Titre: Title:	Cadre mathématique pour l'optimisation de boîtes noires avec variables catégorielles et méta
Auteur: Author:	Edward Hallé-Hannan
Date:	2022
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Hallé-Hannan, E. (2022). Cadre mathématique pour l'optimisation de boîtes noires avec variables catégorielles et méta [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/10286/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/10286/
Directeurs de recherche: Advisors:	Charles Audet, & Sébastien Le Digabel
Programme: Program:	Maîtrise recherche en mathématiques appliquées

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Cadre mathématique pour l'optimisation de boîtes noires avec variables catégorielles et méta

EDWARD HALLÉ-HANNAN
Département de mathématiques et de génie industriel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Mathématiques appliquées

Mai 2022

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Cadre mathématique pour l'optimisation de boîtes noires avec variables catégorielles et méta

présenté par **Edward HALLÉ-HANNAN**
en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

Hanane DAGDOUGUI, présidente
Charles AUDET, membre et directeur de recherche
Sébastien LE DIGABEL, membre et codirecteur de recherche
Antoine LESAGE-LANDRY, membre

REMERCIEMENTS

Avant tout, je tiens à remercier mes deux superviseurs, Charles et Sébastien, de m'avoir proposé un projet de recherche ambitieux, éclectique et particulièrement motivant. De plus, je souligne leur intelligence et sagesse, ainsi que leur écoute et ouverture qui ont m'offert un climat idéal pour la réalisation de ma maîtrise. Je me compte très chanceux de pouvoir continuer mes études doctorales avec ces deux superviseurs que je respecte et apprécie énormément.

Je remercie ensuite les organismes IVADO et Hydro-Québec d'avoir financé ma maîtrise. Je les remercie éternellement pour ces marques de reconnaissance, qui m'aideront grandement dans ma carrière académique.

Je tiens à souligner l'importance de mes parents, mon frère et mes amis pour l'aboutissement de ce mémoire. Difficile d'imaginer un entourage aussi stimulant, enrichissant et encourageant.

En définitive, merci infiniment à ma conjointe Stéphanie, qui à partir de ses multiples qualités humaines, m'apporte un soutien irréprochable dans tous mes projets.

RÉSUMÉ

L'optimisation de boîtes noires est une branche de l'optimisation caractérisée par une fonction objectif et des contraintes sans expression analytique. Concrètement, cela implique que les outils traditionnels de l'optimisation, dont les dérivées, ne peuvent être utilisées. La communauté de recherche a tout de même développé des méthodes algorithmiques ingénieuses pour s'attaquer à ce type de problème. À ce jour, la plupart des méthodologies développées traitent des problèmes contenant uniquement des variables continues ou entières. Les méthodologies pour traiter des problèmes ayant des variables catégorielles ou des structures mathématiques dynamiques (variables méta) sont peu nombreuses. Il ne s'agit pas d'un hasard, puisque ces problèmes délaissés contiennent des difficultés qui sont fondamentalement difficiles à surmonter. Dans ce travail, un cadre mathématique, composé d'un système de notation et de stratégies de résolution, est présenté. Le cadre traite des problèmes d'optimisation mixte dans un contexte de boîtes noires.

Le système de notation permet de modéliser explicitement les variables catégorielles et méta dans un problème mixte. Cette modélisation facilite grandement la résolution de tels problèmes. Le terme méta, introduit dans ce travail, décrit les variables spéciales qui influencent le nombre de variables (dimension) ou le nombre de contraintes. Les variables méta ont un impact important sur les définitions mathématiques de base, telles que le domaine et l'ensemble réalisable. Ceux-ci sont rigoureusement définis, ce qui permet d'exprimer plusieurs difficultés liées aux variables catégorielles et méta.

Les différentes méthodologies et approches de la littérature sont catégorisées en deux type de stratégies : la résolution de sous-problèmes et la résolution d'un problème auxiliaire. Les deux stratégies incorporent les différentes méthodologies et approches de la littérature, dont la recherche directe et l'optimisation bayésienne. Le cadre mathématique est donc compatible avec la littérature.

Le cadre mathématique est illustré sur un problème d'optimisation des hyperparamètres, qui est un problème d'optimisation de boîtes noires avec des méta-variables et variables catégorielles. Ce problème est une motivation importante de ce travail. En pratique, il peut être difficile de déterminer efficacement de bons hyperparamètres, puisque l'espace de recherche est mixte et potentiellement très vaste. Ainsi, déterminer de bons hyperparamètres peut être très coûteux temporellement et énergétiquement. Dans ce travail, les fondements mathématiques et des stratégies de résolution sont développés de sorte qu'ils puissent être appliqués à ce type de problème.

ABSTRACT

In blackbox optimization, the objective function and the constraints have no analytical expression, which prevents the deployment of any method based on derivatives. This limitation did not avert any progress by researchers and practitioners. Indeed, the literature in blackbox optimization has been flourishing in the recent years. However, mixed problems with categorical variables and an unfixed mathematical structure (meta variables) have been dismissed by the community, primarily because of the intrinsically difficulties of such problems.

In this work, a mathematical framework for modelling constrained mixed-variable optimization problems is presented in a blackbox optimization context. The mathematical framework is composed of a new notation system and solution strategies.

The notation system allows meta and categorical variables to be explicitly and efficiently modelled, which facilitates the optimization of such problems. The new term meta variables is used to describe special variables that affect the number of variables (dimension) or the number of constraints. Moreover, the domain of the objective function and the feasible set are rigorously defined, which highlight many subtleties associated to meta variables. The flexibility of the solution strategies supports the main blackbox mixed-variable optimization approaches: direct search methods and surrogate-based methods (Bayesian optimization).

The notation system and solution strategies are illustrated through an example of a hyperparameter optimization problem from the machine learning community. This optimization problem is an important motivation behind this work. The interest in deep learning has been growing exponentially in various fields and applications. However, the performance of deep models may be sensible to the choice of hyperparameters, which may limit the performance of such models or drastically increase the time and energy costs of finding good hyperparameters. This work takes this hyperparameter optimization seriously by providing the mathematical foundation to model the underlying difficulties related to categorical and meta variables, as well as furnished state-of-the-art strategies to tackle the problem.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

GPS	General Pattern Search
MADS	Mesh Adaptive Direct Search
GMESH	Granular Mesh
RBF	Radial basis functions
BO	Bayesian optimization
GP	Gaussian Processes
EI	Expected Improvement
EGO	Efficient Global Optimization
MLE	Maximum likelihood estimation
MLP	Multilayer perceptron
ASGD	Asynchronous Stochastic Gradient Descent
Adam	A Method for Stochastic Optimization
ReLU	Rectified Linear Unit

CHAPITRE 1 INTRODUCTION

Ce mémoire s'intéresse à la formulation générale d'un problème d'optimisation sous contraintes :

$$\min_{x \in \Omega \subseteq \mathcal{X}} f(x), \tag{1.1}$$

où $x \in \mathcal{X}$ est un point qui réside dans le domaine $\mathcal{X}$, $f : \mathcal{X} \to \mathbb{R}$ est la fonction objectif et Ω est l'ensemble réalisable défini par les contraintes du problème.

L'optimisation mathématique est divisée en plusieurs branches dont, l'optimisation linéaire, l'optimisation en nombre entiers, l'optimisation convexe, et l'optimisation non lisse pour en mentionner quelques unes. Ces différentes branches ont leur propre formalisme, conditions d'optimalité et méthodes, qui exploitent les propriétés mathématiques accessibles. Par exemple, en optimisation linéaire, la méthode du simplexe [23] est un algorithme exploitant la forme linéaire de la fonction objectif f et des contraintes. Ainsi, la nature de la fonction objectif f et de l'ensemble réalisable Ω régissent quelle branche devrait être employée [7].

1.1 Optimisation de boîtes noires

En optimisation de boîtes noires, la fonction objectif f et les contraintes définissant l'ensemble réalisable Ω sont des boîtes noires. Le terme boîte noire est utilisé pour décrire une fonction ou une contrainte ayant une forme non-analytique, inconnue ou inaccessible [12]. Dans ce contexte, la fonction objectif et les contraintes ont pratiquement aucune propriété mathématique qui peuvent être exploitée. En d'autres mots, la fonction objectif f et les contraintes fournissent de l'information à partir d'évaluations qui sont généralement coûteuses et issues de programmes informatiques [12]. En particulier, la fonction objectif f, sans forme explicite, retourne l'image $f(x) \in \mathbb{R}$ associée au point $x \in \mathcal{X}$.

À partir des remarques précédentes, l'optimisation de boîtes noires est définie comme étant une branche de l'optimisation mathématique qui étudie, conçoit et analyse des algorithmes dont la fonction objectif et/ou les contraintes sont des boîtes noires [12]. Les boîtes noires sont généralement issues d'un programme informatique coûteux. Ainsi, les méthodes développées en optimisation de boîtes noires sont généralement limitées par un budget d'évaluations. Enfin, un exemple de boîte noire, qui n'est pas issu d'un programme informatique, est le résultat d'une expérience chimique en laboratoire [12]. De nombreuses applications de l'optimisation de boîtes noires en ingénierie, au courant des deux dernières décennies, sont décrites dans [7, 18].

1.2 Problèmes mixtes

Les problèmes mixtes sont notoirement compliqués en optimisation de boîtes noires. En effet, la présence des variables catégorielles et méta complexifient grandement les problèmes.

Une variable méta a la particularité d'avoir la propriété de décret. Cette propriété, exclusivement limitée aux variables méta, est assignée aux variables qui déterminent si d'autres variables ou des contraintes font parties du problème ou non. Ainsi, les variables méta peuvent modifier la dimension ou le nombre de contraintes du problème. Dans ce travail, une attention particulière est portée sur ces variables, notamment sur l'influence qu'elles ont sur le domaine $\mathcal{X}$ et l'ensemble réalisable Ω.

Par la suite, les variables catégorielles, qui sont discrètes et qualitatives, sont fondamentalement difficiles à optimiser. En effet, ces variables appartiennent à des ensembles peu structurés. Par exemple, la variable catégorielle représentant la saveur d'une crème glacée $x \in \{\text{"vanille"}, \text{"chocolat"}, \text{"menthe"}\}$ réside dans un ensemble contenant peu de structure. Entre autres, la distance entre la catégorie "vanille" et la catégorie "chocolat" n'a intrinsèquement pas de signification mathématique. Plus formellement, les notions de distance et de proximité ne sont pas bien établies pour des ensembles catégoriels [27]. Par ailleurs, a priori l'ensemble {"vanille", "chocolat", "menthe"} ne contient aucun ordre, puisque les catégories ne sont pas être ordonnées entre-elles : la variable $x \in \{\text{"vanille"}, \text{"chocolat"}, \text{"menthe"}\}$ est plus précisément une variable nominale. D'ailleurs, les variables binaires sont des variables nominales. À l'inverse des variables nominales, une variable ordinale est une variable catégorielle appartenant à un ensemble ordonné. Par exemple, la variable ordinale représentant le niveau d'éducation d'une personne $x \in \{\text{"primaire"}, \text{"secondaire"}, \ldots, \text{"post-doctorat"}\}$ appartient à un ensemble catégoriel et ordonné, puisque les catégories sont ordonnées de "primaire" à "post-doctorat". Il est difficile de développer des méthodes pour optimiser des variables catégorielles, et plus particulièrement en optimisation de boîtes noires. En effet, les méthodes d'optimisation de boîtes noires reposent généralement sur de l'intensification (recherche dans régions prometteuses) et de l'exploration du domaine $\mathcal{X}$, qui sont généralement guidées par des notions de proximité. Ces méthodes ne s'appliquent généralement pas aux variables catégorielles.

En plus des variables catégorielles et méta, les problèmes mixtes peuvent aussi contenir des variables entières ou continues. D'emblée, les problèmes entiers-continus, appelés problèmes standards, sont difficiles en optimisation de boîtes noires. En ajoutant les variables catégorielles et méta aux problèmes standards, la difficulté des problèmes augmente considérablement. Cependant, il existe des méthodologies efficaces pour traiter ces problèmes standards.

En recherche directe, l'algorithme MADS [10] avec le treillis GMesh [14] permet d'optimiser les problèmes standards. En bref, la recherche directe est une approche algorithmique itérative, qui met à jour une solution courante en évaluant localement des points candidats sur un treillis (domaine discrétisé) construit autour de la solution courante. Puis, en optimisation bayésienne (discuté à la section 2), les variables entières peuvent être traitées comme des variables continues à l'aide de simples transformations [25], ce qui permet d'optimiser des problèmes standards.

1.3 Objectif de la recherche

Les méthodes d'optimisation de boîtes noires sous contraintes s'intéressent à déterminer un point optimal $x_\star \in \mathcal{X}$, qui minimise la fonction objectif f et respecte les contraintes ($x \in \Omega$). Dans un contexte d'optimisation mixte, les objets mathématiques du problème (1.1) doivent être rigoureusement développé de manière à formuler convenablement la nature mixte du problème.

À ce jour, les méthodes d'optimisation mixte sous contraintes de boîtes noires sont peu nombreuses et aucune de ces méthodes ne triomphe par rapport aux autres. Ainsi, il est intéressant de développer un cadre mathématique qui est compatible et général avec les différentes approches, dont la recherche directe par voisinage [5, 9, 30] et l'optimisation bayésienne [20, 25, 29, 37, 39, 40]. À partir de ce cadre algorithmique, plusieurs méthodes pourront être développées plus facilement. Par exemple, une méthode intégratrice, prenant le meilleur de chaque approche, pourrait être développée.

L'objectif de la recherche est de développer un cadre mathématique, qui facilite le développement et l'application de méthodes d'optimisation mixte sous contraintes, tout en intégrant les éléments importants de la littérature. Les objectifs de recherche de ce mémoire sont de développer un système de notation complet et de stratégies de résolution compatibles avec les différentes approches. Il s'agit d'un travail qui s'inscrit dans la modélisation mathématique.

1.4 Démarche du travail

Dans ce mémoire, les définitions du problème (1.1) sont formulées avec une attention particulière aux variables catégorielles et méta, qui posent un défi substantiel. Plus particulièrement, un système de notation approprié est en mesure de mettre en évidence certaines difficultés algorithmiques inhérentes aux variables catégorielles et méta. Notamment, l'influence des variables méta sur les autres variables et les contraintes du problème est soigneusement prise en compte. Ensuite, les variables catégorielles sont convenablement distinguées des variables

entières et continues, puisqu'elles sont traitées avec des stratégies d'optimisation précises.

Ensuite, pour arriver à développer le cadre mathématique, il a été nécessaire d'élucider les stratégies et mécaniques derrière les différentes approches de la littérature. Par exemple, la stratégie de l'optimisation bayésienne consiste conceptuellement à formuler un problème auxiliaire moins coûteux qui permet de sélectionner des points à évaluer par la fonction objectif. Bien discerner ces stratégies ont permis de développer un cadre mathématique général et compatible avec les approches principales de la littérature.

1.4.1 Optimisation des hyperparamètres en apprentissage profond

L'optimisation des hyperparamètres en apprentissage profond est un problème mixte d'optimisation de boîtes noires avec des variables catégorielles et méta. L'objectif de ce problème est de maximiser la performance d'un modèle profond en fonction de ses hyperparamètres, qui sont des paramètres de l'architecture du modèle (couches et unités) et des paramètres contrôlant le processus d'apprentissage (optimiseur et taux d'apprentissage). Pour un ensemble d'hyperparamètres donné, l'évaluation de la fonction objectif f consiste à entraîner le modèle et tester sa performance pour la tâche donnée.

En pratique, les méthodes de *Grid Search* et de *Random Search* [17] sont communément utilisées [26]. Cependant, ces méthodes sont généralement inefficaces, notamment puisque l'information provenant des évaluations effectuées n'est pas utilisée pour les évaluations subséquentes.

Pour le reste du mémoire, un problème d'optimisation des hyperparamètres est modélisé, notamment pour illustrer les parties intégrantes du système de notation.

1.5 Plan du mémoire

Pour la suite de ce mémoire, trois chapitres sont présentés. Tout d'abord, un article en anglais, soumis pour une publication, et qui compose la partie principale du mémoire, est présenté au chapitre 3. Une discussion sur le mémoire est étayée dans le chapitre 4 mettant en évidence la contribution principale de ce travail. Finalement, une conclusion du mémoire est détaillée au chapitre 5, avec des extensions possibles prévues pour des travaux futurs.

CHAPITRE 2 REVUE DE LITTÉRATURE

Dans cette section, une revue de la littérature sur l'optimisation mixte de boîte noires est étayée. Le cadre mathématique de ce mémoire est construit de sorte à être compatible avec les éléments importants de la littérature.

2.1 Premières contributions en recherche directe

Une première tentative pour optimiser les boîtes noires mixtes a été effectuée dans [9]. La méthodologie est basée sur l'algorithme de recherche directe *General Pattern Search* (GPS) [42], qui généralise plusieurs algorithmes de recherche directe dont le *Pattern Search* [28] et le *Multidirectional search* [24]. Dans [9], un point $x = (x^{\mathrm{d}}, x^{\mathrm{c}})$ est partitionné en deux composantes : une composante discrète x^{d} et une composante continue x^{c}. La composante discrète x^{d} est un vecteur contenant toutes les variables discrètes, c'est-à-dire les variables entières et les variables catégorielles (ordinales et nominales). La composante continue x^{c} est un vecteur contenant les variables continues. Cet article contient deux contributions importantes. Premièrement, l'espace continu, dans lequel l'algorithme GPS est appliqué, est généré pour une composante discrète fixée x^{d}). Deuxièmement, l'exploration de l'espace des variables discrètes est effectuée à partir d'une structure mathématique additionnelle notée $\mathcal{N}$. Pour un point $x = (x^{\mathrm{d}}, x^{\mathrm{c}})$ donné, l'ensemble $\mathcal{N}(x)$ est l'ensemble des voisins défini manuellement par l'utilisateur de sorte à pouvoir explorer convenablement les variables discrètes à un coût raisonnable. À partir des contributions de [9], un problème d'optimisation d'isolation thermique est traité et optimisé dans [30].

Par la suite, plusieurs extensions ont été apportées à la méthodologie de [9]. Tout d'abord, plusieurs définitions de l'analyse mathématique, dont une série convergente et la continuité de la structure additionnelle $\mathcal{N}$, sont formellement et rigoureusement développées dans la thèse doctorale [2].

Plusieurs contributions de ce mémoire, dont une formulation du domaine de la fonction objectif, sont basées sur cette thèse. Ensuite, dans [5], une méthode de filtre permet en addition de traiter les contraintes non linéaires. Le problème d'isolation thermique de [30] est repris dans l'article [3]. À partir des extensions provenant de [2, 5] le problème d'isolation thermique est plus rigoureusement modélisé : certains phénomènes physiques, dont l'expansion thermique et le stress mécanique, sont modélisés à partir de contraintes non linéaires. Puis, l'algorithme *Mesh Adaptive Direct Search* (MADS) remplace l'algorithme GPS, initialement employé, dans l'article [4].

En réponse aux articles [2,4,5,9,30], l'article [1] propose une méthodologie alternative qui reformule le problème d'optimisation mixte en un problème contenant seulement des variables entières ou continues. À partir de nombreuses contraintes additionnelles, le problème est reformulé sans variable catégorielle. Des méthodes d'optimisation mixte avec des variables entières ou continues peuvent alors être appliquées au problème reformulé. La méthode employée n'est pas générique, puisque la reformulation demande certaines connaissances du problème.

2.2 Variables dimensionnelles

Les variables dimensionnelles ont été introduites dans les articles [34, 35]. Ces variables discrètes spéciales affectent le nombre de variables, le nombre de contraintes ou la structure du problème d'optimisation. Un point $x = (x^{\text{dim}}, x^{\text{d}}, x^{\text{c}})$ est partitionné en trois composantes : la composante dimensionnelle x^{dim}, la composante discrète x^{d} et la composante continue x^{c}. Les variables dimensionnelles sont une contribution importante des articles [34, 35], puisque ces variables ont permis de modéliser que le nombre de variables (discrètes ou continues) et le nombre de contraintes, peuvent varier en fonction des variables dimensionnelles. Les méta-variables, définies dans ce mémoire, sont une généralisation des variables dimensionnelles.

Dans l'article [9], discuté à la section 2.1, les variables dimensionnelles sont incluses dans la composante discrète. Puis, de manière similaire à [9], un ensemble discret est généré pour une composante dimensionnelle fixée. L'espace continu est généré en fixant la composante dimensionnelle et la composante discrète. Le processus d'optimisation est effectué dans cet espace continu à l'aide d'un algorithme basé sur la recherche linéaire. Un problème concret d'optimisation de boîtes noires est traité dans l'article [34]. Le problème consiste à déterminer la structure optimale d'un appareil à résonance magnétique, qui comporte une variable dimensionnelle pour le nombre d'aimants intégrés dans l'appareil [33].

2.2.1 Substituts et variables catégorielles

En optimisation de boîtes noires, un substitut est une fonction moins coûteuse qui approxime la fonction objectif f ou les contraintes du problème. En étant moins coûteux, un substitut peut évaluer plus efficacement plusieurs points de son domaine. Ainsi, pour la fonction objectif f, un substitut peut fournir des points candidats provenant d'une région prometteuse en optimalité (intensification) ou d'une région sous-explorée (exploration) [29]. L'optimisation bayésienne discutée dans la section 2.3 est une approche dite par substituts (*surrogates*).

Dans l'article [38], une approche par substituts est employée pour traiter les variables catégo-

rielles dans des problèmes mixtes sans variable dimensionnelle (ou méta). Un noyau catégoriel a été ajouté à une méthode par substituts basée sur des fonctions de base radiale (RBF), qui permettait déjà d'optimiser des problèmes entiers-continus. En statistique, un noyau est une fonction symétrique qui prend en argument la distance entre deux éléments d'un espace, puis retourne soit une valeur proche de 1 pour des éléments rapprochés, soit une valeur proche de 0 pour des éléments éloignés. Pour des variables catégorielles, la notion de distance entre les composantes n'est pas bien définie. Ainsi, l'article [38] propose une méthode pour émuler une distance entre les composantes catégorielles (vecteur comportant les variables catégorielles) en calculant le nombre de variables catégorielles qui sont non-identiques entre deux composantes catégorielles. Le substitut est construit à partir d'un noyau composé, tel que les fonctions de base radiale, centrées aux points d'interpolation entier-continus, sont translatées par le nombre de variables catégorielles non-identiques.

2.3 Optimisation bayésienne

L'optimisation bayésienne (OB) est une approche par substituts ayant la particularité d'utiliser des modèles probabilistes basés sur des processus gaussiens [40]. Aujourd'hui, la plupart de la littérature en optimisation mixte de boîtes noires est ancrée dans l'optimisation bayésienne : l'avènement de l'apprentissage automatique a été très bénéfique à l'optimisation bayésienne, puisqu'ils sont reliés. En pratique, le succès de l'OB est partiellement expliqué par l'utilisation de fonctions d'acquisition, qui sélectionnent des points candidats, sur un problème d'optimisation moins coûteux que le problème original. Entre autres, la fonction d'acquisition *Expected Improvement* (EI) [29] est très connue en optimisation continue de boîtes noires. L'EI sélectionne des points candidats dans de régions sous-explorées (exploration) ou provenant des régions prometteuses (intensification).

Les méthodes par substituts utilisant des processus gaussiens et la fonction d'acquisition EI sont appelés *Efficient Global Optimization algorithms* (EGO) [29]. Historiquement, les méthodes EGO ont été utilisées pour traiter des problèmes d'optimisation continue de boîtes noires. Ainsi, les variables entières et catégorielles ont souvent été traitées avec des approches naïves. Par exemple, en pratique ces variables ont souvent été encodées dans des vecteurs binaires, puis relaxées en variables continues (même si la notion de distance entre les variables catégorielles est mal définie) et arrondies pour être compatible avec la fonction objectif du problème original [25]. Ces approches naïves, encore à ce jour utilisées, mènent souvent à plusieurs problématiques. Par exemple, les points sélectionnés par la fonction d'acquisition peuvent être associés aux mauvais points du domaine de la fonction objectif du problème original [25]. Quelques articles, qui ont développé des méthodes plus rigoureuses, sont discutés

ci-dessous.

Dans [20], une méthode basée sur l'optimisation bayésienne a été développée pour traiter les problèmes continus-catégoriques. Dans cette méthodologie, les processus gaussiens sont construits à partir d'un noyau composé d'additions et multiplications tensorielles de noyaux unidimensionnels (un noyau par variable). Le noyau d'une variable catégorielle $x_j \in \{1, 2, \ldots, C\}$ est une matrice $C \times C$, où un élément d'une matrice est la corrélation entre deux catégories de x_j. Les noyaux matriciels des variables catégorielles sont distincts pour les variables ordinales et nominales.

Ensuite, dans [39], l'optimisation bayésienne est étendue aux problèmes mixtes, contenant des variables continues, discrètes (entières et catégorielles) et dimensionnels [34, 35]. Les processus gaussiens sont construits à partir d'un noyau composé d'additions et multiplications de noyaux unidimensionnelles. De plus, deux approches sont proposées : 1) un substitut par composante dimensionnelle (vecteur de composantes discrètes spéciales), ce qui sépare le problème en plusieurs sous-problèmes et 2) un seul substitut avec un noyau composé sur toutes les variables, incluant les variables dimensionnelles.

Enfin, les auteurs de [37] ont utilisé la structure de voisinage $\mathcal{N}$, définit dans [9], pour traiter les variables catégorielles dans une méthodologie EGO. Plus précisément, la structure de voisinage $\mathcal{N}$ est définie aléatoirement à partir d'une distribution discrète basée sur les processus gaussiens. Ainsi, les variables catégorielles sont explorées avec une étape de recherche aléatoire dans la méthodologie EGO.

2.4 Variables latentes pour les variables catégorielles

Les noyaux sont fondamentalement difficiles à définir sur des ensembles catégoriels, puisque la distance entre deux catégories n'est pas bien définie. Dans [8], il est proposé d'encoder les variables catégorielles dans un espace continu appelé l'espace latent. Les catégories $\{1, 2, \ldots, C_j\}$ de chaque variable catégorielle $x_j \in \{1, 2, \ldots, C_j\}$ sont assignées à un point dans un espace continu 2D (espace latent). La position des points dans l'espace latent n'est pas particulièrement importante, ce qui importe est la distance relative entre les points de cet espace. En effet, les catégories sont assignées dans cet espace de sorte que les catégories corrélées soient proches entre-elles et les catégories moins corrélées soient plus éloignées. Les catégories sont assignées par une procédure d'estimation de vraisemblance, où les données sont supposées être distribuées par une loi multinormale. Dans [22], les auteurs ont formalisé un problème contraint de pré-image qui permet de récupérer la composante catégorielle associée au vecteur des variables latentes continues. Plus techniquement, un problème continu EGO est formulé avec un lagrangien augmenté.

CHAPITRE 3 ARTICLE 1 : A GENERAL MATHEMATICAL FRAMEWORK FOR CONSTRAINED MIXED-VARIABLE BLACKBOX OPTIMIZATION PROBLEMS WITH META AND CATEGORICAL VARIABLES

Charles Audet • Sébastien Le Digabel • Edward Hallé-Hannan

Soumis à la revue: Operation Research Forum

Abstract

A mathematical framework for modelling constrained mixed-variable optimization problems is presented in a blackbox optimization context. The framework introduces a new notation and allows solution strategies. The notation framework allows meta and categorical variables to be explicitly and efficiently modelled, which facilitates the solution of such problems. The new term meta variables is used to describe variables that influence which variables are acting or nonacting : meta variables may affect the number of variables and constraints. The flexibility of the solution strategies supports the main blackbox mixed-variable optimization approaches : direct search methods and surrogate-based methods (Bayesian optimization). The notation system and solution strategies are illustrated through an example of a hyperparameter optimization problem from the machine learning community.

Keywords. Blackbox optimization, derivative-free optimization, mixed-variable optimization, categorical variables, meta variables.

3.1 Introduction

This work considers a general constrained optimization problem

$$\min_{x \in \Omega \subseteq \mathcal{X}} f(x), \tag{3.1}$$

where $x \in \mathcal{X}$ is a point that resides in the domain $\mathcal{X}$, $f : \mathcal{X} \to \mathbb{R}$ is the objective function and $\Omega \subseteq \mathcal{X}$ is the feasible set defined by the constraints of the problem.

3.1.1 Context and motivation

In blackbox optimization (BBO), the objective and constraint functions are assumed to be blackboxes. In [12], a mathematical blackbox is defined as : "*any process that when provided an input, returns an output, but the inner working of the process are not analytically available*". In general, these blackboxes are computer programs with expensive processes. Hence, in BBO, the objective and constraint functions can only provide information through evaluations, which are generally costly. For instance, the only information that the objective function f can provide is the mapping of a point $x \in \mathcal{X}$ to its image $f(x) \in \mathbb{R}$ through a given process, which is typically a computer program. Consequently, derivatives are often inaccessible or too costly to compute. Thus, in general, traditional optimization methods cannot be applied to blackbox functions [12].

Mixed-variable problems are notoriously hard to tackle in the blackbox optimization community. This can be partly explained by the presence of meta and categorical variables. Meta variables are a special type of variables that may affect the dimension, the number of constraints and determine which variables are included or excluded in the optimization process. These special variables are a cornerstone of this work and are thoroughly defined in Section 3.3. Moreover, categorical variables are fundamentally difficult to treat since they belong to discrete sets that do not contain any intrinsic metric of distance between the elements and they cannot be relaxed easily in comparison to integer variables. An example of categorical variable is the blood-type of a given person $x \in \{$O-, O+, A-, A, $\ldots\}$ In conjunction, the meta and categorical variables give rise to a substantial challenge in a context of blackbox optimization. In addition, Problem (3.1) may contain continuous or integer variables.

The compact and general formulation of Problem (3.1) does not explicitly model mixed-variable problems. Hence, in order to efficiently tackle these problems, the formulation must be further detailed with a focus on treating the meta variables and the categorical variables. A core aspect of this work is to formally define the domain $\mathcal{X}$, which has many implications in the mathematical framework that consists of a notation system and solution strategies. The notation framework rigorously models constrained mixed-variable problems in an efficient and unambiguous manner, as well as shines the light on some algorithmic subtleties in the solution of these problems. The present work also formalizes solution strategies present in the literature and tackles these problems by being fully compatible with the main blackbox optimization approaches : direct search and surrogate-based Bayesian optimization approaches. The present work does not present any computational experiments, as it focuses on the presentation of the framework, in the same way that the well-known surrogate management framework [19], was proposed without experiments.

3.1.2 Literature review

A first framework to treat mixed-variable optimization problems in a context of blackbox optimization is detailed in [9]. The methodology is based on the general pattern search algorithm (GPS) and the variables are partitioned into two components : discrete and continuous. The discrete component contains both the quantitative and the qualitative discrete variables, *i.e.*, integer variables in $\mathbb{Z}$ as well as categorical variables. The continuous component contains the continuous variables. Two main ideas emerged from this article. First, the continuous space, in which classical continuous blackbox optimization methods can be applied, are generated after fixing the discrete component. Thus, for a fixed discrete component, a continuous space is generated and explored. Second, the exploration of the discrete variables space is being done by defining a set of neighbors function $\mathcal{N}$, which is an additional structure to the domain $\mathcal{X}$, such that $\mathcal{N}(x)$ is a set of neighbors of $x \in \mathcal{X}$. With this additional structure a local minimizer $x_\star$ is defined so that $x_\star$ minimizes the objective function f with respect to the set of neighbors (discrete part) and the continuous space. From the contributions of [9], a practical application of a thermal insulation optimization problem is treated and optimized [30]. In [5], the filter method is added to the methodology proposed in [9, 30]. This addition enables the methodology to treat general nonlinear constraints. In [4], the methodology based on GPS in [5,9,30] is extended to the mesh adaptive direct search (MADS) algorithm [10]. A rigorous convergence analysis based on [5] was improved by using the Clarke generalized derivatives on the continuous space. In [14], the MADS algorithm is equipped with a granular mesh called GMesh, which allows the discretization of granular and continuous variables simultaneously. Granular variables are quantitative variables with a controlled number of decimals. In particular, GMesh enables to treat integer-continuous problems with the MADS algorithm since integer variables are a special type of granular variables without decimals.

An important contribution from [34, 35] is the introduction of dimensional variables. These variables affect the number of variables, the number of constraints and the structure of the optimization problem. A point x is partitioned into three components : a dimensional component, a discrete component and a continuous component. The discrete set, where the discrete component belongs, is generated from a fixed dimensional component. Additionally, the continuous space is generated from both a fixed dimensional component and a fixed discrete component. From the partition of a point x, a domain and a feasible set are implicitly presented in the formulation of an optimization problem. The present work importantly relies on the contributions from [34, 35].

A categorical kernel function is defined in [38] with the aim of tackling mixed-variable optimization problems with a surrogate approach based on radial basis functions (RBF). The

categorical kernel function measures the number of disagreement between two categorical components, where a disagreement is counted when a specific variable of the two compared components is not the same. The surrogate is built upon a composed kernel such that the RBF, centered at some interpolation points, are shifted by the number of categorical disagreements between the fitted point and the interpolation points. The criterion to determine which point is evaluated by the objective function is based on [41]. In essence, the criterion has a high value for points that are distant from the previous evaluations (exploration) or points that have promising surrogate-value (intensification).

Bayesian optimization (BO) has undergone significant development with the recent advent of machine learning. Nowadays, the emerging scientific literature is mainly related to BO based on Gaussian processes (GP), which serves as probabilistic distribution surrogates [40]. The success of BO is explained by an acquisition function that selects which candidate point is to be evaluated. The acquisition function defines a less costly optimization problem with the surrogate. For continuous problems, a well documented acquisition function is the expected improvement (EI) [29], which provides candidate points in unexplored regions (exploration) and candidate points in promising regions (intensification) : algorithms that applies an EI function on a GP are often referred to as efficient global optimization (EGO) algorithms [29]. Historically, BO based on GPs was used to tackle continuous blackbox optimization problems. Hence, in practice the integer and categorical variables (one-hot encoded) are often relaxed as continuous variables and rounded afterwards [25]. This naive approach, used in some modern blackbox solvers, often leads to failure such as a mismatch between the points provided by an acquisition function and where the true evaluation takes place, as well as reevaluating some points [25]. Moreover, an important number of additional variables may be generated by the one-hot encoding of categorical variables. In [20], continuous-categorical optimization problems are modelled with GPs, where a GP surrogate is characterized by a kernel composed of tensor products and additions of one-dimensional kernels : an one-dimensional kernel per variable. The one-dimensional kernel of a given categorical variable $x_j \in \{1, 2, \ldots, C\}$ is a $C \times C$ matrix, where an element of the matrix is a correlation measure between two categories (classes) of x_j. The matrix-kernels for the ordinal and nominal categorical variables are distinguished. In [39], the BO framework is extended to tackle mixed-variable optimization problems with continuous, discrete (categorical and integer) and dimensional variables, such as defined in [34, 35]. Again, the GP surrogate is characterized by a composed kernel built upon products and additions of one-dimensional kernels, each specified by the type of its corresponding variable. Moreover, two approaches are proposed in [39] : multiple surrogates, one surrogate per dimensional component (set of dimensional variables), which separates the main problems into subproblems and a single surrogate with a composed kernel built upon

on all variables, including dimensional variables.

In [37], the authors combined the user-defined set of neighbors in order to tackle categorical variables in an EGO subproblem. More precisely, a user-defined set of neighbors is randomly defined with a discrete probability distribution based on a GP. Thus, the randomly user-defined set of neighbors serves as a randomized categorical exploration strategy for the EGO subproblem.

Covariance functions (kernels) are fundamentally difficult to defined on categorical sets since the distance between two categories (levels) is not defined. To tackle this difficulty, the authors in [8] proposed to map the categories of each categorical variable to a set of quantitative values that represents some underlying latent unobservable quantitative variable. More precisely, the categories of each categorical variable are mapped to a 2D continuous space : for a given categorical variable, the categories are compared into a 2D space. The quantitative values in the vectors does not have any intrinsic meaning. However, the distance between the values encapsulates some information, since the categories are mapped among themselves in a correlated manner. Mathematically, the mapping is done via a maximum likelihood estimation (MLE) procedure that fits the best multivariate Gaussian distribution of some data. A GP model is then constructed on continuous variables and latent variables. Furthermore, the authors in [22] formalized a pre-image problem with a constraint that recovers a categorical component from a vector of continuous latent variables. More technically, a continuous EGO problem is formulated as an augmented Lagrangian with a retrieving constraint on the continuous latent variables.

The document is organized as follows. First, an example of a mixed-variable optimization problem, taken from the machine learning community, is described in Section 3.2. The example is used throughout the paper to facilitate understanding. Second, the notation system is exhaustively detailed in Section 3.3. The notation partitions variables in different types, classifies constraint functions, and formally presents their domain and the feasible set. Finally, solution strategies are presented in Section 3.4 from the framework perspective.

3.2 Hyperparameter multilayer perceptron example

In order to illustrate the mathematical framework, a simplified constrained hyperparameter optimization problem on a multilayer perceptron (MLP) is detailed throughout the document. Some important hyperparameters are internationally left out, such as the mini-batch size or the dropout. The goal of the detailed problem is to model a simple constrained mixed-variable optimization problem in a deep learning context. The objective function is composed of the training and testing of a deep neural network model on a given task. The goal is to find

the set of hyperparameters that maximizes a performance score, which is usually a precision score of accuracy on a untested data set.

In the example, the MLP is defined to perform regression for inputs with $p \in \mathbb{N}$ continuous features. In other words, the MLP approximates a nonlinear function $h : \mathbb{R}^p \to \mathbb{R}$. In order to respect the dimensions of the domain and the codomain of the function h, the architecture of the network must have $u_{in} = p$ units in the input layer and $u_{out} = 1$ unit in the output layer. The hyperparameters of the MLP are described in Table 3.1.

Table 3.1 Hyperparameters of the MLP.

Hyperparameter	Variable	Scope
Learning rate	r	$]0, 1[$
Activation function	a	{ReLU, Sigmoid}
# of hidden layers	l	$\{0, 1, \ldots, l^{\max}\}$
# of units hidden layer i	u_i	$\{u_i^{\min}, u_i^{\min} + 1, \ldots, u_i^{\max}\}$
Optimizer	o	{Adam, ASGD}
if $o =$ ASGD		
decay	λ	$]0, 1[$
power update	α	$]0, 1[$
averaging start	t_0	$]1E3, 1E8[$
if $o =$ Adam		
running average 1	β_1	$]0, 1[$
running average 2	β_2	$]0, 1[$
numerical stability	ϵ	$]0, 1[$

The index i in u_i represents the i-th hidden layer. The number of units in the hidden layers are grouped in the vector $u(l) = (u_1, u_2, \ldots, u_l)$, where l is the number of hidden layers. The situation where there are no hidden layer is modeled by setting $l = 0$. In that case, the variables u_i are said to be nonacting, which signifies that the variables u_i are not part of the optimization problem when $l = 0$. The terminology of acting and nonacting are further detailed in Section 3.3.1.1 and Section 3.3.1.2.

Depending on the choice of the optimizer, different hyperparameters are involved. Indeed, in Table 3.1 the optimizers do not share the same continuous hyperparameters. A given optimizer leads to different variables in the problem. For example, the variable decay λ is only part of the problem (acting) if $o =$ ASGD. This consideration is important and will be discussed throughout the document, but notably in Section 3.3.1.1 that focuses on meta variables.

The first constraint of the problem imposes that the sum of the units in all the hidden layers does not exceed an upper bound $\hat{u} \in \mathbb{N}$, such that $\sum_{i=1}^{l} u_i \leq \hat{u}$. The other constraints are

$u_i \leq u_{i-1}\ \forall i \in \{2, 3, \ldots, l\}$ and they impose that the number of units in subsequent hidden layers are less than or equal, which may help reduce the number of units. These artificial constraints are imposed to illustrate the notation (see Section 3.3.4). An example of a possible architecture of the MLP is schematized in Figure 3.1.

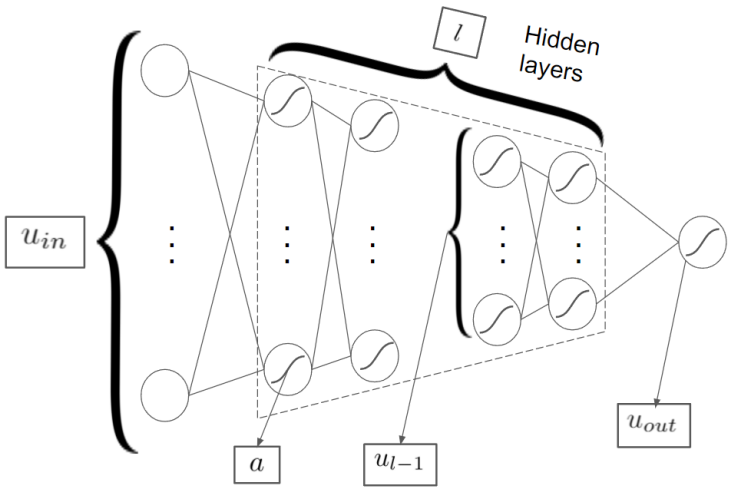


FIGURE 3.1 MLP of the hyperparameter problem (see Table 3.1).

3.3 Notation framework

This section contains the fundamental mathematical definitions that allow modelling mixed-variable problems. In Section 3.3.1, the mathematical objects that define the variables (point and components) are described. Subsequently, the domain $\mathcal{X}$ is detailed in Section 3.3.2. Then, the feasible set $\Omega \subseteq \mathcal{X}$ is precised in Section 3.3.3. Finally, the content in Sections 3.3.1 to 3.3.3 is discussed within the MLP example in Section 3.3.4.

3.3.1 Variables and components of a point

The goal of an optimization algorithm is to find a feasible point $x_\star$ that minimizes the objective function f. In a mixed-variable optimization context, it is necessary to formally define how a point is partitioned into different components.

Definition 1 (Components of a point)**.** *A point $x = (x^m, x^q, x^s)$ is partitioned into three components :*

- *a meta component x^m ;*
- *a categorical component $x^q = (x^{q_u}, x^{q_o})$, which itself is partitioned into the unordered categorical (nominal) component x^{q_u} and the ordered categorical (ordinal) component*

x^{q_o} ;

— *a standard component* $x^s = (x^z, x^c)$*, which itself is the fusion of the integer component* x^z *and the continuous component* x^c*.*

For each $t \in \{m, q, s, q_u, q_o, z, c\}$*, the component* x^t *is a vector containing* $n^t \in \mathbb{N}$ *variables of type* t *:*

$$x^t = (x_1^t, x_2^t, \ldots, x_{n^t}^t). \tag{3.2}$$

The fusion of the integer and continuous components into the standard component x^{s} is justified by several reasons. In practice these variables are generally optimized with standard methods. Moreover, some blackbox optimization algorithms have the ability to simultaneously optimize integer and continuous variables. Thus, it is convenient to group these variables to lighten the notation. However, the standard component $x^{\mathrm{s}} = (x^{\mathrm{z}}, x^{\mathrm{c}})$ can easily be partitioned into its two components if necessary.

The meta, standard and categorical components, as well as their corresponding variables, are respectively discussed in Sections 3.3.1.1, 3.3.1.3 and 3.3.1.4. Additionally, the motivations behind the compact partition $x = (x^{\mathrm{m}}, x^{\mathrm{q}}, x^{\mathrm{s}})$ and the complete partition $x = (x^{\mathrm{m}}, x^{\mathrm{q_u}}, x^{\mathrm{q_o}}, x^{\mathrm{z}}, x^{\mathrm{c}})$ are discussed and illustrated in Section 3.3.1.5. Finally, in Section 3.3.1.2, the roles of variables and constraints are introduced in order to define more clearly the domain $\mathcal{X}$ in Section 3.3.2.

3.3.1.1 Meta component and decree property

The meta component x^{m} contains variables having the decree property, which are called meta variables. The decree property is a special property that only meta variables possess. The property determines if some variables or constraints are either acting or nonacting. The term acting indicates that the variable or constraint is part of the problem and, on the contrary, the term nonacting indicates that the variable or constraint is not part of the problem. More precisely, an acting variable is a decision variable that is included in the domain in which the optimization process is deployed. An acting constraint is a constraint function that defines the feasible set that contains feasible solutions.

The decree propriety is attributed to the meta component x^{m}, since it contains the meta variables. Concretely, meta variables may affect the number of variables (dimension) or the number of constraints. In the MLP example, the number of hidden layers l affect the dimension and the number of constraints of the problem, whereas the optimizer o does not. Indeed, l affect the number of variables (dimension) since it decrees the units u_i in

the hidden layers $i \in \{1, 2, \ldots, l^{\max}\}$, such that $u_i \in \{u_1, u_2, \ldots, u_l\}$ are acting variables and $u_i \in \{u_{l+1}, u_{l+2}, \ldots, u_{l^{\max}}\}$ are nonacting. Moreover, l also decrees the corresponding constraints $u_i \leq u_{i-1}$ $\forall i \in \{2, 3, \ldots, l\}$, thus affecting the number of constraints. Both optimizers ASGD and Adam from Table 3.1 decree three continuous hyperparameters, which does not affect the dimension nor the number of constraints. However, the optimizer decrees some variables. For instance, the decay λ is only an acting variable if $o = \text{ASGD}$.

In that regards, meta variables are a generalization of the strictly discrete dimensional variables defined in [34,35]. First of all, meta variables do not necessarily affect the dimension, in comparison to dimensional variables. Secondly, meta variables can be of any type. For example, a problem could contain a continuous variable frequency that takes its value in the visible spectrum (continuous scope). The visible spectrum could be partitioned into the three intervals that represent the red-blue-green colors. Finally, the frequency could decree some variables or constraints depending in which interval (color) it belongs to. In that particular example, the frequency is a meta-continuous variable.

Additionally, the terminology *dimensional* used in [34, 35, 39] is avoided, since it is used in physical sciences and engineering to describe quantities such as the velocity, mass and time. Many of blackbox mixed-variable optimization problems come from these disciplines.

3.3.1.2 Roles of variables and constraints

The present section introduces the *roles* of variables and constraints. They are introduced for two reasons : 1) they facilitate the comprehension of the influence of meta variables on the other variables of type $t \in \{\mathrm{q}, \mathrm{s}, \mathrm{q_u}, \mathrm{q_o}, \mathrm{z}, \mathrm{c}\}$ and constraints ; 2) the definition of the domain $\mathcal{X}$ in Section 3.3.2 and the feasible set Ω in Section 3.3.3 are made clearer. In essence, the roles of variables and constraints consist of additional terminologies that help elucidate some subtleties of the mathematical framework.

The role of a variable must not be confused with its variable type. In addition to its type $t \in \{\mathrm{m}, \mathrm{q}, \mathrm{s}, \mathrm{q_u}, \mathrm{q_o}, \mathrm{z}, \mathrm{c}\}$, each variable takes a single role amongst meta, decreed or global. A constraint takes a single role amongst decreed or global.

The roles of meta variables is simply their meta type : meta is both a variable type and a role. The role of meta variables is to decree their decreed variables or constraints. More precisely, some variables or constraints may be acting or nonacting accordingly to some specific meta variables. These variables or constraints are said to be decreed. They are called decreed variables and decreed constraints. In the MLP example of Table 3.1, the decay λ is an acting variable if the optimizer $o = \text{ASGD}$ and is nonacting otherwise. Thus, the optimizer o is a

meta variable and the decay λ is a decreed variable. The decay λ is also said to be decreed by the optimizer o. Conceptually, the role of decreed variables or constraints is to be acting or nonacting with respect to their specific meta variables [1].

Additionally, the decree property has been attributed to meta component x^{m} in Section 3.3.1.1. Thus, the meta component x^{m} has an important role in which it decrees all the decreed variables, since the meta component x^{m} contains all the meta variables.

The last role is the simplest one. Global variables or constraints are always acting and do not possess the decree property. In other words, the global variables or constraints are not meta and are always part of the problem. Conceptually, global variables or constraints have an empty role in the sense that they do not influence and are not influenced by other variables or constraints. In the MLP example of Table 3.1, the activation function a is a global variable since it is not decreed by any variable and it does not decree other variable.

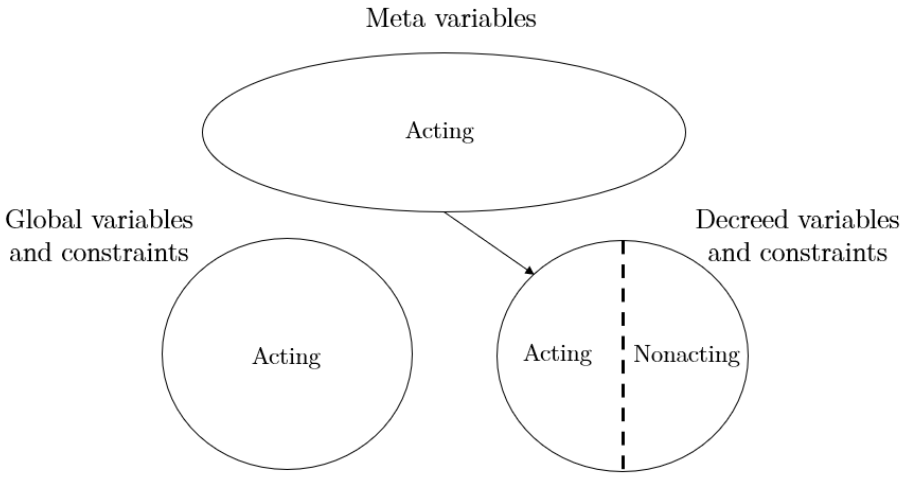


FIGURE 3.2 Role classification of variables and constraints.

Figure 3.2 summarizes the roles of variables and constraints by illustrating some important concepts. First, the arrow symbolizes that meta variables decree some variables or constraints. Second, nonacting variables and constraints are a subset of decreed variables and constraints. This implies that nonacting variables or constraints are necessarily decreed. Third, the global variables and constraints are disjoint, which indicates that they are unaffected by the meta variables.

1. Note that a decreed variable can not be a meta variable : this modeling choice is being done to allow only one instance of meta variables. In other words, meta variables that decrees meta variables are not allowed, since it is very uncommon to encounter such problems in practice and it would complexify the notation even more.

3.3.1.3 Categorical component

The categorical component x^{q} contains qualitative variables, known as categorical variables, that are not meta : a categorical variable may be decreed or global. Categorical variables are discrete variables that take qualitative values called categories. More precisely, a categorical variable x_j^{q} has c_j categories, such that $x_j^{\mathrm{q}} \in \{1, 2, \ldots, c_j\}$.

Categorical variables can be unordered or ordered. Unordered categorical variables are known as nominal variables (*e.g.*, the blood-type) and they are contained in the unordered categorical component $x^{\mathrm{q_u}}$. Note that binary variables are nominal variables. Subsequently, ordered categorical variables are also known as ordinal variables and they are contained in the categorical ordered component $x^{\mathrm{q_o}}$. The size of a pizza $x \in \{\text{small},\ \text{medium},\ \text{large}\}$ is an ordinal variable, since the categories are ordered from small to large. Although the ordinal variables belong to ordered sets, distances between the ordinal variables are inherently unknown : "[...] *there is an ordering between the values, but no metric notion is appropriate*" [27]. For short, the terms nominal and ordinal are prioritized over unordered categorical and ordered categorical respectively.

The categorical component $x^{\mathrm{q}} = (x^{\mathrm{q_u}}, x^{\mathrm{q_o}})$ is composed of the nominal component $x^{\mathrm{q_u}}$ and the ordinal component $x^{\mathrm{q_o}}$, which respectively contains the nominal variables and the ordinal variables that are not meta. In some cases, it might be beneficial to exploit the order of an ordinal variable, motivating the partition of the categorical component into nominal and ordinal components. For instance, [20] used different kernels for ordinal and nominal components. Moreover, a direct search exploration strategy could be generically implemented with a previous and next element mechanism for an ordinal set.

In previous works [5, 9, 30], meta variables were included in the categorical variables ; it is an important distinction from this work.

3.3.1.4 Standard component

The standard x^{s} component contains discrete and continuous quantitative variables that are not meta variables : a standard variable may be decreed or global. Formally, the standard component x^{s} contains variables that belong to intrinsically ordered sets for which a metric of distance is intuitively definable. Simply put, the standard component x^{s} contains the integer variables and the continuous variables.

The integer component x^{z} exclusively contains discrete quantitative variables, called integer variables, that are not meta. Unlike the categorical variables, integer variables are always

ordered and belong to sets with appropriate metric notions [27]. The decision to separate the discrete variables into the categorical component and the integer component differs from some of the current literature. Indeed, in [4,35,39] the discrete component contains both the categorical and the integer variables. Thus, categorical and integer variables are not clearly distinguished : some useful mathematical properties of the integer variables might not be exploited at their fullest. In that regard, integer programming is a well developed optimization field that exploits the properties of the integer variables. In practice, this strengthens the separation of the integer variables from the categorical ones, since integer programming techniques could be implemented in the algorithmic framework to treat the integers variables.

The continuous component x^{c} contains continuous variables that are not meta. Continuous variables have many properties that are generally exploited in a context of blackbox optimization.

3.3.1.5 Variable type classification

Figure 3.3 shows a tree chart that classifies a variable by their type in the proposed mathematical framework. The first question identifies meta variables, the second determines the continuous variables, the third distinguishes integer from categorical variables and the last one separates ordinal from nominal variables. The first question also imply that continuous, integer, ordinal and nominal variables are not meta variables. The doted box in the middle illustrates that standard variables contains the continuous and integer variables, whereas the doted box in the bottom exhibits that two types of categorical variable, which are ordinal and nominal variable.

Mathematically, the partition of a point complete partition $x = (x^{\mathrm{m}}, x^{\mathrm{q_u}}, x^{\mathrm{q_o}}, x^{\mathrm{z}}, x^{\mathrm{c}})$, displayed in Figure 3.3, offers flexibility and extracts most mathematical information accessible to facilitate the optimization process : the modelling choices for the partitions are motivated by these considerations. The compact partition $x = (x^{\mathrm{m}}, x^{\mathrm{q}}, x^{\mathrm{s}})$ implicitly contains the same information and flexibility of the full partition. However, the compact partition alleviates the notation, which is why it is mostly used throughout this work.

3.3.2 Domain

At this stage, the variables have been : 1) classified into different types ; 2) organized into components, which forms a partition of a point x ; 3) attributed roles. The next step is to define the domain $\mathcal{X}$ of the objective function $f : \mathcal{X} \to \mathbb{R}$, such that a point $x \in \mathcal{X}$ resides in that set.

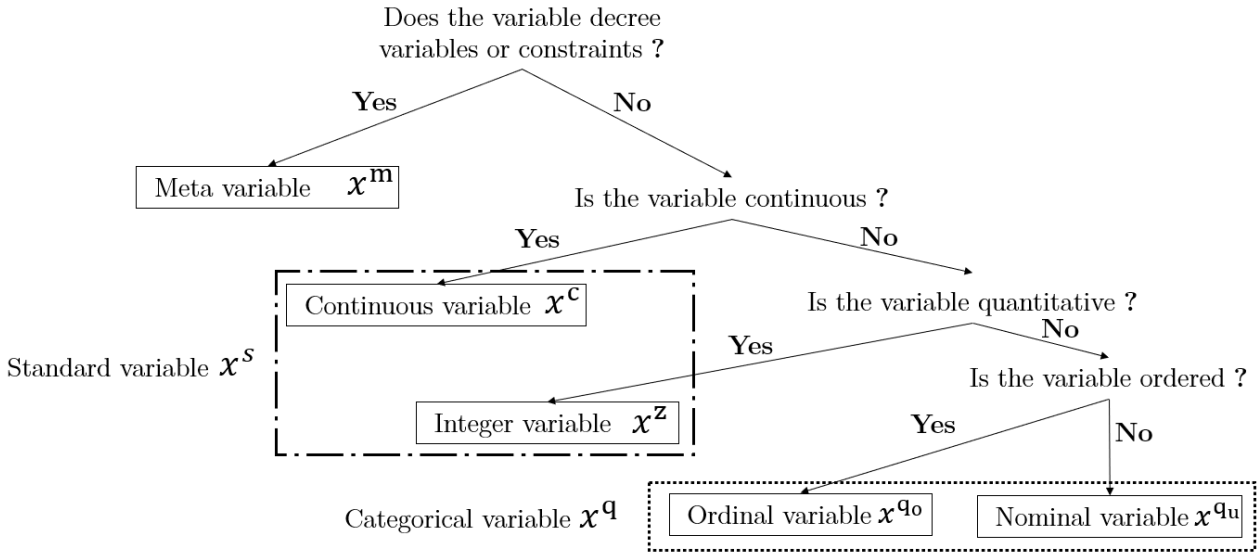


FIGURE 3.3 Variable type classification tree chart.

Definition 2 (Domain)**.** *The domain of objective function is defined by :*

$$\mathcal{X} = \Big\{ \; (x^m, x^q, x^s) \quad : \quad \begin{array}{l} x^m \in \mathcal{X}^m, \\ x^q \in \mathcal{X}^q(x^m), \\ x^s \in \mathcal{X}^s(x^m) \; \Big\} \end{array} \tag{3.3}$$

where $\mathcal{X}^m \subseteq \mathbb{M}^{n^m}$ *is the meta set,* $\mathcal{X}^q(x^m) \subseteq \mathbb{Z}^{n^q(x^m)}$ *is the parametrized categorical set and* $\mathcal{X}^s(x^m) \subseteq \mathbb{Z}^{n^z(x^m)} \times \mathbb{R}^{n^c(x^m)}$ *is the parametrized standard set.*

The dependencies of the parametrized categorical set $\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ and parametrized standard set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ are defined through a parametrization with respect to the meta component x^{m}. These parametrizations are a direct consequence of the decree property of the meta component x^{m}.

Definition 3 (Parametrized set)**.** *A parametrized set* $\mathcal{X}^t(x^m)$ *of type* $t \in \{q, s, q_u, q_o, z, c\}$ *is the set that contains all the components of type* t*, such that*

$$\mathcal{X}^t(x^m) = \Big\{ \; x^t = \left(x_1^t, x_2^t, \ldots, x_{n^t(x^m)}^t\right) \; : \; x_i^t \in S_i^t \text{ is an acting variable } \forall i \in I^t(x^m) \Big\} \tag{3.4}$$

where S_i^t *is the scope of the acting variable* x_i^t *and* $I^t(x^m) = \{1, 2, \ldots, n^t(x^m)\}$ *is the set of indices of the acting variables* x_i^t*, which are either global or decreed by the meta component* $x^m \in \mathcal{X}^m$*.*

From Definition 3, it follows that a component $x^t \in \mathcal{X}^t(x^{\mathrm{m}})$ contains only the acting variables

of type t. The nonacting variables are not contained in the component $x^t \in \mathcal{X}^t(x^{\mathrm{m}})$. Recall that nonacting variables are necessarily decreed variables, whereas acting variables may be global or decreed. Hence, in the component $x^t \in \mathcal{X}^t(x^{\mathrm{m}})$, some acting variables contained may be decreed by the meta component $x^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}}$, which justifies the parametrization of the set $\mathcal{X}^t(x^{\mathrm{m}})$.

Two additional remarks follow. First, the meta variables are always acting variables, thus the meta set $\mathcal{X}^{\mathrm{m}}$ has no dependency. Secondly, a parametrized set $\mathcal{X}^t(x^{\mathrm{m}})$ is a subset of the set that contains all possible components $\mathcal{X}^t$, such that

$$\mathcal{X}^t(x^{\mathrm{m}}) \ \subseteq \ \mathcal{X}^t \ = \bigcup_{x^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}}} \mathcal{X}^t(x^{\mathrm{m}}), \tag{3.5}$$

where $t \in \{\mathrm{q}, \mathrm{s}, \mathrm{q_u}, \mathrm{q_o}, \mathrm{z}, \mathrm{c}\}$. A component y^t is said to be incompatible with the meta component x^{m}, if $y^t \in \mathcal{X}^t$ and $y^t \notin \mathcal{X}^t(x^{\mathrm{m}})$ (more compactly, $y^t \in \mathcal{X}^t \setminus \mathcal{X}^t(x^{\mathrm{m}})$). Again, a set $\mathcal{X}^t$ contains all possible components, hence it must contains these incompatibles components.

In the MLP example, a continuous component y^{c} that contains the decay λ (see Table 3.1) is incompatible with the meta component x^{m}. Indeed, if $o = \mathrm{Adam}$ and y^{c} is a continuous component that contains the decay, then $y^{\mathrm{c}} \in \mathcal{X}^{\mathrm{c}}$ and $y^{\mathrm{c}} \notin \mathcal{X}^{\mathrm{c}}(l, \mathrm{Adam})$.

Moreover, if all variables of type $t \in \{\mathrm{q}, \mathrm{s}, \mathrm{q_u}, \mathrm{q_o}, \mathrm{z}, \mathrm{c}\}$ are global variables (unaffected by the meta component), then no parametrization is necessary, such that $\mathcal{X}^t(x^{\mathrm{m}}) = \mathcal{X}^t$.

The subtlety of incompatible component when some variables are decreed is important in Definition 2 of the domain $\mathcal{X}$. It explains why the domain $\mathcal{X}$ in Definition 2 is formulated with a categorical parametrized set $\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ and a standard parametrized set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ instead of the categorical set $\mathcal{X}^{\mathrm{q}}$ and a standard set $\mathcal{X}^{\mathrm{s}}$. Indeed, for a given meta component $x^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}}$, the categorical and standard components reside in their parametrized sets, such that $x^{\mathrm{q}} \in \mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ and $x^{\mathrm{s}} \in \mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$, in order to take into account that some categorical or standard variables may be decreed by the given meta component $x^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}}$.

Moreover, the meta component x^{m} may affect the dimension $n^t(x^{\mathrm{m}})$ of the component $x^t \in \mathcal{X}^t(x^{\mathrm{m}})$. Indeed, some acting variables of type t contained in the component $x^t \in \mathcal{X}^t(x^{\mathrm{m}})$ may be decreed by the meta component x^{m}, thus the number of acting variables in this component may vary with the meta component x^{m}. In simpler terms, the dimension of the component x^t may vary with the meta component x^{m}. Hence, the dimension of the component x^t is a function $n^t : \mathcal{X}^{\mathrm{m}} \to \mathbb{N}$. Notably in the MLP example in Table 3.1, the number of hidden layers l decrees the number of units u_i in the hidden layers, which affects the number of integer variables. Thus, the dimension of the integer component $x^{\mathrm{z}} \in \mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}})$ is determined

by the meta component x^{m}.

3.3.2.1 Alternative formulation of the domain

The domain $\mathcal{X}$ formulated as Definition 2 offers little insight regarding the visualization and construction of the domain $\mathcal{X}$, especially regarding the parametrized categorical set $\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ and the parametrized standard set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$. Hence, a more visual and algorithmic formulation of the domain, based on [4, 5], is proposed :

$$\mathcal{X} = \bigcup_{x^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}}} \left(\{x^{\mathrm{m}}\} \times \bigcup_{x^{\mathrm{q}} \in \mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})} \left(\{x^{\mathrm{q}}\} \times \mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}}) \right) \right), \tag{3.6}$$

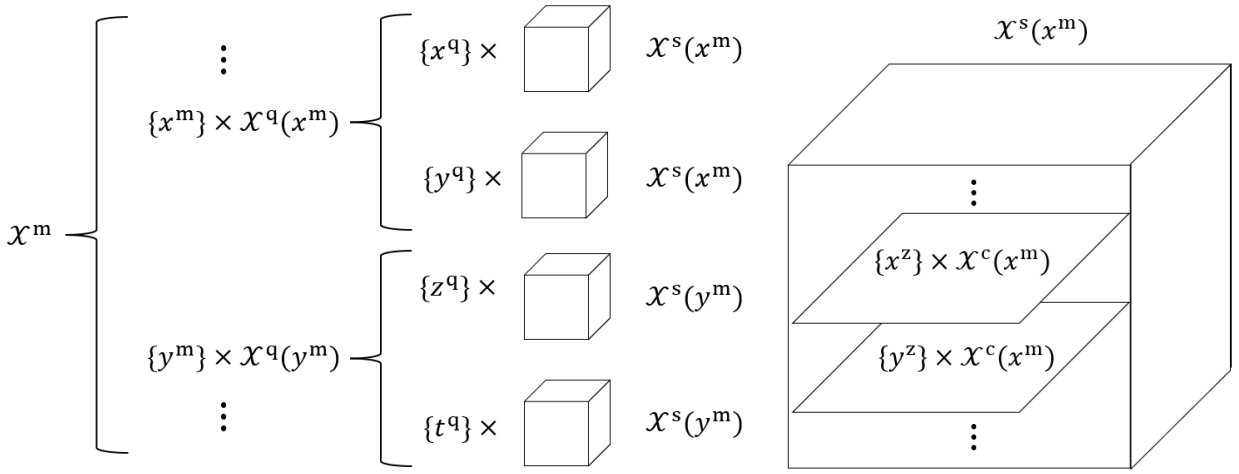
Following the same logic as in Equation (3.6), the parametrized standard set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ is formulated as :

$$\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}}) = \mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}}) \times \mathcal{X}^{\mathrm{c}}(x^{\mathrm{m}}) = \bigcup_{x^{\mathrm{z}} \in \mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}})} \left(\{x^{\mathrm{z}}\} \times \mathcal{X}^{\mathrm{c}}(x^{\mathrm{m}}) \right) \tag{3.7}$$

Schematically, the parametrized standard set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ can be visualized as the union of multiple layers, where each layer is a Cartesian product of a parametrized continuous set $\mathcal{X}^{\mathrm{c}}(x^{\mathrm{m}})$ with an integer component $x^{\mathrm{z}} \in \mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}})$, which is illustrated in Figure 3.4b. In Figure 3.4b, each layer shares the same continuous set $\mathcal{X}^{\mathrm{c}}(x^{\mathrm{m}})$, whereas each layer has a distinct integer component $x^{\mathrm{z}} \in \mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}})$. The standard set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ is represented as a box containing all the possible unions described in Equation (3.7).

A visualization of the entire domain $\mathcal{X}$ can be built upon the abstraction of the standard set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ illustrated in Figure 3.4b. In Figure 3.4a, the standard sets are represented as small boxes, following the abstraction from Figure 3.4b. The left-curly brackets represents the unions in the Equation (3.6), from left to right. Furthermore, the formulation of the domain $\mathcal{X}$ in Equation (3.6) may be understood and visualize as an explicit enumeration of all the possible points, similarly to a set of all possible components $\mathcal{X}^t$ in Equation (3.5).

The domain $\mathcal{X}$, defined in Definition 2 or alternatively formulated in Equation (3.6), is composed of the meta set $\mathcal{X}^{\mathrm{m}}$, the parametrized categorical set $\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ and the parametrized standard set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$. Hence, few details about these important sets are given in the following sections.



(a) Visualization of Equation (3.6). (b) Visualization of Equation (3.7).

FIGURE 3.4 Visualization of the domain $\mathcal{X}$.

3.3.2.2 Meta set

The number of variables in the meta component is denoted $n^{\mathrm{m}} \in \mathbb{N}$. The meta component x^{m} belongs to the meta set $\mathcal{X}^{\mathrm{m}} \subseteq \mathbb{M}^{n^{\mathrm{m}}}$, which contain all the meta component x^{m}. In comparison to a parametrized set, the meta set $\mathcal{X}^{\mathrm{m}}$ is static, since the meta variables are always acting variables. This also implies that the meta component x^{m} has a fixed dimension $n^{\mathrm{m}} \in \mathbb{N}$. Moreover, the set $\mathbb{M}^{n^{\mathrm{m}}}$ is a mixed set consisting of Cartesian products, such that

$$\mathbb{M}^{n^{\mathrm{m}}} = \mathbb{Z}^{n^{\mathrm{m_q}}} \times \mathbb{Z}^{n^{\mathrm{m_z}}} \times \mathbb{R}^{n^{\mathrm{m_c}}}, \tag{3.8}$$

where $n^{\mathrm{m}} = n^{\mathrm{m_q}} + n^{\mathrm{m_z}} + n^{\mathrm{m_c}}$ is the number of meta variables, $n^{\mathrm{m_q}}$ is the number of meta-categorical variables, $n^{\mathrm{m_z}}$ is the number of meta-integer variables and $n^{\mathrm{m_c}}$ is the number of meta-continuous variables. In particular, note that $\mathbb{M}^{n^{\mathrm{m}}} = \mathbb{Z}^{n^{\mathrm{m_q}}} \times \mathbb{Z}^{n^{\mathrm{m_z}}} = \mathbb{Z}^{n^{\mathrm{m}}}$ in the case where meta variables are strictly dimensional variables (discrete variables) [34, 35]. This case is common in practice.

3.3.2.3 Parametrized categorical set

The categories of each categorical variable can be mapped with a bijection to a subset of $\mathbb{Z}$. Hence, without any loss of generality, the parametrized categorical set $\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ is considered to be a subset of $\mathbb{Z}^{n^{\mathrm{q}}(x^{\mathrm{m}})}$. However, this bijection does not imply that a metric notion is appropriate [27]. In other words, this bijection is only useful in terms of algorithmic imple-

mentations.

Since the categorical variable x_j^{q} takes values from the set $\{1, 2, \ldots, c_j\}$, the parametrized categorical set $\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ is defined as

$$\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}}) = \prod_{j=1}^{n^{\mathrm{q}}(x^{\mathrm{m}})} \{1, 2, \ldots, c_j\}. \tag{3.9}$$

It may also be expressed as the Cartesian product between the parametrized unordered and ordered sets

$$\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}}) = \mathcal{X}^{\mathrm{q_u}}(x^{\mathrm{m}}) \times \mathcal{X}^{\mathrm{q_o}}(x^{\mathrm{m}}) = \prod_{i=1}^{n^{\mathrm{q_u}}(x^{\mathrm{m}})} \{1, 2, \ldots, c_i\} \times \prod_{i=j}^{n^{\mathrm{q_o}}(x^{\mathrm{m}})} \{1, 2, \ldots, c_j\}, \tag{3.10}$$

which outlines the distinction between nominal and ordinal variables.

3.3.2.4 Parametrized standard set

The parametrized standard set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ is a compact notation that describes a direct Cartesian product of the parametrized integer and continuous sets :

$$\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}}) \;=\; \mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}}) \times \mathcal{X}^{\mathrm{c}}(x^{\mathrm{m}}) \subseteq\; \mathbb{Z}^{n^{\mathrm{z}}(x^{\mathrm{m}})} \times \mathbb{R}^{n^{\mathrm{c}}(x^{\mathrm{m}})}, \tag{3.11}$$

where $\mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}}) \subseteq \mathbb{Z}^{n^{\mathrm{z}}(x^{\mathrm{m}})}$ is the parametrized integer set and $\mathcal{X}^{\mathrm{c}}(x^{\mathrm{m}}) \subseteq \mathbb{R}^{n^{\mathrm{c}}(x^{\mathrm{m}})}$ is the parametrized continuous set.

Again, the compact notation for the parametrized standard set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ is particularly interesting when for algorithms that optimize simultaneously the integer and continuous variables is employed.

3.3.3 Feasible set

The constraints are separated into two roles, the global and decreed constraints. Global constraints are always acting, whereas decreed constraints may be acting or nonacting depending on the meta variables. The decreed constraints lead to the following definition.

Definition 4 (Set of decreed acting constraints)**.** *The set of decreed acting constraints $C^m(x^m)$ is the set that contains all the acting constraints that are decreed by the meta component x^m.*

Similarly to a parametrized set $\mathcal{X}^t(x^{\mathrm{m}})$ defined in Definition 3, the dependency of $C^{\mathrm{m}}(x^{\mathrm{m}})$ with x^{m} is defined through a parametrization with respect to the meta component x^{m}.

Moreover, the set of decreed acting constraints $C^{\mathrm{m}}(x^{\mathrm{m}})$ is a subset of the set of decreed constraints C^{m}. A constraint $c \in C^{\mathrm{m}}$ is either acting or nonacting, whereas $\hat{c} \in C^{\mathrm{m}}(x^{\mathrm{m}})$ is an acting constraint, decreed by the meta component x^{m}. In the MLP example discussed in Section 3.2, the set of decreed constraints is

$$C^{\mathrm{m}} \;=\; \{u_i - u_{i-1} \leq 0 \;:\; \forall i \in \{2, 3, \ldots, l^{\max}\}\} \tag{3.12}$$

and the set of decreed acting constraints is

$$C^{\mathrm{m}}(x^{\mathrm{m}}) = C^{\mathrm{m}}(l, o) = \begin{cases} \emptyset, & \text{if } l \in \{0, 1\} \\ \{u_i - u_{i-1} \leq 0 \;:\; \forall i \in \{2, 3, \ldots, l\}\} \subseteq C^{\mathrm{m}}, & \text{otherwise} \end{cases} \tag{3.13}$$

where $l \leq l^{\max}$.

Moreover, some constraints are not decreed by the meta component x^{m}. These constraints are called the global constraints. In the MLP example, the global constraint is $c(x) = \sum_{i=1}^{l} u_i - \hat{u} \leq 0$, which is always acting no matter the meta component x^{m}.

To define the feasible set Ω, the global constraints and decreed constraints are distinguished.

Definition 5 (Feasible set)**.** *The feasible set $\Omega \subseteq \mathcal{X}$ is the domain $\mathcal{X}$ defined by constraints :*

$$\begin{aligned} \Omega = \Big\{ \; (x^m, x^q, x^s) \in \mathcal{X} \quad : \quad & c_i(x) \leq 0, \;\; \forall i \in \{1, 2, \ldots, p\}, \\ & c^m(x) \leq 0, \;\; \forall c^m \in C^m(x^m) \; \Big\} \end{aligned} \tag{3.14}$$

where c_i are the global constraints with $p \in \mathbb{N}$ and $C^m(x^m)$ is the set of decreed acting constraints, which parametrized with respect to meta component x^m. The number of acting constraints that are decreed by the meta component x^m is simply $|C^m(x^m)|$.

3.3.4 Mathematical modeling of the MLP example

Each hyperparameter is identified with its variable type and role in Table 3.2.

The following observations can be made. First, the number of units u_i in the hidden layers are typed as integer variables. Although they affect the network architecture, they are not meta variables because they do not decree other variables. More precisely, they do not affect the dimension of the integer component, since they do not decree any other hyperparameters. Second, the number of hidden layers l is a meta variable, since it decrees the units u_i and

Table 3.2 Hyperparameters with their variable type and role.

Hyperparameter	Variable	Scope	Type	Role
Learning rate	r	$]0,1[$	continuous	global
Activation function	a	{ReLU, Sigmoid}	categorical	global
# of hidden layers	l	$\{0,1,\ldots,l^{\max}\}$	meta	meta
# of units hidden layer i	u_i	$\{u_i^{\min}, u_i^{\min}+1,\ldots,u_i^{\max}\}$	integer	decreed
Optimizer	o	{Adam, ASGD}	meta	meta
if o = ASGD				
decay	λ	$]0,1[$	continuous	decreed
power update	α	$]0,1[$	continuous	decreed
averaging start	t_0	$]1E3,1E8[$	continuous	decreed
if o = Adam				
running average 1	β_1	$]0,1[$	continuous	decreed
running average 2	β_2	$]0,1[$	continuous	decreed
numerical stability	ϵ	$]0,1[$	continuous	decreed

thus it affects the dimension of a component $x^{\mathrm{z}} \in \mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}})$. Third, the activation function $a \in \{\mathrm{ReLu}, \mathrm{Sigmoid}\}$ is an unordered variable, since it is a qualitative discrete variable that belongs to a set with no appropriate metric and no order. Fourthly, the optimizer is a meta variable. Indeed, the choice of the optimizer o decrees some continuous hyperparameters of the problem.

3.3.4.1 Components and sets

The meta set $\mathcal{X}^{\mathrm{m}}$ is the Cartesian product between the scopes of the two meta variables, the number of hidden layers l and the optimizer o, thus the meta component x^{m} and the meta set $\mathcal{X}^{\mathrm{m}}$ are :

$$x^{\mathrm{m}} = (l, o) \in \mathcal{X}^{\mathrm{m}} = \{0,1,\ldots,l^{\max}\} \times \{\text{Adam, ASGD}\}. \tag{3.15}$$

Then, the only categorical variable is the activation function a, which is a global variable. Thus, $\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}}) = \mathcal{X}^{\mathrm{q}}$ in the example, since no parametrization of the categorical set is necessary. Following this, the categorical component x^{q} and the categorical set $\mathcal{X}^{\mathrm{q}}$ are :

$$x^{\mathrm{q}} = a \in \mathcal{X}^{\mathrm{q}} = \{\text{ReLU, Sigmoid}\}. \tag{3.16}$$

Moreover, the integer component is directly the vector of units in the hidden layers, such that $x^{\mathrm{z}} = u(l) = (u_1, u_2, \ldots, u_l)$. All the integer variables are decreed by the meta component x^{m}

and more specifically the number of hidden layers l. The integer component x^{z} and the parametrized integer set $\mathcal{X}^{\mathrm{z}}$ are :

$$x^{\mathrm{z}} = \begin{cases} \emptyset \text{ (nonacting)}, & \text{if } l = 0 \\ (u_1, u_2, \ldots, u_l) \in \mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}}) = \mathcal{X}^{\mathrm{z}}(l) = \prod_{i=1}^{l} \{u_i^{\min}, u_i^{\min} + 1, \ldots, u_i^{\max}\} \subseteq \mathbb{N}^l, & \text{if } l \geq 1 \end{cases} \tag{3.17}$$

where $u_i^{\min}$ and $u_i^{\max}$ are respectively the minimum and the maximum of units allowed for each hidden layer $i \in \{1, 2, \ldots, l\}$, l is the number of hidden layers and $u(0)$ is an empty vector.

Finally, all continuous variables are decreed by the optimizer o, except for the learning rate r. Thus, the continuous component x^{c} is decreed by the meta component x^{m}, implying that the continuous set requires a parametrization. The continuous component x^{c} and the parametrized continuous set $\mathcal{X}^{\mathrm{c}}(x^{\mathrm{m}})$ are :

$$x^{\mathrm{c}} \in \mathcal{X}^{\mathrm{c}}(x^{\mathrm{m}}) = \begin{cases} \mathcal{X}^{\mathrm{c}}(\text{Adam}) =]0, 1[^4 \times \subseteq \mathbb{R}^4, & \text{if } o = \text{Adam} \\ \mathcal{X}^{\mathrm{c}}(\text{ASGD}) =]0, 1[^3 \times]1\text{E}3, 1\text{E}8[\subseteq \mathbb{R}^4, & \text{if } o = \text{ASGD} \ . \end{cases} \tag{3.18}$$

For the sake of simplicity, the scope of the units in the hidden layers u_i and the number of hidden layer l are set as : $u_i^{\min} = 100$ and $u_i^{\max} = 300, \ \forall i$ and $l \in \{2, 3\}$ in Table 3.1. With $l \in \{2, 3\}$, the meta set (3.15) can be explicit as :

$$\mathcal{X}^{\mathrm{m}} = \{(\text{Adam}, 2), (\text{Adam}, 3), (\text{ASGD}, 2), (\text{ASGD}, 3)\}. \tag{3.19}$$

Moreover, the parametrized integer set $\mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}})$ can also be explicit :

$$\mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}}) = \mathcal{X}^{\mathrm{z}}(l) = \{100, 101, \ldots, 300\}^l = \begin{cases} \{100, 101, \ldots, 300\}^2, & \text{if } l = 2 \\ \{100, 101, \ldots, 300\}^3, & \text{if } l = 3. \end{cases} \tag{3.20}$$

The parametrized categorical and continuous sets remain unchanged.

3.3.4.2 Constraints

In the example there is a global constraint and decreed constraints. The global constraint can be easily expressed as $c(x) \leq 0$ where

$$c(x) = c(l) = \sum_{i=1}^{l} u_i - \hat{u} = \begin{cases} u_1 + u_2 - \hat{u}, & \text{if } l = 2 \\ u_1 + u_2 + u_3 - \hat{u}, & \text{if } l = 3 \end{cases} \tag{3.21}$$

The set of decreed constraint is

$$C^{\mathrm{m}} = \left\{ u_i - u_{i-1} \leq 0 \ : \ i \in \{2, 3, \ldots, l^{\max}\} \ \right\} \tag{3.22}$$

and the set of acting decreed constraints is

$$C^{\mathrm{m}}(x^{\mathrm{m}}) = C^{\mathrm{m}}(l) = \begin{cases} \emptyset \text{ (nonacting)}, & \text{if } l < 2, \\ \left\{ u_i - u_{i-1} \leq 0 : \ i \in \{2, 3, \ldots, l\}, \ \right\}, & \text{if } l \geq 2, \end{cases} \tag{3.23}$$

which can be further detailed since $l \in \{2, 3\}$

$$C^{\mathrm{m}}(2) = \{u_2 - u_1 \leq 0\}, \quad C^{\mathrm{m}}(3) = \{u_3 - u_2 \leq 0, \ u_2 - u_1 \leq 0\}. \tag{3.24}$$

In this particular example, the number of global constraint is $p = 1$ and the number of acting constraints that decreed by the meta component is $|C(x^{\mathrm{m}})| = l - 1$.

3.3.4.3 Visualization of the domain and the feasible set

The alternative formulation of the domain $\mathcal{X}$ in Equation (3.6) and the feasible set Ω in Definition 3.14 of the MLP example can be visualized in Figure 3.5.

The upper part of Figure 3.5 (above the dotted line) represents the alternative formulation of the domain $\mathcal{X}$ in Equation (3.6). The parametrized standard sets $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ are illustrated as small boxes and the unions from left to right in Equation (3.6) are viewed from top to bottom in Figure 3.5. Moreover, the parametrized standard sets are expressed explicitly, such that $X^{\mathrm{s}}(x^{\mathrm{m}}) = X^{\mathrm{s}}(l, o) = \mathcal{X}^{\mathrm{z}}(l) \times \mathcal{X}^{\mathrm{c}}(o)$. The lower part of Figure 3.5 schematizes the constraints. The acting constraints decreed by a meta component x^{m}, are contained in the set of acting decreed constraints $C^{\mathrm{m}}(x^{\mathrm{m}})$. The global constraint is always acting and unaffected by the meta component x^{m}, hence it is not assign to a specific meta component x^{m} comparatively to decreed constraints : this representation shows the global aspect of global constraints.

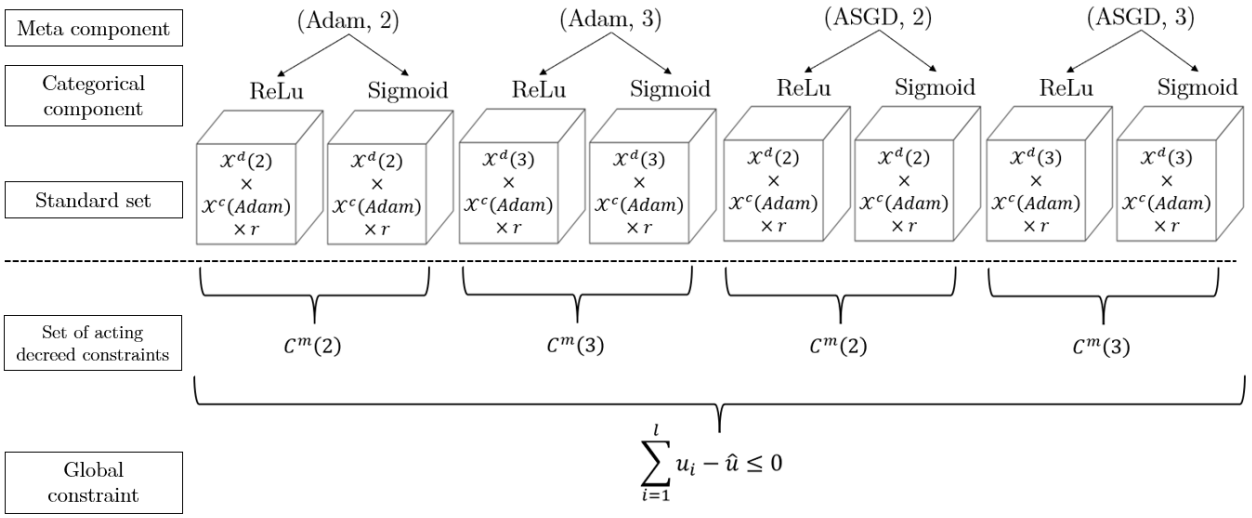


FIGURE 3.5 Diagram of the domain $\mathcal{X}$ and the constraints for the MLP example.

Altogether, the upper and lower parts Figure 3.5 synthesize the feasible set Ω of the MLP example.

In the literature review, it has been discussed that some optimization approaches tackle categorical variables by solving many subproblems in which a categorical component x^{q} is fixed. Indeed, in [5, 10] the MADS algorithm was applied to a continuous space where a discrete component, which contained meta, categorical and integer variables, was fixed. This idea can be generalized to the proposed notation system. For example, assume that $x^{\mathrm{m}} = (\mathrm{Adam}, 2)$ and $x^{\mathrm{q}} = \mathrm{ReLU}$ are selected and fixed. Then, the objective function f could then be optimized on the parametrized standard $\mathcal{X}^{\mathrm{s}}(\mathrm{Adam}, 2)$ with both the meta and categorical components fixed. Subproblems are further discussed in the next Section 3.4 and more particularly in Section 3.4.1.

3.4 Solution strategies

Most blackbox approaches in mixed-variable optimization are built upon two strategies. One solution strategy consists of solving many subproblems in which some selected components are fixed. Another strategy consists of formulating a less costly problem that selects a candidate point to be evaluated by the more costly objective function f. Some methods rely on both strategies.

For example, direct search methods [5, 9, 10, 30] divide the main problem into many subproblems, in which the objective function f is optimized on a continuous space for a fixed discrete component x^d. Bayesian optimization (BO) formulates an auxiliary problem, with a

fixed acquisition function and a probabilistic surrogate, and then selects a candidate point that is subsequently evaluated by the objective function f. The methodology proposed in [39] formulates many auxiliary subproblems, where each problem has a fixed dimensional component [34, 35] and each subproblem has its own surrogate.

The two strategies are respectively defined as the subproblems strategy and the auxiliary problem strategy. These strategies are the basis of the general algorithmic framework, since most algorithms that tackle mixed-variable blackbox optimization conceptually rely on solving many subproblems or on an auxiliary problem.

The purpose of this section is to illustrate that the framework notation may be easily adapted to the main blackbox approaches in mixed-variable optimization. More precisely, direct search and heuristic approaches are discussed through the subproblems strategy in Section 3.4.1 and the BO approach is discussed through the auxiliary problem strategy in Section 3.4.2.

3.4.1 Subproblems

The motivation of dividing a main problem into many subproblems arises from two rationales : 1) there are methods that treat standard problems, or even categorical-standard problems (mostly with an auxiliary problem strategy) ; 2) there are few efficient methods that address mixed-variable optimization problems with both meta (or dimensional) and categorical variables.

In the context of this work, subproblems are obtained by fixing values of meta and categorical components. In [5, 9, 10, 30], the component that is fixed is the discrete component, which contains categorical variables. Secondly, note that there's no particular interest fixing the integer or continuous components, since they can be properly optimized in practice.

To further formalize the subproblems, the objective subfunction must be first defined.

Definition 6 (Objective subfunction)**.** *An objective subfunction g is the objective function f with a single or many fixed components. The objective subfunction is said to be parametrized with respect to the fixed component(s).*

From Definition 6, it should be noted that there is a direct correspondence between the fixed component(s) and its subproblem. In other words, a specific subproblem may be referred by its fixed component(s). Again, in Definition 6, the components that are interesting to fix are the meta component x^{m} and the categorical component x^{q}. In this work, only the standard subproblems strategy, in which both the meta and categorical components, is detailed. Remember that term standard encapsulates integer and continuous.

3.4.1.1 Standard subproblems

In the standard subproblems strategy, the meta component x^{m} and the categorical component x^{q} are fixed, in order to generate standard subproblems (one per couple $(x^{\mathrm{m}}, x^{\mathrm{q}})$). Fixing a meta component $x^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}}$ simplifies the optimization problem, since the acting variables, the acting constraints and the dimension in the subproblems are determined. In addition, fixing the categorical components also further simplifies the optimization problem. Indeed, with both the meta and categorical components fixed, the subproblems are a standard blackbox optimization problem, where the acting variables are either integer or continuous variables. In practice, there are efficient methods to tackle these standard subproblems.

For the standard subproblems strategy, the objective subfunction $g : \mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}}) \to \mathbb{R}$, parametrized with respect to the meta component $x^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}}$ and the categorical component $x^{\mathrm{q}} \in \mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$, is defined as :

$$g(x^{\mathrm{s}}; x^{\mathrm{q}}, x^{\mathrm{m}}) = f(x^{\mathrm{m}}, x^{\mathrm{q}}, x^{\mathrm{s}}), \quad \text{where } x^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}} \text{ and } x^{\mathrm{q}} \in \mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}}) \text{ are fixed.} \tag{3.25}$$

Thus, for a fixed meta component $x^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}}$ and a fixed categorical component $x^{\mathrm{q}} \in \mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$, a standard subproblem may be formulated as

$$\begin{array}{lll} (P^{\mathrm{s}}) \quad \min\limits_{x^{\mathrm{s}} \in \mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})} & g(x^{\mathrm{s}}; x^{\mathrm{m}}, x^{\mathrm{q}}) & \\ \text{s.t.} & c^{\mathrm{m}}(x) \leq 0, & \forall c^{\mathrm{m}} \in C^{\mathrm{m}}(x^{\mathrm{m}}), \\ & c_i(x) \leq 0, & \forall i \in \{1, 2, \ldots, p\}. \end{array} \tag{3.26}$$

where P^{s} stands for standard subproblem. Moreover, note that the constraints of the problem are treated directly within the subproblems of the form (P^{s}).

3.4.1.2 Exploration of subproblems

There is a direct correspondence between the fixed component(s) and their subproblem, hence the exploration of subproblems may be done accordingly to the fixed components. Solving subproblems may be done directly with simple heuristics, such as random searches on the meta and categorical components. However, extra work is required in a direct search framework. Qualitative variables, such as the categorical variables, do not posses intuitive neighborhoods nor directions of exploration. Hence, the meta set $\mathcal{X}^{\mathrm{m}}$, which may contain meta components with meta categorical variables, and the parametrized categorical set $\mathcal{X}^{\mathrm{q}}$ are both endowed with a user-defined neighborhood mapping. To formalize the exploration

of subproblems, the following definition based on [5, 9, 10, 30], is proposed.

Definition 7 (User-defined neighborhood mapping)**.** *For any $t \in \{m, q\}$, a user-defined neighborhood mapping $\mathcal{N}^t$ assigns a user-defined neighborhood $\mathcal{N}^t(x) \subseteq \mathcal{X}^t$ to a point $x \in \mathcal{X}$, such that each neighbor $y^t \in \mathcal{N}^t(x)$ is a component of type t that is determined by a given rule $r^t : \mathcal{X} \to \mathcal{X}^t$:*

$$\begin{aligned} \mathcal{N}^t \ : \ \mathcal{X} &\to \mathcal{P}(\mathcal{X}^t) \\ x &\mapsto \ \Big\{ \ y^t \in \mathcal{X}^t \ : \ y^t = r^t(x), \ r^t \in \mathcal{R}^t(x) \Big\} \subseteq \mathcal{X}^t \end{aligned} \tag{3.27}$$

where $r^t \in \mathcal{R}^t(x)$ is a rule that assigns a neighbor $y^t = r^t(x) \in \mathcal{X}^t$ to a point $x \in \mathcal{X}$, $\mathcal{R}^t(x)$ is a set of rules defined for the given point $x \in \mathcal{X}$ and $\mathcal{P}(\mathcal{X}^t)$ is the powerset of $\mathcal{X}^t$, which is denoted as the codomain of the mapping $\mathcal{N}^t$ to indicate $\mathcal{N}^t(x)$ can either be :

1. *$\mathcal{N}^t(x) = \emptyset$, such that x has no neighbor of type t ;*
2. *$\mathcal{N}^t(x) = \{y^t\}$, such that x has a single neighbor of type t ;*
3. *$\mathcal{N}^t(x) \subseteq \mathcal{X}^t$, such that x^t has multiple neighbors of type t.*

The set of rules $\mathcal{R}^t(x)$ embeds the generality of the user-defined neighborhood $\mathcal{N}^t(x)$. Indeed, a rule $r^t \in \mathcal{R}^t(x)$ must only respect the following mapping $r : \mathcal{X} \to \mathcal{X}^t$, which indicates that a component $y^t = r^t(x) \in \mathcal{X}^t$, called a neighbor, is assigned to a point $x \in \mathcal{X}$. In practice, it is from the these rules that user-defined neighborhoods are generated and implemented. Moreover, two issues are specific to the categorical case $t = \mathrm{q} : 1)$ the set $\mathcal{X}^{\mathrm{q}}$ is the parametrized categorical set : $\mathcal{X}^t = \mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$; 2) the user-defined neighborhood mapping $\mathcal{N}^{\mathrm{q}}$ takes a point $x \in \mathcal{X}$ as an argument, which allows to take into account the decree property of meta variables for the user-defined neighborhood mapping $\mathcal{N}^{\mathrm{q}}$ and its constituent parts, such as the rules r^{q}.

In the MLP example and using Equation (3.15), the meta rules of the form $r^{\mathrm{m}} : \mathcal{X} \to \mathcal{X}^{\mathrm{m}}$, for a given point $y = (y^{\mathrm{m}}, x^{\mathrm{q}}, x^{\mathrm{s}}) \in \mathcal{X}$ with $y^{\mathrm{m}} = (l, o)$, could be

$$\begin{aligned} r_1^{\mathrm{m}}(y) = (l+1, o), \quad r_2^{\mathrm{m}}(y) = (l-1, o), \\ r_3^{\mathrm{m}}(y) = (l, \bar{o}), \quad r_4^{\mathrm{m}}(y) = (l+1, \bar{o}), \quad r_5^{\mathrm{m}}(y) = (l-1, \bar{o}), \end{aligned}$$

where $\bar{o}$ represents the other optimizer available. The set of rules would be :

$$\mathcal{R}^{\mathrm{m}}(y) = \begin{cases} \left\{r_1^{\mathrm{m}}, r_3^{\mathrm{m}}, r_4^{\mathrm{m}}\right\}, & \text{if } l = 0 \\ \left\{r_2^{\mathrm{m}}, r_3^{\mathrm{m}}, r_5^{\mathrm{m}}\right\}, & \text{if } l = l^{\max} \\ \left\{r_1^{\mathrm{m}}, r_2^{\mathrm{m}}, r_3^{\mathrm{m}}, r_4^{\mathrm{m}}, r_5^{\mathrm{m}}\right\}, & \text{otherwise,} \end{cases} \tag{3.28}$$

with corresponding user-defined neighborhood

$$\mathcal{N}^{\mathrm{m}}(y) = \begin{cases} \left\{(l+1, o), (l, \bar{o}), (l+1, \bar{o})\right\}, & \text{if } l = 0 \\ \left\{(l-1, o), (l, \bar{o}), (l-1, \bar{o})\right\}, & \text{if } l = l^{\max} \\ \left\{(l+1, o), (l-1, o), (l, \bar{o}), (l+1, \bar{o}), (l-1, \bar{o})\right\}, & \text{otherwise.} \end{cases} \tag{3.29}$$

The evaluations of the blackbox objective function f are generally costly, which implies that the user-defined neighborhood mappings have to set a trade-off between being exploratory and computationally expensive. Again, in practice, the user-defined neighborhood mappings $\mathcal{N}^{\mathrm{m}}$ and $\mathcal{N}^{\mathrm{q}}$ are based on rules provided by a user. Thus, the compromise is set with the discretion of the user. To lower the number of evaluations, some polling strategies may be used in practice. Indeed, instead of exploring all the neighbors at a given iteration, an opportunistic strategy would stop the iteration if a neighbor that offers a better solution is determined and resume from that neighbor.

3.4.1.3 Direct search framework

Direct search methods with strict decrease are iterative algorithms that start with an initial point $x_{(0)}$ and seek a candidate point t whose objective function value $f(t)$ is strictly less than $f(x_{(k)})$, where $x_{(k)}$ is the current incumbent solution at iteration k. More precisely, at every iteration k, a set of trial points T is generated. Opportunistically, if a trial point $t \in T$ improves the objective function value, then it becomes the next incumbent solution $x_{(k+1)} = t$ and the iteration k terminates. Otherwise, the current incumbent solution remains unchanged, such that $x_{(k+1)} = x_{(k)}$ [12, 14]. In practice, stopping the iteration opportunistically reduces the number of evaluations required [12].

Moreover, direct search methods tackle blackbox optimization problems with two main mechanism : a global search strategy (diversification) and a poll that locally searches better solutions (intensification).

By its own, a poll is prone to miss out good point solutions. Indeed, the poll may get caught

in a region with local minima or may neglect the exploration of promising regions that are far from the poll. For the meta set $\mathcal{X}^{\mathrm{m}}$ and parametrized categorical set $\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ the poll may be emulated with some user-defined neighborhood mapppings $\mathcal{N}^{\mathrm{m}}$ and $\mathcal{N}^{\mathrm{q}}$ respectively. The quality of a poll based on a user-defined neighborhood mapping, such as the meta and categorical polling, depends on the exhaustiveness of the set of rules $\mathcal{R}^{\mathrm{m}}$ and $\mathcal{R}^{\mathrm{q}}$. Therefore, depending on the quality of implementation by the user and the dimensions of the problem, the poll, based on user-defined neighborhood mappings, is likely to neglect some promising components.

In that regard, a global search may help overcome this problem by evaluating scattered trial points (or components) with a flexible strategy that serves as a diversification mechanism. The global search is generally being done before the poll for opportunistic reasons, given that the global search may find a better or interesting point that deserves to be further explored with the poll. The global search is an optional step that often improves the overall quality of a solution and increases the convergence speed. Many generic and low-cost global search strategies exist, such as the random search, Latin hypercube sampling or a Nelder-Mead search [15], and more sophisticated and costly global search strategies can be implemented to generate promising trial points or unexplored regions, such as the Gaussian Processes (surrogate) paired with an acquisition function (auxiliary problem strategy) that quantifies the uncertainty and the potentiality of a point.

Algorithm 1 presents the main steps of a direct search methodology. The methodology consists of a standard subproblems strategy (see Section 3.4.1.1) paired with an exploration of subproblems that is done with user-defined neighborhood mappings $\mathcal{N}^{\mathrm{m}}$ and $\mathcal{N}^{\mathrm{q}}$ from Definition 3.27.

In Algorithm 1, the two main steps to tackle the meta and categorical variables with a direct search approach are compactly presented. For the global search and poll steps, a standard subproblem (P^{s}), which respects the formulation in Problem (3.32), is solved. Hence, the constraints of the problem are handled within the subproblems. Moreover, the solving of a subproblem (P^{s}) encapsulates many algorithmic details, such as a stopping criteria for a subproblem, as well as a global search and poll on the integer and continuous (standard) variables. Note that, a potential solver for the subproblems could be the MADS algorithm [10] which enables to treat simultaneously integer and continuous variables (standard problem). For more details, see [14]. Then, additionally, constraints can be handled with the progressive barrier technique [11].

Algorithm 1: Direct search main steps.

while *stopping criteria not reached* **do**
  1. **Global search**
    Select $t^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}}$ with a global meta exploration strategy
    Select $t^{\mathrm{q}} \in \mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ with a global categorical exploration strategy
    Let t be obtained by solving the subproblem (P^{s}) with t^{m} and t^{q} fixed
  if $f(t) < f\left(x_{(k)}\right)$ **then**
    $x_{(k+1)} \leftarrow t$
  else
    2. **Poll on user-defined neighborhoods**
    for $t^m \in \mathcal{N}^m\left(x^m_{(k)}\right)$ **do**
      for $t^q \in \mathcal{N}^q\left(x^q_{(k)}; t^m\right)$ **do**
        Let t be obtained by solving the subproblem (P^{s}) with t^{m} and t^{q} fixed
        if $f(t) < f\left(x_{(k)}\right)$ **then**
          $x_{(k+1)} \leftarrow t$
          `break` # Opportunistic strategy
        end
      end
    end
end

3.4.2 Auxiliary problem

Auxiliary problems inexpensively allow to select candidate points to be evaluated by the true objective function f. Auxiliary problems are generally built from a surrogate model $\tilde{f}$ of the objective function f, an acquisition function α, as well as surrogates of each global constraint $\tilde{c}_j,\ j \in \{1, 2, \ldots, p\}$ and decreed constraint $\tilde{c}^{\mathrm{m}} \in C^{\mathrm{m}}$. The acquisition function α allows to select candidate points in promising regions (intensification) or in unexplored regions (exploration). The acquisition function α is generally applied to a surrogate model $\tilde{f}$ that quantifies the uncertainty of a point of its domain, and provides a prediction of the true objective function f. This is the case in BO where $\tilde{f}$ is a GP probabilistic surrogate model. Other surrogate models can be considered, such as random forests, however the most common remains the GPs. In this section, only GP surrogate models are adapted to the notation framework, since they are the basis of BO, an important blackbox approach to tackle mixed-variable problems. Before discussing BO, the encoding of variables is discussed.

3.4.2.1 Encoding of variables and auxiliary domain

BO methodologies (from Section 3.1.2) often tackle categorical variables by encoding them as quantitative variables. For instance, the categorical variables may be encoded by the emerging latent variables or simply with the popular one-hot encoding binary vectors relaxed into a continuous vector [25].

Definition 8 (Encoder)**.** *For any $t \in \{q, q_u, q_o\}$ and iteration $k \in \mathbb{N}$, the encoder $\phi^t_{(k)}$, parametrized with respect to the meta component $x^m \in \mathcal{X}^m$, is a mapping that assigns an encoded component e^t to a component x^t, such that*

$$\begin{array}{rccl} \phi^t_{(k)} & : & \mathcal{X}^t(x^m) & \to & \mathcal{E}^t(x^m) \\ & & x^t & \mapsto & e^t = \phi^t_{(k)}(x^t; x^m). \end{array} \tag{3.30}$$

An encoder $\phi^t_{(k)}$ may be updated at every iteration $k \in \mathbb{N}$, such as the latent variables discussed in Section 3.1.2. In order to take into account the decree properties of the meta variables, an encoder is parametrized with respect to the meta component $x^{\mathrm{m}} \in \mathcal{X}^{\mathrm{m}}$. In general, a meta variable may be a meta categorical variable. However, in this work, the meta variables are not encoded for two reasons. First, the decreeing property of encoded meta variables may be ambiguous and difficult to conserve through sophisticated mappings, such as the latent variables. Secondly, there are categorical kernels that allow to avoid encoding categorical variables, hence in a BO framework, meta categorical variables may be treated with these kernels.

One of the main purpose of encoding categorical variables (or equivalently categorical component) is to formulate an auxiliary problem in which these encoded variables possess mathematical properties, making them easier to manipulate. However, by encoding the categorical variables, the domain of the surrogate model may differ from the domain of the objective function $\mathcal{X}$. Hence, the auxiliary domain $\mathcal{X}_{\mathrm{aux}}$ is defined as follows.

Definition 9 (Auxiliary domain)**.** *The auxiliary domain at an iteration $k \in \mathbb{N}$ is defined by :*

$$\begin{array}{rcl} \mathcal{X}_{\mathrm{aux}} = \Big\{ \quad (x^m, l^q, x^s) & : & x^m \in \mathcal{X}^m, \\ & & e^q \in \mathcal{E}^q(x^m), \\ & & x^s \in \mathcal{X}^s(x^m) \quad \Big\} \end{array} \tag{3.31}$$

where $e^q = \phi^q_{(k)}(x^q; x^m)$ and $\mathcal{E}^q(x^m)$ is the encoded parametrized categorical set.

Definition 9 allows to set $\mathcal{E}^{\mathrm{q}}(x^{\mathrm{m}}) = \mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$, so that no encoding is done : $e^{\mathrm{q}} = x^{\mathrm{q}}$. In addition, since some categorical kernels do not require encoding, it follows that the auxiliary

domain $\mathcal{X}_{\text{aux}}$ is compatible with encoded categorical variables or with the original categorical variables.

From Definition 9, the auxiliary maximization problem may be formulated as :

$$\begin{array}{lll} (P^{\text{aux}}) \quad \max\limits_{x \in \mathcal{X}_{\text{aux}}} & \alpha\left(x; \tilde{f}\right) & \\ \text{s.t.} & \tilde{c}_i(x) \leq 0, & \forall i \in \{1, 2, \ldots, p\} \\ & \tilde{c}^{\text{m}}(x) \leq 0, & \forall \tilde{c}^{\text{m}} \in \tilde{C}^{\text{m}}(x^{\text{m}}), \\ & x^{\text{q}} = \phi^{\text{q}}_{(k)}(x^{\text{q}}; x^{\text{m}}) & \text{for some } x^{\text{q}} \in \mathcal{X}^{\text{q}}(x^{\text{m}}), \end{array} \tag{3.32}$$

where (P^{aux}) stands for auxiliary problem, $\alpha : \mathcal{X}_{\text{aux}} \to \mathbb{R}$ is an acquisition function applied to a surrogate model $\tilde{f}$, $\tilde{c}_i$ $\forall i \in \{1, 2, \ldots, p\}$ are surrogate constraints for the global constraints, $\tilde{c}^{\text{m}} \in \tilde{C}^{\text{m}}$ is a surrogate constraint for a decreed constraint. The last constraint imposes the existence of some $x^{\text{q}} \in \mathcal{X}^{\text{q}}(x^{\text{m}})$ such that $x^{\text{q}} = \phi^{\text{q}}_{(k)}(x^{\text{q}}; x^{\text{m}})$ is a pre-image constraint that recovers a categorical component $x^{\text{q}} \in \mathcal{X}^{\text{q}}(x^{\text{m}})$ from the encoded parametrized categorical set $\mathcal{E}^{\text{q}}(x^{\text{m}})$. The pre-image constraint also ensures that the optimal auxiliary problem solution resides in the domain $\mathcal{X}$. For more details on pre-images problem, refer to [22].

3.4.2.2 Bayesian optimization

In this section, the BO approach is formulated as an auxiliary problem (P^{aux}), without detailing the algorithmic steps or the construction of the GP (see [40] or [6] for more details on this subject). For the purpose of this work, it is sufficient to formulate the BO approach as an auxiliary problem (P^{aux}) and to develop the kernel from the notation framework, since the kernel almost entirely characterizes the probabilistic surrogate (GP). A kernel $k : \mathcal{X}_{\text{aux}} \times \mathcal{X}_{\text{aux}} \to \mathbb{R}$ is a positive semi-definite covariance function. Conceptually, the kernel establishes the mathematical properties of the GP, such as the degree smoothness.

In its simplest noise free form, a probabilistic BO distribution is built from a GP, which allows to compute for any given point $x \in \mathcal{X}_{\text{aux}}$, a prediction $\hat{f}(x)$ and an uncertainty measure $\hat{\sigma}^2(x)$, such that

$$\begin{cases} \hat{f}(x) &= \kappa^{\intercal}(x) K^{-1} f(\mathbb{X}) \\ \hat{\sigma}(x)^2 &= k(x, x) - \kappa^{\intercal}(x) K^{-1} \kappa(x) \end{cases} \tag{3.33}$$

where $\mathbb{X}$ is a set of sample points, $f(\mathbb{X})$ is the vector of objective function values of the sample points, $\kappa(x)$ is a vector in which an element is the computed kernel $k(x, y)$ with $(x, y) \in \mathcal{X}_{\text{aux}} \times \mathbb{X}$, K is matrix in containing all pairs $(y, z) \in \mathbb{X} \times \mathbb{X}$, such that an element of

K is $k(y, z)$. In Equation (3.33), everything is computed from the kernel k. In other words, Equation (3.33) displays that the GP is entirely characterized by the kernel : it is assumed that the GP is noise free and that the mean function is zero, which is a common practice [40]. The surrogate probabilistic model $\tilde{f}$ satisfies

$$\tilde{f}(x) \sim \mathcal{N}\left(\hat{f}(x), \hat{\sigma}(x)^2\right). \tag{3.34}$$

where $\mathcal{N}$ is the normal distribution. Moreover, a common acquisition function α applied on GP surrogates is the EI from [29] :

$$EI\left(x; \tilde{f}\right) = \mathbb{E}\left[\max(f_\star - \tilde{f}(x), 0)\right] = \left(f_\star - \hat{f}(x)\right)\Phi\left(\frac{f_\star - \hat{f}(x)}{\hat{\sigma}(x)}\right) + \hat{\sigma}(x)\phi\left(\frac{f_\star - \hat{f}(x)}{\hat{\sigma}(x)}\right) \tag{3.35}$$

where $f_\star = f(x_k)$ is current best known objective function value at iteration $k > 1$, $\hat{\sigma}(x)$ is the standard deviation of the GP, Φ and ϕ are respectively the cumulative distribution and the density function of a standard normal distribution (centered at zero with variance of one). In Equation (3.35), the intensification and exploration trade-off of the EI (acquisition function) is displayed by the two terms : the first term favors promising low surrogate values (intensification) and the second term favors highly uncertain points (exploration). In the auxiliary problem (P^{aux}), the acquisition function could be $\alpha(x; \tilde{f}) = EI(x; \tilde{f})$.

In a similar manner to surrogate model $\tilde{f}$ evaluated at a point $x \in \mathcal{X}_{\text{aux}}$ in Equation (3.34), the surrogate constraints in the auxiliary problem (P^{aux}), may be developed into GP probabilistic surrogates. Thus, a given surrogate constraint $\tilde{c}_i$ would have its own prediction function $\hat{c}_i$ (similarly to $\hat{f}(x)$ in Equation (3.33)), which could be directly used in the auxiliary problem (P^{aux}), i.e., $\tilde{c}_i(x) = \hat{c}_i(x)$. Acquisition functions may also be applied to probabilistic surrogate constraints, which is not covered in this work.

At this stage, the BO framework is formulated in a general manner, which does not explicit the mixed-nature of the optimization problems at stake. To adapt the BO framework on a mixed-variable context, the kernel k, must be further detailed with the support of the notation framework. Many possible kernels can be built with operations of multiplication and additions that respects the RKHS formalism [20, 39]. An example of a specific kernel is detailed next to illustrate the compatibility of the framework with the mixed-variable optimization BO literature.

The kernel k is built piece-by-piece with the partition of a point $x = (x^{\text{m}}, x^{\text{q}_\text{u}}, x^{\text{q}_\text{o}}, x^{\text{z}}, x^{\text{c}})$. The parametrized continuous kernel $k^{\text{c}} : \mathcal{X}^{\text{c}}(x^{\text{m}}) \times \mathcal{X}^{\text{c}}(x^{\text{m}}) \to \mathbb{R}$ is formulated as multiplication

of one-dimensional squared-exponential kernels :

$$k^{\mathrm{c}}(x^{\mathrm{c}}, y^{\mathrm{c}}; x^{\mathrm{m}}) = \exp\left(- \sum_{i=1}^{n^{\mathrm{c}}(x^{\mathrm{m}})} \lambda_i^{\mathrm{c}} \left[x_i^{\mathrm{c}} - y_i^{\mathrm{c}} \right]^2 \right). \tag{3.36}$$

where the λ_i^{c} are weight coefficients (hyperparameters of the surrogate model) that can be adjusted by various methods, such as the MLE.

The parametrized integer kernel $k^{\mathrm{z}} : \mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}}) \times \mathcal{X}^{\mathrm{z}}(x^{\mathrm{m}}) \to \mathbb{R}$ is similar to k^{c}, but applies a transformation T that rounds the relaxed integer variables to the nearest integer [25] :

$$k^{\mathrm{z}}(x^{\mathrm{z}}, y^{\mathrm{z}}; x^{\mathrm{m}}) = \exp\left(- \sum_{i=1}^{n^{\mathrm{z}}(x^{\mathrm{m}})} \lambda_i^{\mathrm{z}} \left[T\left(x_i^{\mathrm{z}}\right) - T\left(y_i^{\mathrm{z}}\right) \right]^2 \right) \tag{3.37}$$

where $x_i^{\mathrm{z}}, y_i^{\mathrm{z}}\ \forall i \in I^{\mathrm{z}}(x^{\mathrm{m}})$ are relaxed integer variables and λ_i^{z} are hyperparameters of the surrogate model. The transformation T conserves the order of an integer variable and ensures that the one-dimensional kernels in (3.37) are piecewise functions [25].

The parametrized standard kernel $k^{\mathrm{s}} : \mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}}) \times \mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}}) \to \mathbb{R}$ is formulated as multiplication of k^{z} and k^{c} :

$$k^{\mathrm{s}}(x^{\mathrm{s}}, y^{\mathrm{s}}; x^{\mathrm{m}}) = k^{\mathrm{z}}(x^{\mathrm{z}}, y^{\mathrm{s}}; x^{\mathrm{m}}) \cdot k^{\mathrm{c}}(x^{\mathrm{s}}, y^{\mathrm{s}}; x^{\mathrm{m}}). \tag{3.38}$$

The parametrized categorical kernel k^{q} may be formulated with an encoding on the categorical variables [8] ($k^{\mathrm{q}} : \mathcal{E}^{\mathrm{q}}(x^{\mathrm{m}}) \times \mathcal{E}^{\mathrm{q}}(x^{\mathrm{m}}) \to \mathbb{R}$), or without any encoding ($k^{\mathrm{q}} : \mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}}) \times \mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}}) \to \mathbb{R}$). With an encoding, the parametrized categorical kernel k^{q} is similar to k^{c} :

$$k^{\mathrm{q}}(e^{\mathrm{q}}, u^{\mathrm{q}}; x^{\mathrm{m}}) = \exp\left(- \sum_{i \in \mathcal{E}_{\mathrm{aux}}^{\mathrm{q}}(x^{\mathrm{m}})} \lambda_i^{\mathrm{q}} \left[e_i^{\mathrm{q}} - y_i^{\mathrm{q}} \right]^2 \right), \tag{3.39}$$

where λ_i^{q} are hyperparameters of the surrogate model and $\mathcal{E}_{\mathrm{aux}}^{\mathrm{q}}(x^{\mathrm{m}})$ is the set of indices of the encoded (acting) categorical variables. Without any encoding, the parametrized categorical kernel k^{q} is formulated as tensor products of matrices (one matrix per categorical categorical) [20] :

$$k^{\mathrm{q}}(x^{\mathrm{q}}, y^{\mathrm{q}}; x^{\mathrm{m}}) = \left(\bigotimes_{i=1}^{n^{\mathrm{q_u}}(x^{\mathrm{m}})} T_i^{\mathrm{q_u}}\left(x_i^{\mathrm{q_u}}, y_i^{\mathrm{q_u}}\right) \right) \otimes \left(\bigotimes_{i=j}^{n^{\mathrm{q_o}}(x^{\mathrm{m}})} T_j^{\mathrm{q_o}}\left(x_j^{\mathrm{q_o}}, y_j^{\mathrm{q_o}}\right) \right), \tag{3.40}$$

where, for $t \in \{\mathrm{q_u}, \mathrm{q_o}\}$ and a categorical variable $x_i^t \in \{1, 2, \ldots, c_i\}$, $T_i^t \in \mathbb{R}^{c_i \times c_i}$ is a positive semi-definite matrix in which an element is the correlation between two categories of the

variable x_i^t. Hence, for two given variables with specific categories $x_i^t = c_1$ and $y_i^t = c_2$, $T_i^t\left(x_i^t, y_i^t\right)$ is a correlation measure between the categories c_1 and c_2. In Equation (3.40), the matrices for the nominal and ordinal variables $T_i^{\mathrm{q_u}}$ and $T_j^{\mathrm{q_o}}$ are distinguished, since there exist more sophisticated matrices for the ordinal variables [20].

Finally, a mixed kernel $k : \mathcal{X}_{\text{aux}} \times \mathcal{X}_{\text{aux}} \to \mathbb{R}$, based on [39], is formulated as :

$$k(x, y) = \begin{cases} \prod_{i=1}^{n^{\mathrm{m}}} k_i^{\mathrm{m}}(x_i^{\mathrm{m}}, y_i^{\mathrm{m}}), & \text{if } x^{\mathrm{m}} \neq y^{\mathrm{m}}, \\ \prod_{i=1}^{n^{\mathrm{m}}} \left(k_i^{\mathrm{m}}(x_i^{\mathrm{m}}, y_i^{\mathrm{m}}) \cdot \left[k^{\mathrm{q}}(l^{\mathrm{q}}, u^{\mathrm{q}}; x^{\mathrm{m}}) k^{\mathrm{s}}(x^{\mathrm{s}}, y^{\mathrm{s}}; x^{\mathrm{m}}) \right] \right), & \text{otherwise,} \end{cases} \tag{3.41}$$

where $k_i^{\mathrm{m}} : S_i^{\mathrm{m}} \times S_i^{\mathrm{m}} \to \mathbb{R}$ is a one-dimensional kernel for a meta variable $x_i^{\mathrm{m}} \in S_i^{\mathrm{m}}$, k^{q} is the parametrized categorical kernel that may take the form in Equation (3.39) or Equation (3.40), k^{s} is the parametrized standard kernel in Equation (3.38). In Equation (3.41), the meta kernel $k^{\mathrm{m}} : \mathcal{X}^{\mathrm{m}} \times \mathcal{X}^{\mathrm{m}} \to \mathbb{R}$ is implicitly decomposed into one-dimensional kernels (one per meta variable), which is again, common practice in the literature. Moreover, in Equation (3.41), the kernel computations for the categorical and standard variables are only being done if the two points in share the same meta component : for $t \in \{\mathrm{q}, \mathrm{s}\}$, the kernel computation $k(x^t, y^t; x^{\mathrm{m}})$ in Equation (3.41) is only done if $x^{\mathrm{m}} = y^{\mathrm{m}}$, which implies that x^t and y^t must both reside in the same parametrized set $\mathcal{X}^t(x^{\mathrm{m}})$.

3.5 Conclusion

This work proposes a thorough notation framework for mixed-variable optimization problems. The framework formally models mixed-variable problems with a careful emphasis on meta and categorical variables. More precisely, a point $x = (x^{\mathrm{m}}, x^{\mathrm{q}}, x^{\mathrm{s}})$, the domain $\mathcal{X}$ of the objective function, and the feasible set Ω, are rigorously defined. Definitions are developed to shed the light on the intrinsic difficulties resulting from the presence of meta variables. Notably, for $t \in \{\mathrm{q}, \mathrm{s}, \mathrm{q_u}, \mathrm{q_o}, \mathrm{z}, \mathrm{c}\}$, a parametrized set $\mathcal{X}^t(x^{\mathrm{m}})$ elucidates that some variables of type t may be acting or nonacting depending on the meta variables.

In addition, the parametrized categorical set $\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ and the parametrized standard set $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ are building blocks of the domain $\mathcal{X}$ that has two equivalent formulations, respectively in Definition 2 and in Equation (3.6). Both formulations provide a different perspective on mixed-variable problems. Furthermore, the constraints are split into global decreed constraints, which allow to formulate a clear feasible set Ω in Definition 3.14.

In Section 3.4, the subproblems strategy and auxiliary problem strategy are exhaustively discussed from the notation framework. These strategies allow to formally adapt the notation framework to the direct search approach through the subproblems strategy, as well as to

the Bayesian optimization approach through the auxiliary problem strategy. Thereby, the notation framework is shown to be compatible with most of the approaches of the literature on mixed-variable optimization with meta (or dimensional) and categorical variables.

Computational experiments will be carried out in future studies with the mathematical framework of this work as a foundation. Moreover, the Bayesian optimization approach will also be extensively developed, with aim of bridging the communities in optimization and machine learning.

CHAPITRE 4 DISCUSSION GÉNÉRALE

Le travail effectué dans ce mémoire est fondamentalement théorique, puisqu'il consiste essentiellement à définir un cadre de modélisation mathématique. Le mémoire ne comporte aucun résultat numérique, ce qui diffère considérablement du format typique d'un article en optimisation ou en apprentissage profond. Cependant, le mémoire a une vocation similaire aux prestigieux articles [19] et [42], qui présentent respectivement un cadre pour la résolution par substituts et un cadre général pour la recherche directe. La contribution principale de ce mémoire est de proposer un cadre mathématique qui permet de modéliser et développer des méthodes algorithmiques pour les problèmes mixtes d'optimisation de boîtes noires sous contraintes.

Ensuite, bien que des stratégies de résolution ont été formalisées et proposées, aucune implémentation concrète n'a été effectuée dans ce travail. Ainsi, plusieurs détails algorithmiques ont été omis afin de ne pas alourdir la présentation. En d'autres mots, le mémoire propose des approches et des stratégies de résolution sans offrir une implémentation concrète et détaillée.

La notation du mémoire est rigoureuse et elle permet d'adéquatement modéliser les problèmes mixtes. Cependant, la notation peut paraître lourde et comporte une terminologie très technique. À cet égard, différents choix de modélisation auraient pu être effectués. Entre autres, au risque d'être moins formel, un système de notation plus simple et plus accessible aurait pu être proposé. La lecture de ce mémoire demande un investissement important, puisqu'il est très technique. Ce mémoire est adressé à un auditoire ayant des connaissances et compétences mathématiques avancées, ce qui pourrait être considéré comme une limitation de celui-ci. Néanmoins, le principal intérêt de ce mémoire est qu'il est compatible avec les approches employées entre les communautés de recherche en optimisation et apprentissage automatique. Ce mémoire ouvre la porte à la collaboration entre ces deux communautés.

CHAPITRE 5 CONCLUSION ET RECOMMANDATIONS

Dans ce mémoire, un cadre mathématique a été développé afin de traiter les problèmes mixtes d'optimisation de boîtes noires sous contraintes. Ce cadre mathématique comporte un système de notation complet, qui modélise formellement les problèmes d'intérêt. Plus précisément, un point $x = (x^{\mathrm{m}}, x^{\mathrm{q}}, x^{\mathrm{s}})$, le domaine de la fonction objectif $\mathcal{X}$ et l'ensemble réalisable Ω, sont rigoureusement définis. De cette manière, la formulation générale d'un problème d'optimisation est adéquatement explicitée pour les problèmes mixtes.

À notre connaissance, aucun cadre mathématique aussi rigoureux n'a été proposé pour de l'optimisation mixte avec des variables catégorielles et méta. Ces définitions ont été développées soigneusement pour mettre en évidence les difficultés causées par la présence de variables méta. Notamment, pour $t \in \{\mathrm{q}, \mathrm{s}, \mathrm{q_u}, \mathrm{q_o}, \mathrm{z}, \mathrm{c}\}$, des ensembles paramétrisés $\mathcal{X}^t(x^{\mathrm{m}})$ explicitent que certaines variables de type t sont agissantes ou nonagissantes, dépendamment des variables méta. Additionnellement, l'ensemble paramétré catégoriel $\mathcal{X}^{\mathrm{q}}(x^{\mathrm{m}})$ et l'ensemble paramétré standard $\mathcal{X}^{\mathrm{s}}(x^{\mathrm{m}})$ permettent de définir méticuleusement le domaine $\mathcal{X}$ en considérant la propriété de décret des variables méta. Les contraintes sont séparées en contraintes globales et contraintes décrétées, ce qui permet de formuler clairement l'ensemble réalisable Ω.

Ensuite, la stratégie de sous-problèmes et la stratégie du problème auxiliaire sont exhaustivement intégrées au système de notation. L'approche par recherche directe est formellement adaptée au système de notation par l'intermédiaire d'une stratégie de sous-problèmes. Puis, l'optimisation bayésienne est adaptée à partir d'une stratégie de problème auxiliaire. Ainsi, il est illustré que les approches importantes de la littérature sont compatibles avec le cadre mathématique (notation et stratégies).

5.1 Travaux futurs

Dans le cadre d'une thèse de doctorat, plusieurs expériences numériques seront effectuées sur la base du cadre mathématique développé dans ce mémoire. En effet, le problème d'optimisation des hyperparamètres sera concrètement traité à partir de l'été 2022. Des méthodes simples et peu coûteuses seront d'abord employées pour établir des résultats préliminaires. Ensuite, de nouvelles méthodologies, développées à partir du cadre mathématique, seront développées et mise en compétition avec les méthodes plus simples. De plus, une analyse plus approfondie de l'optimisation bayésienne sera aussi effectuée, afin d'inciter davantage la collaboration entre les communautés en optimisation et en apprentissage automatique.

RÉFÉRENCES

[1] K. Abhishek, S. Leyffer, and J.T. Linderoth. Modeling without categorical variables : a mixed-integer nonlinear program for the optimization of thermal insulation systems. *Optimization and Engineering*, 11(2) :185–212, 2010.

[2] M.A. Abramson. *Pattern Search Algorithms for Mixed Variable General Constrained Optimization Problems*. PhD thesis, Department of Computational and Applied Mathematics, Rice University, August 2002.

[3] M.A. Abramson. Mixed Variable Optimization of a Load-Bearing Thermal Insulation System Using a Filter Pattern Search Algorithm. *Optimization and Engineering*, 5(2) :157–177, 2004.

[4] M.A. Abramson, C. Audet, J.W. Chrissis, and J.G. Walston. Mesh Adaptive Direct Search Algorithms for Mixed Variable Optimization. *Optimization Letters*, 3(1) :35–47, 2009.

[5] M.A. Abramson, C. Audet, and J.E. Dennis, Jr. Filter pattern search algorithms for mixed variable constrained optimization problems. *Pacific Journal of Optimization*, 3(3) :477–500, 2007.

[6] R.P. Adams, N. De Freitas, B. Shahriari, K. Swersky, and Z. Wang. Taking the human out of the loop : A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1) :148–175, 2015.

[7] S. Alarie, C. Audet, A.E. Gheribi, M. Kokkolaras, and S. Le Digabel. Two decades of blackbox optimization applications. *EURO Journal on Computational Optimization*, 9 :100011, 2021.

[8] D.W. Apley, W. Chen, S. Tao, and Y. Zhang. A Latent Variable Approach to Gaussian Process Modeling with Qualitative and Quantitative Factors. *Technometrics*, 62(3) :291–302, 2020.

[9] C. Audet and J.E. Dennis, Jr. Pattern Search Algorithms for Mixed Variable Programming. *SIAM Journal on Optimization*, 11(3) :573–594, 2001.

[10] C. Audet and J.E. Dennis, Jr. Mesh Adaptive Direct Search Algorithms for Constrained Optimization. *SIAM Journal on Optimization*, 17(1) :188–217, 2006.

[11] C. Audet and J.E. Dennis, Jr. A Progressive Barrier for Derivative-Free Nonlinear Programming. *SIAM Journal on Optimization*, 20(1) :445–472, 2009.

[12] C. Audet and W. Hare. *Derivative-Free and Blackbox Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, Cham, Switzerland, 2017.

[13] C. Audet, S. Le Digabel, V. Rochon Montplaisir, and C. Tribes. NOMAD version 4 : Nonlinear optimization with the MADS algorithm. Technical Report G-2021-23, Les cahiers du GERAD, 2021.

[14] C. Audet, S. Le Digabel, and C. Tribes. The Mesh Adaptive Direct Search Algorithm for Granular and Discrete Variables. *SIAM Journal on Optimization*, 29(2) :1164–1189, 2019.

[15] C. Audet and C. Tribes. Mesh-based Nelder-Mead algorithm for inequality constrained optimization. *Computational Optimization and Applications*, 71(2) :331–352, 2018.

[16] P.-J. Barjhoux, D. Bettebghor, Y. Diouane, S. Grihon, and J. Morlier. A bi-level methodology for solving large-scale mixed categorical structural optimization. *Structural and Multidisciplinary Optimization*, 62 :337–351, 2020.

[17] Y. Bengio and J. Bergstra. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13 :281–305, 2012.

[18] M. Binois and N. Wycoff. A survey on high-dimensional Gaussian process modeling with application to Bayesian optimization. Technical Report 2111.05040, arXiv, 2021.

[19] A.J. Booker, J.E. Dennis, Jr., P.D. Frank, D.B. Serafini, V. Torczon, and M.W. Trosset. A Rigorous Framework for Optimization of Expensive Functions by Surrogates. *Structural and Multidisciplinary Optimization*, 17(1) :1–13, 1999.

[20] A. Clément, Y. Deville, J. Giorla, E. Padonou, G. Perrin, O. Roustant, and H. Wynn. Group kernels for Gaussian process metamodels with categorical inputs. *Uncertainty Quantification*, 8(2) :775–806, 2020.

[21] A.R. Conn, K. Scheinberg, and L.N. Vicente. *Introduction to Derivative-Free Optimization*. MOS-SIAM Series on Optimization. SIAM, Philadelphia, 2009.

[22] J. Cuesta-Ramirez, C. Durantin, A. Gliere, R. Le Riche, G. Perrin, and O. Roustant. A comparison of mixed-variables Bayesian optimization approaches. Technical report, arXiv, 2021.

[23] George B. Dantzig. *Linear Programming and Extensions*. 1963.

[24] J.E. Dennis, Jr. and V. Torczon. Direct search methods on parallel machines. *SIAM Journal on Optimization*, 1(4) :448–474, 1991.

[25] E.C. Garrido-Merchán and D. Hernández-Lobato. Dealing with categorical and integer-valued variables in Bayesian Optimization with Gaussian processes. *Neurocomputing*, 380 :20–35, 2020.

[26] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[27] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

[28] R. Hooke and T.A. Jeeves. “Direct Search" Solution of Numerical and Statistical Problems. *Journal of the Association for Computing Machinery*, 8(2) :212–229, 1961.

[29] D.R Jones, M. Schonlau, and W.J. Welch. . *Journal of Global Optimization*, 13(4) :455–492, 1998.

[30] M. Kokkolaras, C. Audet, and J.E. Dennis, Jr. Mixed variable optimization of the Number and composition of heat intercepts in a thermal insulation system. *Optimization and Engineering*, 2(1) :5–29, 2001.

[31] D. Lakhmiri, S. Le Digabel, and C. Tribes. HyperNOMAD : Hyperparameter Optimization of Deep Neural Networks Using Mesh Adaptive Direct Search. *ACM Transactions on Mathematical Software*, 47(3), 2021.

[32] S. Le Digabel and S.M. Wild. A Taxonomy of Constraints in Simulation-Based Optimization. Technical Report G-2015-57, Les cahiers du GERAD, 2015.

[33] G. Liuzzi, S. Lucidi, V. Piccialli, and A. Sotgiu. A magnetic resonance device designed via global optimization techniques. *Mathematical Programming*, 101(2) :339–364, 2004.

[34] S. Lucidi and V. Piccialli. A Derivative-Based Algorithm for a Particular Class of Mixed Variable Optimization Problems. *Optimization Methods and Software*, 17(3–4) :317–387, 2004.

[35] S. Lucidi, V. Piccialli, and M. Sciandrone. An Algorithm Model for Mixed Variable Programming. *SIAM Journal on Optimization*, 15(4) :1057–1084, 2005.

[36] J.J. Moré and S.M. Wild. Benchmarking Derivative-Free Optimization Algorithms. *SIAM Journal on Optimization*, 20(1) :172–191, 2009.

[37] M. Munoz Zuniga and D. Sinoquet. Global optimization for mixed categorical-continuous variables based on Gaussian process models with a randomized categorical space exploration step. *INFOR : Information Systems and Operational Research*, 58(2) :310–341, 2020.

[38] G. Nannicini. On the implementation of a global optimization method for mixed-variable problems. *Open Journal of Mathematical Optimization*, 2 :1–25, 2021.

[39] J. Pelamatti, L. Brevault, M. Balesdent, E.-G. Talbi, and Y. Guerin. Bayesian optimization of variable-size design space problems. *Optimization and Engineering*, 22 :387–447, 2021.

[40] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[41] R.G. Regis and C.A. Shoemaker. A stochastic radial basis function method for the global optimization of expensive functions. *INFORMS J. Comput.*, 19 :497–509, 2007.

[42] V. Torczon. On the convergence of pattern search algorithms. *SIAM Journal on Optimization*, 7(1) :1–25, 1997.