第二回 Quantum CAE 研究会

AIで加速する
量子・最適化アルゴリズム開発

株式会社 Jij
CEO 山城　悠

会社概要

会社名	株式会社Jij (ジェイアイジェイ)
事業内容	最適化AI事業 ・最適化ソフトウェアJijZeptの提供. ・最適化ソリューションサービスOptXの提供 量子コンピューティング事業 ・量子コンピュータ向けミドルウェアの提供. ・量子アルゴリズムの共同研究・PoCサービス
事務所	日本オフィス(東京・田町) 英国オフィス
設立	2018年11月
従業員数	40＋

社名由来

$$H = \sum_{i,j} J_{ij}\sigma_i\sigma_j$$

「イジングモデル」という
統計物理学における数理モデルに由来

JijZept

JijZeptが提供するすべての機能を簡単に利用できる環境

JijZept SDK※1

最適化AIに必要なプロセスを
全てカバーする開発キット

- 最適化問題の対話的モデリング
- ソルバーの統一的呼び出し
- 量子ゲートの活用
- 数値計算の効率的管理

https://www.jijzept.com/products/sdk/

JijZept Tools※2

最適化計算の分析や専門的なアルゴリズムを
簡単に使えるアドバンスドなツール群

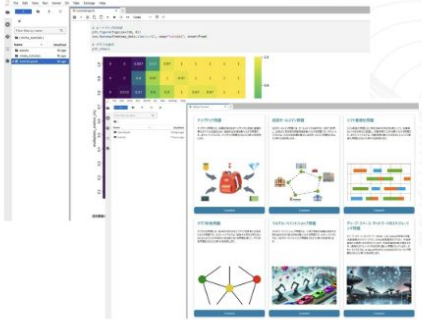
- BIツールライクな分析・探索・可視化
- さまざまなブラックボックス最適化の手法
- 列生成法などの数理最適化アルゴリズム

https://jij-inc.github.io/JijZeptTools/intro.html

JijZept IDE※2

ログインするだけで最適化AIの
開発を実施できる統合環境

すべての機能・ツールがすぐに利用可能



Solvers powered by Jij※2

Jijの経験と技術力を結集させた
最適化ソルバー

- 実用的な組合せ最適化問題の求解
- 複数のGPUベースのイジング最適化アルゴリズム

Quantum Simulator※2

テンソルネットワークベースの
量子回路シミュレータ

- 実機のトポロジーに合わせた効率的なシミュレーション
- 量子計算実行データから学習したノイズエミュレーション
- 多くの量子回路SDKに対応した簡便なインターフェース

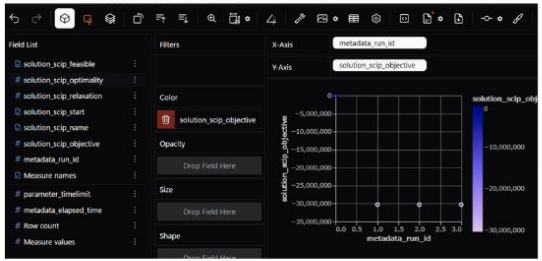
※1 JijZeptの契約に関わらず無料でご利用いただけます
※2 JijZeptの有償契約をご締結いただいたユーザー様のみご利用いただけます

Jij

JijZept Toolsの紹介

BIツールライクな分析

トポロジーを指定することができるため、実機のトポロジーに合わせて効率的な量子回路シミュレーションを実行できます。



ブラックボックス最適化

- **F**actrization **M**achine **O**ptimization
- **B**aysian **O**ptimization for **C**ombinatorial **S**tructures

  今後も追加予定

ワークフロー管理

ノード単位の処理フローをDAGで定義して、ステート管理と併せて一貫した実行を可能とします。

```
fmo_wf = FMOWorkflow(n, latent_dim=5, flip_postprocess4uniq=True)
fmo_wf.workflow.display_workflow("LR")
```

https://jij-inc.github.io/JijZeptTools/intro.html

ドキュメントを随時アップデートしております

JijZept SDK

pip install “jijzept-sdk[all]”

今すぐローカルで使えます！

代数的数理モデラー

数理最適化共通スキーマ (中間表現)

数理最適化ソルバー
イジングマシン
量子アルゴリズム

JijZept SDK

pip install “jijzept-sdk[all]”

今すぐローカルで使えます！

2 モデリング

```python
import jijmodeling as jm

d = jm.Placeholder("d", ndim=1)
n = d.shape[0]
x = jm.BinaryVar("x", shape=(n,))
i = jm.Element("i", (0, n))

problem = jm.Problem("sample")
# Objective Function
problem += jm.sum(i, d[i]*x[i])
# Constraints
problem += jm.Constraint(
  "c1", x[:].sum() <= 1
)
```

JijModelingを使用して、数式と似た表現で最適化問題を記述できます。

JijModeling

3 データの用意 / ソルバー選択と実行

SCIP | HiGHS | Gurobi

SCIP

Open-source mathematical optimization solver. Supports linear programming and mixed-integer programming.

```python
from ommx_pyscipopt_adapter import (
    OMMXPySCIPOptAdapter
)

# Create an instance
instance_data = {
    "d": [1, -2, 3],
}
interpreter = jm.Interpreter(instance_data)
instance = interpreter.eval_problem(problem)

# Solve with SCIP
solution = OMMXPySCIPOptAdapter.solve(instance)
```

複数のソルバーに対応しており、最適なソルバーを選択できます。

OMMX Solver

4 結果の解析

```python
print(solution.decision_variables)

# id    kind  lower  upper name subscripts value
# ------------------------------------------------
# 0   binary    0.0    1.0    x       [0]   0.0
# 1   binary    0.0    1.0    x       [1]   1.0
# 2   binary    0.0    1.0    x       [2]   0.0
```

最適解や目的関数値を簡単に取得でき、結果の可視化や分析も可能です。
MINTOを使用して数値実験の管理も簡単に行えます。

JijModeling

2 モデリング

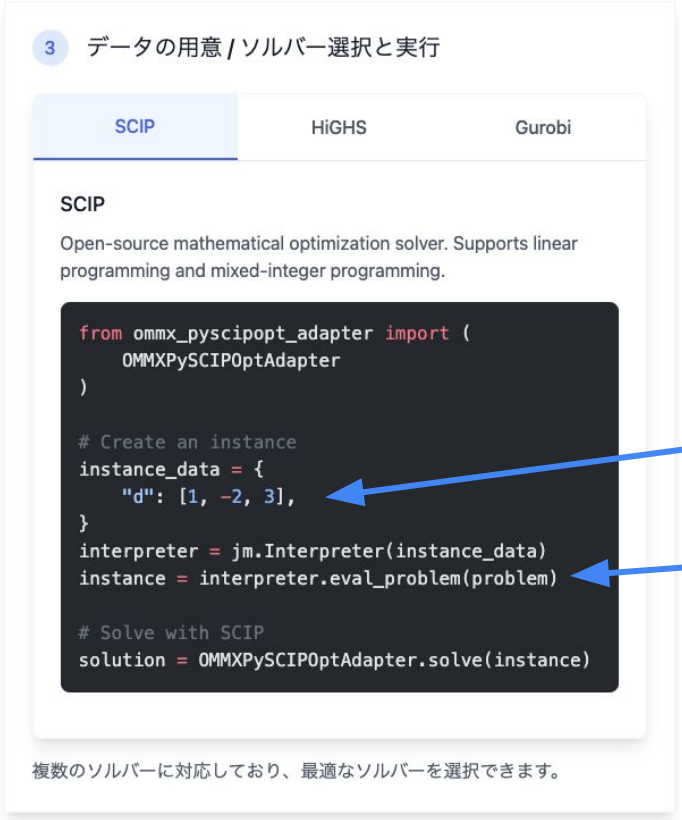
```python
import jijmodeling as jm

d = jm.Placeholder("d", ndim=1)
n = d.shape[0]
x = jm.BinaryVar("x", shape=(n,))
i = jm.Element("i", (0, n))

problem = jm.Problem("sample")
# Objective Function
problem += jm.sum(i, d[i]*x[i])
# Constraints
problem += jm.Constraint(
  "c1", x[:].sum() <= 1
)
```

JijModelingを使用して、数式と似た表現で最適化問題を記述できます。

代数的数理モデラー

特徴:
・代数構造のみを記述
→ 数理モデルとデータの**分離**

このd, nはまだ具体的に値が決まっていない



代数的数理モデラー

特徴:
・代数構造のみを記述
→ 数理モデルとデータの**分離**

ここでd (とn) を具体的に決めて

代入してインスタンスを生成

JijModeling: LaTeXへの変換機能

```
problem = jm.Problem("Jij-Knapsack")

i = jm.Element("i", (0, N))
problem += jm.Constraint("capacity", jm.sum(i, w[i]*x[i]) <= C)

problem += -1*jm.sum(i, v[i]*x[i])

problem
```

Problem: Jij-Knapsack

$$\min \quad -\sum_{i=0}^{N-1} v_i \cdot x_i$$

s.t.

$$\text{capacity} \quad \sum_{i=0}^{N-1} w_i \cdot x_i \leq C$$

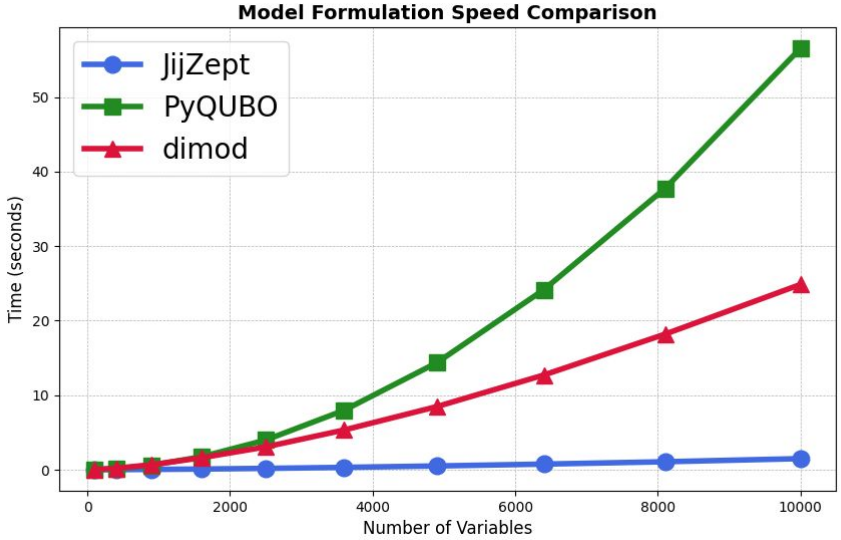
where

x	1-dim binary variable	x[i]: アイテムを入れる1入れない0

代数構造を記述することによるメリット:

Jupyterbook上で確認すると
どういう数理モデルになっているかを
確認しやすい。

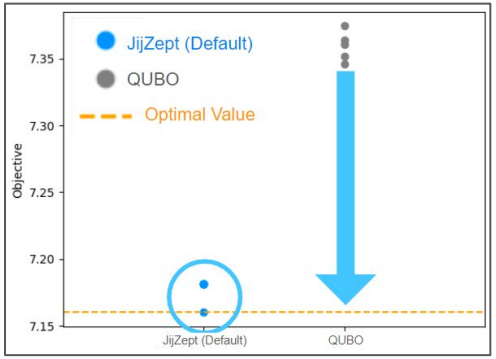
JijModeling: 書き手によらず速い



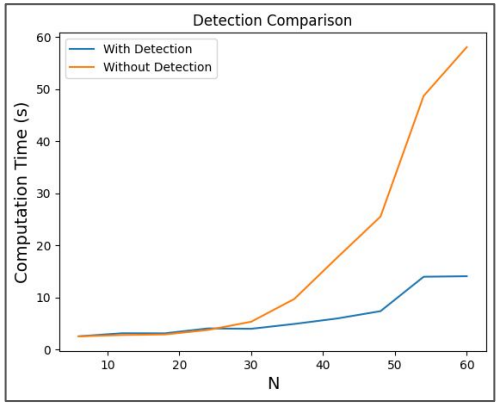
代数構造を記述することによるメリット:

代数構造へのデータの代入は
Rust側で行われ、書き方によらず速い！

JijModeling: ソルバーの性能向上



DA4でのTSPを解いた場合の比較



SOS1を含む問題でのSCIPの性能向上

代数構造を記述することによるメリット:

ソルバーが得意な数理構造を検知
→ 対応したソルバーの機能を活用
→ 専門家じゃなくてもソルバーの性能を引き出せる

人工知能学会 2025 @ 大阪
オーガナイズドセッション (数理最適化)
https://www.ai-gakkai.or.jp/jsai2025/os/#OS-17

JijModeling: 代数構造の静的チェック

Jij

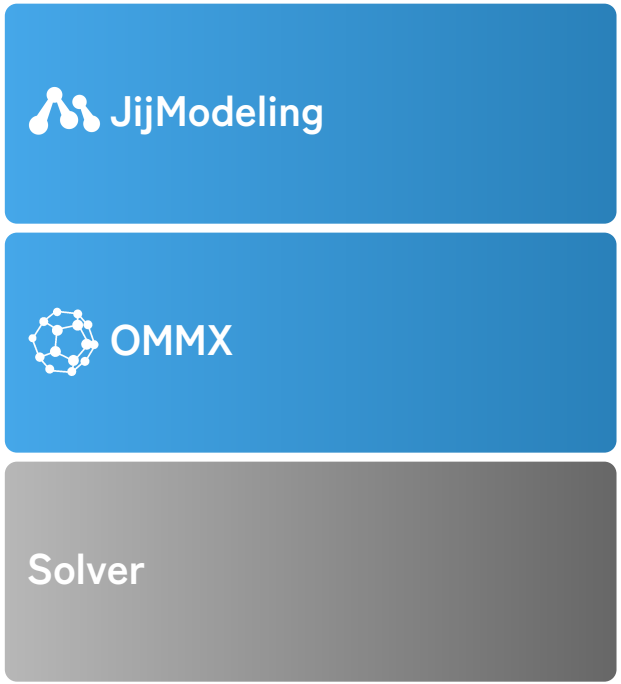
後半で！

OMMX

OMMX Adapter

pip install “jijzept-sdk[all]”

今すぐローカルで使えます！



代数的数理モデラー

数理最適化共通スキーマ (中間表現)

MIP Solver:
- SCIP
- CBC
- Gurobi

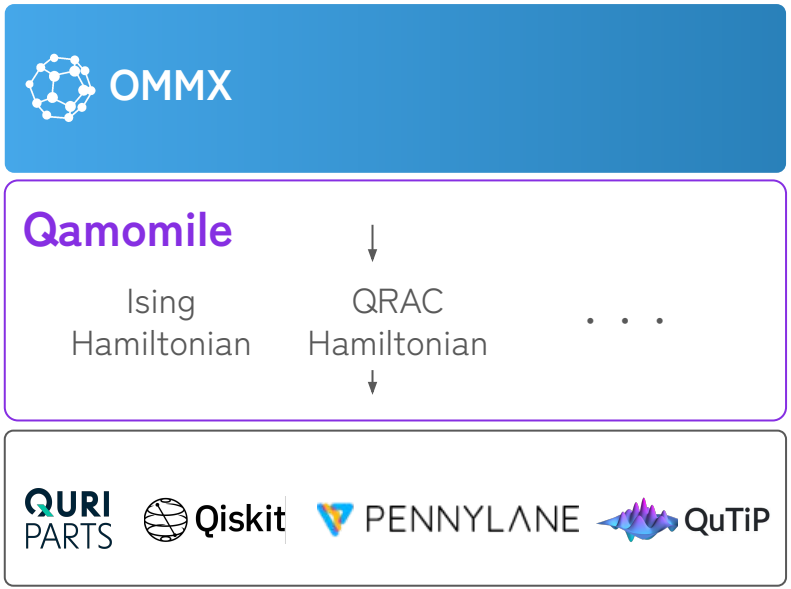
Ising:
- OpenJij
- Fixstars Amplify
- D-Wave
- Digital Annealer

Quantum:
- Qamomile

Qamomile

Qamomile

古典ビット ←→ 量子ビット間のエンコードを行うツール
QAOA, QRAOなどの複数の量子最適化アルゴリズムに対応



Qiskit with QAOA

```
converter = QAOAConverter(compiled_instance)
hamiltonian = converter.get_cost_hamiltonian()
```

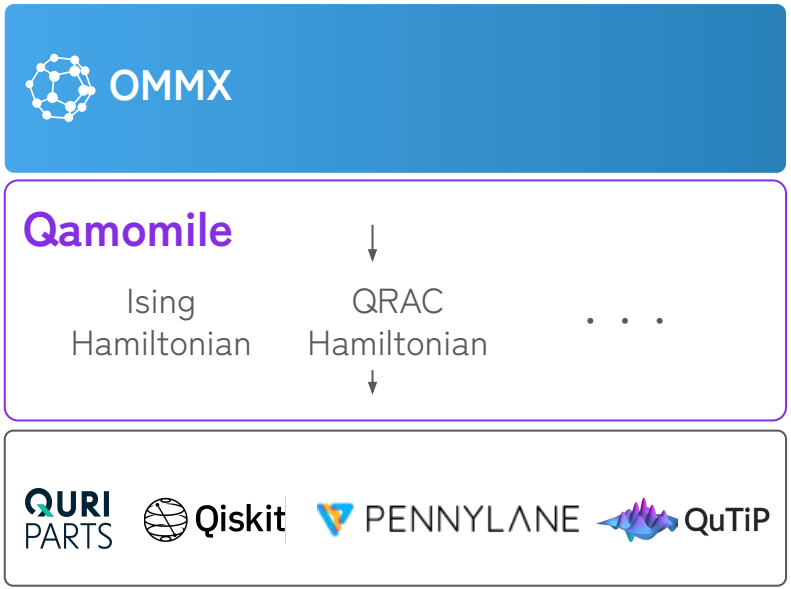
QRAO

```
converter = QRAC31Converter(compiled_instance)
hamiltonian = converter.get_cost_hamiltonian()
```

Qamomile

Qamomile

古典ビット ←→ 量子ビット間のエンコードを行うツール
QAOA, QRAOなどの複数の量子最適化アルゴリズムに対応



Qiskit with QAOA

```
converter = QAOAConverter(compiled_instance)
hamiltonian = converter.get_cost_hamiltonian()
```

QRAO

```
converter = QRAC31Converter(compiled_instance)
hamiltonian = converter.get_cost_hamiltonian()
```

Bloqade Analog

CUDA-Q

Comming Soon….

Jijでの量子アルゴリズムへの取り組み

Jijでの量子計算の方向性

古典最適手法と組み合わせた問題分割

より少ないコストで複雑で大規模な最適化問題を解く手法の開発

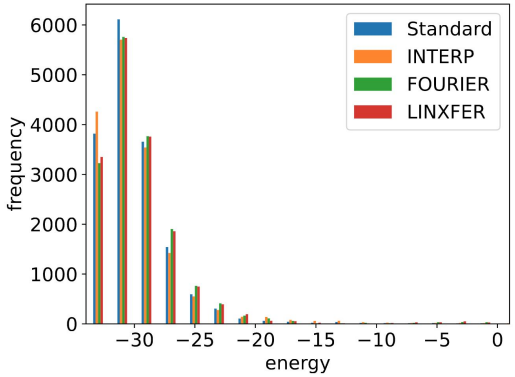
パラメータ転送/機械学習

エンコード手法の適用

Jijでの量子計算の方向性

パラメータ転送/機械学習

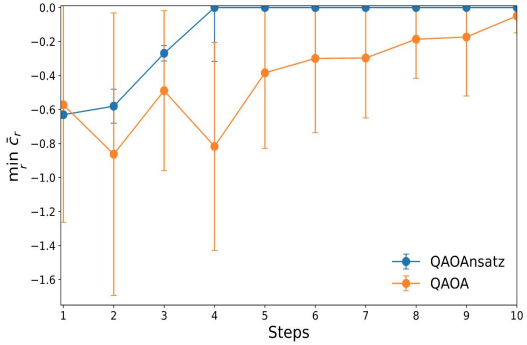
QAOAのパラメータを線形に制限した際のパラメータ転送性の研究



[R. Sakai et al., arXiv:2504.12632 (2025)]

問題分割手法

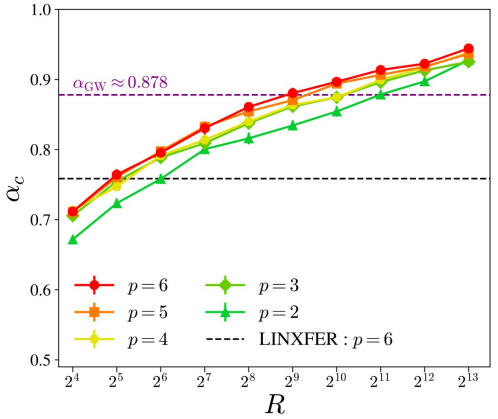
列生成法とQuantum Alternating Operator Ansatzを組み合わせて配送計画問題を解く手法の提案



[W-H Huang et al., arXiv:2503.17051 (2025)]

エンコード手法の適用

Quantum RelaxationとQSCIを組合せた非変分型量子アルゴリズムの提案

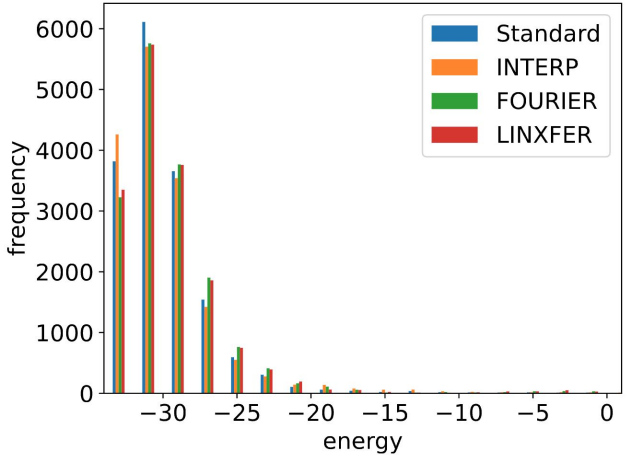


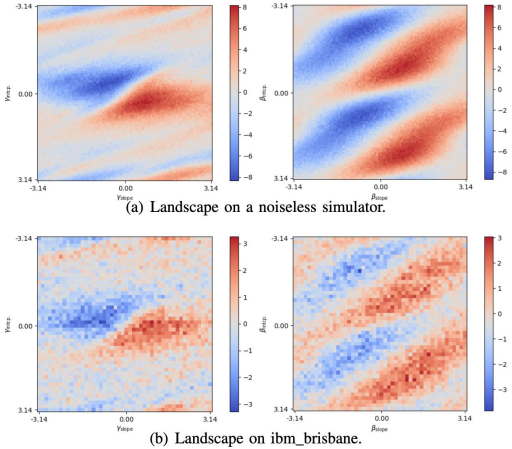
[H. Matsuyama et al., arXiv:2504.12629 (2025)]

Parameter transferability in QAOA

QAOAのパラメータを線形に制限した際のパラメータ転送性の研究

- パラメータ数を2pから4に削減
- 小規模問題で探索した変分パラメータがより大きな問題で利用可能
- ノイズ環境下でもランドスケープの構造は変化しない




(a) Landscape on a noiseless simulator.

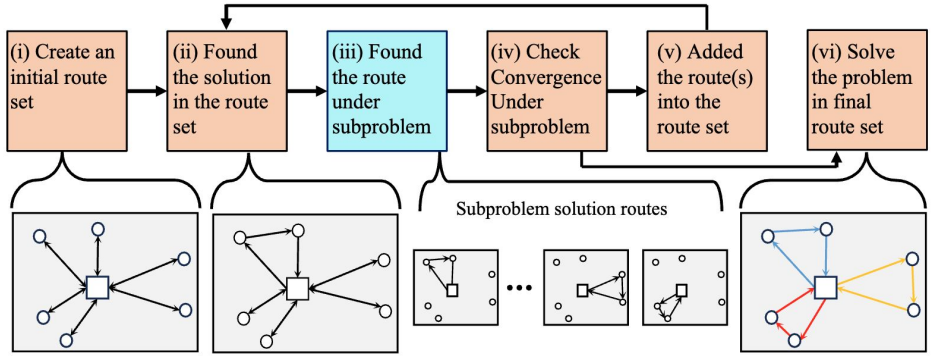
(b) Landscape on ibm_brisbane.

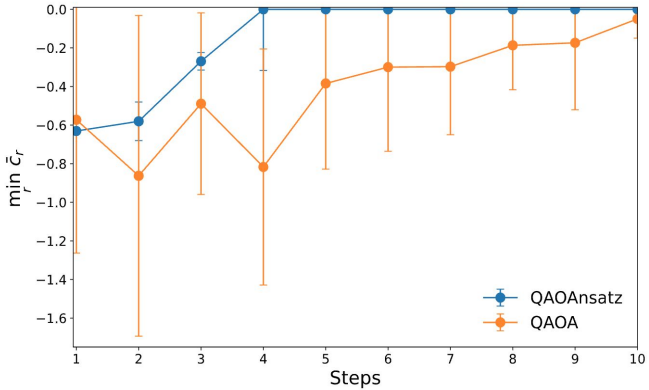
[R. Sakai et al., arXiv:2504.12632 (2025)]

Parameter transferability in QAOA

問題分割アプローチによって配送計画系の問題を解く手法の提案

- 古典最適化手法の列生成法とQuantum Alternating Operator Ansatzを組み合わせてVehicle Routing Problemを解く手法の提案
- 繰り返し操作によって解の質を向上させることができる
- 最適解までの距離を見積もることができる





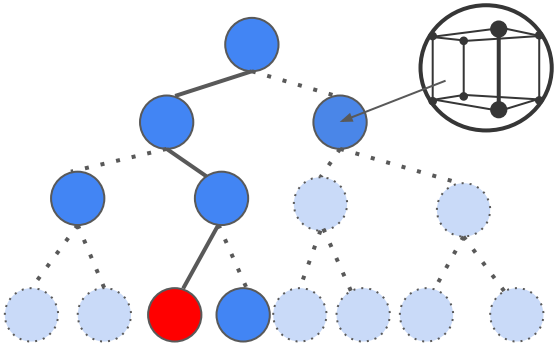
[W-H Huang et al., arXiv:2503.17051 (2025)]

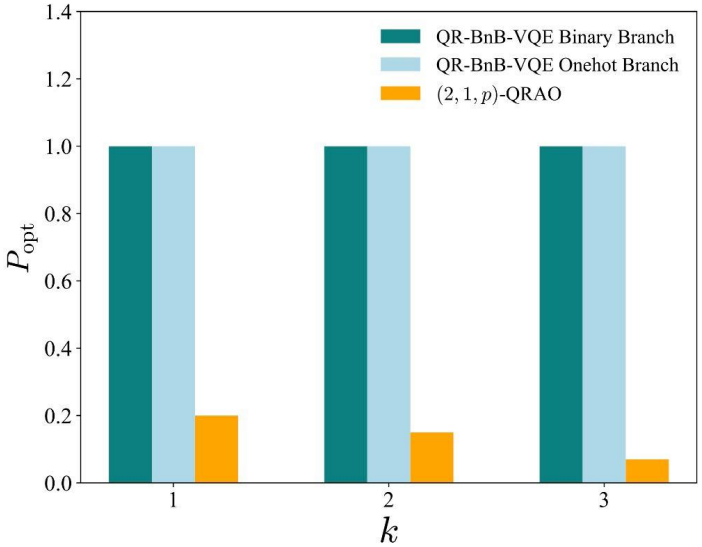
QRAO Branch and Bound

QRAOを利用した分枝限定法ベースのハイブリッドアルゴリズム

分枝限定法は緩和問題による下界の推定と実行可能解の探索による上界の更新によって，最適解の探索を行う厳密解法

- 量子緩和を利用して分枝限定法を行うことで，量子アルゴリズムを用いても最適解を探索できる(理論的には，最適性の保証もできる)

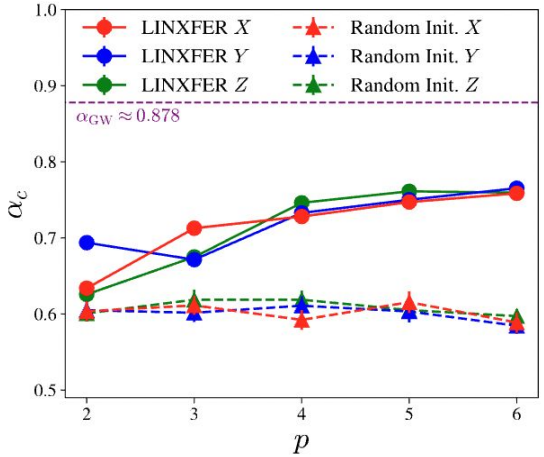


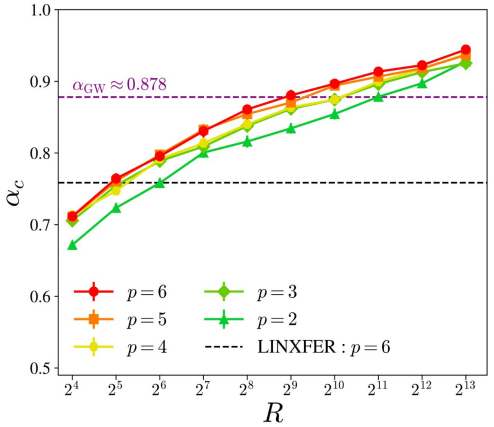


Sampling based Quantum Optimization

Quantum RelaxationとQSCIを組み合わせた非変分型量子アルゴリズムの提案

- 量子化学の分野で提案されたQSCIを最適化問題へと利用する研究.
- Quantum Relaxationを利用して問題を埋め込むことで，最大3倍の圧縮率が達成できる
- Quantum Relaxationに対するAlternating Operator Ansatz
  でもパラメータ転送が行えることを示した.
- パラメータ転送を利用することで大規模問題で変分計算が必要せず，質の高い解が得られる





生成AIを用いた
量子・最適化エンジニアリング

AI Driven Software Engineering

Jij

AI Driven Software Engineering

JijModeling

OMMX

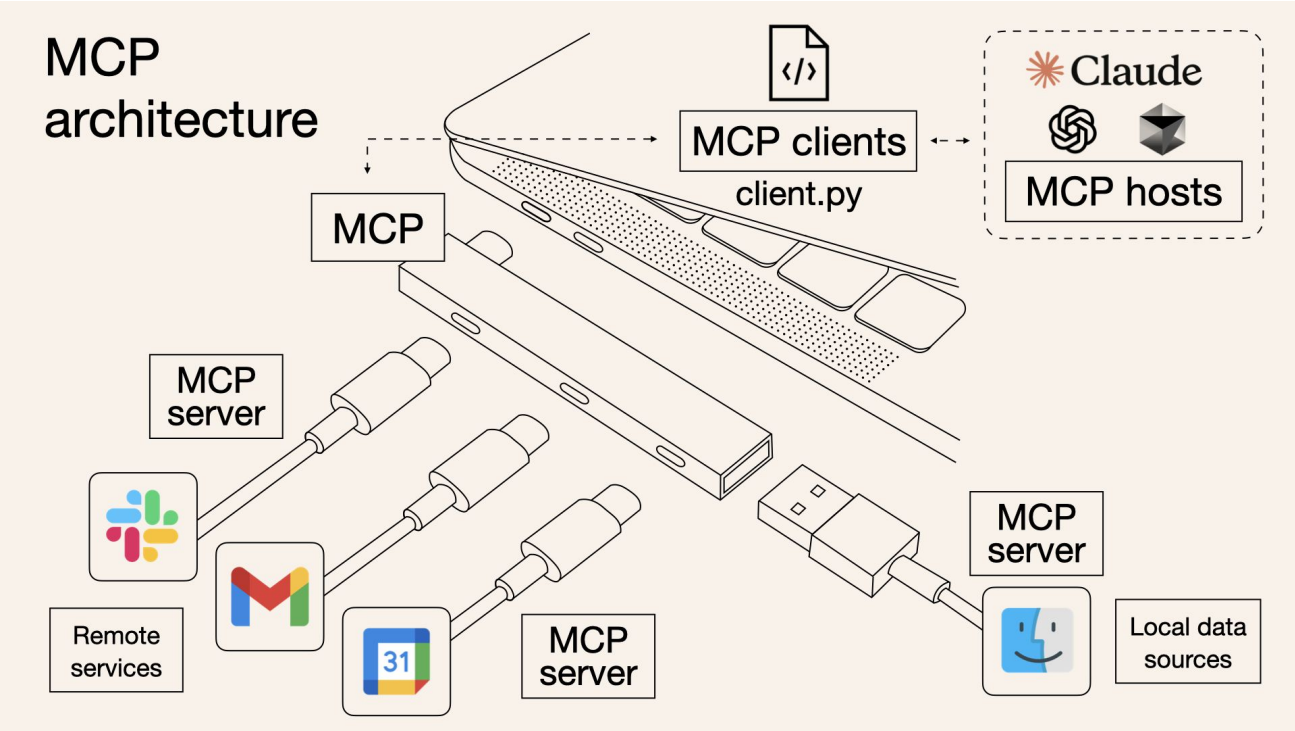
Solver

Qamomile

QURI PARTS Qiskit PENNYLANE QuTiP

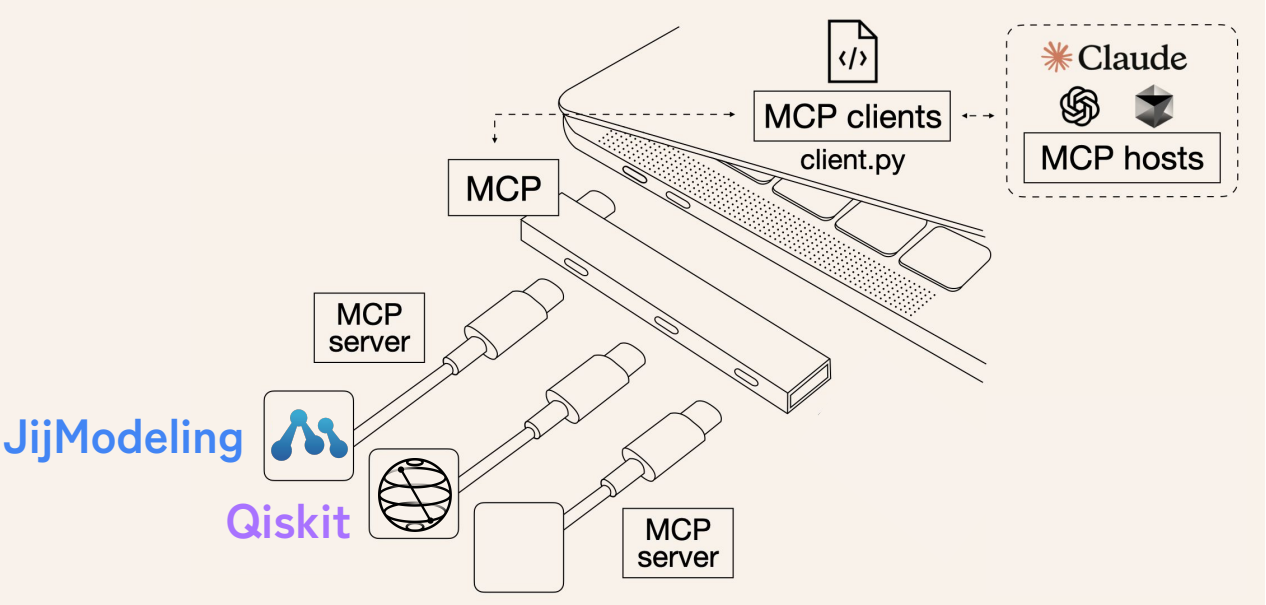
それぞれのライブラリが新しい
バージョン変更による修正も激しい

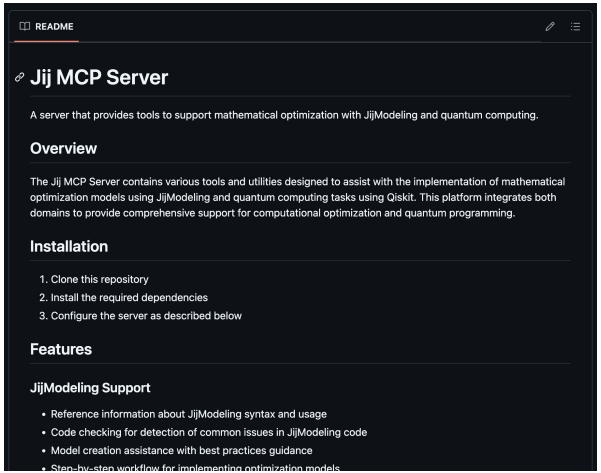
Model Context Protocol (MCP)



What is Model Context Protocol (MCP)? How it simplifies AI integrations compared to APIs

Jij-MCP-Server




README

Jij MCP Server

A server that provides tools to support mathematical optimization with JijModeling and quantum computing.

Overview

The Jij MCP Server contains various tools and utilities designed to assist with the implementation of mathematical optimization models using JijModeling and quantum computing tasks using Qiskit. This platform integrates both domains to provide comprehensive support for computational optimization and quantum programming.

Installation

1. Clone this repository
2. Install the required dependencies
3. Configure the server as described below

Features

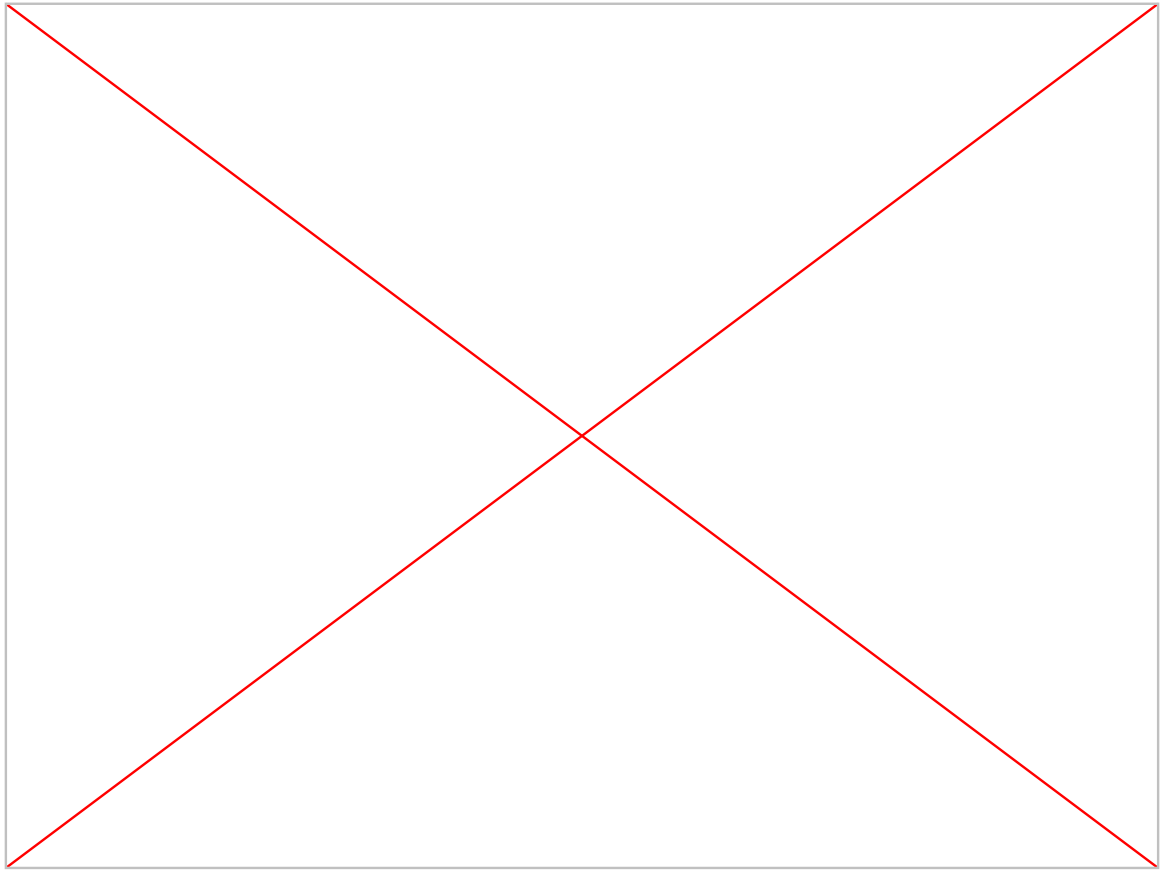
JijModeling Support

- Reference information about JijModeling syntax and usage
- Code checking for detection of common issues in JijModeling code
- Model creation assistance with best practices guidance
- Step-by-step workflow for implementing optimization models

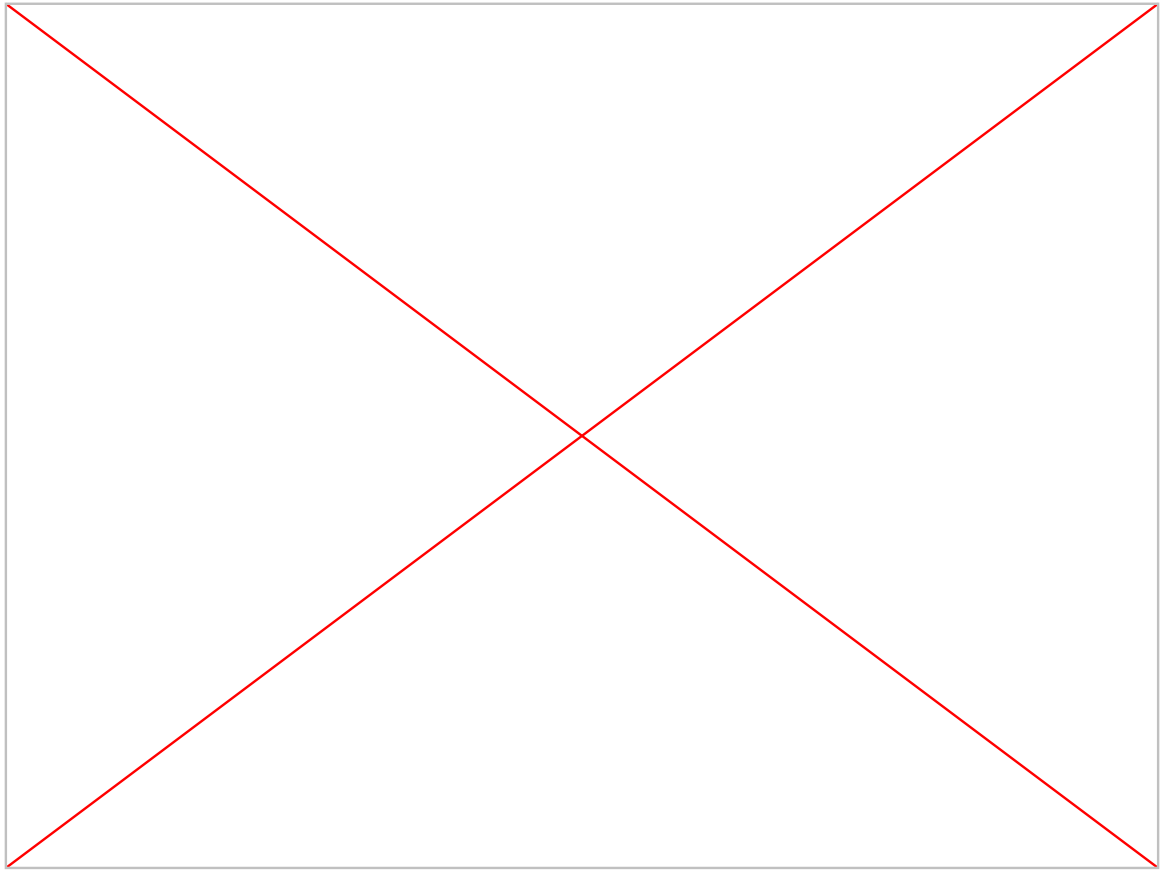
https://github.com/Jij-Inc/Jij-MCP-Server

Jij-MCP-Server: Mathematical Optimization

Jij-MCP-Server: Quantum Computing (Qiskit)

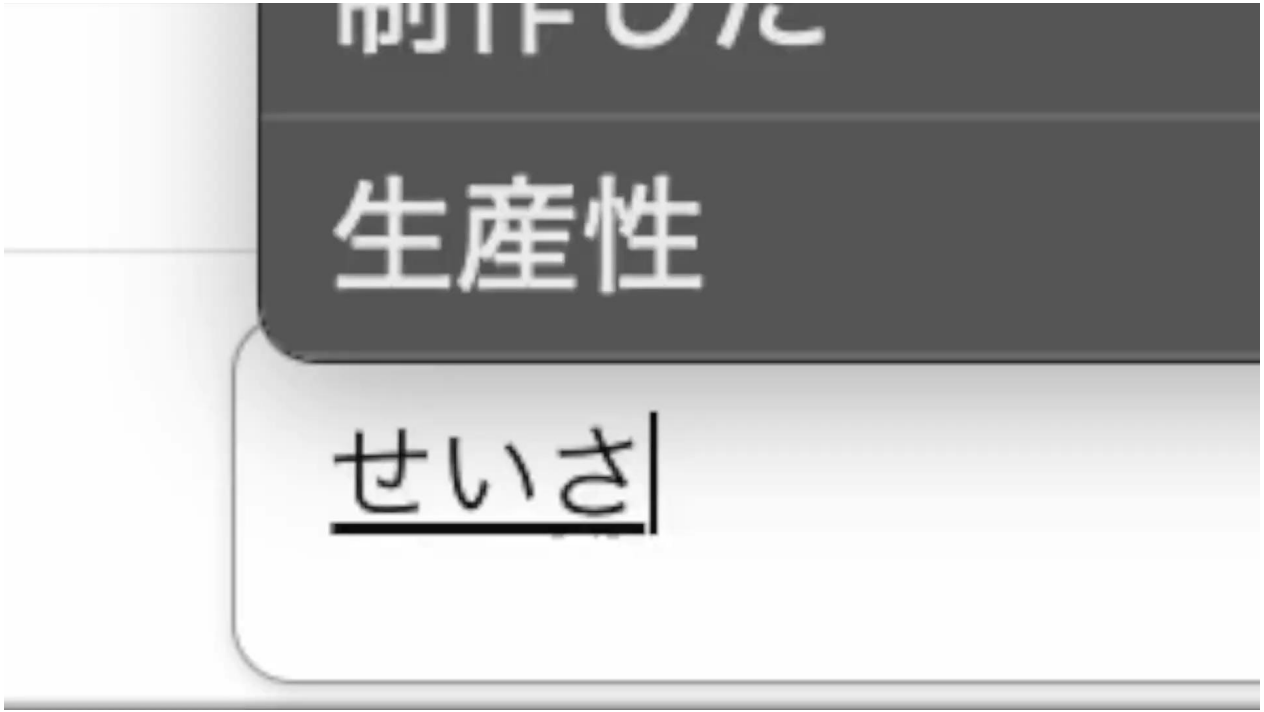


Jij-MCP-Server: Quantum Computing (Qiskit)



Jij-MCP-Serverの大まかな機能

JijZept AI α



Jij

info@j-ij.com

JijZept

Waiting List

Jij
Community