

ナトリウム冷却高速炉の設計最適化フレームワークの開発

(3) ユーザインターフェースを備えたプロトタイプの開発

Development of a Design Optimization Framework for Sodium-cooled Fast Reactors

(3) Development of a prototype with user interface

堂田哲広¹⁾, 中峯由彰²⁾, 吉村一夫³⁾, 桑垣一紀⁴⁾, 浜瀬枝里菜⁵⁾,
横山賢治⁶⁾, 菊地紀宏⁷⁾, 森健郎⁸⁾, 橋立竜太⁹⁾, 田中正暁¹⁰⁾

Norihiro Doda, Yoshiaki Nakamine, Kazuo Yoshimura, Kazuki Kuwagaki, Erina Hamase, Kenji Yokoyama, Norihiro Kikuchi, Takero Mori, Ryuta Hashidate, and Masaaki Tanaka

- 1) 博(工) 日本原子力研究開発機構(〒311-1393茨城県東茨城郡大洗町成田町4002, E-mail: doda.norihiro@jaea.go.jp)
- 2) 株式会社エヌデーデー(〒310-0803茨城県水戸市城南1-1-6, E-mail: y.nakamine@nddhq.co.jp)
- 3) 日本原子力研究開発機構(〒311-1393茨城県東茨城郡大洗町成田町4002, E-mail: yoshimura.kazuo@jaea.go.jp)
- 4) 博(工) 日本原子力研究開発機構(〒311-1393茨城県東茨城郡大洗町成田町4002, E-mail: kuwagaki.kazuki@jaea.go.jp)
- 5) 博(工) 日本原子力研究開発機構(〒311-1393茨城県東茨城郡大洗町成田町4002, E-mail: hamase.erina@jaea.go.jp)
- 6) 博(工) 日本原子力研究開発機構(〒311-1393茨城県東茨城郡大洗町成田町4002, E-mail: yokoyama.kenji09@jaea.go.jp)
- 7) 日本原子力研究開発機構(〒311-1393茨城県東茨城郡大洗町成田町4002, E-mail: kikuchi.norihiro@jaea.go.jp)
- 8) 日本原子力研究開発機構(〒311-1393茨城県東茨城郡大洗町成田町4002, E-mail: mori.takero@jaea.go.jp)
- 9) 日本原子力研究開発機構(〒311-1393茨城県東茨城郡大洗町成田町4002, E-mail: hashidate.ryuta@jaea.go.jp)
- 10) 博(工) 日本原子力研究開発機構(〒311-1393茨城県東茨城郡大洗町成田町4002, E-mail: tanaka.masaaki@jaea.go.jp)

As a part of the development of the "Advanced Reactor Knowledge- and AI-aided Design Integration Approach through the whole plant lifecycle (ARKADIA)" to utilize the knowledge obtained through the sodium-cooled fast reactors (SFRs) and combine the latest numerical simulation technologies, ARKADIA-Design is being developed to support the optimization of SFRs in the conceptual design stage. ARKADIA-Design consists of three systems of Virtual Plant Life System (VLS), Enhanced and AI-aided optimization System (EAS), and Knowledge Management System (KMS). A design optimization framework controls the linkage among the three systems through the interfaces in each system. In this paper, we have developed a prototype of the framework for core design optimization using the coupled analysis functions in VLS and optimization control function in the linkage of EAS and VLS to investigate the applicability of the framework to the SFR design optimization process.

Key Words : User interface, Coupling codes, Framework, Core design optimization, Fast reactor

1. 緒言

日本原子力研究開発機構では、ナトリウム冷却高速炉(SFR)開発で得た豊富な知識(ナレッジ)を活用するとともに、最新の数値シミュレーション技術を組み合わせた統合評価手法(ARKADIA)[1]を開発している。ARKADIAは、最適化検討を進める評価支援・応用システム(EAS: Enhanced and AI-aided optimization System)、最新の解析技術によりプラント挙動を予測する仮想プラントライフシステム(VLS: Virtual plant Life System)、SFRをはじめとする原子炉開発の知見を集約したナレッジマネジメントシステム(KMS: Knowledge Management System)で構成され、これらを連携させることで革新炉の統合的な評価を行うものである。

ARKADIAの開発では、SFRの概念設計段階の最適化を支援するARKADIA-Designと、安全性評価を支援するARKADIA-Safety、知識ベースを活用するためのARKADIA-KMSの開発を行ってきており[2]。本研究では、ARKADIA-Designにて設計最適化を支援するために必要

な機能として、設計最適化フレームワークの整備を進めてきた[3,4]。図-1にARKADIA-Designのシステムと設計最適化フレームワークの構成を示す。設計最適化フレームワークは、ARKADIA-DesignのEAS、VLS、KMSの3つのシステムを連携させる機能をもつプログラム群で、各システムに配置したインターフェースを通じ、共通インターフェースにより任意の評価フローに沿って各システム内のモジュール(解析コードや評価ツール等)を実行する機能を有する。

前報[3]では、VLSにおいて、評価の目的に応じて必要な詳細度のモデルを有する解析コードを組合せた連成解析を行い、プラント挙動解析を可能とする「連成解析制御機能」について報告した。また、前報[4]では、VLSによるプラント挙動解析を実行した結果を次の解析条件に反映し、最適解を得るまで繰返し解析を行うためのVLSとEASとの連携を可能とする「最適化解析制御機能」について報告した。

本報では、これまでに開発してきた個別の機能である

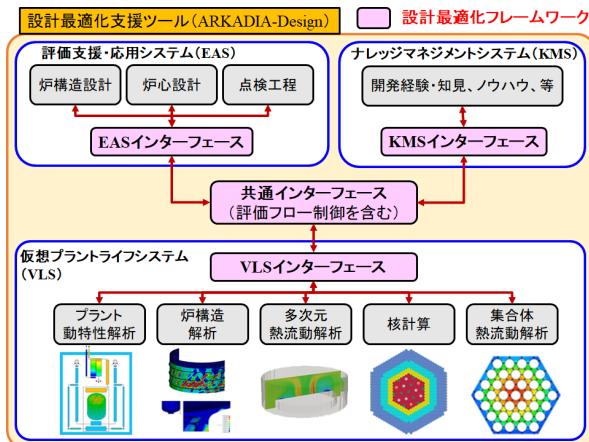


図-1 ARKADIA-Design のシステム構成と
設計最適化フレームワーク

VLSでの連成解析及びVLSとEASとの連携による最適化の制御機能を基に、これらを統合した設計最適化フレームワークのプロトタイプを構築し、SFRの炉心設計最適化における代表例題に適用することで、設計最適化フレームワークのSFR設計最適化プロセスへの適用性を確認する。

2. ARKADIA-Designの設計最適化

(1) 最適化評価フロー

図-2に示す評価フロー[1]に従い、4つのステップで設計最適化を行う。ステップ1で設計対象の最適化のための目的関数を設定する。ステップ2で設計に関する情報をデータベース等から入手するとともに、それに基づき評価条件を選定する。ステップ3で解析を実行し、設計対象の状態や性能を評価する。ステップ4でステップ3の解析結果をもとに目的達成度を確認する。未達成の場合は評価条件の変更等を行いステップ2へ戻り、目的が達成するまでこれを繰り返す。

(2) 設計最適化フレームワーク

ARKADIA-Designでは、図-2の評価フローに従って自動計算を行うため、図-3に示すようにこれまでに整備した連成解析制御機能と最適化解析制御機能を統合して設計最適化フレームワークを構成する。最適化解析制御機能は、EAS、VLS、KMSの各インターフェースに実装される。連成解析制御機能は、VLSのインターフェースに実装される。なお、KMSインターフェースの「(e) 関連情報の抽出」機能は現在整備中である。表-1に評価フローのステップと各インターフェースの機能との関係を示す。ユーザが共通インターフェースの評価フロー制御機能を起動することで、評価フローが開始され、ステップ1とステップ2から4までの繰り返しを最適化解析制御機能が実行を制御し、ステップ3の評価実施を連成解析制御機能が実行を制御する。ARKADIA-Designが対象とする炉心設計、炉構造設計、点検工程の最適化における各例題[2]は、解析環境、最適化問題の内容及び適用する最適化アルゴリズ

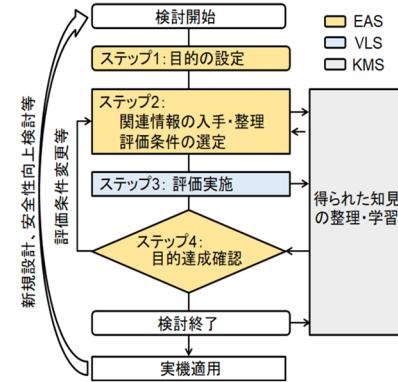


図-2 ARKADIA-Design の最適化評価フロー

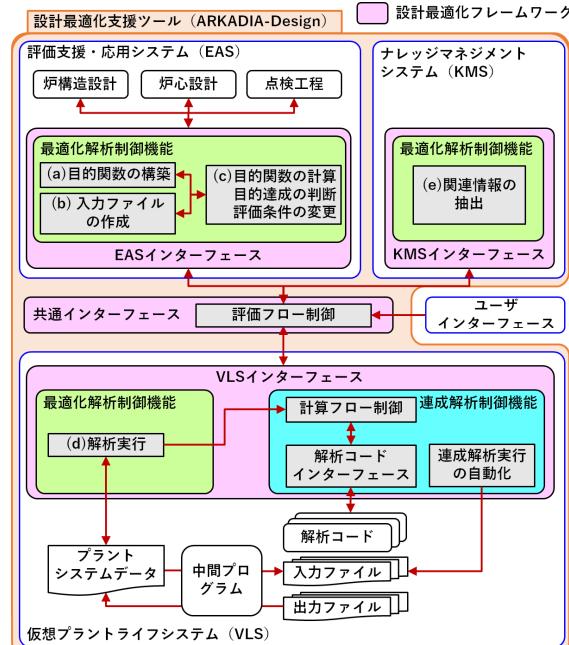


図-3 設計最適化フレームワークの
連成解析制御機能と最適化解析制御機能

ムの種類、プラント状態を評価する解析コード、計算の実行方法、評価結果の処理方法等が問題ごとに異なるため、設計最適化フレームワークの各制御機能は問題に合わせて設定することになる。ただし、基本的な部分は共通化が可能であり、必要な機能の構成を予め設定(プリセット)しておき、ユーザが実施する評価等に合わせて、必要な機能を部分的に変更するのが効率的である。そこで、表-2に示すように、設定内容をシステム開発者とユーザが行うもので分け、さらにユーザの利便性向上のため、ユーザが設定する内容についてはユーザインターフェースを構築した。

3. ユーザインターフェースの構築

炉心設計最適化における例題を対象に、設計最適化フレームワークの各制御機能を設定するためのユーザインターフェースを構築した。

(1) 炉心設計最適化例題

炉心設計分野の例題として、冷却材流量喪失時炉停止

表-1 評価フローにおける各インターフェースの役割

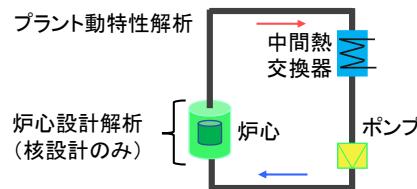
最適化ステップ	インターフェース	機能
-	共通	【評価フロー制御】評価フローを起動する。
1	EAS	【目的の設定(図-3の(a))】ユーザ入力から評価対象とする機器と設計変数を識別し、目的関数を設定する。
2	KMS	【関連情報の抽出(図-3の(e))】評価に必要な関連情報をデータベースから抽出し、EASへ受渡す。
	EAS	【評価条件の選定(図-3の(b))】KMSから関連情報を受け取り、プラント挙動解析の評価条件を選択及び入力データを作成する。
3	VLS	【解析実施(図-3の(d))】連成解析制御機能を呼び出し、プラント挙動を解析する。
4	EAS	【目的関数の計算、目的達成の確認、評価条件の変更(図-3の(c))】VLSの解析結果から目的関数の値を計算し、目的の達成を判断。必要に応じ、評価条件を変更し、ステップ2以降のステップを繰り返す。評価条件の変更を、適切な最適化アルゴリズムで行うによって、最適解探索を効率化する。

失敗事象時の炉心損傷回避と高い炉心性能を実現可能なSFRの炉心仕様の最適化を設定している[5]。ここでは簡単化のため、設計解析は核設計と安全評価とし、プラント体系としては図-4に示すような炉心と、除熱源となる中間熱交換器1基と冷却材を駆動するポンプ1台を配置する1ループの炉心冷却系統である。炉心及び冷却系統の仕様等は既存SFRプラントを参考に設定している。目的関数は炉心性能指標として炉心への初装荷プルトニウム(Pu) フィッサイル重量を最小化することとし、設計変数は炉心燃料集合体の燃料ピンピッチ(p)及び炉心高さ(h)とした。核的及び熱的制約条件としてそれぞれナトリウムボイド反応度及び燃料ピン被覆管最高温度の制限値を設定されている。これは単目的2変数2制約条件付の最適化問題となる。

図-5に炉心設計最適化の評価フローを示す。ステップ3の評価実施の核設計解析では核計算コードMARBLE[6]を

表-2 設計最適化フレームワークの設定内容

設定項目	ユーザ	システム開発者
①解析環境	作業ディレクトリ等の設定	システム環境設定
②最適化問題	最適化問題の設定(設計変数、目的関数、制約条件、最適化アルゴリズム)	最適化ライブラリ等の設定 設計変数の範囲設定の制限
③解析評価	連成解析のオプションの指定(解析モデル、解析条件)	基準データの作成(解析インプット、連成解析計算フロー)
④計算実行	実行オプションの指定(CPU数、等)	計算実行スクリプトの作成
⑤評価結果確認	可視化オプションの指定(グラフ等)	可視化スクリプトの作成



目的関数:
初装荷Puフィッサイル重量を最小化
設計変数:
燃料ピンピッチ(p)
炉心高さ(h)
制約条件:
ボイド反応度 < 制限値
被覆管最高温度 < 制限温度

図-4 炉心設計の最適化例題

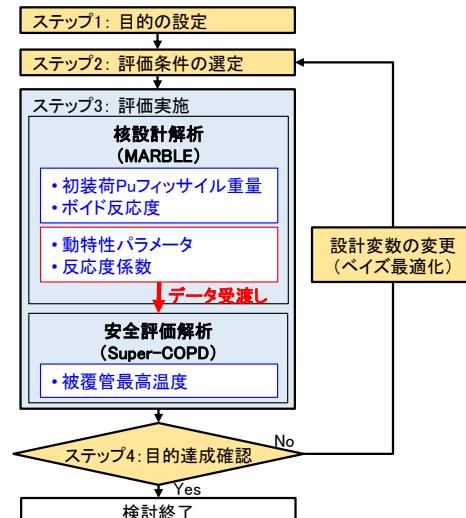


図-5 炉心設計最適化の評価フロー

用い、安全評価解析ではプラント動特性解析コードSuperCOPD[7]を用いた。核設計解析で得られた動特性パラメータ及び反応度係数は、連成解析制御機能を用いて安全評価解析に受渡し、解析に反映させた。また、ステップ4からステップ2への設計変数の変更では、最適化解析制御機能を用いて、ステップ3で得た解析結果を基にTriesteライブラリ[8]のベイズ最適化アルゴリズムに従って設計変数を変更させた。

(2) 炉心設計最適化例題のユーザインターフェース設定

図-6に、3.(1)の炉心設計の最適化例題を対象とした、ユーザインターフェースの構成を示す。ユーザインターフェースは端末の種類を選ばず、また視覚的に操作できるよう、Webブラウザを介して設定を行うWebユーザインターフェースとして構築した。これに、設計最適化フレームワークと同様にPython言語で構築されているWebフレームワークFlask[9]を使用した。ユーザは表-2の設定項目①～⑤を画面の上から順に設定することができる。①解析環境の設定では、システムのルートディレクトリはシステム側で設定し、ユーザはそれ以降の作業ディレクトリ及びケースディレクトリを入力できるようにした。②最適化問題の設定では、ユーザが目的関数を選択し、設計変数の設計範囲を入力するようにした。また、炉心特性や安全評価上の制約条件を必要に応じて選択し、制限値を入力するようにした。ここで、設計変数の範囲は燃料ピン本数等の別の設計要求によって制限されるが、システム上でこれらの制限が入力画面に反映されるようプログラミングした。また、これらの設定内容が、ユーザが選択する最適化アルゴリズムのライブラリの設定へ自動で反映されるようプログラミングした。③解析評価の設定では、連成解析制御機能により、図-7(a)に示す設定ファイルを作成することで様々な解析コードを組み合せた解析が可能である。設定ファイルは図-7(b)に示すように使用する解析コードとそれらの連結状態を計算要素の組合せとして記述するが、設定ファイルの作成はユーザにとって負担となるため、設定ファイルはシステム開発者が事前に作成し、ユーザは用意された設定ファイルを選択することとした。また、設定ファイルの概要を確認できるよう、図-8の計算フロー図を自動生成させるようにプログラミングした。④計算実行の設定では、事前に用意した計算実行スクリプトをユーザが選択し、必要に応じて使用するCPU数等の変更を加え、計算機に計算を投入できるようにした。⑤評価結果確認では、ユーザが評価結果を確認できるよう、グラフを自動作成する可視化スクリプトをプログラミングした。

4. 炉心設計最適化例題を対象とした適用性検討

3.(2)で構築した炉心設計最適化例題を対象としたユーザインターフェースを含む設計最適化フレームワークのプロトタイプを用いて、炉心設計最適化問題の設計変

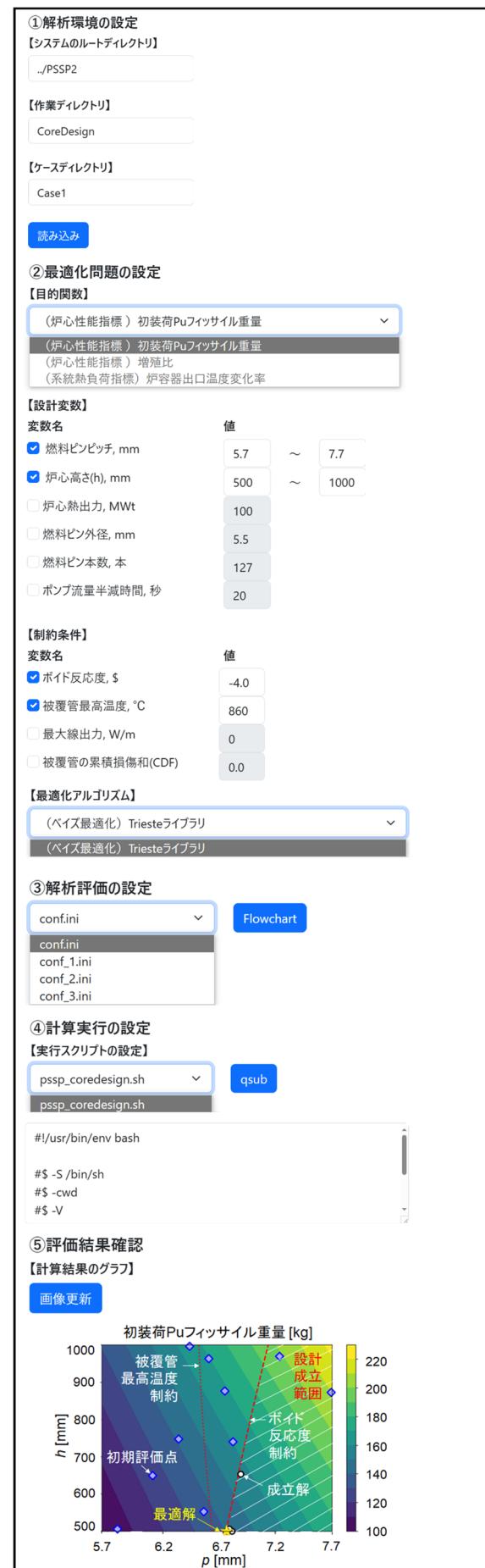


図-6 炉心設計例題のユーザインターフェース

```

[[START]]
function = initialize_001
out      = COUNTER

[[EXIT]]
function = finalize_001

[[COUNTER]]
start    = 0
stop     = 1
out.cont = PROCESS_MARBLE
out.exit  = EXIT

[[PROCESS_MARBLE]]
comm   = SCRIPTMARBLE_1
setup  = marble_setup(**)
exe    = %MARBLEXE
rundir = .
in     = COUNTER
out    = PROCESS_MARBLE_SCOPD

[[PROCESS_MARBLE_SCOPD]]
iproc  = MARBLE2SCOPD()
in     = PROCESS_MARBLE
out    = PROCESS_SCOPD

[[PROCESS_SCOPD]]
comm   = FILESCOPD_1
setup  = scopd_setup(**)
exe    = /bin/S-COPD 1>& out.txt
rundir = .
in     = PROCESS_MARBLE_SCOPD
out    = PROCESS_SCOPD_MARBLE

[[PROCESS_SCOPD_MARBLE]]
iproc  = SCOPD2MARBLE()
in     = PROCESS_SCOPD
out    = COUNTER

```

- 計算要素の組合せで計算フローを表現
- 各計算要素のinとoutで計算要素間の連結を設定

計算要素	内容
START	計算を開始
EXIT	計算を終了
COUNTER	反復数をカウント
PROCESS	解析コード、受渡しデータ変換スクリプトを実行

(a) 設定ファイル

(b) 計算要素の種類

図-7 連成解析の計算フロー設定

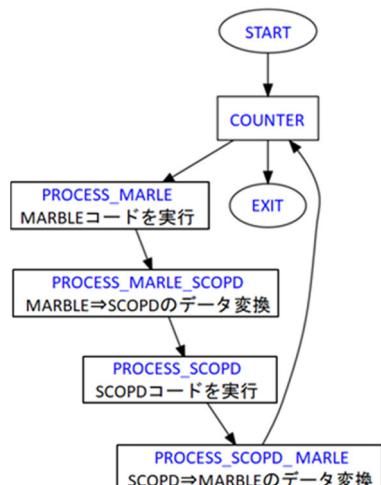


図-8 連成解析の計算フロー図

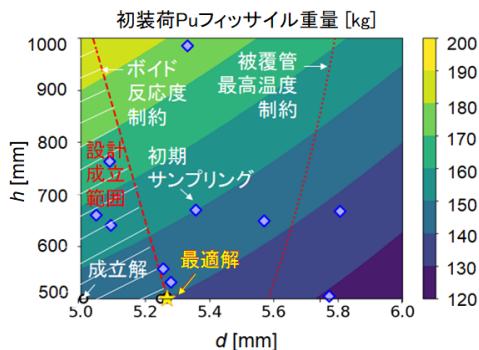


図-9 設計変数を変更した場合の評価結果

数の燃料ピンピッチ (p) を燃料ピン径 (d) に変更した最適化を実施した。図-9に設計最適化フレームワークによる自動実行で得られた評価結果を示す。設計変数を変更するためには設計最適化フレームワークの機能の設定を個別に変更せずに、ユーザインターフェースを介して設計変数の選択を変更するのみで実行可能であることを確認できた。また、設計変数の設定範囲を任意に限定すること、制約条件の項目選択や設定値を変更することも同様に、ユーザインターフェースを介して実行が可能であることを確認した。

以上により、炉心設計最適化例題に対し、設計最適化フレームワークの制御機能の基本的な設定により、ユーザは実施する評価等に合わせて必要な機能をユーザインターフェースを通じて部分的に変更することで、評価を効率よく実施できることを確認した。

5. 結言

設計最適化フレームワークのプロトタイプをこれまでに開発した連成解析及び最適化解析の制御機能にユーザインターフェースを加えて構築し、設計最適化フレームワークが炉心設計最適化に対して機能することを確認した。今後は、KMSとの連携機能の整備を行い、現在はユーザが実施している過去の開発経験・知見、ノウハウ等の抽出や、評価結果を知見の一部とするKMSの更新を行う機能を整備する予定である。

参考文献

- [1] 大島 宏之, 浅山 泰, 古川 智弘, 田中 正暁, 高田 孝, 江沼 康弘: AI支援型革新炉ライフサイクル最適化手法ARKADIAの開発 (1) 全体計画, 2021年日本原子力学会春の年会, 1C01, 2021.
- [2] 田中 正暁, 大木 繁夫, 宮崎 真之, 高屋 茂, 横山 賢治, 安藤 勝訓: AI支援型革新炉ライフサイクル最適化手法ARKADIAの開発 (2) 設計最適化支援ツールARKADIA-Designの開発計画, 2021年日本原子力学会春の年会, 1C02, 2021.
- [3] 堂田 哲広, 中峯 由彰, 井川 健一, 岩崎 隆, 村上 諭, 田中 正暁, ナトリウム冷却高速炉の設計最適化フレームワークの開発 (1) 連成解析制御機能の開発, 計算工学講演会論文集, Vol.27, A-10-02, 2022.
- [4] 堂田 哲広, 中峯 由彰, 桑垣 一紀, 浜瀬 枝里菜, 菊地 紀宏, 吉村 一夫, 松下健太郎, 田中 正暁, ナトリウム冷却高速炉の設計最適化フレームワークの開発 (2) 最適化解析制御機能の開発, 計算工学講演会論文集, Vol.28, G-02-01, 2023.
- [5] 浜瀬 枝里菜, 桑垣 一紀, 堂田 哲広, 横山 賢治, 田中 正暁: ARKADIAにおける炉心設計最適化プロセスの整備 (1) 核-プラント動特性の連携解析による最適化検討, 日本原子力学会 2023年春の年会, 2M13, 2023.

- [6] Yokoyama, K. et al.: Development of comprehensive and versatile framework for reactor analysis, MARBLE, Annals of Nuclear Energy, 66, pp.51-60, 2014.
- [7] Yamada, F. et al.: Development of natural circulation analytical model in Super-COPD code and evaluation of core cooling capability in Monju during a station blackout, Nuclear Technology, Vol.188.3, pp.292-321, 2014.
- [8] Picheny, Victor, et al.: Trieste: Efficiently Exploring the Depths of Black-box Functions with TensorFlow, arXiv preprint arXiv:2302.08436, 2023.
- [9] <https://flask.palletsprojects.com/en/3.0.x/>