

## 量子アニーリングマシンなどで「組合せ最適化問題」を解くための QUBOを自動で構築するドメイン固有言語『PyQUBO』を開発 9月25日（火）よりオープンソースとして公開

株式会社リクルートコミュニケーションズ（本社所在地：東京都中央区、代表取締役社長：清水 淳、以下、リクルートコミュニケーションズ）は、アニーリングマシンの実活用推進に向け、QUBO※を自動で構築するドメイン固有言語（DSL）『PyQUBO（パイキューボ）』を開発しました。2018年9月25日（火）より、GitHub上にてオープンソースとして公開します。

※ QUBO : Quadratic Unconstrained Binary Optimization（二次制約なし二値最適化）

### 『PyQUBO（パイキューボ）』開発について

#### ●背景

デジタルマーケティングの領域をはじめ、配送の最適化や出店計画、AI（人工知能）開発に必要な機械学習処理など、様々なビジネスシーンにおいて「組合せ最適化問題」は多く存在します。しかし、従来のコンピュータでそれら問題を解くには莫大な工数を必要とし、実活用は難しいとされてきました。こうした現状を踏まえ、当社は2015年より量子アニーリング※<sup>1</sup>技術の研究を開始し、デジタルマーケティングの領域における「組合せ最適化問題」を高速に解くための実活用を目指しています。主には、アニーリング技術※<sup>2</sup>を用いたデータ分析手法の理論構築及びプログラムの実装と、実データに適用する考察・検討を進めています。

#### ●『PyQUBO（パイキューボ）』について

これまで当社で進めてきたアニーリング技術活用の考察・検討から得た知見や技術の一部を活用し、この度、量子アニーリングマシンなどのコンピュータで「組合せ最適化問題」を解く際に必要となる専門的な形式「QUBO（Quadratic Unconstrained Binary Optimization：二次制約なし二値最適化）」を自動で構築するドメイン固有言語（DSL※<sup>3</sup>）『PyQUBO』を開発しました。

アニーリングマシンで「組合せ最適化問題」を解く際、ユーザーは解きたい問題を数式として定式化し、さらにQUBO等のアニーリングマシン向けの専門的な形式に手作業で変換する必要があります。このQUBOの形式に変換する処理は複雑なため、ユーザーのミスを誘発し、不具合を招くケースがありました。今回、当社が開発したDSL『PyQUBO』を利用することで「組合せ最適化問題」が自動的にQUBOとして構築され、結果、ユーザーはよりシンプルかつ直観的にアニーリングマシンを利用することが可能になります。

#### <『PyQUBO』なし>

事前到手計算した結果を記述するので  
可読性が低くなる

```
def index(i, j):
    return i+n_city + j

A = 4.0
Q = defaultdict(float)
for i in range(n_city):
    for j in range(n_city):
        Q[index(i, j), index(i, j)] += 2*A

for i in range(n_city):
    for j in range(n_city):
        for k in range(n_city):
            if j != k:
                Q[index(i, j), index(i, k)] += 2.0 * A
                Q[index(j, i), index(k, i)] += 2.0 * A

for u in range(n_city):
    for v in range(n_city):
        for i in range(j, n_city):
            Q[index(u, i), index(v, (i+1)%n_city)] += d(u, v)
```

#### <『PyQUBO』を利用>

数式をそのままの形で記述するとQUBOが自動生成される

```
# Write TSP in DSL
x = Matrix('x', n_city, n_city)
distance = Sum(0, n_city, lambda u: Sum(0, n_city, lambda v:
    Sum(0, n_city, lambda j: d(u, v) * x[j, u] * x[(j+1)%n_city, v])))
const_1 = Sum(0, n_city, lambda v: (Sum(0, n_city, lambda j: x[j, v]) - 1)**2)
const_2 = Sum(0, n_city, lambda j: (Sum(0, n_city, lambda v: x[j, v]) - 1)**2)

# Construct Hamiltonian and compile it
A = Param("A")
H = distance + A * (const_1 + const_2)
model = H.compile()

# Generate QUBO
qubo, offset = model.to_qubo(params={'A': 4.0})
```

$$H = \sum_u \sum_v \sum_j^n d_{uv} \sum_j^n x_{j,u} x_{j+1,v} + A \sum_v^n \left( \sum_j^n x_{j,v} - 1 \right)^2 + A \sum_j^n \left( \sum_v^n x_{j,v} - 1 \right)^2$$

経路の距離                      制約1                      制約2

巡回セールスマン問題における『PyQUBO』の記述例

## オープンソース公開について

当社は、アニーリングマシンの実活用推進への貢献を目的とし、2018年9月25日より『PyQUBO』をGitHub上でオープンソースとして公開します。『PyQUBO』は統計・解析処理およびアニーリングマシン研究で一般的に使用されているプログラミング言語Pythonを使用しています。ユーザーは『PyQUBO』を使用することで、QUBOに関する専門的な知識や経験を持たなくてもアニーリングマシンを利用することが可能になります。今回のオープンソース公開を通じて、より多様な研究者・エンジニアがアニーリングマシンの実活用に向けた研究に参入しやすくなる土壌を作り、結果、応用事例創出の加速に寄与したいと考えています。また、アニーリングマシン利活用に必要な開発環境の整備と利便性向上が、『PyQUBO』によって築かれるエンジニアコミュニティによって推進されていくことも期待します。

▶PyQUBO GitHubサイト：<https://github.com/recruit-communications/pyqubo>

### <『Qubits North America 2018』での発表について>

『PyQUBO』開発の取り組み詳細について、9月25日（火）、米国テネシー州ノックスビルで開催されるD-Wave Users Conference 『Qubits North America 2018』にて発表します。

## 当社におけるアニーリングマシンの実活用に向けた研究について

リクルートコミュニケーションズは、リクルートグループを横断する機能会社として、主にWebマーケティングや広告制作を通じた集客・成約ソリューションをクライアント各社に提供しています。また、リクルートの商品・サービスとユーザーとの接点の拡大・拡充を目的とし、広告・宣伝、流通、カスタマーサポートまでを一貫して担当しています。

個々のサービス利用者の嗜好に合わせて最適な広告やコンテンツなどの情報を提供するデジタルマーケティングの領域には、“サービス利用者×記事内容×広告”のようなマッチング問題を扱うため、従来のコンピュータでは実用的な時間では解くことが難しいとされてきた「組合せ最適化問題」が多く存在します。当社は、2015年より、アニーリングマシンを活用して「組合せ最適化問題」を高速に解き、ビジネスへの実活用を目指す研究を複数実施して参りました。リクルートグループ各社及び、その顧客に提供するデジタルマーケティングにおいて、個人の嗜好など複数の属性情報の組み合わせから導き出される最適なサービスの提案や広告配信など、サービス品質を飛躍的に向上させる新たなマーケティング・テクノロジーの開発を目指します。

## 用語解説

### ※ 1 量子アニーリング

量子アニーリング(Quantum Annealing)は、「組合せ最適化問題」を解くための一つの手法であり、量子効果を反映させた物理系のダイナミクスをシミュレーションすることによって最適化問題を解くものです。量子アニーリングは特に多数の局所解を持つ非線形最適化問題に優位性があると言われており、自然言語処理やマーケティングテクノロジーにおけるクラスタリング処理など、多くの場合、最適化問題として定式化される機械学習に対しても効果的であると考えられています。

### ※ 2 アニーリング技術

最小エネルギー状態を保持した秩序ある構造の状態を作り出す「焼きなまし」をコンピュータ上で再現することで最適化問題について汎用的に解を得ることができるアルゴリズムです。

### ※ 3 ドメイン固有言語・DSL (domain-specific language)

JavaやPythonなどの汎用プログラミング言語とは違い、ある特定領域（ドメイン）の問題の解決に特化したプログラミング言語を指します。

【本件に関するお問い合わせ先】

<https://www.rco.recruit.co.jp/contact/>