms of the high cost of target black-box function evaluation, guaranteeing minimum nutritional content, raw material availability, nutritional content variability in raw materials, and the requirement of adaptability to climate change effects such as overheating, etc. Therefore, the methodology proposed in this work could also be applicable to other animals raised in agricultural settings.

CRediT authorship contribution statement

Gabriel D. Uribe-Guerra: Writing – review & editing, Visualization, Validation, Software, Investigation, Formal analysis, Data curation, Conceptualization. **Danny A. Múnera-Ramírez:** Writing – review & editing, Supervision, Methodology, Formal analysis, Conceptualization. **Julián D. Arias-Londoño:** Writing – review & editing, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Gabriel D. Uribe-Guerra reports financial support was provided by Colombia Ministry of Science Technology and Innovation. Julian D. Arias-Londono reports financial support was provided by Polytechnic University of Madrid. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The code necessary to reproduce the experimental findings can be found at https://github.com/GabrielUribe29/MOBO-feed.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the author(s) used ChatGPT version 3.5 to improve the manuscript's readability. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Acknowledgments

G.D. Uribe-Guerra is supported by Colombia's Ministry of Science, Technology, and Innovation through Bicentennial Doctoral Excellence Scholarship Program - Court 2, 2020.

J.D. Arias-Londoño started this work at the Antioquia University and finished it supported by a María Zambrano grant from the Universidad Politécnica de Madrid, Spain.

Table A.5
Corresponding values of ingredients (in %) for the best MFP and MOBO solutions.

Ingredients	MFP	MOBO		
	[151.4, 1.02, 14.31]	[150.4, 1.15, 14.35]	[150.6, 1.11, 14.36]	[150.53, 1.06, 14.51]
Barley	13.53	40.00	40.00	0.00
Wheat	22.25	0.00	0.00	40.00
Corn	0.00	0.00	0.00	0.00
Alfalfa	0.00	0.00	0.00	0.00
Cassava Meal	0.00	0.00	0.00	0.00
Soybean meal	15.08	9.09	28.74	27.99
Fish meal	0.00	0.11	0.00	0.00
Gluten feed	3.74	8.00	8.00	8.00
Calcium Carbonate	0.94	1.25	3.26	4.01
Lysine 78%	0.00	0.39	0.00	0.00
Sunflower meal	0.00	3.00	0.00	0.00
Animal fat	0.00	0.00	0.00	0.00
Beet pulp	0.00	0.00	0.00	0.00
Lupin	10.00	10.00	0.00	0.00
Peas	14.06	8.15	0.00	0.00
Rye	20.00	20.00	20.00	20.00
Dicalcium	0.4	0.00	0.00	0.00

Table A.6
Nutrient content of the best MFP and MOBO solutions.

	MFP	MOBO		
Cost (€/MT)	151.4	150.35	150.58	150.53
Lysine (%)	1.02	1.15	1.11	1.06
Energy (MJ/kg)	14.31	14.35	14.36	14.51
Crude Fibre (%)	5.09	6.00	4.51	3.79
Calcium (%)	0.60	0.60	1.38	1.66
Dry matter (%)	89.15	89.58	89.53	89.29
Crude protein (%)	19.33	18.00	20.47	20.26
Phosphorus (%)	0.48	0.43	0.45	0.44
Methionine+cystine (%)	0.6	0.58	0.68	0.68
Tryptophan (%)	0.21	0.19	0.25	0.25
Threonine (%)	0.7	0.63	0.77	0.75
Available phosphorus (%)	0.16	0.15	0.15	0.17

This work was partially funded by the Autonomous Community of Madrid through the ELLIS Unit Madrid.

Appendix. Ingredients and nutrients

Tables A.5 and A.6 list the distribution of ingredients and their equivalent nutritional content, respectively, corresponding to the best solutions found by MOBO and the reference solution MFP proposed in Peña et al. (2009).

References

Altun, A.A., Şahman, M.A., 2013. Cost optimization of mixed feeds with the particle swarm optimization method. Neural Comput. Appl. 22, 383–390.

Amit, S.K., Uddin, M.M., Rahman, R., Islam, S., Khan, M.S., 2017. A review on mechanisms and commercial aspects of food preservation and processing. Agric. Food Secur. 6 (1), 1–22.

Babić, Z., Perić, T., 2011. Optimization of livestock feed blend by use of goal programming. Int. J. Prod. Econom. 130 (2), 218–223.

Belakaria, S., Deshwal, A., Doppa, J.R., 2021. Output space entropy search framework for multi-objective Bayesian optimization. J. Artif. Intell. Res. 72, 667–715.

Chappell, A., 1974. Linear programming cuts costs in production of animal feeds. J. Oper. Res. Soc. 25, 19–26.

Cho, S., Han, I.K., Kim, Y., Park, S., Hwang, O., Choi, C., Yang, S., Park, K., Choi, D., Yoo, Y., 2012. Effect of lysine to digestible energy ratio on growth performance and carcass characteristics in finishing pigs. Asian-Aust. J. Anim. Sci. 25 (11), 1582.

D'Alfonso, T.H., Roush, W.B., Ventura, J.A., 1992. Least cost poultry rations with nutrient variability: a comparison of linear programming with a margin of safety and stochastic programming models. Poult. Sci. 71 (2), 255–262.

Daulton, S., Balandat, M., Bakshy, E., 2021. Parallel bayesian optimization of multiple noisy objectives with expected hypervolume improvement. Adv. Neural Inf. Process. Syst. 34, 2187–2200.

Daulton, S., Eriksson, D., Balandat, M., Bakshy, E., 2022. Multi-objective bayesian optimization over high-dimensional search spaces. In: Uncertainty in Artificial Intelligence. PMLR, pp. 507–517.

Emmerich, M., 2005. Single-and Multi-Objective Evolutionary Design Optimization Assisted by Gaussian Random Field Metamodels (Ph.D. thesis). Dortmund, Univ., Diss., 2005.

Frazier, P.I., 2018a. Bayesian optimization. In: Recent Advances in Optimization and Modeling of Contemporary Problems. Informs, pp. 255–278.

Frazier, P.I., 2018b. A tutorial on Bayesian optimization. arXiv preprint arXiv:1807.02811.

Galuzio, P.P., de Vasconcelos Segundo, E.H., dos Santos Coelho, L., Mariani, V.C., 2020. MOBOpt—multi-objective Bayesian optimization. SoftwareX 12, 100520.

Garnett, R., 2023. Bayesian Optimization. Cambridge University Press.

Ghosh, S., Ghosh, J., Pal, D., Gupta, R., 2014. Current concepts of feed formulation for livestock using mathematical modeling. Anim. Nutr. Feed Technol. 14 (1), 205–223.

Ginsbourger, D., Le Riche, R., Carraro, L., 2010. Kriging is well-suited to parallelize optimization. In: Computational Intelligence in Expensive Optimization Problems. Springer, pp. 131–162.

Glen, J.J., 1980. A mathematical programming approach to beef feedlot optimization. Manage. Sci. 26 (5), 524–535.

Innocent, M., Guillemot, S., Gabriel, P., Tamaro, A., 2023. Speeding up the transition to a more sustainable food system: New insights into the links in a system of practices. Rec. Appl. Mark. (Engl. Ed.) 38 (3), 77–109.

Krause, A., Ong, C., 2011. Contextual Gaussian process bandit optimization. Adv. Neural Inf. Process. Syst. 24.

Kushner, H.J., 1964. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. J. Basic Eng..

Lalchand, V., Rasmussen, C.E., 2020. Approximate inference for fully Bayesian Gaussian process regression. In: Symposium on Advances in Approximate Bayesian Inference. PMLR, pp. 1–12.

Li, M., Yao, X., 2019. Quality evaluation of solution sets in multiobjective optimisation: A survey. ACM Comput. Surv. 52 (2), 1–38.

Lisitsyn, A., Chernukha, I., Nikitina, M., 2023. Development of a personalized diet using the structural optimization method. Food Syst. 6 (1), 64–71.

Močkus, J., 1975. On Bayesian methods for seeking the extremum. In: Optimization Techniques IFIP Technical Conference: Novosibirsk, July 1–7, 1974. Springer, pp. 400–404.

Moon, T., Sim, S., Son, J.E., 2023. Calibration of food and feed crop models for sweet peppers with Bayesian optimization. Horticult. Environ. Biotechnol. 1–11.

Naharro, P.S., Toharia, P., LaTorre, A., Peña, J.-M., 2022. Comparative study of regression vs pairwise models for surrogate-based heuristic optimisation. Swarm Evol. Comput. 75, 101176.

Nikitin, A., Fastovets, I., Shadrin, D., Pukalchik, M., Oseledets, I., 2019. Bayesian optimization for seed germination. Plant Methods 15 (1), 1–10.

Noblet, J., Van Milgen, J., 2004. Energy value of pig feeds: Effect of pig body weight and energy evaluation system. J. Anim. Sci. 82 (suppl_13), E229–E238.

Nyhodo, B., Mmbengwa, V., Balarane, A., Ngetu, X., 2014. Formulating the least cost feeding strategy of a custom feeding programme: A linear programming approach. OIDA Int. J. Sustain. Dev. 7 (10), 85–92.

Patil, V., Gupta, R., Rajendran, D., Kuntal, R.S., Chanda, M., 2022. Stochastic programming model in least cost feed formulation for lactating cattle. Indones. J. Agric. Res. 5 (3), 231–248.

Peña, T., Lara, P., Castrodeza, C., 2009. Multiobjective stochastic programming for feed formulation. J. Oper. Res. Soc. 60 (12), 1738–1748.

Pesti, G.M., Miller, B.R., 1993. Animal Feed Formulation: Economic and Computer Applications. Springer Science & Business Media.

Pomar, C., Andretta, I., Remus, A., 2021. Feeding strategies to reduce nutrient losses and improve the sustainability of growing pigs. Front. Vet. Sci. 8, 742220.
Pratiksha, S., et al., 2011. Application of nonlinear programming for optimization of nutrient requirements for maximum weight gain in buffaloes. Int. J. Food Sci. Nutr. Eng. 1 (1), 8–10.
Şahman, M.A., Çunkaş, M., İnal, Ş., İnal, F., Coşkun, B., Taşkiran, U., 2009. Cost optimization of feed mixes by genetic algorithms. Adv. Eng. Softw. 40 (10), 965–974.
Sebastian, C., Festus, K.A., Oluyede, C.A., Gudeta, S., Simon, M., France, M.G., 2008. A simple method of formulating least-cost diets for smallholder dairy production in sub-Saharan Africa. Afr. J. Biotechnol. 7 (16), 2925–2933.
Sharma, A., Jain, A., Gupta, P., Chowdary, V., 2020. Machine learning applications for precision agriculture: A comprehensive review. IEEE Access 9, 4843–4873.
Shu, L., Jiang, P., Shao, X., Wang, Y., 2020. A new multi-objective Bayesian optimization formulation with the acquisition function for convergence and diversity. J. Mech. Des. 142 (9), 091703.
Spronk, J., Spronk, J., 1981. Interactive Multiple Goal Programming. Springer.
Srinivas, N., Krause, A., Kakade, S.M., Seeger, M., 2010. Gaussian process optimization in the bandit setting: No regret and experimental design. In: Proceedings of the 27th International Conference on Machine Learning - ICML. Haifa, Israel.
Tu, B., Gandy, A., Kantas, N., Shafei, B., 2022. Joint entropy search for multi-objective Bayesian optimization. Adv. Neural Inf. Process. Syst. 35, 9922–9938.
Uyeh, D.D., Pamulapati, T., Mallipeddi, R., Park, T., Asem-Hiablie, S., Woo, S., Kim, J., Kim, Y., Ha, Y., 2019. Precision animal feed formulation: An evolutionary multi-objective approach. Anim. Feed Sci. Technol. 256, 114211.
Van der Poel, A., Abdollahi, M., Cheng, H., Colovic, R., den Hartog, L., Miladinovic, D., Page, G., Sijssens, K., Smillie, J., Thomas, M., et al., 2020. Future directions of animal feed technology research to meet the challenges of a changing world. Anim. Feed Sci. Technol. 270, 114692.
Wang, Y.-L., Liao, C.-N., 2023. Assessment of sustainable reverse logistic provider using the fuzzy TOPSIS and MSGP framework in food industry. Sustainability 15 (5), 4305.
Waugh, F.V., 1951. The minimum-cost dairy feed (an application of “linear programming”). J. Farm Econom. 33 (3), 299–310.
Williams, C.K., Rasmussen, C.E., 2006. Gaussian Processes for Machine Learning. MIT Press, Cambridge, MA.
Yang, K., Emmerich, M., Deutz, A., Bäck, T., 2019a. Efficient computation of expected hypervolume improvement using box decomposition algorithms. J. Global Optim. 75, 3–34.
Yang, K., Emmerich, M., Deutz, A., Bäck, T., 2019b. Multi-objective Bayesian global optimization using expected hypervolume improvement gradient. Swarm Evol. Comput. 44, 945–956.
Zitzler, E., Thiele, L., 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE Trans. Evol. Comput. 3 (4), 257–271.