

Wojciech BOŻEJKO, Sergii TROTSKYI,
Mariusz UCHROŃSKI, Mieczysław WODECKI
Politechnika Wrocławska

HYBRYDOWE ALGORYTMY KWANTOWE ROZWIĄZYWANIA PROBLEMÓW OPTYMALIZACJI DYSKRETNEJ

Streszczenie. W pracy przedstawiamy dwa hybrydowe algorytmy kwantowe, *Quantum Approximate Optimization Algorithm* (QAOA) oraz *Variational Quantum Eigensolver* (VQE), do rozwiązywania przepływowego problemu szeregowania zadań (*Flow Shop Scheduling Problem*, FSSP) z kryterium C_{\max} . Obliczenia są wykonywane z wykorzystaniem pakietu Qiskit.

HYBRID QUANTUM ALGORITHMS FOR SOLVING DISCRETE OPTIMIZATION PROBLEMS

Summary. In this paper, we present two hybrid quantum algorithms, *Quantum Approximate Optimization Algorithm* (QAOA) and *Variational Quantum Eigensolver* (VQE), for solving the flow shop scheduling problem (*Flow Shop Scheduling Problem*, FSSP) with the C_{\max} criterion. Calculations are performed using the Qiskit package.

1. Wprowadzenie

Pakiet Optymalizacji Qiskit [25] zaprojektowany przez IBM Research jest frameworkiem stworzonym do definiowania i wdrażania rozwiązań dla problemów optymalizacji. Ujednolicono API (ang. Application Programming Interface) umożliwia implementację i uruchomienie algorytmów kwantowych, w szczególności Quantum Approximate Optimization Algorithm (QAOA) oraz Variational Quantum Eigensolver (VQE). Dodatkowo, częścią pakietu są klasyczne algorytmy optymalizacji (opisane w [26]) stosowane w testowaniu i walidacji wyników algorytmów kwantowych. Poniżej przedstawiamy krótko oba algorytmy kwantowe.

1.1. Variational Quantum Eigensolver (VQE)

VQE jest hybrydowym algorytmem szacującym minimalną wartość energetyczną opisaną Hamiltonianem (wyrażającym energię układu kwantowego). Metoda została zaproponowana przez Peruzzo et al., ze szczególnym wskazaniem na zastosowanie w fonicznych procesorach kwantowym [12] (z kubitami reprezentowanymi jako stany kwantowe fotonów). Kandala i in. wykorzystali VQE zaimplementowane na kubitach nadprzewodnikowych [7] w optymalizacji w kontekście rozwiązywania problemu kwantowego magnetyzmu. Inne zastosowania można znaleźć w pracach [9, 11, 28]. Podstawowa koncepcja algorytmu to minimalizacja $\langle \psi(\vec{\theta}) | \hat{H} | \psi(\vec{\theta}) \rangle$, która obejmuje Hamil-

tonian \hat{H} poprzez znalezienie suboptymalnego $\vec{\theta}$ [9]. Algorytm iteracyjnie aktualizuje parametry $\vec{\theta}$ na komputerze klasycznym i dokonuje pomiaru obwodu na urządzeniu kwantowym (lub symulatorze). W praktyce, obecnie VQE może być wykonany na urządzeniach klasy NISQ (*Noisy Intermediate-Scale Quantum*, [13, 1]).

Kluczową częścią VQE jest Ansatz, który jest parametryzowanym obwodem kwantowym (obwodem kwantowym z programowalnie regulowanymi parametrami). W kolejnych rozdziałach omówimy różne przykłady Ansatzów zaimplementowanych w pakiecie Qiskit oraz odpowiadające im struktury, które możemy wykorzystać do optymalizacji.

1.2. Quantum Approximate Optimization Algorithm (QAOA)

Podobnie jak VQE, QAOA [4] jest algorytmem hybrydowym, który może być używany do rozwiązywania NP-trudnych problemów kombinatorycznych (jest swego rodzaju heurystyką). Algorytm został zaproponowany przez Farhi, Goldstone'a i Gutmanna jako metoda znajdowania rozwiązań przybliżonych za pomocą komputerów kwantowych [4]. Zasady adiabatycznych obliczeń kwantowych zostały opisane w [5] i są również związane z QAOA.

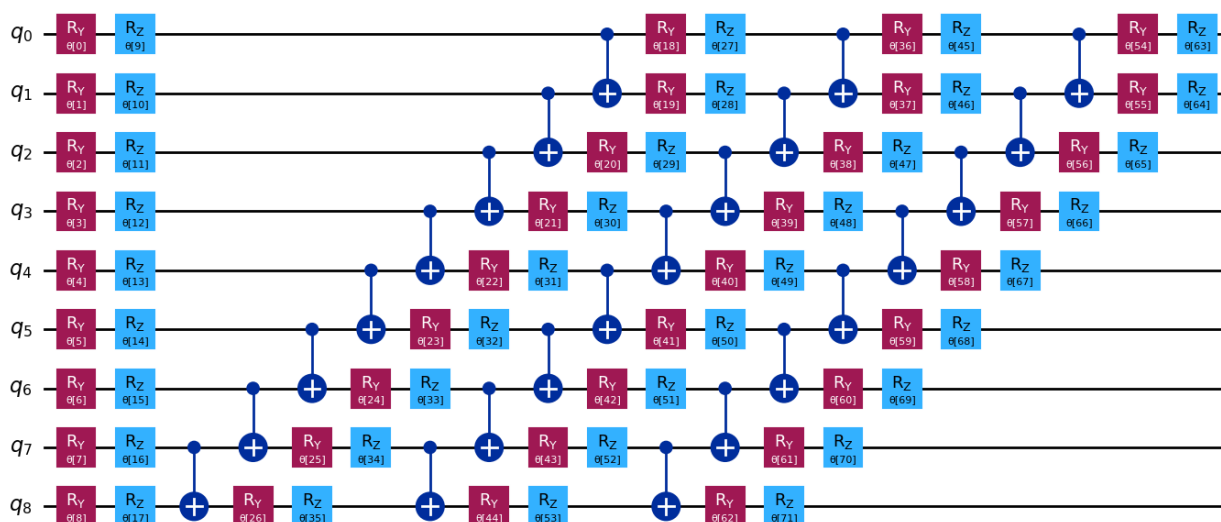
QAOA można sformułować jako metodę, która znajduje rozwiązanie bliskie optymalnemu dla zadanego Hamiltonianu \hat{H}_C , używając obwodu kwantowego z p powtórzeniami pod-obwodu $U_C(\gamma)$ oraz pod-obwodu $U_B(\beta)$ [8]. Można to matematycznie opisać jako:

$$|\gamma, \lambda\rangle = \left(\prod_{k=1}^p U_B(\beta_k) U_C(\gamma_k) \right) |+\rangle^{\otimes n}, \quad (1)$$

jest symetryczną superpozycją wszystkich 2^n obliczeniowych stanów bazowych (n jest liczbą kubitów), a $\gamma = \{\gamma_1, \gamma_2, \dots, \gamma_p\}$ i $\beta = \{\beta_1, \beta_2, \dots, \beta_p\}$ są parametrami wariacyjnymi, które są aktualizowane w każdej iteracji, $U_B(\beta) = e^{-i\beta B}$, $U_C(\gamma) = e^{-i\gamma \hat{H}_C}$, $B = \sum_j X_j$, a X_j opisuje operator X Pauliego działający na kubicie j , $1 \leq j \leq n$. Algorytm QAOA wymaga specyficznego Ansatzu opartego na Hamiltonianie rozważanego problemu.

1.3. Problem przepływowy

Problem przepływowy wykonywania zadań należy do najtrudniejszych (NP-trudnych) i najczęściej badanych problemów optymalizacji dyskretnej. W problemie tym dany jest zbiór maszyn i zbiór zadań. Każde zadanie należy wykonać na każdej maszynie w ustalonym czasie. Kolejność maszyn jest identyczna dla każdego zadania. Zakładamy ponadto, że zadania mogą być wykonywane na każdej maszynie w dowolnej kolejności (rozważamy tzw. niepermutacyjny problem przepływowy). Należy wyznaczyć kolejności wykonywania zadań na maszynach minimalizujące całkowity czas ich wykonania, tj. kryterium C_{\max} . Znacznie częściej w literaturze jest rozważany permutacyjny problem przepływowy, w którym kolejność wykonywania zadań na każdej maszynie jest taka sama. Opublikowano wiele różnych algorytmów rozwiązywania tego problemu (m.in. [6, 10]). Najbardziej efektywne są metaheurystyki typu popraw, oparte na różnych metodach przeszukiwania otoczeń. Praca stanowi kontynuację badań nad konstrukcją kwantowych algorytmów optymalizacji dla problemów NP-trudnych ([2, 3]).



Rys. 1. Ansatz EfficientSU2 dla 9 kubitów z 72 parametrami.

2. Struktura Ansztów VQE oraz QAOA

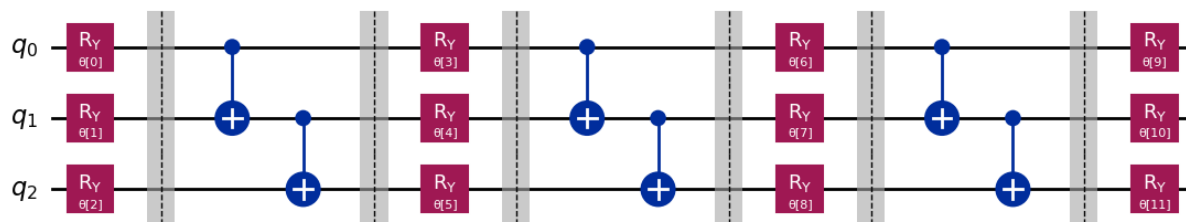
Uruchamianie algorytmów QAOA i VQE w pakiecie Qiskit jest podobne [27, 28]. Jednak w przeciwieństwie do VQE, który może być skonfigurowany za pomocą Ansztów TwoLocal, NLocal lub EfficientSU2, QAOA wymaga Ansatzu o specyficznej strukturze, opierającej się na Hamiltonianie, który jest optymalizowany (Hamiltonian powinien być przekazany jako parametr podczas tworzenia Ansatzu [24, 22]). Dodatkowo, bariery [14] służą do logicznego podziału obwodu kwantowego w taki sposób, by wszelkie optymalizacje były wykonywane tylko w granicach tych barier. Na rysunkach 2-3 przedstawiono przykładowe Ansztze wykonane przy użyciu narzędzi z pakietu Qiskit.

2.1. Ansatz EfficientSU2

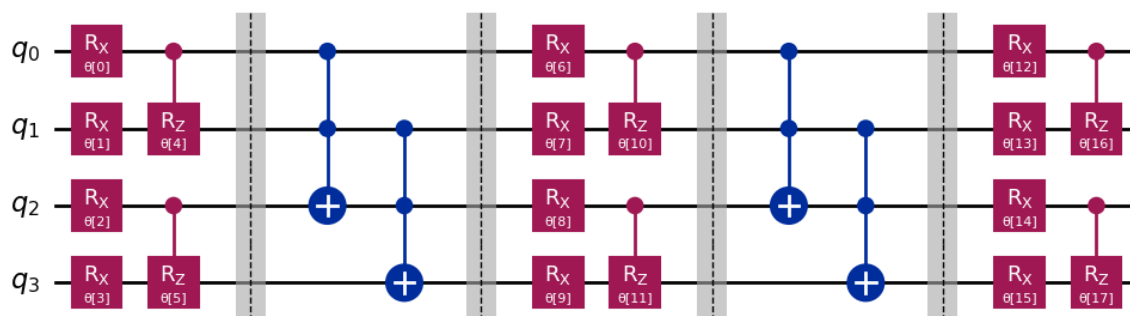
Struktura obwodu kwantowego EfficientSU2 [16] ma serię warstw, z których każda zawiera 1-kubitową rotacyjną bramkę kwantową: Pauli Y, Pauli Z, oraz wykorzystujące splątanie z innymi kubitami za pomocą bramki CX. Na rysunku 1 przedstawiono przykładowy Ansatz wykonany przy użyciu narzędzi z pakietu Qiskit dla 9 kubitów z 72 parametrami.

2.2. Ansztze TwoLocal oraz NLocal.

Ansatz 2-local [23] jest obwodem kwantowym (z parametrami, które aktualizujemy za pomocą klasycznej maszyny), charakteryzującym się kombinacją 1-kubitowych rotacyjnych bramek kwantowych oraz bramek kwantowych wykorzystujących splątanie 2 kubitów. Ansatz n-local [21] składa się również z warstw bramek kwantowych rotacyjnych i splątania, jednak n-local pozwala na większą elastyczność pod względem sposobu splątania kubitów (ta warstwa opiera się na niestandardowym schemacie splątania, który możemy zdefiniować programowo). Kombinacja warstw może być powtarzana wiele razy, ustalony przez parametr powtórzenia.



Rys. 2. Ansatz 2-local z barierami i liczbą powtórzeń równą 3.



Rys. 3. Ansatz n-local z barierami dla 4 kubitów, liczba powtórzeń równa 2, niestandardowy schemat splątania [(0, 1, 2), (1, 2, 3)].

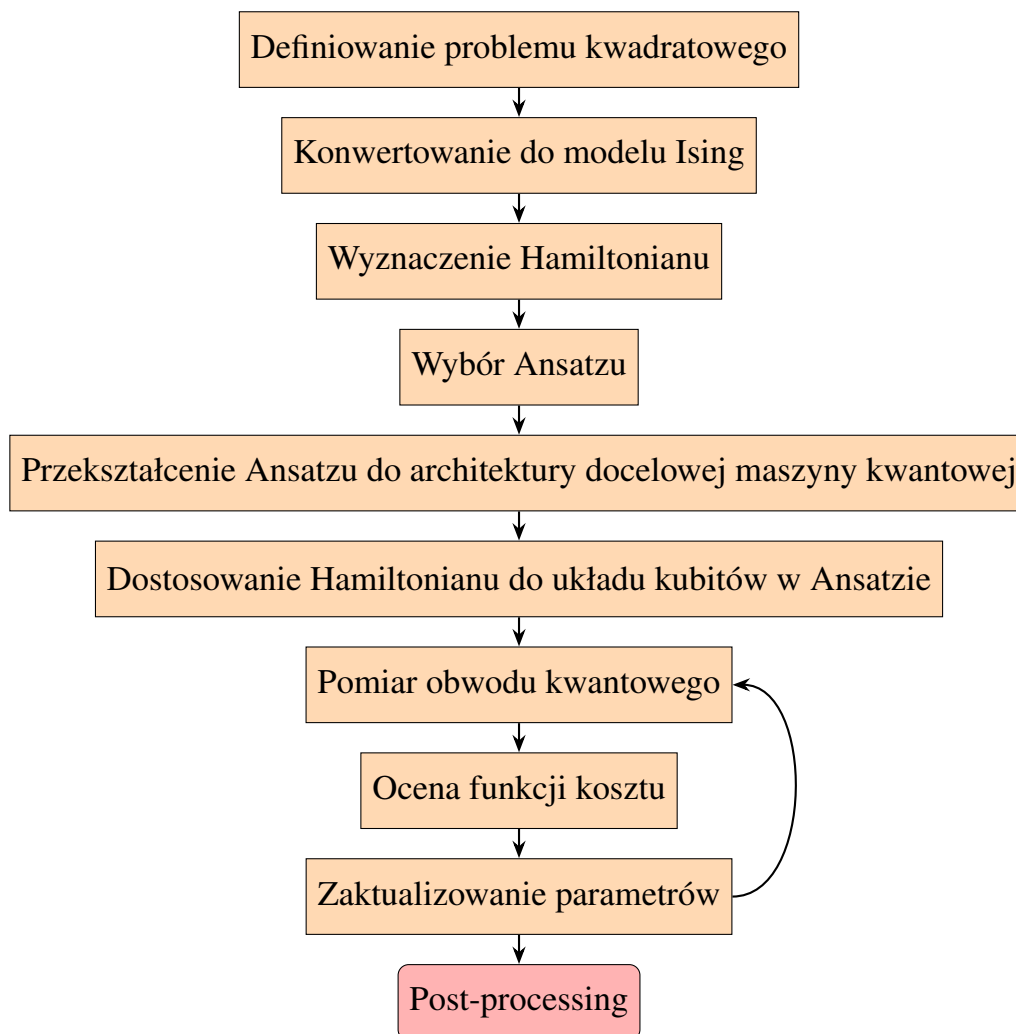
3. Struktura algorytmów

Proces algorytmiczny rozwiązywania zadanego problemu rozpoczyna się od zdefiniowania jego binarnej wersji kwadratowej (quadratic unconstrained binary programming, QUBO), który następnie jest konwertowany do modelu Isinga (ze zmiennymi ± 1 , [27, 28]). Model ten jest następnie mapowany na Hamiltonian, który reprezentuje całkowitą energię układu kwantowego. Następnie konstruowany jest odpowiedni Ansatz [24]. Zarówno Ansatz, jak i Hamiltonian są transformowane tak, aby były kompatybilne z docelowym sprzętem kwantowym poprzez transpiling (adaptacja obwodów kwantowych do architektury docelowej maszyny kwantowej). Ostatecznie uruchamiany jest hybrydowy algorytm VQE lub QAOA; iteracyjnie mierzony jest obwód kwantowy na komputerze kwantowym lub symulatorze, oraz aktualizowane są parametry (rys.4).

4. Studium przypadku

Rozważmy następujący przykład niepermutacyjnego problemu przepływowego. Czasy wykonywania zadań zostały zaczerpnięte z instancji Taillarda [17, 18], dla pierwszych 2 zadań na pierwszych 2 maszynach z przykładu ta001 rozmiaru 20 zadań i 5 maszyn.

Dla rozważanego problemu definiujemy Model Kwadratowy (Quadratic Model, QM), w celu wizualizacji obwodów kwantowych i opisu Hamiltonianu. Model QM został zdefiniowany tak, by minimalizować całkowity czas zakończenia wykonywania zadań C_{\max} . W celu umożliwienia uruchomienia QM na komputerze kwantowym lub w symulatorze Qiskit, wykorzystano implementację konwertera do modelu Isinga [15]



Rys. 4. Schemat algorytmiczny procedury optymalizacji hybrydowej na podstawie analizy dokumentacji Qiskit

Tabela 1
Dane wejściowe z przykładu ta001 instancji Taillarda dla rozmiaru 20 zadań i 5 maszyn

<i>jobs</i>	<i>machines</i>				
	1	2	3	4	5
1	54	79	16	66	58
2	83	3	89	58	56
3	15	11	49	31	20
4	71	99	15	68	85
5	77	56	89	78	53
⋮	⋮	⋮	⋮	⋮	⋮

pakietu Qiskit [19] oraz konwertera QuadraticProgramToQubo [20] by przetransformować QM na QUBO. W efekcie uzyskany został Hamiltonian, który wymagał maszyny

Tabela 2

Liczba bramek kwantowych oraz parametrów dla różnych Ansatzów i dla różnych rozmiarów modelu wejściowego

Rozmiar Modelu	Zadania	Maszyny	Kubity	TwoLocal		EfficientSU2		
				RY	CX	RY	RZ	CX
2×2	2	2	70	210	138	280	280	207
3×3	3	3	180	540	358	720	720	537
4×4	4	4	400	1200	798	1600	1600	1197
5×5	5	5	741	2223	1480	2964	2964	2220
6×5	6	5	1022	3066	2042	4088	4088	3063
7×5	7	5	1474	4422	2946	5896	5896	4419

EfficientSU2 liczba bramek RZ jest równa liczbie bramek RY. Całkowita liczba parametrów do optymalizacji stanowi sumę bramek RY i RZ. Liczba kubitów potrzebnych do znalezienia rozwiązania zależy nie tylko od rozmiaru modelu, ale także od implementacji ograniczeń modelu kwadratowego.

5. Podsumowanie

W artykule omówiono zastosowanie algorytmów optymalizacji kwantowej w rozwiązywaniu NP-trudnego problemu optymalizacji dyskrenej. Skupiono się na dwóch specyficznych hybrydowych algorytmach kwantowych: QAOA i VQE. Dokonano przeglądu i wizualizacji różnych struktur Ansatzów dla VQE, w tym TwoLocal, NLocal i EfficientSU2. Opisano kolejne kroki które należy wykonać aby uruchomić algorytmy VQE i QAOA na platformie kwantowej. Do artykułu dołączono również studium przypadku opisujące proces uzyskiwania Hamiltonianu dla modelu kwadratowego i tworzenia Ansatzu. Omówiono proces transpile'owania Ansatzu w celu zapewnienia zgodności z architekturą docelowego komputera kwantowego. W kolejnej fazie badań chcielibyśmy skupić się na różnych strukturach Ansatzów pod kątem ich wydajności.

LITERATURA

1. Bharti, K., Cervera-Lierta, A., Kyaw, T. H., Haug, T., Alperin-Lea, S., Anand, A., Degroote, M., Heimonen, H., Kottmann, J. S., Menke, T., Mok, W.-K., Sim, S., Kwek, L.-C., Aspuru-Guzik, A.: Noisy intermediate-scale quantum (NISQ) algorithms. *Rev. Mod. Phys.*, 2022, vol. 94, p. 015004.
2. Bożejko W., Burduk A., Pempera J., Uchroński M., Wodecki M.: Optimal solving of a binary knapsack problem on a D-Wave quantum machine and its implementation in production systems. *Annals of Operations Research*, 2024, in press.
3. Bożejko W., Pempera J., Uchroński M., Wodecki M.: Quantum annealing-driven branch and bound for the single machine total weighted number of tardy jobs scheduling problem. *Future Generation Computer Systems*, 2024, vol. 155, p. 245–255.
4. Farhi, E., Goldstone, J., Gutmann, S.: A Quantum Approximate Optimization Algorithm., arXiv:1411.4028 [quant-ph], 2014.
5. Farhi, E., Goldstone, J., Gutmann, S., & Sipser, M.: Quantum Computation by Adiabatic Evolution., arXiv:quant-ph/0001106, 2000,
6. Grabowski J., Wodecki M.: A very fast tabu search algorithm for the permutation flow shop problem with makespan criterion, *Computers and Operations Research*, 2004, vol. 31, p. 1891–1909.
7. Kandala, A., Mezzacapo, A., Temme, K., Takita, M., Brink, M., Chow, J. M., Gambetta, J. M.: Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* 2017, vol. 549, p. 242–246.
8. Mandl, A., Barzen, J., Bechtold, M., Leymann, F., Wild, K.: Amplitude amplification-inspired QAOA: improving the success probability for solving 3SAT. *Quantum Science and Technology*, 2024, vol. 9(1), p. 015028.
9. McClean, J. R., Romero, J., Babbush, R., Aspuru-Guzik, A.: The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics*, 2016, vol. 18, p. 023023.
10. Nowicki E., Smutnicki C.: A Fast tabu search algorithm for permutation flow shop problem, *European Journal of Operational Research*, 1996, vol.91, p. 160–175.
11. O’Malley, P. J. J., Babbush, R., Kivlichan, I. D., Romero, J., McClean, J. R., Barends, R., Kelly, J., Roushan, P., Tranter, A., Ding, N., Campbell, B., Chen, Y., Chen, Z., Chiaro, B., Dunsworth, A., Fowler, A. G., Jeffrey, E., Lucero, E., Megrant, A., ... Martinis, J. M.: Scalable Quantum Simulation of Molecular Energies. *Phys. Rev. X*, 2016, vol. 6(3), p. 031007.
12. Peruzzo, A., McClean, J., Shadbolt, P., Yung, M.-H., Zhou, X.-Q., Love, P. J., Aspuru-Guzik, A., O’Brien, J. L.: A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 2014, vol. 5, p. 4213.
13. Preskill, J.: Quantum Computing in the NISQ era and beyond. *Quantum*, 2018, vol. 2, p. 79.

14. **Barrier Implementation.** Available at: <https://docs.quantum.ibm.com/api/qiskit/qiskit.circuit.library.Barrier>
15. **Definition of function that convert to ising model.** Available at: https://qiskit-community.github.io/qiskit-optimization/stubs/qiskit_optimization.translators.to_ising.html.
16. **EfficientSU2 Ansatz Implementation.** Available at: <https://docs.quantum.ibm.com/api/qiskit/qiskit.circuit.library.EfficientSU2>
17. **E. Taillard, "Benchmarks for basic scheduling problems", EJOR 64(2):278-285, 1993.** Technical Report available at <http://mistic.heig-vd.ch/taillard/articles.dir/Taillard1993EJOR.pdf>
18. **E. Taillard Dataset for 20 jobs and 5 machines.** http://mistic.heig-vd.ch/taillard/problemes.dir/ordonnancement.dir/flowshop.dir/tai20_5.txt
19. **Implementation of function (translator) that convert Quadratic Model to Ising model.** Available at: https://github.com/qiskit-community/qiskit-optimization/blob/main/qiskit_optimization/translators/ising.py.
20. **Implementation of QuadraticProgramToQubo converter.** Available at: <https://docs.quantum.ibm.com/api/qiskit/0.24/qiskit.optimization.converters.QuadraticProgramToQubo>
21. **NLocal Ansatz Implementation.** Available at: <https://docs.quantum.ibm.com/api/qiskit/qiskit.circuit.library.NLocal>
22. **The Quantum Approximate Optimization Algorithm.** Available at: <https://docs.quantum.ibm.com/api/qiskit/0.25/qiskit.algorithms.QAOA>
23. **TwoLocal Ansatz Implementation.** Available at: <https://docs.quantum.ibm.com/api/qiskit/qiskit.circuit.library.TwoLocal>
24. **Qiskit Ansatz Implementations.** Available at: https://github.com/Qiskit/qiskit/tree/stable/1.1/qiskit/circuit/library/n_local
25. **Qiskit Optimization overview.** Available at: <https://qiskit-community.github.io/qiskit-optimization/index.html>.
26. **Qiskit Optimizers.** Available at: https://qiskit-community.github.io/qiskit-algorithms/apidocs/qiskit_algorithms.optimizers.html.
27. **Qiskit QAOA Tutorial.** Available at: <https://learning.quantum.ibm.com/tutorial/quantum-approximate-optimization-algorithm>
28. **Qiskit VQE Tutorial.** Available at: <https://learning.quantum.ibm.com/tutorial/variational-quantum-eigsolver>