7th European Conference on Industrial Engineering and Operations Management
Augsburg, Germany, July 16-18, 2024
Publisher: IEOM Society International, USA DOI: 10.46254/EU07.20240089
Published: July 16, 2024

Integrating Quantum Computing into Smart Maintenance Scheduling Problems

Marius R. Bernreuther
Hochschule Reutlingen, ESB Business School, Stellenbosch University
Reutlingen, Baden-Würtemberg, Germany
marius.bernreuther@student.reutlingen-university.de

Dominik Lucke
Hochschule Reutlingen, ESB Business School
Reutlingen, Baden-Würtemberg, Germany
dominik.lucke@reutlingen-university.de

Johannes L. Jooste
Stellenbosch University
Stellenbosch, Western Cape, South Africa
wyhan@sun.ac.za

Abstract

This paper investigates the integration of quantum computing into smart maintenance, which allows integrated scheduling of maintenance and production to enhance decision-making within manufacturing environments. Uncoordination and the lack of integrated scheduling of maintenance and production plans lead to significant economic inefficiencies. A literature review revealed the gap in the integration of modern and newly emerging quantum computing algorithms and a three-step optimization approach is proposed. The paper showcases the feasibility of quantum computing for smart maintenance scheduling and illustrates a way of including quantum computing in complex integrated scheduling problems. The approach encompasses creating an integrated production and maintenance schedule via simulation-based optimization and metaheuristics and applying the quantum approximate optimization algorithm for prescheduling maintenance tasks.

Keywords
Smart Maintenance, Maintenance Scheduling, Quantum Computing, Optimization, Smart Maintenance Scheduling

1. Introduction
Based on an empirical, inductive research approach Bokrantz et al. (2020) defined smart maintenance as *'an organizational design for managing the maintenance of manufacturing plants in environments with pervasive digital technologies'*, and according to the definition, smart maintenance consists of four dimensions: data-driven decision-making, human capital resource, internal & external integration. One key aspect is the integration of production and maintenance to schedule integrated.

The objectives of the maintenance function and production function are different and their goals are not complementary. Maintenance and production plans are traditionally scheduled separately, which ignores the interdependencies between maintenance and production, leading to significant economic losses (Kolus et al. 2020). The misalignment of maintenance and production schedules results in wrongly chosen maintenance time slots, which impacts the productivity of the production system (Glawar et al. 2018). Determining and evaluating the cost of planned and unplanned downtime is a complex task for production of parts manufacturing companies, but it is indicated that the costs for unplanned downtime are significantly higher than the cost of planned downtime (Salonen and Tabikh 2016). According to Blameuser et al. (2015), 93% of surveyed manufacturing companies think, that maintenance will be of high or very high importance in the future and 97% of surveyed companies, optimize their maintenance processes, leading to better transparency, maintenance quality, better processes as well as higher employee satisfaction. Furthermore, according to ServiceMax and Bourne (2021), for 70% of surveyed companies achieving zero unplanned downtime is a very high priority and 88% think that their organization could improve preventing downtime in general and unplanned downtime impacts the ability to deliver products and services to customers as well as decreasing the production time on critical assets.

To summarize, there is currently significant potential for improvement regarding the planning and scheduling of maintenance activities within manufacturing companies, resulting from the lack of integrated scheduled activities on production assets, which results in the development of smart maintenance and smart maintenance scheduling.

The main objective of this paper is the development of an optimization model, which enables smart maintenance scheduling for practical applications. A major research contribution is the investigation of the feasibility of integrating quantum computing (QC) algorithms, which promise speedups for identifying solutions for complex problems, into smart maintenance scheduling and proposing an approach, and its implementation to do so. The associated results shall be analyzed. The study aims to bridge the gap between real-world problems and the theoretical optimization models and the potential benefits of quantum computing for optimization problems. Due to the current status quo of quantum computing the goal is not an increase of calculation speed. The authors are aware, that quantum supremacy has only been proven for special problems and the true promise of quantum computing, of speeding up practical applications has not been realized yet (Cerezo et al. 2021).

Within the conclusion the researchers synthesize their research contribution and outline the potential impact of quantum computing, its current feasibility for smart maintenance optimization problems and outline future explorations within the field.

2. Literature review

To identify relevant literature within the field of research, multiple search strings and four scientific databases (ScienceDirect, IEEE, Emerald and Web of Science) were considered, and inclusion criteria such as article type, language, access and year were applied. The scientific publications were scanned by their keywords, abstract and full-text and backward snowballing was applied to ensure that the literature basis is relevant and comprehensive.

The corresponding scientific integrated scheduling approaches consider a variety of modelling optimization techniques. Zahedi et al. (2019) propose a quadratic mixed integer programming model for batch production of a single item processed on a flow shop system based on two machines, minimizing the total cost. The cost-saving potential of integrated scheduling is also pointed out by Kolus et al. (2020), who consider a single-machine shop floor setup and emphasize the increasing complexity due to integrated scheduling. Due to the holistic nature of smart maintenance scheduling different aspects are implemented into the objective function and constraints. For instance, Liu et al. (2020) include service level and production capacity constraints, Wang et al. (2019) consider a stochastic deteriorating single-stage production system and Zahedi et al. (2019) consider inventory, rework and different maintenance costs. The Weibull distribution is often used to model maintenance, and an as-good-as-new status after a maintenance job is assumed (Wang et al. 2019; Hafidi et al. 2021; Hafidi et al. 2018). Due to the complex nature of the corresponding problems many authors mention the potential of implementing metaheuristics to discover the complex solution space. Metaheuristics can identify a non-optimal solution in a reasonable amount of time, which is often preferred for practical application.

One strategy is the combination of simulation-based optimization (SBO) with a metaheuristic optimization approach, such as genetic algorithms (GA) or ant colony optimization (ACO). Figure 1 illustrates the basic concept of SBO, where an optimization procedure, such as GA or ACO optimizes input variables to find appropriate solutions based on the value of the objective function for the corresponding input variables.

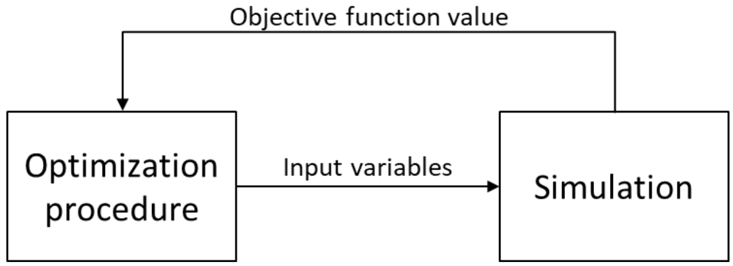

Figure 1. Illustration of integrating simulation-based optimization with metaheuristic optimization procedures

Triska et al. (2021) and Frantzén et al. (2022) emphasize the potential of SBO with metaheuristics to model and solve complex scheduling problems, such as the production of wavers, a job shop scheduling problem, with multiple processing steps on different machines with unique properties. GA and GA-based hybrid metaheuristics such as GA and tabu search (TS) are feasible approaches for optimizing integrated scheduling problems to find appropriate solutions (Hafidi et al. 2021; Boudjelida 2019).

Zhai et al. (2021) present a novel framework to integrate production scheduling with predictive maintenance by utilizing generative deep learning, specifically a conditional variational autoencoder (CVAE) to derive an operation-specific health indicator. The prognosis model can provide a quantitative measure of degradation for a specific production sequence, which can be directly integrated into scheduling algorithms, that can deal with maintenance and production constraints (Zhai et al. 2021). Another approach to integrate raw machine, maintenance and telemetric data into integrated scheduling is proposed by Yeardley et al. (2022) including regression and classification machine learning techniques and subsequent scheduling, modelled as a mixed-integer program. Yeardley et al. (2022) and Wocker et al. (2020) both consider machine learning and compare and evaluate different machine learning algorithms and their performance.

To summarize, classical analytical approaches, metaheuristics, machine learning and CVAE were investigated by the research community extensively. The authors identified that currently the potential of quantum computing has not been investigated for smart maintenance scheduling. With the first gate-based quantum computer for commercial application launched in 2019 by IBM, it is a new and promising technology (Chan 2019). By leveraging quantum mechanical properties, quantum computers promise supremacy over classical computers and decrease computation time compared to classical computers (Arute et al. 2019; Zhong et al. 2020).

3. Approach

A three-step optimization approach is used, consisting of the following steps:

1. Create an integrated production and maintenance schedule through a combination of SBO and GA, which is a common strategy identified by the literature review.
2. Determine the criticality of the production assets based on utilization rate, defined as the utilization time of a machine divided by the total time required to process all production jobs.
3. Starting with the most critical production asset iteratively create an analytical optimization model and solve it via quantum approximate optimization algorithm (QAOA).

The production layout and processing consist of three production machines (M2, M2, M3) and two buffers (B1, B2), as illustrated in Figure 2. The production layout is based on insights from expert interviews focused on part manufacturing and provides a common abstraction representing typical production layouts, which includes serial production.

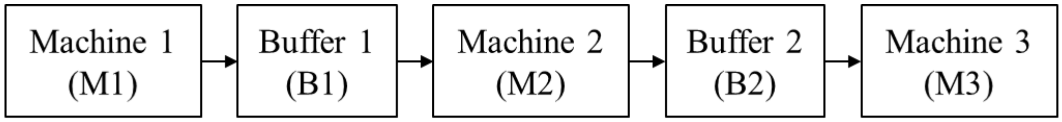

Figure 2. Production layout consisting of three machines and two buffers

Due to the dynamic nature and high complexity resulting from smart maintenance scheduling, production layout and constraints, finding appropriate solutions only via QC is impractical due to the current limitations of QC. The proposed approach combines the ability of SBO and GA to find good solutions in a reasonable amount of time and emphasizes the potential to identify any prescheduling potentials to leverage maintenance jobs. Within the third step, QC is used to determine if any maintenance job can be prescheduled to an earlier maintenance time spot. This increases the value of the production break by doing multiple maintenance measures at the same time, if possible. Due to ordering the production assets by their utilization, it is implicitly included, that prescheduling on a critical machine has a higher value for a manufacturing company since the required worker availability is constrained. The iterative nature of the approach allows a reduction of required qubits since solving the problem for all production assets at the same time would lead to more decision variables.

Similar to Kolus et al. (2020), Zahedi et al. (2019), Liu et al. (2020), Wang et al. (2019), Hafidi et al. (2021) and Hafidi et al. (2018) the necessity of a maintenance job conducted on a machine is determined by the Weibull distribution.
The decision if a maintenance job shall be performed before a production job is determined by:

$$v_{Prev} < v_{corr} \int_0^T \frac{\alpha}{\beta}\left(\frac{x}{\beta}\right)^{\alpha-1} e^{-(x/\beta)^{\alpha}} dx, \qquad \text{for } T \geq 0; \alpha, \beta > 0 \qquad (1)$$

Where T is equal to the runtime, v_{corr} being the total cost for corrective maintenance and v_{Prev} being the total cost of preventive maintenance. v_{Prev} *and* v_{corr} consider the required time, a due date penalty and material cost. A maintenance job is assumed to set T *to* 0, creating an as-good-as-new state, similar to Hafidi et al. (2021), Hafidi et al. (2018) and Wang et al. (2019).

Firstly, based on planned production jobs, maintenance data, layout data and machine status an integrated schedule is created through a genetic algorithm simulation-based optimization (GASBO), which iteratively aims to find approximate sufficient solutions for the job order of the flow shop scheduling problem. The objective function value is determined based on machine processing times, maintenance costs and a time-related penalty. As illustrated in Figure 2 the production layout consists of three production steps, which need to be performed in a given order, with buffers between each production step. Machines and buffers are set to be non-equal, having a unique parametrization. Starting with the most critical production asset, determined by the utilization, an analytical optimization model is created and solved with QAOA to determine if a maintenance job can and shall be prescheduled to an earlier time spot. For practical application, it is often possible to do multiple maintenance measures at the same time, to lower downtimes and boost the value of a break of production due to maintenance.

Firstly introduced by Farhi et al. (2014), QAOA is a quantum algorithm producing approximate solutions for combinatorial optimization problems. QAOA is a hybrid-quantum algorithm since it uses a classical solver to optimize iteratively a parametrized quantum circuit. Figure 3 demonstrates the basic architecture of QAOA, where a set of β and γ parameters is iteratively optimized by a classical solver to minimize a cost Hamiltonian H_c. H_B being the mixer operator, whose role is introducing transition between different quantum states to allow investigating the solution space of the corresponding problem. The initial superposition of the quantum states of a qubit is achieved by applying a Hadamard gate (H) to each qubit. The number of β and γ parameters are often referred to as reps.

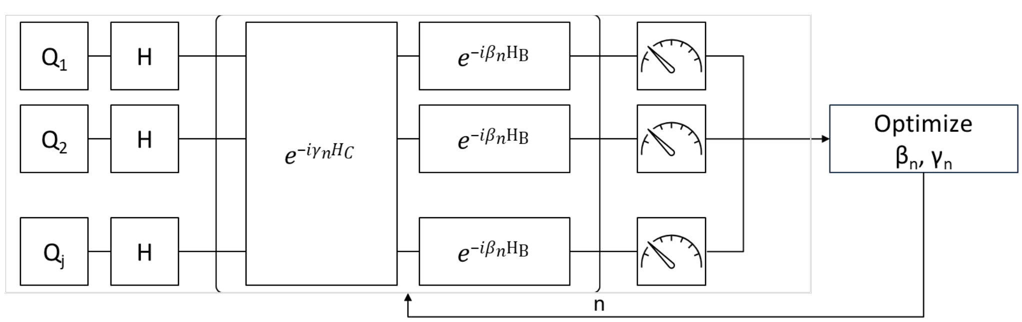

Figure 3. QAOA architecture consisting of $\boldsymbol{j}$ qubits and $\boldsymbol{n}$ $\boldsymbol{\beta}$ **and** $\boldsymbol{\gamma}$ parameters

4. Implementation of solution approach

In the following, the described approach, consisting of three steps is implemented for the scheduling layout, shown in Figure 2. The implementation of the first and second step of the approach is described within section 4.1. and the third step is described in section 4.2.

4.1. Simulation-based GA optimization

The iterative process of each simulation is illustrated Figure 4, a machine can only do a job, if there is a processed job in the previous buffer, and the maintenance decision is done based on the previously presented Weibull distribution.

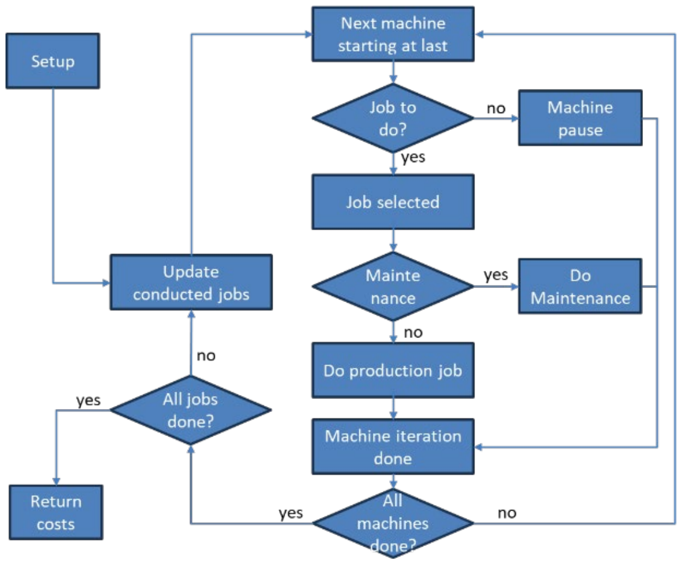

Figure 4. Flowchart of SBO

The symbols used within the GASBO and the objective function to be minimized are described in Table 1.

Table 1. Description of relevant parameters for GASBO

Symbol	Type	Description
m	Index	Machine index
k	Index	Maintenance job index
b	Index	Buffer index
α_m, β_m	Parameters	Weibull parameter for machine m
c_{prev}, t_{prev}	Parameters	Material cost and time parameters for preventive maintenance
c_{corr}, t_{corr}	Parameters	Material cost and time parameters for corrective maintenance
v_{prev}, v_{corr}	Parameters	Total cost for maintenance measures with penalty, worker time and material cost
M_{mk}	Indicator	1 if maintenance job k is conducted on machine m, 0 else
c_p	Parameter	Penalty rate after the due date
T_{total}	Parameter	Total required time to produce all jobs
T_p	Parameter	Penalty threshold
$T_{run,m}$	Parameter	Previous runtime of the machine m
$B_{cap,b}$	Parameter	Capacity of the buffer b
T_s	Parameter	Setup time before conducting a job on a machine

The following objective function aims to be minimized:

$$\text{Min} \quad \sum_{m=1} T_m + c_{prev} \sum_{m=1} \sum_{k=0} M_{mk} + c_p(T_{total} - T_p) \qquad c_{prev}, c_p > 0; T_{total} \geq T_p \quad (2)$$

It consists of the makespan of all machines, the number of maintenance jobs performed on the machines multiplied by a cost term c_{prev} and a penalty term depending on the total time needed to complete all production jobs.

Eight jobs have to be performed and processed in the given order, illustrated in Figure 2. The processing time and corresponding amount of the batch jobs are listed in Table 2. Since all jobs are unique a total of $\mathbf{8}! = \mathbf{40320}$ possible orders are possible. To increase generalization, generic units like time units (TU) and cost units (CU) are used. Before a production job can be conducted a setup, requiring $\boldsymbol{T_s}$ time units needs to be performed.

Table 2. Time units and amount of production jobs for each production asset

	J1 [TU, amount]	J2 [TU, amount]	J3 [TU, amount]	J4 [TU, amount]	J5 [TU, amount]	J6 [TU, amount]	J7 [TU, amount]	J8 [TU, amount]
M1	40, 8	50, 5	45, 3	20, 4	60, 6	30, 2	10, 2	15, 1
M2	56, 8	70, 5	63, 3	28, 4	84, 6	42, 2	14, 2	21, 1
M3	28 ,8	35, 5	32, 3	14, 4	42, 6	21, 2	7, 2	11, 1

4.2. Prescheduling optimization problem

To implement QAOA, firstly an analytical model needs to be constructed, which serves as the starting point to construct the cost Hamiltonian for the QAOA. The parameters for the corresponding prescheduling analytical mixed integer linear program (MILP) are presented in Table 3

Table 3. Description of symbols of prescheduling MILP

Symbol	Type	Description
x_{ij}	Variable	Binary decision variable indicating if job i is scheduled at slot j.
i	Index	Index for maintenance jobs
j	Index	Index for time slots
a	Parameter	Real-valued coefficient for the weight of the first term of the objective function.
b	Parameter	Real-valued coefficient for the weight of the second term of the objective function.
T_{ii}	Parameter	The time associated with conducting a job at its original scheduled time
T_{ij}	Parameter	The time associated with conducting a job at a previous time
I	Index set	Set of indices for maintenance jobs
V_i	Index set	Subset representing feasible prescheduling spots for job i

The analytical mixed integer linear program can be expressed as:

$$\text{Min} \quad a\sum_{i\in I} x_{ii} + b\sum_{i\in I}\sum_{j\in V_i} x_{ij}\left(T_{ii} - T_{ij}\right) \tag{3}$$

$$\text{s.t.} \quad \sum_{j\in V_i} x_{ij} = 1 \qquad \forall i \in I \tag{4}$$

$$x_{ij} \in \{0, 1\} \qquad \forall i \in I, j \in V_i \tag{5}$$

Where we choose

$$I = \{1, \dots, n\} \qquad n \in \mathbb{N} \tag{6}$$

$$V_i \subseteq \{1, \dots, n\} \quad and \quad i \in V_i \qquad n \in \mathbb{N} \tag{7}$$

$$0 < T_{ij} < T_{ij} < T_{ii} \qquad \forall j < j < i \tag{8}$$

$$T_{ij} = T_{ij} \qquad \forall i \in I, j \in V_i \tag{9}$$

Explanation of the model:

- Equation (3) defines the objective function of the mixed integer linear program consisting of two terms. The first term incorporates the $i = j$ condition directly, scaled by a coefficient, and deals with jobs that are not prescheduled. The second term accounts for the residual value of a prescheduled job and adds a penalty of b for the remaining lifetime of a prescheduled part, scaled by its remaining lifetime. In case of prescheduling the replaced part has a residual value, which grows linear with $T_{ii} - T_{ij}$.
- Equation (4) ensures that each job is scheduled exactly once, which is a usual constraint for scheduling problems. Only slots j, which are element of V_i are considered.
- Equation (5) defines the binary decision variables for a job i scheduled at a time spot j.The amount of binary decision variables is determined by I and V_i.
- Equation (6) specifies the set of job indexes, e.g. for 3 maintenance jobs, $n = 3$ and $I = \{1, 2, 3\}$.
- Equation (7) defines the subset of feasible scheduling times slots for a job i, which consists of at least the original time spot, scheduling job i at time slot i. E.g. for $i = 3$, V_i consists at least of $V_i = \{3\}$, but not more than $V_i = \{1, 2, 3\}$. This incorporates the condition, that a job i can not be scheduled at a slot $i < j$.
- Equation (8) assures, that the time spots are in the correct order, time slot $j + 1$ must be greater than the corresponding time of slot j. This ensures, that the second term of the objective function is working correctly.
- Equation (9) explicitly sets, that the slots j are independent from i.

To construct the associated cost Hamiltonian H_c for the MILP problem, the MILP problem is modelled, based on the outcome of the GASBO and the worker availability. A job can only be rescheduled to an earlier time spot if there is an available worker. The MILP was modelled in docplex and linear constraints considering only one binary variable was converted to constant penalty terms within the objective function, because no prescheduling is possible. This preprocessing allows a reduction of required qubits while maintaining the same problem to solve.
Any MILP with binary decision variables can be converted into a QUBO problem, which is considered to be NP-hard, making it non efficiently solvable, and commonly formulated as follows (Egger et al. 2021; Lasserre 2016):

$$\min_{x \in \{0,1\}^n} x^T \sum x + \mu^T x \tag{10}$$

Where x is a vector consisting of binary decision variables, $\sum \in \mathbb{R}^{n*n}$ being a symmetric matrix and $\mu \in \mathbb{R}^n$ being a vector, containing the linear terms coefficients. To convert the MILP into QUBO form, the linear equality constraints, resulting from equation (4), are converted into quadratic penalty terms within the objective function as follows:

$$\sum_{j \in v_j} 1x_{ij} = 1 \longrightarrow \lambda \left(1 - \sum_{j \in V_j} 1x_{ij} \right)^2 \tag{11}$$

Where λ is a high real-valued number. Due to the implementation in qiskit, the Ising Hamiltonian can be expressed as a sparse Pauli operator to increase representation efficiency since the cost operator does not operator on all qubits simultaneously. The resulting $H_c$is in the following form:

$$H_c = \sum_i c_i P_i \tag{12}$$

Where each $\boldsymbol{P_i}$ is a tensor product of Pauli matrices and the identity matrix, and $\boldsymbol{c_i}$ representing the magnitude of the contribution of the individual term

5. Results and discussion
In this section the results for the scheduling problem are presented. Since this paper focuses on the feasibility, the authors focus on the most critical production asset within step 3 of the presented approach. In section 5.1. the first and second step of the implemented approach are presented, and in section 5.2. the optimization of the most critical production asset via QAOA. Finally in section 5.3. improvement measures are suggested and presented.

5.1 Simulation-based GA optimization
The implementation parameters for the GASBO, consisting of the machine parametrization, the buffer capacities, the penalty configuration and the maintenance parameters are provided in Table 4.

Table 4. Parametrization of GASBO

SBO parameter	Value	SBO parameter	Value
M1 ($T_{run,1}, \alpha_1. \beta_1$)	50 TU, 3.5, 100	T_s	1 TU
M2 ($T_{run,2}, \alpha_2. \beta_2$)	20 TU, 3.5, 100	T_p	350 TU
M3 ($T_{run,3}, \alpha_3. \beta_3$)	30 TU, 3.5, 90	c_p	2 CU
$B_{cap,1}$	10 amount	$c_{prev}, t_{prev}, v_{prev}$	5 CU, 5 TU, 20
$B_{cap,2}$	8 amount	$c_{corr}, t_{corr}, v_{corr}$	45 CU, 15 TU, 90

The GA uses a population size of 80, a maximum number of generations of 20, 50 parents and elitarism is included with a parameter of 2. The proposed schedule of the GA is illustrated in Figure 5. When comparing the results to the results from a deterministic approach, which guarantees finding the best solution to the GA result, the GA was able to identify the global minimum with a corresponding cost of 1057 CU and a total time of 441 TU. A total of eight maintenance jobs are to be conducted, three for M1, three for M2, and two for M2. The initial schedule results in 1100 CU and 460 TU needed to complete all jobs, clearly indicating, that the result of the GASBO is superior in terms of total time and cost associated with the schedule.

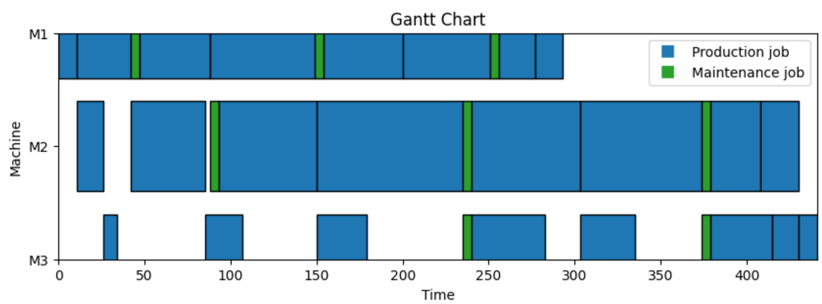

Figure 5. Machine occupancy plan resulting from SBO GA optimization

Since the mean fitness and best fitness are improving with each generation, the convergence of the proposed algorithm is appropriate. The evolution of mean and best fitness over the generations of the GA is illustrated in Figure 6 .

Figure 6. Evolution of best fitness, mean fitness and max fitness within each population of the GA

The utilization rates of the three production assets that determine the order of prescheduling via QC are as follows: M1 has a utilization rate of 61.22%, M2 is utilized at 85.49%, and M3 has a utilization of 42.63%. Therefore, M2 is identified as the most critical production asset, followed by M1, which is reflected in the implementation of QAOA in the next section.

5.2 Prescheduling MILP optimization using QAOA

The optimization prescheduling problem shall be illustrated for the most critical production asset, in this case, M2 since the utilization rate is the highest among all production assets. The energy of the system and the probability of finding the best eigenstate for the optimal parameterized circuit can be used to determine the convergence and result of QAOA. Lower energies, indicate that the system is closer to the ground state, which corresponds to the best eigenstate. Within QAOA, constrained optimization by linear approximation (COBYLA) was shown to obtain good solutions in terms of the energy of the system and computation time for a variety of problems (Fernández-Pendás et al. 2022). The subset V_j is determined by the worker availability and the maintenance times, prescheduling is only possible if the worker is available during the earlier maintenance time. The parameters for the implementation for the highest utilized machine are listed in Table 5.

Table 5. Parameters for MILP problem

Parameter	Value	Parameter	Value
Maintenance times M2	88-93, 235-240, 374-379	b	0.025
Worker availability	0-100, 140-200	λ	100
a	5	Classical solver	COBYLA

The number of eigenstates of the results is equal to $2^4 = 16$, since the circuit operates on four qubits. The best result, calculated via a classical eigensolver, has an objective value of 13.675, with the corresponding energy of the system of -101.7375. The validation via a classical computer can either be done by applying a classical solver like Cplex to the MILP or the NumPy minimum eigensolver to the qubit operator. Table 6 illustrates the energy and probability of finding the best result of the optimal parametrized circuit for an amount of β, γ parameters ranging from 1 to 6. The energy and probability of finding the best eigenstate do not improve significantly for the given problem

Table 6. Comparison of energy and probability of best eigenstate based on the number of parameters for the QAOA circuit

Amount β, γ (reps)	Energy	Probability of best eigenstate
1	-85.84	20.94%
2	-96.37	24.89%
3	-86.72	13.83%
4	-88.61	21.81%
5	-97.24	15.94%
6	-95.57	53.08%

The convergence for different amounts of β and γ (reps) is illustrated in
Figure 7. The initial superposition resulting from the H-gate results in high energy values at the beginning of the iterations as illustrated in Figure 7.

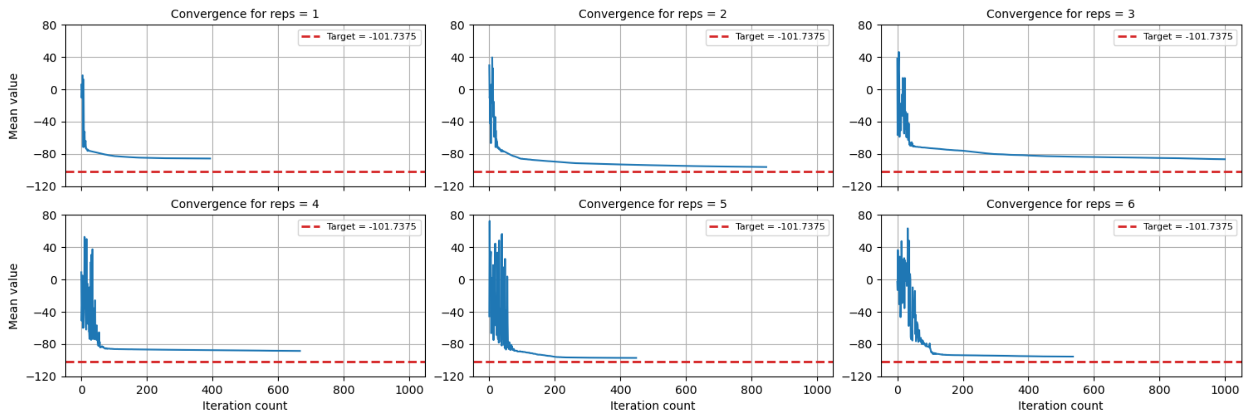

Figure 7. Convergence of QAOA for number of $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ parameters ranging from 1 to 6

The associated energy for the first iterations is significantly higher than the final energy, which indicates, that the starting parametrization is not ideal. It needs to be considered, that increasing the amount of β and γ parameters, leads to a deeper circuit, leading to problems like noisy amplification, and optimization complexity of the classical solver when implementing on a noisy intermediate-scale quantum (NISQ) device. Although the best eigenstate can be identified, a better convergence and higher probability of the best eigenstate would be appreciated since it is a measure how close the system is to its ground state, which is the best eigenstate.

5.3 Proposed improvements

To improve the convergence of the QAOA algorithm a custom initial state and mixer operator are included to warm-start the QAOA. To do so, the QUBO problem is relaxed, making it not NP-hard and solved via a classical solver like Cplex or Guropi (Egger et al. 2021). The initial state for qubit q_j is determined by the solution c_j for the relaxed problem and a R_y rotation with angle $\theta = 2\arcsin\left(\sqrt{c_j^*}\right)$ is applied. The mixer operator has the initial state as ground state and the custom mixer is defined as:

$$H_{B,j} = \begin{pmatrix} 2c_j^* - 1 & -2\sqrt{c_j^*(1-c_j^*)} \\ -2\sqrt{c_j^*(1-c_j^*)} & 1 - 2c_j^* \end{pmatrix} \tag{13}$$

and is implemented applying $R_Y(\theta_j), R_Z(-2\beta), R_Y(-\theta_j)$ to q_j, resulting in a warm-started QAOA (WS-QAOA). Table 7 compares the energy and probability of the best eigenstate of the cold-started QAOA (CS-QAOA) and WS-QAOA. The WS-QAOA provides a higher probability of the best eigenstate and lower energy levels for all amounts of β and γ.

Table 7. Comparison of CS- QAOA & WS- QAOA based on energy & probability of best eigenstate for reps 1 to 6

Amount β, γ (reps)	CS-QAOA energy	WS-QAOA energy	CS-QA–A - Probability of best eigenstate	WS-QAOA Probability of best eigenstate
1	-85.84	-97.68	20.94%	95.80%
2	-96.37	-98.27	24.89%	96.41%
3	-86.72	-101.64	13.83%	99.90%
4	-88.61	-101.73	21.81%	99.99%
5	-97.24	-101.73	15.94%	99.99%
6	-95.57	-101.69	53.08%	99.95%

The corresponding convergence is illustrated in Figure 8. The energies for different amounts of β and γ parameters, ranging from 1 to 6 are lower and the number of evaluations is lower indicating better convergence than the CS-QAOA.

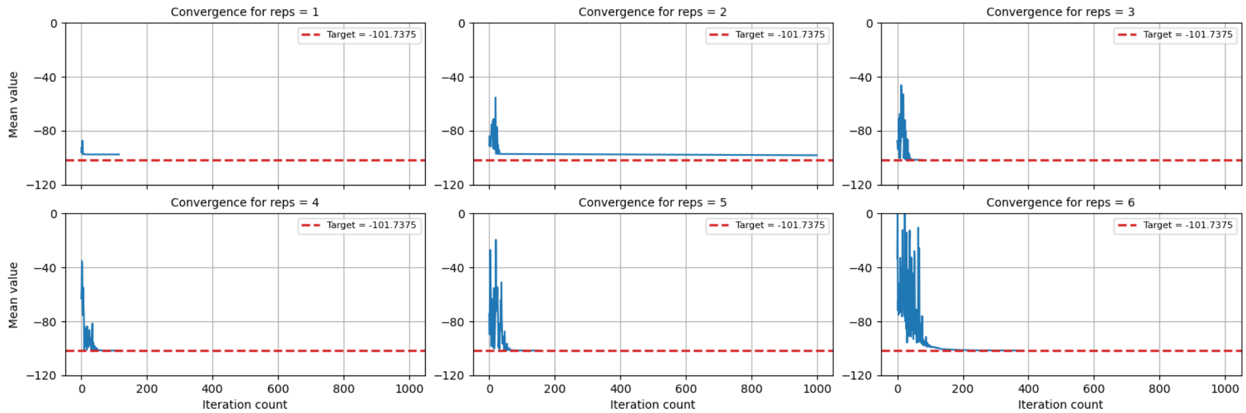

Figure 8. Convergence of WS-QAOA for number of $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ parameters (reps) ranging from 1 to 6

WS-QAOA showed significantly better results in terms of energy, convergence and probability of best eigenstate compared to CS-QAOA, making it a better approach for the given problem.The initial fluctuation is smaller, indicating that the custom initial state superposition is better than the generic H-gate superposition.

6. Conclusion and outlook

The paper proposes an approach to implement quantum computing into smart maintenance scheduling by a three-step optimization approach and illustrates the feasibility of including quantum computing in smart maintenance scheduling. The highest utilized machine within the scheduling problem was identified and an analytical MILP model was formulated. Iteratively optimizing the production assets based on their utilization incorporates the assumption, that optimizing critical production assets is of higher priority than production assets with low criticality. The iterative approach reduces the complexity of the problem, and allows the usage of less qubits.

A method to convert the MILP into a cost Hamiltonian was proposed and the results depending on different sets of β and γ as well as their convergence were presented. To improve performance a WS-QAOA approach was presented, implemented, and compared to CS-QAOA. The WS-QAOA showed significantly better results in terms of convergence, energy, and probability of finding the best eigenstate, which is important since the depth of the quantum circuit grows linear with the amount of β and γ parameters. The feasible prescheduling time slots only considered one parameter, worker availability, which could be enhanced easily to consider more, individual parameters, such as required maintenance equipment, skill levels or multiple workers. The proposed MILP was formulated in a general way, including complex conditions to determine the elements within V_i can be incorporated easily. The modelling of the problem plays a significant role when implementing quantum computing into scheduling problems and trivial decision variables can be converted and included into the objective function to reduce the required amount of qubits. Further research could investigate the implementation of other quantum computing algorithms for smart maintenance scheduling, apply the approach to a real-world scenario for a manufacturing company or investigate approaches to efficiently deal with a high number of maintenance jobs to be considered, which could lead to a high number of required qubits to solve the problem.

References

Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J. C., Barends, R. et al., Quantum supremacy using a programmable superconducting processor, *Nature*, vol. 574, no. 7779, pp. 505-510, 2019.

Blameuser, R., Galonske, M. and Gehrmann, S., Gegenwart und Zukunft der Gegenwart und Zukunft der technischen Instandhaltung. Die technische Instandhaltung im Zeitalter von Industrie 4.0, https://www.greengate.de/fileadmin/users/redakteur/redakteur_upload/GreenGate/RIH_Check/Studie_Instandhaltung_Industrie_4.0.pdf, Accessed 3/25/2024.

Bokrantz, J., Skoogh, A., Berlin, C. and Stahre, J., Smart maintenance: instrument development, content validation and an empirical pilot, *International Journal of Operations & Production Management*, vol. 40, no. 4, pp. 481-506, 2020.

Boudjelida, A., On the robustness of joint production and maintenance scheduling in presence of uncertainties, *Journal of Intelligent Manufacturing*, vol. 30, no. 4, pp. 1515-1530, 2019.

Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S. C., Endo, S., Fujii, K. et al., Variational quantum algorithms, *Nature Reviews Physics*, vol. 3, no. 9, pp. 625-644, 2021.

Chan, R., IBM unveils the world's first quantum computer that businesses can actually use to solve previously impossible problems, 1/13/2019, https://www.businessinsider.com/ibm-unveils-ibm-q-system-one-the-first-commercial-quantum-computer-2019-1, Accessed 3/5/2024.

Egger, D. J., Mareček, J. and Woerner, S., Warm-starting quantum optimization, *Quantum*, vol. 5, p. 479, 2021.

Farhi, E., Goldstone, J. and Gutmann, S., A Quantum Approximate Optimization Algorithm, *arXiv: Quantum Physics,* 2014.

Fernández-Pendás, M., Combarro, E. F., Vallecorsa, S., Ranilla, J. and Rúa, I. F., A study of the performance of classical minimizers in the Quantum Approximate Optimization Algorithm, *Journal of Computational and Applied Mathematics*, vol. 404, p. 113388, 2022.

Frantzén, M., Bandaru, S. and Ng, A. H., Digital-twin-based decision support of dynamic maintenance task prioritization using simulation-based optimization and genetic programming, *Decision Analytics Journal*, vol. 3, 2022.

Glawar, R., Karner, M., Nemeth, T., Matyas, K. and Sihn, W., An Approach for the Integration of Anticipative Maintenance Strategies within a Production Planning and Control Model, *Procedia CIRP*, vol. 67, pp. 46-51, 2018.

Hafidi, N., Barkany, A. E. and Mahmoudi, M., Modelling and Optimization of Integrated Planning of Production and Maintenance with Subcontract Constraint, *International Journal of Engineering Research in Africa*, pp. 184-203, 2018.

Hafidi, N., El Barkany, A., EL Mhamedi, A. and Mahmoudi, M., Joint optimization of production and maintenance for multi-machines subject to degradation and subcontracting constraints, *Journal of Quality in Maintenance Engineering*, vol. 27, no. 1, pp. 64-91, 2021.

Kolus, A., El-Khalifa, A., Al-Turki, U. M. and Duffuaa, S. O., An integrated mathematical model for production scheduling and preventive maintenance planning, *International Journal of Quality & Reliability Management*, vol. 37, no. 6/7, pp. 925-937, 2020.

Lasserre, J. B., A MAX-CUT formulation of 0/1 programs, *Operations Research Letters*, vol. 44, no. 2, pp. 158-164, 2016.

Liu, B., He, K. and Xie, M., Integrated production and maintenance planning for a deteriorating system under uncertain demands, *IFAC-PapersOnLine*, vol. 53, no. 3, pp. 222-226, 2020.

Salonen, A. and Tabikh, M., Downtime Costing—Attitudes in Swedish Manufacturing Industry, In Koskinen, K. T., Kortelainen, H., Aaltonen, J., Uusitalo, T., Komonen, K., Mathew, J., Laitinen, J. (Ed.), *Proceedings of the 10th World Congress on Engineering Asset Management*, pp. 539-544, 2016.

ServiceMax and Bourne, V., 2021, After the Fall. The Costs, Causes & Consequences of Unplanned Downtime, https://lp.servicemax.com/rs/020-PCR-876/images/WP-Unplanned-Downtime-US-042021.pdf, Accessed 1/11/2023.

Triska, Y., Sousa Agostino, Í. R., Penna, P. M., Braghirolli, L. F. and Frazzon, E. M., Integrated production and maintenance planning method with simulation-based optimization, *IFAC-PapersOnLine*, vol. 54, no. 1, pp. 349-354, 2021.

Wang, L., Lu, Z. and Han, X., Joint optimal production planning and proactive maintenance policy for a system subject to degradation, *Journal of Quality in Maintenance Engineering*, vol. 25, no. 2, pp. 236-252, 2019.

Wocker, M., Betz, N. K., Feuersänger, C., Lindworsky, A. and Deuse, J., Unsupervised Learning for Opportunistic Maintenance Optimization in Flexible Manufacturing Systems, *Procedia CIRP*, vol. 93, pp. 1025-1030, 2020.

Yeardley, A. S., Ejeh, J. O., Allen, L., Brown, S. F. and Cordiner, J., Integrating machine learning techniques into optimal maintenance scheduling, *Computers & Chemical Engineering*, vol. 166, 2022.

Zahedi, Z., Salim, A., Yusriski, R. and Haris, H., Optimization of an integrated batch production and maintenance scheduling on flow shop with two machines, *International Journal of Industrial Engineering Computations*, pp. 225-238, 2019.

Zhai, S., Gehring, B. and Reinhart, G., Enabling predictive maintenance integrated production scheduling by operation-specific health prognostics with generative deep learning, *Journal of Manufacturing Systems*, vol. 61, pp. 830-855, 2021.
Zhong, H.-S., Wang, H., Deng, Y.-H., Chen, M.-C., Peng, L.-C., Luo, Y.-H. et al., Quantum computational advantage using photons, *Science*, vol. 370, no. 6523, pp. 1460-1463, 2020.

Biographies

Marius R. Bernreuther is a Master´s student at the ESB Business School of Reutlingen University, pursuing a degree in Digital Industrial Management and Engineering and concurrently, Marius R. Bernreuther is also pursuing Engineering Management at Stellenbosch University.

Dominik Lucke is Professor of Production Technology, Automation and Digitalization of Production at the ESB Business School of Reutlingen University and Project Manager at the Fraunhofer Institute for Manufacturing Engineering and Automation IPA in Stuttgart in the department "Sustainable Production and Quality" since 2017. His main research topics are the development of smart maintenance applications and optimization of maintenance.

Johannes L. Jooste is a Professional Engineer and holds a PhD in Industrial Engineering from Stellenbosch University. He is currently a senior lecturer in the Department of Industrial Engineering and focusses on asset- and asset management related research.