

11th CIRP Global Web Conference (CIRPe 2023)

An optimization approach for a milling dynamics simulation based on Quantum Computing

S. Schröder^{a*}, S. Danz^{b,c}, P. Kienast^a, V. König^d, P. Ganser^a, T. Bergs^{a,c}

^aFraunhofer Institute for Production Technology IPT, Steinbachstraße 17, Aachen 52074, Germany

^bForschungszentrum Jülich, Wilhelm-Johnen-Straße, Jülich 52428, Germany

^cTheoretical Physics, Saarland University, Saarbrücken 66123, Germany

^dModuleWorks GmbH, Henricistraße 50, Aachen 52072, Germany

^eLaboratory for Machine Tools and Production Engineering (WZL) of RWTH Aachen University, Campus-Boulevard 30, Aachen 52074, Germany

*Corresponding author: Stefan Schröder, Tel.: +49-241-8904-488, fax: +49-241-8904-6488, E-mail address: stefan.schroeder@ipt.fraunhofer.de

Abstract

Since the machining of complex aerospace components, like integral compressor-rotors (blade integrated disks), is very cost-intensive, optimizing the process by means of process simulations is an active field of research. With the rise of Quantum Computing, a new instrument with high optimization potential is moving into focus. In this paper, a possible application of Quantum Computing for the machining simulation of multi-axis milling of thin-walled aerospace components is discussed. For this reason, a simulation framework used for the milling simulation is analyzed and each component is evaluated separately in relation to Quantum Computing. Parts of the Harrow, Hassidim, and Lloyd algorithm are proposed to enhance the Finite-Element simulation-based component, like the modal analysis for dynamics simulation. This algorithm can solve linear system problems with exponential speed-up over the classical method. The paper presents a roadmap on how the classical steps of a modal analysis for dynamics simulation could be replaced by quantum algorithms based on quantum phase estimation. The implementation of the first working steps is presented to validate this approach. The linear system problem, arising from the dynamics simulation, is analyzed in detail and a minimal value problem of linear coupled oscillators is derived.

© 2023 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0>)

Peer-review under responsibility of the scientific committee of the 11th CIRP Global Web Conference

Keywords: milling; FEM; modal analysis; optimization; quantum computing; quantum phase estimation; Harrow, Hassidim, and Lloyd algorithm

1. Introduction

In manufacturing applications, the potential of models and simulations for optimization based on digital twins (DT) is not fully exploited due to their computational requirements and the expert knowledge needed to operate them. Therefore, relevant physical effects in industrial practice are either neglected or only approximated by rough estimates. As a result, the quality of the DT and the insights and decisions derived from this suffer significantly, which leads in many cases to significant economic disadvantages. [1] In this context, the Fraunhofer IPT has been developing a domain-specific DT framework for machining, called dPart[®], for several years. It comprises data acquisition, processing, and analysis methods embedded in an Industrial-Internet-of-Things (IIoT) infrastructure (Fraunhofer

Edge Cloud) for the targeted analysis and optimization of machining processes. [2] The following investigated simulation is embedded in the mentioned DT framework. With the advent of Quantum Computing (QC), a new tool with high optimization potential comes into focus. Mathematically it was shown that simulations with given accuracy allow polynomial and in certain cases even exponential acceleration (based on the Harrow, Hassidim, and Lloyd (HHL) algorithm) [3]. A potential application for QC in manufacturing simulation is presented in the machining simulation of multi-axis milling of thin-walled aerospace components like integral compressor-rotors (blade integrated disks (blisks)). Blisks represent one of the most challenging components in turbomachinery manufacturing [4]. Due to the high aspect ratio of the blade geometry resulting in low workpiece rigidity, various effects such as deflection and chatter can occur during the machining process [5]. If vibrations occur during machining, it is often difficult for the operator to locate the cause. To avoid vibration issues, very conservative

process parameters, such as reduced feed, cutting speed, cutting depth, and cutting width, are often used. The selection of an advantageous spindle speed for the present machining case is established based on stability diagrams. For the prediction of the process stability and thus for the generation of the stability diagrams, information about cutting force, dynamics of the total system (superposition of tool and workpiece dynamics), process parameters, and tool geometry is required. [6]

In this paper, the workflow of the coupled technology models used in process simulation is presented in section 2. Then, two possible applications of QC out of the simulation workflow in subsection 2.1 and 2.2 are extracted. In section 3, the approach for enhancing the dynamics simulation of multi-axis milling of thin-walled aerospace components with quantum algorithms is described on the basis of linear coupled oscillators in detail. Moreover, in section 4 and 5 the quantum emulation as proof of concept is summarized so far and reveals the roadmap with the next steps to apply in the process design and optimization for an industrial thin-walled aerospace component.

2. Milling dynamics simulation workflow

The FEM simulation workflow (Figure 1), originally published by Rudel et. al., is briefly summarized below for a better understanding. It shows the detailed sequence of inputs, methods, and variables involved in calculating the virtual representation of the dynamic process behavior. The user programs the toolpath for the milling operation in the first step. Necessary inputs for the dixel-based engagement simulations are the NC path, the stock geometry of the workpiece, and the tool geometry such as diameter D , number of teeth N_{th} , and helix angle λ . The NC path contains all relevant cutting parameters like axial stepover a_p , radial stepover a_e , feed per tooth f_z , and spindle speed n . It is also necessary to select discrete Cutter Locations (CLs) for which the Finite-Element (FE) modal analysis should be performed in the following steps. Since a single toolpath consists of several thousand points, the simulation of all CLs of the toolpath is not realistic in terms of computational time and intensity. [6] In order to calculate the engagement angles between tool and workpiece on the one hand, and to extract CL-dependent In-Process-Workpieces (IPWs) on the other hand, a dixel-based engagement simulation was then executed. Moreover, the user needs to define material properties, e.g. density ρ ; Young's Modulus E ; Poisson's ratio ν , and boundary conditions, e.g. constraints or friction; in order to run FE modal analysis and obtain CL-dependent Frequency Response Functions (FRFs). In addition to the number of selected CLs, for the FE modal analysis of the selected CL resolution, the computational effort also scales exponentially with the number of FE mesh nodes, the selected node resolution, and model size [7, 8]. Additionally, it is also decisive how many of the relevant eigenmodes are considered during the FE modal analysis. In the presented case, the simulation was limited to the first six eigenmodes [9]. Furthermore, specific cutting force coefficients need to be defined to execute a dual-mechanistic cutting force simulation and to obtain the process-related cutting force F_c .

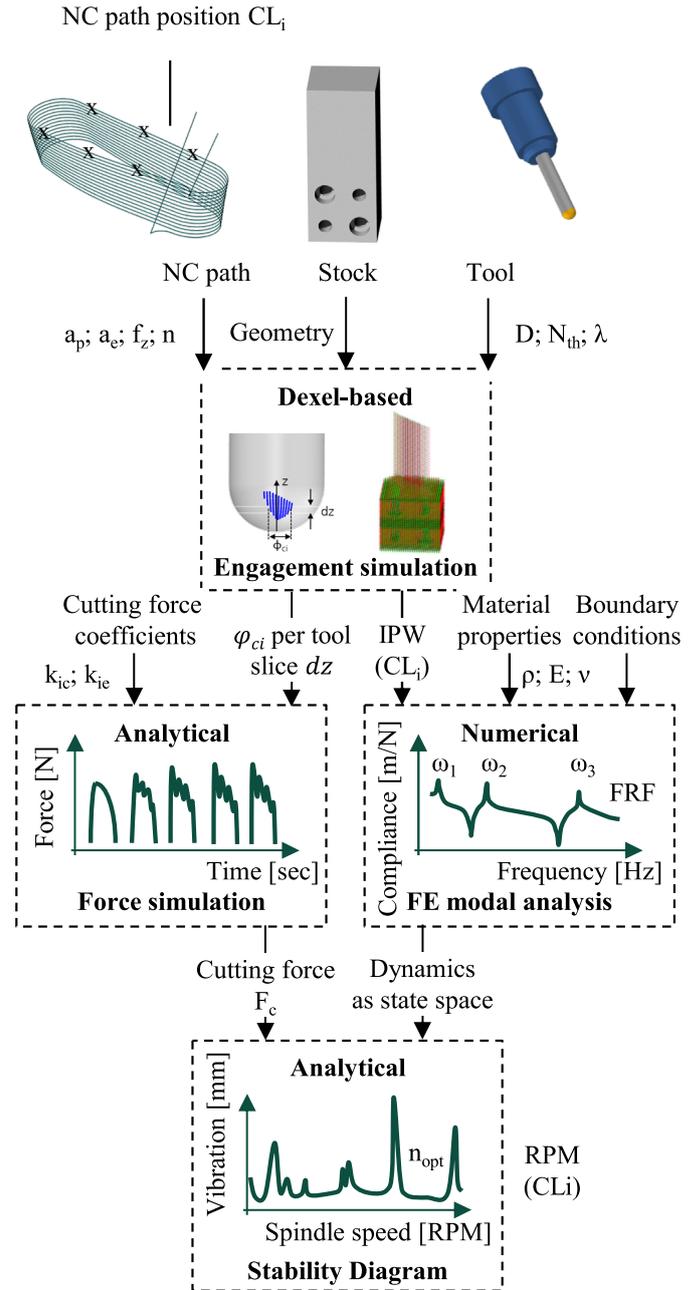


Fig. 1. FEM simulation workflow based on Rudel et. al. [6]

Knowing the position-dependent cutting force and FRFs, stability diagrams can be generated to select suitable spindle speeds to avoid vibrations during the milling process. [6, 9, 10]

The FEM simulations workflow (Figure 1) includes analytical and numerical models. Determination of the relative time required for each step, for a finishing operation at a simplified blade geometry (Figure 2 (a)) showed that the analytical models were negligible in terms of their computing time (see Table 1, computer specifications: Intel® Xeon® Gold-5118 2.30 GHz CPU, 24 cores, 380 GB RAM). Note that the computational time of simulation tasks related to the IPW, e.g. the FE workflow, highly scales with the defined number of cutter locations. As an example, selecting 200 CLs for one operation

Table 1. Measured relative time effort for each simulation step

Simulation steps	Relative time (%)
Dexel-based engagement simulation	77
Analytical force simulation	3
Numerical FE workflow	18
Analytical stability diagram calculation	2

results in a time effort of 4400 seconds. Both analytical methods, force and stability simulation in the milling dynamics simulation workflow, are less computationally extensive compared to the dexel-based engagement simulation and the FE workflow. This disadvantage of numerical simulation has been elaborated on in previous scientific works [11]. For the use case of finishing in blisk milling, the volume of the removed chip is very small. High resolution is required to calculate the engagement precisely leading to a very small dexel size, which negatively affects the computational time. However, with the help of suitable quantum algorithms (e.g. the HHL-Algorithm), it is assumed that the computation time will be exponentially accelerated compared to classical algorithms on classical computers [12]. Analogous behavior applies to the FE modal analysis. Due to the complexity of numerical methods, only discrete cutter locations are currently modeled and considered in the simulation of manufacturing processes. Both methods show the potential to bring advantages in the field of milling dynamics simulation by using quantum algorithms. The two methods are further discussed in the following subsections 2.1 and 2.2.

2.1. Dexelsimulation

Within the dPart[®] framework, the engagement simulation is a dexel-based simulation. The main components of a dexel model are a structured two-dimensional grid on a plane that serves as a nail coordinate system (1), nails, perpendicular to the coordinate system's plane located at the intersection points of the grid (2), and a subset defined by the intersection points between the nail and the volume that is represented by the nails (3). A three-dimensional volume is discretized using three orthogonal depth fields aligned with the axes of a cartesian coordinate system, an example is shown in Figure 2 (b). The engagement simulation provides the tool's contact area, contact length, and width. It also allows tracking removed volume and tool cutting direction to enable an evaluation of the workpiece cut by cut. The algorithm calculates all segments of any dexel (for all three grids) in the nail model that need to be changed. The intersections are performed individually for each of the three dexel grids to take advantage of memory closeness. The calculations performed by the intersector are based on floating-point calculations. As stated above, the engagement simulation was investigated for optimization potential. In the investigated use case, the order of magnitude between tool size and cutting precision is very different (a big tool and compared to a small nail model), leading to a vast amount of intersection calculation

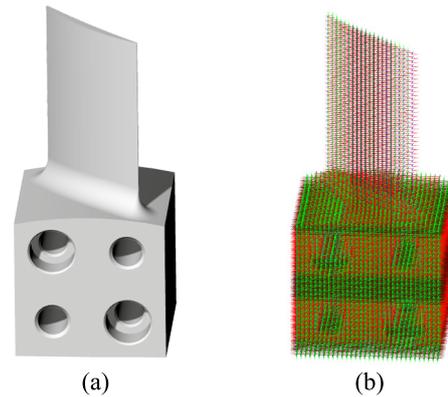


Fig. 2. Simplified blade geometry (a) and three-dimensional dexel model (b)

per cut. A speed-up was reached using parallelization concluding the bottleneck remaining in access operations and therefore out of scope for quantum-based optimization.

2.2. Classical modal analysis

The FE workflow is performed for each IPW in subsequent calculations. Starting after the dexel-based engagement simulation in CAM, the main steps are IPW conversion from dexel to solid, meshing process, and the execution of modal analysis. Considering the condition for quantum acceleration, the workflow was analyzed step by step both in terms of computational time and with regards to extract numerical problems that can be transformed in quantum algorithms. The conversion steps from the dexel representation to the solid geometry and the meshing process consumed the most time. However, the conversion and meshing processes are highly affected by internal software algorithms and were therefore not focused subsequently. In contrast, the step 'modal analysis', i.e. solving the eigenvalue problem, is a well-known mathematical numerical problem and therefore found to be suitable for computation with quantum algorithms. Within the classical step 'modal analysis', the equation of motion for a free and undamped system is consulted in the physical domain on a local simulation computer. Here, the mass matrix and the stiffness matrix are extracted for each respective IPW. With the equation of a harmonic oscillation and the second derivative of the harmonic oscillation, the equation of motion can be transformed into an eigenvalue problem. By knowing eigenvalues and eigenvectors (and damping ratio), the dynamics for each simulated IPW can be described in the state space representation, usually visualized as FRF. The results can be further utilized for the calculation of stability diagrams for optimization of the spindle speed to avoid process vibrations. In the future approach, the quantum-based modal analysis will be connected to a QC via a new microservice in the batch layer, within the Core Domain of the framework dPart[®] [6]. It is planned, that the calculated solutions can be sent back to the local simulation computer via the mentioned microservice. There, the results can be further utilized as described above for optimization of the spindle speed to avoid process vibrations.

3. Quantum based modal analysis

In the beginning, the calculation of a simplified blisk blade geometry was started. The selected geometry and simulation parameters are shown in Figure 2 (a) and Table 2. Despite the highly simplified geometry, the matrix size of 1005 by 1005 was not computable with existing quantum hardware. Therefore, the solution of the modal analysis and the eigenvalue problem using quantum algorithms was investigated within this paper based on a simplification approach of linear coupled oscillators. The corresponding matrix was of size $2^N \times 2^N$, where N is the number of oscillators. However, this approach can be scaled up again as soon as the appropriate quantum hardware is available.

3.1. Linear coupled oscillators

The system of linear coupled oscillators (Fig. 3) was studied with the aim of simulating its behavior under the influence of an external force $F = (f_0, \dots, f_{N-1})^T$. This is described by its equation of motion (Eq. 1), where $X = (x_0, \dots, x_{N-1})^T$ contains the displacement x_i of the N oscillators, M is the mass matrix, K the stiffness matrix, and C the damping.

$$M\ddot{X}(t) + KX(t) + C\dot{X}(t) = F(t) \quad (1)$$

This equation can be solved by switching into frequency space using $X(t) = \int X(\omega) \exp(i\omega t) d\omega$, thus resulting in

$$X(\omega) = \chi(\omega)F(\omega), \quad (2)$$

where the frequency dependent response function $\chi(\omega)$ between the amplitude of the applied force $F(\omega)$ and the displacement $X(\omega)$ is introduced

$$\chi(\omega) = \sum_k \frac{M^{-\frac{1}{2}} V_k V_k^\dagger M^{-\frac{1}{2}}}{-\omega^2 + \omega_k^2 + i\omega(\alpha\omega_k^2 + \beta)}, \quad (3)$$

with the eigenfrequency ω_k and the eigenvector V_k for the matrix $M^{-\frac{1}{2}} K M^{-\frac{1}{2}}$. The damping is represented by the two coefficients α and β , based on Rayleigh Damping and depending on experimental measurements [13]. A full description of the response function depends on the accurate computation of the eigenfrequencies and eigenvectors. Those can be computed using classical algorithms within polynomial time ($O(N^3)$ with the size of the Matrix N [14]). The big drawback of those algorithms is, that they have to deconstruct an equation of vectors into their components and solve those one at a time. A quantum system can be described directly with linear algebra, which makes it optimal for linear algebra computations

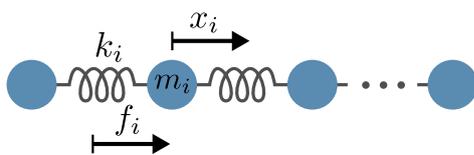


Fig. 3. Linear coupled oscillators as a toy model. The system is fully described by the displacement x_i of the i -th oscillator, its external forces f_i , its mass m_i and the stiffnesses $\{k_i, k_{i+1}\}$ of the surrounding connections.

Table 2. Geometry and simulation parameters

Parameter	Value
Blade height h	60 mm
Chord length l	45 mm
Blade twist φ	0 deg
Blade profile	DCA
Material	Ti-6Al-4V
Damping	0.2 %
Mesh size	0.8 mm
Number of Modes	20
Total nodes	536
Matrix size	1005x1005

in which quantum states represent vectors and quantum operations or gates represent matrices. However, efficient methods are needed to prepare those quantum states and gates without losing any speed-up. One example of quantum algorithms that suits the problem described above is the quantum phase estimation (QPE), which can be used to estimate eigenvalues and eigenvectors and promise exponential speed-up in comparison to classical algorithms [15, 16]. This speed-up is only given for a certain family of problems, as it depends on assumptions. This exact algorithm is further described in subsection 3.2.

3.2. Quantum phase estimation

QPE is a central quantum algorithm that was introduced by Kitaev [15]. Its main function is to find the eigenvalue of a unitary operator U , which finds use in many physics [17] and chemistry [18] problems as well as in Shor's algorithm [19]. In the standard setting for QPE, an eigenstate $|\Psi\rangle$ of the unitary U is given and, in addition, the unitary U is given in the form of a quantum gate. More precisely, the goal is to estimate the phase $\varphi \in [0, 1)$ of the eigenvalue $e^{2\pi i\varphi}$ associated with $|\Psi\rangle$. For a classical algorithm, this task scales exponentially with respect to the number of qubits necessary to represent $U|\Psi\rangle$.

It is further possible to use QPE to determine the eigenvalues of a hermitian matrix H if there exists a quantum circuit for $U = e^{2\pi i H}$. This is not always possible without losing all quantum speed-up. However, QPE can be used without losing the quantum advantage as long as the matrix can be decomposed in a linear superposition of elementary quantum gates [20] or if the rule of how to compute the entries of the matrix is known

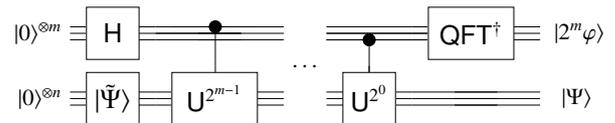


Fig. 4. The QPE starts with the preparation of the eigenstate $|\tilde{\Psi}\rangle$ in the lower register. This is followed by a series of controlled U , which prepares the desired phase φ . The inverse QFT probes for any possible phase, such that the eigenvalues φ and true eigenstates $|\Psi\rangle$ are stored with high probability in the two quantum registers.

[21]. The central element of the original version is the quantum Fourier transform (QFT) [22, 23, 24, 25] as shown in Fig. 4. For a detailed description of the QPE algorithm, see the studies of Cleve et al. (1998), Nielsen et al. (2010), and Scarani (2012) [16, 25, 26].

In its original version, QPE needs a register of m ancilla qubits in which it stores the phase with high probability. This phase can only be exact up to a finite error $\varepsilon < 2^{-(m+1)}$ due to its binary representation [25]. The probability distribution in the ancilla register is peaked around the desired phase φ and its width increases with the error ε . The minimum probability to measure the correct phase never falls below $4/\pi^2$ [16]. The time efficiency of the algorithm depends mostly on the number of queries of U , which scales as $O(\varepsilon^{-1})$ while the number of additional operators in the QFT scales as $O((\log(1/\varepsilon))^2)$. Hence, the algorithm scales inversely proportional with respect to the error ε . Notice that this is the scaling obtained by assuming that the unitary U^k is implemented by applying the unitary U k times. The scaling further reduces if there exists a quantum circuit for the direct implementation of U^k which scales similarly, in the number of gates, as U . This makes it an algorithm that scales in the number of U^k calls logarithmically with the inverse of the error $O(\log(1/\varepsilon))$ and hence, the QFT dominates the computation time [16].

A basic assumption in QPE is that one is capable of implementing the unitary operator U and preparing a state $|\tilde{\Psi}\rangle$, which overlaps significantly with the eigenstate $|\Psi\rangle$, within polynomial time on a quantum device. As already mentioned, this is not always the case for arbitrary $U = e^{iH}$. However, if H can be decomposed into a superposition of elementary gates, one can use a truncated Taylor Series expansion to bypass the exponential function [20]. Another case in which the implementation of the unitary U is efficient is when the matrix H has sparsity properties and its entries can be efficiently evaluated [27]. The preparation of the eigenstate $|\Psi\rangle$ is also a non-trivial task that needs to be accomplished efficiently in the number of quantum gates to preserve the speed-up of the algorithm. Methods to prepare an approximate eigenstate efficiently are known [28, 29]. Here, a good approximation of $|\tilde{\Psi}\rangle$ is sufficient due to the probabilistic nature of the algorithm, which suppresses small errors. The desired eigenvalues φ are hidden in the ancilla register as superposition (see Fig. 4). Extracting them is a non-trivial problem, as a measurement destroys the superposition and yields only one eigenvalue with a certain probability. QPE amplifies the probability of measuring the true eigenvalues, which can be picked out by repeating the whole process multiple times. The corresponding true eigenvectors $|\Psi\rangle$ are in the bottom register (see Fig. 4) and can be further used in other algorithms such as the Hadamard test [30] or extracted by measuring it.

4. Quantum emulation as proof of concept

As proof of concept, we emulated the QPE for a system of $N = 4$ linear coupled oscillators of equal mass $m = 4$ kg, which are connected by springs with stiffness $k = 1$ N m⁻¹, on a classical computer. The resulting FRF is shown in Fig. 5 for an

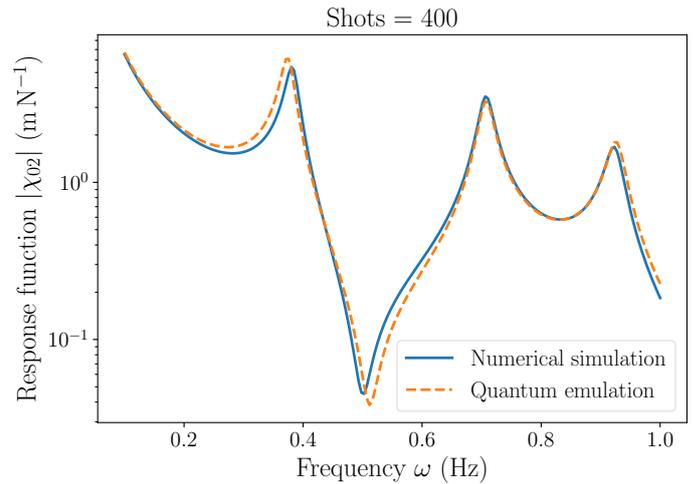


Fig. 5. Numerically simulated and quantum emulated frequency response function for a system of 4 linear coupled oscillators with mass $m = 4$ kg and stiffness $k = 1$ N m⁻¹ as proof of concept.

arbitrarily chosen pair of input and output nodes (2 and 0). In here, the damping was neglected. The blue graph represents the result of the numerically solved eigenvalues, eigenvectors of the matrices and vectors, and their FRF. Conversely, the orange one shows the previous emulation of the FRF based on the presented circuit in Qiskit on a classical computer. The accuracy of the quantum emulation depends directly on the number of qubits and with it on the computation-time of the algorithm. To classify the readiness of the individual simulation steps, it should be noted that the steps of state preparation, gate preparation, and data extraction are implemented, but work inefficiently in terms of computation time. An improved implementation is currently being worked on.

5. Summary & Outlook

The presented paper introduces an optimization approach for a dynamics milling simulation based on QC algorithm. The FEM simulation workflow of the dynamics simulation of multi-axis milling of thin-walled aerospace components, like integral compressor rotors, was presented, and individual steps of it were evaluated for quantum readiness. The FE modal analysis was selected as a suitable use case. A scalable quantum algorithm utilizing QPE was presented based on the minimum value problem of linear coupled oscillators. Nevertheless, the state-of-the-art quantum devices, so-called noisy intermediate scale quantum (NISQ) devices, are characterized by non-trivial qubit numbers (~ 100), but also by relatively high error rates between 10^{-2} and 10^{-3} in the implementation of gates and measurements [31]. This limits the depth of the quantum circuit that can be simulated on current quantum hardware without having a large error on the output to a few thousand gates [32]. QPE generally requires high-depth circuits for exploiting its full potential, and as such it is not thought to be an algorithm that is suited for NISQ devices, but it requires a full fault-tolerant QC [33]. However, further developed algorithms can be used to es-

time the phase even on NISQ technology using the evolution operator for various times [34, 35].

Continuous further development of the presented quantum algorithm will take place. The main emphasis will be on troubleshooting and debugging specific parts of the algorithm. Additionally, efforts will be directed toward scaling up the minimum value problem of linear coupled oscillators to a three-dimensional approach. The ultimate goal is to achieve a practical industrial use case, such as the milling dynamics simulation of integral compressor rotors like a blisk blade. Moreover, QC could support more simulations in the future in the manufacturing industry. For this purpose, a backend, to fill the interface between classical simulation and QC, will be developed in the dPart® framework.

Acknowledgements

This paper has received funding from the 'QUASIM' (01MQ22001A) research, a program of the German Federal Ministry of Economics and Climate Protection (BMWK).

References

- [1] T. Bergs, S. Gierlings, T. Auerbach, A. Klink, D. Schraknepper, T. Augspurger, The concept of digital twin and digital shadow in manufacturing, *Procedia CIRP* 101, p. 81-84, 9th CIRP Conference on High Performance Cutting (2021). doi:10.1016/j.procir.2021.02.010.
- [2] P. Ganser, T. Venek, V. Rudel, T. Bergs, Dpart – A Digital Twin Framework for the Machining Domain, *MM Sci. J.*, p. 5134–5141 (2021). doi:10.17973/MMSJ.2021_11_2021168.
- [3] A. Montanaro, S. Pallister, Quantum algorithms and the finite element method, *Phys. Rev. A* 93, 032324 (Mar 2016). doi:10.1103/PhysRevA.93.032324.
- [4] C. Hubig, Fertigungsgerechtes Konstruieren von Hochdruckverdichterschaukeln in Blisk-Bauweise, Schriftenreihe des PTW: "Innovation Fertigungstechnik", Shaker Verlag, Aachen, 2012.
- [5] D. Liu, M. Luo, G. Urbikain Pelayo, D. O. Trejo, D. Zhang, Position-oriented process monitoring in milling of thin-walled parts, *Journal of Manufacturing Systems* 60, p. 360-372 (2021). doi:10.1016/j.jmsy.2021.06.010.
- [6] V. Rudel, P. Kienast, G. Vinogradov, P. Ganser, T. Bergs, Cloudbased process design in a digital twin framework with integrated and coupled technology models for blisk milling, *Front. Manuf. Technol.* 2, 1021029 (2022). doi:10.3389/fmtec.2022.1021029.
- [7] P. Ganser, Quatencomputing - Potenziale für die Fertigungssimulation in der Zerspanung, Aachen Conference on Machining ACM (21.-22.09.2022).
- [8] F. Klocke, *Fertigungsverfahren 1. Zerspanung mit geometrisch bestimmter Schneide Reihe: VDI-Buch. 9. Aufl.* Berlin, Heidelberg: Springer-Verlag (2018).
- [9] S. Maslo, B. Menezes, P. Kienast, P. Ganser, T. Bergs, Improving dynamic process stability in milling of thin-walled workpieces by optimization of spindle speed based on a linear parameter-varying model, *Procedia CIRP* 93, p. 850-855, 53rd CIRP Conference on Manufacturing Systems 2020 (2020). doi:10.1016/j.procir.2020.03.092.
- [10] S. Maslo, Simulation of the dynamic vibration behaviour and spindle speed optimization during the milling process of turbomachinery components. Apprimus Verlag (2022).
- [11] P. Arrazola, T. Özel, D. Umbrello, M. Davies, I. Jawahir, Recent advances in modelling of metal machining processes, *CIRP Annals* 62 (2013). doi:10.1016/j.cirp.2013.05.006.
- [12] A. W. Harrow, A. Hassidim, S. Lloyd, Quantum algorithm for linear systems of equations, *Phys. Rev. Lett.* 103 (Oct 2009). doi:10.1103/PhysRevLett.103.150502.
- [13] L. Nasdala, *FEM-Formelsammlung Statik und Dynamik*, 1. Aufl. Wiesbaden, Vieweg + Teubner Verlag (2010). doi:10.1007/978-3-8348-9922-4.
- [14] A. Krishnamoorthy, D. Menon, Matrix Inversion Using Cholesky Decomposition (Oct. 2013). doi:10.48550/arXiv.1111.4144.
- [15] A. Y. Kitaev, Quantum measurements and the abelian stabilizer problem (1995). doi:10.48550/ARXIV.QUANT-PH/9511026.
- [16] R. Cleve, A. Ekert, C. Macchiavello, M. Mosca, Quantum algorithms revisited, *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 454, p. 339-354 (1998).
- [17] D. S. Abrams, S. Lloyd, Quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors, *Phys. Rev. Lett.* 83, p. 5162–5165 (Dec 1999). doi:10.1103/PhysRevLett.83.5162.
- [18] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, M. Head-Gordon, Simulated Quantum Computation of Molecular Energies, *Science* 309, p. 1704-1707 (2005). doi:10.1126/science.1113479.
- [19] P. W. Shor, Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, *SIAM Journal on Computing* 26, p. 1484-1509 (1997). doi:10.1137/S0097539795293172.
- [20] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, R. D. Somma, Simulating Hamiltonian Dynamics with a Truncated Taylor Series, *Phys. Rev. Lett.* 114, 090502 (Mar 2015). doi:10.1103/PhysRevLett.114.090502.
- [21] A. M. Childs, On the Relationship Between Continuous- and Discrete-Time Quantum Walk, *Communications in Mathematical Physics* 294, p. 581-603 (Mar 2010). doi:10.1007/s00220-009-0930-1.
- [22] D. Deutsch, R. Penrose, Quantum theory, the church – Turing principle and the universal quantum computer, *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 400, p. 97-117 (1985). doi:10.1098/rspa.1985.0070.
- [23] D. Coppersmith, An approximate Fourier transform useful in quantum factoring, IBM Research Report RC19642, 1994. arXiv:quant-ph/0201067.
- [24] P. Shor, Algorithms for quantum computation: discrete logarithms and factoring, in: *Proceedings 35th Annual Symposium on Foundations of Computer Science*, p. 124-134, 1994. doi:10.1109/SFCS.1994.365700.
- [25] M. A. Nielsen, I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*, Cambridge University Press, 2010. doi:10.1017/CB09780511976667.
- [26] V. Scarani, Quantum Computing: A Gentle Introduction, *Physics Today* 65, p. 53-55 (2012). doi:10.1063/PT.3.1442.
- [27] D. W. Berry, G. Ahokas, R. Cleve, B. C. Sanders, Efficient Quantum Algorithms for Simulating Sparse Hamiltonians, *Commun. Math. Phys.* 270, p. 359-371 (Mar 2007). doi:10.1007/s00220-006-0150-x.
- [28] L. Grover, T. Rudolph, Creating superpositions that correspond to efficiently integrable probability distributions (2002). doi:10.48550/ARXIV.QUANT-PH/0208112.
- [29] A. G. Rattew, B. Koczor, Preparing Arbitrary Continuous Functions in Quantum Registers With Logarithmic Complexity (2022). doi:10.48550/ARXIV.2205.00519.
- [30] D. Aharonov, V. Jones, Z. Landau, A Polynomial Quantum Algorithm for Approximating the Jones Polynomial, *Algorithmica* 55, p. 395-421 (Nov 2009). doi:10.1007/s00453-008-9168-0.
- [31] R. Barends, J. Kelly, A. Megrant, et al., Superconducting quantum circuits at the surface code threshold for fault tolerance, *Nature* 508, p. 500-503 (Apr 2014). doi:10.1038/nature13171.
- [32] F. Arute, K. Arya, R. Babbush, et al., Quantum supremacy using a programmable superconducting processor, *Nature* 574, p. 505-510 (Oct 2019). doi:10.1038/s41586-019-1666-5.
- [33] C. Gidney, M. Ekerå, How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits, *Quantum* 5, p. 433 (apr 2021). doi:10.22331/q-2021-04-15-433.
- [34] R. D. Somma, Quantum eigenvalue estimation via time series analysis (2019). doi:10.48550/ARXIV.1907.11748.
- [35] L. Clinton, J. Bausch, J. Klassen, T. Cubitt, Phase Estimation of Local Hamiltonians on NISQ Hardware (2021). doi:10.48550/ARXIV.2110.13584.