Aurélien MOREL

2A GEN
Année universitaire 2023/2024

Laboratoire de Physique Subatomique et Cosmologie
53 Avenue des Martyrs, 38000 Grenoble

Développement d'outils de calculs pour la conception des réacteurs à sels fondus

du 06/05/2024 au 02/08/2024

Sous la supervision de :

Maître de stage : Axel LAUREAU axel.laureau@cnrs.fr
Encadrant : Thomas SORNAY sornay@lpsc.in2p3.fr

Tuteur école : Anne VILCOT anne.vilcot@grenoble-inp.fr

École nationale supérieure de physique électronique matériaux
Phelma
Bât. Grenoble INP - Minatec
3 Parvis Louis Néel - CS 50257
F-38016 Grenoble Cedex 01
Tél +33 (0)4 56 52 91 00
Fax +33 (0)4 56 52 91 03
http://phelma.grenoble-inp.fr

Remerciements

Je tiens à remercier l'ensemble de l'équipe MSFR pour leur accueil lors de ces semaines passées au LPSC. Un grand merci à Elsa Merle, Axel Laureau et Thomas Sornay pour m'avoir offert l'opportunité de faire ce stage en me laissant autant de libertés dans mon travail, tout en étant toujours disponibles pour m'éclairer et m'aider.

Merci Elsa pour ta gestion et supervision de l'administratif, pour ta réactivité dans ma période d'éclopé et pour tes délicieux gâteaux.

Merci Axel pour m'avoir toujours écouté d'une oreille attentive et m'avoir aiguillé quand j'en avais besoin.

Merci Thomas de m'avoir aidé à amadouer SONGe, pour tes jolis plots, pour m'avoir accompagné dans ma lutte contre JAVA, bien au frais dans ton bureau et pour ton encadrement tout le long de ce stage.

Merci Adrien, Anna, Louiliam, Max, Alexandre et Hugo pour les pauses-café qui ont à chaque fois animé mes journées.

Table des matières

I Introduction . . . 1
- I.1 Énergie, société et climat . . . 1
- I.2 L'énergie nucléaire . . . 1
- I.3 Défis et perspectives pour un nucléaire mondiale . . . 2

II Les MSFR . . . 3
- II.1 Résumé historique des réacteurs à sels fondus . . . 3
- II.2 Le MSFR de référence . . . 4
- II.3 Points forts et limites de la technologie MSFR fluorure . . . 5

III Conception optimisée d'un MSFR . . . 6
- III.1 Objectifs . . . 6
- III.2 Tour d'horizon des algorithmes d'optimisation . . . 7
- III.3 SONGe . . . 9

IV L'optimisation sachant optimiser . . . 12
- IV.1 Discrimination des algorithmes d'optimisations . . . 12
- IV.2 Paramétrisation de la méthode d'optimisation . . . 18
- IV.3 Conclusion et possibilités de poursuite d'étude . . . 25

V Pareto et aide à la décision . . . 26
- V.1 Introduction . . . 26
- V.2 Première idée . . . 27
- V.3 Nouveau coût, nouveau monde . . . 28
- V.4 Comparaison pertes de charge nouvelle méthode . . . 30
- V.5 Étude puissance demandée . . . 30
- V.6 Problème de présentation des résultats . . . 31
- V.7 Conclusion . . . 36

VI Conclusion . . . 36

VII Annexes . . . 38
- VII.1 Résumé en français . . . 38
- VII.2 English abstract . . . 38
- VII.3 Figures annexes . . . 39
- VII.4 Compétences . . . 40

Références **I**

I Introduction

I.1 Énergie, société et climat

La maîtrise de l'énergie procure à l'homme la capacité de modifier et de façonner son environnement afin de subvenir à ses besoins. Sa production, sa gestion et sa répartition sont des sujets vitaux pour la prospérité des sociétés.

En 2022, le mix de consommation énergétique mondiale se répartit entre le pétrole (32 %), le charbon à (27 %), le gaz (23 %), le nucléaire (4.0 %) et les énergies renouvelables (14 %) [1]. La part majeure d'hydrocarbure dans le mix énergétique mondiale est un problème phare dans la lutte contre le réchauffement climatique et l'objectif de le limiter à 2 °C [2]. En effet, leur combustion produit du gaz carbonique (CO_2), un gaz à effet de serre. Accélérer le déploiement d'infrastructures de production d'énergie à émission carbone faible ou nulle constitue une partie de solution à ce problème. Des engagements concernant le déploiement d'installations de production d'énergie renouvelable et de l'utilisation du nucléaire sont partagés à l'échelle internationale dans ce sens [2].

En France, l'ambition de réduction des émissions carbone passe par un objectif de bilan carbone neutre d'ici à 2050 [3]. En 2022, le mix énergétique Français est réparti entre le pétrole (35 %), le charbon (3 %), le gaz (16 %), le nucléaire (32 %) et les énergies renouvelables (14 %) [1]. La part importante du nucléaire dans le mix énergétique français en fait un axe stratégique dans la politique de lutte contre le réchauffement climatique.

I.2 L'énergie nucléaire

Un avantage majeur de l'énergie nucléaire est sa densité énergétique. Une tonne d'uranium-235 libère environ $8,2.10^{16}$ J, c'est 2 millions de fois de plus qu'une tonne de pétrole (42.10^9 J/t) et 5,5 millions de fois de plus qu'une tonne de bois (15.10^9 J/t). Les quantités d'uranium-235 mises en jeu, à production égale d'énergie, sont alors bien plus faibles que celles du pétrole ou du charbon. Dans ce sens, l'utilisation de l'énergie nucléaire permet de limiter l'exploitation minière des sols. De plus, le phénomène de fission nucléaire se fait sans émission de gaz à effet de serre. Son utilisation, en substitution des hydrocarbures, contribue à limiter le réchauffement climatique.

L'utilisation de la fission nucléaire pour la production d'énergie est récente. La découverte du neutron par James Chadwick date de 1932. La première réaction en chaine contrôlée a lieu en 1942 avec l'expérience de la Chicago Pile-1. En 1948, le Réacteur graphite X-10 (États-Unis) est le premier à produire de l'électricité à partir de l'énergie nucléaire [4].

En France, c'est en 1963 que le premier réacteur nucléaire d'EDF de Chinon alimente pour la première fois le réseau électrique. Les premiers réacteurs français sont à uranium naturel graphite gaz (UNGG). Les réacteurs UNGG utilisent de l'uranium naturel comme combustible et du graphite comme modérateur. Du CO_2 est utilisé comme fluide caloporteur pour transporter la chaleur produite par la réaction nucléaire vers les échangeurs de chaleur. Aujourd'hui, le parc de réacteurs nucléaires français comporte exclusivement des réacteurs à eau pressurisée (REP). Cette technologie de réacteur utilise l'eau liquide comme caloporteur et modérateur. Elle circule à haute pression pour transférer la chaleur produite par la réaction nucléaire vers un échangeur de chaleur, où elle est utilisée pour produire de la vapeur qui alimente ensuite une turbine pour générer de l'électricité.

I.3 Défis et perspectives pour un nucléaire mondiale

Le passage à une production nucléaire plus importante poserait des problèmes complexes inexistants dans le contexte actuel d'exploitation.

La durée des stocks d'uranium sont estimés en fonction des besoins actuels de la filière nucléaire mondiale. Une part plus importante de nucléaire dans le mix énergétique mondial ou national, via un déploiement massif de centrale nucléaire, entraînerait des besoins croissants en matière combustible. L'accessibilité et le prix des stocks d'uranium s'en retrouverait alors modifiés. Bien que les réserves d'uranium soient abondantes, la demande accrue pourrait entraîner une pression sur les prix et nécessiterait une exploration et une exploitation plus poussées de nouveaux gisements. Les réserves héritées de la guerre froide [5], jusqu'alors bon marché, ne suffiraient plus à couvrir la demande [6]. Ce scénario souligne l'importance de la gestion des ressources pour atteindre une souveraineté énergétique [7].

La gestion des déchets est une problématique inhérente à l'exploitation de l'énergie nucléaire. En France, tout matériau contenant des substances radioactives pour lesquelles aucune utilisation ultérieure n'est prévue est défini comme un déchet radioactif [8]. Les produits de fission, générés par la fission des noyaux lourds, ainsi que les actinides mineurs, résultant de captures neutroniques et transmutations successives par ces mêmes noyaux, comptent parmi les substances radioactives présentes dans les déchets nucléaires. Les déchets radioactifs peuvent provenir de l'exploitation des installations nucléaires, de leurs démantèlements ou des résidus du traitement du combustible usagé. Ces déchets doivent être éliminés ou stockés de manière sûre et contrôlée. Leur revalorisation permet de limiter l'impact sur l'environnement et de remanier les fonds dédiés au traitement et à l'enfouissement. L'opinion publique autour du déploiement du nucléaire est affectée par la gestion de ces déchets [9].

Maintenir un niveau de sûreté élevé des installations nucléaires est un enjeu permanent. Chaque nouvelle technologie et installation doivent être plus sûres que la précédente. Le déploiement massif du nucléaire ne peut se faire sans gagner l'opinion publique et atténuer les réticences vis-à-vis du danger de ces installations.

Une vigilance vis-à-vis de la prolifération est aussi de mise quant à un déploiement massif d'installations nucléaires. Les installations doivent pouvoir être facilement contrôlées et les dérives possibles difficiles et facilement traçables.

C'est en prenant en considération l'ensemble de ces exigences que les nouveaux réacteurs de demain sont conçus. Le forum Génération IV [1] a proposé en 2008 une liste de six concepts des réacteurs permettant de respecter et d'améliorer les points précédemment énoncés [10]. Cette liste est composée des réacteurs à neutrons rapides refroidis au sodium, des réacteurs à sels fondus, des réacteurs à très haute température, des réacteurs à eau supercritique, des réacteurs à neutrons rapides refroidis au gaz et enfin des réacteurs à neutrons rapides refroidis au plomb.

1. Le forum génération IV (GIF) est une coopération internationale cherchant à développer la recherche pour tester la faisabilité et la performance des systèmes nucléaires de quatrième génération pour leur déploiement industriel. Le GIF rassemble 13 pays (Argentine, Australie, Brésil, Canada, Chine, France, Japon, Corée, Russie, Afrique du Sud, Suisse, Royaume-Uni et États-Unis) ainsi qu'Euratom (UE).

II Les MSFR

Un réacteur à sels fondus (RSF) fonctionne en utilisant comme combustible et caloporteur un mélange de sels fondus, contrairement aux REP qui utilisent de l'eau sous haute pression pour modérer et refroidir le réacteur. Le combustible de sels fondus chauffe au cœur du réacteur du fait de sa géométrie critique. En dehors du cœur, il joue uniquement le rôle de caloporteur. Le sel en question peut être à base de fluorure ou de chlorure. Il se présente sous phase liquide à des températures plus élevées que celles de fonctionnement de l'eau au sein des REP, typiquement entre 600 °C et 1000 °C. Une autre de ses spécificités est qu'il fonctionne à pression atmosphérique. À l'opposé, un REP utilise de l'eau pressurisée aux alentours de 150 bars pour maintenir l'eau liquide proche de 300 °C.

II.1 Résumé historique des réacteurs à sels fondus

L'histoire des RSF remonte à la fin de la Seconde Guerre mondiale, lorsque l'armée de l'air américaine a lancé le programme NEPA pour explorer l'utilisation de l'énergie nucléaire dans la propulsion des avions. Ce programme a ensuite évolué vers l'Aircraft Nuclear Propulsion Program, avec l'objectif de développer des avions nucléaires capables de voler à grande vitesse et basse altitude pour éviter la détection radar. Bien que des progrès aient été réalisés, aucun avion à propulsion nucléaire n'a été opérationnellement utilisé.

Néanmoins, les recherches sur les réacteurs à sels fondus ont continué avec l'Aircraft Reactor Experiment (ARE) développé par le laboratoire d'Oak Ridge (ORNL). Ce réacteur utilisait des sels fluorures liquides comme combustible. Le caractère homogène du combustible liquide et les contre-réactions négatives [2] liées notamment aux variations de densité offraient une stabilité de réactivité accrue. Le succès de l'ARE a conduit au développement du Molten Salt Reactor Experiment (MSRE), un réacteur de démonstration qui a fonctionné entre 1965 et 1969. Le MSRE a montré la faisabilité de l'utilisation des sels fondus pour produire de l'électricité, ainsi que la facilité de maintenance et de gestion du combustible.

Le Molten Salt Breeder Reactor (MSBR), conçu pour utiliser le cycle thorium, a été lancé après l'arrêt du MSRE. Ce réacteur régénérateur commercial visait à être largement déployé en utilisant des ressources abondantes non exploitées aux États-Unis. Cependant, le projet a été interrompu en 1976 en raison de contraintes budgétaires et de défis techniques, notamment liés à la production de tritium et à la fragilisation des matériaux de structure.

Les recherches sur les RSF ne se sont pas limitées aux États-Unis. En Union soviétique (puis Russie), le concept MOSART (MOlten Salt Actinides Recycler and Transmuter - Recycleur et Transmutateur d'actinides à sels fondus) a émergé, visant à recycler et à transmuter [3] les actinides et les combustibles usés. D'autres pays comme la Pologne, la Suisse, le Japon et la Chine ont également mené des études sur les réacteurs à sels fondus, explorant différents concepts et technologies.

En France, le CEA et EDF ont commencé à travailler sur des concepts de réacteurs à sels fondus dans les années 1990. Des projets, tels qu'AMSTER (Actinides Molten

2. « diminution de la réactivité du cœur d'un réacteur due à la variation de certains paramètres de fonctionnement tels que la puissance, la température, la pression ou le taux de vide du caloporteur » [11].

3. « La transmutation est la transformation d'un noyau en un autre par une réaction nucléaire induite par des particules avec lesquelles il est bombardé » [12]

Salt TransmutER - Transmutateur d'actinides à sels fondus) mené par EDF et TASSE (Thorium based Accelerator driven System with Simplified fuel cycle for long term Energy production - Système à base de thorium piloté par accélérateur avec cycle de combustible simplifié pour la production d'énergie à long terme) mené par le CEA, ont alors vu le jour. Le CNRS a quant à lui monté le projet TMSR (Thorium Molten Salt Reactor - Réacteur à sels fondus au thorium) dans les années 2000 [13]. Le TMSR a évolué pour devenir le MSFR (Molten Salt Fast Reactor - Réacteur à sels fondus à spectre rapide), un réacteur à spectre rapide utilisant le cycle thorium, qui a été retenu comme design de référence pour les réacteurs de génération IV.

Ces efforts internationaux ont contribué à acquérir une compréhension des avantages et des défis associés aux réacteurs à sels fondus, ouvrant la voie à leur intégration dans les futurs systèmes énergétiques mondiaux.

II.2 Le MSFR de référence

Le MSFR de référence étudié initialement est un réacteur surgénérateur de 3 GWth à spectre rapide et utilisant le cycle thorium (thorium-232/uranium-233) dans un sel de fluorure de lithium. Le sel combustible circule de la zone critique (le cœur) vers les échangeurs de chaleur, où la chaleur sera transférée à un sel intermédiaire. Il possède une couverture fertile [4] (composée d'un sel de fluorure ou de chlorure de thorium) pour la production d'uranium-233. L'utilisation en cycle uranium-238/plutonium-239 est aussi envisageable avec un sel de fluorure de chlorure [14].

La technologie MSFR offre la possibilité d'agir comme incinérateur, brûleur ou surgénérateur. En tant qu'incinérateurs, ils transmutent les actinides mineurs (américium et curium principalement), réduisant ainsi la quantité et la toxicité des déchets nucléaires nécessitant un stockage à long terme. Le terme d'incinération est alors employé. Le fonctionnement en tant que brûleur est sensiblement le même, mais appliqué spécifiquement au plutonium qui n'est pas considéré comme un déchet nucléaire. En tant que surgénérateurs, les RSF peuvent produire plus de combustible fissile qu'ils n'en consomment en transmutant des matériaux fertiles comme le thorium-232 ou l'uranium-238. La matière fertile qui jusqu'alors n'était pas exploitée est valorisée. Cette capacité à générer du combustible tout en détruisant les déchets existants représente un avantage significatif par rapport aux REP.

Les efforts récents dans le domaine des réacteurs à sels fondus sont également portés par plusieurs startups et entreprises au niveau national, européen et mondial. Parmi elles, Naarea, une entreprise française fondée en 2020, développe une technologie de petits réacteurs modulaires à neutrons rapides basés sur les sels fondus [15]. De son côté, Stellaria, startup portée par le CEA, travaille sur un réacteur à sels fondus chlorures au thorium capable de régénérer entièrement son combustible en continu [16]. Thorizon, une startup néerlandaise récemment implantée à Lyon, développe des réacteurs modulaires à sels fondus en cycle thorium ciblant principalement les applications industrielles à travers la production de chaleur ou d'électricité [17]. Enfin, TerraPower, société crée en 2006 par Bill Gates, a comme projet, en partenariat avec Orano, la construction d'une maquette critique d'un petit RSF à spectre rapide d'ici 2026 [18].

4. Couverture constituée d'un matériau fertile

II.3 Points forts et limites de la technologie MSFR fluorure

Dans un MSFR, le système de circulation du sels fondus est assuré par des pompes. En cas de dysfonctionnement mécanique ou de coupure d'alimentation électrique, le design des réacteurs couplé aux propriétés fluides du combustible et du caloporteur permet à la convection naturelle d'assurer le refroidissement. Les réacteurs à sels fondus sont conçus avec un coefficient de réactivité plus négatif que les réacteurs actuels (Contre-réaction de densité et Doppler). Toute perturbation accidentelle entraîne une réduction automatique de la réactivité du réacteur, contribuant ainsi à prévenir les accidents nucléaires graves.

Un autre avantage de l'utilisation d'un fluide homogène, jouant à la fois le rôle de caloporteur et de combustible, est que sa composition peut être modifiée en service. L'élimination des espèces indésirables (produits de fission, impuretés) et l'ajustement de la composition du combustible sont possibles sans arrêt du réacteur. Effectuer une vidange complète ou partielle est possible.

La technologie des MSFR est conçue pour être résiliente à la prolifération. La manipulation du sel combustible est facilement traçable en raison d'émission gamma très énergétique de 2,6 Mev du cycle thorium-232/uranium-233 lors de la décroissance de thallium-208 en plomb-208 [19]. Il ne peut pas être détourné facilement. Au niveau de la couverture fertile, l'ajout de transuranien (TRU) usagé, afin de dégrader le vecteur [5] plutonium (maintenir 38 % de plutonium-239), ainsi que d'éléments radioactifs à vie courte, tel que le curium, est envisagé [14].

Enfin, la température de fonctionnement du cycle thermique plus élevée qu'en REP procure une meilleure efficacité et augmente le rendement de l'installation par rapport aux REP.

Cependant, le concept de MSFR présente des limites. Le retour d'expérience sur la technologie est très faible, comme présenté dans le résumé historique. Dans l'ensemble, les MSFR présentent des différences technologiques importantes par rapport aux REP existants et à d'autres concepts de réacteurs avancés, ce qui nécessite différentes approches pour l'exploitation et l'évaluation de la sûreté.

L'exploitation d'un combustible en sels fondus présente plusieurs points durs. Le matériau du réacteur doit bien se comporter sous de hautes contraintes (température, corrosion et flux neutronique). Les sels fondus peuvent devenir très corrosifs s'ils sont exposés à des impuretés oxydantes. Des études sont encore en cours pour trouver des aciers plus résistants, ou pour permettre d'ajuster leur composition afin d'éviter ce phénomène. Des solutions basées sur des couples tampons redox sont utilisées en fluorure, mais restent à l'étude en chlorure. À cela vient s'ajouter le fait que les propriétés desdits matériaux (sels composés, aciers, réflecteurs) ne sont pas forcément bien connues sur les plages de température étudiées. La chimie du sel est complexe et peu caractérisée. L'ensemble des recherches repose sur des valeurs expérimentales provenant du faible nombre d'expérimentations ayant eu lieu et de relations d'interpolation empiriques.

La maintenance d'un MSFR est elle aussi problématique. Comme expliqué précédemment, le sel combustible, une fois activé, est émetteur de gamma très énergétique. Le combustible étant liquide, la distribution des radionucléides est étendue dans l'ensemble de la zone critique. Les opérations de maintenance du système peuvent nécessiter de se faire complètement à distance telles que prévues pour les réacteurs historiques américains, ajoutant une contrainte forte sur l'exploitation et devant se coupler avec le caractère li-

5. Composition isotopique

quide du combustible.

Des incertitudes sur l'approvisionnement en matière fissile, nécessaires au lancement du cycle des réacteurs MSFR, subsistent. Le lancer nécessite l'accumulation initiale d'une grande quantité de matière fissile. Une possibilité serait d'utiliser de l'uranium enrichi classiquement en uranium-235. Autrement, de l'uranium-233 ou du plutonium-239 pourrait être utilisé en utilisant la production des réacteurs actuels. Ce processus nécessite une planification sur le long terme et ne peut pas être déployé rapidement.

Enfin, le design thermohydraulique d'un réacteur à sels fondus reste encore en cours de développement. De nombreux paramètres doivent être pris en compte, tels que le positionnement des différents éléments, le nombre d'échangeurs de chaleur, la taille de la cuve, *etc.* Ces éléments sont soumis à une multitude de contraintes. En raison du manque de données concrètes, les plages de conception restent larges, ce qui complique encore davantage la définition d'un design optimal et viable pour ces réacteurs.

III Conception optimisée d'un MSFR

III.1 Objectifs

L'architecture actuelle du MSFR comporte une structure et des éléments de base. Cependant, il reste un objet conceptuel d'étude et continue d'évoluer. Au sein de l'unité de recherche MSFR du LPSC, des travaux sont menés dans l'objectif de fixer l'architecture de référence d'un tel réacteur. Partant de modèles physiques initiaux simplifiés en neutronique et thermohydraulique 0D et 1D, ainsi qu'un ensemble de contraintes et de données initiales, l'objectif est d'explorer les différentes configurations possibles pour le réacteur. Avec plusieurs centaines de paramètres, de variables d'entrée, et un nombre conséquent de relations physiques à considérer, la conception d'un MSFR de référence est un problème d'optimisation contraint multicritères, aussi dit multiobjectifs.

Un problème multicritère $(\mathcal{S})$ est défini de manière générale comme :

$$(\mathcal{S}) \quad \begin{cases} \min \mathbf{f}(\mathbf{x}) \\ \text{avec } \mathbf{x} \in \Omega \end{cases} \tag{1}$$

La fonction vectorielle $\mathbf{f}$ est défini par les fonctions objectifs f_i, $\mathbf{f} = (f_1, \ldots, f_M)$ $(\mathbf{f} : \mathbb{R}^N \to \mathbb{R}^M)$.

Les contraintes du problème sont prises en compte dans la définition de l'espace de décision Ω. Ω contient toutes les solutions $\mathbf{x}$ faisables, c'est-à-dire les solutions qui respectent les C_e contraintes d'égalité et les C_i contraintes d'inégalité définies par les fonctions contraintes h_i et g_j [20] :

$$\Omega = \left\{ \mathbf{x} \in \mathbb{R}^N \mid \forall i \in [\![1, C_e]\!],\ h_i(\mathbf{x}) = 0 \text{ et } \forall j \in [\![1, C_i]\!],\ g_j(\mathbf{x}) \leq 0 \right\} \tag{2}$$

L'espace des objectifs est introduit ici, noté $\mathcal{Y}$, il est l'image de l'espace de décision par $\mathbf{f}$:

$$\mathcal{Y} = \{\mathbf{f}(\mathbf{x}) \mid \mathbf{x} \in \Omega\} = \mathbf{f}(\Omega). \tag{3}$$

La recherche d'une architecture de référence, en raison du nombre élevé de paramètres et de la complexité des interactions entre ces paramètres, ne peut être réalisée analytiquement. C'est pourquoi des algorithmes d'optimisation sont utilisés. Un code d'optimisation multicritère, basé sur une approche thermohydraulique et neutronique, permet de modéliser l'espace des possibles pour le dimensionnement du circuit combustible d'un réacteur à sels fondus et d'étudier la sensibilité des différents paramètres sur la conception.

III.2 Tour d'horizon des algorithmes d'optimisation

Les algorithmes d'optimisation se divisent en deux catégories : les algorithmes déterministes et les algorithmes stochastiques.
Dans cette partie sont présentées les différences notables entre ces deux catégories. La présentation de certains algorithmes présents dans la bibliothèque python scipy.opitmize [21] est également détaillée.

III.2.1 Algorithmes déterministes

Un algorithme d'optimisation déterministe est un algorithme qui, lorsqu'il est exécuté sur les mêmes données et avec les mêmes paramètres, produit toujours la même solution. Les algorithmes d'optimisation déterministes suivent une méthode systématique et reproductible pour trouver la meilleure solution possible.

Les algorithmes d'optimisations locales par directions de descente font partie de cette catégorie. Ces derniers fonctionnent sur le principe suivant :

1. Une solution $\mathbf{x}_0$ est initialisée.
2. Une direction de descente $\mathbf{d}_k$, est recherchée telle que $lim_{t\to 0}\frac{\mathbf{f}(\mathbf{x}_k+t\mathbf{d}_k)-\mathbf{f}(\mathbf{x}_k)}{t} < 0$.
3. La solution est mise à jour l $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha\mathbf{d}_k$, où $\alpha \in \mathbb{R}$ est le pas de la descente.
4. Le processus est réitéré jusqu'à atteindre un critère d'arrêt.

Ces algorithmes ont comme inconvénient de converger vers une solution qui est optimale localement, mais pas globalement.

Plus généralement, les algorithmes déterministes sont très coûteux en ressources computationnelles pour des problèmes de grande taille (*i.e* lorsque l'espace des solutions est très grand).

La bibliothèque scipy.optimize contient, en plus des algorithmes de recherche de minimums locaux classiques de Newton et du gradient, un ensemble d'implémentations d'algorithmes d'optimisation globaux déterministes dont les grandes lignes de fonctionnement sont décrites ci-après. Ce sont ces derniers qui sont les plus convenus pour la recherche d'un optimum global.

L'**algorithme d'optimisation par force brute** cherche à minimiser une fonction sur un espace donné en calculant la valeur de la fonction à chaque intersection d'une grille multidimensionnelle. L'espace de départ est échantillonné de manière uniforme.

L'**algorithme d'optimisation DIRECT**, pour « Dividing Rectangles », fonctionne en divisant l'espace de recherche en hyperrectangles, évaluant systématiquement les sous-régions en fonction de leurs valeurs de fonction et de leur potentiel de réduction. Pour chaque hyperrectangle, DIRECT calcule un score basé sur la valeur de la fonction et le volume de l'hyperrectangle. À chaque itération, l'algorithme divise les hyperrectangles «prometteurs» (ceux qui présentent le score) en plus petits hyperrectangles. Les hyperrectangles de petits volumes avec de bons résultats (indicatif d'une exploitation locale) ou les hyperrectangles de plus grands volumes avec de moins bons résultats (indicatif d'une exploration globale) sont préférés selon un compromis établi par l'utilisateur [22].

L'**algorithme d'optimisation SHGO**, pour « Simplicial Homology Global Optimisation » [23], fonctionne sur le même concept que DIRECT : il découpe en plus petites subdivisions l'espace initial qu'il évalue progressivement. Il se différencie dans le fait que les espaces ne sont pas des hyperrectangles mais des complexes simpliciaux. Il subdivise

par itération les complexes en plus petits simplex, en se concentrant sur les régions prometteuses, tout en effectuant des recherches locales pour affiner la recherche du minimum global. Le processus se poursuit jusqu'à ce qu'un critère d'arrêt soit atteint.

III.2.2 Algorithmes stochastiques

Les algorithmes d'optimisation stochastiques utilisent des éléments de probabilité et d'aléatoire dans leur processus de recherche de solutions optimales. Contrairement aux algorithmes d'optimisation déterministes qui suivent un cheminement précis et reproductible, les algorithmes stochastiques introduisent une composante aléatoire dans leur fonctionnement. Les algorithmes stochastiques sont généralement moins susceptibles de rester piégés dans des optima locaux, car ils peuvent explorer une région différente de l'espace de manière aléatoire. Cela est possible au coût d'une moins bonne garantie de la qualité de la solution trouvée. Les résultats produits par les algorithmes stochastiques peuvent varier d'une exécution à l'autre. Le critère d'arrêt de recherche de solution n'est pas absolu.

La bibliothèque scipy.optimize contient un ensemble d'implémentation d'algorithmes globaux stochastiques d'optimisation dont les grandes lignes de fonctionnement sont décrites ci-après.

L'**algorithme d'optimisation par double recuit** (dual annealing) combine les principes du recuit simulé avec une exploration globale de l'espace de recherche. Cette approche stochastique dérive de la combinaison de la généralisation du « Classical Simulated Annealing » (CSA) et du « Fast Simulated Annealing » (FSA) [24]. Il débute par la sélection aléatoire d'un point de départ et d'une température initiale. À chaque itération, un voisin est généré aléatoirement et évalué selon une fonction de coût. Si ce voisin améliore la solution, il est adopté comme nouveau point de départ. En cas de dégradation de la solution, ce voisin peut être accepté avec une probabilité déterminée par une fonction exponentielle et de la température courante. Cette approche permet au système de se déplacer librement dans l'espace des solutions à haute température, explorant diverses configurations pour éviter les minima locaux. Au fil des itérations, la température est réduite de manière contrôlée, réduisant la probabilité d'accepter des solutions moins favorables et privilégie la convergence vers des minima globaux.

L'**algorithme d'optimisation par saut de bassin** (basin hopping) [25] est très semblable à l'algorithme d'optimisation par simple recuit (simple annealing). Il se distingue de ce dernier par l'ajout d'une étape de descente de gradient, à chaque itération, avant l'évaluation de la fonction de coût. Chaque nouvel état évalué est issu d'une optimisation locale préalable. Cette méthode vise à stabiliser le processus d'optimisation en comparant directement différents minima locaux entre eux plutôt qu'en explorant aléatoirement l'espace des solutions. Cette étape a aussi été ajoutée dans l'algorithme par double recuit décrit précédemment.

L'**algorithme d'optimisation par évolution différentielle** est un algorithme du groupe des **algorithmes évolutionnistes**.

Cette famille d'algorithmes regroupe des méthodes d'optimisation inspirées par les principes de la théorie de l'évolution. Une population initiale d'individus est créée. Chaque individu de la population est évalué en fonction de ses performances par rapport aux objectifs définis dans le problème à résoudre. Des individus sont sélectionnés sur leur performance dans la population actuelle pour devenir les parents de la génération suivante.

Le nombre d'individus dans la population est également crucial. Il définit la taille de l'échantillon à partir duquel les essais sont générés à chaque itération. Une population plus grande peut capturer une plus grande diversité de solutions, favorisant ainsi une exploration plus exhaustive de l'espace de recherche. Toutefois, une augmentation excessive du nombre d'individus peut entraîner des coûts computationnels plus élevés. Un équilibre entre la qualité de la solution et les ressources disponibles est nécessaire.

Des opérateurs de variation, tels que le croisement et la mutation, sont appliqués aux parents sélectionnés pour créer de nouveaux individus, appelés enfants. Le croisement implique le mélange du matériel génétique des parents pour produire une descendance, tandis que la mutation introduit des changements aléatoires dans le génotype de l'enfant. Il faut calibrer correctement les mutations pour parcourir l'espace des solutions et se stabiliser autour des minima. Ajouter la possibilité de croisement permet d'accélérer la convergence vers minimum, mais réduit l'exploration de l'espace des phases en créant une relation de dépendance entre les individus. Il existe un dilemme entre exploration (explorer l'espace de décision pour éviter les optima locaux) et exploitation (exploiter les meilleures solutions connues). Les opérateurs de sélection, de croisement, de mutation et de remplacement, sont ajustés pour équilibrer cet équilibre exploration-exploitation.

Les performances des enfants sont évaluées en fonction des objectifs du problème. Les enfants sont utilisés pour former une nouvelle population, qui remplace la population précédente. Ce processus permet à la population de s'adapter et d'évoluer au fil des générations. L'algorithme continue à générer de nouvelles populations et à évaluer leurs performances jusqu'à ce qu'un critère d'arrêt soit atteint. Ce critère peut être basé sur la performance des solutions ou sur le nombre d'évaluations effectuées.

L'algorithme d'évolution différentielle se distingue des algorithmes évolutionnistes traditionnels principalement par son mécanisme de croisement unique, où la différence entre deux individus de la population est utilisée pour perturber un troisième individu, favorisant ainsi la diversité génétique. Dans les algorithmes d'évolution différentielle, les individus sont représentés par des vecteurs de dimension D, où chaque composant du vecteur est une valeur d'une variable du problème à optimiser.

Une recherche exhaustive est généralement inenvisageable pour résoudre la plupart des problèmes d'optimisation. En effet, la dimension du problème à résoudre est trop importante pour permettre une résolution exacte en un temps raisonnable. Ainsi, les algorithmes stochastiques sont préférés pour la résolution de problèmes « réels ».

III.3 SONGe

Le code Système d'Optimisation Numérique par algorithme Génétique (SONGe), développé au CNRS, permet de faire de l'optimisation multicritère sous contraintes. Le code se scinde en deux parties. Une première partie rassemble une collection de classes qui définissent l'ensemble des paramètres des composants utiles à la réalisation d'un MSFR ainsi que les relations physiques liant ces différents paramètres. Un algorithme d'optimisation constitue la seconde partie du code.

Des plages de valeurs ou des valeurs limites aux variables sont initialement désignées. Ces critères sont réglés par « avis d'expert », ce qui a pour effet de réduire l'espace des décisions, accélérant la convergence de l'algorithme.

Table 1 – Tableau des variables.

Composant	Variable	Min	Max
Échangeur intermédiaire	Température d'entrée du fluide froid [K]	730	800
Cavité centrale	Température de sortie [K]	850	950
Cavité centrale	Débit total de sel combustible [kg/s]	1000	10000
Réacteur	Débit total de fluide froid [kg/s]	1000	10000
Cavité centrale	Facteur de forme du coeur [H/D]	0.5	2
Échangeur intermédiaire	Diamètre calandre [m]	0.2	1.5
Tuyaux	Longueur verticale tuyau haut [m]	0.05	3
Tuyaux	Diamètre [m]	0.1	0.4
Tuyaux	Diamètre pompe réacteur [m]	0.1	1
Tuyaux	Angle tuyau haut	0	20
Tuyaux	Angle tuyau bas	0	20
Cavité centrale	Diamètre du coeur [m]	0.2	3
Cavité centrale	Nombre de secteurs	6	6
Cavité centrale	Puissance spécifique [MW/m3]	10	500
Pompe primaire	Vitesse de rotation [tr/min]	10	2000

En partant de ces variables, l'ensemble des valeurs des paramètres du réacteur, qui respecte les relations physiques 0D et 1D implémentées, peut être retrouvé. Certains paramètres sont soumis à des contraintes. L'objectif est d'obtenir le jeu de variables qui les respecte le mieux [26].

Comme énoncé précédemment, le problème posé est un problème multiobjectifs dans lequel toutes les relations physiques doivent être respectées pour assurer la viabilité physique du réacteur, tout en répondant au mieux aux contraintes définies par l'utilisateur. Cependant, lors de la minimisation de certaines grandeurs pour la conception, des compromis sont souvent nécessaires. Par exemple, limiter le volume de sel combustible et maximiser la puissance ne peuvent se faire sans compromis. Mathématiquement, cela s'exprime par :

$$\nexists \mathbf{x}^* \in \Omega \mid \forall \mathbf{x} \in \Omega, \forall i \in [\![1; M]\!] \ f_i(\mathbf{x}^*) \leq f_i(\mathbf{x}) \tag{4}$$

Dès lors, deux approches sont possibles. En cas d'incapacité à ordonner et à prioriser la réalisation des objectifs, il est nécessaire d'explorer l'ensemble des solutions possibles. Le travail se concentre alors sur la résolution du problème multicritère afin de tracer l'ensemble des solutions pour lesquelles aucune amélioration dans un critère ne peut être réalisée sans détériorer au moins un autre critère. Ces solutions sont les compromis optimaux entre plusieurs objectifs concurrents. Dans l'espace des objectifs, cet ensemble est nommé le front de Pareto. Formellement, il s'écrit :

$$\mathcal{P}(\mathcal{Y}) = \{\mathbf{y}' \in \mathcal{Y} \mid \forall \mathbf{y} \in \mathcal{Y}, \ \mathbf{y}' \prec \mathbf{y}\} \tag{5}$$

où $\mathbf{y}' \prec \mathbf{y}$ signifie que $\mathbf{y}'$ domine $\mathbf{y}$, c'est-à-dire :

$$\mathbf{y}' \prec \mathbf{y}' \Leftrightarrow \forall i \in [\![1, k]\!], \ \mathbf{y}'_i \leq \mathbf{y}_i \wedge \exists j \in [\![1, \ldots, k]\!], \ \mathbf{y}'_j < \mathbf{y}_j. \tag{6}$$

La zone de Pareto, notée Ω^*, est définie dans l'espace de décision comme suit :

$$\Omega^* = \{\mathbf{x}^* \in \Omega \mid \forall \mathbf{x} \in \Omega, \ \mathbf{x}' \prec_{\mathbf{f}} \mathbf{x} \vee \mathbf{x}' \sim_{\mathbf{f}} \mathbf{x}\} \tag{7}$$

Alors :

$$\mathcal{P}(\mathcal{Y}) = \{\mathbf{f}(\mathbf{x}^*) \mid \mathbf{x}^* \in \Omega^*\} = \mathbf{f}(\Omega^*) \tag{8}$$

Si prioriser les objectifs est possible, des préférences vis-à-vis des compromis à réaliser peuvent être formulées a priori. Le problème multicritère peut être transformé en problème en monocritère.

Dans le cadre de SONGe une méthode de formulation a priori des préférences a été choisie pour gérer l'optimisation. À cet effet, il a été choisi une méthode par agrégation pondérée [27] :

$$(\mathcal{S}) \quad \begin{cases} \min \mathbf{f}(\mathbf{x}) \\ \text{avec } \mathbf{x} \in \Omega \end{cases} \rightarrow (\mathcal{S}_0) \quad \begin{cases} \min F(\mathbf{x}) = \sum_{i=1}^{M} \alpha_i f_i(\mathbf{x}) \\ \text{avec } \mathbf{x} \in \Omega \end{cases} \tag{9}$$

Cela permet de passer d'un problème multicritère à un problème monocritère, facilement traitable par les algorithmes présentés jusqu'à maintenant. Sa plus grande qualité reste que les points obtenus par optimisation d'agrégation pondérée appartiennent au front de Pareto, d'après la démonstration de Yair Censor en 1977 [28]. Cependant, cette méthode ne permet pas l'obtention de l'ensemble des points du Front de Pareto uniquement dans le cas où le problème traité est convexe [29] [28].

Initialement dans SONGe $\forall i,\ \alpha_i = 1$. Il était impossible d'accéder aux poids ou de les modifier.

Dans les faits, SONGe dispose d'une liste de contraintes. À chaque contrainte était associé un coût, représenté par l'exponentielle de l'écart à la contrainte notée, c_i divisé par une tolérance notée σ_i. Une évaluation du réacteur était obtenue en utilisant la formule suivante :

$$F(\mathbf{x}) = \sum_{i=1}^{M} \exp\left(\frac{\delta_i(x_i - c_i)}{\sigma_i}\right) \text{ avec } \delta_i = \begin{cases} +1 \text{ si } c_i \text{ est une contrainte maximum} \\ -1 \text{ sinon.} \end{cases} \tag{10}$$

Les configurations $\mathbf{x}$ telles que $F(\mathbf{x})$ soit minimum sont recherchées [30]. L'ensemble des contraintes sont listés dans le tableau suivant.

Il est possible d'étendre la liste des contraintes sur d'autres paramètres du réacteur et ainsi de les prendre en compte dans le calcul du coût.

L'algorithme d'optimisation initialement implémenté en Java dans le code SONGe était un algorithme génétique.

Les algorithmes génétiques font partie des algorithmes évolutionnistes. Chaque individu est représenté par un ensemble de gènes sous forme de string binaire représentant ou non la possession de chromosomes (variables de décision), formant un génotype. Le processus de mutation est alors représenté par la bascule aléatoire d'un chromosome enfant de 0 à 1 ou inversement, indépendamment du patrimoine génétique des parents.

Dans le code SONGe, le critère d'arrêt est laissé à l'avis d'expert de l'utilisateur. L'optimisation s'arrête quand l'utilisateur détermine que le minimum local atteint est proche du minimum global. En effet, l'objectif principal est de caractériser l'espace des solutions et d'identifier de potentielles relations entre les contraintes. La dispersion des solutions pour s'assurer de leur cohérence est aussi étudiée. Une faible dispersion est synonyme d'une stabilité des solutions. Trouver l'optimum absolu est secondaire.

Table 2 – Tableau des contraintes.

Composant	Contrainte	Min	Max
Tuyaux	Vitesse maximum dans les tuyaux [m/s]		5
Échangeur intermédiaire	Vitesse calandre [m/s]		2
Échangeur intermédiaire	Vitesse tube [m/s]		5
Réacteur	Diamètre réacteur [m]		5
Cavité centrale	Volume de sel combustible [m3]		5
Tuyaux	Position y échangeur [m]	0.2	
Cavité centrale	Déséquilibre de température [K]		0.01
Cavité centrale	Puissance demandée [MW]	300	300
Cavité centrale	Déséquilibre de Puissance [MW]		0.0001
Tuyaux	Delta Pompe		0.001
Pompe primaire	Vitesse spécifique		100
Pompe primaire	Pertes de charges [Pa]		1000000
Cavité centrale	Volume du coeur [m3]	1	
Cavité centrale	Fraction de sel combustible en coeur	0.5	

Un problème du code SONGe est lié à la notion d'hyperstatisme[6] du modèle. Pour pallier à ce phénomène certaines relations dans le code restent libres afin de rajouter des degrés de liberté au modèle numérique. Par exemple, la puissance dégagée dans le cœur doit valoir celle échangée, elle n'est pas fixée, mais son respect est vérifié a posteriori. Si elle ne l'est pas, cela montre la défaillance de la solution proposée. Ainsi, un degré de liberté assez grand pour que l'algorithme puisse librement évoluer est garanti.

La version originale de SONGe était entièrement implémentée en Java. Dans un souci de performance, la partie algorithmique d'optimisation de SONGe (choix des configurations) a récemment été transférée en Python pour sa librairie scipy.optimize [21], tandis que l'évaluation (calcul du coût des configurations) reste en Java pour la rapidité de calcul. L'utilisation de tubes nommés (communément appelés named pipes) assure la communication des différentes données entre Java et Python.

IV L'optimisation sachant optimiser

IV.1 Discrimination des algorithmes d'optimisations

Jusqu'à présent, l'algorithme d'optimisation utilisé dans le cadre de SONGe était un algorithme génétique. Le passage de la partie algorithmique de python sur SONGe a permis d'accéder à un ensemble déjà implémenté d'algorithmes d'optimisation dans la bibliothèque scipy.optimize. Cette bibliothèque contient plusieurs algorithmes stochastiques ou déterministes, adaptés à la recherche de minimums globaux, présentés précédemment.

Il convient de choisir le meilleur algorithme d'optimisation parmi ceux disponibles. Pour ce faire, il est nécessaire de fixer des métriques sur lesquelles les différents algorithmes seront comparés à partir des résultats de différents calculs indépendants relancés plusieurs fois. Pour les algorithmes stochastiques, les métriques utilisées durant l'étude sont, à temps de calcul fixe, construits sur la dispersion des résultats avec la fonction de distribution cumulée (FDC). Si besoin, l'écart-type entre les résultats fournis, la moyenne, la médiane,

6. Phénomène de surcontrainte du modèle

la valeur à 33 %, 66 %, et à 95 % de la FDC, le meilleur et le pire résultat fourni sont utilisés pour ordonner les performances. La mesure du nombre d'évaluations ou d'itérations n'est pas utilisée ici. Effectivement, travailler à temps de calcul fixe est plus significatif, car les machines de calcul sont identiques et la fonction à évaluer par le code n'est pas destinée à changer radicalement dans sa forme.

Concernant les algorithmes déterministes, ils fournissent un résultat unique et reproductible (principe de l'algorithme déterministe) pour un temps donné de calcul. Compte tenu de la taille de l'espace des candidats (hyperrectangle de dimension égale à 15), il sera rapidement démontré que ces candidats ne sont pas capables de fournir des solutions satisfaisantes.

Toutefois, chaque algorithme possède des avantages en fonction du problème à optimiser. C'est le principe du « No Free Lunch Theorem » [31] introduit par David H. Wolpert et William G. Macready en 1995. Un algorithme d'optimisation ne présentera jamais des performances égales sur tous les problèmes. Selon le « No Free Lunch Theorem », il n'existe pas d'algorithme qui surpasse les autres sur tous les problèmes possibles. Il faut garder à l'esprit que les comparaisons d'algorithmes sont valables pour un problème donné, à jeu de paramètres donné. Pour un problème différent, peut-être qu'un autre algorithme sera plus efficace, ou un autre jeu de paramètres pour le même algorithme sera meilleur.

Un unique problème sera fixé dans SONGe, sur lequel l'ensemble des algorithmes sera testé. Le problème ne changera pas de manière profonde dans la suite d'utilisation prévue. Les bornes et tolérances seront amenées à changer, mais la structure du problème d'optimisation restera construite sur la minimisation d'une agrégation de coûts pondérés.

Les performances des algorithmes déterministes seront d'abord présentées, suivies par l'examen des algorithmes stochastiques. En conclusion, un algorithme sera sélectionné et les améliorations envisageables seront discutées.

IV.1.1 Algorithme déterministe

Pour la qualification des algorithmes déterministes, il n'est pas nécessaire d'accumuler de la statistique.

Pour l'algorithme par force brute, chacune des 15 dimensions du problème (15 intervalles de variation de nos paramètres) est échantillonnée en 2 ou 3 points. Cela représente respectivement 2^{15} et 3^{15} possibilités évaluées. Les résultats sont compilés dans la table 3.

Table 3 – Résultats pour l'algorithme par force brute.

Nombre de points par dimension	Nombre d'évaluations	Temps de calcul (s)	Coût minimum obtenu
2	3.2768e+4	1.659643	9.766847e+5
3	1.434891e+7	4.142305e+2	6.857281+10
4	1.073741e+9	2.994987e+4	1.772556

Les résultats sont cohérents. Si le temps de calcul est considéré comme majoritairement dédié au calcul des configurations et à l'évaluation de leurs coûts, alors le rapport « nombre d'évaluations » /« temps » devrait être sensiblement constant, ce qui est le cas avec un rapport de l'ordre de 3.10^5. Ainsi, échantillonner à raison de 10 points par variable reviendrait à faire l'évaluation de 10^{15} points, soit, en ordre de grandeur, un calcul de 3.10^{10}s. Même en parallélisation sur un ensemble de 1000 processeurs, il resterait 3.10^7

secondes de calcul, soit plus d'une année de calcul sans arrêt. Impossible d'envisager de trouver une solution avec une telle méthode.

Les paramètres de l'algorithme Direct ont été rapidement configurés de sorte à avoir les meilleurs résultats pour 60 s, 180 s et 600 s de calcul. Le calcul est arrêté de force une fois le temps écoulé, le résultat obtenu est alors un minimum local et non global. Les résultats sont compilés dans la table 4 :

Table 4 – Résultats pour l'algorithme Direct.

Temps de calcul (s)	Coût minimum obtenu
60	3.600263e+4
180	2.572585e+4
600	2.572585e+4

L'algorithme SHGO ne permet pas d'obtenir de résultat intermédiaire. Il ne s'arrête pas avant d'avoir trouvé le minimum global. Dans ces conditions, le temps de calcul est trop important (plusieurs jours sans résultat).

Conclusion : Une limitation de la qualité des résultats, liée au coût computationnel élevé des méthodes déterministes, est bien rencontrée. Bien que ces méthodes permettent d'obtenir à coup sûr le minimum global, elles sont impraticables dans les faits au vu du temps de calcul nécessaire.

IV.1.2 Comparaison des algorithmes stochastiques

Les paramètres de l'algorithme par double recuit sont rapidement configurés de sorte à avoir les meilleurs résultats pour 60 s, 180 s et 600 s de calcul. L'algorithme par double recuit dans l'état actuel de la bibliothèque scipy.optimize autorise un appel à une fonction de rappel (« callback ») que dans le cas où la configuration évaluée améliore la précédente meilleure configuration. Des modifications dans le code source ont été nécessaires pour permettre un appel de la fonction de rappel à chaque évaluation d'une solution, afin de pouvoir définir un critère d'arrêt précis basé sur le temps écoulé.

La figure 1 présente l'histogramme de la répartition des coûts pour l'algorithme par double recuit avec un découpage des coûts compris entre 0 et 6 en 20 canaux. Les barres d'incertitudes apparaissent en bleu foncé et valent $\frac{1}{\sqrt{n}}$, avec n le nombre de résultats obtenus dans le canal concerné. La FDC est affichée en rouge, elle vaut la somme cumulée des nombres d'occurrences normalisés. Des repères pour indiquer le seuil des 66% et 95% du jeu de donnée sont indiqués en pointillé respectivement vert et orange.

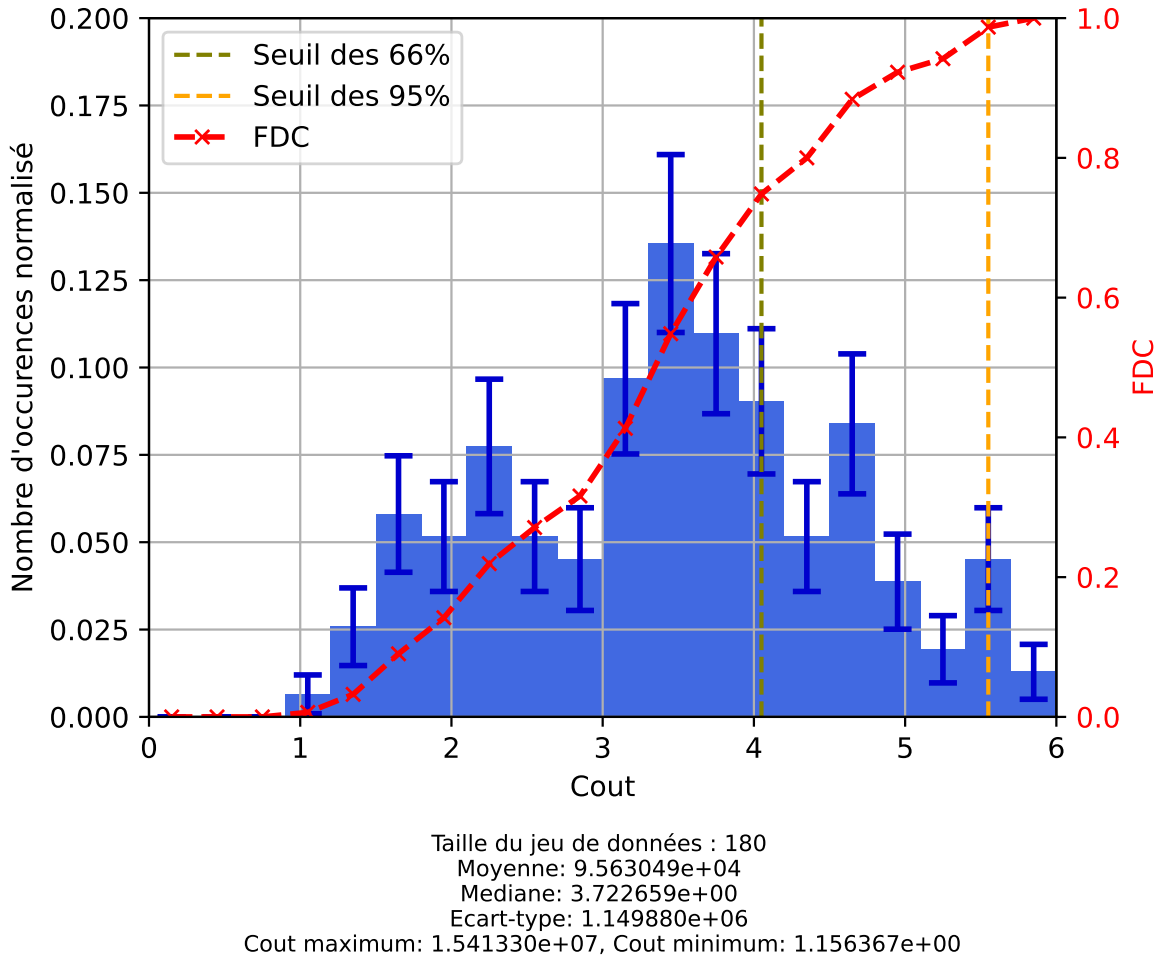

FIGURE 1 – Histogramme de la répartition des coûts pour l'algorithme par double recuit pour 60 s de temps de calcul.

Afin de pouvoir facilement comparer les performances des différents algorithmes, le plus simple et d'afficher directement sur la même figure les FDC correspondantes. Plus la pente de la FDC est raide, meilleur est la précision de l'algorithme considéré. Plus la FDC couvre des coûts faibles, meilleure est la justesse de l'algorithme considéré. Dans les figures 2, 3 et 4 sont affichées les FDC des résultats des calculs réalisés en respectivement 60 s, 180 s et 600 s, en utilisant l'algorithme par double recuit, l'algorithme par saut de bassin et l'algorithme par évolution différentielle, proposés dans la bibliothèque scipy.optimize [21]. En abscisse des figures, on retrouve le coût tandis qu'en ordonnée se trouve la valeur de la FDC.

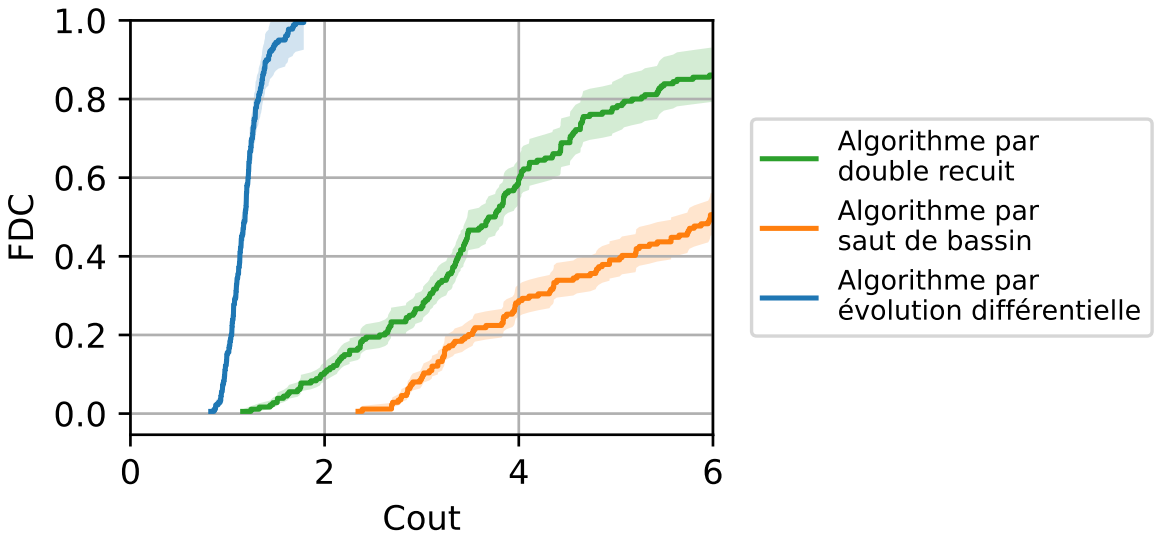

FIGURE 2 – Comparaison des FDC par algorithmes en fonction du coût pour 60 s de calcul.

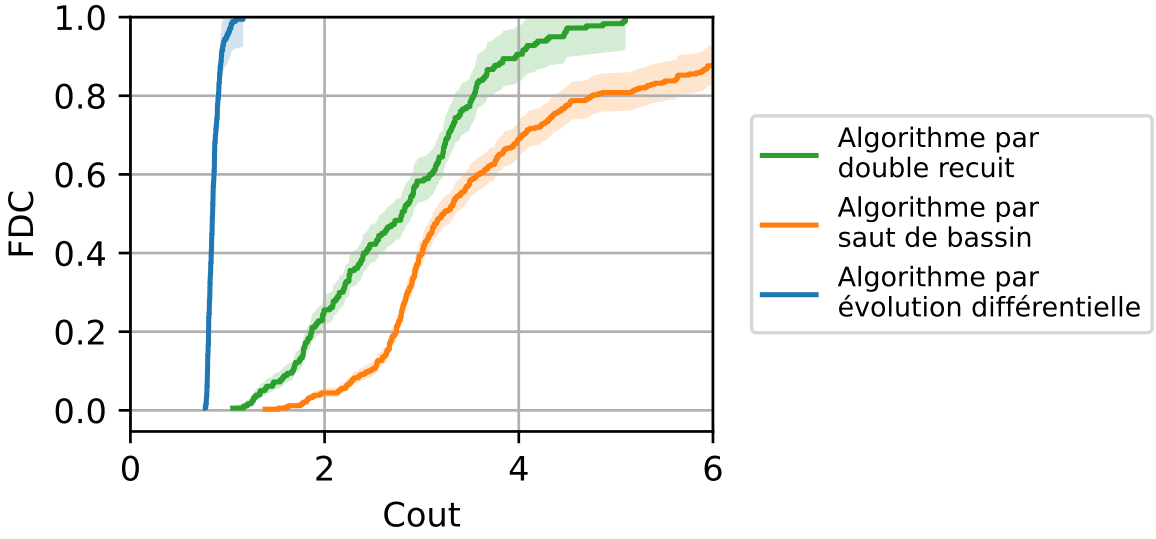

FIGURE 3 – Comparaison des FDC par algo en fonction du coût pour 180 s de calcul.

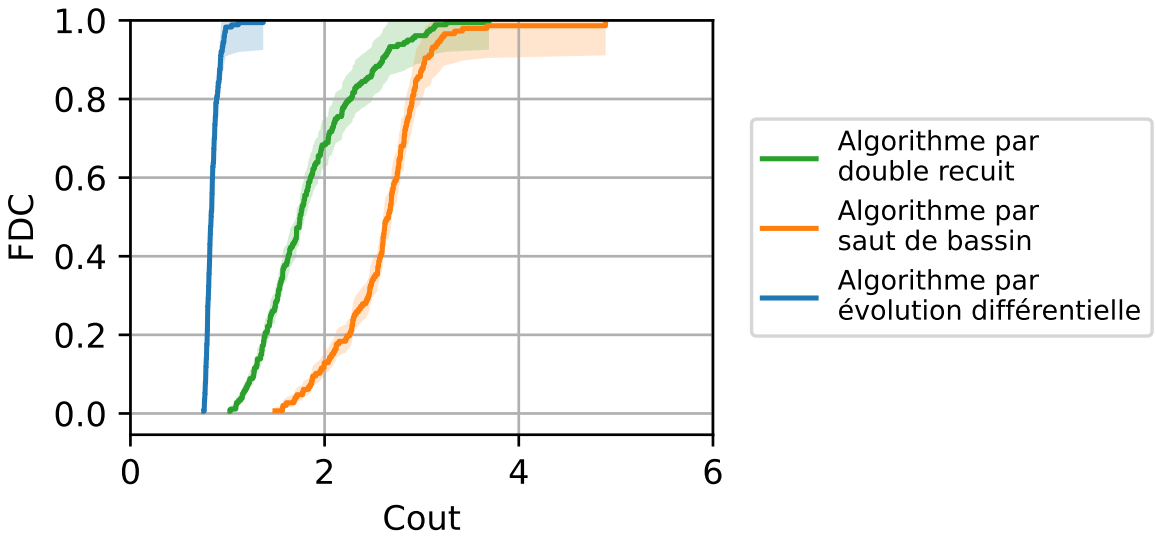

FIGURE 4 – Comparaison des FDC par algo en fonction du coût pour 600 s de calcul.

Les incertitudes affichées en transparence sur les figures 2, 3 et 4 sont calculées comme $\frac{1}{\sqrt{n}}$, où n est le nombre de calculs indépendants dont le résultat est inférieur ou égale au coût considéré.

Conclusion : Après analyse, l'algorithme d'évolution différentielle fournit les meilleurs résultats, avec les coûts les plus bas et un faible écart-type (comme l'indique la pente raide de la FDC), quel que soit le temps de calcul. Conserver un algorithme évolutionniste pour le fonctionnement de SONGe est justifié. L'algorithme d'évolution différentielle proposé dans la bibliothèque scipy.optimize apparaît comme le candidat idéal pour répondre aux exigences de SONGe sur tous les points. Cet algorithme se distingue par ses performances supérieures sur le problème considéré, avec des écarts significatifs par rapport aux autres méthodes, ce qui renforce le choix de son adoption.

IV.1.3 Algorithme d'optimisation par évolution différentielle

L'algorithme par évolution différentielle ayant été choisi comme nouvel outil d'optimisation de SONGe, cette partie présente plus détail le fonctionnement d'un tel algorithme et les paramètres qui le caractérisent.

L'évolution différentielle est une méthode algorithmique utilisée pour résoudre des problèmes d'optimisation globale, introduite par Rainer Storn et Kenneth Price en 1997 [32]. À chaque itération, l'algorithme opère une mutation sur chaque solution candidate en la combinant avec d'autres candidats pour former un nouvel essai. C'est cet algorithme d'optimisation qui se rapproche le plus de l'ancien algorithme génétique implémenté en Java. En effet, ils sont tous deux issus de la grande famille des algorithmes évolutionnistes.

Plusieurs stratégies sont disponibles pour générer de nouveaux individus [33], chacune adaptée à des types de problèmes spécifiques. La stratégie « best1bin » est souvent recommandée comme point de départ pour aborder divers problèmes d'optimisation. Dans cette méthode, deux membres de la population sont choisis de manière aléatoire. Leur différence est alors utilisée pour muter le meilleur individu de la population (le « best » dans « best1bin ») afin de créer un vecteur d'essai : $\mathbf{x}_{\text{essai}} = \mathbf{x}_{\text{best}} + F \times (\mathbf{x_i} - \mathbf{x_j})$ avec F le

facteur de mutation. Plusieurs autres stratégies alternatives existent et sont documentées dans la littérature [34].

Le processus continue par la sélection aléatoire d'un individu. Ses paramètres sont modifiés, en suivant un ordre aléatoire, par ceux provenant du vecteur d'essai ou de l'individu sélectionné, selon une distribution binomiale (le « bin » dans « best1bin »). Un nombre aléatoire compris entre [0, 1[est généré, si ce nombre est inférieur à une constante de recombinaison, le paramètre est remplacé par celui de la différence. Dans le cas contraire, il conserve la valeur du candidat sélectionné. Le dernier paramètre est toujours remplacé par celui du vecteur d'essai. Une fois ce nouveau vecteur construit, il est jugé selon sa performance qui est évaluée par son coût dans notre cas. Si le vecteur d'essai montre une performance supérieure à celle du candidat initial, il le remplace, garantissant ainsi une amélioration continue des solutions au cours du processus d'optimisation [33].

Pour l'algorithme d'évolution différentielle, deux paramètres clé influencent significativement le comportement de l'algorithme : le facteur de mutation F et le taux de recombinaison CR. Le paramètre F, souvent compris entre 0 et 2, contrôle l'amplitude de la différence utilisée pour générer les essais. Une valeur élevée de F favorise une exploration plus large de l'espace de recherche, tandis qu'une valeur plus faible donne des mutations plus subtiles. Quant à CR, qui détermine la proportion de paramètres remplacés par ceux de la différence, il influence directement la diversité génétique de la population et la convergence de l'algorithme.

IV.2 Paramétrisation de la méthode d'optimisation

Une fois le choix de l'algorithme d'optimisation effectué, il reste à correctement paramétrer ce dernier et, plus largement, à construire l'ensemble du processus d'optimisation. Dans cette partie est abordé l'intérêt d'effectuer plusieurs boucles d'optimisation, la recherche de la taille de population ainsi que le temps de calcul idéal. Elle traite aussi de l'intérêt de conserver les optimisations ayant conduit à des itérations sur place. Enfin, la détermination de la méthode d'initialisation la plus efficace est présentée.

IV.2.1 Boucler ou ne pas boucler

Initialement, afin de suivre la progression de la recherche, l'optimisation s'arrêtait toutes les 1000 itérations. Le meilleur individu était alors conservé et réinjecté dans l'optimisation suivante. Ce procédé de boucle avait lieu 64 fois. Un défaut de cette méthode est qu'elle supprime l'ensemble de la population qui avait jusqu'alors profité de l'évolution. Une très bonne diversité dans la population totale était maintenue, limitant ainsi un blocage potentiel dans des minima locaux. En revanche, il entraînait une perte de temps énorme, car le meilleur individu qui profitait de l'évolution n'avait plus à disposition pour ces mutations et croisements des individus déjà sélectionnés lors des itérations précédentes.

Pour suivre l'évolution de la meilleure solution, il est possible d'utiliser une fonction de rappel appelée à chaque itération de l'optimisation.

Il était nécessaire d'évaluer si le rapport entre bénéfice et coût d'un ajout de diversité par suppression de la population en boucle était avantageux.

L'étude effectuée a conduit à la production de la figure 5 représentant la répartition des coûts pour une optimisation avec une population de 8D individus, où D est le nombre de dimensions du problème, et 600 secondes de calcul.

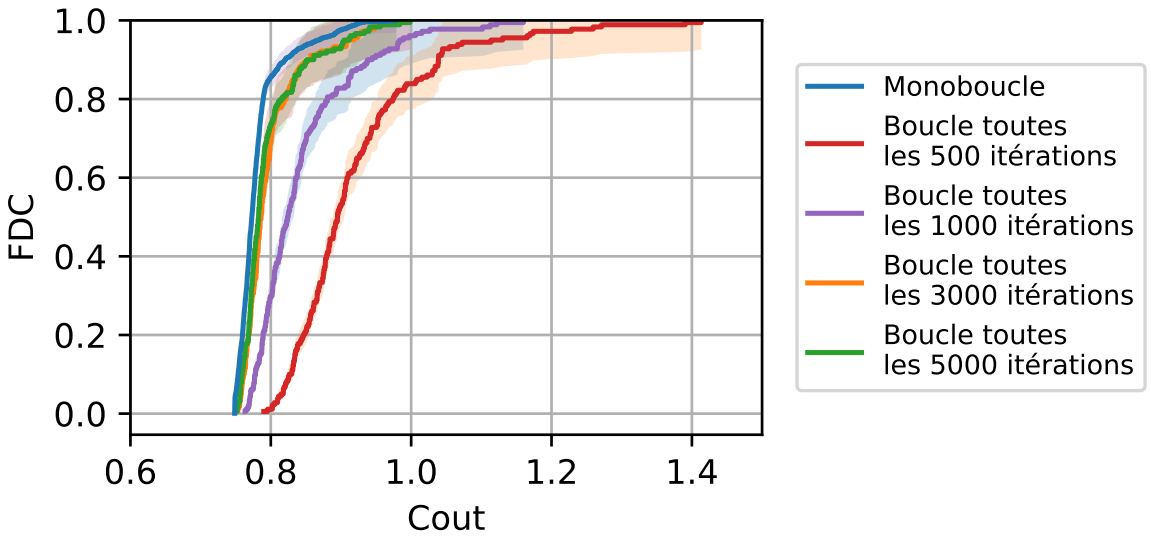

FIGURE 5 – Comparaison des FDC pour différentes taille de boucles pour une population valant 8D et 600 s de calcul.

Conclusion : Le fonctionnement en monoboucle produit de meilleurs résultats vis-à-vis d'une méthode par boucle. Les configurations trouvées sont moins coûteuses et plus centrées. La conservation d'une population stable est plus avantageuse qu'un fort brassage génétique.

Il est aussi possible de garantir une bonne diversité peut aussi se faire en augmentant la taille de la population ou en diminuant le facteur de croisement. Ainsi, le nombre d'individus mutants sera réduit et la diversité initiale sera conservée plus longtemps au prix de ralentir la convergence.

IV.2.2 Population idéale

Le nombre d'individus actifs dans l'algorithme d'évolution différentielle est probablement le paramètre le plus important. Il touche à la fois à la diversité de la population garantissant une résilience vis-à-vis des minima locaux tout en pesant sur le coût computationnel du calcul. Correctement le fixer est un objectif important. Une population comprise entre 3D et 10D, avec D le nombre de dimensions du problème, est conseillée par Rainer Storn et Kenneth Price [32]. Pour des problèmes de grandes dimensions (*i.e* plus de 5) 10D n'est pas souvent l'optimal, un peu moins est conseillé. Dans notre cas D=15, c'est le nombre de variables libres du problème.

Les performances de l'algorithme par évolution différentiel ont été mesurées pour différentes tailles de populations et différents temps de calculs. Les résultats sont présentés dans les figures 6, 7 et 8.

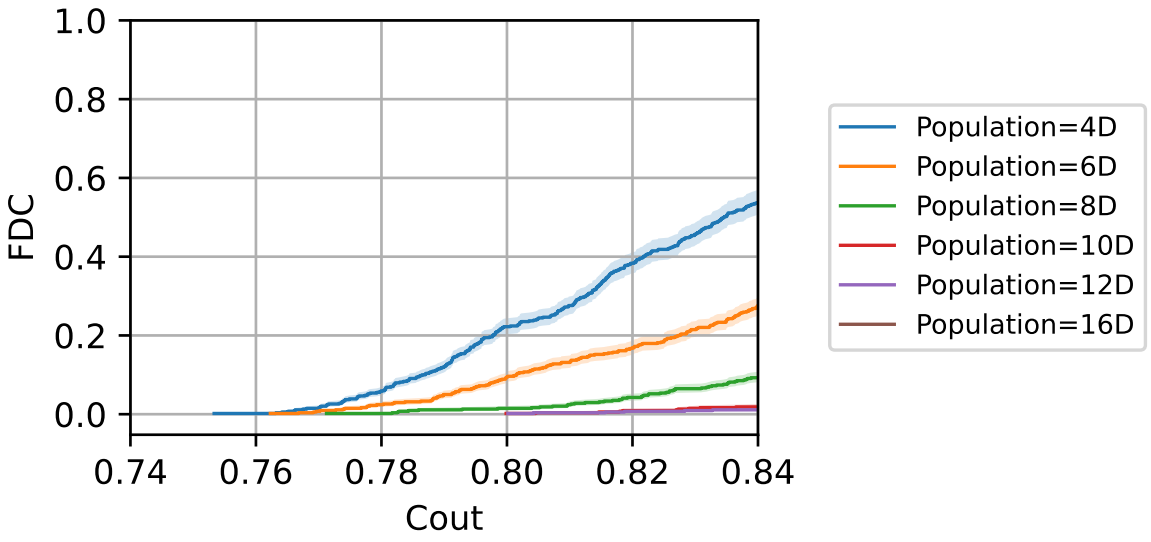

FIGURE 6 – Comparaison des FDC en fonction de la taille de la population pour 180 s de calcul.

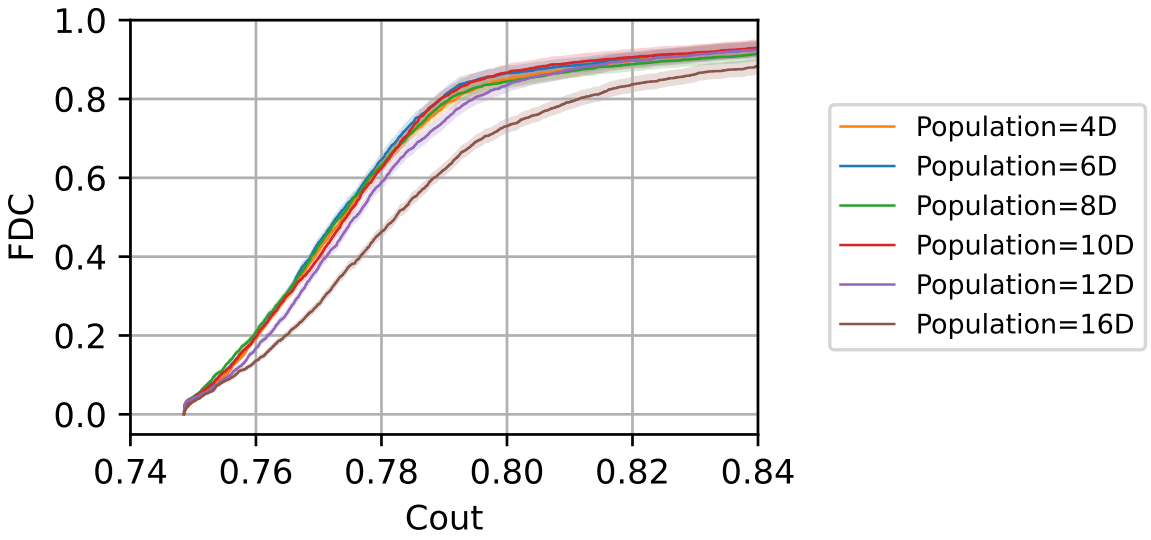

FIGURE 7 – Comparaison des FDC en fonction de la taille de la population pour 600 s de calcul.

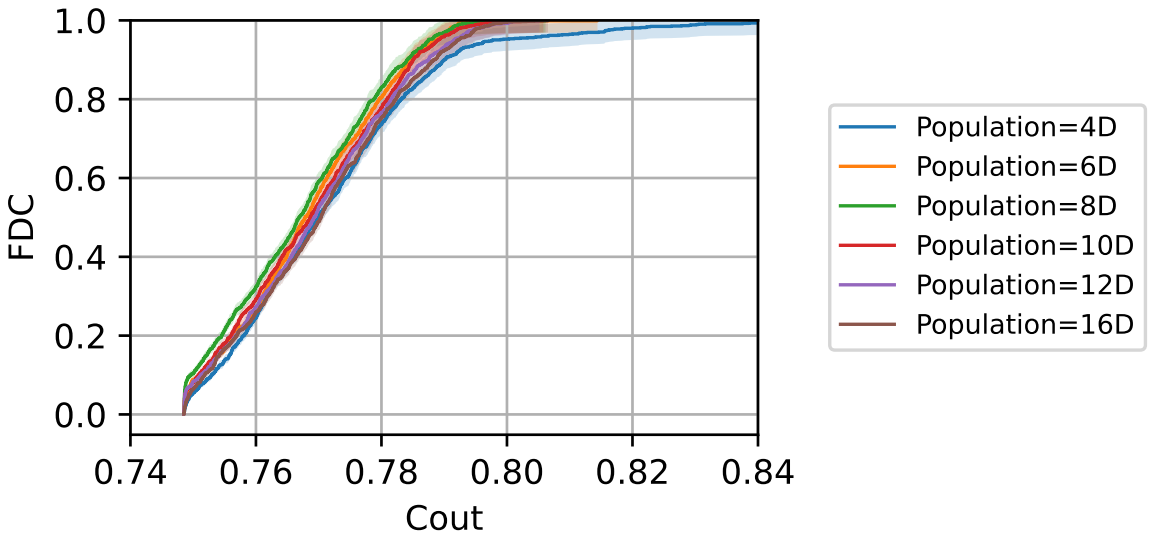

FIGURE 8 – Comparaison des FDC en fonction de la taille de la population pour 3600 s de calcul.

Conclusion : Pour chaque durée de calcul, une taille de population se révèle plus appropriée que les autres, illustrant le principe énoncé par le « No Free Lunch Theorem » [31]. Pour des temps de calcul important, une population de taille égale à 4D n'est pas adaptée car l'optimisation s'arrête avant la fin du temps de calcul. Le critère de convergence est atteint (l'écart type de l'ensemble des individus est inférieur à 1e-12). La diversité de la population n'est pas suffisante pour assurer un résultat satisfaisant. Finalement, pour un temps de calcul compris entre 60 s et 540 s, une population de 4D individus est conseillée, de 540 s à 1200 s de temps de calcul une population de 6D individus est plus adaptée, ce nombre passe à 8D individus au-delà de 3600 s de temps de calcul. Cela reste en accord avec les résultats des travaux détaillés effectués par Kenneth Price [35] sur la caractérisant des paramètres de l'algorithme par évolution différentielle.

Une taille de population plus grande pour un temps de calcul supérieur à 3600 s est potentiellement plus adaptée. Une étude supplémentaire serait nécessaire pour le déterminer.

IV.2.3 Temps de calcul idéal

Un temps de calcul plus long permet généralement d'obtenir un résultat de meilleure qualité. Cependant, il nécessite une mobilisation prolongée des ressources informatiques et peut entraîner une perte en termes de répétabilité statistique. L'objectif de cette section est de déterminer le temps de calcul optimal afin de maximiser le compromis entre la précision des résultats et la robustesse statistique. En d'autres termes, cette partie vise à répondre à la question suivante : est-il plus avantageux d'effectuer un unique calcul de 3600 secondes ou de réaliser dix calculs de 600 secondes chacun ?

La figure 9 présente le résultat de l'étude avec la taille de population la plus adaptée pour chaque temps de calcul.

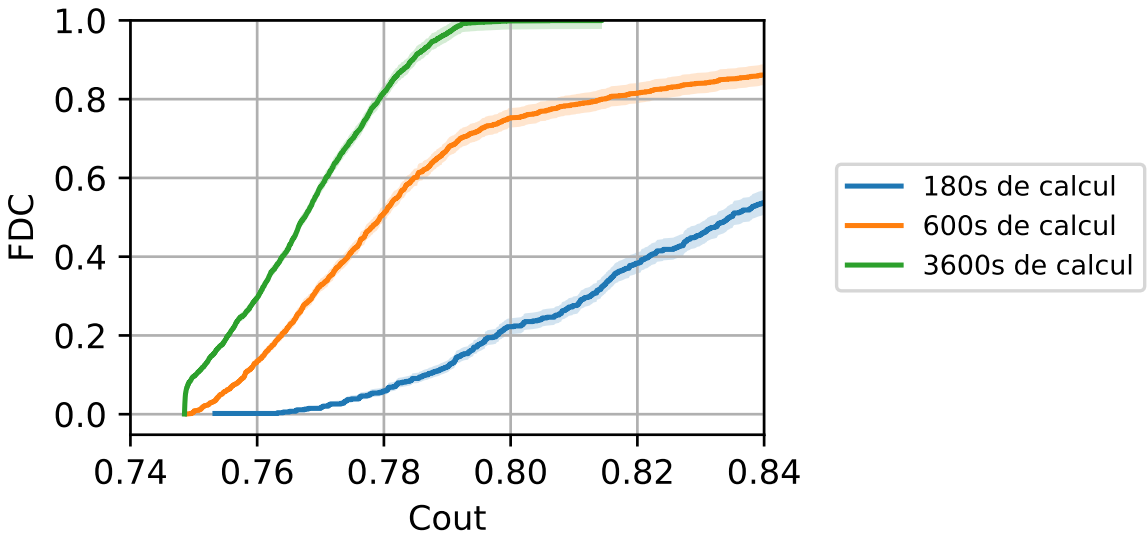

FIGURE 9 – Comparaison des FDC par temps de calcul avec la population la plus adaptée pour chaque temps.

Pour 3600 s de calcul, la FDC n'est pas continue mais marque une discontinuité au niveau d'un coût égale à 0.75. Cela semble marquer l'obtention d'un coût minimum global sur lequel une part des calculs optimisations converge. Une représentation, pour 3600 s et 600 s de temps de calcul, sous forme d'histogramme permet une meilleure étude du phénomène.

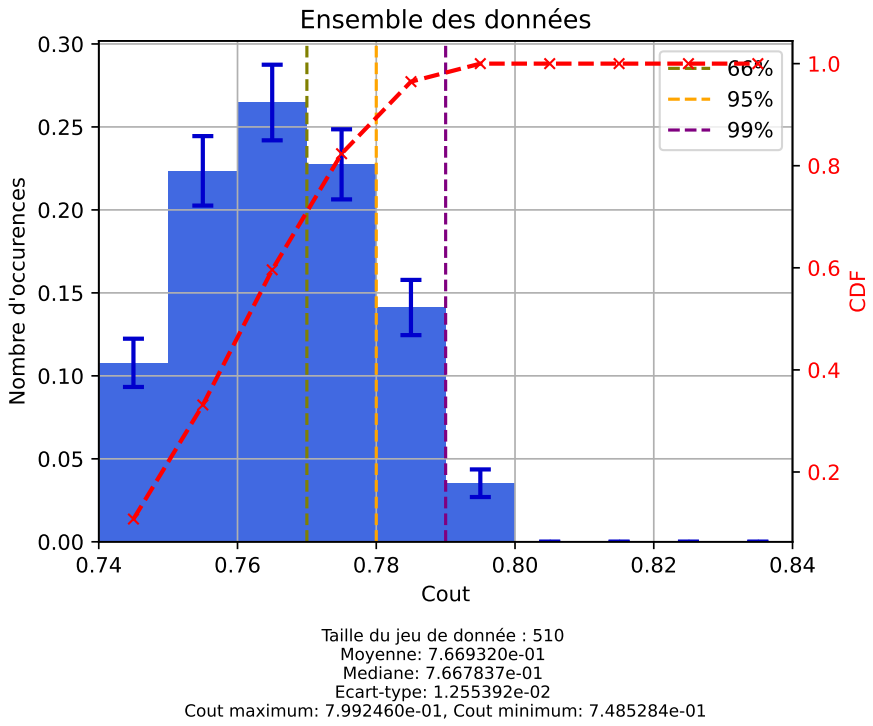

FIGURE 10 – Histogramme de la répartition des coûts pour 3600 s de calcul.

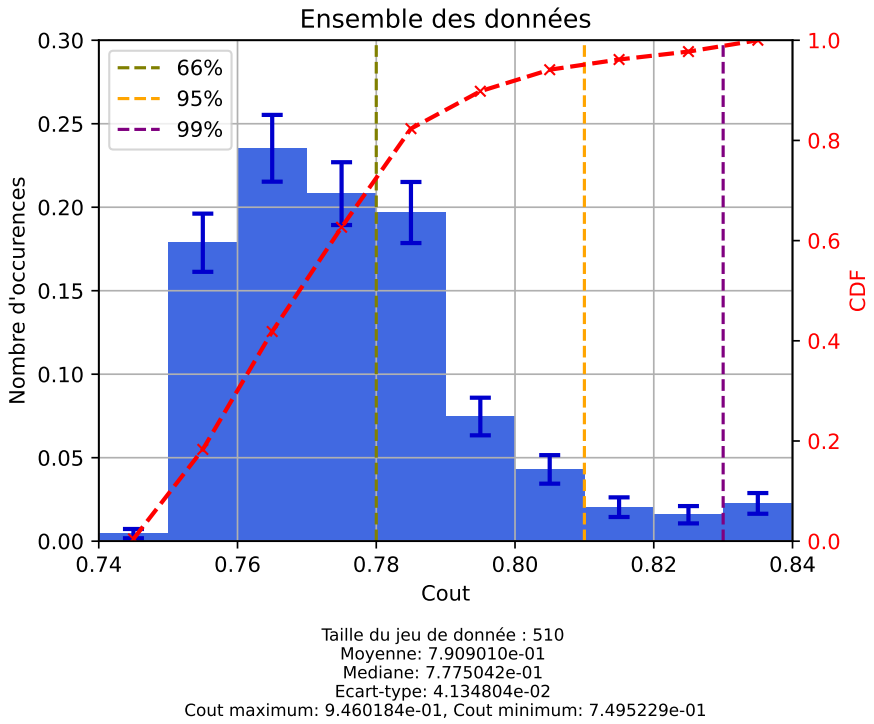

FIGURE 11 – Histogramme de la répartition des coûts pour 600 s de calcul.

Il s'agit de connaître la probabilité d'obtenir un coût compris entre 0,74 et 0,75. Pour un calcul de 3600 s de temps de calcul, la probabilité est explicitement donnée par :

$$P(\text{coût} \in [0,74\ ;\ 0,75]) = 0,187 \pm 0,016$$

Pour un calcul de 600 s, la probabilité est :

$$P(\text{coût} \in [0,74\ ;\ 0,75]) = 0,004 \pm 0,003$$

Pour six calculs de 600 s de temps de calcul, la probabilité est calculée à l'aide de la formule :

$$P(\text{coût} \in [0,74\ ;\ 0,75]) = \sum_{i=1}^{6} \binom{6}{i} \cdot p^i \cdot (1-p)^{6-i}$$

En utilisant $p = 0,004 \pm 0,003$, la probabilité obtenue est :

$$P(\text{coût} \in [0,74\ ;\ 0,75]) = 0,024 \pm 0,018$$

avec l'incertitude δP approximativement égale à $\left|\frac{\partial P}{\partial p}\right| \delta p$.

Conclusion : Un calcul de 3600 s donnera un meilleur résultat que six calculs de 600 s : $P_{1\ calcul\ 3600\ s}(\text{coût} \in [0,74\ ;\ 0.75]) > P_{6\ calculs\ 600\ s}(\text{coût} \in [0,74\ ;\ 0,75])$. La répartition des coûts pour 3600 s de calcul est beaucoup plus centrée avec une queue de distribution quasi inexistante. Les résultats seront non seulement de meilleure qualité, mais aussi plus fiables.

IV.2.4 Itération sur place

Il peut arriver lors du processus d'optimisation que le meilleur individu n'évolue pas lors de plusieurs itérations consécutives. Estimer la confiance dans un résultat obtenu par itération sur place est essentiel pour évaluer la fiabilité des résultats. En effet, un nombre important d'itérations sur place suggère que la configuration est bloquée dans un minimum local dont il peine à s'extraire. Il est pertinent de comparer les résultats obtenus ayant effectué ces itérations sur place et de la comparer à celle obtenue sans itérations sur place pour évaluer l'impact de ce biais. La pertinence de continuer un calcul bloqué dans un minimum local est à déterminer. L'arrêt précoce d'un calcul, lorsqu'il est prévisible que son résultat ne sera pas satisfaisant, permet de disposer de plus de temps pour relancer un nouveau calcul et ainsi économiser des ressources. Il est crucial de peser les avantages de continuer les itérations sur place contre le risque de compromettre la qualité des résultats finaux en raison de la réduction du temps de calcul total.

Sur la figure 12 est représentée en échelle double logarithmique la différence de coût entre le coût final du calcul et la première fois qu'il a itéré 800 fois sur place en fonction du delta de temps entre la première itération sur place et la fin du calcul. Un dégradé de couleurs va du vert (coût le plus faible) au rouge (coût le plus élevé). Les mêmes analyses ont été réalisées pour 400, 500, 600 et 700 itérations sur place.

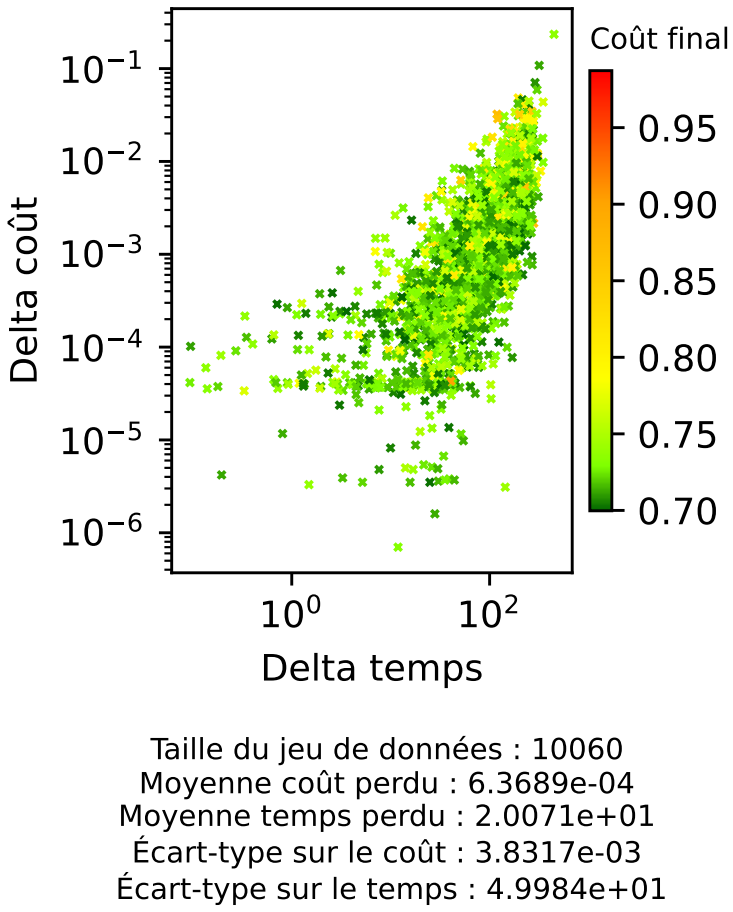

FIGURE 12 – Tracé des résultats avec 800 itérations sur place.

Conclusion : Après avoir testé différents modèles polynomiaux, exponentiels ou logarithmiques, aucune relation n'a pu être établie entre le nombre d'itérations sur place et le coût final des résultats. Les résultats obtenus avec des itérations sur place sont comparables en qualité à ceux obtenus sans itérations sur place. Par conséquent, il n'y a aucune raison de ne pas conserver l'ensemble des résultats. Le chemin suivi pour atteindre une configuration donnée ne semble pas influencer la qualité finale du résultat.

IV.2.5 Type d'initialisation

Il existe différentes méthodes d'initialisation des individus formant la population initiale utilisée par l'algorithme d'évolution différentielle. Deux méthodes courantes d'initialisation sont l'hypercube latin et la méthode sobol. L'initialisation par hypercube latin garantit que chaque individu dans la population possède des paramètres distincts. L'initialisation sobol assure en plus que tous les paramètres des individus soient orthogonaux, offrant une couverture uniforme de l'espace. Cependant, elle est spécifiquement adaptée aux tailles de population de la forme 2^n [33]. L'analyse a été réalisée avec une population égale à 8D et non 6D, comme conseillée précédemment pour 600 s de calcul, du fait de cette particularité.

Le tracé de la figure 13 montre la performance globale et la qualité des solutions obtenues en fonction des différents types d'initialisation.

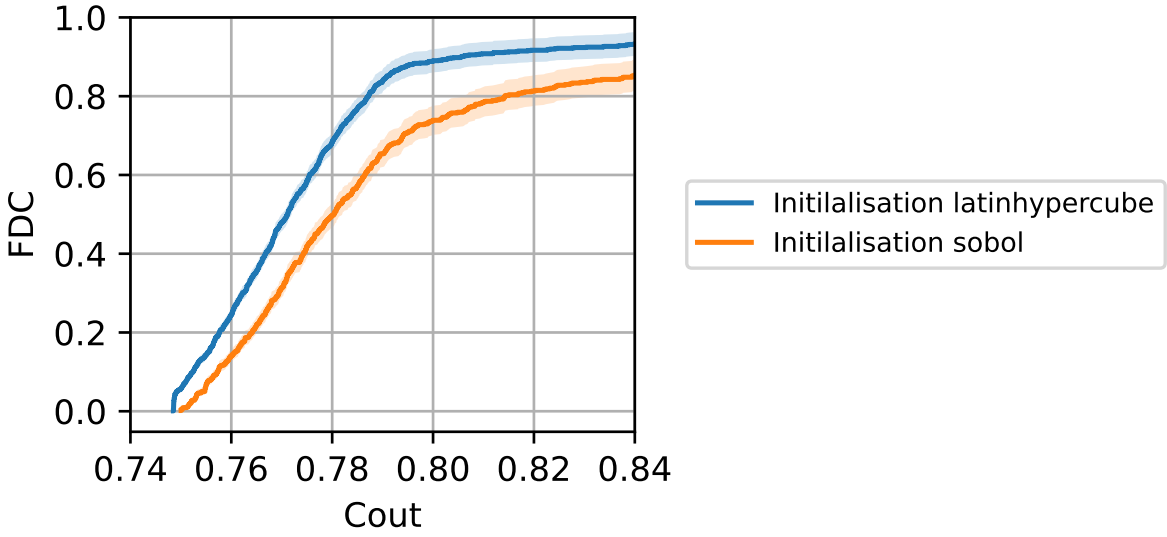

FIGURE 13 – Comparaison des FDC par type d'initialisation en fonction du coût pour une population valant 8D et 600 s de calcul.

Conclusion : Le constat final est que l'initialisation par hypercube latin donne de meilleurs résultats. Le temps gagné grâce à cette méthode d'initialisation constitue un avantage significatif, surpassant l'intérêt d'une couverture plus uniforme de l'espace offerte par une initialisation sobol.

IV.3 Conclusion et possibilités de poursuite d'étude

L'analyse des différentes méthodes et des configurations utilisées pour la partie algorithmique de SONGe révèle plusieurs points importants.

Les méthodes déterministes, bien que garantissant un minimum global, sont impraticables en raison de leur coût computationnel élevé. En revanche, l'algorithme par évolution différentielle, faisant partie des algorithmes stochastiques, se révèle optimal, offrant les meilleurs résultats avec une faible variance. Il répond le mieux aux exigences de SONGe. Le fonctionnement en monoboucle est plus efficace que par boucle, maintenir une population stable est plus avantageux que de favoriser un brassage génétique intense. La diversité peut être préservée en ajustant la taille de la population ou le facteur de croisement. Pour des temps de calcul spécifiques, les tailles de population appropriées varient avec des recommandations précises pour différents intervalles de temps. Une durée de calcul plus longue, comme 3600 s, donne des résultats de meilleure qualité que plusieurs calculs plus courts. Enfin, l'initialisation par hypercube latin offre des performances supérieures par rapport à l'initialisation sobol, grâce à un meilleur compromis entre coût computationnel d'initialisation et couverture de l'espace.

Pour approfondir les résultats obtenus, plusieurs pistes intéressantes peuvent être explorées. Il serait pertinent d'examiner les effets des paramètres de mutation (F) et de croisement (CR) sur la performance de l'algorithme. De plus, l'évaluation de différentes stratégies d'évolution pourrait offrir de nouvelles perspectives pour améliorer les résultats. Une approche prometteuse consisterait à utiliser plusieurs algorithmes d'optimisation au cours du processus d'optimisation pour tirer parti de leurs forces respectives. L'objectif serait de réduire l'exploration exhaustive de l'espace des solutions à la fin du processus, une fois que les zones des minima ont été identifiées, afin d'accélérer la convergence vers des solutions optimales en utilisant des algorithmes déterministes sur un espace de recherche très restreint. Enfin, une initialisation personnalisée basée sur un ensemble de configurations connues pour leur bonne performance, par exemple des individus appartenant au front de Pareto, pourrait également améliorer les résultats en exploitant les résultats antérieurs pour guider l'optimisation.

Désormais, le réglage de la partie algorithmique de SONGe est caractérisé et optimisé. L'observation des sensibilités des paramètres ainsi que l'analyse de leurs relations et possibles corrélations sont facilités. L'identification des configurations optimales est simplifiée et plus rapide. La suite consiste à mener des études approfondies sur les configurations possibles des MSFR.

V Pareto et aide à la décision

V.1 Introduction

Le but est de déterminer comment les paramètres de conception réagissent aux tolérances et aux bornes des contraintes. Il s'agit également de trouver les configurations optimales sur une large gamme de valeurs des variables et contraintes. L'objectif est également d'identifier les éventuelles corrélations entre les coûts des différents paramètres à l'aide du front de Pareto.

Le coût est le critère de sélection prioritaire. La sélection des solutions basée sur le coût implique une intervention préalable du décideur pour orienter les choix. Elle nécessite une compréhension approfondie du problème dès le départ. En revanche, le tracé du front de Pareto est réalisé a posteriori, offrant une vue d'ensemble des meilleures solutions en fonction des compromis entre les différents critères, sans nécessiter une intervention initiale.

L'architecture initiale de SONGe a été détournée d'un fonctionnement a priori, avec

des tolérances et des contraintes établies au départ, vers un fonctionnement a posteriori, avec la construction du front de Pareto. Le but était de pouvoir visualiser l'ensemble des solutions afin de permettre au décideur de pouvoir faire un choix avisé en connaissant l'ensemble le plus exhaustif possible des configurations optimales.

V.2 Première idée

La première méthode envisagée pour échantillonner le front de Pareto consistait à faire varier les tolérances et bornes pour obtenir différentes solutions optimales vis-à-vis des contraintes appliquées. En répétant suffisamment l'opération, il était envisagé d'échantillonner le front de Pareto.

La première étude réalisée portait sur la sensibilité aux pertes de charge des configurations des MSFR. Pour ce faire, la tolérance sur la contrainte correspondant aux pertes de charge a été modifiée. J'ai réalisé 185 optimisations de 3600 s pour 0.1, 1, 2, 3, 4 et 5 bars de tolérance sur les pertes de charge avec comme contrainte maximum 10 bars. Ensuite, une partie des coûts des meilleures configurations pour chaque tolérance ont été affichés dans la figure 14, à l'aide de l'affichage graphique conçu par Axel Laureau et Thomas Sornay, membres de l'équipe MSFR du LPSC. L'affichage de l'ensemble des coûts est disponible figure 21 en annexe.

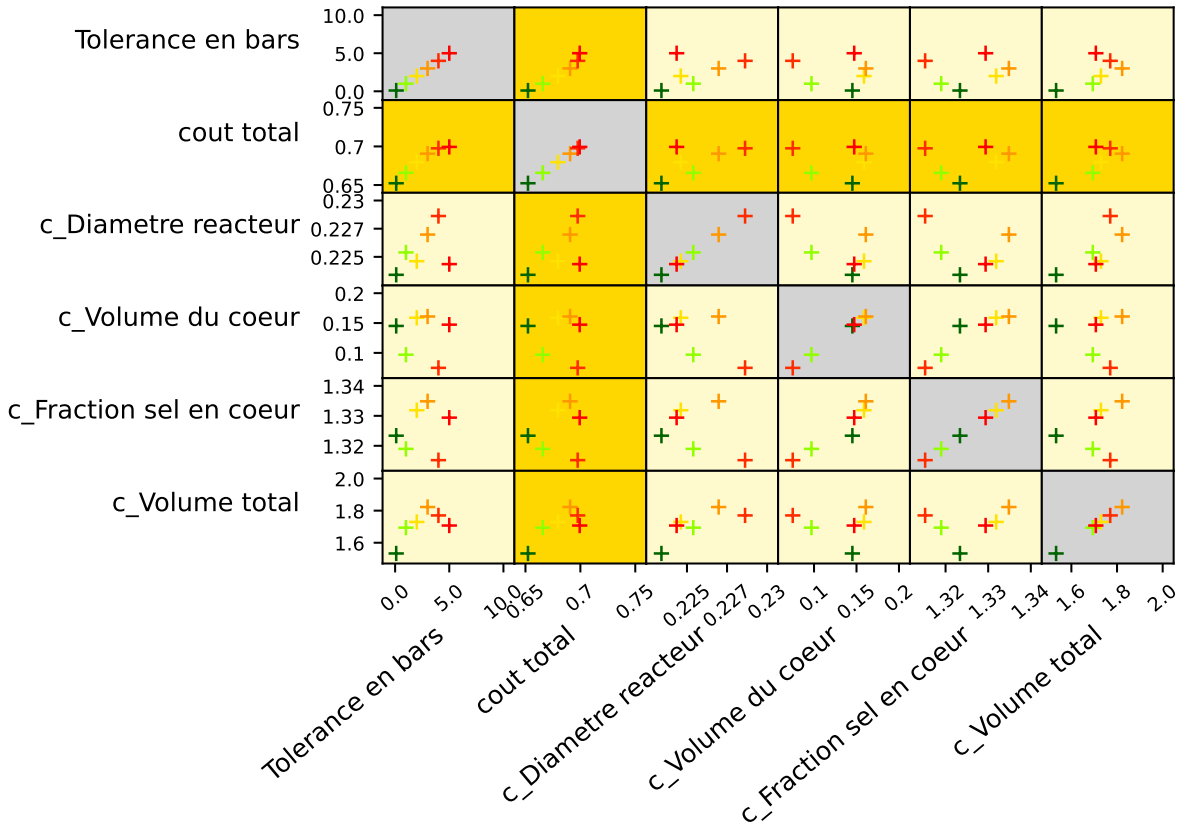

FIGURE 14 – Affichage partiel des coûts pour différentes tolérances sur les pertes de charges.

À l'affichage de ces résultats, un problème a été constaté. Intuitivement, la réduction de la tolérance devait entraîner une augmentation des coûts. Cependant, de manière contre-

intuitive, le coût observé diminue avec le renforcement des contraintes, à condition qu'elles soient respectées.

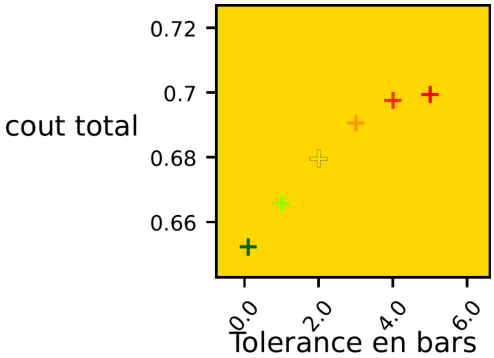

FIGURE 15 – Affichage partiel des coûts pour différentes tolérances sur les pertes de charges.

Cela s'explique par la définition des fonctions objectives, qui est $f_i(x_i) = \exp\left(\frac{\delta_i(x_i - c_i)}{\sigma_i}\right)$. Ces fonctions sont mal définies. Elles compliquent l'ajustement des tolérances ou des bornes pour obtenir des configurations optimales avec différentes valeurs de pertes de charge. En modifiant ces paramètres, les définitions des fonctions objectifs changent, le problème à résoudre n'est plus le même. Par conséquent, les coûts et les résultats ne sont plus comparables et les différents optimums n'appartiennent pas au front de Pareto du même problème. Il fallait trouver une méthode pour échantillonner l'espace sans altérer les différentes fonctions objectifs de coût.

Il subsistait également des interrogations sur la manière dont l'algorithme ajuste le coût lorsque l'objectif est un intervalle fermé et sur l'équité de la pénalisation du non-respect des bornes. Par exemple, dépasser de 1 000 pascals la contrainte sur la perte de charge fixée à 100 000 pascals est pénalisé bien plus sévèrement, à tolérance égale, qu'un dépassement de 0,1 sur la fraction de sel au cœur, même si l'écart relatif à la borne est bien plus grand. Un même écart de 1 % par rapport à la borne de pression entraînait un coût bien plus élevé que pour la fraction de sel en cœur. Cela conduisait à une préférence pour le respect des contraintes avec des bornes très larges.

De plus, les fonctions objectifs de coût restaient décroissantes même à l'intérieur de l'intervalle. Une compétition pouvait avoir lieu entre le gain sur le coût d'une fonction objectif dont la contrainte était déjà respectée et une contrainte dont la borne n'était pas respectée. L'objectif était de passer à une approche plus binaire, où un critère est soit respecté, soit non respecté. En cas de non-respect, la pénalisation doit varier en fonction de l'écart par rapport aux bornes.

V.3 Nouveau coût, nouveau monde

Adopter une nouvelle définition des objectifs était nécessaire pour pouvoir obtenir un front de Pareto correctement échantillonné. Plusieurs méthodes pour construire un front de Pareto sont décrites dans la littérature [36]. Conserver une méthode par agrégation pondérée des fonctions objectifs a l'avantage d'entraîner peu de modifications du fonctionnement de SONGe. C'est le choix qui a été effectué. Il faut néanmoins redéfinir correctement les fonctions objectifs et mettre en place la possibilité de paramétrer des poids pour chacune d'elles.

Pour ce faire, le calcul du coût des fonctions objectifs a été redéfini dans le code Java. Chaque fonction objectif est maintenant définie selon le respect ou non d'un intervalle. L'objectif est atteint lorsque la valeur contrainte est à l'intérieur de l'intervalle cible, sinon c'est un échec auquel est associé un coût défini en fonction de l'écart relatif à la borne dépassée.

Les nouvelles fonctions objectifs sont définies comme :

$$f(x_i) = \begin{cases} \exp\left(\frac{\sigma_i \delta_i (x_i - c_i)}{c_i}\right) - 1 & \text{si } p_{i0} \text{ est une contrainte maximum,} \\ 0 & \text{sinon.} \end{cases}$$
$$\text{avec } \delta_i = \begin{cases} +1 & \text{si } p_{i0} \text{ est une contrainte maximum,} \\ -1 & \text{sinon.} \end{cases} \tag{11}$$

Cette approche accorde une importance identique à toutes les contraintes dans le coût total dans le cas où les poids seraient tous égaux. Pour reprendre l'exemple précédent, la pression et la fraction de sel en cœur contribuent désormais de manière équitable au coût total. Les σ_i ne sont plus associées à des tolérances, mais à des facteurs de sanctions des sorties d'intervalle. Ces facteurs déterminent la sévérité avec laquelle une sortie d'intervalle est pénalisée. En effet, ils accentuent la pente de la fonction objective, renforçant ainsi la sanction appliquée lorsqu'une contrainte dépasse les limites définies. Il est cependant toujours possible d'interpréter σ_i^{-1} comme une tolérance.

La nouvelle fonction à minimiser, construite par somme pondérée, est désormais de la forme :

$$F(\mathbf{x}) = \frac{1}{\sum_{i=1}^{M} \omega_i} \sum_{i=1}^{M} \omega_i f_i(x_i) \tag{12}$$

Normaliser la somme ne change rien au calcul d'optimisation, les solutions obtenues appartiennent toujours au Front de Pareto [28]. En revanche, elle aide à la prise de décision, car le coût total est plus facilement interprétable par le preneur de décision humain.

Cette redéfinition rend particulièrement sensibles certaines relations dans les contraintes à optimiser, comme le delta pompe, le déséquilibre de température, ou le déséquilibre de puissance, qui garantissent l'intégrité physique du réacteur. Ces relations ont été fixées pour offrir plus de degrés de liberté au système d'optimisation. Un respect absolu de ces contraintes est exigé à la fin de l'optimisation car tout compromis sur ces critères compromettrait la validité physique du réacteur.

Précédemment, les bornes de ces contraintes étaient fixées à 0. Avec les nouvelles fonctions objectifs, cela entraînerait des divisions par 0 et n'est donc pas possible. Une solution explorée a été de fixer une borne très faible, de l'ordre de 10^-6. Cependant, cela a posé des problèmes de divergence des coûts en début de simulation. La convergence numérique était impossible. Pour contourner ce problème, les σ_i de ces contraintes ont été adaptés pour éviter que leur coût ne diverge. Avec des σ_i faible le coût initial est réduit et la convergence est de nouveau possible. Ce compromis reste insatisfaisant, car il affecte la sévérité avec laquelle une sortie de l'intervalle est pénalisée alors que c'est le respect de ces mêmes contraintes qui est primordial. Une solution envisageable serait de créer de nouvelles fonctions objectifs spécifiquement pour ces trois contraintes.

V.4 Comparaison pertes de charge nouvelle méthode

Une nouvelle étude de sensibilité sur les pertes de charge a été réalisée. À jeu de poids identique, le poids sur la fonction objectif liée aux pertes de charge a été fixé à 0.1, 1, 2, 3, 4, 5. Pour chaque poids 185 calculs d'optimisation de 3600 s avec une population de 8D individus ont été réalisés. Seul le meilleur résultat de ces 185 optimisations a été conservé. Ces calculs aboutissent au tracé suivant.

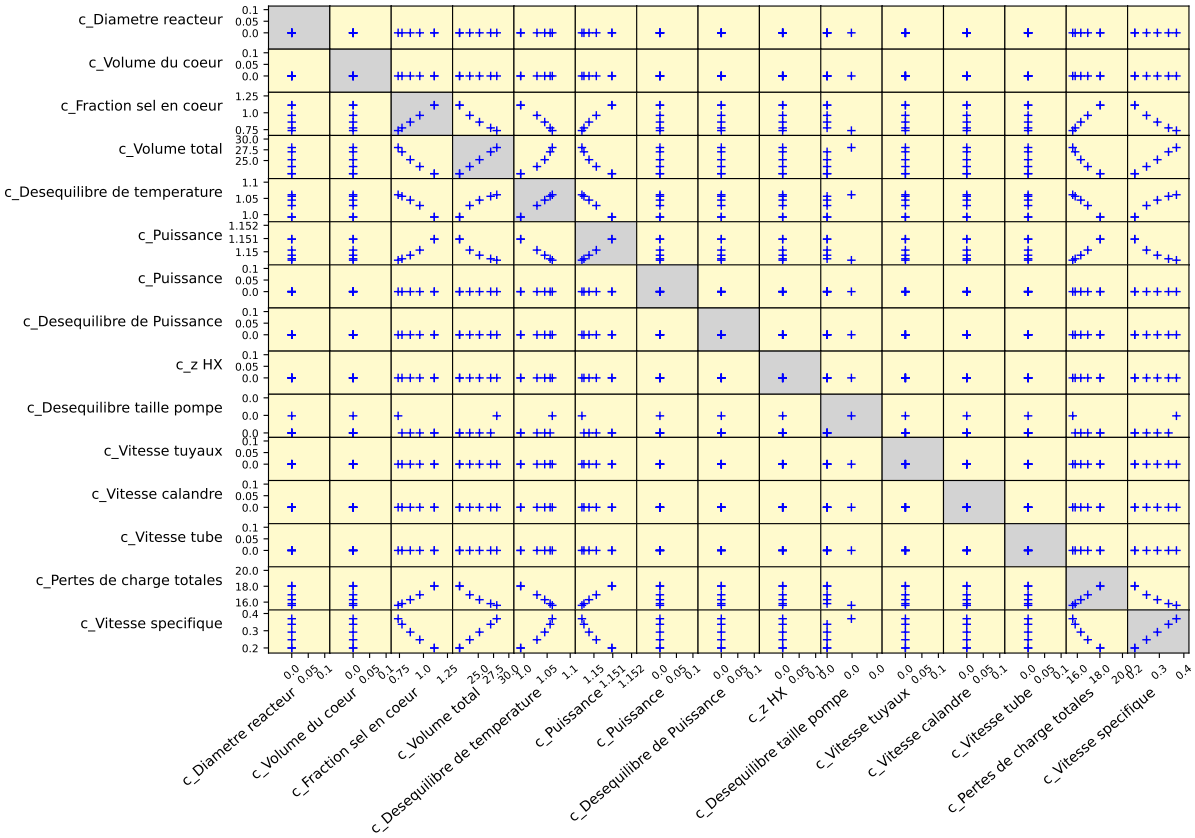

FIGURE 16 – Affichage des coûts de l'étude des pertes de charge en utilisant la nouvelle méthode de calcul des coûts.

L'objectif de cette étude était de vérifier le bon fonctionnement de la nouvelle définition des fonctions de coûts et du système de poids, ainsi que leur capacité à faire évoluer le coût d'une contrainte afin d'explorer les configurations optimales sur une plage de valeurs. Il est observé que huit critères sont respectés, quel que soit le poids de la contrainte sur les pertes de charge. En revanche, les coûts liés aux pertes de charge varient effectivement en fonction de celui-ci. Les coûts les plus élevés correspondent aux poids les plus faibles, et inversement. Bien que non nuls, les coûts des déséquilibres restent faibles et acceptables. Pour prendre l'exemple du déséquilibre de température, un coût de 1 équivaut, dans la configuration testée, à un écart de déséquilibre d'environ 7 K, ce qui est très faible au vu des températures en jeu qui sont aux alentours de 950 K.

V.5 Étude puissance demandée

Une étude sur la contrainte de puissance a également été réalisée. En faisant progressivement varier uniquement le poids sur la puissance demandée, une configuration dans

laquelle tous les critères sont respectés, excepté la puissance demandée, est obtenue. C'est la configuration optimale respectant tous les critères avec la puissance maximale possible. Cette puissance est de 182,4 MW pour le jeu de contraintes présenté dans le tableau 5.

TABLE 5 – Tableau des nouvelles contraintes.

Composant	**Contrainte**	**Min**	**Max**
Tuyaux	Vitesse maximum dans les tuyaux [m/s]		5
Échangeur intermédiaire	Vitesse calandre [m/s]		2
Échangeur intermédiaire	Vitesse tube [m/s]		5
Réacteur	Diamètre réacteur [m]		5
Cavité centrale	Volume de sel combustible [m3]		5
Tuyaux	Position y échangeur [m]	0.2	
Cavité centrale	Déséquilibre de température [K]		0.01
Cavité centrale	Puissance demandée [MW]	300	300
Cavité centrale	Déséquilibre de Puissance [MW]		0.0001
Tuyaux	Delta Pompe		0.001
Pompe primaire	Vitesse spécifique		100
Pompe primaire	Pertes de charges [Pa]		1000000
Cavité centrale	Volume du cœur [m3]	1	
Cavité centrale	Fraction de sel combustible en cœur	0.5	

Ce type d'étude peut être étendu à d'autres paramètres pour avoir un ordre d'idée des valeurs limites pour un jeu de contraintes donné.

V.6 Problème de présentation des résultats

Du point de vue d'un analyste, présenter des résultats d'optimisations à un décideur n'est pas facile. Comment présenter au décideur l'ensemble des options disponibles et lui faire comprendre les compromis possibles et ceux qui ne le sont pas ? L'avantage du calcul a priori est qu'il détermine une unique solution, sur la base d'un choix de jeu de poids décidé à l'avance. Cependant, le choix de ces poids n'est pas évident [20] [36]. Il doit tenir compte de la méthode d'évaluation des objectifs ainsi que du classement de leur importance relative. De plus, le choix des poids ne donne pas l'assurance de fournir le résultat correspondant à l'envie du décideur. Il peut y avoir un problème dans la traduction et la mise en forme des envies et des préférences. Leena Tanner présente un exemple de ce type dans [37] sous la forme d'un problème d'optimisation de choix d'un mari. En outre, des problèmes mathématiques d'incapacité à trouver, avec une méthode par agrégation pondérée, le meilleur compromis recherché dans un problème non convexe peuvent être rencontrés [29] [28].

Finalement, même une fois les résultats acquis, la mise en forme des résultats afin de les présenter au décideur est difficile. Prenons l'exemple suivant simplifié d'optimisation d'un problème selon trois critères. Une recherche d'optimisation par somme pondérée a permis d'obtenir six individus différents. Les vecteurs des coûts selon les fonctions objectifs f1, f2 et f3 sont :

$$V_1 = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \quad V_2 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \quad V_3 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \quad V_4 = \begin{pmatrix} 0.5 \\ 0.5 \\ 1 \end{pmatrix} \quad V_5 = \begin{pmatrix} 1 \\ 0.5 \\ 0.5 \end{pmatrix} \quad V_6 = \begin{pmatrix} 0.5 \\ 1 \\ 0.5 \end{pmatrix}$$

D'après Censor [28] ces points appartiennent tous au front de Pareto.

Une représentation en deux dimensions du problème est proposée figure 17. Les fonctions sont affichées comme suit : f2 en fonction de f1, f3 en fonction de f2, et f3 en fonction de f1.

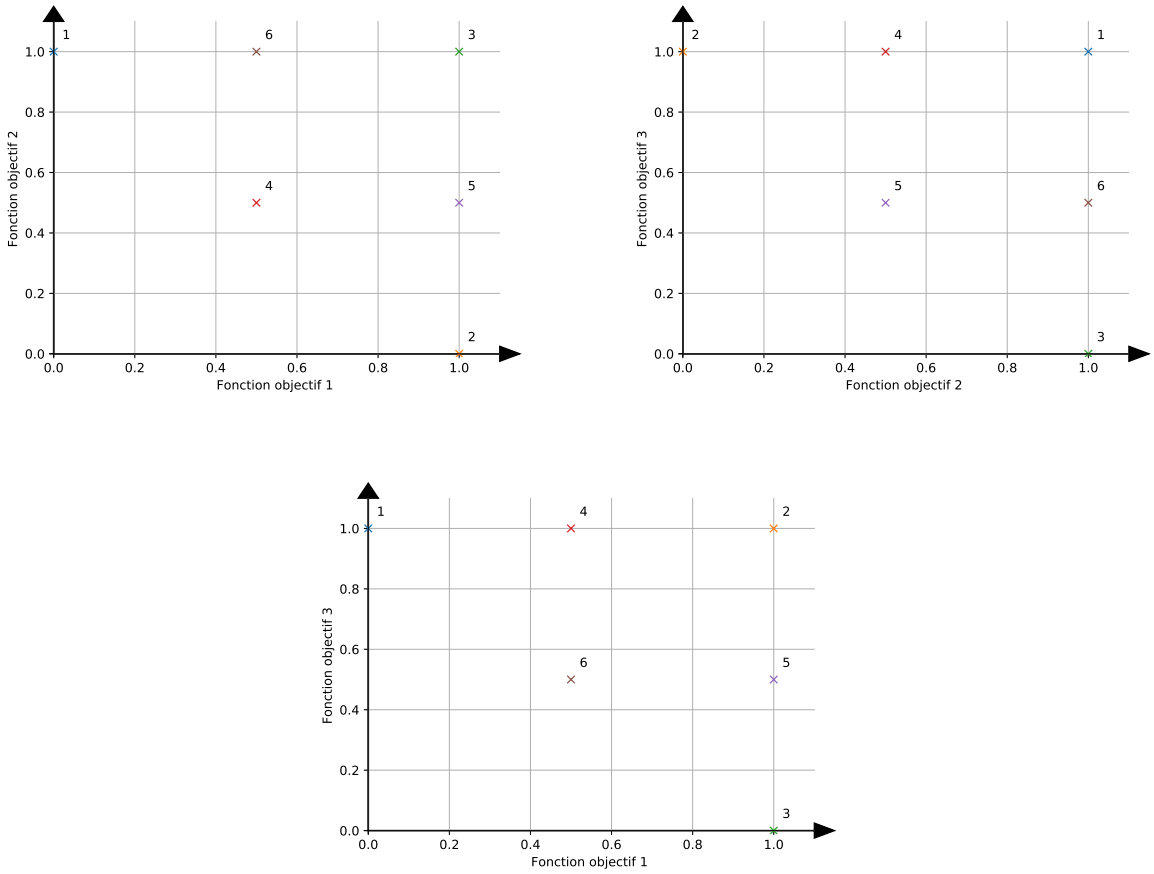

FIGURE 17 – Affichage en deux dimensions des différents vecteurs.

Aucun « front », au sens de ligne, ne se démarque. Sur chaque figure une bordure est visible, mais elle est différente à chaque fois et ne comporte pas l'ensemble des points du front au sens de Pareto.

Un affichage 3D des solutions figure 18 permet de visualiser l'ensemble des solutions sous la forme d'une surface.

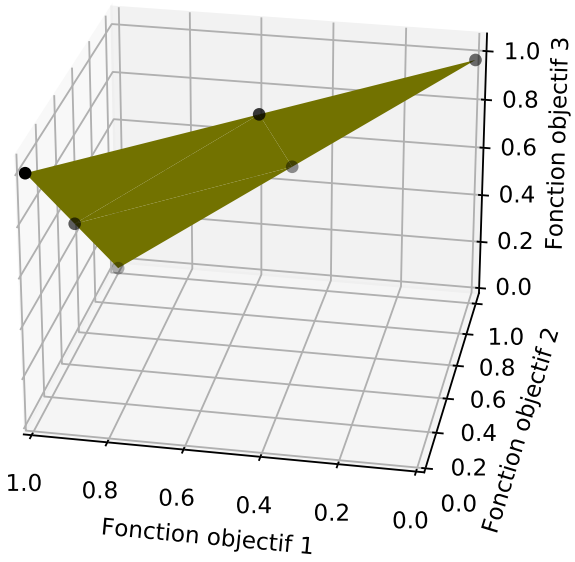

FIGURE 18 – Front de Pareto 3D.

Le front de Pareto est de dimension N-1 au maximum si N est la dimension du problème.

V.6.1 Tracé de l'ensemble des individus non dominés trouvés

Lors des optimisations par variations de poids, il est possible de récupérer l'ensemble des résultats obtenus sur les 185 optimisations réalisées, quel que soit leur coût total, et pas seulement les meilleurs pour chaque jeu de poids. L'ensemble de ces résultats peut ensuite être traité par une fonction qui ne garde que ceux qui ne sont pas dominés, formant ainsi un « pseudo-front » de Pareto. La figure 19 présente l'ensemble des individus non dominés obtenus lors de différents calculs d'optimisations.

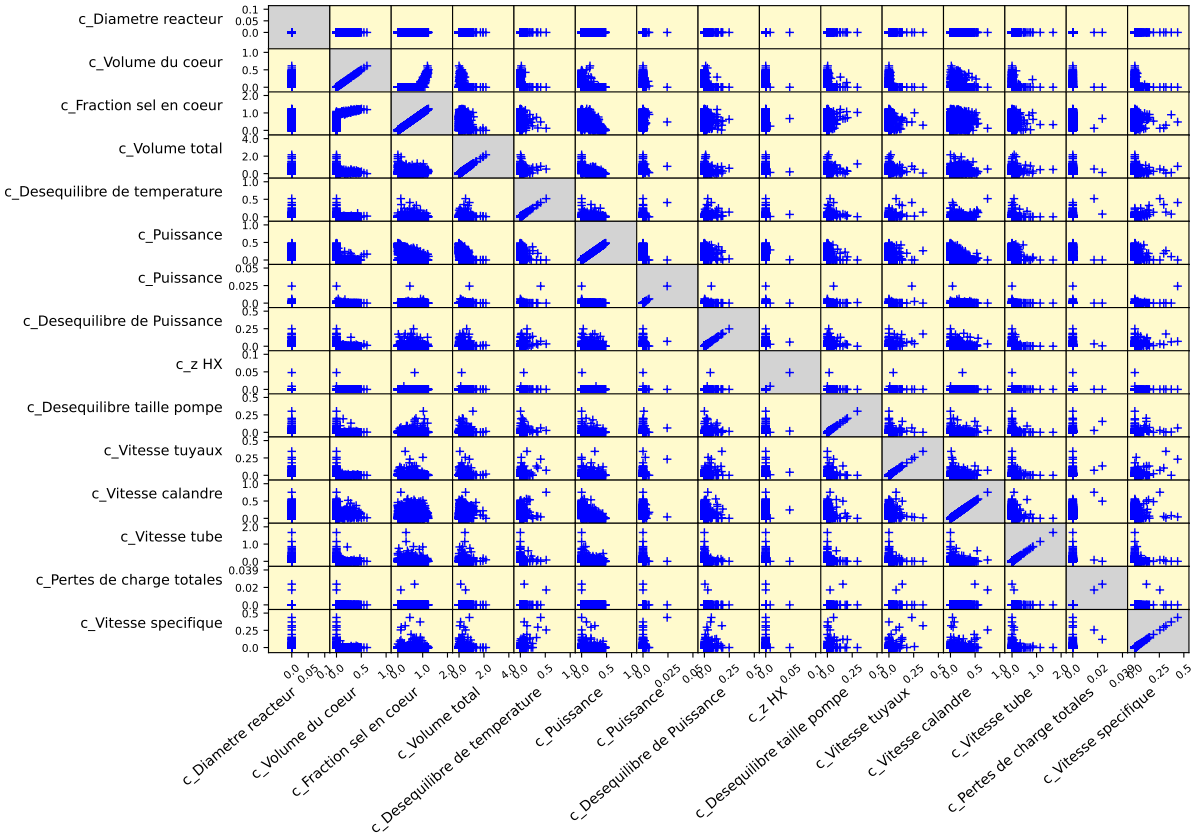

FIGURE 19 – Tracé des coûts de l'ensemble des individus non dominés trouvés.

Certaines fonctions objectifs semblent liées par une relation directe. L'hypothèse est d'autant plus vérifiée que le nombre de points considérés est ici très important (11 000 points environs). Cependant, pour certaines fonctions objectifs aucune relation n'apparaît. La présence d'un nuage de points est observée. Ces derniers peuvent être réduits comme expliqué par la suite.

L'utilisation du terme de « pseudo-front » de Pareto est employée, car aucun meilleur individu n'a été trouvé, mais cela ne signifie pas qu'il n'en existe pas. En effet, Censor [28] démontre uniquement l'appartenance au front de Pareto du meilleur individu pour le jeu de poids donné. Il est donc peu judicieux de prendre tous les points non dominés de l'optimisation, car bien qu'ils fassent partie de l'ensemble des possibles, rien ne garantit qu'ils soient sur le front de Pareto. Il est toujours possible d'en trouver de meilleurs. Cependant, ils permettent tout de même de borner le front de Pareto.

En reprenant l'exemple précédent, si une optimisation permet d'obtenir un vecteur

$$V_7 = \begin{pmatrix} 0.5 \\ 0.5 \\ 0.5 \end{pmatrix}$$

alors les vecteurs V_4, V_5 et V_6 apparaissent désormais comme n'appartenant plus au front de Pareto. Ce genre de cas peut arriver en pratique. Le processus d'optimisation étant stochastique, le minimum global n'est jamais connu. En pratique, il est bon de vérifier à chaque nouvel ajout de solutions parmi le front de Pareto si des configurations ne sont pas à déclasser.

V.6.2 Tracé final

L'affichage des meilleurs individus, après vérification qu'ils soient bien dans le front de Pareto, a été réalisé et est reporté dans la figure 20.

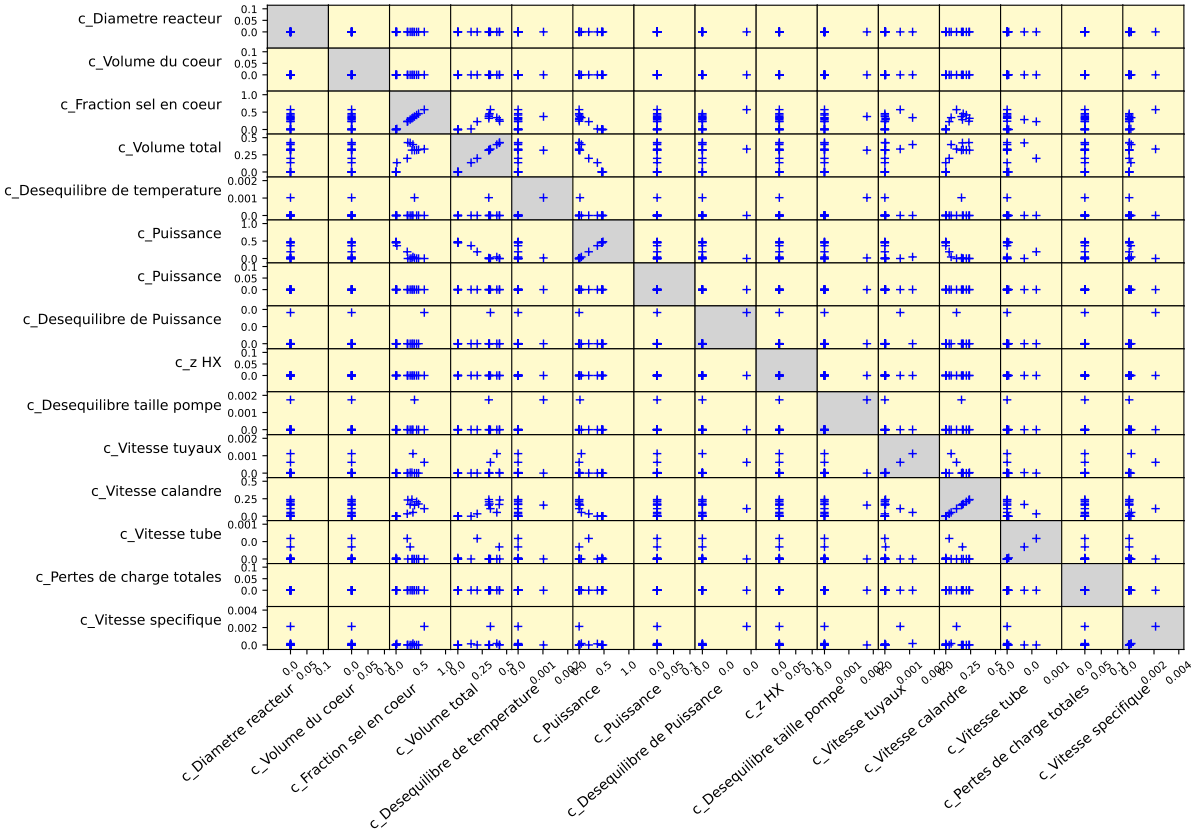

FIGURE 20 – Tracé des coûts des points du front de Pareto.

Lors de la recherche du front de Pareto, certains points obtenus ont été déclassés a posteriori. Ces points correspondaient à des jeux de poids pour lesquels tous les objectifs étaient atteints sauf un. Les différents points étaient alors en compétition uniquement pour obtenir la meilleure optimisation de cet objectif. Leur suppression est due à l'erreur intrinsèque de l'optimisation globale, qui ne trouve pas le minimum global. En théorie, avec une optimisation parfaite, tous les jeux de poids concernés devraient donner le même résultat.

Il n'existe pas de relations directes entre toutes les fonctions objectifs. Par exemple, aucune corrélation ne semble apparaître entre le coût lié à la vitesse de la calandre et le coût associé à la fraction de sel en cœur.

V.6.3 Tracé proposé

Une fois l'étude par variation des poids réalisée, une exploitation possible pour le décideur est de fixer a posteriori un choix de jeu de poids et de visualiser le tracé des points du front de Pareto avec un dégradé de couleurs reflétant le coût total, en fonction de ses préférences. Cet affichage interactif est recommandé pour une meilleure compréhension des options par le décideur, avec la possibilité de modifier les jeux de poids en temps réel et d'obtenir une mise à jour immédiate de l'affichage [36]. Cela lui permet d'avoir à la

fois une vision ciblée de ses préférences et de les nuancer avec une vision plus large des meilleurs compromis possibles, pour aboutir à un choix plus éclairé.

Il est également envisageable que le décideur ne soit intéressé que par un seul compromis entre deux fonctions objectifs. Dans ce cas, seul l'affichage des meilleurs points pour ce compromis sera sélectionné, c'est-à-dire le front de Pareto selon deux fonctions objectifs (deux critères), construit à partir de l'espace du front de Pareto total.

Enfin, il serait intéressant d'identifier l'ensemble des paires de deux ou trois contraintes corrélées. Le tracé en 2D des paires de deux contraintes formerait des courbes, tandis que le tracé en 3D des paires de trois contraintes donnerait lieu à des surfaces. L'utilisation de graphiques tridimensionnels est détaillée dans la section intitulée «Visualization of the Pareto Frontier » du livre « Multiobjective Optimization : Interactive and Evolutionary Approaches » [36].

V.7 Conclusion

Une étude complète d'une optimisation multiobjectifs en 15 dimensions est complexe. De nombreuses méthodes, chacune avec ses avantages et ses défauts, sont disponibles. La somme pondérée présente l'avantage d'être simple à utiliser, mais cette simplicité constitue également sa faiblesse. La bonne compréhension de l'importance des poids, de leur rôle, ainsi que de leur influence sur l'indication d'une solution est subtile. Elle comporte de nombreux biais invisibles et difficilement détectables en raison de la complexité du problème [36]. Pour offrir au décideur la meilleure vision possible sur le problème, la combinaison de méthodes a posteriori, avec le tracé complet du front de Pareto, et de méthodes a priori, avec un classement des solutions basé sur un jeu de poids, semble être l'approche la plus satisfaisante. Profiter des outils technologiques pour construire un affichage interactif, permettant de modifier les jeux de poids en temps réel et d'obtenir une mise à jour immédiate des solutions, constitue un véritable atout pour le décideur.

VI Conclusion

La gestion de l'énergie est un enjeu central de notre ère, notamment dans le contexte de lutte contre le réchauffement climatique. La transition vers des sources d'énergie à faible émission carbone, telles que le nucléaire, est cruciale pour atteindre les objectifs de réduction des émissions de CO_2. Le développement de nouvelles technologies de réacteurs nucléaires, particulièrement des RSF avec l'exemple des MSFR, représente une avancée prometteuse. Les MSFR offrent de nombreux avantages possibles : une meilleure efficacité énergétique, une capacité à transmuter les déchets nucléaires, et une sécurité accrue grâce à des propriétés intrinsèques de réactivité. Cependant, plusieurs défis techniques restent en suspens, tels que la gestion de la corrosion, la maintenance, le lancement du cycle de réacteurs avec un combustible initial adéquat ainsi que l'architecture et le dimensionnement de référence du réacteur.

L'utilisation d'algorithmes d'optimisation, en particulier les algorithmes génétiques implémentés dans le code SONGe, permet de parcourir l'espace des configurations possibles pour les MSFR. SONGe offre une méthode efficace pour approcher des solutions optimales, en prenant en compte les multiples contraintes et les paramètres en jeu. L'analyse de la partie algorithmique de SONGe a permis de tirer plusieurs conclusions importantes. Les algorithmes déterministes, bien qu'assurant un minimum global, sont irréalisables en

raison de leur coût computationnel élevé. En revanche, l'algorithme d'évolution différentielle, un algorithme stochastique, offre des résultats optimaux avec des coûts faibles et une variance réduite, répondant parfaitement aux exigences de SONGe. Le fonctionnement en monoboucle s'avère plus efficace que celui en boucle, et maintenir une population stable est plus avantageux que de favoriser un brassage génétique intense. La diversité peut être maintenue en ajustant la taille de la population ou le facteur de croisement. Les tailles de population idéales varient en fonction du temps de calcul. Par exemple, une durée de 3600 s produit des résultats de meilleure qualité que plusieurs calculs plus courts. Enfin, l'initialisation par hypercube latin est plus performante que l'initialisation sobol, en raison d'un meilleur compromis entre coût computationnel et couverture de l'espace.

L'étude de la construction du front de Pareto et du fonctionnement de l'aide à la décision a permis d'identifier plusieurs points cruciaux. Les variations des tolérances et des bornes modifiaient la fonction de coût, rendant impossible l'obtention d'un front de Pareto commun. La redéfinition des fonctions de coût a permis de résoudre ce problème tout en garantissant l'intégrité physique du réacteur dans une certaine mesure. Une étude sur la contrainte de puissance a permis d'obtenir une configuration optimale avec une puissance maximale de 182,4 MW. Présenter ces résultats à un décideur reste un défi. Le choix des pondérations pour déterminer une solution unique n'est pas évident et peut ne pas correspondre aux attentes du décideur. L'utilisation d'un affichage interactif, permettant de modifier les pondérations en temps réel et de mettre à jour les solutions immédiatement, est un atout considérable pour le décideur et devrait être mis en place. Cela permet d'obtenir une vision précise des préférences, tout en tenant compte des meilleurs compromis possibles.

Pour approfondir les résultats, plusieurs pistes intéressantes peuvent être explorées. Il serait pertinent d'examiner les effets des paramètres de mutation et de croisement sur la performance de l'algorithme. En outre, évaluer différentes stratégies évolutives pourrait offrir de nouvelles perspectives pour améliorer les résultats. Une approche prometteuse serait d'utiliser plusieurs algorithmes d'optimisation au cours du processus pour tirer parti de leurs avantages respectifs. L'objectif serait de réduire la nécessité d'une exploration exhaustive de l'espace des solutions à la fin du processus, une fois que les zones de minima ont été identifiées, et d'accélérer la convergence vers des solutions optimales à l'aide d'algorithmes déterministes sur un espace de recherche restreint. Enfin, une initialisation personnalisée, basée sur des configurations connues pour leurs bonnes performances (comme des individus du front de Pareto), pourrait également améliorer les résultats en exploitant les précédents pour guider l'optimisation.

Désormais, le réglage de la partie algorithmique de SONGe est caractérisé et optimisé. L'observation des sensibilités aux paramètres ainsi que l'analyse de leurs possibles relations et corrélations sont simplifiées. L'identification des configurations optimales est plus rapide qu'auparavant. La prochaine étape consiste à mener des études approfondies sur les configurations possibles des MSFR en ouvrant l'étude sur les valeurs des variables et non plus uniquement sur les coûts des contraintes. Cela donnera un sens beaucoup plus physique à l'utilisation de SONGe.

VII Annexes

VII.1 Résumé en français

Ce rapport traite de l'optimisation des configurations des réacteurs à sels fondus à spectre rapide (MSFR) via l'outil numérique SONGe. Ce travail a permis de sélectionner, parmi différents algorithmes stochastiques et déterministes de la bibliothèque scipy.optimize, l'algorithme par évolution différentielle comme algorithme opérationnel du fait de ses meilleures performances. Une étude non-exhaustive a été effectuée afin de déterminer les meilleurs paramètres d'utilisation de cet algorithme.

Après avoir correctement défini un front de Pareto, son étude a amené à la redéfinition des fonctions de coûts associées aux différentes contraintes. Une méthode par agrégation pondérée des nouvelles fonctions objectifs a été retenue. L'utilisation de SONGe comme outil d'aide à la décision a mis en évidence des défis dans l'acquisition des résultats d'optimisation et aussi dans leur présentation aux décideurs. L'utilisation d'un affichage interactif permettant d'ajuster les poids en temps réel, offrant ainsi une vision plus précise des compromis possibles, a été mise en avant comme solution possible. Plusieurs pistes d'amélioration des résultats ont été proposées, incluant l'analyse des effets des paramètres de mutation et de croisement, l'utilisation successive de différents algorithmes d'optimisation, ainsi que la personnalisation de la population initiale avec des individus admis comme performants.

Finalement, l'optimisation de la partie algorithmique de SONGe a permis de simplifier et d'accélérer l'identification des configurations optimales pour les MSFR. Les prochaines étapes incluront des études plus approfondies sur les configurations possibles des MSFR en ouvrant l'étude sur les valeurs des variables et non plus uniquement sur les coûts des contraintes.

VII.2 English abstract

This report deals with the optimization of molten salt fast reactor (MSFR) configurations via the SONGe digital tool. This work allowed to select, among different stochastic and deterministic algorithms of the scipy.optimize library, the differential evolution algorithm as operational algorithm due to its best performance. A non-exhaustive study was carried out to determine the best parameters for using this algorithm.

After correctly defining a Pareto front, its study leads to the redefinition of the cost functions associated with different constraints. A weighted aggregation method of the new objective functions was adopted. The use of SONGe as a decision support tool pointed out challenges in acquiring optimization results and also in presenting them to decision makers. The use of an interactive display to adjust weights in real time, thus providing a more accurate view of possible compromises, was highlighted as a possible solution. Several ways of improving the results were proposed, including analysis of the effects of mutation and crossing parameters, the successive use of different optimization algorithms, as well as the personalization of the initial population with individuals admitted as performing.

Finally, the optimization of the algorithmic part of SONGe has made possible to simplify and accelerate the identification of optimal design for MSFR. The next steps will include more in-depth studies on possible MSFR design by opening up the study to variable values and no longer only to constraints costs.

VII.3 Figures annexes

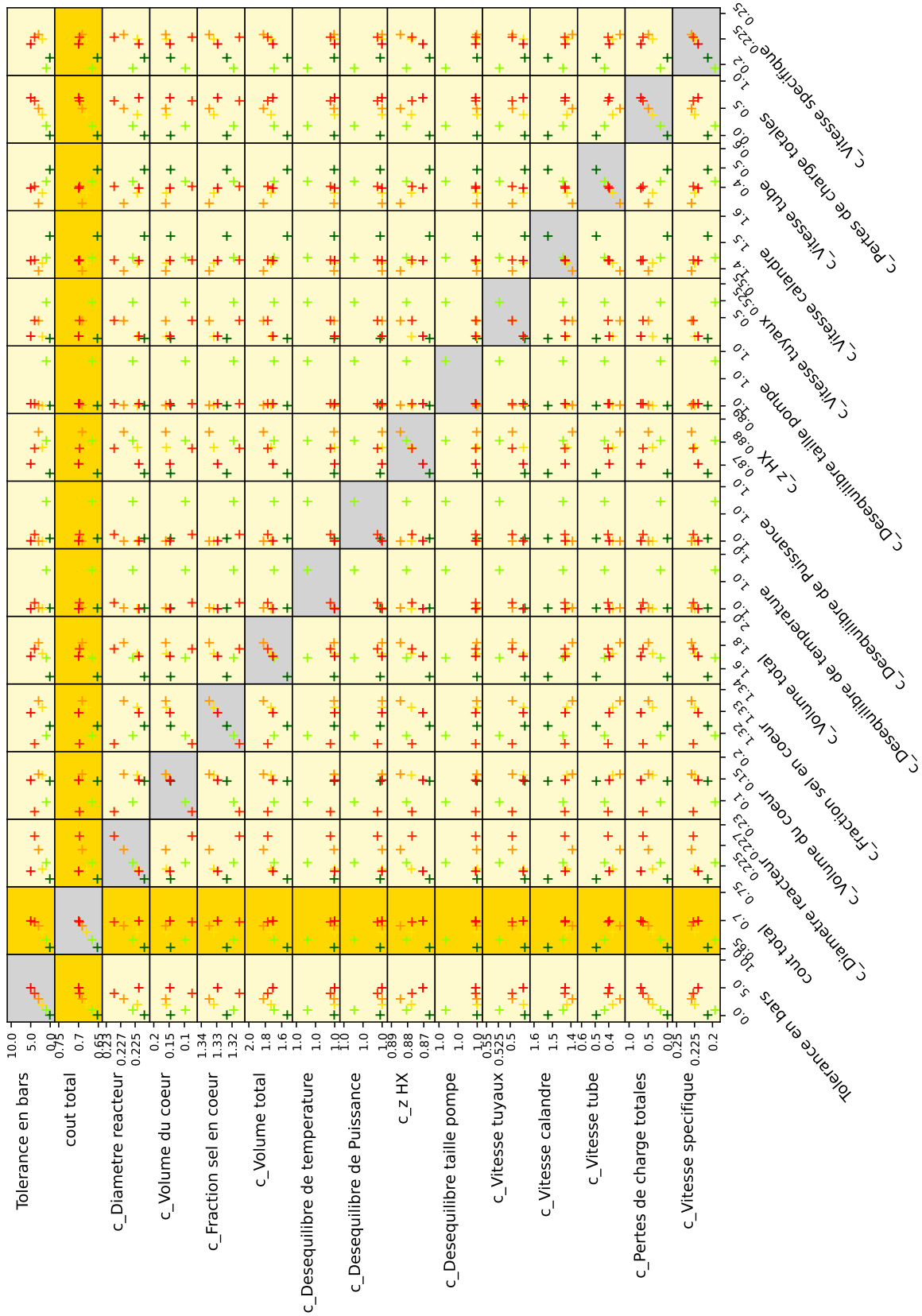

FIGURE 21 – Affichage complet des coûts pour différentes tolérance sur les pertes de charges.

VII.4 Compétences

VII.4.1 Évaluation de la compétence C1 - Concevoir ou réaliser des solutions d'ingénierie, permettant de répondre à un cahier des charges (Niveau 2)

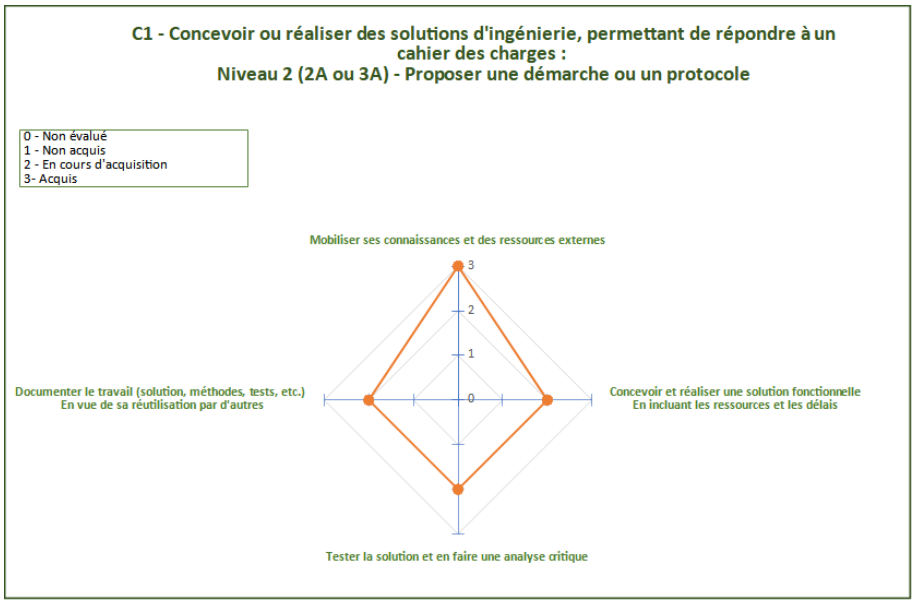

FIGURE 22 – Auto-évaluation de la compétence C1.

Lors de ce stage, j'ai repris un sujet déjà existant dont le cahier des charges s'articulait autour des points suivant :

— Rapidité d'obtention des configurations

— Qualité des configurations obtenues

— Libre paramétrisation des configurations

— Ergonomie/simplicité d'utilisation

Cependant, la documentation sur SONGe était limitée et souvent erronée, car le projet avait évolué sans que les modifications ne soient reportées dans les documents internes.

Pour surmonter ces problématiques, j'ai mobilisé diverses ressources externes. J'ai eu accès à une vaste bibliothèque d'articles scientifiques sur l'optimisation via Beluga (service en ligne fourni par l'UGA). J'ai également eu l'opportunité de discuter avec le créateur du code SONGe, Daniel Heuer, ce qui m'a permis de mieux comprendre les idées et les objectifs initiaux de SONGe. J'ai également consulté des sites et des forums sur les différentes bibliothèques d'optimisation, ainsi que des vidéos de conférences disponibles sur internet [38].

J'ai documenté le travail réalisé durant ce stage et ajouté des commentaires dans le code Python, en essayant de rendre les noms des variables, des paramètres et des fonctions les plus explicites possibles. Ce rapport constitue également un document informatif important sur SONGe.

La conception finale est fonctionnelle et répond aux tâches pour lesquelles elle a été créée. Toutes les ressources utilisées sont citées dans ce rapport.

La solution technique finale a été testée et analysée avec critique tout au long de mon stage, comme en témoigne le changement de direction effectué concernant le calcul

du coût afin de rendre possible le tracé du front de Pareto. Cette critique sur le calcul des coûts a été au cœur de la deuxième partie de mon stage. Il a été très important de vérifier plusieurs fois les résultats et de m'assurer que mon raisonnement était valable et vérifié. D'autant plus qu'aucune documentation préexistante ne permettait de comparer ou d'étayer mes propos, à l'exception des travaux préliminaires de Thomas Sornay, qui m'ont permis de vérifier la qualité et la rapidité des solutions trouvées dans la première partie de mon stage.

Pendant ma formation à Phelma, j'avais développé une maîtrise générale des réacteurs nucléaires et de la thermohydraulique, ainsi que des compétences de base en programmation avec Python et C. Ces compétences m'ont facilité l'apprentissage du Java par la suite. En revanche, l'optimisation n'était pas un domaine que j'avais abordé durant mes études. J'ai principalement appris sur ce sujet via l'étude de documentations.

VII.4.2 Évaluation de la compétence C2 - Mettre en œuvre une démarche de recherche fondamentale ou une démarche appliquée à des fins d'innovation (Niveau 1)

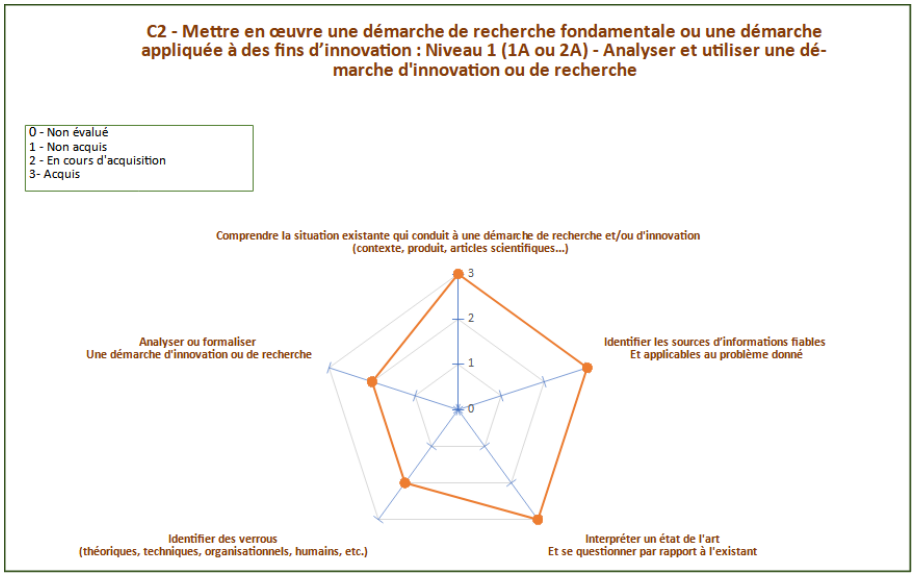

FIGURE 23 – Auto-évaluation de la compétence C2.

La première étape dans ma démarche de recherche à des fins d'innovation a été de rassembler l'ensemble de la documentation disponible sur le projet jusqu'à aujourd'hui. J'ai parcouru la bibliographie disponible sur la page MSFR du LPSC [39] et dialogué avec les différents membres de l'équipe afin de mieux comprendre le code SONGe, que ce soit dans son implémentation ou dans son utilisation. J'ai également pu discuter de leur état d'avancement et des concepts clefs de SONGe. Thomas Sornay a partagé avec moi ses précédents résultats qu'il avait présentés à Framatome lors de sa période au sein de Framatome Lyon. Il m'a introduit aux différents concepts de SONGe et expliqué les points clefs et les blocages qu'il rencontrait.

Une grande phase d'exploration de la structure de SONGe en autonomie a eu lieu, notamment au début de mon stage. Cela m'a permis d'assimiler personnellement son fonctionnement et ses mécanismes, tout en ayant la chance de pouvoir dialoguer sur des points de blocage. J'ai également pu parler avec Daniel Heuer, créateur de SONGe, qui

m'a enseigné davantage sur l'utilisation initiale de SONGe et son utilisation dans d'anciens projets.

J'ai identifié les sources fiables en m'appuyant sur des articles publiés dans des revues à comité de lecture, en citant les experts du domaine et leurs travaux, ainsi que les institutions et instances de référence. Tout ce travail est documenté dans la bibliographie de ce rapport.

Je me suis penché sur l'état de l'art en examinant les évolutions dans les travaux de chercheurs de différentes périodes. J'ai notamment mené cette réflexion pour l'algorithme d'évolution différentielle, qui a été étudié en profondeur par ses créateurs au fil des ans. La caractérisation et l'optimisation de ses paramètres ont ainsi évolué de manière significative au cours du temps.

J'ai également identifié plusieurs verrous techniques, notamment le calcul des coûts lié à des fonctions objectifs dépendantes des bornes et des tolérances. Cela empêchait la variation des coûts dans le cadre d'une étude du front de Pareto, ce qui représentait un obstacle majeur au cours de ce stage.

Un protocole de comparaison des résultats a été établi en amont, et la démarche a été divisée en sous-étapes afin de faire progresser le projet. Commencer par la rapidité et la qualité des solutions, car ces tâches étaient les plus simples, permettant une bonne prise en main et accélérant le processus suivant sur la caractérisation des configurations (cf. IV.3).

VII.4.3 Évaluation de la compétence C3 - Coopérer dans une équipe et en mode projet (Niveau 2)

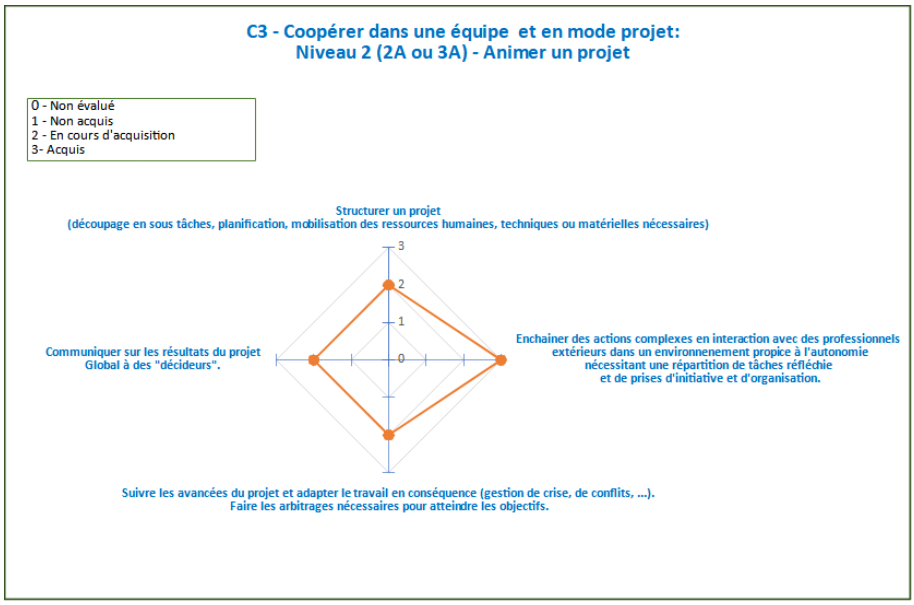

FIGURE 24 – Auto-évaluation de la compétence C3.

Correctement structurer mon stage et m'a permis d'être plus efficace dans mon travail. Pour ce faire, j'ai découpé les tâches de manière progressive et méthodique. Ce découpage a permis de planifier les différentes étapes de mon stage en tenant compte du temps nécessaire pour acquérir les bases théoriques et les connaissances nécessaires avant de passer à la pratique. La planification a également inclus des marges de temps pour gérer

les imprévus, assurant ainsi une gestion flexible du projet. Pendant mon stage, j'ai dû m'adapter à une situation imprévue : une blessure lors de la pratique du basketball le 21 mai, peu après le début de mon stage le 6 mai. Cette blessure a conduit à sept jours d'arrêt, entraînant une modification de mon planning et de l'organisation de mon stage.

J'ai eu l'opportunité de travailler avec une grande liberté et beaucoup d'autonomie. Cette autonomie m'a permis d'orienter mes initiatives en fonction de mes préférences personnelles. J'ai particulièrement apprécié avoir la possibilité d'orienter mon travail sur des sujets qui m'intéressaient le plus. Cette liberté m'a permis de m'intéresser davantage au sujet et de me concentrer sur l'optimisation, la présentation des résultats et le dialogue autour de la prise de décision, plutôt que sur la physique intrinsèque des réacteurs et la technologie MSFR. Ces aspects techniques ont toutefois été abordés pour garantir la cohérence des solutions et comprendre certaines problématiques numériques. J'ai également disposé d'une grande souplesse au niveau de ma durée de présence au laboratoire, toujours en respectant les horaires de présence obligatoires et minimums.

Tout au long de mon stage, j'ai bénéficié d'un encadrement disponible et réactif, avec Thomas Sornay et Axel Laureau toujours prêts à m'aider en cas de besoin. J'ai également eu l'opportunité de dialoguer rapidement avec les acteurs de Framatome sur SONGe lors de ma visite dans les locaux de Lyon. Cela m'a permis d'avoir un point de vue d'industriel sur mon stage et de réaffirmer certains points, notamment vis-à-vis de la non-définition des plages de variations des variables du réacteur. Ces échanges m'ont fourni les ressources nécessaires pour gérer les problématiques et les blocages que j'ai pu rencontrer. J'ai cependant rencontré des difficultés à faire remonter l'information jusqu'à Elsa Merle, responsable de l'équipe MSFR, dans les premières semaines de mon stage. Une fois le problème rapporté, j'ai pu apporter des corrections dans ma communication.

La communication des résultats de mon stage s'est faite de manière régulière, facilitée par la proximité des bureaux situés au même étage. Mes deux interlocuteurs principaux, Axel Laureau et Thomas Sornay, m'ont offert des points de vue et des approches différentes et complémentaires pour traiter les problèmes. Ils soulevaient les erreurs et les imprécisions dans ma démarche lorsque cela était nécessaire, me permettant d'être le plus précis possible. J'ai aussi utilisé le service de chat interne du laboratoire pour échanger avec eux. J'avais également à disposition des tableaux pour schématiser mon raisonnement et appuyer mes explications lorsque cela était nécessaire.

La souplesse laissée par l'équipe dans mon encadrement a permis de contourner facilement les problèmes en choisissant les solutions avec lesquelles j'étais le plus à l'aise. Ma connaissance des outils avec lesquels je travaillais s'est améliorée, notamment en Bash, Python, Java et VS Code. Cela a facilité la gestion de mes problèmes et renforcé ma confiance dans mes capacités.

VII.4.4 Évaluation de la compétence C5 - Tenir compte des transitions technologiques, environnementales, sociétales (Niveau 1)

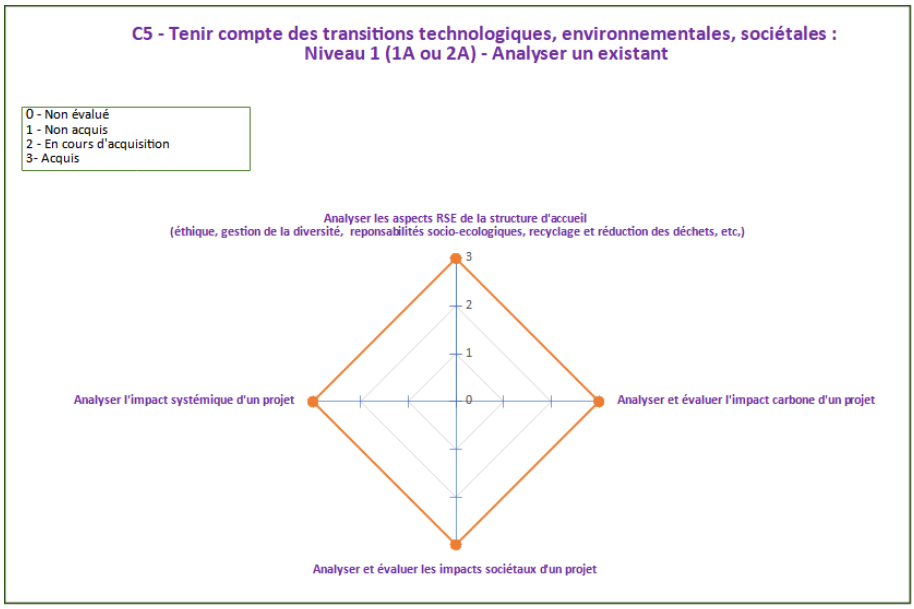

FIGURE 25 – Auto-évaluation de la compétence C5.

Durant mon stage au Laboratoire de Physique Subatomique et de Cosmologie (LPSC) à Grenoble, j'ai pu constater plusieurs initiatives mises en place en matière de responsabilité sociétale des entreprises (RSE). Ces initiatives couvrent divers aspects, allant de l'éthique de la recherche à la gestion de la diversité, en passant par les responsabilités sociales et écologiques ainsi que la réduction des déchets.

L'ensemble du personnel chercheur du laboratoire respecte la charte éthique du CNRS, garantissant ainsi une recherche intègre et responsable. Le LPSC met également en place des initiatives pour favoriser l'inclusion sociale et la parité. Par exemple, l'organisation de stages à destination des classes de seconde permet de faire découvrir les métiers scientifiques et de la recherche avant le choix d'options ou de filière en seconde. Cette initiative permet de toucher un public plus paritaire entre hommes et femmes et des catégories socio-professionnelles différentes, contribuant ainsi à une plus grande diversité dans le domaine scientifique [40] [41].

Le laboratoire encourage les mobilités douces en mettant à disposition plusieurs points d'attache pour vélos et un abri couvert. Un atelier équipé du matériel nécessaire pour effectuer des réparations sur les vélos est également accessible. Cette initiative incite le personnel à adopter des modes de transport plus écologiques, réduisant ainsi l'empreinte carbone liée aux trajets domicile-travail. Cela rejoint l'objectif global de réduire les émissions de gaz à effet de serre en encourageant des pratiques plus durables. De plus, le télétravail est aussi une pratique courante.

Le LPSC organise également un concours photo portant sur la biodiversité sur le site du laboratoire, mettant en avant la richesse de la faune locale, allant des insectes aux lapins et renards. Cette initiative est accompagnée d'un fauchage raisonné des pelouses, contribuant à la préservation de la biodiversité.

Le LPSC organise un « Café du Labo » tous les jeudis, favorisant la cohésion d'équipe et le partage d'idées. Cet événement régulier renforce l'esprit de cohésion au sein du

laboratoire et permet de faire remonter des remarques. Une journée annuelle de festivités est également organisée, contribuant au bien-être psychologique du personnel, notamment des doctorants, qui font face à une charge de travail et à une pression importante.

Lors de mon stage, j'ai évalué l'impact carbone lié aux calculs intensifs réalisés au laboratoire. Les fermes de calcul utilisées ont une consommation énergétique significative. J'ai calculé le temps total de calculs réalisés sur une machine Intel Xeon Gold 6252 @ 2.10 GHz, qui correspond à la machine type sur laquelle j'ai effectué mes calculs. Elles consomment en moyenne 150 W [42]. Au total, durant mon stage, en faisant l'inventaire de l'ensemble des calculs que j'ai effectués, j'arrive à approximativement 21 jours de temps de calcul sans arrêt sur une machine. En France, avec un coût monétaire estimé à 25,16 centimes d'euros par kWh chez EDF en tarification de base [43], avec une consommation de 150 W, cela se traduit par un coût en électricité de 19.02 euros. On estime à 32 gCO_2eq par kilowattheure le coût carbone de l'électricité produite en France [44]. En considérant uniquement que l'électricité consommée a été produite en France (la France étant majoritairement exportatrice d'électricité), la réalisation de mes calculs a un coût carbone de 2,419 kg de CO_2. Ces estimations incluent uniquement l'impact carbone de mes calculs. Elles ne prennent pas en compte l'impact environnemental de l'ensemble du matériel que j'avais à disposition, incluant un ordinateur, un clavier, une souris, un écran, ainsi que le coût carbone de production et d'entretien des infrastructures de calcul.

La cantine dont dépend le laboratoire, propose des paniers repas «Too Good To Go« en fin de service pour limiter le gaspillage alimentaire [45]. Cette initiative est un exemple de responsabilité environnementale et contribue à la réduction des déchets alimentaires. Elle révèle une conscience environnementale dans la gestion des ressources et une volonté de réduction de l'empreinte carbone. Le laboratoire utilise du café en grain plutôt que des capsules, ce qui réduit les déchets générés par les emballages de capsules jetables. Cette pratique est plus respectueuse de l'environnement et s'inscrit dans une démarche globale de réduction des déchets.

Le projet sur lequel j'ai travaillé, porté sur les réacteurs à sels fondus (MSFR), a des implications sociétales profondes. L'évolution des technologies nucléaires, telles que celles étudiées dans le cadre du MSFR, pourrait offrir des solutions pour réduire la dépendance aux ressources fossiles et améliorer la sécurité énergétique mondiale. Aussi, la gestion des déchets nucléaires est un sujet de préoccupation publique majeur. Une meilleure gestion et valorisation des déchets radioactifs pourraient réduire leur impact environnemental et atténuer les réticences sociétales envers l'expansion de l'énergie nucléaire. Leur revalorisation permet de limiter l'impact sur l'environnement et de remanier les fonds dédiés au traitement et à l'enfouissement. Ce point est crucial pour renforcer l'acceptabilité sociale des nouvelles technologies nucléaires. L'impact systémique du projet des réacteurs MSFR va au-delà de la simple production d'énergie. L'évolution des technologies nucléaires pourrait offrir des solutions pour réduire la dépendance aux ressources fossiles et améliorer la sécurité énergétique mondiale. Cette technologie pourrait soulager la pression sur les réserves d'uranium et amener une plus grande souveraineté énergétique à la France du fait de son stock important de déchets radioactifs. C'est une solution dans le scénario d'une demande croissante de combustible nucléaire. En outre, la vigilance accordée à la régulation de l'utilisation des technologies nucléaires est cruciale pour prévenir la prolifération nucléaire tout en maintenant un niveau de sécurité élevé dans les installations. Les réacteurs MSFR sont spécifiquement conçus en tenant compte de ces considérations, afin d'assurer leur intégration sûre et efficace au sein du système énergétique global [19].

Mon stage au LPSC m'a permis de développer une compréhension approfondie des

enjeux liés aux transitions technologiques, environnementales et sociétales, en particulier dans le cadre de la recherche nucléaire. En intégrant des initiatives de responsabilité sociétale, en évaluant l'impact carbone et sociétal des projets, et en analysant l'impact systémique, j'ai acquis les compétences requises pour exercer le métier d'ingénieur dans un contexte de transition énergétique global, où la responsabilité sociétale et environnementale sont parties intégrantes du métier.

Références

[1] Energy Institute : Statistical review of world energy. https://www.energyinst.org/statistical-review/energy-charting-tool/energy-charting-tool, 2022. Consulté le 16/05/2024.

[2] Nations Unies : Convention-cadre sur les changements climatiques : Accord de paris. https://unfccc.int/sites/default/files/french_paris_agreement.pdf, 2015. Consulté le 16/05/2024.

[3] Loi n° 2019-1147 du 8 novembre 2019 relative à l'énergie et au climat. https://www.legifrance.gouv.fr/loda/id/JORFTEXT000039355955/, 2019. Consulté le 16/05/2024.

[4] Oak Ridge National Laboratory : The first nuclear power for peaceful use. https://www.ornl.gov/blog/first-nuclear-power, 2024. Consulté le 26/09/2024.

[5] Nuclear Energy Agency (NEA) : *Uranium 2022 : Resources, Production and Demand*. OECD Publishing, Paris, 2023.

[6] E. Merle-Lucotte, D. Heuer, C. Le Brun et J. M. Loiseaux : Scenarios for a worldwide deployment of nuclear power. *International Journal of Nuclear Governance, Economy and Ecology*, 1(2):168–192, 2006.

[7] LCP, Assemblée nationale : Jean-marc jancovici est auditionné par la commission d'enquête de l'assemblée nationale. https://www.youtube.com/watch?v=851Q-nPNx7I, 2022. Consulté le 16/05/2024.

[8] Institut de radioprotection et de sûreté nucléaire (IRSN) : Savoir et comprendre : Déchets radioactifs. https://www.irsn.fr/savoir-comprendre/surete/dechets-radioactifs. Consulté le 16/05/2024.

[9] Institut de radioprotection et de sûreté nucléaire (IRSN) : Baromètre irsn. https://barometre.irsn.fr/barometre2023v3/#p=3, 2023. Consulté le 16/05/2024.

[10] Forum international génération IV : Rapport annuel. https://www.gen-4.org/gif/jcms/c_43523/2008-annual-report?details=true, 2008. Consulté le 16/05/2024.

[11] FranceTerme : Définition contre-réaction de réactivité. https://www.culture.fr/franceterme/terme/NUCL547, 2013. Consulté le 16/05/2024.

[12] Commissariat à l'énergie atomique et aux énergies alternatives (CEA) : Clefs cea - n° 53. 2006.

[13] A. Nuttin, D. Heuer, A. Billebaud, R. Brissot, C. Garzenne, C. Le Brun, D. Lecarpentier, E. Liatard, J.M. Loiseaux, O. Méplan, E. Merle-Lucotte et J. Wilson : Potential of thorium molten salt reactors. *Progress in Nuclear Energy*, 46:77–99, 2005.

[14] Hugo Pitois : *Conception et optimisation d'un réacteur de type MSFR en sels chlorures et en cycle uranium*. Thèse de doctorat, Université Grenoble Alpes [2020-2023], octobre 2023.

[15] NAAREA : Technologie naarea. https://www.naarea.fr/fr/technologie-naarea, 2024. Consulté le 26/09/2024.

[16] Stellaria : Accueil - stellaria. https://www.stellaria.fr, 2024. Consulté le 26/09/2024.

[17] Thorizon : Technology. https://thorizon.com/technology, 2024. Consulté le 26/09/2024.

[18] Orano Group : Smr / mini réacteurs : un atout pour le nucléaire. https://www.orano.group/fr/decodage/smr-mini-reacteurs-un-atout-pour-le-nucleaire, 2024. Consulté le 26/09/2024.

[19] Michel Allibert et al : Preliminary proliferation study of the molten salt fast reactor. *EPJ Nuclear Sci. Technol.*, 2020.

[20] R. T. Marler et J. S. Arora : The weighted sum method for multi-objective optimization : new insights. *Structural and Multidisciplinary Optimization*, 41:853–862, 2010.

[21] scipy.optimize — scipy reference guide. https://docs.scipy.org/doc/scipy/reference/optimize.html. Consulté le 14-08-2024.

[22] D. R. Jones, C. D. Perttunen et B. E. Stuckman : Lipschitzian optimization without the lipschitz constant. *Journal of Optimization Theory and Applications*, 79(1):157–181, October 1993. Communicated by L. C. W. Dixon.

[23] Stefan Endres : A simplicial homology algorithm for lipschitz optimisation. Master of engineering (chemical engineering), University of Pretoria, Pretoria, South Africa, October 2017. A dissertation submitted in partial fulfillment of the requirements for the degree Master of Engineering in the Department of Chemical Engineering, Faculty of Engineering, the Built Environment and Information Technology.

[24] scipy.optimize.dual_annealing — scipy v1.14.0 manual. https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.dual_annealing.html#rbaa258a99356-5. Consulté le 16/08/2024.

[25] David J. Wales et Jonathan P. K. Doye : Global optimization by basin-hopping and the lowest energy structures of lennard-jones clusters containing up to 110 atoms. *The Journal of Physical Chemistry A*, 101(28):5111–5116, July 1997.

[26] Thomas Sornay : Calcul d'optimisation songe pour le prédimensionnement d'aramis-a. Rapport technique, LPSC, 2023.

[27] David Freynet : *Développement d'une méthodologie pour l'optimisation multicritère de scénarios d'évolution du parc nucléaire*. Thèse de doctorat, Université Grenoble Alpes.

[28] Yair Censor : Pareto optimality in multiobjective problems. *Applied Mathematics and Optimization*, 4:41–59, 1977. Department of Mathematics, University of Haifa, Mt. Carmel, Haifa, Israel.

[29] I. Das et J. E. Dennis : A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation in multicriteria optimization problems. *Structural Optimization*, 14:63–69, 1997.

[30] Daniel Heuer : Méthode de dimensionnement du msfr. Rapport technique, LPSC, 2012.

[31] David H. Wolpert et William G. Macready : No free lunch theorems for search. *In Proceedings of the 1995 International Conference on Evolutionary Computation*, 1995.

[32] Rainer Storn et Kenneth Price : Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359, 1997.

[33] scipy.optimize.differential_evolution — scipy v1.14.0 manual. https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.differential_evolution.html. Consulté le 16/08/2024.

[34] Ji Qiang et Chad Mitchell : A unified differential evolution algorithm for global optimization. *IEEE Transactions on Evolutionary Computation.*, 6 2014.

[35] Kenneth V. Price : *Differential Evolution*, pages 187–214. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[36] Jürgen Branke, Kalyanmoy Deb, Kaisa Miettinen et Roman Słowiński, éditeurs. *Multiobjective Optimization : Interactive and Evolutionary Approaches*, volume 5252 de *Lecture Notes in Computer Science*. Springer Berlin, Heidelberg, Berlin, Heidelberg, 1 édition, 2008. Published : 15 October 2008 (Softcover), 18 October 2008 (eBook).

[37] Leena Tanner : Selecting a text-processing system as a qualitative multiple criteria problem. *European Journal of Operational Research*, 50:179–187, 1991. Helsinki School of Economics, Runeberginkatu 14-16, SF-O0100 Helsinki, Finland.

[38] Sylvain Gubian : Dual annealing for global optimization problems. https://www.youtube.com/watch?v=OP4Z4_Csn1s, 2023. Consulté le 16/08/2024.

[39] Équipe MSFR du LPSC : Bibliographie. https://lpsc.in2p3.fr/index.php/fr/groupes-de-physique/msfr/lang-fr-anciennes-pages-lang-lang-en-old-pages-lang-reacteurs-msfr/rsf-reacteurs-a-sels-fondus/lang-fr-msfr-bibliographie-lang-lang-en-msfr-bibliography-lang. Consulté le 16/08/2024.

[40] Ministère de l'Enseignement Supérieur de la Recherche et de l'Innovation : La parité dans l'enseignement supérieur, 2023. Consulté le : 31 août 2024.

[41] Observatoire des inégalités : L'inégal accès au bac des catégories sociales. https://inegalites.fr/L-inegal-acces-au-bac-des-categories-sociales, 2023. Consulté le : 31 août 2024.

[42] PassMark Software : Intel xeon gold 6252 @ 2.10ghz cpu benchmark. https://www.cpubenchmark.net/cpu.php?cpu=Intel+Xeon+Gold+6252+%40+2.10GHz&id=3532&cpuCount=2, 2024. Consulté le : 31 août 2024.

[43] EDF SA : Grille de prix de l'offre de fourniture d'électricité Tarif Bleu, 2024.

[44] RTE France : Bilan électrique 2023 - Émissions. https://analysesetdonnees.rte-france.com/bilan-electrique-2023/emissions#Introduction, 2024. Consulté le : 31 août 2024.

[45] Too Good To Go : Site officiel de too good to go. https://www.toogoodtogo.com, 2024. Consulté le : 31 août 2024.