

量子ゲート方式によるジョブショップスケジューリング問題のアルゴリズムの提案

Quantum Algorithm for Job-shop Scheduling Problem Based on Grover's Search Algorithm

量子ゲート方式の量子コンピュータでは、重ね合わせを利用した並列計算を実行できるため、膨大な解空間での最適解の探索が必要となるスケジューリング問題への応用が期待される。本稿ではジョブショップスケジューリング問題への量子コンピューティングの応用を検討し、グローバーのアルゴリズムを基にした量子アルゴリズムを提案する。提案手法では、解の探索を行う過程で実行不可能なスケジュールが解となる確率を低くすることができる。小規模の問題によるシミュレーションで、実際にその確率を低くできることを確認した。

飯塚 新司 Iizuka Shinji

1. はじめに

量子ゲート方式による量子コンピュータは、2016年にIBMがクラウドからアクセスできる5量子ビットの超伝導方式のデバイスを公開して以来、年々扱うことのできる量子ビット数が増え続け、2025年には4,000量子ビット超のデバイスの公開が予定されている¹⁾。しかし、現状のデバイスは量子ビット数がまだ少なく、ノイズによる量子誤りが多いNISQ (Noisy Intermediate-Scale Quantum) デバイスである。ビジネス適用に向けて実証が進んでいるが、実用的な応用で従来のコンピュータに対して優位性があるような利用方法は現時点で存在しない。だが、量子ビット数の多い量子誤り耐性をもつデバイスが実現されると、状況は大きく変化すると考えられている²⁾。現在、(株)日立ソリューションズ東日本ではSynPLAなど生産計画分野に強みを持つソリューションを展開しているが、上記のような将来においても引き続き優位性を保てるよう、生産計画で用いられるスケジューリング問題のアルゴリズムに量子コンピューティングを応用する研究を行った。

2. スケジューリング問題への量子コンピューティングの応用とその課題

現在、スケジューリング問題では量子アニーリング方式の適用が進んでいる³⁾。この方式ではスケジューリング問題を二次制約なしパイナリ最適化 (QUBO) と呼ば

れるコスト最小化問題に帰着して解く。生産計画では、ジョブショップスケジューリング問題 (以下、JSP) のように、工程の順序制約やマシンの同時処理禁止制約など、遵守しなければならない制約がある。コスト最小化問題に帰着する際、上記の制約はソフト制約としてコスト関数のペナルティ項で表現される。量子コンピューティングでは解は確率的なサンプリングにより取得されるが、得られた解が最適解でない場合、一部の制約を満たさない実行不可能なスケジュールとなる可能性がある。ペナルティ項の重みを大きくすると全体の作業時間のような本来最小化したいコストが大きくなるトレードオフがあるため、実行不可能なスケジュールが解となる確率を低くすることができず、制約の充足に課題がある。

量子ゲート方式では、量子演算を行う量子ゲートを組み合わせ、量子回路をプログラミングすることでアルゴリズムを実装し計算を行う。量子重ね合わせを利用した並列計算が可能であるため、探索問題などを従来のコンピュータよりも高速に計算することができる。スケジューリング問題への量子ゲート方式の応用については、VQE, QAOA, F-VQEなどの量子変分アルゴリズムを適用した先行研究がある⁴⁾。この研究もスケジューリング問題をコスト最小化問題に帰着する方式である。IBMのデバイスによる23量子ビットを使った検証で、コストを最小化する最適解の測定確率をF-VQEにより約25%まで高めた実験結果が報告されている。しかし、この方式

も実行不可能なスケジュールが解となる確率を低くすることはできず、問題規模が大きいくほど最適解が得られる確率は小さくなる。そのため、現実的な時間では実行可能なスケジュールが得られない可能性がある。このようにスケジューリング問題をコスト最小化問題に帰着して解く方式では、制約の充足が課題となっている。

3. 提案手法

本研究では、スケジューリング問題としてJSPを取り扱う。2章で述べた課題を解決するため、提案手法ではdisjunctive graph⁵⁾と呼ばれるグラフ構造でJSPを表現し、disjunctive arcの最適な向きの組合せを探索する。組合せの探索はグローバラーのアルゴリズム⁶⁾を基に行い、その過程で実行不可能なスケジュールが解となる確率を低くする。量子ゲートの操作回数が多く、現行のNISQデバイスでは実行できないが、2章で述べた制約の充足の課題を解決できる手法である。

disjunctive graphとは、ジョブのオペレーションをノードとし、オペレーション間の順序制約をエッジで表現したグラフ構造である。2ジョブ×2マシンのJSPの場合のdisjunctive graphを図1に示す。O₁とO₂がジョブ1、O₃とO₄がジョブ2のオペレーションである。p_jの値はオペレーションO_jの作業時間を表す。実線のエッジはconjunctive arcと呼ばれ、ジョブ内のオペレーションの順序制約を表す。ダミーのソースからはジョブの最初のオペレーションへのエッジがあり、シンクへはジョブの最後のオペレーションからのエッジがある。破線のエッジはdisjunctive arcと呼ばれ、マシン内のジョブの順序制約を表す。例えばO₁からO₃へとdisjunctive arcの向きを決めると、マシン1ではジョブ1のO₁を実行してからジョブ2のO₃を実行するというように、各マシンのジョブ

の実行順序が決まる。disjunctive arcの向きの組合せが決まると、disjunctive graph上に閉路（ループ）が存在しなければ、オペレーションの順序制約が決定する。その順序制約を満たすスケジュールのうち、全体の作業時間（メイクスパン）が最小になるスケジュールを自動的に決定できる。そのときのメイクスパンは、作業時間を重みとしたソースからシンクへのパス長の最大値となる。なお、図1は一例であり、提案手法はdisjunctive graphで表現された任意のJSPに適用可能である。

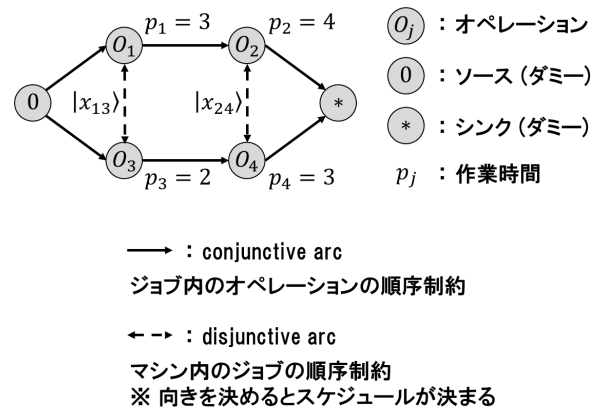


図 1 disjunctive graph の例 (2 ジョブ × 2 マシン)

提案アルゴリズムでは、disjunctive arcの向きの組合せを表現する量子ビット列 $|x\rangle = \otimes_{ij} |x_{ij}\rangle$ と、disjunctive graphのパスを表現する量子ビット列 $|p\rangle$ を用いる。量子ビットによる表現の仕方を、それぞれ図2と図3に示す。

$$|x_{ij}\rangle = \alpha|0\rangle + \beta|1\rangle, \quad |x_{ij}\rangle = \begin{cases} |1\rangle & \text{if } O_i \dashrightarrow O_j \\ |0\rangle & \text{if } O_j \dashrightarrow O_i \end{cases}$$

$$P(|x_{ij}\rangle = |0\rangle) = |\alpha|^2, \quad P(|x_{ij}\rangle = |1\rangle) = |\beta|^2$$

図 2 disjunctive arc の向きを表現する量子ビット

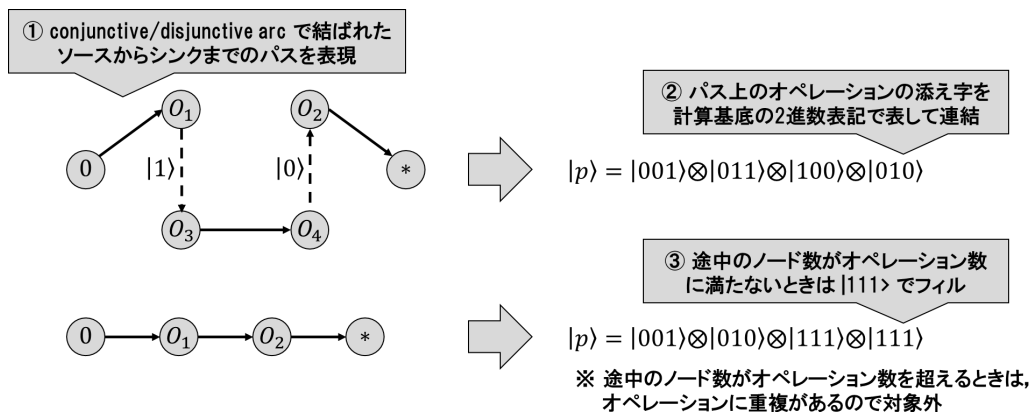


図 3 disjunctive graph のパスを表現する量子ビット列

※ 途中のノード数がオペレーション数を超えるときは、オペレーションに重複があるので対象外

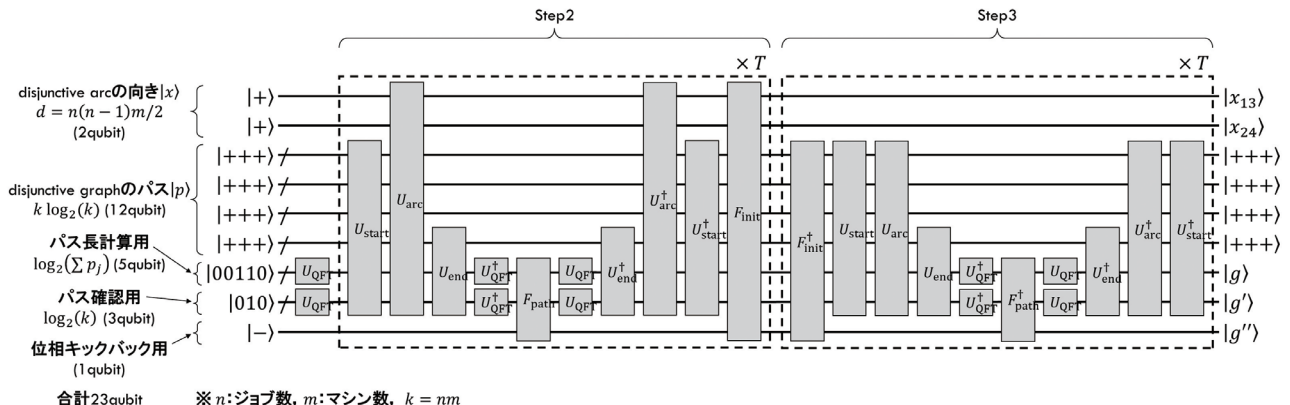


図 4 提案手法の量子回路図 (Step2, Step3)

以下、JSPのジョブ数を n 、マシン数を m とする。オペレーションの個数は $k := nm$ 、disjunctive arcの個数は $d := n(n-1)m/2$ となり、 $|x\rangle$ と $|p\rangle$ の量子ビット数は、それぞれ d と $k \log_2(k)$ になる。提案アルゴリズムの入力はdisjunctive graphで表現されたJSPと、パス長の閾値を表す自然数 N である。入力 N に対して、JSPの最小メイクスパンが N 未満であれば、提案アルゴリズム実行後、量子測定により高い確率でどのパスも長さ N 未満になるdisjunctive arcの向きの組合せが得られる。このとき得られるスケジュールのメイクスパンは N 未満であり、このことからJSPの最小メイクスパンが実際に N 未満であることが分かる。一方、JSPの最小メイクスパンが N 以上のときは、どのようなdisjunctive arcの向きの組合せでもメイクスパンは N 以上となり、提案アルゴリズム実行後に量子測定を繰り返してもメイクスパンが N 未満となるようなスケジュールは得られない。十分な回数でそのような結果となれば、高い確率で最小メイクスパンが N 以上であることが分かる。例えば二分探索で N の値を変えながら提案アルゴリズムを繰り返し実行することで、JSPの最小メイクスパンを決定できる。

提案アルゴリズムは以下の4つのステップからなる。

- (Step1) $|x\rangle$ と $|p\rangle$ の各量子ビットを、量子状態 $|0\rangle$ と $|1\rangle$ の均等な重ね合わせである $|+\rangle$ で初期化する。
- (Step2) グローバーのアルゴリズムにより、長さ N 以上のパスを表現している $|x\rangle$ と $|p\rangle$ の量子状態の組合せ $|x\rangle \otimes |p\rangle$ の確率振幅を増幅し、パスを表現していないビット列や長さ N 未満のパスの確率振幅を0に近づける。
- (Step3) Step2の逆演算により、 $|p\rangle$ の各量子ビットを初期状態の $|+\rangle$ に戻す。ただし、 $|x\rangle$ の量子ビットには量子ゲート操作を行わない。こうすることで、 $|x\rangle$ を量子測定したときの確率振幅をStep2実行後の状態から変化しない

ようにできる。これによりStep3実行後の $|x\rangle$ の確率振幅は、長さ N 以上のパスが1つ以上存在するときは高くなり、どのパスも長さ N 未満のときは0に近くなる。

(Step4) グローバーオペレータにより $|x\rangle$ の振幅確率を反転させ、長さ N 以上のパスが1つ以上存在する $|x\rangle$ の確率振幅が低くなるようにする。この状態で $|x\rangle$ の量子測定を行う。disjunctive graph上に閉路がある場合、2回以上実行されるオペレーションが存在するので実行不可能なスケジュールとなるが、その場合はいくらかでも長いパスが存在する。Step4でそのような実行不可能なスケジュールが解となる確率を低くすることができる。

図4にStep2とStep3の量子回路図を示す。 U_{QFT} は量子フーリエ変換の回路、 $U_{start}, U_{arc}, U_{end}$ は $|p\rangle$ がパスを表現しているかチェックするための回路で、それぞれ開始ノード、エッジ、終了ノードの正常性をチェックする。 F_{path} は $|p\rangle$ がパスを表現していて、かつ長さが N 以上のときに位相を -1 倍する。 F_{init} はグローバーオペレータで、位相を -1 倍した量子状態の振幅を増幅する。

図5にStep4のグローバーオペレータによる確率振幅の反転の様子を示す。実線で示した量子状態のベクトルを、破線の量子状態のベクトルを軸として反転させる。例えば、Step3実行後の $|x\rangle$ を $|+\rangle^{\otimes d}$ を軸として反転することで、 $|x\rangle$ と $|+\rangle^{\otimes d}$ のなす角 θ を2倍の 2θ にした $|x_{2\theta}\rangle$ が得られる(①)。同様の操作を繰り返すことで、 $|x_{3\theta}\rangle$ や $|x_{4\theta}\rangle$ など、任意の整数 T' に対して $|x_{T'\theta}\rangle$ が得られる(②③)。

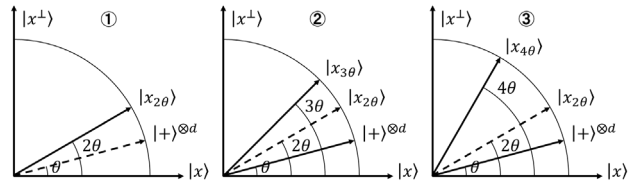


図 5 確率振幅の反転 (Step4)

表 1 小規模の問題によるシミュレーション結果

disjunctive arcの向きを表す量子ビット $ x_{13}\rangle$ $ x_{24}\rangle$		disjunctive graph 上の 可能なパス				パス長	量子ビット列の測定確率 $ x\rangle\otimes p\rangle$			長さ10以上のパス が存在するか？
							Step2	Step3	Step4	
0	0	O_1	O_2	*	*	7	0.0%	0.0%	49.0%	No
		O_3	O_4	*	*	5	0.0%			
		O_3	O_4	O_2	*	9	0.0%			
		O_3	O_1	O_2	*	9	0.0%			
0	1	O_1	O_2	*	*	7	0.0%	31.3%	0.8%	Yes
		O_1	O_2	O_4	*	10	25.2%			
		O_3	O_4	*	*	5	0.0%			
		O_3	O_1	O_2	*	9	0.0%			
1	0	O_3	O_1	O_2	*	12	24.8%	25.0%	1.3%	Yes
		O_1	O_2	*	*	7	0.0%			
		O_1	O_3	O_4	*	8	0.0%			
		O_1	O_3	O_4	O_2	12	25.1%			
1	1	O_3	O_4	*	*	5	0.0%	25.0%	0.0%	Yes
		O_3	O_4	O_2	*	9	0.0%			
		O_1	O_2	*	*	7	0.0%			
		O_1	O_2	O_4	*	10	25.0%			
		O_1	O_3	O_4	*	8	0.0%			
		O_3	O_4	*	*	5	0.0%			
		O_3	O_4	*	*	5	0.0%			

4. 小規模の問題によるシミュレーション結果

図1に示した小規模のJSPで提案アルゴリズムのシミュレーションを行った。このJSPではdisjunctive arcの向きが O_3 から O_1 , O_4 から O_2 のときにメイクスパンが最小値の9となる。 $N = 10$ として提案アルゴリズムをシミュレーションにより実行したときの結果を表1に記す。実行にはIBM Qiskit 0.36.1を用いた。Step2実行後の $|x\rangle\otimes|p\rangle$ の測定確率では、長さ10以上のパスを表す量子ビット列のみ確率が大きくなっている。Step3ではそのようなパスが1つ以上存在する $|x\rangle$ の確率のみ大きくなり、Step4でその確率振幅を反転することで、長さ10以上のパスが存在する $|x\rangle$ の確率が1.3%以下と十分低くなっていることが分かる。これにより実行不可能なスケジュールが解となる確率を十分低くすることができ、想定通りの結果となることを確認できた。

5. おわりに

スケジューリング問題をコスト最小化問題に帰着して量子コンピュータで解く方式では、制約の充足が課題となる。これを解決するため、disjunctive graphでJSPを表現し、disjunctive arcの最適な向きの組合せをグローバルのアルゴリズムを基に探索する量子アルゴリズムを考案した。小規模のJSPによるシミュレーションでは、想定通りの測定結果となることを確認した。

提案アルゴリズムではStep2とStep3で $T = O(2^{k/2})$ 回、Step4では $T' = O(2^{d/2})$ 回の繰り返しの計算がある。全体では $O(2^{(d+k)/2})$ 回の繰り返し計算が必要となり、量子ゲ

ートの操作回数が多いため、現在のNISQデバイスでは実行できない。ただし、従来のコンピュータではdisjunctive arcの向きの探索に $O(2^d)$ の計算量が必要となり、量子誤り耐性のあるデバイスが実現されれば、2乗加速での計算速度の向上が見込める。

参考文献

- 1) IBM Research Blog (2022/12/6 閲覧), <https://rease.arch.ibm.com/blog/ibm-quantum-roadmap-2025>
- 2) 量子情報技術 科学技術に関する調査プロジェクト報告書, 国立国会図書館 (2022年3月)
- 3) D. Venturelli, et al.: Job shop scheduling solver based on quantum annealing, Proceedings of ICAPS-16 Workshop on Constraint Satisfaction Techniques for Planning and Scheduling (2016)
- 4) David Amaro, et al.: A case study of variational quantum algorithms for a job shop scheduling problem, EPJ Quantum Technology, Vol. 9, No. 1 (December 2022)
- 5) A.M.S. Zalzal, P.J. Fleming: Genetic algorithms in engineering systems, The Institution of Electrical Engineers (1997)
- 6) Lov K. Grover: A fast quantum mechanical algorithm for database search, Proceedings of the twenty-eighth annual ACM symposium on Theory of computing (1996)



飯塚 新司 2008年入社
 研究開発部
 クラウドDWH, データ分析, AI,
 量子コンピュータ関連技術の研究