Quantum Computation in Robotic Science and Applications

Christina Petschnigg, Mathias Brandstötter*, Horst Pichler, Michael Hofbaur, and Bernhard Dieber

***Abstract*— Using the effects of quantum mechanics for computing challenges has been an often discussed topic for decades. The frequent successes and early products in this area, which we have seen in recent years, indicate that we are currently entering a new era of computing. This paradigm shift will also impact the work of robotic scientists and the applications of robotics. New possibilities as well as new approaches to known problems will enable the creation of even more powerful and intelligent robots that make use of quantum computing cloud services or co-processors. In this position paper, we discuss potential application areas and also point out open research topics in quantum computing for robotics. We go into detail on the impact of quantum computing in artificial intelligence and machine learning, sensing and perception, kinematics as well as system diagnosis. For each topic we point out where quantum computing could be applied based on results from current research.**

I. Introduction

Much has been said about quantum computing for decades. From the initial postulations [1], [2], ever new results have brought this idea closer to realization. With the recent successes and the worldwide significant investments in technology development, e.g., the European FET Flagship on Quantum Technologies and the National Laboratory for Quantum Information Sciences in China, we find ourselves at the edge of a new age of computing [3]. This revolution will also impact the field of robotic science and its applications. Many areas of robotics pose challenges that require intensive computation where today, we typically use GPGPUs (general-purpose GPU) to offload expensive tasks. With quantum computing techniques, new approaches to solve those challenges but also new fields of research are on the horizon. While quantum computers are in theory capable of performing all kinds of calculations, it is not to assume that there will be computers (or robots for that matter) that are entirely quantum-powered even tough quantum robots have been proposed in literature [4], [5]. Instead, there will be quantum computing cloud services initially and potentially quantum co-processors (QPUs) that work together with classical CPUs.

In this position paper, we want to point out possible fields of application for quantum computing in robotic research and robotic applications. We are convinced that many areas within robotics can benefit from this promising new technology. We shortly summarize the basics of quantum computing as well as the quantum computing application to general problems in Section II. Then, in Section III, we present fields from the robotic science where the application of quantum computing algorithms is promising to achieve significant improvements. In Section IV, we present open research questions that regard the application of quantum computing to robotics.

All authors are with ROBOTICS – Institute for Robotics and Mechatronics, JOANNEUM RESEARCH Forschungsgesellschaft mbH, Lakeside B13b, 9020 Klagenfurt am Wörthersee, Austria `firstname.lastname@joanneum.at`

*Corresponding author

II. Background

A quantum computer uses quantum bits – in short qubits – that have several interesting properties. In contrast to classical bits, qubits can be in multiple states at the same time (i.e., $|0\rangle$ and $|1\rangle$) before being measured, which is referred to as superposition. This superposition state can be represented by a linear combination of the ground states and is irreversibly disturbed during measurement. Additionally, two qubits are able to influence each others states without having a physical connection, which is called entanglement. These are particles that are generated in a way such that one cannot be fully described without the other. A system of such entangled qubits can be described, however, as a superposition. While a large number of classical states can be represented simultaneously by a single superposition state, the computational power grows exponentially with the number of qubits through entanglement. Therefore, adding qubits to a quantum computer can exponentially increase its computing power [6]. A major advantage of quantum computers is their ability to solve some computationally intensive mathematical problems at all, more efficiently or exactly compared to classical computers [7]. However, the complexity of algorithm formulation in order to be executed on a quantum computer and the number of error corrected qubits necessary is still an obstacle. In addition to the properties mentioned above, the fragility of qubits should be noted. Any interaction, such as measuring, observing, disturbing, with a qubit, which represents a two-state system, leads to a reliably distinguishable state. However, this apparent disadvantage can be exploited specifically for applications.

A. Application areas

There is a high probability that quantum computers will have a positive impact on many scientific disciplines and their applications in future. In several fields, useful applications are already being explored, including scientific computing [3], cryptography [8], [9], chemistry [10], [11], drug development [12], and finance [13]. In addition, there are approaches to tackle general problems like solving linear systems of equations [14], linear differential equations [15] or finding discrete logarithms and factoring integers [16].

TABLE I
QUANTUM ACCELERATION FOR VARIOUS ALGORITHMS

Method	Acceleration	HHL	QRAM
Bayesian Inference [21], [22]	$O(\sqrt{n})$	yes	no
Online Perceptron [23]	$O(\sqrt{n})$	no	optional
Least Squares Estimation [24]	$O(\log(n))$	yes	yes
Quantum Principal Component Analysis [25]	$O(\log(n))$	yes	optional
Quantum Support Vector Machine [26]	$O(\log(n))$	yes	yes
Quantum Reinforcement Learning [27]	$O(\sqrt{n})$	no	no

B. Quantum algorithms and quantum acceleration

Quantum algorithms substantially outperform their classical counterparts for several problems like searching in an unsorted list. This circumstance is called quantum acceleration. However, the great benefit of quantum computers is not the discovery of a solution for problems that have already been solved in a practical way. More interesting are exponentially growing problems where classical computers fail and, moreover, those that can be directly described by the combination of quantum gates [17].

Prominent and widely used quantum algorithms include quantum basic linear algebra subroutines (qBLAS) that are used to calculate Fourier transforms, to find eigenvalues and the corresponding eigenvectors as well as to solve linear equation systems [18]. Another popular quantum algorithm is the Grover's algorithm that searches for entries in an unsorted database of size n in $O(\sqrt{n})$ steps, which is provably faster than any classical algorithm [19]. The eigenvalues and eigenvectors of an $n \times n$ matrix can be found in $O(\log(n))$ steps by quantum algorithms. The Harrow-Hassidim-Lloyd (HHL) quantum algorithm is used for solving linear systems of equations and is exponentially faster than classical algorithms tackling the same problem [20]. Table I, which was partly adopted from [18], describes the acceleration achieved by quantum algorithms over their classical counterparts. The table item $O(\sqrt{n})$ depicts quadratic acceleration. Similarly $O(\log(n))$ describes exponential acceleration compared to classical algorithms. The column labeled by HHL provides information on whether the HHL algorithm is applied. Quantum Random Access Memory (QRAM) is used in some algorithms to translate data vectors into quantum states. The dedicated column indicates the need for QRAM to execute the respective algorithm. The acceleration of these techniques is based on different quantum algorithms like Grover's algorithm, HHL or qBLAS.

C. Optimization problems

In robotics research, optimization problems are omnipresent. Application areas include trajectory planning, robot vision, task allocation and kinematics. The goal of a one dimensional optimization problem is to find a best element with regard to a given cost function and a problem definition. The solution space of an optimization problem generally consists of several feasible solutions for the basic problem, however, the aim is to find a best solution, which is unique in the convex case. The solution effort increases linearly, polynomially or exponentially depending on the nature of the problem and the solution algorithm. Yet, the classical approach to solve problems with exponentially increasing complexity quickly reaches its limits.

Optimization problems can alternatively be formulated as search problems. In order to solve general or specific search problems, there exist different quantum algorithms like the Grover's algorithm, quantum annealing or the Shor algorithm. Quantum annealing is a heuristic to find a global minimum of an objective function or the ground state of a system and is mainly used for problems with discrete search spaces where a high number of minima is assumed [28]. The analogy to classical random walks in quantum computing is the quantum random walk. It is used to design randomised quantum algorithms and to speed up several problem classes, e.g. to determine whether a graph is triangle free [29]. The Shor algorithm accelerates factoring integers by efficiently finding discrete logarithms, which lies at the base of several cryptosystems and is considered a hard problem for classical computers [30].

Multi-objective optimization is another form of mathematical optimization that involves several objective functions, which are optimized simultaneously. The most common multi-objective optimization technique is Pareto optimality [31]. Similar to one dimensional optimization problems, it is possible to find a formulation as search problem. Thus, there lies huge potential in considering quantum algorithms for solving multi-objective optimization. A model for Pareto optimization of composite materials based on quantum behaved particle optimization is presented in [32].

D. Machine Learning

Modern robots make vital use of machine learning techniques for various tasks like the analysis and mining of sensor data used for perception, robot localization, learning controllers and planners as well as learning human robot interaction [33]. Application algorithms range from quantum principal component analysis (PCA) over quantum regression to quantum artificial neural networks [33]. In order to construct models for machine learning applications, a dimensionality reduction step is often applied. In the quantum world, quantum PCA is used to this end and provides exponential speedup compared to the classical algorithm [25].

Generally, learning tasks can be grouped into three major categories, namely supervised, unsupervised and reinforcement learning. Quantum computing techniques can be used to speed up several supervised learning approaches. Curve fitting or regression, which are supervised learning strategies, can be accelerated exponentially using the HHL algorithm for solving linear systems of equations [20]. Different variants of Quantum Support Vector Machines (SVM) have been introduced, some of which outperform classical methods exponentially [33]. Grover's algorithm in conjunction with a special oracle function may be used for quantum cluster analysis [34], which is an unsupervised learning method.

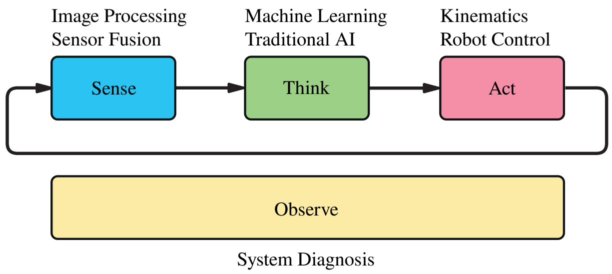

Fig. 1. Application of quantum computing in the sense-think-act cycle.

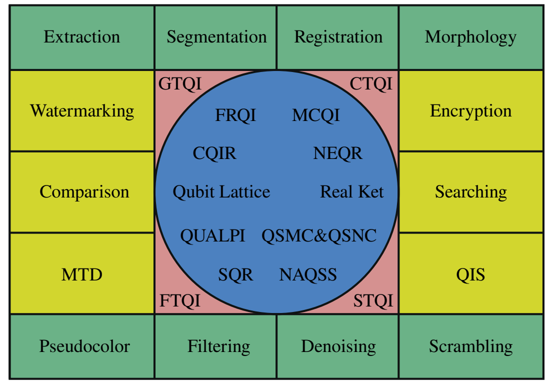

Fig. 2. Quantum image representations (blue) and their transformations (red), algorithms (green), and applications (yellow) [38].

Quantum reinforcement learning can be realized using the properties of state superposition principle and quantum parallelism [35].

The potential impact of quantum machine learning on robotics is vast. The above mentioned methods are widely applied in robotics and can be significantly accelerated by using quantum principles. Classical PCA is used in various robotic applications like environment modelling and simultaneous localization and mapping. SVMs are frequently used for problems such as object recognition or data fusion. Clustering methods in contrast are used among other things to understand image and trajectory data.

III. Possible applications in robotics

In this section, we want to go into more detail on the research areas in robotics where we assume the biggest impact of quantum computing. We will structure our remarks according to the classical sense-think-act cycle extended by an overall observe action that represents system diagnosis (see Fig. 1).

A. Sense: Perception, Vision, and Sensor Data Processing

Modern autonomous robots require fast vision capabilities in order to perceive and assess their environment. Computer vision and image processing algorithms are computationally expensive as they have to compute results on millions of pixels. Naturally, the hopes to better understand the nature of visual information, and to secure, store and process them efficiently with the help of quantum properties, like entanglement and parallelism, and above mentioned quantum computing algorithms are very high. Related endeavours resulted in a sub-discipline called quantum image processing (QIMP). The basic idea is that properties of an image, like the colors at certain positions, can be encoded as qubit-lattices [36], which was widely accepted and formally extended by many representations and possible applications, including videos [37].

As visualized in Fig. 2 the field is categorized [38], [39] into quantum image representations (QIRs), transformations, applications, and algorithms, some of which are also required for robotic perception. For instance, registration algorithms transform images, recorded by different sensors, in different times, with different depth, or from different view points, into one common coordinate system. Although several quantum-inspired registration methods [40], [41] have been proposed, registration is still rarely touched in QIMP. Another example falls into the category of segmentation algorithms, namely for visual tracking of moving objects. [42] claims to achieve a quadratic speedup, compared to conventional approaches, for detecting the moving object's position, and even greater improvements for object disappearance detection and motion behavior matching.

However, above mentioned approaches only deal with two-dimensional images, which is not sufficient when dealing with robotic perception, where the input of multiple sensors are often fused into a three-dimensional point cloud, in order to locate and identify objects and environments. Currently, only few methods exist to express a three-dimensional image by quantum representation in the form of quantum point cloud [43].

As with other quantum technologies, the general expectation is that QIMP will surpass the capabilities and performance of it's traditional equivalents by far. Some approaches already show potential benefits, but it is still required to show that QIMP is superior to conventional image processing by presenting high impact applications and by providing corresponding hardware and toolkits [39].

B. Think: Traditional Artificial Intelligence in Robotics

Traditional AI, in contrast to modern machine learning approaches, is based on formal knowledge representations (e.g., by rules and facts) and algorithms in order to optimize the robot behaviour or mimic intelligent (human) behaviour. AI-applications are frequently used in robotics, like path planning, the deduction of goal-oriented action plans, system diagnosis, the coordination of multiple agents, or reasoning and deduction of new knowledge [44], [45]. Many of these applications use variations of uninformed (blind) or informed (heuristic) search algorithms, which are based on traversing trees or graphs, where each node represents a possible state in the search space, connected to further follow-up states.

Table II lists a complexity comparison of basic search algorithms that are used in robotic and AI applications, where d is the depth of a solution within the search tree, b is the branching factor of the search tree, and n is a subset of b, which the algorithm will actually process [46]. The

TABLE II

COMPARISON OF BASIC GRAPH SEARCH ALGORITHMS, ADAPTED FROM [46].

Algorithm	Time	Memory	Complete	Optimal	Type
Breadth First [47]	$O(b^d)$	$O(b^d)$	Yes	Yes	Blind
Depth First [48], [49]	$O(b^d)$	$O(d)$	No	No	Blind
Hill Climbing [50], [51]	$O(b^d)$	$O(1) - O(b^d)$	No	No	Heuristic
Best First [52]	$O(b^d)$	$O(b^d)$	Yes	No	Heuristic
A* [53]	$O(b^d)$	$O(b^d)$	Yes	Yes	Heuristic
Beam Search [54]	$O(n^d)$	$O(n^d)$	No	No	Heuristic

references in the table lead to robotics related papers that additionally apply these search strategies mainly for multi-robot coordination and trajectory planning. As already stated in Section II, combinatorial search algorithms can be formulated as quantum algorithmic problem by applying Grover's algorithm, or quantum annealing [55], which reduces the complexity tremendously. For graph search algorithms there is also a quantum alternative based on quantum random walks [56].

Another group of AI algorithms deals with decision making under uncertainty by means of stochastic processes. This is usually accomplished by using Bayesian networks or Markov chains which are basically graphs where transitions between states are described by a stochastic properties [55]. Quantum algorithms for these applications are already formulated, like quantum Markov chain and quantum Markov processes, which replace the classical definitions of probability with quantum probability [57], [58]. Simulated annealing, genetic algorithms and some Monte Carlo methods can be interpreted as stochastic processes with a finite state space and can thus profit from a quantum formulation, which provides a quadratic speed up in most of the cases [57]. Simulated annealing as well as genetic algorithms are used in several approaches to tackle robot trajectory planning [59], which may as well profit from the quantum representation. Quantum random walks display, however, different properties compared to classical random walks [60].

C. Act: Kinematics and Dynamics

For many years now, attempts have been made to solve classical robotic tasks using artificial intelligence methods as an alternative. For this reason, it is not surprising that analogous works can be found, which want to solve kinematic problems with quantum neural networks, e.g., the inverse kinematics problem [61] or using quantum genetic algorithm, e.g., for trajectory planning [62]. Figure 3 shows a qualitative classification of basic optimization options that can be applied to the design of a manipulator and its kinematics control [63]. These freedoms can be understood as incomplete specifications of a planner, e.g. pick and place poses are defined, but not the way in between. In addition, all of this options lead to a discrete or continuous optimization problem, which grows exponentially with the degree of surplus actuators.

Optimizing the sequence of subtasks (or atomic tasks) within a given problem is a comprehensive issue that can be formulated as traveling salesperson problem, which is in the complexity class NP-complete [64] and has a chance to be solved efficiently by the use of a quantum computer [65]. Redundancy resolution of an over-determined robot system is known as an nonlinear optimization problem. Global solutions can actually be computed only for simple kinematic chains, therefore the local solution by evaluating the Jacobian matrix is typically used [66]. Whether globally optimal solutions can be found for general redundant manipulators is currently not known. Trajectory variation directly changes the pose function of the end-effector between the start and target positions to minimize or maximize a cost function. This well researched area will again become an important field of robotics science, under consideration of the handling of non-rigid (flexible) and deformable objects [67]. The complexity of calculating an optimized motion execution increases significantly when dynamic models are examined rather than just kinematics. Moreover, we expect that the two levels of control in robotics, i.e. abstract task-planning and specific movement-planning, that are currently often solved separately, due to their combined complexity, can be solved in a more integrative way through quantum computing. In particular, the potential of quantum optimization will provide interesting new opportunities for control schemes that utilize on-line optimization such as model-predictive control [68], [69], [70] or even classic dynamic programming formulations of control problems [71].

The manipulator placement problem and the optimization of the manipulator design are even more complex in terms of the search space size. In addition, classical approaches fail completely in finding the optimal solution to the overall problem, which takes all optimization options into account at the same time. Same applies if the exactness of mathematical modeling is extended to the dynamic case, where, e.g., the

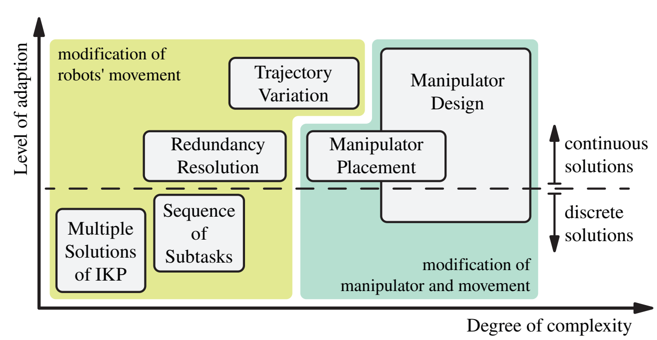

Fig. 3. Classification of optimization approaches for robot manipulators.

moments of inertia of manipulator parts and joint friction are taken into consideration. There are opinions in the robotics community that this problem class can only be handled with quantum reinforcement learning, as with models that learn to improve themselves [72]. Another potent method to get such combinatorial problems under control seems to be hybrid quantum-classical algorithms [73]. Furthermore, the use of quantum computers in a multi-state operation appears to be useful to find the global solution of complex optimization problems. For this purpose, a first quantum computer resource for the generation of start solutions and a second quantum computer resource for the finding of optimal solutions are used [74].

D. Observe: Diagnosis and Data Mining

In modern robotics, experience shows that even carefully designed and manufactured robots encounter faults due to reasons like degradation of components over time or incomplete knowledge of the environment in which a robot operates. Therefore, methods to detect and specify these faults are essential. The problem of diagnosis can generally be characterized as the problem of finding the components in a system that describe the discrepancy between the observed and the expected behaviour of the system best. The set of methods build either on systems theory [75], AI methodologies [76] or hybrid approaches [77]. The classic consistency based approach [78], [79], for example, deduces diagnoses on the basis of solving a minimal hitting-set problem, which is a common optimization problem in AI. The hitting-set problem is one of the 21 classical NP-hard problems of Karp [80]. It can be reformulated as a vertex cover problem for which quantum computing approaches already exist [81].

Data mining is the process of extracting information from and discovering patterns in large data sets. Thus, data mining is a valuable tool for knowledge discovery in order to diagnose systems [82], e.g. to identify the sources and reasons for erroneous robot behavior by mining the robot's log files. Data mining and analysis involves methods from machine learning, statistics, and database systems (like index searches), which are optimally suited to be tackled with quantum algorithmic approaches, as stated in several of the above sections [83].

IV. Open Research Topics

On the road to using quantum computing techniques effectively in robotics, there is still a variety of research topics open. Besides the application of quantum computing techniques to improve existing algorithms, quantum technologies will allow us to approach the underlying problems from totally new directions and thus also find new solutions.

In this section, we present some of the research challenges that we deem important to enable effective use of quantum computing technologies in future robotic systems.

A. High-level languages and their application to robotics

While several higher-level languages for quantum computing have been presented (e.g., [84], [85], [86], [6]), they are still far from a broad applicability since they require significant know-how in the principles of quantum computing and quantum mechanics in order to be used. A high-level language for quantum computing should be accessible to developers in order to enable broad use. To solve specific challenges of robotics, a domain-specific language (DSL) for quantum robotics is desirable just as there are many DSLs for robotics already today. The transfer of quantum computing know-how into compilers and tools could relieve the developers by performing optimization and allocation decisions automatically. Thus, there may be much work done to the language infrastructure transferring the DSL source code to quantum-computer compatible machine code. Ideally, the impact on the DSL should be small to not require extra quantum know-how from developers.

B. Architectures for the integration of quantum computing with robotic systems

It is obvious that the main computational unit of a robot will not be quantum-only. Instead, we foresee two hybrid architectures that will be predominant. First, initially probably the only architecture will be quantum computing services in the cloud that can be remotely accessed on demand. Similar approaches are currently available for GPU-based processing. However, this is currently only suitable for long-running tasks and non-realtime tasks that allow delayed responses.

Second, once the miniaturization has progressed sufficiently, quantum computing co-processors (QPUs) will be integrated on robots that only take over tasks for which they are better suited. Of course, those local QPUs will not have the same power (in terms of available qubits) as their cloud-based counterparts. Thus, smart algorithms for splitting the workload will be a crucial success factor for using quantum computing.

For both cases and also their combination (i.e., local QPU and a cloud service with high power), intelligent scheduling between CPU and QPU must be ensured. Similar techniques as for CPU-GPU scheduling either supported by a DSL [87] (possibly integrated in the DSL mentioned above) or performed by means of program profiling [88] can be starting points. However, special characteristics have to be observed like the mobility of the robots or the high latency in accessing cloud-services. In addition, GPU-scheduling typically assumes fixed program flows for optimization (thus, also profiling works). However, since robots' operations are highly interdependent with their operating environments, no such fixed program flow exists or at least the problem sizes for algorithms scale dynamically according to the environment. Thus, methods for context-sensitive on-demand scheduling are needed. This requires that first, the executed robot program can be executed on CPU as well as QPU and second, that there is a decision mechanism that operates on partial information and is real-time capable. Those factors make the CPU-QPU scheduling a distinctly harder problem. Potentially however, also the scheduling algorithm itself could be quantum-powered.

C. Distributed quantum computing

Multi-robot systems are inherently distributed systems. Thus quantum-enabled robot fleets must have methods for distributed computing that respects and harnesses the power of quantum computing. Distributed algorithms like consensus methods or complex task planning must profit from the elevated capabilities available with the new technology. Thus, also methods for state synchronization on different hosts must be found. Initial works in distributed quantum computing show that the parallel use of smaller quantum computers can be used to scale the computation [89]. The power of entanglement is used to remotely manipulate the quantum state in remote computers [90].

D. Development, simulation and test environments

Appropriate tool support is a key factor for developer productivity and software quality. The special properties of quantum computing development must be incorporated into robotic development environments (RDEs) which includes the two different paradigms of programming classical computers and programming quantum algorithms. Again here, a unifying DSL would be advantageous.

Simulation is a primary tool in robotics research and in the development of robotic applications. Quantum-aware simulators will have to reflect the impacted fields described in Section III. With the advent of quantum-powered robots, simulations must also reflect the CPU-QPU task scheduling appropriately. There might even be techniques to use the simulation already for the required program profiling.

Even more challenging will be the creation of testing tools that reflect the combination of CPU and QPU. With the advent of quantum programming languages, also verification and validation will quickly become important. This will be a new research field in the future of quantum computing. Debugging and testing quantum computing-algorithms in conjunction with robotic hardware will correspondingly also pose interesting challenges to researchers.

E. Robot safety

In Section III-C the complexity of the optimization possibilities under inclusion of a kinematic or dynamic robot model has been discussed. In particular, the trajectory planning and the redundancy resolution are tasks which have to be calculated online if the environment is a priori unknown [91]. This is crucial in real-world collaborative robotic applications where robots and humans work side-by-side and unplanned situations are very likely to occur. The physical task-execution of the robot, which depends on the robot's autonomous control schemes, has a decisive influence on the contact-forces that the robot will exert on its environment in the case of unintended collisions. The technical standard [92] provides the regulatory framework for the permissible forces and pressures that may act on humans in the event of a contact. This means that the motion execution is influenced by another criterion, which depends not only on the robot system itself, but also on its dynamic environment. The open problem can thus be identified as finding adaptive and optimized robot trajectories in a real world environment while maintain safety and completeness guarantee.

V. Conclusion

This position paper discussed current work and the expected influence of quantum computing on the field of robotics. We are convinced that this key future technology will significantly impact the field of robotic science and the application of robots. Therefore, we have described potential applications of quantum computing in key methodologies and components of robots and discussed likely challenges related to the use of quantum computing in robotics. The following years will show how the problem descriptions of robot's key functionalities and their solutions will change under the influence of this new technology. In our future work, we will detail quantum algorithms for the application areas described above and work on resulting and complementary open research questions. Our goal is to make quantum computers an advantageous technology for robotics. We will focus on the transformation of robot-related problems that are in the class NP and those that are not known to be in P or to be NP-complete into the world of quantum computation.

References

[1] R. P. Feynman, "Simulating physics with computers," *International Journal of Theoretical Physics*, vol. 21, no. 6, pp. 467–488, Jun 1982. [Online]. Available: https://doi.org/10.1007/BF02650179

[2] ——, "Quantum mechanical computers," *Foundations of Physics*, vol. 16, no. 6, pp. 507–531, Jun 1986. [Online]. Available: https://doi.org/10.1007/BF01886518

[3] M. Möller and C. Vuik, "On the impact of quantum computing technology on future developments in high-performance scientific computing," *Ethics and Information Technology*, vol. 19, no. 4, pp. 253–269, 2017. [Online]. Available: http://dx.doi.org/10.1007/s10676-017-9438-0

[4] P. Benioff, "Quantum robots and environments," *Physical Review A*, vol. 58, no. 2, p. 893, 1998.

[5] D. Dong, C. Chen, C. Zhang, and Z. Chen, "Quantum robot: structure, algorithms and applications," *Robotica*, vol. 24, no. 4, pp. 513–521, 2006.

[6] D. S. Steiger, T. Häner, and M. Troyer, "ProjectQ: an open source software framework for quantum computing," *Quantum*, vol. 2, p. 49, Jan. 2018. [Online]. Available: https://doi.org/10.22331/q-2018-01-31-49

[7] D. A. Sofge, "A survey of quantum programming languages: History, methods, and tools," in *Quantum, Nano and Micro Technologies, 2008 Second International Conference on*. IEEE, 2008, pp. 66–71.

[8] V. Mavroeidis, K. Vishi, M. D. Zych, and A. Jøsang, "The impact of quantum computing on present cryptography," *arXiv preprint arXiv:1804.00200*, 2018.

[9] C. Cheng, R. Lu, A. Petzoldt, and T. Takagi, "Securing the internet of things in a quantum world," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 116–120, 2017.

[10] H. Primas, *Chemistry, quantum mechanics and reductionism: perspectives in theoretical chemistry*. Springer Science & Business Media, 2013, vol. 24.

[11] R. Improta, F. Santoro, and L. Blancafort, "Quantum mechanical studies on the photophysics and the photochemistry of nucleic acids and nucleobases," *Chemical reviews*, vol. 116, no. 6, pp. 3540–3593, 2016.

[12] E. Lang and A. Mulholland, "Molecular dynamics, quantum mechanics, and combined quantum mechanics/molecular mechanics methods for drug discovery and development," in *Comprehensive Medicinal Chemistry III*, S. Chackalamannil, D. Rotella, and S. E. Ward, Eds. Oxford: Elsevier, 2017, pp. 51–66. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780124095472123443

[13] P. Rebentrost, B. Gupt, and T. R. Bromley, “Quantum computational finance: Monte carlo pricing of financial derivatives,” *arXiv preprint arXiv:1805.00109*, 2018.

[14] Y. Subaşi, R. D. Somma, and D. Orsucci, “Quantum algorithms for systems of linear equations inspired by adiabatic quantum computing,” *Phys. Rev. Lett.*, vol. 122, p. 060504, Feb 2019. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.122.060504

[15] T. Xin, S. Wei, J. Cui, J. Xiao, I. Arrazola, L. Lamata, X. Kong, D. Lu, E. Solano, and G. Long, “A quantum algorithm for solving linear differential equations: Theory and experiment,” *arXiv preprint arXiv:1807.04553*, 2018.

[16] D. Vaishnav, “Quantum computing and quantum algorithms,” *IRE Journals*, vol. 1, no. 9, 2018.

[17] T. F. Rønnow, Z. Wang, J. Job, S. Boixo, S. V. Isakov, D. Wecker, J. M. Martinis, D. A. Lidar, and M. Troyer, “Defining and detecting quantum speedup,” *Science*, vol. 345, no. 6195, pp. 420–424, 2014.

[18] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, “Quantum machine learning,” *Nature*, vol. 549, no. 7671, pp. 195–202, 2017.

[19] L. K. Grover, “A fast quantum mechanical algorithm for database search,” in *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. ACM, 1996, pp. 212–219.

[20] A. W. Harrow, A. Hassidim, and S. Lloyd, “Quantum algorithm for linear systems of equations,” *Physical review letters*, vol. 103, no. 15, p. 150502, 2009.

[21] G. H. Low, T. J. Yoder, and I. L. Chuang, “Quantum inference on bayesian networks,” *Physical Review A*, vol. 89, no. 6, p. 062315, 2014.

[22] N. Wiebe and C. Grandade, “Can small quantum systems learn,” *Quantum Info. Comput.*, vol. 17, no. 7-8, pp. 568–594, June 2017. [Online]. Available: http://dl.acm.org/citation.cfm?id=3179553.3179555

[23] A. Kapoor, N. Wiebe, and K. Svore, “Quantum perceptron models,” in *Advances in Neural Information Processing Systems*, 2016, pp. 3999–4007.

[24] N. Wiebe, D. Braun, and S. Lloyd, “Quantum algorithm for data fitting,” *Physical review letters*, vol. 109, no. 5, p. 050505, 2012.

[25] S. Lloyd, M. Mohseni, and P. Rebentrost, “Quantum principal component analysis,” *Nature Physics*, vol. 10, no. 9, pp. 631–633, 2014.

[26] P. Rebentrost, M. Mohseni, and S. Lloyd, “Quantum support vector machine for big data classification,” *Physical review letters*, vol. 113, no. 13, p. 130503, 2014.

[27] V. Dunjko, J. M. Taylor, and H. J. Briegel, “Quantum-enhanced machine learning,” *Physical review letters*, vol. 117, no. 13, p. 130501, 2016.

[28] M. Herrero-Collantes and J. C. Garcia-Escartin, “Quantum random number generators,” *Reviews of Modern Physics*, vol. 89, no. 1, p. 015004, 2017.

[29] F. Magniez, M. Santha, and M. Szegedy, “Quantum algorithms for the triangle problem,” *SIAM Journal on Computing*, vol. 37, no. 2, pp. 413–424, 2007.

[30] P. W. Shor, “Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer,” *SIAM review*, vol. 41, no. 2, pp. 303–332, 1999.

[31] W. J. Gutjahr and A. Pichler, “Stochastic multi-objective optimization: a survey on non-scalarizing methods,” *Annals of Operations Research*, vol. 236, no. 2, pp. 475–499, 2016.

[32] S. Omkar, R. Khandelwal, T. Ananth, G. N. Naik, and S. Gopalakrishnan, “Quantum behaved particle swarm optimization (qpso) for multi-objective design optimization of composite structures,” *Expert Systems with Applications*, vol. 36, no. 8, pp. 11 312–11 322, 2009.

[33] P. Tandon, S. Lam, B. Shih, T. Mehta, A. Mitev, and Z. Ong, “Quantum robotics: A primer on current science and future perspectives,” *Synthesis Lectures on Quantum Computing*, vol. 6, no. 1, pp. 1–149, 2017.

[34] E. Aïmeur, G. Brassard, and S. Gambs, “Quantum speed-up for unsupervised learning,” *Machine Learning*, vol. 90, no. 2, pp. 261–287, 2013.

[35] D. Dong, C. Chen, H. Li, and T.-J. Tarn, “Quantum reinforcement learning,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 38, no. 5, pp. 1207–1220, 2008.

[36] *Storing, processing, and retrieving an image using quantum mechanics*, vol. 5105, 2003. [Online]. Available: https://doi.org/10.1117/12.485960

[37] F. D. Abdullah M. Iliyasu, Phuc Q. Le and K. Hirota, “A framework for representing and producing movies on quantum computers,” *International Journal of Quantum Information*, vol. 09, no. 06, pp. 1459 – 1497, 2011.

[38] F. Yan, A. M. Iliyasu, and S. E. Venegas-Andraca, “A survey of quantum image representations,” *Quantum Information Processing*, vol. 15, no. 1, pp. 1–35, Jan 2016. [Online]. Available: https://doi.org/10.1007/s11128-015-1195-6

[39] N. Abura’ed, F. S. Khan, and H. Bhaskar, “Advances in the quantum theoretical approach to image processing applications,” *ACM Comput. Surv.*, vol. 49, no. 4, pp. 75:1–75:49, Feb. 2017. [Online]. Available: http://doi.acm.org/10.1145/3009965

[40] H. Talbi, A. Draa, and M. C. Batouche, “A genetic quantum algorithm for image registration,” in *Information and Communication Technologies: From Theory to Applications, 2004. Proceedings. 2004 International Conference on*. IEEE, 2004, pp. 395–396.

[41] H. Talbi, M. Batouche, and A. Draa, “A quantum-inspired genetic algorithm for multi-source affine image registration,” in *International Conference Image Analysis and Recognition*. Springer, 2004, pp. 147–154.

[42] C.-H. Yu, F. Gao, C. Liu, D. Huynh, M. Reynolds, and J. Wang, “Quantum algorithm for visual tracking,” *Physical Review A*, vol. 99, no. 2, p. 022301, 2019. [Online]. Available: https://doi.org/10.1103/PhysRevA.99.022301

[43] N. Jiang, H. Hu, Y. Dang, and W. Zhang, “Quantum point cloud and its compression,” *International Journal of Theoretical Physics*, vol. 56, no. 10, pp. 3147–3163, 2017.

[44] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2009.

[45] R. R. Murphy, *Introduction to AI Robotics*, 1st ed. Cambridge, MA, USA: MIT Press, 2001.

[46] A. Chandel and M. Sood, “Searching and optimization techniques in artificial intelligence: A comparative study & complexity analysis,” *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume*, vol. 3, 2014.

[47] R. Sim and N. Roy, “Global a-optimal robot exploration in slam,” in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*. IEEE, 2005, pp. 661–666.

[48] N. Sariff and N. Buniyamin, “An overview of autonomous mobile robot path planning algorithms,” in *Research and Development, 2006. SCOReD 2006. 4th Student Conference on*. IEEE, 2006, pp. 183–188.

[49] S. Giordani, M. Lujak, and F. Martinelli, “A distributed algorithm for the multi-robot task allocation problem,” in *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*. Springer, 2010, pp. 721–730.

[50] S. Thrun and Y. Liu, “Multi-robot slam with sparse extended information filers,” in *Robotics Research. The Eleventh International Symposium*. Springer, 2005, pp. 254–266.

[51] B. P. Gerkey, S. Thrun, and G. Gordon, “Parallel stochastic hill-climbing with small teams,” in *Multi-Robot Systems. From Swarms to Intelligent Automata Volume III*. Springer, 2005, pp. 65–77.

[52] K. Shubina and J. K. Tsotsos, “Visual search for an object in a 3d environment using a mobile robot,” *Computer Vision and Image Understanding*, vol. 114, no. 5, pp. 535–547, 2010.

[53] X. Liu and D. Gong, “A comparative study of a-star algorithms for search and rescue in perfect maze,” in *Electric Information and Control Engineering (ICEICE), 2011 International Conference on*. IEEE, 2011, pp. 24–27.

[54] J. W. S. Chong, S. Ong, A. Y. Nee, and K. Youcef-Youmi, “Robot programming using augmented reality: An interactive method for planning collision-free paths,” *Robotics and Computer-Integrated Manufacturing*, vol. 25, no. 3, pp. 689–701, 2009.

[55] A. Wichert, *Principles of Quantum Artificial Intelligence*. World Scientific Publishing Co. Pte. Ltd., 2014.

[56] A. M. Childs and J. Goldstone, “Spatial search by quantum walk,” *Physical Review A*, vol. 70, no. 2, p. 022314, 2004.

[57] M. Szegedy, “Quantum speed-up of markov chain based algorithms,” in *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*. IEEE, 2004, pp. 32–41.

[58] B. Kraus, H. P. Büchler, S. Diehl, A. Kantian, A. Micheli, and P. Zoller, “Preparation of entangled states by quantum markov processes,” *Physical Review A*, vol. A78, p. 042307, 2008.

[59] P. Raja and S. Pugazhenthi, “Optimal path planning of mobile robots: A review,” *International Journal of Physical Sciences*, vol. 7, no. 9, pp. 1314–1320, 2012.

[60] J. Kempe, “Quantum random walks: an introductory overview,” *Contemporary Physics*, vol. 44, no. 4, pp. 307–327, 2003.

[61] H. M. Abdulridha and Z. A. Hassoun, "Control design of robotic manipulator based on quantum neural network," *Journal of Dynamic Systems, Measurement, and Control*, vol. 140, no. 6, p. 061002, 2018.

[62] G. Qingda, Q. Yanming, L. Peijie, and C. Jianwu, "Trajectory planning of robot based on quantum genetic algorithm," in *International Conference on Mechatronics and Intelligent Robotics*. Springer, 2017, pp. 561–567.

[63] M. Brandstötter, "Adaptable serial manipulators in modular design," Ph.D. dissertation, UMIT, Institute of Automation and Control Engineering, 2016.

[64] D. Harel and Y. A. Feldman, *Algorithmics: the spirit of computing*. Pearson Education, 2004.

[65] A. Berthiaume and G. Brassard, "Oracle quantum computing," *Journal of modern optics*, vol. 41, no. 12, pp. 2521–2535, 1994.

[66] Z. Li, M. Brandstötter, and M. Hofbaur, "Global optimisation analysis for a planar redundant serial manipulator," *International Journal of Mechanisms and Robotic Systems*, vol. 4, no. 4, pp. 352–367, 2018.

[67] F. Ruggiero, A. Petit, D. Serra, A. C. Satici, J. Cacace, A. Donaire, F. Ficuciello, L. R. Buonocore, G. A. Fontanelli, V. Lippiello, *et al.*, "Nonprehensile manipulation of deformable objects: Achievements and perspectives from the rodyman project," *IEEE Robotics & Automation Magazine*, 2018.

[68] D. Q. Mayne and H. Michalska, "Receding horizon control of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 35, no. 7, pp. 814–824, July 1990.

[69] M. Morari and J. H. Lee, "Model predictive control: past, present and future," *Computers and Chemical Engineering*, vol. 23, no. 4, pp. 667 – 682, 1999.

[70] F. Allgoewer and A. Zheng, Eds., *Nonlinear Model Predictive Control*, ser. Progress in Systems and Control Theory. Birkhaeuser Basel, 2000.

[71] D. Bertsekas, *Dynamic Programming*, 4th ed. Athena Scientific, 2017.

[72] V. Dunjko, J. M. Taylor, and H. J. Briegel, "Advances in quantum reinforcement learning," in *Systems, Man, and Cybernetics (SMC), 2017 IEEE International Conference on*. IEEE, 2017, pp. 282–287.

[73] G. G. Guerreschi and M. Smelyanskiy, "Practical optimization for hybrid quantum-classical algorithms," *arXiv preprint arXiv:1701.01450*, 2017.

[74] D. Garrison, A. E. Fano, and J. A. Weichenberger, "Multi-state quantum optimization engine," US Patent US10 095 981B1, Oct, 2018.

[75] J. Chen and R. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Kluwer, 1999.

[76] W. Hamscher, L. Console, and J. de Kleer, Eds., *Readings in Model-Based Diagnosis*. Morgan Kaufmann, 1992.

[77] M. W. Hofbaur and B. C. Williams, "Hybrid estimation of complex systems," *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 34, no. 5, pp. 2178–2191, October 2004.

[78] R. Reiter, "A theory of diagnosis from first principles," *Artificial intelligence*, vol. 32, no. 1, pp. 57–95, 1987.

[79] R. Greiner, B. A. Smith, and R. W. Wilkerson, "A correction to the algorithm in reiter's theory of diagnosis," *Artificial Intelligence*, vol. 41, no. 1, pp. 79–88, 1989.

[80] R. M. Karp, *Reducibility among Combinatorial Problems*. Boston, MA: Springer US, 1972, pp. 85–103. [Online]. Available: https://doi.org/10.1007/978-1-4684-2001-2_9

[81] W. Chang, T. Ren, M. Feng, S. Huang, L. C. Lu, K. W. Lin, and M. Guo, "Quantum algorithms of the vertex cover problem on a quantum computer," in *2009 WASE International Conference on Information Engineering*, vol. 2, July 2009, pp. 100–103.

[82] X. Zhu, *Knowledge Discovery and Data Mining: Challenges and Realities: Challenges and Realities*. Igi Global, 2007.

[83] P. Wittek, *Quantum Machine Learning: What Quantum Computing Means to Data Mining*. Elsevier Science, 2016.

[84] P. Selinger, "Towards a quantum programming language," *Mathematical Structures in Computer Science*, vol. 14, no. 4, p. 527586, 2004.

[85] A. S. Green, P. L. Lumsdaine, N. J. Ross, P. Selinger, and B. Valiron, "Quipper: A scalable quantum programming language," in *Proceedings of the 34th ACM SIGPLAN Conference on Programming Language Design and Implementation*, ser. PLDI '13. New York, NY, USA: ACM, 2013, pp. 333–342. [Online]. Available: http://doi.acm.org/10.1145/2491956.2462177

[86] K. Svore, A. Geller, M. Troyer, J. Azariah, C. Granade, B. Heim, V. Kliuchnikov, M. Mykhailova, A. Paz, and M. Roetteler, "Q#: Enabling scalable quantum computing and development with a high-level dsl," in *Proceedings of the Real World Domain Specific Languages Workshop 2018*, ser. RWDSL2018. New York, NY, USA: ACM, 2018, pp. 7:1–7:10. [Online]. Available: http://doi.acm.org/10.1145/3183895.3183901

[87] T. R. W. Scogland, B. Rountree, W. Feng, and B. R. de Supinski, "Heterogeneous task scheduling for accelerated openmp," in *2012 IEEE 26th International Parallel and Distributed Processing Symposium*, May 2012, pp. 144–155.

[88] Z. Wang, L. Zheng, Q. Chen, and M. Guo, "CPU+GPU scheduling with asymptotic profiling," *Parallel Computing*, vol. 40, no. 2, pp. 107 – 115, 2014, special issue on programming models and applications for multicores and manycores. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167819113001415

[89] A. Yimsiriwattana and S. J. Lomonaco, "Distributed quantum computing: a distributed shor algorithm," *Proc. SPIE*, vol. 5436, 2004. [Online]. Available: https://doi.org/10.1117/12.546504

[90] ——, "Generalized GHZ states and distributed quantum computing," *arXiv preprint quant-ph/0402148*, 2004. [Online]. Available: https://arxiv.org/abs/quant-ph/0402148

[91] K. Hauser, "On responsiveness, safety, and completeness in real-time motion planning," *Autonomous Robots*, vol. 32, no. 1, pp. 35–48, 2012.

[92] Technical Specification, "ISO/TS 15066:2016, Robots and robotic devices – Collaborative robots," International Organization for Standardization, Tech. Rep., 2016.